



# Automatic variogram modeling by iterative least squares. Univariate and multivariate cases.

Nicolas Desassis, Didier Renard

## ► To cite this version:

Nicolas Desassis, Didier Renard. Automatic variogram modeling by iterative least squares. Univariate and multivariate cases.. 2012. hal-00707887v1

**HAL Id: hal-00707887**

**<https://minesparis-psl.hal.science/hal-00707887v1>**

Preprint submitted on 13 Jun 2012 (v1), last revised 25 Sep 2012 (v2)

**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.

# Automatic variogram modeling by iterative least squares.

Univariate and multivariate cases.

N. Desassis and D. Renard

## **Abstract :**

In this paper, we propose a new methodology to automatically find a model that fits on an experimental variogram. Starting with a linear combination of some basic authorized structures (e.g spherical, exponential,...), a numerical algorithm is used to compute the parameters which minimize a distance between the model and the experimental variogram. The initial values are automatically chosen and the algorithm is iterative. After this first step, parameters with a negligible influence are discarded from the model and the more parsimonious model is estimated by using the numerical algorithm again. This process is iterated until no more parameter can be discarded.

A procedure based on a profiled cost function is also developed in order to use the numerical algorithm for multivariate data sets (possibly with a

lot of variables) modeled in the scope of a linear model of coregionalization. The efficiency of the method is illustrated on several examples (including variogram maps) and on a multivariate case study.

**Key-words :** Automatic variogram fitting, variogram maps, linear model of coregionalization, anisotropy, weighted least squares, over-fitting.

## 1 Introduction

A key stage for most of geostatistical studies is the modeling of the structural model which characterizes the spatial behavior of the variables of interest (Chilès and Delfiner, 2012). For this purpose, some methods are based on the likelihood in a distributional framework. For instance Mardia and Marshall (1984) jointly estimate the drift terms and the covariance parameters by numerical maximization of the likelihood under a multigaussian assumption. Even if they show that such an estimator is asymptotically unbiased and consistent, the bias can be important for a finite sample size. For this reason, other authors propose to work with the restricted maximum likelihood estimator (Cressie and Lahiri, 1996). Also based on the Gaussian likelihood, some works are made in the Bayesian paradigm (see for instance Handcock and Wallis, 1994).

Nevertheless, all these iterative methods need several evaluations of the likelihood and each one requires to solve a linear system with a dimension equal to the size of the data set. This can be prohibitive when working with

several thousands of samples (see e.g, Stein, 1999) as it is common in a lot of domains, for instance in the mining industry. Furthermore, the distributional assumptions that have to be made (as the multigaussian one) cannot be easily checked on the data.

For these reasons, most geostatistical studies are based on weaker assumptions such as the intrinsic stationarity (i.e the variance of the increments between two points is only a function of the lag between these two points). In such a framework, the experimental variogram, which measures the spatial continuity, is computed. Then a valid model for the theoretical variogram must be fitted on the experimental variogram. Indeed, this model cannot be any function : it must be a conditionally definite negative function in order to ensure that the variance of any linear combination of the data always remains positive, as long as their weights add up to 0. For that sake, it is recommended to define the model as a linear combination of a small set of authorized functions, called basic structures (nugget effect, exponential, spherical, linear structures to name only few of them) with positive coefficients. Each basic structure depends on a limited set of parameters, such as the sill, the scale parameter, the anisotropy ratios and rotation angles. Other basic structures may need some shape parameters, for instance the exponent in the power model.

Once the set of basic structures is defined, and if all their parameters (except the sills) are given, the optimal determination of the sills can be obtained by using standard minimization procedures. The principle is to find

the model that minimizes a cost function measuring the distance between the model and the experimental variogram. In the univariate case it is sufficient that the sills fulfill a positivity constraint and they can be fitted by ordinary least squares (OLS), weighted least squares (WLS) or generalized least squares (GLS) (Cressie, 1985, 1993). In the multivariate case and in the scope of the linear model of coregionalization (LMC), it is sufficient that the fitted matrices of sills for each basic structure are positive definite (Wackernagel, 1993). For this purpose, the algorithm proposed by Goulard and Voltz (1992) allows us to find the optimal matrix of sills for each basic structure, under the positive definiteness constraint. These techniques are sufficient when the choice of the necessary basic structures is obvious and when the scale and shape parameters can be visually guessed. However, things become more complicated when the experimental variograms are calculated in different directions and show an anisotropy for example. The complexity may even increase when a different anisotropy is attached to each structure, or in the 3-D case. Finally, in the moving geostatistics approach (Magneron *et al.*, 2009), the structural model varies across the domain. Therefore this fitting step must be repeated several times. For all these reasons, automatic procedures for the entire set of variogram parameters are needed.

Some statistical software already propose automatic variogram fitting procedures. These routines are usually based on non-linear optimizations tools. For instance, the function `variofit` in the R package `geoR` (Diggle and Ribeiro, 2007) calls the R function `optim`. But they have not been fully adapted

to the specific case of variogram modeling and may fail to converge, mostly in the case where nested models are used.

Other methods are based on stochastic optimization. For instance the `varfit` program (Pardo-Igúzquiza, 1999, Larrondo *et al.*, 2003) tries to minimize the cost function by randomly perturbing the value of one parameter at each iteration. The new parameter value is accepted if it reduces the cost function. More recently, Emery (2010) uses simulated annealing to fit a linear model of coregionalization.

In this paper, a new algorithm is presented. It has been developed and heavily tested in many different configurations. It is based on a numerical minimization (deterministic) of a sum of squares and uses specific techniques to cope with numerical singularities and convergence problems often encountered with variogram fitting (mostly when several basic structures are used). Each parameter is restricted to a definition domain consistent with the geo-statistical framework (e.g the range must be a positive quantity) but this domain can also be defined by the user. The user can input his own set of basic structures and the algorithm tries to obtain a good fit within this set while avoiding over-fitting.

This paper first gives the mathematical basis of the numerical algorithm and details its different steps. Then an extension of the method is proposed in order to fit all the parameters of a LMC. Finally the method efficiency is illustrated through several examples of automatic fits performed on one or more variables, isotropic or not, starting from experimental variograms or

variogram maps.

## 2 Problem formulation

We start with a collection of experimental structure quantities  $\hat{\gamma}_j = \hat{\gamma}(\vec{h}_j)$  (e.g an experimental variogram) for the lag vectors  $\vec{h}_j \in \mathbb{R}^d$ ,  $j = 1, \dots, n$ . We also consider a collection of weights  $\omega_1, \dots, \omega_n$  (for instance  $\omega_j = N_j / \|\vec{h}_j\|$  where  $N_j$  is the number of pairs which have been used to compute  $\hat{\gamma}(\vec{h}_j)$ ).

We consider  $\mathcal{G} = \{g^{(1)}, \dots, g^{(p)}\}$  a family of normalized basic structures with a sill parameter equal to one. Note that each  $g^{(i)}$  is parameterized by a vector  $\theta_i$  such as  $g^{(i)} \equiv g_{\theta_i}^{(i)}$ . For instance  $g_{\theta_1}^{(1)}$  could be the exponential variogram model in  $\mathbb{R}^3$  and  $\theta_1$  is the vector containing the parameters of this model (range, anisotropy angles and ratios).  $\mathcal{G}$  can contain the same basic structure several times since a “good” model could be a linear combination of two exponential structures, one with a short range and the other one with a longer range.

The aim is to find a linear combination  $\gamma_\Psi = \sum_{i=1}^p \lambda_i g_{\theta_i}^{(i)}$  with positive coefficients such as the cost function

$$S(\Psi) = \frac{1}{2} \sum_{j=1}^n \omega_j \left( \gamma_\Psi(\vec{h}_j) - \hat{\gamma}(\vec{h}_j) \right)^2 \quad (1)$$

is minimum with respect to the full vector of parameters  $\Psi = (\theta_1, \dots, \theta_p, \lambda_1, \dots, \lambda_p)$ .

In other words, the target  $\Psi^*$  is the value that minimizes the cost func-

tion :

$$\Psi^* = \underset{\Psi \in \mathcal{D}}{\operatorname{argmin}} S(\Psi),$$

where  $\mathcal{D} \subset \mathbb{R}^N$ , the definition domain of  $\Psi$ , is an hyper-rectangle, and  $N$  is the total number of parameters.

In the following, we will denote  $\Lambda = (\lambda_1, \dots, \lambda_p)$  (for the set of sills),  $\Theta = (\theta_1, \dots, \theta_p)$  (for the other parameters) and  $\mathcal{D}_\Lambda$  and  $\mathcal{D}_\Theta$  their respective definition domains.

### 3 Algorithm

In order to work with a parsimonious model, the following iterative procedure is used :

- *Initialisation* : at  $k = 0$ , an initial model family  $\mathcal{G}_0$  is given.
- *Iterations* : at step  $k$ ,
  - (i) start from an initial model in  $\mathcal{G}_k$ , say  $\gamma_{\Psi_0}^{(k)}$ ,
  - (ii) find  $\gamma_{\Psi^*}^{(k)}$  the best model in  $\mathcal{G}_k$  (i.e find the “best” vector of parameters  $\Psi^*$  using an iterative and numerical algorithm),
  - (iii) if  $\mathcal{G}_k$  can be reduced, then take  $\mathcal{G}_{k+1} \subset \mathcal{G}_k$  and return to (i). Otherwise set  $\gamma_{\Psi^*} = \gamma_{\Psi^*}^{(k)}$  and exit.

The algorithm of step (ii) (presented in section 3.1) is only able to find a local minimizer for  $\Psi$ . For this reason, care has to be taken for the choice of the initial vector of parameters  $\Psi_0^{(k)}$  in step (i) (as described in section



3.2.2). The way to reduce the model family in step (iii) is detailed in section 3.2.3.

## 3.1 Numerical algorithm

In this part, we describe the point (ii) of the main algorithm. The numerical algorithm we propose is a variant of the Gauss-Newton algorithm. We call it *foxleg* by reference to the close Powell's Dog Leg algorithm (see e.g Madsen *et al.*, 2004a) and the translation of the French name of the second author.

### 3.1.1 Principle of the Gauss-Newton algorithm

We start from an initial value, say  $\Psi_0$ . If  $\Psi^*$  denotes a value of  $\mathcal{D}$  that corresponds to a local minimizer of the cost function  $S(\Psi)$  given by equation (1), then the aim is to produce a sequence  $(\Psi^{(t)})_{t \in \mathbb{N}^*}$  of elements of  $\mathcal{D}$  such that  $\Psi^{(t)}$  converges towards  $\Psi^*$  when  $t$  tends to infinity. For  $j = 1, \dots, n$ , let us denote :

$$r_j(\Psi) = \gamma_\Psi(\vec{h}_j) - \hat{\gamma}(\vec{h}_j)$$

the residual associated to the  $j^{\text{th}}$  lag. With this notation, the cost function can be written :

$$S(\Psi) = \frac{1}{2} \sum_{j=1}^n \omega_j r_j^2(\Psi). \quad (2)$$

First, note that for a vector  $\varepsilon = (\varepsilon_1, \dots, \varepsilon_N)$  such as  $\|\varepsilon\|$  is small, the

Taylor expansion of each  $r_j$  around  $\Psi$  gives :

$$r_j(\Psi + \varepsilon) \simeq r_j(\Psi) + \varepsilon^T J_j(\Psi)$$

where  $^T$  stands for the transposition and  $J_j(\Psi)$  is the gradient vector of  $r_j$  with respect to each component of  $\Psi$ . Then the  $k^{\text{th}}$  component of  $J_j(\Psi)$  is :

$$J_j^k = \frac{\partial r_j}{\partial \Psi_k}(\Psi),$$

for  $k = 1, \dots, N$ .

By replacing  $r_j$  by its approximation in equation (2), we obtain an approximation function  $L_\Psi$  of  $S$  around  $\Psi$  :

$$\begin{aligned} S(\Psi + \varepsilon) &\simeq L_\Psi(\varepsilon) = \frac{1}{2} \sum_{j=1}^n \omega_j r_j^2(\Psi) + \sum_{j=1}^n \omega_j r_j(\Psi) \varepsilon^T J_j(\Psi) + \frac{1}{2} \sum_{j=1}^n \omega_j (\varepsilon^T J_j(\Psi))^2 \\ &= S(\Psi) + \varepsilon^T \mathbf{J}(\Psi) W \mathbf{r}(\Psi) + \frac{1}{2} \varepsilon^T \mathbf{J}^T(\Psi) W \mathbf{J}(\Psi) \varepsilon \\ &= S(\Psi) + \varepsilon^T \nabla S(\Psi) + \frac{1}{2} \varepsilon^T \mathbf{J}^T(\Psi) W \mathbf{J}(\Psi) \varepsilon \end{aligned}$$

where  $\mathbf{r}(\Psi)$  is the vector of size  $n$  whose  $j^{\text{th}}$  component is  $r_j(\Psi)$ ,  $\nabla S(\Psi) = \mathbf{J}(\Psi) W \mathbf{r}(\Psi)$  is the gradient vector of  $S$  with respect to  $\Psi$ ,  $\mathbf{J}(\Psi)$  is the  $N \times n$  matrix whose the  $j^{\text{th}}$  column is  $J_j(\Psi)$ , and  $W$  is the  $n \times n$  diagonal matrix of the weights, i.e  $W_{jj} = \omega_j$  and  $W_{ij} = 0$  for  $i \neq j$ ,  $i, j = 1, \dots, n$ .

The principle of the algorithm is to find at the  $(t+1)^{\text{th}}$  iteration, the vector  $\Psi^{(t+1)} = \Psi^{(t)} + \varepsilon^{(t)}$  such that :

$$\varepsilon^{(t)} = \underset{(\Psi^{(t)} + \varepsilon) \in \mathcal{D}}{\operatorname{argmin}} L_{\Psi^{(t)}}(\varepsilon) \quad (3)$$

Since the matrix  $\mathbf{M}(\Psi) = \mathbf{J}^T(\Psi)W\mathbf{J}(\Psi)$  is generally positive definite (see section 3.1.3 for the cases where  $\mathbf{M}(\Psi)$  is only positive semidefinite), the global minimum of  $L_{\Psi}$  can be obtained by differentiating  $L_{\Psi}$  with respect to the components of  $\varepsilon$  and equating to zero. This leads to :

$$\varepsilon^{(t)} = -\mathbf{M}^{-1}(\Psi^{(t)})\nabla S(\Psi^{(t)}).$$

Hence the updating equation is :

$$\Psi^{(t+1)} = \Psi^{(t)} - \mathbf{M}^{-1}(\Psi^{(t)})\nabla S(\Psi^{(t)}). \quad (4)$$

The Gauss-Newton method is known to have a good convergence rate when  $\Psi^{(t)}$  is not far from the local minimum  $\Psi^*$  and when the weighted residuals  $\sqrt{\omega_j}r_j(\Psi^*)$  are close to zero (see e.g Madsen *et al.*, 2004a). This case is generally encountered for variogram fitting since the weights are small for large lag vectors and the variogram at the vicinity of the origin can be well fitted in most of the cases by adequately choosing the initial values (see subsection 3.2.2 for the automatic choice of initial values that are not far from the solution). Nevertheless, there are some cases for which the algorithm fails to converge. In section 3.1.2 we present a modification based on trust regions which makes the Gauss-Newton algorithm more robust to the choice of initial

values.

### 3.1.2 Trust region based method

Starting from the consideration that a Taylor approximation is all the more accurate as  $||\varepsilon||$  is small, trust region based methods assume that the quadratic approximation is accurate inside a region around the current value  $\Psi^{(t)}$ . The quadratic optimization problem of equation (3) is solved under the constraint that  $||\varepsilon|| \leq \delta_t$  where  $\delta_t$  is a positive radius. Then the gain observed on  $S$  is compared to the expected gain, i.e the gain of  $L_{\Psi^{(t)}}$ . If the expected gain is close to the observed gain, then the radius of the trust region is increased, otherwise it is decreased.

At the  $(t+1)^{\text{th}}$  iteration, we compute a candidate  $\Psi_c$  for the next iteration by solving :

$$\Psi_c = \underset{\Psi \in \mathcal{D}}{\operatorname{argmin}} L_{\Psi^{(t)}}(\Psi - \Psi^{(t)})$$

under the following set of constraints on the vector  $\mathbf{R} = \Psi - \Psi^{(t)}$  :

- (i)  $\Psi$  has to belong to the set of its allowed values  $\mathcal{D}_{\Psi}$
- (ii)  $|R_i| = |\Psi_i - \Psi_i^{(t)}|$  has to be lower than  $\delta_t \gamma_i$  where  $\gamma_i$  is a scaling parameter.

The method to obtain the constrained minimum  $\Psi_c$  of  $L_{\Psi}$  is a classical quadratic optimization problem under linear constraints. The Powell's Dog Leg algorithm provides an approximation to this minimum. Here we use an

exact algorithm as described for instance in Madsen *et al.* (2004b). It gives the exact result in a finite number of iterations.

Then we compare the gain in the cost function with the predicted gain by computing the ratio :

$$\alpha = \frac{S(\Psi_c) - S(\Psi^{(t)})}{L_{\Psi^{(t)}}(\Psi_c - \Psi^{(t)}) - L_{\Psi^{(t)}}(\mathbf{0})}$$

First note that the denominator of  $\alpha$  is always positive.

If  $\alpha < 0$ , we set  $\Psi^{(t+1)} = \Psi^{(t)}$  rejecting the candidate value  $\Psi_c$  as  $S(\Psi_c) > S(\Psi^{(t)})$ .

If  $\alpha > 0$ , we set  $\Psi^{(t+1)} = \Psi_c$ .

Furthermore, we update  $\delta_t$  as follows :

- if  $\alpha > 0.75$ , we set  $\delta_{t+1} = 2\delta_t$  since  $L_{\Psi^{(t)}}$  gave a good approximation of the gain,
- if  $\alpha < 0.25$ , we set  $\delta_{t+1} = \delta_t/2$  since the approximation poorly agreed with the effective gain.

### 3.1.3 Singularity of the Gauss-Newton matrix $M(\Psi)$

In some cases the matrix  $\mathbf{M}(\Psi)$  is not invertible. It means that the quadratic function  $L_\Psi$  does not have a unique minimum on  $\mathbb{R}^N$ . Instead there is a subspace  $E$  of  $\mathbb{R}^N$  on which  $L_\Psi$  is minimum. In these cases, we can obtain a particular minimum on  $E$  by replacing  $\mathbf{M}(\Psi)^{-1}$  by  $\mathbf{M}(\Psi)^-$ , a particular generalized-inverse of  $\mathbf{M}(\Psi)$ . In this work we use the Moore-Penrose

generalized-inverse (see Horn and Johnson, 1985).

#### **3.1.4 Impossible steps**

If some components are on the boundary of their definition domains and if the candidate values obtained by equation (4) are located outside the authorized range of values (by crossing the domain boundary), these components are said to be unmodifiable for the current iteration. At each iteration, the algorithm starts by computing the Gauss-Newton step, and while some components are unmodifiable, they are suppressed from the model and the Gauss-Newton step is computed again without them. The current iteration will then leave these unmodifiable components unchanged.

### **3.2 Automatic settings**

The numerical procedure described in the previous section is used to fit the parameters of the basic structures (sill, range, anisotropy ratios and rotation angles) in order to minimize the distance between an experimental quantity and the model.

Note that all basic structures do not share the same number and type of parameters. For example :

- the nugget effect component is simply defined by its sill ;
- the spherical basic structure requires the definition of its sill and range, and possibly the anisotropy ratios and rotation angles ;

- the gamma basic structure requires the definition of an additional scale parameter ;
- the linear basic structure simply requires a slope coefficient, in addition to possible anisotropies.

The efficiency of the algorithm heavily relies on the correct choice of some algorithmic options which are described in the next paragraphs.

### 3.2.1 Algorithmic options

- **Scaling.** During the iterative procedure, each parameter is modified in turn in order to measure the impact of its variation on the cost function. These variations must be equalized over the different parameters, in order this technique to be applied for parameters as different as the range (expressed in field units), the sill (expressed as square units of the variable), the anisotropy angle (generally in degrees) or the adimensional anisotropy ratio. This is obtained by a careful scaling of each parameter.
- **Stopping criterion.** It is needed to stop the iterative algorithm. It is based on a combination of criteria :
  - the maximum number of iterations has been reached,
  - the distance between the experimental variogram and the model is small enough,
  - the size of the search rectangle is small enough.
- **Numerical gradient.** The sensitivity of the cost function must be cal-

culated. This involves the calculation of its partial derivative against each parameter. This derivative is approximated by the numerical gradient obtained as follows :

$$\frac{df(x_1, x_2, \dots, x_n)}{dx_1} = \frac{f(x_1 + \epsilon, x_2, \dots, x_n) - f(x_1 - \epsilon, x_2, \dots, x_n)}{2\epsilon}$$

This simplification avoids having to check the differentiability of the cost function.

### 3.2.2 Initial values and bounds

When starting the *foxleg* iterative procedure, each parameter must be set to an initial value and the corresponding bounds must be defined.

The rules are different depending on the parameter type :

- the sill of each basic structure is set by default to a value equal to the total variance divided by the number of structures ;
- the range of each basic structure is set by default to half of the maximum distance divided by the number of basic structures (the nugget effect component is discarded). The maximum distance is obtained as the longest distance for which the experimental variogram has been calculated ;
- the anisotropy ratio is set to 1 (isotropic hypothesis) ;
- the anisotropy angle is arbitrarily set to 0.

Similarly the bounds depend on the parameter type. For example :



- the sill and the range must be positive,
- the rotation angle has no bound.

### 3.2.3 Model reduction

Obviously, the cost function evaluation is improved when the set of parameters gets larger. On the other hand, the principle of parsimony leads to favor the model with the smallest number of parameters. According to this principle, an additional feature has been included in the procedure in order to find a parcimonious model.

When the iterative procedure is ended, the resulting model is analyzed : if a basic structure represents a too small part of the total variability (e.g 5%), this structure is discarded of the model with its corresponding parameters. Note that the model reduction criterion can be different for each basic structure. For instance the nugget effect may be added on purpose (for relaxing the kriging system when the model is composed of a Gaussian covariance only) and should not be removed by the automatic fitting procedure.

As soon as a basic structure is discarded, *foxleg* iterative procedure is started again. If the previous *foxleg* procedure had converged before the maximum number of iterations, the remaining parameters are kept with their current values. Otherwise, their values are automatically reinitialized.

## 4 Adaptation to the multivariate case

We consider  $K$  variables and for each  $i, k = 1, \dots, K$ , we have the cross-variograms (or univariate variograms when  $i = k$ )  $\hat{\gamma}^{(ik)}(\vec{h}_j)$ ,  $j = 1, \dots, n$ . Note that  $n$  is the total number of lag vectors  $\vec{h}_j$ . In the heterotopic case, we can have some pairs  $(i, k)$  for which  $\hat{\gamma}^{(ik)}(\vec{h}_j)$  is not defined for a given  $j$ . In this case, we consider that  $\hat{\gamma}^{(ik)}(\vec{h}_j) = 0$ .

For the coregionalization model we consider  $p$  basic normalized variogram structures (with sill equal to one) and each simple and cross-variogram  $\hat{\gamma}^{(ik)}$  is modeled as follows :

$$\gamma^{(ik)}(\vec{h}) = \sum_{r=1}^p \lambda_{ik}^{(r)} \gamma_{\theta_r}(\vec{h}), \quad (5)$$

Each coregionalization matrix  $\Gamma^{(r)}$ , defined by  $\Gamma_{ik}^{(r)} = \lambda_{ik}^{(r)}$  for  $i, k = 1, \dots, K$ , must be positive definite. In the following,  $\mathcal{D}_\Lambda$  will denote the subset of  $\mathbb{R}^{pn(n-1)/2}$  in which  $\Lambda = (\lambda_{ik}^{(r)})_{\substack{i,k=1,\dots,K \\ r=1,\dots,p}}$  is well defined. The positive definiteness of the coregionalization matrices is difficult to obtain by the *foxleg* algorithm since it imposes non-linear constraints on the terms  $\lambda_{ik}^{(r)}$  for  $i, k = 1, \dots, K$ .

In order to ensure the positive definiteness of the coregionalization matrices, a first possibility is to change the parameterization of the problem by using a decomposition of each matrix  $\Gamma^{(r)}$ . For instance, if  $\Gamma^{(r)} = L^{(r)}L^{(r)T}$  is the Cholesky decomposition of  $\Gamma^{(r)}$  with  $L^{(r)}$  a lower triangular matrix, then the terms of  $L^{(r)}$  can vary in  $\mathbb{R}$  and we can perform the optimization for this

new set of parameters.

This method can be used when the number of variables  $K$  and the number of basic structures  $p$  are small enough. Otherwise, the number of parameters is too high and the optimization algorithm often fails to converge. For these latter cases, we propose to reduce the space dimension by using an implicit profiled cost function based on the algorithm proposed by Goulard and Voltz (1992) as explained below.

#### 4.1 Minimization of a profiled cost function

To fit the coregionalization model of equation (5), we minimize the following cost function :

$$S(\Psi) = \sum_{j=1}^n \sum_{i,k=1}^K \omega_{ikj} \left( \gamma_{\Psi}^{(ik)}(\vec{h}_j) - \hat{\gamma}^{(ik)}(\vec{h}_j) \right)^2$$

An improvement has been carried out compared to the initial implementation : as for the multi-directional case, the weights must be normalized to give the same weight to each cross-variogram. In the heterotopic case, the weights  $\omega_{ikj}$  are set to 0 if  $\hat{\gamma}^{(ik)}(\vec{h}_j)$  is not computed.

Since  $\Psi = (\Theta, \Lambda)$  where  $\Theta$  stands for the non-linear parameters (scale, anisotropy parameters and shape parameters) and  $\Lambda$  stands for the sill, we can consider  $S$  as a function of  $\Theta$  and  $\Lambda$ . When  $\Theta$  is given, the algorithm proposed in Goulard and Voltz (1992) to solve the ordinary least-squares problem gives :

$$\Lambda^*(\Theta) = \underset{\Lambda \in \mathcal{D}_\Lambda}{\operatorname{argmin}} S(\Theta, \Lambda).$$

We use these remarks to reduce the dimension of the space over which the minimization is performed as explained above. We consider the reduced cost function also called profiled cost function :

$$\begin{aligned} S_r(\Theta) &= S(\Theta, \Lambda^*(\Theta)) \\ &= \sum_{j=1}^n \sum_{i,k=1}^K \omega_{ijk} \left( \gamma_{(\Theta, \Lambda^*(\Theta))}^{(ik)}(\vec{h}_j) - \hat{\gamma}^{(ik)}(\vec{h}_j) \right)^2 \end{aligned}$$

then

$$\Theta^* = \underset{\Theta \in \mathcal{D}_\Theta}{\operatorname{argmin}} S_r(\Theta)$$

can be obtained with the algorithm presented above. Note that, at each iteration, the Goulard's algorithm has to be used several times : once for each evaluation of  $S_r$  and twice for each evaluation of the partial derivatives of  $S_r$  with respect to each  $\theta_k$ .

Finally, we obtain :

$$\Psi^* = (\Theta^*, \Lambda^*(\Theta^*)).$$

Note that in this formulation, we cannot fix the definition domain  $\mathcal{D}_\Lambda$  since the Goulard's algorithm only provides  $\Lambda$  such that all the coregionali-

zation matrices are positive definite and *foxleg* algorithm only works on the vector  $\Theta$ . To fix more constraints on the sill parameters  $\lambda_{ik}^{(r)}$ , a modification of the Goulard's algorithm should be developed.

Therefore the final fitting procedure combines the classical iterative procedure for fitting all the parameters (except the sill) with the Goulard's algorithm used to obtain the optimal matrices of sills (assuming that all the other parameters are fixed).

## 5 Examples

### 5.1 Variogram

In this first test, we fit the experimental variogram plotted on the left side of figure (1). The fitting algorithm is performed (using a nugget effect, a cubic variogram and a spherical variogram as basic functions), convergence is reached after 33 iterations only. The model plotted after each iteration shows that convergence is almost immediate for such a simple case.

Another illustration consists in representing the cost function (i.e. the distance between the experimental variogram and the model) as a function of the number of iterations. We fit the same experimental variogram as before, but we consider an initial set of 5 basic structures (nugget effect, gaussian, cubic, exponential and spherical components) : the number of parameters is equal to 9 (nugget effect and the sill and the range for each basic structure). The left part of figure (2) shows the monotoneous decrease of the cost function

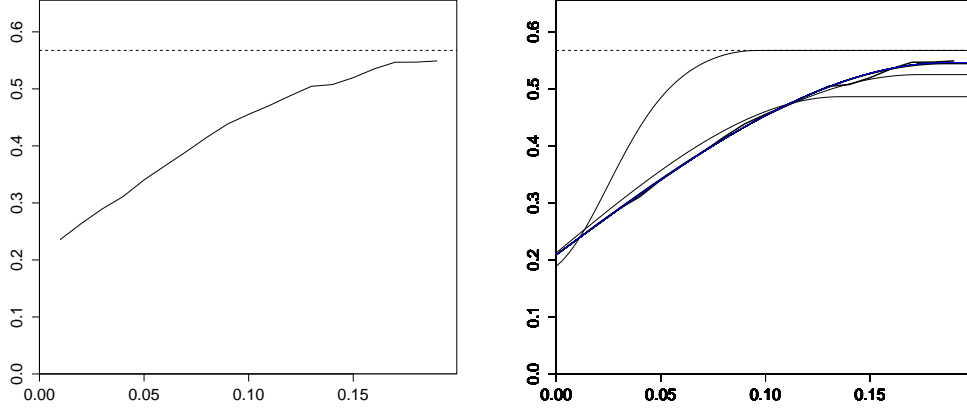


FIG. 1 – When fitting the experimental variogram [left], iterations represented in increasing dark lines [right].

with the rank of the iteration ; the right part shows the evolution of the  $\delta$  parameter. During the first eight iterations, the cost function decreases rapidly, while the  $\delta$  parameter increases (in a non-monotoneous way). When the cost function flattens, the  $\delta$  parameter starts decreasing in order to confirm that the minimum score has really been reached. At iteration #33, the iterative algorithm is ended.

In this experience, all the basic structures are kept (even when their sill becomes negligible). In particular, at iteration #3, the gaussian and the cubic components become useless ; at iteration #7, the exponential term disappears. The other two basic structures (i.e. the nugget effect and the spherical basic structures) are kept in the final model. Usually, when some basic structures are discarded, the optimization step is launched again with

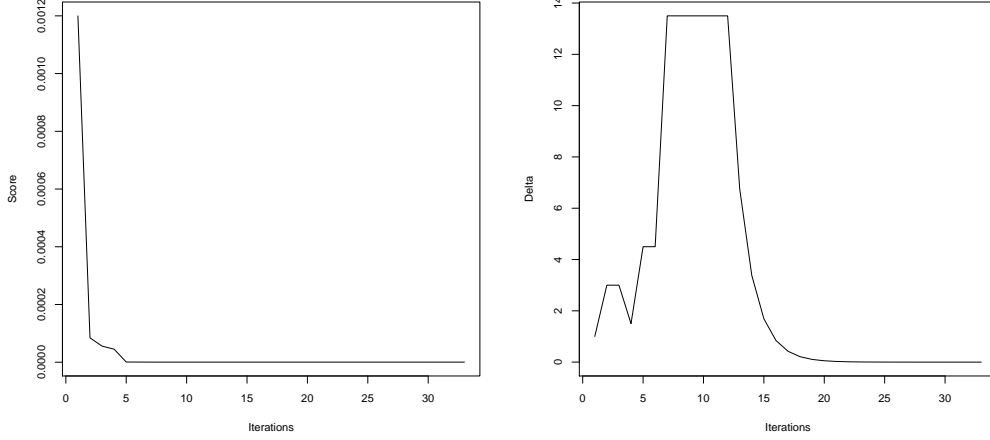


FIG. 2 – Cost function [left] and  $\delta$  parameter [right] as a function of the number of iterations

the reduced set of basic functions. This step is bypassed here.

The second example illustrates the automatic model fitting for an anisotropic variable defined in the 2-D space, based on two directional experimental variograms. Obviously in this experience, we are not able to infer the anisotropy directions (we assume that the main axes of the anisotropy ellipse are parallel to the directions where the experimental variogram has been calculated). We rather focus on the sills and ranges along the main axes of the anisotropy ellipse, for each basic structure. Figure (3) shows the experimental variograms and the corresponding model.

The initial set of basic structures consists of a nugget effect, and gaussian, cubic, exponential and spherical components. The number of parameters is equal to 13. The first descent is ended after 229 iterations. The nugget effect

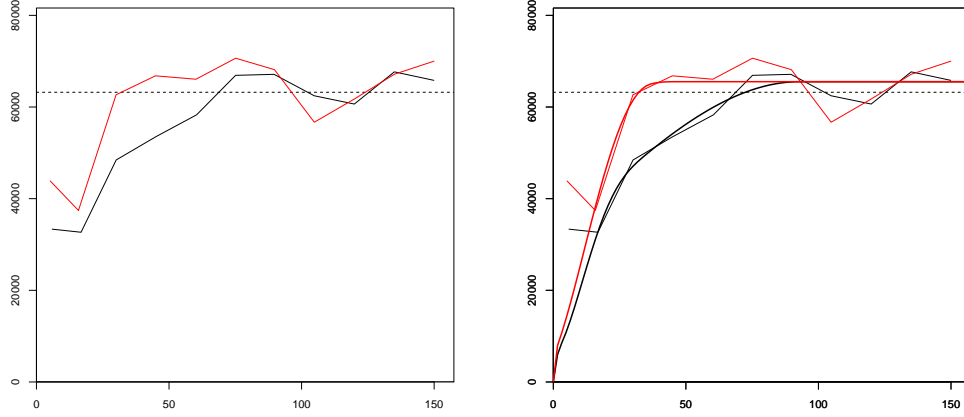


FIG. 3 – The experimental variogram [left] and the corresponding anisotropic model [right]

and exponential basic structures are then discarded and the fitting procedure is launched again. After 92 more iterations, the final anisotropic model is obtained.

## 5.2 Variogram map

Due to its generality, *foxleg* algorithm has been applied to various experiments. A promising one is to fit the model of a variable starting from an experimental variogram map. Obviously, such an information provides much more knowledge on the spatial structure than the traditional experimental variogram, even when calculated in several directions. The number of samples changes from the number of lags for experimental variograms to the number of nodes of the grid where the variogram map is calculated. For the sake of



demonstration, we first define a model composed of two nested structures (4) :

- a short anisotropic spherical component with ranges equal to 4 and 10 units and 20 degrees for the rotation angle of the main axis of the anisotropy ellipse,
- a long anisotropic spherical component with ranges equal to 20 and 40 units and an angle of -30 degrees.

Note that the model is rather complex as both structures show anisotropy but do not share the same rotation. A realization of this model is (unconditionally) simulated on a 2-D grid of 100 by 100 units. Finally a variogram map is calculated on a grid of 101 by 101 grid meshes (from -50 to +50).

We start *foxleg* algorithm from the information carried by the experimental variogram map : it uses 101\*101 data for inferring the few parameters of the model. The process is therefore rather slow but the result is better than with experimental variograms as the fit simultaneously accounts for all directions and all distances up to 50 units (half of the variogram grid extent). Although the initial information carried by the variogram map is perturbed by the statistical fluctuations of the simulation, the resulting model is quite close to the initial model as it is composed of :

- a first spherical component with ranges of 4.8 and 9.9 and a rotation of 16.9 degrees,
- a second spherical component with ranges of 19.8 and 51.6 and a rotation of -29.5 degrees.

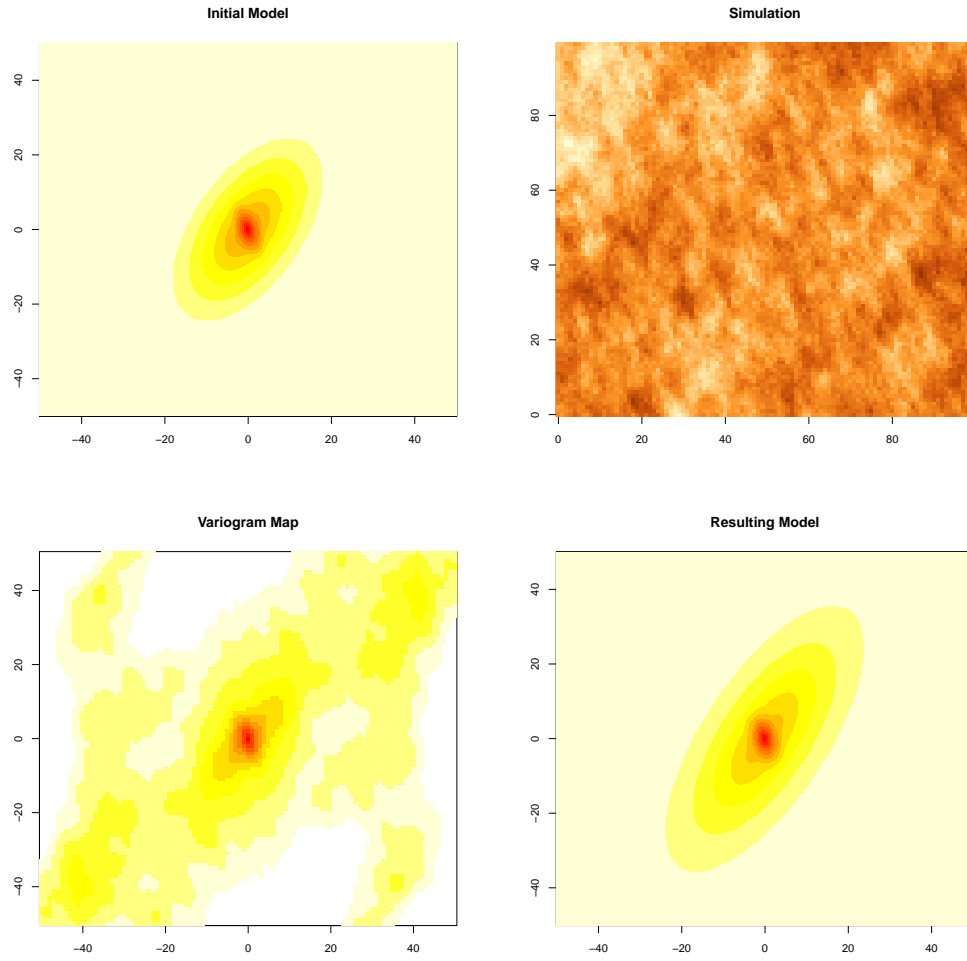


FIG. 4 – Variogram map of the initial model [upper-left], simulation [upper-right], experimental variogram map [lower-left] and variogram map of the resulting model [lower-right]

### 5.3 Multivariate case

This section demonstrates the ability of *foxleg* algorithm to operate in the multivariate case (2 variables in this case, called A and B) when simple and cross experimental variograms have been calculated in two directions. A preliminary remark shows that, in this 2-D setup, the anisotropy angle cannot be inferred : the only parameters that can be obtained are the ranges in the calculation directions and the sill matrices for the different structures.

Let us consider that we wish to infer the model with an initial set of basic structures composed of a nugget effect, a cubic, a spherical and an exponential basic structures. The total number of parameters depends upon the technique used to fit the sills of the structures :

- if Goulard’s algorithm is used : 2 for the ranges of each structure (except the nugget effect) for a total of 6 parameters. The sill matrices are inferred by the Goulard’s algorithm,
- otherwise : to the 6 previous parameters, we must add one sill matrix (i.e. 3 more parameters for the corresponding  $L$  matrix) per structure (including the nugget effect), for a total of 18 parameters.

In the first trial, we use *foxleg* combined with Goulard’s algorithm. A first convergence is reached after 30 iterations. The resulting model is then cleaned : the nugget effect and the cubic basic structures are discarded and the iterative algorithm is launched again with only 4 parameters left. A set of 41 additional iterations is needed to reach the final minimum score of the cost function. The resulting model is displayed in Figure (5).

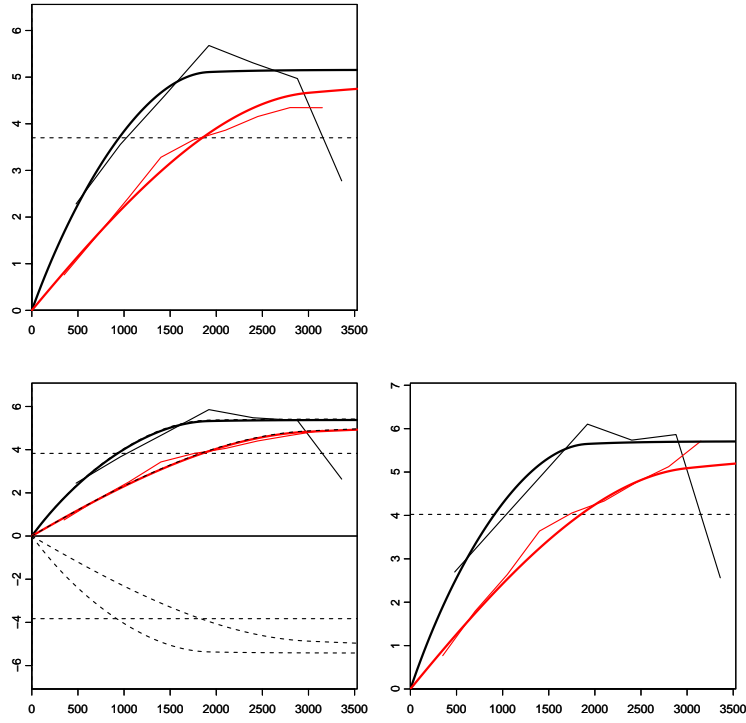


FIG. 5 – Simple variogram of A (upper-left), cross-variogram of A and B (lower-left) and simple variogram of B (lower-right). Multivariate fit using Goulard's algorithm.

In a second trial, we let *foxleg* algorithm also determine the sill matrices (through the Cholesky decomposition terms). Note that, due to the large number of parameters, the number of iterations is much larger. The first set of iterations is interrupted as convergence is not reached before the maximum number of iterations (1000). The same basic structures as previously are discarded : the nugget effect and the exponential basic structures. As the convergence has not been reached, the remaining basic structures are reset to their initial values, which leads to a severe increase of the cost function. *foxleg* algorithm is resumed for 854 more iterations. The resulting model is displayed in Figure (6).

Note that, although the fitting procedure is different, the resulting models are very similar. However, the computing time is significantly larger.

## 6 Discussion

In this paper, we propose an algorithm to automatically fit a model starting from experimental variograms, variogram maps or cross-variograms. The method is able to characterize the usual anisotropies (angles and ratios) even for high ratios in 3D. By implementing a procedure for reducing the model dimension, this algorithm provides a trade-off between the fit quality of the resulting model and its parsimony. The multivariate case is treated by combining the *foxleg* method with the algorithm from Goulard and Voltz (1992). Several illustrations show the efficiency of the method which is implemented

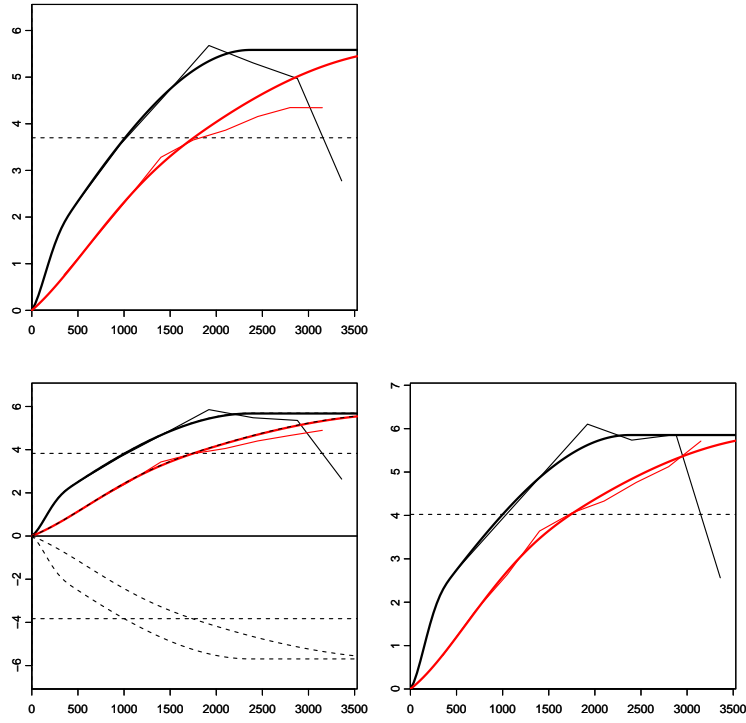


FIG. 6 – Simple variogram of A (upper-left), cross-variogram of A and B (lower-left) and simple variogram of B (lower-right). Multivariate fit without using Goulard's algorithm.

in the R package RGeoS (see Renard *et al.*) and in the commercial software Isatis<sup>®</sup> of Geovariances<sup>TM</sup>.

The main limitation of the algorithm arises in cases where the cost function presents several distinct modes leading to strong differences between the different fits. This is mainly the case when trying to fit a periodic variogram (in one dimension) or pseudo-periodic ones (see e.g de Fouquet *et al.*, 2011). The result for the periodicity parameter strongly depends on the initial values given to the numerical algorithms. Finding the global mode would require to test different initial values or to fix the periodicity parameter.

Future work will concern the indirect fit of the indicator variograms in the plurigaussian framework (see Armstrong *et al*, 2003). In this case, the model of the underlying Gaussian random functions is obtained through the automatic fit performed on the experimental variograms of the facies indicators which are defined by truncating the Gaussian random function.

## 7 References

- Armstrong, M., Galli, A., Beucher, H., Le Loc'h, G., Renard, D., Doligez, B., Eschard, R., Geffroy, F. (2011) Plurigaussian simulation in geosciences. *Springer-Verlag*, Berlin.
- Chilès, J-P. and Delfiner, P. (2012) Geostatistics : Modeling spatial uncertainty. 2<sup>nd</sup> edition. *Wiley*, New York.
- Cressie, N. (1985) Fitting variogram models by weighted least squares. *Ma-*

*thematical Geology*. Vol. 17 (5), p. 563-586.

Cressie, N. (1993) Statistics for Spatial Data. *Wiley*, New York.

Cressie, N. and Lahiri, N. (1996) Asymptotics for REML estimation of spatial covariance parameters. *Journal of Statistical Planning and Inference*. Vol. 50.

de Fouquet, C., Malherbe, L. and Ung, A. (2011) Geostatistical analysis of the temporal variability of ozone concentrations. Comparison between CHIMERE model and surface observations. *Atmospheric Environment*. Vol. 45, p. 3434-3446

Diggle, P. J. and Ribeiro, P. J. (2007) Model-based Geostatistics. *Springer Series in Statistics*.

Emery, X. (2010) Iterative algorithms for fitting a linear model of coregionalization. *Computers & Geosciences*. Vol. 36., p 1150-1160.

Goulard, M. and Voltz, M. (1992) Linear coregionalization model : Tools for estimation and choice of cross-variogram matrix. *Mathematical Geology*. Vol. 24, p 269-286.

Handcock, M. S. and Wallis, J. R. (1994) An approach to statistical spatial-temporal modeling of meteorological fields. *Journal of the American Statistical Association*. Vol. 89 (426), p 368-390.

Horn, R. A. and Johnson, C. R. (1985) Matrix Analysis. *Cambridge University Press*.



- Larrondo, P. F., Neufeld, C. T., Deutsch, C. V., (2003) VARFIT : A program for semi-automatic variogram modeling. *In : Deutsch, C.V. (Ed.), Fifth Annual Report of the Centre for Computational Geostatistics. University of Alberta, Edmonton, 17 pp.*
- Mardia, K. V. and Marshall, R.J. (1984). Maximum likelihood estimation of models for residual covariance in spatial regression. *Biometrika*. Vol. 71 (1), p. 135-146.
- Madsen, K., Nielsen, H.B. and Tingleff, O. (2004a) Methods for non-linear least squares problems, 2<sup>nd</sup> edition. *Technical report, Informatics and Mathematical Modeling, Technical University, Denmark.*
- Madsen, K., Nielsen, H.B. and Tingleff, O. (2004b) Optimization with constraints, 2<sup>nd</sup> edition. *Technical report, Informatics and Mathematical Modeling, Technical University, Denmark.*
- Magneron, C., Jeanne, N., Le Moine, O. and Bourillet, J-F. (2009) Integrating prior knowledge and locally varying parameters with Moving-GeoStatistics : methodology and application to bathymetric mapping. *GeoEnv VII - Geostatistics for Environmental Applications*. Springer.
- Pardo-Igúzquiza, E. (1999) VARFIT : A FORTRAN-77 program for fitting variogram models by weighted least squares. *Computers & Geosciences*. Vol. 25, p 251-261.
- Renard, D., Bez, N., Desassis, N. and Laporte, F., (2010). RGeoS : The Geostatistical package [6.1.2]. Ecole des Mines de Paris. Free download from

[www.geosciences.mines-paristech.fr](http://www.geosciences.mines-paristech.fr)

Stein, M. (1999) Interpolation of Spatial Data : Some Theory for Kriging.  
*Springer*, New York.