Holistic multidisciplinary method for optimisation of mechatronic systems

Fariba Rahimi

Department Machine Design,
KTH Royal Institute of Technology,
Stockholm, Sweden
Email: frahimi@kth.se

Abstract: Design method for mechatronic systems relies on multidisciplinary, collaborative research, and information sharing between engineers from different domains. However, there are many hurdles for information sharing such as expensive cost to provide required tools. Therefore, a holistic co-design method is needed to integrate these domains concurrently and facilitate designs, and verification. In this paper, a method based on holistic approaches is presented. A design case including a DC-motor, a belt drive and a load component is considered. An observer-based optimal pole placement control design is employed. The system is optimised for three objectives as volume, sampling frequency, and sensor resolution to reduce size, hardware and implementation costs. A weighted sum function is applied on the result of optimisation as a Pareto optimal set to provide even weights for the objectives. The optimum system is derived, control performance is evaluated, and the correlations between parameters from different domains are discussed.

Keywords: holistic design; multidisciplinary; information sharing; integrated design.

Reference to this paper should be made as follows: Rahimi, F. (2021) 'Holistic multidisciplinary method for optimisation of mechatronic systems', *Int. J. Mechatronics and Automation*, Vol. 8, No. 2, pp.80–91.

Biographical notes: Fariba Rahimi is a PhD researcher in Machine Design Department at the KTH Royal Institute of Technology. She has been supervising Master and Bachelor thesis in mechatronics and involved in teaching mechatronics course. Her research interests are related to mechatronic systems design, control design, optimisation, and robotic systems. She has published research papers in both conference proceedings and international journals.

This paper is a revised and expanded version of a paper entitled 'Multi-criteria co-design optimization of mechatronic systems' presented at IEEE International Conference on Mechatronics and Automation (ICMA), Beijing, China, October 2020.

1 Introduction

Design optimisation of mechatronic systems is a cumbersome task that requires engineers from different disciplines to work together and share information. As the information about design process increases, the freedom for engineers to use the information decreases (Ullman, 2003) (shown in Figure 1).

Since, integration of several tools to consider design aspects from different domains is highly cost inefficient, a multidisciplinary design method with a supported software framework considering cross-domain synergies and targeted product properties can be useful. This paper extends methodology and supported software framework as integrated design optimisation of mechatronic systems (IDIOM) that was introduced in previous studies by Frede et al. (2013), Rahimi et al. (2017) and Rahimi and Wikander (2020) on developing a holistic co-design optimisation to cover embedded control implementation impact on structural design. A design case including a DC-motor, a belt drive and a load component is modelled and a physical design approach is developed for the belt component. A multi-objective multidisciplinary method is employed to optimise this conceptual system. The system model is tried to be simple enough for time efficient evaluations while it captures important system behaviour and properties. The aim with presented method and design case is to broaden use cases of the approach and to evaluate the application of extended framework to a more complex system. The co-design steps in our method are as follows:

1 a conceptual mechatronic system is considered
2 the system concept is configured by drag and drop of components from the library of IDIOM framework
3 system specification and requirements are provided

4 mechanical/electrical components are designed and dimensioned with respect to the imposed internal physical constraints and cost functions

5 dynamics of the system is symbolically derived by Mathematica solver

6 if the system is nonlinear, it is linearised at specified operational points

7 the controller is designed for the evaluated open loop system

8 the system is discretised using calculated sampling time

9 control constraints are evaluated by substituting the numerics of physical design parameters into the symbolic system

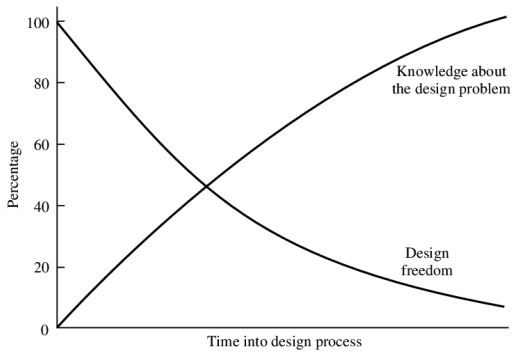
10 if the constrains are all satisfied, then the possible optimal solutions are derived

11 the optimisation is terminated when the generation and population size are achieved

12 a weighted sum function is applied on the Pareto-front set and normalised objectives to obtain the final optimum solution.

In our method, the requirement on a conceptual system is a position profile that is connected to a mass/inertia body called a load component which impose requirement as force/torque that should be handled by the system. Component models are dimensioned/designed for this type of a profile while satisfying physical and control constraints. The objectives of optimiser can be size, energy, cost, efficiency, reliability, safety, and hardware or implementation expenses. This approach assists in designing a proper controller with optimised configuration by treating engineering domains, simultaneously.

Figure 1 The design process



Source: Ullman (2003)

The outline of the paper is as follows: Section 1 is an introduction of the presented method and contribution. Section 2 briefly presents related work and the existing gaps. Section 3 includes the applied method. Section 4 presents physical models relevant to a design case. Section 5 is a presentation of physical design constraints. Section 6 describes the control method. Section 7 presents multi-objective optimisation problem, and the achieved results, and discussions. The paper is concluded in Section 8.

2 Background

The value of integrated design approaches for mechatronic systems has been recognised especially for systems with couplings between physical and control system design (Youcef-Toumi, 1996). A method for designing mechanical structures considering dynamic specifications is presented by Asada et al. (1991). They have applied their method on a flexible robot arms and located the poles and zeros by modifying geometry of the arms. Inverse mapping approach is employed in their method to derive the mentioned geometry. Ravichandran et al. (2006) used co-design to minimise energy of a counterbalanced manipulator, and employed gravity balance as a proxy objective, which resulted in a modified sequential process. Park and Asada (1994) used co-design on a manipulator and employed simplified system performance metrics, like settling time.

Allison (2012) applied plant-limited co-design (PLCD) method on a robotic manipulator. PLCD method is usually used on an existing system to modify control or physical design at minimum cost and attempts to meet new specifications and requirements. The new requirement in their method for this design case was defined to minimise energy consumption for a pick and place task. Subsequently, a formal method to identify plant modifications was introduced by Allison (2013). In their method, a co-design approach was used to minimise the cost of plant modification, as PLCD.

O'Driscoll (2002) used design for manufacture (DFM) method to address the coupling between product design and manufacturing by considering manufacturing needs during product development. DFM is to design products by taking manufacturing issues into account. Early consideration of manufacturing related requirements shortens the product development cycle time, minimises the cost for development, and ensures a smooth transition into production. Jain and Tsiotras (2008) presented a multi-resolution-based approach for direct trajectory optimisation. Their method generated a non-uniform grid automatically and increased numerical efficiency and robustness which assisted in handling control and state constraints without additional computational complexity.

Kurdi and Beran (2008) used temporal spectral element method for discretisation of time-dependent differential equation and optimisation of a dynamic system by computing monolithic-time responses and constraints. Becerra (2010) introduced PSOPT (open source optimal control solver written in C++), which is used for the solution of complex optimal control problems. They used an illustrative example to explain features and capabilities of this software. Trivedi et al. (2008) proposed an optimal design method for OctArm manipulators which involves specifications of air muscle actuators and configuration

of sections to maximise dexterity and load capacity for a given actuation pressure. In their method, after generating design rules for the optimisation problem, optimal solutions for pneumatic and hydraulically actuated soft robotic manipulators were obtained. Their method addressed trade-off in soft manipulator design between dexterity and strength.

Prevalent design processes are usually based on sequential steps, where the physical system is designed first and followed by a controller design which reduces flexibility and has costly iterations in the case of errors appearing in the system. Therefore, in order to achieve a true optimal design, exploiting synergies between physical and control design decisions is required. Designing mechatronic systems including mechanical properties and dynamic response involves constraints that need to be satisfied over a time interval where the objective functions should also be evaluated over this interval. Efficiently designing a constrained optimisation is challenging, since the responses are implicitly correlated with the design variables. In this paper, a method for holistic co-design optimisation of mechatronic systems is presented which considers physical structure, control and implementation factors, simultaneously, and results in a time efficient optimum solution. The relevant studies in the literature ignore the impact of control implementation on the physical design and lack integration of several domains in an early stage of design. In this work, a design case is considered and sampled. A sensor quantisation model is used and the impact of design variables on the optimisation objectives is discussed.

3 Method

The method for system evaluation together with the pseudo code is presented in Algorithm 1. For an arbitrary system composed of K physical components, the method goes through a few steps as shown in Algorithm 1 to evaluate the optimum solution. The inputs for this algorithm are the system concept (G_p), design parameters ($\mathbf{p}^k$), design variables ($\mathbf{v}^k$) and their ranges (D^k). The output of this algorithm is a multidisciplinary optimum solution (f).

In lines 1 and 2, the static load transfer model (St_k) is executed for all components except the load component (K) to evaluate physical dimension model (Pd_k), in line 3, for design purposes. Static transfer model captures static transformation of the load profile (position/torque) from one side of a component to the other. These models are needed for initial physical dimensioning of the components. Regarding the component type, the physical dimension is based on load properties such as RMS (root mean square) and peak values of the given load (motion and torque profiles). The physical dimension model captures load carrying capacity of the component as a function of component properties such as size, energy or cost (Rahimi et al., 2017).

In line 4, the design answers ($\mathbf{As}^k$) are calculated in component level which are related to the targeted optimisation criteria. If the physical design constraints are satisfied, the design answers ($\mathbf{As}^k$) are real values, so the algorithm goes through dynamic (Dy_k) evaluation in lines 6–11. Therefore, Wolfram Mathematica solver starts to evaluate open loop system model (Tf_{op} or SS_{op}). For nonlinear systems, the model is linearised and sent back to MATLAB in line 9. The closed loop system (C_{cl}) is constructed in line 10 and the control performance constraints (cpc_j) for $j = 1, ..., \gamma$ are evaluated in line 10. If the constraints are in a defined boundary (b_j), then the final optimisation objectives are calculated. This iteration is repeated until the generation and population size for the optimiser are achieved. A weighted sum function is applied on the result which is in a form of Pareto front set to derive the final optimum solution (f) for the system.

Algorithm 1 Dynamic system configuration (DSC)

[min f] = **DSC**(G_p, $\mathbf{p}^k$, D^k, $\mathbf{v}^k$)
Require: system concept (G_p), design parameters ($\mathbf{p}^k$), range for design variables (D^k), design variables ($\mathbf{v}^k$);
Ensure: min f;
1: Repeat
2: Execute St_k, for $k = 1, ..., K-1$
3: Evaluate Pd_k, for $k = 1, ..., K-1$
4: Calculate As^k, for $k = 1, ..., K-1$
5: **if** $\mathbf{As}^k \in \mathbb{R}^{Q_k}$ **then**
6: Read Dy_k
7: Start Wolfram Mathematica
8: Evaluate numerical TF_{op}/SS_{op}
9: Read TF_{op}/SS_{op} by MATLAB
10: Construct closed loop system (C_{cl})
11: Compute cpc_j for $j = 1...\gamma$
12: **if** $cpc_j \leq b_j$ is true **then**
13: Out numerical cpc_j
14: Evaluate $\min_{\mathbf{v}^k \in \mathcal{D}^k} \sum_{j=1}^{\chi} obj_j(\mathbf{v}^k)$
15: **else**
16: $cpc_j = \varnothing$
17: $\sum_{j=1}^{\chi} obj_j(\mathbf{v}^k) = invalid$
18: **end if**
19: **else**
20: $\sum_{j=1}^{\chi} obj_j(\mathbf{v}^k) = invalid$
21: **end if**
22: Until (generation and population size for the optimiser are achieved)
23: Terminate the Optimiser
24: Apply (weighted sum function)
25: Output f

4 Physical model

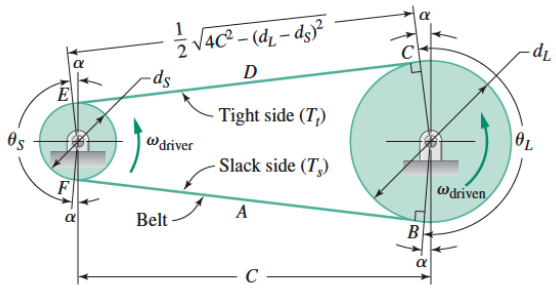
The models described in here are relevant to the case study that will be discussed in the next sections. Dynamics of the physical component models together with the interface equations given in equations (1)–(16) are configured to formulate the continuous time state-space model.

4.1 DC-motor dynamics

$$K_T i_{in} = T_{m,out} + J_m \ddot{\phi}_m \tag{1}$$

where K_T is the motor torque constant and i_{in} is the current. J_m is the rotor inertia and $T_{m,out}$ is the output torque.

Figure 2 Belt terminology and geometry (see online version for colours)



Source: Collins et al. (2009)

4.2 Belt drive dynamics

The nonlinear dynamics for belt drive are adopted from Yang (2000) considering Figure 2.

$$J_{p_1}\ddot{\theta}_1 = T_{v,in} + r_1(T_t - T_s) \tag{2}$$

$$J_{p_2}\ddot{\theta}_2 = r_2(T_s - T_t) + T_{v,out} \tag{3}$$

$$T_s = F_0 + \frac{EA + F_0}{l_c - r_1\theta_1 + r_2\theta_2}(r_1\theta_1 - r_2\theta_2) \tag{4}$$

$$T_t = F_0 + \frac{EA + F_0}{l_c - r_2\theta_2 + r_1\theta_1}(r_2\theta_2 - r_1\theta_1) \tag{5}$$

where J_{p_1}, J_{p_2}, r_1, r_2, θ_1 and θ_2 are the driving and driven pulley inertias, pitch radii and angular positions, respectively. $T_{v,in}$ and $T_{v,out}$ are the input and output torques to and from the pulleys. T_s and T_t are the belt tension both on the slack and tight side. F_0 is the pre-tension force, A is the cross sectional area of the belt, E is the Young's modulus of the material and C is the distance between pulley centres as shown in Figure 2. The belt itself is considered without inertia. By substituting equations (4) and (5) in equations (2) and (3) we get,

$$\begin{aligned} &J_{p_1}\ddot{\theta}_1 \\ &= T_{v,in} + r_1(EA + F_0) \\ &\left(\frac{r_2\theta_2 - r_1\theta_1}{l_c - r_2\theta_2 + r_1\theta_1} - \frac{r_1\theta_1 - r_2\theta_2}{l_c - r_1\theta_1 + r_2\theta_2}\right) \end{aligned} \tag{6}$$

$$\begin{aligned} &J_{p_2}\ddot{\theta}_2 \\ &= r_2(EA + F_0) \\ &\left(\frac{r_1\theta_1 - r_2\theta_2}{l_c - r_1\theta_1 + r_2\theta_2} - \frac{r_2\theta_2 - r_1\theta_1}{l_c - r_2\theta_2 + r_1\theta_1}\right) + T_{v,out} \end{aligned} \tag{7}$$

By linearising the model in equations (6) and (7) at specified points, $\theta_{1_0} = 0$ and $\theta_{2_0} = 0$, we have,

$$\begin{aligned} &J_{p_1}\ddot{\theta}_1 \\ &= T_{v,in} - \frac{2r_1^2(EA + F_0)\theta_1}{l_c} \\ &+ \frac{2r_1r_2(EA + F_0)\theta_2}{l_c} \end{aligned} \tag{8}$$

$$\begin{aligned} &J_{p_2}\ddot{\theta}_2 \\ &= T_{v,out} + \frac{2r_1r_2(EA + F_0)\theta_1}{l_c} \\ &- \frac{2r_2^2(EA + F_0)\theta_2}{l_c} \end{aligned} \tag{9}$$

4.3 Load dynamics

The load is an inertia body with following dynamics:

$$J_l\ddot{\phi}_{l,in} = T_{l,out} - T_{l,in} \tag{10}$$

$$\phi_{l,in} = \phi_{l,out} \tag{11}$$

where J_l is the load inertia connected to the driven pulley, $\phi_{l,in}$ and $\phi_{l,out}$ are the input and output load position profiles, and $T_{l,in}$ and $T_{l,out}$ are the load input and output torques.

4.4 Interface dynamics

The interface equations between dynamics of each component are as following:

$$\phi_m = \theta_1 \tag{12}$$

$$T_{m,out} = T_{v,in} \tag{13}$$

$$T_{v,out} = T_{l,in} \tag{14}$$

$$\theta_2 = \phi_{l,in} \tag{15}$$

$$T_{l,out} = 0 \tag{16}$$

4.5 System model

Considering angular position and velocity of the pulleys as the state variables, we derive the state space model as following:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{2r_1^2(EA+F_0)}{l_cJ_{tot}} & \frac{2r_1r_2(EA+F_0)}{l_cJ_{tot}} & 0 & 0 \\ \frac{2r_1r_2(EA+F_0)}{l_cJ_{out}} & -\frac{2r_2^2(EA+F_0)}{l_cJ_{out}} & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{K_T}{J_{tot}} \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}, D = 0$$

where J_{tot} and J_{out} are the total inertia on the motor side and load side as:

$$J_{tot} = J_m + J_{p_1}$$
$$J_{out} = J_{p_2} + J_l$$

5 Physical design constraints

The constraints on the physical design components are related to mechanical properties of each component without impact of dynamic behaviour models. These constraints include both linear and nonlinear models.

5.1 Physical design of DC-motor

The design model for DC-motor is adopted from Roos (2007) as following,

$$T_{m,rated} \geq T_{rms} \tag{17}$$

$$T_{m,rated} = C_m l_m r_m^{2.5} \tag{18}$$

where C_m is the motor type constant, l_m is the motor's rotor length and r_m is the radius of stator. The motor's RMS torque is derived as,

$$T_{rms} = \sqrt{\frac{1}{\tau}\int_0^{\tau} ((C_{mj} l_m r_m^4 + J_0)\ddot{\phi}_m + T_{m,out})^2 dt} \tag{19}$$

where C_{mj} is a constant for specific machine type and is derived from a reference motor of the same type, τ is the cycle time of the output profile, i.e., the time period during which the output profile is valid, and $\ddot{\phi}_m$ is the angular acceleration of the motor output. $T_{m,out}$ is the output torque of the motor and J_0 is the inertia of motor's shaft and bearings. Combining equations (17)–(19) results in equation (20).

$$C_m l_m r_m^{2.5} \geq \sqrt{\frac{1}{\tau}\int_0^{\tau} ((C_{mj} l_m r_m^4 + J_0)\ddot{\phi}_m + T_{m,out})^2 dt} \tag{20}$$

5.2 Physical design of belt drive

Physical design model for the belt drive considers designing a feasible belt drive structure with a suggested driven pulley's radii as optimisation variable. The angle of wrap for smaller and larger pulleys are θ_S and θ_L, respectively (as shown in Figure 2), and are calculated using equations (21) and (22).

$$\theta_S = \pi - 2sin^{-1}\left(\frac{r_2 - r_1}{C}\right) \tag{21}$$

$$c\theta_L = \pi + 2sin^{-1}\left(\frac{r_2 - r_1}{C}\right) \tag{22}$$

The total length of belt is L_b that is derived as following (Carvill, 1993):

$$L_b = 2\sqrt{(C)^2 - (r_2 - r_1)^2} + r_2\theta_L - r_1\theta_S \tag{23}$$

The length of belt segments A and D is l_c (shown in Figure 2), that is used in dynamic behaviour model and is derived using equation (24).

$$l_c = \sqrt{(C)^2 - (r_2 - r_1)^2} \tag{24}$$

Designed timing belt using the above equations should handle the pre-tension force, F_0, that is implemented in dynamic equations. For the tension forces, we have,

$$T_t - T_s = F_{out} \tag{25}$$

By substituting equations (4) and (5) in equation (25), we derive expression (26) for F_{out},

$$F_{out} = (EA + F_0)\frac{2l_c(r_2\theta_2 - r_1\theta_1)}{l_c^2 - (r_1\theta_1 - r_2\theta_2)^2} \tag{26}$$

By solving equation (26) for $(r_1\theta_1 - r_2\theta_2)$, we obtain,

$$r_1\theta_1 - r_2\theta_2 = \frac{l_c(EA + F_0) + l_c\sqrt{(EA + F_0)^2 + F_{out}^2}}{F_{out}} \tag{27}$$

By substituting equation (27) in equations (4) and (5), we get,

$$T_s = \frac{F_0 - F_{out} - EA - \sqrt{(EA + F_0)^2 + F_{out}^2}}{2} \tag{28}$$

$$T_t = \frac{F_0 + F_{out} - EA - \sqrt{(EA + F_0)^2 + F_{out}^2}}{2} \tag{29}$$

The required area for the belt is determined by equation (30),

$$T_{s,t} = A_{1,2}\sigma_y \tag{30}$$

where σ_y is the yield stress of the belt, and by solving each equation for the area we derive,

$$A_1 = \frac{4(\sigma_y+E)F_0 - 2F_{out}(2\sigma_y+E)}{8\sigma_y(E+\sigma_y)} + \frac{\sqrt{(-4F_0(\sigma_y+E)+2F_{out}(2\sigma_y+E))^2 + 32F_0F_{out}\sigma_y(\sigma_y+E)}}{8\sigma_y(E+\sigma_y)} \tag{31}$$

$$A_2 = \frac{4(\sigma_y+E)F_0 - 2F_{out}(2\sigma_y+E)}{8\sigma_y(E+\sigma_y)} + \frac{\sqrt{(-4F_0(\sigma_y+E)+2F_{out}(2\sigma_y+E))^2 - 32F_0F_{out}\sigma_y(\sigma_y+E)}}{8\sigma_y(E+\sigma_y)} \tag{32}$$

Since σ_y is very small in comparison with E, for simpler computations, we can consider, $\sigma_y + E \approx E$ and $2\sigma_y + E \approx E$.

$$A_1 = \frac{2F_0 - F_{out}}{4\sigma_y} + \frac{\sqrt{(-4F_0(\sigma_y+E)+2F_{out}(2\sigma_y+E))^2 + 32F_0F_{out}\sigma_y(\sigma_y+E)}}{8\sigma_y(E+\sigma_y)} \tag{33}$$

$$A_2 = \frac{2F_0 - F_{out}}{4\sigma_y} + \frac{\sqrt{(-4F_0(\sigma_y+E)+2F_{out}(2\sigma_y+E))^2 - 32F_0F_{out}\sigma_y(\sigma_y+E)}}{8\sigma_y(E+\sigma_y)} \tag{34}$$

The maximum of these areas is considered in design analysis.

5.3 Detailed design of the belt drive

Detailed design of the belt aids in obtaining a reasonable optimum result (according to mechanical engineering principles) and reduces the number of free optimisation variables which yields a computationally less expensive design. When a belt drive system is initially pre-tensioned to F_0, a free body analysis gives the relationship as in equation (35) (Collins et al., 2009).

$$F_0 = \frac{T_s + T_t}{2} \tag{35}$$

A slip equation for a moving belt based on equilibrium requirements is as given in equation (36),

$$e^{\mu\theta_S} = \frac{T_t - F_c}{T_s - F_c} \tag{36}$$

where μ and F_c are the coefficient of friction and centrifugal-force-induced belt tension (inertia force), respectively. The inertia force may be also expressed as follows,

$$F_c = \frac{M_b v^2}{g} \tag{37}$$

where M_b, v and g are the unit weight of belt in N/m, belt velocity in m/s and gravitational acceleration approximated to 9.81 m/s^2. M_b can be calculated as following,

$$M_b = b_{belt}\, T_{belt}\, M_{sw} \tag{38}$$

where b_{belt}, T_{belt} and M_{sw} are the belt width, thickness and specific weight factor in N/m^3, respectively. M_{sw} is derived from standard belt specifications presented by Collins et al. (2009). Using the above equations and by replacing the equivalent of F_c in equation (36), the allowed width of belt, b_{belt}, is calculated.

6 Control design

An optimal pole placement control with two structures are presented and applied on the design case as following. This control method is simple enough and a common approach in industry.

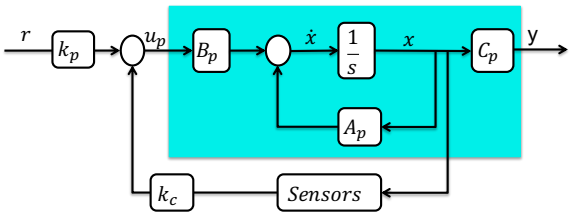
6.1 State feedback control

A state feedback control with sensor quantisation model is depicted in Figure 3.

The open loop system is as follows,

$$\begin{aligned} \dot{x} &= A_p x + B_p u_p \\ y &= C_p x \end{aligned} \tag{39}$$

Figure 3 Pole placement control (see online version for colours)



Using this structure, the sensor quantisation can be applied on any of the state variables. The control system is derived as a gain, k_c, and the control output is $y_c = k_c u_c$ (u_c is the sensor output). When the sensor quantisation is applied on the first state (sn_1), we have,

$$u_c = [q(x_1); x_2; x_3; x_4] \tag{40}$$

where q is the quantisation model which was neglected in previous studies (Malmquist et al., 2014, 2013), as one main aspect of a control system. Since the implementations are done by digital computers, therefore, the sensor quantisation model should be considered in the design process. Quantisation can be modelled using the $floor$ function that quantises the value of input for the sensor. The sensor model, $q(x)$, is adopted from Ferrante et al. (2014) and presented as follows,

$$q(x) = \delta \left\lfloor \frac{x}{\delta} \right\rfloor \tag{41}$$

The plant input is $u_p = k_p r - y_c$ (k_p is a proportional feed-forward control gain). By substituting equation (40) in u_p, we have,

$$u_p = k_p r - k_c \begin{bmatrix} q(x_1) \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{42}$$

By simplifying further, the closed loop system is obtained as,

$$\begin{aligned} \dot{x} &= A_p x + B_p \left(k_p r - k_c x - k_c \begin{bmatrix} q(x_1) - x_1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right) \\ y &= C_p x \end{aligned} \tag{43}$$

We consider a function $\psi(x)$ (Ferrante, 2015) as in equation (44),

$$\begin{aligned} \psi(x) : \mathbb{R}^n &\to \mathbb{R}^n \\ x &\mapsto q(x) - x \end{aligned} \tag{44}$$

For every $x \in \mathbb{R}^n$ and $i = 1, 2, ..., n$, we have,

$$\mid q(x_i) - x_i \mid \leq \delta_i \tag{45}$$

where $q(x_i) - x_i$ is a time variant vector and at this stage we do not have information about it. In order to avoid unnecessary simplifications of the method by applying algorithms that reduce the impact of δ, we let Simulink

to solve the above model. Therefore, a Simulink model is integrated inside constraint function in optimisation algorithm.

By implementing sn_2 in the configuration, i.e., applying quantisation on the second state, we derive the closed loop system as,

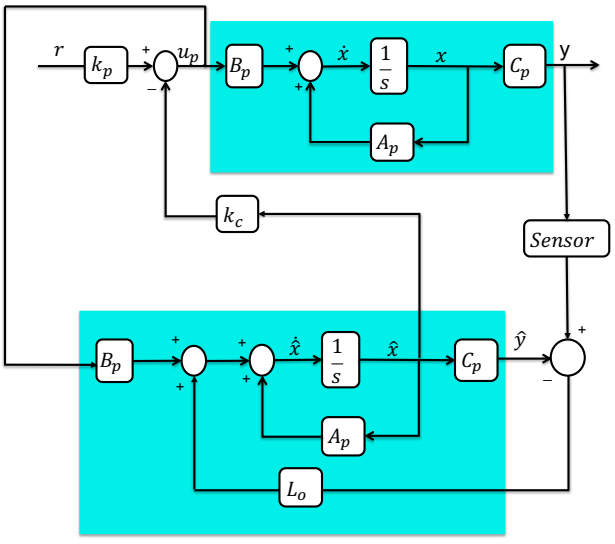
$$\begin{aligned} \dot{x} &= A_p x + B_p \left(k_p r - k_c x - k_c \begin{bmatrix} 0 \\ q(x_2) - x_2 \\ 0 \\ 0 \end{bmatrix} \right) \\ y &= C_p x \end{aligned} \tag{46}$$

The sensor quantisation in estimating states acts as a measurement noise and disturbs the tracking performance of the reference, therefore, a proportional feed-forward control gain, k_p, that is defined to be an optimisation variable is applied which assists in reducing the rise time and steady state error. The response achieved with the above structure is based on the assumption of full-state feedback, which is not necessarily valid. To address the situation where not all state variables are measured, a state estimator must be designed.

6.2 Observer-based control

An observer control provides an estimate of the internal state of a given real system, from measurements of the input and output of the real system. In this structure, the sensor quantisation model should be applied on the output of the system. An observer-based optimal pole placement control is designed by defining the desired pole locations for both the state and observer control as optimisation variables. This method allows estimation of the physical states of a system that cannot be observed/measured. The structure of controller is depicted in Figure 4.

Figure 4 Observer-based pole placement control (see online version for colours)



The system is continuous-time with a quantised output as given in equation (47),

$$\begin{aligned} \dot{x} &= A_p x + B_p u_p \\ y &= q(C_p x) \end{aligned} \tag{47}$$

We assume that the system given in equation (47) is controlled with a dynamic output feedback control of the form in equation (48).

$$\begin{aligned} \dot{\hat{x}} &= A_p \hat{x} + B_p u + L_o (y - C_p \hat{x}) \\ u_p &= k_p \, r - k_c \, \hat{x} \end{aligned} \tag{48}$$

Tracking a non-constant reference is called a servomechanism problem, therefore, we added a feed forward gain, k_p, to assist the tracking performance. This system structure is based on a Luenberger observer (Ellis, 2012), and the state $\hat{x}$ is considered as an estimate of the state x. Hence, the closed-loop system can be written as,

$$\begin{aligned} \dot{x} &= A_p x + B_p u_p \\ \dot{\hat{x}} &= (A_p - B_p k_c - L_o C_p)\hat{x} + B_p r + L_o q(C_p x) \end{aligned} \tag{49}$$

The continuous time closed loop system is discretised using calculated sampling time with respect to the specified constraints and objectives in the following section. Discrete nature of the plant, quantisation in the sensor, sampling and computational delay in the digital controller cause nonlinearity. The nonlinearity limits control performance and to some extent the cost and size of the system which leads to a challenging structural and control design method. Hence, keeping the model and requirements as simple as possible while capturing the main behaviour and properties of the system is crucial in design analysis.

7 Multi-objective optimisation

Multidisciplinary design optimisation of mechatronic systems is a trade-off between optimal objectives, control performance, energy efficiency, cost and safety. Complexity in designing mechatronic systems requires to dig into details in the used parameters. In this paper, a multi-objective optimisation approach is used that considers three objectives from the three engineering disciplines. These objectives are: minimising volume of the physical system, v_{tot}, minimising the sampling frequency, w_s, and maximising the sensor resolution, δ.

Since the continuous time closed loop system and defined load profile are sampled and discretised, it is important to have a good choice of sampling time in the approach. Clearly, a *shorter* sampling time gives the advantage of better performance (closer to the corresponding continuous time system). On the other hand, a *longer* sampling time gives the advantages of lower computational load and may reduce numerical challenges. Therefore, to address this trade-off issue the sampling

frequency is defined as one of the optimisation objectives. The sampling time is derived using equation (50),

$$T_s = \frac{2\pi}{c_b w_b} \tag{50}$$

where w_b is the fastest frequency of the closed loop system and c_b is a frequency coefficient that is defined to be a real valued optimisation parameter in the range of $[2 \dots 30]$. The sampling frequency, w_s, is evaluated using equation (51),

$$w_s = \frac{2\pi}{T_s} \tag{51}$$

To avoid the aliasing effect in sampling of signals, we must set an optimisation constraint on the sampling time, T_s. A good sampling time should be in a defined range such as in equation (52).

$$5w_c \leqslant \frac{2\pi}{T_s} \leqslant 100w_c \tag{52}$$

where w_c is the desired bandwidth of the continuous time closed loop system.

Defining the sampling frequency, w_s, as the second optimisation function aims at reducing computational costs. On the other hand, a too low sampling frequency deteriorates control performance. To avoid this problem, control performance constraint(s) are considered. Integrated square error (ISE) and maximum error ($max(er)$) are defined as control constraints on the optimisation problem.

$$ISE = \sqrt{\sum_{0}^{m} (r_{des} - y_{out})^2} \tag{53}$$

$$max(er) = max(\mid r_{des} - y_{out} \mid) \tag{54}$$

where r_{des} and y_{out} are the desired reference input and the controlled output, respectively.

To avoid expensive sensors, $\frac{1}{\delta}$ is minimised. On the other hand, too large δ deteriorates control performance. Hence, bounding the control constraint, $max(er)$ or ISE, aids in compensating this fact. Hence, the problem is formulated to get a less expensive sensor resolution which can at the same time deliver great control performance regarding the definition of boundary.

A multi-objective optimisation problem generally is defined as given in equation (55) where a set of $m > 1$ conflicting objective functions obj_i are maximised (Hoffmeister and Bäck, 1992). The maximisation is to avoid loss of generality and $min\{obj(x)\} = -max\{-obj(x)\}$.

$$\begin{aligned} &\max\ obj(x) = [obj_1(x), obj_2(x), ..., obj_m(x)] \\ &\text{s.t.}\ \ g_n(x) \leqslant 0;\ \ n = 1, ..., N \end{aligned} \tag{55}$$

where m is the number of objectives, g is a constraint function, n is the number of inequality constraints and x is a vector of design variables.

Normally, a single point that minimises all objectives altogether does not exist. Therefore, Pareto optimality is usually used to extract solutions for multi-objective optimisation problems. In the Pareto optimal set, it is not possible to improve one objective without deteriorating the others. A typical solution of Pareto optimisation is a compromise. Consequently, there is a need to have a proper definition of a function that yields a reasonable optimal solution for all of the objectives. Hence, a weighted sum function is defined as given in equation (56). This function aggregates the objective values to a single quality measure (Jakob and Blume, 2014).

$$f = min(k_1\ obj_1 + k_2\ obj_2 + ... + k_m\ obj_m) \tag{56}$$

where k_1, k_2, ..., k_m are the Pareto front weight factors and we have, $k_1 + k_2 + ... + k_m = 1$. Considering the definition of weight factors, if k_1, k_2, ..., k_m are positive, minimising f yields in Pareto optimality solution (Coelingh et al., 2002).

7.1 Weighted sum method

Defining uneven weights for the weighted sum function given in equation (56) yields a non-optimum system which is only optimum with respect to one of the objectives that has higher weight. However, it is still difficult to decide the weights to indicate the relative importance of each objective due to the difference in objectives magnitudes. One proper method to have even weight factors is to normalise the objectives. This is done using equations (57) and (58) for minimising, and maximising objectives, respectively (Grodzevich and Romanko, 2006).

$$obj_i^{norm} = \frac{max(obj_i) - obj_i}{max(obj_i) - min(obj_i)} \tag{57}$$

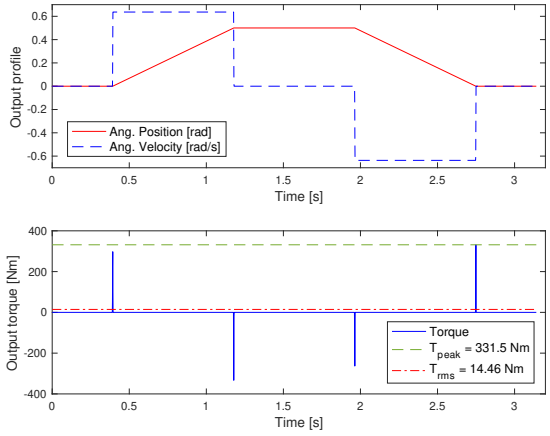
$$obj_i^{norm} = 1 - \frac{max(obj_i) - obj_i}{max(obj_i) - min(obj_i)} \tag{58}$$

The system is optimised to reduce physical size, $obj_1 = min(v_{tot})$, implementation cost, $obj_2 = min(w_s)$, and hardware cost, $obj_3 = min(\frac{1}{\delta})$. The final optimisation is evaluated for $k_1 = k_2 = k_3 = \frac{1}{3}$.

7.2 Nominal load profile/requirement on the system

To dimension and optimise a system, there is a need to have a clear definition of the requirements in system which is a load profile. The load profile gives the required target which the system should deliver within a tolerance set by an operator and is a worst case torque and angular speed that the system should handle. It also represents the reference angular trajectory for the control loop. The torque and velocity of the load profile in this paper are known and derived from a position profile connected to an inertia of $1.1kg/m^2$ and is sampled during an optimisation run to evaluate the system. The profile is given in Figure 5.

Figure 5 Load profile (see online version for colours)



7.3 System design

The approach is tested on a case study that includes a DC-motor, connected to the driving pulley of a belt drive and an inertia as a load connected to the driven pulley. Some parameters are defined to be fixed during optimisation run (presented in Table 1) to reduce complexity of the system and to be able to track the influence of significant parameters.

Table 1 Fixed parameters

Variables	*Fixed value*	*Physical definition*
r_m	0.03 (m)	Radius of the DC-motor
r_1	0.01 (m)	Radius of driving pulley

A few parameters are chosen as free variables to be designed by the approach as given in Table 2. The parameter boundaries for physical constraints are defined based on basic mechanical engineering principles.

Table 2 Design variables

Variables	*Ranges*	*Physical definition*
l_m	$[0.03-0.15]$ (m)	Length of the DC-motor
r_2	$[0.07-0.25]$ (m)	Radius of driven pulley
b_b	$[0.01-0.1]$ (m)	Width of belt
C	$[0.5-1]$ (m)	Centre distance of the pulleys
c_b	$[2-30]$	Sampling rate coefficient
p_1	$[1-400]$	Desired closed loop pole
k_p	$[1-600]$	Proportional control gain
$\frac{1}{\delta}$	$[2-2{,}000](\frac{\text{pulse}}{\text{rad}})$	Sensor factor

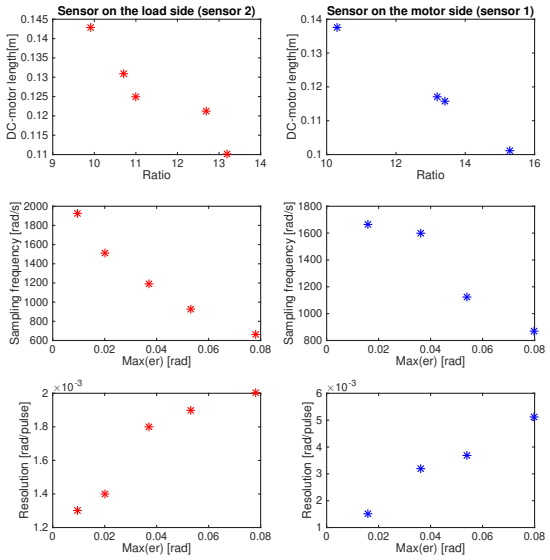
The closed loop system has four poles and we defined the desired pole locations to be $[-p_1; -p_2; -p_3; -p_4] = [-p_1; -p_1 - 10; -p_1 - 20; -p_1 - 30]$ to reduce the complexity of optimiser and computational time. The system is optimised for defined objectives and constraints, and a few tests are examined (presented in Tables 3 and 4). To be noted that the units for objectives in all tests are as follows; v_{tot} $[\text{cm}^3]$, w_s $[\frac{\text{rad}}{\text{s}}]$, and δ $[\frac{\text{rad}}{\text{pulse}}]$.

7.4 Results

Comparing numerical results in these tables, it is clear that a lower (more optimal) sampling frequency is possible with a more loose definition of control constraint boundaries which is in line with presented results by Rahimi and Wikander (2020) (true for both scenarios of sensor locations, sn_1 and sn_2). The results for c_b show that the sampling frequency, w_s, of the system satisfies basic requirement for embedded control implementation as being at least two times the highest frequency of the system. As the boundary for $max(er)$ increases, derived optimal value for δ increases which is intuitively correct. Larger ratio, $n = \frac{r_2}{r_1}$, yields less torque requirement on the motor, consequently, smaller length of the motor, l_m that is in line with the results achieved by Roos (2007).

From Tables 3 and 4, we see that when sn_2 is applied we have a better resolution (performance-wise) for the system than when sn_1 is implemented; i.e., quantising the position of first pulley degrades resolution on the output side. Moreover, when sn_2 is applied, better control performance, $max(er)$, is achieved than when sn_1 is applied. It should be noted that the cost for unity sensors are not considered in any of the results presented in these Tables. The analysed behaviour of the system is depicted in Figure 6.

Figure 6 Patterns generated from system behaviour when sn_1 and sn_2 are applied (see online version for colours)



The control performance of the optimal pole placement with feed forward proportional gain is verified by the position tracking of the output position. Figure 7 shows tracking performance for test 1 in Table 4. The control performance of the optimal control method satisfies the constraint by adequately tracking the reference input.

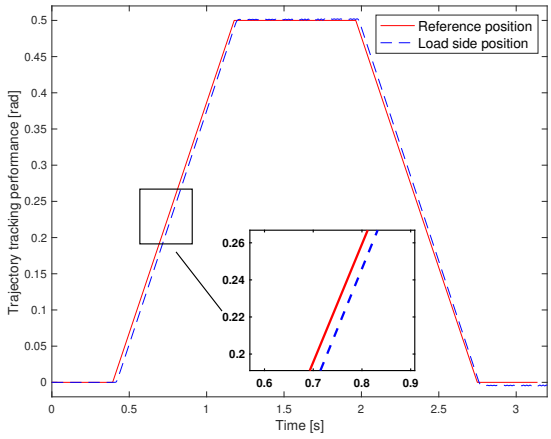
Table 3 Design optimisation when sn_2 is applied on the load side (see online version for colours)

$Test$	l_m	r_2	b_b	C	c_b	$max(er)$	v_{tot}	w_s	δ	T_s	$-p_4$	k_p
1	0.1212	0.127	0.077	0.55	5.0	$0.0095 \leq 0.01$	$1.29e^4$	1920	0.0013	0.0033	−384	456.8
2	0.11	0.132	0.070	0.62	3.8	$0.02 \leq 0.02$	$1.534e^4$	1510	0.0014	0.0041	−397.4	391
3	0.143	0.099	0.081	0.59	4.1	$0.037 \leq 0.04$	$9.43e^3$	1192	0.0018	0.0053	−290.7	402.3
4	0.125	0.11	0.073	0.71	2.7	$0.053 \leq 0.06$	$1.24e^4$	927	0.0019	0.0068	−343.3	272.3
5	0.131	0.107	0.082	0.67	3.4	$0.078 \leq 0.08$	$1.17e^4$	658	0.002	0.0095	−194.5	193

Table 4 Design optimisation when sn_1 is applied on the motor side (see online version for colours)

$Test$	l_m	r_2	b_b	C	c_b	$max(er)$	v_{tot}	w_s	δ	T_s	$-p_4$	k_p
1	0.1157	0.134	0.0707	0.8	6	$0.016 \leq 0.02$	$1.86e^4$	1667	0.0015	0.0038	−277.8	248.5
2	0.117	0.132	0.0718	0.74	5.9	$0.036 \leq 0.04$	$1.69e^4$	1600	0.0032	0.004	−271.2	156
3	0.101	0.153	0.074	0.66	3.6	$0.054 \leq 0.06$	$2.05e^4$	1126	0.0037	0.0056	−312.8	295.1
4	0.1374	0.103	0.0752	0.79	2.9	$0.08 \leq 0.08$	$1.17e^4$	870	0.0051	0.0072	−300	213

Figure 7 Trajectory tracking performance (see online version for colours)



7.5 *Weighted sum factors impact on the co-design*

To evaluate the final solution of the system, different weight factors considering $k_1 + k_2 + k_3 = 1$ are applied on normalised objectives in test 5 (Table 3) and are shown in Table 5. As given, test 1 has higher weight on the volume of system, v_{tot}, and therefore, volume is more optimal (minimised) than the two other objectives compared to test $main$. Test 2 gives higher importance to the sampling frequency, w_s, hence the results prioritise minimising sampling frequency rather than the two other objectives compared to test $main$. In test 3, resolution, δ, has higher weight and the method results in better (maximised) solution for δ.

Table 5 Results of the co-design for different weight factors (test 5 Table 3) (see online version for colours)

$Test$	k_1	k_2	k_3	v_{tot}	w_s	δ
$main$	1/3	1/3	1/3	$1.17e^4$	658	0.002
1	0.8	0.1	0.1	$9.87e^3$	887	0.0019
2	0.1	0.8	0.1	$1.43e^4$	571	0.002
3	0.1	0.1	0.8	$1.2e^4$	693	0.0022

Comparing these results verifies the significant importance of weighted sum factors. In order to derive a final solution that is optimum without deteriorating any objective or giving high priority on just one, these factors should have exact the same weights, i.e., equal numerics.

7.6 *Detailed physical design of the system using an observer-based control*

Using the detailed physical design method presented in Subsection 5.3 for the belt drive, the number of free design variables is reduced. Therefore, the complexity of system analysis and evaluation is decreased. Defined fixed and free variables are given in Tables 6 and 7, respectively.

Table 6 Fixed parameters

Variables	*Fixed value*	*Physical definition*
r_m	0.03 (m)	Radius of the DC-motor
r_1	0.01 (m)	Radius of the driving pulley
T_{belt}	0.002 (m)	Thickness of the belt

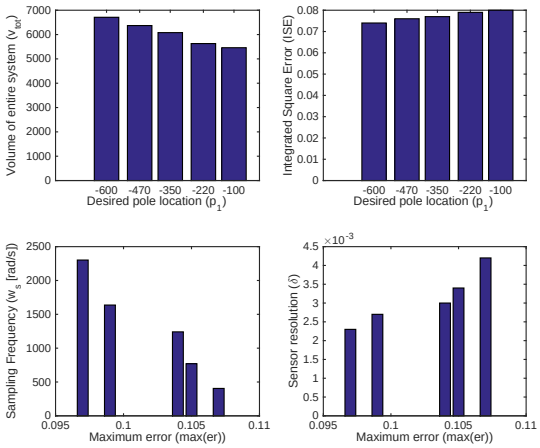
Table 7 Design variables

Variables	*Ranges*	*Physical definition*
l_m	[0.03−0.12] (m)	Length of the DC-motor
r_2	[0.05−0.15] (m)	Radius of driven pulley
C	[0.5−1] (m)	Centre distance of the pulleys
c_b	[2−30]	Sampling rate coefficient
k_p	[1−300]	Proportional control gain
$\frac{1}{\delta}$	[10−1,000] $(\frac{\text{pulse}}{\text{rad}})$	Sensor factor

An observer-based control is used with this design method and the desired pole locations for both the state feedback and observer are defined to be at $[-p_1; -p_1 - 5; -p_1 - 10; -p_1 - 15]$. To track the impact of poles on the physical and control design, a number of optimisations with different fixed values for p_1, and fixed boundaries for $max(er)$ and ISE are executed. The change for physical properties of the system (volume) with respect to the dynamic specifications is investigated.

As it is depicted in Figure 8, faster poles put higher demand on actuator effort [in line with presented results by Malmquist et al. (2014)]. The number of free design variables is reduced by using the detailed physical design approach for the belt drive component, and accordingly, the volume of entire system is in a reasonable level based on basic mechanical engineering principles. The sampling frequency of the system and sensor resolution have similar patterns as presented by Rahimi and Wikander (2020), i.e., as the boundary for control constraints, $max(er)$, and ISE, increase, the optimisation allows larger solution space for both of the objectives (decreased w_s and increased δ).

Figure 8 System behaviour analysis (see online version for colours)



8 Conclusions

Neglecting the influence of control implementation factors on co-design optimisation of mechatronic systems might result in over-dimensioning in one of the involved engineering domains. In this paper, a multidisciplinary co-design approach is presented to include some of the gaps in closed loop system design. The method integrates physical design, control design, and control implementation, simultaneously. An optimal pole placement with two structures; state feedback and observer-based design is implemented. The optimisation problem is formulated to minimise volume and sampling frequency, and maximise sensor resolution. The approach is applied to a design case including a DC-motor, a belt drive and a load component and a few tests are examined to investigate relations between physical, control and implementation parameters. The supported software framework is extended to cover physical and control design aspects, and the coupling between implementation and co-design is considered. The system is optimised and the results are compared for two scenarios; when the sensor quantisation is applied on the first and second state variables, respectively. Control performance is checked over output of the load component and it is evaluated by tracking a reference signal. The objectives are computed for different fixed boundaries of the control constraints. The optimum system solution is evaluated for different weight factors of a scalar weighted sum function using derived Pareto front set and the impact of each parameter on the objectives are studied.

References

Allison, J.T. (2012) 'Plant-Limited co-design of an energy-efficient counterbalanced robotic manipulator', *Proceedings of the ASME 2012 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, No. DETC2012-71108.

Allison, J.T. (2013) 'Engineering system co-design with limited plant redesign', *Eng. Optimiz.*, ePub.10.1080/0305215X.2013.764999.

Asada, H., Park, J.H. and Rai, S. (1991) 'A control-configured flexible arm: integrated structure control design', *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pp.2356–2362.

Becerra, V.M. (2010) 'Solving complex optimal control problems at no cost with PSOPT', *Proceedings of the 2010 IEEE International Symposium on Computer-Aided Control System Design*, pp.1391–1396.

Carvill, J. (1993) *Towards a Methodology for Integrated Design of Mechatronic Servo Systems*, Butterworth-Heinemann.

Coelingh, E., Vries, T.J.A. and Koster, R. (2002) 'Assessment of mechatronic system performance at an early design stage', *IEEE/ASME Transactions on Mechatronics*, Vol. 7, No. 3, pp.269–279.

Collins, J.A., Busby, H.R. and Staab, G.H. (2009) *Mechanical Design of Machine Elements and Machines: A Failure Prevention Perspective*, 2nd edition, 912pp, John Wiley & Sons.

Ellis, G. (2012) *Control System Design Guide*, 4th ed., 520pp, Butterworth-Heinemann.

Ferrante, F. (2015) *On Quantization and Sporadic Measurements in Control Systems: Stability, Stabilization, and Observer Design*, PhD thesis, De L'Université de Toulouse.

Ferrante, F., Gouaisbaut, F. and Tarbouriech, S. (2014) 'Observer-based control for linear systems with quantized output', *European Control Conference (ECC)*, pp.964–969.

Frede, D., Malmquist, D. and Wikander, J. (2013) 'Holistic design optimization in mechatronics', *IFAC Proceedings Volumes*, Vol. 46, No. 5, pp.655–662.

Grodzevich, O. and Romanko, O. (2006) 'Normalization and other topics in multi-objective optimization', *Proceedings of the Fields–MITACS Industrial Problems Workshop*.

Hoffmeister, F. and Bäck, T. (1992) *Genetic Algorithm and Evaluation Startegies: Similarities and Differences*, Technical Report SYS-1/92, FB Informatik, University of Dortmund, Dortmund, Germany.

Jain, S. and Tsiotras, P. (2008) 'Trajectory optimization using multiresolution techniques', *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 5, pp.1424–1436.

Jakob, W. and Blume, C. (2014) 'Pareto optimization or cascaded weighted sum: a comparison of concepts', *Algorithms*, Vol. 7, No. 1, pp.166–185.

Kurdi, M. and Beran, P. (2008) 'Optimization of dynamic response using temporal spectral element method', *Proceedings of the 46th AIAA Aerospace Sciences Meeting and Exhibit*, No. AIAA-2008-0903, AIAA.

Malmquist, D., Frede, D. and Wikander, J. (2014) 'Holistic design methodology for mechatronic systems', *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, Vol. 228, No. 10, pp.741–757.

Malmquist, D., Wikander, J. and Frede, D. (2013) 'Optimal design of harmonic drive servo', *IEEE/ASME International Conference on Advanced Intelligent Mechatronics: Mechatronics for Human Wellbeing*, pp.1579–1584.

O'Driscoll, M. (2002) 'Design for Manufacture', *Journal of Materials Processing Technology*, Vol. 122, Nos. 2–3, pp.318–321.

Park, J.H. and Asada, H. (1994) 'Concurrent design optimization of mechanical structure and control for high speed robots', *ASME J. Dyn. Syst., Meas., Control*, Vol. 116, No. 3, pp.344–356.

Rahimi, F. and Wikander, J. (2020) 'Concurrent and optimal structure, control and implementation design', *2020 IEEE 11th International Conference on Mechanical and Intelligent Manufacturing Technologies (ICMIMT)*, pp.87–97.

Rahimi, F., Feng, L., Wikander, J. and Frede, D. (2017) 'Early phase design-optimization of mechatronic systems', *Proceedings of the 5th International Conference on Control, Mechatronics and Automation*, pp.42–49.

Ravichandran, T., Wang, D. and Heppler, G. (2006) 'Simultaneous plant-controller design optimization of a two-link planar manipulator', *Mechatronics*, Vol. 16, Nos. 3–4, pp.233–242.

Roos, F. (2007) *Towards a Methodology for Integrated Design of Mechatronic Servo Systems*, PhD thesis, KTH Royal Institute of Technology.

Trivedi, D., Dienno, D. and Rahn, C.D. (2008) 'Optimal, model-based design of soft robotic manipulators', *Journal of Mechanical Design*, Vol. 130, No. 9, pp.091402.

Ullman, D.G. (2003) *The Mechanical Design Process*, 3rd ed., 432pp, McGraw-Hill Science.

Yang, X. (2000) *Modeling and Control of Two-Axis Belt-Drive Gantry Robots*, 123pp, Georgia Institute of Technology, Atlanta, GA, USA.

Youcef-Toumi, K. (1996) 'Modeling, design, and control integration: a necessary step in mechatronics', *IEEE/ASME Transactions on Mechatronics*, Vol. 1, No. 1, pp.29–38.