Proceedings of the 2002 IEEE
International Conference on Robotics & Automation
Washington, DC • May 2002

The Operational Space Formulation Implementation to Aircraft Canopy Polishing Using a Mobile Manipulator*

Rodrigo Jamisola, Marcelo H. Ang, Jr., Denny Oetomo
National University of Singapore
Singapore

Oussama Khatib
Stanford University
U.S.A.

Tao Ming Lim, Ser Yong Lim
Gintic Inst. of Manufacturing Technology
Singapore

Abstract

The Operational Space Formulation provides a framework for the analysis and control of manipulator systems with respect to the behavior of their end-effectors. Its application to aircraft canopy polishing is shown using a mobile manipulator. The mobile manipulator end-effector maintains a desired force normal to the canopy surface of unknown geometry in doing a compliant polishing motion, while, at the same time, its mobile base moves around the shop floor, effectively increasing the mobile manipulator's workspace. The mobile manipulator consists of a PUMA 560 mounted on top of a Nomad XR4000. Implementation issues are discussed and simultaneous motion and force regulation results are shown.

1 Introduction

Dynamic interaction between a manipulator and its environment is one of the most important goals of robotic systems. To be able to do this, the concept of simultaneous force and motion control must be well evaluated and implemented.

Manipulator force and motion control as the manipulator interacts with its environment can be categorized into two families [25]. The first is when force is controlled along the directions constrained by the environment while motion is controlled along the direction of free motion ([20], [16], [22], [14], [17], and [24]). The second is when manipulator position is controlled and its relationship with the environment interaction force are simultaneously specified ([8], [11], [23], and [21]).

In this paper, we apply the *operational space formulation* [14] to control of mobile manipulator polishing an aircraft canopy of unknown geometry (Figure 1). The mobile manipulator consists of the six-axis articulated arm (PUMA 560) mounted on an omni-directional mobile base capable of holonomic motion (Nomad XR4000). The operational space formulation analyzes the robot dynamics as seen from the operational space (i.e., end-effector or tool frame) and realizes force and motion control at this frame according to the desired dynamic behavior. In our polishing application, the robot maintains a contact force normal to the canopy surface while the polishing tool is moving tangentially to cover the polishing area. The canopy is of unknown geometry and the robot is mounted on a mobile base whose motion is controlled by a human operator and is unknown to the robot. The operator controlling the joy stick provides higher level intelligence to move the mobile manipulator to cover the entire canopy, while our compliant motion control represents lower level intelligence, so human operator need not bother with commanding the motion of the polishing tool to follow the surface, which is automatically done. This paper shows the robustness of the operational space formulation to achieve compliant motion during force and motion control tasks in spite of robot base disturbances and unknown contact geometries between the robot and environment.

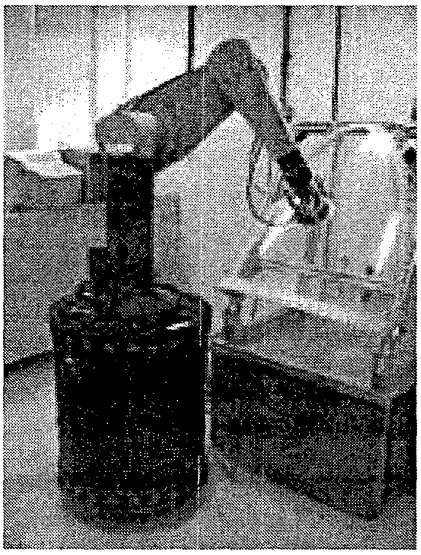

Figure 1: Mobile manipulator polishing an aircraft canopy

The dynamics of the PUMA 560 robot is modelled and the model parameters are identified. The PUMA is tasked to polish by moving its end-effector back and forth on the aircraft canopy surface with simultaneous force and moment control. The Nomad is then moved via joystick, effectively moving the PUMA base as the PUMA polishes the aircraft canopy. The Nomad motion is taken as disturbance to the PUMA polishing task. Robust force control maintains the contact of the PUMA end-effector with the canopy surface and maintains the desired force to 10 N $\pm$ 4 N as the Nomad moves around the canopy. The motion control on $\vec{x}$ and $\vec{y}$ position and *yaw* rotation of the PUMA, and force control on the rest of the axes, allows it to assume different robot configurations while maintaining desired force and moments at the end-effector as the base is moved. Implementation issues and results are discussed.

*This work is sponsored by Gintic Institute of Manufacturing Technology.

2 Operational Space Force and Motion Control in an Unknown Environment

The end-effector equations of motion in operational space $\mathbf{x}$ (i.e., end-effector or tool frame coordinates) can be written in the form [12, 14]:

$$\mathbf{\Lambda}(\mathbf{x})\ddot{\mathbf{x}} + \mu(\mathbf{x}, \dot{\mathbf{x}}) + \mathbf{p}(\mathbf{x}) = \mathbf{F} \quad (1)$$

where $\mathbf{\Lambda}(\mathbf{x})$ is the inertia matrix, $\mu(\mathbf{x}, \dot{\mathbf{x}})$ is the centrifugal and Coriolis forces, $\mathbf{p}(\mathbf{x})$ is the gravity vector, and $\mathbf{F}$ is the operational force exerted on the end-effector. Motion and force control is achieved by specifying the required force, $\mathbf{F}$:

$$\mathbf{F} = \hat{\mathbf{\Lambda}}(\mathbf{x})\left(\mathbf{\Omega}\ \mathbf{F}^*_{motion} + \bar{\mathbf{\Omega}}\ \mathbf{F}^*_{force}\right) + \hat{\boldsymbol{\mu}}(x, \dot{x}) + \hat{\mathbf{p}}(x) + \bar{\mathbf{\Omega}}\mathbf{f}_d \quad (2)$$

where $\mathbf{F}^*_{motion}$ and $\mathbf{F}^*_{force}$ are the desired motion and force responses:

$$\mathbf{F}^*_{motion} = \ddot{\mathbf{x}}_d - \mathbf{k}_{v\_motion}(\dot{\mathbf{x}} - \dot{\mathbf{x}}_d) - \mathbf{k}_{p\_motion}(\mathbf{x} - \mathbf{x}_d) \quad (3)$$

$$\mathbf{F}^*_{force} = \mathbf{k}_{p\_force}(\mathbf{f}_d - \mathbf{f}) + \mathbf{k}_{i\_force}\int(\mathbf{f}_d - \mathbf{f}). \quad (4)$$

$\ddot{\mathbf{x}}_d$, $\dot{\mathbf{x}}_d$, and $\mathbf{x}_d$ are the desired operational space acceleration, velocity, and displacement, respectively. $\mathbf{f}$ is the actual force exerted by the manipulator on the environment and is related to the force sensor reading, $\mathbf{F}_{sensor}$, by $\mathbf{f} = -\mathbf{F}_{sensor}$; and $\mathbf{k}$ are the corresponding gains. $\mathbf{\Omega}$ and $\bar{\mathbf{\Omega}}$ are selection matrices that define the directions of motion and force control respectively [14]. The generalized joint forces $\mathbf{\Gamma}$ to produce the operational force $\mathbf{F}$ in Eq. (2) are:

$$\mathbf{\Gamma} = \mathbf{J}^T(q)\,\mathbf{F}, \quad (5)$$

which form the basis of the actual control of manipulators in operational space.

3 Canopy Polishing Application

The choice of operational space frame is crucial in the implementation of this work. The chosen operational space for motion control, is at the force sensor frame, which is at an offset distance from the wrist frame (Fig. 2) . This frame, with its origin at the motion operational space point, $\mathcal{O}_{motion}$, is specified as $\mathbf{S}_{\mathcal{O}}$ ($\mathcal{O}_{motion}$, x_e, y_e, z_e). While the operational space frame for force control is at the tip of the tool that is attached to the force sensor. This frame, with its origin at the force operational space point, $\mathcal{O}_{force}$, is specified as $\mathbf{S}_{\mathcal{O}}$ ($\mathcal{O}_{force}$, x_e, y_e, z_e).

It would have been simpler to assign the operational space points for motion and force at the same point in the end-effector. There are several reasons why the operational space points were separated. The force operational space point must be placed at the tip of the polishing tool to have better force and moment control at the surface of the aircraft canopy. If the motion operational space point was also placed at the tip of the polishing tool together with the force operational space point, the motion control response would not be robust because of the considerable degree of flexibility of the polishing tool, compared to the highly rigid links of the PUMA arm. The robot control is based on the dynamics of rigid bodies. This degree of flexibility of the polishing tool creates higher penalty on the motion control.

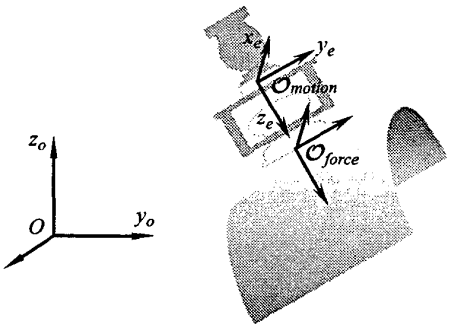

Figure 2: The choice of operational point for simultaneous force and motion control.

Force is controlled in the direction of the $\vec{z}_e$ axis, which is along the line of contact between the manipulator end-effector and the aircraft canopy, releasing (not controlling) the position along this direction. This released axis moves together with the end-effector as it moves about the canopy surface, thus, maintaining axis of force control to be along $\vec{z}_e$ axis that is normal to the aircraft canopy surface. A desired force is specified along this axis which is the desired force that the manipulator applies to the aircraft canopy.

Moment is controlled about the $\vec{x}_e$ and $\vec{y}_e$ axes. This plane is perpendicular to the axis of force control and is parallel to a tangent plane that contains the contact point on the aircraft canopy. By specifying the desired moments about these two axes to be zero, the tool can comply to the surface of the aircraft canopy as it moves about it. Control actions to achieve the desired moments to be zero will naturally move the polishing tool tangent to the surface.

Motion is controlled along the remaining axes which are not controlled in force or moment: position control along $\vec{x}_e$ and $\vec{y}_e$ axes and orientation control about the $\vec{z}_e$. Fig. 3 shows superimposed snapshots of the polishing with compliant motion during our real-time implementation. The geometry of the surface is unknown. The moments about $\vec{x}_e$ and $\vec{y}_e$ axes are controlled to be zero, thus allowing the polishing to automatically orient itself to be tangent to the surface while maintaining a desired normal force to the surface.

Given this choice of operational frames and axes to control motion and force, proper transformations need to be done before combining the forces of motion and force as in Eq. 2.

Forces and moments at the force operational space point, which includes desired and actual forces/moments and forces of force control, are expressed with respect to the force operational space frame $\mathbf{S}_{\mathcal{O}}$ ($\mathcal{O}_{force}$, x_e, y_e, z_e). These are then converted to forces/moments at the force operational space frame expressed with respect to the base frame. And lastly, these force/moments need to be converted as equivalent forces/moments at the origin of the wrist frame expressed with respect to the base frame. The reason for this is for consistency in multiplying these force operational space point forces/moments with the operational space lambda matrix, $\mathbf{\Lambda}(x)$, and the manipulator Jacobian matrix, $\mathbf{J}(q)$. The operational space lambda matrix, $\mathbf{\Lambda}(x)$, is derived as the equivalent

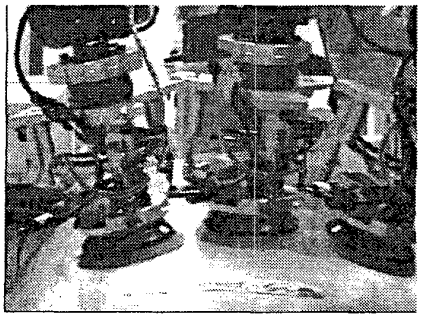

Figure 3: The figure shows a sequence of overlaid snapshots of the mobile manipulator end-effector during a compliant motion polishing task.

operational space manipulator inertia matrix at the wrist frame expressed with respect to the base frame, while the manipulator Jacobian matrix is derived from the operational space velocities at the wrist frame expressed with respect to the base frame.

4 Robot Arm Singularity Handling

Singularities, if not properly addressed, limit a manipulator's achievable configurations needed to do a desired task. In actual implementation, a region around each singularity is defined. At this region, modified algorithms (such as [18, 1, 7, 6, 15]) have to be designed and implemented to compensate for the lost degree(s) of freedom.

Outside the singular region, usual control resumes and in our work we use the operational space control as described in Section 2. Smoothness in the transition in and out of the singular region is important. At a singular configuration, the operational space inertia matrix, $\mathbf{\Lambda}(x)$, does not exist because the manipulator Jacobian matrix, $\mathbf{J}(q)$, that is used to solve for the operational space inertia matrix, $\mathbf{\Lambda}(x)$ has its rank reduced. We employ a method motivated by Chang, and Oetomo [5, 19] to achieve singularity robust operational space control in the singular region. The singularity handling is done by first expressing the manipulator Jacobian matrix with reduced rank, $\mathbf{J}(q)$, to a reference frame where one or more axes represent the lost degree of freedom. The Jacobian is truncated by removing the row(s) corresponding to the lost degree(s) of freedom. The truncated Jacobian $\mathbf{J}(q)^*$, would have its number of rows equal to its rank. The reduced operational space inertia matrix expressed as $\mathbf{\Lambda}(x)^*$ would have the same size as the rank of the truncated Jacobian matrix, $\mathbf{J}(q)^*$. The truncated control law vector, $\mathbf{F}^{**}$, is expressed in the same frame as the truncated Jacobian matrix, $\mathbf{J}(q)^*$, and has zero control along the axis of singularity.

For a PUMA 560 robot, there are three singularity configurations: head, elbow, and wrist singularities. We apply the method of [19] in a region around the wrist singularity of the PUMA 560 robot. We express the manipulator Jacobian, $\mathbf{J}(q)$ in link four frame, $\mathbf{S}_4$ ($\mathbf{4}$, x_4, y_4, z_4), where the fourth row represents the lost degree-of-freedom (i.e., rotation about the x-axis of Frame 4). The torque to be sent to the manipulator joints, considering only pure motion, would be

$$\boldsymbol{\tau} = {}^4\mathbf{J}_e^{*T}(q)\ {}^4\mathbf{\Lambda}_e^*(x)\,{}^4\mathbf{F}_{motion}^{**} + {}^\circ\mathbf{J}_e(q)^T[\hat{\boldsymbol{\mu}}(x,\dot{x}) + \hat{\mathbf{p}}(x)] \quad (6)$$

where

$${}^4\mathbf{\Lambda}_e^*(x) \;=\; [\,{}^4\mathbf{J}_e^*(q)\ \hat{\mathbf{A}}^{-1}(q)\ {}^4\mathbf{J}_e^{*T}(q)]^{-1}. \quad (7)$$

${}^4\mathbf{\Lambda}(x)_e^*$ is the operational space matrix expressed in link four frame with its size reduced by 1. ${}^4\mathbf{J}_e(q)^*$ is derived from the manipulator Jacobian matrix expressed in the link four frame, ${}^4\mathbf{J}_e(q)$, with the fourth row truncated. It was observed that the manipulator motion was not smooth during the transition between the singular and non-singular regions, and trajectory modifications were done to alleviate this problem [19].

This singularity handling needs some joint space damping control as the manipulator enters and leaves the singularity region. Truncating the 6×6 ${}^4\mathbf{J}_e(q)$ results in a 5×6 Jacobian ${}^4\mathbf{J}_e(q)^*$ and ${}^4\mathbf{F}_{motion}^*$ has one less element representing the component which cannot be controlled because of the lost degree of freedom. Thus, given this condition, the manipulator would undergo from a *with-control-state* to a *no-control-state* as it enters the region of singularity and then undergo from a *no-control-state* to a *with-control-state* as it leaves the region of singularity [4]. This non-smooth transition in the manipulator control would make the robot jerk as it enters and leaves the singularity region. Even with damping, this jerking is still considerably present, and it creates greater penalty in its motion control especially during simultaneous force and motion control.

To avoid such a non-smooth transition, we employ a slightly different strategy where the manipulator is made to undergo from *with-dynamics-control-state* to *non-dynamics-control state* as it enters the singularity region and from *non-dynamics-control-state* to *with-dynamics-control-state* as it leaves the singularity region. This control strategy creates lower penalty in the force and motion control within the singularity region.

The singularity handling strategy used here plays around with the operational space lambda matrix, $\mathbf{\Lambda}(x)$, and the rest of the dynamics control terms would remain the same. Note that at this point, the proposed control strategy does not only eliminate the need for damping in joint space around the region of singularity, but also helps in easier singularity implementation by doing matrix manipulation of $\mathbf{\Lambda}(x)$ only around the region of singularity. We apply this to the wrist singularity. Instead of truncating ${}^4\mathbf{J}_e(q)$ which results in a reduced 5×5 ${}^4\mathbf{\Lambda}_e^{-1}(x)$, the fourth row and fourth column of ${}^4\mathbf{\Lambda}_e^{-1}(x)$ are padded with zeros except for element ${}^4\lambda_{44}^*$ which is set to unity. This results in

$${}^4\mathbf{\Lambda}_e^{-1}(x) = \begin{bmatrix} {}^4\lambda_{11}^* & \dots & {}^4\lambda_{13}^* & 0 & {}^4\lambda_{15}^* & {}^4\lambda_{16}^* \\ . & \dots & . & 0 & . & . \\ {}^4\lambda_{31}^* & \dots & {}^4\lambda_{33}^* & 0 & {}^4\lambda_{35}^* & {}^4\lambda_{36}^* \\ 0 & 0 & 0 & 1 & 0 & 0 \\ {}^4\lambda_{51}^* & \dots & {}^4\lambda_{53}^* & 0 & {}^4\lambda_{55}^* & {}^4\lambda_{56}^* \\ {}^4\lambda_{61}^* & \dots & \lambda_{63}^* & 0 & {}^4\lambda_{65}^* & {}^4\lambda_{66}^* \end{bmatrix} \quad (8)$$

where λ_{ij}^* represents the element of the inverse of the operational space lambda matrix, $\mathbf{\Lambda}(x)^{-1}$. Now, the inverse of this matrix *does* exist. The operational space lambda matrix, $\mathbf{\Lambda}(x)$,

exists, and would turn out to be,

$$ {}^4\Lambda_e(x) = \begin{bmatrix} {}^4\lambda_{11} & \dots & {}^4\lambda_{13} & 0 & {}^4\lambda_{15} & {}^4\lambda_{16} \\ . & \dots & . & 0 & . & . \\ {}^4\lambda_{31} & \dots & {}^4\lambda_{33} & 0 & {}^4\lambda_{35} & {}^4\lambda_{36} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ {}^4\lambda_{51} & \dots & {}^4\lambda_{53} & 0 & {}^4\lambda_{55} & {}^4\lambda_{56} \\ {}^4\lambda_{61} & \dots & {}^4\lambda_{63} & 0 & {}^4\lambda_{65} & {}^4\lambda_{66} \end{bmatrix} \quad (9) $$

where λ_{ij} represents the element of the of the operational space lambda matrix, $\Lambda(x)$. If the element ${}^4\lambda_{44}$ of the above expression is set to *zero*, this approach would effectively be equivalent to truncating the fourth row and fourth column of the matrix as done in [19]. The only difference is the size of the matrix.

The above expressions have several implications. By truncating the operational space lambda matrix, $\Lambda(x)$, to a size equal to its rank, or using an equivalent approach of padding the fourth row and fourth column with zeros and retaining operational space lambda matrix's, $\Lambda(x)$'s, original size, the fourth element of the motion control law, ${}^4\mathbf{F}^*_{motion}[4]$ is forced to a zero value within the bounds of singularity. This is done in the previous approach to singularity handling and this effectively creates zero control along the axis of singularity within the bounds of singularity. The significance of setting ${}^4\lambda^*_{44} = 1$ is to get value of the operational space lambda matrix, $\Lambda(x)$, without changing the size of ${}^4\Lambda_e^{-1}(x)$ and getting its inverse. By setting ${}^4\lambda^*_{44} = 1$, the operational space lambda matrix, $\Lambda(x)$, already exists. The unity value of ${}^4\lambda^*_{44}$ does not affect the value of the rest of the elements in the matrix. However, it *does* eliminate the dynamics contribution along the degenerate direction. It should be noted, that as opposed to the previous approach, the fourth element of the motion control law, ${}^4\mathbf{F}^*_{motion}[4]$, still applies and is not set to zero. Having ${}^4\lambda^*_{44} = 1$ is basically a *non-dynamics control state* with the absence of the dynamics values of dominant inertia and coupling inertias in the fourth row and fourth column of the $\Lambda(x)$ matrix.

The final operational space lambda matrix is then transformed back to the base frame before applying the operational space force in Eq. 2. The implementation results are shown in Section 7.

5 Impact Loading and Control

The mobile manipulator goes through the following 3 stages in our polishing application. First is pure motion control on approaching the canopy, followed by impact loading control upon contact, and finally force and motion control during contact with the canopy.

A smooth transition from contact and non-contact state between the manipulator tool and the environment is a crucial part in the interaction between the manipulator and the environment. Here impact loading and control is used [13].

On its first stage of approaching the aircraft canopy, a point below the surface of canopy and along the end-effector $\vec{z}_e$ axis is taken as the desired point to set collision scenario between the canopy and the polishing tool. Selection matrices are appropriately chosen to set all the axes in motion control mode.

Force along the $\vec{z}_e$ is monitored throughout the motion with a threshold of 10 N. As the tool collides with the canopy, the force sensor would sense a force along $\vec{z}_e$ that is way above a preset threshold. The manipulator would then enter into the second stage using full motion control but instead of applying Eq. 3, $\mathbf{F}^*_{motion}$ is changed to

$$ \mathbf{F}^*_{motion} = - \mathbf{k}_{v\_motion}\dot{\mathbf{x}}. \quad (10) $$

This control law would dissipate the impact of the collision between the tool and the aircraft canopy [13].

After dissipating the impact with the force normal to the aircraft canopy falling below the threshold value of 10 N, the manipulator then enters into the final stage of polishing where both motion and force are controlled with the appropriate choice of selection matrices (as discussed in Section 3). Motion and force control are done using Eqs. 2 to 4 with singularity handling done by Eqs. 8 to 9. Impact, force, and motion responses are shown in Section 7.

6 Friction Parameters

Modelling and compensating frictional effects can improve the performance of robots. Friction is modeled to include static, kinetic (or Coulomb), and fluid friction [3]. The friction model we use is:

$$ \tau_{friction} = f_s \left(\frac{\text{sgn}(\dot{q})}{1 + (\frac{\dot{q}}{x_s})^2} \right) + f_k \tanh(\dot{q}) + k_{vn}\,\dot{q} \quad (11) $$

where f_s is static friction; x_s is a constant to correct static friction due to Stribeck Effect; f_k is kinetic friction; and, k_{vn} is fluid friction.

Experiments are performed to identify these friction parameters. Fluid friction is identified together with the parameters of the inertia matrix [9]. Static and kinetic friction are derived as discussed in [10] and [2]. The identified parameters are shown in Table 1. The torques to compensate for friction

Table 1: Friction parameters of PUMA 560.

Link	f_s	f_k	k_{vn}	x_s
1	5	2	1	0.1
2	5	2	1	0.1
3	2.5	1	1	0.1
4	0.3	0.1	0.05	0.1
5	0.2	0.1	0.05	0.1
6	0.2	0.1	0.05	0.1

in Eq. 11 is then added to the joint torques required to produce the operation force in Eq. 5, yielding:

$$ \mathbf{\Gamma} = \mathbf{J}^T(q)\,\mathbf{F} + \tau_{friction}. \quad (12) $$

7 Implementation Results

Implementation results of the aircraft canopy polishing using a mobile manipulator are shown here. Emphasis on the implementation results are focused on how well the mobile manipulator can achieve the desired normal force to the aircraft canopy surface, with its end-effector moving in a compliant behavior as it does polishing motion around the surface of the aircraft canopy. Disturbances are introduced to the manipulator end-effector in the form of the vibration by the active

grinding tool, and the motion of the mobile base via joystick operation of the human operator.

Fig. 4 shows tracking errors during free motion at a maximum speed of 1.9 m/s. DPhi is the end-effector orientation

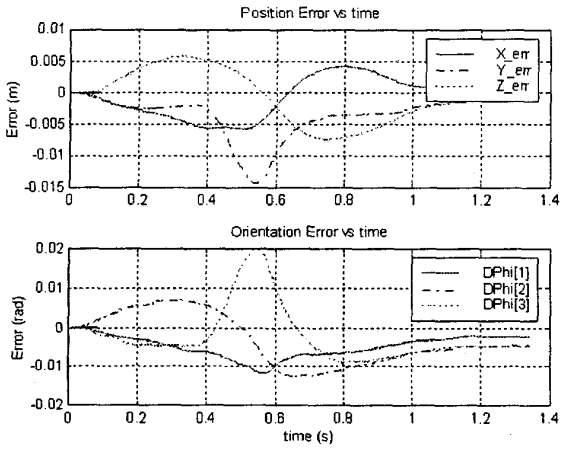

Figure 4: Error response of the PUMA in free motion running at a maximum speed of 1.9 m/s in Operational Space.

error expressed as,

$$\delta\,\Phi\;=\;-\frac{1}{2}(\lfloor {}^{o}\mathbf{r}_{e_1}\times\rfloor {}^{o}\mathbf{r}_{e_{1d}}\;+\;\lfloor {}^{o}\mathbf{r}_{e_2}\times\rfloor {}^{o}\mathbf{r}_{e_{2d}}\;+\;\lfloor {}^{o}\mathbf{r}_{e_3}\times\rfloor {}^{o}\mathbf{r}_{e_{3d}}) \tag{13}$$

where ${}^{o}\mathbf{R}_e = [{}^{o}\mathbf{r}_{e_1}\ {}^{o}\mathbf{r}_{e_2}\ {}^{o}\mathbf{r}_{e_3}]$ and the subscript d specifies the desired orientation equivalents. Position errors are shown as *X_err*, *Y_err*, and *Z_err* in units of meters while orientation errors are shown as DPhi[1], DPhi[2], and DPhi[3] in units of radians. Maximum position error is along the $\vec{y}$ axis being 0.014 m where the possible cause could be the PUMA motors saturating at its maximum velocity. Maximum orientation error around $\delta\,\Phi_z$ at around 1.16 degrees.

Fig. 5 shows the error response of the PUMA as it does its canopy polishing task. The end-effector is running in a non-terminating sinusoidal path along its $\vec{y}$ axis at 0.15 m amplitude and period of 5 s. The maximum position error is along the base frame axis $\vec{y}_o$ at 0.12 m. The desired normal force is more critical and position errors of this scale is tolerable in this application. Maximum force error along the end-effector frame $\vec{z}_e$ is at 4 N. We note here that the base is moved as the PUMA is doing its polishing task. Thus, the force error reflected here can be a measure at how well the mobile manipulator setup could maintain its desired force when doing its polishing task with the given disturbance.

The robustness of the singularity handling algorithm presented here is tested by the letting the PUMA polish the aircraft canopy as it goes in and out of the wrist singularity region. Fig. 6 shows the force reading exerted by the PUMA normal to the aircraft canopy surface at it moves in and out of the singularity region. The PUMA is set to move along its $\vec{x}_o$ while going through the wrist singularity, as shown by the Determinant of the manipulator Jacobian. 10 N is specified as the desired force normal to the canopy surface, applied as the robot goes in and out of the singularity region. The maximum force error reading is 3.7 N. This force error reading is lower

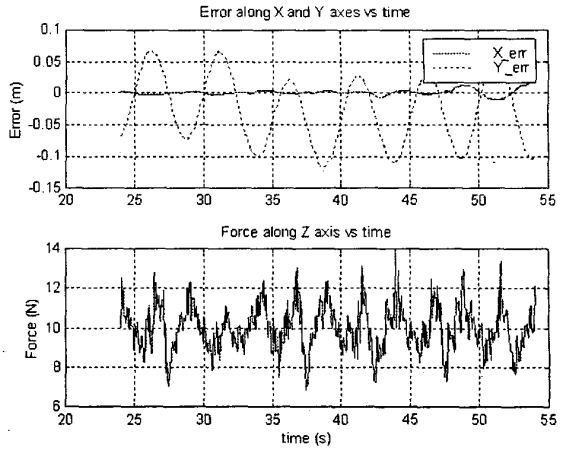

Figure 5: Error response of the PUMA doing polishing on the aircraft canopy with 10 N desired normal force and with Nomad base moving

compared to the force error reading in Fig. 5 because the base is not moved in this setup to force the PUMA to stay within the singularity region.

Fig. 7 shows the dissipation of impact forces as the end-effector makes contact with the aircraft canopy surface. Force is monitored as the end-effector approaches the canopy surface. And at the moment of impact, impact loading control is done using Eq. 10. Once the impact forces are dissipated, the PUMA then starts it polishing task. A threshold of 10 N is used to determine the moment of impact and the dissipation of impact forces.

Firction compensation was employed according to Eq. 12 in all the results presented. It was observed that friction compensation was able to reduce maximum force tracking errors by up to 32%.

8 Conclusion

It has been shown in this implementation that robust simultaneous force and motion control is possible for a mobile manipulator setup where the full dynamics of the manipulator arm is modelled while the mobile base motion is treated as disturbance. Further work focuses in implementing a fully integrated mobile manipulator incorporating full arm/base dynamics and control.

Acknowledgments

The financial support of Gintic Institute of Manufacturing Technology in this collaborative research project is gratefully acknowledged.

References

[1] E. W. Aboaf and R. P. Paul, "Living with the singularity of robot wrists," *IEEE Intl. Conf. for Robotics and Automation,*

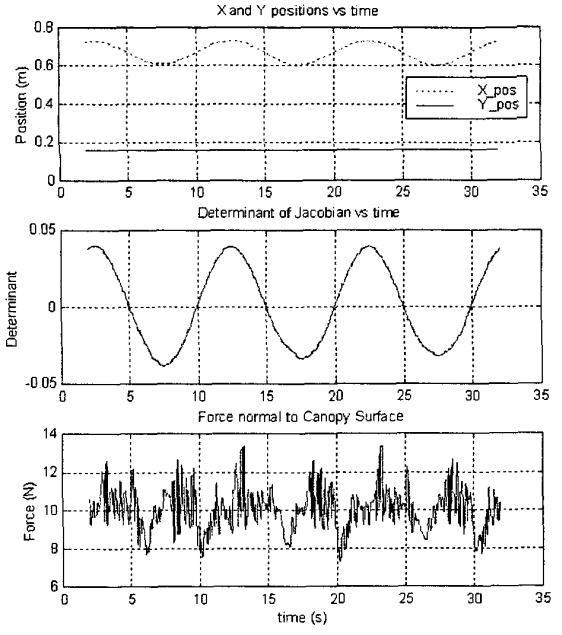

Figure 6: Force reading exerted by PUMA normal to the aircraft canopy surface at it moves in and out of wrist singularity region.

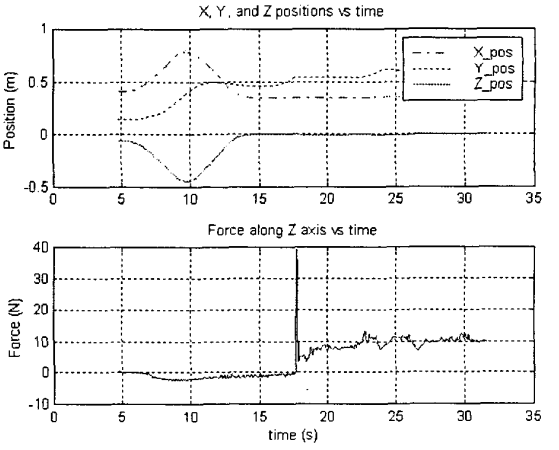

Figure 7: Force reading exerted by PUMA normal to the aircraft canopy surface before, during, and immediately after impact loading.

pp. 1713–1717, 1987.

[2] Armstrong-Hélouvry, B., "Control of Machines with Friction," Kluwer Academic Press, 1991.

[3] Beer, F.P., and Johnston, E.R., "Mechanics for Engineers: Statics and Dynamics," Mc-Graw Hill, 3rd ed., 1976.

[4] Bejczy, A. K., "Robot Arm Dynamics and Control," Technical Memo 33-669, Jet Propulsion Laboratory, Pasadena, Calif., 1974.

[5] Chang, K., and Khatib, O., "Manipulator Control at Kinematic Singularities: A dynamically consistent Strategy," Proc. IEEE/RSJ Int. Conference on Intelligent Robots and Systems, Pittsburgh, August 1995, vol. 3, pp. 84-88.

[6] Cheng, F.T., Hour, T.L., Sun, Y.Y., and Chen, T.H., "Study and Resolution of Singularities for a 6-DOF PUMA Manipulator", IEEE Trans. on Systems, Man, and Cybernetics- PartB: Cybernetics, vol. 27, no. 2, April 1997.

[7] S. Chiaverini and O. Egeland, "A solution to the singularity problem for six-joint manipulators," *Proc. IEEE for Robotics and Automation*, vol. 1, pp. 644–649, 1990.

[8] Hogan, N., "Impedance Control: An Approach to Manipulation ", ASME Journal of Dynamic Systems, Measurement, and Control, vol. 107, no. 1,pp. 1-24, March 1985.

[9] Jamisola, R., Ang, M.H., Jr., Lim, T.M., Khatib, O., Lim, S.Y., "Dynamics Identification and Control of an Industrial Robot", The Ninth International Conference on Advance Robotics, pp. 323-328, Oct. 1999.

[10] Jamisola, R. "Full Dynamics Identification and Control of PUMA 560 and Mitsubishi PA-10 Robots," Master's Thesis, National University of Singapore, 2001.

[11] Kazerooni, H. and Kim, S., "Contact Stability of the Direct Drive Robot when Constrained by a Rigid Environment", Proc. ASME Winter Annual Meeting, San Francisco, CA; Dec. 1989.

[12] Khatib, O., "Commande Dynamique dans l'Espace Operationnel des Robots Manipulateurs en Presence d'Obstacles," Ph.D. thesis, Ecole Nationale Superieure de l'Aeronautique et de l'Espace, Toulouse, France, 1980.

[13] Khatib, O., and Burdick, J. " Motion and Force Control of Robot Manipulators", IEEE Int. Conf. on Robotics and Automation, 1986.

[14] Khatib, O., "A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation", IEEE Journal on Robotics and Automation, vol. RA-3, no. 1, pp. 43-53, Feb. 1987.

[15] Kircanski, M.V., "Symbolical Singular Value Decomposition for a 7-DOF Manipulator and its Application to Robot Control," IEEE Conf. on Robotics and Automation,pp. 895-900, 1993.

[16] Mason M.T., "Compliance and Force Control for Computer Controlled Manipulators", IEEE Transactions on Systems, Man, and Cybernetics SMC-11(6):418-432, June 1982.

[17] J.K. Mills and A.A. Goldenberg, "Force and Position Control of Manipulators During Constrained Motion and Tasks", IEEE Transactions on Robotics and Automation, Vol.5, No.1,pp.304-320, Feb.1989.

[18] Y. Nakamura and H. Hanafusa, "Inverse kinematics solutions with singularity robustness fob robot manipulator control," *ASME J. of Dynamic Systems, Measurement, and Control*, vol. 108, pp. 163–171, 1986.

[19] Oetomo, D., Ang, M.H., Jr., Lim, S.Y., "Singularity Handling on PUMA in Operational Space Formulation," The Seventh Int. Symposium on Experimental Robotics, Hawaii, Dec. 2000.

[20] Paul, R. P. C., Shimano, B., "Compliance and Control", Proceedings of the Joint Automatic Control Conference, pp. 694-699, San Francisco, 1976.

[21] Pelletier, M., and Daneshmend, L. K., "An Approach to Compliant Motion Planning using Uncertain Impedance Models," IASTED International Conference on Control and Robotics, pages 58-61, Vancouver, Canada, 1992.

[22] Raibert, M. H. Craig, J. J., "Hybrid Position/Force Control of Manipulators", ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 102, June 1981.

[23] Salisbury, J.K., "Active stiffness control of a manipulator in Cartesian coordinates", Proc. 19th IEEE Conference on Decision and Control, Dec. 1980.

[24] Whitcomb, L.L., Arimoto, S., Naniwa, T., Ozaki, F., "Adaptive Model-Based Hybrid Control of Geometrically Constrained Robot Arms" , IEEE Transactions on Robotics and Automation, vol. 13, no. 1., Feb. 1997.

[25] Whitney, D.E., "Historical Perspective and State of the Art in Robot Force Control", The International Journal of Robotics Research, vol. 6, no.1, Spring 1987.