2023-01-0550 Published 11 Apr 2023

SAE INTERNATIONAL

Nonlinear, Concave, Constrained Optimization in Six-Dimensional Space for Hybrid-Electric Powertrains

Szabolcs Sovenyi and Feisel Weslati FCA US LLC

Citation: Sovenyi, S. and Weslati, F., "Nonlinear, Concave, Constrained Optimization in Six-Dimensional Space for Hybrid-Electric Powertrains," SAE Technical Paper 2023-01-0550, 2023, doi:10.4271/2023-01-0550.

Received: 25 Oct 2022 Revised: 10 Jan 2023 Accepted: 22 Jan 2023

Abstract

One of the building blocks of the Stellantis hybrid powertrain embedded control software computes the maximum and minimum values of objective functions, such as output torque, as a function of engine torque, hybrid motor torque and other variables. To test such embedded software, an offline reference function was created. The reference function calculates the ideal minimum and maximum values to be compared with the output of the embedded software. This article presents the offline reference function with an emphasis on mathematical novelties. The reference function computes the minimum and maximum points of a linear objective function as a function of six independent variables, subject to 42 linear and two nonlinear constraints. Concave domains, curved surfaces, disjoint domains and multiple local extremum points challenge the algorithm. As a theorem, the conditions and methods for running trigonometric calculations in 6D Euclidean space are presented. More complex geometric structures can be built by joining multiple triangles, adding one triangle at a time. Extensions to 4D, 5D, 7D, etc. are shown to be straightforward. Based on the rules of 6D trigonometry, a probing straight line is constructed. The overlap zone of the constraints is determined along a 6D probing straight line. The optimum point is found within the overlap zone. The algorithm iteratively changes the location and orientation of the straight line to find the optimum point. A formula for rotating any straight line by a given angle in 6D space is derived. The iteration with the probing straight line runs with 2.1 million starting points. In any iteration algorithm, the number of starting points must be determined. This is usually done based on intuition and experience. In this article, the minimum number of required iteration starting points is determined as a function of the required probability of finding the global optimum.

Introduction

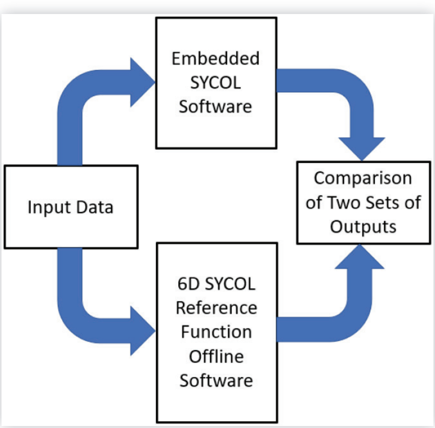
Hybrid electric and EV powertrains are augmented with one, two or more electromotors. Hybrid vehicles and range-extended electric vehicles also have an internal combustion engine to provide additional propulsion and/or battery charging capability. Each electric machine can operate in motoring mode or generator mode. The control software of such a system finds the best operating point of electrical and mechanical power and torque levels to attain a variety of sometimes contradictory objectives in a balanced fashion, such as fuel efficiency, high acceleration and deceleration, maneuverability, and so on. To this end, an optimization process is implemented as an algorithm that maximizes an objective function, (output torque for example), with respect to at most six independent variables the software can control, for instance, generalized electromotor torque variables and engine torque. While a lower number of control effort variables would suffice for the current Stellantis vehicle programs, the number was set to six because this is the highest imaginable combined number of engines and electromotors a Stellantis passenger vehicle could have in the foreseeable future. Each independent variable is a torque value produced by an electromotor or the engine. When a powertrain has fewer than six torque actuators, one or more of the dimensions of the 6D system will be turned off. The optimization process must find the two points where an objective function reaches its minimum and maximum values while honoring some system constraints. For instance, the torque levels must stay below the limits allowed for clutches and drive belts, the high voltage battery charging and discharging power must stay in the safe operating zones, and so on. These constraints are expressed as linear and quadratic inequalities of the independent variables. Within the Stellantis powertrain control software, this constrained optimization problem is solved by the Systems Constraints Library (SYCOL). SYCOL is a software routine that is called by other control software systems many times a second. SYCOL is a static system; that is, it has no memory and no dynamic response. It was implemented by Stellantis engineers other than the authors of this paper in a block diagram of over 366,000 blocks. Depending on which side of which constraint surface the operating point or test point falls, the software keeps switching between different calculation algorithms and keeps searching for different constraint surface intersection points. Also, the Stellantis Electrified Powertrain Team has specific rules for handling mutually contradictory constraints. The constraints are ranked based on priority. Constraints with higher priority must be satisfied first. If a constraint cannot be satisfied, then the optimum point must be placed within the last constraint boundary that is still satisfied, and at a minimum distance from the first constraint surface that is not satisfied. In these situations, the objective function is ignored, so the optimum point could potentially be placed on a surface where the objective function gives the least favorable value. All of this leads to a switched, sometimes stepwise, nonlinear behavior in the SYCOL outputs as a function of the inputs.

Due to the limited execution time available in the embedded processor, SYCOL is designed based on a speed-accuracy trade-off. Years of driving experience in test vehicles and production vehicles proves that SYCOL provides good results in most driving situations. To make sure SYCOL provides the best attainable result in every driving situation, the embedded SYCOL library must be validated. The overall process flow of the validation test is illustrated in Figure 1.

To validate the embedded SYCOL software, an input dataset (representative of the driving situations and vehicle powertrain architectures that the embedded software must handle) was created. The input data set, composed of hundreds of use cases, is sent through the embedded SYCOL and the results were recorded. To calculate the ideal output for each input dataset, a reference function was created. The reference function runs offline on a workstation and is given up to an hour to calculate the output for each use case if necessary. The reference function is given the time and processing power to calculate the ideal outputs as best as it can. The reference function does not take shortcuts or compromise accuracy in any other way. The input data is also sent through the embedded SYCOL software and the outputs of the two systems are compared. If the two do not match, then both systems are scrutinized for the cause of the inaccuracy.

This article focuses on the 6D SYCOL reference function and some mathematical novelty behind it. This reference function must find the minimum and maximum locations and values of a linear objective function of six independent variables. These optimum points must satisfy 42 linear

FIGURE 1 Black box software validation process.



(hyper-planar) and two quadratic (hyper-cylindrical) constraints in the $\mathfrak{R}^6$ space, as a function of six variables: T_i, T_v, T_w, T_x, T_y, T_z. T_i is engine torque, while the other five variables are generalized electromotor torque variables actuated at various parts of the powertrain. To match the nature of the SYCOL embedded software, the reference function must also be a static system with no memory.

Literature Review

In most disciplines of science, e.g. biology, psychology, a rule of thumb states that at least 32 independent samples are required before a statement of predictive power about a population can be made. Some sources recommend 30. As a hypothesis test, it is possible to start with a lower sample size, 25 for instance, and check on the test results. In rare instances, when the data of the first 25 samples shows strongly conclusive results, will a sample size of 25 be accepted. The default recommendation for minimum sample size is 32. By extension, this should also be the minimum number of test vehicles in a test fleet, the minimum number of components subjected to accelerated durability testing, the minimum number of test drivers to give feedback on our vehicles, and so on. With literature review being a sampling process, 39 publications will be reviewed in this article.

Even within controls theory only, a variety of authors call a variety of procedures "optimization." For instance, it can refer to the optimal decentralized controls for interconnected systems, solved with linear quadratic control laws, as on page 785 in [1]. Other meanings are also listed on pp. 822 & 1426 in [1].

In the context of the constrained optimization in electric powertrains at Stellantis, and in several handbooks of mathematics, optimization is a process of finding the minimum or maximum value of an objective function, which is called Tm_1 in case of SYCOL, as a function of several independent variables, in our case T_i, T_v, T_w, T_x, T_y, T_z, while the independent variables must satisfy a set of constraint inequalities.

An optimization problem is dynamic if some variables or events depend on time, and static otherwise, see page 878 [2]. The optimization task solved by SYCOL is static.

A dynamic optimization process is continuous if the variables are continuous, and discrete if isolated, stepwise events occur, see page 878 in [2]. SYCOL is a continuous system, but the outputs are not differentiable functions of the inputs.

While in case of SYCOL the objective function is linear, the optimization is nonlinear due to the two quadratic constraints.

The optimization can be single-objective and multi-objective, as explained on page 211 in [3]. Multi-objective optimization processes are also called vector optimization. The multiple objective function values are combined into a single performance metric, and then the single-objective techniques are used. SYCOL runs a single-objective optimization.

In case of SYCOL, the quadratic constraint of smaller radius creates concave or even hollow shapes and it turns the problem into a concave one. Also, in case of the SYCOL reference function, the problem is finite (six) dimensional. After specifying the features of the optimization problem the SYCOL reference function has to solve, the review of the possible solution approaches will follow.

Linear programming is a popular optimization toolkit. It is only applicable to situations where the constraints and the objective function are linear. [4, 5]. Due to the quadratic constraints SYCOL has to work with, these techniques are not applicable to the SYCOL reference function or the embedded SYCOL software.

The simplex method is only applicable for solving problems that have linear constraints only. [5] Also, the simplex method assumes a single domain. Because SYCOL sometimes faces two or more disjoint domains, this method is not applicable to the present work.

Optimization over linear matrix inequalities (LMIs) is presented on page 752 in [1]. Unfortunately, because of the quadratic constraints, this method is not applicable to SYCOL.

Newton's method is presented on page 665 in [6] and on page 879 in [7]. Newton's method finds the independent variable value(s) that lead to a function to be zero. All by itself it is not an optimization method, but it can be used as a part of a larger optimization algorithm that, for instance, aims to find the location where a gradient or some derivative is zero. The method also relies on the calculation of the Hessian matrix, the matrix of second derivatives. In general, in an embedded controller environment, the first derivative of any signal is usually noisy. The second derivative is even noisier and practically useless in most situations. In particular, in case of SYCOL, the switched application of the constraint surfaces and the edges and corners where the optimum points tend to sit make the constrained domain surface non-differentiable, so gradients, derivatives and the elements of the Hessian matrix are not possible to calculate. Also, Newton's method does not consider constraints. Some other method must be found to serve as the backbone of the 6D SYCOL reference function.

The gradient descent method, see page 666 in [6] requires the calculation of partial derivatives. For reasons already mentioned, this is not possible in most scenarios tackled by the 6D SYCOL.

The topic of minimum search in N-dimensional Euclidean vector-spaces is summarized based on pp. 878-888 in [7]. The discussion in [7] covers minimum values, but it is equally applicable to maximum values. Suppose f is a scalar function $f(\underline{x})$ of the N-dimensional vector $\underline{x}$. Our goal is to find the minimum location, x^* of $f(x)$. The algorithm takes the elements of the N-dimensional vector $\underline{x}$ one at a time. With respect to the k-th element, $(k = 1, \dots N)$ the algorithm optimizes the objective function $f(\underline{x})$. The algorithm runs this process for each element of the objective function vector, and then loops back to start over. Since this method does not require differentiation, this could be a good candidate for the 6D SYCOL reference function.

Various other linear programming optimization methods are also documented to solve various problems, such as transportation problem, material distribution problem, traveling salesman problem pp. 857-861 in [2]. Because these address linear systems, these methods are not applicable to the 6D SYCOL.

Evolution strategies, a.k.a. genetic algorithms are covered on page 870 in [2], also in [8]. These methods imitate the evolution processes seen in nature. The process starts with a randomized pool of solution point candidates. The quality of each solution point candidate $\underline{x}^r$ is evaluated against a set of criteria. In case of SYCOL, these are the objective function value and the satisfaction of the constraints. Based on the evaluation, a performance metric is assigned to each candidate. The best half (or some other portion) of the ranked point candidates is kept, the others are discarded. Among the candidates that are kept, a recombination process is executed to merge and mix the properties of the good candidates and arrive at the second generation of candidates. A small percentage of the properties is randomized to imitate the effect of mutations. The process then repeats for the second generation of candidates, and so on, until the process converges to a sufficiently good extremum point. This approach might work for solving the 6D SYCOL constrained optimization problem. However, the 6D SYCOL reference function task may benefit from a more intended, purposeful search process with a more analytical evaluation and overlapping of constraint surfaces and objective function values.

Machine learning could in principle handle constrained optimization problems with reinforcement learning. The Stellantis Electrified Powertrain Team (ePT) has already attempted to implement artificial intelligence to solve 2D SYCOL optimization problems. Even with 200,000 training datasets, the ANN could not learn to find intersection points with the accuracy required for SYCOL. 6D optimization problems would likely pose an even greater challenge to AI. As a result, the ePT team kept searching for other approaches.

The gradient method for problems with inequality type constraints has many sub-variants. One of them is the method of allowed gradients, a.k.a. direction search program or method of feasible directions, as presented on page 871 in [2], also on pp. 169-170 in [9]. This method is based on the gradient descent approach and considers the gradient of the constraint surfaces as well as the gradient of the objective function. As previously stated, no method that works with derivatives can be applied as a part of the 6D SYCOL reference function.

A gradient method for problems with inequality type constraints is presented on page 873 in [2], also a gradient projection method is presented on pp. 219-224 in [9]. This method only applies to convex domains that are bounded by linear inequalities. In case of SYCOL the domains can be concave and two of the boundary surfaces are quadratic. Therefore, this method is not applicable to SYCOL or its reference function.

Penalty function and barrier methods are covered on pp. 885-886 in [7], on pp. 875-876 in [2], on pp. 207-218 [9], pp. 178 & 183 in [3]. The idea is that the constraints are turned into penalty functions and are added to the objective function. The objective function becomes a single function that has to be minimized or maximized, now also representing the constraints. As an experiment, for a 4D example, this approach was attempted by the ePT team. This method may work in case of the 6D SYCOL reference function, but it is unclear if or how it could handle the 44 sometimes contradictory and/or switched constraints and/or pinched constraints the 6D

TABLE 3 Results obtained for Example 3.

Coordinate of Tm_{1Max} solution point	Expected result	Result of 6D SYCOL Reference Function
T_i	0	0.001603644
T_v	$5/\sqrt{2} = 3.535534$	3.536335
T_w	$-5/\sqrt{2} = -3.535534$	-3.534731
T_x	0	-7.69278E-08
T_y	0	-1.66214E-07
T_z	0	1.85973E-08

Conclusions

An optimization algorithm was presented for finding the minimum and maximum value of a linear objective function in six-dimensional space, while satisfying 42 linear and two quadratic constraints in a prioritized fashion.

The actuators of a typical powertrain, such as electromotors, engine, etc. are only about ±1% accurate, so the three-to-nine decimal digit accuracy achieved by the 6D SYCOL reference function is adequate for the purpose of the powertrain calculations considered in this article. Should any other application necessitate the attainment of finer accuracy, the speed-accuracy tradeoff of the iteration runs can be adjusted in favor of greater accuracy by lowering the iteration step size and increasing the maximum limit for the number of iteration steps.

This article presents the following mathematical novelty:

1. How and under what conditions a triangle can be moved from 2D to ND Euclidean space and back, where $3 \leq N$.
2. The Neville-Sovenyi Theorem that states that as long as angles and distances are defined in a certain way, all the 2D trigonometric relationships between distances and angles apply to any three points in N-dimensional Euclidean spaces, $2<N$. More complex structures can be created from multiple triangles, adding one triangle at a time in N-dimensional space.
3. A formula for modifying one of the coordinates of an N-dimensional vector in a way that the angle between the original and the modified vector will be equal to a predetermined value, α. See appendix A.5.
4. A theorem that states that in N-dimensional space, if two free vectors are parallel, then they are scalar multiples of each other.
5. A formula for calculating the minimum number of iteration starting points needed as a function of the required probability of finding the global optimum point.
6. A probing straight line for algebraically overlaying intervals that satisfy various constraints.
7. The observation that writing a literature review is a sampling process.

The 6D SYCOL reference function has already been deployed to test the SYCOL embedded software in hundreds of driving scenarios. Most results confirm that the SYCOL embedded software runs accurately. In some other test cases the SYCOL reference function and the SYCOL embedded software gave different outputs. In those cases, one or the other or both systems may be off target. As future work, the analysis of those test cases and the correction of the system or systems that gave the inaccurate results is planned.

References

1. Levine, W.S. (Eds), *The Control Handbook* (IEEE Press, CRC Press, 1996), ISBN:0-8493-8570-9
2. Bronshtein, I.N., Semendyayev, K.A., Musiol, G., and Muehlig, H., *Handbook of Mathematics*, Fifth ed. (Springer-Verlag, 2007), ISBN:978-3-540-72121-5
3. Mykel, J., Kochenderfer, and Tim, A., *Wheeler: Algorithms for Optimization* (The MIT Press, 2019), ISBN:978-0-262-03942-0
4. *Marks' Standard Handbook for Mechanical Engineers*, Eleventh Edition, McGraw Hill, 2007, 2-50, (2–50 means chapter 2, page 50) and page 17–8. 978-0-07-142867-5, ISBN-10 0–07–142867-4.
5. Bronstejn, I.N., Szemengyajev, K.A., and Zsebkonyv, M., (*Pocket Book of Mathematics*), 6th edition, Muszaki Konyvkiado, Budapest, 1987, 1046-1099, 963 1053091.
6. *CRC Standard Mathematical Tables and Formulas*, 33rd edition, Edited by Zwillinger, Dan, CRC Press, 2018, 665-667, 978-1-4987-7780-3, (Hardback).
7. Bronstejn, I.N., Szemengyajev, K.A., and Musiol, G., Muhlig, H., Kezikonyv, M., (*Handbook of Mathematics*), 4th, reworked edition, TypoTEX Kiado, Budapest, 2000, 963-9132-59-4.
8. Rahi, K.H., Singh, H.K., and Ray, T., "Partial Evaluation Strategies for Expensive Evolutionary Constrained Optimization," *IEEE Transactions on Evolutionary Computation* 25, no. 6 (December 2021): 1103-1117, doi:10.1109/TEVC.2021.3078486.
9. Gill, Philip E., Murray, Walter, and Margaret, H., "Wright: Practical Optimization," *Society of Industrial and Applied Mathematics*, 2020, This is an unaltered republication of the same work originally published by Academic Press, in 1981, 978-1-611975-59-8.
10. Kennedy, J., and Eberhart, R., "Particle Swarm Optimization," In *Proc. IEEE Int Conf. on Neural Networks*, 1942-1948, Piscataway, NJ, 1995.
11. Eberhart, R., and Kennedy, J., "A New Optimizer Using Particle Swarm Theory," In *Proc. 6th Int Symposium on Micro Machine and Human Science*, 39-43, Nagoya, Japan, 1995.
12. Qin, J., and Xie, B., "A Naive Particle Swarm Algorithm for Constrained Optimization Problems," in *2013 25th Chinese Control and Decision Conference (CCDC)*, 2013, 46-50, 10.1109/CCDC.2013.6560892.
13. Dong, Y., Tang, J., Xu, B., and Wang, D., "An Application of Swarm Optimization to Nonlinear Programming," *Computers and Mathematics with Applications* 29 (2005): 1655-1668.

14. Sahin, Mehmet Fatih, Eftekhari, Armin, Alacaoglu, Ahmet, Latorre, Fabian, et al., *An Inexact Augmented Lagrangian Framework for Nonconvex Optimization with Nonlinear Constraints. Optimization and Control*, arXiv: 1906.11357v1 [math.OC] LIONS, Ecole Polytechnique Federale de Lausanne, Switzerland, 26 Jun 2019, 1-29.
15. Hestenes, M.R., "Multiplier and Gradient Methods," *Journal of optimization theory and applications* 4, no. 5 (1969): 303-320.
16. Powell, M.J., "A Method for Nonlinear Constraints in Minimization Problems," *Optimization* (1969): 283-298.
17. Wang, Z.L., Qiu, L., Fu, Q., and Liang, C., "Applications of Chaos Optimization Algorithm to Nonlinear Constrained Programming (in Chinese)," *Journal of North China Institute of Water Conservancy and Hydroelectric Power* 23, no. 2 (2002): 1-3.
18. Dorigo, M., Maniezzo, V., and Colori, A., "Ant System Optimization by a Colony of Cooperating Agents," *IEEE Transactions of Systems, Man and Cybernetics* 26, no. 1 (1996): 28-41.
19. Yan, L.X. and Ma, D., "Global Optimization of Non-Convex Nonlinear Programs Using Line-Up Competition Algorithm," *Computers and Chemical Engineering* 25 (2001): 1601-1610.
20. Zulkefli, M.A.M. and Sun, Z., "Fast Numerical Powertrain Optimization Strategy for Connected Hybrid Electric Vehicles," *IEEE Transactions on Vehicular Technology* 68, no. 9 (Sept. 2019): 8629-8641, doi:10.1109/TVT.2019.2929412.
21. Li, M., and Lv, R., "The Binary Algorithm for Finding the Best Weak-Constraints in Contradictory Weak-Constrained Optimization Problems," in *2014 10th International Conference on Natural Computation (ICNC)*, 2014, 475-479, 10.1109/ICNC.2014.6975881.
22. Gill, Philip E., Murray, Walter, and Saunders, Michael A., *User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming*, 2008.
23. David, V., *Widder: Advanced Calculus*, 2nd edition, Dover publications Inc., New York, 1989, 61-87, 0-486-66103-2.
24. Murray, R., and Spiegel, P.H.D., *Schaum's Outlines: Mathematical Handbook of Formulas and Tables*, Schaum's Outline Series, McGraw Hill, © 1968. 35th printing, 1996, 46-52, also pp. 116–118. (No ISBN shown.).
25. William H Beyer, P.H.D., *CRC Standard Mathematical Tables*, 26th Edition, 1981, CRC Press Inc, Boca Raton, Florida, 0-8493-0626-4.
26. Myškis, A.D., *Advanced Mathematics for Engineers – Translation from the Russian*, by: V. M. Volosov, D.Sc., and Volosova, I.G., MIR Publishers, Moscow, © English Translation, Mir Publishers, 1975, 15-35, (No ISBN shown.).
27. Graham, I.S. and Kohr, G., *Geometric Function Theory in One and Higher Dimensions* (Marcel Dekker Inc., 2003), 1-477, ISBN:0-8247-0976-4
28. Debarre, Olivier, *Higher-Dimensional Algebraic Geometry 2001*, 1-217, Universitext, ISBN-13: 978–0387952277, 0387952276.
29. Reinhardt, Fritz, and Soeder, Heinrich, *SH Atlasz – Matematika*, © 1993 for the Hungarian Translation: Springer Hungarica Kiadó Kft., Budapest, 963-7775-60-9.
30. Zill, Dennis G., and Cullen, Michael R., *Advanced Engineering Mathematics*, 2nd Edition, Jones and Bartlett Publishers, Sudbury, Massachusetts, 2000, 297-337, 0-7637-1065-2.
31. Bay, J.S., *Fundamentals of Linear State Space Systems, WCB/McGraw Hill* (The McGraw-Hill Companies, Inc, 1999), 46-84, ISBN:0-256-24639-4
32. Manning, Henry Parker, *The Fourth Dimension Simply Explained, Dover Books on Mathematics*, New York, Munn & Company, Inc., 1910. Reprint: FB &c Ltd., 2016, Available on www.ForgottenBooks.com.
33. Neville, E.H., *The Fourth Dimension*, Cambridge at the University Press, 1921, (No ISBN is shown.) Available on www.ForgottenBooks.com
34. Sommerville, D.M.Y., *An Introduction to the Geometry of N Dimensions*, Reprint 2015 in India by Facsimile Publisher, Delhi. 9789332857902, First publication printed in Great Britain at the University Press, Aberdeen, 1929.
35. Baker, Henry F., "Principles of Geometry. Volume IV, Higher Geometry. Being Illustrations of the Utility of the Consideration of Higher Space," *Especially of Four and Five Dimensions*, Frederich Ungar Publishing Co., New York, 1963, Originally published by Cambridge University Press in 1925. (No ISBN shown.).
36. Kendall, M.G., *A Course in the Geometry of N Dimensions*, Dover Publications, Inc., Mineola, New York, 2004, 0-486-43927-5, This book is an unabridged republication of another book published by Charles Griffin and Company, Limited, London, 1961.
37. Zhu, D., Pritchard, E., Dadam, S.R., Kumar, V. et al., "Optimization of Rule-Based Energy Management Strategies for Hybrid Vehicles Using Dynamic Programming," *Combustion Engines* 184, no. 1 (2021): 3-10, doi:10.19206/CE-131967.
38. Jinquan, G., He, H., Jiankun, P., and Nana, Z., "A Novel MPC-Based Adaptive Energy Management Strategy in Plug-in Hybrid Electric Vehicles," *Energy* 175 (2019): 378-392.
39. Xie, S., Hu, X., Xin, Z., and Brighton, J., "Pontryagin's Minimum Principle Based Model Predictive Control of Energy Management for a Plug-in Hybrid Electric Bus," *Applied energy* 236 (2019): 893-905.

Nomenclature

$\boldsymbol{a}$ - Length of one of the sides of a triangle in 2D or ND

$\boldsymbol{b}$ - Length of one of the sides of a triangle in 2D or ND

$\boldsymbol{c}$ - Length of one of the sides of a triangle in 2D or ND

$\underline{\boldsymbol{A}_2}$ - Point A in two-dimensional space

$\underline{\boldsymbol{A}_3}$ - Point A in three-dimensional space

$\boldsymbol{A}_4$ - Point A in four-dimensional space

$\overrightarrow{\boldsymbol{AB}}$ - Vector pointing from point A to point B

$\overrightarrow{\boldsymbol{AC}}$ - Vector pointing from point A to point C

$\overrightarrow{\boldsymbol{AC}}_{parl}$ - Parallel projection of vector $\overrightarrow{AC}$ to vector $\overrightarrow{AB}$

$\overrightarrow{\boldsymbol{AC}}_{perp}$ - Component of vector $\overrightarrow{AC}$ orthogonal to vector $\overrightarrow{AB}$

$\boldsymbol{A}_w$ - The W coordinate of point A

$\boldsymbol{A}_x$ - The X coordinate of point A

$\boldsymbol{A_y}$ - The Y coordinate of point A

$\boldsymbol{A_z}$ - The Z coordinate of point A

$\underline{\boldsymbol{B_2}}$ - Point B in two-dimensional space

$\underline{\boldsymbol{B_3}}$ - Point B in three-dimensional space

$\underline{\boldsymbol{B_4}}$ - Point B in four-dimensional space

$\boldsymbol{B_w}$ - The W coordinate of point B

$\boldsymbol{B_x}$ - The X coordinate of point B

$\boldsymbol{B_y}$ - The Y coordinate of point B

$\boldsymbol{B_z}$ - The Z coordinate of point B

C - A very large scalar value added to the *i*-th coordinate of vector *x*

$\underline{\boldsymbol{C_2}}$ - Point C in two-dimensional space

$\underline{\boldsymbol{C_3}}$ - Point C in three-dimensional space

$\underline{\boldsymbol{C_4}}$ - Point C in four-dimensional space

$\boldsymbol{CTm_1}$ - A constant term in the linear equation used to calculate the objective function Tm_1

$\boldsymbol{CTm_5}$ - A constant term in the linear equation used to calculate the constraint variable Tm_5

$\boldsymbol{C_w}$ - The W coordinate of point C

$\boldsymbol{C_x}$ - The X coordinate of point C

$\boldsymbol{C_y}$ - The Y coordinate of point C

$\boldsymbol{C_z}$ - The Z coordinate of point C

D - Distance between two points

***f*(*x*)** - Scalar function of vector *x*

g - Running parameter of an equation of a straight line

GMHR - Global Maximum Hit Ratio

GSL - Given straight line

$\mathbf{H_0}$ - Null hypothesis

j - Index of one particular constraint surface among the full set of constraint surfaces

k - Index of dimensions in a multi-dimensional vector or a function or Euclidean space

$\boldsymbol{L_1}$ - Scalar weight of a constraint equation, used for forming the Lagrangian function

$\boldsymbol{L_2}$ - Scalar weight of a constraint equation, used for forming the Lagrangian function

m - Integer multiplier of π

N - The total number of dimensions of a vector or a function or an Euclidean space

ND - *N*-dimensional

$\boldsymbol{N_s}$ - Sample size, in the context of this article equivalent to the number of starting points of an iteration

$\boldsymbol{p_k}$ **, *k*=1 … 6** - *k*-th coordinate of position vector $\underline{P}$

$\boldsymbol{p_{sk}}$ **, *k*=1 … 6** - *k*-th coordinate of position vector $\underline{P_s}$

$\underline{\boldsymbol{P}}$ - Position vector of point P

$\boldsymbol{P_1}$ - A generic point in *N*-dimensional Euclidean space

$\boldsymbol{P_{1j}}$ - The *j*-th coordinate of point P_1

$\boldsymbol{P_2}$ - A generic point in *N*-dimensional Euclidean space

$\boldsymbol{P_{2j}}$ - The *j*-th coordinate of point P_2

$\underline{\boldsymbol{P_s}}$ - Position vector of the starting point of a straight line

Q - If a load case presents contradictory constraints, then *Q* is the index of the lowest priority constraint that is still possible to satisfy

R - If a load case presents contradictory constraints, then *R* is the index of the highest priority constraint that is already impossible to satisfy

$\boldsymbol{R_{min}}$ - Minimum value of quadratic constraint expression

$\boldsymbol{R_{max}}$ - Maximum value of quadratic constraint expression

s - Half of the perimeter of a triangle

Sh - A geometric shape in *N*-dimensional space that is described by an equation that is missing one of the *N* variables

SL - Straight line

$\boldsymbol{S_w}$ - The same scalar value substituted for the W coordinate of various points

t - Time

$\boldsymbol{T_i}$, - Engine torque

$\boldsymbol{Ti2Tm_1}$ - Coefficient of engine torque T_i, expresses the contribution of engine torque T_i to the objective function Tm_1

$\boldsymbol{Ti2Tm_5}$ - Coefficient of engine torque T_i, expresses the contribution of engine torque T_i to the constraint variable Tm_5

TiMin - Minimum value of engine torque

TiMax - Maximum value of engine torque

$\boldsymbol{Tm_1}$ - Objective function value

Tm1_box_max - The maximum value the objective function Tm_1 reaches anywhere on the motor torque limit box

Tm1_box_min - The minimum value the objective function Tm_1 reaches anywhere on the motor torque limit box

Tm1_max - Maximum value of the Tm_1 objective function

Tm1_max_real - The highest existing values among all *Tm1_max* values retrieved by iteration runs

Tm1_max_sampled - The highest among all Tm_1 values retrieved by the iteration runs based on all the starting points

$\boldsymbol{Tm_5}$ - Constraint variable

$\boldsymbol{Tm_{16}}$ - Constraint variable

$\boldsymbol{Tm_{5max}}$ - Maximum value for the constraint *k* limit Tm_5

$\boldsymbol{Tm_{5min}}$ - Minimum value for the constraint *k* limit Tm_5

$\boldsymbol{Tm_{kmax}}$ - Maximum value for the constraint *k* limit Tm_k

$\boldsymbol{Tm_{kmin}}$ - Minimum value for the constraint *k* limit Tm_k

$\boldsymbol{T_v}$ - Electromotor V torque

$\boldsymbol{Tv2Tm_1}$ - Coefficient of electromotor torque T_v expresses the contribution of engine torque T_v to the objective function Tm_1

$\boldsymbol{Tv2Tm_5}$ - Coefficient of electromotor torque T_v expresses the contribution of electromotor torque T_v to the constraint variable Tm_5

$\boldsymbol{T_{vMin}}$ - Minimum value of motor V torque

$\boldsymbol{T_{vMax}}$ - Maximum value of motor V torque

$\boldsymbol{T_w}$ - Electromotor W torque

$\boldsymbol{Tw2Tm_1}$ - Coefficient of electromotor torque T_w expresses the contribution of electromotor torque T_w to the objective function Tm_1

***Tw2Tm*$_5$** - Coefficient of electromotor torque T_w expresses the contribution of electromotor torque T_w to the constraint variable Tm_5

TwMax - Maximum allowed torque value of motor W

TwMin - Minimum allowed torque value of motor W

$\boldsymbol{T_x}$ - Electromotor X torque

***Tx2Tm*$_1$** - Coefficient of electromotor torque T_x expresses the contribution of electromotor torque T_x to the objective function Tm_1

***Tx2Tm*$_5$** - Coefficient of electromotor torque T_x expresses the contribution of electromotor torque T_x to the constraint variable Tm_5

TxMax - Maximum allowed torque value of motor X

TxMin - Minimum allowed torque value of motor X

$\boldsymbol{T_y}$ - Electromotor Y torque

***Ty2Tm*$_1$** - Coefficient of electromotor torque T_y expresses the contribution of electromotor torque T_y to the objective function Tm_1

***Ty2Tm*$_5$** - Coefficient of electromotor torque T_y expresses the contribution of electromotor torque T_y to the constraint variable Tm_5

TyMax - Maximum allowed torque value of motor Y

TyMin - Minimum allowed torque value of motor Y

$\boldsymbol{T_z}$ - Electromotor Z torque

***Tz2Tm*$_1$** - Coefficient of electromotor torque T_z expresses the contribution of electromotor torque T_z to the objective function Tm_1

***Tz2Tm*$_5$** - Coefficient of electromotor torque T_z expresses the contribution of electromotor torque T_z to the constraint variable Tm_5

TzMax - Maximum allowed torque value of motor Z

TzMin - Minimum allowed torque value of motor Z

u - Index of constraints

v - Spatial coordinate v

$\underline{\boldsymbol{v}}$ - A vector in N-dimensional space

$\boldsymbol{v_i}$ **, *i* = 1, ..., 6** - i-th coordinate of vector v in 6D space

$\underline{\boldsymbol{v}}_{\boldsymbol{u}}$ - Unit vector of vector v

$\underline{\boldsymbol{V}}$ - Direction vector of a straight line

w - Spatial coordinate w

$\underline{\boldsymbol{x}}^{\boldsymbol{r}}$ - Solution point candidate after the r-th iteration step

$\underline{\boldsymbol{x}}^{*}$ - Minimum location of $f(x)$

x - Spatial coordinate x

$\underline{\boldsymbol{x}}$ - A vector in 6D space

$\boldsymbol{x_k}$ - The k-th coordinate of vector X and x

$\underline{\boldsymbol{X}}$ - A vector in N-dimensional space

$\boldsymbol{X_k}$ - The k-th coordinate variable of an N-dimensional object

y - Spatial coordinate y

$\boldsymbol{y_k}$ - The k-th coordinate of vector $\underline{Y}$

$\underline{\boldsymbol{Y}}$ - A vector in N-dimensional space

z - Spatial coordinate z

$\boldsymbol{\alpha}$ - The angle between two vectors in N-dimensional space

$\boldsymbol{\alpha_s}$ - Significance level

$\boldsymbol{\alpha_t}$ - Angle across side "a" in a triangle in 2D or ND

$\boldsymbol{\alpha_{-\infty}}$ - Limit of α, as the maximum negative rotation angle achievable in response to a negative infinite value of Δ

$\boldsymbol{\alpha_{+\infty}}$ - Limit of α, as the maximum positive rotation angle achievable in response to a positive infinite value of Δ

$\boldsymbol{\beta}$ - Angle across side "b" in a triangle in 2D or ND

$\boldsymbol{\gamma}$ - Angle across side "c" in a triangle in 2D or ND

Δ - Increment added to the i-th coordinate of vector x

Δ_1 - A negative increment such that, when added to the i-th coordinate of vector x, will rotate vector x by 20 degrees

Δ_2 - A positive increment such that, when added to the i-th coordinate of vector x, will rotate vector x by 20 degrees

$\boldsymbol{\Phi}$ - Lagrangian function

$\underline{\nabla}\boldsymbol{Tm_1}$ - Gradient vector of the objective function Tm_1

Contact Information

Szabolcs Sovenyi, Ph.D.
Szabolcs.Sovenyi@stellantis.com
sovenyi@hotmail.com
1 248 962 5344

Dr. Sovenyi works for FCA US LLC, a member entity of Stellantis, in the Chrysler Technology Center in Auburn Hills, Michigan, USA as a powertrain torque controls subject matter expert. Previously, he worked on projects sponsored by Volvo, Rolls Royce and Associates, General Motors and the Truck Technology Team of Eaton Corporation. He has a Ph.D. in mechanical engineering from the University of Michigan, Ann Arbor and two M.Sc. degrees in mechanical engineering from the University of Massachusetts, Amherst and the Technical University of Budapest. His interests include creating mathematical models for electromechanical and hydraulic systems, and writing numerical algorithms for signal processing, system identification, spectral analysis, wavelet analysis and optimization.

Acknowledgments

The authors thank FCA US LLC, a member entity of Stellantis, for keeping exciting projects in house and fostering an engineering work culture of innovation, exploration, minimal bureaucracy and high autonomy. Also, many thanks to ePT supervisors Andryas Mawardi, Ph.D. and Ravi Shah and ePT manager Nadirsh Patel for making this paper possible.

Author Affiliation

Szabolcs Sovenyi, Ph.D. works for FCA US LLC, a member entity of Stellantis, as an electrified powertrain torque control subject matter expert.

Feisel Weslati, Ph.D. also works for FCA US LLC, a member entity of Stellantis, as the senior manager of the Electrified Powertrain Team.