Lecture Notes
Multibody Dynamics B, wb1413

A. L. Schwab & Guido M.J. Delhaes

Laboratory for Engineering Mechanics
Mechanical Engineering
Delft University of Technolgy
The Netherlands

April 28, 2011

Contents

Chapter 8

Closed Loop Systems

Finding $\mathbf{x}_i = \mathbf{F}_i(\mathbf{q}_j)$ for closed loops is <u>not</u> easy. Lets look at for instance at a four-bar linkage. $\mathbf{x}_i = (x_1, y_1, \phi_1, x_2, y_2, \phi_2, x_3, y_3, \phi_3)$ and $\mathbf{q}_i = (\alpha)$.

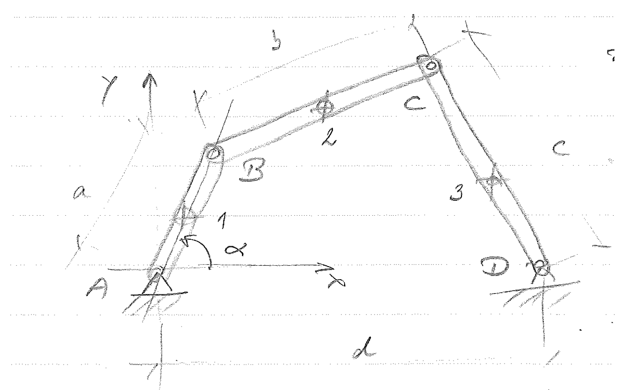


Figure 8.1: Four-bar linkage system

Why four-bar?
So we have to write down $\mathbf{x}_i = \mathbf{F}_i(\mathbf{q}_j)$.
Lets start,look at the figure above:

$$\begin{aligned} x_1 &= a/2\cos(\alpha) \\ y_1 &= a/2\sin(\alpha) \\ \phi_1 &= \alpha \\ x_2 &= a\cos(\alpha) + \cdots \end{aligned}$$

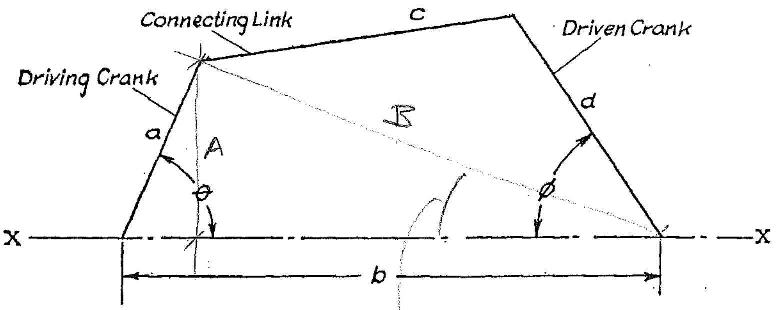
I do not know! Look at the paper by Talbourdet from 1941 [1].

Part I—Analysis of Single 4-Bar Linkage

By Guy J. Talbourdet

Research Division

United Shoe Machinery Corporation



Calculate the values of the following factors from the geometry of the linkage:

$$K = a^2 + b^2 - c^2 + d^2$$

$$A = a \, Sin \, \theta$$

$$B = a^2 + b^2 - 2ab \, Cos \, \theta$$

$$D = K - 2ab \, Cos \, \theta$$

$$S = \sqrt{4d^2B - D^2}$$

θ = Angular displacement of driving crank.

ϕ = Angular displacement of driven crank.

$\frac{d\theta}{dt}$ and $\frac{d^2\theta}{dt^2}$ = Angular velocity and acceleration respectively of the driving crank.

$\frac{d\phi}{dt}$ and $\frac{d^2\phi}{dt^2}$ = Angular velocity and acceleration respectively of the driven crank.

Then:

$$\phi = Tan^{-1} \frac{A}{b - a \, Cos \, \theta} + Cos^{-1} \frac{D}{2d\sqrt{B}}$$

And:

$$\frac{d\phi}{dt} = \frac{d\theta}{dt}\left[\frac{a}{B}(b \, Cos \, \theta - a) - \frac{Ab}{S}\left(2 - \frac{D}{B}\right)\right]$$

And:

$$\frac{d^2\phi}{dt^2} = \frac{d^2\theta}{dt^2}\left[\frac{a}{B}(b \, Cos \, \theta - a) - \frac{Ab}{S}\left(2 - \frac{D}{B}\right)\right]$$

$$+ \left(\frac{d\theta}{dt}\right)^2\left[\frac{2A^2b^2}{BS}\left(1 - \frac{D}{B}\right) + \left(2 - \frac{D}{B}\right)\left(\frac{2A^2b^2(2d^2 - D)}{S^3} - \frac{ab \, Cos \, \theta}{S}\right)\right.$$

$$\left. - \frac{Ab}{B}\left(1 + \frac{2a(b \, Cos \, \theta - a)}{B}\right)\right]$$

When the motion of the driving crank is uniform, $\frac{d\theta}{dt} = \omega$, and $\frac{d^2\theta}{dt^2} = 0$, in which case we have:

$$\frac{d\phi}{dt} = \omega\left[\frac{a}{b}(b \, Cos \, \theta - a) - \frac{Ab}{S}\left(2 - \frac{D}{B}\right)\right]$$

And:

$$\frac{d^2\phi}{dt^2} = \omega^2\left[\frac{2A^2b^2}{BS}\left(1 - \frac{D}{B}\right) + \left(2 - \frac{D}{B}\right)\left(\frac{2A^2b^2(2d^2 - D)}{S^3} - \frac{ab \, Cos \, \theta}{S}\right)\right.$$

$$\left. - \frac{Ab}{B}\left(1 + \frac{2a(b \, Cos \, \theta - a)}{B}\right)\right]$$

But I do know how to do a triple-pendulum. This is a double pendulum with an extra pendulum at the end and the I just add two constraints to get back to the original 1 dof system.

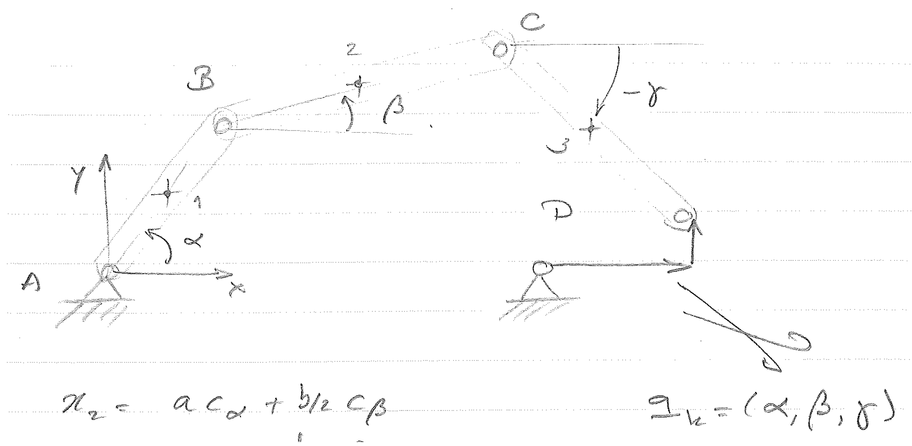


Figure 8.2: four-bar linkage generalized coordinate definition

So cut the loop at D and add two generalized coordinates β and γ. Next write down the positions and orientations of the rigid bodies in terms of the generalized coordinates, $\mathbf{x} = \mathbf{F}(\mathbf{q})$, where $\mathbf{q} = (\alpha, \beta, \gamma)$:

$$\begin{aligned}
x_2 &= a\cos(\alpha) + \tfrac{b}{2}\cos(\beta)\\
y_2 &= a\sin(\alpha) + \tfrac{b}{2}\sin(\beta)\\
\phi_2 &= \beta\\
x_3 &= a\cos(\alpha) + b\cos(\beta) + \tfrac{c}{2}\cos(\gamma)\\
y_3 &= a\sin(\alpha) + b\sin(\beta) + \tfrac{c}{2}\sin(\gamma)\\
\phi_3 &= \gamma
\end{aligned}$$

Now add two constraints to close the loop again at D,

$$\begin{array}{lll} \epsilon_1 & = \Delta x_D & = 0 \\ \epsilon_2 & = \Delta y_D & = 0 \end{array} \Rightarrow \begin{array}{lll} \epsilon_1 & = a\cos(\alpha) + b\cos(\beta) c\cos(\gamma) - d & = 0 \\ \epsilon_2 & = a\sin(\alpha) + b\sin(\beta) c\sin(\gamma) & = 0 \end{array}$$

And finally we can form the DAE for this problem as,

$$\begin{array}{l} 3\{ \\ 2\{ \end{array} \begin{bmatrix} \mathbf{F}_{i,l}\mathbf{M}_{ij} & \mathbf{D}_{c,l} \\ \mathbf{D}_{c,k} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}}_k \\ \boldsymbol{\lambda}_c \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_L + \mathbf{F}_{i,l}(\mathbf{f}_i - \mathbf{M}_{ij}\mathbf{g}_j) \\ -\mathbf{D}_{c,kl}\dot{\mathbf{q}}_k\dot{\mathbf{q}}_l \end{bmatrix}$$

From this we solve for $\ddot{\mathbf{q}}_k$ and $\boldsymbol{\lambda}_c$ and then we integrate the state $\begin{bmatrix} \mathbf{q}_k \\ \dot{\mathbf{q}}_k \end{bmatrix}$ like in:

$$\begin{bmatrix} \tilde{\mathbf{q}}_k \\ \tilde{\dot{\mathbf{q}}}_k \end{bmatrix} = \begin{bmatrix} \mathbf{q}_k \\ \dot{\mathbf{q}}_k \end{bmatrix} + \int_0^h \begin{bmatrix} \dot{\mathbf{q}}_k \\ \ddot{\mathbf{q}}_k \end{bmatrix} \mathrm{d}t$$

The approximate values at $t+h$ $\tilde{\mathbf{q}}_{n+1}$ and $\tilde{\dot{\mathbf{q}}}_{n+1}$ will in general NOT fulfill the constraints. Remember the results from the 2^{nd} lecture things fly apart, see Figure 8.3. One can picture the constraints as a sort of surface in a higher dimensional space, where a state $\mathbf{q}$ is represented by points in that space, see

Figure 8.3: flying apart of the two bars of the pendulum

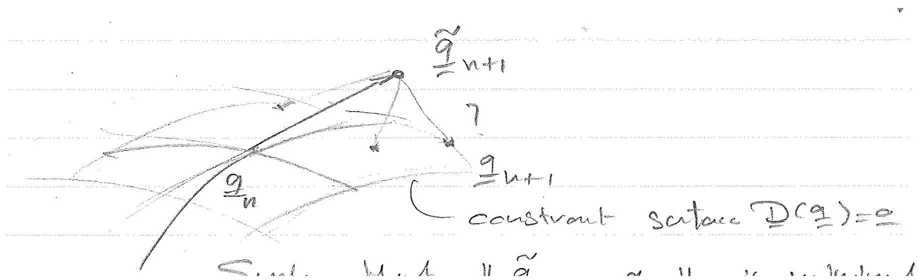


Figure 8.4: constraint surface

Figure 8.4. The constraint surface has the form $\mathbf{D}(\mathbf{q}) = \mathbf{0}$. Now a predicted solution $\tilde{\mathbf{q}}_{n+1}$ will in general not be on the constraint surface. We have to find a way to get back on the surface with minimal effort. Lets formulate this as a minimization problem such that the distance from the predicted solution $\tilde{\mathbf{q}}_{n+1}$ to the solution which is on the constraint surface is minimal: $||\tilde{\mathbf{q}}_{n+1} - \mathbf{q}_{n+1}||_2$ is minimal where all $\mathbf{q}_{n+1}$ have to fulfill the constraints $\mathbf{D}(\mathbf{q}_{n+1}) = \mathbf{0}$

This is what we call a non-linear constrained least-square problem,

$$\begin{array}{l} ||\tilde{\mathbf{q}}_{n+1} - \mathbf{q}_{n+1}||_2 = \min_{\mathbf{q}_{n+1}} \\ \mathbf{D}(\mathbf{q}_{n+1}) = \mathbf{0} \end{array}$$

We solve this by a Gauss-Newton method: First linearize about $\tilde{\mathbf{q}}_{n+1}$.

$$\mathrm{q}_{n+1} = \tilde{\mathrm{q}}_{n+1} + \Delta\mathbf{q}_{n+1}$$

Which leads to:

$$\begin{array}{ll} ||\Delta\mathbf{q}_{n+1}||_2 = min & \Sigma_i \left(\Delta q_i^2\right)_{n+1} = min \\ \mathbf{D}(\tilde{\mathbf{q}}_{n+1}) + \mathbf{D}_{,\mathbf{n}}(\tilde{\mathbf{q}}_{n+1})\,\Delta\mathbf{q}_{n+1} = \mathbf{0} & \end{array}$$

This constrained least square problem can be solved by introducing the so-called Lagrange multipliers $\boldsymbol{\mu}$ for the constraints leading to the linear system of equations,

$$\begin{bmatrix} \mathbf{I} & \mathbf{D}_{,\mathbf{q}}^T \\ \mathbf{D}_{,\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{q}_{n+1} \\ \boldsymbol{\mu} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ -\mathbf{D}(\tilde{\mathbf{q}}_{n+1}) \end{bmatrix}$$

Or in a shorthand form, where we use $\boldsymbol{\Delta} = \Delta\mathbf{q}_{n+1}$, $\mathbf{D} = \mathbf{D}_{,\mathbf{q}}$, and $\mathbf{e} = -\mathbf{D}(\tilde{\mathbf{q}}_{n+1})$,

$$\begin{bmatrix} \mathbf{I} & \mathbf{D}^T \\ \mathbf{D} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Delta} \\ \boldsymbol{\mu} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{e} \end{bmatrix}$$

We have to solve for the vectors $\mathbf{\Delta}$ and $\boldsymbol{\mu}$.

Start with $\mathbf{\Delta} = -\mathbf{D}^T\boldsymbol{\mu}$, and substitute this in the second equation, $\mathbf{D\Delta} = \mathbf{e}$, as,

$$-\mathbf{D}\mathbf{D}^T\boldsymbol{\mu} = \mathbf{e}$$

Note the dimension of the matrix $\mathbf{D}\,(m \times n)$ where $m < n$ and the product $\mathbf{D}\mathbf{D}^T\;(m \times m)$ which is now square in the smallest dimension m. If this matrix has full rank, which it usually will have, then we can solve for $\boldsymbol{\mu}$ and $\mathbf{e}$,

$$\boldsymbol{\mu} = -\left(\mathbf{D}\mathbf{D}^T\right)^{-1}\mathbf{e}$$
$$\mathbf{\Delta} = \mathbf{D}^T\left(\mathbf{D}\mathbf{D}^T\right)^{-1}\mathbf{e}$$

For an undetermined linear system of equations with full rank matrix $\mathbf{D}$, the matrix,

$$\mathbf{D}^+ = \mathbf{D}^T\left(\mathbf{D}\mathbf{D}^T\right)^{-1}$$

is called the Moove-Penrose pseudo inverse and gives us the least square solution of the problem.

Example:

$$\begin{array}{ll} x_2 - x_1 & = 0 \\ x_2 - x_3 & = 0 \end{array} \quad \text{with values} \quad \begin{array}{l} x_1 = 0.9 \\ x_2 = 1 \\ x_3 = 1 \end{array}$$

The equations are linear so the Jacobian, $\mathbf{D}, \mathbf{x}$ is simply the matrix,

$$\mathbf{D} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \quad \tilde{\mathbf{x}} = \begin{bmatrix} 0.9 \\ 1 \\ 1 \end{bmatrix}$$
$$\mathbf{D}\tilde{\mathbf{x}} = \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} \qquad \text{"errors"}$$

$$\begin{array}{ll} \mathbf{x} = \tilde{\mathbf{x}} + \mathbf{\Delta x} & \rightarrow \mathbf{D}\tilde{\mathbf{x}} + \mathbf{D\Delta x} = 0 \\ \mathbf{D\Delta x} = -\mathbf{D}\tilde{\mathbf{x}} & \rightarrow \mathbf{\Delta x} = \mathbf{D}^+\left(-\mathbf{D}\tilde{\mathbf{x}}\right) \end{array}$$

$$\mathbf{D}^+ = \mathbf{D}^T\left(\mathbf{D}\mathbf{D}^T\right)^{-1}$$
$$\mathbf{D}\mathbf{D}^T = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}\begin{bmatrix} -1 & 0 \\ 1 & -1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

$$\left(\mathbf{D}\mathbf{D}^T\right)^{-1} = \tfrac{1}{3}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$
$$\mathbf{D}^+ = \mathbf{D}^T\left(\mathbf{D}\mathbf{D}^T\right)^{-1} = \tfrac{1}{3}\begin{bmatrix} -2 & -1 \\ 1 & -1 \\ 1 & 2 \end{bmatrix}$$

$$\Delta\mathbf{x} = \mathbf{D}^+\left(-\mathbf{D}\tilde{\mathbf{x}}\right) = \tfrac{1}{3}\begin{bmatrix} -2 & -1 \\ 1 & -1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} -0.1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.0666 \\ -0.0333 \\ -0.0333 \end{bmatrix}$$
$$||\Delta\mathbf{x}|| = 0.0816$$

$$\mathbf{x} = \begin{bmatrix} 0.9 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0.0666 \\ -0.0333 \\ -0.0333 \end{bmatrix} = \begin{bmatrix} 0.9666 \\ 0.9666 \\ 0.9666 \end{bmatrix}$$

Note that this solution (which is on teh constraint surface) is really at the shortest distance from the approximate solution $(0.9, 1, 1)$. You can easily come up with other solutions which are on the surface, like $(1, 1, 1)$, but they are always farther away (check this).

The Gauss-Newton iteration scheme is now,

```
set iterat = 0
set tol = 1e-12
set x_n+1 x_n
set maxiterat = 10
evaluate eps=D(x_n+1)
repeat
  dx_n+1 = -D,x^T(D,x D,x^T) eps
  x_n+1 = x_n+1 + dx_n+1
  eps = D(x_n+1)
  iterat = iterat+1
until max(abs(eps))<tol or iterat>maxiterat
```

Next we determine the speeds which fulfill the constraints, these are linear equations so we have a linear least square problem which we can solve in <u>one</u> step:

```
epsdot=D,x xdot_n+1
dxdot_n+1 = -D,x^T(D,x D,x^T) epsdot
xdot_n+1 = xdot_n+1 + dxdot_n+1
```

Now take a look at the Hrones & Nelson [2] four-bar linkage atlas. A 700 page folio book from 1951, which shows 500.000 solutions of the coupler curve for a general four-bar linkage. And how are these constructed? with a mechanism!

Bibliography

[1] G. J. Talbourdet. Mathematiscal Solutions of Four-Bar Linkages; part I - Analysis of Single 4-Bar Linkage. *Machine Design*, May 1941.

[2] J. A. Hrones and G. L. Nelson. *Analysis of the four bar linkage*. MIT press and John Wiley & Son, New York, 1951.