IGA in Plasma Physics and Electromagnetism

A. RATNANI,[a] E. SONNENDRUCKER,[b] AND N.CROUSEILLES[a]

[a]INRIA-Nancy Grand Est, Projet CALVI

[b]IRMA-Université de Strasbourg et INRIA-Nancy Grand Est, Projet CALVI

ABSTRACT

We present in this work, various applications of the IGA in Plasma Physics and Electromagnetism. We also introduce the new Fast-IGA approach, that enables to reduce considerably the time spent in solving linear systems, involved in IGA, for a class of problems.

IsoGeometric Analysis

The underlying idea behind the IsoGeometric Analysis (IGA), developed recently by Hughes [8], is to use the functions (*B-splines/NURBS*) that describe the geometry in order to approach the numerical solution of *pdes*.
Compared to the standard FEM, IGA posses many advantages:

- the geometry description remains the same while refining. This reduces the communication between the CAD and FEM modules,
- reduces the number of degrees of freedom,
- as based on tensor products, refinement is very cheap,
- IGA provides smoother solutions.

Another important feature of IGA, is that can be used in either its isoparametric version (with the approximation power of curves, using *B-splines/NURBS*), or in its p-FEM version by using *splines* of higher order.

B-splines

Let $T = (t_i)_{1\leqslant i\leqslant N+k}$ be a non-decreasing sequence of knots.
The j-th B-Spline of order k is defined by the recurrence relation:

$$N_j^k = w_j^k N_j^{k-1} + (1 - w_{j+1}^k) N_{j+1}^{k-1} \quad \text{where} \quad w_j^k(x) = \frac{x - t_j}{t_{j+k-1} - t_j} \qquad N_j^1(x) = \chi_{[t_j, t_{j+1}[}(x).$$

We note some important properties of a B-splines basis:

- B-splines are piecewise polynomial of degree $p = k - 1$
- Positivity
- Compact support; the support of N_j^k is contained in $[t_j, .., t_{j+k}]$
- Partition of unity : $\sum_{i=1}^{N} N_i^k(x) = 1, \forall x \in \mathbb{R}$
- Local linear independence
- If a knot t has a multiplicity m then the B-spline is $\mathcal{C}^{(p-m)}$ at t

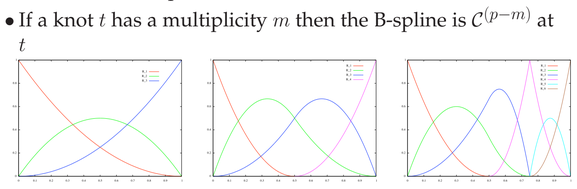

Figure 1: Illustration of h-refinement with $p = 2$, $T = \{000, 111\}$, $T = \{000, \frac{1}{2}, 111\}$ and $T = \{000, \frac{1}{2}, \frac{33}{44}, 111\}$.

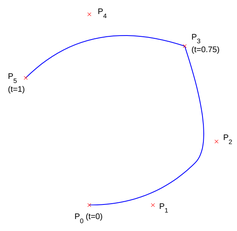

Figure 2: A B-spline curve and its control points, $T = \{000\ \frac{1}{2}\ \frac{33}{44}\ 111\}$

NURBS

Let $\omega = (\omega_i)_{1\leqslant i\leqslant N}$ be a sequence of non-negative reals. The NURBS functions are defined by a projective transformation:
The i-th NURBS of order k associated to the knot vector T and the weights ω, is defined by

$$R_i^k = \frac{\omega_i N_i^k}{\sum_{j=1}^{N} \omega_j N_j^k}.$$

Using *NURBS* curves, **we can exactly model all conics**. For example, a circle can be defined with 9 control points.

Maxwell time domain

We developed in [2] a new formulation of the exact sequence of Finite Element spaces based on splines, introduced by Buffa et al. [1], having the same properties as the Whitney Finite Element spaces traditionally used for the Finite Element solution of Maxwell's equations. As with the Whitney elements, one of Ampere's or Faraday's law can be discretized with a relation between the spline coefficients of the electric and magnetic fields independent of the topology of the mesh. The metric comes in through a discrete Finite Element Hodge operator which appears as the mass matrix involved in the other equation. This method allows us to inverse only one matrix at each time step.
using a **H-div** formulation, the following function spaces are involved,

$$\begin{array}{ccccc} & \mathbf{curl} & & \mathrm{div} & \\ H^1(\mathcal{P}) & \longrightarrow & H(\mathrm{div}, \mathcal{P}) & \longrightarrow & L^2(\mathcal{P}) \\ \cup & & \cup & & \cup \\ V & \longrightarrow & W_{div} & \longrightarrow & X \\ = & & = & & = \\ \mathcal{S}_{\alpha,\alpha}^{p,p} & \longrightarrow & \mathcal{S}_{\alpha,\alpha-1}^{p,p-1} \times \mathcal{S}_{\alpha-1,\alpha}^{p-1,p} & \longrightarrow & \mathcal{S}_{\alpha-1,\alpha-1}^{p-1,p-1} \end{array} \tag{1}$$

The key point of our method is the use of the recursion formula for the derivatives:

$$N_i^{k\prime}(t) = D_i^k(t) - D_{i+1}^k(t), \qquad \text{where} \quad D_i^k = k\frac{N_i^{k-1}(t)}{t_{i+k} - t_i} \tag{2}$$

which leads to the finite spaces,

$$V = \mathbf{span}\left\{N_i^p(\xi) N_j^p(\eta)\right\},\ W_{div} = \mathbf{span}\left\{\begin{pmatrix} N_i^p(\xi) D_j^p(\eta) \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ D_i^p(\xi) N_j^p(\eta) \end{pmatrix}\right\},\ X = \mathbf{span}\left\{D_i^p(\xi) D_j^p(\eta)\right\}. \tag{3}$$

this leads to the linear system, with R the incidence matrix, which does only depend on topology and not on geometry,

$$\begin{cases} \dot{\mathbf{e}} = Rh \\ M_V \dot{h} = -K^T \mathbf{e} \end{cases} \tag{4}$$

For time discretization, we used a classical and 4th order Leap Frog scheme. Next, we show the CFL numbers in function of the splines degree. The expected theoretical convergence order are achieved in this case.

	$LF2^{Th}$	$LF2^{num}$	$LF4^{Th}$	$LF4^{num}$
p = 2	0.3044	0.3056	0.8676	0.8720
p = 3	0.2058	0.1840	0.5866	0.5872
p = 4	0.1496	0.1520	0.4265	0.4272
p = 5	0.1151	0.1168	0.3281	0.3280

Table 1: CFL numbers (theoretical and numerical values), for splines of degree $p = 2, \cdots, 5$

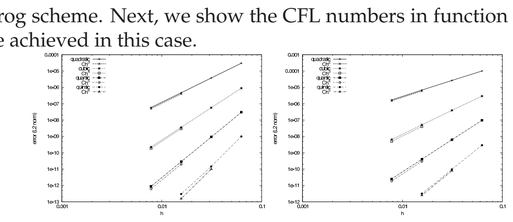

Figure 3: Square test: the L^2 norm error for (top) the magnetic field, (bottom) the electric field

Resistive MHD

We studied the current hole problem:

$$\begin{cases} \partial_t \psi = [\psi, \phi] + \eta\,(J - J_c)\,, \\ \partial_t \omega = [\omega, \phi] + [\psi, J] + \nu \Delta\omega, \\ \quad J = \Delta\psi, \\ \Delta\phi = \omega. \end{cases}$$

Using a semi-implicit time scheme, the discretization of the incompressible *MHD* can be written as:

$$\begin{aligned} A_\nu^0[\omega^{n+1}] &= M^0[\omega^n] + \Delta t \mathcal{C}_{\mathcal{V}_h^0}[J^n, \psi^n] + \Delta t \mathcal{C}_{\mathcal{V}_h^0}[\phi^n, \omega^n], \\ S^0[\phi^{n+1}] &= -M^0[\omega^{n+1}], \\ A_\eta^0[\psi^{n+1}] &= M^0[\psi^n] + \Delta t \mathcal{C}_{\mathcal{V}_h^0}[\psi^n, \phi^{n+1}] - \eta \Delta t M^0[J_c], \\ M^0[J^{n+1}] &= -S^0[\psi^{n+1}]. \end{aligned}$$

The following results were obtained using quadratic *NURBS*, a grid of 64×64 meshes and a time step $dt = 0.1$.

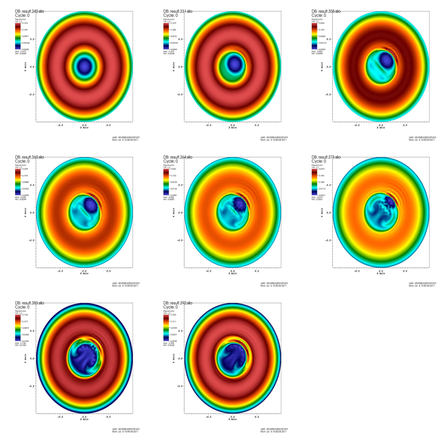

Figure 4: Evolution of the current density as a function of time, using $\nu = 10^{-6}$ and $\eta = 10^{-5}$.

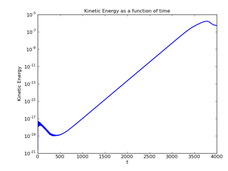

Figure 5: Kinetic energy of the plasma for $\nu = 10^{-6}$ and $\eta = 10^{-5}$.

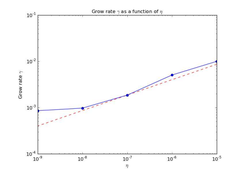

Figure 6: The grow rate γ as a function of the resistivity η. The dashed line indicates the $\eta^{\frac{1}{3}}$ asymptote.

Quasi-Neutrality equation

Linearization around the equilibrium density n_0 together with the approximation $B \approx B_0$ are usually performed; these simplifications provide only a radial dependence for the anisotropic factor

$$-\frac{1}{B_0^2}\nabla_\perp \cdot (n_0 \nabla_\perp \phi) + \frac{n_0}{T_e}(\phi - \langle\phi\rangle) = \bar{n}_i - n_0. \tag{5}$$

Test case : Chaotic solution We test our solver on a chaotic solution on a polar coordinate Laplacian $-\nabla^2\phi = n$. This analytic solution takes the form

$$\phi(r, \theta) = \left[\sin(2\pi\xi) + \epsilon \sum_M B_M \sin(2\pi M \xi)\right] \sum_l A_l \cos(l\theta + \Theta_l), \tag{6}$$

where $0 \leq \epsilon \leq 1$, $\xi = \frac{r - r_{\min}}{r_{\max} - r_{\min}}$, A_l, B_M and $\frac{\Theta_l}{2\pi}$ are random numbers which range between 0 and 1, with $|M|, |l| \leq 40$.

Spline degree	dof	L^2 error norm
2	17408	$7.20\ 10^{-3}$
3	18060	$1.07\ 10^{-3}$
4	18720	$1.71\ 10^{-4}$
5	19388	$2.84\ 10^{-5}$
6	20064	$4.81\ 10^{-6}$
7	20748	$1.67\ 10^{-6}$

Table 2: *Chaotic solution*: Number of degree of freedom and L^2 norm of the error for each spline degree. Used parameters are $\epsilon = 0.4$, $r_{\min} = 0.2$ and $r_{\max} = 0.4$

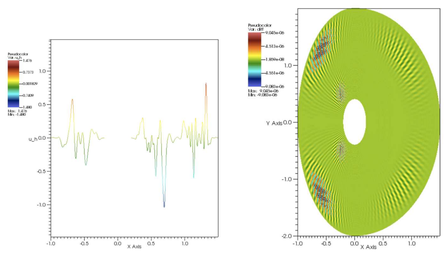

Figure 7: (left) the numerical solution, with $l = m = 10$ and $p = 4$, (right) the difference $u - u_h$ for a turbulence test in the tokamak model, with $l = m = 2$ and $p = 4$.

The Fast-IGA approach

We consider the classical linear system obtained after discretization $MU = F$. The next strategy works for evolution problem's, elliptic equations (under some assumptions), it can be extended to more general problems (but still under some restrictive assumptions).

The case of annular domains

We consider here, annular domain $\Omega \subset \mathbb{R}^2$, defined between an *interior* curve $\mathcal{C}^i$ and an *exterior* curve $\mathcal{C}^e$, where both $\mathcal{C}^e$ and $\mathcal{C}^i$ must be periodic, and defined with the same uniform knot sequence (*c.f.* figure 8). We consider the following mapping $\mathbf{F}(\xi, \eta) = (1 - \xi)\mathcal{C}^i(\eta) + \xi\mathcal{C}^e(\eta)$, that maps a patch onto Ω. Therefore, the jacobian can be written in the form:

$$\det(J_{\mathbf{F}}) = u_1(\xi)v_1(\eta) + u_2(\xi)v_2(\eta)$$

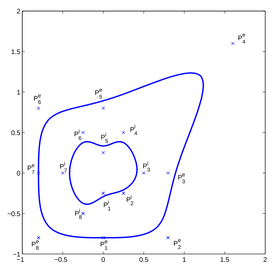

Figure 8: The *interior* and *exterior* curves are defined using the same knot vector $T = \{0\ 1\ \cdots\ 13\ 14\}$, the *interior* curve is defined using the control polygon $P_8^iP_1^iP_2^iP_3^iP_4^iP_5^iP_6^iP_7^iP_8^iP_1^iP_2^i$, the *exterior* curve is defined using the control polygon $P_8^eP_1^eP_2^eP_3^eP_4^eP_5^eP_6^eP_7^eP_8^eP_1^eP_2^e$.

In this case, the mass matrix can be written as a Kronecker tensor product, $M = M_\xi^1 \otimes M_\eta^1 + M_\xi^2 \otimes M_\eta^2$. Thus a linear system $MU = F$, can be solved as

$$M_\xi^1 \mathbf{vec}^{-1} U M_\eta^1 + M_\xi^2 \mathbf{vec}^{-1} U M_\eta^2 = \mathbf{vec}^{-1} F \tag{7}$$

where both M_η^1 and M_η^2 are circulant and can be diagonalized in the same basis (Fourier modes). To solve eq (8), one needs to solve the N^η linear $1D$ linear systems : $\left(\lambda_j^1 M_\xi^1 + \lambda_j^2 M_\xi^2\right)\widehat{U}_j = \widehat{F}_j$, where $M_\eta^1 = P\Lambda_1 P^\star$ and $M_\eta^2 = P\Lambda_2 P^\star$, with $\Lambda_1 = \mathrm{diag}\{\lambda_j^1\}$ and $\Lambda_2 = \mathrm{diag}\{\lambda_j^2\}$. The cost of this method is $N\log(N)$.

	Initializing		Solving	
p	*FIGA*	*SPLU*	*FIGA*	*SPLU*
1	0.021	3.38	0.074	0.067
2	0.043	31.40	0.076	0.967
3	0.052	197.31	0.075	3.505
4	0.060	330.28	0.075	16.070
5	0.069	415.63	0.077	32.852

Figure 9: CPU-time, in seconds, spent in solving (left) and initializing (right) the linear system, using the Fast IGA, compared to SuperLU. Test done on a grid 256×256

Current Research and perspectives

PyIGA Library

PyIGA is a library written in Fortran and Python. Using *PyIGA*, you can solve scalar or vectorial (system of) *pde's*,

- assemble differential operators,
- Input (XML files) and Output (SILO/HDF5 files), possible use of Xdmf,
- use existing *Python*-modules,
- minimum communication between the user and the library.

Current Research

- Semi-Lagrangian schemes using *NURBS* to solve the Vlasov-equation, [4]
- Coupling PIC method using IGA, [5]
- Numerical results for the Fast-IGA, and extend it to the $3D$ case.

References

[1] A. Buffa, G. Sangall, R. Vazquez, Isogeometric analysis in electromagnetics: B-splines approximation, *Comput. Methods Appl. Mech. Engrg - 2009.*
[2] A.Ratnani, E.Sonnendrucker, Arbitrary High-Order Spline Finite Element Solver for the Time Domain Maxwell equations, Accepted in *Journal of Scientific Computing.*
[3] A.Ratnani. E. Sonnendrucker, N. Crouseilles, An Isogeometric Analysis Approach for the study of the gyrokinetic quasi-neutrality equation, Submitted.
[4] A.Ratnani. J. Abiteboul, G. Latu, V. Grandgirard, E. Sonnendrucker, A. Strugarek, Solving the Vlasov equations in complex geometries, Submitted.
[5] A.Ratnani. A. Back, A. Crestetto, E. Sonnendrucker, An Axisymmetric PIC code based on Isogeometric Analysis, Submitted.
[6] A.Ratnani, The Fast-IGA approach, In preparation.
[7] A.Ratnani, PyIGA, Isogeometric Analysis using Python, In preparation.
[8] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Computer Methods in Applied Mechanics and Engineering, Volume 194, number 39 - 41, pages 4135 - 4195, (2005).*