Isogeometric analysis in electronic structure calculations

Robert Cimrman[a,*], Matyáš Novák[b], Radek Kolman[c], Miroslav Tůma[d], Jiří Vackář[b]

[a]*New Technologies Research Centre, University of West Bohemia, Univerzitní 8, 306 14 Plzeň, Czech Republic*
[b]*Institute of Physics, Academy of Sciences of the Czech Republic, Na Slovance 1999/2, Prague, Czech Republic*
[c]*Institute of Thermomechanics, Academy of Sciences of the Czech Republic, Dolejškova 5, 182 00 Prague, Czech Republic*
[d]*Institute of Computer Science, Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 2, 182 07, Prague, Czech Republic*

Abstract

In electronic structure calculations, various material properties can be obtained by means of computing the total energy of a system as well as derivatives of the total energy w.r.t. atomic positions. The derivatives, also known as Hellman-Feynman forces, require, because of practical computational reasons, the discretized charge density and wave functions having continuous second derivatives in the whole solution domain. We describe an application of isogeometric analysis (IGA), a spline modification of finite element method (FEM), to achieve the required continuity. The novelty of our approach is in employing the technique of Bézier extraction to add the IGA capabilities to our FEM based code for ab-initio calculations of electronic states of non-periodic systems within the density-functional framework, built upon the open source finite element package SfePy. We compare FEM and IGA in benchmark problems and several numerical results are presented.

Keywords: electronic structure calculation, density functional theory, finite element method, isogeometric analysis

---

*Corresponding author
Email address:* `cimrman3@ntc.zcu.cz` (Robert Cimrman)

1. Introduction

The electronic structure calculations allow to predict and understand material properties such as stable atomic arrangements by minimizing the total internal energy of a system of atoms, as well as to determine derived properties such as elasticity, hardness, electric and magnetic properties, etc.

We are developing a real space code [29] for electronic structure calculations based on

- the density functional theory (DFT), [10, 22, 18, 23];

- the environment-reflecting pseudopotentials [28];

- a weak solution of the Kohn-Sham equations [15].

The code is based on the open source finite element package SfePy (Simple Finite Elements in Python, http://sfepy.org) [4], which is a general package for solving (systems of) partial differential equations (PDEs) by the finite element method (FEM), cf. [11, 27].

The key required ability for practical computations is the calculation of the *Hellman-Feynman forces* (HFF), which correspond to the derivatives of the total energy w.r.t. atomic positions. The HFF can efficiently provide gradients in a gradient-based optimizer searching the total energy minimum. A major hurdle to overcome in computing the HFF is the requirement that the discretized charge density and wave functions should have continuous second derivatives in the whole solution domain — implementing a globally C^2 continuous basis in FEM is not easy. Therefore we decided to enhance the SfePy package with another PDE discretization scheme, the *Isogeometric analysis* (IGA), see [6, 2].

IGA is a modification of FEM which employs shape functions of different spline types such as B-splines, NURBS (Non-uniform rational B-spline), T-splines [1], etc. It was successfully employed for numerical solutions of various physical and mathematical problems, such as fluid dynamics, diffusion and other problems of continuum mechanics [6].

IGA has been reported to have excellent convergence properties when solving eigenvalue problems connected to free vibrations in elasticity [17], where the errors in frequencies decrease in the whole frequency band with increasing the approximation order, so even for high frequencies the accuracy is very good. It should be noted that dispersion and frequency errors are reported to decrease with increasing spline order [13]. Moreover, IGA solution excellently approximates not only eigenvalues in the full frequency spectrum but also accurately approximates eigen-modes. There are no optical modes as in higher-order FEM, where the errors in higher frequencies grow rapidly with the approximation order and band gaps in the frequency band exist, see [7, 16, 12]. The Kohn-Sham equations are a highly non-linear eigenvalue problem so the above properties of IGA seem to further support our choice.

The drawbacks of using IGA, as reported also in [17], concern mainly the increased computational cost of the numerical integration and assembling. Also, because of the higher global continuity, the assembled matrices have more nonzero entries than the matrices corresponding to the C^0 FEM basis. A comparison study of IGA and FEM matrix structures, the cost of their evaluation, and mainly the cost of direct and iterative solvers in IGA has been presented by [5] and [25].

Recently, using FEM and its variants in electronic structure calculation context is pursued by a growing number of groups, cf. [8], where the hp-adaptivity is discussed, [20, 21] where spectral finite elements as well as the hp-adaptivity are considered, or [19], where NURBS-based FEM is applied.

In the paper we first outline the physical problem of electronic states calculations in Section 2, then introduce the computational methods and their implementation in terms of both FEM and IGA in Section 3. Finally, we present a comparison of FEM and IGA using a benchmark problem and show some numerical results in Section 4.

3

2. Calculation of electronic states

The DFT allows decomposing the many-particle Schrödinger equation into the one-electron Kohn-Sham equations. Using atomic units they can be written in the common form

$$\left(-\frac{1}{2}\nabla^2 + V_{\mathrm{H}}(\boldsymbol{r}) + V_{\mathrm{xc}}(\boldsymbol{r}) + \hat{V}(\boldsymbol{r})\right)\psi_i = \varepsilon_i \psi_i \ , \tag{1}$$

which provide the orbitals ψ_i that reproduce, with the weights of occupations n_i, the charge density ρ of the original interacting system, as

$$\rho(\boldsymbol{r}) = \sum_i^N n_i |\psi_i(\boldsymbol{r})|^2 \ . \tag{2}$$

$\hat{V}$ is a (generally) non-local Hermitian operator representing the effective ionic potential for electrons. In the present case, within pseudopotential approach, $\hat{V}$ represents core electrons, separated from valence electrons, together with the nuclear charge. V_{xc} is the exchange-correlation potential describing the non-coulomb electron-electron interactions. We use local-density aproximation (LDA) of this potential [18]. V_{H} is the electrostatic potential obtained as a solution to the Poisson equation. The Poisson equation for V_{H} has the charge density ρ at its right-hand side and is as follows:

$$\Delta V_{\mathrm{H}} = 4\pi\rho \ . \tag{3}$$

Denoting the total potential $V := V_{\mathrm{H}} + V_{\mathrm{xc}} + \hat{V}$, we can write, using Hartree atomic units,

$$\left(-\frac{1}{2}\nabla^2 + V(\boldsymbol{r})\right)\psi_i = \varepsilon_i \psi_i \ . \tag{4}$$

Note that the above mentioned eigenvalue problem is highly non-linear, as the potential V depends on the orbitals ψ_i. Therefore an iterative scheme is needed, defining the DFT loop for attaining a self-consistent solution.

2.1. DFT loop

For the global convergence of the DFT iteration we use the standard algorithm outlined in Fig. 1. The purpose of the DFT loop is to find a self-consistent solution. Essentially, we are seeking a fixed point of a function of

4

$V_{\mathrm{hxc}} := V_{\mathrm{H}}[\rho] + V_{\mathrm{xc}}[\rho]$. For this task, a variety of nonlinear solvers can be used. We use Broyden-type quasi-Newton solvers applied to

$$DFT(V_{\mathrm{hxc}}^{i}) - V_{\mathrm{hxc}}^{i} = V_{\mathrm{hxc}}^{i+1} - V_{\mathrm{hxc}}^{i} = 0 \; , \tag{5}$$

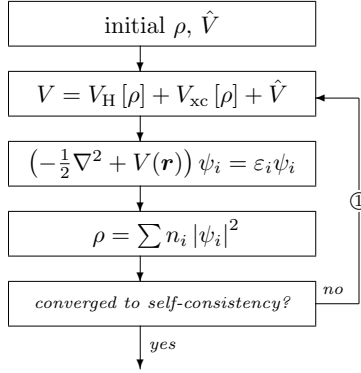where DFT denotes a single iteration of the DFT loop.

Figure 1: DFT, iterative self-consistent scheme.

After the DFT loop convergence is achieved, the derived quantities, particularly the total energy, are computed. By minimizing the total energy as a function of atomic positions, the equilibrium atomic positions can be found. Therefore the DFT loop itself can be embedded into an outer optimization loop, where the objective function gradients are the HFF.

2.2. Forces acting on atoms

The force acting on atom α at position $\boldsymbol{\tau}_{\alpha}$ is equal to the derivative of the total energy functional with respect to an infinitesimal displacement of this atom $\delta\boldsymbol{\tau}_{\alpha}$:

$$\boldsymbol{F}^{\alpha} = -\frac{\delta E}{\delta\boldsymbol{\tau}_{\alpha}} \; . \tag{6}$$

Making use of the Hellmann-Feynman theorem that relates the derivative of the total energy, with respect to a parameter λ, to the expectation value of the derivative of the Hamiltonian operator, w.r.t. the same parameter,

$$\frac{\mathrm{d}E}{\mathrm{d}\lambda} = \left\langle \Psi_{\lambda}^{*} \left| \frac{\mathrm{d}\hat{H}_{\lambda}}{\mathrm{d}\lambda} \right| \Psi_{\lambda} \right\rangle \; , \tag{7}$$

5

within the density functional theory we can write

$$\boldsymbol{F}^{\alpha} = \boldsymbol{F}_{\mathrm{HF,es}}^{\alpha} - \frac{\left(\sum_i n_i \delta\varepsilon_i - \int \rho(\boldsymbol{r})\delta V_{\mathrm{eff}}(\boldsymbol{r})d^3\boldsymbol{r}\right)}{\delta\boldsymbol{\tau}_\alpha} \ , \tag{8}$$

where the first term is the electrostatic Hellmann-Feynman force and the second term is the "Pulay" force, also known as "incomplete basis set" force, that contains the corrections that depend on technical details of the calculation. The electrostatic Hellmann-Feynman force is given by the sum over all the atoms $\beta \neq \alpha$ of electrostatic forces between the charges of atomic nuclei Z_α and Z_β and by the force acting on the charge Z_α in the electric field of the charge density ρ:

$$\boldsymbol{F}_{\mathrm{HF,es}}^{\alpha} = Z_\alpha \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\tau}_\alpha} \left(-\sum_{\beta\neq\alpha} \frac{Z_\beta}{|\boldsymbol{\tau}_\alpha - \boldsymbol{\tau}_\beta|} + \int \frac{\rho(\boldsymbol{r})}{|\boldsymbol{\tau}_\alpha - \boldsymbol{r}|} d^3\boldsymbol{r} \right) \ . \tag{9}$$

The effective potential

$$V_{\mathrm{eff}} = -\sum_\alpha \frac{Z_\alpha}{|\boldsymbol{r} - \boldsymbol{\tau}_\alpha|} + V_{\mathrm{hxc}}(\boldsymbol{r}) \tag{10}$$

used in the Pulay force term, together with the kinetic energy operator, forms the total energy (for more details see e.g. [31], [30], [9]).

Within the pseudopotential formalism, as it was shown by Ihm, Zunger and Cohen [14], the electrostatic HF force (9) transforms into

$$-\sum_{\beta\neq\alpha} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{\tau}_\alpha} \left(\frac{Z_\alpha Z_\beta}{|\boldsymbol{\tau}_\alpha - \boldsymbol{\tau}_\beta|} \right) + \sum_l \int \rho_{\mathrm{ion},l}(|\boldsymbol{r} - \boldsymbol{\tau}_\alpha|)E_l(\boldsymbol{r})d^3\boldsymbol{r} \ , \tag{11}$$

where $\rho_{\mathrm{ion},l}$ is "virtual ionic partial charge" derived from the l-component $U_{\mathrm{ps},l}$ of the semilocal form of the pseudopotential,

$$\rho_{\mathrm{ion},l}(\boldsymbol{r}) = -\frac{\pi}{4}\nabla^2 U_{\mathrm{ps},l}(\boldsymbol{r}) \ , \tag{12}$$

and

$$E_l(\boldsymbol{r}) = -\nabla_{\boldsymbol{r}} \int \frac{\rho_{\mathrm{ps},l}(\boldsymbol{r}')}{|\boldsymbol{r}' - \boldsymbol{r}|} d^3\boldsymbol{r}' \ . \tag{13}$$

Here $\rho_{\mathrm{ps},l}$ is the l-projected (via an integration over angles on a unit sphere) charge w.r.t. atomic center α

$$\rho_{\mathrm{ps},l}(\boldsymbol{r}') = \sum_i n_i \int \psi_i^*(\boldsymbol{r}')\hat{P}_l^\alpha \psi_i(\boldsymbol{r}')d\theta d\phi \ , \tag{14}$$

6

where $\hat{P}_l^\alpha$ is the Legendre-polynomial-based projector to the l-subspace w.r.t. site α.

The continuity of the derivatives of the wave functions ψ_i up to the second order is the necessary condition for the validity of the derivation above; otherwise everything would become much more complicated and unsuitable for practical use, in connection with FEM/IGA approach (for more details see e.g. [31], [30], [9]).

3. Computational methods and their implementation

3.1. Weak formulation

Let us denote $H^1(\Omega)$ the usual Sobolev space of functions with L^2 integrable derivatives and $H_0^1(\Omega) = \{u \in H^1(\Omega) | u = 0 \text{ on } \partial\Omega\}$.

The eigenvalue problem (4) can be rewritten using the weak formulation: find functions $\psi_i \in H^1(\Omega)$ such that for all $v \in H_0^1(\Omega)$ holds

$$\int_\Omega \frac{1}{2} \nabla\psi_i \cdot \nabla v \ \mathrm{d}V + \int_\Omega v V \psi_i \ \mathrm{d}V = \varepsilon_i \int_\Omega v\psi_i \ \mathrm{d}V + \oint_{\partial\Omega} \frac{1}{2} \frac{\mathrm{d}\psi_i}{\mathrm{d}\boldsymbol{n}} \ \mathrm{d}S \ . \qquad (15)$$

If the solution domain Ω is sufficiently large, the last term can be neglected. The Poisson equation (3) has the following weak form:

$$\int_\Omega \nabla v \cdot \nabla V_\mathrm{H} = 4\pi \int_\Omega \rho v \ . \qquad (16)$$

Equations (15), (16) then need to be discretized — the continuous fields are approximated by discrete fields with a finite set of degrees of freedom (DOFs) and a basis, typically piece-wise polynomial:

$$u(\boldsymbol{r}) \approx u^h(\boldsymbol{r}) = \sum_{k=1}^N u_k \phi_k(\boldsymbol{r}) \text{ for } \boldsymbol{r} \in \Omega \ , \qquad (17)$$

where u is a continuous field (ψ, v, V_H in our equations), u_k, $k = 1, 2, \ldots, N$ are the discrete DOFs and ϕ_k are the basis functions. From the computational point of view it is desirable that the basis functions have a small support, so that the resulting system matrix is sparse.

7

3.2. Finite element method

In the FEM the discretization process involves the discretization of the domain Ω — it is replaced by a polygonal domain Ω_h that is covered by small non-overlapping subdomains called *elements* (e.g. triangles or quadrilaterals in 2D, tetrahedrons or hexahedrons in 3D), cf. [11, 27]. The elements form a FE *mesh*.

The basis functions are defined as piece-wise polynomials over the individual elements, have a small support and are typically globally C^0 continuous. The discretized equations are evaluated over the elements as well to obtain local matrices or vectors that are then assembled into a global sparse system. The evaluation usually involves a numerical integration on a reference element, and a mapping to individual physical elements [11, 27]. The nodal basis of Lagrange interpolation polynomials or the hierarchical basis of Lobatto polynomials can be used in our code.

3.3. Isogeometric analysis

In IGA, the CAD geometrical description in terms of NURBS patches is used directly for the approximation of the unknown fields, without the intermediate FE mesh — the meshing step is removed, which is one of its principal advantages. A D-dimensional geometric domain can be defined by

$$\boldsymbol{r}(\underline{\xi}) = \sum_{A=1}^{n} \boldsymbol{P}_A R_{A,p}(\underline{\xi}) = \boldsymbol{P}^T \boldsymbol{R}(\underline{\xi}) \,, \tag{18}$$

where $\underline{\xi} = \{\xi_1, \ldots, \xi_D\}$ are the parametric coordinates, and $\boldsymbol{P} = \{\boldsymbol{P}_A\}_{A=1}^{n}$ is the set of control points, $R_{A,p}$, $A = 1, 2, \ldots n$ are the NURBS solid basis functions and p is the NURBS solid degree.

If $D > 1$, the NURBS solid can be defined as a tensor product of univariate NURBS curves. First a mapping is defined, see [2], that maps between the tensor product space and the global indexing of the basis functions. Let $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$ and $k = 1, 2, \ldots, l$, then

$$\tilde{A}(i, j, k) = (l \times m)(i - 1) + l(j - 1) + k \,.$$

8

If $N_{i,p}(\xi)$, $M_{j,q}(\eta)$ and $L_{k,r}(\zeta)$ are the univariate B-spline basis functions with degrees p, q and r, respectively, then with $A = \tilde{A}(i,j,k)$ and $\hat{A} = \tilde{A}(\hat{i},\hat{j},\hat{k})$

$$R_A^{p,q,r}(\xi,\eta,\zeta) = \frac{L_{i,r}(\zeta)M_{j,q}(\eta)N_{k,p}(\xi)w_A}{\sum_{\hat{i}=1}^n \sum_{\hat{j}=1}^m \sum_{\hat{k}=1}^l L_{\hat{i},r}(\zeta)M_{\hat{j},q}(\eta)N_{\hat{k},p}(\xi)w_{\hat{A}}}$$

are the NURBS solid basis functions (here $R_A^{p,q,r}$ corresponds to $R_{A,p}$ in (18)), and w_A are the weights (products of univariate NURBS basis weights). Below we will denote (ξ,η,ζ) by a vector $\underline{\xi}$. The univariate B-spline basis functions are defined by a *knot vector*, that is the vector of non-decreasing parametric coordinates $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$, where $\xi_A \in \mathbb{R}$ is the Ath knot and p is the polynomial degree of the B-spline basis functions [24]. Then for $p = 0$

$$N_{A,0}(\xi) = \begin{cases} 1 \text{ for } \xi_A \leq \xi < \xi_{A+1} \,, \\ 0 \text{ otherwise.} \end{cases}$$

For $p > 0$ the basis functions are defined by the Cox-de Boor recursion formula (defining $\frac{0}{0} \equiv 0$)

$$N_{A,p}(\xi) = \frac{\xi - \xi_A}{\xi_{A+p} - \xi_A}N_{A,p-1}(\xi) + \frac{\xi_{A+p+1} - \xi}{\xi_{A+p+1} - \xi_{A+1}}N_{A+1,p-1}(\xi) \,.$$

Note that it is possible to insert knots into a knot vector without changing the geometric or parametric properties of the curve by computing the new set of control points by the knot insertion algorithm, see e.g. [2]. The continuity of approximation does not change when inserting control points. The basic properties of the B-spline basis functions can be found in [24].

In IGA, the same NURBS basis, that is used for the geometry description, is used also for the approximation of PDE solutions. For our equation (15) we have

$$\psi(\underline{\xi}) \approx \psi^h(\underline{\xi}) = \sum_{A=1}^n \psi_A R_{A,p}(\underline{\xi}) \,, \quad v(\underline{\xi}) \approx v^h(\underline{\xi}) = \sum_{A=1}^n v_A R_{A,p}(\underline{\xi}) \,, \qquad (19)$$

where ψ_A are the unknown DOFs - coefficients of the basis in the linear combination, and v_A are the test function DOFs.

Complex geometries cannot be described by a single NURBS outlined above, often called *NURBS patch* — many such patches might be needed, and special

9

care must be taken to ensure required continuity along patch boundaries and to avoid holes. Usually, the patches are connected using C^0 continuity only, as the individual patches have the open knot vectors [24].

However, on a single patch, the NURBS basis can be smooth as needed for the HFF calculation — a degree p curve has $p-1$ continuous derivatives, if no internal knots are repeated, as follows from the B-spline basis properties [24]. The basis functions $R_{A,p}$, $A = 1, \ldots, n$ on the patch are uniquely determined by the knot vector for each axis, and cover the whole patch. Due to our continuity requirements, only single-patch domains are considered in this paper. Also, we set $w_A = 1$, $A = 1, \ldots, n$, thus using a B-spline basis instead of full NURBS basis, because our domain is simply a cube, see Section 4.

3.4. IGA implementation in SfePy

Our implementation [3] uses a variant of IGA based on *Bézier extraction operators* [2] that is suitable for inclusion into existing FE codes. The code itself does not see the NURBS description at all. It is based on the observation that repeating a knot in the knot vector decreases continuity of the basis in that knot by one. This can be done in such a way that the overall shape remains the same, but the "elements" appear naturally as given by non-zero knot spans. The final basis restricted to each of the elements is formed by the Bernstein polynomials $\boldsymbol{B}$, cf. [2]. The Bézier extraction process is illustrated in Fig. 2. The depicted basis corresponds to the second parametric axis of the domain shown in Fig. 3, see below.

In [2] algorithms are developed that allow computing the Bézier extraction operator $\boldsymbol{C}$ for each element such that the original (smooth) NURBS basis function $\boldsymbol{R}$ can be recovered from the local Bernstein basis $\boldsymbol{B}$ using $\boldsymbol{R} = \boldsymbol{C}\boldsymbol{B}$. The Bézier extraction also allows construction of the Bézier mesh, see Fig. 2, right and Fig. 3, right. The code then loops over the Bézier elements and assembles local contributions in the usual FE sense. The operator $\boldsymbol{C}$ is a function of the knot vectors only — it does not depend on the positions of control points.

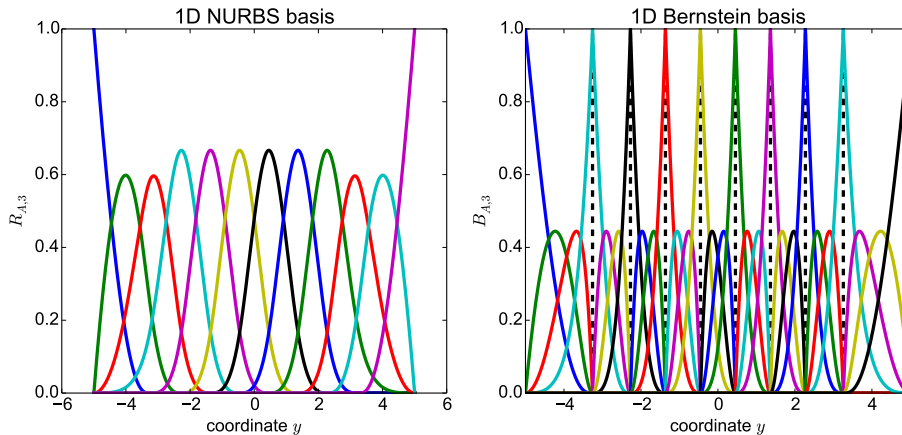Several kinds of grids (or "meshes") can be constructed for a NURBS patch,

Figure 2: Left: NURBS basis of degree 3 that describes the second axis of the parametric mesh in Fig. 3. Right: the corresponding Bernstein basis with Bézier elements delineated by vertical lines.

as depicted in Fig. 3. The parametric mesh is simply the tensor product of the knot vectors defining the parametrization — the lines correspond to the knot vector values. The control mesh has vertices given by the NURBS patch
215 control points and connectivity corresponding to the tensor product nature of the patch. The Bézier mesh has been introduced above and its vertices are the control points of the individual Bézier elements. In our code we use the corner vertices of the Bézier elements to construct a *topological Bézier mesh*, which can be used for subdomain selection (e.g. parts of the boundary, where boundary
220 conditions need to be applied), because its vertices are interpolatory, i.e., they are in the NURBS domain or on its boundary.

In our implementation, full Gauss quadrature rules with the 1D quadrature order $r = p + 1$ are used to integrate over the Bézier elements.

4. Numerical examples

225 In this section we show some results based on our initial tests with the IGA implementation.
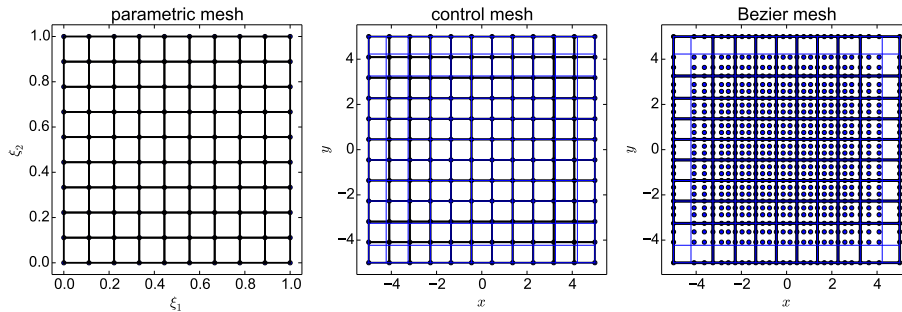
11

Figure 3: From left to right: parametric mesh (tensor product of knot vectors), control mesh, Bézier mesh. The corner vertices of Bézier mesh elements form the topological Bézier mesh. The thin blue lines are iso-lines of the NURBS parametrization.

4.1. Nitrogen atom benchmark

The nitrogen atom serves us as a benchmark problem. A cube domain with the size of $10 \times 10 \times 10$ atomic units was used for all the computations. The FEM approximation used Lagrange polynomial basis of order three (tri-cubic Lagrange polynomials) on a uniform hexahedral mesh. The IGA approximation used degree 3 B-splines and a uniform knot vector in each parametric axis. The control points were also spaced uniformly, so that the placement of the basis in the physical space was not coarser in the middle of the cube, see Fig. 3.

We compared IGA and FEM solution convergence given the increasing number of DOFs (grid size). The number of DOFs corresponds to the sizes of the matrices that come from the FEM- or IGA- discretized (15), formally written as

$$(\boldsymbol{K} + \boldsymbol{V}(\boldsymbol{\psi}_i))\boldsymbol{\psi}_i = \varepsilon_i \boldsymbol{M} \boldsymbol{\psi}_i \ . \tag{20}$$

The following numbers of DOFs per cube side (including DOFs fixed by boundary conditions) were used:

- FEM: 16, 19, 22, 28, 34, 40, 46, 52, 58, 64;

- IGA: 12, 14, 16, 18, 20, 22, 24, 26, 28.

This corresponds to:

12

- FEM: 5, 6, 7, 9, 11, 13, 15, 17, 19, 21 cubic elements per cube side;

- IGA: 9, 11, 13, 15, 17, 19, 21, 23, 25 Bézier elements per cube side.

The grids corresponding to the coarsest IGA approximation used are shown (2D projection) in Fig. 3. Because the difficulty of solving (20) depends not only on the number of DOFs, but also on the number of non-zeros in the sparse matrices, we show the convergence with respect to both of these parameters, see Figs. 4, 5. The non-zeros are determined structurally by the compact support of each basis function and the pattern (allocated space) is the same for both $(\boldsymbol{K} + \boldsymbol{V}(\boldsymbol{\psi}_i))$ and $\boldsymbol{M}$.
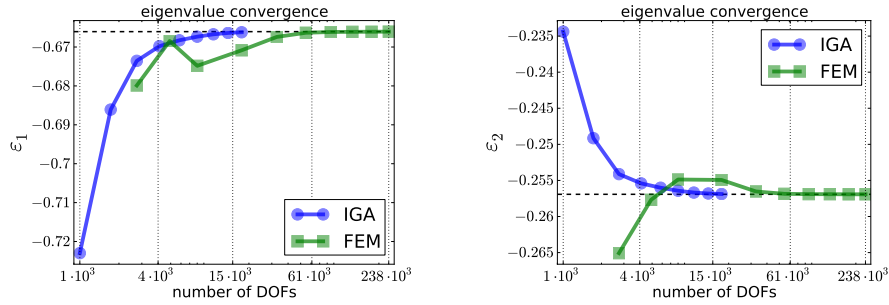
Figure 4: Convergence of eigenvalues ε_1 and ε_2 w.r.t. the number of DOFs.
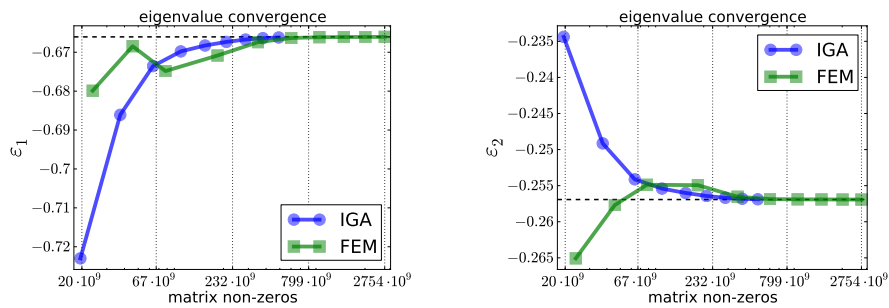
Figure 5: Convergence of eigenvalues ε_1 and ε_2 w.r.t. the number of non-zeros of the matrices.

It should be noted that the converged values are neither exact physical binding energies of electrons nor the ionization energies. They are just the Kohn-

Sham eigenvalues for a given particular problem under given conditions and approximations, in our case with a relatively small physical domain size, since the aim was to test quickly the numerical properties of different bases. However, note that both methods converge to exactly the same values, which confirms the numerical validity of the FEM/IGA calculation.

The above results suggest that IGA converges to a solution both for a much smaller number of DOFs and for smaller number of non-zeros in the matrices. This is in agreement with the initial prognosis that the higher-order smooth basis functions can improve the convergence and accuracy of the finite element electronic structure calculations. Note the non-oscillating convergence of the IGA values in contrast to the oscillating convergence of FEM values.

4.2. Examples of computed quantities

In physical simulations we are interested in other quantities, besides the eigenvalues ε_i. As an example we show the charge density and the orbitals ψ_i of the nitrogen atom in Fig. 6 for some of the grids used in the convergence study above. It shows that the electron states form into shapes of spherical harmonics even if no preliminary shape-anticipating assumption is done. Note that depending on the grid resolution, the orientation of orbitals without spherical symmetry (ψ_2, ψ_3, and ψ_4) changes.

For illustration of a more complex structure, we also present the distribution of the charge density ρ of the tetrafluormethane molecule (CF_4) in Fig. 7. Even though the parametric grid used in this computation was only $20 \times 20 \times 20$, with C^2 continuous B-spline basis, good results were obtained: our code correctly reproduced both the angles of C-F branches and the inter-atomic distances, by minimizing the total energy.

5. Conclusion

We introduced our approach to electronic structure calculations, based on the density functional theory, the environment-reflecting pseudopotentials and
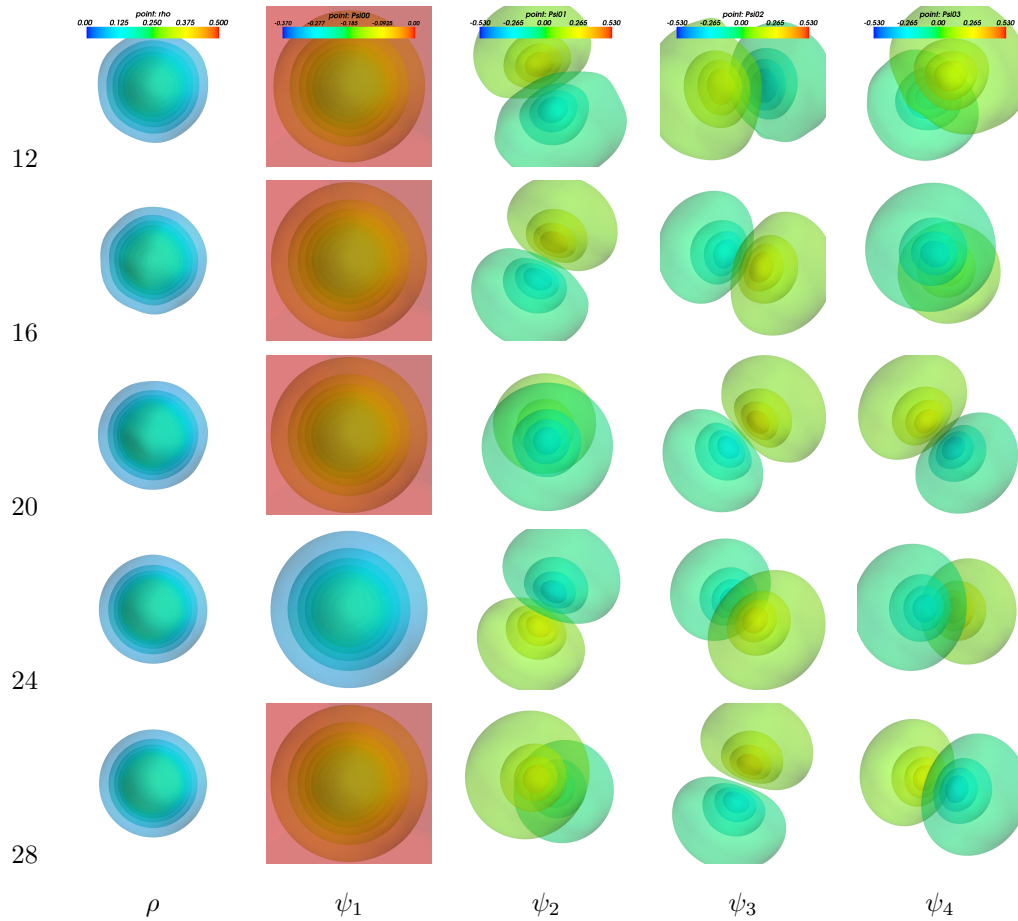
14

Figure 6: Nitrogen atom: iso-surfaces of charge density ρ and orbitals ψ_i for several IGA grid sizes. The number of DOFs per axis is shown in the left-most column. The sizes of the individual images are in proportion. The color range shown in the top row is used in the corresponding column, except the ψ_1 plot for the 24 DOFs/axis grid — there ψ_1 has the opposite sign then in the other rows. This is a side effect of the eigenvalue solver implementation and is physically insignificant.

a weak solution of the Kohn-Sham equations. Our computer implementation built upon the open source package SfePy supports computations both with the finite element basis and the NURBS or B-splines basis of isogeometric analysis. The latter allows a globally C^2 continuous approximation of unknown fields, which is crucial for computing of derivatives of the total energy w.r.t. atomic
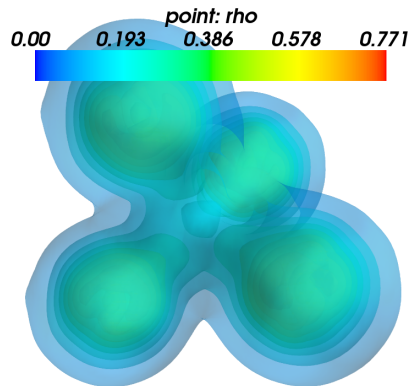
15

Figure 7: Tetrafluormethane molecule: iso-surfaces of charge density ρ.

positions etc., as given by the Hellmann-Feynman theorem.

Numerical results comparing the FEM and IGA calculations were presented on the benchmark problem of nitrogen atom. These results suggest significantly better convergence properties of IGA over FEM for our application, due to the higher smoothness of the approximation. This will be further studied on more complex substances, together with the implementation of the calculation of total energy derivatives. Finally, other quantities (charge density and related orbitals) that can be computed were illustrated using figures.

To alleviate the numerical quadrature cost, reduced quadrature rules has been proposed for the context of the Bézier extraction [26], which we plan to assess in future.

16

References

References

[1] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, T. W. Sederberg, Isogeometric analysis using T-splines, Comput. Methods Appl. Mech. Engrg. 199 (2010) 229–263.

[2] M. J. Borden, M. A. Scott, J. A. Evans, T. J. R. Hughes, Isogeometric finite element data structures based on Bezier extraction of NURBS, Int. J. Numer. Meth. Engng. 87 (2011) 15–47.

[3] R. Cimrman, Enhancing SfePy with isogeometric analysis, in: P. de Buyl, N. Varoquaux (eds.), Proceedings of the 7th European Conference on Python in Science (EuroSciPy 2014), 2014, pp. 65–72.
URL http://arxiv.org/abs/1412.6407

[4] R. Cimrman, SfePy - write your own FE application, in: P. de Buyl, N. Varoquaux (eds.), Proceedings of the 6th European Conference on Python in Science (EuroSciPy 2013), 2014, pp. 65–70.
URL http://arxiv.org/abs/1404.6391

[5] N. Collier, D. Pardo, L. Dalcin, M. Paszynski, V. M. Calo, The cost of continuity: A study of the performance of isogeometric finite elements using direct solvers, Computer Methods in Applied Mechanics and Engineering 213–216 (2012) 353 – 361.

[6] J. A. Cottrell, T. J. R. Hughes, Y. Bazilevs, Isogeometric Analysis: Toward Integration of CAD and FEA, John Wiley & Sons, Chichester, West Sussex, U.K., 2009.

[7] J. A. Cottrell, T. J. R. Hughes, A. Reali, Studies of refinement and continuity in isogeometric structural analysis, Comput. Methods Appl. Mech. Engrg. 196 (2007) 4160–4183.

[8] D. Davydov, T. D. Young, P. Steinmann, On the adaptive finite element analysis of the Kohn-Sham equations: Methods, algorithms, and implementation, International Journal for Numerical Methods in Engineering (2015) n/a–n/a.

[9] A. Di Pomponio, A. Continenza, R. Podloucky, J. Vackář, Symmetrization of atomic forces within the full-potential linearized augmented-plane-wave method, Phys. Rev. B 53 (1996) 9505–9508.

[10] R. M. Dreizler, E. K. U. Gross, Density Functional Theory, Springer-Verlag, Berlin, 1990.

[11] T. J. R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis, Dover Publications, Mineola, New York, USA, 2000.

[12] T. J. R. Hughes, J. A. Evans, A. Reali, Finite element and NURBS approximations of eigenvalue, boundary-value, and initial-value problems, Comput. Methods Appl. Mech. Engrg. 272 (2014) 290–320.

[13] T. J. R. Hughes, A. Reali, G. Sangalli, Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: Comparison of p-method finite elements with k-method NURBS, Computer Methods in Applied Mechanics and Engineering 197 (49–50) (2008) 4104–4124.

[14] J. Ihm, A. Zunger, C. M. L., Momentum-space formalism for the total energy of solids, J. Phys. C: Solid State Phys. 12 (1979) 4409–4422.

[15] W. Kohn, L. J. Sham, Self-consistent equations including exchange and correlation effects, Phys. Rev. 140 (4A) (1965) A1133–A1138.

[16] R. Kolman, J. Plešek, M. Okrouhlík, Complex wavenumber Fourier analysis of the B-spline based finite element method, Wave Motion 51 (2013) 348–359.

18

[17] R. Kolman, S. V. Sorokin, B. Bastl, J. Kopačka, J. Plešek, Isogeometric analysis of free vibration of simple shaped elastic samples, Journal of the Acoustical Society of America 137 (4) (2015) 2089–2100.

[18] R. M. Martin, Electronic Structure: Basic Theory and Practical Methods, Cambridge University Press, Cambridge, New York, USA, 2005.

[19] A. Masud, R. Kannan, B-splines and {NURBS} based finite element methods for Kohn–Sham equations, Computer Methods in Applied Mechanics and Engineering 241–244 (2012) 112–127.

[20] P. Motamarri, V. Gavini, Subquadratic-scaling subspace projection method for large-scale Kohn-Sham density functional theory calculations using spectral finite-element discretization, Phys. Rev. B 90 (2014) 115127.

[21] P. Motamarri, M. R. Nowak, K. Leiter, J. Knap, V. Gavini, Higher-order adaptive finite-element methods for Kohn–Sham density functional theory, Journal of Computational Physics 253 (2013) 308–343.

[22] R. G. Parr, Y. Weitao, Density-Functional Theory of Atoms and Molecules, Oxford University Press, Oxford, USA, 1994.

[23] W. E. Pickett, Pseudopotential methods in condensed matter applications, Comp. Phys. Reports 9 (1989) 115–198.

[24] L. Piegl, W. Tiller, The NURBS Book, (2nd ed.) ed., Springer-Verlag, New York, New York, USA, 1995–1997.

[25] D. Schillinger, J. A. Evans, A. Reali, M. A. Scott, T. J. R. Hughes, Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical {NURBS} discretizations, Computer Methods in Applied Mechanics and Engineering 267 (2013) 170 – 232.

[26] D. Schillinger, S. J. Hossain, T. J. R. Hughes, Reduced Bézier element quadrature rules for quadratic and cubic splines in isogeometric analysis, Computer Methods in Applied Mechanics and Engineering 277 (2014) 1–45.

19

[27] G. Strang, G. Fix, An Analysis of the Finite Element Method, Wellesley-Cambridge Press, Wellesley, 2008, pp. 414.

[28] J. Vackář, A. Šimůnek, Adaptability and accuracy of all-electron pseudopotentials, Phys. Rev. B 67 (2003) 125113.

[29] J. Vackář, O. Čertík, R. Cimrman, M. Novák, O. Šipr, J. Plešek, Advances in the Theory of Quantum Systems in Chemistry and Physics, vol. 22 of Prog. Theoretical Chem. and Phys., chap. Finite Element Method in Density Functional Theory Electronic Structure Calculations, Springer Netherlands, Netherlands, 2011, pp. 199–217.

[30] M. Weinert, J. W. Davenport, Fractional occupations and density-functional energies and forces, Phys. Rev. B 45 (1992) 13709–13712.

[31] R. Yu, D. Singh, H. Krakauer, All-electron and pseudopotential force calculations using the linearized-augmented-plane-wave method, Phys. Rev. B 43 (1991) 6411–6422.