MyPAM Modelling

Contents

[1. Forward Kinematics 3](#_Toc530477180)

[2. Inverse Kinematics 4](#_Toc530477181)

[3. Assigning Denavit-Hartenberg Frames to obtain the Homogeneous Transformation Matrix 6](#_Toc530477182)

[4. Forming the Jacobian Matrix using the Homogenous Transformation Matrices 8](#_Toc530477183)

[5. Forming the Jacobian Matrix using Partial Differentiation 10](#_Toc530477184)

[6. Using Energy Equivalence to obtain a Jacobian Torque-Force Relationship 11](#_Toc530477185)

[7. Accounting for Mass, Inertia and Gravity 12](#_Toc530477186)

[7a. Dealing with Mass and Inertia 13](#_Toc530477187)

[7b. Accounting for gravity (just for completeness) 18](#_Toc530477188)

[8. Finalising the Control signal (not citable) 19](#_Toc530477189)

[9. Finalising the Control signal (citeable) 20](#_Toc530477190)

[10. Operational Space Control 21](#_Toc530477191)

[11. Trajectory Planning using Parametric Equations. 24](#_Toc530477192)

[12. A Simple Position Controller 25](#_Toc530477193)

# Forward Kinematics

|  |
| --- |
|  |
| Figure 1.1: Forward Kinematics for the MyPAM |

The End Effector (Hand) position is given by:

The Elbow position is given by:

The Shoulder (origin) position is given by:

# Inverse Kinematics

|  |
| --- |
|  |
| Figure 2.1: Inverse Kinematics for MyPAM |

|  |  |
| --- | --- |
|  | Eqn 2.1 |

By the Cosine rule:

|  |  |
| --- | --- |
|  | Eqn 2.2 |
|  | Eqn 2.3 |
|  | Eqn 2.4 |
|  | Eqn 2.5 |

|  |
| --- |
|  |
| Figure 2.2: Inverse Kinematics for MyPAM |

By the Cosine rule:

|  |  |
| --- | --- |
|  | Eqn 2.6 |
|  | Eqn 2.7 |
|  | Eqn 2.8 |

# Assigning Denavit-Hartenberg Frames to obtain the Homogeneous Transformation Matrix

|  |
| --- |
|  |
| Figure 3.1: The assigned Denavit-Hartenberg frames |

Populating the Denavit-Hartenberg parameter table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **θ** | **α** | **r** | **d** |
| **1** | θ0 | 0 | a2 | a1 |
| **2** | θ1 | 0 | a4 | a3 |

Form the Homogeneous Transformation Matrices, given by Equation 3.1:

|  |  |
| --- | --- |
|  | Eqn 3.1 |

Thus:

|  |  |
| --- | --- |
|  | Eqn 3.2 |
|  | Eqn 3.3 |

The Homogenous Transformation Matrix to transform from the base frame to the end effector frame is given by Equation 3.4:

|  |  |
| --- | --- |
|  | Eqn 3.4 |

Thus:

|  |  |
| --- | --- |
|  | Eqn 3.5 |

Consider the trigonomic identities shown by Equations 3.6 and 3.7:

|  |  |
| --- | --- |
|  | Eqn 3.6 |
|  | Eqn 3.7 |

Thus:

|  |  |
| --- | --- |
|  | Eqn 3.8 |

The Homogenous Transformation Matrix contains information about the rotational transformation and the displacement transformation, as shown by Equation 3.9:

|  |  |
| --- | --- |
|  | Eqn 3.9 |

Thus:

|  |  |
| --- | --- |
|  | Eqn 3.10 |
|  | Eqn 3.11 |
|  | Eqn 3.12 |

# Forming the Jacobian Matrix using the Homogenous Transformation Matrices

The Jacobian Matrix for MyPAM relates how movements of **θ** cause movements of **X**, as shown by Equation 4.5:

|  |  |
| --- | --- |
|  | Eqn 4.1 |
|  | Eqn 4.2 |
|  | Eqn 4.3 |
|  | Eqn 4.4 |
|  | Eqn 4.5 |

The Jacobian Matrix is formed using the information shown in table 4.1:

|  |  |  |
| --- | --- | --- |
|  | Prismatic | Revolute |
|  |  |  |
|  |  |  |

Thus, the Jacobian Matrix to transform joint velocities into cartesian end effector velocities is given by Equation 4.6:

|  |  |
| --- | --- |
|  | Eqn 4.6 |
|  | Eqn 4.7 |
|  | Eqn 4.8 |
|  | Eqn 4.9 |
|  | Eqn 4.10 |
|  | Eqn 4.11 |
|  | Eqn 4.12 |

Thus, the entire Jacobian Matrix for the end effector is given by Equation 4.13:

|  |  |
| --- | --- |
|  | Eqn 4.13 |

Substituting a2 = L0 and a4 = L1:

|  |  |
| --- | --- |
|  | Eqn 4.14 |

# Forming the Jacobian Matrix using Partial Differentiation

It is useful to check the Jacobian Matrix shown in Equation 4.11 by forming the Matrix using a different method and comparing them.

The Jacobian Matrix for MyPAM relates how movements of **θ** cause movements of **X**, as shown by Equation 5.1:

|  |  |
| --- | --- |
|  | Eqn 5.1 |

Considering only the x and y positions of the end effector, (x2, y2), since there is no change in z:

|  |  |
| --- | --- |
|  | Eqn 5.2 |

From Equation 1.1:

|  |  |
| --- | --- |
|  | Eqn 5.3 |
|  | Eqn 5.4 |
|  | Eqn 5.5 |
|  | Eqn 5.6 |
|  | Eqn 5.7 |
|  | Eqn 5.8 |

This gives:

|  |  |
| --- | --- |
|  | Eqn 5.9 |

It can be seen that the Jacobian Matrix shown by Equation 5.9 matches the top 2 rows shown by Equation 4.14, thus the 2X2 relationship shown by Equation 5.10 is proven (which is expected since the system is planar):

|  |  |
| --- | --- |
|  | Eqn 5.10 |

# Using Energy Equivalence to obtain a Jacobian Torque-Force Relationship

Conservation of energy is a property of all physical systems. This useful property means that the energy expended is the same in both operational space (cartesian space) and generalised coordinate space (joint space).

Work is the application of Force over a distance:

|  |  |
| --- | --- |
|  | Eqn 6.1 |
|  | Eqn 6.2 |
|  | Eqn 6.3 |
|  | Eqn 6.4 |
|  | Eqn 6.5 |

Power is the rate of work:

|  |  |
| --- | --- |
|  | Eqn 6.6 |
|  | Eqn 6.7 |

Equating power:

|  |  |
| --- | --- |
|  | Eqn 6.8 |

Substitute Equation 4.4:

|  |  |
| --- | --- |
|  | Eqn 6.9 |
|  | Eqn 6.10 |
|  | Eqn 6.11 |

Thus, there are now 2 uses for the Jacobian Matrix.

|  |  |
| --- | --- |
|  | Eqn 4.4 |
|  | Eqn 6.11 |

Equation 4.4 equates end effector velocities to joint velocities. Equation 6.11 equates end effector forces to joint torques.

# Accounting for Mass, Inertia and Gravity

It is necessary to account for inertia, mass and gravity in order to build a stable and accurate control signal for a normal system. For MyPAM it is possible to ignore gravity, since the system is planar.

Equation 7.1 characterises the Motor Torque Vector:

|  |  |
| --- | --- |
|  | Eqn 7.1 |

Where:

This allows the opportunity to rearrange for a Joint Torque control signal:

|  |  |
| --- | --- |
|  | Eqn 7.2 |

Where:

It is clear that setting the control signal equal to the torque vector shown in Equation 6.11 is not sufficient, since it does not account for the full dynamics of the MyPAM.

## 7a. Dealing with Mass and Inertia

The mass distribution for the MyPAM is shown by figure 7a.1:

|  |
| --- |
|  |
| Figure 7a.1: Mass distribution for the MyPAM |

Where:

COM1 = Centre of mass of Link L0.

COM2 = Centre of mass of Joint J1 (motor and gearbox).

COM3 = Centre of mass of Link L1.

COM4 = Centre of mass of end effector.

Figure 7a.2 shows the added Denavit-Hartenberg frames added to the centre of mass of the links:

|  |
| --- |
|  |
| Figure 7a.2: Mass distribution for the MyPAM with added frames |

A Jacobian Matrix must be found for each centre of mass. The first step is to find the Homogenous Transformation Matrix to transform the base frame to each centre of mass.

|  |  |
| --- | --- |
|  | Eqn 7a.1 |
|  | Eqn 7a.2 |
|  | Eqn 7a.3 |
|  | Eqn 7a.4 |
|  | Eqn 7a.5 |
|  | Eqn 7a.6 |
|  | Eqn 7a.7 |
|  | Eqn 7a.8 |
|  | Eqn 7a.9 |
|  | Eqn 7a.10 |

Thus, the Homogenous Transformation Matrixes can be formed:

|  |  |
| --- | --- |
|  | Eqn 7a.11 |
|  | Eqn 7a12 |
|  | Eqn 7a.13 |
|  | Eqn 7a.14 |

Which allows the formation of the overall Homogeneous Transformation Matrices for each COM:

|  |  |
| --- | --- |
|  | Eqn 7a.15 |
|  | Eqn 7a.16 |

Thus, the four Jacobian Matrices can be found (note that the Jacobian Matrix for COM4 is the same as the Jacobian Matrix for the end effector for obvious reasons):

|  |  |
| --- | --- |
|  | Eqn 7a.17 |
|  | Eqn 7a.17 |
|  | Eqn 7a.18 |
|  | Eqn 7a.19 |

With the Jacobians found for each COM, it is necessary to consider the energy in the system. It is useful to do so since energy is a scalar term which is the same in all coordinate systems. The total energy in the system can be calculated as the sum of energy from each source. The Jacobians are used to calculate the kinetic energy each link generates during motion.

|  |  |
| --- | --- |
|  | Eqn 7a.20 |
|  | Eqn 7a.21 |
|  | Eqn 7a.22 |
|  | Eqn 7a.23 |

Where:

mi is the mass of COMi.

Equation 7a.23 shows the Mass/Inertia Matrix in Cartesian space. It is necessary to find its equivalent in Joint space.

The effects of mass of each component must be calculated and summed:

|  |  |
| --- | --- |
|  | Eqn 7a.24 |

We can use the Jacobian relationship to transform from generalised cartesian coordinates into joint space.

Substitute the Jacobian relationship **:**

|  |  |
| --- | --- |
|  | Eqn 7a.25 |
|  | Eqn 7a.26 |
|  | Eqn 7a.27 |

Which contains the definition of the overall Mass/Inertia Matrix in Joint Space, shown by Equation 7a.28:

|  |  |
| --- | --- |
|  | Eqn 7a.28 |

## 7b. Accounting for gravity (just for completeness)

Again, using conservation of energy (work done is the same in all coordinate systems).

In cartesian coordinates, shown by Equation 7b.1:

|  |  |
| --- | --- |
|  | Eqn 7b.1 |

Where:

**F**gi is the gravitational force on each segment (simply -9.81 multiplied by the mass of the segment along the z-axis).

In Operational (joint) space, shown by Equation 7b.2:

|  |  |
| --- | --- |
|  | Eqn 7b.2 |

Substitute the Jacobian relationship and equate Equations 7b.1 and 7b.2:

|  |  |
| --- | --- |
|  | Eqn 7b.3 |
|  | Eqn 7b.4 |
|  | Eqn 7b.4 |

None of this is necessary for MyPAM, since it is planar in the xy plane.

# Finalising the Control signal (not citable)

Equation 7.2 shows the system dynamics with the control signal for controlling torque:

|  |  |
| --- | --- |
|  | Eqn 7.2 |

Rearranging the system dynamics:

|  |  |
| --- | --- |
|  | Eqn 8.1 |

Since the MyPAM is planar in xy, the effects of gravity can be omitted.

The Coriolis and centrifugal effects need to be accounted for with a highly accurate model of the moments of inertia. This is notoriously difficult to do. If the moments of inertia are inaccurate then the controller can introduce instability. Further to this, since the anticipated velocities are small, which means that the Coriolis and centrifugal effects will be negligible. For the above reasons, the Coriolis and centrifugal effects will also be omitted. This is generally considered fine to do if the Δt is small.

Thus, the motor torque control signal is given by Equation 8.2:

|  |  |
| --- | --- |
|  | Eqn 8.2 |

Implemented using a standard PD control formula:

|  |  |
| --- | --- |
|  | Eqn 8.3 |

Which gives:

|  |  |
| --- | --- |
|  | Eqn 8.4 |

# Finalising the Control signal (citeable)

Working from equation 8.2, the last point at which the maths is true/citeable:

|  |  |
| --- | --- |
|  | Eqn 8.2 |

In the above equation, is equivalent to a force on a unit mass (F=ma) in operational generalised coordinate (joint) space. Using Khatibs’ convention, we can define this:

|  |  |
| --- | --- |
|  | Eqn 9.1 |

This allows us to use a separate feedback controller for each joint, with the following feedback rule:

|  |  |
| --- | --- |
|  | Eqn 9.2 |

Giving:

|  |  |
| --- | --- |
|  | Eqn 9.3 |

Fully formed:

|  |  |
| --- | --- |
|  | Eqn 9.4 |

Whilst useful, this is difficult to implement because we don’t know what the requirements are to follow a trajectory in terms of joint variables.

# Operational Space Control

The control signal defined in section 8 is defined in generalised coordinates (joint space). It would be much more useful to define a control signal in operational space (cartesian coordinates relative to the base frame).

Using the relationship defined in Equation 6.11:

|  |  |
| --- | --- |
|  | Eqn 6.11 |

Rewriting **Fx** in its component parts:

|  |  |
| --- | --- |
|  | Eqn 10.1 |

Calculating **M(θ)** using Equation 7a.28 allows inertia to be accounted for by the control signal in generalised coordinates (joint space), but to account for inertia in Operational Space is more involved. **Mxee(θ)** shown above in Equation 10.1 is not just the normal inertial matrix, and must be derived.

The acceleration must be calculated in operational space:

|  |  |
| --- | --- |
|  | Eqn 10.2 |

By the product rule:

|  |  |
| --- | --- |
|  | Eqn 10.3 |

Consider Equation 8.1:

|  |  |
| --- | --- |
|  | Eqn 8.1 |

Ignoring gravity, the Coriolis and the centrifugal effects, as previously discussed:

|  |  |
| --- | --- |
|  | Eqn 10.4 |

Substituting Equation 9.4 into Equation 9.3:

|  |  |
| --- | --- |
|  | Eqn 10.5 |

Define the control signal using Equation 6.11:

|  |  |
| --- | --- |
|  | Eqn 6.11 |

Substituting Equation 10.1:

|  |  |
| --- | --- |
|  | Eqn 10.6 |

This gives:

|  |  |
| --- | --- |
|  | Eqn 10.7 |

Ignoring the first term due to the complexity of modelling:

|  |  |
| --- | --- |
|  | Eqn 10.8 |

Cancelling the accelerations, and rearranging:

|  |  |
| --- | --- |
|  | Eqn 10.9 |

Checking:

|  |  |
| --- | --- |
|  | Eqn 10.10 |
|  | Eqn 10.11 |

Returning to the control signal:

|  |  |
| --- | --- |
|  | Eqn 10.12 |

Finally giving:

|  |  |
| --- | --- |
|  | Eqn 10.13 |

Or, with a different PID equation shown in equation 9.2:

|  |  |
| --- | --- |
|  | Eqn 10.14 |

It is useful to note here that the following relationships exist:

|  |  |
| --- | --- |
|  | Eqn 10.15 |
|  | Eqn 10.16 |
|  | Eqn 10.15 |

The defined control signal converts desired end effector accelerations into torque commands, whilst compensating for inertia.

Its common in control engineering that

Further to this, for MyPAM a further gain term should multiply the whole equation ranging from 0-1, where 1 is complete assistance and 0 is no assistance.

# Trajectory Planning using Parametric Equations.

The End Effector should follow a straight trajectory, as shown by figure 10.1:

|  |
| --- |
|  |
| Figure 11.1: Desired path |

This is achieved by the Parametric Equation shown in Equation 10.1:

|  |  |
| --- | --- |
|  | Eqn 11.1 |

Where:

|  |  |
| --- | --- |
|  | Eqn 11.2 |
|  | Eqn 11.3 |

And *v* is the maximum allowed velocity.

Thus we have and **.**

Further to this, the current position can be evaluated by forward kinematics, and the current velocity can be estimated using Equations 10.4 and 10.5:

|  |  |
| --- | --- |
|  | Eqn 11.4 |
|  | Eqn 11.5 |

# A Simple Position Controller

In order to have a comprehensive series of controllers to compare, it is useful to have a baseline. A useful place to start is with a position controller. The simplest system is shown by the descriptive closed loop block diagram shown in figures 12.1 and 12.2:

|  |
| --- |
|  |
| Figure 12.1: Descriptive Closed Loop Block Diagram for position control |

|  |
| --- |
|  |
| Figure 12.2: Descriptive Closed Loop Block Diagram for position control |

Where the control law can be as simple as:

|  |  |
| --- | --- |
|  | Eqn 12.1 |

Or a little more complicated:

|  |  |
| --- | --- |
|  | Eqn 12.2 |

Clearly this oversimplified position controller will have significant issues. It may become unstable due to interaction forces with the patient. It may also significantly overshoot the target, or indeed fail to reach it.