



LAWRENCE  
LIVERMORE  
NATIONAL  
LABORATORY

# Towards Multi-level Optimization: Space-Mapping and Manifold-Mapping

D. Echeverria, C. Tong

August 1, 2006

## **Disclaimer**

---

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This work was performed under the auspices of the U.S. Department of Energy by University of California, Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

# Towards Multi-level Optimization: Space-Mapping and Manifold-Mapping

D. Echeverría<sup>1</sup> and C. Tong<sup>2</sup>

<sup>1</sup> Centrum voor Wiskunde en Informatica (CWI), Amsterdam, The Netherlands

<sup>2</sup> Lawrence Livermore National Laboratory (LLNL), Livermore, United States

July 24, 2006

## Abstract

In this report we study space-mapping and manifold-mapping, two multi-level optimization techniques that aim at accelerating expensive optimization procedures with the aid of simple auxiliary models. Manifold-mapping improves in accuracy the solution given by space-mapping. In this report, the two mentioned techniques are basically described and then applied in the solving of two minimization problems. Several coarse models are tried, both from a two and a three level perspective. The results with these simple tests confirm the speed-up expected for the multi-level approach.

## 1 Introduction

Many optimization problems in practice are based on models that typically have an excessive computational cost. A typical example is a finite element discretization. Accurate models yield precise solutions but at the expense of a time-expensive optimization procedure. By using fast (simplified) models one can expect some gain in the computational cost. But of course, the solution found will be only an approximation of the desired one; the simpler the model, the greater the error. According to space-mapping terminology we will denote the two types of models described as fine and coarse, respectively.

The purpose of this research is the combined employment of these models for precise and efficient optimization algorithms. The main idea is as follows: the coarse model optimum approximates efficiently the fine one, the fine model corrects that estimate using the coarse model optimization routine as a workhorse in an iterative procedure. We see a clear analogy with multigrid [1, 2] and defect correction [3] algorithms. Only the precise forward problem is solved; inversion is carried out with respect to the coarse model. And as with multigrid, provided a hierarchy of models is available, we can proceed in a multi-level fashion.

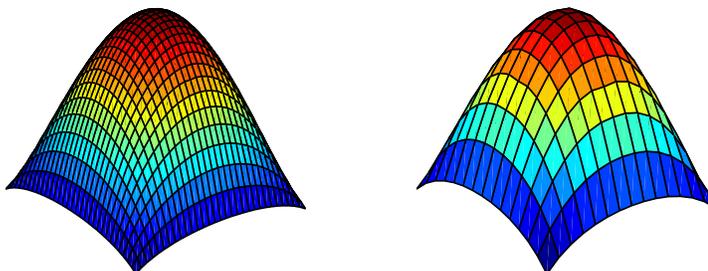


Figure 1: Post-processing of fine and coarse discretizations yield fine and coarse models. In this figure, Poisson equation solved with two different grids.

## 2 Multi-Level Optimization

Let us consider an optimization problem in the design space  $X \in \mathbb{R}^n$ . The fine model  $\mathbf{f} : X \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$  is also defined on that space and tries to meet the problem specifications  $\mathbf{y} \in Y \subset \mathbb{R}^m$ . The associated optimum is referred to as  $\mathbf{x}_f^*$  and we will assume it to be unique. Formally

$$\mathbf{x}_f^* = \operatorname{argmin}_{\mathbf{x} \in X} \|\mathbf{f}(\mathbf{x}) - \mathbf{y}\|. \quad (1)$$

If for example  $X$  is a non-empty compact set and  $\mathbf{f}$  is continuous, this problem has always at least one solution. The design is called *reachable* if the optimum meets the problem specifications, i.e.,  $\mathbf{f}(\mathbf{x}_f^*) = \mathbf{y}$ .

The coarse model  $\mathbf{c} : Z \subset \mathbb{R}^n \rightarrow \mathbb{R}^m$  is defined over a general set  $Z \subset \mathbb{R}^n$ . The type of coarse models used in this report take  $Z = X$ . The coarse optimum  $\mathbf{x}_c^*$  minimizes the proximity of the coarse model with respect to the problem specifications

$$\mathbf{x}_c^* = \operatorname{argmin}_{\mathbf{x} \in Z} \|\mathbf{c}(\mathbf{x}) - \mathbf{y}\|. \quad (2)$$

We will also assume uniqueness for  $\mathbf{x}_c^*$ . Examples of fine and coarse models could be any post-processing of the solution of a discretized partial differential equation. Finer discretizations yield in general more accurate models (Figure 1).

The two optimization procedures described below are based on alignments of the two models. Space-mapping operates in the design space  $X$  and manifold-mapping in the target space  $Y$ .

### 2.1 Space-Mapping

Space-mapping [4, 5] aligns the two models. The space-mapping function  $\mathbf{p} : X \rightarrow Z$  is introduced with that purpose

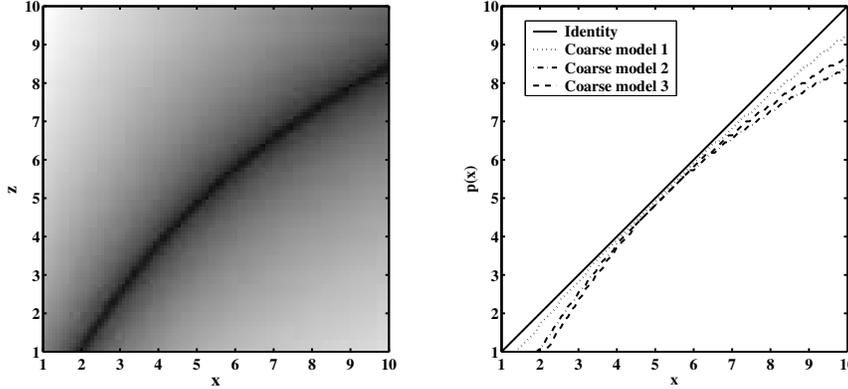
$$\mathbf{p}(\mathbf{x}) = \operatorname{argmin}_{\mathbf{z} \in Z} \|\mathbf{c}(\mathbf{z}) - \mathbf{f}(\mathbf{x})\|. \quad (3)$$

The quantity  $\|\mathbf{c}(\mathbf{z}) - \mathbf{f}(\mathbf{x})\|$  can be understood as a measurement of the model misalignment (see Figure 2). According to the definition of  $\mathbf{p}$ , we can expect that  $\mathbf{c}(\mathbf{p}(\mathbf{x})) \approx \mathbf{f}(\mathbf{x})$  and thus that the space-mapping solution

$$\mathbf{x}_{sm}^* = \operatorname{argmin}_{\mathbf{x} \in X} \|\mathbf{c}(\mathbf{p}(\mathbf{x})) - \mathbf{y}\| \quad (4)$$

improves the approximation of the fine optimum given by  $\mathbf{x}_c^*$ . Space-mapping can be interpreted as a general preconditioner since when applied to a linear

Figure 2: Misalignment and space-mapping function.



The left figure shows the misalignment function  $\|\mathbf{c}(\mathbf{z}) - \mathbf{f}(\mathbf{x})\|$  for a fine and a coarse model for the problem in Section 3.2. Darker shading shows a smaller misalignment. The right figure shows the identity function and, for different coarse models, their corresponding space-mapping functions.

```


$$\mathbf{x}_0 = \mathbf{x}_c^* = \operatorname{argmin}_{\mathbf{x} \in Z} \|\mathbf{c}(\mathbf{x}) - \mathbf{y}\|$$


$$\mathbf{B}_0 = I$$


$$\mathbf{p}_0(\mathbf{x}) = \mathbf{p}(\mathbf{x}_0) + \mathbf{B}_0 (\mathbf{x} - \mathbf{x}_0)$$

for  $k = 0, 1, \dots$ , while ...
do
    
$$\mathbf{x}_{k+1} = \operatorname{argmin}_{\mathbf{x} \in X} \|\mathbf{c}(\mathbf{p}_{k+1}(\mathbf{x})) - \mathbf{y}\|$$

    
$$\mathbf{h}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$$

    break if ...
    
$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{(\mathbf{p}(\mathbf{x}_{k+1}) - \mathbf{p}(\mathbf{x}_k) - \mathbf{B}_k \mathbf{h}_k) \mathbf{h}_k^T}{\mathbf{h}_k^T \mathbf{h}_k}$$

    
$$\mathbf{p}_{k+1}(\mathbf{x}) = \mathbf{p}(\mathbf{x}_{k+1}) + \mathbf{B}_{k+1} (\mathbf{x} - \mathbf{x}_{k+1})$$

enddo

```

Figure 3: The space-mapping algorithm.

problem, we obtain linear preconditioning [6]. The actual space-mapping algorithm used in this report is shown in Figure 3.

We should stress that an evaluation of  $\mathbf{p}$  is at least as expensive as one of the fine model. Therefore, it is not clear that the computation of  $\mathbf{x}_{sm}^*$  offers some advantages with respect to  $\mathbf{x}_f^*$ . But in practice, the space-mapping function is considerably simpler than the fine model, and thus, easier to approximate. In many cases  $\mathbf{p}$  is a slight deviation of a linear function and this fact allows a fast computation of the space-mapping solution. Some space-mapping functions are shown in Figure 2.

The space-mapping solution  $\mathbf{x}_{sm}^*$  does not always coincide with the fine optimum  $\mathbf{x}_f^*$ . A space-mapping function is *perfect* iff  $\mathbf{p}(\mathbf{x}_f^*) = \mathbf{x}_c^*$ . Then, if  $\mathbf{p}$  is perfect and injective, we have  $\mathbf{x}_{sm}^* = \mathbf{x}_f^*$  [6]. Though injectivity is not really a strong requirement, only in very special problems we can expect the space-mapping function to be perfect (it is easy to check that this is the case for any reachable design). But very often, a mapping is perfect by approximation,

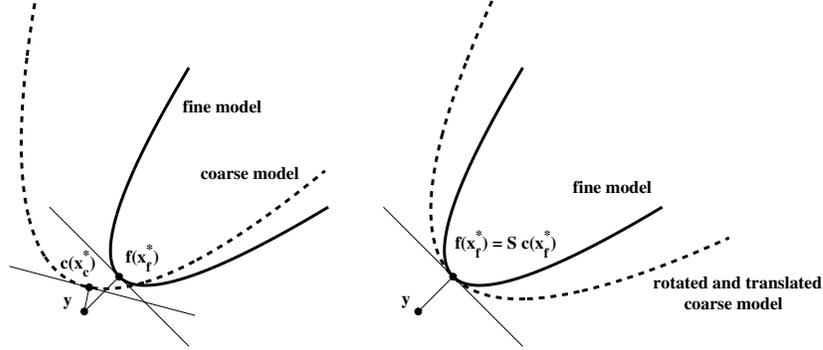


Figure 4: Manifold-mapping model alignment.

yielding space-mapping solutions that are acceptable in practice.

## 2.2 Manifold-Mapping

In [6] we analyze under which circumstances space-mapping yields the precise optimum. Essentially it depends on the behavior of the manifolds  $\mathbf{f}(X)$  and  $\mathbf{c}(Z)$ . In general these sets are different in the region close to the specifications  $\mathbf{y}$ , and this is the main reason why  $\mathbf{x}_{sm}^*$  does not always coincide with  $\mathbf{x}_f^*$ . The *manifold mapping*  $\mathbf{S} : \mathbf{c}(Z) \rightarrow \mathbf{f}(X)$  is introduced in [6] with the aim of correcting the misalignment between the manifolds  $\mathbf{f}(X)$  and  $\mathbf{c}(Z)$ . With  $\mathbf{S}$ , the point  $\mathbf{c}(\mathbf{x}_f^*)$  is mapped to  $\mathbf{f}(\mathbf{x}_f^*)$  and the tangent plane for  $\mathbf{c}(Z)$  at  $\mathbf{x}_f^*$  to the tangent plane for  $\mathbf{f}(X)$  at  $\mathbf{x}_f^*$  (see Figure 4). With these conditions, the optimum for  $\mathbf{S}(\mathbf{c}(\mathbf{x}))$  coincides with the fine optimum

$$\mathbf{x}_f^* = \operatorname{argmin}_{\mathbf{x} \in X} \|\mathbf{S}(\mathbf{c}(\mathbf{x})) - \mathbf{y}\|. \quad (5)$$

The mapping  $\mathbf{S}$  is assumed to be affine and can be determined in a non-unique way. It is also not known a priori, because it depends on the solution of the optimization problem. We propose an algorithm (see Figure 5) that, when it converges, yields the mapping  $\mathbf{S}$  and, as a consequence, the desired optimum  $\mathbf{x}_f^*$ . The  $\dagger$  denotes a pseudoinverse operator.

This algorithm can be slightly simplified. It can be shown that it is asymptotically equivalent to

$$\mathbf{x}_{k+1} = \operatorname{argmin}_{\mathbf{x} \in X} \|\mathbf{c}(\mathbf{x}) - \bar{\mathbf{y}}_k\| \quad (6)$$

where  $\bar{\mathbf{y}}_k = \mathbf{c}(\mathbf{x}_k) - [\Delta C \Delta F^\dagger + (I - U_c U_c^T)] (\mathbf{f}(\mathbf{x}_k) - \mathbf{y})$ . The model alignment can also be improved with an additional mapping  $\bar{\mathbf{p}} : X \rightarrow Z$ . In the algorithm in Figure 5,  $\bar{\mathbf{p}}$  is taken as the identity.

The desirable properties of the mapping  $\mathbf{S}$  are fulfilled at convergence. This is because the matrix  $S_k$  satisfies

$$S_k (\mathbf{c}(\mathbf{x}_k) - \mathbf{c}(\mathbf{x}_{k-l})) = \mathbf{f}(\mathbf{x}_k) - \mathbf{f}(\mathbf{x}_{k-l}) \quad (7)$$

for  $k > 0$  and  $l = 1, \dots, \min(n, k)$ . Then, since  $\mathbf{S}_k \mathbf{c}(\mathbf{x}_k) = \mathbf{f}(\mathbf{x}_k)$ , in case of convergence the limit is the fine optimum  $\mathbf{x}_f^*$ .

```

x0 = xc* = argminx ∈ Z ||c(x) - y||
S0 • = f(x0) + (• - c(x0))
for k = 0, 1, ..., while ...
do
    xk+1 = argminx ∈ X ||Sk(c(x)) - y||
    break if ...
    ΔF = [f(xk+1) - f(xk), ..., f(xk+1) - f(xmin(n,k))]
    ΔC = [c(xk+1) - c(xk), ..., c(xk+1) - c(xmin(n,k))]
    ΔF = Uf Σf VfT singular value decomposition
    ΔC = Uc Σc VcT singular value decomposition
    Sk+1 = ΔF ΔC† + (I - Uf UfT) (I - Uc UcT)
    Sk+1 • = f(xk+1) + Sk+1 (• - c(xk+1))
enddo

```

Figure 5: The manifold-mapping algorithm. The † denotes a pseudoinverse operator.

### 2.3 Multi-level Optimization

In some occasions, a whole hierarchy of models (ordered with respect to computational cost or accuracy) is available. This is the case of any optimization procedure that is based on solving a discretized partial differential equation by means of any type of grid. Inspired by multigrid we can apply the two-level ideas described up till now.

The fine optimum is computed through a sequence of coarse model optimizations. Each of these optimizations can be accelerated if some other efficient two-level approach is found. It can be seen that the convergence of the two-level approach is related to the similarity between the fine and the coarse model [6]. Thus, the lack of convergence between two distant models, can be alleviated by introducing an intermediate one, at the expense of (slightly) increasing the total computational cost. The real difficulty in the multi-level approach is the process of determining the appropriate sequence of coarse models.

## 3 Test problems

The methods described in the previous sections will be applied in the solving of two test problems. The first one represents a simple case of partial differential equation based optimization. We control the quality of the underlying models by selecting the size of the associated grid. The second one is more realistic and is related to photonic transmission and absorption in random media. The length of the propagation interval and the number of experiment realizations are used to tune the fidelity of the response obtained.

In the algorithms in Figure 3 and Figure 5 the user is free to choose the optimizers in the argmin function. In this report we use two different schemes: Nelder-Mead simplex [7] and Powell's Cobyly [8]. Both are derivative-free local minimizers and they often present a robust though slow behavior. With no other reason than diversity in the schemes applied, we will associate Powell's Cobyly with space-mapping and Nelder-Mead simplex with manifold-mapping. The stopping criterion chosen for each algorithm is the same. Every optimization

process stops when the increase of the Euclidean norm of the design parameters is smaller than  $10^{-4}$ .

### 3.1 A Poisson-based optimization

We first consider the Poisson equation in the unit square  $[0, 1]^2$  with homogeneous Dirichlet boundary conditions. There are two fixed point sources at  $(1/4, 1/2)$  and  $(1/2, 3/4)$  and their unknown magnitude is the two-dimensional design variable  $\mathbf{x} = (x_1, x_2)^T$ , corresponding  $x_1$  and  $x_2$  to the first and second source respectively. The model response is a four-dimensional vector with the solution of the discretized partial differential equation at  $(3/8, 5/8)$ ,  $(5/8, 5/8)$ ,  $(5/8, 3/8)$  and  $(3/8, 3/8)$ . The specifications are equal to  $\mathbf{y} = (1, 1, 1, 1)^T$ . The fine model  $\mathbf{f}$  is based on a  $257 \times 257$  rectangular grid and it is computed in around three seconds by means of the backslash operator in MatLab and in around seven when implemented in C and conjugate gradient is used for inverting the linear system. The coarse model  $\mathbf{c}$  considers only a  $7 \times 7$  rectangular grid and it is around 3000 times faster than the fine one in the first case and 1400 times in the second. Again, for sake of variety, the backslash operator is employed in conjunction with Powell's Cobyly and conjugate gradient with the Nelder-Mead simplex scheme. The sets  $X$  and  $Z$  coincide and they are equal to  $[0.01, 100]^2$ . We use the following cost function  $F(\mathbf{x})$  for the fine model

$$F(\mathbf{x}) = \frac{\|\mathbf{f}(\mathbf{x}) - \mathbf{y}\|_2}{\|\mathbf{y}\|_2} \times 100, \quad (8)$$

i.e., a relative Euclidean norm of the model discrepancy. A barrier term is added to take into account the compact design space  $X$ . The cost function for the coarse model is analog to that for the fine one. With the specifications above,  $F(\mathbf{x})$  has a unique minimum.

Table 1 shows the results of the optimization for the four schemes tried. Both Powell's Cobyly and Nelder-Mead simplex start from the coarse model optimum  $\mathbf{x}_c^* = (4.0992, 6.3160)^T$ . For this reason, the field corresponding to the number of coarse model evaluations for these two mentioned methods, is not in blank. Nelder-Mead simplex yields the fine optimum  $\mathbf{x}_f^* = (4.0761, 4.0761)^T$  and Powell's Cobyly solution practically coincides, with roughly the same computational effort. Space-mapping represents an important speed-up with respect to the one-level approach, but the solution differs from the precise optimum<sup>1 2</sup>. Manifold-mapping conserves the low computational cost of space-mapping and according to theory, gives the correct answer. Table 2 shows a comparison of both multi-level schemes, iteration by iteration until the stopping criterion is met.

A three-level approach was considered for manifold-mapping applied to this test problem. The fine model was again based on a  $257 \times 257$  grid and the two companion models were constructed upon a  $33 \times 33$  and a  $4 \times 4$  grid. These two coarse models are around 250 and 3000 times faster than the fine one respectively. Though even simpler models could be tried ( $3 \times 4$  for example),

<sup>1</sup>Indeed, perfect mapping is not held:  $\mathbf{p}(\mathbf{x}_f^*) = (3.9171, 6.7854)^T \neq \mathbf{x}_c^*$ .

<sup>2</sup>The results obtained with space-mapping combined with Nelder-Mead are essentially the same as those with Powell's Cobyly: identical final design and cost function in 6 and 818 fine and coarse model evaluations respectively.

Table 1: Results for the Poisson-based optimization.

	#f evals.	#c evals.	$F(\cdot)$	Final design
<b>Cobyla</b>	90	92	28.129	$(4.0762, 4.0760)^T$
<b>Nelder-Mead</b>	74	125	28.129	$(4.0761, 4.0761)^T$
<b>SM</b>	7	1175	28.194	$(4.3165, 3.7769)^T$
<b>MM</b>	5	424	28.129	$(4.0761, 4.0761)^T$

Table 2: Iteration history for the Poisson-based optimization.

# iter.	SM		MM	
	#(f, c) evals.	$(F(\cdot), h_k)$	#(f, c) evals.	$(F(\cdot), h_k)$
1	(1, 187)	(39.354, )	(1, 126)	(39.354, )
2	(2, 447)	(36.388, 4.1500)	(2, 220)	(36.388, 4.1500)
3	(3, 676)	(28.196, 1.6400)	(3, 303)	(28.340, 1.4500)
4	(4, 867)	(28.194, 0.0189)	(4, 369)	(28.129, 0.6070)
5	(5, 980)	(28.194, 0.0026)	(5, 424)	(28.129, 0.0000)
6	(6, 1105)	(28.194, 0.0002)		
7	(7, 1175)	(28.194, 0.0000)		

Powell's Cobyla and space-mapping (SM) are used in conjunction with conjugate gradient and Nelder-Mead simplex and manifold-mapping (MM) with MatLab's backslash operator. This yield coarse models around 1400 and 3000 times faster than the fine ones in the first and second case respectively.  $F(\cdot)$  represents the cost function. The stopping criterion is in every case  $h_k = \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 < 10^{-4}$ .

the last coarse model can be consider as the lower limit in the hierarchy. Table 3 shows the results concerning the three-level approach.

We denote the three-level scheme by **MM-(257/33/4)**. **MM-(257/33)** and **MM-(257/4)** refer to the two-level algorithms that compute the fine optimum aided by the  $33 \times 33$  and  $4 \times 4$  grid based models respectively. **MM-(257/33)** presents a monotonous convergence in just four iterations. With the three-level approach, the convergence history is almost identical and the total computational cost is slightly reduced (from 16.84 seconds to 14.95). **MM-(257/4)** also yields the fine optimum, but the cost function does not decrease monotonously (the total computing time is 16.31 seconds)<sup>3</sup>.

The three-level manifold-mapping approach does not represent a significant speed-up with respect to the two-level scheme. But we can think of the multi-level idea as a procedure as efficient as any based on two levels, but with the additional advantage of a higher robustness. Multi-level manifold mapping is subject of further research. Whenever a hierarchy of models is available, it would be useful to know how to select the best intermediate ones.

We will conclude this section with a simple experiment, in which the potential of coarsening techniques similar to those employed in algebraic multigrid [9, 2] is studied. Again we are interested in the Poisson equation in the unit square (homogeneous Dirichlet boundary conditions), but this time with a vary-

<sup>3</sup>Two-level space-mapping is not performing as its manifold-mapping counterpart, when the fine model is aided with that extremely coarse model. The space-mapping solution obtained  $\mathbf{x}_{sm}^* = (0.0100, 7.3243)^T$  lies very far away from the precise minimum.

Table 3: Three-level approach for the manifold-mapping Poisson-based optimization.

MM-(257/33)				
# iters.	#f evals.	#c evals.	$F(\cdot)$	$\ \mathbf{x}_{k+1} - \mathbf{x}_k\ _2$
1	1	(120, 0)	28.130	
2	2	(176, 0)	28.129	0.0135
3	3	(230, 0)	28.129	0.0012
4	4	(283, 0)	28.129	0.0000

MM-(257/4)				
# iters.	#f evals.	#c evals.	$F(\cdot)$	$\ \mathbf{x}_{k+1} - \mathbf{x}_k\ _2$
1	1	(0, 105)	34.881	
2	2	(0, 175)	29.165	1.4900
3	3	(0, 237)	30.454	0.9010
4	4	(0, 313)	28.129	2.1100
5	5	(0, 369)	28.129	0.0000

MM-(257/33/4)				
# iters.	#f evals.	#c evals.	$F(\cdot)$	$\ \mathbf{x}_{k+1} - \mathbf{x}_k\ _2$
1	1	(5, 365)	28.130	
2	2	(10, 693)	28.129	0.0134
3	3	(15, 1024)	28.129	0.0012
4	4	(20, 1352)	28.129	0.0000

MM-(257/33/4) denotes the three level approach based on the  $257 \times 257$ ,  $33 \times 33$  and  $4 \times 4$  grid based models. MM-(257/33) and MM-(257/4) refer to the two-level algorithms that consider the two previous coarse models. These two coarse models are around 250 and 3000 times faster than the fine one respectively. Every linear system is solved by means of MatLab's backslash operator. Nelder-Mead simplex was used for the coarsest minimization in all cases.

ing diffusion coefficient that takes the value 1 inside a circle centered in (0.5, 0.5) with radius  $1/3$  and  $10^6$  in the remaining part of the domain. The rest of the description is analog to the problem studied above. The fine model  $\mathbf{f}$  is based on a  $257 \times 257$  rectangular grid. The first coarse model is obtained by taking a rectangular grid with larger step size (geometric coarsening). The second coarse model is derived with the aid of algebraic multigrid. The coarsening used by algebraic multigrid aims at the essential structure of the underlying linear system and this enables to capture features that sometimes escape to the traditional geometric approach. All the operators needed (restriction, prolongation and coarse linear system) are obtained with the classical Ruge-Stuben coarsening [2].

The performance of manifold-mapping using both coarse models is compared in Table 4 for a reachable design problem with specifications  $\mathbf{y} = \mathbf{f}((12.345, 54.321)^T) = (3.9622, 3.4256, 1.0644, 1.6010)^T$ . The two models

Table 4: Results for manifold-mapping and geometric and algebraic multigrid based coarse models.

grid size MG / levels AMG	#f evals.	#c evals.	speed-up
182 / 2	5 / 2	361 / 158	0.8 / 7.0
92 / 3	5 / 4	362 / 300	2.4 / 4.0
47 / 4	5 / 5	377 / 658	6.3 / 2.8
25 / 5	5 / 5	367 / 673	10.4 / 2.6
14 / 6	5 / 5	383 / 611	17.2 / 3.8

Every row expresses the performance of manifold-mapping using two different coarse models. The number of degrees of freedom of the grid used for solving the partial differential equation is about the same. The number of fine and coarse model evaluations needed to solve the problem is also shown and additionally the speed-up factor when these two methods are compared with the Nelder-Mead simplex scheme. The first number refers to the geometric coarsening and the second one to the algebraic multigrid approach. The stopping criterion is in every case  $h_k = \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 < 10^{-4}$ .

rely on discretizations with a similar number of degrees of freedom<sup>4</sup>. In every row, the results for two coarse models of different type are shown. The first number in every field refers to the geometric coarsening and the second one to the algebraic multigrid approach. The speed-up refers to the performance of Nelder-Mead simplex starting from the optimum of the geometric coarse model. The algebraic multigrid idea seems useful when the coarse model approximates faithfully the fine one. For really simple coarse models, the standard approach is more efficient. Better results for the algebraic based technique are expected in more complicated problems, where algebraic multigrid truly represents an improvement over the traditional geometric scheme.

### 3.2 A photon transport based problem

In this problem we study the performance of the two-level methods explained above with a simplified problem of practical interest. The phenomenon under analysis is the photon transmission over random media. The one-dimensional steady state transport equation is solved over a domain of (dimensionless) length  $L$ . Due to the stochastic nature of the media involved, the solution is computed  $N$  times, and statistics are taken afterwards. We consider two different optimization problems. In the first one, the model output is scalar, a global transmission coefficient. The second problem in addition computes its variance. For both problems, the design variable  $\mathbf{x}$  describes some behavior of the random media considered. In the second case  $\mathbf{x}$  is a vector with dimension two and in the first one is the scalar obtained by keeping its second component  $x_2$  equal to 5. The sets  $X$  and  $Z$  coincide and they are equal to  $[1, 10]$  and to  $[1, 10]^2$  in the one and two dimensional case respectively.

The model is assumed to be fine enough for practical purposes when  $L = 1$  and  $N = 100000$ . Different coarse models are obtained by taking smaller values for these two parameters. This is shown in Figure 6 and Figure 7 where the

<sup>4</sup>The number of grid nodes in every case is approximately the same.

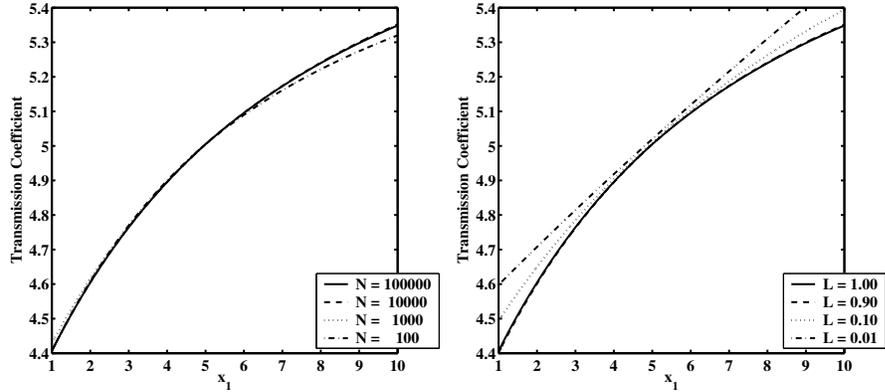


Figure 6: Different models for the transmission coefficient. The second component of the design variable  $x_2$  is kept constant and equal to 5. The length  $L$  is taken as 1 in the left figure. The number of realizations  $N$  is 100000 for the right plot.

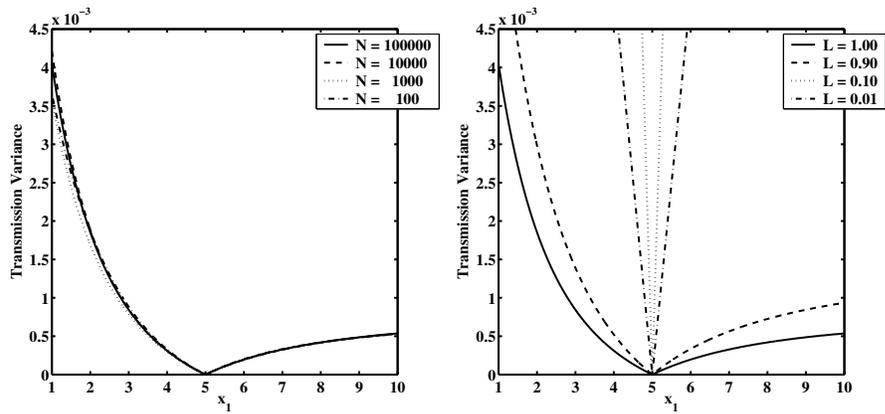


Figure 7: Different models for the transmission variance. The second component of the design variable  $x_2$  is kept constant and equal to 5. The length  $L$  is taken as 1 in the left figure. The number of realizations  $N$  is 100000 for the right plot.

transmission coefficient and its variance are computed for different values of  $L$  and  $N$ . The second component of the design variable  $x_2$  is kept constant and equal to 5. We observe that for the transmission coefficient  $L$  and  $N$  can be reduced to a very low value without (apparently) losing much accuracy in the model. The transmission variance is very sensitive to the varying of the length  $L$ , but concerning the number of realizations  $N$  the behavior is analog to the one observed in Figure 6. Thus, the same pair  $(L, N)$  can be acceptable for computing the transmission coefficient but not for its variance.

We will take coarse models given by the pairs  $(L, N) = (0.01, 100000)$  and  $(L, N) = (1, 100)$  for the first and second design problem respectively. The first coarse model is around 30 times faster than the fine model and the second one around 40 times. The difference in computational cost of the selected

Table 5: Results for the one-dimensional photon transport based optimization.

	#f evals.	#c evals.	$F(\cdot)$	Final design
<b>Cobyla</b>	27	40	0.000	9.1234
<b>Nelder-Mead</b>	30	34	0.000	9.1234
<b>SM</b>	6	300	0.000	9.1234
<b>MM</b>	5	155	0.000	9.1234

Table 6: An iteration history for the one-dimensional photon transport based optimization.

# iter.	SM		MM	
	#(f, c) evals.	$(F(\cdot), h_k)$	#(f, c) evals.	$(F(\cdot), h_k)$
1	(1, 26)	(4.016, )	(1, 43)	(4.016, )
2	(2, 60)	(3.740, 0.0587)	(2, 83)	(2.713, 1.7000)
3	(3, 76)	(3.740, 0.0000)	(3, 116)	(0.254, 0.6970)
4			(4, 143)	(0.018, 0.0712)
5			(5, 167)	(0.000, 0.0049)
6			(6, 191)	(0.000, 0.0000)

The coarse models employed are around 30 times faster than the fine one.  $F(\cdot)$  represents the cost function. The stopping criterion is in every case  $h_k = \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 < 10^{-4}$ .

coarse models when compared with the those given by  $(L, N) = (0.01, 100)$  or  $(L, N) = (0.9, 100)$  is almost inappreciable. The cost functions considered are as in (8).

In the one-dimensional case we aim two different reachable designs. In the first problem we take  $\mathbf{y} = \mathbf{f}(9.1234) = 5.3044$  and the results obtained are shown in Table 5. Both two-level approaches yield the fine optimum  $\mathbf{x}_f^*$  but the speed-up is only of around two for space-mapping and three for manifold-mapping. For this problem, no faster coarse models are available. In some cases,  $N$  is recommended to be at least 1000000. There we can depreciate the number of coarse model evaluations when compared to those for the fine model and then obtain higher speed-up values.

The space-mapping solution  $\mathbf{x}_{sm}^*$  coincides with  $\mathbf{x}_f^*$  because the design is reachable, i.e.,  $\mathbf{f}(\mathbf{x}_f^*) = \mathbf{y}$ . The space-mapping function  $\mathbf{p}$  for the models used is shown in Figure 2, labeled **Coarse model 2** in the legend. It should be noticed that  $\mathbf{p}(\mathbf{x}) = 1$  for  $\mathbf{x} \lesssim 2$ . We can easily see that  $\mathbf{p}$  is injective in the region of interest (see Figure 2 and Figure 6).

The specifications for the second design problem are  $\mathbf{y} = \mathbf{f}(2.0000) = 4.6044$ . In this case the coarse model optimum is very close to one and because, in this region  $\mathbf{p}$  is not injective, we can expect space-mapping to fail. We check that in Table 6 together with the expected correct behavior of manifold-mapping. Space-mapping converges in only three iterations but to the wrong point 1.1173. The other methods find the right answer. Nelder-Mead simplex needs 35 fine model evaluations to reach the fine model optimum. The speed-up obtained with manifold-mapping with respect to this one-level optimization procedure is around two.

Table 7: Results for a reachable two-dimensional photon optimization.

	#f evals.	#c evals.	$F(\cdot)/F_s(\cdot)$	Final design
<b>Cobyla</b>	37	45	0.000 / 0.092	$(3.2428, 7.7157)^T$
<b>Nelder-Mead</b>	72	89	0.000 / 0.000	$(3.4549, 7.6548)^T$
<b>SM<sub>Cob</sub></b>	3	200	0.000 / 0.095	$(3.2405, 7.7164)^T$
<b>SM<sub>NMS</sub></b>	13	1711	0.000 / 0.000	$(3.4556, 7.6546)^T$
<b>MM</b>	6	436	0.000 / 0.000	$(3.4507, 7.6560)^T$

Table 8: Results for a non-reachable two-dimensional photon optimization.

	#f evals.	#c evals.	$F(\cdot)/F_s(\cdot)$	Final design
<b>Nelder-Mead</b>	64	90	0.020 /	$(1.0002, 5.7685)^T$
<b>SM</b>	4	468	0.020 /	$(1.0015, 5.7680)^T$
<b>MM</b>	5	344	0.020 /	$(1.0034, 5.7674)^T$

The coarse models employed are around 40 times faster than the fine ones.  $F(\cdot)$  represents the cost function.  $F_s(\cdot)$  represents the scaled cost function. The stopping criterion is in every case  $h_k = \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2 < 10^{-4}$ .

In the two-dimensional problem we first try a reachable design. As mentioned before, the coarse model selected is the one given by  $(L, N) = (1, 100)$  which is around 40 times faster than the fine model. The specifications are  $\mathbf{y} = \mathbf{f}((3.4567, 7.6543)^T) = (7.0348, 0.0002)^T$ . We observe from Figure 6 and Figure 7 (and also from  $\mathbf{y}$ ) that the transmission variance takes values much smaller than the transmission coefficient. This difference in size suggests that a scaling could be necessary since the cost function considered gives more emphasis to the first component of every model than to the second one. Consequently, values far away from the fine model optimum could yield, for practical purposes, the same cost function value. The model response is scaled as follows. An average value of the transmission variance is first computed based on the coarse model. Then, the output of both fine and coarse models is weighted according to that value. From another perspective, a scaled cost function  $F_s(\cdot)$  is used. The results of this reachable two-dimensional optimization problem are shown in Table 7.

First, it should be noted that Powell’s Cobyla method, with the obvious tuning parameters, does not yield the fine model optimum<sup>5</sup>. The scaling could be inadequate for this algorithm. We observe that space-mapping, combined with Powell’s Cobyla (denoted in the table by **SM<sub>Cob</sub>**) also fails, and we associate this wrong behavior to the use of that one-level optimizer. This assumption is corroborated by the fact that the Nelder-Mead simplex scheme leads space-mapping to the fine model optimum (again perfect mapping holds). But this alternative space-mapping (**SM<sub>NMS</sub>**) takes almost the same time as the one-level Nelder-Mead simplex. Manifold-mapping represents a speed-up of around four and this result is consistent with the one-dimensional case. If no scaling is introduced, the problem is ill-conditioned and the solutions obtained differ clearly from the fine optimum  $((3.2784, 7.7052)^T$  for Nelder-Mead simplex and

<sup>5</sup>The implementation chosen for Cobyla is the package by J.S. Roy. The source code can be found at <http://plato.la.asu.edu/topics/problems/nlores.html>.

$(3.3043, 7.6975)^T$  for manifold-mapping).

The manifold-mapping solution is slightly different to the precise optimum. Even with the additional scaling, the matrices  $\Delta F$  (see the manifold-mapping algorithm, Figure 5) have very small singular values. Since we are interested in computing the pseudoinverse of this matrix  $\Delta F^\dagger$ , at a given point, the next iterant can be found very far away from their predecessors. This behavior clearly spoils convergence and can be prevented by the use of a trust region methodology. The one implemented for this report is very simple: the inverse of those singular values smaller than a given tolerance is taken as zero. Large steps are then discouraged at the expense of precision in finding the optimum. More involved trust region procedures may yield better results.

If manifold-mapping is applied with the same coarse model employed in the one-dimensional problem, the solution  $(3.4516, 7.6557)^T$  is obtained after 24 and 1855 fine and coarse model evaluations respectively. The associated computational cost is even higher than that for one-level Nelder-Mead simplex. If we use the model with  $(L, N) = (1, 100)$  just for computing the coarse model optimum and then we continue with  $(L, N) = (0.01, 100000)$ , the resulting procedure accelerates Nelder-Mead simplex by a factor of three ( $(3.4526, 7.6554)^T$  is obtained after 9 and 538 fine and coarse model evaluations). The conclusion is that intermediate coarse models can be used for computing a better initial guess that improves a previous observed speed-up.

We also try a non-reachable design, given by the specifications  $\mathbf{y} = (5, 0.004)^T$ . Again the selected implementation of Powell's Cobyta does not yield results. The second component in the specifications is about 20 times larger than in the reachable case and this seems to be enough for not needing the additional weighting. The results without scaling are shown in Table 8. The speed-up factors for space-mapping and manifold-mapping are five and four respectively. With the use of scaling, the convergence behavior is about the same and the limit point is  $(1.0000, 5.6710)^T$  for space-mapping and  $(1.0007, 5.6967)^T$  for manifold-mapping. The coarse model given by  $(L, N) = (0.01, 100000)$  yields also the correct answer with no scaling and using manifold-mapping but after 13 and 1117 evaluations of the fine and coarse model respectively. If scaling is applied, the solution differs significantly from  $\mathbf{x}_f^*$ .

The difference in computational cost between the fine and coarse models considered is not enough to fully exploit the advantages of the multi-level approach. Moreover, a really simple model as  $(L, N) = (1, 100)$  presents no important convergence problems in the two-level scheme. An intermediate model would presumably be of not much help. If higher accuracy is desired in solving the transport equations, the distance between the fine and coarse models will increase, and then, a multi-level scheme might offer some noticeable improvement over the two-level strategy.

## 4 Conclusions

Some optimization procedures can be accelerated by a sequence of inexpensive approximations of the desired optima. We have introduced space-mapping and manifold-mapping, two schemes that enhance the efficiency, with the use of information from models with different levels of accuracy. Most of the time space-mapping yields an acceptable approximation of the precise solution. How-

ever, if we want to find the accurate optimum manifold-mapping should be used. Two experiments have illustrated theory, in the two and three level approach, and the effect of the choice from a great number of several coarse models has been studied.

## 5 Acknowledgement

D. Echeverría thanks P.W. Hemker for his useful suggestions for the improvement of this manuscript.

## References

- [1] W. Hackbush, *Multigrid methods and Applications*, Series in Computational Mathematics, vol. 4, Springer-Verlag, Berlin, 1985.
- [2] U. Trottenberg, C. W. Oosterlee and A. Schüller, *Multigrid*, Academic Press, London, 2001.
- [3] K. Böhmer and P. W. Hemker and H. J. Stetter, *Defect Correction Methods: Theory and Applications*, The defect correction approach, Computing Suppl. 5, pp. 1–32, K. Böhmer and H. J. Stetter ed., Springer-Verlag, Berlin, Heidelberg, New York, Tokyo, 1984.
- [4] J. W. Bandler, R. M. Biernacki, C. H. Chen, P. A. Grobelny and R. H. Hemmers, *Space Mapping Technique for Electromagnetic Optimization*, pp. 2536–2544, IEEE Trans. on Microwave Theory and Techniques, 42 (12), 1994.
- [5] J. W. Bandler, Q. S. Cheng, S. A. Dakroury, A. S. Mohamed, M. H. Bakr, K. Madsen and J. Søndergaard, *Space Mapping: The State of the Art*, pp. 337–361, IEEE Trans. on Microwave Theory and Techniques, 52 (1), 2004.
- [6] D. Echeverría, and P.W. Hemker, *Space mapping and defect correction*, Comp. Methods in Appl. Math., vol. 5(2), pp. 107–136, 2005.
- [7] J.C. Lagarias, J.A. Reeds, M.H. Wright and P.E. Wright, *Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions*, SIAM Journal of Optimization, vol. 9(1), pp. 112–147, 1998.
- [8] J.M.D. Powell, *A Direct Search Optimization Method that Models the Objective and Constraint Functions by Linear Interpolation*, DAMTP/NA5, Cambridge, England, 1992.
- [9] A. Brandt, S. F. McCormick, and J. W. Ruge, *Algebraic multigrid for sparse matrix equations*, Sparsity and Its Applications, D. J. Evans, ed., Cambridge University Press, Cambridge, 1984.

This work was performed under the auspices of the U. S. Department of Energy by University of California, Lawrence Livermore National Laboratory under contract W-7405-Eng-48.