Dual Link Manipulator

Linearisation

{Learn, Create, Innovate};

Linearisation of a State Space System

- For a dynamical system of the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u})$$

- We can linearise it (local model to be analysed) about $\{\mathbf{x}^*, \mathbf{u}^*\}$ using a 1^{st} order **multivariate Taylor series****.

$$\Delta\dot{\mathbf{x}} = \frac{\partial \mathbf{f}(\mathbf{x}^*, \mathbf{u}^*)}{\partial \mathbf{x}} \Delta\mathbf{x} + \frac{\partial \mathbf{f}(\mathbf{x}^*, \mathbf{u}^*)}{\partial \mathbf{u}} \Delta\mathbf{u}$$

$$\Delta\mathbf{y} = \frac{\partial \mathbf{h}(\mathbf{x}^*, \mathbf{u}^*)}{\partial \mathbf{x}} \Delta\mathbf{x} + \frac{\partial \mathbf{h}(\mathbf{x}^*, \mathbf{u}^*)}{\partial \mathbf{u}} \Delta\mathbf{u}$$

- The linear state space model can then be described as:

$$\Delta\dot{\mathbf{x}} = \mathbf{A}\,\Delta\mathbf{x} + \boldsymbol{B}\,\Delta\mathbf{u}$$

$$\Delta\mathbf{y} = \mathbf{C}\,\Delta\mathbf{x} + \mathbf{D}\,\Delta\mathbf{u}$$

*** Named after the mathematician Brook Taylor (1685-1731). This is not related to Taylor Swift nor a TV series about Taylor Swift.*

- Where the matrices are defined as follows

$$\boldsymbol{A} = \frac{\partial \mathbf{f}(\mathbf{x}^*, \mathbf{u}^*)}{\partial \mathbf{x}}, \qquad \boldsymbol{B} = \frac{\partial \mathbf{f}(\mathbf{x}^*, \mathbf{u}^*)}{\partial \mathbf{u}}$$

$$\boldsymbol{C} = \frac{\partial \mathbf{h}(\mathbf{x}^*, \mathbf{u}^*)}{\partial \mathbf{x}}, \qquad \boldsymbol{D} = \frac{\partial \mathbf{f}(\mathbf{x}^*, \mathbf{u}^*)}{\partial \mathbf{u}}$$

- The states and control are now incremental

$$\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}^*, \quad \Delta\mathbf{u} = \mathbf{u} - \mathbf{u}^*$$

**Usually the incremental variables are implicit (remove Δ)

SLM Linearisation Example

- The SLM dynamical model is given by

$$\ddot{q} + \frac{mga\,cos(q)}{J + ma^2} = \frac{1}{J + ma^2}\,\tau$$

- Linearising the model using using a 1^{st} order **multivariate Taylor series** around the operating point $\{q^*, \tau^*\}$ gives the same linearized system as previously obtained:

$$\Delta\ddot{q} - \frac{mga\sin(q^*)}{(J + ma^2)}\Delta q = \frac{1}{(J + ma^2)}\Delta\tau$$

- This can be expressed in state space form as

$$\begin{bmatrix}\Delta\dot{q}\\ \Delta\ddot{q}\end{bmatrix} = \begin{bmatrix}0 & 1\\ \frac{mga\sin(q*)}{J+ma^2} & 0\end{bmatrix}\begin{bmatrix}\Delta q\\ \Delta\dot{q}\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{J+ma^2}\end{bmatrix}\Delta\tau$$

SLM Linearisation Example

- The SLM state space dynamical model is given by

$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ (\tau - mga\cos(q))/(J + ma^2) \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}, \quad u = \tau$$

- The linearised model is then given by

$$\frac{\mathrm{d}}{\mathrm{dt}}(\Delta \boldsymbol{q}) = \mathbf{A}\,\Delta \boldsymbol{q} + \boldsymbol{B}\,\Delta\tau$$

$$\mathbf{A} = \begin{bmatrix} \partial f_1/\partial x_1 & \partial f_1/\partial x_2 \\ \partial f_2/\partial x_1 & \partial f_2/\partial x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ mga\sin(q^*)/(J + ma^2) & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \partial f_1/\partial u_1 \\ \partial f_2/\partial u_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1/(J + ma^2) \end{bmatrix}$$

- The same equation is obtained using this method.
- This is called Jacobian.

Definition (Jacobian): Given a vectorial function $f: \mathbb{R}^n \to \mathbb{R}^n$, then the Jacobian matrix $\mathcal{J}_f \in \mathbb{R}^{n\times n}$ is defined by:

$$\mathcal{J}_f = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

Linearisation around an operating point (generalisation)

Let us consider the nonlinear system given by:

$$\dot{x} = f(x, u)$$

$$y = h(x, u)$$

Then if $f(x_0, u_0) = 0$, the point (x_0, u_0) is referred to as an operating point. Under this condition, we can perform a linearisation of the system. Let us define the new input, state, and output as the variation around x_0, u_0, and y_0.

$$\Delta u = u - u_0$$

$$\Delta x = x - x_0$$

$$\Delta y = y - y_0$$

Linearisation around an operating point (generalisation)

Using the Jacobian, it is possible to linearise $f(x,u)$ and $h(x,u)$ around the point (x_0,u_0) and we get:

$$f(x,u) \simeq f(x_0,u_0) + \mathcal{J}_f^x(x_0,u_0)(x-x_0) + \mathcal{J}_f^u(x_0,u_0)(u-u_0)$$

$$h(x,u) \simeq h(x_0,u_0) + \mathcal{J}_h^x(x_0,u_0)(x-x_0) + \mathcal{J}_h^u(x_0,u_0)(u-u_0)$$

where $f(x_0,u_0) = 0$ and $h(x_0,u_0) = y_0$.

The linearised system is given by:

$$\frac{d}{dt}(\Delta x) \simeq \mathcal{J}_f^x(x_0,u_0)\Delta x + \mathcal{J}_f^u(x_0,u_0)\Delta u$$

$$\Delta y \simeq \mathcal{J}_h^x(x_0,u_0)\Delta x + \mathcal{J}_h^u(x_0,u_0)\Delta u$$

where the superscripts x and u in the Jacobian matrices indicate the parameter that is considered as a variable.

- In other words

$$\frac{\mathrm{d}}{\mathrm{dt}}(\Delta\mathbf{x}) = \mathbf{A}\,\Delta\mathbf{x} + \boldsymbol{B}\,\Delta\mathbf{u}$$

$$\boldsymbol{y} = \mathbf{C}\,\Delta\mathbf{x} + \mathbf{D}\,\Delta\mathbf{u}$$

- where

$$\mathbf{A} = \mathcal{J}_f^x(x_0,u_0), \qquad \mathbf{B} = \mathcal{J}_f^u(x_0,u_0)$$

$$\mathbf{C} = \mathcal{J}_h^x(x_0,u_0), \qquad \mathbf{D} = \mathcal{J}_h^u(x_0,u_0)$$

Dual Link Manipulator Model

DLM Manipulator (Model)

- The dual-link manipulator model is given by:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$$

$$\frac{d}{dt}\begin{bmatrix}\mathbf{q} \\ \dot{\mathbf{q}}\end{bmatrix} = \begin{bmatrix}\dot{\mathbf{q}} \\ \mathbf{M}(\mathbf{q})^{-1}(\boldsymbol{\tau} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{g}(\mathbf{q}))\end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix}\mathbf{q} \\ \dot{\mathbf{q}}\end{bmatrix}, \quad \mathbf{u} = \boldsymbol{\tau}$$

where

$$M(\boldsymbol{q}) = \begin{bmatrix} m_1 a^2 + M_1 l_1^2 + m_2(l_1^2 + d^2 + 2 l_1 d C_2) + M_2(l_1^2 + l_2^2 + 2 l_1 l_2 C_2) & m_2 d(l_1 C_2 + d) + M_2 l_2(l_1 C_2 + l_2) \\ m_2 d(l_1 C_2 + d) + M_2 l_2(l_1 C_2 + l_2) & m_2 d^2 + M_2 l_2^2 \end{bmatrix}$$

$$\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} -2 l_1 S_2(m_2 d + M_2 l_2)\dot{q}_2 & -l_1 S_2(m_2 d + M_2 l_2)\dot{q}_2 \\ l_1 S_2(m_2 d + M_2 l_2)\dot{q}_1 & 0 \end{bmatrix}$$

$$\mathbf{g}(\mathbf{q}) = g\begin{bmatrix} m_1 a C_1 + M_1 l_1 C_1 + m_2(l_1 C_1 + d C_{12}) + M_2(l_1 C_1 + l_2 C_{12}) \\ C_{12}(m_2 d + M_2 l_2) \end{bmatrix}$$

$$\boldsymbol{\tau} = \begin{bmatrix}\tau_1 \\ \tau_2\end{bmatrix}$$

Dual Link Manipulator Model

DLM Manipulator (Model)

- Expanding the model in MATLAB:

$$\frac{d}{dt}\begin{bmatrix}\mathbf{q}\\ \dot{\mathbf{q}}\end{bmatrix} = \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dfrac{\left(m_2\, d^2 + M_2\, l_2^2\right)\sigma_3}{\sigma_1} + \dfrac{\sigma_4\,\sigma_2}{\sigma_1} \\ -\dfrac{\sigma_2\,\sigma_3}{\sigma_1} - \dfrac{\sigma_4\,\left(M_1\, l_1^2 + M_2\, l_1^2 + M_2\, l_2^2 + a^2\, m_1 + d^2\, m_2 + l_1^2\, m_2 + 2\, M_2\, l_1\, l_2 \cos(q_2) + 2\, d\, l_1\, m_2 \cos(q_2)\right)}{\sigma_1} \end{pmatrix}$$

where

$$\sigma_1 = -M_2^2\, l_1^2\, l_2^2 \cos(q_2)^2 + M_2^2\, l_1^2\, l_2^2 + m_1\, M_2\, a^2\, l_2^2 + M_2\, d^2\, l_1^2\, m_2 - 2\, M_2\, d\, l_1^2\, l_2\, m_2 \cos(q_2)^2 + M_2\, l_1^2\, l_2^2\, m_2 + M_1\, M_2\, l_1^2\, l_2^2 + m_1\, a^2\, d^2\, m_2 - d^2\, l_1^2\, m_2^2 \cos(q_2)^2 + d^2\, l_1^2\, m_2^2 + M_1\, d^2\, l_1^2\, m_2$$

$$\sigma_2 = m_2\, d^2 + l_1\, m_2 \cos(q_2)\, d + M_2\, l_2^2 + M_2\, l_1 \cos(q_2)\, l_2$$

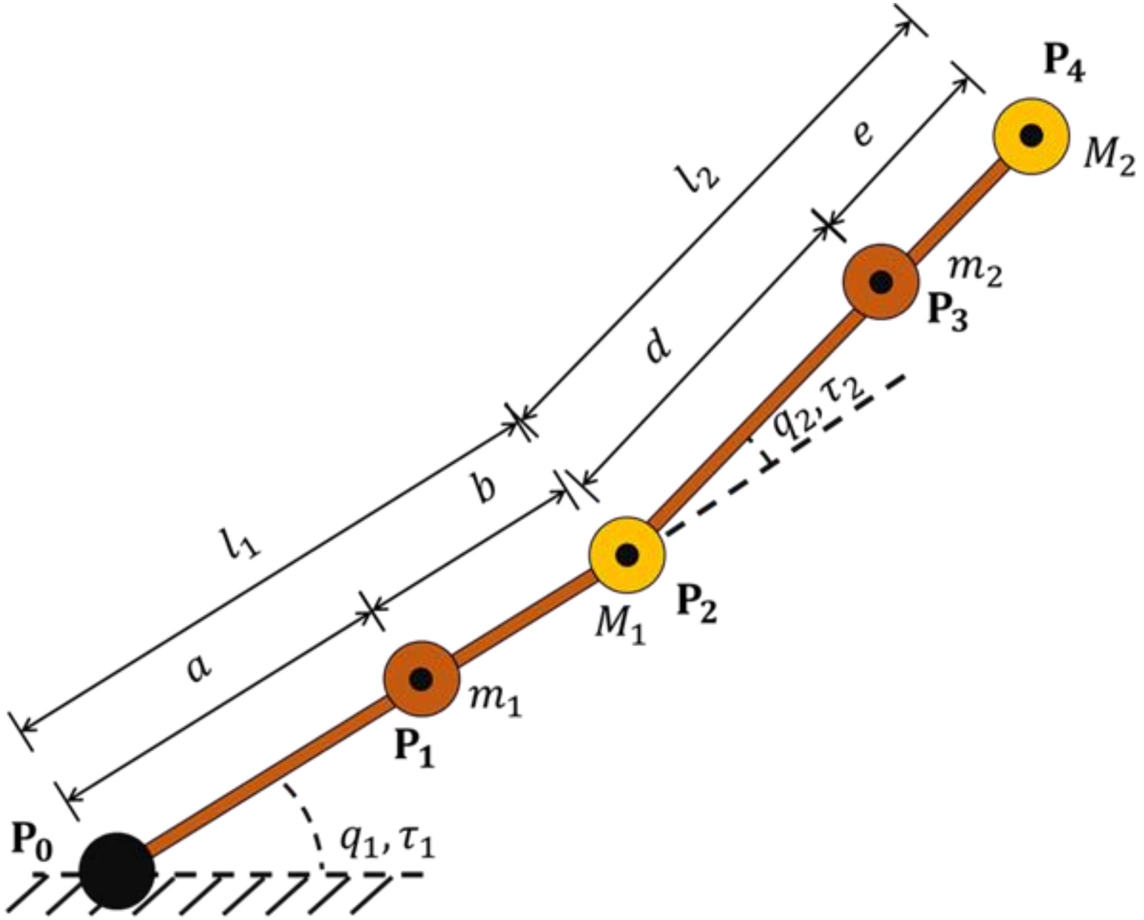
$$\sigma_3 = l_1 \sin(q_2)\, \sigma_5\, \dot{q}_2^{\,2} + 2\, l_1\, \dot{q}_1 \sin(q_2)\, \sigma_5\, \dot{q}_2 + \tau_1 - g\, \left(M_2\, \left(l_2 \cos(q_1 + q_2) + l_1 \cos(q_1)\right) + m_2\, \left(d \cos(q_1 + q_2) + l_1 \cos(q_1)\right) + a\, m_1 \cos(q_1) + M_1\, l_1 \cos(q_1)\right)$$

$$\sigma_4 = l_1 \sin(q_2)\, \sigma_5\, \dot{q}_1^{\,2} - \tau_2 + g \cos(q_1 + q_2)\, \sigma_5$$

$$\sigma_5 = M_2\, l_2 + d\, m_2$$

DLM Linearisation Example

- Let the parameters of the DLM to be:
  - $m1 = 3\ kg$
  - $m2 = 3\ kg$
  - $M1 = 1.5\ kg$
  - $M2 = 1\ kg$
  - $a = 0.2\ m$
  - $b = 0.2\ m$
  - $d = 0.2\ m$
  - $e = 0.2\ m$
  - $g = 9.8\ ms^{-2}$



DLM Nonlinear Model

- Substituting the values in the model using MATLAB:

$$\frac{d}{dt}\begin{bmatrix} q_1 \\ q_2 \\ \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ -\frac{7\,\sigma_1}{25\,\sigma_3} - \frac{\sigma_4\,\sigma_2}{\sigma_3} \\ \frac{25\,\sigma_4\,\sigma_1}{4\cos(q_2)^2 - 7} + \frac{25\left(\frac{4\cos(q_2)}{5} + \frac{32}{25}\right)\sigma_2}{4\cos(q_2)^2 - 7} \end{pmatrix}$$

where

$$\sigma_1 = \frac{2\sin(q_2)\,\dot{q}_2^{\,2}}{5} + \frac{4\,\dot{q}_1\sin(q_2)\,\dot{q}_2}{5} + \tau_1 - \frac{49\cos(q_1+q_2)}{5} - \frac{686\cos(q_1)}{25}$$

$$\sigma_2 = \frac{2\sin(q_2)\,\dot{q}_1^{\,2}}{5} - \tau_2 + \frac{49\cos(q_1+q_2)}{5}$$

$$\sigma_3 = \frac{4\cos(q_2)^2}{25} - \frac{7}{25}$$

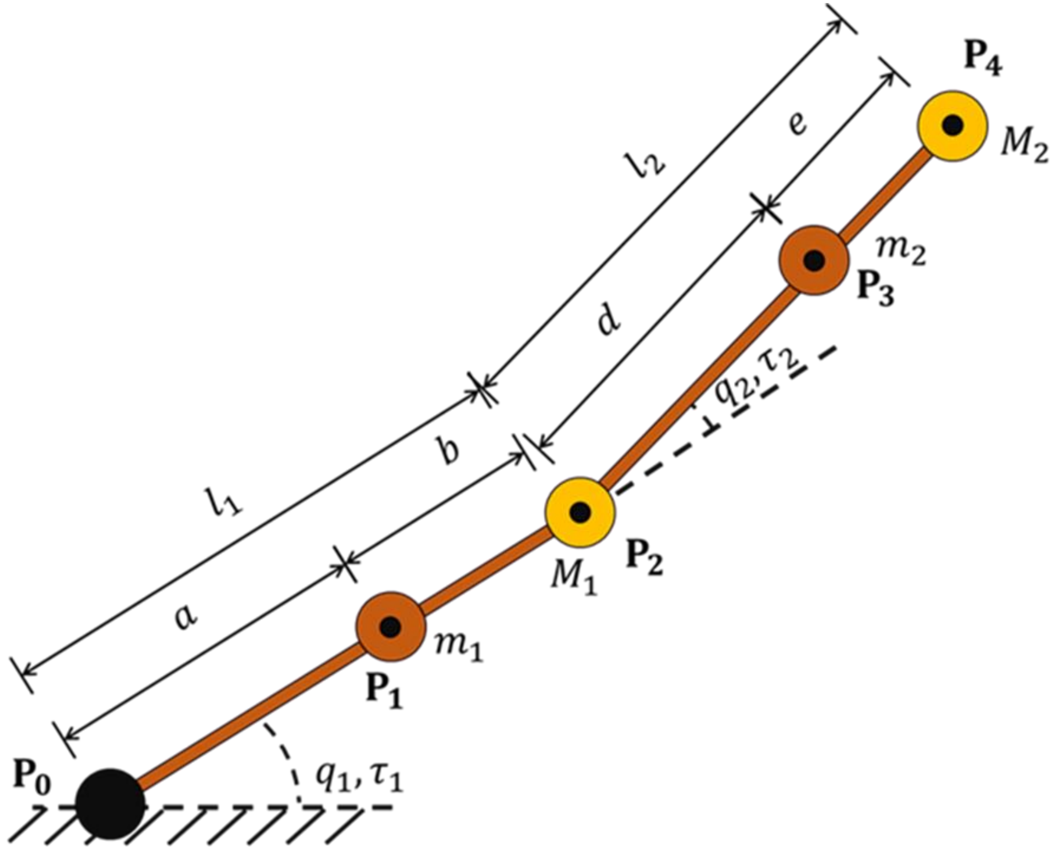
$$\sigma_4 = \frac{2\cos(q_2)}{5} + \frac{7}{25}$$

While the non-linear, state space equations appear complex there are a few observations to make:

- The denominator is the determinant of the mass-inertia matrix, always positive and always invertible

DLM Linearisation Example

- Having obtained the DLM Nonlinear model, it is time now to linearise it around an operating point.
- This can be done using the concept of the Jacobian.
- In this case, we will use MATLAB symbolic toolbox to make the linearisation easier.



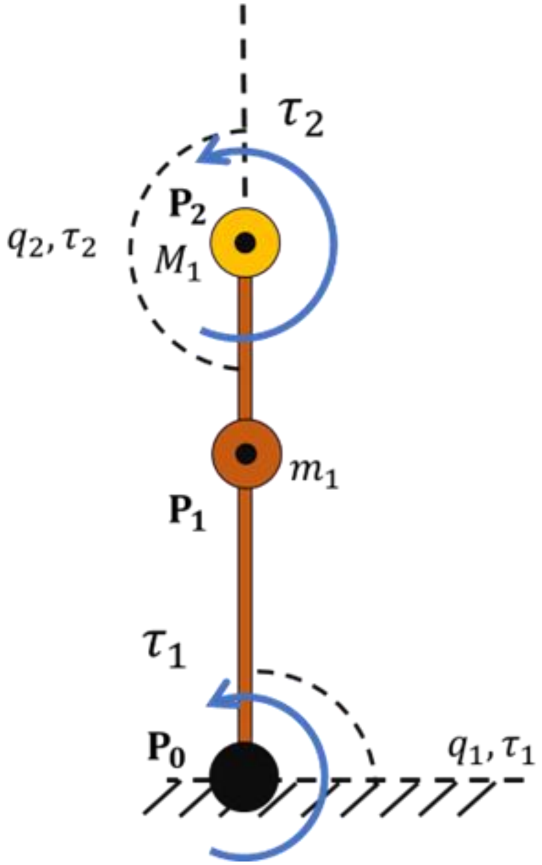
DLM Linearisation Example

- For this example, the linearisation of the Dual Link Manipulator will be about the static upright position:

$$\mathbf{q}^* = \left[\frac{\pi}{2}, \pi, 0, 0\right]^T$$

$$\boldsymbol{\tau}^* = [0,0]^T$$

- This is in order to analyse the dynamics and design linear feedback controllers.

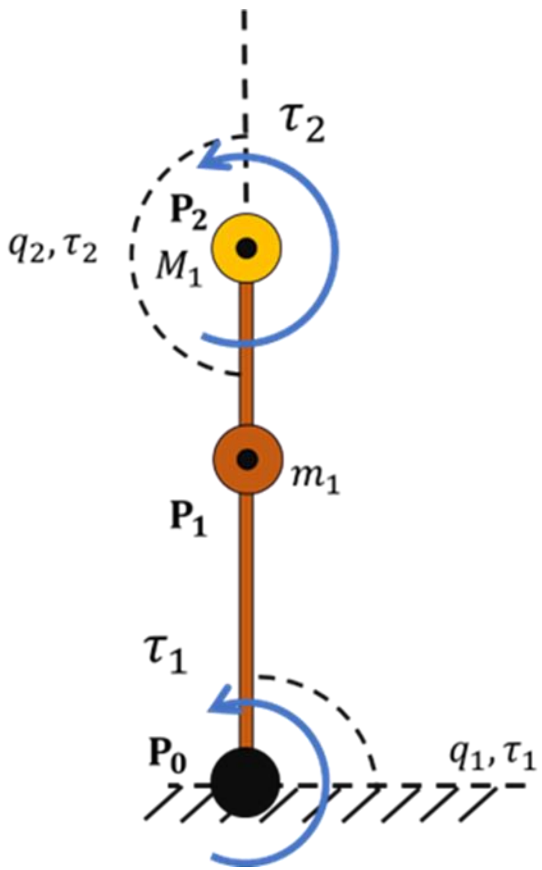


DLM Linearisation Example

- In general, using linear feedback methods (PID) to perform joint control i.e., $\mathbf{r} \rightarrow \mathbf{q}$ for a manipulator or a humanoid robot, isn't optimal, as the dynamics are inherently non-linear

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau}$$

- However, we observed that for a single link manipulator, a PID controller could give acceptable performance for all orientations.



DLM Linearisation Example

- The linear system of a DLM will have the following form

$$\Delta\dot{\boldsymbol{q}} = \mathbf{A}\,\Delta\mathbf{q} + \boldsymbol{B}\,\Delta\boldsymbol{\tau}$$

$$\Delta\mathbf{y} = \mathbf{C}\,\Delta\mathbf{q} + \mathbf{D}\,\Delta\boldsymbol{\tau}$$

- Using the Jacobian, to linearise the system and substituting the values we obtain the $\mathbf{A}$ Matrix:

$$\mathbf{A}(\mathbf{q}^*, \dot{\mathbf{q}}^*, \boldsymbol{\tau}^*) =$$

$$\begin{pmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\frac{49\,(25\sin(q_1+2\,q_2) - 73\sin(q_1))}{5\,(10\cos(2\,q_2) - 25)} & \frac{280\,\dot{q}_1^{\,2}\cos(q_2)^3 + 1000\,\dot{q}_1^{\,2}\cos(q_2)^2 - 70\,\dot{q}_1^{\,2}\cos(q_2) - 700\,\dot{q}_1^{\,2} + 560\,\dot{q}_1\,\dot{q}_2\cos(q_2)^3 - 140\,\dot{q}_1\,\dot{q}_2\cos(q_2) + 280\,\dot{q}_2^{\,2}\cos(q_2)^3 - 70\,\dot{q}_2^{\,2}\cos(q_2) - 24500\sin(q_1)\cos(q_2)^2 + 4116\cos(q_1)\cos(q_2)\sin(q_2) - 1000\,\tau_2\sin(q_2)^3 + 2750\,\tau_2\sin(q_2) + 17150\sin(q_1) - 700\,\tau_1\sin(2\,q_2) + 700\,\tau_2\sin(2\,q_2)}{25\,(\sigma_7 - 7)^2} & -\frac{4\sin(q_2)\left(\frac{7\,\dot{q}_1}{25} + \frac{7\,\dot{q}_2}{25} + \frac{2\,\dot{q}_1\cos(q_2)}{5}\right)}{\sigma_6} & \frac{\sin(q_2)\,(28\,\dot{q}_1 + 28\,\dot{q}_2)}{20\sin(q_2)^2 + 15} \\
\frac{49\left(\sin(q_1+q_2) + \frac{14\sin(q_1)}{5}\right)\sigma_1}{\sigma_6} - \frac{49\sin(q_1+q_2)\,\sigma_2}{\sigma_6} & \frac{25\,\sigma_1\left(\frac{2\cos(q_2)\,\dot{q}_2^{\,2}}{5} + \frac{4\,\dot{q}_1\cos(q_2)\,\dot{q}_2}{5} + \sigma_4\right)}{\sigma_7 - 7} - \frac{25\left(\sigma_4 - \frac{2\,\dot{q}_1^{\,2}\cos(q_2)}{5}\right)\sigma_2}{\sigma_7 - 7} - \frac{2\sin(q_2)\,\left(10\sin(q_2)\,\dot{q}_2^{\,2} + 20\,\dot{q}_1\sin(q_2)\,\dot{q}_2 + 25\,\tau_1 - 245\cos(q_1+q_2) - 686\cos(q_1)\right)}{20\cos(q_2)^2 - 35} - \frac{4\sin(q_2)\,(\sigma_3 - 5\,\tau_2 + \sigma_5)}{\sigma_7 - 7} + \frac{100\sin(2\,q_2)\,\sigma_1\left(\frac{2\sin(q_2)\,\dot{q}_2^{\,2}}{5} + \frac{4\,\dot{q}_1\sin(q_2)\,\dot{q}_2}{5} + \tau_1 - \frac{\sigma_5}{5} - \frac{686\cos(q_1)}{25}\right)}{(\sigma_7 - 7)^2} + \frac{100\sin(2\,q_2)\,\sigma_2\left(\frac{\sigma_3}{5} - \tau_2 + \frac{\sigma_5}{5}\right)}{(\sigma_7 - 7)^2} & \frac{4\sin(q_2)\left(\frac{32\,\dot{q}_1}{25} + \frac{7\,\dot{q}_2}{25} + \frac{4\,\dot{q}_1\cos(q_2)}{5} + \frac{2\,\dot{q}_2\cos(q_2)}{5}\right)}{\sigma_6} & \frac{\sin(q_2)\,(20\,\dot{q}_1 + 20\,\dot{q}_2)\,\sigma_1}{\sigma_7 - 7}
\end{pmatrix}$$

where

$$\sigma_1 = \frac{2\cos(q_2)}{5} + \frac{7}{25}$$

$$\sigma_2 = \frac{4\cos(q_2)}{5} + \frac{32}{25}$$

$$\sigma_3 = 2\sin(q_2)\,\dot{q}_1^{\,2}$$

$$\sigma_4 = \frac{49\sin(q_1+q_2)}{5}$$

$$\sigma_5 = 49\cos(q_1+q_2)$$

$$\sigma_6 = 5\left(\frac{\sigma_7}{25} - \frac{7}{25}\right)$$

$$\sigma_7 = 4\cos(q_2)^2$$

- Verifying the odd entry is possible
- Computer algebra was the right choice …
- This is just for two links …

DLM Linearisation Example

- The $\mathbf{B}$ Matrix is obtained using MATLAB as follows

$$\mathbf{B}(\mathbf{q}^*) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ -\dfrac{7}{4\cos(q_2)^2 - 7} & \dfrac{10\cos(q_2) + 7}{4\cos(q_2)^2 - 7} \\ \dfrac{10\cos(q_2) + 7}{4\cos(q_2)^2 - 7} & -\dfrac{20\cos(q_2) + 32}{4\cos(q_2)^2 - 7} \end{pmatrix}$$

- Only depends on the relative joint angle q_2, as the lower submatrix is simply the inverse of the mass–inertia matrix
- The torque signals enter the state space equations multiplied by the inverse mass/inertia matrix.
- The original non-linear dynamics are affine: $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}$
- The output of the system is q_1 and q_2, therefore the matrix $\mathbf{C}$ can be defined as follows

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

DLM Linearisation Example

- Linearising around the previously defined operating point:

$$\mathbf{q}^* = \left[\frac{\pi}{2}, \pi, 0, 0\right]^T$$

$$\boldsymbol{\tau}^* = [0,0]^T$$

- The model then becomes (MIMO):

$$\Delta\dot{\boldsymbol{q}} = \mathbf{A}\,\Delta\mathbf{q} + \boldsymbol{B}\,\Delta\boldsymbol{\tau}$$

$$\Delta\mathbf{y} = \mathbf{C}\,\Delta\mathbf{q} + \mathbf{D}\,\Delta\boldsymbol{\tau}$$

where

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{784}{25} & -\frac{98}{3} & 0 & 0 \\ -\frac{539}{25} & -49 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{7}{3} & 1 \\ 1 & 4 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Linearised MIMO system

- Converting from SS to TF using MATLAB it is possible to obtain the MIMO model of the system.

$$G_1 = \frac{2.33s^2 + 81.67}{s^4 + 17.64s^2 - 2241}$$

$$G_2 = \frac{s^2 + 81.67}{s^4 + 17.64s^2 - 2241}$$

$$G_3 = \frac{s^2 - 81.67}{s^4 + 17.64s^2 - 2241}$$

$$G_4 = \frac{4s^2 - 147}{s^4 + 17.64s^2 - 2241}$$

- The matrix described the interactions of the inputs and outputs of the system.

$$\begin{bmatrix} q_1(s) \\ q_2(s) \end{bmatrix} = \begin{bmatrix} \frac{2.33s^2 + 81.67}{s^4 + 17.64s^2 - 2241} & \frac{s^2 - 81.67}{s^4 + 17.64s^2 - 2241} \\ \frac{s^2 + 81.67}{s^4 + 17.64s^2 - 2241} & \frac{4s^2 - 147}{s^4 + 17.64s^2 - 2241} \end{bmatrix} \begin{bmatrix} \tau_1(s) \\ \tau_2(s) \end{bmatrix}$$

DLM Linearisation Example

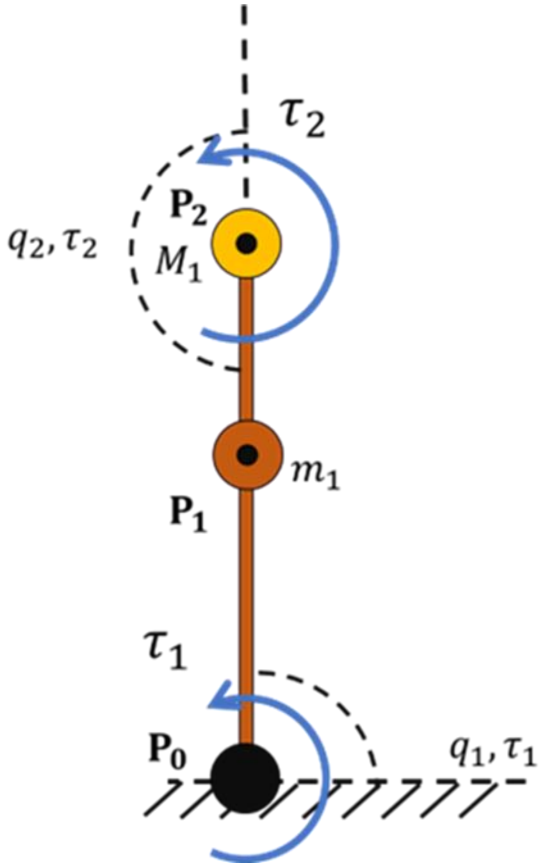
- Analysing the poles of the linearised system
- This corresponds to analysing the eigenvalues of the matrix $\mathbf{A}$.

Link 1   Link 2

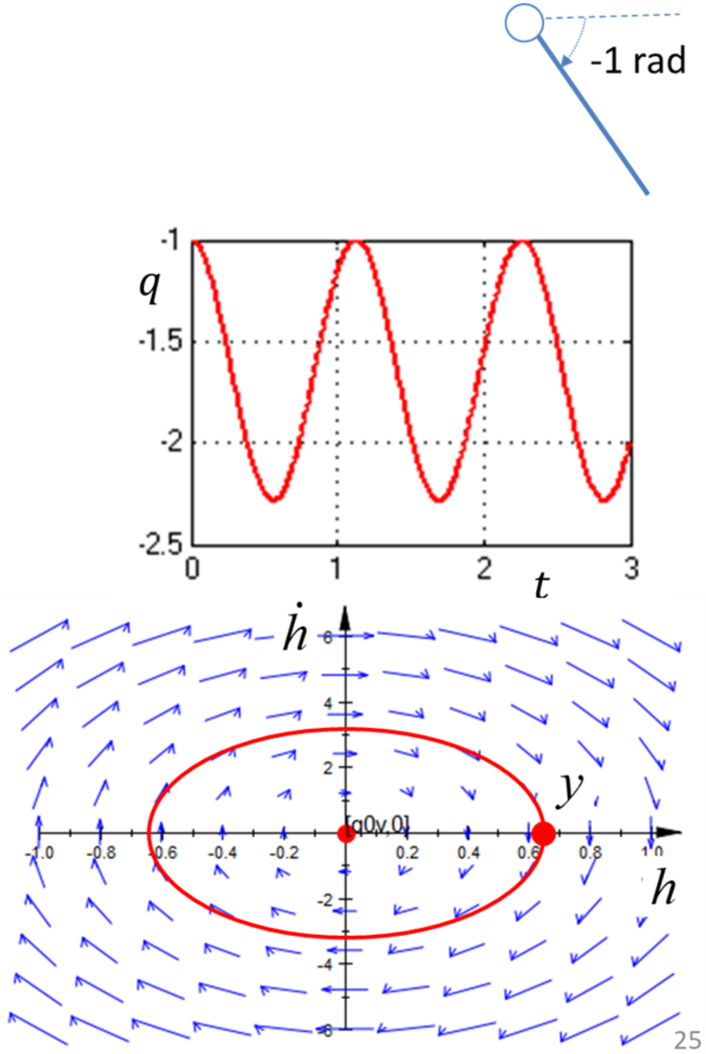
$$\lambda(\mathbf{A}) = \{-6.2716, \quad 6.2716, \quad 7.54i, \quad -7.54i\}$$

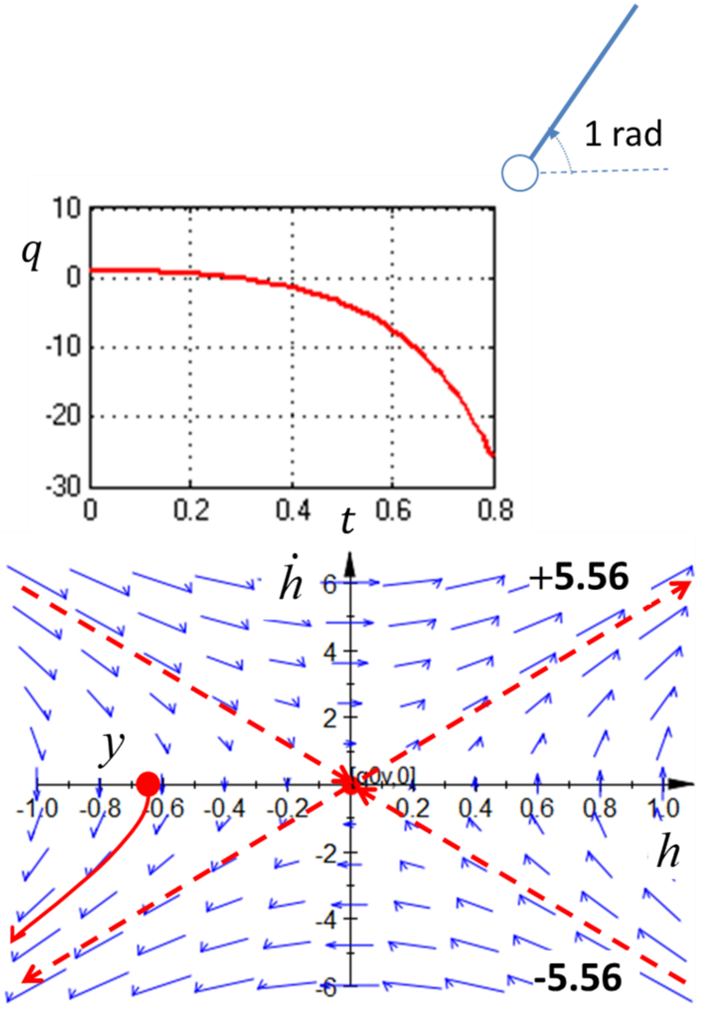
$$\tau = 0.15 \;, \omega = 7.55$$

- Analysing the poles it can be observed:
- Oscillatory behaviour for the Link 2 (SLM in the downright position)
- Unstable for the link 1 (SLM in the upright position)



DLM Linearisation Example



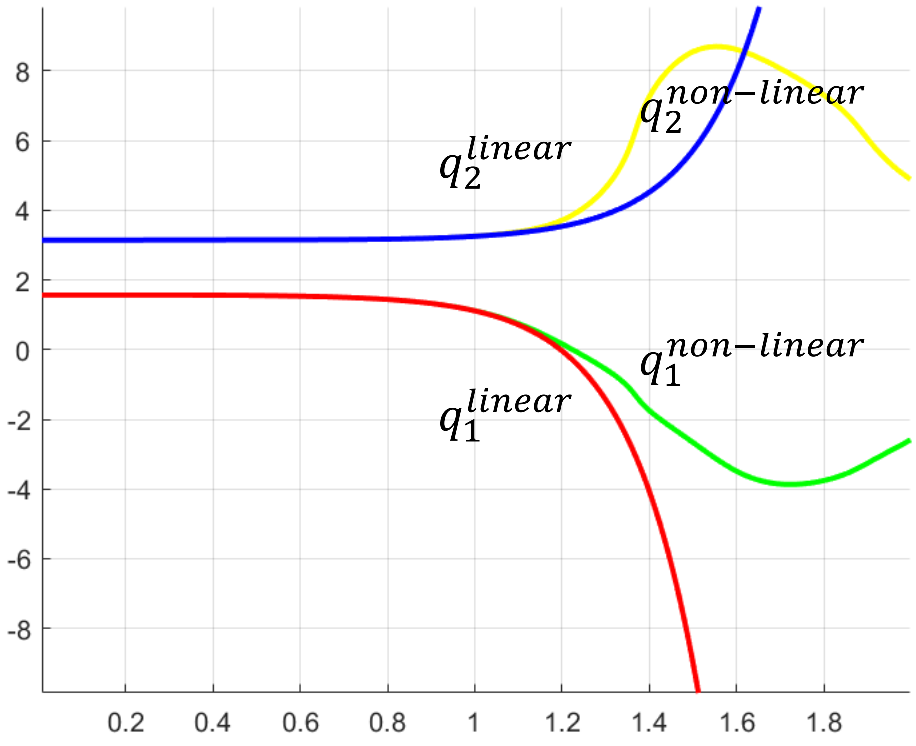


Linearized v Non-linear Comparison Simulation

Compare a linearized and a non-linear (unforced) simulation in MATLAB.

- Linearization point $\{\boldsymbol{q}^*, \boldsymbol{\tau}^*\}=\{[\pi/2,\pi,0,0], [0,0]\}$
- Initial conditions
  - Non-linear: $\mathbf{q}(0)=[\pi/2-0.003,\pi+0.003,0,0]$
  - Linear: $\Delta\mathbf{q}(0)=[-0.003,0.003,0,0]$
- As can be seen the unforced, non-linear plant behaves like before, the links begin to **circle** around in an **unpredictable fashion**
- The linearized model initially agrees quite well (for between ½ or 1 s) and then **exponentially diverges**, both the **states and errors**.
- This should be expected for an **unstable**, linear model

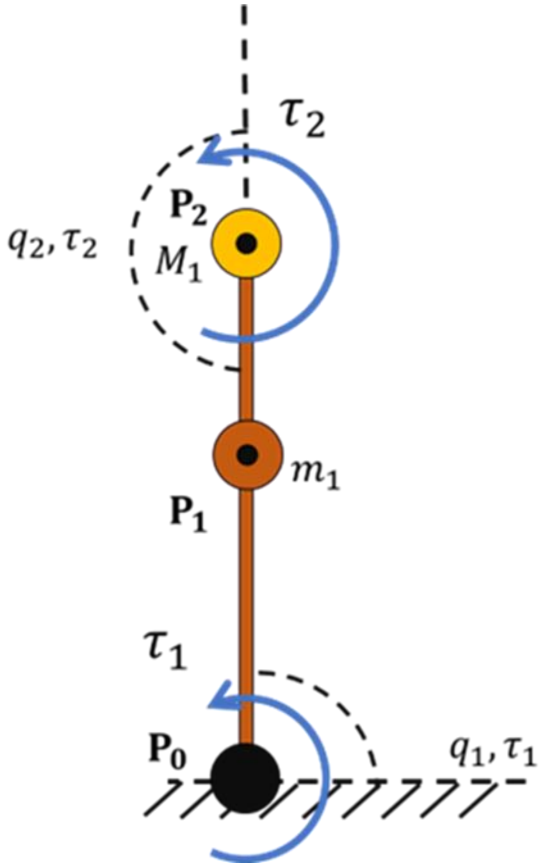
The evolution of the linear and nonlinear joint angles and the corresponding errors are shown below



Joint Angle Evolution

Conclusions

- The linearized model is a good approximation to the non-linear model for the DLM and can be used for linear, feedback control (PID) design
- One key insight is that the eigenvalues consist of an stable/unstable pair (Link 1) and an oscillatory pair (Link 2).
- In subsequent section, design a linear (PID) joint controller for each joint.



Thank you

{Learn, Create, Innovate};

MCR²

Manchester **Robotics**

T&C

Terms and conditions

{Learn, Create, Innovate};

Manchester **Robotics**

Terms and conditions

- *THE PIECES, IMAGES, VIDEOS, DOCUMENTATION, ETC. SHOWN HERE ARE FOR INFORMATIVE PURPOSES ONLY. THE DESIGN IS PROPRIETARY AND CONFIDENTIAL TO MANCHESTER ROBOTICS LTD. (MCR2). THE INFORMATION, CODE, SIMULATORS, DRAWINGS, VIDEOS PRESENTATIONS ETC. CONTAINED IN THIS PRESENTATION IS THE SOLE PROPERTY OF MANCHESTER ROBOTICS LTD. ANY REPRODUCTION, RESELL, REDISTRIBUTION OR USAGE IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF MANCHESTER ROBOTICS LTD. IS STRICTLY PROHIBITED.*
- *THIS PRESENTATION MAY CONTAIN LINKS TO OTHER WEBSITES OR CONTENT BELONGING TO OR ORIGINATING FROM THIRD PARTIES OR LINKS TO WEBSITES AND FEATURES IN BANNERS OR OTHER ADVERTISING. SUCH EXTERNAL LINKS ARE NOT INVESTIGATED, MONITORED, OR CHECKED FOR ACCURACY, ADEQUACY, VALIDITY, RELIABILITY, AVAILABILITY OR COMPLETENESS BY US.*
- *WE DO NOT WARRANT, ENDORSE, GUARANTEE, OR ASSUME RESPONSIBILITY FOR THE ACCURACY OR RELIABILITY OF ANY INFORMATION OFFERED BY THIRD-PARTY WEBSITES LINKED THROUGH THE SITE OR ANY WEBSITE OR FEATURE LINKED IN ANY BANNER OR OTHER ADVERTISING.*