Chapter #6

EEE3001 - 8013

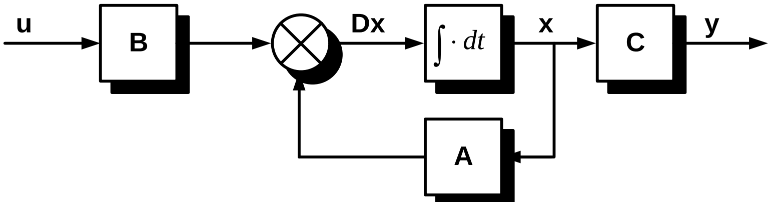
State Space Analysis and Controller Design

- **State feedback**
- **Linear Quadratic Regulator (LQR)**
- **Open and closed loop estimators**
- **Reduced Order Estimators (not assessed)**
- **Tracking**

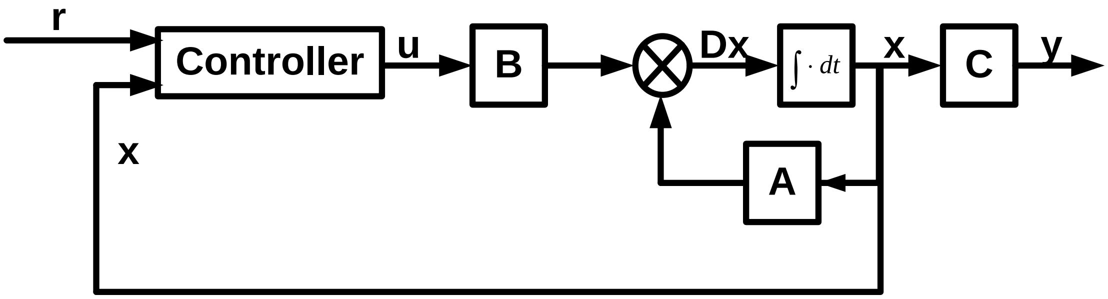
Module Leader: Dr Damian Giaouris - damian.giaouris@ncl.ac.uk

State feedback

The BD of the ss model is:



In state space systems we can have two kinds of feedback; the output and the state feedback (we will only study the state feedback method):



The task of the controller is to produce the appropriate control signal **u** that will insure that **y**=**r**. Let's assume the simplest form of control: $\mathbf{u}(t)=\mathbf{K}_1\mathbf{r}(t)-\mathbf{K}\mathbf{x}(t)$, where $\mathbf{K}_1$ and $\mathbf{K}$ are appropriately selected gain vectors/matrices. The exact dimensions of these vectors will be defined later.

Using the standard state space equation:

$$\left.\begin{aligned}\mathbf{u}(t)&=\mathbf{K}_1\mathbf{r}(t)-\mathbf{K}\mathbf{x}(t)\\ \dot{\mathbf{x}}(t)&=\mathbf{A}\mathbf{x}(t)+\mathbf{B}\mathbf{u}(t)\end{aligned}\right\}\Rightarrow$$

$$\dot{\mathbf{x}}(t)=\mathbf{A}\mathbf{x}(t)+\mathbf{B}(\mathbf{K}_1\mathbf{r}(t)-\mathbf{K}\mathbf{x}(t))\Leftrightarrow\dot{\mathbf{x}}(t)=(\mathbf{A}-\mathbf{B}\mathbf{K})\mathbf{x}(t)+\mathbf{B}\mathbf{K}_1\mathbf{r}(t)$$
$$\mathbf{y}(t)=\mathbf{C}\mathbf{x}(t)+\mathbf{D}\mathbf{u}(t)=(\mathbf{C}-\mathbf{D}\mathbf{K})\mathbf{x}(t)+\mathbf{D}\mathbf{K}_1\mathbf{r}(t)$$

But the closed loop system is a state space model, so it must be described by a state equation:

$$\left.\begin{aligned}\mathbf{x}(t)&=\mathbf{A}_{CL}\mathbf{x}(t)+\mathbf{B}_{CL}\mathbf{r}(t)\\ \mathbf{y}(t)&=\mathbf{C}_{CL}\mathbf{x}(t)+\mathbf{D}_{CL}\mathbf{u}(t)\end{aligned}\right\}\text{where }\left.\begin{aligned}\mathbf{A}_{CL}&=\mathbf{A}-\mathbf{B}\mathbf{K}\\ \mathbf{B}_{CL}&=\mathbf{B}\mathbf{K}_1\\ \mathbf{C}_{CL}&=\mathbf{C}-\mathbf{D}\mathbf{K}\\ \mathbf{D}_{CL}&=\mathbf{D}\mathbf{K}_{\mathbf{1}}\end{aligned}\right\}$$

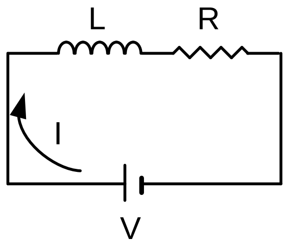
are the closed loop matrices. If **r**=0 the system is called regulator.

The CL state matrix is a function of **K**, therefore by appropriate changing **K** we can change the eigenvalues of $\mathbf{A}_{\mathrm{CL}}$) which means that we can improve's performance, i.e. to make it faster/stable. This method is called pole placement.

<u>**WE MUST CHECK IF THE SYSTEM IS CONTROLLABLE**</u>**.**

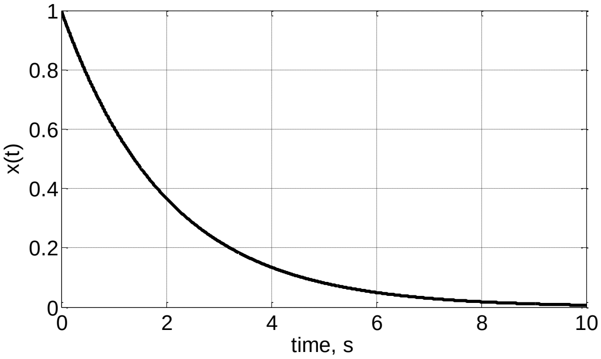
Example 1:

Assume that we have an RL circuit:



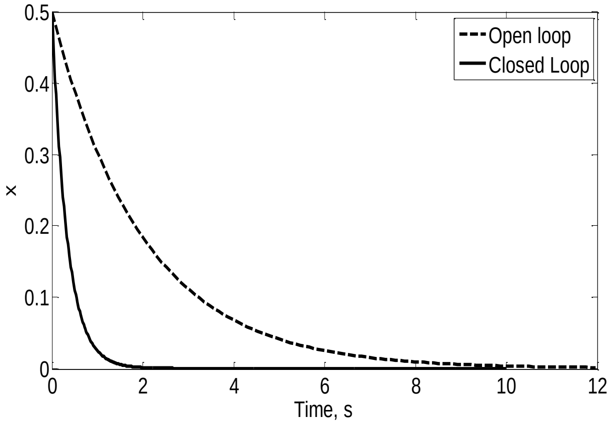
The system is described by the equation: $\frac{di}{dt}=\frac{1}{L}(V-iR)$. We want to control the system in such a way that by changing V we will achieve a satisfactory value of *i* (steady state). Also we want to control the dynamics of the system (oscillations, settling time, pole location…).

The state space model: $\frac{di}{dt} = \frac{1}{L}(V - iR) \Rightarrow \frac{di}{dt} = -\frac{R}{L}i + \frac{1}{L}V$, this implies that $\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}$, where **A**=-*R/L*, **x**=*i*, **B**=1/*L* and **u**=*V*. Obviously the eigenvalue is –*R*/*L* and the system will exponentially converge to zero when **x**(0) is not zero (assume that this eigenvalue is 0.5) and *V*=0:



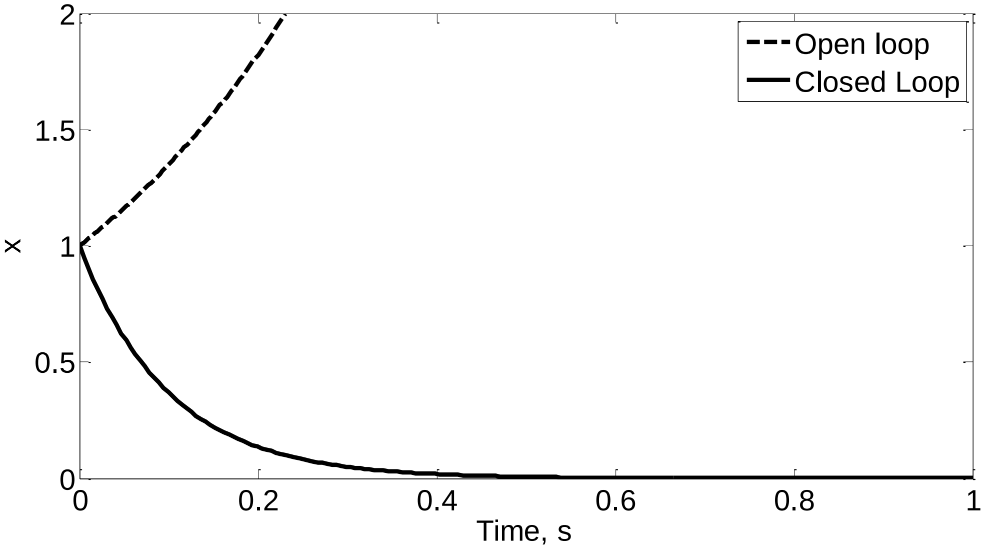
If we want to make the system faster then we can use a state feedback control strategy. The new signal **u** is **-Kx**

This implies that $\mathbf{A_{CL}} = \mathbf{A} - \mathbf{BK} = \frac{-R}{L} - \frac{K}{L} = \frac{-R-K}{L}$. The new system has an eigenvalue at $\frac{-R-K}{L}$. If we want to place the eigenvalue at -6*R*/*L* then $\frac{-R-K}{L} = -6\frac{R}{L} \Rightarrow K = 5R$:



Example 2:

Assume $\dot{x} = 3x + u$ (unstable system), $x(0)=1$. We use a state feedback controller $u = -kx$ which implies that the Closed Loop (CL) state equation is $\dot{x} = (3-k)x$, hence the eigenvalue of the CL system is 3-*k*. If we want the CL eigenvalues at -10 (stable and very fast) then *k*=13:



Example 3:

Find the eigenvalues of $\left.\begin{aligned}\dot{\mathbf{x}}(t) &= \begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix}\mathbf{x}(t) + \begin{bmatrix}1\\ 0\end{bmatrix}u\\ \mathbf{y}(t) &= \begin{bmatrix}1 & 0\end{bmatrix}\mathbf{x}(t)\end{aligned}\right\}$. If the system is unstable create a controller that will stabilise the system.

The eigenvalues are: -0.3723, 5.3723. Hence we need to create a feedback controller to stabilise the system. Before that we need to see if we can influence both states, i.e. if the system is controllable:

$\mathbf{AB}=\begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix}\begin{bmatrix}1\\ 0\end{bmatrix}=\begin{bmatrix}1\\ 3\end{bmatrix}\Rightarrow[\mathbf{B}\quad \mathbf{AB}]=\begin{bmatrix}1 & 1\\ 0 & 3\end{bmatrix}$, since the controllability matrix has 2 LI vectors, the system is controllable. To check that we can also use the determinant of the controllability matrix: $\begin{vmatrix}1 & 1\\ 0 & 3\end{vmatrix}=3\neq 0$.

So the system is controllable and hence we can use a pole placement strategy. The new CL state matrix is:

$$\mathbf{A_{CL}}=\begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix}-\begin{bmatrix}1\\ 0\end{bmatrix}[k_1\quad k_2]=\begin{bmatrix}1-k_1 & 2-k_2\\ 3 & 4\end{bmatrix}$$

The eigenvalues of the closed loop system are:

$$eig\left(\begin{bmatrix}1-k_1 & 2-k_2\\ 3 & 4\end{bmatrix}\right)=\begin{vmatrix}\lambda-1+k_1 & -2+k_2\\ -3 & \lambda-4\end{vmatrix}=0.$$

Assume that we want to place the CL poles at -10 and -11:

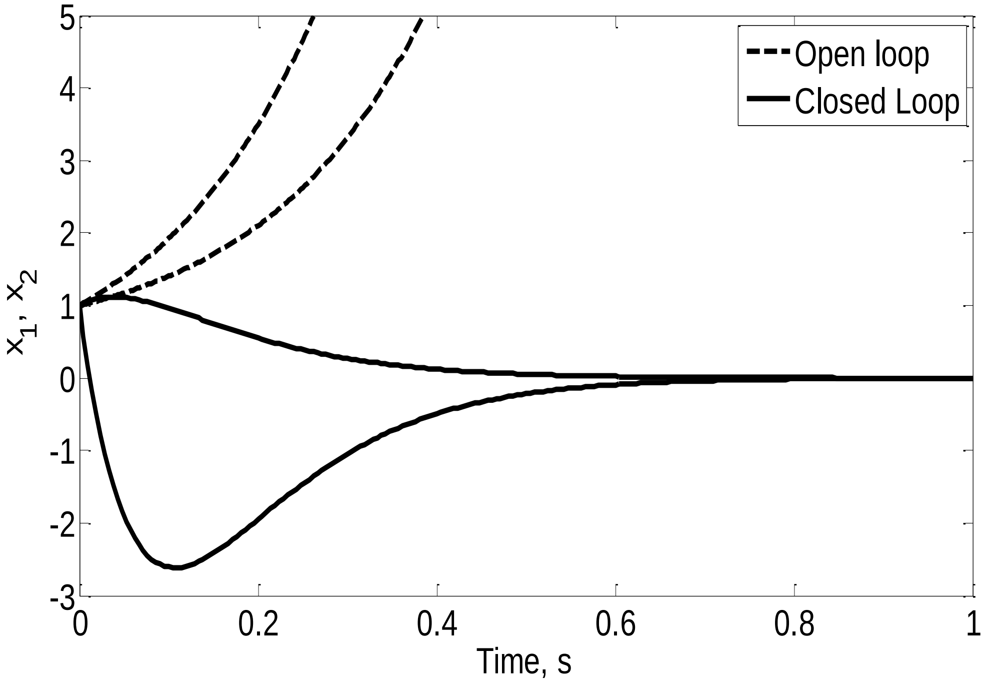
- $\begin{vmatrix}-10-1+k_1 & -2+k_2\\ -3 & -10-4\end{vmatrix}=0\Leftrightarrow\begin{vmatrix}-11+k_1 & -2+k_2\\ -3 & -14\end{vmatrix}=0\Leftrightarrow$

$$-14(-11+k_1)+3(-2+k_2)=0\Leftrightarrow 154-14k_1-6+3k_2=0$$

- $\begin{vmatrix}-11-1+k_1 & -2+k_2\\ -3 & -11-4\end{vmatrix}=0\Leftrightarrow\begin{vmatrix}-12+k_1 & -2+k_2\\ -3 & -15\end{vmatrix}=0\Leftrightarrow$

$$-15(-12+k_1)+3(-2+k_2)=0\Leftrightarrow 180-15k_1-6+3k_2=0$$

So a 2 by 2 system: $\left.\begin{aligned}14k_1-3k_2&=148\\ 15k_1-3k_2&=174\end{aligned}\right\}\Leftrightarrow k_1=26 \text{ so } k_2=72$.

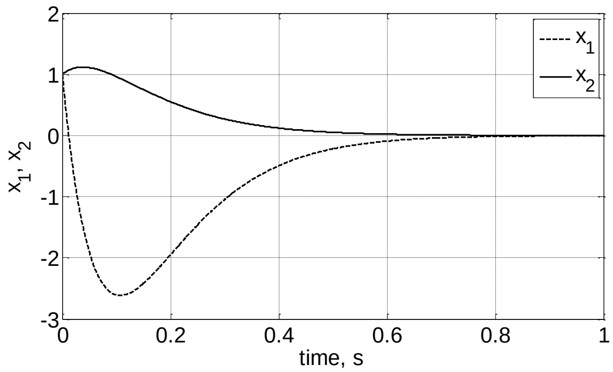

5
4
3
2
1
0
-1
-2
-3
0
0.2
0.4
0.6
0.8
1
x_1, x_2
Time, s
Open loop
Closed Loop

Linear Quadratic Regulator (LQR)

Previously we saw that the necessary gains for the system:

$\dot{\mathbf{x}}(t)=\begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix}\mathbf{x}(t)+\begin{bmatrix}1\\ 0\end{bmatrix}\mathbf{u}(t),\ \mathbf{y}(t)=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\mathbf{x}(t)$ are 26 and 72.

This will place the poles at -10 and -11 and we have a nice stable response:

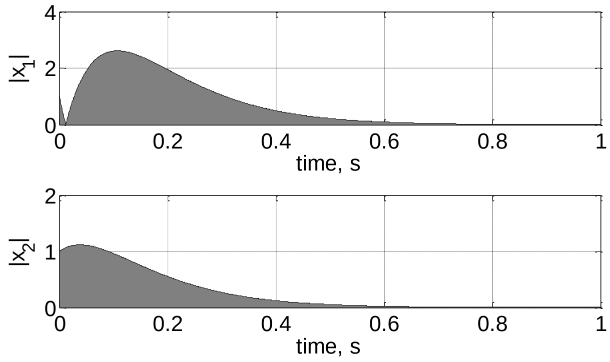


Obviously in that system we want to "converge to zero as fast as possible". In order to quantify the phrase "converge to zero as fast as possible" we can use various performance indexes like:

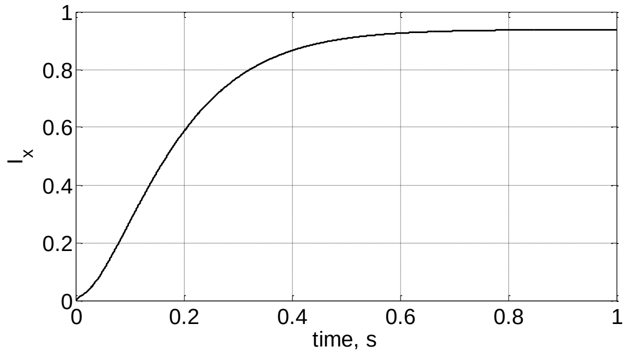
$$I_x=\int\left(|x_1|+|x_2|\right)dt\,,$$

Question: Why do we need to take the absolute values?

The last index effectively tries to minimize the area under the curves produced by x_1, x_2:



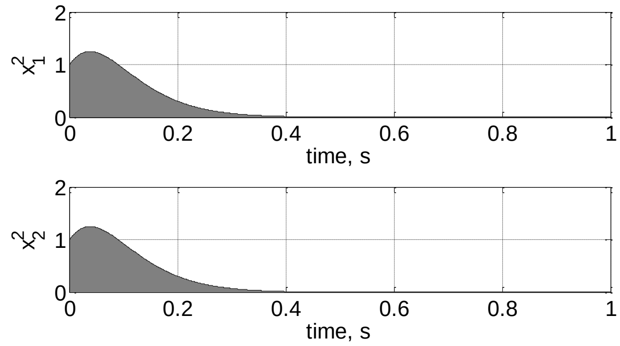
Obviously the faster the system, the smaller the above areas and hence the smaller the performance index, in this case the index has as a final value 0.9383 and its graph is:



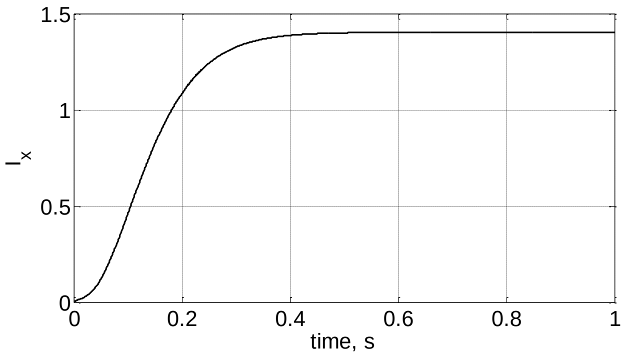
Another way to quantify the speed is to use the following index:

$$I_x = \int \left(x_1^2 + x_2^2 \right) dt$$

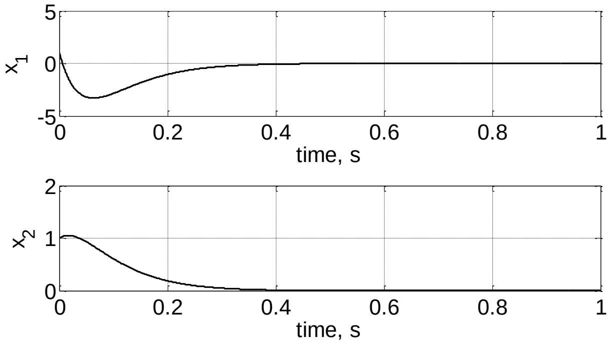
This is preferred as the graphs will be smoother:



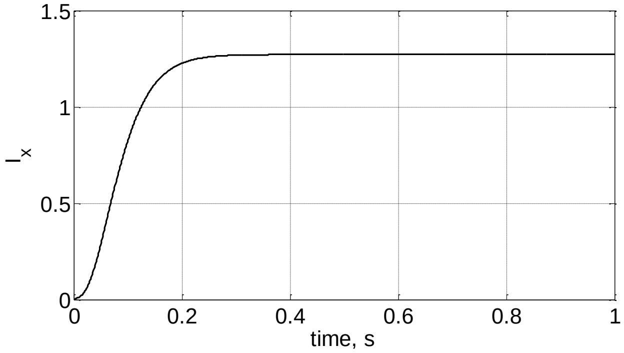
In this case we have I_x=1.403:



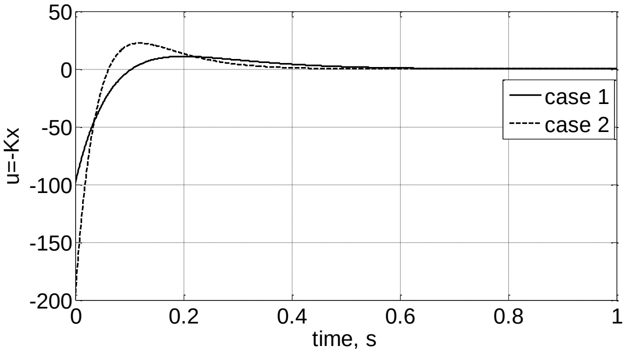
In order to see that indeed this index quantifies the speed, we see that this is reduced if we have a faster system. To do that we choose the pole location at: -15 -20:



And the index I_x=1.271

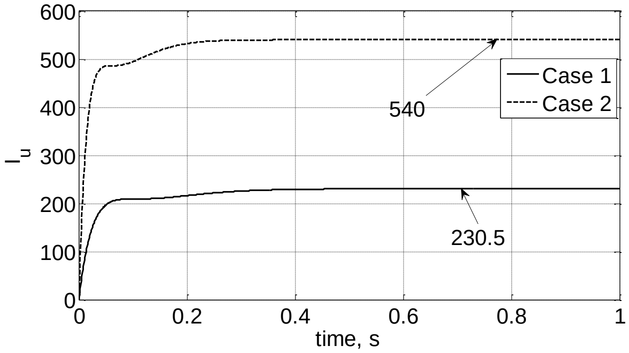


So someone could state that the smaller the desired eigenvalues the “better” the system. This is not true as the smaller the eigenvalues, the higher the gains that are required. Which effectively means that the signal **u** will be higher in the second case:



This practically means that we need to use more energy, and more expensive equipment (for example a higher rating dc converter to drive a DC motor).

In order to quantify this "energy" we use a similar index as before $I_u = \int u^2 dt$:



Thus we see that the "best" system is the one that minimises both indexes:

$I_x = \int \left(x_1^2 + x_2^2\right)dt$ and $I_u = \int u^2 dt$ or

$$I = \int \left(x_1^2 + x_2^2\right) + u^2 dt$$

If we have a system where the speed is more important than the energy then we use: $I = \int q\left(x_1^2 + x_2^2\right) + u^2 dt$

In general we can use $I = \int q\left(x_1^2 + x_2^2\right) + ru^2 dt$ and we chose the positive gains *q* and *r* to denote the "importance" of the speed or of the energy.

In general we use:

$$\mathbf{I} = \int \mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)dt$$

Where **Q** and **R** are positive definite matrices (the equivalent of positive numbers in matrices, they are square symmetric matrices with positive eigenvalues and other properties like $\mathbf{x}^T\mathbf{Q}\mathbf{x} > 0$ for all nonzero x). In general **Q** and **R** can be any positive definite matrices but we will use only diagonal matrices where all the elements are equal.

Our task: Design a controller (called LQR) $\mathbf{u} = -\mathbf{K}\mathbf{x}$ for $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ such as it is going to minimise: $J = \int_0^\infty \left(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right)dt$, where **Q** and **R** are positive definite matrices. **Q**: Importance of the error, **R**: Importance of the energy that we use.

The optimum controller can be found:

Not assessed material

$\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}$ (assume that $\mathbf{A} - \mathbf{B}\mathbf{K}$ is stable)

So: $J = \int_0^\infty \left(\mathbf{x}^T\mathbf{Q}\mathbf{x} + (-\mathbf{K}\mathbf{x})^T\mathbf{R}(-\mathbf{K}\mathbf{x})\right)dt$

$$\Rightarrow J = \int_0^\infty \mathbf{x}^T\left(\mathbf{Q} + \mathbf{K}^{\mathbf{T}}\mathbf{R}\mathbf{K}\right)\mathbf{x}\,dt$$

Assume that $\mathbf{x}^T\left(\mathbf{Q} + \mathbf{K}^{\mathbf{T}}\mathbf{R}\mathbf{K}\right)\mathbf{x} = -\frac{d}{dt}\left(\mathbf{x}^{\mathbf{T}}\mathbf{P}\mathbf{x}\right) = \dot{\mathbf{x}}^{\mathbf{T}}\mathbf{P}\mathbf{x} + \mathbf{x}^{\mathbf{T}}\mathbf{P}\dot{\mathbf{x}}$

P is positive definite.

By replacing $\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{B}\mathbf{K})\mathbf{x}$ and $\dot{\mathbf{x}}^T = \mathbf{x}^T(\mathbf{A} - \mathbf{B}\mathbf{K})^T$:

$$\mathbf{x}^T\left(\mathbf{Q}+\mathbf{K}^\mathbf{T}\mathbf{R}\mathbf{K}\right)\mathbf{x}=\mathbf{x}^T(\mathbf{A}-\mathbf{B}\mathbf{K})^T\mathbf{P}\mathbf{x}+\mathbf{x}^\mathbf{T}\mathbf{P}(\mathbf{A}-\mathbf{B}\mathbf{K})\mathbf{x}$$
$$=\mathbf{x}^T\left((\mathbf{A}-\mathbf{B}\mathbf{K})^T\mathbf{P}+\mathbf{P}(\mathbf{A}-\mathbf{B}\mathbf{K})\right)\mathbf{x}$$

and since the previous equation holds for all **x**:

$$\mathbf{Q}+\mathbf{K}^\mathbf{T}\mathbf{R}\mathbf{K}=(\mathbf{A}-\mathbf{B}\mathbf{K})^T\mathbf{P}+\mathbf{P}(\mathbf{A}-\mathbf{B}\mathbf{K})$$

This equation can be solved only if **A-BK** is stable and it can be shown that $\mathbf{K}=\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$

By replacing $\mathbf{K}=\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$ into $\mathbf{Q}+\mathbf{K}^\mathbf{T}\mathbf{R}\mathbf{K}=(\mathbf{A}-\mathbf{B}\mathbf{K})^T\mathbf{P}+\mathbf{P}(\mathbf{A}-\mathbf{B}\mathbf{K})$=>

$$\mathbf{A}^T\mathbf{P}+\mathbf{P}\mathbf{A}-\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}+\mathbf{Q}=0$$

The last equation is called “**Reduced Riccati Equation**”.

Steps to design an LQR controller:

1. Solve $\mathbf{A}^T\mathbf{P}+\mathbf{P}\mathbf{A}-\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}+\mathbf{Q}=0$ to find the optimum **P**.
2. Use $\mathbf{K}=\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}$ to find the optimum gain **K**.

In general it is difficult to solve the Reduced Riccati Equation and for that reason we need to use Matlab.

Example:

A system is given by $\mathbf{A}=\begin{bmatrix}0 & 1\\ 0 & -1\end{bmatrix}, \mathbf{B}=\begin{bmatrix}0\\ 1\end{bmatrix}$

Find **K**, the eigenvalues of **A-BK**, and the response of the system for **R**=1 and **Q**=eye(2) and **Q**=2*eye(2) (**X**(0)=[1 1]). It is given that the matrix P in the first case is $\mathbf{P}=\begin{bmatrix}2 & 1\\ 1 & 1\end{bmatrix}$ and in the second $\mathbf{P}=\begin{bmatrix}3.4 & 1.4\\ 1.4 & 1.4\end{bmatrix}$

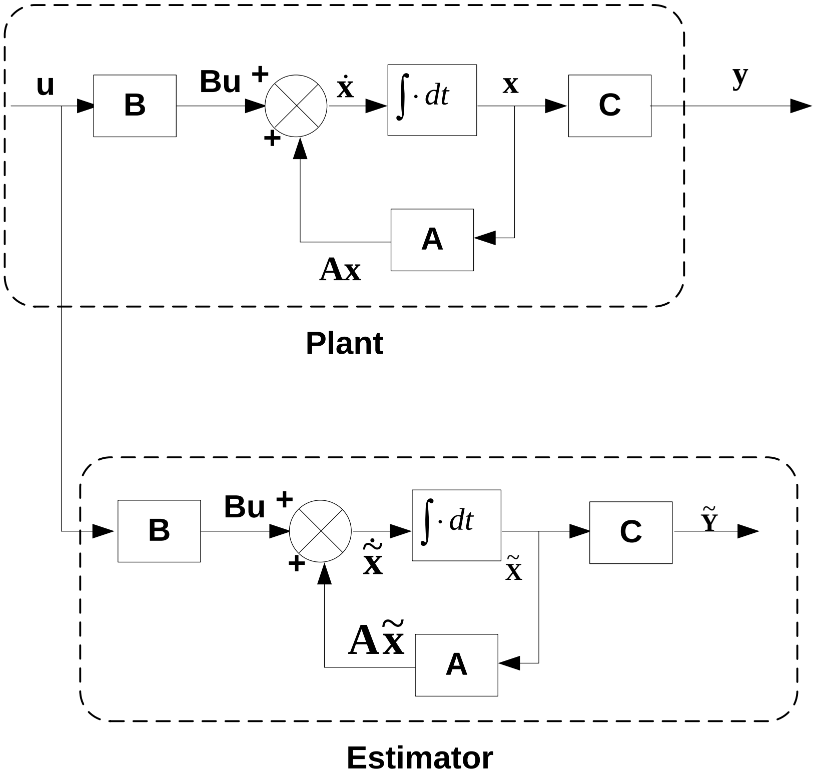
We know that $\mathbf{K}=\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{P}$, hence: $\mathbf{K}=1\times\begin{bmatrix}0 & 1\end{bmatrix}\times\begin{bmatrix}2 & 1\\ 1 & 1\end{bmatrix}=\begin{bmatrix}1 & 1\end{bmatrix}$ and

$\mathbf{K}=1\times\begin{bmatrix}0 & 1\end{bmatrix}\times\begin{bmatrix}3.4 & 1.4\\ 1.4 & 1.4\end{bmatrix}=\begin{bmatrix}1.4 & 1.4\end{bmatrix}$

Thus we see that in the 2nd case where we have a higher importance in the system's error we have higher gains.

Estimating techniques

Until now we assumed that we can measure all states for our state feedback laws. But usually we only have **y** and not **x** at our disposal. If we had a perfect model of the system then we can use a mathematical representation of the actual system:

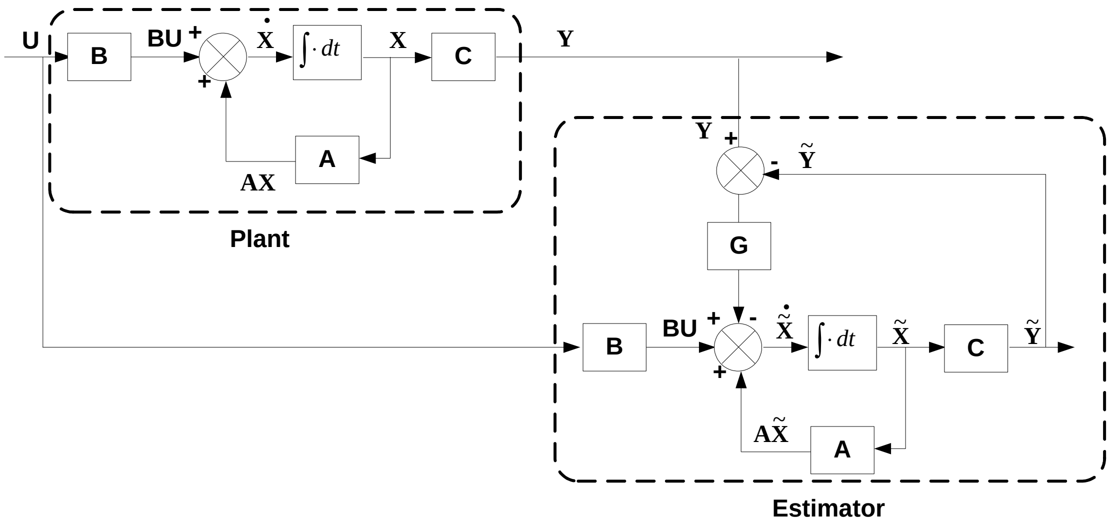


Where $\tilde{\mathbf{x}},\tilde{\mathbf{y}}$ are the estimated (or calculated) state and output vector respectively. Then we can use in the state feedback the estimated states instead of the actual ones. The error between the estimated and real state is

$$\mathbf{e}(t)=\mathbf{x}(t)-\tilde{\mathbf{x}}(t)\Rightarrow\dot{\mathbf{e}}(t)=\dot{\mathbf{x}}(t)-\dot{\tilde{\mathbf{x}}}(t)\Leftrightarrow$$
$$\mathbf{Ax}(t)+\mathbf{Bu}(t)-\mathbf{A}(t)\tilde{\mathbf{x}}(t)-\mathbf{Bu}(t)\Rightarrow\dot{\mathbf{e}}(t)=\mathbf{Ae}(t)$$

Therefore the error is described by a homogeneous ODE. If $\mathbf{e}(0) \neq 0$ and **A** has unstable eigenvalues, then the error will diverge to infinity. Even if **A** is stable the error will depend on the dynamics of **A**. If **A** is slow then the error will slowly converge to zero.

Obviously one way to quantify the estimation is to monitor the error between the actual and the estimated state vector. Unfortunately we cannot monitor the actual state vector (if we could then we would not need the estimator) so we have to use the error between the actual and the estimated output $\mathbf{y} - \tilde{\mathbf{y}}$. We feed this signal into the estimator:



The gain **G** will be used so that we have a fast and stable estimator:

$$\mathbf{e}(t) = (\mathbf{A} - \mathbf{GC})\mathbf{x}(t) - (\mathbf{A} - \mathbf{GC})\tilde{\mathbf{x}}(t) = (\mathbf{A} - \mathbf{GC})\mathbf{e}(t)$$

Hence by appropriately choosing **G** we can force the error to converge to zero very fast.

<u>But the system must be observable</u>

Reduced Order Estimators (ROE) – Not assessed Material

In the previous case the estimator was estimating all the states of the controller. On the other hand, we usually want to estimate just some of the states since the others can be measured. With this way we have a simpler observer and the estimation process is better. The estimator or observer that is estimating some states is called Reduced Ordered Estimator.

Assume that we can only measure one state (as it is usually the case), then we can use a different approach for the estimator design.

Imagine that the only state that can be measured is x_a. The rest of the state vector $\mathbf{X}_b$ cannot be measured: $\mathbf{X} = \left[\begin{array}{c} x_a \\ \hline \mathbf{X}_b \end{array}\right]$. By partitioning the state space equations we can have:

$$\begin{bmatrix} \overset{\bullet}{x}_a \\ \mathbf{X}_b \end{bmatrix} = \begin{bmatrix} A_{aa} & \mathbf{A}_{ab} \\ \mathbf{A}_{ba} & \mathbf{A}_{bb} \end{bmatrix} \begin{bmatrix} x_a \\ \mathbf{X}_b \end{bmatrix} + \begin{bmatrix} B_a \\ \mathbf{B}_b \end{bmatrix} \mathbf{U}$$

$$\mathbf{Y} = \begin{bmatrix} 1 & \mathbf{0} \end{bmatrix} \begin{bmatrix} x_a \\ \mathbf{X}_b \end{bmatrix}$$

where A_{aa} is a scalar number
$\mathbf{A}_{ab}$ is 1 by (n-1) vector
$\mathbf{A}_{ba}$ is (n-1) by 1 vector
$\mathbf{A}_{bb}$ is (n-1) by (n-1) matrix
B_a is scalar
$\mathbf{B}_b$ is (n-1) by 1 vector
Then a new state space model can be defined by the non-measurable state:

$$\overset{\bullet}{\mathbf{X}}_b = \mathbf{A}_{bb}\mathbf{X}_b + \mathbf{A}_{ba}x_a + \mathbf{B}_b\mathbf{U}$$

In the last eqn. the matrix $\mathbf{A}_{bb}$ is the new state matrix and $\mathbf{A}_{ba}\,x_a+\mathbf{B}_b\mathbf{U}$ is the new known input.

The known state is:

$$\dot{x}_a=\dot{\mathbf{Y}}=\mathbf{A}_{aa}x_a+\mathbf{A}_{ab}\mathbf{X}_b+\mathbf{B}_a\mathbf{U}$$

$$\Leftrightarrow\dot{\mathbf{Y}}-\mathbf{A}_{aa}x_a-\mathbf{B}_a\mathbf{U}=\mathbf{A}_{ab}\mathbf{X}_b\Leftrightarrow\mathbf{Y}_n=\mathbf{C}_n\mathbf{X}_n=\mathbf{A}_{ab}\mathbf{X}_b$$

At the last eqn. the term $\dot{\mathbf{Y}}-\mathbf{A}_{aa}x_a-\mathbf{B}_a\mathbf{U}$ is a known measurement.

Hence the new state space model will have states: $\mathbf{X}_\mathbf{n}=\mathbf{X}_b$, state matrix $\mathbf{A}_\mathbf{n}=\mathbf{A}_{bb}$, Input $\mathbf{BU}_\mathbf{n}=\mathbf{A}_{ba}\,x_a+\mathbf{B}_b\mathbf{U}$, $\mathbf{C}_\mathbf{n}=\mathbf{A}_{ab}$ and the output is

$$\mathbf{Y}_n=\dot{\mathbf{Y}}-\mathbf{A}_{aa}x_a-\mathbf{B}_a\mathbf{U}$$

$$\left.\begin{array}{l}\dot{\mathbf{X}}(t)=\mathbf{AX}(t)+\mathbf{BU}(t)\\ \mathbf{Y}(t)=\mathbf{CX}(t)\end{array}\right\}\Rightarrow\left.\begin{array}{l}\dot{\mathbf{X}}_n=\mathbf{A}_n\mathbf{X}_n+\mathbf{BU}_n\\ \mathbf{Y}_n=\mathbf{C}_n\mathbf{X}_n\end{array}\right\}\Rightarrow\left.\begin{array}{l}\dot{\mathbf{X}}_b=\mathbf{A}_{\mathbf{bb}}\mathbf{X}_b+A_{ba}\;x_a+\;\mathbf{B}_\mathbf{b}\mathbf{U}\\ \mathbf{Y}_n=\dot{\mathbf{Y}}-\mathbf{A}_{aa}x_a-\mathbf{B}_a\mathbf{U}=\mathbf{A}_{ab}\mathbf{X}_b\end{array}\right\}$$

Hence the estimator eqn. is:

$$\left.\begin{array}{l}\dot{\tilde{\mathbf{X}}}=\mathbf{A}\tilde{\mathbf{X}}+\mathbf{BU}+\mathbf{G}\left(\mathbf{Y}-\tilde{\mathbf{Y}}\right)\\ \tilde{\mathbf{Y}}=\mathbf{C}\tilde{\mathbf{X}}\end{array}\right\}\Rightarrow$$

$$\dot{\tilde{\mathbf{X}}}_b=\mathbf{A}_{bb}\tilde{\mathbf{X}}_b+\mathbf{A}_{ba}x_a+\mathbf{B}_b\mathbf{U}+\mathbf{G}\left(\dot{\mathbf{Y}}-\mathbf{A}_{aa}x_a-\mathbf{B}_a\mathbf{U}-\mathbf{A}_{ab}\tilde{\mathbf{X}}_b\right)$$

or: $\dot{\tilde{\mathbf{X}}}_b = (\mathbf{A}_{bb} - \mathbf{GA}_{ab})\tilde{\mathbf{X}}_b + (\mathbf{A}_{ba} - \mathbf{GA}_{aa})x_a + (\mathbf{B}_b - \mathbf{GB}_a)\mathbf{U} + \mathbf{G}\dot{\mathbf{Y}} \Leftrightarrow$

$$\dot{\tilde{\mathbf{X}}}_b = (\mathbf{A}_{bb} - \mathbf{GA}_{ab})\tilde{\mathbf{X}}_b + (\mathbf{A}_{ba} - \mathbf{GA}_{aa})\mathbf{Y} + (\mathbf{B}_b - \mathbf{GB}_a)\mathbf{U} + \mathbf{G}\dot{\mathbf{Y}}$$

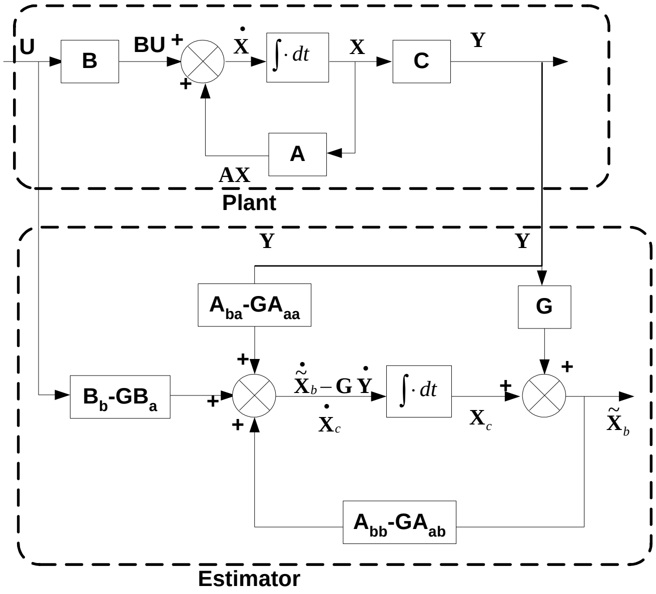
And hence the estimating error:

$$\dot{E}_b = (\mathbf{A}_{bb} - \mathbf{GA}_{ab})E_b$$

To avoid differentiating the **Y** we can define as a new state: $\mathbf{X}_c = \tilde{\mathbf{X}}_b - \mathbf{GY}$ hence the state eqn. is:

$$\dot{\tilde{\mathbf{X}}}_c = (\mathbf{A}_{bb} - \mathbf{GA}_{ab})\tilde{\mathbf{X}}_b + (\mathbf{A}_{ba} - \mathbf{GA}_{aa})\mathbf{Y} + (\mathbf{B}_b - \mathbf{GB}_a)\mathbf{U}$$

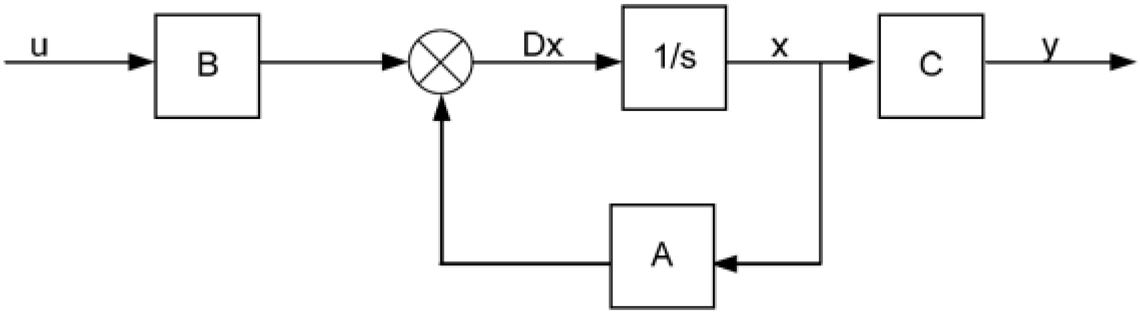
Notice that the term $\mathbf{G}\dot{\mathbf{Y}}$ is not at the LHS. The block diagram of this ROE is:



With this method the gain matrix G is found from:
G1=place(A'$_{bb}$,A'$_{ab}$,P$_e$)
G=G1'

Revision & Tracking

The block diagram of a state space system is:

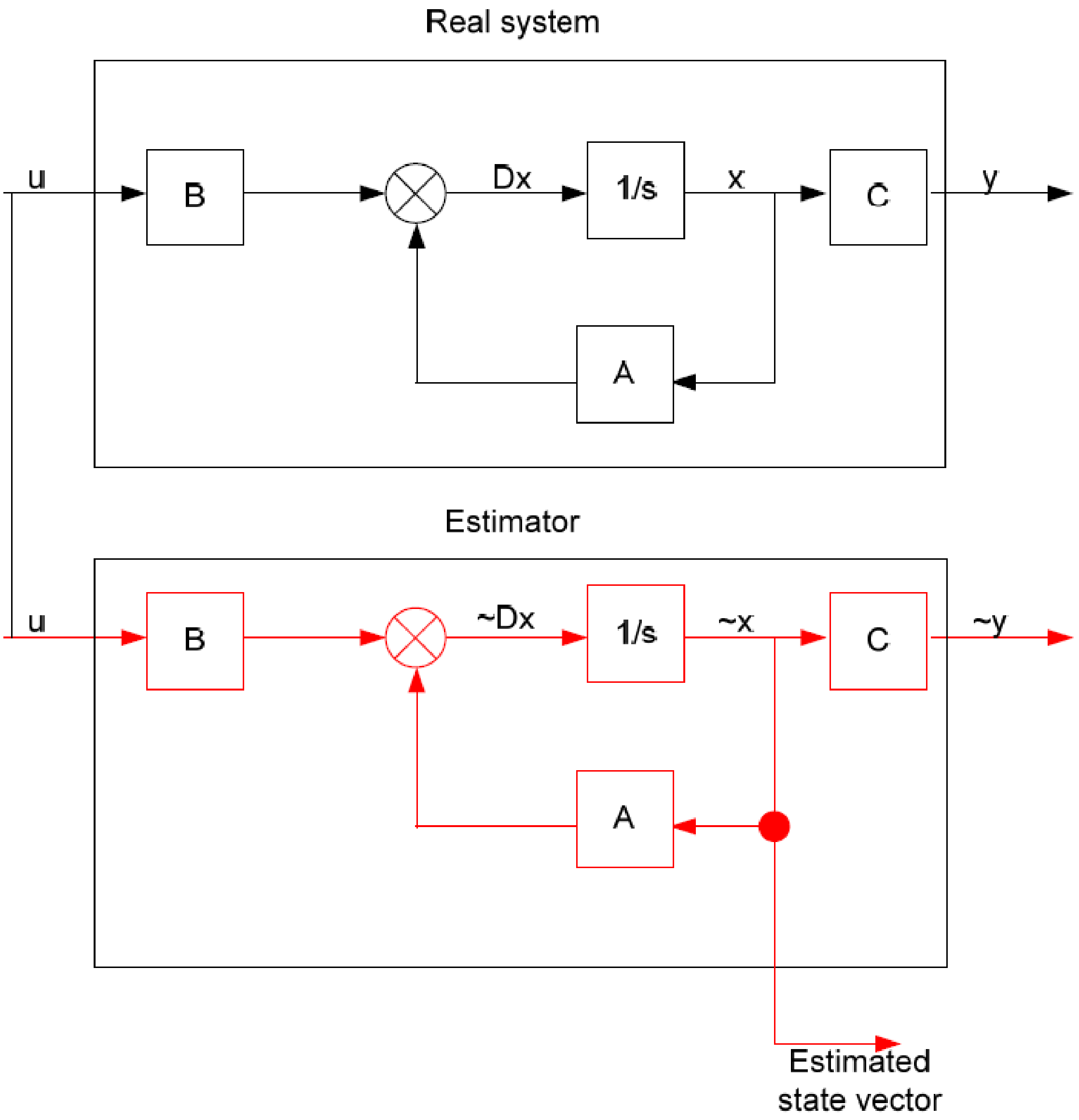


With equations: $\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}$, $\mathbf{y} = \mathbf{Cx}$

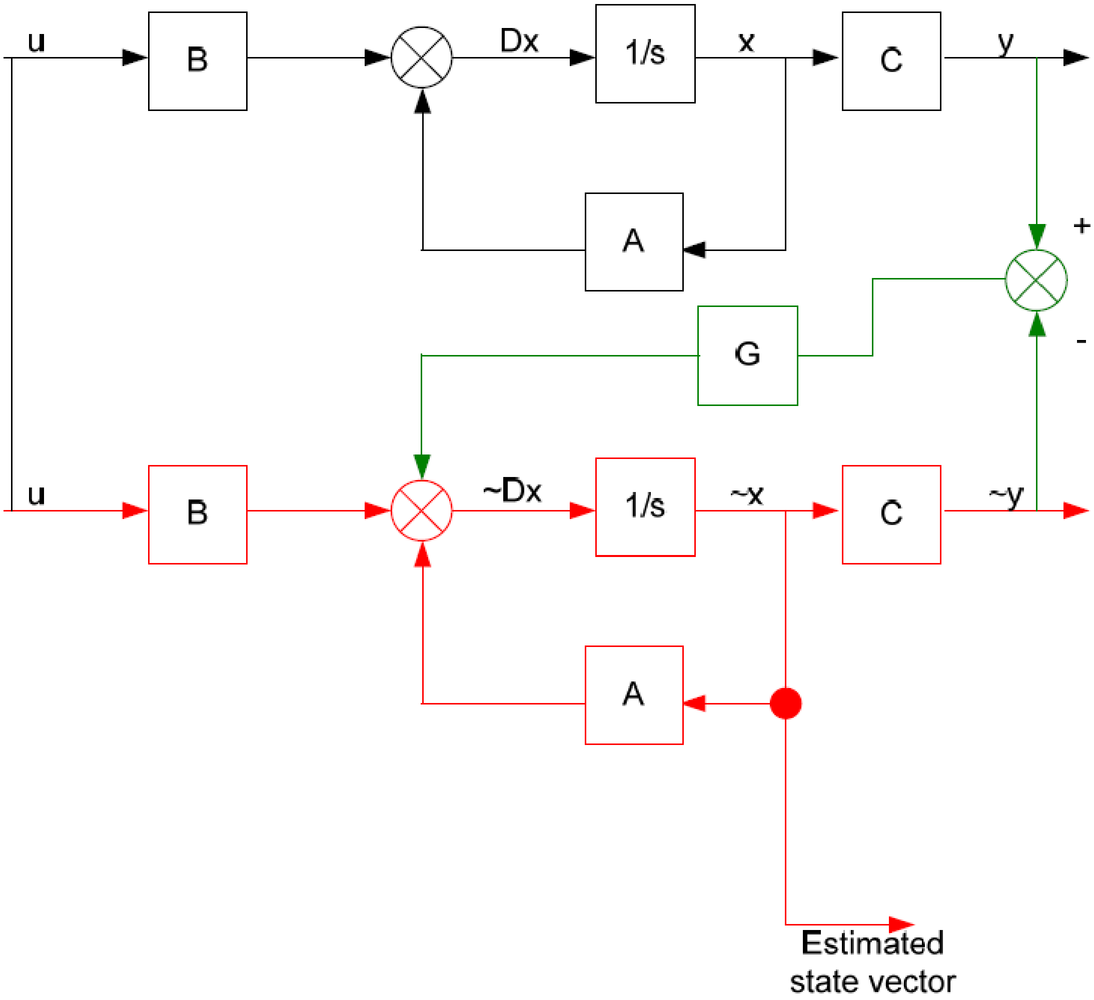
The qualitative properties of the system depend on the eigenvalues of **A**.

In reality we can only access/measure the output vector **y**. So unless the output matrix **C** is the identity matrix we need to estimate the state vector (either for monitoring or if we want to stabilise the system using a state feedback controller).

The simplest case is an open estimator, i.e. a mathematical model of the system. We assume that we have a linear system and we perfectly know the system's matrices (**A**, **B** and **C**):



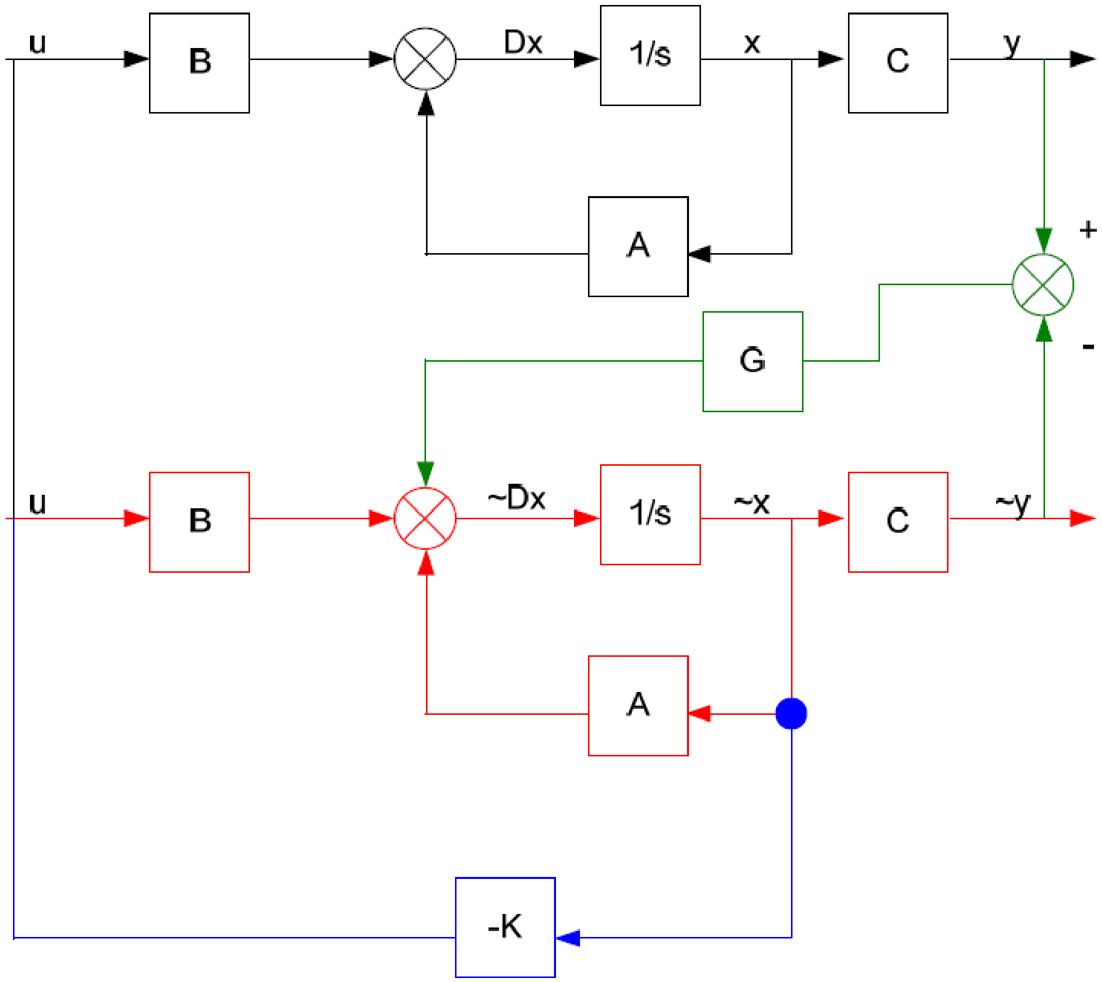
We have proved that the error dynamics between the actual state vector and the estimated state vector are given by: $\dot{\mathbf{e}} = \mathbf{A}\mathbf{e}$. Hence if $\mathbf{e}(0) \neq 0$ and the state matrix has unstable eigenvalues the error will diverge to infinity. To overcome this problem we create a closed loop estimator (the system under study is still open loop):



Then the error dynamics of the estimator will be given by: $\dot{\mathbf{e}} = (\mathbf{A} - \mathbf{GC})\mathbf{e}$

Note: If the system is unstable then the response of the estimator will also be "unstable" but at the same rate as the original system such as their difference converges to zero.

Now, we can design a pole placement controller to stabilise the system:

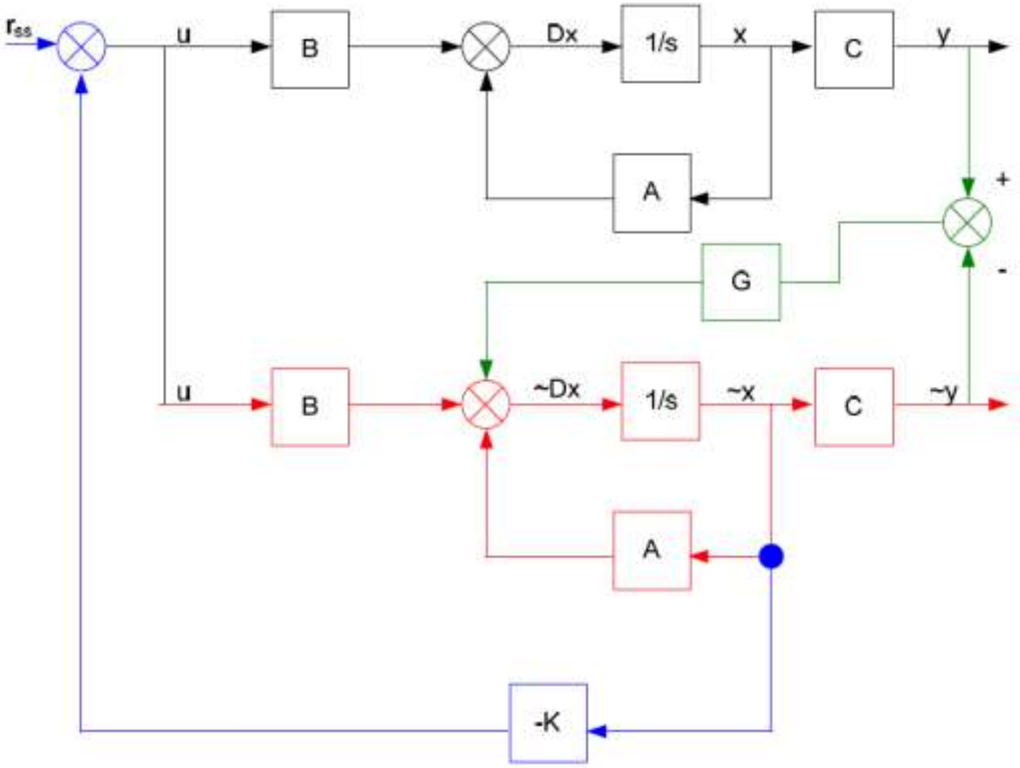


So in this case the estimator must be a lot faster than the original system’s dynamics (governed by **A-BK** and hence by **K**) so that the error between the actual state vector and estimated state vector will converge to zero very fast and hence the estimated state vector that is being used to control the actual system and the estimator (they have the same input) will be the correct one.

This means that the original system has dynamics: $\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{B}(-\mathbf{K}\tilde{\mathbf{x}})$ but if $\mathbf{x} \to \tilde{\mathbf{x}}$ very fast then we have $\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{BK})\mathbf{x}$

Note: To stabilise the system we can also use an LQR controller.

Now let’s assume that we want to converge to a nonzero value. If we just change the diagram to:



Then while the system will be stable (from the state feedback controller) it will be probably not converge to r_{ss}. To avoid that and assuming that we have a DC input (for SISO systems):

$$y \to y_{ss}, \mathbf{x} \to \mathbf{x}_{ss}, u \to u_{ss}$$

Thus we can say that if $u = u_{ss} - \mathbf{K}(\mathbf{x} - \mathbf{x}_{ss})$ then $u \to u_{ss}$ as $\mathbf{x} \to \mathbf{x}_{ss}$.

(the gain matrix K can be found by pole placement or LQR).

The state equation at the steady state is: $\left.\begin{aligned} 0 &= \mathbf{A}\mathbf{x}_{ss} + \mathbf{B}\mathbf{u}_{ss} \\ y_{ss} &= \mathbf{C}\mathbf{x}_{ss} \end{aligned}\right\}$

Now we can define 2 new vectors: $\mathbf{x}_{ss} = \mathbf{N}_x r_{ss}, u_{ss} = \mathbf{N}_u r_{ss}$ and hence our problem is transformed into find these 2 vectors.

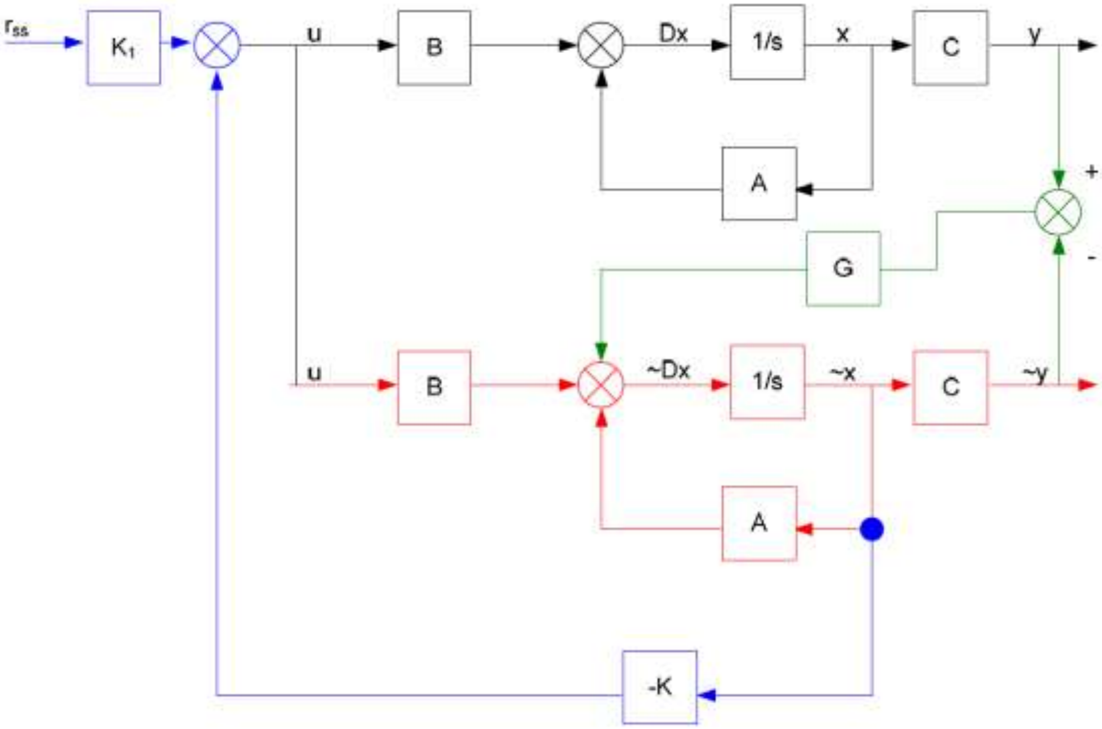
$$\left.\begin{array}{l} \mathbf{0} = \mathbf{A}\mathbf{x}_{ss} + \mathbf{B}u_{ss} \\ y_{ss} = \mathbf{C}\mathbf{x}_{ss} \end{array}\right\} \Rightarrow \left.\begin{array}{l} \mathbf{0} = \mathbf{A}\mathbf{N}_x r_{ss} + \mathbf{B}\mathbf{N}_u r_{ss} \\ r_{ss} = \mathbf{C}\mathbf{N}_x r_{ss} \end{array}\right\} \Rightarrow \left.\begin{array}{l} \mathbf{0} = \mathbf{A}\mathbf{N}_x + \mathbf{B}\mathbf{N}_u \\ 1 = \mathbf{C}\mathbf{N}_x \end{array}\right\} \Rightarrow$$

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{N}_x \\ \mathbf{N}_u \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{N}_x \\ \mathbf{N}_u \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}$$

Also

$$u = u_{ss} - \mathbf{K}(\mathbf{x} - \mathbf{x}_{ss}) = \mathbf{N}_u r_{ss} - \mathbf{K}(\mathbf{x} - \mathbf{N}_x r_{ss}) = -\mathbf{K}\mathbf{x} + (\mathbf{N}_u + \mathbf{K}\mathbf{N}_x) r_{ss} = -\mathbf{K}\mathbf{x} + K_1 r_{ss}$$

And thus the block diagram is:



Example:

Previously we saw that the necessary gains for the system:

$\dot{\mathbf{x}}(t)=\begin{bmatrix}1 & 2\\3 & 4\end{bmatrix}\mathbf{x}(t)+\begin{bmatrix}1\\0\end{bmatrix}\mathbf{u}(t),\ \mathbf{y}(t)=\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\mathbf{x}(t)$ are K=[26, 72]

If we assume that **C**=[1 0] then the matrix $\begin{bmatrix}\mathbf{A} & \mathbf{B}\\\mathbf{C} & \mathbf{0}\end{bmatrix}^{-1}$ is:

$$\begin{bmatrix}0 & 0 & 1\\0 & 0.25 & -0.75\\1 & -0.5 & 0.5\end{bmatrix}$$

Hence $\begin{bmatrix}\mathbf{N}_x\\\mathbf{N}_u\end{bmatrix}=\begin{bmatrix}\mathbf{A} & \mathbf{B}\\\mathbf{C} & \mathbf{0}\end{bmatrix}^{-1}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}0 & 0 & 1\\0 & 0.25 & -0.75\\1 & -0.5 & 0.5\end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}1\\-0.75\\0.5\end{bmatrix}$, therefore:

$$\mathbf{N}_x=\begin{bmatrix}1\\-0.75\end{bmatrix},\ \mathbf{N}_u=0.5.$$

$$K_1=(\mathbf{N}_u+\mathbf{K}\mathbf{N}_x)=0.5+\begin{bmatrix}26 & 72\end{bmatrix}\begin{bmatrix}1\\-0.75\end{bmatrix}=0.5+26-54=-27.5$$