POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

BOSCH

In collaboration with Robert Bosch GmbH Corporate Research

Executive Summary of the Thesis

Global optimization of pulse patterns for an electrical drive via Set Membership methods

Laurea Magistrale in Automation and Control Engineering - Ingegneria dell'Automazione

Authors: Mattia Alborghetti, Giulio Montecchio

Advisor: Prof. Lorenzo Fagiano

Co-advisors: Stefan Gering, Martin Loehning, Maximilan Manderla

Academic year: 2022-2023

---

1. Introduction

In the field of modulation techniques in electrical drives, the optimization of pulse patterns is a promising research area [1]. The modulation via Optimal Pulse Patterns (OPP) computes the switches of the inverter as the optimization variables of an optimal control problem. In this work, the objective is the minimization of the distortion of the currents flowing in the coils of the electric motor. This problem is challenging due to its high dimensionality and non-convexity.

Set Membership Global Optimization (SMGO) is an innovative optimization method that is characterized by its global nature. Its functioning is black-box and data-driven, hence an explicit model of the cost function is unnecessary [5]. In this work, the SMGO method is enhanced with some novel concepts and mechanisms. Moreover, these modifications produce a method applicable to the OPP problem. Finally, the effectiveness of SMGO is compared with two other methods: gradient-based optimization, representing the state of the art, and Bayesian optimization, which is a well-known global optimizer.

2. Enhancing SMGO

The principles of SMGO are vastly reviewed in [5]. This method fits a Set Membership (SM) model to the cost and constraint functions, which are black-box functions defined on the *search space*, i.e. the space of the optimization variables. Such a surrogate model is then refined through wise handling of the budget of function evaluations. This last term refers to a simultaneous evaluation of the cost and all the constraints, that is performed once per iteration and produces a known point in the search space. This point is defined as *sample*. The algorithm chooses its next sample according to two different modes:

1. *Exploitation*, trying to find a better point.
2. *Exploration*, in regions where the cost function is uncertain.

The balance of these two modes is key to have a high rate of convergence towards the global optimum. On every new iteration n, the SM-model is updated on a subset of the search space, i.e. the set of candidate points $\boldsymbol{E}^{\langle n \rangle}$. This set is enlarged at each iteration according to a specific *points generation* mechanism.

2.1. Enforcement of linear constraints

The first addition to SMGO is the ability to account for linear constraints. This feature is important in the OPP problem, where the requirement of a sequence of ordered switches can be translated into a set of linear constraints. Generally, a set of linear constraints on the optimization variables $\boldsymbol{x}$ defines a convex polytope $\mathcal{X}$ of feasible points [2] . To easily enforce these constraints, it is possible to modify the candidate points generation, so that the search space of the algorithm is practically limited to $\mathcal{X}$. As a consequence, the *Sobol* distribution implemented in SMGO to populate the set $\boldsymbol{E}^{\langle n \rangle}$ is no longer good. This distribution of pseudo-random points offers good coverage of the hyper-rectangular space, but is not suited for a polytopic $\mathcal{X}$. Instead, the points generated from the symmetric *Dirichlet* distribution can be used to cover uniformly a *simplex* [3]. The density function of this symmetric distribution is

$$\mathrm{Dir}(\boldsymbol{q}, a) = \frac{\Gamma(m\, a)}{\Gamma(a)^m} \prod_{i=1}^{m} q_i^{a-1}, \tag{1}$$

where $\Gamma(a)$ is the gamma distribution with the shape parameter a and vector $\boldsymbol{q}$ contains the coordinates of the point on the *standard simplex*, i.e. the *(m-1)*-simplex with the standard unit vectors of $\mathbb{R}^m$ as vertices. If $a = 1$, the distribution on the standard simplex is uniform. This uniform distribution can then be mapped into a uniform distribution in any simplex of dimension $m - 1$.
Therefore, $\mathcal{X}$ is decomposed into the product of simplices and in all these simplices a uniform distribution of points is obtained through (1). Finally, recomposing the original polytope, a uniform distribution over the full $\mathcal{X}$ is obtained.

2.2. Extended trust region

It is observed from the application of SMGO to the OPP problem, that the optimization suffers of *over-exploitation*, i.e. it wastes too many iterations refining a local minimum. To fight this issue, a modification is applied. Instead of limiting the exploitation in the surroundings of the current best point, the enhanced SMGO exploits always the full $\boldsymbol{E}^{\langle n \rangle}$. Previously a better point in a different region could be found only via exploration, i.e. searching in regions with high uncertainty. With this change, it is easier for the algorithm to jump from a local optimum to a superior one in a different region: Exploitation could eventually reach these minima, enhancing the rate of convergence.

2.3. Adaptive alpha

The trade-off between exploitation and exploration is a crucial feature for the rate of convergence of a global optimizer. The best compromise depends on the user's objectives. In SMGO, this trade-off is controlled by the tuning parameter α, which acts as a threshold that limits the exploitation. However, this α-tuning has various drawbacks. The application of a fixed α limits the accuracy of the optimum that the algorithm can possibly achieve. Moreover, the relationship between α and the amount of exploitation performed is highly dependent on the specific cost function. Proper tuning can be achieved only via trial and error.
A new approach is here proposed. The value of α is set to be the result of a discrete PI controller, that adjusts its value throughout the optimization. The user's preference is now accommodated through R_{ref}, which represents a reference for the value of

$$R(n) = \frac{N_\phi(n)}{N_\theta(n)}. \tag{2}$$

At iteration n, (2) describes the ratio between the amount of iterations spent for exploration over the amount of iterations spent for exploitation. Therefore, after defining the error

$$e(n) = R_{\text{ref}} - R(n), \tag{3}$$

the PI controller that outputs α is

$$\alpha(n) = \max\left(K_P\, e(n) + K_I\, \Sigma_e(n),\, \alpha_{\min}\right), \tag{4}$$

where K_P and K_I are the proportional and the integral gain, $\alpha_{\min}$ corresponds to the minimum value of α and $\Sigma_e(n)$ is the integral error. The update of $\Sigma_e(n)$ is characterized by a clamping anti-windup:

$$\Sigma_e(n+1) = \begin{cases} \Sigma_e(n), & \alpha(n) = \alpha_{\min} \\ \Sigma_e(n) + e(n) & \text{otherwise.} \end{cases} \tag{5}$$

The primary purpose of this structure is not to keep the error $e(n)$ close to 0, but to adapt the value of α to a large variety of situations in terms of different SM models and different distributions of candidate points.

2.4. Sunburst points generation

The default mechanism for the update of the data set $\boldsymbol{E}^{\langle n \rangle}$ is the *spiderweb generation* [5]. However, in this mechanism the number of candidate points increases following the law

$$n(B-1)\left(2D + \frac{n-1}{2}\right), \tag{6}$$

where n is the iteration, B is the user-defined gridding granularity and D is the dimension of $\mathcal{X}$. Hence, it is polynomial w.r.t. the number of samples n. The pulse pattern optimization is carried out offline, so there are ideally no limits on the number of iterations that can be performed, and therefore of samples that can be collected. In this work, a new candidate points generation is introduced. The *sunburst generation* adds to $\boldsymbol{E}^{\langle n \rangle}$ only the midpoints of the segments along the coordinate directions and between the last sample and the N_{cdpt} closest candidates, represented in orange in Figure 1. With this method, the number of candidate points generated at each iteration is fixed, and the total number of candidate points grows linearly with n, according to the law

$$n(2D + N_{\text{cdpt}}). \tag{7}$$

The tuning parameter N_{cdpt} offers a further degree of freedom to the user.
The sunburst generation speeds up the algorithm, which is vital for $D > 5$. On top of that, the distribution of candidate points gets finer only in the most interesting regions.
The enhancement to SMGO can be assessed

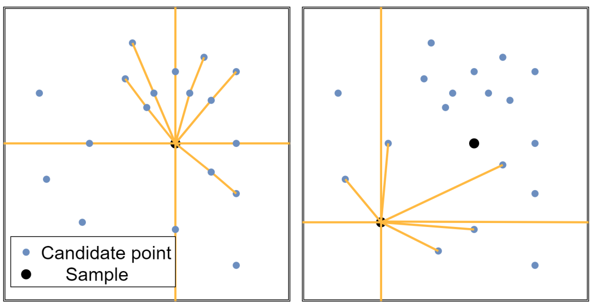

Figure 1: Two consecutive iterations of sunburst generation.

on the 3D OPP problem. Table 1 and Table 2 report the results of different tunings of the exploitation vs exploration trade-off. For clarity, in Table 2 the tuning parameter R_{ref} is substituted with Ξ, which is defined as

$$\Xi = \frac{1}{R_{\text{ref}} + 1} \tag{8}$$

and describes the fraction of the budget that should be preferably reserved for exploitation.
The metrics presented in the Tables are: **Mean**, the mean optimum found by SMGO over ten trials, **Exploitation**, the fraction of exploitation out of the full budget and **Converged**, the number of trials that converged to the global optimum, which is known to be 32.373. In the captions is also reported the mean *self-time*, i.e. the run time of the algorithm only, without the time for cost and constraint evaluation.

$\boldsymbol{\alpha}$	**Mean**	**Exploitation**	**Converged**
0.001	36.563	293.5/500	5/10
0.005	37.646	108.4/500	4/10
0.01	38.339	66.8/500	4/10

Table 1: Original SMGO for different tuning of α. Average self-time: 68.3 s.

$\Xi[\%]$	**Mean**	**Exploitation**	**Converged**
50%	33.627	179.7/500	9/10
33%	33.358	151.0/500	9/10
17%	32.574	69.5/500	10/10

Table 2: Enhanced SMGO for different tuning of R_{ref}. Average self-time: 2.0 s.

The improvement with respect to the original version is manifold: The self-time is greatly reduced, the number of convergent optimums is increased and the mean optimum is generally closer to the global one. Notice how the indicator Ξ does not correspond exactly to the fraction reported in the third column. This happens because factors other than α influence the exploitation, e.g. the available candidate points.
Analogous results are confirmed also on a set of 14 benchmark functions, taken from literature on global optimization.

3. Pulse pattern optimization

The goal of OPPs is to minimize the total losses of the electric drive system. The optimization in this work is based on the *Total Harmonic Distortion* (THD) concept. The THD is a performance criterion computed with the power spectrum of a signal. In particular, it compares the magnitude of the first harmonic with the higher harmonic content. This methodological choice follows the literature on this topic [4], as much as recent trails of research [1], and even if it is a simplified approach, still represents a valid and relevant starting point. The following multi-objective cost function is used:

$$F(\boldsymbol{\sigma}) = \sqrt{\sum_{i=2}^{N} (h^i(\boldsymbol{\sigma}))^2} + Qh^1(\boldsymbol{\sigma}). \qquad (9)$$

Here, the optimization variable $\boldsymbol{\sigma} \in \mathbb{R}^D$ is a sequence of switching angles that define the pulse pattern. $\boldsymbol{\sigma}$ is linked to the amplitude of the i-th spectrum harmonic of the electrical machine phase current $h^i(\boldsymbol{\sigma})$ by a non-linear dynamical relationship. The first term penalizes the harmonic content of the current, while the second favors the lowest fundamental current that ensures a certain mean torque, enforced via black-box constraint. The parameter Q is a weight empirically tuned. Overall, the problem results to be a challenging *constrained non-convex optimization*.

4. Comparative analysis

4.1. Setup of the methods

In order to assess SMGO performance, the algorithm is compared with two well-established optimization methods: gradient-based multistart and Bayesian optimization.

The former represents the state of the art in the industry and the latter is a well-established black-box global optimization method.

Gradient-based methods make use of the first- and second-order optimality conditions. They are very powerful, but they exploit only local information in the quest for a minimum, converging to local minima in case of non-convex optimization problems. In order to find the global optimum of a non-convex cost function, the gradient-based local optimization must be initialized on different points. This structure gives the name *multistart* to the method. In the thesis, the initialization point is drawn from the Dirichlet random distribution, which enforces the linear constraints on the starting point.

The optimizer is `fmincon` of `MATLAB`. Among the manifold of gradient-based approaches, this is a Sequentially Quadratic Programming (SQP) that makes use of the quasi-newton method Broyden–Fletcher–Goldfarb–Shanno (BFGS) to approximate the Hessian matrix. In addition, the computation of the gradient is done with the central difference approach. Details on these features are discussed in [2] in an exhaustive manner.

Differently, *Bayesian optimization* is a global black-box optimization method, like SMGO. Nevertheless, they approximate the value of the cost function with two different surrogate models. SMGO uses the SM-model, while Bayesian fits a Gaussian process on the evaluated samples. The problem is solved with the `MATLAB` function `bayesopt`. Lastly, SMGO solutions make use of the enhancements described in Section 2.

4.2. Comparative results

In this comparative study, every algorithm is tested for ten trials, with a given number of function evaluations. The first chart that we report summarizes in a synthetic way the results obtained with the different methods. The square represents the mean value of the optimum found over all the trials, with the bracket delimited by the largest and the smallest optimum. The blue numbers tell the average amount of local gradient-based optimizations carried out in the multistart optimization. The second chart shows the performance of each method in terms of total running time. For bars out of scale, the number is reported on top.

Results for the 3D OPP problem case are contained in Figure 2.

All the solvers reach the same global minimum, with a reasonable budget of function evaluations. SMGO converges with a smaller number of function evaluations compared to multistart. Bayesian employs even fewer evaluations, but it requires way more computational time, one order of magnitude higher than SMGO. Overall, SMGO for this low dimensional case can be considered the best one, in the trade-off between computational burden and convergence rate.

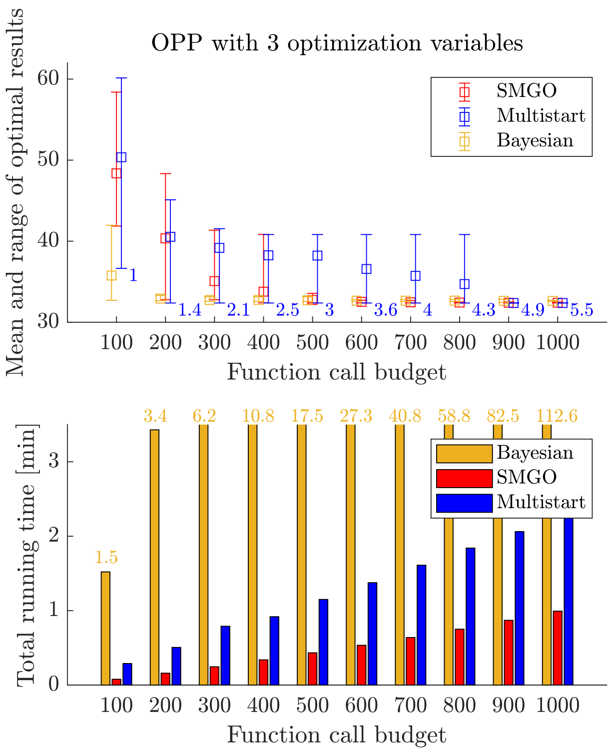

Figure 2: Comparison charts for $D = 3$.

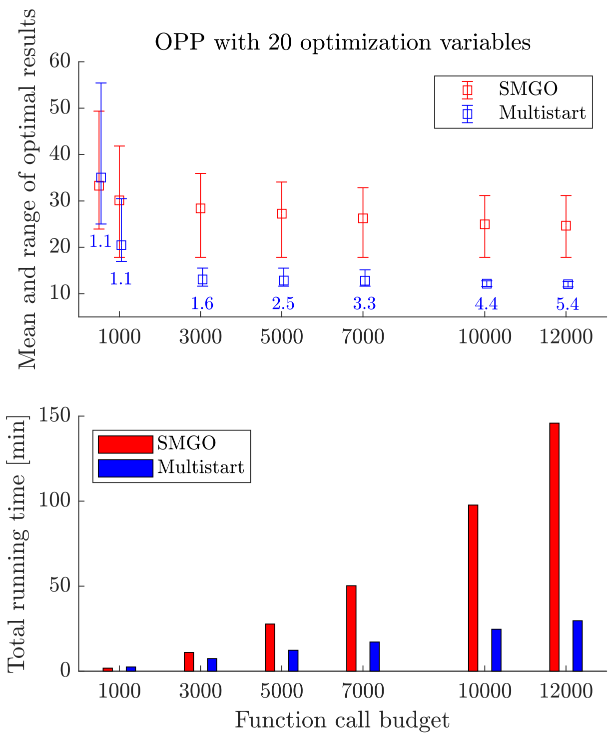

Figure 3: Comparison charts for $D = 20$.

Note that, in multistart optimization, the whole budget is just sufficient to initialize the gradient-based optimization in a few different points; This can be inferred from the blue numbers of the first chart.
It is known that the OPP problem requires more than three optimization variables to have interesting results, especially for higher motor speed. Up to five switching angles, SMGO maintains a good quality-efficiency trade-off and result. However, starting from $D = 6$, SMGO performance deteriorates. In the following, the comparative charts for $\boldsymbol{\sigma} \in \mathbb{R}^{20}$ are presented. This is an interesting case, because applications in the automotive field require similar number of switches.
Notice that Bayesian optimization is not even applied for this high dimension. The requested time is prohibitive and, in addition, it suffers in finding the region feasible for the linear constraints.
For budgets of a few hundred cost function evaluations, SMGO provides slightly better performance in terms of mean and variability of the resulting optima, and also in terms of the computational burden. Nonetheless, with larger budgets, the roles are reversed: multistart can converge very reliably to a better minimum, in a short time. Looking at Figure 3, we notice that multistart can converge with just between 5 and 6 random initializations of local gradient-based optimization.
Therefore, the problem at hand can be considered smooth enough. Normally, for this kind of problem, gradient-based methods are the best choice. Nevertheless, the OPP can be computed with different cost and constraint functions. A cost function with a more accurate loss model or different constraints can lead to the failure of gradient-based methods, due to non-convexity and discontinuity. Hence, SMGO could still be a superior method with respect to gradient-based for different formulations of the OPP problem.

5. Conclusion

The Optimal Pulse Patterns are specific drive modulations able to dramatically reduce the harmonic distortion introduced by the power converters in the electrical machines for steady-state operating points. For low dimensions of the THD-based OPP problem with one inequality constraint on torque, SMGO appears to be a good option. It is faster than Bayesian and multistart, and the budget of function evaluations it requires is in-between the two methods. However, in practical terms, OPPs modulation requires a high number of switching angles, that correspond to a high number of optimization variables. In this case, SMGO does not perform better than state-of-the-art gradient-based multistart, because the formulated problem is quasi-smooth. On the other side, SMGO shows the ability to handle the problem where other global optimization methods fail. These interesting results are achieved thanks to several modifications to the original SMGO implementation. An extended trust region, an adaptive alpha, and a sunburst candidate point generation mechanism reveal to be enhancements to the solver, that becomes faster and less memory demanding. These changes are also tested on 14 benchmark functions that prove the generality of the enhancements. In conclusion, SMGO turns out to be appealing for two branches of optimization:

- **offline optimization** in case of few optimization variables, expensive and time-consuming cost function evaluation or highly non-smooth problems.
- **online optimization** where the real bottleneck is a low budget of available function cost evaluations.

References

[1] F. Berkel, M. Löhning, and S. Reimann, *Loss optimal pulse patterns for electrical drives*, tech. rep., Robert Bosch GmbH, 2022.

[2] L. Fagiano, *Constrained Numerical Optimization for Estimation and Control.* Lecture Notes, Politecnico di Milano, September 2021.

[3] B. A. Frigyik, A. Kapila, and M. R. Gupta, *Introduction to the dirichlet distribution and related processes*, tech. rep., University of Washington, 2010.

[4] T. Geyer, *Model Predictive Control of High Power Converters and Industrial Drives*, John Wiley and Sons, Inc., 2016.

[5] L. J. Sabug, *On data-driven optimization in the design and control of autonomous systems*, PhD thesis, Politecnico di Milano, 2023.