# Systems Theory

## Laboratory 5: *Analogue and digital signal filtering methods. Kalman state observer for discrete-time systems.*

**Purpose of the exercise:**
Learning analogue and digital signal filtering methods, including Kalman state observer for discrete-time systems and LQG control, using MATLAB/Simulink environment.

### 1. Introduction

A filter is the primary block used in signal processing. Due to the type of signals processed, filters are divided into:

- Analogue – processing analogue signals (i.e. signals, which information parameter can take an infinitely large number of values).
- Digital – processing digital signals (a digital / binary signal information parameter can take a certain number / two value levels); types of digital filters:
  - Finite Impulse Response (FIR),
  - Infinite Impulse Response (IIR),
  - Moving Average,
  - Adaptive Filters.

Due to the energy demand needed to operate, filters are divided into:

- Passive – they do not need an additional external energy source for proper operation (are based on input signal energy only), e.g. simple RC or LC circuits,
- Active – they need an additional external power source for proper operation (they use signal amplifiers), e.g. operational amplifier circuits.

Due to the range of frequencies passed through and blocked by the filter, they can be divided into (see Fig. 1):

- Lowpass – filters that transmit signals with a frequency lower than the threshold and suppress signals with a higher frequency.
- Highpass – filters that transmit signals with a frequency higher than the threshold frequency and suppress signals with a lower frequency.
- Bandpass – filters that transmit signals within a certain frequency range and suppress signals with frequencies outside that range.
- Bandstop – filters that suppress signals from a certain frequency range and transmit signals with frequencies outside that range.

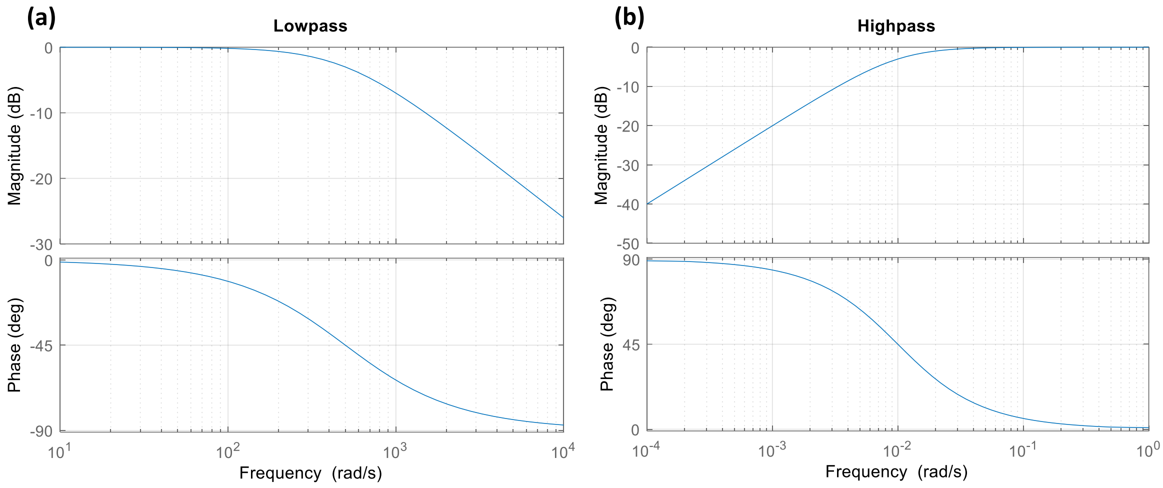



Fig. 1. Bode diagrams of 1st order (e.g. single RC passive) filters: (a) lowpass, (b) highpass

## 2. Filter designs and characteristics

*Chebyshev* filter

Chebyshev filters are used when the most important criterion is high attenuation in the stopband, and at the same time magnitude change in the passband is allowed. Their amplitude-frequency characteristics are characterised by ripples, and a very sharp roll off (the sharper roll off, the greater ripples, associated with an increase of the filter order).

- The magnitude response of a Chebyshev type I filter is equiripple in the passband and monotonic in the stopband (Fig. 2(a)). The magnitude response of a Chebyshev type II filter is monotonic in the passband and equiripple in the stopband (Fig. 2(b)).

*Butterworth* filter

Butterworth filters are used when the most important criterion is that the amplitude-frequency characteristics in the passband are as flat as possible with sharp roll off at the cut-off frequency.

- The magnitude response of a Butterworth filter is flat in the passband and monotonic overall (Fig. 2(c)).

*Bessel* filter

Bessel filters are used where the most important criterion is the most accurate representation of the signal waveform in the passband, which is achieved for the possibly flat characteristics of group delay. Amplitude-frequency characteristics is flat in the passband, while its roll off sharpness is greater than for simple RC filters. This is a preferred filter for rectangular pulses thanks to meeting the condition of constant of group delay over a wide frequency range (phase shift proportional to frequency).

- The magnitude response of a Bessel filter is possibly flat in the passband and monotonic overall. The filter has a linear phase response (Fig. 2(d)).

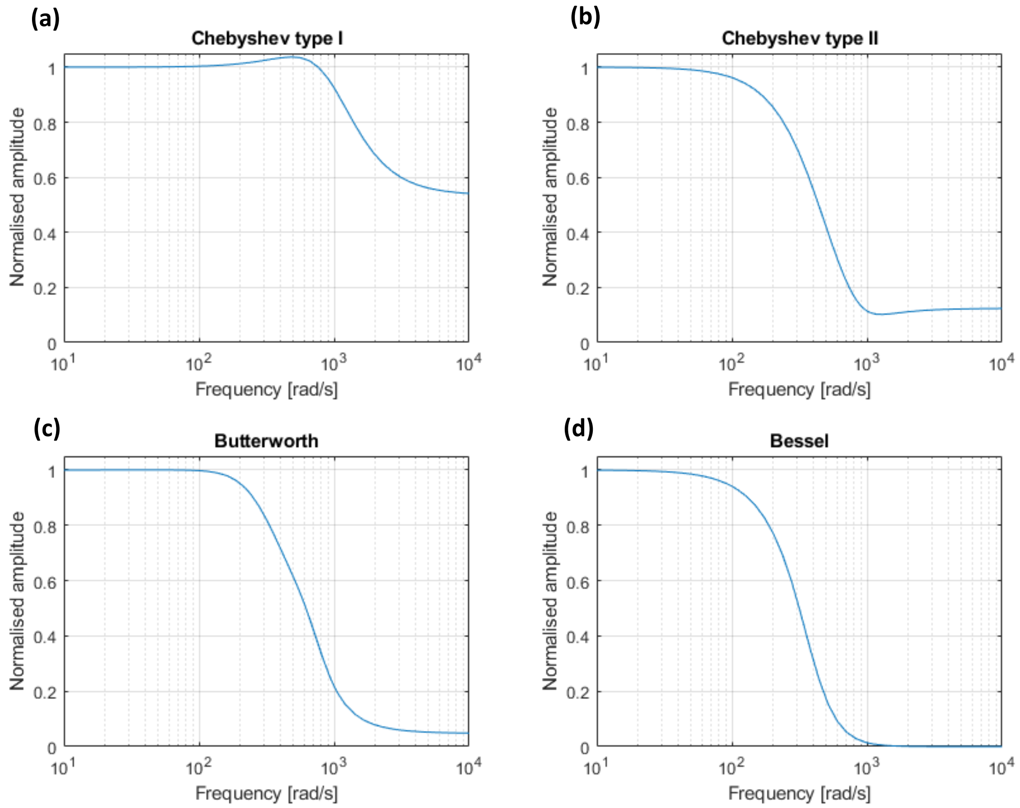



Fig. 2. Amplitude-frequency characteristics of 4th order filters:
a) Chebyshev type I, b) Chebyshev type II, c) Butterworth, d) Bessel

## 3. Analogue vs. digital filters

*Analogue filters* characteristic features:

- Soft, indistinct transition between the passband and the stopband
- Non-linear phase
- Less accuracy due to component tolerances
- Drift due to component variations
- Difficult to simulate and design
- Adaptive filters difficult to implement
- Analog filters are required at high frequencies and for anti-aliasing filters
- No A/D and D/A converters nor DSP required

*Digital filters* characteristic features:

- High accuracy
- Linear phase (FIR filters)
- No drift due to component variations

- High resistance to disturbances and amplitude distortions
- Flexible, adaptive filtering possible
- Easy to simulate and design
- The computation must be completed within the sampling period – this limits the real-time operation
- Require high performance A/D and D/A converters along with DSP to operate in real-time (the requirements depending on the sampling frequency and filter complexity)

Assuming a bandwidth $f_a$ of the analogue signal to be processed, the A/D sampling frequency $f_s \geq 2f_a$ is required (Nyquist-Shannon condition), thus the sampling period is $1/f_s$. All digital filter computations (including overhead) must be completed during this sampling interval. The computation time depends on the number of taps (i.e. filter length – number of input samples processed) in the filter structure and the speed/efficiency of the DSP. Each tap on the filter requires one multiplication and one addition (multiply-accumulate).

There are two fundamental types of digital filters: finite impulse response (FIR) and infinite impulse response (IIR). By varying the weight of the coefficients and the number of taps, virtually any frequency response characteristic can be realised with an FIR filter. FIR filters can achieve performance levels which are not possible with analogue filter techniques (such as perfect linear phase response). However, high performance FIR filters generally require a large number of multiply-accumulates and therefore require fast and efficient DSPs. On the other hand, IIR filters may mimic the performance of traditional analogue filters and make use of feedback. Therefore their impulse response extends over an infinite period of time. Because of feedback, IIR filters can be implemented with fewer coefficients than for an FIR filter. The digital filters may be used in adaptive filtering applications due to their speed and ease with which the filter characteristics can be altered by varying the filter coefficients.

FIR filters features:

- Linear phase
- Easy to design
- Computationally intensive

IIR filters features:

- Based on classical analogue filters
- Computationally efficient

*General advantages of analogue circuits over digital techniques*:

- Speed – even simple operational amplifiers can operate at 100 kHz to 1 MHz,
- Amplitude dynamic range – the ratio between the largest signal that can be passed through, and the inherent noise of the system; e.g. a 12 bit A/D converter yields a dynamic range of about 14000; in comparison, a standard operational amplifier has a dynamic range of about ten million,
- Frequency dynamic range – an operational amplifier circuit can handle frequencies between 0.01 Hz and 100 kHz (7 decades); in digital techniques this would require $f_s$ of 200 kHz, thus 20 million time samples are needed to capture one complete cycle at 0.01 Hz. The digital filters often use a linear frequency scale to show their performance, while analogue filters need the logarithmic scale to show their large dynamic range.

*General advantages of digital techniques over analogue circuits*:

- Passband flatness achievable with analogue filters is limited by the accuracy of the resistors and capacitors. Even for a Butterworth filter design (0% ripple), a residual ripple of ca. 1% is expected. The flatness of digital filters is primarily limited by round-off error, making them hundreds of times flatter than their analogue counterparts.
- The digital filters are preferable with regard to both roll-off and stopband attenuation. Even if the analogue performance is improved by adding additional stages, it still can't compare to the digital filter.

In MATLAB/Simulink environment, *Analog Filter Design* block may be used to design an analogue filter of the demanded type, order, passband edge frequency and passband ripple, while *Digital Filter Design* block is intended for both FIR and IIR filters design, analysis, and realisation according to the demanded characteristics.

## 4. Kalman filter (state-observer)

The Kalman filter (Kalman state-observer) is used to restore unmeasurable states. This method provides a full state estimation, considering the measurement and process noises. Consider the following system:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{A}_k\boldsymbol{x}(k) + \boldsymbol{B}_k u(k) + \boldsymbol{\vartheta}(k) \\ y(k) &= \boldsymbol{H}_k\boldsymbol{x}(k) + \psi(k) \end{aligned} \tag{1}$$

where: $\boldsymbol{x} = [q \;\; v \;\; a\,]^T$ is a state vector that includes: displacement $q$ ($q_1$ or $q_2$), velocity $v$ ($v_1$ or $v_2$), and acceleration $a$ ($a_1$ or $a_2$, respectively; acceleration is estimated for other applications), while $\boldsymbol{\vartheta}(k)$ and $\psi(k)$ are independent process and measurement (respectively) white gaussian noises (i.e. random signals with constant power spectral density and normal/gaussian probability density distribution) satisfying:

$$E[\boldsymbol{\vartheta}(k)] = \boldsymbol{0}, \quad E[\psi(k)] = 0, \quad E[\boldsymbol{\vartheta}(k)\boldsymbol{\vartheta}^T(k)] = \boldsymbol{Q}_k, \quad E[\psi(k)\psi^T(k)] = R_k,$$

where $\boldsymbol{Q}_k$, $R_k$ are covariance matrices.

Assuming that only displacements are being measured, i.e. $y(k) = q(k)$ ($q_1(k)$ or $q_2(k)$) is the measured displacement value at a $k^{\text{th}}$ time step, the following matrices $\boldsymbol{A}_k$, $\boldsymbol{B}_k$, and $\boldsymbol{H}_k$ of the equation set (1) are considered:

$$\boldsymbol{A}_k = \begin{bmatrix} 1 & T_0 & T_0^2/2 \\ 0 & 1 & T_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{B}_k = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \boldsymbol{H}_k = [1 \quad 0 \quad 0],$$

where $T_0$ is the sampling (discretisation) period. For the calculation purposes, the following values of the covariance matrices $\boldsymbol{Q}_k$, $R_k$ are assumed ($\nu$, $r$ are tuning constants):

$$\boldsymbol{Q}_k = \nu \begin{bmatrix} T_0^5/20 & T_0^4/8 & T_0^3/6 \\ T_0^4/8 & T_0^3/3 & T_0^2/2 \\ T_0^3/6 & T_0^2/2 & T_0 \end{bmatrix}, \quad R_k = [r].$$

The considered Kalman filter algorithm consists of the two basic steps: *prediction* and *correction*.

Prediction step:

$\hat{\boldsymbol{x}}_k^- = \boldsymbol{A}_k \hat{\boldsymbol{x}}_k$ – predicted value of the state $\boldsymbol{x}$,
$\boldsymbol{P}_k^- = \boldsymbol{A}_k \boldsymbol{P}_k \boldsymbol{A}_k^T + \boldsymbol{Q}_k$ – predicted value of the error covariance: $\boldsymbol{P}_k = E[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k)^T]$.

Correction step:

$\boldsymbol{K}_k = \boldsymbol{P}_k^- \boldsymbol{H}_k^T (\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + R_k)^{-1}$ – gain of the Kalman filter,
$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k [q\,(k) - \boldsymbol{H}_k \hat{\boldsymbol{x}}_k^-]$ – optimal, estimated value of the state $\boldsymbol{x}$,
$\boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k)\boldsymbol{P}_k^-$ – optimal, estimated value of the covariance ($\boldsymbol{I}$ is the identity matrix).

The above algorithm may be implemented in the form of a Simulink diagram. Figs. 3 and 4 present comparison of the $q_1$ displacements and $v_1$ velocities time responses of the tower-nacelle system, determined from the experiment and estimated by the Kalman filter. The estimated velocity $v_1$ was compared to the one calculated by simple differentiation of $q_1$ (Euler method). Regarding the displacements, time responses practically coincide (Fig. 3). Analysis of the velocity patterns (Fig. 4) shows the advantage of the Kalman filter over the simple differentiation calculation method.

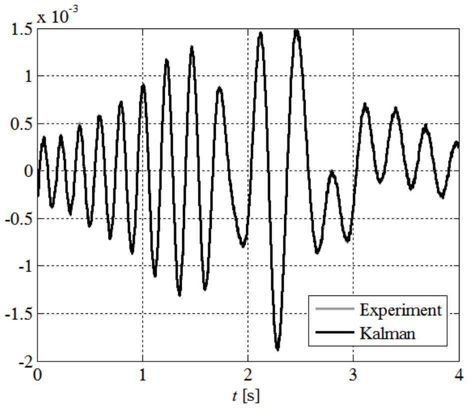



Fig. 3. Comparison of the $q_1$ [m] time responses

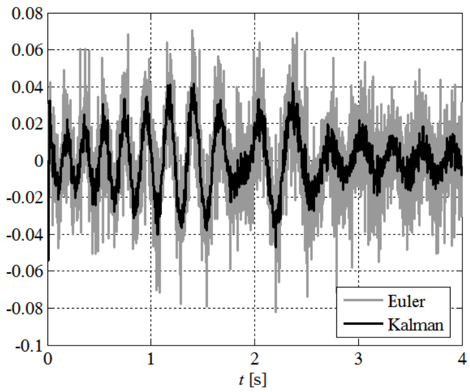



Fig. 4. Comparison of the $v_1$ [m/s] time responses

## 5. Tasks

1. build *Simulink* model of the Kalman filter according to *section 4*; assume: $T_0$=1 ms, $v$=4, $r$=$10^{-9}$, and zero initial conditions for $\boldsymbol{x}$ and $\boldsymbol{P}_k$;
2. build *Simulink* model of the lowpass 1st order inertial filter of 500 [rad/s] cut-off angular frequency (at –3dB magnitude), and the highpass 1st order differential filter with inertia of 1.0 [rad/s] cut-off angular frequency (at –3dB magnitude) – see *Appendix1*;
3. Using *From Workspace* blocks, input *ScopeData* displacement ($q_1$) and *ScopeData1* minus acceleration ($-a_1$) measurement signals from files available at: https://dysk.agh.edu.pl/s/2mBDLwrPLHX5YCZ
4. determine velocity $v_1$ time pattern using both displacement $q_1$ and acceleration $a_1$ signals, and *Derivative* (or *Discrete Derivative*) and *Integrator* (or *Discrete-Time Integrator*) blocks, respectively, without any filters, and with lowpass (along with differentiation) and highpass (along with integration) continuous or discrete-time filters versions build in *Task 2*;
5. determine velocity $v_1$ time pattern using the Kalman filter build in *Task 1* and displacement $q_1$ signal; compare all the obtained $v_1$ time patterns.

*Appendix1: Real-time signal processing*

Two of the most frequent real-time signal processing challenges are noise attenuation and value drift.

Measurement and process noises are intrinsic phenomena present in real-world measurement-control systems. These noises are associated with imperfectness of real-world structures/systems and measurement sensors/conditioners as well as external disturbances (e.g. electromagnetic or acoustic noise, vibration), quantisation, etc. To remove noise content from the measurement signal, lowpass filters are used. In real-time control implementation, a close-to-zero phase shift of the measured signal is one of the major elements of control efficiency; therefore, simple low order filters are used as, e.g. $1^{st}$ / $2^{nd}$ order inertial elements (RC circuits) of a time constant tuned to the demanded cut-off frequency (Fig. 1(a)).

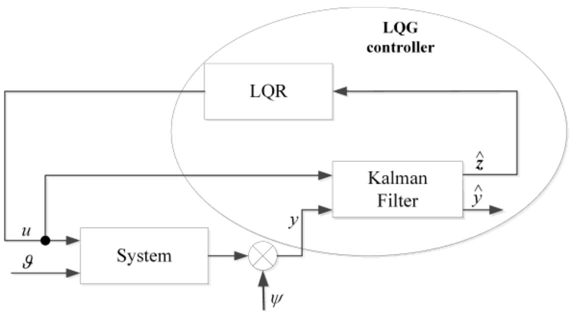
Signal value drift phenomena are often present when piezoelectric transducers are used or signal integration performed. Piezoelectric effect yields constant electric charge of the crystal element used under static stress. However, due to nonzero resistance, its discharge occurs over time; thus, output signal low frequency drift. Signal integration (e.g. to obtain velocity based on acceleration measurement), on the other hand, is associated with integration constant, which added at every sample step yields mean value drift. Again, as a close-to-zero phase shift of signals utilised in real-time control systems is crucial, simple low order filters are used as, e.g. $1^{st}$ order differential elements with inertia (RC circuits) of a time constant tuned to the demanded cut-off frequency (Fig. 1(b)).

Both of the above real-time signal processing challenges may be addressed using a Kalman filter (see *section 4*).

*Appendix2: LQG controller synthesis*

The LQG (Linear-Quadratic-Gaussian) controllers are built for uncertain linear systems disturbed by additive white gaussian noises, having incomplete state information. The LQG is a combination of the Kalman filter (Kalman state-observer) with a Linear-Quadratic Regulator (LQR) – see Fig. 5. The LQR problem is defined for the system dynamics described by a set of linear differential (continuous-time) / difference (discrete-time) equations and a quadratic cost function. The discrete-time LQR (DLQR) optimisation problem may be solved using *dlqr.m* function (or *dlqry.m* with output weighting) from *MATLAB/Simulink Optimization toolbox*.

The *separation principle* allows that each of these two parts of the LQG can be designed and tested independently. LQG controller may be applied to both linear time-invariant and linear time-varying systems. It should be noted that the LQG control problem is one of the most fundamental problems of optimal control. Application of a Kalman filter enables to restore unmeasured state variables and then use them in the state-feedback LQR controller. A typical structure of the LQG controller is shown in Fig. 5.




$u$ – control input of the system,
$\vartheta$ – system noise (stochastic),
$\psi$ – measurement noise (stochastic),
$y$ – output of the system,
$\hat{y}$ – estimation of the system output,
$\hat{z}$ – estimation of the system state.

Fig. 5. Structure of the LQG controller

The description of the LQG controller focuses on the following discrete-time linear system of equations:

$$\boldsymbol{z}(k+1) = \boldsymbol{A}_d\boldsymbol{z}(k) + \boldsymbol{B}_d u(k) + \boldsymbol{\vartheta}(k)$$
$$y(k) = \boldsymbol{C}_d\boldsymbol{z}(k) + \boldsymbol{D}_d u(k) + \psi(k)$$

where $\boldsymbol{z}$ is a state vector, while process and measurement noises, respectively: $\vartheta(k)$ and $\psi(k)$ are independent, zero mean, white gaussian random processes.

The LQG control algorithm can be employed for e.g. semi-active vibration control of the wind turbine tower-nacelle system with MR TVA, assuming $u=P_{MR}$ (MR damper force) as a control input – see *Lab. 1* and DLQR implementation in *Lab. 4*. Using this algorithm, the optimal control signal $P_{MR}$, which is the force generated by an actuator (MR damper) is obtained. To induce the actuator (MR damper) to generate the desired optimal control force, the force tracking / inverse model of the MR damper is used. As $q_1$ and $q_2$ displacements of the tower-nacelle system are measured only, the full state vector: $\boldsymbol{z} = [z_1 \quad z_2 \quad z_3 \quad z_4]^T$ with $z_1 = q_1$, $z_2 = v_1$, $z_3 = q_2$, $z_4 = v_2$ is not accessible for the DLQR controller. To solve this problem, the state is replaced in the control law by an optimal state *estimate* generated by a *Kalman filter*.

The LQR controller and Kalman filter are integrated, forming the LQG controller. The integration stage is executed according to the scheme shown in Fig. *A1*.

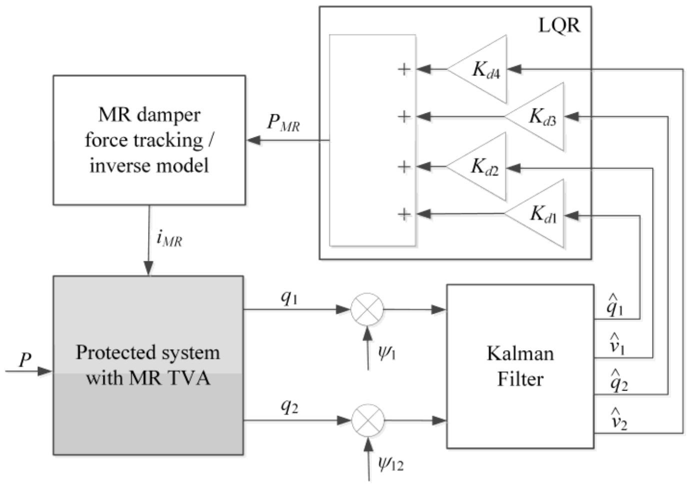



Fig. *A1*. Structure of the integrated LQG controller

Most of the applications of the LQG controller concern control of the civil structures (buildings) excited by severe earthquakes or strong winds. The existing solutions of the LQG semiactive control algorithm use, most frequently, the mathematical model of the analysed mechanical structure. In opposition to the Linear-Quadratic-Regulator (LQR) algorithm, they do not need a measurement of the full-state for all DOFs. The Kalman state observer is responsible for the estimation of unmeasurable state variables, based on the measured positions or accelerations. In many cases, obtaining sufficiently accurate model is difficult, therefore a model-free LQG control is also used, computing the LQG parameters directly from the measurement data. Output feedback strategy based on measured acceleration (or position) at limited number of structure points may be realised. The LQG controller calculates demanded actuator (e.g. MR damper) force on the basis of state variables vector restored by the Kalman filter. The LQR / DLQR problem is solved using a linearised model, considering two state variables (displacement $q$ and velocity $v$) for each vibrating body ($m_1$ and $m_2$, see Fig. 1, *Lab.*

4) – in this LQG design two Kalman filters of the same structure may be used: one for $v_1$ (and possibly $a_1$) estimation based on $q_1$ measurement, the other for $v_2$ ($a_2$) estimation based on $q_2$.

**References:**


[1] M. Athans "Special Issue on the LQG Problem", IEEE Transaction on Automatic Control, Vol. 16, No. 6, December 1971.

[2] T. Singhal, A. Harit, D.N. Vishwakarma "Kalman Filter Implementation on an Accelerometer sensor data for three state estimation of a dynamic system", IJRET, Vol. 1, No. 6, 2012.

[3] M. Rosół, P. Martynowicz "Implementation of the LQG controller for a wind turbine tower-nacelle model with an MR tuned vibration absorber", Journal of Theoretical and Applied Mechanics, Vol. 54, No. 4, 2016.

[4] A.D. Ioffe, V.M. Tihomirov „Theory of extremal problems. Studies in mathematics and its applications", Amsterdam–New York–Oxford: North-Holland Publishing Company, 1979.

[5] D.S. Naidu „Optimal Control Systems", CRC Press, 2003.

[6] Digital Filters - Analog Devices, https://www.analog.com/media/en/training-seminars/design-handbooks/MixedSignal_Sect6.pdf

[7] S.W. Smith "The Scientist and Engineer's Guide to Digital Signal Processing", copyright ©1997-1998 by Steven W. Smith. For more information visit the book's website at: www.DSPguide.com

[8] MATLAB/Simulink documentation: http://www.mathworks.com.