Convex Optimization for Holistic Mechatronic System Design

Yuchao Li
School of Mechanic and
Mechatronic Engineering
Royal Institute of Technology
Stockholm, Sweden
Email: Yuchao@kth.se

Anqing Duan
School of Mechanic and
Mechatronic Engineering
Royal Institute of Technology
Stockholm, Sweden
Email: Anqingd@kth.se

Alexander Gratner
School of Mechanic and
Mechatronic Engineering
Royal Institute of Technology
Stockholm, Sweden
Email: Gratner@kth.se

***Abstract*—Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.**

I. Introduction

Could the current resaerch trend of using genetric algorithms as a solver of mechatronic design optimization problems be improved by adapting a diciplinary convex optimization approach? The multidiciplinary nature of mechatronic systems, which could be characterised as systems with synergetic integrations of mechanical engineering with eletronics and intelligent computer control [**?**], makes early design optimization a difficult process. Optimizations are often done in the latter detailed design phases [**?**], [**?**] whereas early design decisions becomes a limiting factor. Research on the topic of optimization in early design stages for mechatronic systems has presented both holistic and non-holistic approaches which share the common feature of using genetic algorithms, either as a complement or as a standalone solver. The disadvantages of GA when it comes to accuracy and computation time justifies research on applying diciplinary convex optimization.

Optimizing mechatronic systems using genetic algorithms has been widely explored by researchers today. Hammadi [**?**] proposes an emergent multi-agent approach where the system is decoposed into smal sub-systems (or agents) that in turn is optimized using the genetic algorithms NSGA II. Another approach that uses NSGA II is proposed by Guizani [**?**] where a partioning method is utilized to decompose the mechatronic system and classification of the interactions between partions are done. An approach developed by Seo [**?**] and further developed by Behbahani [**?**] uses a two loop optimization process which incorperates genetic algorithms and bond graphs in an outer loop to otimize system topology and genetic programming in an inner loop to fin the elite solution within the optimal topology. The research in this paper will be based upon the work done by Malmquiest et al. [**?**] where they created a framework named IDIOM which optimizises mechatronic systems in a holistic manner by the use of genetic algorithms.

The IDIOM framework developed by Malmquist et al. extends on the Roos [**?**] on a research on a methodology for integrated design och mechatronic servo systems. In his research, Roos reflects on the choice of using genetic algorithms to optimize the servo systems by stating that "The drawback of this method for system optimization is that it is computationally intensive and that there exists no mathematical proof that it actually finds the true optimum". This statement justifies the exploration of other optimization techniques which could provide true optimizations in a less computationally intensive manner, and for which convex optimization has been chosen. Diciplinary convex programming is a methodology developed by Grant et al. [**?**] which purpose, according to the authors, is to "allow much of the manipulation and transformation required to analyze and solve convex programs to be automated". The methodology originates from procedures taken by those who regularly study convex optmization problems and has been implemented in the modelling framework called CVX [**?**].

The argument made by Roos serve as a starting point in the research presented in this report. It challenges the current trend of using evolutionary algorithms when optimizing the design of mechatronic systems and could provide a optimization approach that is less compututionally expensive and more accurate.

The remainder of this paper will introduce the prevuous work done by Malmquist et al. of which this study aim to extend on. $3 describes the method used when applying convex optimization to the mechatronic system. Finally, $4 introduces a case study where a system optimization is performed, and compared to the results obtained by Malmquist et al. , of a mechatronic system composed of a motor, planetary gear and a load.

VIMTESTLOLOLOLOL [1]

II. Previous Work

Roos presented a methodology for the optimal design methodology[???]. According to his proposed methodology, an optimal mechatronic servo system consisting of a motor, planetary gear transmission and load can be achieved regarding system volume, weight and efficiency. The physical system configuration is shown in fig ???. Roos built various dimensioning models for the physical system upon the most critical constraints and designed the corresponding controller by time expensive simulation for minimized control error so that the complete system would be designed at the very early design phase. However, Roos′ methodology treats the static model and the dynamic model separately relying on two optimization loops, so the methodology is not very holistic. In fact, only one fixed system configuration: servo actuator was considered and could only be optimized mainly for a single objective in his methodology.

Malmquist et al. extended Roos′ work further. A prototype software tool was developed for implementing the methodology. The software allows for arbitrary system configuration including PMDC motor, planetary gear, harmonic drive, solid shaft, linear timing belt drive, load as well as full state feedback controller and multiple objectives can be optimized. With these elementary components, Malmquist presented design examples of haptic steering wheel and two axis gantry system. The methodology is completed to some extent and shows applicable potential to solve more general engineering problems compared with Roos.

The modeling part in Malmquist′s work also includes static model and dynamic model. The static model is built for each component based on the critical constraints of the component using objective-relevant design variables and can reflect the relationship between the capability of the component and its design variables. The model is scalable according to the required load. The complexity of the model is aimed at as low as possible so they did not use FEM-analysis for the reason of time efficiency.

The dynamic model in Malmquist′s work is used to evaluate the dynamic performance such as maximum integrated square error, maximum error and overshoot etc. Usually dynamics evaluation involves simulation and that would increase complexity as well as time consumption. The method proposed by Malmquist is that the dominant frequencies of the input signal are identified by applying Fourier transforms and a series of sinusoidal waves would be used to mimic the input signal. Their corresponding outputs would be calculated based on the transfer function separately considering the frequency dependent gain and phase shift. The sum of all the individual output is regarded as the approximation of the time domain response. The transfer function is determined symbolically outside the optimization loop for the reason of time efficiency. During the optimization process of the dynamic part, the controller design without satisfying the dynamic requirements would be rejected. In addition, when the physical system is implemented with a controller, the static model derived earlier might be needed to be checked if it also fulfills the requirements of the dynamic behavior. Thus, a dimensioning factor is introduced to over- or under-dimension the static model so that an optimum could be obtained with both the static and dynamic properties taken into consideration.

The optimization method employed by Malmquist is genetic algorithm, which is one of the most widely used evolutionary methods. It is a non-gradient based optimization. The algorithm is inspired by the Darwinian principle of natural selection. The possible solution with design variables can be compared to an individual with a number of genes. The optimum would be reached after breeding several generations. This method can handle a diverse range of problems but at a cost of low computation efficiency. So replacing it with a more efficient optimization method such as convex optimization is worth being studied further.

III. Optimization Approach

A. Mathematical Preliminaries

The following introduction regarding convex optimization mainly come from [2]. Note that the notations for different functions and sets will be used in later discussions. Before bringing the definition of convex optimization, we need to introduce what convex set and convex function are. A set C is a convex set if for any two elements $x_1, x_2 \in C$, $\theta x_1 + (1-\theta)x_2 \in C$ holds for any $\theta \in [0,1]$. If the set C represents a geometric region, any line segments defined by points which belong to C shall be in the set. A function $f : \mathbf{R^n} \to \mathbf{R}$ is convex if $\mathbf{dom}\ f$ is a convex set and any chord between any two points on f lies above the graph of f. Based on the definition of convex function, a convex optimization (CO) problem is defined as

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \le 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{aligned} \tag{1}$$

where all the functions in 1 are convex functions. $f_0(x)$ is named as objective function, $f_i(x)$ inequality constraint and $h_i(x)$ equality constraint. Although it is now a mature technique to solve a CO problem, it is not easy to formulate objective and constraint functions as convex ones. In the case of mechatronic system design, some typical objective functions, such as volume of the system, are not convex functions and not allowed to be manipulated. However they do follow polynomial form. Luckily, it is still possible to take the advantage of CO technique if those optimization problems can be formulated as another classic form, which is geometric programming (GP).

Before discussion about GP, we need to introduce the form GP follows. A monomial function is defined as 2

$$f(x) = c x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}, \quad \mathbf{dom}\, f = \mathbf{R}^{\mathbf{n}}_{++} \tag{2}$$

where c is positive and a_i can be any real number. A posynomial function is a sum of monomials, which is

$$f(x) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}, \quad \mathbf{dom}\, f = \mathbf{R}^{\mathbf{n}}_{++} \tag{3}$$

A GP problem follows the same format as 1, except that objective and inequality constraints are posynomials and equality constraints are monomials. In fact, by conducting variable substitution, it is easy to show a GP problem can be transferred as a CO problem [2].

In order to formulate an optimization as GP, we don't need to verify if a function is convex or not. Instead, we only need to make sure both objective and constraint functions follow either monomial or posynomial forms. In this way, it is avoided to manipulate functions to be convex and it provides a well defined principle for researchers to create constraints.

B. Modeling Methodology for Mechanical Components

GP is considered to be promising in solving optimization problems of mechatronic system design since many of the constraints can be expressed by polynomial approximation. Although there is some distance between posynomials and polynomials, it is possible to get posynomial form since the parameters we deal with are all in positive orthant and all exponents in posynomials can be any real numbers while only non-negative integers for polynomial models.

In order to create constraints, many of those properties need to be expressed by objective variables as posynomials for inequalities and monomials for equalities. In general, the constraint functions can be obtained by either analysis according to some theoretical principles or data fitting based on existing data. The former way would be easier to get a constraint, but in a rather complex form. Then it is necessary to manipulate the constraint function and make it to be proper. At this stage, there is no systematic way to do this. In the following case, we use a convergent sequence to change a nonlinear inequality to be posynomial. In principle any convergent sequences without negative terms can be used to approximate non-posynomial part of constraints.

In case that no analytic expression is available or the ones we get are too complicated to maneuver, data-fitting approach may be conducted. The data can come from experimental data collection, existing lookup tables or simulation results of obtained expression. In this way identification of models becomes another optimization problem, the optimal of which is a posynomial which has smallest deviation from the data according to some criterion. Although polynomial approximation is a rather mature technology, very few contributions can be found regarding posynomial identification. The method used to create posynomial is adapted from [3].

Suppose $f_i(x)$ is right-hand side of one inequality containing n different optimization variables which is not posynomial and there is a posynomial approximation which has smallest fitting error, which is denoted as

$$f_i^\star(x) = \sum_{k=1}^{K^\star} c_k^\star x_1^{a_{1k}^\star} x_2^{a_{2k}^\star} \cdots x_n^{a_{nk}^\star}$$

So the model identification is equivalent to minimize the fitting error, which is a optimization problem and all coefficients and exponents c_k, a_{ik}, $i = 1, \ldots, n$, $k = 1, \ldots, K^\star$ and cardinality $K^\star$ are optimization variables. It's clear that this is a rather complicated problem. Since the function is used for mechatronic product design in early design stages, a good estimate of the optimal posynomial would be enough. Define an over-parametrized posynomial family as

$$\hat{f}_i(x) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}$$

where $K \gg K^\star$. The range of exponents can be estimated from the slops of plot. By fixing all the other variables, the range of one variable can be calculated and they can be considered as reasonable. After properly discretizing the range, the only left variables are over-parametized coefficients. It is proved in [3] that a good estimation of optimal posynomial can be achieved by following this approach.

IV. CASE STUDY

The physical system studied in our case is a mechatronic servo system consisting of a motor, a shaft and a planetary gear as shown in figure ?????. Both the static and dynamic models for different components have been previously developed by Roos and Daniel. Here we will focus only on how to arrange the existing mathematical models to geometrical form which can be accepted by the CVX toolbox. In order to perform a fair comparison, the load used in our case is the same as that used in DanielâĂŹs, i.e. a load inertia J_{load} 1.1 kg m2 with a specified motion profile (shown in fig ?????) . Similarly, the optimization objective is still the system volume. The dimensioning factor introduced in the dynamic model of DanielâĂŹs method is applied to every optimization result from the static model, while in our case, the dimensioning factor can only be calculated for the optimum from the static model.

Motor

The motor model was derived by Roos[???] based on the constraint of the rated torque no less than the expected RMS torque:

$$T_{m,rated} \geqslant \tilde{T}_m \tag{4}$$

Roos et al derived the rated torque of the motor considering mechanical, magnetic and thermal effects as

$$T_{m,rated} = C_m l_m r_m^{2.5} \tag{5}$$

Where C_m is a constant for a specific motor type and cooling condition, l_m is the motor length and r_m the radius of the stator. The motorâĂŹs root mean square (RMS) torque is derived to drive the given load torque T_l on the load side transferred to the motor side with gear ratio n:

$$\tilde{T}_m = \sqrt{\frac{1}{\tau}\int_0^\tau ((C_{mj}l_m r_m^4 + J_m)\ddot{\varphi}_m + \frac{T_l}{n})^2 \mathrm{d}t} \tag{6}$$

Where C_{mj} is constant for a specific motor type and derived from a reference motor of the same type, J_m is the rotor inertia φ_m is the angular position of the output shaft. Rewriting the formula 4 by combining formula 5 and 6 results in

$$C_m l_m r_m^{2.5} \geqslant \sqrt{\frac{1}{\tau}\int_0^\tau ((C_{mj}l_m r_m^4 + J_m)\ddot{\varphi}_m + \frac{T_l}{n})^2 \mathrm{d}t} \tag{7}$$

Hence, the geometric form of this constraint is derived as

$$\frac{n_j^2 n^2 C_{mj}^2 T_{rms}^2 r_m^3}{J_{load}^2} + \frac{2n_j C_{mj} T_{rms}^2}{J_{load} r_m l_m} + \frac{T_{rms}^2}{n^2 l_m^2 r_m^5} \leqslant C_m^2 \tag{8}$$

where n_j is defined as $\frac{C_{mj}l_m r_m^4 + J_m}{C_{mj}l_m r_m^4}$ and approximated as 10/9. A complementary constraint for the motor model is form factor constraint

$$0.5 \leqslant \frac{l_m}{r_m} \leqslant 5 \tag{9}$$

The dynamic model of the motor is derived as

$$K_t i = \frac{T_l}{n} + J_m \ddot{\varphi}_m \tag{10}$$

Where K_t is the constant and i is the electric current.

Planetary gear

A three wheel planetary gear is used in the servo system instead of a spur gear since the volume of the planetary gear is less than a spur gear when transmitting the same torque. The main constraints on the planetary gear are bending stress in the root of a gear teeth and Hertzian pressure at the teeth contact surfaces. However, if the number of the sun gear teeth is small and/or the the gears are made from ductile steel, the dominant constraint would be the Herzian pressure. Every teeth surface needs to fulfill the following constraints by Roos:

$$r_g^2 b \geqslant Z_H^2 Z_M^2 Z_\varepsilon K_{H\alpha} K_{H\beta} \frac{T_l(n-1)^2}{6(n-1)\sigma_{H,max}^2} \tag{11}$$

Applying the standard parameters defined in Roos work, equation 11 can be simplified as

$$r_g^2 b \geqslant 4 \cdot 10^{10} C_{gr}^2 \frac{T_l(n-1)^2}{(n-1)\sigma_{H,max}^2} \tag{12}$$

where $\sigma_{H,max}$ is the maximum allowed flank pressure given by

$$\sigma_{H,max} = \frac{\sigma_{H,lim}}{SF} \tag{13}$$

$\sigma_{H,lim}$ the maximum allowed Herzian pressure for the particular gear material and SF is the safe factor.

V. Conclusion

Conclusion...

Acknowledgment

References

[1] K. Janschek, *Mechatronic systems design: methods, models, concepts*. Springer Science & Business Media, 2011.

[2] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[3] G. C. Calafiore, L. M. El Ghaoui, and C. Novaraa, “Sparse Identification of Posynomial Models,” *Automatica*, vol. 59, pp. 27–34, 2015.