Finite Elements in Analysis and Design 100 (2015) 77–90

Contents lists available at ScienceDirect

Finite Elements in Analysis and Design

journal homepage: www.elsevier.com/locate/finel

Isogeometric local *h*-refinement strategy based on multigrids

Alexandre Chemin [a,*], Thomas Elguedj [a], Anthony Gravouil [a,b]

[a] Université de Lyon, INSA-Lyon, LaMCoS, CNRS UMR 5259, F-69621 Villeurbanne, France
[b] Institut Universitaire de France, France

ARTICLE INFO

Article history:
Received 13 October 2014
Received in revised form
4 February 2015
Accepted 19 February 2015

Keywords:
Isogeometric analysis
Local *h*-refinement
Controlled accuracy
Full-multigrid method

ABSTRACT

This paper presents an isogeometric local *h*-refinement algorithm based on localized multigrid resolution dedicated to computational mechanics. This algorithm leads to a solution on a quasi-optimal refined mesh initially unknown for a given precision criterion. Moreover, it allows us to circumvent the obstacle of refinement of non-straight geometric boundaries existing in full multigrid algorithms with isoparametric finite element analysis.

1. Introduction

Multigrid algorithms are iterative solvers showing interesting rates of convergence [1–3]. They were originally introduced to solve fluid mechanics problems using finite differences by Brandt and Livne [4] and were extended to solve solid mechanics problems with finite element analysis by Parsons and Hall [5,6]. Multigrid resolution has already been used with IGA as a way to reduce computational time by Gahalaut et al. [7]. This kind of resolution is designed for problems where the fine discretization is known a priori.

There has been in the recent years quite an extensive work on designing efficient solvers and preconditioners that are well suited to isogeometric analysis (IGA). Up to our knowledge the first work on specific isogeometric preconditioners is attributed to Beirão da Veiga et al. [8] and Buffa et al. [9]. Multigrid (MG) and multilevel methods were also coupled with IGA to design efficient iterative solvers, see for example the work of Gahalaut et al. [7,10] and Donatelli et al. [11]. Finally one can also cite the work of Collier et al. [12] on the performance of standard iterative solvers for IGA. It is important to note that in the work presented in this paper, our goal is not to design efficient iterative solvers for IGA based on multigrid methods. We suggest that multigrid methods can be used to obtain a new local *h*-refinement strategy for IGA with controlled accuracy of the solution. Consequently, the proposed work falls into the class of isogemetric methods with local *h*-refinement such as T-splines [13,14], LR splines [15] and hierarchical b-splines [16,17].

The use of the full multigrid algorithm (FMG) with finite element analysis (FEA) combined with an error indicator has been developed as a solver with adaptative refinement. This algorithm is very efficient but presents a loss of accuracy when the refinement of non-straight boundaries is required as shown in Adams and Taylor [3] and Biboulet et al. [18]. One way to deal with this difficulty is to go back to the CAD geometry each time a refinement is needed. We propose here a way to circumvent this problem using IsoGeometric Analysis. Indeed, IGA allows an exact description of the CAD geometry even with non-straight boundaries [19]. Refinement of such meshes can be done keeping the exact description of the CAD geometry. In this paper, the MG resolution is used as an automatic local mesh refinement tool with controlled accuracy. Here the desired fine mesh is not known a priori and depends on the required level of accuracy on the solution.

This paper is organized as follows. First a short introduction to MG techniques with FEA and a short introduction to IGA with NURBS are presented. Then, we present a localized FMG with IGA algorithm. Finally, the performances of this algorithm are analyzed on several examples.

1.1. Multigrid methods applied to finite element analysis

1.1.1. Principle of multigrid resolution

MG methods are mainly based on the smoothing properties of iterative solvers such as Gauss–Seidel or preconditioned conjugate gradients. In other words, MG methods are based on

* Corresponding author.
E-mail addresses: alexandre.chemin@insa-lyon.fr (A. Chemin), thomas.elguedj@insa-lyon.fr (T. Elguedj), anthony.gravouil@insa-lyon.fr (A. Gravouil).

http://dx.doi.org/10.1016/j.finel.2015.02.007
0168-874X/

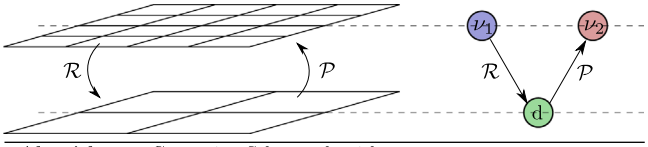
the ability of these solvers to capture the high frequency error in the solution (see [5,6,20]). One iteration of the chosen iterative solver is called a smoothing step. The first idea is to exploit this property using two grids, in the well-known correction scheme. These two nested meshes of the same geometry cover the whole considered domain which is therefore a global MG approach. Quantities on the fine and coarse grid are noted respectively by the f and c subscript. $\mathbf{K}$ is the stiffness matrix, $\mathbf{F}$ are the generalized forces and $\mathbf{U}$ is the discretized displacement field we are looking for using a FE discretization of a given linear solid mechanics problem.

Here, the aim of the CS is to solve the equilibrium equation arising from a static linear elastic analysis on the fine grid $\mathbf{K}_f\mathbf{U}_f=\mathbf{F}_f$.

1. The first step is to do ν_1 smoothing steps on the fine grid with an iterative solver initialized with an initial guess $\mathbf{U}_f^0$, obtaining a solution $\overline{\mathbf{U}}_f^0$.
2. Assuming that the iterative solver decreases the high frequency error, then the difference $\mathbf{E}_f^0$ between $\overline{\mathbf{U}}_f^0$ and the exact solution we are looking for $\mathbf{U}_f$ ($\mathbf{E}_f^0=\mathbf{U}_f-\overline{\mathbf{U}}_f^0$) is only composed by low frequency error. Consequently $\mathbf{E}_f^0$ can be computed on the coarse grid.
3. Therefore, the residual $\mathbf{R}_f^0=\mathbf{F}_f-\mathbf{K}_f\overline{\mathbf{U}}_f^0$ is transferred on the coarse grid with a restriction operator $\mathcal{R}$, and the low frequency error is computed with a direct solve on the coarse grid $\mathbf{K}_c\mathbf{E}_c^0=\mathcal{R}(\mathbf{R}_f^0)$.
4. Then, the displacement coarse correction is transferred on the fine grid with a prolongation operator $\mathcal{P}$, and the solution on the fine grid is corrected as $\widehat{\mathbf{U}}_f^0=\overline{\mathbf{U}}_f+\mathcal{P}(\mathbf{E}_c^0)$.
5. Finally, ν_2 smoothing steps are done on $\widehat{\mathbf{U}}_f^0$ to correct the high frequency error introduced by the prolongation operator giving the solution $\mathbf{U}_f^1$.

These operations form a two grids MG cycle. The two grids MG cycle is repeated until convergence of the solution is reached. The algorithm is presented in Fig. 1.

This algorithm can be generalized for N grids. Indeed, if the direct solve on the coarse grid is too expensive, the CS can be applied recursively with a limited number of MG cycles γ. Fig. 2 shows a MG cycle with three grids for $\gamma=1$ (V-cycle) and $\gamma=2$ (W-cycle). Applied to N grids, MG solvers can be quasi-optimal and reach a complexity in $\mathcal{O}(n)$ (where $n=\text{size}(K_f)$).



Algorithm 1: Correction Scheme algorithm

$k \leftarrow 0$; $\mathbf{U}_f^0 \leftarrow \mathbf{0}$;
while $res > \epsilon$ **do**
  $\overline{\mathbf{U}}_f^k \leftarrow smooth(\mathbf{K}_f, \mathbf{U}_f^k, \mathbf{F}_f, \nu_1)$;
  $\mathbf{R}_f^k \leftarrow \mathbf{F}_f - \mathbf{K}_f\overline{\mathbf{U}}_f^k$;
  $\mathbf{E}_c^k \leftarrow solve(\mathbf{K}_c\mathbf{E}_c^k = \mathcal{R}(\mathbf{R}_f^k))$;
  $\widehat{\mathbf{U}}_f^k \leftarrow \overline{\mathbf{U}}_f^k + \mathcal{P}(\mathbf{E}_c^k)$;
  $\mathbf{U}_f^{k+1} \leftarrow smooth(\mathbf{K}_f, \widehat{\mathbf{U}}_f^k, \mathbf{F}_f, \nu_2)$;
  $res \leftarrow criterion(\mathbf{K}_f, \mathbf{U}_f^{k+1}, \mathbf{F}_f)$;
  $k \leftarrow k+1$;
$\mathbf{U}_f \leftarrow \mathbf{U}_f^k$;

Fig. 1. Correction Scheme algorithm.

In practice, the convergence rate of a MG resolution can depend on the quality of the initial guess $\mathbf{U}_f^0$. A way to avoid such a dependency is the FMG resolution. For a two grid resolution with the CS, the idea consists in using as an initial guess on the fine grid $\mathbf{U}_f^0$ as the prolongation of the solution computed on the coarse grid $\mathbf{U}_f^0=\mathcal{P}(\mathbf{U}_c)$ with $\mathbf{K}_c\mathbf{U}_c=\mathbf{F}_c$. The generalization to N grids is represented in Fig. 3. Solutions boxed in Fig. 3 represent the converged solutions on a given level of discretization.

1.1.2. Definition of prolongation and restriction operators

In the previous part prolongation and restriction operators $\mathcal{P}$ and $\mathcal{R}$ are mentioned. There are some constraints on these operators dealing with primal quantities ($\mathbf{U}$) and dual quantities ($\mathbf{R}$). We can see in Fig. 4 a representation of the relation between these quantities in the two grids case.

A condition we can impose to these operators is to preserve internal work for any couple $(\mathbf{U}_c, \mathbf{F}_f)$

$$\begin{aligned}\langle \mathbf{U}_c, \mathbf{F}_c\rangle = \langle \mathbf{U}_f, \mathbf{F}_f\rangle, \quad & \forall (\mathbf{U}_c, \mathbf{F}_f)\\ \Leftrightarrow \mathbf{U}_c^T\mathcal{R}\mathbf{F}_f = \mathbf{U}_c^T\mathcal{P}^T\mathbf{F}_f, \quad & \forall (\mathbf{U}_c, \mathbf{F}_f)\\ \Leftrightarrow \mathcal{R} = \mathcal{P}^T & \end{aligned} \tag{1}$$

The conservation condition on the internal work gives us a relation between the two operators $\mathcal{P}$ and $\mathcal{R}$. For a given prolongation operator $\mathcal{P}$ we have no choice on the restriction operator if we want this condition to be verified (Eq. (1)).

It is possible to impose another kind of condition: for all $\mathbf{U}_c$, a prolongation, a solve on the fine grid, a restriction and a solve on the coarse grid have to keep $\mathbf{U}_c$ unchanged

$$\begin{aligned}\mathbf{K}_c\mathbf{U}_c = \mathbf{F}_c, \quad & \forall \mathbf{U}_c\\ \Leftrightarrow \mathbf{K}_c\mathbf{U}_c = \mathcal{R}\mathbf{K}_f\mathcal{P}\mathbf{U}_c, \quad & \forall \mathbf{U}_c\\ \Leftrightarrow \mathbf{K}_c = \mathcal{R}\mathbf{K}_f\mathcal{P} & \end{aligned} \tag{2}$$

It gives a condition on stiffness matrices of the two grids (Eq. (2)). MG resolution using this approach are algebraic MG resolutions (see [21]). We are not going to use that kind of approach here because it implies a new computation of all stiffness matrices each time a new grid is added which can become expensive.

Usually in MG resolution with FEA, interpolation operators are used as prolongation operators. We will see further in this paper that this solution is not pertinent in MG resolution with IGA.

1.1.3. Adaptative local h-refinement based on multigrids

The FMG solver presented in Section 1.1.1 is suited for problems with a known final mesh. It is possible to build an algorithm with automatic refinement using an error indicator with an FMG resolution. In this case only the coarse discretization is known a priori and the successive grids are built locally with information from the error indicator (see Fig. 5).

This algorithm uses several levels of discretization, but all the meshes are defined on the whole structure. It implies that quantities of interest (displacement in the linear elastic case) are computed on areas with identical discretization on several meshes. An example with three levels of discretization is displayed in Fig. 6. With a MG algorithm with global refinement, quantities of interest are computed on blue areas which have the same discretization on the three grids and on red areas which have the same discretization on the last two grids. It is possible to save computational cost by avoiding the calculation on blue and red areas for each grid. It is the principle of localization in MG algorithms.

The idea of localization is to refine only the area where the level of accuracy is not satisfied (see Fig. 7).