Preparation of Papers for IEEE Sponsored Conferences & Symposia*

Yuchao Li[1], Anqing Duan[2] and Alexander Gratner[3]

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

The trend within modern research on mechatronic system design optimization is to use evolutionary algorithms due to their ease of implementation and their ability to handle non-smooth, non-differentiable and non-explicit functions in a good manner. Evolutionary algorithms does however suffer from disadvantages concerning accuracy and computational time, two performance indexes that this paper aims to reduce by introducing a convex optimization approach for mechatronic servo systems in early design phases.

The multidisciplinary nature of mechatronic systems, which could be characterised as systems with synergistic integrations of mechanical engineering with electronics and intelligent computer control [1], makes early design optimization a difficult process. Optimizations are often done in the latter detailed design phases [2] 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 evolutionary algorithms (e.g. genetic algorithms), either as a complement or as a standalone solver. Hammadi et al. [3], [4] propose an emergent multi-agent approach where the system is decomposed 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 [5] where a partitioning method is utilized to decompose the mechatronic system and classification of the interactions between partions are done. An approach developed by Seo [6] and further developed by Behbahani [7], [8] uses a two loop optimization process which incorporates genetic algorithms and bond graphs in an outer loop to optimize 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. [9], they introduce a tool called IDIOM for holistic optimization of mechatronic design concept by utilizing genetic algorithms.

The IDIOM framework developed by Malmquist et al. extends on Roos$'$ research [10] concerning a methodology for integrated design and 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. Disciplinary convex programming is a methodology developed by Boyd et al. [11] which purpose, according to the authors, is to "allow much of the manipulation and transformation required to analyse and solve convex programs to be automated". The methodology originates from procedures taken by those who regularly study convex optimization problems and has been implemented in the modelling framework called *CVX* [12] which serves as a package for specifying and solving convex programs. 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 computationally expensive and more accurate.

The remainder of this paper will introduce the previous work done by Malmquist et al. of which this study aim to extend on. Paragraph 3 describes the method used when applying convex optimization to the mechatronic system. Finally, paragraph 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.

II. PREVIOUS WORK

Roos presented a methodology for the optimal design methodology[10]. 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

*This work was not supported by any organization

[1] Yuchao Li.. yuchao@kth.se
[2] Anqing Duan.. anqingd@kth.se
[3] Alexander Gratner.. gratner@kth.se

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 is non-holistic and treats the static and the dynamic model separately by relying on two optimization loops, 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 by developing the prototype software tool IDIOM, which built on Roos′ methodology and incorporated a holistic approach where the static and dynamic model is evaluated together. The tool allows for optimization of desired objectives in arbitrary constructed system using components such as PMDV motors, planetary gears, harmonic drives, solid shafts, linear timing belts and loads. With these elementary components, Malmquist presented design examples of haptic steering wheel and two axis gantry system. A simplification of not taking FEM-analysis into account was made to reduce calculation time.

The dynamic model in Malmquist′s work is used to evaluate the dynamic performance such as maximum integrated square error, maximum error and overshoot. Usually dynamics evaluation involves simulation which would increase complexity as well as time consumption. The proposed method 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 fulfils the requirements of the dynamic behaviour. 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 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 the work presented by Boyd and Vandenberghe [11]. 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 $f : \mathbf{R^n} \rightarrow \mathbf{R}$ 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

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

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

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

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 [11].

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. Proposed Modeling Approach for Static Constraints

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. One way may be using 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 [13].

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 [13] that a good estimation of optimal posynomial can be achieved by solving this optimization problem. More details can be found there.

Another way to handle posynomial constraints would be solving the GP problem with posynomial constraints which are imprecise. In other words, instead of finding good posynomial estimate of $f_i(x)$, a rough estimate with ranges of parameters are obtained. In this case, the estimate is denoted as

$$\hat{f}_i(x) = \sum_{k=1}^{K} \hat{c}_k x_1^{\hat{a}_{1k}} x_2^{\hat{a}_{2k}} \cdots x_n^{\hat{a}_{nk}}$$

where the ranges of parameters are known. The optimization result of the design problem would be a interval. Related work can be found in [14], [15] and [16].

Although it is possible to express many constraints by polynomial with either good estimattion after optimization or with fuzzy coefficients, it is still recommended to simplify the model to maneuver the constraint function first. If it's after all not possible to do so, then those method can be applied then. In the following case, convergent sequence is used to change a non-posynomial constraint to be solvable.

C. *Criteria for Dynamic Performance*

The above-introduced approaches may be applied for constructing constraints of some static features of the system. It is also of interest the dynamic performance of the system under some controller. As introduced in section (previous work), controller is designed by pole placement method. Therefore, it is supposed that the poles of transfer function $G_{\rm cl}(s)$ for the closed-loop system would be desired ones and there is not any new zeros introduced to the system. The focus is put on zeros of closed loop transfer function which are introduced by the system itself. There are three different criteria introduced to evaluate and optimize the dynamic performance of the system, which are overshoot, slope delay and integrated square error (ISE).

Overshoot of the closed-loop system can be rather easy to get since it has obtained by pole placement and has well-designed poles. The overshoot of step response M is given by

$$M = \max_{t \in \mathbf{R}_+} \mathcal{L}^{-1}(\frac{1}{s} G_{\rm cl}(s)) - 1$$

By simply removing insignificant part of the result, a constraint which aiming at increasing the stiffness of components is created if the maximum deviation is given by the user.

Slope delay D is used to describe delay of closed loop system under slope input. If the system has no overshoot, ISE is mainly caused by delay. Therefore, slope delay can roughly describe the information of ISE although it loses some details. It is proved (??? reference needed) that slope delay gets stable after some time, as shown in figure 1 and the static delay can be expressed as

$$D = t_{\rm s} - x(t_{\rm s})$$

where $t_{\rm s}$ is a sufficient large time and $x(t) = \mathcal{L}^{-1}(\frac{1}{s^2} G_{\rm cl}(s))$. By removing insignificant part, another constraint is created if delay time is given.

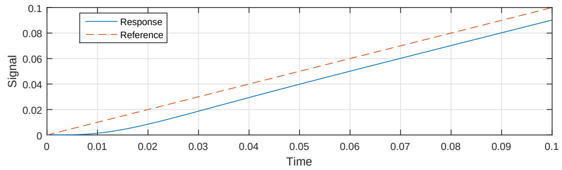

Fig. 1. System reaction under a slope signal.

IV. CONCLUSIONS

V. CASE STUDY

To be written..

VI. CONCLUSION

To be written..

APPENDIX

Appendixes should appear before the acknowledgment.

ACKNOWLEDGMENT

The preferred spelling of the word ÒacknowledgmentÓ in America is without an ÒeÓ after the ÒgÓ. Avoid the stilted expression, ÒOne of us (R. B. G.) thanks . . .Ó Instead, try ÒR. B. G. thanksÓ. Put sponsor acknowledgments in the unnumbered footnote on the first page.

REFERENCES

[1] F. Harashima, M. Tomizuka, and T. Fukuda, “Mechatronics—"what is it, why, and how?" an editorial,” *IEEE Transactions on Mechatronics*, vol. 1, no. 1, pp. 1–4, 1996.

[2] G. Pahl, W. Beitz, J. Feldhusen, and K.-H. Grote, *Engineering Design : A systematic approach*, 3rd ed., Springer, Ed., 2007.

[3] M. Hammadi, J. Y. Choley, O. Penas, a. Riviere, J. Louati, and M. Haddar, “A new multi-criteria indicator for mechatronic system performance evaluation in preliminary design level,” *2012 9th France-Japan and 7th Europe-Asia Congress on Mechatronics, MECATRONICS 2012 / 13th International Workshop on Research and Education in Mechatronics, REM 2012*, pp. 409–416, 2012.

[4] M. Hammadi, J.-y. Choley, and P. Hehenberger, “Mechatronic design optimization using multi- agent approach,” no. JUNE, 2014.

[5] A. Guizani, M. Hammadi, J.-y. Choley, T. Soriano, M. S. Abbes, and M. Haddar, “Multidisciplinary approach for optimizing mechatronic systems : Application to the optimal design of an electric vehicle,” 2014.

[6] K. Seo, Z. Fan, J. Hu, E. D. Goodman, and R. C. Rosenberg, “Toward a unified and automated design methodology for multi-domain dynamic systems using bond graphs and genetic programming,” *Mechatronics*, vol. 13, no. 8-9 SPEC., pp. 851–885, 2003.

[7] S. Behbahani and C. W. De Silva, “Mechatronic design evolution using bond graphs and hybrid genetic algorithm with genetic programming,” *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 1, pp. 190–199, 2013.

[8] ——, “Niching genetic scheme with bond graphs for topology and parameter optimization of a mechatronic system,” *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 1, pp. 269–277, 2014.

[9] D. Malmquist, “A tool for holistic optimization of mechatronic design concepts,” 2015.

[10] F. Roos, “Towards a methodology for integrated design of mechatronic servo systems,” 2007.

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

[12] M. Grant and S. Boyd, “CVX: Matlab software for disciplined convex programming, version 2.1,” http://cvxr.com/cvx, Mar. 2014.

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

[14] S.-T. Liu, “Geometric programming with fuzzy parameters in engineering optimization,” *International journal of approximate reasoning*, vol. 46, no. 3, pp. 484–498, 2007.

[15] ——, “Using geometric programming to profit maximization with interval coefficients and quantity discount,” *Applied Mathematics and Computation*, vol. 209, no. 2, pp. 259–265, 2009.

[16] G. Mahapatra and T. Mandal, “Posynomial parametric geometric programming with interval valued coefficient,” *Journal of Optimization Theory and Applications*, vol. 154, no. 1, pp. 120–132, 2012.

[17] Z. Affi, B. EL-Kribi, and L. Romdhane, “Advanced mechatronic design using a multi-objective genetic algorithm optimization of a motor-driven four-bar system,” *Mechatronics*, vol. 17, no. 9, pp. 489–500, 2007.

[18] A. Albers and J. Ottnad, “Integrated Structural and Controller Optimization in Dynamic Mechatronic Systems,” *Journal of Mechanical Design*, vol. 132, no. 4, p. 041008, 2010. [Online]. Available: http://mechanicaldesign.asmedigitalcollection.asme.org/article.aspx?articleid=1450018

[19] D. K. Didier Casner, Rémy Houssin, Jean Renaud, “Contribution to the embodiment design of mechatronic system by evolutionary optimization approaches,” in *CONTRIBUTION TO THE EMBODIMENT DESIGN OF MECHATRONIC SYSTEM BY EVOLUTIONARY OPTIMIZATION APPROACHES*, Toulouse, France, 2014, pp. 1–7.

[20] Y. He and J. McPhee, *A design methodology for mechatronic vehicles: application of multidisciplinary optimization, multibody dynamics and genetic algorithms*, 2005, vol. 43, no. 10. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-27744456652\&partnerID=tZOtx3y1

[21] M. Iwasaki, M. Miwa, and N. Matsui, “GA-based evolutionary identification algorithm for unknown structured mechatronic systems,” *IEEE Transactions on Industrial Electronics*, vol. 52, no. 1, pp. 300–305, 2005.

[22] T. Lee, M. Leok, and N. H. McClamroch, “Geometric numerical integration for complex dynamics of tethered spacecraft,” *Proceedings of the 2011 American Control Conference*, no. January, pp. 1885–1891, 2011.