Feedback Linearisation

We have seen that we can place the poles of a completely controllable LTI system almost anywhere we wish using linear state feedback. Does anything similar exist for nonlinear systems?
We will look at "geometric control methods" which in a sense mimic the approach developed with linear systems. Our development will concentrate on input-state linearisation for single input affine control systems

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u \tag{1}$$

where we assume that this system has a fixed point $\mathbf{x}_e = \mathbf{0}$ corresponding to $u \equiv 0$. $\mathcal{N}$ represents a neighbourhood of this fixed point. We shall describe the conditions under which there exists a nonlinear transformation $\mathbf{z} = \Phi(\mathbf{x}),\ \Phi : \mathcal{N} \to \mathbb{R}^n,\ \Phi(\mathbf{0}) = \mathbf{0}$ and nonlinear feedback law $u = \alpha(\mathbf{x}) + \beta(\mathbf{x})v$ such that the system (1) can be transformed to the linear controllable canonical form

$$\dot{\mathbf{z}} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \mathbf{z} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} v \tag{2}$$

In this form the system is readily seen to be completely controllable and thus can be stabilised using linear feedback

$$v = K\mathbf{z} + w = \begin{pmatrix} k_1 & k_2 & \cdots & k_n \end{pmatrix} \mathbf{z} + w$$

Back in the original variables, this means that the feedback control

$$u = \alpha(\mathbf{x}) + \beta(\mathbf{x})[K\Phi(\mathbf{x}) + w] \tag{3}$$

stabilises the system (1).
Feedback linearisation is to be distinguished from (*Jacobian*) linearisation. The latter approximates the system by a linear model evaluated at an operating point(e.g. a fixed point); this approximation is only valid in some neighbourhood of the operating point. Feedback linearisation on the other hand is a combination of a state transformation followed by the application of a feedback control which replaces the nonlinear system *exactly* by a linear one. Thus any control law designed using this linear system will result in trajectories that will be followed exactly rather than approximately. If the neighbourhood in which Φ exists is the entire state space, then the strategy will be globally valid.

After the transformation $\mathbf{z} = \Phi(\mathbf{x})$, but before the application of the control $u = \alpha(\mathbf{x}) + \beta(\mathbf{x})v$, the system equations have become

$$\begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \vdots \\ \dot{z}_{n-1} \\ \dot{z}_n \end{pmatrix} = \begin{pmatrix} z_2 \\ z_3 \\ \vdots \\ z_n \\ A(\mathbf{x}) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ B(\mathbf{x}) \end{pmatrix} u$$

where A and B are nonlinear functions of $\mathbf{x}$ and thus implicitly of $\mathbf{z}$. Thus the application of the feedback control is designed to *cancel* these nonlinearities in order to render the state equation for z_n as $\dot{z}_n = v$.

Algorithm:

1. compute the set of vector fields

$$F_n = \{\mathbf{g},\ ad_{\mathbf{f}}\,\mathbf{g}, ad_{\mathbf{f}^2}\,\mathbf{g}, \ldots, ad_{\mathbf{f}^{n-1}}\,\mathbf{g}\}$$

2. check that there exists a region $\mathcal{N}$ such that (i) F_n is linearly independent (equivalent to the system being controllable) and (ii) F_{n-1} is involutive (meaning that a certain set of pdes can be solved). If so then...

3. Solve the system of pdes

$$\begin{aligned} L_{ad_{\mathbf{f}^i}\,\mathbf{g}}\, z_1 &= 0, \qquad\qquad i = 0, 1, \ldots, n-2 \\ L_{ad_{\mathbf{f}^{n-1}}\,\mathbf{g}}\, z_1 &\neq 0 \end{aligned}$$

for z_1 the first component of $\mathbf{z}$.

4. Compute the state transformation

$$\mathbf{z}(\mathbf{x}) = \Phi(\mathbf{x}) = \big(z_1,\quad L_{\mathbf{f}}\, z_1,\quad L_{\mathbf{f}^2}\, z_1,\quad \ldots,\quad L_{\mathbf{f}^{n-1}}\, z_1\big)$$

and the control functions

$$\beta(\mathbf{x}) = \frac{1}{L_{\mathbf{g}} L_{\mathbf{f}^{n-1}}\, z_1}, \qquad\qquad \alpha(\mathbf{x}) = -\frac{L_{\mathbf{f}^n}\, z_1}{L_{\mathbf{g}} L_{\mathbf{f}^{n-1}}\, z_1}.$$

Example: Consider the 2nd order affine system

$$\dot{\mathbf{x}} = \underbrace{\begin{pmatrix} x_2 + 4x_1^2 x_2 \\ -x_1 \end{pmatrix}}_{\mathbf{f}(\mathbf{x})} + \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{\mathbf{g}(\mathbf{x})} u$$

Applying the algorithm, we get

Step 1.

$$[\mathbf{f}, \mathbf{g}] = \frac{\partial \mathbf{g}}{\partial \mathbf{x}}\mathbf{f} - \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\mathbf{g} = \mathbf{0} - \begin{pmatrix} 8x_1x_2 & 1+4x_1^2 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -1-4x_1^2 \\ 0 \end{pmatrix}$$

Hence

$$F_2 = \{\mathbf{g},\ ad_{\mathbf{f}}\mathbf{g}\} = \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} -1-4x_1^2 \\ 0 \end{pmatrix} \right\}.$$

Step 2. F_2 is by inspection linearly independent, and $F_1 = \{\mathbf{g}\}$ is involutive. So ....

Step 3.

$$\begin{aligned} L_{\mathbf{g}} z_1 = 0 \qquad & \Rightarrow \qquad \begin{pmatrix} \frac{\partial z_1}{\partial x_1} & \frac{\partial z_1}{\partial x_2} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 0 \\ & \Rightarrow \qquad \frac{\partial z_1}{\partial x_2} = 0 \end{aligned} \tag{4}$$

$$\begin{aligned} L_{ad_{\mathbf{f}}\mathbf{g}}\, z_1 \neq 0 \qquad & \Rightarrow \qquad \begin{pmatrix} \frac{\partial z_1}{\partial x_1} & \frac{\partial z_1}{\partial x_2} \end{pmatrix} \begin{pmatrix} -1-4x_1^2 \\ 0 \end{pmatrix} \neq 0 \\ & \Rightarrow \qquad (1+4x_1^2)\frac{\partial z_1}{\partial x_1} \neq 0 \\ & \Rightarrow \qquad \frac{\partial z_1}{\partial x_1} \neq 0 \\ & \Rightarrow \qquad \frac{\partial z_1}{\partial x_1} = 1 \ \text{(say)} \end{aligned} \tag{5}$$

A solution of Eqs (4) and (5) is

$$z_1 = x_1$$

Step 4.

$$\begin{aligned} z_2 &= L_{\mathbf{f}}\, z_1 \\ &= \begin{pmatrix} \frac{\partial z_1}{\partial x_1} & \frac{\partial z_1}{\partial x_2} \end{pmatrix} \begin{pmatrix} x_2 + 4x_1^2 x_2 \\ -x_1 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_2 + 4x_1^2 x_2 \\ -x_1 \end{pmatrix} \\ &= x_2 + 4x_1^2 x_2 \end{aligned}$$

We compute

$$
\begin{aligned}
L_{\mathbf{g}}\, z_2 &= \begin{pmatrix} \frac{\partial z_2}{\partial x_1} & \frac{\partial z_2}{\partial x_2} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\
&= \begin{pmatrix} 8x_1x_2 & 1+4x_1^2 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\
&= 1+4x_1^2
\end{aligned}
$$

and

$$
\begin{aligned}
L_{\mathbf{f}}\, z_2 &= \begin{pmatrix} \frac{\partial z_2}{\partial x_1} & \frac{\partial z_2}{\partial x_2} \end{pmatrix} \begin{pmatrix} x_2+4x_1^2x_2 \\ -x_1 \end{pmatrix} \\
&= \begin{pmatrix} 8x_1x_2 & 1+4x_1^2 \end{pmatrix} \begin{pmatrix} x_2+4x_1^2x_2 \\ -x_1 \end{pmatrix} \\
&= x_1(8x_2^2-1)(1+4x_1^2)
\end{aligned}
$$

then

$$
\begin{aligned}
\beta(\mathbf{x}) &= \frac{1}{L_{\mathbf{g}}\, z_2} \\
&= \frac{1}{1+4x_1^2}
\end{aligned}
$$

and

$$
\begin{aligned}
\alpha(\mathbf{x}) &= -\frac{L_{\mathbf{f}}\, z_2}{L_{\mathbf{g}}\, z_2} \\
&= -\frac{x_1(8x_2^2-1)(1+4x_1^2)}{1+4x_1^2} \\
&= -x_1(8x_2^2-1)
\end{aligned}
$$

As a check,

$$
\begin{aligned}
z_1 &= x_1 \\
\Rightarrow \dot{z}_1 &= \dot{x}_1 = x_2+4x_1^2x_2 = z_2
\end{aligned}
$$

$$
\begin{aligned}
z_2 &= x_2+4x_1^2x_2 \\
\Rightarrow \dot{z}_2 &= (1+4x_1^2)\dot{x}_2 + 8x_1x_2\dot{x}_1 \\
&= (1+4x_1^2)(-x_1+u) + 8x_1x_2(x_2+4x_1^2x_2) \\
&= x_1(8x_2^2-1)(1+4x_1^2) + (1+4x_1^2)u \qquad (6)
\end{aligned}
$$

With the feedback control

$$
u = \alpha(\mathbf{x}) + \beta(\mathbf{x})v = -x_1(8x_2^2-1) + \frac{1}{1+4x_1^2}\, v
$$

Eq (6) becomes

$$\dot{z}_2 = x_1(8x_2^2 - 1)(1 + 4x_1^2) + (1 + 4x_1^2)\left(-x_1(8x_2^2 - 1) + \frac{v}{1 + 4x_1^2}\right) = v$$

In summary, the system

$$\dot{\mathbf{x}} = \begin{pmatrix} x_2 + 4x_1^2 x_2 \\ -x_1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u$$

can be input-state linearised by means of

- the state transformation

$$\begin{aligned} \mathbf{z} &= \Phi(\mathbf{x}) = \begin{pmatrix} x_1 \\ x_2 + 4x_1^2 x_2 \end{pmatrix} \\ \Leftrightarrow \mathbf{x} &= \Phi^{-1}(\mathbf{z}) = \begin{pmatrix} z_1 \\ \frac{z_2}{1+4z_1^2} \end{pmatrix} \end{aligned}$$

  (Note that this transformation is globally defined.)

- the feedback control

$$u = \alpha(\mathbf{x}) + \beta(\mathbf{x})v = -x_1(8x_2^2 - 1) + \frac{1}{1 + 4x_1^2}\, v.$$

It becomes

$$\dot{\mathbf{z}} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \mathbf{z} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v.$$

Example (cont'd):
Use the feedback linearised system to find a stabilising control for the original system.
With the control $v = k_1 z_1 + k_2 z_2 + w$, the transformed system becomes

$$\dot{\mathbf{z}} = \begin{pmatrix} 0 & 1 \\ k_1 & k_2 \end{pmatrix} \mathbf{z} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} w.$$

Thus by choosing $k_1 < 0$ and $k_2 < 0$, both of its poles will have $\Re < 0$, i.e. the system will have been stabilised. Back in the original system , this is equivalent to using the control

$$\begin{aligned} u = -x_1(8x_2^2 - 1) + \frac{1}{1 + 4x_1^2}\, v &= -x_1(8x_2^2 - 1) + \frac{1}{1 + 4x_1^2}(k_1 z_1 + k_2 z_2 + w) \\ &= -x_1(8x_2^2 - 1) + \frac{1}{1 + 4x_1^2}\left(k_1 x_1 + k_2(x_2 + 4x_1^2 x_2) + w\right) \\ &= -x_1(8x_2^2 - 1) + \frac{k_1 x_1}{1 + 4x_1^2} + k_2 x_2 + \frac{1}{1 + 4x_1^2} w. \end{aligned}$$