Optimal control

Quadratic criterion

- System to be controlled:
  \[
  \begin{align*}
  z(k) &= G(q)u(k) + v(k), \\
  y(k) &= z(k) + e(k),
  \end{align*}
  \]
  \[
  \begin{align*}
  \text{constraints:} & \quad \begin{cases} 
  |u(k)| \leq C_u, \\
  |z(k)| \leq C_z.
  \end{cases}
  \end{align*}
  \]

- Minimizing the cost function
  \[
  V = \lim_{N \to \infty} \frac{1}{N} \sum_{k=0}^{N} [z^T(k)Q_1z(k) + u^T(k)Q_2u(k)],
  \]
  disregarding from the constraints, gives the LQG controller.

- Minimizing
  \[
  V = \sum_{k=0}^{N} [z^T(k)Q_1z(k) + u^T(k)Q_2u(k)], \quad N < \infty,
  \]
  accounting for the constraints, gives a nonlinear, time-varying controller, and in general no explicit solution.
The importance of constraints

An example

LQG control of a sampled double integrator,

\[ y(k) = \frac{0.5h^2(q + 1)}{(q - 1)^2} u(k). \]

Simulated with and without the constraint \(|u(k)| \leq 1\).

Sampling period \(h = 1\), weightings \(Q_1 = 64, Q_2 = 1\).

The output \(y\) to the left, the input \(u\) to the right.
Restating the problem formulation
Alternative/modified criterion

Consider the following system:

\[
\begin{align*}
  y(k) &= G(q)u(k) + v(k), \\
  y_m(k) &= y(k) + e(k),
\end{align*}
\]

constraints:

\[
\begin{align*}
  |u(k)| &\leq C_u, \\
  |y(k)| &\leq C_y.
\end{align*}
\]

Minimize the cost function

\[
V = \sum_{k=1}^{M} y^T(k)Q_1(k)y(k) + \sum_{k=0}^{N} u^T(k)Q_2(k)u(k), \quad N \leq M
\]

with respect to the input sequence \(u(0), \ldots, u(N)\), and

with \(u(N+1), \ldots, u(M)\) predefined (e.g. \(u(k) = u(N)\) for \(k > N\)).

The solution can be found by standard numerical optimization methods. However, this is open loop control rather than feedback...
Evaluating the cost function

- For the numerical optimization it must be possible to evaluate the cost function $V$.
- For this the future outputs $y(1), \ldots, y(M)$ are needed — these must be *predicted*!
- Use the model!

$$
\begin{align*}
x(k + 1) &= Fx(k) + Gu(k) + Nv_1(k), \\
y(k) &= Cx(k), \\
y_m(k) &= Cx(k) + v_2(k).
\end{align*}
$$

- The Kalman filter gives $\hat{x}(k|k)$, and the $m$-step predictor gives

$$
\hat{x}(k + m|k) = F^m \hat{x}(k|k) + \sum_{l=0}^{m-1} F^{m-1-l} Gu(k + l)
$$

$\Rightarrow \hat{y}(k + m|k) = C\hat{x}(k + m|k)$. 

hans.rosth@it.uu.se
Model Predictive Control

MPC combines this optimization technique with feedback:
At time $k$

1. Measure $y_m(k)$, use a Kalman filter to get $\hat{x}(k|k)$
2. Minimize the cost function

$$V_k = \sum_{l=k+1}^{k+M} \hat{y}^T(l|k)Q_y(l)\hat{y}(l|k) + \sum_{l=k}^{k+N} u^T(l)Q_u(l)u(l)$$

numerically (w.r.t. constraints) $\Rightarrow u(k), \ldots, u(k+N)$
3. Apply $u(k)$ on the system
4. Wait for the next sample instant, $k+1$, and go to step 1

This is also called *receding horizon optimization*. 
MPC: Prediction and control horizons

What the MPC controller “sees” at time $k$

- **Past**: $y_m$
- **Future**: $\hat{y}(k+j)$
- **Reference**: $y_m$
- **Control horizon**: $k$ to $k+N$
- **Prediction horizon**: $k+N$ to $k+M$
Constraints: LQG vs MPC

- MPC and LQG control of the sampled double integrator, with the constraint $|u(k)| \leq 1$.
- Design parameters: $Q_y = 64$, $Q_u = 1$ (for LQG and MPC), $M = 20$, $N = 5$ (for MPC).

- The output $y$ to the left, the input $u$ to the right.
MPC handles control constraints efficiently
Complex (big MIMO) systems are no problem
Flexible technique, e.g. nonlinear systems/models can be handled
There are a great number of industrial applications
There are many commercial MPC systems available
Computationally demanding
Hard to analyse
Design parameters: $h$, $M$, $N$, $Q_y$, $Q_u$, ...