



HAL
open science

Composites Manufacturing Processes: Towards an Advanced Simulation

Brice Bognet, Chady Ghnatios, Anaïs Barasinski, Francisco Chinesta, Francoise Masson, Adrien Leygue, Arnaud Poitou

► **To cite this version:**

Brice Bognet, Chady Ghnatios, Anaïs Barasinski, Francisco Chinesta, Francoise Masson, et al.. Composites Manufacturing Processes: Towards an Advanced Simulation. *Revue des composites et des matériaux avancés = Journal of Composite and Advanced Materials*, 2011, 21 (1), pp.23-32. 10.3166/rcma.21.23-32 . hal-01005984

HAL Id: hal-01005984

<https://hal.science/hal-01005984v1>

Submitted on 8 Feb 2017

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.



Distributed under a Creative Commons Attribution 4.0 International License

Composites manufacturing processes

Towards an advanced simulation

Brice Bognet — Chady Ghnatios — Anais Barasinski — Francoise Masson — Francisco Chinesta — Adrien Leygue — Arnaud Poitou

*EADS Corporate Foundation International Chair
"Advanced Modeling of Composites Manufacturing Processes"
GEM, UMR CNRS-ECN,
1 rue de la Noe, BP 92101, F-44321 Nantes cedex 3, France
Francisco.Chinesta@ec-nantes.fr*

RÉSUMÉ. La modélisation avancée des procédés de mise en forme de composites est confrontée à de nombreux verrous malgré les avancées récentes en matière de modélisation mécanique, analyse numérique, stratégies de discrétisation et capacité de calcul. La mise en forme de composites nécessite la prise en compte des multiples échelles en espace et temps, des comportements non-linéaires anisotropes et fortement couplés, définis dans des géométries très complexes. L'optimisation des procédés ainsi que l'indentification inverse, ou encore le contrôle des procédés nécessite plusieurs résolutions. Dans ce contexte les techniques de réduction de modèles offrent de nouvelles possibilités, permettant d'accélérer les calculs de quelques ordres de grandeur, et même de résoudre des modèles jamais résolus jusqu'à présent.

ABSTRACT. Efficient simulation of composites manufacturing processes remains today a challenging issue despite the impressive progresses reached in mechanical modeling, numerical analysis, discretization techniques and computer science during the last decade. Composites manufacturing involve multiscale models in space and time, highly non-linear and anisotropic behaviors, strongly coupled multiphysics and complex geometries. Moreover, optimization (shape and process optimization), inverse analysis (parameter identification, non destructive testing, ...) or process control need the solution of many direct problems, as fast and accurately as possible. In this context, model reduction techniques constitute an appealing simulation choice, accelerating the computations of several orders of magnitude, and even, enabling the solution of models never until now solved.

MOTS-CLÉS : Composites, Procédés de mise en forme, Modélisation avancée, Réduction de modèles

KEYWORDS: Composites, Manufacturing processes, Advanced modeling, Model reduction

1. Introduction

Composites manufacturing processes involve many different physics. All these coupled physics coexist and exhibit multiscale and localized behaviors in space and time. Because the multiscale description micro-macro modeling is mandatory, and appropriate inter-scale bridges, other than classical homogenization, must be defined. Another important issue concerns the nature of the macroscopic models defined in plate or shell domains characterized by having a dimension (the thickness) several orders of magnitude lower than the other representative in-plane dimensions. This fact, even if it is not a major conceptual issue, is a real nightmare for simulation purposes. This situation is not new, plate and shell theories were successfully developed many years ago and still are ; they were also intensively used in structural mechanics. These theories make use of some kinematic and static hypotheses to reduce the 3D nature of mechanical models to 2D reduced models defined in the shell or plate middle surface. In the case of elastic behaviors the derivation of such reduced models is quite simple and it constitutes the foundations of classical plate and shell theories. Today, most commercial codes for structural mechanics applications propose different types of plate and shell finite elements, even in the case of multilayered plates or shells composites.

However, in composites manufacturing processes the physics encountered in such multilayered plate or shell domains is much more rich, because as previously indicated it involves chemical reactions, crystallization and strongly coupled thermomechanical behaviors. The complexity of the involved physics makes the introduction of pertinent hypotheses to reduce the dimensionality of the model from 3D to 2D impossible. In that case a full 3D modeling is compulsory, and because the richness of the thickness' description (many coupled physics and many plies differently oriented) the approximation of the fields involved in the model needs thousands of nodes distributed along the thickness direction. Thus, full 3D descriptions involve thousands of millions of degrees of freedom that should be solved many times because of the history dependent thermomechanical behavior. Moreover, when we are considering optimization or inverse identification, many direct problems have to be solved in order to reach the minimum of a certain cost function. In the case of inverse analysis such cost function is the difference between the predicted and measured fields.

An other important issue encountered in the simulation of composites manufacturing is the one related to the process control and optimization. In general, optimization implies the definition of a cost function and the search of the optimum process parameters defining the minimum of that cost function. The process starts by choosing a tentative set of process parameters. Then the process is simulated by discretizing the associated model. The solution of the process model is the most costly step of the optimization procedure. As soon as the solution is available, the cost function can be evaluated and its optimality checked. If the chosen parameters do not define a minimum (at least local) of the cost function, the process parameters should be updated and the solution recomputed. The procedure continues until reaching the minimum of the cost function. Obviously, nothing ensures that such minimum is global, so more

sophisticated procedures exist in order to explore the domain defined by the parameters to escape to local minimums traps. The parameters updating is carried out in a way ensuring the highest variation of the cost function, *i.e.* in order to move along the direction of the cost function gradient. However, to identify the direction of the gradient one should compute not only the fields involved in the process model but also the derivatives of such fields with respect to the different process parameters. The evaluation of these derivatives is not in general an easy task. Conceptually, one could imagine that by perturbing slightly only one of the parameters involved in the process optimization and then solving the resulting model, one could estimate by using a finite difference formula, the derivative of the cost function with respect to the perturbed parameter. By perturbing sequentially all the parameters we could have access to the derivatives of the cost function with respect to all the process parameters (also known as sensibilities) and then defining the cost function gradient direction, on which the new trial set of parameters should be chosen. There are many strategies for updating the set of process parameters and the interested reader can find most of them in the books focusing on optimization procedures. Our interest here is not a discussion on optimization strategies, but only evidence the necessity to reduce the number of direct solutions of the process model. As we discussed in the previous paragraphs, the solution of the process model is a tricky task that demands important computational resources and usually implies extremely large computing times. Usual optimization procedures are inapplicable because they need numerous solutions of the problem defining the model of the process, one for each trial set of process parameters. The same issues are encountered when dealing with inverse analysis in which material or process parameters are expected to be identified from numerical simulation, by looking for the unknown parameters such that computed fields agree in minute with the ones measured experimentally.

Until now the getaway consisted in using the more and more powerful computing platforms (parallel and distributed computing architectures, the use of GPUs, the acceleration techniques based on the preconditioning of the algebraic systems obtained after discretization, the use of multigrid techniques or the ones making use of multi-domains, among many others). But the verdict is implacable : all these techniques are not enough, neither today nor in the next decade, when the complexity of models approach the ones of industrial interest. The brute force approach is no more a valuable alternative and consequently new proposals are urgently needed.

One of these alternatives lies in the use of model reduction strategies. Model reduction is based on the fact that the solutions of many models contain much less information than the one a priori assumed when the discrete model was built. The solution of some models usually encountered in computational engineering and science only involves a reduced amount of information that can be extracted by applying the so-called Proper Orthogonal Decomposition -POD-. The main drawback in using such a strategy lies in the fact that some a priori knowledge is compulsory in order to define the reduced model that could then be applied for solving "*similar*" problems. However, because the POD leads to a separated representation of the problem's solution, this separated description could be postulated "a priori" and then the functions involved in

such decomposition calculated. This idea is the heart of the so-called Proper Generalized Decomposition -PGD- that we revisit in Section 2. Then, PGD will be applied to solve different problems encountered in composite structures manufacturing : process optimization, on-line process control and high resolution solutions of models defined in degenerated domains, that is, in plate or shell geometries.

2. Illustrating the Proper Generalized Decomposition of a generic parametric model

Imagine for example that you are interested in solving the heat equation but that you do not know the material's thermal conductivity, because it has a stochastic nature or simply because prior to solve the thermal model you should measure it. You have three possibilities : (i) you wait to know the conductivity before solving the heat equation (a conservative solution !); (ii) you solve the equation for many values of the conductivity (a sort of Monte Carlo) and then the work is done (a sort of brute force approach !); or (iii) you solve the heat equation only once for any value of the conductivity (the cleverest alternative !). Obviously the third alternative is the most exciting one. To compute this "magic" solution it suffices to introduce the conductivity as an extra coordinate, playing the same role than the standard space and time coordinates, even if there are no derivatives concerning this extra-coordinate. This procedure runs, very well, and can be extended to introduce many other extra-coordinates : the source term, initial condition, etc. It is easy to understand that after performing this type of calculations, a posteriori inverse identification or optimization can be easily handled, but we will come back to these potential applications later.

In what follows, we illustrate the construction of the Proper Generalized Decomposition (Ladeveze 1999 ; Ammar *et al.*, 2006 ; Ammar *et al.*, 2007) of the solution by considering the following simple parametric heat transfer equation :

$$\frac{\partial u}{\partial t} - k\Delta u - f = 0 \quad [1]$$

where $(\mathbf{x}, t, k) \in \Omega \times I \times \mathfrak{S}$, and for the sake of simplicity the source term is assumed to be constant, *i.e.* $f = \text{constant}$. Because the conductivity is considered unknown, it is assumed to be a new coordinate defined in the interval \mathfrak{S} . Thus, instead of solving the thermal model for different values of the conductivity parameter we are introducing it as a new coordinate. Therefore, we solve a more general problem, but obviously the price to pay is the increase of the model's dimensionality. However, as the complexity of PGD scales only linearly (and not exponentially) with the space dimension, as we prove later, the consideration of the conductivity as a new coordinate still allows to obtain a fast and cheap solution. Thus, in this case, the solution of equation [1] is searched under the form :

$$u(\mathbf{x}, t, k) \approx \sum_{i=1}^{i=N} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) \quad [2]$$

To describe the PGD algorithm, we assume that the approximation at iteration n is already known :

$$u^n(\mathbf{x}, t, k) = \sum_{i=1}^{i=n} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) \quad [3]$$

and that at the present iteration we look for the next functional product $X_{n+1}(\mathbf{x}) \cdot T_{n+1}(t) \cdot K_{n+1}(k)$ that, to alleviate the notation, will be denoted by $R(\mathbf{x}) \cdot S(t) \cdot W(k)$. Before solving the resulting non linear model related to the calculation of these three functions, a model linearization is performed. The simplest choice consists in using an alternating-directions fixed-point algorithm. First of all, we proceed by assuming $S(t)$ and $W(k)$ given at the previous iteration of the non-linear solver and then computing $R(\mathbf{x})$. From the just updated $R(\mathbf{x})$ and the previously used $W(k)$ we can update $S(t)$. Finally from the just computed $R(\mathbf{x})$ and $S(t)$ we update $W(k)$. The procedure continues until reaching convergence. The converged functions $R(\mathbf{x})$, $S(t)$ and $W(k)$ allow us to define all the needed functions : $X_{n+1}(\mathbf{x}) = R(\mathbf{x})$, $T_{n+1}(t) = S(t)$ and $K_{n+1}(k) = W(k)$. The interested reader can refer to Chinesta *et al.* (2010) and the references therein for more details on the functional constructor.

Discussion

The construction of each term in Equation [2] needs a certain number of iterations because of the non-linearity of the problem related with the approximation given by Equation [2]. Denoting by m_i the number of iterations needed to compute the i -th sum in Equation 2], let $m = \sum_{i=1}^{i=N} m_i$ be the total number of iterations involved in the construction of the separated approximation, Equation [2]. It is easy to note that the solution procedure involves the solution of m 3D problems related to the construction of the space functions $X_i(\mathbf{x})$, $i = 1, \dots, N$; m 1D ordinary differential equations related to the construction of functions $T_i(t)$ and, finally, m diagonal linear systems related to the definition of functions $K_i(k)$. In general m rarely exceeds ten. On the other hand, the number N of sums needed to approximate the solution of a given problem depends on the solution's regularity itself, but all the experiments carried out until now reveal that this number ranges from a few tens to slightly more than one hundred. Thus, we can conclude that the complexity of the solution procedure is of some tens of 3D solutions (the cost related to the one dimensional problems being negligible in respect to the one related to the 3D problems). On the contrary, if we follow a classical approach, we should solve a 3D problem at each time step and for each value of the parameter k . In usual applications, the complexity can easily reach millions of 3D solutions. The CPU time saving, by applying the PGD, can be of several orders of magnitude.

Note also that another possibility exists :it consists in the separation of the three-dimensional physical space into a sequence of one-dimensional ones :

$$u(\mathbf{x}, t, k) \approx \sum_{i=1}^{i=N} X_i(x) \cdot Y_i(y) \cdot Z_i(z) \cdot T_i(t) \cdot K_i(k) \quad [4]$$

This possibility reduces drastically the complexity mentioned before, allowing to solve models involving hundreds of dimensions and the equivalent of 10^{300} degrees of freedom (Ammaret *al.*, 2007) in a few minutes using a standard laptop. Techniques to cope in this framework with non-paralelepipedic domains have been analyzed in Gonzalez *et al.* (2010).

The models involving moving discontinuities or hyperbolic equations could involve non-separable solutions. In that case the procedure just described converges towards the full tensor product of the bases considered in each space, that is, to the solution that would be computed by using a standard mesh based strategy. In these cases the use of the PGD technology doesn't represent nay advantage.

3. Application to processes optimization

3.1. Off-line construction of the parametric solutions

In what follows we consider a generic model involving the space and time coordinates $(\mathbf{x}, t) \in \Omega \times \mathcal{I}$, $\Omega \subset \mathcal{R}^3$ and $\mathcal{I} \subset R$:

$$\mathcal{L}(u(\mathbf{x}, t)) = \mathcal{F} \quad [5]$$

where the unknown fields depend of a set of parameters $p_1, \dots p_M$.

Optimizing the model consists of determining the best set of parameters able to minimize a certain physically derived cost function. To perform such optimization one could consider different values of the parameters and then solve the resulting model. However, optimization or inverse identification needs many solutions, and then when the number of the parameters involved in the model increases, standard approaches fail to compute optimal solutions in a reasonable computing time. In this case defining "on-line" optimization or inverse analysis is a challenging issue and in all cases real time remains unreachable.

Our proposal is to introduce all the process parameters $p_1, \dots p_M$ as extra coordinates, and then to compute the field u at each point and time and for any possible value of the model parameters, $u(\mathbf{x}, t, p_1, \dots p_M)$. As soon as this multidimensional solution is available, we could try to particularize it for any value of the process parameters without the necessity of further solutions. Thus, optimization procedures can proceed from the only knowledge of an "off-line" pre-computed parametric solution.

To circumvent the curse of dimensionality related to the high dimensional space in which the solution $u(\mathbf{x}, t, p_1, \dots, p_M)$ is defined we consider a separated representation of that field :

$$u(\mathbf{x}, t, p_1, \dots, p_M) \approx \sum_{i=1}^{i=N} X_i(\mathbf{x}) \cdot T_i(t) \cdot P_{1i}(p_1) \cdot \dots \cdot P_{Mi}(p_M) \quad [6]$$

where all the functions involved in such separated representation are computed by applying the Proper Generalized Decomposition technique widely described in Section 2.

3.2. From "off-line" optimization to "on-line" control strategies

Optimization procedures look for optimal parameters minimizing an appropriate single or multi objective cost function (sometimes subjected to many constraints). In this work we consider a simple scenario, in which the cost function reads :

$$\mathcal{C}(p_1, \dots, p_M) = \frac{1}{2} \sum_{i=1}^{i=Q} (u(\mathbf{x}_i, t_i, p_1, \dots, p_M) - \tilde{u}_i)^2 \quad [7]$$

Now, the optimal process parameters $p_1^{op}, \dots, p_M^{op}$ must be calculated by minimizing the cost function. There exist many techniques to look for such minimization, the interested reader can refer to any book on optimization. Many of them proceed by evaluating the gradient of the cost function and then moving on that direction. The gradient computation involves the necessity of performing first derivatives of the cost function with respect to the process parameters. Other techniques need computing second derivatives.

When using separated representations involving the process parameters as extra-coordinates, this task is quite simple because as the solution depends explicitly on the parameters, its derivation is straightforward :

$$\begin{aligned} \frac{\partial u(\mathbf{x}, t, p_1, \dots, p_M)}{\partial p_j} &\approx \sum_{i=1}^{i=N} X_i(\mathbf{x}) \cdot T_i(t) \cdot \\ &\cdot P_{1i}(p_1) \cdot \dots \cdot P_{j-1i}(p_{j-1}) \frac{\partial P_{ji}(p_j)}{\partial p_j} \cdot P_{j+1i}(p_{j+1}) \cdot \dots \cdot P_{Mi}(p_M) \end{aligned} \quad [8]$$

and similarly for the second derivatives. The calculation of the solution's derivatives is a tricky point when proceeding from standard discretization techniques because the parametric dependency of the solution on the parameters is not explicit.

By computing off-line the parametric solution, and then minimizing on-line cost functions, one could control real time processes.

Figure 1. *Tape placement : illustration of the different thermal cycles experienced at a certain position*

4. Full 3D simulation of thermomechanical models defined in plate domains

4.1. High resolution thermal modeling

In this section we consider the tape placement process. The process consists in considering two initial plies whose interface is heated by using a heat source that moves along the interface while a heated roller moving with the heat source press both plies ensuring the adhesion. After welding both plies, the process is repeated to add a third ply and so on until applying all the plies composing the laminate.

Thus, the process involves different thermal cycles. In practice between two consecutive cycles the temperature reaches the ambient one, fact that allows to decouple the different thermal cycles. Thus, the computation of the thermal history requires solving the different scenarios depicted in Figure 1.

If we denote by θ_i the orientation of ply i and by R_i the contact thermal resistance due to an imperfect adhesion of the plies, one could compute off-line the parametric solutions related to each laminate consisting of L plies, with $N \geq L > 1$:

$$u(\mathbf{x}, t, \theta_1, \dots, \theta_L; R_1, \dots, R_{L-1}), \quad L = 2, \dots, N \quad [9]$$

The solution for a particular orientation of the different plies can be obtained by particularizing the above solution. Thus, a single multidimensional solution suffices to cover all the possible laminates.

The thermal history at a certain position \mathbf{x} results from the superposition of the contributions related to each laminate $L = 2, \dots, N$. Knowing the thermal history, we could compute the thermal induced stress $\sigma_0(\mathbf{x})$ that will be introduced as an

initial stress field to solve the mechanical model at the structure level. The efficient 3D solution of the resulting mechanical modeling is addressed in the next section.

4.2. Mechanical structural analysis

In this section, we apply the PGD to the simulation of the thermomechanical models defined in plates-shaped domains. The PGD method allows us to separately search for the in-plane and the out-of-plane contributions to the full 3D solution, allowing significant savings in computing time and memory resources.

In an attempt of solving full 3D models keeping a 2D complexity in regard to the computing cost, we use the Proper Generalized Decomposition - PGD - widely described in previous sections, by assuming a separated representation of the displacement field

$$\mathbf{u}(x, y, z) = \begin{pmatrix} u(x, y, z) \\ v(x, y, z) \\ w(x, y, z) \end{pmatrix} \approx \sum_{i=1}^N \begin{pmatrix} u_{xy}^i(x, y) \cdot u_z^i(z) \\ v_{xy}^i(x, y) \cdot v_z^i(z) \\ w_{xy}^i(x, y) \cdot w_z^i(z) \end{pmatrix} \quad [10]$$

where $u_{xy}(x, y)$, $v_{xy}(x, y)$ and $w_{xy}(x, y)$ are functions of the in-plane coordinates whereas $u_z(z)$, $v_z(z)$ and $w_z(z)$ are functions involving the thickness coordinate.

Because neither the number of terms in the separated representation of the displacement field nor the dependence on z of functions $u_z^i(z)$, $v_z^i(z)$ and $w_z^i(z)$ are assumed *a priori*, the approximation is flexible enough to represent the full 3D solution, being obviously more general than the classical plate theories that assume particular *a priori* behaviors in the thickness direction.

The separated representation solution converges towards the solution obtained by using a full tensor product of the approximation bases defined by the in-plane and thickness approximation basis, similar to a full 3D finite element discretization, enabling fast and accurate solutions of problems unapproachable in practice by using the more experienced finite element method.

5. Conclusions

The simulation of composites manufacturing processes remains today a challenging issue despite the recent impressive progresses in computational resources. The simulation of processes of industrial interest needs the proposal of advanced numerical strategies, radically different to the ones until now widely used, usually based on finite element, finite difference or finite volumes discretizations.

In this work we explored the potentiality of techniques based on model reduction, and in particular those making use of the so-called Proper Generalized Decomposition.

This technique allows circumventing the curse of dimensionality when addressing models defined in high dimensional spaces. We proved that it could constitute a real breakthrough for addressing shape or process optimization as well as inverse analysis.

On the other hand thanks to the separated representation that the PGD uses, full 3D solutions in plate domains could be achieved by separating the in-plane and out-of-plane (thickness) coordinates. Thus full 3D solutions could be computed by keeping a 2D computational complexity.

Thanks to this novel approach we solved models never until now solved, without the necessity of using powerful computing platforms. Many solutions can be achieved by using Matlab and a laptop, and even light computing devices, as for example smartphones.

Model reduction strategies open new promises in the field of simulation of composites manufacturing processes. It is too early to define the possibilities and the limitations of such an approach. In any case, the first balance seems very positive and it encourages further developments.

The comparison between numerical predictions and the experimental industrial data is a work in progress.

6. References

- Ammar A., Mokdad B., Chinesta F., Keunings R., "A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids", *J. Non-Newtonian Fluid Mech.*, vol. 139, 2006, p. 153-176.
- Ammar A., Mokdad B., Chinesta F., Keunings R., "A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. Part II : transient simulation using space-time separated representations", *J. Non-Newtonian Fluid Mech.*, vol. 144, 2007, p. 98-121.
- Chinesta F., Ammar A., Cueto E., "Recent advances in the use of the Proper Generalized Decomposition for solving multidimensional models", *Archives of Computational Methods in Engineering*, vol. 17, no. 4, 2010, p. 327-350.
- Gonzalez D., Ammar A., Chinesta F., Cueto E., "Recent advances in the use of separated representations", *International Journal for Numerical Methods in Engineering*, In press.
- Ladeveze P., *Nonlinear computational structural mechanics*, Springer, NY, 1999.