

Toward the Design and Simulation of a New Generation of Reconfigurable Space Manipulators using Telescoping Passive Joints

P. Hebert¹, C. Tatossian², M. Cairns³, F. Aghili⁴, K. Parsa⁵, A. Morozov⁶

¹*Department of Aeronautics, California Institute of Technology, hebert@caltech.edu*

²*McGill Computational Fluid Dynamics Laboratory, McGill University, ctatossian@cfdlab.mcgill.ca*

³*Department of Mechanical Engineering, McGill University, malcolm.cairns@mail.mcgill.ca*

⁴*Spacecraft Engineering, Canadian Space Agency, farhad.aghili@space.gc.ca*

⁵*Spacecraft Engineering, Canadian Space Agency, kourosh.parsa@space.gc.ca*

⁶*McGill Centre for Intelligent Machines, McGill University, alexvit@cim.mcgill.ca*

Abstract

This work presents the analysis, simulation, design, and assembly of a three-degree-of-freedom reconfigurable robotic arm. Unlike conventional reconfigurable robots, the final design illustrated in this paper does not achieve re-configurability through modular joints; instead, it is equipped with passive telescopic joints. These passive joints allow the robotic arm to change its Denavit-Hartenberg parameters via an innovative breaking system. The robotic arm itself presents many advantages; not only is it versatile to perform various tasks, but it can be packed into a small volume, as usually required for launch in space applications.

Keywords: design, reconfigurable robotic arm, passive telescopic joints

Vers le design et la simulation d'une nouvelle génération de robots manipulateurs employant le télescopage de joints passifs

Résumé

Ce travail présente l'analyse, la simulation, le design et l'assemblage d'un bras télémanipulateur reconfigurable à trois degrés de liberté. Contrairement aux robots reconfigurables traditionnels, le design final présenté dans cet article n'achève pas la reconfigurabilité à travers des joints modulaires; au lieu de cela, il est équipé de joints télescopiques passifs, sans actionneurs et capteurs. Ces joints passifs permettent au bras de changer ses paramètres Denavit-Hartenberg à travers un système de freinage novateur. Le bras télémanipulateur lui-même présente de nombreux avantages; non seulement il est polyvalent pour effectuer différentes tâches, mais il peut être comprimé dans un petit volume, ce qui est normalement exigé pour le lancement à l'intérieur d'applications spatiales.

Mots clés: design, bras télémanipulateur reconfigurable, joints télescopiques passifs

1 INTRODUCTION

It is difficult to imagine space exploration without any robotic assistance. However, the selection of today's extreme environments manipulators is exceptionally limited; achieving several complex reconfigurations using a limited amount of resources isn't child's play. One possible solution to this problem is to explore reconfigurable manipulators, which can change their geometry according to the operation at hand. This paper presents the analysis, simulation, design, and assembly of a three-degree-of-freedom reconfigurable robotic arm which fulfill these requirements.

The process of reconfiguration used in this work consists of the following stages¹:

1. The robotic arm sits in free space, either idle or finishing an assigned task
2. The arm grasps a fixed ball joint; the position of the end-effector is thus constrained
3. The length and twist angle of the first link are configured, keeping the second link fixed
4. The first link's configuration is secured via the internal breaking mechanism
5. The arm repeats the same process for the second link, keeping the first link fixed
6. The second link's configuration is secured via the internal breaking mechanism
7. When the final configuration is formed, the ball joint is released
8. The reconfigured robot can now operate in free space, and reach its target value

Each passive joint is equipped with a built-in brake mechanism which is normally locked, but that can be released whenever the changing of the parameters is required. This brake mechanism consists of a sophisticated internal subsystem that ultimately expands the inner section of the cylindrical joint against its outer layer. A detailed description of the brake system is illustrated in this paper.

2 KINEMATIC SIMULATION

2.1 Inverse Kinematics of the Closed Chain

Inverse kinematics will be used to solve the kinematic relations of the reconfigurable manipulator for its generalized coordinates when the position of the end-effector is given. These generalized coordinates are typically angles for revolute joints and lengths for prismatic joints. A detailed schematic of these coordinates is shown in Figure (1). This section also demonstrates a series of transformation matrices which relates the relative rotation and translation between the links^{2,5}. The variables shown in Figure (1) are described on the next page.

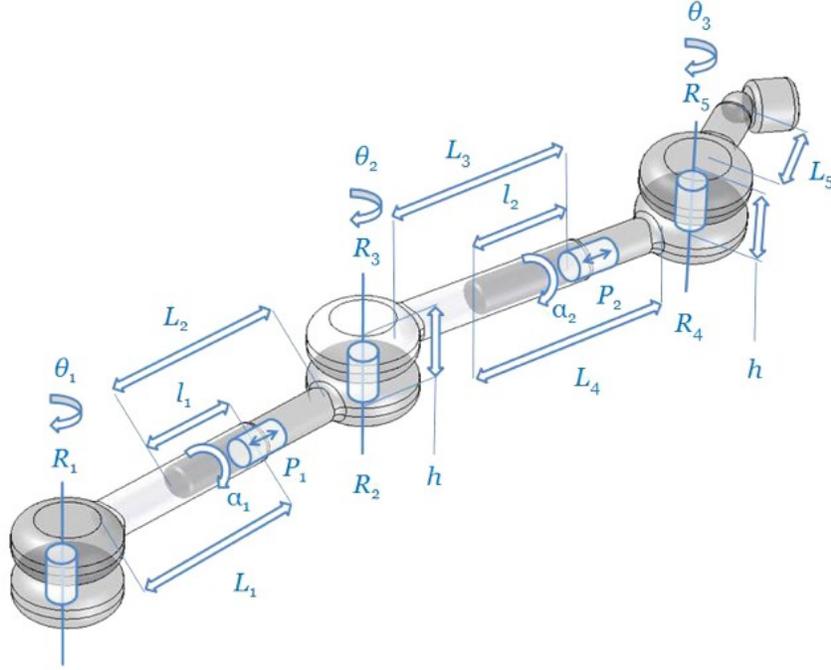


Figure 1: Kinematics of the reconfigurable robot

Referring to Figure (1),

$\theta_1, \theta_2, \theta_3$ represent the final joint angles which are defined for joints R_1, R_3, R_5 ;

α_1, α_2 represent the relative angular displacements of the links in the passive joints R_2 and R_4 ;

L_1, L_2, L_3, L_4, L_5 represent the lengths of each link;

l_1, l_2 represent the relative displacements of the links in passive joints P_1 and P_2 ;

h consists of the offset in the joints R_3 and R_5

One should note that the location of the base and the location of the end-effector is given in X,Y,Z coordinates. The analyzing of this system can then be carried out when it forms a closed loop, i.e., the end-effector grips onto an object. This will enable the change in the joint variables of the passive joints (once the passive joint is unlocked). The new configuration of the particular passive joint will be known (new α_1 , the offset h , all L 's, and l_1); and the variables of the locked passive joint remain unchanged. Thus, the only unknowns are $\theta_1, \theta_2, \theta_3$, which describe the final motor angles, R_1, R_3 , and R_5 .

There will be 10 transformation matrices for all our given links and joints, and three non-linear equations, which are to be solved for three unknowns, hence our decision to implement a computational approach, described further in subsection 2.3.

$$\begin{bmatrix} X_{origin} \\ Y_{origin} \\ Z_{origin} \\ 1 \end{bmatrix} = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \cdot {}^3_4T \cdot {}^4_5T \cdot {}^5_6T \cdot {}^6_7T \cdot {}^7_8T \cdot {}^8_9T \cdot {}^9_{10}T \begin{bmatrix} X_{endpoint} \\ Y_{endpoint} \\ Z_{endpoint} \\ 1 \end{bmatrix} = {}^0_{10}T \cdot \begin{bmatrix} X_{endpoint} \\ Y_{endpoint} \\ Z_{endpoint} \\ 1 \end{bmatrix}$$

where,

$${}^0_1T = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 & 0 & 0 \\ \sin \theta_1 & \cos \theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1_2T = \begin{bmatrix} 1 & 0 & 0 & L_1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2_3T = \begin{bmatrix} \cos \alpha_1 & -\sin \alpha_1 & 0 & 0 \\ -\sin \alpha_1 & \cos \alpha_1 & 0 & 0 \\ 0 & 0 & 1 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^3_4T = \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4_5T = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & 0 \\ \sin \theta_2 & \cos \theta_2 & -1 & -h \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5_6T = \begin{bmatrix} 1 & 0 & 0 & L_3 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^6_7T = \begin{bmatrix} \cos \alpha_2 & -\sin \alpha_2 & 0 & 0 \\ -\sin \alpha_2 & \cos \alpha_2 & 0 & 0 \\ 0 & 0 & 1 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^7_8T = \begin{bmatrix} 1 & 0 & 0 & L_4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^8_9T = \begin{bmatrix} \cos \theta_3 & -\sin \theta_3 & 0 & 0 \\ \sin \theta_3 & \cos \theta_3 & -1 & -h \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^9_{10}T = \begin{bmatrix} 1 & 0 & 0 & L_5 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.2 Trajectory Planning

In this section, we present a solution for simple trajectory planning used for both our animation and the operation of the robotic arm. The animation was used primarily for illustration purposes; it can be viewed on the internet⁶. For the animation, we used a third-order trajectory plan for the motor angles³. This plan (Figure 2a) is a third-order polynomial which specifies the trajectory of the motor angle in time as it departs from the initial angle and reaches the final destination angle. These constraints are

$$\begin{aligned}\theta(0) &= \theta_i \\ \theta(t_f) &= \theta_f\end{aligned}$$

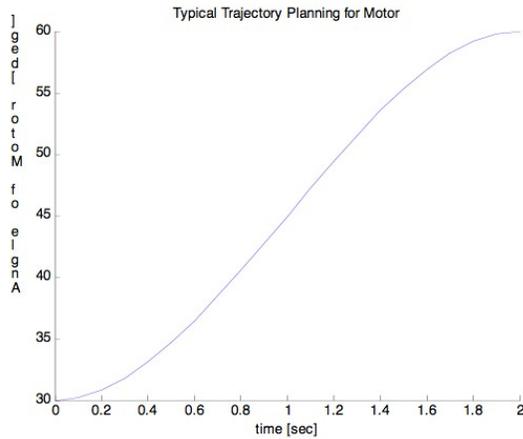
With this order, we can specify that the angular velocity of the motor will be zero at the beginning and at the end of the trajectory.

$$\begin{aligned}\dot{\theta}(0) &= 0, \text{ for each motor} \\ \dot{\theta}(t_f) &= 0,\end{aligned}$$

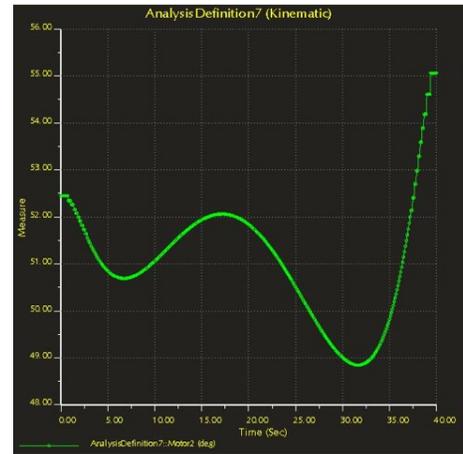
The general equation of the trajectory plan of each motor angle is given by

$$\begin{aligned}\theta(t) &= a_0 + a_1t + a_2t^2 + a_3t^3 \\ a_0 &= \theta_i, \\ a_1 &= 0, \\ a_2 &= \frac{3}{t_f^2}(\theta_f - \theta_i), \\ a_3 &= -\frac{2}{t_f^3}(\theta_f - \theta_i)\end{aligned}$$

One polynomial is generated for each of the three motor angles. However, it was observed that if we were to simultaneously plan all three motors with these simple plans, when the arm forms a closed loop, the motion would often fail as the joints could not be easily operated simultaneously. We then noticed that we could plan the trajectory of the passive joints as we did with the motor angles. Reconfiguring would give two polynomials, one for the angular displacement variable α_1 and the other for the displacement variable l_1 . Once we implemented this in Pro/ENGINEER, we could capture the complex trajectory plan for all three motors in a closed-loop configuration. In Figure 2b, an example of trajectory plan for one of the motors is shown. For actual operation of our robotic arm, we would need to send this input to the respective motor during reconfiguration in order to obtain the desired passive-joint values.



(a) Theoretical



(b) Simulation

Figure 2: Trajectory planning for the motor (deg./sec)

2.3 Robotic Analysis Software - MOBILE

The inverse kinematics problem, in our case, is a non-linear type. Instead of setting up our own forward kinematic matrices (rotational, translational), we decided to use a software package for multi-body dynamics analysis called MOBILE^{4,5}.

The software has specifically been designed for analyzing robotic systems. Thus, it was an appropriate choice of software for analyzing our system. It has built in joints such as prismatic and revolute, and also the capability for modeling curved links. Moreover, MOBILE automatically attaches the appropriate frames to each joint so as to use in the calculation of the transformation matrices for forward and inverse kinematics. The joints and the links of the manipulator implemented by MOBILE are shown in Figure 4. MOBILE also has the ability to implement the appropriate Denavit-Hartenberg offsets needed for escaping planar singularities, thus ensuring that the robotic arm is able to reach any spatial configuration. Another important characteristic of MOBILE is its internal non-linear solver. Once the manipulator is defined with the appropriate boundary conditions and constraints, it is very simple to determine the joint variables (motor angles) needed. This saved us plenty of time in setting up different algorithms for solving the inverse kinematics.

In our simulation, we implemented a ball-joint constraint, which is fixed in space, called Mochord3DPosition; it is defined as the difference between position vectors. This enables us to determine three unknowns pertaining to the system. They can be determined because of the three spatial constraints X, Y, and Z are fixed. This allows us to determine the needed motor angles.

For the execution of the code, it is needed to input the ball-joint location, as well as the initial locked passive joint parameters that are not going to be reconfigured during the initial stage. The desired parameters such as length and angle of revolution of the passive joint that will be reconfigured during the first stage need to be supplied as well.

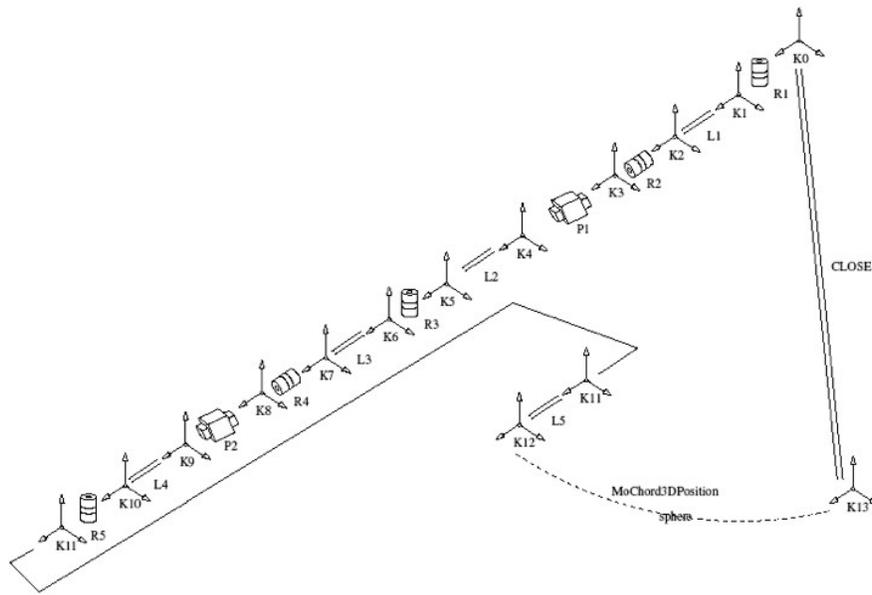


Figure 3: MOBILE chain

This will be iterated again for the second stage; in that, the second passive joint will be reconfigured to another set of values. Once the code has calculated the inverse kinematics, it will then find two sets of motor angles to achieve the reconfigured passive joint parameters. With each set of motor angles, trajectory planning will take place to properly animate the motion of the manipulator when there is a reconfiguration change from initial motor angles to the output motor angles found by MOBILE. Note that for this simulation code, for successful passive joint parameters, the length variable should be negative, as the manipulator is defined in its maximum extended length.

2.4 Results

This section presents a two-step reconfiguration example. It is worth reminding ourselves that the MOBILE program will compute the inverse kinematics of the first reconfiguration for the desired joint parameters, and the final position of the first reconfiguration will be used as the initial conditions for the second reconfiguration. Please note that, for the followings results, θ represents the motor angle, while α represents the revolute passive joint angle, and finally *Length* represents the prismatic passive joint length.

For the first reconfiguration, the end point will be set at $[12.0, 18.0, 0.98]$. The locked passive joint parameters for prismatic joint length and revolute angle is set to $[0]$ in length and $[60^\circ]$, respectively. The desired reconfigured passive joint parameters is $[0]$ in length and $[0^\circ]$, and the time for the trajectory planning motion (from one motor angle to final motor angle) will be $10s$.

OUTPUT FROM MOBILE

First Stage of Inverse Kinematics with the 2nd passive-joint locked
Desired final motor angles: Theta1 0.0161207; Theta2 90.9397; Theta3 2.53694
Desired first joint passive parameters: Alpha1 0; Length1 0
Locked Second joint passive parameters: Alpha2 60; Length2 0
Position of Mid Joint = [12;0.00337631;0]
Position of tip = [12;18;0.98]

The coefficients of the third-order polynomials are then computed, using,

$$\begin{aligned}a_0 &= \theta_i, \\a_1 &= 0, \\a_2 &= \frac{3}{t_f^2} (\theta_f - \theta_i), \\a_3 &= -\frac{2}{t_f^3} (\theta_f - \theta_i),\end{aligned}$$

and imported back into the polynomial as,

$$\theta(t) = \theta_i + \frac{3}{t_f^2} (\theta_f - \theta_i) t^2 - \frac{2}{t_f^3} (\theta_f - \theta_i) t^3.$$

OUTPUT FROM MOBILE

Theta 1 Polynomial Coefficients, 30, 0, -0.899516, 0.0599678
Theta 2 Polynomial Coefficients, 30, 0, 1.82819, -0.121879
Theta 3 Polynomial Coefficients, 30, 0, -0.823892, 0.0549261

The process is then repeated for the reconfiguration of the second passive joint parameters. It is important to note that the first reconfigured passive joint will now remain in a locked position. For the second reconfiguration, the end point will be set at [12.0, 18.0, 0.98]. The locked passive joint parameters for prismatic joint length and revolute angle is set to [0] in length and [0°], respectively. The desired reconfigured passive joint parameters is [1.5] in length and [45°], and the time for the trajectory planning motion (from one motor angle to final motor angle) will be 10s.

OUTPUT FROM MOBILE

Second Stage of Inverse Kinematics with 1st passive joint locked
Desired final motor angles: Theta1 7.18736; Theta2 83.2739; Theta3 1.70788
Desired Second joint passive parameters: Alpha2 45; Length2 -1.5
Locked First joint passive parameters: Alpha1 0; Length1 0
Position of Mid Joint = [11.9057;1.50137;0]
Position of tip = [12;18;0.98]

The corresponding polynomial coefficients are,

OUTPUT FROM MOBILE

Theta 1 Polynomial Coefficients, 0.0161207, 0, 0.215137, -0.0143425
Theta 2 Polynomial Coefficients, 90.9397, 0, -0.229975, 0.0153317
Theta 3 Polynomial Coefficients, 2.53694, 0, -0.0248718, 0.00165812

3 FINAL DESIGN

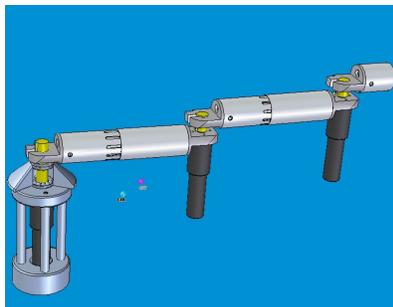
3.1 Robotic Arm

(Author's Note: Imperial units are used in order to comply with tubing industry standards)

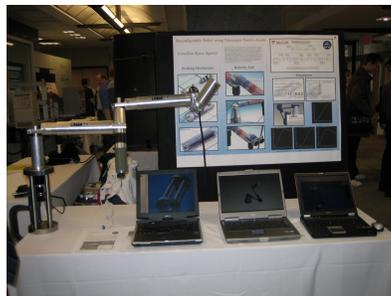
The final assembly of the robotic arm incorporates three motors (a base motor, an intermediate motor, and the gripper motor), and two telescoping passive joints (located on branches 1 and 2 of the robotic arm). The simplified solid model is shown in Figure 4a. The built prototype is shown in Figure 4b. To create a closed kinematic chain, the shortest link attaches to a fixed point (on a wall, the ground, etc) via a gripper (not shown). Once this chain is created, the robot is able to reconfigure passively.

The base of the robotic arm consists of 4 steel struts. The struts were designed to eliminate load on the base motor housing. Simultaneously, they provide sufficient ventilation, which eliminates the possibility of the motor overheating. Two thrust bearings hug the motor shaft. These thrust bearings protect the motor shaft by eliminating any radial or axial loads. Without these bearing the loading of the arm would have significantly impacted the motor's operating range and could lead to breaking of the motor shaft. A similar assembly is located on the second motor (not shown in the figure).

Each kinematic branch consists of two tubes: the outer tube (designated as male) and an inner tube (female). The male tube's inner diameter is 2 inches with a $\frac{1}{8}$ inch wall thickness. The female tube has a $\frac{1}{16}$ inch wall thickness with an inner diameter of 2.5 inches. The length of the tubes themselves would vary from 7 inches (for the second kinematic branch of the robot) to 11 inches (for the 1st branch).



(a) Simplified Model



(b) Complete Prototype



(c) Cylindrical Link

Figure 4: Robotic arm

3.2 Brake Mechanism

The breaking mechanism of the second kinematic branch is shown in Figure 5a. The final design was the eighth design that was considered. The breaking mechanism makes passive kinematic reconfiguration possible, and is what separates this design from generic robotic arm. Any change in the design of the breaking mechanism would require changes in the design of the entire robotic arm. The sizes of motors, tubes, bracket, and the gripper all hinged on the brake mechanism.

The breaking mechanism itself is located inside the male portion of the telescopic arm. When the mechanism is not engaged, the telescopic link can move freely: both translational and rotational movements between the male and female tubes are possible. When the mechanism is in this state, reconfiguration is possible. When the mechanism is engaged, the telescopic link is locked in its place. In this state, no relative motion is permitted between the male and female parts. During normal operation of the robot the breaking mechanism is to remain locked. It was hence important to design a breaking mechanism which would not require electrical power when engaged.

To lock the two links into place, a disk radially fitted with bearings is pushed forward by a linear actuator (Figure 5a). The wall of the male tube is tapered with a 3.6 degree angle, in such a way that tubing's wall thickness is increased from $\frac{1}{16}$ inch at the end to $\frac{1}{8}$ inch. This tapered section of the male tubing is then milled into 8 separate slits (Figure 5b/c).

When the disk is pushed forwards, the tapered slits are forced outwards in the radial direction. As the male tube is inside the female tube, the expanding slits catch the inner diameter of the female tube. The normal force (relative to the female tube) is strong enough to lock the joint. It is important to appreciate the mechanical advantage of this mechanism: the tapered slits transmit and enhance the axial force by a factor of approximately 4 due to the geometry of the system. The exploded view of the brake system is shown in Figure 5b.

The finite element analysis program, ALGOR, was used to compute the deflections and the stress on the flanges. It was found that for the displacement which these slits undergo, the deflection would not result in permanent deformation of the male tubing. Hence the male tubing will return to it's original state.

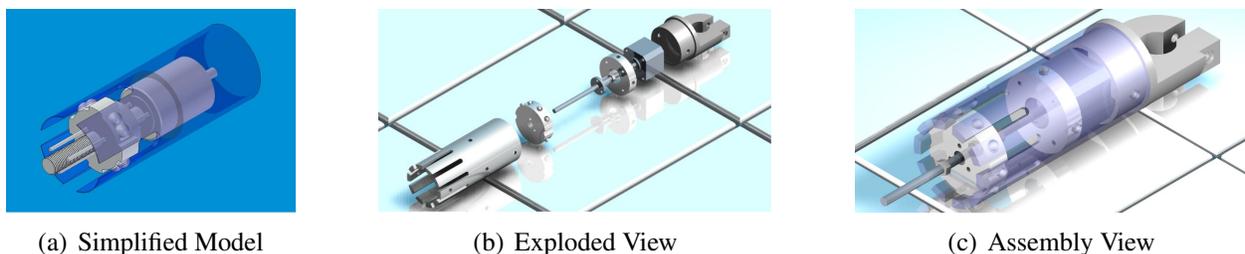


Figure 5: Solid model of the brake system

4 CONCLUSION

The simulations carried out for this particular robotic arm proved that telescopic passive joints are a plausible solution for space exploration. A short description of the theory and methods were presented in this paper, along with a possible design and fabrication of the arm. The major design aspects and flaws of the robotic arm were also presented. Rigorous testing of the arm has yet to be completed.

5 ACKNOWLEDGEMENTS

The authors would like to thank *LINAK, Inc*, for their generous donation of two linear actuators; and *Martin Tändl* for his outstanding support in this project.

6 REFERENCES

1. Parsa K., Aghili F. (2006) On the Design and Control of a New Generation of Reconfigurable Space Manipulators with Passive Joints, ICES06
2. Craig, John J. (2005) Introduction to Robotics: Mechanics and Control, Addison-Wesley, Massachusetts.
3. Kecskemethy, Andreas, and Tändl Martin (2005)
<http://www-mechanik.uni-duisburg.de/downloads/MOBILEWorkshop/MobileManual.pdf>
4. Tändl, Martin (2005) <http://www-mechanik.uni-duisburg.de/downloads/MOBILEWorkshop/MOBILEWorkshop.pdf>
5. Morecki, Adam (1999) Basics of Robotics: Theory and Components of Manipulators and Robots, Springer.
6. Hebert, P, Tatossian C. Cairns, M. (2006) Animation of Reconfigurable Robot, Retrieved from Web, 2007, <http://video.google.ca/videoplay?docid=7899765817085674745&hl=en-CA>