Appendix

The Matrix Computations of the Transformation Matrices and the Jacobian using MATLAB

Definition of the Variables used in the Computation

```
syms L1 L2 L3 q1 q2 q3 pi_val
```

Obtaining the Intermediate Transformation Matrices using DH-formalism

```
T01 = DHtransffxn(0, 0, 0, q1)
```

T01 =

$$\begin{pmatrix} \cos(q_1) & -\sin(q_1) & 0 & 0 \\ \sin(q_1) & \cos(q_1) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
T12 = subs(DHtransffxn(L1, -pi_val/2, 0, q2), [sin(pi_val/2), cos(pi_val/2)], [1, 0])
```

T12 =

$$\begin{pmatrix} \cos(q_2) & -\sin(q_2) & 0 & L_1 \\ 0 & 0 & 1 & 0 \\ -\sin(q_2) & -\cos(q_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
T23 = DHtransffxn(L2, 0, 0, q3)
```

T23 =

$$\begin{pmatrix} \cos(q_3) & -\sin(q_3) & 0 & L_2 \\ \sin(q_3) & \cos(q_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
T34 = DHtransffxn(L3, 0, 0, 0)
```

T34 =

$$\begin{pmatrix} 1 & 0 & 0 & L_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Obtaining the Transformation Matrices to the Base

```
T02 = T01 * T12
```

T02 =

$$\begin{pmatrix} \cos(q_1)\cos(q_2) & -\cos(q_1)\sin(q_2) & -\sin(q_1) & L_1\cos(q_1) \\ \cos(q_2)\sin(q_1) & -\sin(q_1)\sin(q_2) & \cos(q_1) & L_1\sin(q_1) \\ -\sin(q_2) & -\cos(q_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
T03 = simplify(T01 * T12 * T23)
```

T03 =

$$\begin{pmatrix} \cos(q_2+q_3)\cos(q_1) & -\sin(q_2+q_3)\cos(q_1) & -\sin(q_1) & \cos(q_1)\,\sigma_1 \\ \cos(q_2+q_3)\sin(q_1) & -\sin(q_2+q_3)\sin(q_1) & \cos(q_1) & \sin(q_1)\,\sigma_1 \\ -\sin(q_2+q_3) & -\cos(q_2+q_3) & 0 & -L_2\sin(q_2) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where

$$\sigma_1 = L_1 + L_2\cos(q_2)$$

```
T04 = simplify(T01 * T12 * T23 * T34)
```

T04 =

$$\begin{pmatrix} \cos(q_2+q_3)\cos(q_1) & -\sin(q_2+q_3)\cos(q_1) & -\sin(q_1) & \cos(q_1)\,\sigma_1 \\ \cos(q_2+q_3)\sin(q_1) & -\sin(q_2+q_3)\sin(q_1) & \cos(q_1) & \sin(q_1)\,\sigma_1 \\ -\sin(q_2+q_3) & -\cos(q_2+q_3) & 0 & -L_3\sin(q_2+q_3) - L_2\sin(q_2) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where

$$\sigma_1 = L_1 + L_3\cos(q_2+q_3) + L_2\cos(q_2)$$

Computing the Jacobian Using Indirect Method

The Jacobian was first calculated using Indirect Method because it appears easier and less prone to error.

```
P = T04(1:3, 4)
```

P =

$$\begin{pmatrix} \cos(q_1)\,(L_1 + L_3\cos(q_2+q_3) + L_2\cos(q_2)) \\ \sin(q_1)\,(L_1 + L_3\cos(q_2+q_3) + L_2\cos(q_2)) \\ -L_3\sin(q_2+q_3) - L_2\sin(q_2) \end{pmatrix}$$

```
J_indirect = Jacobian(P, [q1; q2; q3])
```

J_indirect =

$$\begin{pmatrix} -\sin(q_1)\,\sigma_2 & -\cos(q_1)\,\sigma_1 & -L_3\sin(q_2+q_3)\cos(q_1) \\ \cos(q_1)\,\sigma_2 & -\sin(q_1)\,\sigma_1 & -L_3\sin(q_2+q_3)\sin(q_1) \\ 0 & -\sigma_3 - L_2\cos(q_2) & -\sigma_3 \end{pmatrix}$$

where

$$\sigma_1 = L_3\sin(q_2+q_3) + L_2\sin(q_2)$$

$$\sigma_2 = L_1 + \sigma_3 + L_2\cos(q_2)$$

$$\sigma_3 = L_3\cos(q_2+q_3)$$

Obtaining Determinant of the Jacobian

```
JacDet = simplify(det(J_indirect))
```

JacDet = $L_2\,L_3\,\left(L_3\sin(q_2)\cos(q_3)^2 + L_3\cos(q_2)\sin(q_3)\cos(q_3) + L_1\sin(q_3) - L_3\sin(q_2) + L_2\cos(q_2)\sin(q_3)\right)$

```
result = solve(JacDet == 0, [q3, q2], "ReturnConditions",true, 'Real',true)
```

```
result = struct with fields:
            q3: [3×1 sym]
            q2: [3×1 sym]
    parameters: [1×2 sym]
    conditions: [3×1 sym]
```

```
simplify(result.q2)
```

ans =

$$\begin{pmatrix} x \\ x \\ x \end{pmatrix}$$

```
simplify(result.q3)
```

ans =

$$\begin{pmatrix} \pi\,k \\ 2\,\mathrm{atan}\left(\dfrac{\sigma_1 + L_3\sin(x)}{L_1 + L_2\cos(x) - L_3\cos(x)}\right) + 2\,\pi\,k \\ 2\,\pi\,k - 2\,\mathrm{atan}\left(\dfrac{\sigma_1 - L_3\sin(x)}{L_1 + L_2\cos(x) - L_3\cos(x)}\right) \end{pmatrix}$$

where

$$\sigma_1 = \sqrt{-{L_1}^2 - 2\,L_1\,L_2\cos(x) - {L_2}^2\cos(x)^2 + {L_3}^2}$$

```
result.parameters
```

```
ans = (k   x)
```

```
result.conditions
```

ans =

$$\begin{pmatrix} (k \in \mathbb{Z} \land x \in \mathbb{R} \land L_2 \neq 0 \land L_3 \neq 0 \land \sigma_1 < \sigma_2) \lor (k \in \mathbb{Z} \land x \in \mathbb{R} \land L_2 \neq 0 \land L_3 \neq 0 \land \sigma_2 \leq \sigma_1) \\ k \in \mathbb{Z} \land x \in \mathbb{R} \land L_2 \neq 0 \land L_3 \neq 0 \land \sigma_2 \leq \sigma_1 \\ k \in \mathbb{Z} \land x \in \mathbb{R} \land L_2 \neq 0 \land L_3 \neq 0 \land \sigma_2 \leq \sigma_1 \end{pmatrix}$$

where

$$\sigma_1 = L_3^2\ (\cos(x)^2 + \sin(x)^2)$$

$$\sigma_2 = (L_1 + L_2 \cos(x))^2$$

The above solution shows that singularity occurs at q3 = $\pi\ k$ where $k \in \mathbb{Z}$.

Computing the Jacobian Using Direct Method

```
O1 = T01(1:3, 4); R01 = T01(1:3, 1:3);
O2 = T02(1:3, 4); R02 = T02(1:3, 1:3);
O3 = T03(1:3, 4); R03 = T03(1:3, 1:3);
O4 = T04(1:3, 4); R04 = T04(1:3, 1:3);

v1 = SkewSymMat(R01 * [0; 0; 1]) * (O4-O1)
```

v1 =

$$\begin{pmatrix} -\sin(q_1)\ (L_1 + L_3 \cos(q_2 + q_3) + L_2 \cos(q_2)) \\ \cos(q_1)\ (L_1 + L_3 \cos(q_2 + q_3) + L_2 \cos(q_2)) \\ 0 \end{pmatrix}$$

```
v2 = SkewSymMat(R02 * [0; 0; 1]) * (O4-O2)
```

v2 =

$$\begin{pmatrix} -\cos(q_1)\, \sigma_2 \\ -\sin(q_1)\, \sigma_2 \\ \cos(q_1)\ (L_1 \cos(q_1) - \cos(q_1)\, \sigma_1) + \sin(q_1)\ (L_1 \sin(q_1) - \sin(q_1)\, \sigma_1) \end{pmatrix}$$

where

$$\sigma_1 = L_1 + L_3 \cos(q_2 + q_3) + L_2 \cos(q_2)$$

$$\sigma_2 = L_3 \sin(q_2 + q_3) + L_2 \sin(q_2)$$

```
v3 = SkewSymMat(R03 * [0; 0; 1]) * (O4-O3)
```

v3 =

$$\begin{pmatrix} -L_3 \sin(q_2+q_3)\cos(q_1) \\ -L_3 \sin(q_2+q_3)\sin(q_1) \\ \cos(q_1)\ (\cos(q_1)\ (L_1 + L_2\cos(q_2)) - \cos(q_1)\ \sigma_1) + \sin(q_1)\ (\sin(q_1)\ (L_1 + L_2\cos(q_2)) - \sin(q_1)\ \sigma_1) \end{pmatrix}$$

where

$$\sigma_1 = L_1 + L_3\cos(q_2+q_3) + L_2\cos(q_2)$$

```
J_direct = simplify([v1, v2, v3])
```

J_direct =

$$\begin{pmatrix} -\sin(q_1)\ \sigma_2 & -\cos(q_1)\ \sigma_1 & -L_3\sin(q_2+q_3)\cos(q_1) \\ \cos(q_1)\ \sigma_2 & -\sin(q_1)\ \sigma_1 & -L_3\sin(q_2+q_3)\sin(q_1) \\ 0 & -\sigma_3 - L_2\cos(q_2) & -\sigma_3 \end{pmatrix}$$

where

$$\sigma_1 = L_3\sin(q_2+q_3) + L_2\sin(q_2)$$

$$\sigma_2 = L_1 + \sigma_3 + L_2\cos(q_2)$$

$$\sigma_3 = L_3\cos(q_2+q_3)$$

Verify that both DIrect and Indirect Approach are the same

```
J_indirect - J_direct
```

ans =

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

```
function T = DHtransffxn(d, alpha, r, theta) % Function for DH-Formalism
R = [cos(theta), -sin(theta), 0;
    sin(theta)*cos(alpha), cos(theta)*cos(alpha), -sin(alpha);
    sin(theta)*sin(alpha), cos(theta)*sin(alpha), cos(alpha)];
P = [d;
    -r*sin(alpha);
    r*cos(alpha)];

T = [R, P;
    0 0 0 1];
end

function S = SkewSymMat(vec) % Function for Obtaining the Skew Symmetric Matrix of a Vector
x = vec(1); y = vec(2); z = vec(3);
```

```
S = [0, -z, y;
    z, 0, -x;
    -y, x, 0];
end

function Jac = Jacobian(f, x) % Function for computing Jacobian

for f_i=1:size(f, 1)
    for x_i=1:size(x, 1)
        Jac(f_i, x_i) = diff(f(f_i), x(x_i));
    end
end

end
```