Journal of Computational Physics 464 (2022) 111101

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A generalised formulation of G-continuous Bezier elements applied to non-linear MHD simulations

S.J.P. Pamela [a,*], G.T.A. Huijsmans [b,c], M. Hoelzl [d] and the JOREK Team [1]

[a] CCFE, Culham Science Centre, Abingdon, Oxon, OX14 3DB, UK
[b] CEA, IRFM, F-13108 Saint-Paul-lez-Durance, France
[c] Eindhoven University of Technology, 5612 AZ Eindhoven, the Netherlands
[d] Max-Planck Institute for Plasma Physics, 85748 Garching, Germany

ARTICLE INFO

Article history:
Received 27 March 2021
Received in revised form 31 January 2022
Accepted 24 February 2022
Available online 4 March 2022

Keywords:
FEM
Bezier
MHD
Fusion
Plasma

ABSTRACT

The international tokamak ITER is progressing towards assembly completion and first-plasma operation, which will be a physics and engineering challenge for the fusion community. In the preparation for ITER experimental scenarios, non-linear MHD simulations are playing an essential role to actively understand and predict the behaviour and stability of tokamak plasmas in future fusion power plant. The development of MHD codes like JOREK is a key aspect of this research effort, and provides invaluable insight into the plasma stability and the control of global and localised plasma events, like Edge-Localised-Mode and disruptions. In this paper, we present an operational implementation of a new, generalised formulation of Bezier finite-elements applied to the JOREK code, a significant advancement from the previously G1-continuous bi-cubic Bezier elements. This new mathematical method enables any polynomial order of Bezier elements, with a guarantee of G-continuity at the level of $(n-1)/2$, for any odd n, where n is the order of the Bezier polynomials. The generalised method is defined, and a rigorous mathematical proof is provided for the G-continuity requirement. Key details on the code implementation are mentioned, together with a suite of tests to demonstrate the mathematical reliability of the finite-element method, as well as the practical usability for typical non-linear tokamak MHD simulations. A demonstration for a state-of-the-art simulation of an Edge-Localised-Mode instability in the future ITER tokamak, with realistic grid geometry, finalises the study.

1. Introduction

1.1. Motivations

In the current alarming climate change situation [1], nuclear fusion could provide an abundant energy source with a minimal level of greenhouse gas emissions and no long-lived radioactive nuclear waste. Together with renewable energies, fusion could contribute to the electricity of future societies, without the limit of exhaustible natural resources. Currently, the most promising candidate for industrial fusion reactors is the tokamak device [2], which uses a magnetic field to confine

* Corresponding author.
E-mail address: stanislas.pamela@hotmail.com (S.J.P. Pamela).
[1] See author list of M. Hoelzl et al., Nucl. Fusion (submitted); [https://arxiv.org/abs/2011.09120].

https://doi.org/10.1016/j.jcp.2022.111101

a hot plasma composed of hydrogen isotopes. The toroidal, periodic nature of the tokamak ensures that the plasma's ions and electrons, which approximately follow the magnetic field lines, are not lost at the end of open field lines, like in linear plasma devices. However, this periodicity can be subject to resonance and instability. These resonant and unstable modes typically involve the plasma and the magnetic field, and are often studied using magnetohydrodynamics (MHD), a model that combines the Navier–Stokes with Maxwell's equations. In tokamak physics, *extended* MHD models are typically used, to account for additional kinetic or geometric effects, like diamagnetic rotation or bootstrap current [3,4].

Beyond the theoretical study of the MHD equations and the properties of various waves and unstable modes in a tokamak [5], numerical simulations can provide essential insight into the dynamics of plasma instabilities. In particular, these can determine how MHD instabilities can limit the operational capabilities of tokamak devices, which is a key aspect of both present machines analysis, as well as the design of future reactors. The most limiting MHD instabilities in tokamaks include Edge-Localised-Modes (ELMs), Toroidal Alfven Eigenmodes (TAEs) and Global instabilities (Disruptions). ELMs eject plasma from the edge region onto the first wall of the machine, leading to large heat-fluxes and thus potential damage to surface materials [6–9]. TAEs are excited by the 3.5 MeV alpha-particles born from fusion reactions, and can limit the core plasma pressure [10–13]. Global MHD instabilities, during which the entire plasma is affected, can lead to the total loss of plasma control, which is often called a disruption. During disruption events, the kinetic and magnetic energy of the plasma can be transferred to the wall, leading to material heat-fluxes and/or wall-currents that can damage the structural components of the machine [14–19]. In order to study, understand and predict these MHD instabilities, numerical simulations are performed using codes like JOREK [20–23], M3D-C1 [24,25], NIMROD [26,27], XTOR [28], BOUT++ [29,30], MEGA [31–33], HALO [12] (and many others).

1.2. Numerical context

There is a wide range of finite-element methods used in the community of fusion modelling. The use of G1- or C1-continuous finite-elements has proven essential for the practical reliability of codes like JOREK [21,23] and M3D-C1 [24,25]. Note: G-continuity means geometrical continuity, so that derivatives are collinear on both sides of element edges, whereas C-continuity means parametric continuity, such that the collinear derivatives on both sides also need to have the same amplitude in the parametric space, which is a restricted sub-case of G-continuity. Throughout the remainder of the text, the term '*practical reliability*' is meant from a user's perspective, to describe simulations that are capable of running through highly non-linear phases and numerically challenging MHD dynamics. This term is used to avoid any confusion with '*numerical stability*' in the variational sense, where a variation leads to a minimisation of energy-like (squared) terms, which is not necessarily a property of higher continuity finite-elements. Additionally, it should be pointed out that higher-order continuity constraints, for a given polynomial order of finite-elements, implies a subspace of the full (unconstrained) solution space of the finite-elements; as such, higher-continuity may in fact restrict a computation from finding solutions that are, potentially, numerical more stable.

Nevertheless, G1/C1-continuity can be highly beneficial when dealing with physics models that include high-order spatial derivatives, typically 2nd order derivatives for diffusive terms like viscosity, resistivity and particle/heat diffusion, as well as 4th order derivatives for hyper-diffusion terms. Higher-order derivatives can be reduced numerically, either by using the Weak-Formulation method [34], or by using auxiliary variables, like toroidal vorticity and toroidal current in reduced-MHD [21]. Thus, some MHD models only contain first-order derivatives in the numerical sense, but higher-order derivatives may be present for other terms, like hyper-diffusion, or physics effects like the Ohmic heating. Although most 2nd order derivatives in the JOREK full-MHD model [35] can be removed with the Weak-Formulation method, the Ohmic heating contains squares of 2nd order derivatives of the poloidal magnetic flux ψ, like $(\partial^2_R \psi)^2$, which cannot be integrated by parts and removed with the Weak-Formulation. In other words, it is debatable whether higher-order elements and higher-order continuity are practically advantageous or not for MHD applications. However, this question can only be properly addressed if these higher-order methods are implemented and available, which is the primary purpose of this work, for the JOREK code.

Simulations of MHD models typically evolve around an equilibrium between two large force terms: the kinetic pressure gradient force ∇p, and the magnetic (Lorentz) force $\vec{J} \times \vec{B}$. At equilibrium, these two terms in the momentum equation cancel each other out, but when dealing with instabilities, it is precisely this balance that breaks, leading to spatially finely localised differences arising from the $\nabla p - \vec{J} \times \vec{B}$ imbalance. In particular, the thin current layers observed in previous non-linear MHD studies [36,37] inevitably involve large second-order derivatives, which may potentially be better resolved with higher-order elements. In this highly-nonlinear environment, having the flexibility to control spatial resolution either through *h*-refinement (size of the finite-elements) or *p*-refinement (order of polynomial basis inside finite-elements) can be an advantage. Although higher-order elements can be subject to Gibbs phenomenon oscillations (when localised jumps in the physical solution causes ripples in its polynomial representation), in the cases of reduced-MHD tested here with JOREK, it is observed that the higher-order elements do not have worse levels of numerical noise than the previous bi-cubic G1-continuous finite-element implementation, for spatially under-resolved grids. The convergence tests, presented in the last section of this paper, clearly show that higher-order elements become computationally cheaper if the numerical errors have to be minimised below a given threshold. Another motivation to include higher order elements is for the evolution of kinetic and/or gyro-kinetic particles with the MHD fields, as in [38,39], where 2nd (or higher) order derivatives may be required to evaluate the necessary magnetic and electric fields. Moreover, with coupled kinetic/fluid models, as in [11], the

number of particles inside each element needs to be projected onto the degrees-of-freedom of the element, and the overall noise due to this projection could potentially be reduced with a larger number of degrees-of-freedom (although this has not been verified in this paper).

In the wider computational physics community, there are several advanced Finite-Element platforms, such as MFEM [40,41], GMSH [42,43] or FIREDRAKE [44,45], that specialise in high-order p-refinement as well as continuous *H(div)* and *H(Curl)* methods, and that include MHD applications [46,47]. The method presented here is not yet available in these FEM platforms, and implementing it would likely require significant efforts, which is well beyond the scope of this paper. The aim of the work presented here is to certify the feasibility of the method itself, and further detailed studies will certainly be required to assess how it compares to these alternative finite-element methods. Nevertheless, since there is a clear benefit (for some applications) in using C1- and G1-continuous elements, as well as continuous *H(div)* and *H(Curl)* methods, it is conceivable that higher order continuity between elements may also be advantageous in certain specific cases, particularly for physics models that require high-order derivatives. In addition, the finite-element formulation presented here inherently links the order of G-continuity with the order of the element's polynomial basis. The main reason for this is that it reduces the number of degrees-of-freedom, but having the flexibility to choose a certain level of G-continuity independently of the finite element basis could be interesting, both from a user's perspective, as well as from a computational and numerical perspective. The NIMROD code [27] provides a study of MHD instabilities while varying the degree of the basis functions at fixed continuity level, but the ability to change the level of continuity at fixed polynomial degree could also be of interest. However, this would require a reformulation of the finite-element numerical structure inside JOREK, which is beyond the work addressed here.

1.3. Mathematical context

Substantial mathematical work has been performed in geometry regarding Bezier patches and their Gn-continuity properties [48–51], and similarly extensive work has been published on C2- and G2-continuous Bezier patches [52–54]. Before proceeding to the method presented here, the reader should be aware of the remarkable work of Aumann & Bentz [48], whose method bears great similarities to this manuscript.

In particular, it is worth noting the close resemblance of Aumann & Bentz's last formula (18) with our formulation (45). Unfortunately, Aumann & Bentz's formula (18) would not be sufficient to obtain a construction method for our particular case, for two key reasons. Firstly, our formulation is a nodal formulation, where neighbour finite-elements are described using a common vector basis at their common (shared) nodes. This nodal formulation is essential for a practical implementation in the JOREK code and enables us to reduce the number of degrees-of-freedom, while imposing continuity by construction of the finite-elements, independently of the values of the degrees-of freedom themselves. Secondly, Aumann & Bentz only considers continuity between 2 Bezier patches (i.e. in one direction), whereas our case requires continuity between 4 patches (i.e. in two directions). Interestingly, Aumann & Bentz's formula is simultaneously more elaborate and simpler than our case: it is generalised for two patches of arbitrary polynomial order (whereas we require all patches to have the same odd polynomial order), but it does not provide a formulation for 4 patches meeting at the same corner node, requiring continuity in two directions, which our formulation provides.

However, we expect there may exist an alternative mathematical proof to the one we have provided in this manuscript, where one would start from our formula (45), and proceed to recover Aumann & Bentz's formula (18), in the two directions across the two boundaries of the 4 elements that share the corner node. Unfortunately, we provide here only the direct proof of Gn-continuity, and cannot ascertain that this alternative proof exists, due to the nodal nature of our formulation. In addition, this alternative proof might be equivalent in length and complexity to the one we have provided here. Nevertheless, obtaining a bridge from our work to Aumann & Bentz's formula (18) would be of mathematical interest.

1.4. Overview of the work

In this paper, we present a generalised method for high-order Bezier finite-elements, which ensures that G-continuity increases linearly with the polynomial order of the elements. The method is defined, proven, and implemented in the JOREK code with several numerical tests and benchmark cases. Section-2 introduces the G2-continuous Bezier finite-elements, which serves as an introduction to the generalised method. Section-3 presents the definition the generalised Bezier formulation, together with the mathematical proofs to demonstrate G-continuity. Section-4 describes some of the key details that were required for the implementation of this new method in the JOREK code, and Section-5 presents the series of tests conducted to demonstrate the mathematical accuracy of the new method, together with the practical usability and advantages of using higher-order finite-element methods for non-linear MHD instabilities in tokamaks. Finally Section-6 summarises the work and lays out the further improvements required for future studies of tokamak instabilities.

2. G2-continuous bi-quintic Bezier elements

2.1. Bezier curves and Bezier elements

The work presented in this paper relies on established methods of Bezier curves and elements [22,34]. Iso-parametric Bezier curves of order n are commonly described by the formulation of any function (including spatial coordinates) as

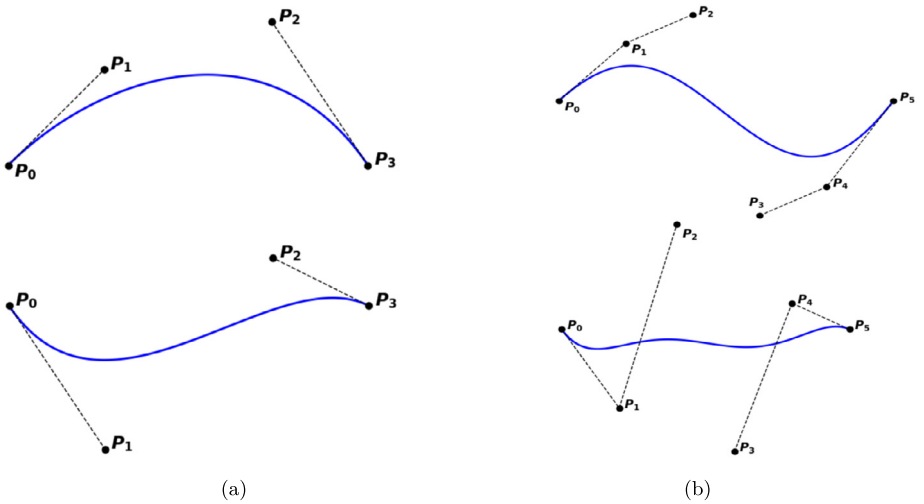


Fig. 1. (a) Cubic Bezier segments and (b) quintic Bezier segments, showing control points.

$$\boldsymbol{F}(s) = \sum_{i=0}^{n} B_i(s)\, \boldsymbol{P}_i \tag{1}$$

Where s is the element's local coordinate, the points $\boldsymbol{P}_i$ are the control points of the curve, and the Bernstein polynomials are defined as

$$B_i(s) = \frac{n!}{i!(n-i)!} s^i (1-s)^{n-i} \tag{2}$$

Note that the control points $\boldsymbol{P}_i$ can be position vectors in any spatial dimension. Whether in 3D, 2D, or even 1D, the above Bezier formulation $\boldsymbol{F}$ will describe a curve. This is represented for Bezier segments of order 3 and 5 in Fig.-1. For cubic curves, the end control points $\boldsymbol{P}_0$ and $\boldsymbol{P}_3$ determine the position of the segment vertices (or nodes), while the vectors $[\boldsymbol{P}_0, \boldsymbol{P}_1]$ and $[\boldsymbol{P}_3, \boldsymbol{P}_2]$ control the first derivatives (or directions) of the segments. Cubic Bezier segments can have up to two inflexion points. For quintic curves, the end control points $\boldsymbol{P}_0$ and $\boldsymbol{P}_5$ determine the position of the segment vertices, the first vectors $[\boldsymbol{P}_0, \boldsymbol{P}_1]$ and $[\boldsymbol{P}_5, \boldsymbol{P}_4]$ control the first derivatives, and the second vectors $[\boldsymbol{P}_1, \boldsymbol{P}_2]$ and $[\boldsymbol{P}_4, \boldsymbol{P}_3]$ control the second derivatives. Quintic Bezier segments can have up to four inflexion points.

Similarly, iso-parametric Bezier finite elements of order n are described in the literature [22,34] by the formulation of any function (including spatial coordinates) as

$$\boldsymbol{F}(s,t) = \sum_{i=0}^{n}\sum_{j=0}^{n} B_i(s) B_j(t)\, \boldsymbol{P}_{ij} \tag{3}$$

Where s and t are the element's local coordinates, the points $\boldsymbol{P}_{ij}$ are the control points of the elements, and the Bernstein polynomials are the same as defined above by (2). Again, note that the control points $\boldsymbol{P}_{ij}$ can be position vectors in any dimension, but even in 3D, the above Bezier formulation $\boldsymbol{F}$ will describe a surface. In the remainder of this paper, for simplicity, the control points will always be assumed to be in 2D, although this changes nothing to the mathematical derivations, which are identical for the 3D case. 2D Bezier elements of order 3 and 5 are represented in Fig.-1. The iso-parametric property of Bezier elements means that the spatial coordinates have the same formulation (3) as any other variable.

2.2. Continuity between Bezier curves or elements

The continuity between two Bezier curves or two Bezier patches is also well established in the literature [22,34], and repeated here for clarity. Consider two cubic Bezier curves ζ^1 and ζ^2, with control point $[\boldsymbol{P}_0, \boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}_3]$ and $[\boldsymbol{Q}_0, \boldsymbol{Q}_1, \boldsymbol{Q}_2, \boldsymbol{Q}_3]$ respectively. The two curves are continuous, or G0-continuous, provided the two control points $\boldsymbol{P}_3$ and $\boldsymbol{Q}_0$ are identical. The curves are smooth, or G1-continuous, provided the vectors $[\boldsymbol{P}_4, \boldsymbol{P}_3]$ and $[\boldsymbol{Q}_0, \boldsymbol{Q}_1]$ are aligned, such that $(\boldsymbol{P}_3 - \boldsymbol{P}_2) = \alpha(\boldsymbol{Q}_1 - \boldsymbol{Q}_0)$ for any non-zero positive scalar α. This is represented by Fig.-3, where α has been chosen to be $\alpha = 1$. This condition is important with respect to the Bezier formalism of finite elements: the freedom of allowing $\alpha \neq 1$ means that finite elements can have different sizes on each side of a node, implying that the derivatives along the local coordinate are not continuous, only derivatives in real space are (i.e. G1 continuity).

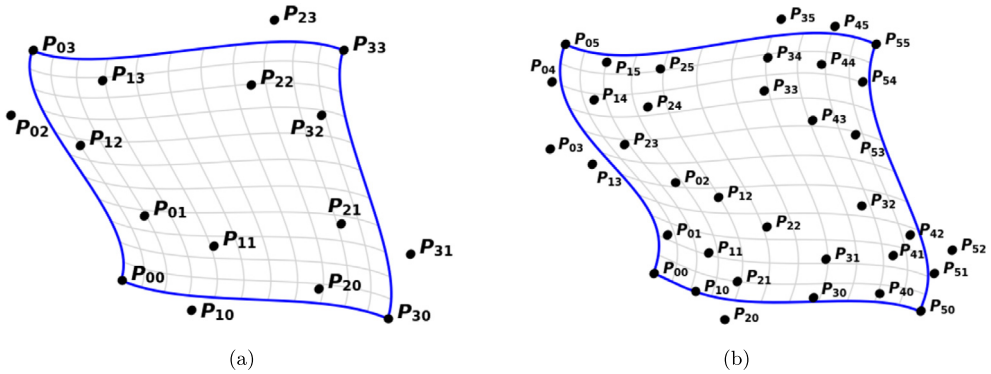


Fig. 2. (a) Bi-cubic Bezier element and (b) bi-quintic Bezier element, showing control points.

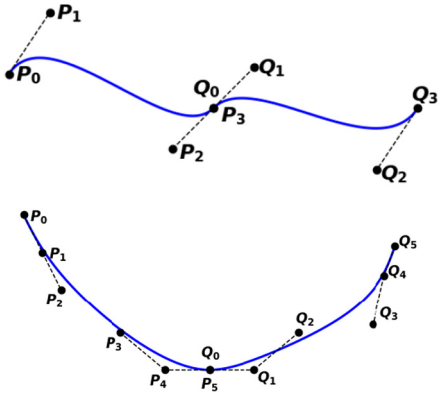


Fig. 3. G1-continuity between two cubic Bezier curves (top). G2-continuity between two quintic Bezier curves (bottom).

In order to achieve G2-continuity, such that the curvature (or second derivative) of the curves is continuous at the point $\boldsymbol{P}_3 = \boldsymbol{Q}_0$, the following rule (where 'rule' is used as synonym of 'constraint') must be satisfied:

$$(\boldsymbol{P}_1 - \boldsymbol{P}_2) + (\boldsymbol{P}_3 - \boldsymbol{P}_2) = (\boldsymbol{Q}_2 - \boldsymbol{Q}_1) + (\boldsymbol{Q}_0 - \boldsymbol{Q}_1) \tag{4}$$

Note that in the bottom Fig.-3, the G1-continuity has also been chosen with $\alpha = 1$, however the G2-continuity rule (4) is the same regardless of the choice of α. In addition, the curve lengths on either side of $\boldsymbol{Q}_0$ in Fig.-3 are equal, but this is arbitrary, and rule (4) is independent of the lengths of each curve, which are determined by the positions of the end control points $\boldsymbol{P}_0$ and $\boldsymbol{Q}_5$.

There is an important implication from this rule for cubic Bezier lines/elements. G2 continuity at vertex $\boldsymbol{P}_3$ in the top Fig.-3 would impose the choice of $\boldsymbol{P}_1$, such that if the position of $\boldsymbol{P}_0$ is fixed, then the first derivative value at $\boldsymbol{P}_0$, controlled by $\boldsymbol{P}_1$ is also imposed by the G2-continuity at $\boldsymbol{P}_3$. In other words, G2-continuity at one vertex uses the control points from another vertex. While this may be acceptable in some simple symmetrical geometries, it greatly diminishes the flexibility of finite elements nodes to have their own 1st and 2nd derivatives, regardless of neighbouring nodes. This is the main reason to increase from cubic to quintic finite elements: since quintic curves have 6 control points, each node has its own set of 3 control points to determine 2nd order derivatives and G2-continuity with the next curve.

2.3. Nodal formulation of bi-quintic elements

The novelty of this work consists in building upon the ground method established by [22], which provides a nodal formulation of bi-cubic Bezier patches with G1-continuity. As in [22], it is preferable to use a nodal formulation of the finite elements, and set constraints that will guarantee G0, G1 and G2 continuity of all variables for any solution of the system to be solved. The advantage of a nodal formulation is that it centralises the degrees-of-freedom from the four parent elements of a node (since with quadrangular elements, four neighbour elements have one node in common). It is an effective way of using Bezier patches as finite elements, and simplifies the numerical formulation required for the linear system in the implicit time discretisation. With Bezier patches each patch has its own, independent, control points. In a nodal formulation the control points of an element are split into 4 groups (one group for each node), like points $[\boldsymbol{P}_{00}, \boldsymbol{P}_{01}, \boldsymbol{P}_{10}, \boldsymbol{P}_{11}]$ at node $\boldsymbol{P}_{00}$ in Fig.-2a. Then, G-continuity is used to constrain the control points of all 4 parent elements at a given node, such that the corresponding groups of control points of the 4 parent elements share the same degrees of freedom. For example, in

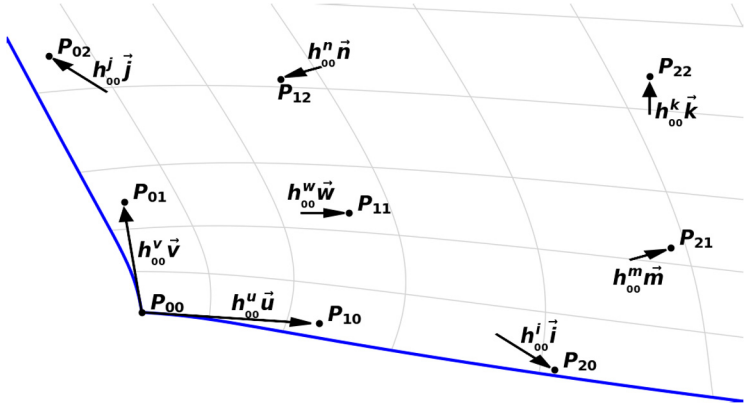


Fig. 4. Nodal formulation of bi-quintic elements, focused on the first node P_{00} of an element similar to that represented in Fig.-2b.

Fig.-2a, there would be only 4 degrees of freedom at each node, effectively determining the position of 16 control points in total (one group of 4 control points for each of the 4 parent elements). To achieve this nodal formulation, we propose a vector basis used to locate all control points.

In this nodal formulation, each of the 4 nodes of an element serves as a reference point for its neighbouring control points. In Fig.-2a, for example, the 3 control points $[\boldsymbol{P}_{01}, \boldsymbol{P}_{10}, \boldsymbol{P}_{11}]$ are associated to the node $\boldsymbol{P}_{00}$. Given that $\boldsymbol{P}_{00}$ is known, the control point $\boldsymbol{P}_{10}$ can be defined with a vector $\vec{U}_{00}$ between $\boldsymbol{P}_{00}$ and $\boldsymbol{P}_{10}$ such that $\boldsymbol{P}_{10} = \boldsymbol{P}_{00} + \vec{U}_{00}$. However, in order to impose continuity constraints, it is necessary to define this point as $\boldsymbol{P}_{10} = \boldsymbol{P}_{00} + h^u_{00}\vec{u}_{00}$. The scalar quantity h^u_{00} can be thought of as the *size* or *extension* of the element, at node $\boldsymbol{P}_{00}$, in the *horizontal* direction. Likewise, in the *vertical* direction, the point $\boldsymbol{P}_{01}$ will be defined using another vector $\vec{v}_{00}$ and element size h^v_{00} such that $\boldsymbol{P}_{01} = \boldsymbol{P}_{00} + h^v_{00}\vec{v}_{00}$. Of course, the two dimensions *horizontal* and *vertical* are relative terms here, since an element can be in any direction. It is important to note that the vectors $\vec{u}_{00}$ and $\vec{v}_{00}$ are not unit vectors, they will represent the degrees-of-freedom of the final linear system. These degrees-of-freedom will be shared by all 4 parent elements that share the common node $\boldsymbol{P}_{00}$, so that the values of $\vec{u}_{00}$ and $\vec{v}_{00}$ will be the same for all elements, but the element sizes h^u_{00} and h^v_{00} may be different for each element. It is by constraining the sizes h^u_{00} and h^v_{00} on the 4 parent elements that continuity will be ensured.

In bi-cubic elements, the node value plus 3 vectors are necessary to define the 4 control points associated to each node. For bi-quintic elements, the node value plus 8 vectors are required. The nodal formulation, at node $\boldsymbol{P}_{00}$ in Fig.-2b, is defined as

$$
\begin{aligned}
&\boldsymbol{P}_{00} \\
&\boldsymbol{P}_{10} = \boldsymbol{P}_{00} + h^u_{00}\vec{u}_{00} \\
&\boldsymbol{P}_{01} = \boldsymbol{P}_{00} + h^v_{00}\vec{v}_{00} \\
&\boldsymbol{P}_{11} = \boldsymbol{P}_{00} + h^u_{00}\vec{u}_{00} + h^v_{00}\vec{v}_{00} + h^w_{00}\vec{w}_{00} \\
&\boldsymbol{P}_{20} = \boldsymbol{P}_{00} + 2h^u_{00}\vec{u}_{00} + h^i_{00}\vec{i}_{00} \\
&\boldsymbol{P}_{02} = \boldsymbol{P}_{00} + 2h^v_{00}\vec{v}_{00} + h^j_{00}\vec{j}_{00} \\
&\boldsymbol{P}_{21} = \boldsymbol{P}_{00} + 2h^u_{00}\vec{u}_{00} + h^v_{00}\vec{v}_{00} + h^i_{00}\vec{i}_{00} + 2h^w_{00}\vec{w}_{00} + h^m_{00}\vec{m}_{00} \\
&\boldsymbol{P}_{12} = \boldsymbol{P}_{00} + 2h^v_{00}\vec{v}_{00} + h^u_{00}\vec{u}_{00} + h^j_{00}\vec{j}_{00} + 2h^w_{00}\vec{w}_{00} + h^n_{00}\vec{n}_{00} \\
&\boldsymbol{P}_{22} = \boldsymbol{P}_{00} + 2h^u_{00}\vec{u}_{00} + 2h^v_{00}\vec{v}_{00} + h^i_{00}\vec{i}_{00} + h^j_{00}\vec{j}_{00} + 2h^m_{00}\vec{m}_{00} + 2h^n_{00}\vec{n}_{00} + 4h^w_{00}\vec{w}_{00} + h^k_{00}\vec{k}_{00}
\end{aligned}
\tag{5}
$$

where the vectors $\vec{u}_{00}$, $\vec{v}_{00}$, $\vec{w}_{00}$, $\vec{i}_{00}$, $\vec{j}_{00}$, $\vec{m}_{00}$, $\vec{n}_{00}$, $\vec{k}_{00}$, and the element sizes h^u_{00}, h^v_{00}, h^w_{00}, h^i_{00}, h^j_{00}, h^m_{00}, h^n_{00}, h^k_{00} have been introduced. A representation of this formulation is shown in Fig.-4.

Similar definitions are used for the nodal formulation at nodes $\boldsymbol{P}_{05}$, $\boldsymbol{P}_{50}$ and $\boldsymbol{P}_{55}$, to locate all control points of an element. It is worth noting that, as will become evident later, the choice of this particular formulation is not random. For example, point $\boldsymbol{P}_{22}$ could simply have been defined as $\boldsymbol{P}_{00} + h^k_{00}\vec{k}$. However, using this mixture of vectors to represent various points is essential in order to simplify (as far as possible) the various G2-continuity constraints. As will be shown further on, in Corollary-1, this definition has the advantage that each vector corresponds to a derivative with respect to the local coordinates (s, t) of the element. Since the continuity constraints involve these derivatives, it is desirable to have a direct correspondence between the vectorial basis and the derivatives. Using a different nodal formulation would mean that each derivative has to be expressed as a linear combination of multiple vectors. In that case, continuity is still possible, of course, but the formulations required to describe continuity constraints would become increasingly complex, and render the mathematical demonstration difficult to follow.

Using the nodal formulation (5) at the four nodes of an element, all control points (except $\boldsymbol{P}_{00}$, $\boldsymbol{P}_{50}$, $\boldsymbol{P}_{05}$ and $\boldsymbol{P}_{55}$) can now be substituted into the Bezier definition of a finite element (3), to write $\boldsymbol{F}(s,t)$ as

$$\begin{aligned}\boldsymbol{F}(s,t) = \sum_{xx} \Big[& F^{P}_{xx}\,\boldsymbol{P}_{xx} + F^{u}_{xx} h^{u}_{xx} \vec{u}_{xx} + F^{v}_{xx} h^{v}_{xx} \vec{v}_{xx} \\ & + F^{w}_{xx} h^{w}_{xx} \vec{w}_{xx} + F^{i}_{xx} h^{i}_{xx} \vec{i}_{xx} + F^{j}_{xx} h^{j}_{xx} \vec{j}_{xx} \\ & + F^{m}_{xx} h^{m}_{xx} \vec{m}_{xx} + F^{n}_{xx} h^{n}_{xx} \vec{n}_{xx} + F^{k}_{xx} h^{k}_{xx} \vec{k}_{xx} \Big] \\ & \text{with } xx = [00, 05, 50, 55],\end{aligned} \tag{6}$$

where a new set of scalar basis functions F^{P}_{xx}, F^{u}_{xx}, ..., F^{k}_{xx}, has been obtained by factorizing $\boldsymbol{F}(s,t)$ with respect to each vector. Thus, the vectors $\vec{u}$, $\vec{v}$, etc. represent the degrees of freedom that are used and solved in the system of linearised equations, while the element sizes h^{u}, h^{v}, etc. are fixed in time, and chosen to ensure G2-continuity between elements, for any values of the vectors $\vec{u}$, $\vec{v}$, etc. In practice, the finite elements can be constructed either by positioning the control points themselves, as would traditionally be done for Bezier patches (like in Fig.-2), or by defining the vectors $\vec{u}$, $\vec{v}$, $\vec{w}$, etc. directly at each node. However, if the elements are constructed by positioning the control points, then the vectors will need to be calculated by inverting (5), because it is the vectors that act as degrees of freedom, which are needed inside the code (not the control points). Note that inverting (5) is a trivial linear exercise.

In this case, the Bezier function $\boldsymbol{F}(s,t)$ is actually a 2D vector, but this can simply be considered to be two separate scalar functions $X(s,t)$ and $Y(s,t)$, with each set of vectors being actual scalar degrees of freedom for each function. This is true for any variable ψ of the system to be evaluated, and which can be seen as extra dimensions in which the Bezier patch $\boldsymbol{F}$ resides. It is written here as vectors just to simplify the visualisation with the use of plots of 2D elements like Fig.-4.

2.4. G0-, G1- and G2-continuity constraints

Finally, constraints must be set on the elements sizes in order to ensure G2-continuity for any set of node vectors. Rather than simply defining those constraints and showing that G0-, G1- and G2-continuity is thus satisfied, it is better to start from the continuity requirements, and deduce the constraints from those. Despite being more lengthy, it will provide a clearer understanding why each set of constraints is chosen. To describe these constraints, let the 4 Bezier elements ξ_{11}, $\xi_{\text{-}11}$, $\xi_{1\text{-}1}$, and $\xi_{\text{-}1\text{-}1}$ have the common node $\boldsymbol{P}_{00}$. The control points of the 4 parent elements around node $\boldsymbol{P}_{00}$ can be defined, similarly to (5), using the subscripts "11", "1-1", "-1-1" and "-11" for each of the 4 parent elements. Note that, by definition, all elements use the same vectors for the nodal formulation, but have different element sizes. Hence, on element ξ_{11}, the nodal formulation is

$$\begin{aligned}
\boldsymbol{P}_{10} &= \boldsymbol{P}_{00} + h^{u}_{11}\vec{u} \\
\boldsymbol{P}_{01} &= \boldsymbol{P}_{00} + h^{v}_{11}\vec{v} \\
\boldsymbol{P}_{11} &= \boldsymbol{P}_{00} + h^{u}_{11}\vec{u} + h^{v}_{11}\vec{v} + h^{w}_{11}\vec{w} \\
\boldsymbol{P}_{20} &= \boldsymbol{P}_{00} + 2h^{u}_{11}\vec{u} + h^{i}_{11}\vec{i} \\
\boldsymbol{P}_{02} &= \boldsymbol{P}_{00} + 2h^{v}_{11}\vec{v} + h^{j}_{11}\vec{j} \\
\boldsymbol{P}_{21} &= \boldsymbol{P}_{00} + 2h^{u}_{11}\vec{u} + h^{v}_{11}\vec{v} + h^{i}_{11}\vec{i} + 2h^{w}_{11}\vec{w} + h^{m}_{11}\vec{m} \\
\boldsymbol{P}_{12} &= \boldsymbol{P}_{00} + 2h^{v}_{11}\vec{v} + h^{u}_{11}\vec{u} + h^{j}_{11}\vec{j} + 2h^{w}_{11}\vec{w} + h^{n}_{11}\vec{n} \\
\boldsymbol{P}_{22} &= \boldsymbol{P}_{00} + 2h^{u}_{11}\vec{u} + 2h^{v}_{11}\vec{v} + h^{i}_{11}\vec{i} + h^{j}_{11}\vec{j} \\
&\quad + 2h^{m}_{11}\vec{m} + 2h^{n}_{11}\vec{n} + 4h^{w}_{11}\vec{w} + h^{k}_{11}\vec{k}
\end{aligned} \tag{7}$$

while on the other three elements, minus subscripts will be used for the corresponding points and element sizes. For example, point $\boldsymbol{P}_{\text{-}21}$ (on element $\xi_{\text{-}11}$), point $\boldsymbol{P}_{2\text{-}1}$ (on element $\xi_{1\text{-}1}$), and point $\boldsymbol{P}_{\text{-}2\text{-}1}$ (on element $\xi_{\text{-}1\text{-}1}$), are defined as

$$\begin{aligned}
\boldsymbol{P}_{\text{-}21} &= \boldsymbol{P}_{00} + 2h^{u}_{\text{-}11}\vec{u} + h^{v}_{\text{-}11}\vec{v} + h^{i}_{\text{-}11}\vec{i} + 2h^{w}_{\text{-}11}\vec{w} + h^{m}_{\text{-}11}\vec{m} \\
\boldsymbol{P}_{2\text{-}1} &= \boldsymbol{P}_{00} + 2h^{u}_{1\text{-}1}\vec{u} + h^{v}_{1\text{-}1}\vec{v} + h^{i}_{1\text{-}1}\vec{i} + 2h^{w}_{1\text{-}1}\vec{w} + h^{m}_{1\text{-}1}\vec{m} \\
\boldsymbol{P}_{\text{-}2\text{-}1} &= \boldsymbol{P}_{00} + 2h^{u}_{\text{-}1\text{-}1}\vec{u} + h^{v}_{\text{-}1\text{-}1}\vec{v} + h^{i}_{\text{-}1\text{-}1}\vec{i} + 2h^{w}_{\text{-}1\text{-}1}\vec{w} + h^{m}_{\text{-}1\text{-}1}\vec{m}
\end{aligned} \tag{8}$$

Again, note that throughout the text, the terms 'continuity rules' are often used with the same meaning as 'continuity constraints'.

2.5. G0-continuity

G0-continuity requires that, apart from the common node $\boldsymbol{P}_{00}$, all control points along the element sides must coincide on both sides of an element, such that vector sizes used to define these common points must be equal on either sides of an element boundary, namely

$$\begin{aligned} h^u_{1\text{-}1} &= h^u_{11} \quad \text{and} \quad h^u_{\text{-}1\text{-}1} = h^u_{\text{-}11} \\ h^v_{\text{-}11} &= h^v_{11} \quad \text{and} \quad h^v_{\text{-}1\text{-}1} = h^v_{1\text{-}1} \\ h^i_{1\text{-}1} &= h^i_{11} \quad \text{and} \quad h^i_{\text{-}1\text{-}1} = h^i_{\text{-}11} \\ h^j_{\text{-}11} &= h^j_{11} \quad \text{and} \quad h^j_{\text{-}1\text{-}1} = h^j_{1\text{-}1} \end{aligned} \tag{9}$$

2.6. G1-continuity

Firstly, considering G1-continuity at the node itself, $\boldsymbol{P}_{00}$, the requirement, as mentioned above, is that opposite vectors are aligned, but this is already achieved by the nodal formulation (5), since we use the same vectors $\vec{u}$ and $\vec{v}$ for all elements. Therefore, the first constraints are

$$\begin{aligned} h^u_{\text{-}11} &= -\alpha\, h^u_{11} \quad \text{with} \quad \alpha > 0 \\ h^v_{1\text{-}1} &= -\beta\, h^v_{11} \quad \text{with} \quad \beta > 0 \end{aligned} \tag{10}$$

Note that the scalars α and β are related to the geometrical property of the continuity: if $\alpha = \beta = 1$ is imposed, then the elements are C-continuous (with parametric continuity), but if they are not restricted to 1, then the elements are G-continuous (geometrically continuous). In other words, $\alpha = \beta = 1$ would impose that the parametric derivatives are equal on both sides of element boundaries. This would be a strongly limiting restriction, potentially problematic if two neighbour elements have very different sizes (areas).

Secondly, considering G1-continuity at the points directly adjacent to the node $\boldsymbol{P}_{00}$, namely the control points $\boldsymbol{P}_{01}$, $\boldsymbol{P}_{10}$, $\boldsymbol{P}_{0\text{-}1}$ and $\boldsymbol{P}_{\text{-}10}$, this requires the alignment of control points, such that

$$\begin{aligned} (\boldsymbol{P}_{11} - \boldsymbol{P}_{01}) &= -\delta_1(\boldsymbol{P}_{\text{-}11} - \boldsymbol{P}_{01}) \quad \text{with} \quad \delta_1 > 0 \\ (\boldsymbol{P}_{1\text{-}1} - \boldsymbol{P}_{0\text{-}1}) &= -\delta_2(\boldsymbol{P}_{\text{-}1\text{-}1} - \boldsymbol{P}_{0\text{-}1}) \quad \text{with} \quad \delta_2 > 0 \\ (\boldsymbol{P}_{11} - \boldsymbol{P}_{10}) &= -\delta_3(\boldsymbol{P}_{1\text{-}1} - \boldsymbol{P}_{10}) \quad \text{with} \quad \delta_3 > 0 \\ (\boldsymbol{P}_{\text{-}11} - \boldsymbol{P}_{\text{-}10}) &= -\delta_4(\boldsymbol{P}_{\text{-}1\text{-}1} - \boldsymbol{P}_{\text{-}10}) \quad \text{with} \quad \delta_4 > 0 \end{aligned} \tag{11}$$

Substituting the formulations of control points from (7), (8), and using the identities (9) and (10), gives

$$\begin{aligned} (1-\alpha\delta_1)h^u_{11}\vec{u} \;+\; (h^w_{11} + \delta_1 h^w_{\text{-}11})\vec{w} &= 0 \\ (1-\alpha\delta_2)h^u_{11}\vec{u} \;-\; (h^w_{1\text{-}1} + \delta_2 h^w_{\text{-}1\text{-}1})\vec{w} &= 0 \\ (1-\beta\delta_3)h^v_{11}\vec{v} \;+\; (h^w_{11} + \delta_3 h^w_{1\text{-}1})\vec{w} &= 0 \\ (1-\beta\delta_4)h^v_{11}\vec{v} \;+\; (h^w_{\text{-}11} + \delta_4 h^w_{\text{-}1\text{-}1})\vec{w} &= 0 \end{aligned}$$

Hence, introducing the following constraints on the w element sizes:

$$\begin{aligned} &h^w_{xx} = h^u_{xx} h^v_{xx} \\ &\text{with } xx = [11,\ 1\text{-}1,\ \text{-}11,\ \text{-}1\text{-}1], \end{aligned} \tag{12}$$

and using again (9) and (10), the above simplifies to give

$$\begin{aligned} (1-\alpha\delta_1)h^u_{11}\vec{u} \;+\; h^u_{11}h^v_{11}(1-\alpha\delta_1)\vec{w} &= 0 \\ (1-\alpha\delta_2)h^u_{11}\vec{u} \;-\; \beta h^u_{11}h^v_{11}(1-\alpha\delta_2)\vec{w} &= 0 \\ (1-\beta\delta_3)h^v_{11}\vec{v} \;+\; h^u_{11}h^u_{11}(1-\beta\delta_3)\vec{w} &= 0 \\ (1-\beta\delta_4)h^v_{11}\vec{v} \;-\; \alpha h^u_{11}h^u_{11}(1-\beta\delta_4)\vec{w} &= 0 \end{aligned}$$

which is trivially satisfied for $\delta_1 = \delta_2 = \alpha^{-1}$ and $\delta_3 = \delta_4 = \beta^{-1}$. Finally, G1-continuity is ensured at the points $\boldsymbol{P}_{20}$, $\boldsymbol{P}_{02}$, $\boldsymbol{P}_{\text{-}20}$ and $\boldsymbol{P}_{0\text{-}2}$ by aligning the points

$$\begin{aligned} (\boldsymbol{P}_{12} - \boldsymbol{P}_{02}) &= -\lambda_1(\boldsymbol{P}_{\text{-}12} - \boldsymbol{P}_{02}) \quad \text{with} \quad \lambda_1 > 0 \\ (\boldsymbol{P}_{1\text{-}2} - \boldsymbol{P}_{0\text{-}2}) &= -\lambda_2(\boldsymbol{P}_{\text{-}1\text{-}2} - \boldsymbol{P}_{0\text{-}2}) \quad \text{with} \quad \lambda_2 > 0 \\ (\boldsymbol{P}_{21} - \boldsymbol{P}_{20}) &= -\lambda_3(\boldsymbol{P}_{2\text{-}1} - \boldsymbol{P}_{20}) \quad \text{with} \quad \lambda_3 > 0 \\ (\boldsymbol{P}_{\text{-}21} - \boldsymbol{P}_{\text{-}20}) &= -\lambda_4(\boldsymbol{P}_{\text{-}2\text{-}1} - \boldsymbol{P}_{\text{-}20}) \quad \text{with} \quad \lambda_4 > 0 \end{aligned} \tag{13}$$

The constraints to be imposed here are

$$\begin{aligned} h^n_{-11} &= -\alpha\, h^n_{11} \quad \text{and} \quad h^n_{-1\text{-}1} = -\alpha\, h^n_{1\text{-}1} \\ h^m_{1\text{-}1} &= -\beta\, h^m_{11} \quad \text{and} \quad h^m_{-1\text{-}1} = -\beta\, h^m_{-11} \end{aligned} \tag{14}$$

Indeed, as before, substituting the control points definitions from (7), (8), and using the constraints (9), (10), (12) and (14), gives

$$\begin{aligned} h^u_{11}(1-\alpha\lambda_1)\vec{u} \;+\; 2h^u_{11}h^v_{11}(1-\alpha\lambda_1)\vec{w} \;+\; h^n_{11}(1-\alpha\lambda_1)\vec{n} &= 0 \\ h^u_{11}(1-\alpha\lambda_2)\vec{u} \;+\; -2\beta h^u_{11}h^v_{11}(1-\alpha\lambda_2)\vec{w} \;+\; h^n_{1\text{-}1}(1-\alpha\lambda_2)\vec{n} &= 0 \\ h^v_{11}(1-\beta\lambda_3)\vec{v} \;+\; 2h^u_{11}h^v_{11}(1-\beta\lambda_3)\vec{w} \;+\; h^m_{11}(1-\beta\lambda_3)\vec{m} &= 0 \\ h^v_{11}(1-\beta\lambda_4)\vec{v} \;+\; -2\alpha h^u_{11}h^v_{11}(1-\beta\lambda_4)\vec{w} \;+\; h^m_{-11}(1-\beta\lambda_4)\vec{m} &= 0 \end{aligned}$$

which holds for $\lambda_1 = \lambda_2 = \alpha^{-1}$ and $\lambda_3 = \lambda_4 = \beta^{-1}$.

2.7. G2-continuity

As for G1-continuity, the G2-continuity constraints is first considered at the node itself, where the control points must satisfy the conditions

$$\begin{aligned} \boldsymbol{P}_{00} + \boldsymbol{P}_{20} - 2\,\boldsymbol{P}_{10} &= \boldsymbol{P}_{00} + \boldsymbol{P}_{\text{-}20} - 2\,\boldsymbol{P}_{\text{-}10} \\ \boldsymbol{P}_{00} + \boldsymbol{P}_{02} - 2\,\boldsymbol{P}_{01} &= \boldsymbol{P}_{00} + \boldsymbol{P}_{0\text{-}2} - 2\,\boldsymbol{P}_{0\text{-}1} \end{aligned} \tag{15}$$

which can be expanded, using (7), (8) and the G0- and G1-constraints derived above, to give

$$\begin{aligned} h^i_{11}\vec{i} &= h^i_{-1\text{-}1}\vec{i} \\ h^j_{11}\vec{j} &= h^j_{-1\text{-}1}\vec{j} \end{aligned} \tag{16}$$

implying that the G0-constraint (9) must be extended to

$$\begin{aligned} h^i_{1\text{-}1} = h^i_{-11} = h^i_{-1\text{-}1} = h^i_{11} \\ h^j_{1\text{-}1} = h^j_{-11} = h^j_{-1\text{-}1} = h^j_{11} \end{aligned} \tag{17}$$

Now, consider the G2-continuity conditions at the four control points adjacent to $\boldsymbol{P}_{00}$, namely

$$\begin{aligned} \boldsymbol{P}_{01} + \boldsymbol{P}_{21} - 2\,\boldsymbol{P}_{11} &= \boldsymbol{P}_{01} + \boldsymbol{P}_{\text{-}21} - 2\,\boldsymbol{P}_{\text{-}11} \\ \boldsymbol{P}_{0\text{-}1} + \boldsymbol{P}_{2\text{-}1} - 2\,\boldsymbol{P}_{1\text{-}1} &= \boldsymbol{P}_{0\text{-}1} + \boldsymbol{P}_{\text{-}2\text{-}1} - 2\,\boldsymbol{P}_{\text{-}1\text{-}1} \\ \boldsymbol{P}_{10} + \boldsymbol{P}_{12} - 2\,\boldsymbol{P}_{11} &= \boldsymbol{P}_{10} + \boldsymbol{P}_{1\text{-}2} - 2\,\boldsymbol{P}_{1\text{-}1} \\ \boldsymbol{P}_{\text{-}10} + \boldsymbol{P}_{\text{-}12} - 2\,\boldsymbol{P}_{\text{-}11} &= \boldsymbol{P}_{\text{-}10} + \boldsymbol{P}_{\text{-}1\text{-}2} - 2\,\boldsymbol{P}_{\text{-}1\text{-}1} \end{aligned} \tag{18}$$

which, once expanded, gives only the remaining terms

$$\begin{aligned} h^m_{11}\vec{m} &= h^m_{-11}\vec{m} \\ h^m_{1\text{-}1}\vec{m} &= h^m_{-1\text{-}1}\vec{m} \\ h^n_{11}\vec{n} &= h^n_{1\text{-}1}\vec{n} \\ h^n_{-11}\vec{n} &= h^n_{-1\text{-}1}\vec{n} \end{aligned} \tag{19}$$

Hence, the set of constraints (14) must be extended to

$$\begin{aligned} h^n_{-11} = h^n_{-1\text{-}1} = -\alpha\, h^n_{11} = -\alpha\, h^n_{1\text{-}1} \\ h^m_{1\text{-}1} = h^m_{-1\text{-}1} = -\beta\, h^m_{11} = -\beta\, h^m_{-11} \end{aligned} \tag{20}$$

At last, the G2-continuity at the remaining nodes is expressed as

$$\begin{aligned} \boldsymbol{P}_{02} + \boldsymbol{P}_{22} - 2\,\boldsymbol{P}_{12} &= \boldsymbol{P}_{02} + \boldsymbol{P}_{\text{-}22} - 2\,\boldsymbol{P}_{\text{-}12} \\ \boldsymbol{P}_{0\text{-}2} + \boldsymbol{P}_{2\text{-}2} - 2\,\boldsymbol{P}_{1\text{-}2} &= \boldsymbol{P}_{0\text{-}2} + \boldsymbol{P}_{\text{-}2\text{-}2} - 2\,\boldsymbol{P}_{\text{-}1\text{-}2} \\ \boldsymbol{P}_{20} + \boldsymbol{P}_{22} - 2\,\boldsymbol{P}_{21} &= \boldsymbol{P}_{20} + \boldsymbol{P}_{2\text{-}2} - 2\,\boldsymbol{P}_{2\text{-}1} \\ \boldsymbol{P}_{\text{-}20} + \boldsymbol{P}_{\text{-}22} - 2\,\boldsymbol{P}_{\text{-}21} &= \boldsymbol{P}_{\text{-}20} + \boldsymbol{P}_{\text{-}2\text{-}2} - 2\,\boldsymbol{P}_{\text{-}2\text{-}1} \end{aligned} \tag{21}$$

which, once expanded, gives the remaining terms

$$\begin{aligned} h^k_{11}\vec{k} &= h^k_{-11}\vec{k} \\ h^k_{1\text{-}1}\vec{k} &= h^k_{-1\text{-}1}\vec{k} \\ h^k_{11}\vec{k} &= h^k_{1\text{-}1}\vec{k} \\ h^k_{-11}\vec{k} &= h^k_{-1\text{-}1}\vec{k} \end{aligned} \tag{22}$$

Now, this leads to the final constraint on the element size for the $\vec{k}$ vectors:

$$\begin{aligned} &h^k_{xx} = h^i_{xx} h^j_{xx} \\ &\text{with } xx = [11,\ 1\text{-}1,\ \text{-}11,\ \text{-}1\text{-}1], \end{aligned} \tag{23}$$

Using this rule, together with (17), the above is trivially satisfied.

For convenience, the G2-continuity constraints are summarised here:

$$h^u_{-1\text{-}1} = h^u_{-11} = -\alpha\, h^u_{1\text{-}1} = -\alpha\, h^u_{11} \tag{24}$$

$$h^v_{-1\text{-}1} = h^v_{1\text{-}1} = -\beta\, h^v_{-11} = -\beta\, h^v_{11} \tag{25}$$

$$h^i_{1\text{-}1} = h^i_{11} = h^i_{-1\text{-}1} = h^i_{-11} \tag{26}$$

$$h^j_{-11} = h^j_{11} = h^j_{-1\text{-}1} = h^j_{1\text{-}1} \tag{27}$$

$$h^m_{1\text{-}1} = h^m_{-1\text{-}1} = -\beta\, h^m_{11} = -\beta\, h^m_{-11} \tag{28}$$

$$h^n_{-11} = h^n_{-1\text{-}1} = -\alpha\, h^n_{11} = -\alpha\, h^n_{1\text{-}1} \tag{29}$$

$$h^w_{xx} = h^u_{xx} h^v_{xx} \tag{30}$$

$$h^k_{xx} = h^i_{xx} h^j_{xx} \tag{31}$$

$$\text{with } xx = [11,\ 1\text{-}1,\ \text{-}11,\ \text{-}1\text{-}1],$$

3. Generalised formulation: beyond G2 continuity

The nodal formulation (5), and the element-size constraints ((24)-(31)) can be generalised to provide Bezier elements of arbitrary odd order n, with G-continuity of order $(n-1)/2$. This generalisation requires the definition of a nodal formulation at any odd order n. The order n must be odd such that all nodes have the same number of associated control points. Note that continuity of order $(n-1)/2$ is obtained since in a given direction, there are $(n+1)$ control points, including the end nodes, hence $(n+1)/2$ control points for each node, one being the value of a variable, and then one for each derivative, such that $(n-1)/2$ derivatives can be controlled. At each node, there are $(n+1)^2/4$ degrees of freedom. Note that this is specific to the 2D case: in 3D, with eight elements sharing a common node, there would be $[(n+1)/2]^3$ degrees of freedom.

3.1. Notation for the derivatives of control points

Before defining the general formulation, it helps to understand the structure of G-continuity requirements.

Notation-1. G1-continuity

Let direction x denote the first index of control points $\boldsymbol{P}_{ij}$, and direction y the second index. The first derivative, in the direction x, at a border point $\boldsymbol{P}_{0j}$ is defined by

$$\Delta^1_x \boldsymbol{P}_{0j} = \boldsymbol{P}_{0j} - \boldsymbol{P}_{1j},\ \text{for any j.} \tag{32}$$

On the opposite element, it is defined as

$$\Delta^{-1}_x \boldsymbol{P}_{0j} = \boldsymbol{P}_{0j} - \boldsymbol{P}_{\text{-}1j},\ \text{for any j.} \tag{33}$$

Likewise, the first derivative in the y direction is defined by

$$\Delta^1_y \boldsymbol{P}_{i0} = \boldsymbol{P}_{i0} - \boldsymbol{P}_{i1},\ \text{for any i} \tag{34}$$

and, on the opposite element,

$$\Delta^{-1}_y \boldsymbol{P}_{i0} = \boldsymbol{P}_{i0} - \boldsymbol{P}_{i\text{-}1},\ \text{for any i.} \tag{35}$$

The four parent elements of node $\boldsymbol{P}_{00}$ are G1-continuous provided

$$\Delta_x^{-1}\,\boldsymbol{P}_{0j} = -\lambda_1 \Delta_x^{1}\,\boldsymbol{P}_{0j}, \tag{36}$$

$$\Delta_y^{-1}\,\boldsymbol{P}_{i0} = -\lambda_2 \Delta_y^{1}\,\boldsymbol{P}_{i0}, \tag{37}$$

for any indices i, j in interval $[-(n\text{-}1)/2, (n\text{-}1)/2]$, and any positive real scalars λ_1, λ_2.

Notation-2. G-continuity of order *m*
The m-th derivative, in direction x, at point $\boldsymbol{P}_{0j}$, is defined by

$$\Delta_x^{m}\,\boldsymbol{P}_{0j} = \Delta_x^{m\text{-}1}\,\boldsymbol{P}_{0j} - \Delta_x^{m\text{-}1}\,\boldsymbol{P}_{1j} \tag{38}$$

Just like standard derivatives, it is trivial to demonstrate by induction that this is equivalent to

$$\Delta_x^{m}\,\boldsymbol{P}_{0j} = \sum_{i=0}^{m} (-1)^i \binom{m}{i} \boldsymbol{P}_{ij} \tag{39}$$

Likewise, on the opposite element, the m-th derivative can be written as

$$\Delta_x^{\text{-}m}\,\boldsymbol{P}_{0j} = \sum_{i=0}^{m} (-1)^i \binom{m}{i} \boldsymbol{P}_{\text{-}ij} \tag{40}$$

Similarly, at point $\boldsymbol{P}_{i0}$, the m-th derivatives, in direction y, on two opposite elements, are defined as

$$\Delta_y^{m}\,\boldsymbol{P}_{i0} = \sum_{j=0}^{m} (-1)^j \binom{m}{j} \boldsymbol{P}_{ij} \tag{41}$$

$$\Delta_y^{\text{-}m}\,\boldsymbol{P}_{i0} = \sum_{j=0}^{m} (-1)^j \binom{m}{j} \boldsymbol{P}_{i\text{-}j} \tag{42}$$

Unlike the G1-continuity requirement ((36)-(37)), G-continuity of order m requires equality of the m-th derivatives on either sides of a node, namely

$$\Delta_x^{m}\,\boldsymbol{P}_{0j} = \Delta_x^{-m}\,\boldsymbol{P}_{0j}, \tag{43}$$

$$\Delta_y^{m}\,\boldsymbol{P}_{i0} = \Delta_y^{-m}\,\boldsymbol{P}_{i0}, \tag{44}$$

It is important to note that we use a sub-case version of G-continuity definition. Generic G1- and G2-continuity definitions can be found in [52,53] respectively. In the most general case, G1-continuity would not require derivatives to be collinear, but in this particular sub-case, it does. Likewise, in the generic case, G2-continuity involves both the second, first and zero derivatives of control points, but in this study, we only consider the second derivatives for the G2-continuity constraint (which does not mean that lower derivatives are not continuous). The main advantage of the more generic continuity definition is that they allow the degrees of freedom to be shifted from the control points to the polynomial representation of boundaries between elements. One of the main outcomes of such a generic method is that it allows control points to be "misaligned", enabling strong angles between elements at a given node. For example, it would allow G1-continuity without requiring the points $\boldsymbol{P}_4$, $\boldsymbol{Q}_0$ and $\boldsymbol{Q}_1$ in the bottom Fig.-3 to be collinear. Such a generic method is, however, beyond the scope of this paper and should be considered in future studies.

The notation of control point derivatives (39), (41), together with the continuity constraints (43), (44) can be derived simply by evaluating the derivative of the Bezier formula (3) at $s = 0$ or $s = 1$ (or $t = 0$ and $t = 1$). This derivation is not detailed here, but its method can be seen in [22] for G1-continuity, and is identical for higher-order derivatives.

Now, consider the generic Bezier 1D-derivative identity (39) as a function of the control points. The ideal nodal formulation to satisfy these conditions is one that automatically simplifies the derivative at any order. This should be, of course, in both directions i and j. As will become clear below, the generic formulation is nothing less than a construction similar to mixed 2D-derivatives.

3.2. Generalised nodal formulation of Bezier elements

Definition-1. For Bezier elements of any odd polynomial order n, the nodal formulation of control points $\boldsymbol{P}_{ij}$ and vectors $\vec{u}^{\,ij}$ is defined as

$$\boldsymbol{P}_{ij} = h^{ij}\vec{u}^{\,ij} + \sum_{k=0}^{i}\sum_{l=0}^{j}(-1)^{1+i+j+k+l}(1 - \delta_{ki}\delta_{lj})\binom{i}{k}\binom{j}{l}\boldsymbol{P}_{kl} \tag{45}$$

where

$0 \le i < (n+1)/2$

$0 \le j < (n+1)/2$

δ_{ab} is the Kronecker delta

h^{ij} are the sizes of each component of the nodal vectorial basis.

$\boldsymbol{P}_{00} = h^{00}\vec{u}^{\,00} := \boldsymbol{P}_{00}$

Definition-2. The notation of the element sizes for the 4 parent elements of a node are expressed as h^{ij}, h^{-ij}, h^{i-j} and h^{-i-j}, and they are constrained with the following rules

for any j, h_{-ij} is constrained by

$$h^{-ij} = \begin{cases} -\alpha\, h^{ij} & \text{for } i = 1 \text{ and } \alpha > 0 \\ h^{ij} & \text{for } i \neq 1 \end{cases} \tag{46}$$

for any i, h_{i-j} is constrained by

$$h^{i-j} = \begin{cases} -\beta\, h^{ij} & \text{for } j = 1 \text{ and } \beta > 0 \\ h^{ij} & \text{for } j \neq 1 \end{cases} \tag{47}$$

Note that Definition-1 is clearly just a mixture of the derivatives (39) in two dimensions. The Kronecker-delta functions are simply required to remove the re-occurrence of the point $\boldsymbol{P}_{ij}$ in the double-sum term. Also, note that formula (45) can be used, recursively, to define either the points $\boldsymbol{P}_{ij}$ given the vectors $\vec{u}^{ij}$, or inversely to define the vectors $\vec{u}^{ij}$ given the points $\boldsymbol{P}_{ij}$. Using these two definitions, it can be demonstrated that G-continuity is ensured at order $(n-1)/2$. Before this can be achieved, some necessary properties of the definitions must first be obtained.

3.3. G1- and G2-continuity demonstration

The final proof of G-continuity is done by induction, and therefore G1- and G2-continuity must first be proven.

Proposition-1. *The nodal formulation $\boldsymbol{P}_{ij}$ defined in Definition-1, together with the element size constraints defined in Definition-2, ensures G1-continuity between Bezier elements.*

Proof. Consider G1-continuity in the x direction. G1-continuity at the node $\boldsymbol{P}_{00}$ itself (i.e. at $j = 0$) is trivial, since expanding the nodal formulation (45) at points $\boldsymbol{P}_{10}$ and $\boldsymbol{P}_{-10}$ gives

$$\boldsymbol{P}_{10} = \boldsymbol{P}_{00} + h^{10}\vec{u}^{10} \tag{48}$$

$$\boldsymbol{P}_{-10} = \boldsymbol{P}_{00} + h^{-10}\vec{u}^{10} \tag{49}$$

where of course, by definition, the nodal formulation implies that $\vec{u}^{\,-10} = \vec{u}^{10}$. The above can be rearranged and written as

$$\Delta_x^1\,\boldsymbol{P}_{00} = -h^{10}\vec{u}^{10} \tag{50}$$

$$\Delta_x^{-1}\,\boldsymbol{P}_{00} = -h^{-10}\vec{u}^{10} = \alpha h^{10}\vec{u}^{10} = -\alpha\,\Delta_x^1\,\boldsymbol{P}_{00} \tag{51}$$

where rule (46) of Definition-2 was used. Thus, the G1-continuity rule (36) is satisfied with $\lambda_1 = \alpha$. Similarly, at $j = 1$, points $\boldsymbol{P}_{11}$ and $\boldsymbol{P}_{-11}$ can be expressed, using (45), as

$$\boldsymbol{P}_{11} = h^{11}\vec{u}^{11} - \boldsymbol{P}_{00} + \boldsymbol{P}_{01} + \boldsymbol{P}_{10} = h^{11}\vec{u}^{11} - \Delta_x^1\,\boldsymbol{P}_{00} + \boldsymbol{P}_{01} \tag{52}$$

$$\boldsymbol{P}_{-11} = h^{-11}\vec{u}^{11} - \boldsymbol{P}_{00} + \boldsymbol{P}_{01} + \boldsymbol{P}_{-10} = -\alpha h^{11}\vec{u}^{11} - \Delta_x^{-1}\,\boldsymbol{P}_{00} + \boldsymbol{P}_{01} \tag{53}$$

where rule (46) of Definition-2 was used. However, G1-continuity at $\boldsymbol{P}_{00}$ already provides an expression for $\Delta_x^{-1}\,\boldsymbol{P}_{00}$ in (51), such that the above can be rearranged as

$$\Delta_x^1 \boldsymbol{P}_{01} = -h^{11}\vec{u}^{11} + \Delta_x^1 \boldsymbol{P}_{00} \tag{54}$$

$$\Delta_x^{-1} \boldsymbol{P}_{01} = +\alpha h^{11}\vec{u}^{11} - \alpha\Delta_x^1 \boldsymbol{P}_{00} \tag{55}$$

Thus, the G1-continuity rule (36) is satisfied at $\boldsymbol{P}_{01}$ with $\lambda_1 = \alpha$.

We now proceed by induction. Assume G1-continuity holds at point $\boldsymbol{P}_{0,(j-1)}$, such that $\Delta_x^{-1}\boldsymbol{P}_{0l} = -\alpha\Delta_x^1\boldsymbol{P}_{0l}$ for all $l \le j-1$. Then, expressing $\boldsymbol{P}_{1j}$ using the nodal formulation (45) gives

$$\begin{aligned}
\boldsymbol{P}_{1j} &= h^{1j}\vec{u}^{1j} + \sum_{k=0}^{1}\sum_{l=0}^{j}(-1)^{j+k+l}(1-\delta_{k1}\delta_{lj})\binom{j}{l}\boldsymbol{P}_{kl} \\
&= h^{1j}\vec{u}^{1j} + \sum_{l=0}^{j}(-1)^{j+l}\binom{j}{l}\boldsymbol{P}_{0l} + \sum_{l=0}^{j}(-1)^{1+j+l}(1-\delta_{lj})\binom{j}{l}\boldsymbol{P}_{1l} \\
&= h^{1j}\vec{u}^{1j} + \boldsymbol{P}_{0j} + \sum_{l=0}^{j-1}(-1)^{j+l}\binom{j}{l}\boldsymbol{P}_{0l} + \sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}\boldsymbol{P}_{1l} \\
&= h^{1j}\vec{u}^{1j} + \boldsymbol{P}_{0j} + \sum_{l=0}^{j-1}(-1)^{j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l}) \\
&= h^{1j}\vec{u}^{1j} + \boldsymbol{P}_{0j} + \sum_{l=0}^{j-1}(-1)^{j+l}\binom{j}{l}\Delta_x^1\boldsymbol{P}_{0l}
\end{aligned} \tag{56}$$

Similarly, point $\boldsymbol{P}_{-1j}$ is expanded to give

$$\begin{aligned}
\boldsymbol{P}_{-1j} &= h^{-1j}\vec{u}^{1j} + \boldsymbol{P}_{0j} + \sum_{l=0}^{j-1}(-1)^{j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{-1l}) \\
&= -\alpha h^{1j}\vec{u}^{1j} + \boldsymbol{P}_{0j} + \sum_{l=0}^{j-1}(-1)^{j+l}\Delta_x^{-1}\boldsymbol{P}_{0l}
\end{aligned} \tag{57}$$

where rule (46) of Definition-2 was used. However, since the sum term in (57) is only up to $j-1$, G1-continuity applies for $\Delta_x^{-1}\boldsymbol{P}_{0l}$ by assumption, which can be replaced by $-\alpha\Delta_x^1\boldsymbol{P}_{0l}$. Therefore, (56) and (57) can be rearranged to write

$$\Delta_x^1 \boldsymbol{P}_{0j} = -h^{1j}\vec{u}^{1j} - \sum_{l=0}^{j-1}(-1)^{j+l}\binom{j}{l}\Delta_x^1\boldsymbol{P}_{0l} \tag{58}$$

$$\Delta_x^{-1} \boldsymbol{P}_{0j} = +\alpha h^{1j}\vec{u}^{1j} + \alpha\sum_{l=0}^{j-1}(-1)^{j+l}\binom{j}{l}\Delta_x^1\boldsymbol{P}_{0l} \tag{59}$$

In other words, the G1-continuity rule (36) is satisfied at $\boldsymbol{P}_{0j}$ with $\lambda_1 = \alpha$. Hence, by induction, since we know that G1-continuity is valid at point $\boldsymbol{P}_{01}$, it follows that G1-continuity is valid at all points $\boldsymbol{P}_{0j}$ for $j \le (n-1)/2$. The demonstration for negative j is almost identical. Likewise, the above demonstration for G1-continuity in the y direction, at any point $\boldsymbol{P}_{i0}$, is also identical, by swapping indices. This completes the proof, that the nodal formulation of Definition-1 and Definition-2 satisfies G1-continuity between Bezier elements. □

Proposition-2. *The nodal formulation of $\boldsymbol{P}_{ij}$ defined in Definition-1, together with the element size constraints defined in Definition-2, ensures G2-continuity between Bezier elements.*

Proof. As with the G1-continuity proof, consider G2-continuity in the x direction. Expand point $\boldsymbol{P}_{20}$ using the nodal formulation (45),

$$\begin{aligned}
\boldsymbol{P}_{20} &= h^{20}\vec{u}^{20} + \sum_{k=0}^{2}(-1)^{1+k}(1-\delta_{k2})\binom{2}{k}\boldsymbol{P}_{k0} \\
&= h^{20}\vec{u}^{20} - \boldsymbol{P}_{00} + 2\boldsymbol{P}_{10}
\end{aligned} \tag{60}$$

However, the second derivative is defined as $\Delta_x^2\boldsymbol{P}_{00} = \boldsymbol{P}_{00} - 2\boldsymbol{P}_{10} + \boldsymbol{P}_{20}$, so that the above can be rearranged as

$$\Delta_x^2 \boldsymbol{P}_{00} = h^{20}\vec{u}^{20}$$

Likewise, on the opposite side,

$$\begin{aligned}\Delta_x^{-2} \boldsymbol{P}_{00} &= h^{-20}\vec{u}^{20} \\ &= h^{20}\vec{u}^{20}\end{aligned}$$

where rule (46) of Definition-2 was used. Thus, $\Delta_x^2 \boldsymbol{P}_{00} = \Delta_x^{-2} \boldsymbol{P}_{00}$, so that G2-continuity is satisfied at $\boldsymbol{P}_{00}$.

Next, we proceed by induction. Assume G2-continuity holds at point $\boldsymbol{P}_{0,(j-1)}$, such that $\Delta_x^2 \boldsymbol{P}_{0l} = \Delta_x^{-2} \boldsymbol{P}_{0l}$ for all $l \leq j-1$. Then, expressing $\boldsymbol{P}_{2j}$ using the nodal formulation,

$$\begin{aligned}\boldsymbol{P}_{2j} &= h^{2j}\vec{u}^{2j} + \sum_{k=0}^{2}\sum_{l=0}^{j}(-1)^{1+j+k+l}(1-\delta_{k2}\delta_{lj})\binom{2}{k}\binom{j}{l}\boldsymbol{P}_{kl} \\ &= h^{2j}\vec{u}^{2j} + \sum_{l=0}^{j}(-1)^{1+j+l}\binom{j}{l}\boldsymbol{P}_{0l} + 2\sum_{l=0}^{j}(-1)^{j+l}\binom{j}{l}\boldsymbol{P}_{1l} + \sum_{l=0}^{j}(-1)^{1+j+l}(1-\delta_{lj})\binom{j}{l}\boldsymbol{P}_{2l} \\ &= h^{2j}\vec{u}^{2j} - \boldsymbol{P}_{0j} + 2\boldsymbol{P}_{1j} \\ &\quad + \sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}\boldsymbol{P}_{0l} + 2\sum_{l=0}^{j-1}(-1)^{j+l}\binom{j}{l}\boldsymbol{P}_{1l} + \sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}\boldsymbol{P}_{2l}\end{aligned} \tag{61}$$

which can simply be rearranged as

$$\begin{aligned}\Delta_x^2 \boldsymbol{P}_{0j} &= h^{2j}\vec{u}^{2j} + \sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - 2\boldsymbol{P}_{1l} + \boldsymbol{P}_{2l}) \\ &= h^{2j}\vec{u}^{2j} + \sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}\Delta_x^2 \boldsymbol{P}_{0l}\end{aligned} \tag{62}$$

Likewise, on the opposite side,

$$\begin{aligned}\Delta_x^{-2} \boldsymbol{P}_{0j} &= h^{-2j}\vec{u}^{2j} + \sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - 2\boldsymbol{P}_{-1l} + \boldsymbol{P}_{-2l}) \\ &= h^{2j}\vec{u}^{2j} + \sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}\Delta_x^{-2} \boldsymbol{P}_{0l}\end{aligned} \tag{63}$$

where rule (46) of Definition-2 was used for $h^{-2j} = h^{2j}$. However, since G2-continuity holds for $l \leq j-1$, the sum term in (63) is equivalent to the sum term in (62), such that $\Delta_x^2 \boldsymbol{P}_{0j} = \Delta_x^{-2} \boldsymbol{P}_{0j}$. Hence, by induction, G2-continuity in the x direction is satisfied for all j between 0 and $(n-1)/2$. For negative j, the demonstration is identical. Likewise, in the y direction, the demonstration is identical, by switching indices. □

3.4. G-continuity demonstration

Proposition-3. *If the Bezier elements are G-continuous at order m, in direction x, then for any index i such that* $1 < i \leq m$, *the following identity holds,*

$$\boldsymbol{P}_{ij} - \boldsymbol{P}_{-ij} = -i\,(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j}) \tag{64}$$

for any index j, and where α *is the positive scalar of the G1-continuity rule in the x direction from (46).*

Proof. First, consider G1-continuity at point $\boldsymbol{P}_{0j}$, which implies that

$$\begin{aligned}&\Delta_x^{-1} \boldsymbol{P}_{0j} = -\alpha\, \Delta_x^1 \boldsymbol{P}_{0j} \\ \Longrightarrow\ &\boldsymbol{P}_{0j} - \boldsymbol{P}_{-1j} = -\alpha(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j})\end{aligned} \tag{65}$$

Then, consider G2-continuity at point $\boldsymbol{P}_{0j}$, so that

$$\begin{aligned}
&\Delta_x^2 \boldsymbol{P}_{0j} = \Delta_x^{-2} \boldsymbol{P}_{0j} \\
\implies &\boldsymbol{P}_{0j} - 2\boldsymbol{P}_{1j} + \boldsymbol{P}_{2j} = \boldsymbol{P}_{0j} - 2\boldsymbol{P}_{-1j} + \boldsymbol{P}_{-2j} \\
\implies &\boldsymbol{P}_{2j} - \boldsymbol{P}_{-2j} = 2\boldsymbol{P}_{1j} - 2\boldsymbol{P}_{-1j} \\
\implies &\boldsymbol{P}_{2j} - \boldsymbol{P}_{-2j} = 2(\boldsymbol{P}_{1j} - \boldsymbol{P}_{0j}) - 2(\boldsymbol{P}_{-1j} - \boldsymbol{P}_{0j})
\end{aligned} \tag{66}$$

Substituting for (65) into (66) gives

$$\begin{aligned}
&\boldsymbol{P}_{2j} - \boldsymbol{P}_{-2j} = 2(\boldsymbol{P}_{1j} - \boldsymbol{P}_{0j}) + 2\alpha(\boldsymbol{P}_{-1j} - \boldsymbol{P}_{0j}) \\
\implies &\boldsymbol{P}_{2j} - \boldsymbol{P}_{-2j} = -2(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j})
\end{aligned} \tag{67}$$

Next, we proceed by induction. Assume that identity (64) holds for all values of $i-1$ up to some $i-1 < m$. Then, since G-continuity holds at order i, the following identity can be written,

$$\begin{aligned}
&\Delta_x^i \boldsymbol{P}_{0j} = \Delta_x^{-i} \boldsymbol{P}_{0j} \\
\implies &\sum_{k=0}^{i} (-1)^k \binom{i}{k} \boldsymbol{P}_{kj} = \sum_{k=0}^{i} (-1)^k \binom{i}{k} \boldsymbol{P}_{-kj} \\
\implies &(-1)^i \boldsymbol{P}_{ij} + \sum_{k=0}^{i-1} (-1)^k \binom{i}{k} \boldsymbol{P}_{kj} = (-1)^i \boldsymbol{P}_{-ij} + \sum_{k=0}^{i-1} (-1)^k \binom{i}{k} \boldsymbol{P}_{-kj} \\
\implies &\boldsymbol{P}_{ij} - \boldsymbol{P}_{-ij} = -(-1)^i \sum_{k=0}^{i-1} (-1)^k \binom{i}{k} (\boldsymbol{P}_{kj} - \boldsymbol{P}_{-kj})
\end{aligned} \tag{68}$$

However, since the right-hand side sum term only goes up to $i-1$, by assumption, we can apply identity (64), to write (68) as

$$\boldsymbol{P}_{ij} - \boldsymbol{P}_{-ij} = +(-1)^i \sum_{k=0}^{i-1} (-1)^k \binom{i}{k} k(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j}) \tag{69}$$

Now, most of the terms in the right-hand side sum term are independent of the summation index k, so they can simply be extracted from the sum, such that (69) becomes

$$\boldsymbol{P}_{ij} - \boldsymbol{P}_{-ij} = +(-1)^i(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j}) \sum_{k=0}^{i-1} (-1)^k \binom{i}{k} k \tag{70}$$

The sum term is now close to a known binomial identity [55], which states that

$$\begin{aligned}
&\sum_{k=0}^{i} (-1)^k \binom{i}{k} k = 0 \\
\implies &\sum_{k=0}^{i-1} (-1)^k \binom{i}{k} k = -(-1)^i i
\end{aligned} \tag{71}$$

Hence, (70) becomes

$$\begin{aligned}
\boldsymbol{P}_{ij} - \boldsymbol{P}_{-ij} &= -(-1)^{2i} i(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j}) \\
&= -i(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j})
\end{aligned} \tag{72}$$

By induction, since identity (64) was shown for $i=2$, this completes the proof up to $i=m$. □

Theorem- 1. *With the nodal formulation of $\boldsymbol{P}_{ij}$ defined in Definition-1, together with the element size constraints defined in Definition-2, Bezier elements are G-continuous at order $(n-1)/2$ for any value of the vectors $\vec{u}^{ij}$.*

Proof. This proof is a little intricate, because it contains a proof-by-induction within a proof-by-induction. Since Proposition-2 already shows that G-continuity holds at order $=2$, the first induction is on the order of the G-continuity, proving that if G-continuity holds for $i-1$, then it also does for i. However, this procedure is also done by induction for the second index j of the point $\boldsymbol{P}_{0j}$, showing that if G-continuity of order i holds for $\boldsymbol{P}_{0,(j-1)}$, then it also does for $\boldsymbol{P}_{0j}$.

Hence, assume that G-continuity holds up to order $i-1$ for some $2 \leq i-1 < (n-1)/2$. Consider the G-continuity requirement, in direction x, at order i. Exactly like the above (68), this gives

$$\begin{aligned} &\Delta_x^i \boldsymbol{P}_{0j} = \Delta_x^{-i} \boldsymbol{P}_{0j} \\ \implies &\boldsymbol{P}_{ij} - \boldsymbol{P}_{\text{-}ij} = -(-1)^i \sum_{k=0}^{i-1} (-1)^k \binom{i}{k} (\boldsymbol{P}_{kj} - \boldsymbol{P}_{\text{-}kj}) \end{aligned} \tag{73}$$

Again, since the right-hand side sum term only goes up to $i-1$, by assumption, we can apply identity (64), to write (73) as

$$\boldsymbol{P}_{ij} - \boldsymbol{P}_{\text{-}ij} = +(-1)^i \sum_{k=0}^{i-1} (-1)^k \binom{i}{k} k(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j}) \tag{74}$$

which, as in the demonstration of Proposition-3, gives

$$\boldsymbol{P}_{ij} - \boldsymbol{P}_{\text{-}ij} = -i\,(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j}) \tag{75}$$

Thus, in order to prove that G-continuity holds at order i, we need to show that the above identity (75) is true. First, consider $\boldsymbol{P}_{ij} - \boldsymbol{P}_{\text{-}ij}$ for $j=0$, by expanding using the nodal formulation (45) of Definition-1,

$$\begin{aligned} \boldsymbol{P}_{i0} - \boldsymbol{P}_{\text{-}i0} &= \vec{u}^{i0}(h^{i0} - h^{\text{-}i0}) + \sum_{k=0}^{i} (-1)^{1+i+k}(1-\delta_{ki}) \binom{i}{k} (\boldsymbol{P}_{k0} - \boldsymbol{P}_{\text{-}k0}) \\ &= \sum_{k=0}^{i-1} (-1)^{1+i+k} \binom{i}{k} (\boldsymbol{P}_{k0} - \boldsymbol{P}_{\text{-}k0}) \end{aligned} \tag{76}$$

where rule (46) of Definition-2 was used for $h^{\text{-}i0} = h^{i0}$, for any $i > 1$. Now, since the sum term in (76) only goes up to $i-1$, we can use identity (64) to write it as

$$\boldsymbol{P}_{i0} - \boldsymbol{P}_{\text{-}i0} = -(-1)^{1+i}\,(\alpha+1)(\boldsymbol{P}_{00} - \boldsymbol{P}_{10}) \sum_{k=0}^{i-1} (-1)^k \binom{i}{k} k \tag{77}$$

which, using (71), is simply

$$\begin{aligned} \boldsymbol{P}_{i0} - \boldsymbol{P}_{\text{-}i0} &= (-1)^{1+2i} i\,(\alpha+1)(\boldsymbol{P}_{00} - \boldsymbol{P}_{10}) \\ &= -i\,(\alpha+1)(\boldsymbol{P}_{00} - \boldsymbol{P}_{10}) \end{aligned} \tag{78}$$

which demonstrates G-continuity for $j=0$. Next, consider $j=1$, and expand $\boldsymbol{P}_{i1} - \boldsymbol{P}_{\text{-}i1}$ using the nodal formulation (45) of Definition-1,

$$\begin{aligned} &\boldsymbol{P}_{i1} - \boldsymbol{P}_{\text{-}i1} \\ &= \cancel{\vec{u}^{i1}(h^{i1} - h^{\text{-}i1})} + \sum_{k=0}^{i} \sum_{l=0}^{1} (-1)^{i+k+l}(1-\delta_{ki}\delta_{l1}) \binom{i}{k} (\boldsymbol{P}_{kl} - \boldsymbol{P}_{\text{-}kl}) \\ &= \sum_{k=0}^{i} (-1)^{i+k} \binom{i}{k} (\boldsymbol{P}_{k0} - \boldsymbol{P}_{\text{-}k0}) + \sum_{k=0}^{i} (-1)^{1+i+k}(1-\delta_{ki}) \binom{i}{k} (\boldsymbol{P}_{k1} - \boldsymbol{P}_{\text{-}k1}) \\ &= \sum_{k=0}^{i} (-1)^{i+k} \binom{i}{k} (\boldsymbol{P}_{k0} - \boldsymbol{P}_{\text{-}k0}) + \sum_{k=0}^{i-1} (-1)^{1+i+k} \binom{i}{k} (\boldsymbol{P}_{k1} - \boldsymbol{P}_{\text{-}k1}) \\ &= (\boldsymbol{P}_{i0} - \boldsymbol{P}_{\text{-}i0}) + \cancel{\sum_{k=0}^{i-1} (-1)^{i+k} \binom{i}{k} (\boldsymbol{P}_{k0} - \boldsymbol{P}_{\text{-}k0})} + \cancel{\sum_{k=0}^{i-1} (-1)^{1+i+k} \binom{i}{k} (\boldsymbol{P}_{k1} - \boldsymbol{P}_{\text{-}k1})} \\ &= (\boldsymbol{P}_{i0} - \boldsymbol{P}_{\text{-}i0}) \end{aligned} \tag{79}$$

where the cancellation of the two sum terms is obvious since $(-1)^1 = -1$ can simply be extracted from the sum. The cancellation of $(h^{i1} - h^{\text{-}i1})$ comes from the definition of element sizes (46) for $i > 1$. Next, (78) implies that the above can be written as

$$\boldsymbol{P}_{i1} - \boldsymbol{P}_{\text{-}i1} = -i\,(\alpha+1)(\boldsymbol{P}_{00} - \boldsymbol{P}_{10}) \tag{80}$$

which demonstrates G-continuity for $j = 1$. Now, we proceed by induction. Assume that identity (75) holds up to some $j-1$. Then, expanding $\boldsymbol{P}_{ij} - \boldsymbol{P}_{\text{-}ij}$ with the nodal formulation (45) of Definition-1,

$$
\begin{aligned}
&\boldsymbol{P}_{ij} \;-\; \boldsymbol{P}_{\text{-}ij} \\
&= \cancel{\vec{u}^{ij}(h^{ij} = h^{-ij})} + \sum_{k=0}^{i}\sum_{l=0}^{j}(-1)^{1+i+j+k+l}(1-\delta_{ki}\delta_{lj})\binom{i}{k}\binom{j}{l}(\boldsymbol{P}_{kl} - \boldsymbol{P}_{\text{-}kl}) \\
&= \sum_{l=0}^{j}(-1)^{1+j+l}(1-\delta_{lj})\binom{j}{l}(\boldsymbol{P}_{il} - \boldsymbol{P}_{\text{-}il}) \;+\; \sum_{k=0}^{i-1}\sum_{l=0}^{j}(-1)^{1+i+j+k+l}\binom{i}{k}\binom{j}{l}(\boldsymbol{P}_{kl} - \boldsymbol{P}_{\text{-}kl}) \\
&= \sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{il} - \boldsymbol{P}_{\text{-}il}) \;+\; \sum_{k=0}^{i-1}\sum_{l=0}^{j}(-1)^{1+i+j+k+l}\binom{i}{k}\binom{j}{l}(\boldsymbol{P}_{kl} - \boldsymbol{P}_{\text{-}kl})
\end{aligned}
\tag{81}
$$

Now, by assumption, since the first term only goes up to $j-1$, identity (75) can be used. Likewise, by assumption, since we have assumed G-continuity up to $i-1$, identity (64) can be used for the second term. Thus, (81) can be written as

$$
\begin{aligned}
&\boldsymbol{P}_{ij} \;-\; \boldsymbol{P}_{\text{-}ij} \\
&= -(\alpha+1)i\sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l}) \\
&\quad -(\alpha+1)\sum_{k=0}^{i-1}\sum_{l=0}^{j}(-1)^{1+i+j+k+l}\binom{i}{k}\binom{j}{l}i(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l})
\end{aligned}
\tag{82}
$$

Now, the second term is separated to isolate the k-sum as much as possible, such that

$$
\begin{aligned}
&\boldsymbol{P}_{ij} \;-\; \boldsymbol{P}_{\text{-}ij} \\
&= -(\alpha+1)i\sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l}) \\
&\quad -(\alpha+1)i\sum_{l=0}^{j}(-1)^{1+i+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l})\left[\sum_{k=0}^{i-1}(-1)^{k}\binom{i}{k}\right]
\end{aligned}
\tag{83}
$$

However, this is also a well known binomial identity, such that

$$
\begin{aligned}
&\sum_{k=0}^{i}(-1)^{k}\binom{i}{k} \;=\; 0 \\
\Longrightarrow\; &\sum_{k=0}^{i-1}(-1)^{k}\binom{i}{k} \;=\; -(-1)^{i}
\end{aligned}
$$

Hence, (83) can be written as

$$
\begin{aligned}
&\boldsymbol{P}_{ij} \;-\; \boldsymbol{P}_{\text{-}ij} \\
&= -(\alpha+1)i\sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l}) \\
&\quad +(\alpha+1)i\sum_{l=0}^{j}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l}) \\
&= \cancel{-(\alpha+1)i\sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l})} \\
&\quad -i\,(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j}) \;+\; \cancel{(\alpha+1)i\sum_{l=0}^{j-1}(-1)^{1+j+l}\binom{j}{l}(\boldsymbol{P}_{0l} - \boldsymbol{P}_{1l})}
\end{aligned}
$$

$$= -\, i\,(\alpha+1)(\boldsymbol{P}_{0j} - \boldsymbol{P}_{1j}) \tag{84}$$

which demonstrates, by induction, that identity (75) holds for all positive $j \leq (n-1)/2$. The proof for all negative j is identical.

Thus, this completes the proof that G-continuity of order i holds if it is assumed at order $i-1$, in the x direction. Therefore, G-continuity holds for all i up to $(n-1)/2$, since G-continuity was shown for $i=2$. The same holds for the y-direction, for which an analogue of Proposition-3 can be obtained in the y-direction, such that the above proof can be repeated, with switched indices, in the y-direction. □

3.5. Derivatives identity

Note that the generalised nodal formulation of Definition-1 is very similar to the derivatives definition (39), but in two dimensions. In fact, a notable property of Definition-1 is the following corollary.

Corollary-1. *For the nodal formulation of $\boldsymbol{P}_{ij}$ defined in Definition-1, any derivative of order (f, g) at the element node $\boldsymbol{P}_{00}$, with $f+g>0$, corresponds to the vector $\vec{u}^{\,fg}$, namely*

$$\partial_s^f \partial_t^g \boldsymbol{F}(0,0) = \frac{n!}{(n-f)!}\frac{n!}{(n-g)!}\, h^{fg}\vec{u}^{\,fg} \tag{85}$$

Proof. Expanding the Bezier formulation (3) gives

$$\boldsymbol{F}(s,t) = \sum_{i=0}^{n}\sum_{j=0}^{m}\binom{n}{i}\binom{m}{j} s^i (1-s)^{n-i} t^j (1-t)^{m-j}\, \boldsymbol{P}_{ij} \tag{86}$$

where the polynomial orders have been separated between n and m, because it is important to distinguish between the two for this derivation, but in reality, $m=n$. Now, consider the derivative of order f for the two composites of a Bernstein polynomial,

$$\partial_s^f\left[s^i\right] = \begin{cases} \frac{i!}{(i-f)!}s^{i-f} & \text{for } 0 \leq f \leq i \\ 0 & \text{for } f > i \end{cases}$$

$$\partial_s^f\left[(1-s)^{n-i}\right] = \begin{cases} \frac{(n-i)!}{(n-i-f)!}(-1)^f (1-s)^{n-i-f} & \text{for } 0 \leq f \leq n-i \\ 0 & \text{for } f > n-i \end{cases}$$

Evaluated at $s=0$, this gives

$$\partial_s^f\left[s^i\right]\Big|_{s=0} = \begin{cases} i! & \text{for } f=i \\ 0 & \text{for } f \neq i \end{cases} \tag{87}$$

$$\partial_s^f\left[(1-s)^{n-i}\right]\Big|_{s=0} = \begin{cases} \frac{(n-i)!}{(n-i-f)!}(-1)^f & \text{for } 0 \leq f \leq n-i \\ 0 & \text{for } f > n-i \end{cases} \tag{88}$$

Next, consider the derivative of the factor of two functions of $I(s)$ and $J(s)$, namely,

$$\partial_s^f\left[IJ\right] = \sum_{a=0}^{f}\binom{f}{a}\left(\partial_s^a I\right)\left(\partial_s^{f-a} J\right)$$

and substitute the functions with the Bernstein polynomial components, such that $I = s^i$ and $J = (1-s)^{n-i}$. Then

$$\partial_s^f\left[s^i(1-s)^{n-i}\right] = \sum_{a=0}^{f}\binom{f}{a}\left(\partial_s^a\left[s^i\right]\right)\left(\partial_s^{f-a}\left[(1-s)^{n-i}\right]\right) \tag{89}$$

From (87), it follows that the above is zero at $s=0$ unless $a=i$. In addition, since the summation goes from $a=0$ to $a=f$, the above is also zero if $i>f$. Hence, evaluating (89) at $s=0$ gives

$$\partial_s^f\left[s^i(1-s)^{n-i}\right]\Big|_{s=0} = \begin{cases} \binom{f}{i} i!\left(\partial_s^{f-i}\left[(1-s)^{n-i}\right]\right) & \text{for } i \leq f \\ 0 & \text{for } i > f \end{cases} \tag{90}$$

However, from (88), it also follows that the above is zero if $f-i>n-i$, such that (90) becomes

$$\partial_s^f \left[s^i (1-s)^{n-i} \right] \Big|_{s=0} = \begin{cases} \binom{f}{i} i! (-1)^{f-i} \frac{(n-i)!}{(n-f)!} & \text{for } i \le f \le n \\ 0 & \text{otherwise} \end{cases} \tag{91}$$

Now, the Bezier function (86) can be derived with respect to s and t, such that

$$\partial_s^f \partial_t^g \boldsymbol{F}(s,t) = \sum_{i=0}^{n} \sum_{j=0}^{m} \binom{n}{i} \binom{m}{j} \partial_s^f \left[s^i (1-s)^{n-i} \right] \partial_t^g \left[t^j (1-t)^{m-j} \right] \boldsymbol{P}_{ij}$$

Thus, evaluating at $s=0$ and $t=0$, and using (91), the above becomes

$$\begin{aligned} & \partial_s^f \partial_t^g \boldsymbol{F}(s,t) \Big|_{s=0,t=0} \\ &= \sum_{i=0}^{n} \sum_{j=0}^{m} \binom{n}{i} \binom{m}{j} \left[\binom{f}{i} i! (-1)^{f-i} \frac{(n-i)!}{(n-f)!} \right] \left[\binom{g}{j} j! (-1)^{g-j} \frac{(m-j)!}{(m-g)!} \right] \boldsymbol{P}_{ij} \\ &= \sum_{i=0}^{n} \sum_{j=0}^{m} (-1)^{f+g-i-j} \frac{n!}{(n-f)!} \frac{m!}{(m-g)!} \binom{f}{i} \binom{g}{j} \boldsymbol{P}_{ij} \\ &= \frac{n!}{(n-f)!} \frac{m!}{(m-g)!} \sum_{i=0}^{n} \sum_{j=0}^{m} (-1)^{f+g-i-j} \binom{f}{i} \binom{g}{j} \boldsymbol{P}_{ij} \\ & \quad \text{for } i \le f \le n \text{ and } j \le g \le m \quad \text{(zero otherwise)} \end{aligned} \tag{92}$$

However, since the above is valid only for $i \le f$ and $j \le g$, the two sums are valid only up to f and g respectively, so that (92) simplifies to

$$\partial_s^f \partial_t^g \boldsymbol{F}(s,t) \Big|_{s=0,t=0} = \frac{n!}{(n-f)!} \frac{m!}{(m-g)!} \sum_{i=0}^{f} \sum_{j=0}^{g} (-1)^{f+g+i+j} \binom{f}{i} \binom{g}{j} \boldsymbol{P}_{ij} \tag{93}$$

where, of course, the identity $(-1)^{-i} = (-1)^i$ was used. Now, note that at $i=f$ and $j=g$ the summation term in (93) simply gives $\boldsymbol{P}_{fg}$. By using the Kronecker delta function, (93) can thus be written as

$$\begin{aligned} & \partial_s^f \partial_t^g \boldsymbol{F}(s,t) \Big|_{s=0,t=0} \\ &= \frac{n!}{(n-f)!} \frac{m!}{(m-g)!} \left[\boldsymbol{P}_{fg} - \sum_{i=0}^{f} \sum_{j=0}^{g} (-1)^{1+f+g+i+j} (1 - \delta_{if}\delta_{jg}) \binom{f}{i} \binom{g}{j} \boldsymbol{P}_{ij} \right] \end{aligned} \tag{94}$$

This completes the proof, since the bracket term is none other than the nodal formulation (45) of Definition-1, namely

$$\partial_s^f \partial_t^g \boldsymbol{F}(s,t) \Big|_{s=0,t=0} = \frac{n!}{(n-f)!} \frac{m!}{(m-g)!} h^{fg} \vec{u}^{\,fg} \quad \square$$

4. Implementation in the non-linear MHD code JOREK

4.1. The JOREK code

The above formulation is implemented in the JOREK code [20–23]. JOREK uses a finite-element grid in the poloidal plane, with finite Fourier series in the toroidal direction. The finite elements are quadrangular Bezier elements using a nodal formulation equivalent to the generalised formulation in Definition-1 and Definition-2 with polynomials of order 3, and thus G-continuity of order 1 [22]. The Fourier series used in the toroidal direction can be chosen with arbitrary toroidal periodicity, meaning that with 3 Fourier modes and a toroidal periodicity of 4, the physical toroidal mode numbers simulated would be $n = 4, 8, 12$. Typically, when producing linear stability scans, a single Fourier mode is used, while scanning the toroidal periodicity. For non-linear simulations, multiple Fourier modes are used, with a periodicity of 1 or 2.

The time discretisation used in JOREK is fully implicit, with a choice between the Crank-Nicolson or the Gear's scheme. This implicit method results in a linearised system of equations, for which a sparse matrix needs to be inverted. This is done either with a direct solve, or using a preconditioned GMRES iterative solver. In practice, the direct solve approach is

only used for the toroidally axisymmetric kinetic equilibrium $n = 0$, while for non-linear cases, the GMRES method is employed. The GMRES preconditioner is equivalent to a Block-Jacobi preconditioner, which is obtained by solving each toroidal harmonic block of the matrix individually (i.e. without the harmonic coupling), and these blocks are dealt with individually, in parallel, with a direct solve. The direct solves (either for the preconditioner harmonic-blocks, or for the whole system) is done using open-source solvers like MUMPS [56–58], PASTIX [59,60] or STRUMPACK [61,62]. Further developments on the preconditioner have recently been implemented in the JOREK code, allowing the preconditioner blocks to include multiple Fourier harmonics, with their respective non-linear interactions [63].

The Weak Formulation method [34] is used for all equations, which are multiplied by a test-function and integrated over the element surfaces. The test-functions are chosen to be the basis functions obtained from the nodal formulation, as in (6), or as described in the next section. One of the main advantages of the weak formulation is that it allows integration by parts of divergence terms, which allows the removal of all 2nd order derivatives from the system. This was a significant advantage when using G1-continuous finite-elements in the past, like those implemented in JOREK [22], since all terms in the equations were guaranteed to be smoothly represented, thus reducing numerical noise and improving practical reliability.

The physics models of JOREK includes visco-resistive models for both full-MHD and reduced-MHD [35]. The equations are normalised using two constants: the vacuum permeability μ_0, and the central density ρ_0. This normalisation is similar to the Alfven time normalisation, such that for a deuterium plasma with central particle density $n_o = 6 \times 10^{19}\ \mathrm{m}^{-3}$, a normalized time unit corresponds to approximately $0.5\mu\mathrm{s}$. Note that this is a pseudo-normalisation, where not all variables are dimensionless in the final formulation. In particular, the magnetic field is not normalised.

4.2. The basis functions

The full formulation of any variable ψ (as well as spatial variables R and Z) on one finite element, including the Fourier series, is given by

$$\psi(s,t) = \sum_{j=1}^{4} \sum_{k=0}^{\frac{n-1}{2}} \sum_{l=0}^{\frac{n-1}{2}} \sum_{m=0}^{N} \left[\psi_{jm}^{kl}\, F_{klj}(s,t)\, h_j^{kl}\, e^{im\phi} \right] \tag{95}$$

j : sum on 4 element nodes

k : sum on first vector index

l : sum on second vector index

m : sum on N Fourier modes

$F_{j,k,l}(s,t)$: nodal basis functions as in (6)

h_j^{kl} : element sizes

ϕ : toroidal angle

ψ_{jm}^{kl} : variable values for each degree of freedom

Note, with respect to the generalised formulation (45), the ψ_{jm}^{kl} values are the equivalent of the vectorial basis $\vec{u}^{\,ij}$, but for any variable ψ, including spatial coordinates. Once the nodal formulation (45) has been calculated for a given polynomial order n, the basis functions F_{jkl} of this nodal basis need to be calculated as the coefficients obtained from the factorization of the Bezier formula (3) with respect to each $\vec{u}^{\,ij}$ term, as in (6). For bi-cubic G1-continuous elements, this is easily done by hand. For bi-quintic G2-continuous elements, the basis functions and their derivatives with respect to s and t can be calculated by hand, although this is already tedious. The s- and t-derivatives are required up to order 2 in the code, which means 5 derivatives in total (plus the absolute value), for each basis function component. For bi-quintic G2-continuous elements, this gives a total of: 4 nodes, times $[(n+1)/2]^2$ basis vectors, times 6 derivatives (including zeroth derivative); hence 216 functions to calculate and include in the code. For elements of order $n = 7$, there are 384, and for $n = 11$, there are 864.

Thus, although the basis functions and their derivatives can be calculated by hand for bi-quintic elements, at higher order this is not feasible without a considerable risk of human error. It is much safer to do this analytical calculation numerically, which can be done relatively easily with analytical algebra libraries like Python's symbols library, Sympy [64]. Using this library, the Fortran module of JOREK that includes the basis function formulas is auto-generated for any polynomial order n.

It is worth highlighting the fact that continuity is not 'imposed' in the matrix of the final linear system. Expressions (45) and (95) are nodal formulations, meaning that the degrees of freedom, namely the $\vec{u}^{\,ij}$ in (45) and the ψ_{jm}^{kl} in (95), are shared between neighbouring elements at their common nodes. Thus, any solution of the linear system will always be continuous, provided the elements have been properly constructed, respecting the continuity constraints (46) and (47). In other words, continuity is not imposed inside the linear system, it is ensured by the geometrical construction of the elements with respect to their neighbours, for which expressions (45), (46) and (47) are the recipe. Formula (95) is the

recipe to reconstruct the solution inside one element, given the values of the degrees-of-freedom at each of the 4 nodes, but with the finite-element method, as explained in Section-3.2 of [22], the final linear system uses the shared degrees-of-freedom at each node independently, with contributions from the 4 parent elements through the Weak-Form integration. Thus, at each node, there are $N[(n-1)/2]^2$ degrees-of-freedom.

4.3. Gaussian integration

The weak-formulation of JOREK requires integration over each element. This is done using Gaussian quadrature, for which the number of Gaussian integration points (and weights) needs to correspond to the polynomial order of the Bezier elements. Although trivially obtained from Python libraries like Numpy [65], this also requires the auto-generation of a small module in the JOREK code (for polynomial orders $n > 7$), to register these Gaussian integration points and weights.

4.4. The mesh generators and initial conditions

The JOREK code includes a large variety of mesh generators, adapted to complex magnetic configurations in tokamaks, and with adaptive extensions to arbitrary wall structures [66]. These grid generators had to be adapted to include higher order polynomials. The alignment of the grids, along the background magnetic flux surfaces, is done up to second derivatives, which is largely sufficient for most applications, and thus for polynomial orders $n \geq 7$, the derivatives of order ≥ 3 are set to zero (only for the isoparametric spatial variables, not for the physics model variables). In the future, if higher accuracy is required, for Bezier elements of order $n \geq 7$, these higher-order derivatives could be included in the grid generators.

It is difficult to quantify accurately the effect of precision in the alignment of the 2nd grid, along flux-surfaces. In linear cases, where the plasma equilibrium does not change, this could be tested by using various misaligned and aligned grids. However, for non-linear cases, the background equilibrium typically shifts slightly from the grid, due to kinetic contributions to the momentum balance, which can change the Grad-Shafranov equilibrium. Of course, in cases where the equilibrium pressure and current profiles change, misalignment of flux-surfaces with the grid becomes even more pronounced. Therefore, in realistic simulations, the equilibrium is never perfectly aligned to the grid. Nevertheless, having some alignment to flux surfaces is systematically better than the opposite. For example, using a simple polar grid for a ballooning mode simulation with an X-point plasma will result in numerical noise, particularly around the X-point, where the magnetic field is sheared, and where parallel transport plays a major role in propagating filaments in the parallel direction. Of course, this is not the case if the resolution of this polar grid is very high, but in practice this is numerically too costly, and flux-aligned grids are highly preferable. A quantification of these alignment effects on anisotropic diffusion and conduction can be found in [20].

Finally, initial conditions must be set on the generated grids, before time-evolution is simulated. This is done simply by projecting the initial condition on the node coefficients $\psi^{kl}_{j,m=0}$.

4.5. Replacing root solvers

Several root-solver routines inside JOREK cannot be used with bi-quintic (or higher-order) elements. These root-solvers are used to find values along the sides of elements, either to determine the spatial minima/maxima of an element, or to find the intersections of flux-surfaces along element edges, used for flux-aligned grid construction. For example, with bi-cubic elements, finding the location of a ψ-value along the side of an element simply requires the analytical formula for the solution of a 3rd order polynomial. Since there is no analytical solution for polynomials of order 5 or higher, locating root values along element sides requires converging algorithms like Newton's method. Of course Newton's method can only be used after first determining the location of minima/maxima along the side of an element; once this is achieved, Newton's method can be used to locate roots between consecutive minima/maxima. Note that locating minima/maxima along the side of a bi-quintic element requires the zero solutions of the derivative of a given variable ψ, thus a 4th order polynomial. While there exists a formulation for the roots of 4th order polynomials, since this cannot be done for polynomial orders $n \geq 7$, Newton's method is also used to find these roots to determine the minima/maxima.

4.6. Long-integer matrix solvers

In theory, for a given simulation grid, if higher polynomial Bezier elements are used, the spatial resolution of the grid number of poloidal elements can be diminished, because numerical precision increases with the order of the finite-elements [34], as will be shown in Section-5 with convergence tests. However, in practice, particularly when addressing performance and convergence properties of the elements, long-integers (64 bit) may be needed for the sparse-matrix representation and the solvers. For large non-linear problems, the matrix sent to direct solvers like PastiX is determined by the grid size, i.e. the number of nodes n_n, the number of degrees of freedom per node n_d, and the number of variables n_v in the physics system. Indeed, since the GMRES preconditioner is obtained by solving each Fourier harmonic individually, the size of these matrices is independent of the number of Fourier modes. Thus, each matrix size is, at most

$$m \times m = (2\, n_n\, n_d\, n_v)^2 \tag{96}$$

where the factor 2 comes from the *sine* and *cosine* component of each Fourier mode. Of course, these are sparse matrices, but in practice, their number of non-zero entries will scale with the square of the number of degrees of freedom, n_d^2. Since the number of degrees of freedom per node scales as the square of the polynomial basis order, n^2, it means the size of the matrices will scale as n^4. In order to enable simulations with large spatial resolutions and large polynomial basis orders, parts of the JOREK code required modification to allow for a compressed sparse row (CSR) representation of the matrices using long-integers. Although straight-forward in most instances of the code, this also requires MPI communications of the matrix to be split into batches, since MPI communication indexing is restricted to short integers.

5. Testing the JOREK code

5.1. Linear MHD tests

Three linear benchmarks are conducted for core and edge MHD instabilities in toroidal geometry, using the reduced-MHD model [21]. The first two linear benchmarks are a low-β $m = n_\phi = 1$ internal kink mode, and a low-β $m = 2$, $n_\phi = 1$ tearing-mode (where m is the dominant poloidal mode, and n_ϕ the toroidal Fourier mode). Both instabilities were studied in previous publications [22,35], and they are included here for completeness and clarity. The two equilibria are similar, but differ in safety q-profile: the internal kink mode has a q-profile in the range [0.7, 1.6], crossing $q = 1$ at $\psi_n = 0.5$, while the tearing mode has a q-profile in the range [1.7, 3.9], crossing $q = 2$ at $\psi_n = 0.3$. The Grad-Shafranov equilibrium quantities and profiles for these two cases are described in more details in [35]. Since the equilibria both have circular poloidal cross-sections, polar grids are used.

The first two benchmark cases are run for a scan in resistivity. The kink mode is run with resistivity alone (without viscosity, and without particle or thermal diffusion), while the tearing mode is run including all diffusions, with $\mu_0 = 10^{-8}\ \mathrm{kg.m^{-1}.s^{-1}}$, $D_\perp = 0.7\ \mathrm{m^2.s^{-1}}$, and $\kappa_\perp = 1.7 \times 10^{-8}\ \mathrm{kg.m^{-1}.s^{-1}}$. For simplicity, the resistivity and viscosity are taken to be spatially constant for both cases. Note that only the toroidal mode $n_\phi = 1$ is simulated here, such that coupling with higher toroidal modes is not present in these linear benchmarks. Note, for a resistivity value of $\eta = 10^{-6}\Omega.m$, the corresponding Lundquist number for the kink and tearing cases is approximately $S = 2.5 \times 10^6$.

The third benchmark case is an X-point plasma with peeling-ballooning instabilities. This is an artificial equilibrium similar to the JET tokamak. It is run using the kinetic stationary background equilibrium flows ($n_\phi = 0$), together with a single toroidal harmonic, which is ranged from $n_\phi = 1$ up to $n_\phi = 20$. The equilibrium characteristics, together with the diffusive parameters, are identical to those described in [67].

Fig.-5 shows the benchmark of the internal kink mode, the tearing mode, and the X-point peeling-ballooning modes, for various polynomial orders up to $n = 7$ with G3-continuity (and up to $n = 9$, G4-continuity for the internal kink). Poloidal cross-sections of the main Fourier mode perturbation are shown for the toroidal current j_ϕ, together with the growth rates of the modes, plotted as a function of resistivity for the internal kink and tearing mode cases, and as a function of toroidal mode number for the peeling-ballooning mode. The linear theory in the ideal (low resistivity) regimes is also plotted, with a $\eta^{1/3}$ scaling for the internal kink mode [68], and a $\eta^{3/5}$ scaling for the tearing mode [69]. For the peeling-ballooning mode, the poloidal cross-section pictures the toroidal Fourier mode $n_\phi = 17$. The benchmarks are, as expected, identical in all cases.

5.2. Non-linear MHD tests

A simple non-linear simulation is run to test that results are well reproduced at higher polynomial orders in non-linear regimes. This case is an Edge-Localised-Mode in a discharge from the MAST tokamak. The particularity of this case, also described in details in [67], is that plasma filaments are expelled from the edge plasma, travelling at large radial valocities (several $\mathrm{km/s}$) into the far Scrape-Off Layer (SOL). In JOREK simulations, the SOL is a near-vacuum region, which still has a finite background density and temperature (2% of the core values in this case, before the filaments appear), and where the same MHD equations are used as for the rest of the simulation domain. For this benchmark, the G1- and G2-continuous versions are run (i.e. with polynomial orders $n = 3$ and $n = 5$ respectively) and compared after the filaments have crossed the separatrix. Slightly higher spatial resolution (number of poloidal elements) is used for the G1 case than for the G2 case, to ensure the filaments are accurately resolved. At the outboard mid-plane, where the filaments evolve, both cases use a radial resolution of 2.5 mm, but the G1 cases uses a poloidal resolution of 2.5 cm, compared to 4.5 cm for the G2 case.

Fig.-6 shows a poloidal cross-section of the MAST tokamak, as the filaments travel into the SOL. A radial profile of density is taken across the filament, and compared between the G1 and G2 runs. Negligible differences are found between the two profiles, which is to be expected due to the accuracy differences between the two cases, and the resulting effect this may have due to the non-linear nature of the run. Beyond this early phase of the filament evolution, both simulations keep evolving without numerical issue, but these small differences between profiles become more pronounced at a later stage. Nevertheless, this benchmark is also extremely positive.

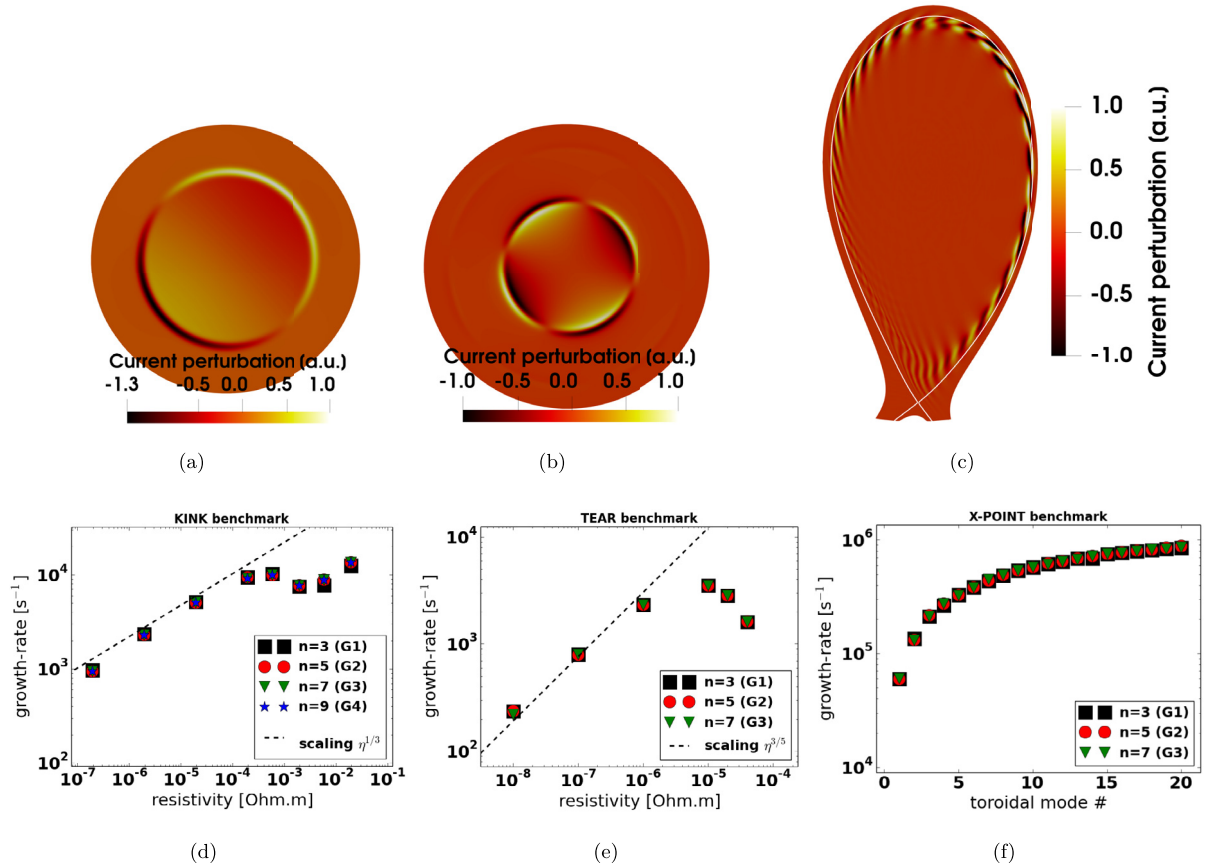


Fig. 5. $m = 1$, $n_\phi = 1$ internal kink mode benchmark: (a) Poloidal cross-section of the normalised perturbation of the toroidal current j_ϕ, and (d) comparison of the growth rates between various polynomial orders, ranging from $n = 3$ to $n = 9$, as a function of resistivity η. Case (a) corresponds to the G2-continuous point with $\eta = 2.10^{-5}\Omega.m$ on Figure-(d). $m = 2$, $n_\phi = 1$ low-β tearing mode benchmark: (b) Poloidal cross-section of the normalised perturbation of the toroidal current j_ϕ, and (e) comparison of the growth rates between various polynomial orders, ranging from $n = 3$ to $n = 7$, as a function of resistivity η. Case (b) corresponds to the G2-continuous point with $\eta = 10^{-6}\Omega.m$ on Figure-(e). Peeling-ballooning mode benchmark with X-point equilibrium: (c) Poloidal cross-section of the normalised perturbation of the toroidal current j_ϕ for the Fourier mode $n_\phi = 17$, and (f) comparison of the growth rates between various polynomial orders, ranging from $n = 3$ to $n = 7$, as a function of the toroidal mode number n_ϕ. (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

5.3. G-continuity verification

Verification of G-continuity is done up to polynomial order $n = 7$ (G3-continuous), by comparing between subsequent polynomial orders of finite elements, at fixed spatial resolution (i.e. fixed element size). This is done by measuring linear profiles across a sharp perturbation of the toroidal current j_ϕ.

Fig.-7 shows the comparison for 2nd and 3rd derivatives between corresponding polynomial orders for the 1D profiles. This result clearly shows that the discontinuity of derivatives at element boundaries (represented by jumps in the red profiles), disappears entirely at the relevant polynomial order. Fig.-8 shows 3D warp surfaces of the corresponding quantities $\partial_R j_\phi$ and $\partial_R^2 j_\phi$ for the G2-continuous case, revealing the smoothness of derivatives.

Note that the derivatives shown in Fig.-7 are computed exactly, or 'strongly'. The Weak Formulation of derivatives is only used inside the code to evolve the equations in time. It can be used to transfer derivatives from a variable to its counterpart test-function [34]; but in the case of Fig.-7, these are the physical quantities.

5.4. Convergence tests

One of the main interests of using higher-order polynomials with finite elements is that the local numerical errors diminish significantly. Convergence of growth rates as a function of spatial grid resolution is tested for a standard ballooning benchmark, the so-called CBM18 ballooning case, also described in details in [35]. The grid resolution is scanned homogeneously in the radial and poloidal directions, scanning from $(n_{flux}, n_{tht}) = (54, 180)$ to $(267, 890)$, where n_{flux} and n_{tht} are the number of radial (flux surfaces) and poloidal (theta) grid-points, respectively, both of which are equidistant in real-space for this case.

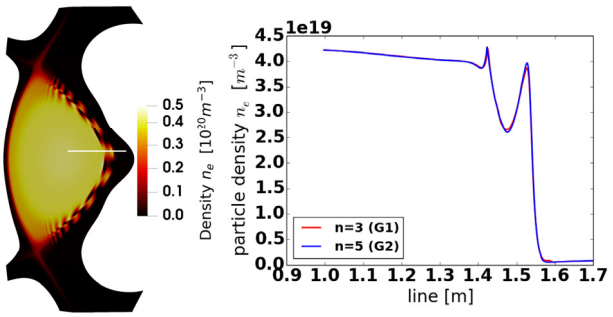


Fig. 6. (a) Poloidal cross-section picturing the particle density n_e during an Edge-Localised-Mode (ELM) in the MAST tokamak, during which plasma filaments are expelled at several km/s into the Scrape-Off Layer. (b) Particle density profiles across a filament while it is moving away from the plasma, with a comparison between the G1-continuous ($n = 3$) and G2-continuous ($n = 5$) finite elements. These radial profiles are taken along the white horizontal line pictured in (a).

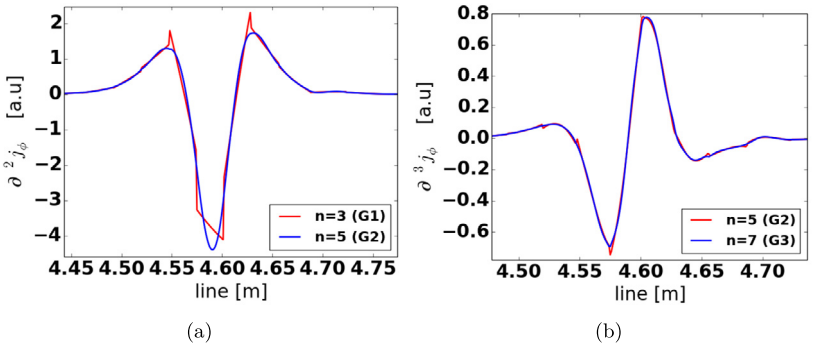


Fig. 7. (a) Comparison of second derivatives across a sharp perturbation of the toroidal current j_ϕ, between the G1-continuous ($n = 3$) and G2-continuous ($n = 5$) finite elements. (b) Comparison of third derivatives across a sharp perturbation of the current j, between the G2-continuous ($n = 5$) and G3-continuous ($n = 7$) finite elements. Note that all cases are done at equal spatial resolution (i.e. elements are the same size in all cases). The plots reveal the discontinuity of the current derivatives (red curves), which becomes smooth at the relevant G-continuous polynomial order (blue curves).

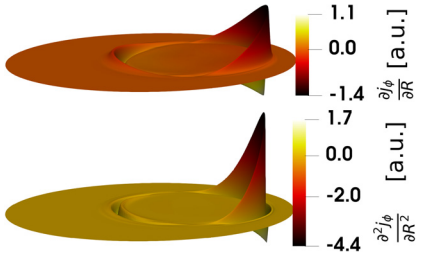


Fig. 8. 3D warp surfaces of $\partial_R j_\phi$ (top) and $\partial_R^2 j_\phi$ (bottom).

With finite elements, any smooth function $\boldsymbol{u}$ differs from its numerical representation $\tilde{\boldsymbol{u}}$ by $||\boldsymbol{u} - \tilde{\boldsymbol{u}}|| \leq Ch^p ||\boldsymbol{u}^{(p)}||$, for some constant C, where h is the element size, $p = n + 1$ is the polynomial order of the finite elements, and $\boldsymbol{u}^{(p)}$ is the p-th derivative of the function [34]. Thus, assuming that the analytical solution of the system has a bounded derivative $\boldsymbol{u}^{(p)}$, the error should scale as $E \sim h^p$. In cases where the non-linear MHD system does not possess a finite $\boldsymbol{u}^{(p)}$, convergence would occur at a lower rate. Here, $(\sqrt{n_{flux} n_{tht}})^{-1}$ is used as an approximation of the element size h. Note that these tests evaluate the growth rates of toroidal modes, which are obtained by integrating the mode energies over the domain (i.e. over the elements), hence adding another factor $(\sqrt{n_{flux} n_{tht}})^{-1}$ to the error estimate. Thus, the error of the growth rates should scale with the $(p + 1)$ power of the spatial resolution, i.e. $(\sqrt{n_{flux} n_{tht}})^{-5}$ for bi-cubic elements, $(\sqrt{n_{flux} n_{tht}})^{-7}$ for bi-quintic elements, etc.

Fig.-9b shows the convergence of the growth rate error, as a function of spatial resolution. The scaling of this convergence is, as expected, dependent on the corresponding p-refinement level of the finite-elements, as long as the resolution is above a certain minimum (vertical dashed line). The converged growth rates, used to evaluate the errors in Fig.-9, are obtained using the highest resolution cases available. A least-square fit is done for the expected convergence, such that the converged

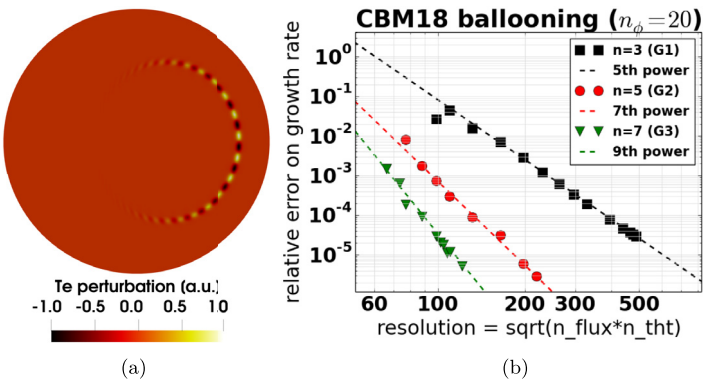


Fig. 9. (a) Poloidal cross section of the ballooning perturbations (mode number $n_\phi = 20$) of the electron temperature T_e for the CBM18 test-case used to verify convergence of the error with respect to resolution. (b) Convergence of the error on the growth rates, for the CBM18 test-case with the ballooning mode number $n_\phi = 20$, as a function of spatial grid resolution (using log-scales for both axes). The error converges according to the estimate $E \sim h^p$ [34].

growth rate is estimated at "infinite" resolution. Note that although separate fits are done for each polynomial order, the differences of the converged growth rates, between the G1, G2 and G3 cases, are of the order of 10^{-7}, which is below the lowest error obtained. Also, scans in the amplitude of the time-step were performed to ensure that an accurate (converged) time-resolution was used for the runs.

This CBM18 test case is a representative example of both the advantages and the limitations of higher-order finite-elements. As can be seen in Fig.-9b, below a certain poloidal resolution, the fine ballooning structure of Fig.-9a cannot be resolved properly, and the error starts deviating from the scaling. Both high radial and poloidal resolution is required to resolve the ballooning filaments, and this requirement increases with the toroidal mode number n_ϕ. Depending on the physics addressed, higher-order finite-elements may not always be the preferred choice. For example if addressing turbulence, which requires mode numbers upwards of $n_\phi = 40$, the high minimal resolution required may become computationally too expensive when combined with high-order polynomial elements. Users need to find the balance between the spatial scale of non-linear structures to be resolved for a given instability, and the computational cost of simulations. The highest resolution cases required 240 nodes (almost 12,000 cores) on the CINECA-Marconi cluster, due to the memory requirement for solving the larger sparse matrices at higher polynomial order (240 nodes on CINECA-Marconi corresponds to approximately 47*TB* of memory). Efforts are ongoing in the JOREK team to reduce memory consumption [63], although this is a common feature of fully-implicit schemes. The next subsection provides a more detailed evaluation of the computational gains versus accuracy. The highest resolutions used here were $(n_{flux}, n_{tht}) = (267, 890)$ for the G1-elements, $(n_{flux}, n_{tht}) = (120, 400)$ for the G2-elements, and $(n_{flux}, n_{tht}) = (66, 221)$ for the G3 elements.

In principle, the h^p convergence property of the growth rates should be independent from the level of G-continuity. However since G-continuity is ensured by the geometrical construction of the finite-elements, a voluntarily misalignment of the control points between elements would artificially enforce discontinuity, due to the nodal formulation and the resulting shared degrees-of-freedom. Since this situation is not of practical interest for JOREK applications, the convergence tests were not conducted with voluntarily discontinuous elements.

5.5. Performance tests

Fig.-10a shows the computation time as a function of the polynomial order of the finite elements, up to $n = 13$ (G6-continuous). As expected, the computation time scales as the square of the number of degrees of freedom per node (because the matrix dimension scales as $n_{degrees}$ in both directions), hence a 4th order scaling with the order of the polynomial basis. Fig.-10b shows the computation time as a function of the error on the growth-rate, corresponding to the CBM18 test-case shown in Fig.-9. Higher accuracy requires higher computational cost, as expected. However, the important point to notice here is that, in order to achieve low error levels (below 10^{-3}), the higher-order finite-element cases become systematically more efficient, in some cases by an order of magnitude.

In Fig.-10b, for the G1-cases, there is an almost linear dependence between the computation time and the error (at least above 10^{-4}). This can be explained by estimating that the computation time scales at least as fast as the total size of the matrix of the final linear system, which scales as the square of the number of nodes in the poloidal grid, such that $T_{cpu} \sim n_{nodes}^2$. For a polar grid, the number of nodes is $n_{nodes} \approx n_{flux} n_{tht}$. Hence, since the estimate for the element size, as given in Fig.-9b, is $h \approx (\sqrt{n_{flux} n_{tht}})^{-1}$, it follows that the computation time scales as $T_{cpu} \sim h^{-4}$. Now, for cubic elements, the error bound from [34] is $E \sim h^4$, such that the computation time should scale at least linearly with the error level, $T_{cpu} \sim E^{-1}$. Of course, since the solving efficiency of sparse linear system does not scale perfectly, this dependency clearly breaks at low error levels (i.e. large matrix sizes).

Note that an additional modifications, left for future development, could further reduce the cost of higher-order finite elements. Firstly, the reduced-MHD model, frequently used in JOREK, uses separate equations for auxiliary variables w_ϕ and

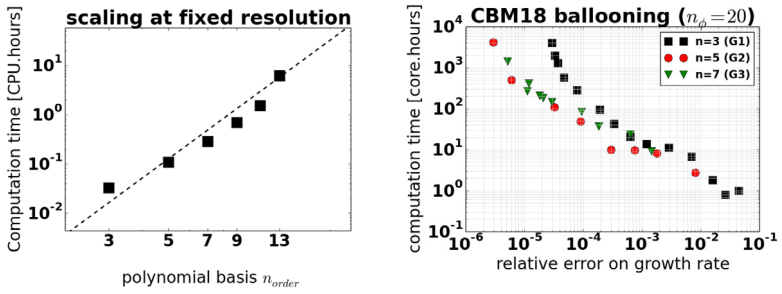


Fig. 10. (a) Scaling of the computation time, at fixed spatial grid resolution, as function of the polynomial basis, up to $n = 13$ (G6-continuous). The dashed line shows the scaling of n^4, which matches the calculations as expected. (b) Computation cost of the convergence test case from Fig.-9, as a function of error. This clearly shows that, between G1- and G2- cases, at equivalent accuracy, the G2 runs are computationally cheaper.

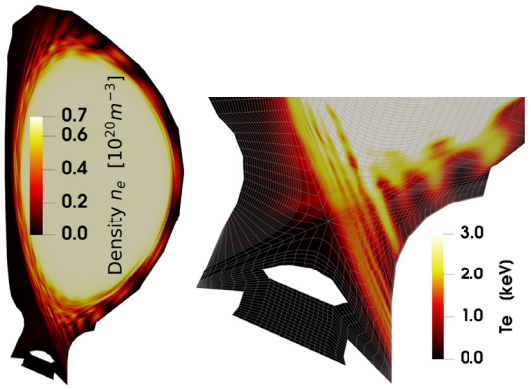
j_ϕ. The reason for this is that these variables are themselves 2nd order derivatives of other variables, such that solving these separately reduces numerical errors in G1-continuous elements. With higher G-continuity, these could be removed, reducing the number of variables from 7 to 5, which would significantly reduce the size of the problem. Secondly, using localised p-refinement would also result in lower computational costs, while keeping the desired accuracy where it is needed (for example in the region where ballooning structures are dominant in Fig.-9a). These modifications would however require significant development and testing, which is beyond the scope of this paper and left for future work. In particular, localised p-refinement would require a far more elaborate nodal formulation, to satisfy the conditions proposed by Aumann & Bentz [48].

5.6. Advanced simulation demonstration

Finally, to demonstrate the practical usability of the new finite-element method implemented in JOREK, a non-linear simulation is run, with multiple Fourier mode numbers $n_\phi = (1, 2, 3, ..., 8)$, for an Edge-Localised-Mode in ITER. This is done for a 15MA equilibrium, with a pedestal electron temperature of $T_e^{ped} = 8$ keV. An advanced grid geometry is used, aligned to the first-wall of ITER, including the divertor dome geometry below the X-point, based on the grid-patches method developed in [66]. Note that the exact geometry of the ITER divertor dome is 3-dimensional and far more complex in reality [70], but demonstrating the feasibility of such an isolated dome in JOREK simulations is an important aspect of current challenges, particularly for detachment studies [71], where the dome can play a major role for neutral particles dynamics. The simulation is run with the reduced-MHD model [21,72], using conservative physical diffusivity coefficients, since its purpose is simply to demonstrate numerical feasibility of violent MHD non-linear instabilities. The resistivity is taken to be 300 times the neoclassical resistivity. More advanced studies of ITER plasmas, at experimental resistivity and with diamagnetic effects, are the current focus of other experts in the JOREK team, and are well beyond the scope and purpose of this paper.

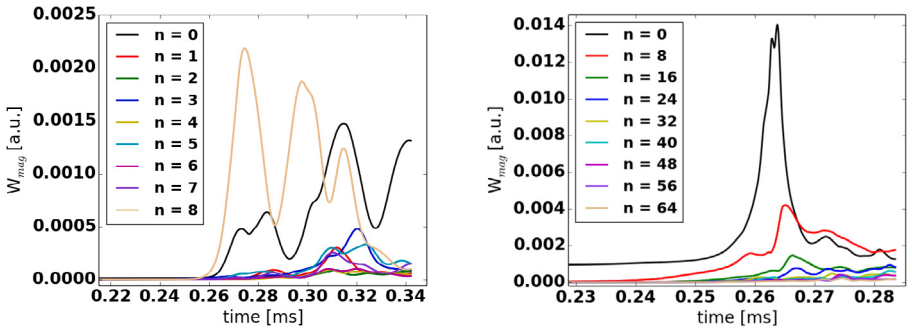
Fig.-11 shows the density and electron temperature as ELM filaments are expelled from the plasma. Fine, poloidally well-resolved filamentary structures can be observed reaching the divertor. The finite-element edges are also pictured, in the divertor and X-point region, in Fig.-11b. The magnetic energy of the toroidal modes is pictured in Fig.-12a, where the ELM starts with the toroidal mode number $n_\phi = 8$, followed by coupling with lower harmonics, as is typical of non-linear ELM dynamics [73]. However, truncating the number of toroidal modes at $n_\phi = 8$ can have important consequences for the dynamics of MHD instabilities. In fact, as suggested by Aydemir et al. [74], the toroidal resolution should be about one order of magnitude higher than the most unstable mode number. Although JOREK cannot yet handle all toroidal modes from $n_\phi = 1$ to $n_\phi = 80$, another simulation was run with octant periodicity, including the modes $n_\phi = 8, 16, 24, ..., 64$. This simulation is run with diamagnetic effects [72], which stabilises the higher modes, such that $n_\phi = 8$ is the dominant mode number in the linear phase. The evolution of the magnetic energy of all toroidal modes is shown in Fig.-12b. Indeed, while the dominant mode is $n_\phi = 8$ throughout the simulation, the modes $n_\phi = 16$ and $n_\phi = 24$ have a non-negligible contribution. At the peak of the MHD activity, around $t = 0.265ms$ in Fig.-12b, the contribution of higher mode numbers is much weaker, and it rapidly decreases as the mode number increases: the modes $n_\phi = 32$ and $n_\phi = 40$ are more than one order of magnitude lower than the $n_\phi = 8$ amplitude, while the contribution from $n_\phi = 48, 56, 64$ are negligible, two orders of magnitude below the amplitude of $n_\phi = 8$. Nevertheless, moderate non-linear coupling with higher modes does occurs later, after the peak of the filament activity. These simulations should provide sufficient confidence to the JOREK community that the high-order finite-element method presented here, and its implementation in the code, has the numerical robustness to handle advanced future studies of tokamak MHD, at least as well as the previous G1 implementation. Note that improving the computational efficiency and scalability of the code, in order to enable higher toroidal resolution, is a continuing effort and one of the main priorities of the JOREK developers community.

Beyond the demonstration of the feasibility of advanced simulations like the one portrayed in Fig.-11 and Fig.-12, some qualitative comparisons were performed between G1- and G2-continuous grids for under-resolved cases. The same ITER



(a) (b)

Fig. 11. ITER simulation with G2-continuous elements, with a wall-aligned grid, including the divertor Dome below the X-point. (a) Poloidal cross-section of the density, showing filaments travelling radially in the Scrape-Off Layer. (b) Zoom on the divertor region, showing thin stripes of temperature reaching the divertor target, due to heat conduction from upstream filaments along magnetic field-lines.



(a) (b)

Fig. 12. (a) Magnetic energies of the ballooning modes as a function of time, for the simulation shown in Fig.-11, where $n_\phi = 8$ dominates the early phase of the ELM, followed by non-linear coupling with lower modes in the later phase. (b) Magnetic energies of the ballooning modes for the simulation with $n_\phi = 8, 16, 24, ..., 64$ including diamagnetic effects, where the modes $n_\phi = 32$ and $n_\phi = 40$ have a clear contribution to the dominant $n_\phi = 8$ mode number.

simulation of an ELM with $n_\phi = (1, 2, 3, ..., 8)$ was conducted with more challenging visco-resistive and diffusive parameters, and with a poloidal grid resolution which results in the same computational cost for both the G1 and G2 cases (approximately a factor 2.5 in both the radial and poloidal directions). One of the most typical numerical issues encountered in practice with the JOREK code is when the ELM filament dynamics is very violent, creating areas of low plasma density and temperature. In the worst cases, the positivity of density and temperature is violated, which in turn leads to further numerical issues: the current density and toroidal vorticity become numerically noisy, with fine-scale oscillations below the size of the elements, which may be attributed to Gibbs oscillations. These oscillations may grow exponentially if untreated, thus restricting the simulation. For the simulation shown in Fig.-11 and Fig.-12, in order to avoid negative density, a conservatively small time-step is used (approximately 1/20th of the Alfven time), and the perpendicular diffusion $D_\perp$ is increased locally by a factor 100 wherever the density comes below 0.1% of the core density (7×10^{16} m^{-3} in this case). This allows the density to remain approximately above 0.01% of the core density during the peak of the MHD activity.

All parameters are identical in both simulations, using a conservatively small time-step size. With numerical diffusion, the level of numerical noise in the region of the filaments can be reduced, allowing the simulation to run through without a numerical blow-up of the Gibbs oscillations (which typically leads to the failure of the GMRES convergence). It could be expected that Gibbs oscillations will change as the order of the finite-element increases, leading to even finer structure, but this is not what is observed in these case. In fact, as the numerical diffusion was progressively scanned, either the G1 and G2 cases both failed, or they both succeeded. Individual scans were performed for numerical diffusion in the continuity equation, the poloidal momentum equation, and the parallel momentum equation, but in all cases the conclusion was the same: with identical diffusive parameters, either the G1 and G2 cases both failed, or they both succeeded. This suggests that, for non-linear cases, G2 elements are practically as reliable as G1 elements. Nevertheless, as could be expected from Fig.-10b, there may be regimes, at higher resolutions, where this could be different, but demonstrating this particular result would require a large amount of computational resources, since the non-linear case would need to be run several times,

while scanning the hyper-diffusive parameters, and for several grid sizes of higher resolutions which are computationally even more expensive.

Note that numerical stabilisation methods, like the Taylor-Galerkin method and the Variational Multi-Scale method [75], are also available in JOREK to handle numerical issues like Gibbs oscillations. However these methods were not used in this study, and would need to be addressed in future works.

6. Conclusion

6.1. Summary

This paper presents the significant development and generalisation of the finite-element method used in the JOREK code. A generalised formulation was developed for a nodal representation of Bezier elements, which ensures that G-continuity of order $(n-1)/2$ is respected, where n is the order of the polynomial basis of the Bezier elements. Rigorous proof was provided to show that the definition of our nodal formulation ensures G-continuity. Some details about the key aspects of the implementation into the code were provided, followed by a series of tests and benchmark studies. Beyond the fact that the G-continuity of the method is demonstrated numerically, and that all benchmark tests provide identical results to the previous JOREK version, one of the key achievements of this study is that, at equivalent numerical precision, having higher-order finite-elements can be computationally cheaper. As such, it is expected that the G2-continuous version of the JOREK code, possibly the G3-version, could become routinely used by JOREK users in their studies of tokamak MHD instabilities in the future.

6.2. Further work

There are, nevertheless, several areas where improvements can be brought to the JOREK code, following the implementation of this new method.

The first aspect is that a more generic definition of continuity, as in [52,53], could be considered at least for the G1- and G2-continuous versions. This is far more elaborate than the mathematical methods presented here, and would require significant code developments. However, there is an invaluable advantage to such generic methods, because they allow sharp angles between elements at a given node, which would enable more advanced element constructions, with arbitrary numbers of elements meeting at a given node, and the construction of triangular element sub-structures. This great flexibility could prove beneficial in the context of arbitrary grid construction, particularly in the context of alignment to arbitrary wall structures like in Fig.-11.

The second point that could be improved, and of significant impact, would be the elimination of the two equations for the auxiliary variables w_ϕ and j_ϕ in the reduced-MHD model. These equations are simply the identity definitions of $w_\phi = \nabla^2\Phi$ and $j_\phi = \nabla^2\psi$, where Φ and ψ are the electric and magnetic potentials respectively. These identities are time-independent, but they are included as separate equations in the model because it significantly reduces the level of numerical noise in the simulations, and improves the code's practical reliability. If higher-order continuity is available, these equations could be removed entirely, in principle, such that 3rd and 4th order derivatives of Φ and ψ would then be directly used in the momentum and induction equations (as opposed to 1st and 2nd derivatives of w_ϕ and j_ϕ at present). This would reduce the number of equations/variables from 7 to 5 for the standard reduced-MHD model, which would significantly reduce the overall cost of simulation.

The third aspect concerns boundary conditions for Mach-1 Sheath conditions for boundaries with incident magnetic field lines. At present, the Sheath boundary conditions are constrained with the weak-formulation, such that no assumption is made for higher-order derivatives, but the Mach-1 boundary conditions are enforced by a penalisation method projected onto specific node degrees of freedom. With higher-order finite-elements, this is done for first-order derivatives, but not 2nd or higher derivative degrees of freedom. It would be highly desirable to implement this feature in the code, but numerical difficulties were found in our initial attempts to implement this, and so this will require further investigation. However, the current version of the code, with Mach-1 boundary conditions applied to the values and first derivatives of the plasma velocity, is numerically very reliable and sufficient for production studies. It is, of course, at least as precise as the previous version.

The fourth area, where additional development could bring significant flexibility, is the possibility of having so-called "localised p-refinement". However, as demonstrated by Aumann & Bentz's work [48], ensuring continuity (at the lowest order of two adjascent elements) requires a far more elaborate formulation. In particular, if two patches meet with different p-refinements, their control points at the common boundary do not match, which is assumed in our case. Hence, for localised p-refinement, our formulation would need to be generalised to Aumann & Bentz's formula (18), but in nodal form and in 2-directions (for 4 elements). Nevertheless, p-refinement is an appealing capability to consider for future developments, and one that is already used in many other state-of-the-art modelling tools like Nektar++, Firedrake and MFEM [40,41,44,47,76,77].

Similarly, an interesting generalisation of the finite-element formulation in the JOREK code would be to allow for unconstrained finite-elements, where the continuity is not imposed. At present, the formulation inherently includes continuity,

which has the advantage of reducing the number of degrees-of-freedom, but the disadvantage of being less flexible. Removing this automatic constraint may lead to a computationally more costly formulation, but it could have other advantages, and the flexibility to impose continuity or not would certainly be of numerical interest. However, this would require a reformulation of the finite-element numerical structure inside the code.

Additionally, as mentioned earlier, finding an alternative proof to derive the G-continuity of our formulation (45), by providing a bridge to Aumann & Bentz's condition (18), would be mathematically interesting.

In terms of the mathematical method itself, the existence of such a generalised method for triangular Bezier elements would need to be demonstrated. It may also be of interest to address more generic finite elements than Bezier, like B-splines, and using basis functions other than the Bernstein polynomials.

Finally, now that this new method is available in the JOREK code, extensive studies and routine usability still remains to be demonstrated by the JOREK team, hopefully leading to cheaper, more precise simulations in the future.

CRediT authorship contribution statement

Both co-authors M. Hoelzl and G.T.A. Huijsmans provided invaluable advice in the redaction of the paper and conduction of the work therein.

In addition, G.T.A. Huijsmans is the main author of the JOREK code, into which the generalised formulation proposed here was implemented. The modern and modular implementation of the JOREK code by G.T.A. Huijsmans allowed for a surprisingly "effortless" implementation of the new FEM method. G.T.A. Huijsmans also provided considerable technical advice for many of the numerical tests performed for the manuscript.

Also, M. Hoelzl is currently the main administrator and facilitator of the JOREK code. His contributions are not only to the JOREK code itself, but also to the entire development structure in which JOREK resides (git platform, non-regression tests, pull-request organisations, wiki documentation, communications, etc.). This work would simply not have been possible without the background infrastructure provided by M. Hoelzl. Furthermore, M. Hoelzl provided invaluable input and comments for the redaction of the manuscript, as well as technical aspects of the numerical implementation.

Certified by main author: S.J.P. Pamela. Culham, UK, 20/Oct/2021

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was performed with the support of the JOREK Team [See https://www.jorek.eu for the present list of team members].

This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No. 101052200 — EUROfusion). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

This work has been carried out within the framework of the RCUK Energy Programme [grant number EP/I501045].

This work was performed using the MARCONI computer at CINECA in Italy, within the EUROfusion framework.

This work was performed using the Cambridge Service for Data Driven Discovery (CSD3), part of which is operated by the University of Cambridge Research Computing on behalf of the STFC DiRAC HPC Facility (www.dirac.ac.uk). The DiRAC component of CSD3 was funded by BEIS capital funding via STFC capital grants ST/P002307/1 and ST/R002452/1 and STFC operations grant ST/R00689X/1. DiRAC is part of the National e-Infrastructure.

The authors would like to thank Dr. James Buchanan, who kindly offered to provide an internal review of this manuscript before submission, and who pointed out several typos in the proof derivations, as well as many other useful comments.

References

[1] V. Masson-Delmotte, P. Zhai, A. Pirani, S.L. Connors, C. Pean, S. Berger, N. Caud, Y. Chen, L. Goldfarb, M.I. Gomis, M. Huang, K. Leitzell, E. Lonnoy, J.B.R. Matthews, T.K. Maycock, T. Waterfield, O. Yelekci, R. Yu, B. Zhou, Climate Change 2021: The Physical Science Basis. Contribution of Working Group i to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change, Cambridge University Press, 2021, in press.
[2] J. Wesson, D.J. Campbell, Tokamaks, Oxford University Press/Clarendon Press, Oxford/New York, 2004.
[3] D.D. Schnack, D.C. Barnes, D.P. Brennan, C.C. Hegna, E. Held, C.C. Kim, S.E. Kruger, A.Y. Pankin, C.R. Sovinec, Phys. Plasmas 13 (2006) 058103.
[4] O. Sauter, et al., Phys. Plasmas 6 (1999) 7.
[5] J.P. Freidberg, Ideal MHD, Cambridge University Press, ISBN 9780511795046, 2014.
[6] T. Eich, A. Herrmann, J. Neuhauser, Phys. Rev. Lett. 91 (2003) 195003.
[7] B. Sieglin, M. Faitsch, T. Eich, A. Herrmann, W. Suttrop, Phys. Scr. 2017 (2017) T170.
[8] C. Ham, A. Kirk, S. Pamela, H. Wilson, Nat. Rev. Phys. 2 (2020) 159–167.
[9] A.W. Leonard, Phys. Plasmas 21 (2014) 090501.
[10] C. Cheng, L. Chen, M.S. Chance, Ann. Phys. 161 (1985) 21.

[11] A. Dvornova, G.T.A. Huijsmans, S. Sharapov, F.J. Artola Such, P. Puglia, M. Hoelzl, S. Pamela, A. Fasoli, D. Testa, Phys. Plasmas 27 (2020) 012507.
[12] M. Fitzgerald, J. Buchanan, R.J. Akers, B.N. Breizman, S.E. Sharapov, Comput. Phys. Commun. 252 (2020) 106773.
[13] S.D. Pinches, I.T. Chapman, Ph.W. Lauber, H.J.C. Oliver, S.E. Sharapov, K. Shinohara, K. Tani, Phys. Plasmas 22 (2015) 021807.
[14] A.H. Boozer, Phys. Plasmas 19 (2012) 058101.
[15] P.C. de Vries, G. Pautasso, D. Humphreys, M. Lehnen, S. Maruyama, J.A. Snipes, A. Vergara, L. Zabeo, Fusion Sci. Technol. 69 (2016) 471–484.
[16] M. Lehnen, K. Aleynikova, P.B. Aleynikov, D.J. Campbell, P. Drewelow, N.W. Eidietis, Yu. Gasparyan, R.S. Granetz, Y. Gribov, N. Hartmann, E.M. Hollmann, V.A. Izzo, et al., J. Nucl. Mater. 463 (2015) 39–48.
[17] F.J. Artola, K. Lackner, G.T.A. Huijsmans, M. Hoelzl, E. Nardon, A. Loarte, Phys. Plasmas 27 (2020) 032501.
[18] D. Hu, E. Nardon, M. Lehnen, G.T.A. Huijsmans, D.C. van Vugt, Nucl. Fusion 58 (2018) 12.
[19] V. Bandaru, M. Hoelzl, F.J. Artola, G. Papp, G.T.A. Huijsmans, Phys. Rev. E 99 (2019) 063317.
[20] M. Hoelzl, G.T.A. Huijsmans, S.J.P. Pamela, M. Becoulet, E. Nardon, F.J. Artola, B. Nkonga, C.V. Atanasiu, V. Bandaru, A. Bhole, D. Bonfiglio, A. Cathey, O. Czarny, A. Dvornova, T. Feher, A. Fil, E. Franck, S. Futatani, M. Gruca, H. Guillard, J.W. Haverkort, I. Holod, D. Hu, S.K. Kim, S.Q. Korving, L. Kos, I. Krebs, L. Kripner, G. Latu, F. Liu, P. Merkel, D. Meshcheriakov, V. Mitteraur, S. Mochalskyy, J.A. Morales, R. Nies, N. Nikulsin, F. Orain, J. Pratt, R. Ramasamy, P. Ramet, C. Reux, K. Sarkimaki, N. Schwarz, P. Singh Verma, S.F. Smith, C. Sommariva, E. Strumberger, D.C. van Vugt, M. Verbeek, E. Westerhof, F. Wieschollek, J. Zielinski, The JOREK non-linear extended MHD code and applications to large-scale instabilities and their control in magnetically confined fusion plasmas, Nucl. Fusion 61 (2021) 065001.
[21] G.T.A. Huysmans, O. Czarny, Nucl. Fusion 47 (2007) 659.
[22] O. Czarny, G.T.A. Huysmans, J. Comput. Phys. 227 (2008) 7423.
[23] JOREK, https://www.jorek.eu/, 2020.
[24] M3D-C1, https://w3.pppl.gov/~nferraro/m3dc1.html, 2020.
[25] S.C. Jardin, J. Comput. Phys. 200 (133) (2004).
[26] NIMROD, https://nimrodteam.org/, 2020.
[27] C.R. Sovinec, A.H. Glasser, T.A. Gianakon, D.C. Barnes, R.A. Nebel, S.E. Kruger, D.D. Schnack, S.J. Plimpton, A. Tarditi, M.S. Chufth, the NIMROD Team, J. Comput. Phys. 195 (2004) 355.
[28] H. Lutjens, J.F. Luciani, J. Comput. Phys. 227 (14) (2008) 6944–6966.
[29] BOUT++, https://boutproject.github.io/, 2020.
[30] B.D. Dudson, M.V. Umansky, X.Q. Xu, P.B. Snyder, H.R. Wilson, Comput. Phys. Commun. 180 (2009) 1467–1480.
[31] Y. Todo, T. Sato, Phys. Plasmas 5 (1321) (1998).
[32] Y. Todo, H.L. Berk, B.N. Breizman, Nucl. Fusion 52 (2012) 094018.
[33] A. Konies, S. Briguglio, N. Gorelenkov, T. Feher, M. Isaev, Ph. Lauber, A. Mishchenko, D.A. Spong, Y. Todo, W.A. Cooper, R. Hatzky, R. Kleiber, M. Borchardt, G. Vlad, A. Biancalani, A. Bottino, Nucl. Fusion 58 (2018) 12.
[34] G. Strang, G. Fix, An Analysis of the Finite Element Method, 2nd ed., Wellesley-Cambridge Press, 2008, ISBN-10: 0980232708, ISBN-13: 978-0980232707.
[35] S.J.P. Pamela, A. Bhole, G.T.A. Huijsmans, B. Nkonga, M. Hoelzl, I. Krebs, E. Strumberger, Phys. Plasmas 27 (2020) 102510.
[36] E. Nardon, D. Hu, F.J. Artola, D. Bonfiglio, M. Hoelzl, A. Boboc, P. Carvalho, S. Gerasimov, G. Huijsmans, V. Mitteraur, N. Schwarz, H. Sun, Plasma Phys. Control. Fusion 63 (2021) 115006, https://doi.org/10.1088/1361-6587/ac234b.
[37] G.T.A. Huijsmans, et al., Phys. Plasmas 22 (2015) 021805.
[38] D.C. van Vugt, G.T.A. Huijsmans, M. Hoelzl, A. Loarte, N.J. Lopes Cardozo, Kinetic modelling of elm-induced w transport in asdex upgrade, Phys. Plasmas 26 (2019) 042508, https://doi.org/10.1063/1.5092319.
[39] C. Sommariva, E. Nardon, P. Beyer, M. Hoelzl, G.T.A. Huysmans, D. van Vugt, Test particles dynamics in the JOREK 3d non-linear MHD code and application to electron transport in a disruption simulation, Nucl. Fusion 58 (2018) 016043, https://doi.org/10.1088/1741-4326/aa95cd.
[40] R. Anderson, J. Andrej, A. Barker, J. Bramwell, J.-S. Camier, J. Cerveny, V. Dobrev, Y. Dudouit, A. Fisher, Tz. Kolev, W. Pazner, M. Stowell, V. Tomov, I. Akkerman, J. Dahm, D. Medina, S. Zampini, MFEM: a modular finite element methods library, Comput. Math. Appl. 81 (2021) 42–74, https://doi.org/10.1016/j.camwa.2020.06.009.
[41] MFEM: modular finite element methods [Software]. `mfem.org`, https://doi.org/10.11578/dc.20171025.1248.
[42] C. Geuzaine, J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities, Int. J. Numer. Methods Eng. 79 (11) (2009) 1309–1331.
[43] Gmsh website, https://gmsh.info/.
[44] F. Rathgeber, D.A. Ham, L. Mitchell, M. Lange, F. Luporini, A.T.T. Mcrae, G. Bercea, G.R. Markall, P.H.J. Kelly, Firedrake: automating the finite element method by composing abstractions, ACM Trans. Math. Softw. 43 (3) (2016) 24.
[45] Firedrake website, https://www.firedrakeproject.org/.
[46] Q. Tang, L. Chacon, T.V. Kolev, J.N. Shadid, X.-Z. Tang, An adaptive scalable fully implicit algorithm based on stabilized finite element for reduced visco-resistive MHD, Submitted article, https://arxiv.org/abs/2106.00260, 2021.
[47] F. Laakmann, P.E. Farrell, L. Mitchell, An augmented Lagrangian preconditioner for the magnetohydrodynamics equations at high Reynolds and coupling numbers, Submitted article, https://arxiv.org/abs/2104.14855, 2021.
[48] G. Aumann, K. Bentz, Geometrische stetigkeit beliebiger ordnung zwischen tensorprodukt-Bezier-flachen, J. Geom. 47 (1993) 1–15, https://doi.org/10.1007/BF01223799.
[49] X. Ye, Y. Liang, H. Nowacki, Geometric continuity between adjacent Bezier patches and their constructions, Comput. Aided Geom. Des. 13 (6) (1996) 521–548, https://doi.org/10.1016/0167-8396(95)00043-7.
[50] E. Hartmann, Numerical implicitization for intersection and G^n-continuous blending of surfaces, Comput. Aided Geom. Des. 15 (4) (1998) 377–397, https://doi.org/10.1016/S0167-8396(97)00040-X.
[51] T. Hermann, G. Lukacs, A new insight into the G^n continuity of polynomial surfaces, Comput. Aided Geom. Des. 13 (1996) 697–707, https://doi.org/10.1016/0167-8396(96)00005-2.
[52] W.-H. Du, F.J.M. Schmitt, On the G1 continuity of piecewise Bezier surfaces: a review with new results, Comput. Aided Des. 22 (9) (1990) 556–573, https://doi.org/10.1016/0010-4485(90)90041-A.
[53] J. Zheng, G. Wang, Y. Liang, Curvature continuity between adjacent rational Bezier patches, Comput. Aided Geom. Des. 9 (5) (1992) 321–335, https://doi.org/10.1016/0167-8396(92)90027-M.
[54] G. Hu, H. Cao, X. Wang, X. Qin, G2 continuity conditions for generalized Bezier-like surfaces with multiple shape parameters, J. Inequal. Appl. 2017 (2017) 248, https://doi.org/10.1186/s13660-017-1524-7.
[55] Binomial coefficient identities, https://en.wikipedia.org/wiki/Binomial_coefficient.
[56] MUMPS website: http://mumps.enseeiht.fr/.
[57] P.R. Amestoy, I.S. Duff, J. Koster, J.-Y. L'Excellent, SIAM J. Matrix Anal. Appl. 23 (1) (2001) 15–41, https://doi.org/10.1137/S0895479899358194.
[58] P.R. Amestoy, A. Guermouche, J.-Y. L'Excellent, S. Pralet, Parallel Comput. 32 (2) (2006) 136–156, https://doi.org/10.1016/j.parco.2005.07.004.
[59] P. Henon, P. Ramet, J. Roman, Parallel Comput. 34 (2008) 345–362.
[60] G. Pichon, E. Darve, M. Faverge, P. Ramet, J. Roman, J. Comput. Sci. 27 (2018) 255–270, https://doi.org/10.1016/j.jocs.2018.06.007.

[61] STRUMPACK website: https://portal.nersc.gov/project/sparse/strumpack/index.html.
[62] P. Ghysels, S.L. Xiaoye, C. Gorman, F. Rouet, in: IEEE International Parallel and Distributed Processing Symposium, IPDPS, 2017, pp. 897–906.
[63] I. Holod, M. Hoelzl, P.S. Verma, G.T.A. Huijsmans, R. Nies, the JOREK Team, Enhanced preconditioner for JOREK MHD solver, Plasma Phys. Control. Fusion 63 (2021) 114002, https://doi.org/10.1088/1361-6587/ac206b.
[64] Sympy Python library, https://www.sympy.org/en/index.html.
[65] Numpy Python library, https://numpy.org/doc/stable/contents.html.
[66] S. Pamela, G. Huijsmans, A.J. Thornton, A. Kirk, S.F. Smith, M. Hoelzl, T. Eich, Comput. Phys. Commun. 243 (2019) 41–50.
[67] S. Pamela, G. Huysmans, S. Benkadda, Plasma Phys. Control. Fusion 52 (2010) 7.
[68] R.J. Hastie, T.C. Hender, B.A. Carreras, L.A. Charlton, J.A. Holmes, Phys. Fluids 30 (1987) 1756.
[69] P.L. Pritchett, Y.C. Lee, J.F. Drake, Phys. Fluids 23 (1980) 1368.
[70] R.A. Pitts, X. Bonnin, F. Escourbiac, H. Frerichs, J.P. Gunn, T. Hirai, A.S. Kukushkind, E. Kaveeva, M.A. Miller, D. Moulton, V. Rozhansky, I. Senichenkov, E. Sytovaaf, O. Schmitz, P.C. Stangeby, G. De Temmerman, I. Veselova, S. Wiesen, Physics basis for the first ITER tungsten divertor, Nucl. Mater. Energy 20 (2019) 100696, https://doi.org/10.1016/j.nme.2019.100696.
[71] G. Huijsmans, D. van Vugt, S. Franssen, S. Korving, M. Becoulet, Non-linear MHD simulations of ELMs in a high recycling divertor, in: 46th European Physical Society Conference on Plasma Physics, EPS 2019, Milan, Italy, 2018, https://hal.archives-ouvertes.fr/cea-02555093.
[72] F. Orain, M. Becoulet, G. Dif-Pradalier, G. Huijsmans, S. Pamela, E. Nardon, C. Passeron, G. Latu, V. Grandgirard, A. Fil, A. Ratnani, I. Chapman, A. Kirk, A. Thornton, M. Hoelzl, P. Cahyna, Phys. Plasmas 20 (2013) 102510.
[73] A. Cathey, M. Hoelzl, K. Lackner, G. Huijsmans, M. Dunne, E. Wolfrum, S. Pamela, F. Orain, S. Gunter, Nucl. Fusion 60 (2020) 124007.
[74] A.Y. Aydemir, J.Y. Kim, B.H. Park, J. Seol, On resistive magnetohydrodynamic studies of sawtooth oscillations in tokamaks, Phys. Plasmas 22 (2015) 032304, https://doi.org/10.1063/1.4914090.
[75] B. Nkonga, J. Tarcisio-Costa, Jeaniffer Vides, Vms finite element for MHD and reduced-MHD in tokamak plasmas, [Research Report] RR-8892, Inria Sophia Antipolis/Université de Nice-Sophia Antipolis, 2016, https://hal.inria.fr/hal-01294788.
[76] C.D. Cantwell, D. Moxey, A. Comerford, A. Bolis, G. Rocco, G. Mengaldo, D. De Grazia, S. Yakovlev, J-E. Lombard, D. Ekelschot, B. Jordi, H. Xu, Y. Mohamied, C. Eskilsson, B. Nelson, P. Vos, C. Biotto, R.M. Kirby, S.J. Sherwin, Comput. Phys. Commun. 192 (2015) 205–219.
[77] D. Moxey, C.D. Cantwell, Y. Bao, A. Cassinelli, G. Castiglioni, S. Chun, E. Juda, E. Kazemi, K. Lackhove, J. Marcon, G. Mengaldo, D. Serson, M. Turner, H. Xu, J. Peiro, R.M. Kirby, S.J. Sherwin, Comput. Phys. Commun. 249 (2020) 107110.