

MANIFOLD MAPPING OPTIMIZATION WITH OR WITHOUT TRUE GRADIENTS

Benoît Delinchant, D. Lahaye, Frédéric Wurtz, Jean-Louis Coulomb

▶ **To cite this version:**

Benoît Delinchant, D. Lahaye, Frédéric Wurtz, Jean-Louis Coulomb. MANIFOLD MAPPING OPTIMIZATION WITH OR WITHOUT TRUE GRADIENTS. Electrimacs 2011, Jun 2011, Cergy-Pontoise, France. hal-00600301

HAL Id: hal-00600301
https://hal.science/hal-00600301

Submitted on 21 Jan 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Manifold mapping optimization with or without true gradients

B.Delinchant[1], D.Lahaye[2], F.Wurtz[1], J.-L. Coulomb[1]

1. Electrical Engineering Lab - G2ELab, Grenoble University, France.
2. Numerical Analysis Unit, Delft University of Technology, The Netherlands
e-mail: benoit.delinchant@G2ELab.grenoble-inp.fr, d.j.p.lahaye@tudelft.nl

Abstract – This paper deals with the Space Mapping optimization algorithms in general and with the Manifold Mapping technique in particular. The idea of such algorithms is to optimize a model with a minimum number of calls using a less accurate but faster model. In this optimization procedure, fine and coarse models interact at each iteration in order to adjust themselves in order to converge to the real optimum. The Manifold Mapping technique guarantees mathematically this convergence but requires gradients of both fine and coarse model. Approximated gradients can be used for some cases but are subject to divergence. True gradients can be obtained for many numerical model using adjoint techniques, symbolic or automatic differentiation. In this context, we have tested several Manifold Mapping variants and compared their convergence in the case of real magnetic device optimization.

Keywords – Space Mapping, Manifold Mapping, Optimization, Surrogate model, Gradients, Symbolic derivation, Automatic differentiation.

1. Introduction

The space-mapping technique [1] allows computationally expensive simulation based optimization procedures to be speeded up through the use of approximate models. In the space mapping literature the so-called fine and coarse models are conveived as mappings from the design space to the space of model responses. The key element is the space mapping function. It reparametrises the coarse model domain in such a way to minimize the discrepancy between the fine and coarse model responses. The composition of the space-mapping function and the coarse model response defines a surrogate for the fine model. Instead of solving the fine model problem directly, space-mapping solves the surrogate optimization problem through a sequence of approximations of the space-mapping function. This in turn defines a sequence of coarse model optimization problems whose solution by definition converges to the space-mapping solution. The computational efficiency of this procedure stems from the fact that it takes less fine model evaluations to converge than it takes to solve the fine model optimization problem. The drawback is that the space-mapping solution does not necessarily coincide with the fine model optimum.

In the manifold-mapping technique [2], the surrogate model is constructed in such a way that in a neighbourhood of the fine model optimum, the surrogate model response closely ressembles its fine fine model counterpart. This guarantees that the solutions of the surrogate and fine model optimization problem do coincide. The space-mapping function is replaced by the so-called manifold-mapping function. The latter is an affine transformation between the tangent manifolds of the fine and coarse model image spaces. Manifold-mapping is computationally as efficient as space-mapping.

Space mapping techniques have been used in electromagnetic device optimization for several years now [8][12]. Different techniques can be used, but manifold mapping, which is the only one proved to converge to the fine model optimum is always using approximated gradients of the fine model since true gradients are not always available.

This paper details the manifold mapping technique and argues that exact gradients can be available more or less easily nowadays. The computational cost of these gradients is generally small compared with that of the fine model, and the convergence of the manifold mapping algorithm is improved. This property will be required in the future when optimization specifications becomes more and more constrained.

2. Manifold mapping algorithm

2.1. Mathematical background

Let us consider an optimization problem with design variables x in the design space $x \in X \subset \Re^n$ and specifications $y \in \Re^m$ which can be approximated by minimizing a cost functional $F(x) \in \Re$ (e.g. equation 4).

The manifold-mapping function $S : c(X) \mapsto f(X)$ is a mapping between the coarse model $c(X) \subset \Re^m$ and fine model $f(X) \subset \Re^m$ image spaces. This function maps the point $c(x_f^*)$ to $f(x_f^*)$ and the coarse model tangent space at $c(x_f^*)$ to the fine model tangent space at $f(x_f^*)$. It allows to define the surrogate model $S(c(x))$ and to write the manifold-mapping solution as follows:

$$\begin{aligned} &find\ x_{mm}^* \in X \quad such\ that \\ &x_{mm}^* = \arg\min_{z \in X} \left\| S(c(z)) - y \right\| \end{aligned} \tag{1}$$

The manifold-mapping function $S(x)$ is approximated by a sequence $\{S_k(x)\}_{k \geq 1}$ yielding a sequence of iterands $\{x_{k,mm}\}_{k \geq 1}$ converging to x_{mm}^*. The individual iterands are defined by coarse model optimization:

$$\begin{aligned} &find\ x_{k,mm}^* \in X \quad such\ that \\ &x_{k,mm}^* = \arg\min_{z \in X} \left\| S_k(c(z)) - y \right\| \end{aligned} \tag{2}$$

At each iteration k, the construction of S_k is based on tangent planes of coarse and fine model, i.e., $S_k = J_c(x_k^*).J_f^+(x_k^*)$ where matrices $J_c(x_k^*)$ and $J_f^+(x_k^*).$ of size $m \times n$ are $c(x_k)$ Jacobian, and pseudo inverse of the $f(x_k)$ Jacobian, respectively. The pseudo inverse can be computed by a simple QR decomposition or using the singular value decomposition.

If the Jacobians are not available, S_k can be approximated using ΔC and ΔF of size $m \times \min(k,n)$ defined as follow:

$$\Delta C = [c(x_k) - c(x_{k-1}),\ c(x_k) - c(x_{k-2}), \cdots, c(x_k) - c(x_{\max(k-n,0)})]$$
$$\Delta F = [f(x_k) - f(x_{k-1}),\ f(x_k) - f(x_{k-2}), \cdots, f(x_k) - f(x_{\max(k-n,0)})]$$

During the first n iterations, these matrices are not fully describing the tangent planes but are enough to define a search direction until k becomes greater than n.

In order to improve robustness of the approximation S_k is defined with a complementary term $S_k = \Delta C(x_k^*).\Delta F^+(x_k^*) + (I - U_{k,c}U_{k,c}^T)$ where $U_{k,c}$ is provided by the singular value decomposition of $\Delta C = U_c.\Sigma_c.V_c^T$ at each k iteration.

Using S_k mapping function, an update objective $y_k = c(x_k) - S_k\left(f(x_k) - y\right)$ can be introduced leading to an asymptotically equivalent problem:

$$\begin{aligned} &find\ x_{k,mm}^* \in X \quad such\ that \\ &x_{k,mm}^* = \arg\min_{z \in X} \left\| c(z) - y_k \right\| \end{aligned} \tag{3}$$

In other words, the mapping is transfer from the coarse model to the optimization objectives which leads to an easier algorithm implementation. And by construction $x^*_{k,mm} = x^*_f$ corresponding to the fine model optimum.

A trust region strategy has to be implemented in addition to this algorithm in order to prevent arbitrary large step size $x_k - x_{k-1}$. To prevent manifold mapping from taking such steps, a trust-region stabilization was presented in [13].

3. Validation on a simple test case

A first test case has been computed in order to show that algorithm proposed can find good results. The problem is defined by 3 degrees of freedom and an objective function is a least square functional with 3 components. A coarse model has been defined using minor modifications in some formulas in order to approximate the fine model.

The objective function has been plotted according the optimization iterations for several space mapping strategies:

- OM : Output Mapping, which is a manifold mapping with the identity matrix for the mapping function S. The objective update is then $y_k = y + c(x_k) - f(x_k)$.
- MM Approx: Manifold mapping using a tangent plane approximation for ΔC and ΔF.
- MM Approx without SVD: Manifold mapping using a tangent plane approximation without correction : $S_k = \Delta C(x^*_k).\Delta F^+(x^*_k)$.
- MM Approx trust: Trust Region manifold mapping in order to ensure convergence.
- MM Exact: Manifold mapping using true gradients.
- MM Exact Trust: Trust Region manifold mapping in order to ensure convergence.

Fig. 1 shows that MM with true gradient converge extremely fast to a good solution, while OM and MM with approximated gradients converge slower with some estimations. Trust region leads to slower convergence but in the case of MM with approximated gradient the same solution is reached with only one iteration more. In the case of approximated gradient correction term $I - U_{k,c}U^T_{k,c}$ in the mapping function, is important.

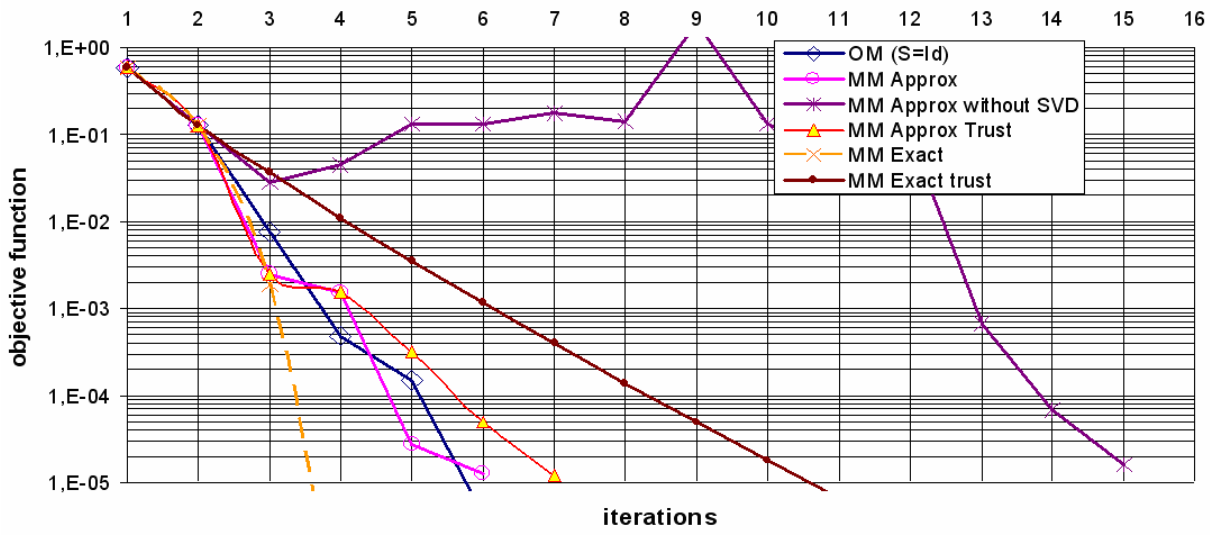


Fig. 1. Optimization algorithms convergence on a simple test case.

4. Manifold mapping on a real test case

4.1. Device description

Functional design goal is to produce MEMS based translation to rotation contactless transducer, with linear law. It can be used for sensors in order to have very sensitive position measures.

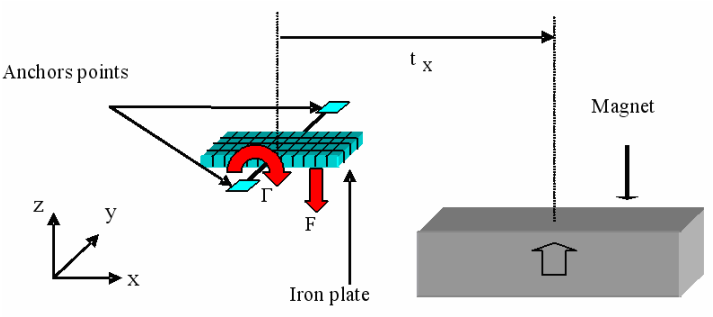


Fig. 2. Magnetic MEMS topology

A MEMS magnetic actuator topology has been defined to reach these requirements Fig. 2. It is made of 2 parts:

- mobile magnet with x-axis translation degree of freedom,
- an iron plate with y-axis rotation degree of freedom.

4.2. Optimization specifications

The objective is to find mobile magnet dimensions in order to obtain a torque as linear as possible. To do this, a least squares objective functional is defined in Equation 4 and Fig. 3:

$$F(x)=\sqrt{\sum_i\left(\Gamma(tx_i)-\Gamma_{\max}.\frac{tx_i}{tx_{\max}}\right)^2} \tag{4}$$

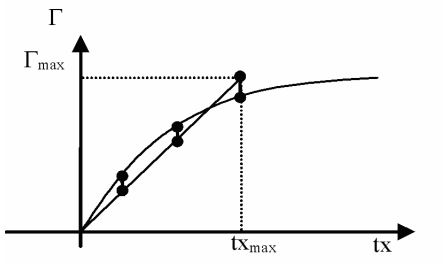


Fig. 3. Design specification: torque has to be linear

3 torque computations, equally distributed along translation position, have been chosen in order to do the least square minimization.

Table I. Design variables and constants

Parameter	**Values**
Magnet width	[1;25] mm
Magnet high	[1;25] mm
Magnet length	[1;30] mm
Iron plate width	600 µm
Iron plate high	8 µm
Iron plate length	600 µm
Magnet Polarization	1 T
Gap between magnet and iron	1 mm

4.3. Fine modeling

For the mobile magnet, a Coulombian equivalent charge approach is used in order to compute magnetic field applied on the ferromagnetic plate.

For this last one, a steady-state Method of Moments (MoM) is applied. It consists in the meshed of ferromagnetic bodies along the X, Y and Z axes, with uniform induced magnetized elementary blocks [4]. This method does not require to mesh the air and is particularly efficient for "radiating" systems. The issues with such an integral method are full matrices and computation memory limitation.

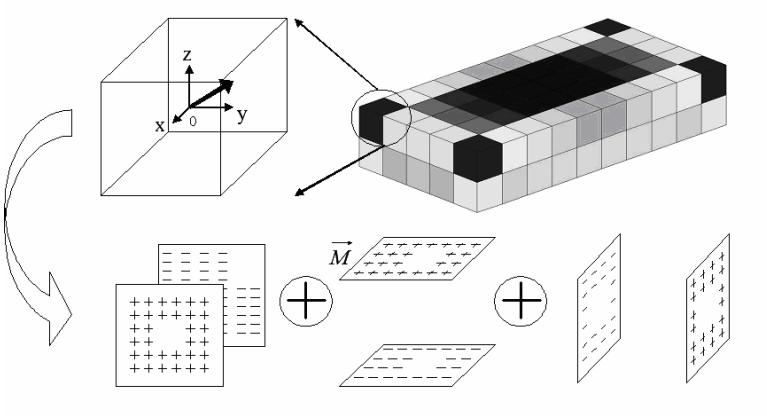


Fig. 4. Surface charge method to compute torque when magnetization of each bloc is known.

Fig. 4 shows the magnetization of each block which depends on the external field (H_{ext} produced by field sources such as magnets or conductors) as well as on the field produced by other blocks depending on their own magnetization.

$$H = H_{ext} + Q \cdot M \tag{5}$$

Blocks interaction is defined by the interaction matrix (Q in equation 5) : Q is a square matrix of size (3m)x(3m) (m = number of blocks), composed of 3x3 square matrices which represent the magnetic excitation created by a block to an other.

The ferromagnetic material behavior law is defined by non linear law, parameterized by the saturation induction and initial permeability of the material. The unknown ferromagnetic magnetizations can then be found by using a Newton-Raphson solver with a relaxation method to ensure convergence.

When magnetizations are known, each block can be seen as a set of parallelepiped magnets (Fig. 4), in order to compute the magnetic field or force and torque (equation 6).

$$\mathbf{\Gamma} = \mu_0 \sum_{i=1}^{N} \sum_{j=1}^{6} \iint_{s_{ij}} \mathbf{OP}_{s_{ij}} \times \left[\sigma_{s_{ij}} . \mathbf{H_{ext}} (\mathbf{P}_{s_{ij}}) \right] ds \tag{6}$$

where O is the pivot point where the torque is computed, P_{ij} are barycenter of each bloc and σ_{ij} there equivalent surface charge derived from their magnetization.

Computation time of this model depends on the number of blocks (see Fig. 5). Computation is fast, a good accuracy is reached in less than one minute, but we have always to think about increasing model computation and optimization time if our objective is system simulation and design.

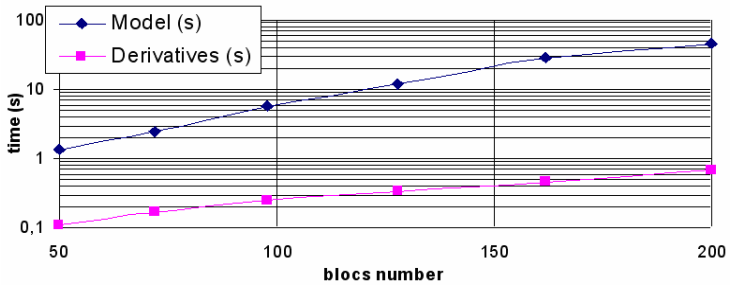


Fig. 5. Fine model computation time and its derivatives, depending on the discretizing.

4.4. Coarse modeling

The iron plate is not discretized and global demagnetisation coefficients in 3D space are computed analytically using classical rectangular shapes formulas. Under the assumption that the plate is saturated along the x-axis, the magnetization along z-axis remains in the linear domain. It is then possible to solve explicitly the 2 magnetizations M_x and M_z in order to compute torque Γ_y by equation 7.

$$\Gamma_y = \mu_0 . V_{plate} . \left(- Mx . H_0 + Mz . H_0 \right) \tag{7}$$

This model is fully analytical and an optimization using its gradients is less than one second

4.5. Modeling comparison

A FEM simulation has been done but can not be considered as a reference (just a good approximation) due to the mesh issues for such thin geometries.

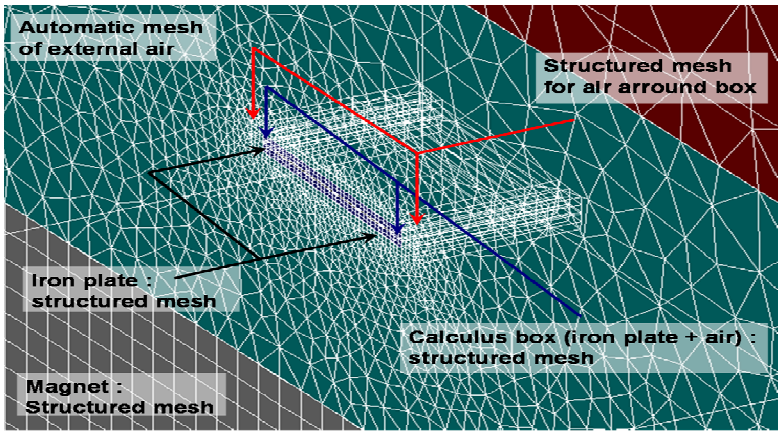


Fig. 6. Difficult mesh in 3D FEM software Flux3D™

The fine model was tuned using a variation of the discretization parameter in order to appreciate the accuracy convergence (see Fig. 7). This tuning could be used to produce both fine and coarse modelling but it was decided in this work to compare manifold mapping technique with two different kinds of model, one analytical and the other numerical.

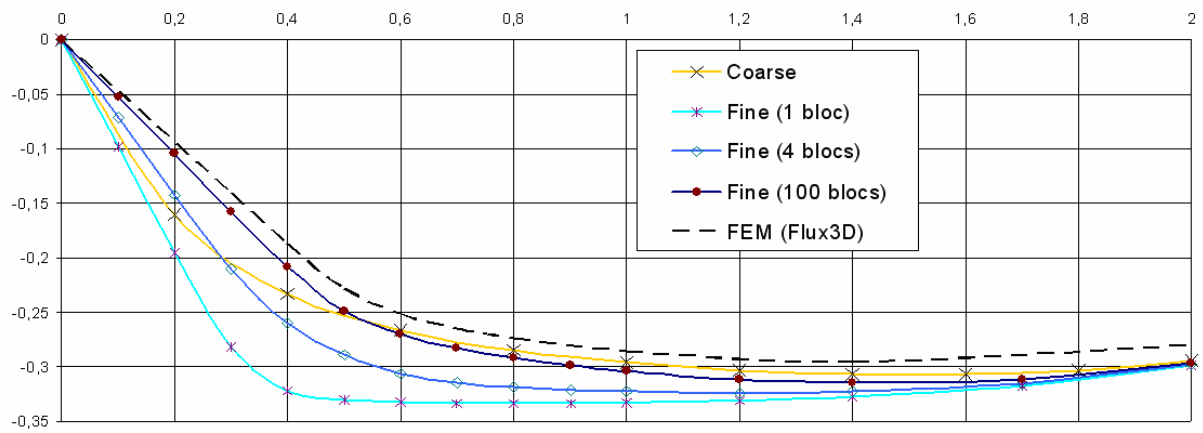


Fig. 7. Modeling comparison on torque regarding mobile magnet position for both Fine and Coarse models but also with FEM

5. Model derivation

Nowadays, many techniques and tools are available in order to compute model Jacobian of analytical model and numerical models. Two kind of derivation techniques exist, the symbolic one which examines computation model in order to simplify as far as possible the derivatives expressions; and the automatic one which is more systematic but less efficient.

5.1. Symbolic derivation

First is the symbolic derivation based on mathematical theorems which tries to express derivatives of a model for each kind of modeling methods. For example, if an unknown 'I' is expressed by numerical integration (eq 8) partial derivatives of function *I* regarding parameters *p* can be given by (eq 9).

$$I(p) = \int_{lo(p)}^{up(p)} f(p,x).dx \tag{8}$$

$$\begin{aligned} \frac{\partial I(p)}{\partial p} = & \int_{lo(p)}^{up(p)} \frac{\partial f(p,x)}{\partial p}.dx \\ & + \frac{\partial up(p)}{\partial p} \cdot f(p,up(p)) \\ & - \frac{\partial lo(p)}{\partial p} \cdot f(p,lo(p)) \end{aligned} \tag{9}$$

Other well known applications of such kind of derivation are :

- implicit theorem [3]: which allows symbolic derivation of unknwons computed using implicit solver such as Newton-Raphson procedure.

- Adjoint method [5]: which allows performant computing of multiple calls to gradients using a dual code which depends on the numerical problem solved (linear system, ...) .

5.2. Automatic derivation

Defining derivative of each basic mathematical operators can lead to automatic differentiation (AD) tools [6]. Typically, AD can be implemented using either the operator overloading (ADOL-C, CppAD, etc) or the source transformation technique (ADIFOR, ADiJaC, etc). In operator overloading one overloads the operators which

are applied on new variable types, with the routine call performing the actual derivative computation. The source transformation approach examines the source code of the original function and generates new code that computes the desired derivative together with the original function value.

5.3. Software implementation

CADES framework[1] [7] implements symbolic and automatic differentiation techniques. So it is easy for several kind of models to add Jacobian computation in order to perform optimization.

This software will be used in this work in order to produce Jacobians on both coarse and fine model.

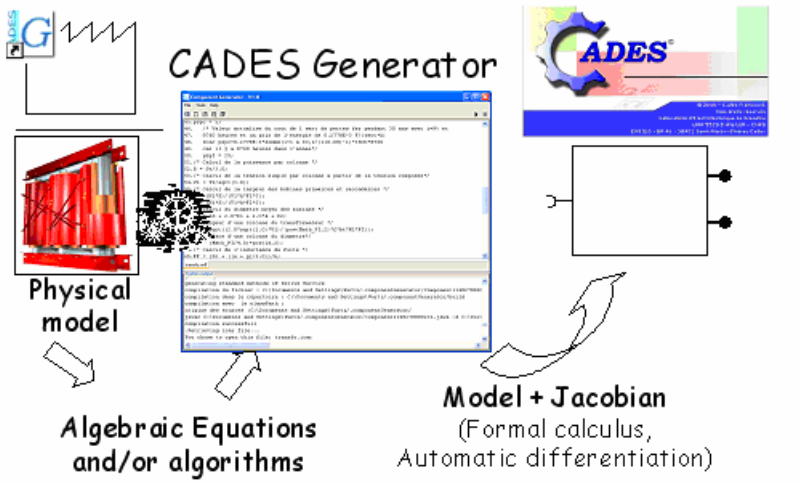


Fig. 8. CADES Generator: an automated tool performing symbolic and automatic differentiation composition and produce software component.

5.4. Model derivation

As defined in the previous section, several techniques are available in order to compute gradients. For the fine modeling, derivatives have been computed using implicit theorem and an adjoint code [5] leading to a low cost compared to the computation of the magnetization vector itself. This is also due to the fact that no non-linear solving procedure is required. For the coarse model which is essentially based on analytical equations, a simple derivation has been done automatically using CADES framework. The torque gradient with respect to the magnet position is plotted in Fig. 9.

[1] CADES framework : Component Architecture for the Design of Engineering Systems (available : http://forge-mage.g2elab.grenoble-inp.fr/project/cadesframework)

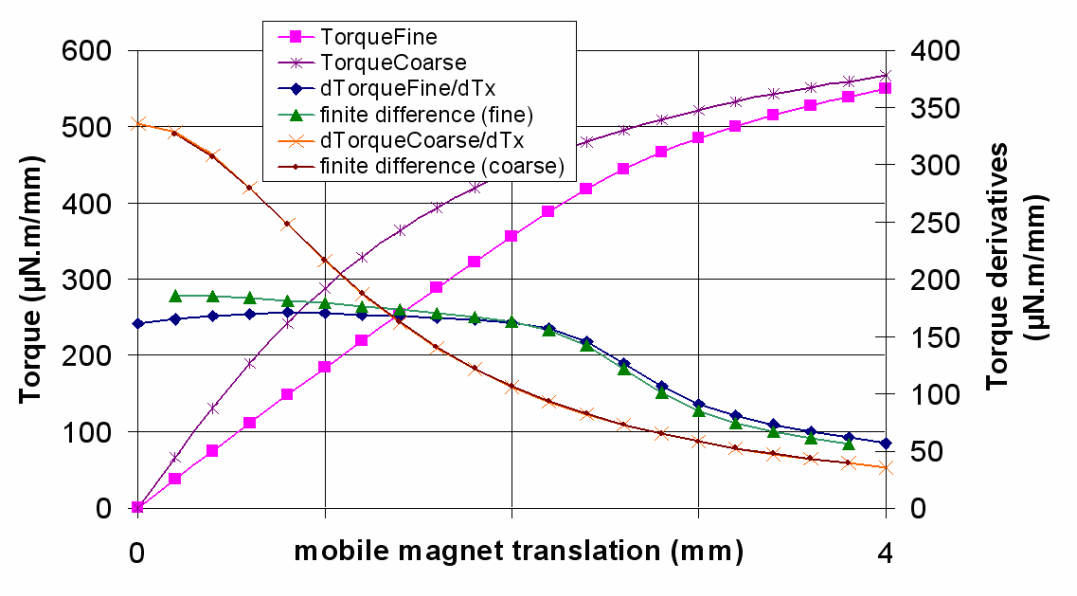


Fig. 9. Fine and coarse torques and their formal derivatives, versus the magnet translation.

5.5. Optimization results

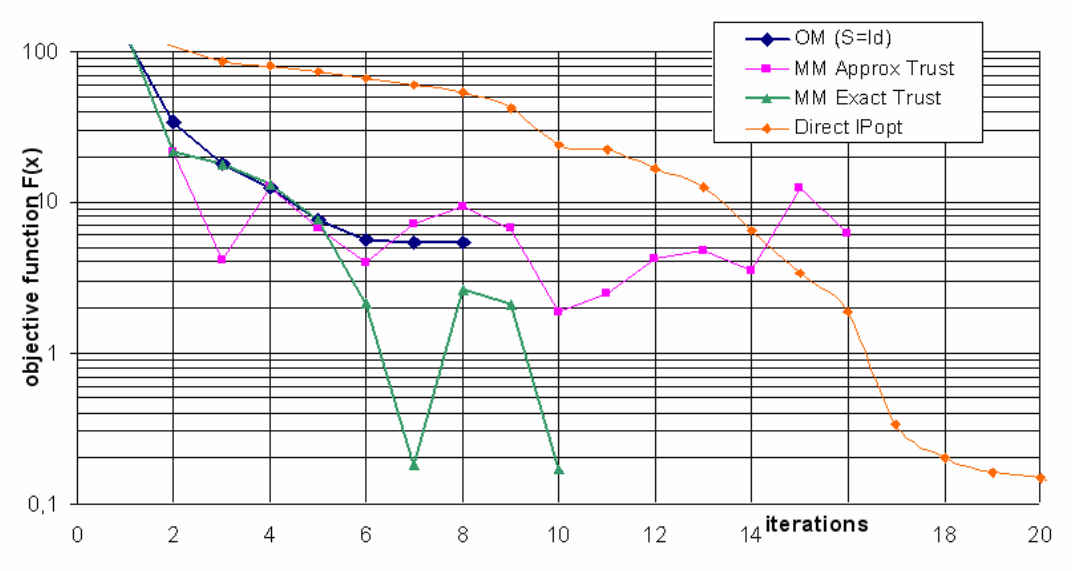


Fig. 10. Optimization algorithms convergence on the electromagnetic device optimization.

A direct optimization has been performed using fine model and its gradients in order to get a reference solution. The initial solution is provided by the coarse model optimal solution. The algorithm used is an Interior Point algorithm. A good solution has been found after 20 iterations, which means 20 fine model computations and 20 Jacobian computations which represent a high cost.

Simple output mapping convergence is good and the algorithm is stopped after 8 iterations. But it might fail to converge to the fine model optimal solution. Indeed, any interaction between variables in this mapping is missing.

Regarding Manifold Mapping, both true gradients and approximated ones need a trust region adaptation. Indeed the problem is very sensible and original algorithms failed. Default trust region parameters given in [14] have been used.

MM using true gradients converges with the highest speed. The consequence is a bouncing effect because it overcomes the target and has to return slower. This can be improved by trust region parameters modification (default values are used).

MM using approximated gradients failed to converge even with trust region strategy. The examination of S_k during a convergence process shows that $S_k = J_c(x_k^*).J_f^+(x_k^*)$, created with Jacobian matrices, and S_k created from approximated gradients $S_k = \Delta C(x_k^*).\Delta F^+(x_k^*) + (I - U_{k,c}U_{k,c}^T)$ are different, but in the same level of value and generally with the same elements sign. Our approximation is based on the previous steps but if steps are in the same direction, the gradients are not really well identified. This approximation has to be improved in order to have a robust algorithm.

6. Conclusions

In this work, several variants of the manifold mapping technique have been compared. Results on a realistic test case show that the use of exact gradients allows to convergence to more accurate solutions than reached by gradient approximations. These accurate solutions are reached three times faster than an interior algorithm iterating solely on the fine model. The manifold mapping algorithm with exact gradients therefore opens interesting perspectives on solving more complex optimization problems in the future.

References

[1] J.W. Bandler et.al., Space mapping: the state of the art, IEEE Trans. On Microwave Theory and Techniques, Vol. 52, No 1, pp. 337-361. 2004.

[2] D. Echeverria and P.W. Hemker, Space Mapping and Defect Correction, Comp. Methods in Appl. Math., Vol. 5, No. 2, pp. 107-136. 2005.

[3] K. Jittorntrum, An Implicit Function Theorem, Journal of Optimization Theory and Applications, 25(4), 1978.

[4] O. Chadebec, J. L. Coulomb and F. Janet, A Review of Magnetostatic Moment Method, IEEE Transactions on Magnetics, Vol: 42, No: 4, pp.515-520, April 2006

[5] H.J. Rakotoarison et al. Formal Sensitivity Computation of Magnetic Moment method, IEEE Transactions on Magnetics, vol. 44, No 6, pp. 1014-1017, June 2008

[6] P. Enciu, F. Wurtz, L. Gerbaud, B. Delinchant Automatic differentiation for electromagnetic models used in optimization, COMPEL, Vol 28, No: 5, pp 1313 - 1326, 2009

[7] B. Delinchant et al An optimizer using the software component paradigm for the optimization of engineering systems, COMPEL Vol 26, No: 2, pp 368 - 379, 2007

[8] D. Lahaye, A. Canova, G. Gruosso, M. Repetto, Adaptive manifold-mapping using multiquadric interpolation applied to linear actuator design, COMPEL Vol. 26 No. 2, pp.225 – 235, 2007

[9] L. Encica et al., Space mapping optimization of a cylindrical voice coil actuator. IEEE Transactions on Industry Applications, Vol 42, No 6, 1437-1444. 2006

[10]D. Echeverría et al., Manifold-mapping optimization applied to linear actuator design, IEEE Transactions on Magnetics, vol. 42, no. 4, April 2006

[11]T.V. Tran, F. Moussouni, S. Brisset, P Brochet,. Adapted Output Space-Mapping Technique for a Bi-Objective Optimization, IEEE Transactions on Magnetics, vol 46, no. 8, pp. 2990 - 2993 Aug. 2010

[12] G. Crevecoeur et al. Space mapping optimization of the magnetic circuit of electrical machines including local material degradation, IEEE Trans. on Magnetics, vol. 43, no 6, pp. 2609-2611, June 2007

[13]P.W. Hemker and D. Echeverría. A Trust-Region Strategy for Manifold Mapping Optimization. JCP Journal of Computational Physics, Vol 224, No 1, pp. 464-475, 2007.

[14]D. Echeverría, Two new variants of the manifold-mapping technique, COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, Vol. 2, No: 2, pp.334 – 344. 2007