Speeding up in SSFEM computation using Kronecker tensor products

Roman Gaignaire, Frédéric Guyomarc'H, O Moreau, Stéphane Clénet, B Sudret

SPEEDING UP IN SSFEM COMPUTATION USING KRONECKER TENSOR PRODUCTS

R. Gaignaire[1], F. Guyomarc'h[2], O. Moreau[3], S. Clénet[4], B. Sudret[5]

[1]ACE, Montefiore Institute, Sart Tilman Campus, Grande Traverse, 10 (B28), B-4000 Liège, Belgique, rgaignaire@gmail.com
[2]INRIA, Lille Nord-Europe, 59640 Villeneuve d'Ascq, France, frederic.guyomarch@irisa.fr
[3]Electricité de France, R&D Division, 1 Avenue du Général de Gaulle, 92141 Clamart, France, olivier.moreau@edf.fr
[4]L2EP, ENSAM, 8 Boulevard LOUIS XIV, 59046 Lille, France, stephane.clenet@lille.ensam.fr
[5]Electricité de France, R&D les Renardières 77818 Moret-sur-Loing, France, bruno.sudret@edf.fr

Spectral Stochastic Finite Element Method (SSFEM), makes it possible to convey some random aspects of input data to the output data. However, the system size dramatically increases with the number of input random variables. Using matrix Kronecker tensor products for system solving reduces noticeably both the computation time and the storage requirements.

***Index Terms*—Finite Element Method, Electrokinetic's, Hermite polynomial chaos, Linear system resolution, Random media.**

I. INTRODUCTION

DUE TO AGING, manufacturing process or a lack of knowledge in state variables (pressure, temperature …), some uncertainties may appear in the classical input data of numerical electromagnetism modeling (such as material characteristics, loading or geometrical dimensions). An approach to take into account such uncertainties consists in modeling input data as random variables and broadcast this randomness to the output data. The methods, we will deal with, are related to Finite Element Method (FEM). First method, Monte Carlo Simulation method (MCSM) builds a sample of R numerical values of input data and then to solve the R associated deterministic problem to obtain a sample of size R of the output data. Then, statically treatment should be applied to analyze the results. This method has been remaining our reference for all of our work [1-2]. Other methods which have been proposed in mechanical engineering are based on the so-called Hermite's polynomial chaos. One of them, called non intrusive owing to they use FEM code as black-box [3], will be used in the last section. As in MCSM, numerous deterministic finite element problems have to be solved. Another kind, called intrusive as it requires the finite element code to be deeply modified, is the Spectral Stochastic Finite Element Method (SSFEM) [1-2]. The underline of this method is to write both input and output data as an expansion of Hermite chaos polynomials. Then, solving the random problem is equivalent to find deterministic values of an expansion in a Hilbertian base. Such resolution is performed by a Galerkin approach. Only one linear system have to be solved meaning that the matrix coefficient are related both to spatial and random mesh. This method features a general theoretical frame but unfortunately quickly leads to large systems. Standard numerical techniques lead both to long computation time and huge RAM storage requirements which until now prevent from tackling many industrial applications. Kronecker tensor products method [4] has been tested to overcome this drawback on an actual electrokinetic case. First, mathematical framework and discretization schemes applied to obtain the SSFEM discrete equations will be reminded. Secondly, from the equations, the Kronecker tensor product approach will be detailed. The validations have be carried out by comparing SSFEM results involving Conjugate Gradient (CG) with either standard stiffness matrix assembly or Kronecker product technique. A discussion about numerical consideration will be proposed. To illustrate the efficiency of this last approach, global current flux will be compute on an industrial case of a power line using both tensor approach SSFEM and standard non intrusive method [3].

Digital Object Identifier inserted by IEEE

II. GLOBAL FRAMEWORK

A. Problem Definition and Notations

Let us consider a spatial domain D which represents the geometry of the device (fig. 1) and Θ the random domain. The boundary of D is divided into three complementary parts ∂D_1, ∂D_2, ∂D_3. x and θ will denote the spatial and random dependence and $\boldsymbol{n}$ an outward normal vector to D. $\boldsymbol{grad}_x(.)$, $div_x(.)$, and $\boldsymbol{curl}_x(.)$ stand for the standard gradient, divergence and curl operator through the spatial dimension. The random conductivity $\sigma(x,\theta)$ is an input data whereas the current density $J(x,\theta)$, the electrical field $E(x,\theta)$ and the global current through the surface ∂D_1 $I(\theta)$ are the output data.

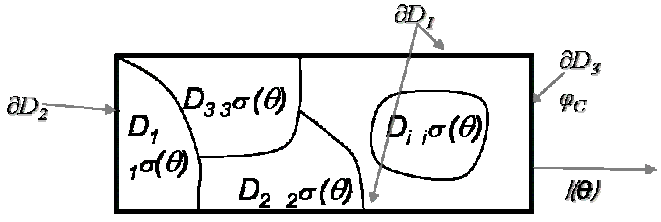

Fig. 1. Domain D divided in M sub domain D_i

D is assumed to be divided into M sub domains D_i where the conductivity ${}_i\sigma(\theta)$ is supposed to be a uniform but independent random with finite variance. Let us denote $1_{Di}(x)$

the indicator function of the sub domain D_i which is equal to 1 if x belongs to D_i and 0 elsewhere. Conductivity may then be written as:

$$\sigma(x,\theta) = \sum_{i=1}^{M} {}_i\sigma(\theta)\, 1_{D_i}(x). \quad (1)$$

The constitutive law linking the current density to the electrical field through the conductivity can be written as:

$$J(x,\theta) = \sigma(x,\theta)\boldsymbol{E}(x,\theta). \quad (2)$$

On ∂D_3, the normal component of J is supposed to be zero, whereas ∂D_2 and ∂D_3 are supposed to be equipotential surfaces with a difference of potential φ_C. Let us introduce a function $\alpha(x)$ such as:

$$\alpha(x) = \begin{cases} 0 \text{ on } \partial D_2, \\ 1 \text{ on } \partial D_3 \end{cases}. \quad (3)$$

As the electrical field is curl free, we know it exists a scalar potential, denoted $\varphi(x,\theta)$ such that :

$E(x,\theta) = \boldsymbol{grad}_x(\varphi(x,\theta) + \varphi_C\, \alpha(x))$.

This definition allows imposing boundary conditions, and Maxwell equations can then be written as:

$$\begin{cases} div_x(J(x,\theta)) = 0 \text{ on } D; \\ \boldsymbol{E}(x,\theta) = \boldsymbol{grad}_x(\varphi(x,\theta) + \varphi_C\alpha(x)), \text{ on } D; \\ J(x,\theta) = \sigma(x,\theta)\boldsymbol{E}(x,\theta); \\ J(x,\theta).\boldsymbol{n} = 0 \text{ on } \partial D_1 \otimes \Theta; \\ \varphi(x,\theta) = 0 \text{ on } \partial D_2 \otimes \Theta; \\ \varphi(x,\theta) = \varphi_C \text{ on } \partial D_3 \otimes \Theta. \end{cases} \quad (4)$$

B. Discretization Scheme SSFEM

To numerically solve (4), a spatial mesh is needed. Let us consider a Finite Element mesh. In the deterministic case, the scalar potential is spanned in the space of the nodal function related to the node i $\lambda_i(x)$. We assume there are n spatial unknowns (the scalar potential) among the number N of nodes. They are degrees of freedom related to the spatial dimension (SDoF). Then, there are N-n nodes where the scalar potential is imposed by the boundary conditions on ∂D_2 and ∂D_3 (see (4)). We now may choose the function α as a unitary linear combination of shape functions related to the nodes belonging to ∂D_3 [1,3].

A random mesh is also necessary to characterize those n unknown random variables ${}_i\varphi(\theta)$. As there are M different independent random variables ${}_i\sigma(\theta)$ as input, M-multidimensional Hermite polynomial will be used [1-3]. Let us denote ${}^g\psi(\xi(\theta))$ the gth multidimensional Hermite polynomial with variable ξ which is a random normal vector of size M. We will use the so-called polynomial chaos of M dimensions and of order p which is the sub space of random variables with finite variance spanned by the M dimension Hermite polynomials with order up to p. Such space is of dimension $P = C_{M+p}^{\,p}$. As input random variables are of finite variance and independent, we may expand them in the space of dimension P_{in}:

$$\sigma(x,\theta) \approx \sum_{i=1}^{M} \sum_{j=1}^{P_{in}} {}_i^j\sigma\ {}^j\psi((\xi_k(\theta))_{k=1}^{M})1_{D_i}(x). \quad (5)$$

The scalar potential may be now expand both in spatial (n SDoF) and random dimension with P_{out} degrees of freedom (RDoF) as:

$$\varphi(x,\theta) + V\alpha(x) = \sum_{i=1}^{n} \sum_{j=1}^{P_{out}} {}_i^j\varphi\ {}^j\Psi((\xi_k)_1^M)\lambda_i(x) + \varphi_C\alpha(x). \quad (6)$$

Where ${}_i^j\varphi$ will be the scalar unknowns of our problem.

C. Linear System

Whereas in the deterministic case a node is related to one unknown, in our case a node is related to P_{out} unknowns which correspond to the random discretization. Let us store in a list of P_{in} matrices E^j of size P_{out}^2 the mathematical expectation E(.) of the product of the j, g and m Hermite polynomials:

$$E^j(g,m) = E({}^g\Psi(\xi(\theta))\ {}^j\Psi(\xi(\theta))^m\Psi(\xi(\theta))). \quad (7)$$

By using the weak formulation of (4) with the Galerkin method, and after simple algebra, we want to find ${}_i^j\varphi$ such that [1]:

$$\begin{aligned} &\forall (f,g) \in \|1;n\| \times \|0,P_{out}\| \\ &\sum_{l=1}^{n}\sum_{m=1}^{P_{out}} {}_l^m\varphi \sum_{i=1}^{M}\sum_{j=1}^{P_{in}} \{{}_i^j\sigma E^j(g,m) \int_{D_i} (\boldsymbol{grad}_x(\lambda_l(x))\boldsymbol{grad}_x(\lambda_f(x))\} \\ &= -\varphi_C \sum_{i=1}^{M}\sum_{j=1}^{P_{in}} \{{}_i^j\sigma E^j(g,1) \int_{D_i} (\boldsymbol{grad}_x(\lambda_f(x)\boldsymbol{grad}_x(\alpha(x)))\}, \end{aligned} \quad (8)$$

From a continuous problem, a discrete problem of nP_{out} equations with nP_{out} unknowns has been defined, it is a linear system of the shape $A\varphi=B$. A is the stiffness matrix, φ is the vector of the nP_{out} unknowns and B is the load vector which contains only boundary conditions in our case. Building the stiffness matrix and solving the linear problem with a Cholesky preconditionnated CG will be denoted method A1.

III. SSFEM Kronecker Product Approach

A. Computational Issue

Let us consider a spatial domain where three sub domains have random conductivities, a spatial mesh with 6,949 spatial unknown and the scalar potential will be searched in a polynomial chaos of 3 dimensions with order until 6, that's to say we have 84 RDoF related to the random dimension. Then the problem will present 6,949*84 = 583,717 unknowns. We can overestimate the number of non-zero terms (nnz) by the following way. In a 3D tetrahedral mesh, each node is connected to about 30 other nodes. Then, on each lines of the stiffness matrix, we should have about CN*RDOF = 30*84 non-zero terms (CN: connectivity). The RAM storage requirement will be about 10.9GO. This prevent from tackling many industrial applications. Kronecker product approach [4] enables to avoid whole stiffness matrix assembly.

B. Mathematical Framework

The Kronecker product of two matrices H with n lines and m columns and K with p lines and m columns gives a matrix C with $n*p$ lines and $m*p$ columns, such that:

$$C = H \otimes K = \begin{bmatrix} H_{11}K & \cdots & H_{1m}K \\ \vdots & \ddots & \vdots \\ H_{n1}K & \cdots & H_{nm}K \end{bmatrix}. \quad (9)$$

Let us define now a list of M deterministic matrices $(A_i)_{1\le i\le M}$ related to each sub domain of D such that:

$$A_i(l,f) = \int_{D_i} (\boldsymbol{grad}_x(\lambda_l(x))\boldsymbol{grad}_x(\lambda_f(x))\}. \quad (10)$$

Each matrix A_i has less than SDoF * CN non zero terms. From this list of matrices, we may define another list of P_{in} matrices $(B^j)_{1\le j\le Pin}$ which mix spatial discretization and value of conductivity expansion:

$$B^j = \sum_{k=1}^{M} {}_k^j\sigma\, A_k. \quad (11)$$

Matrix B^j is related to the expansion of the conductivity on the j^{th} Hermite polynomial through all each domain. Each matrix B^j has less than SDoF * CN non zero terms. Moreover, as the conductivities are modeled by independent random variables, it exists some j such that for all k, ${}_k^j\sigma$ is equal to zero. For such j, the matrix B^j is empty.

Let us consider a line li defined by the index f and g such that $li = (g\text{-}1)n+f$ and a column co defined by the index l and m such that $co = (m\text{-}1)n+l$. Simple algebra from (8) show A may be written as:

$$A(li,co) = \sum_{j=1}^{P_{in}} E^j(m,g) \sum_{i=1}^{M} {}_i^j\sigma \int_{D_i} (\boldsymbol{grad}_x(\lambda_l(x))\boldsymbol{grad}_x(\lambda_f(x))\}. \quad (12)$$

By using (11),(12) and the definition (9) one can notice A may be written as a sum of Kronecker product:

$$A = \sum_{j=1}^{P_{in}} E^j \otimes B^j. \quad (13)$$

C. CG Involving Kronecker Products

In CG algorithm (like in most of iterative solvers), assembly the stiffness matrix is not necessary: we just need to know how to compute a matrix vector product. Taking into account that the operator *vect*(.) converts a n×m matrix into a transposed vector of $n*m$ lines and considering a matrix X with n lines and P_{out} columns, Kronecker product features the following interesting property:

$$(\sum_{j=1}^{P_{in}} E^j \otimes B^j)vect(X) = \sum_{j=1}^{P_{in}} vect(B^j X E^j). \quad (14)$$

By using (14), only P_{in} matrix computations are required. Each of those products involved a product of a matrix of size $n*CN$ by a matrix of size nP_{out} ($B^j X$) and then to compute the product of previous resulting matrix of size ($n*P_{out}$) by a matrix P_{out}^2. Some optimized algorithms have been developed and test to perform such operation owing to special structures of B^j and E^j.

D. Advantages and Difficulties of Kronecker Product

By using the Kronecker product approach, we just have to store the list of P_{in} matrices B^j and E^j. Then, the total storage is of the order: $P_{in}(P_{out}^2 + n * CN)$, therefore the sequential approach need to store $n * CN * P_{out}^2$ terms. By considering the same example than in the section A, one can show that we just need 0.7GO of RAM storage by using the Kronecker approach. In fact, the gain by using Kronecker product is obvious (owing to $P_{in}* P_{out}^2$ is negligible comparing to $n*CN*P_{in}$). In addition, tests carried out with 1D matrix vector using Kronecker approach have been quicker than direct stiffness matrix vector product.

Main issue concerns the preconditioning method for GC algorithm. As far as standard CG algorithm is concerned, it is quite simple to use Cholesky preconditioners (method A1). It becomes unfortunately not so obvious when the stiffness matrix is not assembled. As a fist step, present work will only deal with a Jacobi preconditioner for the CG involving tensor products (method A2).

IV. Validation And Numerical Consideration

A. Academic Case

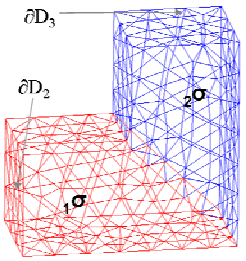

Fig. 2. L-Shape academic case

Let us consider a mesh (fig.2) with 912 tetrahedral, 276 nodes giving 234 scalar potential unknowns. The difference of potential between ∂D_2 and ∂D_3 is equal to 1. The domain is divided in two sub domains where the conductivity is supposed to be a random variable following lognormal law whose means and standard deviations are given in tab. I.

TABLE I
RANDOM PROPERTIES OF THE CONDUCTIVITY, L SHAPE

Conductivity	Mean	Standard Deviation
${}_1\sigma$~Ln(5.19,0.22)	200	100
${}_1\sigma$~Ln(3.84,0.15)	50	20

This problem has been solved with the three previous SSFEM algorithms for $p_{in} = 2*p_{out}$ with p_{out} chosen equal to 4, 5, and 6. That leads to a polynomial chaos size (P_{out}) equal to 15, 21, and 28. Previously, the SSFEM classical algorithm (A1) had been validated by comparing with a MCSM [1-2]. To validate the Kronecker approach, the value of each unknown coefficient ${}_i^j\varphi$ obtained by the three methods has been compared. By denoting ${}_i^j\varphi_1$ (respectively ${}_i^j\varphi_2$) the value obtained by A1 (resp. A2), let us define an error criterion (err) in percents by:

$$err = \max_{(i,j)\in\{1,..,n\}\otimes\{1,..,P_{out}\}} \left(\frac{|{}_i^j\varphi_2 - {}_i^j\varphi_1|}{|{}_i^j\varphi_1|} * 100\right). \quad (15)$$

TABLE II
ERROR CRITERION ON ALL THE MESH, L-SHAPE (10^{-3}%)

	$p_{out} = 4$	$p_{out} = 5$	$p_{out} = 6$
err	0.6	0.5	0.5

As we can see in tab.2, the difference between the methods is negligible: the maximum error on all the mesh for selected output order is of $0.6.10^{-3}$%.

TABLE III
STANDARD AND TENSOR CG ALGORITHM CHARACTERISTICS

Method	Unk	NbIter	Tb (s)	TCG (s)	Rate	Total Time (s)
A1 $p_{out} = 4$	3510	30	5.06	0.31	0.061	5.37
A2 $p_{out} = 4$	3510	283	2.29	0.54	0.24	2.84
A1 $p_{out} = 5$	4914	30	13.35	0.78	0.058	14.13
A2 $p_{out} = 5$	4914	353	2.7	1.22	0.45	3.92
A1 $p_{out} = 6$	6552	30	31.90	1.67	0.052	33.57
A2 $p_{out} = 6$	6552	431	3.12	2.51	0.80	5.63

Tab.3 summarizes some numerical considerations: Unk is the number of unknowns involved in the problem; NbIter is the number of iterations needed by CG to converge; Tb represents the time to build the stiffness matrix for A1 and the time needed to build both list B^j and E^j for A2; TCG is the time needed for CG convergence; Rate is the ratio between Tb and TCG, Total Time represents the complete solving time.

First, it can be noticed that the number of unknowns depends only on p_{out} as the spatial mesh is the same for all the configurations. As far method A1 is concerned, NbIter remains constant with regard to p_{out} thanks to the Cholesky preconditioner efficiency. Moreover, the CPU time for CG convergence is negligible compared with the time needed to build stiffness matrix ("rate" column). However the CG CPU time increases with p_{out} due to the fact that the size of the stiffness matrix increases dramatically with p_{out}, so that each matrix vector product involved by CG iterations become more and more CPU time consuming. So, time to build the stiffness matrix increases with about the square of p_{out}.

Concerning method A2, using Kronecker approach makes it possible to nearly get ride of the assembly time. However, the weak efficiency of the Jacobi preconditioner makes the number of iterations increases with p_{out}. Nevertheless, it turns out that A2 becomes more and more efficient as p_{out} increases (until 6 times faster than A1 for $p_{out} = 6$). This result points out that focusing on the preconditioning aspects dedicated to Kronecker products should make this technique even faster.

B. Industrial Case

Line joints are used to connect high power transmission conductors (fig.3). The study consists in computing the total current taking into account some uncertainties in three contact conductivities ($_1\sigma$,$_2\sigma$,$_3\sigma$) [2].

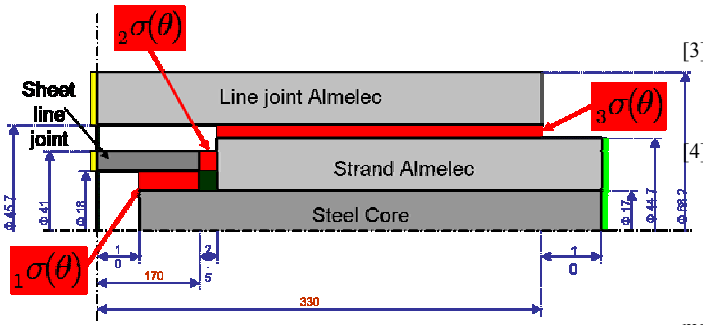

Fig. 3. Line joint

According to expert sayings, they have been modelled as random variables with uniform laws (tab. IV). SSFEM problem has been solved thanks to tensor technique whereas it was previously unfeasible to store the whole stiffness matrix (III.A).

TABLE IV
RANDOM PROPERTIES OF THE CONDUCTIVITY LINE JOINT

Conductivity laws	mean	Standard deviation
$_1\sigma \sim U([500;10\ 000])$	5,250	2,742
$_2\sigma \sim U([57;2\ 270])$	1,163.5	638.8
$_3\sigma \sim U([1\ 120;4\ 770])$	2,945	1,053.8

The validation has been carried out by comparing with the non intrusive polynomial chaos method using Hermite-Gauss projection (NIHG) [3]. It is worth noticing that unlike SSFEM, some theoretical properties about random global quantities [2] are not yet available with NIHG. Nevertheless, as far as the line joint problem is concerned, a very good agreement can be observed, with the same discretization parameters, between the NIHG and SSFEM results (fig. 2) with computation times of same order.

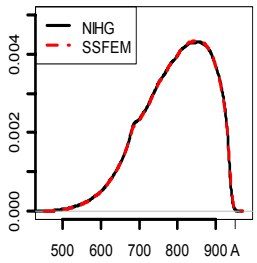

Fig. 4. probabilistic distribution of the global current in the line joint for NIHG (-) and tensor product SSFEM (- -) methods

V. CONCLUSION

CG algorithm involving Kronecker tensor products makes the SSFEM computations run faster than standard implementation. In addition, this algorithm requires much less RAM storage allowing dealing with further industrial studies. Preconditioners dedicated to this tensor technique, like block-SSOR, should increase performances even further.

REFERENCES

[1] R. Gaignaire, S. Clenet, O. Moreau, B. Sudret, 3D Spectral Stochastic Finite Element Method in Electromagnetism, IEEE Trans. Magn., vol. 43, no. 4, pp. 1209-1212, 2007.

[2] R. Gaignaire, S. Clenet, O. Moreau, B. Sudret, Current calculation in Electrokinetics Using a Spectral Stochastic Finite Element Method, IEEE Trans. Magn., to be published in 2008.

[3] Berveiller, M., Sudret, B., Lemaire, M., Structural reliability using a non intrusive stochastic finite element method, 12th IFIP WG 7.5 Working Conference on Reliability and Optimization of Structural Systems, Aalborg (Danemark), May 2005.

[4] Hermann G. Matthies and Andreas Keese, Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations, Computer Methods in Applied Mechanics and Engineering, Volume 194, Issues 12-16, , Special Issue on Computational Methods in Stochastic Mechanics and Reliability Analysis, 8 April 2005, Pages 1295-1331.

Manuscript received May 9, 2008. Corresponding author: O. Moreau (e-mail: Olivier.moreau@edf.fr; optional phone: 33-1-47-65-31-89; optional fax: 33-1-47-65-32-51).