# Dynamic Inversion based Control of a Docking Mechanism

Nilesh V. Kulkarni* and Corey Ippolito†
*QSS Group, Inc., Moffett Field, CA - 94035*

Kalmanje Krishnakumar‡
*NASA Ames Research Center, Moffett Field, CA - 94035*

**The problem of position and attitude control of the Stewart platform based docking mechanism is considered motivated by its future application in space missions requiring the autonomous docking capability. The control design is initiated based on the framework of the intelligent flight control architecture being developed at NASA Ames Research Center. In this paper, the baseline position and attitude control system is designed using dynamic inversion with proportional-integral augmentation. The inverse dynamics uses a Newton-Euler formulation that includes the platform dynamics, the dynamics of the individual legs along with viscous friction in the joints. Simulation results are presented using forward dynamics simulated by a commercial physics engine that builds the system as individual elements with appropriate joints and uses constrained numerical integration.**

## Nomenclature

| | | |
|---|---|---|
| $\mathbf{A}$ | = | Angular acceleration of the leg in inertial coordinates |
| $\mathbf{a}$ | = | Platform linear acceleration |
| $\mathbf{a}_d$ | = | Linear acceleration of the lower portion of the leg in inertial coordinates |
| $\mathbf{a}_u$ | = | Linear acceleration of the upper portion of the leg in inertial coordinates |
| $\boldsymbol{\alpha}$ | = | Platform angular acceleration in inertial coordinates |
| $C_u$ | = | Coefficient of viscous friction at the universal joint |
| $C_p$ | = | Coefficient of viscous friction at the prismatic joint |
| $\mathbf{F}_{\text{ext}}$ | = | External force acting on the platform in the platform body axes. |
| $\mathbf{f}_i$ | = | Viscous friction at the joint between the platform and the $i^{\text{th}}$ leg |
| $\mathbf{g}$ | = | Gravitational acceleration |
| $\mathbf{I}$ | = | Platform moment of inertia in the inertial coordinates |
| $\mathbf{I}_d$ | = | Moment of inertia of the lower portion of the leg in inertial coordinates |
| $\mathbf{I}_u$ | = | Moment of inertia of the upper portion of the leg in inertial coordinates |
| $L_i$ | = | Length of the $i^{\text{th}}$ leg |
| $\dot{L}$ | = | Rate of change of the length of the leg. |
| $M$ | = | Platform mass |
| $\mathbf{M}_{\text{ext}}$ | = | External moment acting on the platform in inertial coordinates |
| $m_d$ | = | Mass of the lower portion of the leg. |
| $m_u$ | = | Mass of the upper portion of the leg. |
| $\mathbf{q}_1 - \mathbf{q}_6$ | = | Position vector of each of the Platform-leg joint in the Inertial coordinates |
| $\mathbf{R}$ | = | Position vector of the platform CG. |
| $R_{\text{platform}}$ | = | Platform Rotation Matrix |
| $\mathbf{r}_d$ | = | Position of the CG of the lower portion of the leg in inertial coordinates |
| $\mathbf{r}_u$ | = | Position of the CG of the upper portion of the leg in inertial coordinates |

* Scientist, M/S 269-2, AIAA Member, nilesh@email.arc.nasa.gov
† Engineer, M/S 269-1, AIAA Member.
‡ Group Lead, Adaptive Control and Evolvable Systems Group, M/S 269-2, AIAA Associate Fellow.

$s_1 - s_6$ = Unit vector along each of the leg expressed in the Inertial coordinate
$W$ = Angular velocity of the leg in inertial coordinates
$\omega$ = Platform angular velocity in inertial coordinates
$x_1 - x_6$ = Unknown joint forces at each of the six joints along the direction of the legs

## I. Introduction

Autonomous rendezvous and docking has been identified as a strategic technical challenge for all future space missions[1]. An androgynous docking mechanism is being developed by the Johnson Space Center towards meeting that goal, which can be used across different space platforms. Figure 1 illustrates such a docking mechanism. The docking mechanism will be attached to the chaser spacecraft, which on close proximity will extend the mechanism to dock itself to the target spacecraft. Using force sensors, the two spacecrafts can then lock in to complete the docking maneuver. Precise position and attitude control of the docking mechanism is, therefore, a critical need to achieve such an autonomous docking capability.

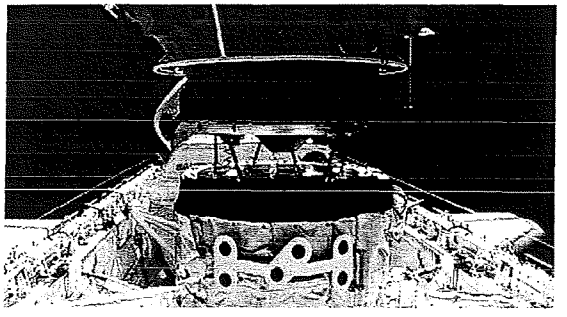

**Figure 1: Illustration of the space docking mechanism**

We propose to design the control system for the docking mechanism based on the intelligent flight control architecture being developed at the NASA Ames Research Center (ARC). The intelligent flight control (IFC) research program began in ARC in 1992 to address adaptive aircraft flight control to provide a real-time system capable of compensating for a broad spectrum of model uncertainties and failures[2-4]. The core of the IFC architecture is a neural control architecture developed by Rysdyk and Calise[5]. The control system consists of a dynamic inversion based controller with proportional-integral (PI) and neural network augmentation. NASA ARC has implemented and demonstrated this IFC architecture via several different vehicle simulations varying from small unmanned aircraft to large transport aircraft and highly maneuverable fighter aircraft.

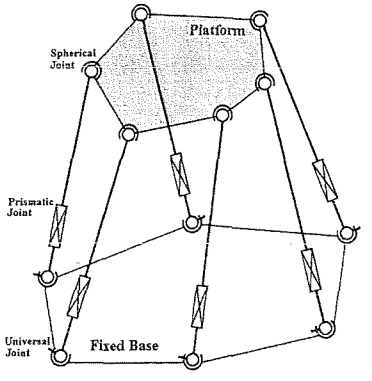


**Figure 2: 6 UPS Stewart Platform**

In this paper, the baseline control system without neural network augmentation is designed for the docking mechanism. The docking mechanism is modeled as a Stewart platform with six legs connecting the fixed base with the moving platform. Figure 2 illustrates the Stewart platform. The Stewart platform, as its name suggests, was first designed by Stewart for motion flight simulators[6-7]. The mechanical device, given its better stiffness and precise pointing capabilities, was suggested as a parallel manipulator in the robotic community and soon enough gained wide popularity[8-9]. The control design for the commercially available motion flight simulators as well as the robotic manipulators is, however, typically given by kinematical control laws using hand tuned PID controllers that try to achieve the necessary leg velocities to produce the desired platform motion. The control design outlined in this paper consists of a control law that considers the coupled dynamics of the platform and the individual legs. The control problem corresponds to computing the forces to be input at the leg prismatic joint to produce the necessary platform states. Adequate sensing is assumed that provides the target position and orientation, along with platform position, orientation and its linear and angular velocities. Similarly the lengths and the rate of change of the lengths of each of the six legs are assumed to be sensed.

## II. Docking Mechanism Control Architecture

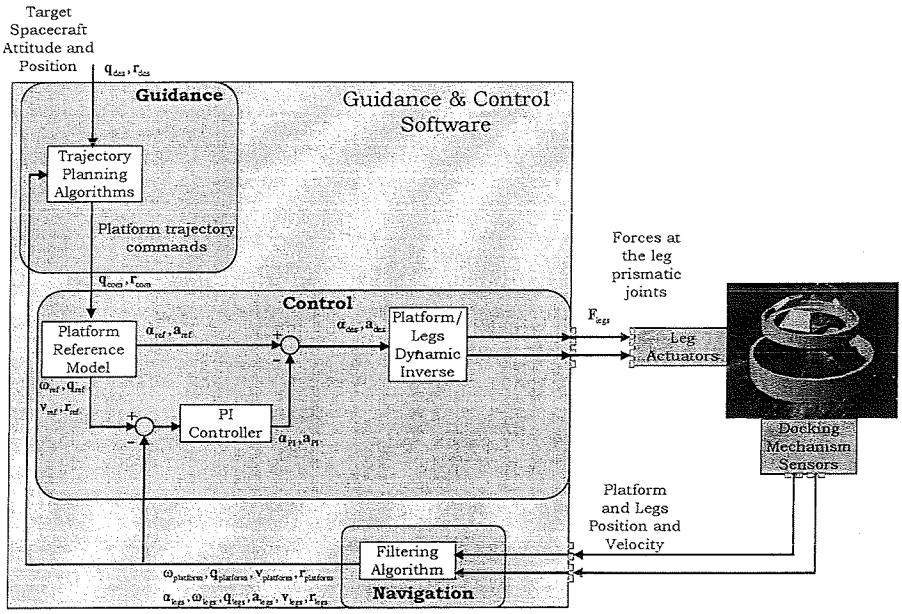


**Figure 3: Proposed docking mechanism control architecture**

Figure 3 outlines the proposed docking mechanism control architecture. The guidance law is typically designed as an optimal outer-loop trajectory or using waypoints that satisfy the relevant system and mission constraints. For the purpose of the current implementation, it is assumed that the guidance law provides the target position and orientation for the platform. These values are provided to the platform reference model that computes the reference trajectory for the platform. The reference model is divided into two parts that individually address the translational and rotational motions separately. The reference translational motion of the center of gravity (CG) of the platform is given using a second order reference model on the commanded position.

$$x_{\text{ref}}(s) = \frac{1}{s^2 + 2\xi w_n s + w_n^2} x_c(s) \tag{1}$$

$$y_{\text{ref}}(s) = \frac{1}{s^2 + 2\xi w_n s + w_n^2} y_c(s) \tag{2}$$

$$z_{\text{ref}}(s) = \frac{1}{s^2 + 2\xi w_n s + w_n^2} z_c(s) \tag{3}$$

This translational reference model provides the reference position, velocity and acceleration of the platform CG. The desired acceleration of the platform CG is given using a proportional-integral augmentation as

$$\mathbf{a}_{\text{des}} = \mathbf{a}_{\text{ref}} + \mathbf{K}_{\text{P}}(\mathbf{v}_{\text{ref}} - \mathbf{v}) + \mathbf{K}_{\text{I}}(\mathbf{r}_{\text{ref}} - \mathbf{r}) \tag{4}$$

so that the platform CG position follows a stable second order dynamics towards the reference position.

$$\mathbf{r}_{\text{ref}} = \begin{bmatrix} x_{\text{ref}} \\ y_{\text{ref}} \\ z_{\text{ref}} \end{bmatrix} \tag{5}$$

$$\begin{aligned} \mathbf{v}_{\text{ref}} &= \dot{\mathbf{r}}_{\text{ref}} \\ \mathbf{a}_{\text{ref}} &= \dot{\mathbf{v}}_{\text{ref}} \end{aligned} \tag{6}$$

$$\mathbf{K}_{\text{P}} = \begin{bmatrix} \mathbf{K}_{\text{Px}} & 0 & 0 \\ 0 & \mathbf{K}_{\text{Py}} & 0 \\ 0 & 0 & \mathbf{K}_{\text{Pz}} \end{bmatrix} \text{ and } \mathbf{K}_{\text{I}} = \begin{bmatrix} \mathbf{K}_{\text{Ix}} & 0 & 0 \\ 0 & \mathbf{K}_{\text{Iy}} & 0 \\ 0 & 0 & \mathbf{K}_{\text{Iz}} \end{bmatrix} \tag{7}$$

The reference angular motion of the platform is given as a second order motion around the eigen axis between the initial and commanded quaternion. In relation to the initial platform orientation, the commanded quaternion, $\mathbf{q}_{\text{c}}$, is given as

$$\mathbf{q}_{\text{c}} = \begin{bmatrix} e_1 \sin\left(\frac{\phi_c}{2}\right) \\ e_2 \sin\left(\frac{\phi_c}{2}\right) \\ e_3 \sin\left(\frac{\phi_c}{2}\right) \\ \cos\left(\frac{\phi_c}{2}\right) \end{bmatrix} \tag{8}$$

Inverting Eq. (8) provides the eigen axis, $\hat{\mathbf{e}} = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}^T$ and the commanded angle, $\phi_c$. The reference angular motion is given as

$$\phi_{\text{ref}}(s) = \frac{1}{s^2 + 2\xi w_n s + w_n^2} \phi_c(s) \tag{9}$$

The quaternion for the reference orientation can now be given as

$$\mathbf{q}_{\text{ref}} = \begin{bmatrix} e_1 \sin\left(\frac{\phi_{\text{ref}}}{2}\right) \\ e_2 \sin\left(\frac{\phi_{\text{ref}}}{2}\right) \\ e_3 \sin\left(\frac{\phi_{\text{ref}}}{2}\right) \\ \cos\left(\frac{\phi_{\text{ref}}}{2}\right) \end{bmatrix} \tag{10}$$

Similarly the reference angular velocity and acceleration for the platform can be given as

$$\begin{aligned} \boldsymbol{\omega}_{\text{ref}} &= \dot{\phi}_{\text{ref}} \hat{\mathbf{e}} \\ \boldsymbol{\alpha}_{\text{ref}} &= \ddot{\phi}_{\text{ref}} \hat{\mathbf{e}} \end{aligned} \tag{11}$$

The desired angular acceleration of the platform, similar to the translational case, is given using a PI augmentation as

$$\boldsymbol{\alpha}_{\text{des}} = \boldsymbol{\alpha}_{\text{ref}} - \mathbf{K}_{\text{P}}^{\text{rot}} \left( \boldsymbol{\omega}_{\text{platform}} - \boldsymbol{\omega}_{\text{ref}} \right) + \mathbf{K}_{\text{I}}^{\text{rot}} \mathbf{q}_{\text{e}}' \tag{12}$$

If the system angular acceleration equals this desired angular acceleration, it has been shown that this PI augmentation represents a stable angular motion[10]. $\mathbf{q}_{\text{e}}'$ represents the vector portion of the quaternion error given by

$$\mathbf{q}_{\text{e}}' = \begin{bmatrix} q_{\text{ref4}} & q_{\text{ref3}} & -q_{\text{ref2}} & -q_{\text{ref1}} \\ -q_{\text{ref3}} & q_{\text{ref4}} & q_{\text{ref1}} & -q_{\text{ref2}} \\ q_{\text{ref2}} & -q_{\text{ref1}} & q_{\text{ref4}} & -q_{\text{ref3}} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{13}$$

The desired translational and angular accelerations given by Eqs. (4) and (12) respectively are input to the inverse dynamics of the system to compute the leg actuation forces.

The inverse dynamics of the platform is based on the Newton-Euler formulation given by Dasgupta and Mruthyunjaya[11]. This formulation incorporates the dynamics of the platform and the dynamics of each of the legs. Viscous friction is assumed at each of the joints. The inverse dynamics is set up such that the linear and angular accelerations of the platform are given in terms of six unknown forces, one at each of the joint platform-leg joint and along the direction of the leg. Knowing these linear and angular acceleration values, these six unknown forces can be computed. They are then used in each of the six leg dynamic equations to compute the leg actuator force that is needed to be given at the prismatic joint for supporting these joint forces.

The six unknown joint forces along the direction of the legs are given by

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \mathbf{s}_3 & \mathbf{s}_4 & \mathbf{s}_5 & \mathbf{s}_6 \\ \mathbf{q}_1 \times \mathbf{s}_1 & \mathbf{q}_2 \times \mathbf{s}_2 & \mathbf{q}_3 \times \mathbf{s}_3 & \mathbf{q}_4 \times \mathbf{s}_4 & \mathbf{q}_5 \times \mathbf{s}_5 & \mathbf{q}_6 \times \mathbf{s}_6 \end{bmatrix}^{-1} * \begin{bmatrix} R_{\text{platform}}\mathbf{F}_{\text{ext}} + M(\mathbf{g}-\mathbf{a}) - \sum_{i=1}^{6}\mathbf{K}_i \\ M\mathbf{R}\times(\mathbf{g}-\mathbf{a}) - \mathbf{I}\boldsymbol{\alpha} - \boldsymbol{\omega}\times\mathbf{I}\boldsymbol{\omega} + R_{\text{platform}}\mathbf{M}_{\text{ext}} - \sum_{i=1}^{6}(\mathbf{q}_i \times \mathbf{K}_i - \mathbf{f}_i) \end{bmatrix} \tag{14}$$

$$\mathbf{K}_i = \frac{\mathbf{C}_i \times \mathbf{s}_i}{L_i} \tag{15}$$

$$\mathbf{C}_i = \left[ m_d \mathbf{r}_d \times \mathbf{a}_d + m_u \mathbf{r}_u \times \mathbf{a}_u - (m_d \mathbf{r}_d + m_u \mathbf{r}_u) \times \mathbf{g} + (\mathbf{I}_d + \mathbf{I}_u)\mathbf{A} + \mathbf{W} \times (\mathbf{I}_d + \mathbf{I}_u)\mathbf{W} + C_u \mathbf{W} + \mathbf{f} \right]_i \tag{16}$$

$$\mathbf{f} = C_s(\mathbf{W} - \boldsymbol{\omega}) \tag{17}$$

Having found these forces, the individual leg actuator forces are given by

$$F_i = \left[ m_u \mathbf{s} \bullet (\mathbf{a}_u - \mathbf{g}) + C_p \dot{L} \right]_i - x_i \tag{18}$$

The description of the various variables is given in the Nomenclature section. The detailed analysis of the kinematics and dynamics of the platform and the legs can be obtained by referring to Dasgupta and Mruthyunjaya[11]. The dynamic inversion controller can now be designed by specifying the desired values of the translational and angular accelerations ($\mathbf{a}_{\text{des}}$ and $\boldsymbol{\alpha}_{\text{des}}$) and substituting them in place of $\mathbf{a}$ and $\boldsymbol{\alpha}$ in Eq. (14) and computing the individual leg actuator forces.

## III. Simulation Results

The control system outlined in the previous section is coded in Simulink® and provides the actuating forces at the leg prismatic joints to the forward dynamics of the system. The forward dynamics is simulated using a physics engine that builds the docking mechanism as multiple bodies with the appropriate joints connecting them. The physics engine is coded in C/C++ and accepts inputs from the Simulink® controller through a shared memory interface. The physics engine also renders the output of the simulation as a real time graphic.

The simulation results are presented for a test case consisting of commanded rotation without any commanded translation.

$$\mathbf{q}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{q}_c = \begin{bmatrix} 0.25 \\ -0.25 \\ 0.067 \\ 0.933 \end{bmatrix}$$

The commanded quaternion corresponds to Euler angles (yaw, pitch, roll) of 30 deg., -30 deg. and 0 respectively. Figure 4 illustrates the graphic output of the docking mechanism for the first 105 seconds of the simulation.

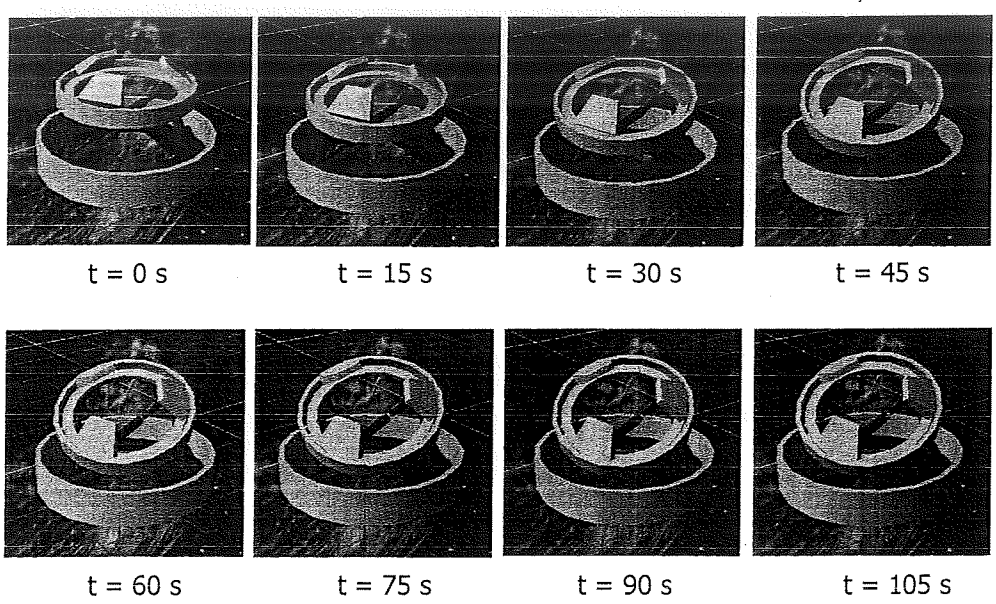


**Figure 4: Platform attitude change graphical output for the first 105 seconds of simulation**

Figure 5 presents the comparison of the platform quaternion with the reference quaternion. Figure 6 compares the platform and reference angular velocities. Figure 7 presents the actuator force inputs at each of the six prismatic leg joints and figure 8 presents the evolution of the lengths of each of the six legs with time. It can be inferred that the control system is successful in following the specified reference trajectories towards achieving the commanded attitude.

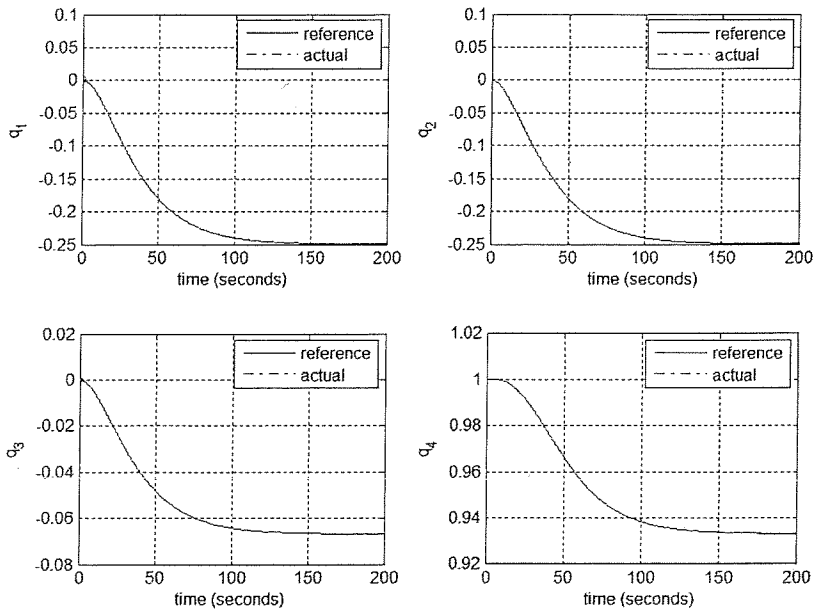


**Figure 5: Platform reference and actual quaternion trajectories**

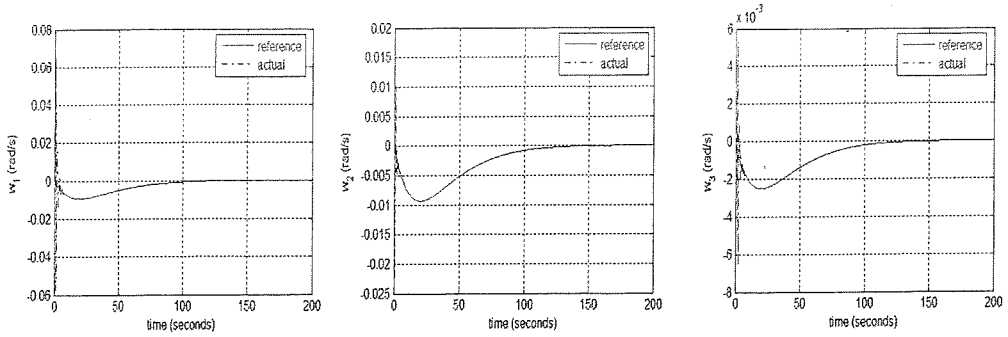


**Figure 6: Platform reference and actual angular velocities**

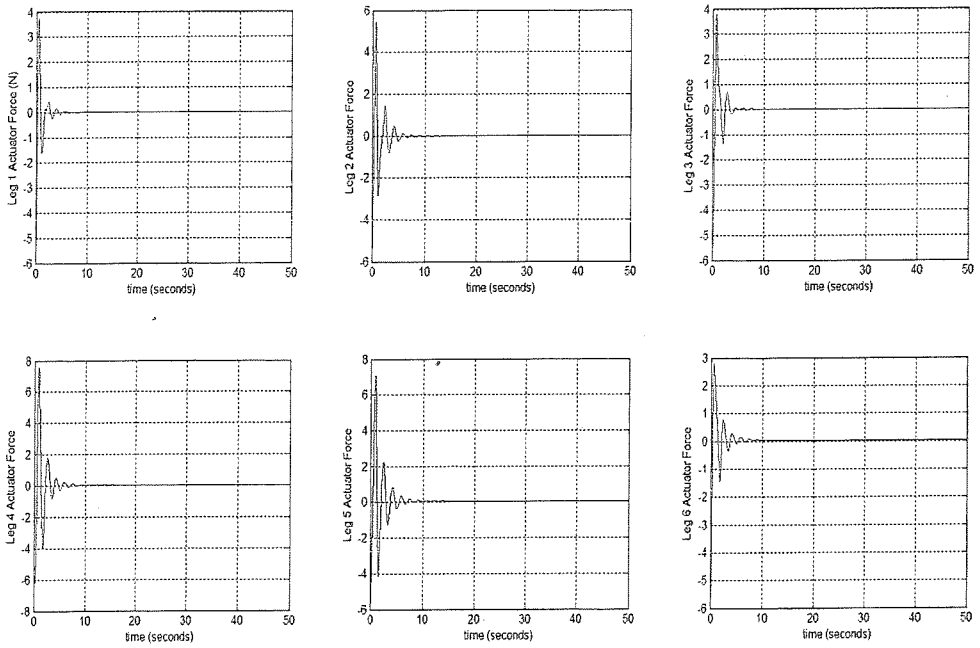


**Figure 7: Force inputs from the actuators in the legs**

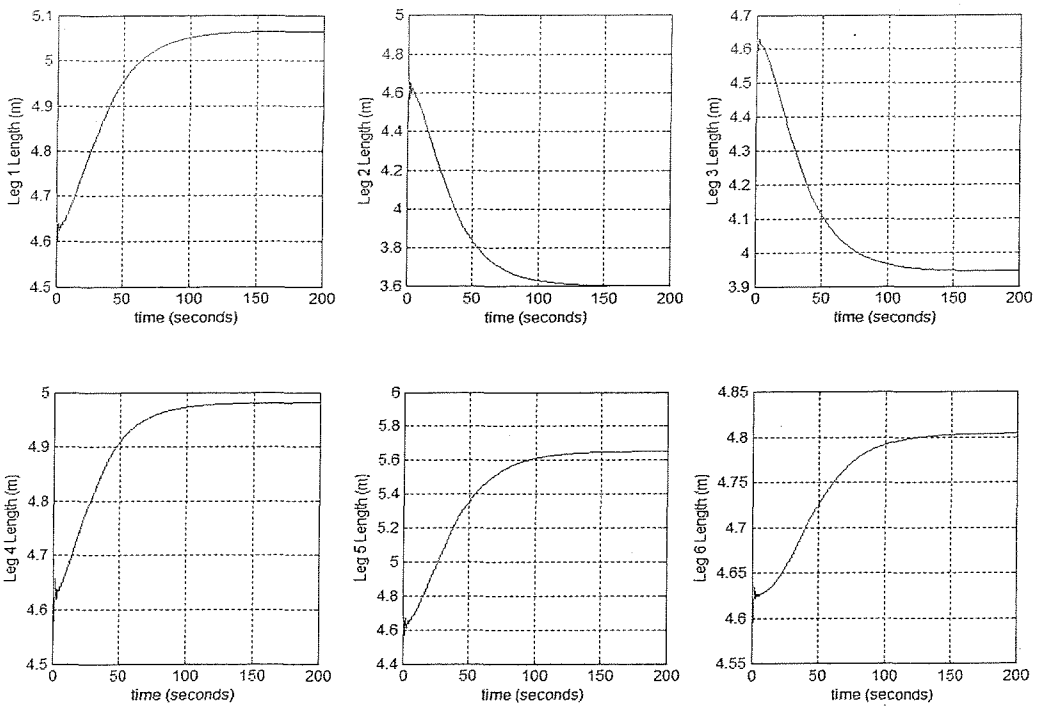


**Figure 8: Time evolution of the lengths of the six legs**

## IV. Conclusion

A dynamic inversion based control system with proportional-integral augmentation has been designed and simulated for the space docking mechanism. The control system fits within the intelligent flight control (IFC) architecture that has been designed and implemented for several aircraft platforms at the Ames Research Center. The current control design for the docking mechanism represents the baseline control system in the IFC architecture. This baseline control system, as in the IFC framework, can be adapted using neural network augmentation. The neural network will provide additional translational and angular acceleration commands to account for modeling errors or faults in the system. The adaptive control implementation is currently being designed. The proposed control architecture, while designed with the space docking motivation, can also be applied to general purpose Stewart platforms in flight simulator and robotic applications.

## References

1

[2]Kaneshige, John and Gundy-Burlet, Karen, Integrated Neural Flight and Propulsion Control System, AIAA-2001-4386, August 2001

[3]An Adaptive Critic Approach to Reference Model Adaptation, K. KrishnaKumar, G. Limes, K. Gundy-Burlet, D. Bryant, AIAA-2003-5790, Presented at the AIAA GNC

[4]Control Reallocation Strategies for Damage Adaptation in Transport Class Aircraft, K. Gundy-Burlet, K. KrishnaKumar, G. Limes, D. Bryant, D., AIAA-2003-5642, Presented at the AIAA GNC Conference, Austin, Texas, August 11-14, 2003

Conference, Austin, Texas, August 11-14, 2003.

[5]Rysdyk, Rolf T., and Anthony J. Calise, Fault Tolerant Flight Control via Adaptive Neural Network Augmentation, AIAA 98-4483, August 1998.

[6]Stewart, D., Proceedings of the Institute of Mechanical Engineers (Part 1) 180(15), pp. 371-386, 1965-66.

[7]Gough, V. E., and WhiteHall, S. G., Proceedings of the 9th International Technical Congress F.I.S.I.T.A., May 1962, Institution of Mechanical Engineers, 117, 1962.
[8]Hunt, K. H., Kinematic Geometry of Mechanisms, Clarendon Press, Oxford, 1978.
[9]McCallion, H., and Truong, P. D., Proceedings of the 5th World Congress on Theory of Machines and Mechanisms, pp. 611-616, 1979.
[10]Zhou, Z., and Colgren, R., "Spacecraft Nonlinear Attitude Tracking Control with Non-constant Rate Command," AIAA-2004-5128, *AIAA Guidance, Navigation and Control Conference*, Providence, RI, Aug. 2004.
[11]Dasgupta, B., and Mruthyunjaya, T. S., "A Newton-Euler Formulation for the Inverse Dynamics of the Stewart Platform Manipulator", Mechanism and Machine Theory, Vol. 33, No. 8, pp. 1135-1152, 1998.