An isogeometric analysis approach for coupled multi-field problems at large strain

Lei Zhang

ECOLE CENTRALE MARSEILLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'ECOLE CENTRALE MARSEILLE

Par

LEI ZHANG

le 5 décembre 2016

SUR UNE APPROCHE ISOGEOMETRIQUE POUR PROBLEMES MULTI-CHAMPS COUPLES EN GRANDES TRANSFORMATIONS

JURY

Rapporteur: Thomas Elguedj, INSA de Lyon

Rapporteur: Jean-Claude Léon, ENSE3 - Grenoble-INP

Examinateur: Salim Bouabdallah, Altair Engineering

Examinateur: Marc Médale, Polytech' Marseille

Directeur de thèse: Dominique Eyheramendy, Centrale Marseille

Acknowledgements

This dissertation has been written from 2013 to 2016 during my time as Phd student in the equip of material and structure of LMA (Laboratoire de Mécanique et d'Acoustique), Marseille, France. I would like to thank sincerely Prof. Dominique Eyheramendy for giving me the opportunity to work with him, and for his helpful and patient guidance as my doctoral supervisor. I also want to express my thanks to Dr. Stéphane Lejeunes, who has been of great assistance and inspiration to me, and also contributed to the work included in this thesis.

Furthermore, I would like to address my thanks to the member of my examining jury Dr. Thomas Elguedj and Prof. Jean-Claude Léon. Their interest in my work is gratefully appreciated. Also, I want to thank Dr. Salim Bouabdallah and Prof. Marc Médale to participate to the judging panel.

The funding for my whole work as research scholar was granted by the China Scholarship Council (CSC). This funding is gratefully acknowledged.

Finally, I want to thank my parents: Laquan Zhang and Shuhui Chen for their support and encouragement at all times.

张 磊

Lei Zhang

Marseille, France

October, 2016

Abstract

Recently proposed as a general purpose numerical method, the Isogeometric Analysis (IGA) offers great perspective to bridge the gap between CAD and CAE. The IGA is closely related to the finite element method (FEM) as the method is based on the same variational framework. Moreover, this method has shown in many circumstances to be have a better accuracy than the FEM (large mesh distortions…). Our final aim in this work is to simulate complex multiphysics problems for elastomers industrial parts. As matter of fact, the two main numerical issues in this context is the incompressibility/quasi-incompressibility of the material and the thermochemical coupling in Galerkin formulations. First, we propose, a programming paradigm of the IGA in an existing Java object-oriented hierarchy initially designed for solving multi-fields coupled problems at finite strains. We develop an approach that fully take benefit of the original architecture to reduce developments for both FEM and IGA (one problem developed in FEM can be run in IGA and vice versa). Second, we investigate volumetric locking issues persisting for low order NURBS element observed with standard displacement formulation as finite elements. To cure the problem, we adopt two-fields mixed formulation (displacement/pressure) for the sake of simplicity and target at assessing different discretizations in stability (*inf-sup* condition). The basic idea is to first to increase the internal knot's multiplicity or to subdivide the patch for displacements. These ideas that are directly inspired from patches properties, have been found in the literature for the Stokes problem and extended to large strain in solid mechanics. The comparison between the two-fields mixed formulation and a strain projection method is lead at small and large strains. At last, we originally adopt a similar strategy for thermomechanical problem at small and large strains. In the context two-fields formulation, displacement/temperature, the LBB stability condition must be fulfilled to guaranty stability. Thus, we investigate the choices of patches for two-fields formulation displacement/temperature fields for IGA applied to thermoelasticity. Several numerical results for thermomechanical problems at small and finite strains, linear and nonlinear have been presented. At last, an incompressible viscous thermo-hyperelastic model is evaluated in the IGA framework with the proposed approach.

Key words: object-oriented programming, isogeometric analysis, volumetric locking, multiphysics, large strains

Résumé

La méthode isogéométrique (IGA) récemment proposée en tant que méthode numérique générique offre de réelles perspectives dans l'unification des modèles géométriques et computationnel. La méthode isogéométrique est intiment liée à la méthode des éléments finis (FEM) étant donné que la méthode est basée sur le même cadre variationnel. Cette méthode a montré dans de nombreuses circonstances de très bonne qualités numériques notamment avec des maillages grossiers (précision numérique, capacité à supporter de grandes déformations…). Notre objectif final dans ce travail est de fournir un environnement de base, numérique et logiciel, pour la simulation de problèmes à champs et physiques multiples pour des pièces élastomériques de type industriel. Dans ce contexte, les points numériques à développer pour l'IGA sont le traitement de l'incompressibilité et le caractère multi-champs du problème thermique dans la formulation de Galerkin. Ainsi dans ce travail nous proposons en premier, un paradigme objet de l'IGA intégré au sein d'une architecture orientée objet en Java, initialement conçue pour résoudre des problèmes multi-champs couplés en transformations finies. L'approche proposée s'appuie pleinement sur le contexte variationnel existant dans le code dans le cadre des éléments finis pour réduire les développements pour MEF et IGA (une formulation développée en IGA tourne en MEF et vice versa). Dans un second temps, nous avons étudié le problème de l'incompressibilité pour notamment réduire le verrouillage numérique existant toujours sur l'IGA standard. Par un souci de simplicité, nous adoptons des formulations mixtes à 2 champs (déplacement/pression). Afin d'essayer de satisfaire la condition *inf-sup* en relâchant la contrainte sur le déplacement, nous avons développé deux idées de la littérature (naturelle en NURBS) qui consiste à soit dupliquer une fois les nœuds intérieurs du patch des déplacements ou subdiviser les éléments du patch des déplacements. Nous avons étendu ce type d'éléments aux transformations finies. Enfin, et de manière originale, nous avons adopté la même stratégie pour les problèmes à 2-champs pour la thermomécanique. Différentes simulations à petites et grandes déformations confirment le potentiel de l'approche. Enfin, nous évaluons l'ensemble sur un modèle quasi-incompressible thermo-visco-élastique de type Zener sur des éprouvettes classiques dans un contexte physique complexe.

Mots clés : programmation orientée objet, analyse isogéométrique, verrouillage numérique, problèmes multiphysiques, grandes déformations.

Résumé Etendu

- **Principes de l'analyse isogéométrique**

De nos jours, l'utilisation de fonction splines est très répandu en infographie. Dans le domaine de la CAO, les splines les plus utilisées sont les NURBS. Les B-Splines et NURBS appartiennent à la famille des fonction paramétriques. L'attrait de ce type de fonction est que la définition de cet espace paramétrique et l'élaboration des fonctions de base associée est simple. En 1D, l'espace paramétrique est défini sur un intervalle sur lequel on définit un vecteur de nœuds, ensemble de nœuds ordonnés (dupliquée ou non) de manière croissante. Les fonctions de base des B-Spline se définissent alors de manière récursive par la relation de Cox-De Boor :

Soit le vecteur de nœuds : $\Xi = \{\xi_1, \xi_2, \xi_3, \dots, \xi_m\}$

Les fonctions de bases sont alors définies par :

- *If p=0*

$$N_{i,0}(\xi) = \begin{cases} 1 & if\ \xi_i < \xi < \xi_{i+1} \\ 0 & else \end{cases}$$

- *If p>=1*

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi)$$

où ξ_i est le $i^{ème}$ nœud du vecteur de nœuds Ξ est le vecteur des nœuds, p est degré du polynôme. Ces fonctions sont C^p sur l'intervalle entre 2 nœuds et C^{p-k} si l'ordre de multiplicité du nœud *i* est *k*. Ces fonctions de base permettent de construire le changement de base de l'espace paramétrique vers l'espace physique. Par exemple, une courbe B-Spline sera donnée par $C(\xi) = \sum_{i=1}^{n} N_{i,p} \cdot B_i$ Où les B_i sont les points de contrôle. L'extension à des surfaces ou volume se fait par produit tensoriel des fonctions 1D. Les fonctions B-Splines permettent de représenter des formes complexes mais pas des coniques. Les NURBS (Non-Uniform Rational B-Splines) le permettent. Leur construction se base sur des fonctions rationnelles construite à partie des fonction B-Splines. Un des enjeux pour

les B-Splines et NURBS est l'évaluation efficace des fonctions de bases et de leurs dérivées. Ces bases peuvent être enrichies soit par ajout de nœuds, soit par élévation de degré. Il faut noter que l'enrichissement continue de représenter exactement la même géométrie. Il peut se faire, soit en insérant un nouveau nœud, soit élevant le degré des fonctions de base. Dans les 2 cas, le nombre de points de contrôles des géométrie (courbe, surface, volume) augmente. L'idée clé de l'IGA est d'utiliser ces fonctions B-Splines ou NURBS en lieu et place des fonctions d'interpolation classiques de la méthode des éléments finis. Le cadre variationnel du problème continue reste identique à celui des éléments finis. De la même manière qu'en éléments finis, pour avoir des solutions plus précises on cherchera à affiner le maillage. Comme en éléments finis, en IGA on peut faire du raffinement-h (insertion de nœuds) ou du raffinement-p (augmentation du degré des fonctions de bases utilisées). En IGA, on peut aussi avoir une stratégie couplée le raffinement-k obtenu en premièrement insérant des nœuds et en deuxièmement élevant le degré. Ce raffinement s'avère être très performant en termes de convergence.

D'un point de vue historique, on trouve des publications dans lesquelles des auteurs ont très tôt proposé d'utiliser des B-Splines ou NURBS comme fonctions d'interpolation en éléments finis ou méthodes voisines (voir par exemple Schultz et al. [1], Höllig [2], Natekar [3], Cassale et al. [4]). D'autre auteurs, en CAO, ont utilisés des méthodes numériques type éléments finis à des fins d'optimisation des géométries (voir par example Celinker et al. [5],, Terzopoulos et al. [6]).

Le travail qui a fondé l'IGA en posant les bases est celui de Hughes et al. [7]. On peut dire qu'aujourd'hui tous les domaines de la mécanique computationnelle. Dans ce résumé nous ne détaillerons pas une bibliographie compète que l'on trouvera dans ce manuscrit, mais citons les sujets actifs des années passées : en mécanique des structures (plaques, coques, poutres…), mécanique des fluides (Stokes, Navier-Stokes, interaction fluides structures), simulations des milieux incompressibles, milieux biphasé, problèmes de contact, vibrations, méthodes POD, électromagnétisme, acoustique, lubrification, milieux poreux… D'un point de vue numérique de nombreux point été soulevés, et celui de l'intégration numérique des formes élémentaires dans le cadre de l'IGA reste le plus difficile car très couteux dès que l'on passe au 3D. Une adaptation des méthodes de collocation permet de ne pas être

confronté à ce problème et a été développé en élasticité linéaire statique ou dynamique, plaques…

Dans le même temps, des alternatives aux B-Splines NURBS se sont développées avec des objectifs variables : B-Splines hiérarchiques qui permettent de se séparer de la structure tensorielle en multidimensionnel et donc avoir des stratégies locales, T-Splines pour décrire des topologies complexes, surface de subdivision pour représenter des géométries complexes…

- **Intégration de l'Analyse Isogéométrique dans un environnement orienté objet pour éléments finis**

Depuis les années 50 et les origines de l'informatique, le génie logiciel a considérablement évolué s'appuyant sur l'évolution des langages de programmation. Dans les années 90, dans le développement logiciel, y compris en mécanique computationnelle, les approches de programmation traditionnelles (approches procédurales en Fortran, C…) ont rapidement posé des problèmes dans le développement à grande échelle (complexité et taille du code). La maintenance et l'extension des codes s'est avérée difficile à assurer. Introduite dans le domaine des éléments finis par Rehak et al [8] et Miller [9], la programmation orientée objet a permis, grâce notamment à l'encapsulation des données, de mieux contrôler les flux de données. Zimmermann et Pèlerin [10] ont été les premiers à proposer une implémentation complète de la méthode des éléments finis en élastodynamique en introduisant notamment l'objet degré de liberté. De nombreux auteurs ont proposés dans le même temps des implémentations similaires. Besson et al. [11] ont été les premier à proposer une étude approfondie de l'objet matériaux. Depuis cette époque, la programmation orientée objet a été appliquée dans un grand nombre de domaines de la mécanique. Plus récemment, des langages tel que Java ou C# ont été utilisés dans le but soit d'améliorer la structuration des codes, d'utiliser les capacité réseaux pour effectuer du calcul parallèle, de profiter de la portabilité des codes ou de mixer des développements multi-langages avec de bonnes performances (voir Eyheramendy et al. [12]). Une voie de recherche a été ouverte dans les années 70 afin d'accélérer le développement de codes éléments finis et d'en généraliser la structuration par l'introduction des formes

mathématiques dans des environnements à haut niveau d'abstraction (voir Saad et al. [13], [14], Korelc et al. [15] et Logg et al. [16]).

Dans le domaine de l'analyse isogéométrique, de nombreux développements ont été publiés dans la littérature. On trouve principalement des codes de type recherche écrits en Matlab (Vuong et al. [17], Nguyen et al. [18], Falco et al. [19],…). Certains de ces développements proposent des structurations de type objet (Falco et al. [19]). De nombreux développements dans des langages performants existent aujourd'hui en général comme extension de codes de calculs éléments finis classiques ou de codes industriels : PetIGA extension de PETSc pour l'analyse isogéométrique Dalcin et al. [20] bénéficiant des implémentation hautes perfomances de PETSc et utilisé dans de nombreuses applications, une intégration à Abaqus proposée par Duval et al. [21], IEFM de Kvarving et al. [22] implémentation C++ pour solides en linéaire et non linéaire, … La plupart des packages se concentrent sur l'aspect approche variationnel de l'analyse isogéométrique, certains cependant abordent les aspects constitutifs : OOFEM et son extension IGA Rypl et al. [23] et Duval et al. [21] basé sur Abaqus. Dans ce travail, nous proposons un modèle orienté objet complet et totalement intégré, transparent dans un code en langage Java de type éléments finis (code FEMJava). Le code FEMJava est à l'origine un code orienté objet en langage Java élaboré pour les modèles éléments multiphysiques couplés à champs multiples. Les différentes notions nécessaires au code sont séparées en packages thématique dont les noms sont naturels : ***algorithm***, ***distributedcomputing***, ***fem***, ***field***, ***fortranlibraries***, ***geometry***, ***graphics***, ***imposedvalues***, ***material***, ***mathool***, ***mesh***, ***quadrature***. La description de la géométrie est disjointe des champs inconnus, on peut donc changer la discrétisation des champs pour une même formulation. La clé de voute de la structuration de l'ensemble est la notion de champ (scalaire, vectoriel, tensoriel d'ordre 2, …). D'un point de vue pratique, le champ discret (donnée globale sur le domaine) s'appuie sur le maillage du domaine (ensemble de géométries élémentaires –triangle, quadrangle, hexaèdre…-), qui lui-même s'appuie sur le domaine de calcul décrit par des géométrie (points, lignes, surfaces…). L'élément, au sens des éléments finis implémente les formes élémentaires de la formulation et couple les champs élémentaires nécessaires à la formulation. L'originalité de ce travail consiste à l'intégration d'un nouvel objet **Patch** pour lequel l'élément (au sens analyse isogéométrique) joue le rôle de support d'intégration,

tout comme l'objet éléments dans les codes éléments finis orientés objets traditionnels. Dans ce contexte, la généralisation du support d'intégration numérique permet une intégration naturelle du patch de l'analyse isogéométrique au sein du même cadre éléments finis. Une adaptation de la gestion des conditions de bord a été nécessaire. Le cadre à objets proposé est ainsi totalement unifié MEF et IGA. Nous illustrons l'approche sur des problèmes d'élasticité linéaire simples.

- **Modélisation du comportement des élastomères et implémentation du matériau « élastomère »**

Une des caractéristiques importantes des élastomères est de supporter de grandes déformations qui se font à variation de volume nulle ou quasi nulle. On les modélise souvent par un comportement incompressible ou quasi incompressible.

Les méthodes éléments finis de type déplacement classique échouent lorsque l'on tend vers l'incompressibilité du matériau, c'est à dire quand le coefficient de Poisson tend vers 0.5. Dans ces conditions, on peut observer des oscillations importantes dans les solutions pour les contraintes et pour des problèmes confinés on peut observer un verrouillage des déplacements. De nombreuses méthodes ont été développées en éléments finis pour gérer l'incompressibilité/quasi-incompressibilité. Une des premières méthodes proposées par Naylor [24] a été de réduire le nombre de points d'intégration pour la partie volumétrique des déformations dans le calcul des contraintes moyennes redistribuées. Cette méthode a été généralisée par Malkus et Hughes [25], dans lequel ils ont montré l'équivalence des méthode d'intégration réduite sélective avec les méthodes mixtes sous certaines conditions. Les méthodes de Galerkin moindres-carrés largement développés par Hughes et al [26]–[28] est basée sur l'ajout de termes de type moindres-carrés à la formulation de Galerkin de base et dont la fonction est de stabiliser mathématiquement. D'autres méthodes ont été développées à la même époque. Citons par exemple : la méthode FIC d' Oñate [29], les méthodes stabilisées par enrichissement des interpolations par une fonction bulle Brezzi et al [30]–[33]. Dans les années 90, Hughes et al [34], [35] ont développé une méthode variationnelle multi-échelle dont le formalisme a permis de développer des méthodes pour les milieux incompressibles ou faiblement compressibles (voir Masud et al. [36] pour

l'incompressibilité en fluide, Nakshatrala et al. [37] pour l'incompressibilité en solide et Cervera et al [38] pour la plasticité). Les méthodes mixtes à 2 champs (déplacement/pression) ont été initialement développées par Hermann et al. [39] et Taylor et al. [40] en petites perturbations et par Argyris et al. [41] en grandes déformations. Ces méthodes permettent, avec un choix judicieux de fonctions d'interpolation, d'avoir des formulations stables. Des méthodes à trois champs (déplacement/pression/déformations) basés sur une formulation de Hu-Washizu (voir par exemple Simo & Taylor [42], Jankovich [43], ou Oden et al. [44] pour l'élasticité incompressible). Les méthodes EAS (Enhanced Assumed Strain) proposée par Simo et al [45] permettent de faire de plus avoir de verrouillage volumique en relâchant les contraintes entre les différents champs. Dans le cadre l'analyse isogéométrique, des approches purement déplacement ont été développées. Elguedl et al. [46] ont étendu le formalisme $\bar{B}$ and $\bar{F}$ à l'IGA. Adams et al. [47] ont proposé une extension des approches d'intégration sélective réduites. En éléments finis mixtes, la seule possibilité de relâcher les contraintes entre déplacement et pression (vitesse/pression en fluides) et de faire varier l'ordre des interpolations pour les 2 champs. Avec la méthode isogéométrique, on peut en plus faire varier la continuité inter éléments en augmentant l'ordre de multiplicité des nœuds internes au patch ou alors en subdivisant un élément en deux. Dans les deux cas, on augmente la proportion de de degrés de libertés d'inconnues cinématique par rapport à ceux en pression, ce qui peut permettre sous certaines conditions de satisfaire la condition *inf-sup*. Dans les approches mixtes à 2 champs, on trouvera les travaux de Buffa et al [48] où sont comparés la version IGA des éléments de Raviart-Thomas et Nédélec à des éléments pour lesquels l'ordre de multiplicité des nœuds intérieurs est augmenté $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ où p est le degré des NURBS ou B-Spline avec la nomenclature du paragraphe 4.1). De même dans les travaux Nielsen et al [49], des couples d'interpolations qui garantissent la stabilité des formulations pour les problèmes de Stokes et Navier-Stokes sont proposés, de la même manière que Buffa et pour un schéma de subdivision $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ avec la nomenclature du paragraphe 4.1. Kapada étend lui à la mécanique des solide pour l'incompressibilité (petites perturbations et grandes déformations) avec des schémas de subdivision. Des méthode IGA mixtes à trois champs sont proposées dans Taylor et al. [50], Cardoso et al. [51] sur des bases de formulation de Hu-Washizu. Dans cette étude, au regard de la bibliographie, nous allons chercher plutôt à

travailler avec des formulations simples de type déplacement ou mixte à 2 champs. On vise effectivement des applications pour élastomères en thermomécanique pour lesquelles à terme nous introduirons d'autres physiques. Nous avons donc besoins de schémas plutôt légers. D'autre part, les schémas avec intégration réduite sélective ne sont pas forcements généralisables pour des formulations très nonlinéaires (anisotropie des opérateurs tangents pour les applications visées). Nous évaluerons tout de même les formations de type $\bar{B}$ and $\bar{F}$ bien que nous sachions que le calcul des opérateurs tangents pour la thermomécanique est extrêmement lourd en calcul, et nous le développerons pas. Les résultats obtenus dans la littérature par Buffa et Nielsen en petites déformation couvre des combinaisons d'interpolation $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p-1_0^1)$ $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p-1_0^1)$ $(\boldsymbol{u}\, p_0^1, \boldsymbol{p}-\boldsymbol{1}\, p_0^1)$. Nous allons les comparer dans le cadre des petites perturbations et des grandes déformations et les comparer aux approches $\bar{B}$ and $\bar{F}$.

La première série de comparaisons concerne les petites déformations. Le problème de base est une formulation mixte du problème d'élasticité incompressible. Le premier test est une cavité de forme carré. La transformation de l'espace paramétrique vers l'espace physique est ici triviale, ce qui permet de se concentrer sur la convergence de la méthode. On impose pour ce problème des charges volumiques qui équilibre une solution en déplacement et pression connue. On constate que pour des patches d'interpolation identiques, la solution en vitesse semble correcte mais la pression est instable comme on s'y attend. Par contre, instabilité diminuent quand on augmente le degré des NURBS mais il n'y a pas convergence en norme L^2 de l'erreur en pression. Pour les combinaisons $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ $(\boldsymbol{u}\, p+1_0^1, \boldsymbol{p}\, p_0^1)$ les convergences en norme L^2 de l'erreur en déplacement et pression sont optimales. On remarque que cependant le niveau de l'erreur est d'un ordre plus bas pour le déplacement. On a les mêmes observations pour des éléments $(\boldsymbol{u}\, p+1_1^1, \boldsymbol{p}\, p_0^1)$ et $(\boldsymbol{u}\, p+1_0^2, \boldsymbol{p}\, p_0^1)$. On remarque cependant que le niveau de l'erreur à maillage équivalent est plus faible d'un ordre pour le déplacement pour $(\boldsymbol{u}\, p+1_0^2, \boldsymbol{p}\, p_0^1)$. Cela ne prouve pas la stabilité de ce type d'élément mais rend optimiste sur sa stabilité. Le problème de la cavité cisaillée (équivalent au problème de la cavité entrainée pour le problème de Stokes) vise à vérifier la capacité de la formulation pour le calcul en pression aux coins de la cavité et sa capacité à gérer le verrouillage (fort gradient en pression, et

problème très confiné). Les profils de déplacement pour en *x* et *y* sont équivalents pour $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ et pour les éléments finis classiques Q2/Q2 et Q2/Q1 (éléments quadrangulaires de degré 1 ou 2). Par contre, les pressions sont instables pour $(\boldsymbol{u}\, p_0^1, \boldsymbol{p}\, p_0^1)$ et Q2/Q2. Tout de même pour $(\boldsymbol{u}\, p_0^1, \boldsymbol{p}\, p_0^1)$ les instabilités sont faibles. Enfin on peut remarquer que pour la pression au coin, l'interpolation $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ produit un pic plus raide (moins diffusif) que $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$. Le problème de la poutre de Cook en petites perturbations est dominant en cisaillement et flexion. On étudie ici le déplacement du coin droit supérieur de la poutre, et son évolution en fonction du nombre de degrés de liberté ou du nombre d'éléments. On montre que pour les éléments $\bar{B}$ Qp/Qp-1 (déplacement de degré *p* et projection sur l'espace de degré *p-1*) de Elguedj (formulation déplacement), et les éléments $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ et $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ le verrouillage est considérablement réduit pour les bas degrés, voir annulé pour des degrés plus élevés. Le test en degré 3 pour comparer $\bar{B}$ Q3/Q2, $(\boldsymbol{u}\, 3_0^2, \boldsymbol{p}\, 3_0^1)$ et $(\boldsymbol{u}\, 3_1^1, \boldsymbol{p}\, 3_0^1)$ et Q3 (qui verrouille) montre des résultats équivalents, les éléments $\bar{B}$ Q3/Q2 et $(\boldsymbol{u}\, 3_0^2, \boldsymbol{p}\, 3_0^1)$ étant identiques. EN ce qui concerne la trace des contraintes, on remarque que les formulations $\bar{B}$ il reste un peu d'instabilité alors que pour $(\boldsymbol{u}\, 2_0^2, \boldsymbol{p}\, 2_0^1)$ et $(\boldsymbol{u}\, 2_1^1, \boldsymbol{p}\, 2_0^1)$ il n'y en a pas du tout. Cela nous fait préférer les formulations mixtes à ce niveau. En grandes déformations, on considère des matériaux hyperélastiques. Pour le problème de Cook, avec un matériau élastique de type Néo-Hookéen pour lequel l'énergie libre est scindée en deux parties, une isochore l'autre volumique ; le matériau est considéré faiblement compressible. On montre que pour divers ordres, les éléments $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ et $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ et la formulation $\bar{F}$ d'Elguedj ne verrouillent pas. Plus le degré est élevé, plus le verrouillage disparaît rapide (avec la finesse du maillage). A même degré, les formulations mixtes $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ et $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ sont un tout petit peu meilleures que la formulation $\bar{F}$ (en degré 3 pour le déplacement). On observe comme en petits déplacements que pour la trace des contrainte de Cauchy de petites instabilités persistent pour $\bar{F}$ mais pas pour $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ et $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$. Enfin, dans les itérations de type Newton, les formulations $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ et $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ convergent bien mieux que la formulation $\bar{F}$. On teste ensuite la capacité des formulations à supporter de grandes déformations dans le cadre de la compression d'un bloc hyper-élastique.

L'originalité de cette partie est d'avoir testé et comparé ces différentes formulations et éléments en transformations finis. La grandeur d'intérêt est le niveau de compression. Dans ces tests, les éléments de type $(\boldsymbol{u}\ p_0^2, \boldsymbol{p}\ p_0^1)$ et $(\boldsymbol{u}\ p_1^1, \boldsymbol{p}\ p_0^1)$ converge vers la solution plus vite que l'élément $\bar{F}$.

- **Thermomécanique et IGA**

L'idée est de reproduire ces tests dans le cadre de la thermomécanique. On va se concentrer sur des combinaisons du type $(\boldsymbol{u}\ p_0^2, \boldsymbol{p}\ p_0^1)$ et $(\boldsymbol{u}\ p_1^1, \boldsymbol{p}\ p_0^1)$ qui nous sont apparues être le plus performantes du point de vue de la stabilité dans le traitement de l'incompressibilité aussi en petites perturbations qu'en transformations finies Cette problématique n'a pas été étudiée à notre connaissance en IGA.

En éléments finis, les champs de déplacement et de température doivent respecter une règle de compatibilité. Dans le contexte d'un problème à deux champs, la condition de stabilité de Ladyzhenskaya-Babuska-Brezzi ou condition inf-sup. Le lecteur trouvera dans Xue et al. [52] une extension aux formulations à multiples variables mixtes basée sur la condition inf-sup. Dans Prathap et al [53], un exemple simple en 1D illustre l'incompatibilité entre un champ de déplacement linéaire et un champ de température linéaire. Pourtant dans la pratique, on trouve de nombreuses publications dans lesquelles cette condition n'est pas respectée, et des solutions acceptables du point de physique sont calculées. Depuis les années 70 de nombreux auteurs ont proposés des formulations, schémas éléments finis et algorithmes associés (couplés, étagés, …) pour résoudre le problème. En ce qui concerne les formulations éléments finies, on peut classer la littérature abondante en trois catégories. Dans la première, on trouvera les formulations 2 champs déplacement/température basées sur des interpolations compatibles pour déplacement/température, par exemple éléments de Lagrange déplacement quadratique, température linéaire (voir par exemple Keramidas et al [54], Carter et al [55], Rao et al [56]…). Comme en incompressibilité, des auteurs ont proposé d'utiliser des techniques d'intégration réduite sélective, par exemple Juhre et al. [57]. Enfin des méthodes mixtes ont été développées afin d'obtenir de solutions en contraintes et flux thermiques précises. Basée sur des formulations de type Hu-Washizu,

Hellinger-Reissner, ces formulations incluent des champs en plus des déplacements et températures: contrainte, déformations, flux thermique, gradient thermique… Un choix judicieux basé sur des extensions de la condition inf-sup sont alors nécessaires (voir Miranda et al. [58], Prathap [53], Zhu et al. [59], Cannarozzi et al. [60]…). Enfin, dans Dittmman et al. [61] , une formulation du contact par la méthode 'mortar', la discrétisation en déplacement et température étant de type NURBS. Des patchs de même ordre sont utilisés pour déplacement et la température et donnent pourtant des résultats très corrects. Cela n'est cependant pas complètement étonnant en IGA ou pour des maillages que l'on pourrait considérer assez grossier, on peut déjà atteindre des niveaux de précision suffisants pour réduire d'éventuelles oscillations. Dans ce travail, nous avons cherché à développer des formulations à 2 champs (déplacement/température) pour les mêmes soucis de simplicité que précédemment avec des combinaisons que nous cherchons stables du même type que pour traiter les problèmes de faible compressibilité, à savoir : $(\boldsymbol{U}\ p_1^1, \boldsymbol{T}\ p_0^1)$ $(\boldsymbol{T}\ p_0^2, \boldsymbol{T}\ p_0^1)$ $(\boldsymbol{U}\ p_1^1, \boldsymbol{T}\ p-1_0^1)$ $(\boldsymbol{U}\ p_1^1, \boldsymbol{T}\ p-1_0^1)$ $(\boldsymbol{U}\ p_0^2, \boldsymbol{T}\ p-1_0^1)$.

Dans des tests en élasticité linéaire en petites perturbations, on montre sur un exemple simple (cavité carré, chargement imposé élaboré à partir d'une solution visée) la convergence optimale des combinaisons d'interpolation (presque optimale en ce qui concerne les déplacements) pour un couplage faible des équations d'élasticité et de thermique (coefficient de dilatation thermique petit devant le module de Young). Par contre, lorsque le couplage augmente, c'est à dire lorsque on augmente le coefficient de dilatation thermique, le taux de convergence décroit pour l'erreur en déplacement. Cela signifie qu'il faudra prendre garde si l'on souhaite utiliser ce type de formulation, à ce que le couplage entre les équations de la mécanique et de la thermique soit trop important. Un test sur cylindre épais dont on connait la solution exacte, avec couplage fort des équations (coefficient de dilatation thermique et module d'Young du même ordre de grandeur) montre des résultats encourageant : convergence optimale pour la température (ce qui est normal étant donné qu'il n'y a pas de couplage sur l'équation de la thermique) et convergence pour la norme L^2 de l'erreur en déplacement converge avec une pente de 2 quel que soit le degré d'interpolation. On remarque que l'erreur en déplacement est un plus petite pour le couple $(\boldsymbol{T}\ p_0^2, \boldsymbol{T}\ p_0^1)$ à taille de maille égale. Il faut noter que pour ce test, le couplage est fort, et que la transformation qui passe de l'espace paramétrique vers l'espace

physique est plus complexe que pour la cavité carrée. C'est peut-être une des origines de la non convergence optimale de l'erreur en déplacement. Sur le même problème, un test préliminaire de couplage de patch par la méthode de Nitsche nous donne des résultats de convergence tout à fait similaires. On réalise des tests en transformations finies, pour des matériaux hyperélastiques. Pour un problème d'évolution dont on connait une solution 1D exacte (poutre encastrée soumise à un déplacement imposé dépendant du temps et une énergie libre pour le matériau couplant mécanique et température, on montre que l'on peut capturer la solution avec un maillage régulier très grossier en IGA par rapport aux EF (dix fois moins de degrés de liberté). L'erreur de la norme L^2 en température à un instant donné, est plus petite en IGA par rapport aux éléments finis, et le taux de convergence se rapproche du taux de convergence optimal. On voit que pour des patchs de faible ordre sur maillage grossier, il peut subsister des oscillations en contrainte qui disparaissent en augmentant l'ordre du patch (raffinement k) ou la finesse du maillage rapidement (raffinement h).

Nous avons enfin appliqué la méthode à un modèle de comportent d'intérêt industriel. Il s'agit d'un modèle thermomécanique pour milieu quasi-incompressible en transformations finies. Une des originalités de ce modèle vient de la décomposition de la partie volumique du gradient des déformations en une partie mécanique et une partie venant de la température. La partie déviatorique est ensuite classiquement séparée en partie élastique et partie inélastique. Ce modèle est associé à des fins expérimentale à un modèle de Zener pour la partie mécanique. Trois tests numériques pour vérifier la physique des modèles sont étudiés. Dans un test de compression statique/cisaillement cyclique en 3D dans lequel on ne tient pas compte de la température (choix des paramètres pour avoir un problème purement mécanique), on vérifie la stabilisation du cycle hystérétique (contrainte de cisaillement fonction du déplacement horizontal) dû à la dissipation. On vérifie qualitativement la solution en contraintes et déplacement qui pour un maillage assez grossier ne montre pas de problèmes d'oscillations. Sur une éprouvette de type haltère en 2D, on vérifie la capacité du modèle à simuler l'auto-échauffement d'un élastomère chargé. Au centre de l'éprouvette (zone de déformations uniformes), on retrouve physiquement l'inversion de température par une précharge, et on vérifie que le matériau s'échauffe sous charge cyclique. Un test équivalent est conduit sur la même éprouvette en 3D. Ces tests valide globalement le travail d'implémentations et de modélisation numérique.

Contents

Introduction

Isogeometric Analysis has probably open a new area in the design of simulation tools as an attempt to reduce the gap between CAD and CAE. From an industrial point of view, it makes sense to keep a same geometric primitive all along the design and production processes. The fundamental concept formalized by T.J.R Hughes & al [7] was to base the discretization of the unknowns in the FEM on basis functions usually used for geometric description. Until now this method has been applied in many contexts in fluid, solid and structural mechanics.

Our aim in this work is to build a tool to simulate complex multiphysics problems for elastomers industrial parts. Roughly speaking, in aeronautics for example, the elastomeric supplies have rather simple shapes because there are in general used as energy absorbers or dampers: cylinders, spheres… It means that for our concern the management of complex geometries is not essential today. In this context, the matter that the gap between full 3D volume representation and CAD surface representation is secondary. The promising gain in accuracy of model is much more attractive for us. We can except to develop light and simple models to simulate long term behavior under thermochemical heavy loading of rubber-like supplies rather easily. We will address to major issues in this work, first, the design of an integrated approach for IGA multiphysics applications in an object-oriented paradigm, and second, the numerical modeling for the simulations of long term time dependent problems under heavy multiphysics loading.

The key issue for simulation is to have a convenient computational tool that enable a fast and accurate solutions. During the recent 50's, the developments of FEM (including material models, problem formulation and resolution algorithms, etc.), numerous reliable and efficient tools where developed in the industry and academic researches. In the 90's object-oriented programming has proven its capability to deal with complexity. It brought modularity to codes and enabled the programmer fast extendibility and maintenance. Today, in an industrial context, the optimal solution to integrate new computational methods remains their availability in existing resources. In a research context, the fast extendibility remains an important issue. In presence of a new numerical method, we think

that its analysis in the context of computer implementation is important and mandatory. We are convinced that the object oriented paradigm remains one of the best analysis and implementation paradigm for structuring large scale software. In this work, we'll study the IGA from the point of view of the computer implementation in an object-oriented context but these ideas could be applied in any other context. The aim is to take benefit of the existing frameworks in the context of IGA. It will be quite natural to take advantage of the common variational framework between FEM and IGA to fasten developments. Besides, with the intension to adopt flexible and stable discretizations for multi-field problems (e.g. mixed formulation method, thermomechanical formulations and the other coupled multiphysics formulations) a prototype of multilayer discretization scheme is put forward and established. With the IGA integrated Java code, we can test effortlessly the application of NURBS based isogeometric Galerkin method. All the models and formulations intended for FEM method are compatible with the novel analysis method. Thus it facilitates the validation and comparison of the numerical results deduced from IGA with reference to that from FEM.

The point of departure of this work is the material elastomer. As matter of fact, the two main numerical issues have to be addressed in this context is the incompressibility/quasi-incompressibility of the material and the thermochemical coupling in Galerkin formulations, including the coupling with constitutive law.

Most rubber-like materials such as elastomers undergo large deformations without significant volumetric changes. They are modeled as incompressible or weakly compressible materials. Under incompressible constraint, volumetric locking issues persist for low order element in FEM and in IGA for standard displacement models. To overcome this difficulty, the treatment of incompressibility and quasi-incompressibility has followed mains tracks for FEM: pure displacement formulations or mixed approaches. Several curing methods been approved successful for FEM have been extended and adapted to IGA. An extension of the $\bar{B}$ and $\bar{F}$ have been proposed by Elguedj et al [46]. The volumetric part of the strain is projected on patch one order lower than the original one. Adams et al [47] have proposed selective and reduced integration approach applied in the context of NURBS interpolations. In the context of mixed approaches, 2- or more fields approaches exist. The

fulfillment of the so-called inf-sup stability condition for mixed variational formulations guaranty the stability of the formulation. For the discrete case, this requirement is mandatory for any choice of discretization for the fields involved in the formulation. The mathematical proof is not always attainable. Numerical strategies to evaluate the inf-sup condition can be developed but depends on the shape of the domain on which the test is done. The last way to check the stability is to check to rate of convergence of the error for a given proven. Once again the stability is not proven but if several well-chosen tests exhibits optimal convergence, it might be sufficient to trust the formulation. We follow this last idea for mixed formulations and target at assessing the numerical performance of different discretizations of two-field mixed formulation. We will compare results to pure displacements methods and/or FEM when possible if it is considered pertinent.

In the context of thermomechanical models, we investigate the choices of discretization for multi-fields formulations for thermomechanical problems. Similar ideas developed in the context of mixed formulations for incompressibility treatment are developed, e.g. aiming at satisfying the *inf-sup* stability condition. Its performance is evaluated through several linear and nonlinear thermoelastic problems. At last, the full complexity of the problem, i.e. an IGA formulation for incompressible or quasi-incompressible thermomechanical problem at large strains will be addressed.

The manuscript is decamped in 5 chapters and 3 appendices. In chapter 1, a deep state of the art is provided for IGA and basic principles are discussed. In chapter 2, all the implementation aspect in an object-oriented paradigm are advocated. In chapter 3, miscellaneous aspects of rubber-like materials are given, and a trial object-oriented model for hyperelastic material complete the software design of the application in a multiphysics context. In chapter 4, the problem of the incompressible or quasi-incompressible constraint for IGA is addressed and in chapter 5, the thermomechanical coupling is similarly discussed. In appendices, the reader will find: A/ the premise of the study for coupling patches, B/ the details of the projection methods for quasi-incompressible medias, and C/ basic statement for mixed formulations for linear incompressible and quasi-incompressible media.

1. Principles of Isogeometric Analysis

1.1. Geometric description

1.1.1. B-spline and NURBS

Today the spline functions are widely used in computer graphics to create surfaces and geometries. Among these various sorts of splines, the spline the most commonly used in industrial CAD software is NURBS (Non-Uniform Rational B-Splines). In the following, we present firstly the basic conception of B-splines, and then describe the principle to build the geometries based on NURBS from the B-splines.

Parameterization

Mathematically, parametrization is a process involving the identification of a complete set of effective coordinates or degrees of freedom of the system, process or model, without regarding their utility in some design. Parametrization of a line, surface or volume, implies identification of a set of coordinates that allows one to uniquely identify any point (on the line, surface, or volume).

One makes a distinction between the parametric space dimension and the physic space one. E.g., for a parametric curve, its parametric space dimension is 1, whereas its physical space dimension could be 2 (plane curve) or 3 (spatial curve). The geometries with parametric dimension equals to 2 are surfaces (plane or spatial). A geometry with a parametric space dimension equal to 3 is volumetric.

Let us consider a trial example of the unit circle (circle of radius 1 centered at origin). The circle's implicit form expression is:

$$x^2 + y^2 = 1 \tag{1-1}$$

It can be rewritten under the parametric form using trigonometric functions:

$$x = \cos\xi\ , y = \sin\xi\ \ and\ 0 \leq \xi \leq 2\pi \tag{1-2}$$

in which ξ denotes the parameter limited in the interval from 0 to 2π. Here, the parametric space is a one-dimensional interval $[0, 2\pi]$. The parametric definition maps every point

coordinated by parameter ξ to a point (x, y) on the physical space (the origin centered unit circle).

The parametric space of a B-spline is determinated by its knot vectors. A knot vector is a sequence of ordered numbers. For example:

$$\Xi = \{\xi_1, \xi_2, \xi_3, \dots, \xi_m\} \; with \; \forall i = 1,2, \dots, m \;\; \xi_i \in \mathbb{R} \quad and \;\; \xi_i \leq \;\; \xi_{i+1} \;\; \forall i \tag{1-3}$$

In this expression, ξ_i is the i^{th} element of the sequence named by knot. And the interval $[\xi_i, \xi_{i+1})$ is called the i^{th} knot span.

A knot designates the coordinate of a one-dimensional parametric space. When the knots are equidistance, the knot vector is said to be uniform, such as in the knot vector $\Xi = \{0,1,2,3,4,5\}$. Otherwise, it is non-uniform, e.g. $H = \{0,2,5,6,7,9\}$. Knot values may be repeated, i.e. $\Xi = \{0,0,0,1,1,2,3,5,5,5\}$.

The multiplicity of a knot value is the number of times the knot is repeated. Knot's multiplicity has important implications for the properties of the basis function which will detailed later. The parametric space is fully determined by its knot vectors.

Basis functions of B-splines

The basis functions of B-splines are recursively defined with the Cox-De Boor's algorithm which is a fast and numerically stable algorithm for evaluating spline curves in B-spline form. It can be regarded as a generalization of the de Casteljau's algorithm for Bezier curves as in following equations:

Considering the following knot vector:

$$\Xi = \{\xi_1, \xi_2, \xi_3, \dots, \xi_m\} \tag{1-4}$$

then, we define the basis functions as following formulas.

- *If p=0*

$$N_{i,0}(\xi) = \begin{cases} 1 & if \; \xi_i < \xi < \xi_{i+1} \\ 0 & else \end{cases} \tag{1-5}$$

- *If p>=1*

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi)$$

where ξ_i indicates the i^{th} knot of knot vector Ξ, p denotes the basis function's degree.

For a specified knot vector, the relation between the number of knots m, the degree of basis function p and the number of basis functions n holds:

$$m = n + p + 1 \tag{1-6}$$

The equality states that, once the knot vector and degree functions have been chosen, the number of basis functions is automatically defined.

Let's consider a trial example based on a simple uniform knot vector which contains 6 knots (m = 6), $\Xi = \{0,1,2,3,4,5\}$. For degree $p = 0$, the number of basis functions equals to five n = 5. For degree $p = 1$, the number of basis functions is n = 4. For degree $p = 2$, there are 3 quadratic basis fucntions n = 3. These basis functions are plotted in Figure 1.

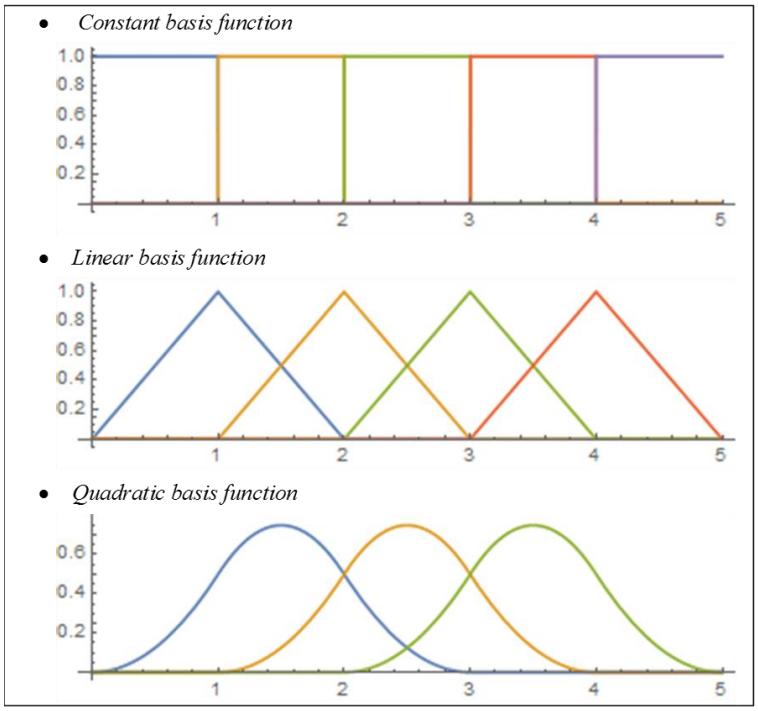


Figure 1. B-Spline functions of degree 0, 1 and 2 for knot vector Ξ = {0,1,2,3,4,5}.

Knot's multiplicity and basis function's continuity

From the definition, it is obvious to remark that B-splines basis function are in fact piecewise polynomials. A p^{th} order basis function has $p-1$ continuous derivatives across the knots. Basis functions of order p have $p-k$ continuous derivatives across knot ξ, where k is the multiplicity of the value of ξ in the knot vector. Thus, when the multiplicity of a knot value is exactly p, the basis is interpolatory at that knot. If the multiplicity is $p+1$, the basis becomes discontinuous and the patch boundary is formed. A knot vector is said to be open if the multiplicity of its first and last knot values equals to $p+1$. Open knot vectors are the standard in the CAD literature. In dimension 1, basis functions built from open knot vectors are interpolatory at both ends of the parameter space interval, $[\xi_1, \xi_m]$, and at the corners of patches in multiple dimensions, but they are not, in general, interpolatory at interior knots.

Consider the open knot vector given as $\Xi = \{0,0,0,1,1,3,3,3\}$, the quadratic basis functions ($p = 2$) are drawn in Figure 2. The knot vector has 8 knots (m=8). As matter of fact, the number of basis functions is n $= 5$. The head and tail knots have both a multiplicity order of $p+1$, the knot vector is open. The knot value $\xi = 1$ is repeated twice ($k = 2$), thus the basis functions are C^0 at knot $\xi = 1$.

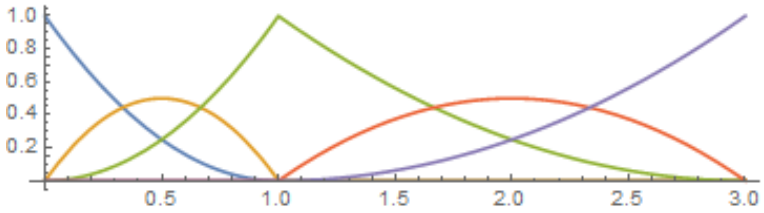


Figure 2. An example of B-spline basis functions

Geometric modeling based on B-splines

In order to construct a geometry based on B-splines, we need some indicated points in physical space which are referred as control points. B-spline curves in $\mathbb{R}^d$ are constructed by considering a linear combination of B-spline basis functions, such as the construction based on Bezier. A piecewise linear interpolation of the control points defines the control polygon. Given a knot vector with n basis functions defined such that $N_{i,p}(\xi), i =$

$1,2,\dots,n$, and corresponding control points in the physical space, $B_i \in \mathbb{R}^d, i = 1,2,\dots,n$. The one-dimensional B-spline curve is given by:

$$C(\xi) = \sum_{i=1}^{n} N_{i,p} \cdot B_i \tag{1-7}$$

Control points and basis functions are obviously in a one-to-one correspondence relationship. n should be the number of basis functions likewise number of control points.

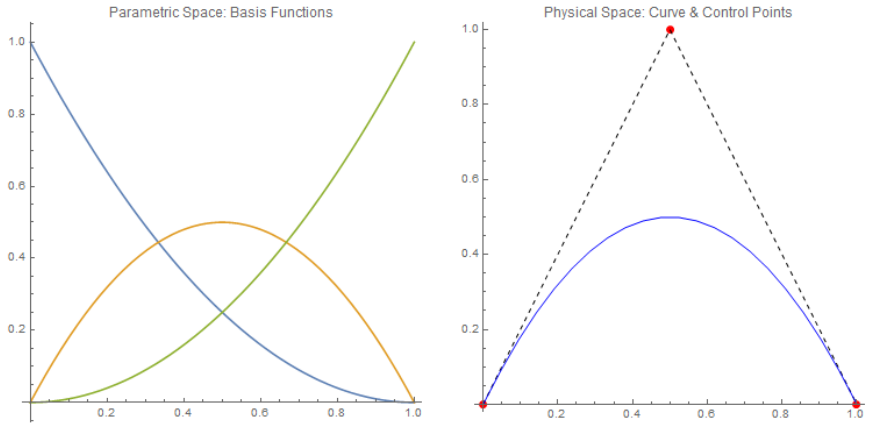


Figure 3. B-spline based parabola.

The surfaces and volumes on B-splines are respectively built by tensor product of 2 or 3 B-splines curves. Mathematically, the tensor product is realized by a rewriting of the basis functions. Consider the 2D example, the knot vectors:

$$\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\} \; and \; H = \{\eta_1, \eta_2, \dots, \eta_{m+q+1}\}$$

The parametric 2D basis function of B-splines are rewritten as:

$$N_{i,p}(\xi) \cdot M_{j,q}(\eta)$$

where $N_{i,p}(\xi)$ and $M_{j,q}(\eta)$ are the univariate B-spline basis function of order p and q corresponding to knot vectors Ξ and H, respectively.

As matter of fact, the tensor-product B-splines surface is formulated as:

$$S(\xi,\eta) = \sum_{i=1}^{n}\sum_{j=1}^{m} N_{i,p}(\xi) \cdot M_{j,q}(\eta) B_{i,j} \tag{1-8}$$

For simplicity, the above relationship can be rewritten in a more compact form:

$$S(\xi,\eta) = \sum_{i=1}^{n}\sum_{j=1}^{m} N_{i,j}^{p,q}(\xi,\eta)B_{i,j} \tag{1-9}$$

where $N_{i,j}^{p,q}(\xi,\eta) = N_{i,p}(\xi)\cdot M_{j,q}(\eta)$ is the bivariate B-spline basis function. The parametric domain is a rectangle $(\xi,\eta) \in [\xi_1,\xi_{n+p+1}]\times[\eta_1,\eta_{m+q+1}]$. In Figure 4, an example of quadratic B-spline in $\mathbb{R}^3$ surface is presented.

From B-splines to NURBS

B-splines are powerful for free-form modeling, but they lack the ability to exactly represent the conics like circles, cylinders which are usual in mechanical engineering. An extension of B-splines which is no more based on polynomials are necessary: Non-Uniform Rational B-spline (NURBS). NURBS are commonly used in computer-aided design (CAD), manufacturing (CAM), and engineering (CAE) and are part of numerous industry wide used standards, such as IGES, STEP, ACIS, and PHIGS.

Like B-splines, NURBS need a degree, and a knot vector, and a set of control points to be fully defined. The difference from simple B-splines is that every control points of NURBS is associated to a weight. NURBS can be regarded as a generalization of B-splines like what B-spline was a generalization of Bézier functions. The difference is the weighting of the control points which makes NURBS curves "rational". When all the weights equal to 1, NURBS is simply B-splines.

A rational basis function of NURBS is expressed with respect to B-splines function such as:

$$R_i^p(\xi) = \frac{N_{i,p}(\xi)\omega_i}{W(\xi)} = \frac{N_{i,p}(\xi)\omega_i}{\sum_{j=1}^{n} N_{j,p}(\xi)\omega_j} \tag{1-10}$$

where ω_i denotes the weight of control point B_i. Obviously, this rational basis will degenerate to B-spline basis function when $\omega_i = 1$ for every control point. The expression of a geometry on NURBS has the same form as B-spline. It is a linear combination of the basis functions R_i^p including a control point $B_i \in \mathbb{R}^d$. The expression of a NURBS curve is given just as equation (1-7) by:

$$C(\xi) = \sum_{i=1}^{n} R_{i,p}(\xi) \cdot B_i \tag{1-11}$$

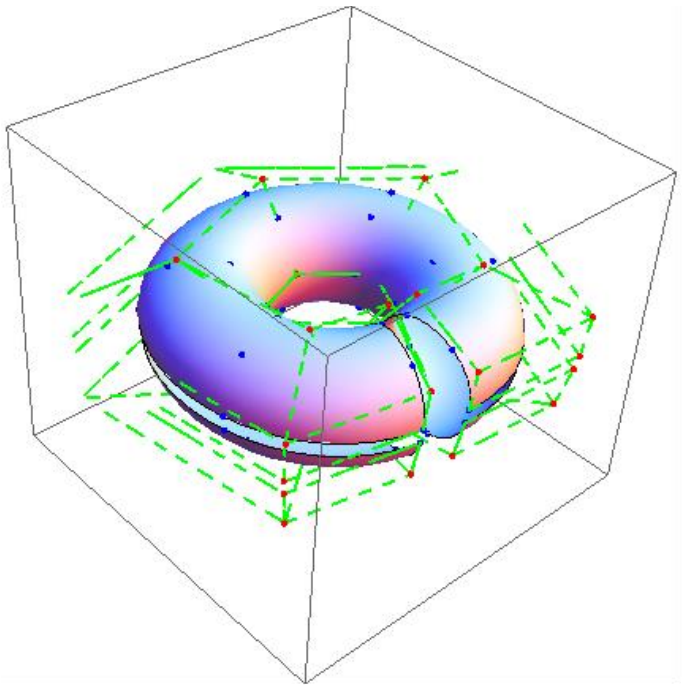


Figure 4. B-spline based spatial surface

NURBS based surfaces or volumes are constructed in the same way as surfaces or volumes on B-splines, using the tensor-product of rational basis function $R_{i,p}(\xi)$ to obtain the two or three parametric dimensional basis functions for surfaces $R_{i,j}^{p,q}(\xi,\eta) = R_{i,p}(\xi) \cdot R_{j,q}(\eta)$ or volumes $R_{i,j,k}^{p,q,r}(\xi,\eta,\zeta) = R_{i,p}(\xi) \cdot R_{j,q}(\eta) \cdot R_{k,r}(\zeta)$.

In addition to the mathematical explanation. A NURBS geometry modeling in $\mathbb{R}^{n_{sd}}$ can be obtained by the projection transformation of a B-spline geometry modeling in $\mathbb{R}^{n_{sd}+1}$. With a given projective B-spline curve and its associated projective control points in hand, the control points for the NURBS curve are obtained thorough equation:

$$\omega_i = ({B_i}^{\omega})_{d+1}, \;\; (B_i)_j = \frac{{B_i}^{\omega}}{\omega_i} \; for \; j = 1,2,\dots,d \tag{1-12}$$

In this expression, ${B_i}^{\omega}$ indicates a control point of the original B-spline. And the B_i represents a weighted control point for the projected B-spline that is the NURBS. $(B_i)_j$ is the j^{th} component of its coordinate.

Let us consider the constructing a plane circle based on NURBS functions. The plane circle is in $\mathbb{R}^2$, hence we need a spatial curve based on B-spline in $\mathbb{R}^3$ (see Figure 5). The spatial B-spline is represented by the black curve in the Figure 5 with its control points indicated by the blue lozenges. A radial projection on the plane $z = 1$ leads to define the red circle, and the projected control points denotes by these green lozenges. By looking through the viewport right above, we can verify that the circle is precisely constructed basing on NURBS.

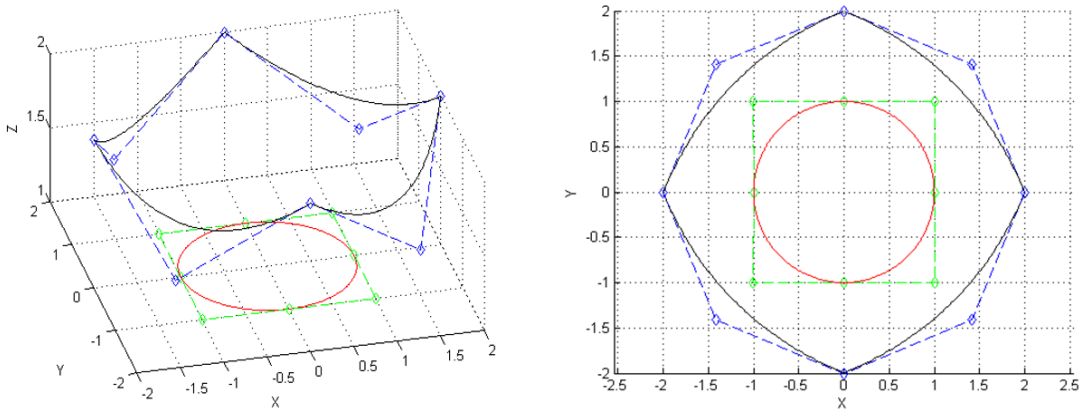


Figure 5. Projection from a B-spline based spatial curve to a NURBS based perfect circle

1.1.2. Knot insertion and Degree elevation algorithms

In this section, the fundamental algorithms for B-spline and NURBS are presented keeping in mind the perspective of implementation. In the first part, we consider some implementation aspects of the evaluation of the basis functions and their derivatives. Then, we focus on some B-splines and NURBS basic requirements for the so-called h, p and k refinement algorithms: knot insertion and degree elevation. Both algorithms play a crucial role to manage basis function's inter-span continuity order. Note for that, NURBS functions, some advanced algorithms such as point inversion or, parametrization redefinition may be necessary for a full management. As these advanced methods go beyond the scope of this work, wo de not present them (see [63] for a thorough treatment).

Evaluation of basis functions and their derivatives

The definition of B-splines basis function reveals its locality property: for a parameter value ξ in the i^{th} knot span $[\xi_i, \xi_{i+1})$, only $(p+1)$ basis functions do not vanish, i.e. functions $\{N_{i-p,p}, N_{i-p+1,p} \dots, N_{i,p}\}$. Thus, the value of B-splines at parameter value ξ is determined through the control points related to these nonzero basis functions.

The De Boor's algorithm is an efficient algorithm to evaluate the B-spline function $C(\xi)$. Consider a parameter value $\xi \in [\xi_i, \xi_{i+1})$, and the control points $P_j^{[0]} = P_j$ for $j = i - p, \dots, i$. Consider the iterative evaluation:

$$P_j^{[k]}(\xi) = \left(1 - \alpha_j^{[k]}(\xi)\right) P_{j-1}^{[k-1]} + \alpha_j^{[k]} P_j^{[k-1]} \tag{1-13}$$

where $k = 1, \dots, p$ and $j = i - p + k, \dots, i$, where $\alpha_j^{[k]}$ is a function of ξ written as in equation:

$$\alpha_j^{[k]}(\xi) = \frac{\xi - \xi_j}{\xi_{j+p+k} - \xi_j} \tag{1-14}$$

The procedure of De Boor's algorithm could be visualized by a triangular scheme in Figure 6. E.g. to compute the term $P_2^{[3]}$ one needs to compute $P_3^{[2]}$ and $P_2^{[2]}$, and so one until the level *0*, the level of the control points coordinates at parameter value $\xi \in [\xi_i, \xi_{i+1})$.

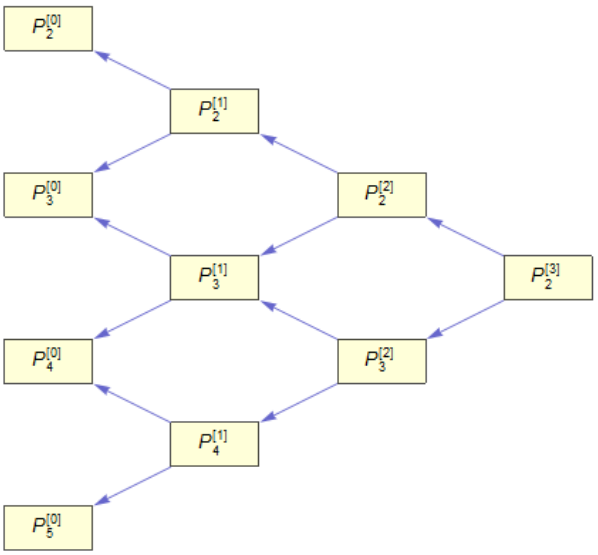


Figure 6. De Boor's Algorithm

In Figure 7, we show an example of a cubic B-spline curve evaluated at parameter value $\xi \in [\xi_i, \xi_{i+1})$ by De Boor's algorithm. The geometric point corresponding to parameter value ξ is indicated by the $P_{2,3}$ in red.

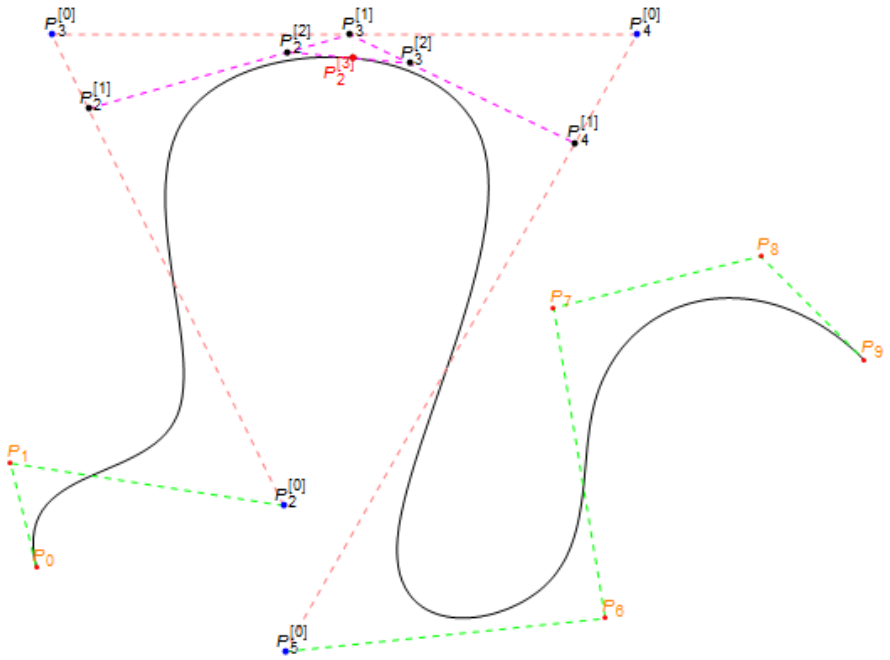


Figure 7. A cubic B-spline curve's evaluation by De Boor's algorithm

Considering definitions in equation (1-7) and the fact that $\xi \in [\xi_i, \xi_{i+1})$, we can limit the calculation on the basis functions $\{N_{i-p,p}, N_{i-p+1,p} \dots, N_{i,p}\}$, as the other functions vanish all the time in considered span. We also need to compute derivatives of basis functions.

Use $N_{i,p}^{(k)}$ to denote the k^{th} derivation of basis function $N_{i,p}$. Since $N_{i,p}$ is basically a polynomial function, its derivative $N_{i,p}^{(k)}$ will be zero for every $k > p$. For $k \leq p$ the general formula for the derivation of $N_{i,p}$, $N_{i,p}^{(k)}$ depends on terms $\{N_{i,p-k}, N_{i+1,p-k} \dots, N_{i+k,p-k}\}$ is given in equations:

$$N_{i,p}^{(k)} = \frac{p!}{(p-k)!} \sum_{j=0}^{k} a_{k,j}\, N_{i+j,p-k} \tag{1-15}$$

where

$$\begin{cases} a_{0,0} = 1 \\ a_{k,0} = \dfrac{a_{k-1,0}}{\xi_{i+p-k+1} - \xi_i} \\ a_{k,j} = \dfrac{a_{k-1,j} - a_{k-1,j-1}}{\xi_{i+p+j-k+1} - \xi_{i+j}} \\ a_{k,k} = \dfrac{-a_{k-1,k-1}}{\xi_{i+p+1} - \xi_{i+k}} \end{cases} \quad (1\text{-}16)$$

where $j = 1, \dots k-1$. Attention should be paid on the computation of the denominator. A fraction is defined to be zero when its denominator vanishes.

Basic refinement algorithms: knot insertion

An important advantage of NURBS based geometric description comes from its refinement algorithms: knot insertion and degree elevation. They offer advantage to obtain a finer grid without altering the underlying geometry or the parameterization. Besides, the combination of those two algorithms permits to control the continuity order of functions (for a thorough discussion, see [63]).

To reduce the size of the knot span and to improve the mesh resolution, one can perform the so-called knot insertion method. The result in a new spline space with an enriched basis.

Consider a NURBS based curve $C(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) P_i^w$, for which the NURBS are defined on the knot vector $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$. The new knot to be inserted $\bar{\xi}$ is located in the span $[\xi_k, \xi_{k+1})$. We can give the new knot vector:

$$\bar{\Xi} = \{\bar{\xi}_1 = \xi_1, \dots, \bar{\xi}_k = \xi_k, \bar{\xi}_{k+1} = \bar{\xi}, \bar{\xi}_{k+2} = \xi_{k+1}, \dots, \bar{\xi}_{n+p+1} = \xi_{n+p+1}\}$$

The number of control points is now $n+1$. The coordinates for these $n+1$ new control points $\{Q_i^w, i = 1, \dots, n+1\}$ can be computed through the relation:

$$C(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) \cdot P_i^w = \sum_{i=1}^{n+1} \bar{N}_{i,p}(\xi) \cdot Q_i^w \quad (1\text{-}17)$$

The general formula is deduced from the equation:

$$Q_i^w = \alpha_i P_i^w + (1 - \alpha_i) P_{i-1}^w \quad (1\text{-}18)$$

with

$$\alpha_i = \begin{cases} 1 & i \le k-p \\ \dfrac{\bar{\xi} - \xi_i}{\xi_{i+p} - \xi_i} & k-p+1 \le i \le k \\ 0 & i \ge k+1 \end{cases} \quad (1\text{-}19)$$

We can note that the new spline space contains the original spline space as a subspace. Thus any spline curve in the original space can also be represented in the refined space. The knot insertions algorithm can be used in many situations such as: partitioning the parametric geometry, evaluating points and derivatives on curves and surfaces, improving the flexibility by adding supplementary control points. Consider the simplest case for a quadratic NURBS curve constructed on knot vector $\Xi = \{0, 0, 0, 1, 1, 1\}$ with $\mathrm{n} = 3$ control points. When a new knot valued $\xi = 0.4$ has been inserted, the knot vector becomes $\Xi = \{0, 0, 0, 0.4, 1, 1, 1\}$. In the meantime, a novel control point emerges $\mathrm{n} = 4$ as in Figure 8.

The relation between the continuity order and the multiplicity of a knot value ($\mathrm{c} = p - k$) allows us to adjust the continuity order of basis function at a given knot value. It can be done by inserting a repeated knot value. Increasing the order of multiplicity of the knot value, causes the decrease of the continuity order at this knot value.

Basic refinement algorithms: knot removal

The inverse process of knot insertion is called knot removal. This algorithm is aimed at removing an inner knot from the knot vector without altering the geometry. The usual motivation is to accurately construct a B-spline with minimum data (minimum control points and knots). Note that removing a repeated knot value allows us to augment the continuity order of function at the knot value. However, it is not possible to remove whatever knots when the geometry and parametrization is unchanged. Thus it is necessary to check if a knot value can be removed or not.

Suppose a NURBS based curve of degree p, $C(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) \cdot P_i^w$, defined on the knot vector $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$. The knot to be removed is $\xi = \xi_r \neq \xi_{r+1}$ and its order of multiplicity is m where $1 \le m \le p$. We know that the basis functions $N_{i,p}(\xi)$ that do not vanish at the knot ξ_r are C^{p-m}. The order of continuity of curve $C(\xi)$ depends on the positions of the control points and can be different from the one of its basis functions which

are supposed to be C^{p-m+t}. Pose t the disparity, the criterion to determine if the knot value ξ can be removed or not. If $t > 0$, the knot is t times removable. The relation is expressed as follows, where the superscription indicates processing the number of times of the knot is removed.

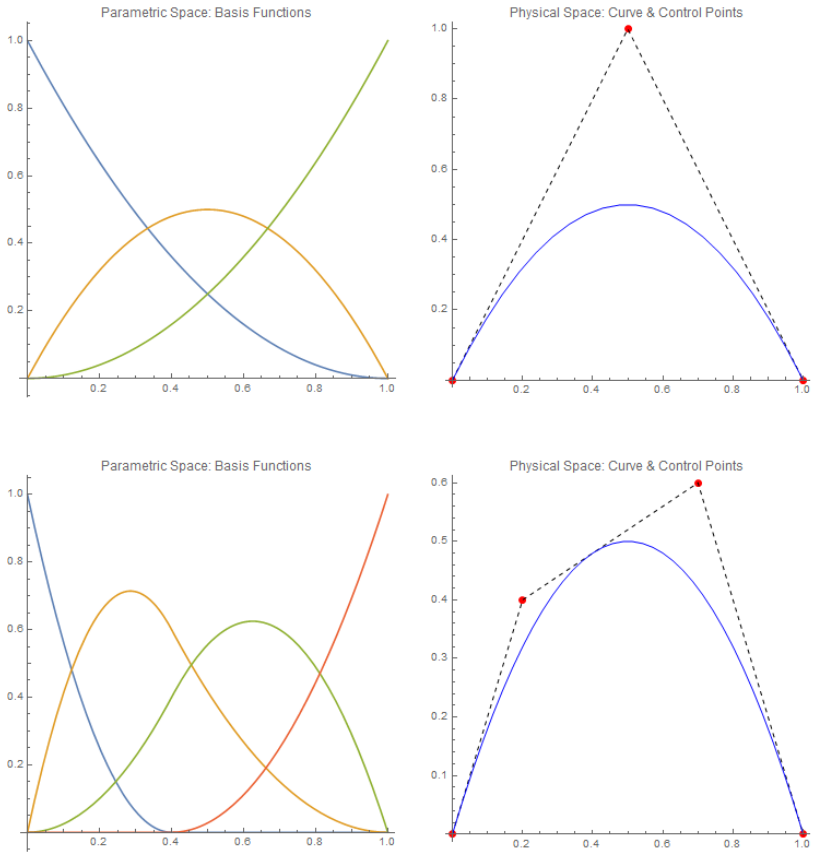


Figure 8. Knot insertion performed on a quadratic NURBS curve

The general formulas to remove t times the knot ξ when assuming it is at least t time removable are:

$$C(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) \cdot P_i^0 = \sum_{i=1}^{n-t} \overline{N}_{i,p}(\xi) \cdot P_i^t \tag{1-20}$$

The equations for computing the new control points are:

$$\begin{cases} P_i^t = \dfrac{P_i^{t-1} - (1-\alpha_i)P_{i-1}^t}{\alpha_i} & r-p-t+1 \le i \le \dfrac{1}{2}(2r-p-s-t) \\ P_{j-1}^t = \dfrac{P_j^{t-1} - \alpha_j P_j^t}{(1-\alpha_j)} & \dfrac{1}{2}(2r-p-s+t+1) \le j \le r-s+t-1 \end{cases} \quad (1\text{-}21)$$

where

$$\alpha_i = \frac{\xi - \xi_i}{\xi_{i+p+t} - \xi_i} \; and \; \alpha_j = \frac{\xi - \xi_{j-t+1}}{\xi_{j+p+1} - \xi_{j-t+1}} \quad (1\text{-}22)$$

An illustration of the knot removal algorithm is given in Figure 9. The initial geometry is a cubic NURBS based curve$(p = 3)$defined by control points $\{P_1^0, \dots, P_7^0\}$ and the knot vector $\Xi = \{0, 0, 0, 0, 0.5, 0.5, 0.5, 1, 1, 1, 1\}$. The superscript on the control point indicates the step number in the knot removal process. Suppose the knot $\xi = 0.5$ is to be removed 3 times from the initial knot vector. By means of the formulas, all new control points are computed step by step. The control points and the control mesh are plotted in red in Figure 9. The dashed lines in green indicates the last control mesh, the initial control mesh is in blue.

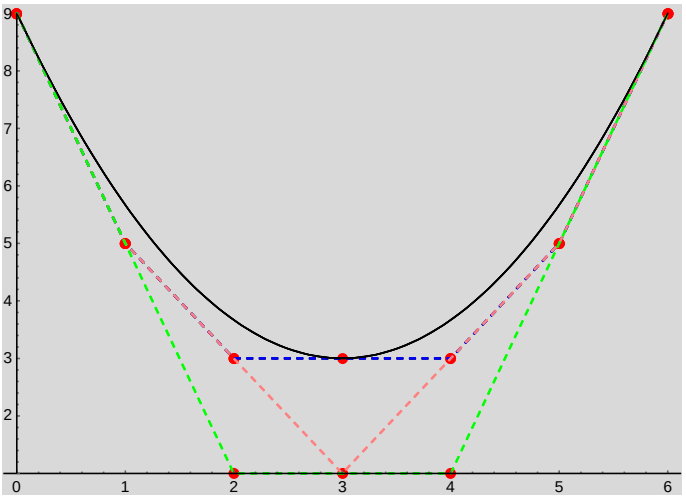


Figure 9. Knot Removal from a cubic curve with a triple knot

Basic refinement algorithms: degree elevation

The augmentation of the degree of functions follows a similar strategy to enrich the representation of NURBS curves. The continuity order of basis functions is maintained even across the knot spans. The algorithm which consist in raising up the degree of function of a NURBS geometry can be decomposed in three steps:

- Decompose the knots span into Bézier segments: all the interior knots are duplicated until their multiplicity order equals to the polynomial degree p.
- Increase the degree of functions of each Bézier segments.
- Remove unnecessary knots to recover the initial basis function's continuity order across knot spans.

The first step of the process can be accomplished by repeating the knot insertion algorithm described above. The last step of degree elevation is simply obtained by applying the knot removal algorithm. Note that it is necessary to store the number of knots that have been inserted during Step 1 in order to recover the exact continuity order of basis functions during the Step 3.

Therefore, the degree elevation can be completed by the degree elevations algorithm for Bézier segment for the second step. The general formulas to elevate a Bézier curve from degree p to degree $p+1$ is given as follows. Consider a Bézier curve of degree p, $C_p(\xi) = \sum_{i=1}^{p} B_{i,p}(\xi)P_i$. Its representation as a Bézier curve of degree $(p+1)$ is given by $C_{p+1}(\xi) = \sum_{i=1}^{p+1} B_{i,p+1}(\xi)Q_i$. Consider the more general case to raise degree from p to $p+t$ in one step. The expression for the new curve is $C_{p+t}(\xi) = \sum_{i=1}^{p+t} B_{i,p+t}(\xi) \cdot P_i^t$. At last, the formula for the general case is given as in equation:

$$P_i^t = \sum_{j=max(0,i-t)}^{min(p,i)} \frac{\binom{p}{j}\binom{t}{i-j}P_j}{\binom{p+t}{i}} \; for\; i = 0, \dots, p+t \qquad (1\text{-}23)$$

At this stage, all the basic tools necessary for proceeding the degree elevation on NURBS have been described. In the following, consider the example of B-spline curve's degree being elevated from $p = 2$ to $p = 4$ as shown in Figure 10. During the degree elevation procedure, continuity order of all the basis function stays unchanged. Furthermore, just as those two-precedent refinement algorithms, the geometry remains exact and unchanged.

1.2. NURBS as a basis for analysis

In the previous section, the principles of NURBS as a geometric construction tool have been introduced. The flexibility and precision of NURBS make them ubiquitous in computer aided design. The key point of isogeometric analysis is to employ NURBS as

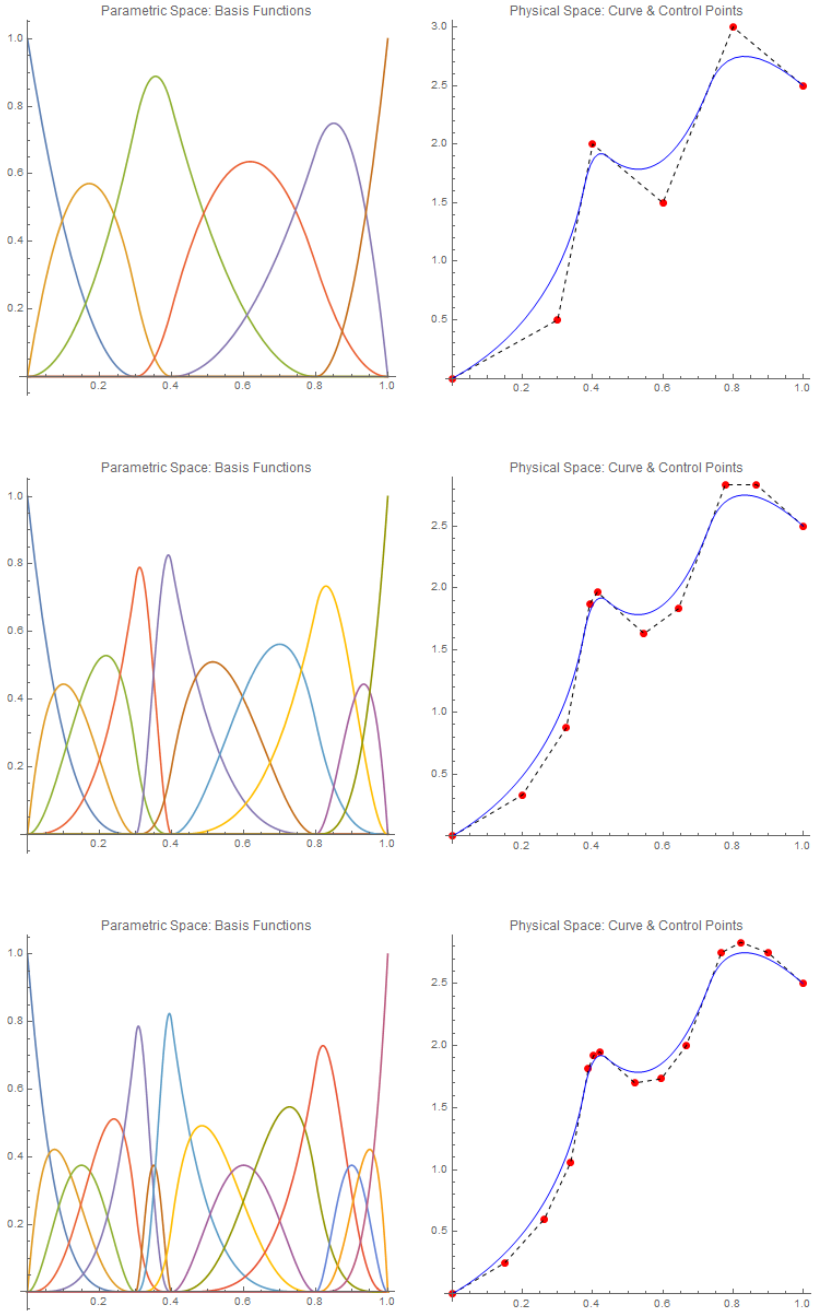


Figure 10. Degree elevation for a B-spline curve.

basis of analysis. Due to their high order continuity and exact geometry description, NURBS functions are good candidates aiming at improving the analysis performance over traditional piecewise polynomial functions.

1.2.1. A trial boundary values problem -Weak form - Discrete form

Many continuum problems arising in engineering and physics are usually described by appropriate differential equations and boundary conditions. For the sake of simplicity, let's consider a steady-state heat conduction equation. A formal statement of this boundary value problem is as follows:

Given $f: \Omega \to \mathbb{R}$, $g: \Gamma_D \to \mathbb{R}$ and $h: \Gamma_N \to \mathbb{R}$, find $u: \Omega \to \mathbb{R}$ temperature such that:

$$\begin{aligned} \nabla \cdot \vec{q} + f &= 0 \ \ in\ \Omega \\ u &= g \ \ on\ \Gamma_D \\ \vec{q} \cdot \vec{n} &= h\ on\ \Gamma_N \end{aligned} \tag{1-24}$$

where Ω denotes the domain and its boundary $\partial\Omega$ which is decomposed into two parts with $\Gamma_D \cup \Gamma_N = \partial\Omega$ and $\Gamma_D \cap \Gamma_N = \emptyset$. $\vec{n}$ is the unit outward normal vector on $\partial\Omega$.

The unknown function is the temperature $u: \Omega \to \mathbb{R}$ and $\vec{q}: \Omega \to \mathbb{R}^{n_{sd}}$ represents the heat flux vector. The domain Ω is defined by a NURBS geometry. $n_{sd} = 2\ or\ 3$ is the space dimension.

$$\vec{q} = k\nabla u \tag{1-25}$$

where the conductivity k ($(n_{sd})^{th}$ order tensor) is constant for a homogeneous body and for a linear problem.

In the following, we focus on the finite element method as general framework to describe the isogeometric analysis.

The weak formulation of strong form (1-24) of the boundary-value problem to be solved is built by multiplying the heat equation by an arbitrary test function $w \in V$. The result is integrated over the domain and integrated by parts. The weak form of the steady-state heat

conduction problem introduced in the previous section is obtained by taking into consideration the boundary conditions and the Fourier's law and goes as follows:

Given $f: \Omega \to \mathbb{R}$, $g: \Gamma_D \to \mathbb{R}$ and $h: \Gamma_N \to \mathbb{R}$, find $u \in S$ such that for all $w \in V$

$$\int_\Omega \nabla w \cdot \nabla u \, d\Omega = \int_\Omega w \, f \, d\Omega + \int_{\Gamma_N} w \, h \, d\Gamma \tag{1-26}$$

where $S = \{u | u \in H^1(\Omega), u|_{\Gamma_D} = g\}$ and $V = \{w | w \in H^1(\Omega), w|_{\Gamma_D} = 0\}$ are respectively the trial solution space and variation space and $H^1(\Omega)$ denotes a Sobolev space on the domain Ω (remark: $H^1(\Omega) = \left\{u \,/\, \int_\Omega \nabla w \cdot \nabla u \, d\Omega < \infty\right\}$).

It is more convenient to write the weak form under the abstract form:

$$\begin{gathered} a(w,u) = l(w) \\ a(w,u) = \int_\Omega \nabla w \cdot \nabla u \, d\Omega \\ l(w) = \int_\Omega w \, f \, d\Omega + \int_{\Gamma_N} w \, h \, d\Gamma \end{gathered} \tag{1-27}$$

where $a(w,u)$ is symmetric and bilinear and $l(w)$ is linear.

The first step of the Galerkin method is to construct finite dimensional approximation spaces for trial solution $S = \{u | u \in H^1(\Omega), u|_{\Gamma_D} = g\}$ and variation $V = \{w | w \in H^1(\Omega), w|_{\Gamma_D} = 0\}$. These discretization approximations are respectively noted S^h and V^h. They are subsets of original spaces $S^h \subset S$ and $V^h \subset V$.

The discretized variation function is chosen within the approximation space V^h as $w^h \in V^h$. The discretized trial function is decomposed using the Dirichlet boundary condition such as $u^h = v^h + g^h$, where $v^h \in V^h$ and $g^h \in S^h$ with $g^h|_{\Gamma_D} = g$. The Galerkin formulation for the steady-state heat conduction problem can be expressed as:

Given g^h, h and r, find $u^h = v^h + g^h$, where $v^h \in V^h$, such that for all $w^h \in V^h$

$$a(w^h, u^h) = l(w^h) \tag{1-28}$$

Considering the splitting based of the Dirichlet Substituting u^h using the relation $u^h = v^h + g^h$ in the previous equation, we obtain:

$$a(w^h, v^h) = L(w^h) - a(w^h, g^h) \tag{1-29}$$

Consider that the solution and trial solution spaces consist of linear combinations of n NURBS functions defined on the domain Ω. For the solution $u \in S^h$, there exist $d_A, A = 1,2, \dots, n$ such that

$$u^h = \sum_{A=1}^{n} N_A \, d_A + g^h \tag{1-30}$$

In similar way, for the trial solution $w^h \in V^h$, there exist constants $c_A, A = 1,2, \dots, n$, such that

$$\mathbf{w^h} = \sum_{\mathbf{A=1}}^{\mathbf{n}} \mathbf{N_A} \, \mathbf{c_A} \tag{1-31}$$

Inserting these relations into the approximated bilinear form (1-27), and taking advantage of linearity, we obtain the expression

$$\sum_{A=1}^{n} (\sum_{B=1}^{n} a(N_A, N_B) d_B - l(N_A) + a(N_A, g^h)) = 0 \tag{1-32}$$

It must be satisfied for all $w^h \in V^h$, so the coefficients c_A are arbitrary. Thus for $A = 1,2, \dots, n$, one has:

$$\begin{aligned} \boldsymbol{K_{AB}} &= \boldsymbol{a(N_A, N_B)} \\ \boldsymbol{F_A} &= \boldsymbol{l(N_A) - a(N_A, g^h)} \end{aligned} \tag{1-33}$$

The problem can be characterized by the matrix equation

$$\mathrm{Kd} = \mathrm{F} \tag{1-34}$$

Due to historical origins, **K** is usually named as stiffness matrix, **d** and **F** are usually called displacement vector and force vector. These terminologies are applied independently of the actual problem being solved.

Since the NURBS basis functions N_A are generally highly localized, the stiffness matrix **K** is a sparse matrix. Therefore, instead of looping through all the global shape functions, taking global integrals to build **K** one entry at a time, we will loop through the elements, building element stiffness matrices as we go.

1.2.2. Isoparametric discretization

The isoparametric concept implies the use of same basis function for both the geometry and the unknown field discretization. The so called isoparametric element is quite commonly used in finite element analysis. Nevertheless, they differ from the chosen entry point. In traditional finite element method, the basis function used to interpolate the unknown field is applied directly as approximate basis for geometry. Contrariwise, isogeometric analysis consist in employing the geometric basis functions such as NURBS to approximate unknown field function. That is reason we say the isogeometric analysis can preserve the exact geometry stems from computer aided design.

Isoparametric Discretization		
FEM:	Geometry ← Analysis	Lagrange Polynomials
IGA:	Geometry → Analysis	NURBS

Figure 11. Isoparametric element: FEM vs IGA

The polynomials gain this favor in traditional finite element method is mainly due to their simplicity, they are easy to program and easy to prove theorems. Polynomials that satisfy the basic following conditions are convergent (see Hughes [64]):

- C^1 in the element interior,
- C^0 on the element boundaries,
- Complete.

For NURBS basis functions, the first two condition are satisfied straightforwardly. Set of functions that satisfy the partition of unity fulfill the completeness condition. Obviously, the basis functions of NURBS verify that:

$$\forall \xi, p \quad \sum_{i=1}^{n} N_{i,p}(\xi) = 1 \tag{1-35}$$

We illustrate the approximating unknown function on a NURBS based domain in (1-35), where the physic domain is denoted by Ω and the parametric domain by $\widehat{\Omega}$. We see that the geometrical mapping from parametric space to physical space is given as equation by:

$$C(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) \cdot P_i \tag{1-36}$$

In an isoparametric formulation, the unknown field variable u is approximated by the same basis functions as geometry which gives the relation:

$$u(C(\xi)) = \sum_{i=1}^{n} N_{i,p}(\xi) \cdot u_i \tag{1-37}$$

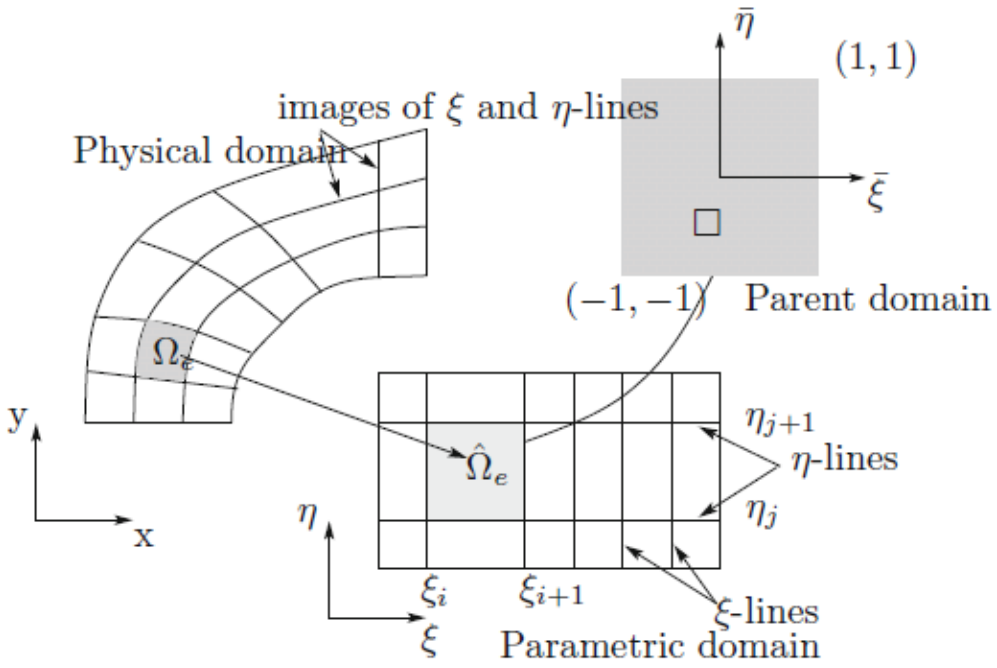


Figure 12. Definition of domains in isogeometric analysis (from Nguyen [65])

where the coefficient u_i is the control variables associated to control point P_i. However, unlike to nodes for finite element method, the control variable u_i does not represent a physical nodal value, since control points of NURBS is generally not interpolant.

1.2.3. Mesh refinement strategies

Mesh refinement indicates the strategies we use frequently in classical finite element method to reduce the errors once an approximate solution has been obtained. Numerous procedures exist for the mesh refinement of FEM solution. Roughly speaking, these falls into two categories h-refinement and p-refinement. Through h-refinement, the same class of elements continue to be used but are decreased in finesse. And in p-refinement, the element size stays unchanged but the order of the polynomials used as basis of geometry and analysis is increased.

In NURBS based isogeometric analysis, these two types of refinement strategy can be easily reproduced by combination of the algorithms knot insertion and degree elevation presented previously. Furthermore, the new refinement procedure: k-refinement which allows us to adjust the inter-element continuity order is introduced.

k-refinement

Knot insertion method clearly has similarities with the classical h-refinement in finite element analysis as it subdivides existing elements into new ones. Despite this, the number of new basis functions created is different, so as their continuity order across recently created span borders, which is C^{p-1} in this case. To accurately reproduce the finite element method's h-refinement, one should insert each new knot values p times. This will give newly created basis function with C^0 continuity across all the knot span borders just as what we always have in finite element.

Degree elevation has much in common with the classical p-refinement strategy in finite element method as both increase the degree of the basis functions. What differs is that the basis functions of finite element analysis remain C^0 at the element boundaries, while NURBS based isogeometric analysis's degree elevation is compatible with arbitrary continuity order basis functions in the unrefined parametric space. Nevertheless, if we begin with a NURBS geometry in which all the functions are already C^0 across knot span borders, order elevation coincides exactly with the traditional notion of finite element method's p-refinement strategy.

When both the knot insertion and degree elevation are employed to perform refinement on a NURBS geometry, the order of combination changes the resulting description since these two processes do not commute. Consider a p^{th} degree spline curve as the non-refined geometry. Insertion of a unique knot value ξ and elevation of degree to $q(q > p)$ would be performed on it in different order:

- Knot insertion first then degree elevation will result in a curve whose basis functions are C^{p-1} at knot ξ.
- Degree elevation first then knot insertion give a curve in which the basis functions have C^{q-1} continuity order at knot ξ.

Since $q > p$, the second scheme leads to basis functions with higher order continuity. This scheme is referred as k-refinement and there is no analogous practice in finite element analysis.

The h-p-k refinement space

As discussed in the previous section, for NURBS based isogeometric analysis, the principal refinement strategies are realized thanks to these primitive algorithms of NURBS: knot insertion and degree elevation. Compared with the FEM refinement procedure, NURBS based IGA provides greater flexibility to reduce the approximating errors. Furthermore, with the possibility of adjusting inter-element continuity order, we introduce a notion of an h-p-k refinement space.

Recalling that B-spline curves may have no more than $(p - 1)$ continuous derivatives across an element boundary, the set of possible refinements may be characterized. Pure k-refinement keeps h fixed but increases the continuity along with the polynomial order. Pure p-refinement increases the polynomial order while the basis remains. Increasing the multiplicity of existing knot values decreases the continuity without introducing new elements. Inserting new knot values with a multiplicity of p results in classical h-refinement, whereby new elements are introduced that have C^0 boundaries. Inserting new knot values with a multiplicity order of 1 decreases h without decreasing the minimum continuity already found in the mesh. Considering all the techniques results in a multitude of refinement options beyond simple h-, p- and k-refinement.

1.3. An historical point of view for Isogeometric Analysis

Before the publication of the founding paper of isogeometric analysis by T.J.R Hughes & al [7] in 2005, a few developments sharing similar perspectives has emerged in the domain of Computer Aided Engineering as well as in the Computer Aided Design domain. There we can see some noteworthy efforts to take benefit from the high order continuity properties of reduce the gap between design and analysis in mechanical engineering.

In the field of CAE, Schultz & al. [1] has proposed to use the cubic spline functions for Finite Element analysis basis functions for different kind of linear problems, including linear elasticity, providing a higher order continuity compared to classical Hermite interpolation (continuity C^2 instead of C^1). As stated in the paper, the spline interpolation needs half as many degrees of freedom as the piecewise cubic Hermite interpolation for the same geometric description. In the context of fluid mechanics, spline type functions have also been used to solve some problems involving different spatial scales. In Kwok [66], the simulation of turbulent flows has been investigated. In the latter, the Galerkin Method based on spline approximations enables the authors to improve the resolution thanks to spline's inter-element higher order continuity. Furthermore, spline functions have also been used in the collocation method. In contrast to compact finite difference method with the same bandwidth, the spline collocation method trades away a potentially higher convergence rate for a straightforward and robust formulation. In the book Höllig [2], the B-spline functions are used to construct a weighted and extended B-splines like functions as solution space basis. However, the geometry description is constructed by a mapping different from the one of the approximation space. This does not respect the isoparametric concept which could guarantee the convergence of approximation. Some applications of NURBS functions in shape optimization can been found in Natekar [3]. In this paper, NURBS functions are used as basis functions for both the design and the analysis to perform shape optimization procedures in linear elasticity. Hence, frequent remeshing procedures are needed. A geometric representation for CAD and CAE using NURBS reduces considerably the computational cost. In this work, the physical domains are defined by multiple and overlapping NURBS patches. Thus, it is necessary to manage the overlapping regions with multiple parametric description and complex quadrature rules are necessary. As matter of fact, once again the method is not isoparametric. Moreover, the

author advocates the extension of his work to the optimal design for fracture analysis and large deformation analysis where significant geometry changes between sequential analysis steps exists. More recently, Renken [67] used NURBS to represent the shape of droplets as well as to determine the solid–liquid and liquid–vapor interaction energies at the surfaces. These latter quantities were obtained by integrating surface energy coefficients over the appropriate surfaces. Cassale [4] has utilized trimmed surface patches to develop a boundary elements methods for linear elasticity. The interpolation of the solution is developed over the geometry generated by trimming surfaces somehow enhancing classical mesh of the boundary element method. Beyond NURBS functions, some other CAD technologies have also been introduced into analysis, such as the subdivision surfaces to model shells in Cirak [68]. In the publications mentioned above, we note similar intensions shared with isogeometric analysis, but bridging between geometry description and analysis is not fully achieved.

In CAD, a lot of work were done to develop physical based methodologies to improve the design of geometries. In the work of Celnikera et Gossard [5], the classical Ritz or Galerkin methods for linear elasticity was implemented into to a surface modeling tool (code ShapeWright) to define complex free-form shapes. The initial geometry is progressively deformed to the desire shape by applying convenient boundary conditions (loads and displacements). In this work, the shape functions are piecewise polynomial such as in the classical finite element method. In a similar context, several researchers have also proposed the use of the same basis functions for the geometric modeling and for the approximation of the solution fields. E.g., NURBS functions have been introduced by Terzopoulos and Qin [6] into this kind of interactive design paradigm. They introduced the concept of Dynamic NURBS which are physics-based models that incorporate the analysis in the context of Lagrangian mechanics at the interactive design stage. NURBS functions are used as analysis basis functions in the so-called D-NURBS element for calculation. Kagan et al. [69] and [70] developed a B-spline based finite element scheme for linear elasticity to interactively design shapes by applying loads. The link with the analysis is not straightforward. Some analysis concepts are used to optimize the way to freely define

shapes. Control points and associated weights are adjusted by applying forces on a fictive structure to make the shape design more intuitive. The strategies developed in this context have some similitude with Isogeometric Analysis.

1.3.1. Isogeometric Analysis: a tentative state of the art

In the field of Computer Aided Design (CAD), the Non-Uniform Rational B-spline (NURBS) is the predominant technology that is used to represent complex geometries. Its ability to characterize exactly some essential geometries including conics such as cylinders and spheres, which can only be approximated by polynomial functions, make it superior to these ancient tools like Beziers and B-spline functions. There exists a large number of literatures focused on NURBS, we can cite [63] and [71]. As a result of several decades of research, many efficient computer algorithms exist to perform the fast evaluation and refinement for NURBS based geometry. The fundamental concept sketched out by T.J.R Hughes & al [7] was to employ the basis function for geometric description as discretization tool for analysis. Considering NURBS as a typical example, in the NURBS based isogeometric analysis, the geometry and objective function space are both constructed on NURBS. Since this seminal paper, a monograph J. Cottrell & al [72] dedicated entirely to IGA has been published and applications can now be found in many different fields including structural mechanics, solid mechanics, fluid-structure interaction problems and contact problems… In the following, we draw a panel of different fields and problematics concerning isogeometric analysis.

Shell and plate problems

Shell and plate problems are a field where Isogeometric Analysis has been frequently applied to complete the simulations. It is important to note that in CAD software, volumetric geometries are represented by a surface description of the boundaries. It seems that this approach is especially well suited for surface-based geometries such as shell and plates. Moreover, it has been demonstrated that IGA has compelling benefits over conventional approaches for shell and plates because the high order continuity of basis functions. The use of the well-known k-refinement (smooth order elevation; see J. Cottrell

& al [73]) , in which C^{p-1} continuity is achieved using discretization of order p, leads to improved accuracy and robustness compared to the Finite Element Method. Numerous works have been initiated to assess the effectiveness of the method in the case of shells. Kiendl & al [74] has developed a Kirchhoff-Love NURBS based element which possesses an accurate description for the shell's curvature. Ensuing this work, they have proposed a method to connect multiple shell patches in J. Kiendl & al [75]. These structural patches have a C^1 or higher-order continuity in the interior, and are glue with a C^1continuity. Simultaneously, Benson & al [76] concentrated themselves on a Reissner-Mindlin type shell element based on a degraded 3D shell approach. Afterwards, the authors turned to Kirchhoff-Love type element for which the computational cost was less important in [77]. In Benson & al [78], the coupling of the this type of element with the previous Reissner-Mindlin element is carried out to solve the problem of imposing boundary condition and to achieve the coupling of multiple patches. Dornisch & al [79] proposed a method to calculate exactly the normal vector of the shell based on NURBS in the model of Reissner-Mindlin. At last, Echter & al [80] has constructed a group of hierarchical shell elements. This group begins from an element of type Kirchhoff-Love with three parameters similar to the one developed in [74] to a seven parameters solid-shell element based on the five parameters of a Reissner-Mindlin model, which takes into account the variation of the thickness. An important issue is the mesh locking phenomena occurring during the shear deformation of shells and plates. In the papers presented in this paragraph, the locking issue for NURBS based shell element is treated by employing high order approximations, which is natural for NURBS but is not sufficient to resolve completely the locking problem. Some technologies need to be developed to overcome these troubles.

In Elguedj & al [46], the authors developed a strategy to handle volumetric locking for which the projection method was applied to the strain for linear and nonlinear case. This work is an extension of the well-known $\bar{B}$ method developed by Hughes in [81]. In this approach, the strain is spitted into its isochoric and volumetric parts. The volumetric part is projected to a lower order interpolation space than that of displacement.

In the context of shells and plates, Echter & al [82] developed a so-called NURBS DSG element with a discrete shear gap method which is locking free. Bouclier & al [83]

investigated the traditional methods applied classical methods developed for locking finite elements to isogeometric analysis formulations: selective and reduced integration and $\bar{B}$ projection in thick plates and shells. Hereafter, they developed mixed formation for NURBS based solid shell elements under small perturbations. From this mixed formulation, a $\bar{B}$ projection for a solid shell element was derived for IGA. These two approaches with low order element and coarse mesh were proved to be efficient enough to produce accurate results when standard NURBS element suffers from locking issue. More recently, in Bouclier & al [84], the mixed method has been extended to the geometrically nonlinear static analysis of elastic shell structures. Moreover, the locking issue for Timoshenko beam was also addressed with an isogeometric collocation method in Beirão da Veiga & al [85] and in Auricchio & al [86]. A three-field variational approach was developed for NURBS based isogeometric analysis to simulate locking free incompressible deformations in Taylor & al [50].The Enhanced Assumed Strain method (EAS) has been extensively used to cure locking phenomena in finite elements. In Cardoso & al [51], EAS was extended to the NURBS-based isogeometric analysis. The proposed stabilized three-field formulation alleviates successfully the volumetric locking associated to the incompressible deformation, and prevents from spurious oscillations in the solution when high degree NURBS basis function was used. A similar formulation with the Assumed Natural Strain (ANS) method was implemented with quadratic NURBS based solid-shell element in Caseiro & al [87]. The locking pathologies like shear locking and membrane locking appeared to be effectively relived.

Incompressibility

It has been noted that standard displacement formulation surfers from volumetric locking when the material becomes incompressible. Actually, problems arise even when the deformation is nearly incompressible, in the case of linear elasticity, that means the Poisson's ratio ν is close to 0.5 (e.g. 0.49 and higher). Incompressible and nearly incompressible behaviors are encountered in a variety of real engineering problems with materials such as deformations of elastomer and undrained soils or elastic-plastic response of metals. Identical problem occurs for incompressible fluids as well.

In order to resolve the locking issue, several techniques have been developed for classical finite element methods. Here we can list: mixed finite element method, selective and reduced integration, enhanced assumed strain which is an extension of incompatible mode technique, and finally the strain projection method such as $\bar{B}$ projection in Hughes & al [81].

It was shown in Elguedj & al [46] that the standard NURBS based element of Isogeometric analysis is also not stable when incompressible constraints exit in the materials. As mentioned above, the authors extended the $\bar{B}$ projection of FEM to IGA and developed a $\bar{F}$ projection for nonlinear problems to simulate incompressible deformations of rubber-like materials. Later, this $\bar{F}$ projection method was applied to large strain plasticity in [88]. Selective and reduced integration approach for NURBS based element was studied in Adam & al[47]. They have assessed the performance of different rules on weakly compressible linear elasticity problem. Note that the problem of incompressibility is also addressed in fluid mechanics for Stokes and incompressible Navier-Stokes problems. Enhanced and assumed strain method for IGA was investigated in Cardoso & al [51] and Caseriro & al [89]. A three field mixed formulation method combined with isogeometric analysis was implemented in Taylor & al [50]. To stabilized the mixed formulation, Galerkin Least Squares method was developed later in Cardoso & al [90] and in Kadapa & al [91]. More recently, in the work of Kadapa & al [62], inf-sup stable displacement-pressure combinations for mixed formulation are investigation for NURBS based element. Based on subdivision property of B-splines, a group of stable displacement-pressure elements for IGA were proposed. Numerical inf-sup tests are performed on several problems exhibiting invariant inf-sup constants when the mesh is refined. The performances were investigated on nearly incompressible elastic and incompressible elasto-plastic materials.

Fluid mechanics and Fluid-Structure Interaction

Since the origin of IGA, the method has been extended to fluid problems because of the accuracy obtained in numerical solutions. In Y.Bazilevs & al. [92], a variational multiscale

method is presented to simulate turbulence. It is an LES-type formulation derived from the Navier-Stokes equations within a space-time formulation. Variational projections in place of traditional filtered equations is proposed in the context of NURBS approximations. The proposed implementation for a fixed spatial domain is based on the semi-discrete generalized-α method. The superiority of the NURBS basis compared with finite elements is shown on various numerical applications. Beyond the special care of fine scale modeling in this work, the authors show the high accuracy performance of NURBS. Quadratic NURBS significantly improve accuracy of results over linear elements. E.g. on a turbulent channel flow, they show that on "coarse" mesh quadratic NURBS are more accurate than a spectral Galerkin LES, and on a finer mesh the results are identical to DNS ones. Further investigations based on similar formulation can be found e.g. in Bazilives & al. [93] and prove the accuracy of NURBS in the contest of turbulence modeling. Note that the authors proposed alternative methods to weakly impose Dirichlet boundary conditions particularly for the lower IGA elements in fluid mechanics especially when coarse boundary-layer mesh are employed (see Bazilev & al [94], [95]). Similar IGA formulations may be found in Chang & al [96], Coloms & al[97] and Golshan & al [98], in [99] for the simulation of coupled multi-ion transport in turbulent flows.

From a general point of view, the main numerical difficulties in fluid mechanics occurring for FEM or IGA are the mixed character of the variational formulation when velocity and pressure are solved, and the advective character of Navier-Stokes equations. Each issue has been addressed in the context of IGA. The mixed character of velocity/pressure formations of incompressible Navier-Stokes equations in the context of IGA is studied in Nielen & al [49]. A similar study for the Stokes problem is given in Buffa & al [48]. This proposed a set of pair of NURBS interpolations for velocity and pressure including equal order interpolations but different inter-element continuity order and different knots number. Similar pair of stable B-splines interpolations are proposed in Rüberg & al [100] but the velocity/pressure interpolation are build using the two-scale properties of B-spline functions. The element is stabilized by projecting the pressure on a grid coarser than the one of velocity. This defined a new family of inf-sup stable elements which is used here for an immersed finite element method to compute flows with complex moving boundaries. In Hosseini & al [101], similar velocity/pressure combination with B-splines element of

type Taylor-Hood are evaluated for the Navier-Stokes equations. In Niemi et al [102], a discontinuous Petrov-Galerkin method based on optimal test space norm is proposed. This approach used Bézier element for 1D convection dominated flow problems with diffusion, and seems to be robust. In Manni & al [103], the authors present an extension to advection-diffusion problems. Exponential B-Splines and variable degree B-Splines are used to approximate the solution. Stable results are obtained for 1D and 2D test without stabilization schemes such as SUPG, GLS or Douglas Wang with high order B-Splines. It is important to note that with standard B-Splines, a stabilized formulation is mandatory. Similar results are found in Dhawan & al [104].

It is obvious that one of the most important feature of IGA is its capability to accurately deal with interface problems, either in capturing interfaces or describing them.

In Gomez & al [105], Caquero & al [106] and Bueno & al [107], IGA has been applied to the simulation of Navier-Stokes-Korteweg equations that enables to model interfacial phenomena. This is a phase-field to describe phase transition phenomena representation of water/water-vapor two phase flow. A refinement technique based on physical considerations has been utilized to capture thin transitions. High order partial difference operators are treated in straightforward manner benefiting by IGA. Similar results are obtained with a formulation developed in Vignal & al [108] for the simulation of two phase field with the coupled Navier-Stokes and Cahn-Hilliard equations.

A major feature of IGA remains the use of CAD functions for the analysis. Immerse boundary methods have been developed in that context, B-Splines or NURBS functions providing accuracy in the capturing of interfaces. In Schilinger & al [109], a hierarchical refinement method for NURBS is presented for the cell method and providing flexibility compared to classical CAD description. In Hsu & al [110], the boundary representation is directly built from B-rep representation for the immersogeometric fluid flow simulation. Alternative numerical methods have been developed within the context of NURBS geometries. In Heinrich & al [111], a finite volume method based on a domain parametrized by NURBS is proposed for incompressible Navier-Stokes flows simulation. This method combined with a computational structure method allows to develop a FSI solver with on gaps or overlaps between domains. Free-surface flow simulation (air/water)

is addressed in Akkerman & al [112] within the framework of Y.Bazilevs & al. [92] and is based on a level-set approach. The accuracy of IGA makes the formulation efficient to capture free-surface. In Kees & al [113], a conservative formulation for handling complex free-surface simulations that can be combined with isogeometric analysis is propsed. The basic idea here is to couple the level-set equation with the volume fraction equation.

The smoothness of NURBS basis function is also attractive for analysis of Fluid-Structure Interaction (FSI) problems. The first endeavor goes back to 2006, in the work of Y. Bazilevs & al [114] where a framework based on NURBS is introduced for Fluid-Structure Interaction problem. This framework contains a subdomain of fluid modeled by the incompressible Navier-Stokes equations. The motion description is based on an Arbitrary Lagrange-Eulerian Method (ALE). The solid subdomain is modeled by a Saint Venant Kirchhoff model with a finite strain elasticity constitutive law (Lagrangian description, geometrical nonlinearity). Interaction between fluid and structure is achieved by coupling both the velocity field and the stress field between solid and fluid domains. The compatibility of velocity and stress are respectively strongly and weakly imposed. The results given by isogeometric analysis show a good agreement with the reference computations. In this paper, a computation of a patient-specific abdominal aorta is also performed, giving good qualitative agreement with similar models. Y. Bazilevs & al [115] makes a pedagogic presentation of NURBS based IGA fluid-structure interactions. Careful attention is paid to the derivation of various forms of the conservation equations (conservation properties of the semi-discrete and fully discretized systems) and to time integration algorithm (unified presentation of the generalized-α time integration method for FSI). Compared with the previous work [114], a nonlinear hyperelastic model is employed to simulate the behavior of the structure. The model is tested on three computations: a flow over an elastic beam, the inflation of a balloon, and blood flow in a patient- specific model of an abdominal aortic aneurysm. The formulation is robust in all cases. In Bazilevs & al [116], the author has presented the first 3D patient-specific FSI simulation of Left Ventricular Assist Devices (LVADs). As in previous papers, a similar NURBS based isogeometric analysis FSI framework is applied to a patient specific model of the thoracic aorta. The computational results are in qualitative good agreement with clinical observations. Isogeometric analysis FSI is performed also for the flows around

rotating components in Bazilevs & al [117]. Finite element causes gap and overlapping between the rotating and stationary subdomains. These issues can be solved automatically by a NURBS based geometric description. Although geometric compatibility is exactly achieved, the discretization of the flow velocity and pressure remains incompatible at the interface between the stationary and rotating subdomains. This incompatibility is handled by using a weak enforcement of the continuity of solution fields at the interface (similar to the coupling terms in Discontinuous Galerkin method). The new methodology was successfully applied to the problems of two propellers inside a rectangular box filled with a viscous flow. In the work of Bazilevs & al [118], isogeometric FSI analysis with non-matching discretization on interface is studied. For the coupling, the augmented Lagrangian multiplier method is adopted. However, the interface Lagrange multiplier is formally eliminated such that the final FSI formulation is written in terms of primal variables. The aerodynamic domain is modeled by volumetric quadratic NURBS then discretized by low-order Finite element method while the rotor structure is modeled with a cubic T-Spline based discretization of a rotation-free Kirchhoff-Love shell. The IGA-FEM hybrid framework is successfully applied to the simulation of a 5MW wind turbine rotor at full scale. The author has compared the T-Spline/NURBS discretization with T-Spline/FEM discretization. Through which it can be seen that the significantly finer NURBS mesh is capable of resolving some of the trailing edge turbulence; while the coarse FEM mesh produces a visually smoother solution. Nevertheless, the large scale features of the flow are qualitatively and quantitatively very similar for both discretization.

Contact problems

It might be a natural intuition to think that smooth, compactly-supported basis functions of Isogeometric Analysis might improve the modeling of contact problems. The classical C^0 continuous finite element basis function often create serious convergence problem for contact mechanics due to the gaps and overlapping on contact interface. Hence, various surface smoothing algorithms have been developed. However, the smoothed surface discretization are mostly incompatible with the volumetric discretization. In Temizer & al [119], a systematic mortar-based study of contact problems with isogeometric analysis was

initiated. The design of NURBS surface patches to define the contact zone in the Knot-To-Surface (KTS) algorithm is directly inherited from the NURBS volume parametrization in a straightforward manner. Consequently, it makes it possible to achieve arbitrary smoothness across contact element interfaces while preserving a consistency between surface and volume discretization. These efforts were simultaneously reported in Lu [120] in a frictionless setting through alternative robust contact treatments based on the works of Papadopoulos & al [121]. Convenient qualitative results for thermomechanical frictionless contact constraints were obtained even at coarse resolutions in 2D and 3D. Moreover, the pressure distributions in the classical Hertz contact problem is much smoother than those arising from Lagrange discretization. In particular, the oscillations that were reported for the Hertz problem for higher-order Lagrange discretization were significantly alleviated with NURBS discretization. Subsequently, a two-dimensional mortar-based approach with friction was investigated in De Lorenzis & al [122]. High order NURBS discretization was investigated to deliver smoother global interactions while ensuring the local quality of the solution. A three-dimensional mortar-based frictional contact problem with NURBS interpolations in the finite deformation regime is contributed in Temizer & al [123]. The contact integrals are evaluated through a mortar approach where the geometrical and frictional contact constraints are treated through a projection to control point quantities. It was shown that this framework approach offers robust local results even at coarse resolutions of the contact interface with smooth pressure and tangential traction distributions. Lastly, in Dittemann & al [124], thermomechanical Mortar contact algorithms and their application to NURBS based Isogeometric Analysis are investigated in the context of nonlinear elasticity. Mortar methods are applied to both the mechanical field and the thermal field in order to model the frictional contact, the energy transfer between the surfaces as well as the frictional heating. Compared with traditional approaches, the benefits of using Isogeometric analysis over contact problems are evident, since smooth contacts surfaces are obtained, leading to more physically accurate stress.

Aside from the applications we have talked about above, this novel isogeometric analysis has also been studied for structural vibration problems (Cottrell & al [125], Hughes & al [126] and Wang & al [127]), optimization problems (Wall & al [128], Qian & al [129] and Manh & al [130]) and others problems. It is to be noted that today the isogeometric

analysis is being widely utilized and rapidly developing. The lectures we have listed here cannot be expected as all-inclusive.

1.3.2. Alternative geometric descriptions

In most of contributions to the IGA, splines and NURBS have been chosen as support of IGA, since these technologies are ubiquitous in CAD. First, NURBS are convenient to model freeform surface and can exactly represent all elliptic surfaces and volumes such as cylinders, spheres, ellipsoids, etc. Many efficient and numerically stable algorithms to generate and refine NURBS have been developed and are widely used in CAD software. Moreover, NURBS possess valuable mathematical properties, such as the ability to be refined conserving the exact geometry up to C^{p-1} continuity for a NURBS of degree p, and the convex hull properties.

In spite of this, NURBS has some innate deficiencies which cause difficulties for numerical simulations. In order to model topologically complex geometries, multi-patch of NURBS have to be brought in order to build complex geometries. In real life geometries issues from CAD systems, gaps and overlaps at intersections of surface cannot be avoided. From the simulation point of view, NURBS requires the insertion of an entire row of control points to perform the knot insertion refinement (h-refinement) because of its intrinsic tensor product structure. Adaptive and local refinement on single patch of NURBS is untouchable. Besides, single patch based geometries suffer from the topological limitation, e.g. a 2D NURBS must have a rectangular topology. In order to overcome these difficulties, some endeavors relate to the development of computational geometry technologies. For those frequently seen, we can cite Hierarchical B-splines, T-splines, Subdivision Surface and Spline forest.

Hierarchical B-splines

NURBS based geometry has a tensor-product structure. That's why a knot insertion refinement on NURBS causes the increasing of control points far from region of interest.

Hierarchical B-splines introduced in D.Forsey & al[131], [132] offered a method of localizing the effect of refinement through the use of overlays. These overlays are hierarchically controlled subdivisions. In the framework of computational engineering hierarchical refinement of NURBS has recently received increasing attention. In D. Schillinger & al [133], this technology was adapted with B-splines finite element method. Their numerical experiments illustrated the computational performances of the method. They describe the imposition of unfitted boundary conditions and some fast technique to generate hierarchical grids of Hierarchical B-splines. A combination with IGA and Hierarchical B-splines was presented in A. Vuong & al [134]. Apart from the application of adaptive local refinement, some fundamental properties like linear independence and partition of unity of splines space were also investigated. Recently, hierarchical refinement of NURBS for some elementary fluid and structural analysis problems in two and three dimension combined with immersed boundary methods were tested in D. Schillinger & al [135]. The authors have made up a design-through-analysis procedure and computed some problems with complex engineering part like a ship propeller. The technology of hierarchical B-splines possesses principally two advantages. First, hierarchical B-splines rely on the principle of B-spline subdivision, which makes it possible to maintain linear independence throughout the refinement process. C^{p-1} continuity of NURBS is also maintained in its hierarchically refined basis. Second, hierarchical B-splines rely on a local tensor product structure, and they can be easily generalized to arbitrary dimensions. However, due to its hierarchically tensor-product structure, hierarchical B-splines suffers also from topological limitations such as NURBS.

T-splines

A T-splines surface can be regarded as a generalization of NURBS. In contrast to NURBS. A row of T-splines control points is allowed to terminate without traversing the entire surface. The final control point in a partial row is called a T-junction. This recent computational geometry technology was first presented in the work T. W. Sederberg et al [136]. A T-mesh is basically a rectangular grid that allows T-junctions that serves the purpose to localize the control point and deduce its correspondent knot vectors as shown

in Figure 13. It is proved that T-spines can model watertight and topologically complex geometry, such as a propeller in Figure 14. Moreover, a primary local refinement algorithm and a method for merging of several B-splines surfaces that have different knot vectors into a single gap-free model are described in this paper. Lately, in T. W. Sederberg et al[137], a T-spline simplification algorithm for eliminating NURBS superfluous control points was presented. Based on the latter, the local refinement of T-splines has been improved such that the number of additional control points needed for inserting a requested control is significantly reduced. The linear independence of functions corresponding to the remaining control points has been proved in X.Li et al [138]. This paper shows that, for any given T-spline, the linear independence of its blending functions can be determined by computing the nullity of the T-spline-to-NURBS transform matrix. Furthermore, An analysis-suitable class of T- splines were recently introduced in M.Scoot et al [139] and X.Li et al [140] where T-splines are shown, to be linearly independent, to form a partition of unity. They can be refined in a highly localized manner.

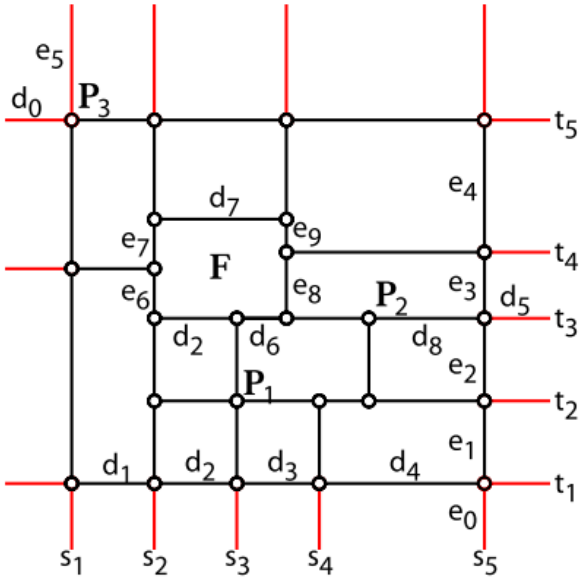


Figure 13. A Pre-image of T-mesh from [136].

T-splines as basis for isogeometric analysis was initially explored in the paper Y.Bazeilevs & al [141]. Tests of T-splines on certain trial two-dimensional and three-dimensional fluid and structural analysis problems confirms that this technology provides a nearly complete

basis analysis. Its local refinement property allows to obtain an acurate numerical solution with less control points, thus less degree of freedom and chipper computational cost. The author has also mentioned a challenging issue about the generation of tri-variate T-spline geometry models. In [142], a posteriori error estimation techniques have been combined with T-spines to realize an adaptive refinement strategy. In order to facilitate the implementation of T-Splines IGA M.Sccot & al [143] presented an extraction operator for T-splines on Bézier by generalizing their previous work on NURBS (see M.Sccot & al [144]). Thus, it is possible to adapt the T-splines based IGA to a classical FEM code by simply modifying the shape functions subroutine. This operator localizes the topological and global smoothness information at the level of the element. It represents a canonical treatment of T-junctions, referred to as 'hanging nodes' in finite element analysis and a fundamental feature of T-splines. Error estimation of local h-refinement provides a theoretical foundation to evaluate the accuracy and the convergence of isogeometric analysis.

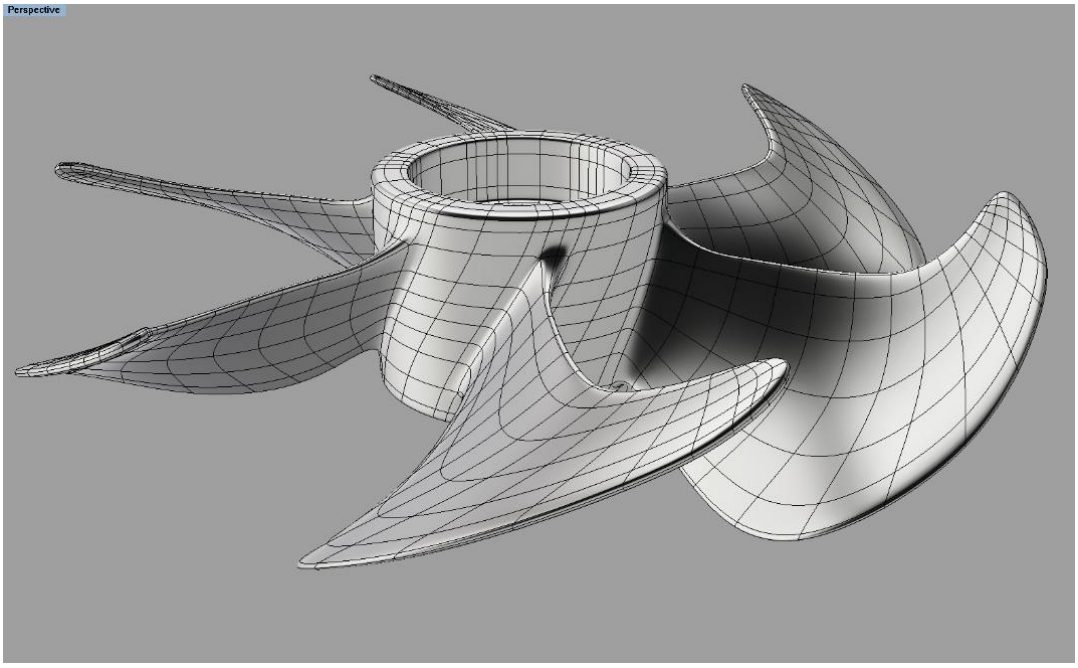


Figure 14. A propeller based on T-splines from [145].

Subdivision Surface

Subdivision is another powerful technique in surface modeling. It is compatible with NURBS as the standard in CAD systems. The geometry models can be refined with a well-chosen approximation scheme to achieve a required accuracy of the numerical simulation. The subdivision schemes are simple, efficient and can be applied to meshes with arbitrary topology. Recently, subdivision surfaces or solids have been applied to Isogeometric analysis. Volumetric IGA based on Catmull-Clark solids was investigated in D. Burkhart & al [146]. With a C^2 continuous element, a faster convergence than tri-linear and tri-quadratic elements of FEM is proved by the experiments with a quite simple shape domain. Powell-Sabin splines were used as IGA tools for advection-diffusion-reaction problems H. Speleers & al [147]. It is proved that subdivision based isogeometric method can also handle complex geometry domain problems. A robust and efficient implementation of an isogeometric discretization approach to partial differential equations on surfaces using subdivision methodology has been discussed in B. Jüttler & al [148]. A discretization of Kirchhoff–Love thin shells based on a subdivision algorithm that generalizes NURBS to arbitrary topology is proposed in A. Riffnaller-Schiefer [149]. In Q. Pan & al [150], they have developed the finite element method based on the extended Catmull-Clark surface subdivision which can be integrated into the framework of IGA scheme. Applications to the Poisson equation proved that their approach is faster than a linear FEM element calculation.

Conclusion

Aside from these three computational geometry technologies we have discussed above, there are others candidates which were explored such as Splines forest in M.Scott & al [151], polynomial splines over T-meshes in J. Deng & al [152] and LR B-splines in T. Dokken & al [153]. And along with development in the domain of computational geometries. There will be more powerful technology emerging in the future. Under current circumstances, we believe that the ideal computational geometry technology should have capabilities as follows.

- **Exact geometric representation**. This guaranties the numerical solution will not be disturbed by the approximations at the geometry level. With the exact geometry at the

analysis step, the refinement procedure could be facilitated since redundant communication with CAD software is not obligatory.

- **Local refinement ability.** It is an import capability of which the NURBS is short. Local refinement allows to obtain more precise numerical solution with few control points added. Compared to global refinement, it reduces importantly the computational cost and strengthens the stability of geometric model and numerical solution.

- **Arbitrary Topological geometry modeling ability.** The realistic engineering level simulation usually needs to solve problems defined on complex geometries, such as a ship propeller in Figure 14 which can be model with only one T-splines patch. With NURBS based isogeometric analysis, this kind of problems can only be solved with multi-patch. That means additional computational cost is inevitably paid to impose the solution's inter-patch continuity. The division caused by geometry modeling may cause errors in the numerical solution, thus degrade the precision and convergence velocity in analysis procedure.

- **Tri-variate volumetric parametrization.** Different from the need of geometric modeling, the analysis demands the solution inside of a three-dimensional model. As today's CAD software model, a three-dimensional part by define its closed exterior surfaces. The most significant challenge facing isogeometric analysis is developing three-dimensional spline parameterizations from those surfaces (see e.g. H. Al Akhrasa et al. [154]).

1.3.3. Numerical integration for Splines and NURBS

Numerical Integration

The Isogeometric analysis method is based on the Galerkin method. A weak form equivalent to the strong form description of the problem needs to be derived. The matrix contributions corresponding to the integrals of weak form has to be evaluated by numerical integration stays an issue of efficiency specially when high order basis functions are employed. The classical Gauss-Legendre quadrature rule which has been widely used in

FEM cannot properly take into account the high order inter-element continuity of NURBS basis function. Subsequently, it leads to non-optimal assembly costs, significantly affecting the performance of IGA methods. Some endeavors have been taken to develop an optimal or sub-optimal quadrature rule by taking into account the inter-element higher continuity of NURBS basis functions. Improvised quadrature rules have been proposed by Hughes et al. [155], Auricchio et al.[156], and Schillinger et al.[157], but the development of a general effective solution for Galerkin-based IGA methods still remains an open problem.

Isogeometric Collocation method

Beyond the effects in improving the quadrature rules, another way to avoid the costly numerical integration turns up: Isogeometric Collocation method (IGA-C). As opposed to Galerkin method, collocation method is based on the discretization of the strong form of the governing partial differential equations. The requirements to fulfill the continuity and derivability requirements for solution and the test functions for the collocation method are naturally fulfilled by the basis function, e.g. B-splines and NURBS functions. Auricchio et al.[158] developed a one-dimensional theoretical analysis of the method, which served the dual purpose of providing the theoretical background and guiding the selection of collocation points. They presented numerical tests on simple elliptic problems in one, two and three dimensions. They studied the accuracy of the method, the behavior of the discrete eigen spectrum and discussed the performance of the scheme with respect to the choice of collocation points. In Auricchio et al. [159], a variational interpretation of the collocation scheme is developed and included special considerations about the patch interfaces and external boundaries. The proposed framework has also been extended to dynamics and described explicit predictor multi-corrector time integration algorithms. Shcillinger et al.[160] compared IGA-C with isogeometric Galerkin (IGA-G) and standard finite element methods (FEA-G) in terms of their computational efficiency. They first assessed the computational cost in floating point operations for the evaluation and assembly of stiffness matrices and residual vectors. By the way of operation counts, IGA-C significantly reduces the computational cost compared to IGA-G and FEA-G. They also showed that for IGA-C the bandwidth of the stiffness matrix and the cost of matrix-vector products are much

chiper than in IGA-G and FEA-G. The results showed that IGA-C can be orders of magnitude faster than IGA-G and FEA-G for the same level of accuracy. The IGA-C is employed to treat the locking issue of shell problems in Beirão da Veiga et al. [85] for the approximation of initially straight planar Timoshenko beams. Following the same issue, Auricchio et al. [86] extended the investigation to curved spatial Timoshenko rods. The proposed schemes, based on standard mixed formulations, were shown theoretically and computationally to be free of shear locking. In Kiendl et al.[161] IGA-C approach has been devoted to the solution of Reissner-Mindlin plate problems. De Lorenzis et al.[162] addressed two important issues of IGA-C method's development namely the imposition of Neumann boundary conditions and the enforcement of contact constraints between multi-patch with non-conforming discretization. The authors proposed a frictionless contact formulation in the collocation setting.

1.3.4. Other studies of IGA

Isogeometric Analysis has been in many other contexts. Many domains in computational mechanics are today addressed. As the rate of publication increases every month, let's mention among them

- IGA combined to reduced order modeling techniques based on proper orthogonal decomposition for minimizing computational time for repetitive simulations in shape flow optimization computations for Stokes flow in Salmoiraghi & al [163]; Similar strategies developed in Manzoni & al [164] for the simulation of potential flows about NACA profiles airfoils.
- Electromagnetism applications in Buffa & al [165].
- NURBS for discretization in time (Lagrange polynomial for discretization in space) in space-time formulation for the modeling of fluid-structure interactions problems with application to spacecraft parachutes and flapping-wing aerodynamic Takizawa & al [166].

- Lubricated piston dynamics for the resolution of the Reynolds equation in Liu & al [167] for which stability is achieved, without stabilization scheme compared with Habchi & al [168] in which SUPG stabilization has been used.
- Flows in porous media, anisotropic porous media Shahrbanozadeh & al [169], flow in porous deformable media – Darcy like flow – mixed formulation flows in Vuong & al[170].
- Discontinuous isogeometric analysis (discontinuous Galerkin method) for Neutron transport equation in Owens & al [171].
- Strategies to mix Lagrangian FE and IGA application to incompressible flow problems Rasool & al [172]

2. Integration of IGA to an Object-Oriented Code for FEM

2.1. High level abstraction computational mechanics

Software engineering in computational mechanics has significantly evolved since the 50s following the development of programming languages. Until 90s, the major approaches were based on procedural languages like Fortran, C and Pascal. These traditional procedural programming languages usually suffer from difficulties in case of large scale developments such as:

a) The global access to data structure and/or too many function parameters decrease the flexibility of the system.

b) Clear modularity in language level is hard. A particular algorithm usually associates to many data structure and functions. It must be documented clearly for end users.

c) Reuse of existing codes to adapt them for new models, new algorithms or slightly different applications is difficult or impossible. Sometimes, a high level of knowledge of the codes is necessary.

Maintainability and extendibility has been initially achieved through progress in code modularity. This could be obtained using a sequentially organized code: sequential call to functions or subroutines. Data structuring capabilities appeared when using languages such as C or Pascal providing capabilities of controlling data flows. Even though, when software complexity rapidly increased with the type of problem addressed, code maintainability and extendibility have become overmuch difficult in this context.

Object-oriented programming whose applications to the finite element method emerged in late 80s enforce a better control data flow (e.g. see Rehak et al.[8]). The key concept of object-oriented programming is the object. This is an entity that contains both data (attributes) and actions (methods). Objects are instances of classes and communicate through messages telling the receiver WHAT is expected but leaving it to each object to determine HOW to achieve the requested task. Objects encapsulate their own data which can only be accessed by the object itself, upon receiving a message from another object (e.g. see Zimmermann et Pèlerin [10]). The object-oriented approach organizes the code in a hierarchy of classes taking advantage of inheritance and polymorphism.

This programming paradigm enables the developers to better control large systems with a high level of reusability. It offers a powerful alternative of structuring codes.

2.1.1. Object-oriented finite elements in computational mechanics

In the pioneering work of Miller [9] and Rehak et al[8], some basic structuring concepts for the FEM were represented. In the article of Miller [9], a LISP framework for finite element computations described. In the work of Fenves [173], the properties of modularity and code reuse are highlighted, and the efficiency in maintenance and implementation is described as a key idea of the approach. The program developed was limited to linear elasticity. Roughly speaking, the main objects proposed in these pioneering works are related to basic structures such as nodes and elements, and also to some linear algebra features. In the same time, complete approaches have been developed for static and dynamic finite element analysis (e.g. see Zimmermann et Pèlerin [10], Baugh et Rehak [174], Scholz [175], Devloo [176] and Pèlerin et Zimmermann [177]). The global structuring of the FEM for linear elasticity is addressed through introducing new features such as the object degree of freedom in Zimmermann et Pèlerin [10], and objects covering the time integration algorithms. Nonlinear FEM has then been addressed later, e.g. elasto-plasticity in Menbtrey et al [178]. One of the most complete approach for nonlinear material modeling was the one proposed in Besson et al [11] and Foerch et al [179]. Note that in the same time, some successful attempts of structuring classical FE codes have been developed: see e.g. SIC (Interactive System Design) of Golay et al [180] and CAST3M of Verpeaux et al [181]. In the latter, some structuring capabilities have been developed based on advanced memory management systems in Fortran. However, the object-oriented design remains one of the most efficient strategies to manage complexity.

Since the origin, the object-oriented programming has been widely applied in all the fields of computational mechanics and related domains (e.g. see Mackerle [182] for a tentative exhaustive bibliography). We can mention a large number of papers covering a wide range of applications and computing frameworks. Among them, we can cite: numerical tools for linear algebra Zeglinski et al [183], creation of interactive codes Mackie [184], integration

of artificial intelligence in finite element systems Bomme et Zimmermann [185], fractures and damage problems Fang et al [186], parallel computing in solid and fluid mechanics Adeli et al [187], mechanics of deformable solids in large transformations Rio et al [188], multiphysics problems Eyheramendy [189], Dadvand et al [190] and Tonks et al [191], contact problems Fang et al [186] and Ma et Wei [192]. This list is of course not exhaustive and shows that the object-oriented paradigm is now widely spread out in the scientific computing community. The object-oriented paradigm is today a common modeling tool to tackle the most challenging problems. Following this track, new approaches have been developed in the middle 1990's bringing both, additional structuring capabilities and better software integration. Among them, the most popular is based on the Java language. Roughly speaking, the key points of Java are:

a) an object-oriented programming language allowing high abstraction level data structures

b) the Java virtual machine which ensures a wide portability of the applications

c) the Java platform which provides a large number of predefined classes: I/O, object persistency, networking, multiple process management, GUIs development, security, internationalization, …

Java initially retained some attention for its networking capabilities and its easy Internet portability. E.g. in Nuggehally et al [193], a trial application based on a boundary element method is proposed. Similarly, a web-based application for fracture mechanics can be found in Nikishkov et Kanda [194]. In these works, only a few innovative structuring features are proposed. An original way to consider Java is use it to couple and manage traditional codes written in C/C++/Fortran. This permits the developers to use ancient codes or part of code in coupled applications, preserving the original computational efficiency. E.g. in Miller et al [195], an interactive finite element application based on a coupled C++/Java is described. Comparative tests with Fortran and C are conducted on small problems using direct solvers based on tensor computations, this aims at illustrating the high efficiency computational potential of Java in the context of code coupling. To go

further, similar conclusions were drawn in Bull et al [196], Häuser et al [197] and Eyheramendy [12], where good performances of pure Java application are exhibited on simple matrix/vector products. In Marchand et al [198], the development of GUIs is put in prominent position on an unstructured mesh generator. Most of the computational applications have been conducted in the computer science community, including computational mechanics applications. In Padial-Collins et al [199] and VanderHeyden et al [200], the Java environment for distributed computations of complex multiphase flows CartaBlanca is presented. Based on a finite volume approach, a solution scheme based on a Newton-Krylov algorithm is described. CartaBlanca exhibits good performances as shown in Padial-Collins et al [199]. A similar environment has been developed to simulate electromagnetisms problems in Baduel et al [201]. Both applications show the high potential of the approach to design more complex and general computational tools in mechanics including complex parallelism paradigms. These developments exhibit the networking facilities provided by Java. A large number of publications shows the interest of Java and its efficiency in different context of numerical analysis: direct solution of linear systems Nikishkov [202], FFT and iterative and direct linear systems solvers on Euler type flows Bull et al [196], solution of Navier-Stokes flows Häuser et al [197] and Riley et al [203]. More recently in Nikishkov [204] [205], the description of a finite element code in Java was proposed. The proposed design remains rather similar to the existing ones based on C++ approaches. Note that in Eyheramendy et al [206], two new programing principles are proposed to maintain consistency of finite element codes. Based on similar principle, C# has also been used to design finite element applications (see e.g. Heng et Mackie [207] and Mackie [208]). Thus, the code is integrated with the platform ".NET" which allows developers to mix codes developed using different programming languages. Roughly speaking, the OO paradigm has brought a real modularity and robustness to finite elements software but it is now possible to consider extendibility capabilities of traditional codes in a new manner by e.g. introducing high abstraction level paradigms to directly consider mathematical models. Even, it is not directly link to our purpose, we propose a quick review of mathematical algebraic approaches for finite elements and related schemes, because they might open a new era for computational tools.

2.1.2. Generic approaches for fast extendibility capabilities in computational mechanics

Today, focusing only on the design of a computational code is not enough to fast extend codes in computational mechanics. Some works have shown that high level abstraction may help this way to consider variational forms for given problems. Even this goes beyond the scope of this work, we consider important to recall main tracks opened in the domain of the automation of elemental contributions computation and their integration into software.

The use of algebraic manipulation software has always been a point of interest for finite elements developments since the 70's. The first related works seems to be Luft et al [209], Gunderson et Cetiner [210] and Noor et Andersen [211]. They described a methodology to automatically generate finite element matrices. Later on, many codes were presented for solving different kind of finite element problems. As proposed in Eyheramendy et Zimmermann [212], the symbolic approaches can be grouped into three main categories. In the first one, we may found the works in which the authors have developed procedures for solving finite element environments in computer algebra. A classical finite element approach is used in a conventional symbolic computing environment such as Maple, Mathematica, Matlab, etc. Thus, some variables may be stored in symbolic form, which allows evaluating their influence on the numerical results (see Choi et Nomura [213] and Iokimidis [214]). The semi-analytical numerical approaches offer a perfect environment for parametric studies using finite element solving method. But in view of current developments of symbolic software, such an approach can only apply to mechanical problems of limited size. This strategy remains difficult to extend to actual problems due to the necessary computational efficiency needed to lead symbolic computations. The second category corresponds to approaches whose main goal is to improve the computational efficiency of conventional finite element codes. These approaches aim to perform a preliminary symbolic computing in order to improve performance in which preliminary floating point operators are introduced (see e.g. Yagawa et al [215], Yang [216] and Silvester et Chamlian [217]). The use of computer algebra software showed that, on

one hand, it is possible to use the efficiency and flexibility of such environments to optimize the expressions needed for the evaluation of element matrices, and on the other hand, the numerical code can be automatically generated from the symbolic environment. This kind of strategy is today widely adopted to optimize computational efficiency and/or avoid though hand calculations. Finally, some authors have sought to accelerate and automate the development of numerical code using symbolic manipulations tools. Initially, some authors proposed to develop finite elements matrices such as in Gunderson et Cetiner [210] (see Eyheramendy et Zimmermann [218] for a thorough state of the art). More recently Eyheramendy et al [219] [212] proposed an alternative way to develop finite element models and codes. The concept is based on a hybrid symbolic/numerical approach for solving mechanical problem and on high-level software, object-oriented programming (Smalltalk and C++). This environment manages all the necessary concepts to the physical problems solution: manipulation of partial differential equations, variational forms, integration by parts, weak forms, finite element approximations… The result of these algebraic operations is a set of data items to be introduced into a conventional numerical code. At the same time, Korelc [15] [220] has presented an approach based on Mathematica that aims at generating finite element formulations. The environment is based on two main libraries. The AceGen tool which automate the generation of the code and computational templates combines Mathematica with automatic differentiation techniques. The approach was designed for complex problems including constitutive law modeling. It allows the generation of multi-environment finite element codes resulting from the same symbolic description. The automatic differentiation techniques allow the management of strongly nonlinear problems. The time needed for deriving the code is much less than the simulation time of the typical industrial problems. More recently, Logg [16] have proposed the FEniCS project, aiming at automating the finite element modeling. FEniCS is a global project that consists of multiple tools. FEAT (Finite Element Automatic Tabular) is a spreadsheet of finite element. Its main role is to automate the generation of the basic functions of finite elements. FFC (FEniCS Form Compiler) is a compiler for variational forms that automates the key step in the implementation of the finite element method for solving partial differential equations. In some senses, the software commercial package COMSOL Multiphysics (see [221]) aims at a similar strategy. In this tool, the equations of

the problem can be introduced either as strong or weak form. The advantage of the tool is that it provides a relatively user-friendly infrastructure to manage the totality of a problem since the equations of the problem to the definition of the geometry, external actions... But the algorithms related to finite element resolution are somehow locked. In the solution of strongly coupled nonlinear problems in a Lagrangian framework, it is not so obvious that generic algorithms can be sufficiently efficient, even if a wide range of problems can be tackled. In the mathematical tool freefem++ (see Hecht [222]), the user defines the variational formulation and may have access to various discretizations. The variational formulation can be built from pre-defined differential operators, and the advantage of the language developed gives access to the numerical algorithm to solve the problem. The constitutive aspects are not considered here.

Thus, it is often essential to have access to the different levels of algorithmic patterns in the construction of solution schemes for new problems, as shown in most recent works (e.g. Hecht [222], Eyheramendy et Zimmermann [212], [218], [219], Korelc [15], [220] and Logg [16]) where mathematical structures are provided to build the finite elements solver. It seems obvious that these approaches probably open the most promising tracks in the context of modern computational tools.

Note that if we consider the works presented in Eyheramendy et Saad [13], [14] attached to the FEMJava platform for finite elements, the present approach for the isogeometric analysis could be naturally integrated in the previous developments.

2.1.3. Isogeometric Analysis implementations

In isogeometric analysis, researcher initially developed their own computational tools generally based on Matlab like applications.

In Vuong et al [17], the authors present a 2D tutorial Matlab code for IGA called ISOGAT. The key step of the method for trivial problem are outlined in a pedagogic fashion. The approach is valuable to understand the basis of IGA and offer a simple tool that may be enhanced to develop more complex formulations.

In Nguyen et al [18] a detailed study of the IGA is given and implementations aspects in Matlab are detailed. The incorporation of enrichment functions through the partition of unity is also discussed. The implementation in Matlab follows classical procedural algorithms for FEM/IGA: mesh generation, shape functions and spatial derivatives of shape functions, core of the code for forming stiffness matrices and loads, h- k- and p-refinement, post-processing, XFEM functions for enrichment and crack propagation. This package offers a pedagogic and clear implementation of the IGA method.

GeoPDEs (see Falco et al [19] and Vazque [223]) is a compatible Octave/Matlab package for the solution of PDEs using isogeometric analysis. Compared to the approaches presented above, this work is more generic and the code is obviously more structured. This research tool evolved rapidly since the initial version following the evolution of isogeometric analysis (see Falco et al [19]). From the structure point of view, the main characteristic of this last implementation are: the implementation is independent from the dimension (curves, surfaces, volumes), use of Octave classes instead of structures and enhanced object-orientedness of the code compared to the initial version (Falco et al [19]). Roughly speaking the code is organized around a few classes to manage geometries and fields:

- **geometry**: manage the characteristic of the NURBS geometry, is built from a file, compute the parametrization
- **msh_cartesian**: manage the quadrature rule for each parametric direction, is built from the knots vectors
- **sp_scalar:** manage a scalar field, manage the computation of the basis functions (corresponding to a discrete space), generate the boundary fields data
- **sp_vector:** manage a vector-valued field such as a velocity or a displacement

Several particular formulations are also available in the tool: div- and curl-conforming discretizations even in multi-patch domains, multi-patch classes that automatically manage strong c^0 continuity. Convenient functions to compute solution, evaluate errors are also provided. The whole framework makes GeoPDEs a convenient tool either for beginners in IGA or researchers who would like to evaluate easily their own formulations.

As the IGA is quite recent paradigm, some researchers develop extension to well established computational frameworks.

PetIGA is a code that aims at solving PDEs using IGA (see Dalcin et al [20]). The code is implemented in C (a few routines are written in Fortran) and is based on PETSc (see Balay et al [224]). PETSc is a collection of data structures and algorithm for the solution of PDEs including high performance computing capabilities (based on MPI model for communications). It provides both, an environment for modeling problems, and an environment allowing algorithms customization and extension. PetIGA reuses high performances capabilities of PETSc for extending it to IGA. It offers interfaces to manage IGA features in the context of PETSc. In Dalcin et al [20], the authors discuss first, the algorithmic choices made for PetIGA (periodic boundary conditions implementation, numerical differentiation, quadrature rules, tool IGAkit to manually handle CAD…), and second, technical aspects for high performance computing (adjacency graph computation, parallel partitioning and inter-process communication, assembling procedure. Challenging computations examples are provided. Scalability up to 4096 cores on a Navier-Stokes simulation is shown on a distributed memory system. A wide range of applications have been developed based on PetIGA: hyperelastic material in Bernal et al [225], a package for multi-field HPC for structure preserving B-splines spaces Vignal et al [226] and Sarmiento et al [227].

Commercial FEA tools provide in general the introduction of user defined capabilities. The use of established computational tools allows the partial or total reuse of existing capabilities: formulations, material formulation, CAD tools pre-treatment and post-treatment, solution algorithms, linear system solvers… In the case of isogeometric analysis, it seems clear that reuse of CAD capabilities is not straightforward because of the definition of computational domain based on splines family functions. In Duval et al [21], the authors propose a package called abqNURBS based on Abaqus. The implementation of the isogeometric analysis is done through, first, the routine UELMAT allowing both the introduction of user's elements and the use of existing constitutive laws, and second, the routine UEXTERNALDB to allow the definition of the NURBS (knots, control points, weights, connectivity tables). The pre-processing is done through GeoPDEs (see Vazque

[223]) and Rhino. Note that the tool Rhino provides only surface definitions whereas "solid" domain definition is needed for 3D computation. For post-processing purposes, the NURBS solution is projected on classical meshes allowing classical post-processing in Abaqus. This example illustrates the difficulty of classical FEA software to be extended to isogeometric analysis. The CAD integration for commercial tools is definitely not straightforward.

The development of object-oriented codes in C++ brings efficiency (computational efficiency) that cannot be expected in Matlab.

IFEM(see Kvarving [22]) is a general purpose object-oriented framework based on splines like functions to perform linear and nonlinear IGAs of solid problems. IFEM is developed in C++. It was initially developed with the aim of developing a suitable alternative to the traditional finite element approaches by utilizing spline finite elements. The main feature of IFEM is the core of IFEM is independent of the problem and that it widely uses external libraries: BLAS-subroutine in the implementation of elemental contribution computation, linear equation solvers (SuperLU, PETSc…). IFEM provides tools for linear and non-linear, stationary and dynamic time-domain analyses and eigenvalue analyses. Classical object-oriented design is used to handle CAD primitives (see Sorli et al [228]).

Geometry+Simulation Module (G+SMO) is an open-source C++ library for IGA (e.g. see Jüttler et al [229]). It is developed on a pure object-oriented concept. Note that G+SMO provides hierarchical splines.

The library igatools of Pauletti et al [230] is a general purpose object-oriented library for isogeometric analysis developed in C++. Advanced object-oriented and generic techniques are widely used in igatools. The design of igatools is closely related to the FEM. In Pauletti et al [230], the authors thoroughly compare the FEM and the IGA. The authors state that main similarities between FEM and IGA is the the Galerkin method with basis functions defined on a small support, the main difference being that the absence of master element and association of degrees of freedoms and geometries, and the treatment of boundary conditions. Even if this statement needs to be qualified, the global analysis remains pertinent. The first feature of igatools is its independence with regard to the dimension of the domain. The main objects of the code are:

- **ReferenceSpace** (BSplineSpace and NURBSSpace): for the reference spaces, definition of the parametric functions
- **PhysicalSpace**: contains the geometry definition and the transformation between the ReferenceSpace
- **Geometry**: the geometry defined by the transformation of the reference domain using a Mapping
- **Mapping**: contains a geometry and manages the transformation of the reference domain; it mays be analytical or isogeometric
- **PushForward**: the operator manages the construction of the physical space from the reference space
- **ElementIterator**: this is the mechanism to compute and access all kind of quantities for a collection of object
- **Other objects**: objects to manage global quantities are provided: fields, operators and linear algebra tools.

The library igatools provides:

a) a cache mechanism to precompute and store quantities needed in the computations

b) a wide use of templates to achieve genericity

c) input/output facilities based on XML specifications

d) interaction with CAD solid 3D CAD systems

Such as GeoPDEs, igatools offers a good compromise for testing new ideas in the research community. It is important to note that none of these tools have been initially designed for implementing constitutive law modeling which is an important feature in computational mechanics.

An object-oriented implementation of the T-spline based isogeometric analysis is presented in Rypl et Patzák [23]. This work is an IGA extension of OOFEM Patzák et Bittnar [231]

which was directly inspired from Pèlerin et Zimmermann [177] and [10]. The implementation of the IGA element is done in similar way than the one of the FEM due to the similitude in the management of the DOFS in both IGA and FEM. A class IGAElement is inherited from base class Element for basic operations: list of nodes maintenance, boundary conditions management, integration rules, access to interpolation, material model, abstract services for the evaluation of elemental contributions (e.g. stiffness matrix, load vector, etc.) This enables a similar treatment for both IGA and FEM elements types thanks to inheritance and polymorphism. The integration rule, which is held on a portion of the parametric space is made in a class IGA_IntegrationElement. This class has an attribute knotSpan that enables to control the integration over the elements (in the IGA sense) of the patch. Thus, the integration scheme consists in an outer loop over the elements, and an inner loop over the integration points. Special classes support the definition of splines and NURBS interpolation definitions. In this implementation, the user can implement either new FEM or new IGA elements. Note that here, even for similar forms of variational formulations (same form of elemental contribution) both implementations for IGA and FEM are needed.

To go further, a wide range of packages such as the Python library SfePy (see Cimrman [232], [233]) or the C++ library MFEM (see [234]) includes capabilities for isogeometric analysis. More specific implementations are developed such as a concurrent integration algorithm on GPU for IGA (see Karatarakis et al [235] and Woniak [236]).

Most of the packages presented here are focused on the variational framework of IGA and the design of patches. Just a few of them explicitly take into account constitutive modeling as expected in computational mechanics. Let cite for example: OOFEM and its IGA extension in Rypl et Patzák [23], abqNURBS from Duval et al [21] which is based on Abaqus. For large scale development of FEM or IGA, object-oriented design remains today an important feature. Most of the packages discussed here implement this paradigm: OOFEM and its IGA extension in Rypl et Patzák [23], library igatools in Pauletti et al [230], Geometry+Simulation Module (G+SMO) in Jüttler et al [229], GeoPDEs in Falco et al [19] and Vazque, etc. In the approach we propose in this work, we realize an extension of a Java code for the FEM to the IGA. The aim is to take benefit of all the framework in the context

of IGA. Another important feature is to take advantage of the similarities of the variational forms between FEM and IGA to reduce developments for same problems.

2.2. FEMJava: A Java Finite Element Code

In this section, the base code for the implementation of isogeometric analysis method is presented. It is an object-oriented general purpose FEM program written in Java: FEMJava.

2.2.1. Package exploration

The code is classically organized in thematic packages grouped into the FEMJava package as shown in Figure 15. The core of the code includes packages *algorithm, fem, field, quadrature, geometry, material, mesh* and *imposedvalues*. The data are mainly structured around both, the definition of the geometry of the computational domain (package *geometry*) and the definition of the different fields involved in the formulation of the problem (package *field*). The definition of a field is the one of a discrete field. Data may be supported on either nodes or integration points (e.g. Gauss points as often used in finite elements). Numerical integration rules, i.e. roughly speaking a list of points (1D points, 2D points,…) are defined in the package *quadrature*. The package *fem* contains the main classes corresponding to the mathematical formulation (definition of the elemental contributions derived from the physical problem for a given integration algorithm). It contains classes to organize the elemental matrices evaluation and the basic kinematic discrete operators to compute them. The package *algorithm* encompasses the classes needed to manage the resolution algorithms such as: time stepping, iterative schemes, time integration, etc. The package *material* holds the classes to locally integrate constitutive laws. Thus, the global solution procedure is provided by the classes in package *algorithm* and the local algorithms involved in the elemental contribution evaluation manage calls to local procedure in which constitutive laws are engaged. The package *mesh* contains classes to manage mesh generation. Only basic algorithms are presented here. The boundary conditions are managed in the package *imposedvalues*.

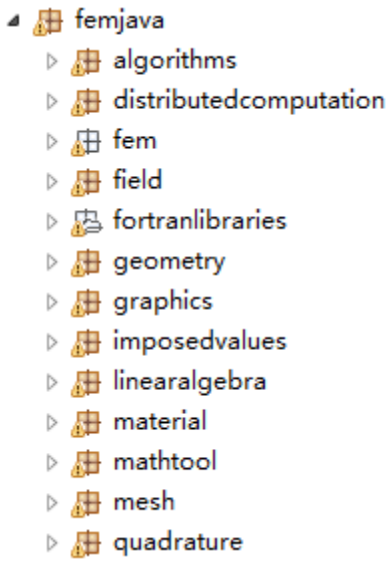


Figure 15. FEMJava: first level packages exploration

There are some additional packages to supply tools to complete the framework:

a) package *fortranlibrairies*: BLAS libraries
b) package *graphics*: a complete GUI (Graphical User Interface) pre-processing and post-processing capabilities)
c) package *linearalgebra*: linear system solution, assembling elemental matrices.

This brief introduction to FEMJava packages permits the reader to realize the areas covered by the code. In the following, we briefly discuss the principal packages and classes of the framework as in Figure 16 and their interactions focusing on the variational formulations, discretization schemes, material and constitutive integration, global solution algorithms.

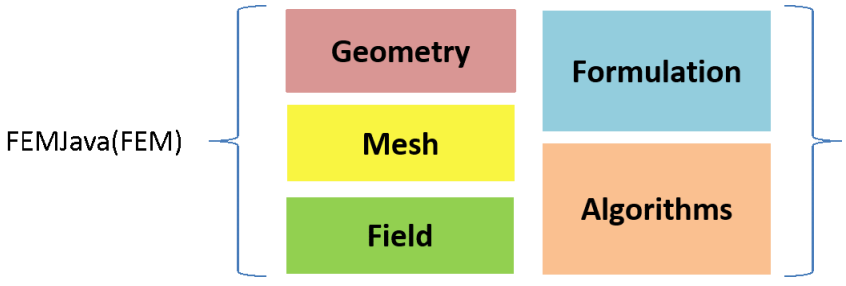
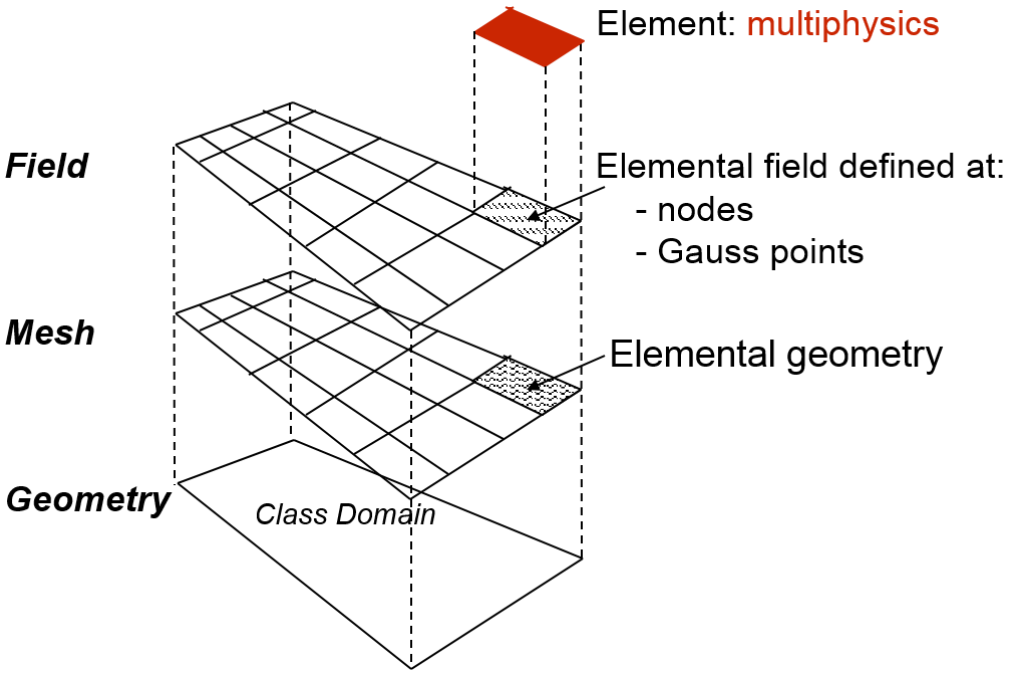


Figure 16. FEMJava: main thematic packages

2.2.2. Data structures for Multiphysics

The code FEMJava was originally designed for treating problems in which multi-physical phenomenon and their interactions take place (see Eyheramendy [189] and [237]). This objective demands careful attention to design data structures the management of multi-fields formulations, in the context of local and global level interactions. In this section, the mainly aspects upon the data structure of framework FEMJava is to be presented.

- **A brief overview of FEMJava structuration**



- Figure 17. FEMJava: multiphysics data structure

Firstly, the domain on which a boundary value problem stated is abstracted by the class **Domain**. As a physical domain for industrial applications could be fairly complex, an instance of class **Domain** holds typically multiple instances of class **Subdomain**. As a result, the class **Subdomain** conforms to the concept of computational domain at the level of geometry. These two anther layers named respectively “mesh” and “field”. The mesh layer defined in class **Mesh**, represents the discretized geometric domain in the sense of finite elements like concept (domain decomposed in a set elemental volumes representing

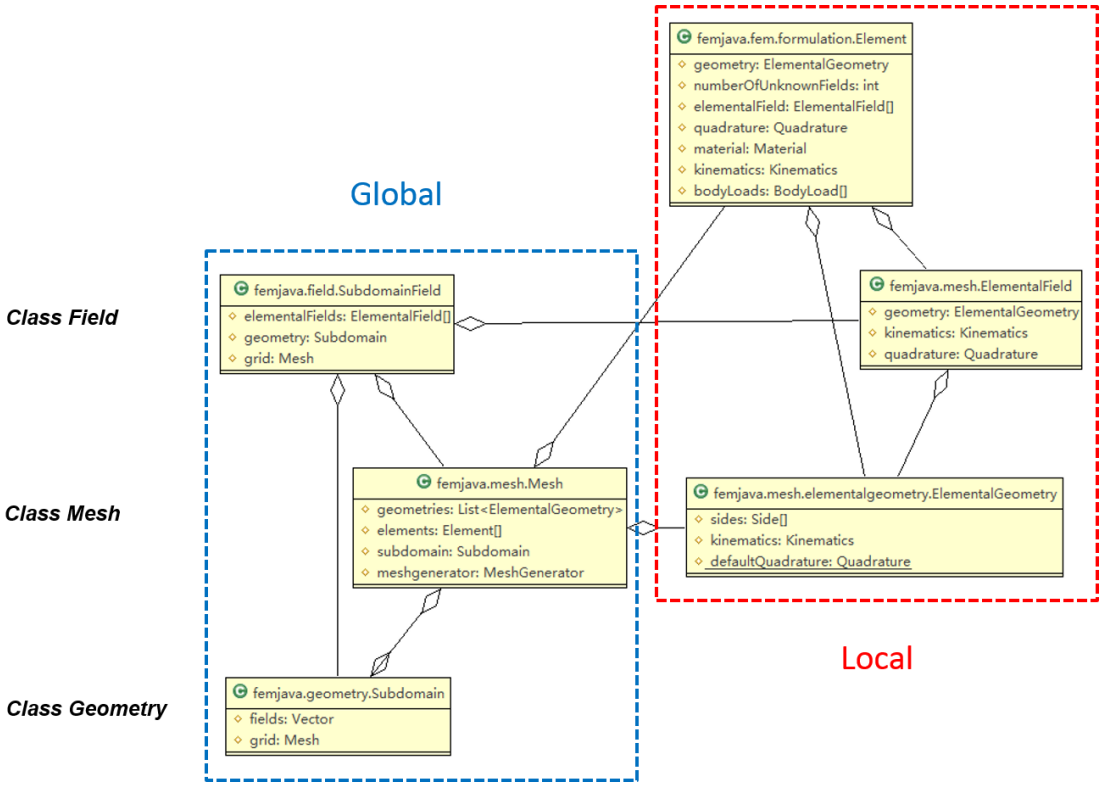


Figure 18. FEMJava: UML class diagram for data structure

the global physical domain). Thus, all the discrete elemental geometries are characterized by subclasses of class **ElementalGeometry**. The key idea for structuring the framework lies on classes **Field** and **SubdomainField** which represent the mathematical concept of discrete field (physical or nonphysical quantity defined at a finished number of points in a physical domain). Their respective supports are classes **Domain** and **Subdomain**. The global and local architecture of the concept of field, mesh and domain is shown in Figure 17. The geometry of the domain is the basic support of physical or mathematical entities. The mesh is built on the geometry (mesh level in Figure 17). The mesh is a set of nodes associated to a set of elemental geometries (triangles, quadrangles, hexahedrons…). The fields and their associated nodes to support scalar, vector or tensor values are built on the stencil the mesh and its elemental geometries. The definition of the field at the level if the subdomain is a set of elemental fields. The definition of the elemental field embeds the local concept of interpolation in the case of a nodal field. The UML class diagram corresponding to the corresponding data structure is given in Figure 18. At the global level, the object “subdomain field” knows the subdomain on which he is defined and know the

discretization on which it is built. At the local level, the basic definition of the subdomain field is the elemental field which processes either its nodes or Gauss points (depending on the type of field, defined at nodes or integration points). The element field knows the elemental geometry from which it has been created and eventually a quadrature used during computations. Therein, the global and local perspectives and their relation have been clearly clarified. The object Element is the key of the local definition of multiphysics. The class **Element** main attribute is a set of instances of **ElementalField**. This allows the taking into account of the coupling between the different fields at the local local of elemental contributions computations.

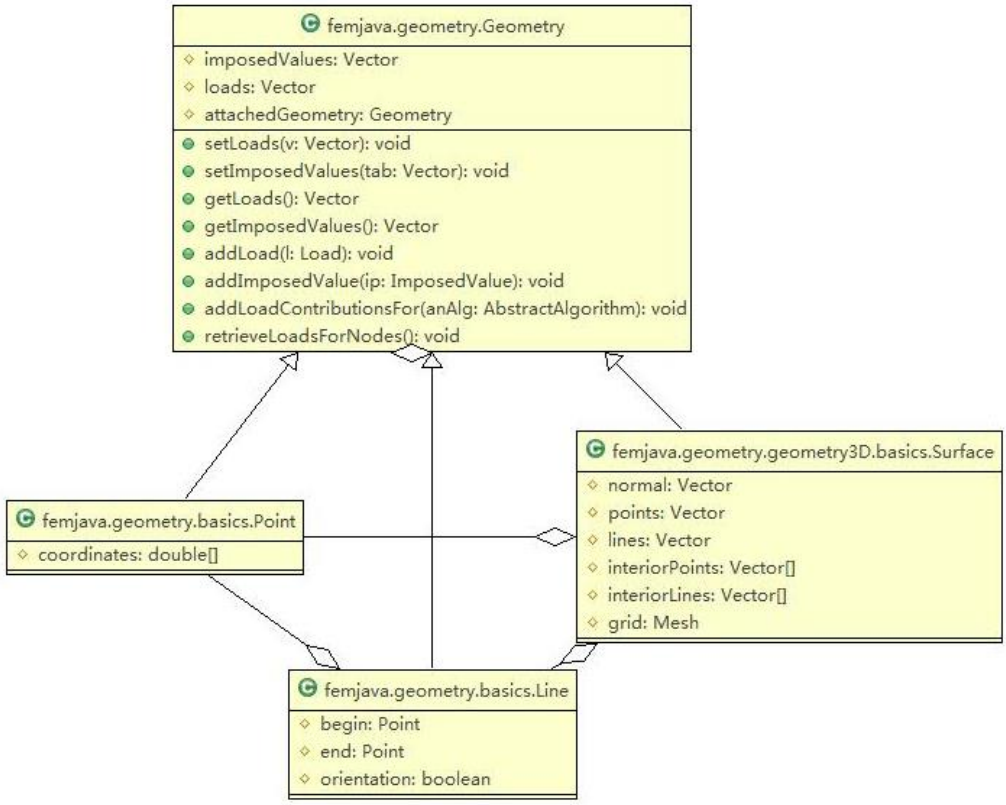


Figure 19. FEMJava: primary geometry classes

- **Class Geometry and its subclasses**

With some classes for primary geometries such as **Point**, **Line**, **Surface** and **Volume**, it allows to describe the primitive geometric models in FEMJava. The subdomain is composed of these geometrical entities describing its boundaries (interior and exterior

faces). These boundary geometries support the boundary conditions. These primitives share some common operations and properties that they inherent from the abstract superclass **Geometry**. The corresponding UML class diagram is shown in Figure 19. The surface (2D model) is made of lines and points. The main functionality of the superclass **Geometry** is the management of boundary conditions.

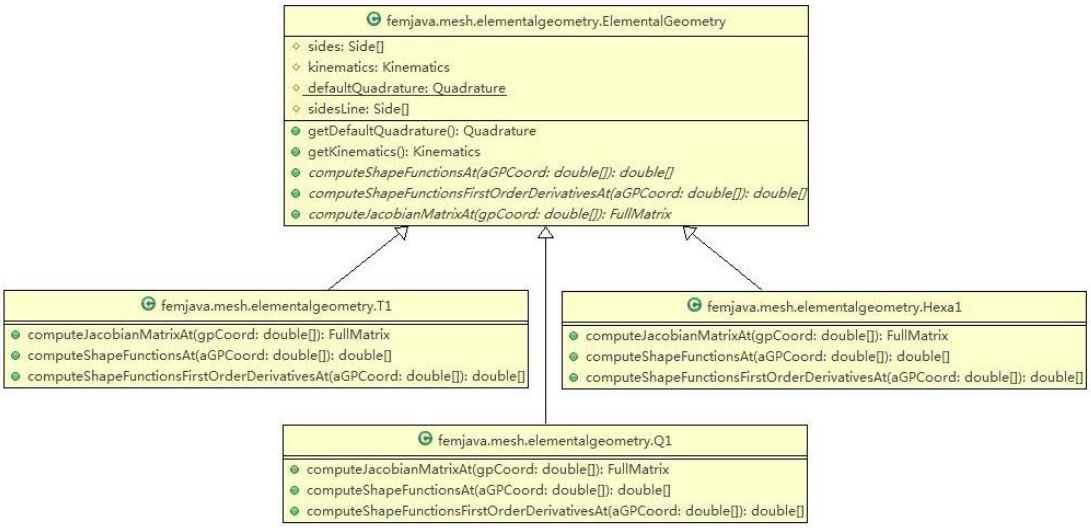


Figure 20. FEMJava: elemental geometry classes

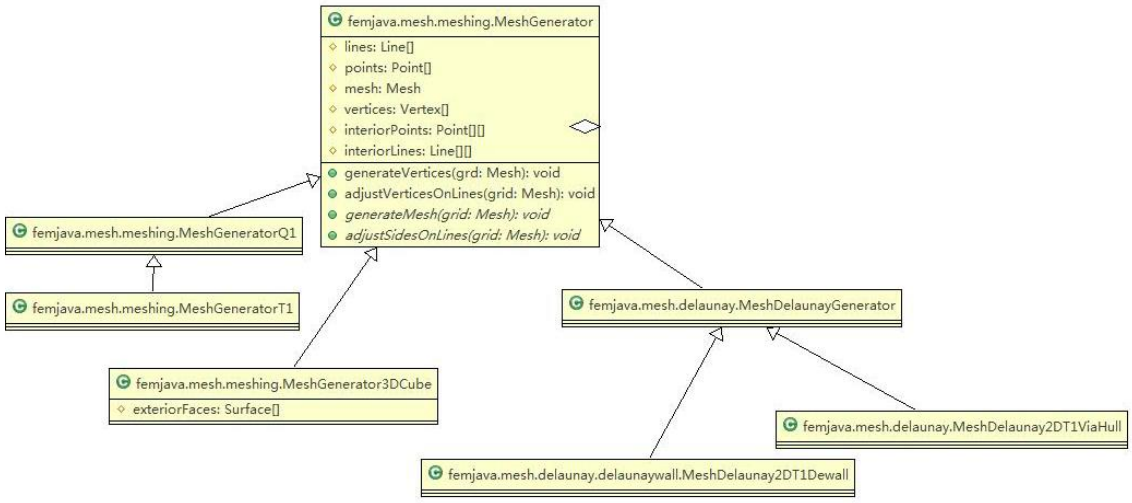


Figure 21. FEMJava: examples of mesh generator classes

- **Mesh and meshing management**

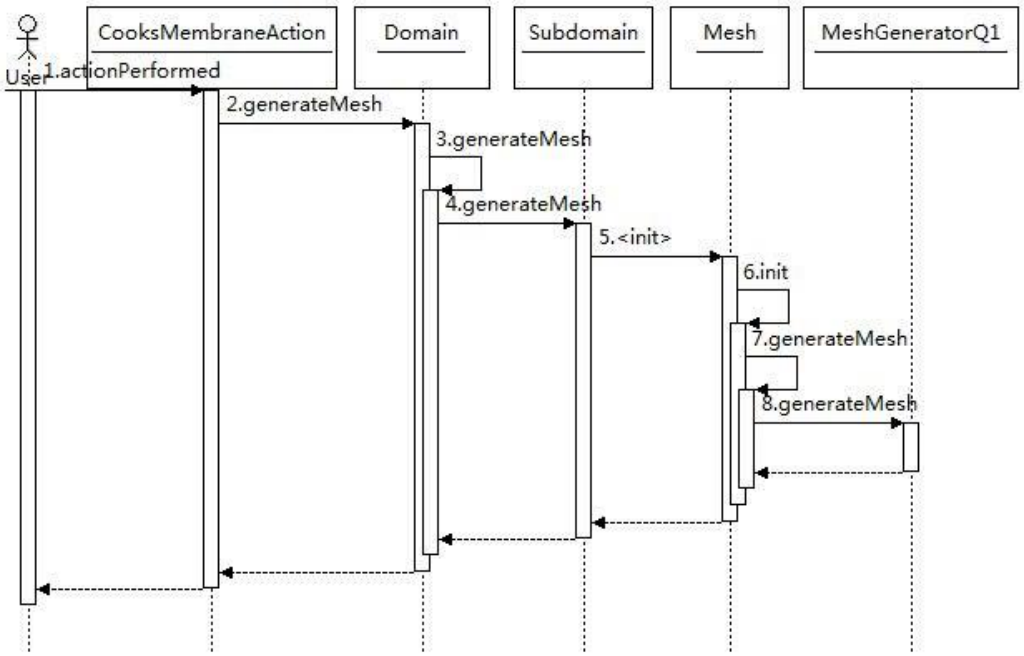


Figure 22. FEMJava: mesh generation sequence

The mesh can be considered as a set of instances of class **ElementalGeometry**. As shown in Figure 20, every type element inherits form class **ElementalGeometry**, e.g. class **Q1** for linear quadrilateral plane element, class **T1** for a linear triangular plane element and class **Hexa1** for a linear hexahedral element. The abstract superclass **ElementalGeomery** is mainly characterized by an array of vertices and an array of sides whenever needed and declares several abstract methods. The parametric change of variable between local and global coordinates axis is managed at the level of elemental geometry defined in the computation of the Jacobian matrix. These methods are implemented in all subclasses that implement the particular behavior for shape function computations and Jacobian matrix computation:

a) computeShapeFunctionsAt(double[])

b) computeShapeFunctionsFirstOrderDerivativesAt(double[])

c) computeJacobianMatrixAt(double[]).

The second feature of the elemental geometry is to have access to the set of discrete forms of partial differential operators through the attribute kinematics, instance of class **Kinematics**. At last, the elemental geometry defines the default quadrature scheme (in general the minimum Gauss points to achieve exact integration in each direction).

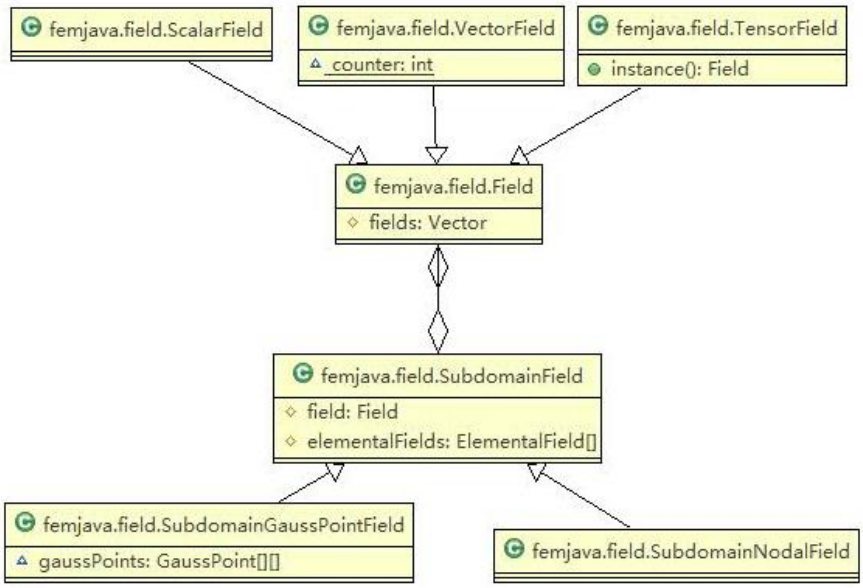


Figure 23. FEMJava: Field and Subdomain field classes

The meshing procedure consists mainly in building a set of vertices and a set elemental geometries. It is performed on the computational domain. The code accomplishes mesh generation through a class subclass of the abstract superclass **MeshGenerator**, as shown in Figure 21. Each specific subclass of **MeshGenerator** is implemented for a particular elemental geometry. Some frequently used mesh generation algorithms has been implemented in the code, such as the Delaunay triangulation, which make it possible to mesh irregular geometries.

The process for generating linear quadrilateral element (here for generating linear 2D quadrangular elements in class **MeshGeneratorQ1**), an UML sequence diagram is given in Figure 22. The command of meshing comes from **Domain**, **Subdomain** to **Mesh**. At last, it is actually carried out by the method inside a subclass of **MeshGenerator**.

- **Fields management**

Finally, for the field level, the abstract class **Field** lies on the discretized domain. Classes **ScalarField**, **VectorField** and **TensorFiled** inherit from class **Field** and implement for

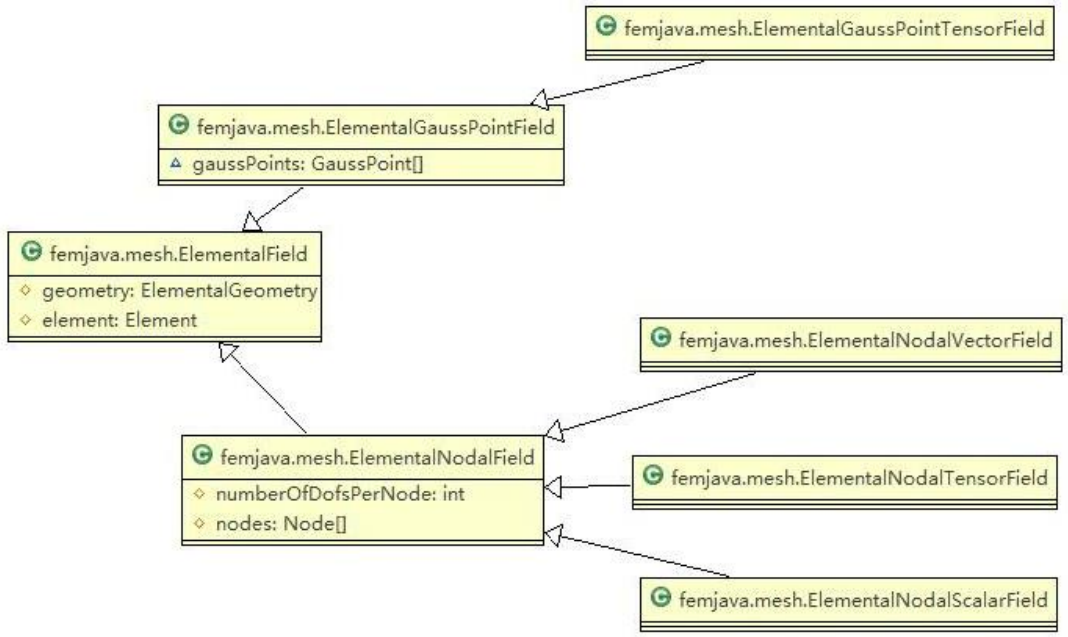


Figure 24. FEMJava: elemental field classes

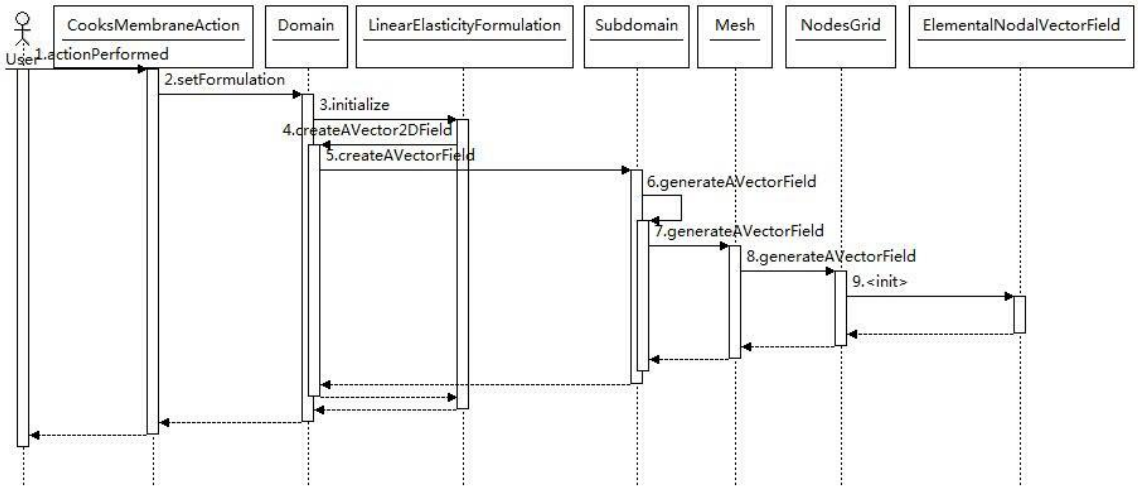


Figure 25. FEMJava: Field generation sequence

specific fields respectively: scalar, vector and tensor fields. Such as the domain is defined by a set of subdomains, the field has equivalent definition on each subdomain, class **SubdomainField**. The class subdomain field is specialized into a field defined either at Gauss points or at nodes. Nodes and elemental fields are stored at the level of the subdomain field, either in classes **SubdomainNodalField** or **SubdmainGaussPointField**. This UML class diagram is shown in Figure 23.

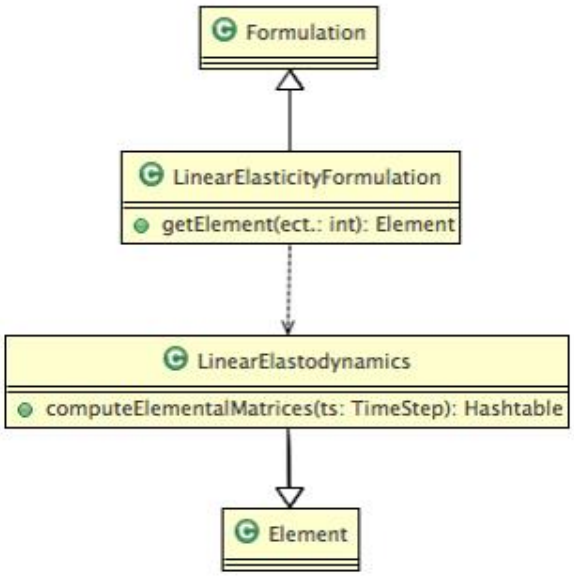


Figure 26. FEMJava: UML class diagram of formulation and algorithm for linear elasticity

Each field has a local definition base on an elemental geometry. The basic entity that defines a field is the elemental field, instance of a subclass of class **ElementalField** (see Figure 24 for the UML class diagram). The interpolation scheme is managed at this level for both, the field defined at Gauss points and the nodal field. Note that an extrapolation interpolation is needed to recover nodal values in the case of a field defined at Gauss point. Class **ElementalField** represents this local field. The elemental field is based on an elemental geometry (class **ElementalGeometry**). The parametric change of variable between local and global coordinates axis is managed at the level of the elemental geometry. Local data, either nodal values or values at Gauss points are managed in subclasses of **ElementalField**, i.e. repectively classes **ElementalNodalField** and **ElementalGaussPointField**. The class **ElementalNodalField** contains a set of nodes (instances of class Node that stores the unknowns corresponding to the field), the class

ElementalGaussPointField a set of Gauss points storing the values of the field at the corresponding local coordinates. Those two classes correspond to global classes **SubdomainNodalField** and **SubdomainGaussPointField** at the level of the subdomain.

2.2.3. Formulations and Algorithms

As shown in Figure 18, the class **Element** contains all the objects needed to compute elemental contributions: instances of elemental fields (support multiple fields computations), an instance of **Quadrature** (numerical integration), the instance of elemental geometry (parametric change of coordinates axis), an instance of material (integration of the constitutive law described in the following), an instance of class **Kinematics** (manage the discrete forms of differential operators applied to a given field (plane strain, plane stress, 3D,…). Typical formulations are implemented in subclasses of class **Element** (see Figure 3). As the local definition of the variational formulation is done in class **Element**, the global definition of the formulation, i.e. mainly the definition of the fields involved in the multi-fields multiphysics formulation is done in class **Formulation** (see Figure 26). The subclasses of **Formulation** implement specialized formulations. E.g. in Figure 26) in class **LinearElasticityFormulation** that implement classical linear elasticity, the formulation has one unknown field, the displacement field which is a vector. The consequence of it is an automatic construction of the nodal field based on a given mesh. The class **LinearElasticity**, inheriting from class **Element**, defines the specialized behavior to compute the finite elements elemental contributions. To enforce consistency between the global formulation and the elemental definition of the formulation, the class **LinearElasticity** is defined as an inner class in class **LinearElasticityFormulation**. It requires the programmer to consistently program the elemental contribution by only using the elemental fields originally provided by the global fields unknowns defined in the formulation.

2.3. An object-oriented implementation of NURBS in Java

In the section, we propose an object-oriented implementation of NURBS. As the fundamental of analysis code (initially designed for finite element method) being realized in Java, here we choose to implement the previously introduced geometric description library of NURBS with the same language.

2.3.1. Patch classes

A geometry entity of NURBS is modeled by the class **Patch** in our implementation, as shown in Figure 27. This class being designed for arbitrary parametric dimension geometry stays abstract. For these typical cases, three subclasses for problems of 1D, 2D and 3D have been defined: **Patch1D**, **Patch2D**, **Patch3D**. It has no difficulty to create another specific subclass by extending the superclass, for instance we may need a **Patch4D** for a spatially three-dimensional problem with space-time method.

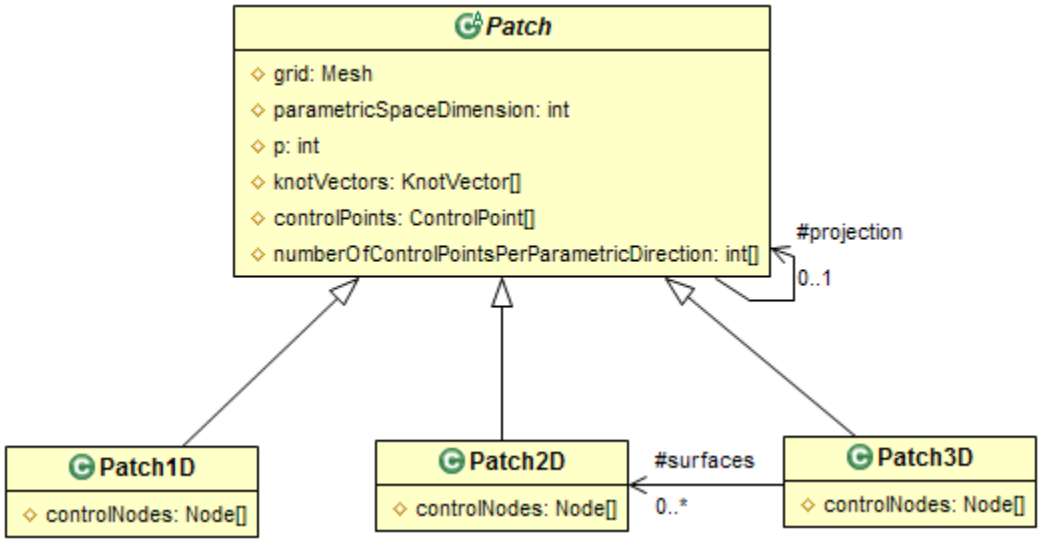


Figure 27. Java NURBS library: Patch classes

The abstract class **Patch** has principally four attributes:

- **parametricSpaceDimension**: the dimension of geometry's parametric space, curves, surfaces and volumes correspondent to the values 1, 2 and 3 respectively.
- **p**: the degree of basis functions in each direction.

- **knotVectors**: the array of class **KnotVector**'s instances. Its size equals imperatively to the value of **parametricSpaceDimension**
- **controlPoints**: the array of class **ControlPoint**'s instances. The number of control points is determined automatically by its knot vectors and degree.

These two last attributes: knot vector and control point are encapsulated as classes to guarantee a right way of data getting and setting as schemed in Figure 28,.

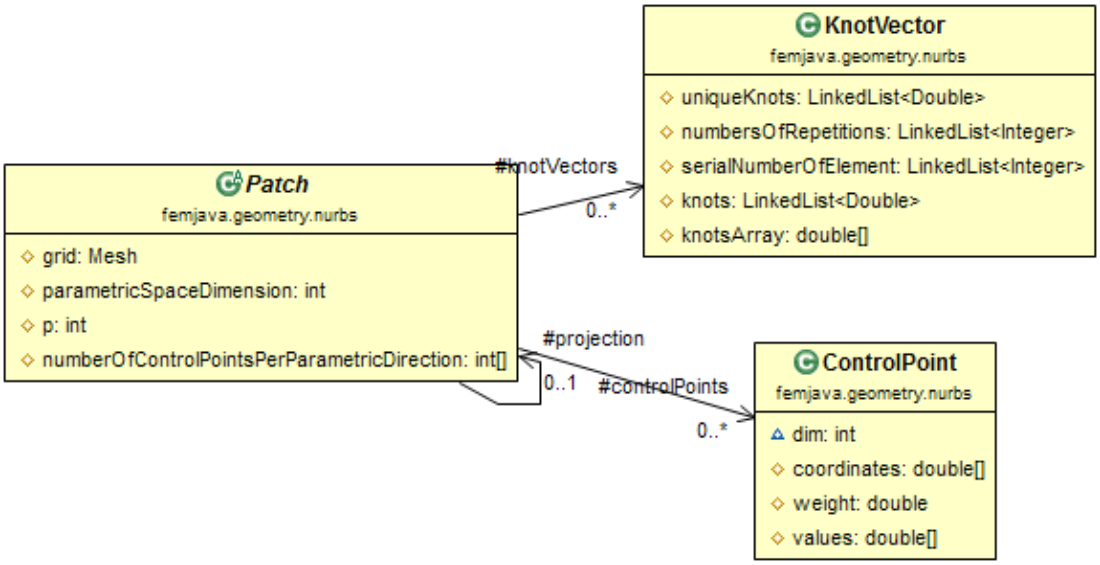


Figure 28. Java NURBS library: KnotVector and ControlPoint

Knot vectors

The knots in a knot vector are not stored directly as an array of float numbers. However, an instance of **KnotVector** can accept this form as its constructor's parameter. The initial array knots, will be parsed and stored principally in two float number lists, **uniqueKnots** and **numberOfRepetitions.** Furthermore, another integer list **serialNumberOfElement** related to the knot spans with the intention of facilitating their location are retrieved.

- **uniqueKnots**: a double list contains the unique knot values without repetitions.
- **numberOfRepetitions**: an integer list stores the multiplicities of each unique knot value.

- **serialNumberOfElement**: an integer list stores the serial numbers of every no vanishing knot span.

For example, a knot vector initialized by a double vector:

[0.0 0.0 0.0 0.5 0.5 1.0 2.0 3.0 3.0 3.0]

after the parsing process commanded inside the constructor, the attributes find their values as:

uniqueKnots: [0.0 0.5 1.0 2.0 3.0]

numberOfRepetitions: [3 2 1 1 3]

serialNumberOfElement: [3 5 6 7]

The storage way implemented allows to simplify the indexing for knots and knots spans. More importantly, it avoids the machine precision problems caused by frequently operations on knots.

Control Point

The class **ControlPoint** has been implemented in a more intuitive way. As presented in Figure 28, there are only four attributes for a control point.

- **dim**: an integer denotes the physical dimension of control point decides the size of its physical coordinates.
- **coordinates**: a double array physical coordinates of control point
- **weight**: a double represent control point's weight.
- **value**: a double array stores the control variables of control point.

Many NURBS based geometry's algorithm previously introduced are achieve by the operations on control point, for example the multiplication to a coefficient, addition or subtraction between two control points. All these processes demands to operate the control point's coordinates, weight and also the arbitrarily dimensional control variables. The encapsulation prevents the inconsistent operations of these variables.

2.3.2. Implementation of algorithms for arbitrary NURBS

NURBS based geometry's necessary and important operations such as evaluations of basis functions and their derivatives and refinement algorithms are naturally considered as methods of the abstract class **Patch.** All these algorithms have been implemented in this abstract superclass. On account of the intrinsic tensor product structure of high order NURBS geometries, it is possible to program these methods that can work directly for arbitrary parametric or physical dimensions. In fact, this feature is the very novel and challenging point of this implementation.

Reviewing the algorithms of refinement introduced in 1.1.2, it is found that a modification of parametric space (e.g. insert or remove a value in a knot vector) requests recalculating the associated control points. In case of parametrically multi-dimensional NURBS, the same operation need to be performed on control points indexed by all the other parametric direction.

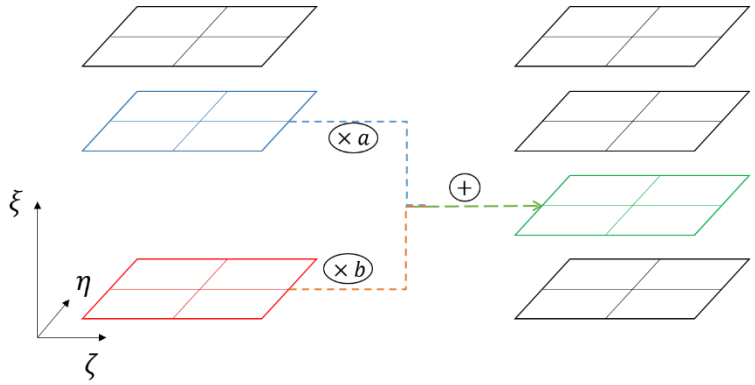


Figure 29. Knot insertion scheme for control point mesh

Taking the knot insertion for a parametrically three-dimensional NURBS as an example, see the scheme in Figure 29. A new knot introduced in ξ knot vector demands to adjoin a new two-dimensional control point net that indicated by meshed parallelogram. When the basis function is linear $p = 1$, only these two adjacent control point net (blue and red) are concerned. Thus, once we get the coefficients calculated from the neighboring knot values, the new one order lower control net (the green parallelogram) inferred from the summation of these two weighted nets.

The previous example shows that the operation on one order lower control net is frequently performed during knot insertion, so as these other refinement algorithms knot removal, degree elevation and degree reduction. We realize this kind of operation in FEMJava through a single index to multiple index numbering switching for control points, and vice visa.

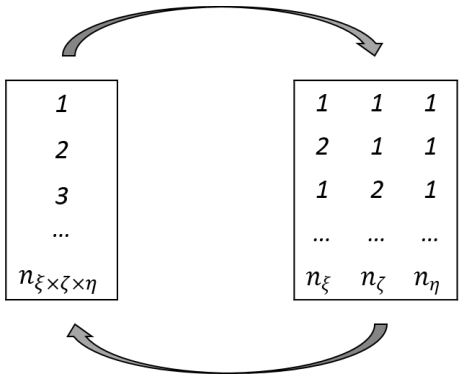


Figure 30. Single index to multiple index numbering switching

Considering once again a parametrically three-dimensional volume based on NURBS, the control point for each parametric direction are set to be $[n_\xi,\ n_\zeta,\ n_\eta]$. Thus, it has entirely $n_{\xi\times\zeta\times\eta}$ control points. The numbering switching can be easily made with a convention that control point's multiple index counts firstly in direction ξ from 1 to n_ξ, then performs a carry bit for the second direction ζ, and so on.

The important methods such as evaluation of the multi-dimensional basis functions and their partial derivatives, refinement algorithm knot insertion, knot removal and degree elevation are all implemented once for all in the abstract super class **Patch** with the preceding presented techniques. In the following section, this minor library for NURBS geometry will contribute to integrate isogeometric to FEMJava.

2.4. Integration of Isogeometric Analysis

From object oriented programming point of view, the isogeometric Galerkin method's implementation has very limited difference from the traditional finite element method.

Therefore, in this section, we demonstrate that it is straightforward to integrate IGA to the previously introduced code FEMJava which is initially designed for FEM.

These two main differences between IGA and FEM implementation in the framework of FEMJava lie in the data structure of discretization and the field creation basing on discretized mesh. The FEM has two space: physical space where the problem is defined and the reference space in which the shape function and numerical integration are evaluated. But for NURBS based isogeometric Galerkin method, we need a third parametric space between the reference space and physical space. The isoparametric basis functions stay in this parametric space. And all the numerical integration for the terms of variational formulations still takes place in reference space since we work it with the Gaussian quadrature rules. The FEMJava framework's multiple levels (geometry, mesh and field) data structure, and its OOP characteristics (encapsulation, inheritance and polymorphism) make it possible to bridge the divides without radical changes in code architecture.

In the following, we firstly present two element geometries specified for NURBS based IGA, and their extraction process from NURBS patches which are equivalent to subdomain of FEM. Then, the field creation based the former discretization is explained. At the last, a performance optimization strategy has been implemented to accelerate the computing.

2.4.1. NURBS elemental geometry

In Figure 17, the data structure for FEM in the code FEMJava has been schemed at three levels, geometry, mesh and field. To adapt the NURBS based isogeometric method to the same architecture, the correspondence relationship for a field based on NURBS discretization is shown in Figure 31.

The most evident difference is observed that for NURBS, there is no more mesh level, since that a NURBS patch has an intrinsic discretized structure retrieved from its tensor product parametric space that is defined by the knot vectors. These knot vectors partition the parametric, and further discretize the geometry. The conception equivalent to an element of FEM rises naturally with this intrinsic discretization. In FEMJava, an element

of NURBS based IGA consists of a partitioned parametric section and these control points whose basis function do not vanish in this section. Thus, we can tell that there are always $(p+1)^{n_{pd}}$, where p denotes the basis function's degree and n_{pd} represents the parametric dimension. It is noted that in this implementation, we have considered only the general case that the NUBRS has the same degree in each parametric direction.

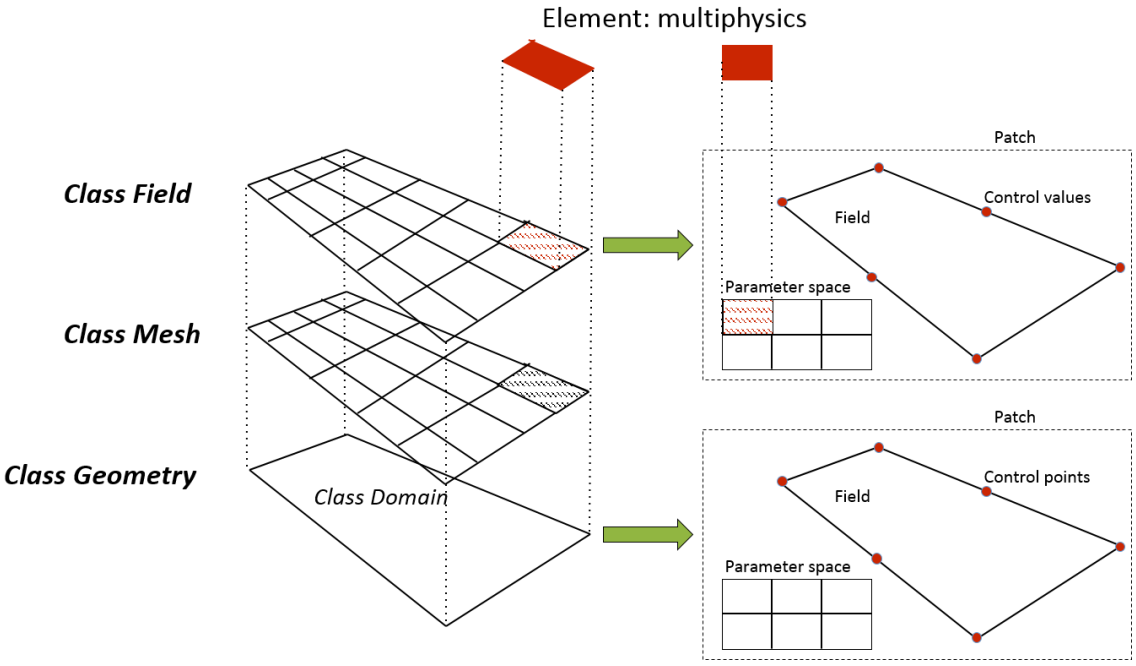


Figure 31. Integration of IGA: Multiphysics data structure.

Then, a patch entity that contains a parametric space and the control net is regarded as the geometry class to support the boundary conditions such as imposed values and loads. With the intention to represent a field created from this patch, every control point holds a variable for the field value. It makes a control net with control variables as schemes in Figure 31 on the right.

Figure 32 displays these UML diagram of those classes that serve the purpose to carry out the previously introduced data structure in an OOP code like Java for our implementations. Considering a 2D patch of NURBS as an example, the elemental geometry has been named by **Q1NURBS** because its parametric construction is as simply as the linear quadrilateral element of FEM. This subclass of **ElementalGeometry** support the **ElemetnalField** which constitutes the **SubdomainField** in the same way as the FEM of FEMJava that has been

depicted in Figure 18. Compared to the diagram in Figure 18, here we have one more global class: **Patch**. This NURBS class that has been presented in detail in 2.3 presents itself as the geometry support for other classes global classes of level geometry, mesh and field.

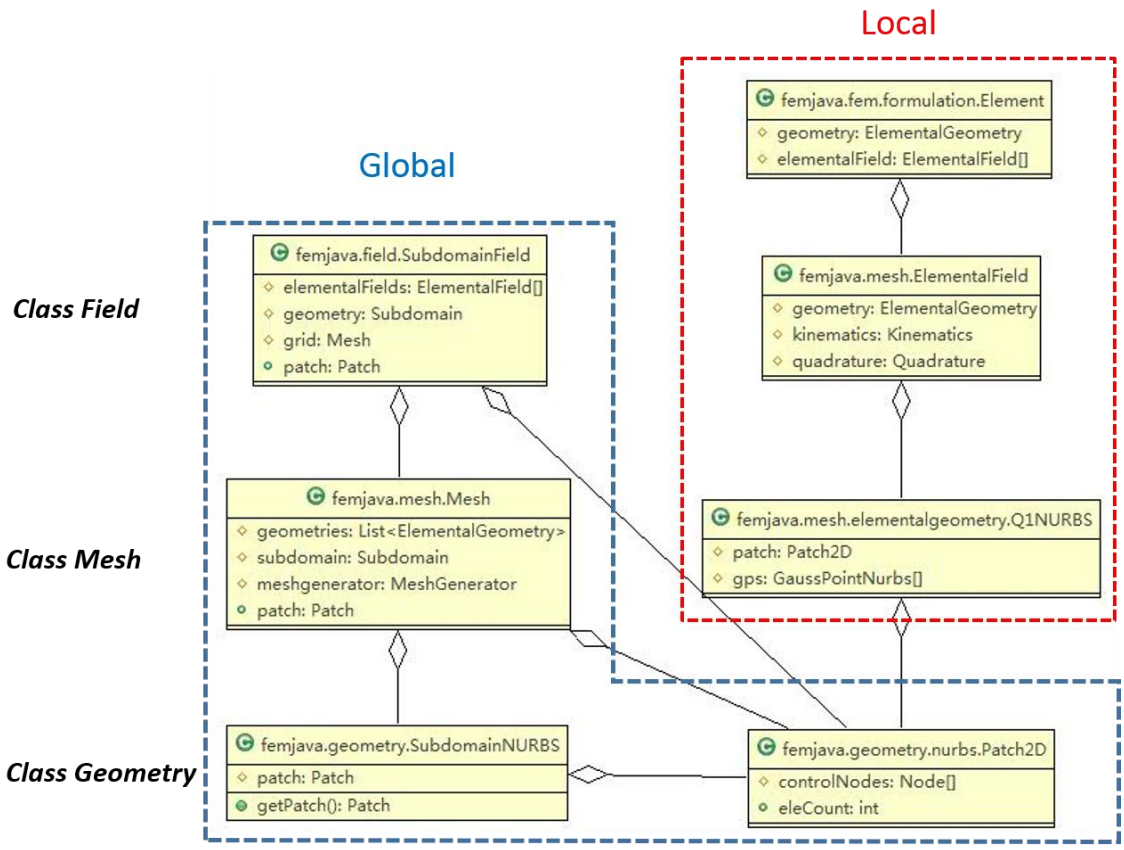


Figure 32. Integration of IGA: UML class diagram for data structure.

Lastly, we give the UML diagram to demonstrate the inheritance of **Q1NURBS** and **H1NURBS** from **ElementalGeometry**, and their aggregations to **Patch2D** and **Patch3D**. The **H1NURBS** for 3D NURBS element is so named since it has a linear hexagon form in parametric space.

2.4.2. Performance optimization

The computational costs of NURBS based element are usually much higher than the commonly used linear and quadratic element of FEM. We can list several raisons here: NURBS element usually has high degree basis function; recursive definition of basis

function is computationally expensive; the tensor product structure NURBS computation rapidely grows with dimension.

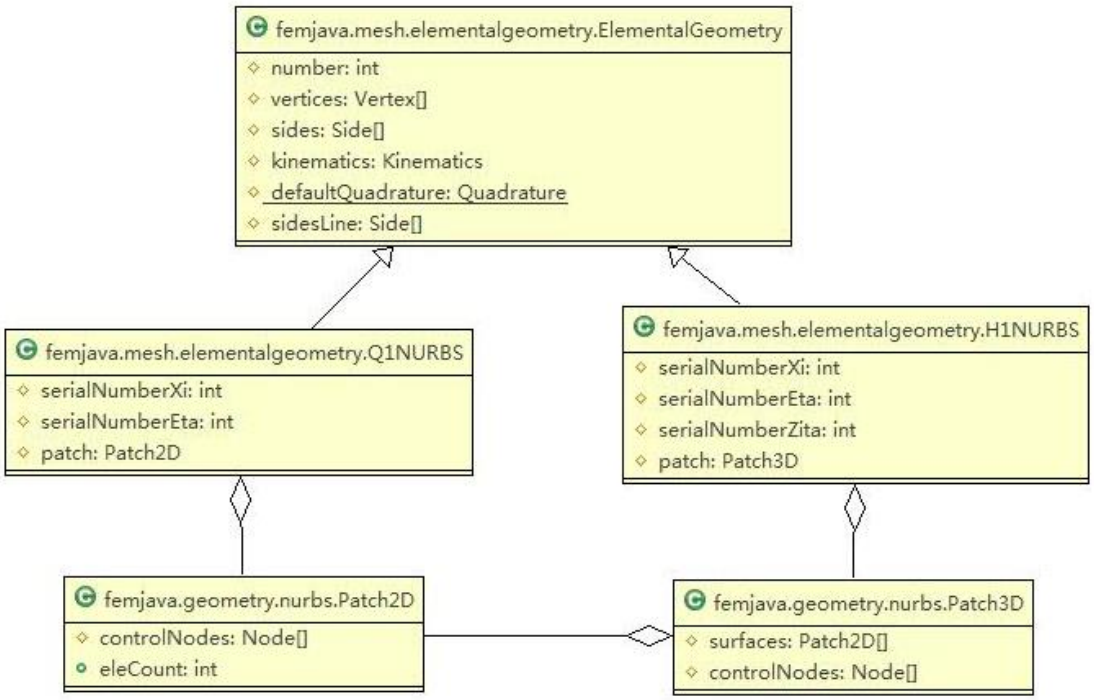


Figure 33. Integration of IGA: NURBS element classes and their patch classes.

In our implementation, we have proposed a strategy to sacrifice some memory space to improve the computational performances. This optimization scheme has been realized at the level of the integration point in a new class called **GaussPointNurbs** as shown in Figure 34.

```
/**
 * basisFuncs[i][j][k]
 * Index i indicates the direction of parametric space:(xi,eta,zeta)
 * Index j indicates the derivation order: 0<=j<=p
 * Index k indicates the offset of the element in the specific direction
 */
public double[][][] basisFuncs;
```

During the calculation, once these one-dimensional basis functions their and derivations have been evaluated at a numerical integration point. An instance of class **GaussPointNurbs** is created to store all these results. Thus, when these values need to be

computed another time for the point (if a second iteration or a second time step exits), these values can be directly reuse. With this performance optimization, it allows to solve a linear elasticity problem within about 8 times less computational time that the first version of code.

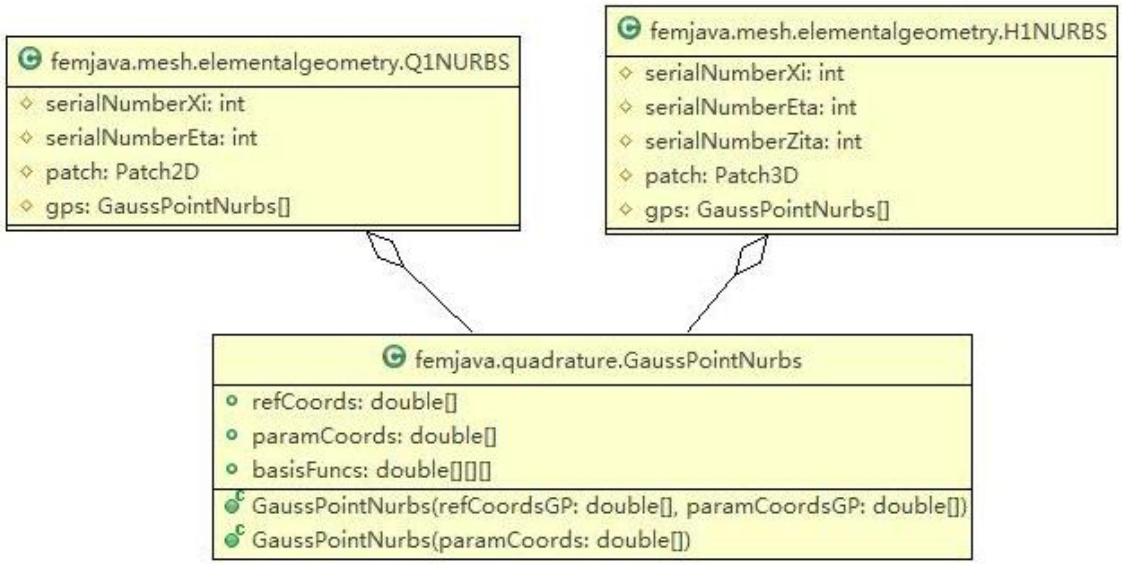


Figure 34. integration of IGA: performance optimization class.

Note that this optimization serves only to alleviate the costly computation of recursively defined basis functions for each parametric direction. It does not handle the complexity coming from the tensor product structure for multi-dimensional NURBS. Thus, the code efficiency is seriously reduced for 3D problems. A solution inspired by the algebraic optimization computation done in FEM could store as well theses multi-dimensional basis functions and their derivatives at the integration points.

2.4.3. 2D Hole plate

We present the classical 2D example (see e.g. Cotterell et Hughes [72]) to validate the code for linear elasticity problem. This problem has an exact solution for an infinite plate with

a circular hole under constant in-plane tension at infinity. In this case, a finite quarter plate is modelled.

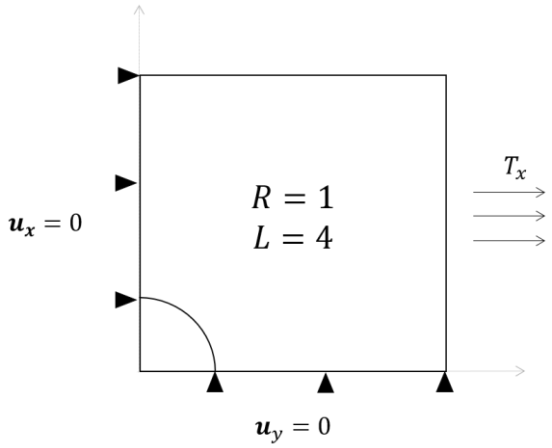


Figure 35. Elastic hole plate: Problem definition.

The exact solution of stress is given in [72] as follows:

$$\sigma_{rr}(r,\theta) = \frac{T_x}{2}\left(1-\frac{R^2}{r^2}\right)+\frac{T_x}{2}\left(1-4\frac{R^2}{r^2}+3\frac{R^4}{r^4}\right)\cos 2\theta$$
$$\sigma_{\theta\theta}(r,\theta) = \frac{T_x}{2}\left(1+\frac{R^2}{r^2}\right)-\frac{T_x}{2}\left(1+3\frac{R^4}{r^4}\right)\cos 2\theta \qquad (2\text{-}1)$$
$$\sigma_{r\theta}(r,\theta) = -\frac{T_x}{2}\left(1+2\frac{R^2}{r^2}-3\frac{R^4}{r^4}\right)\sin 2\theta$$

where T_x denotes the magnitude of the applied stress for the infinite plate.

The geometry and the boundary conditions are described in Figure 35. *R* is the radius of the hole, amd *L* is the edge length of the quarter of the plate. The material parameters and imposed stress are set as:

$$E = 10^5,\ \nu = 0.3$$
$$T_x = 1000 \qquad (2\text{-}2)$$

NURBS are able to exactly model the geometry even the hole at center of plate. Thus it could overcome the geometry error that exit for traditional FEM method. The amplified deformed mesh is shown in Figure 36.

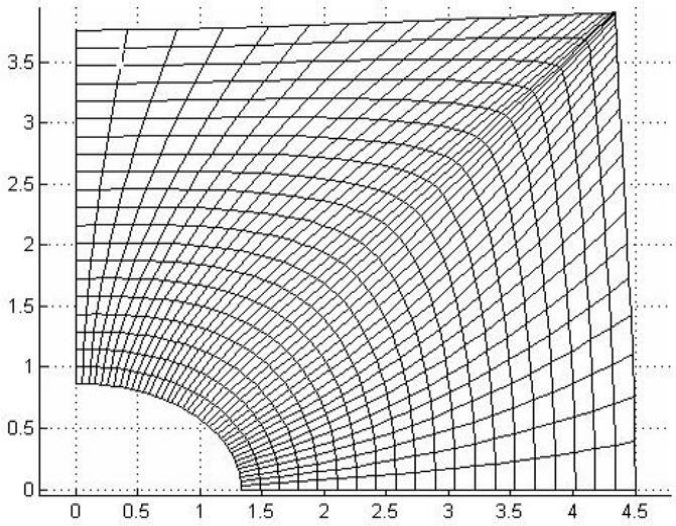


Figure 36. Elastic hole plate: Deformed mesh.

Figure 37 shows the distribution of the stress components. Under this horizontal charge, the component σ_{xx} is dominant. There is a clear stress concentration close to the hole.

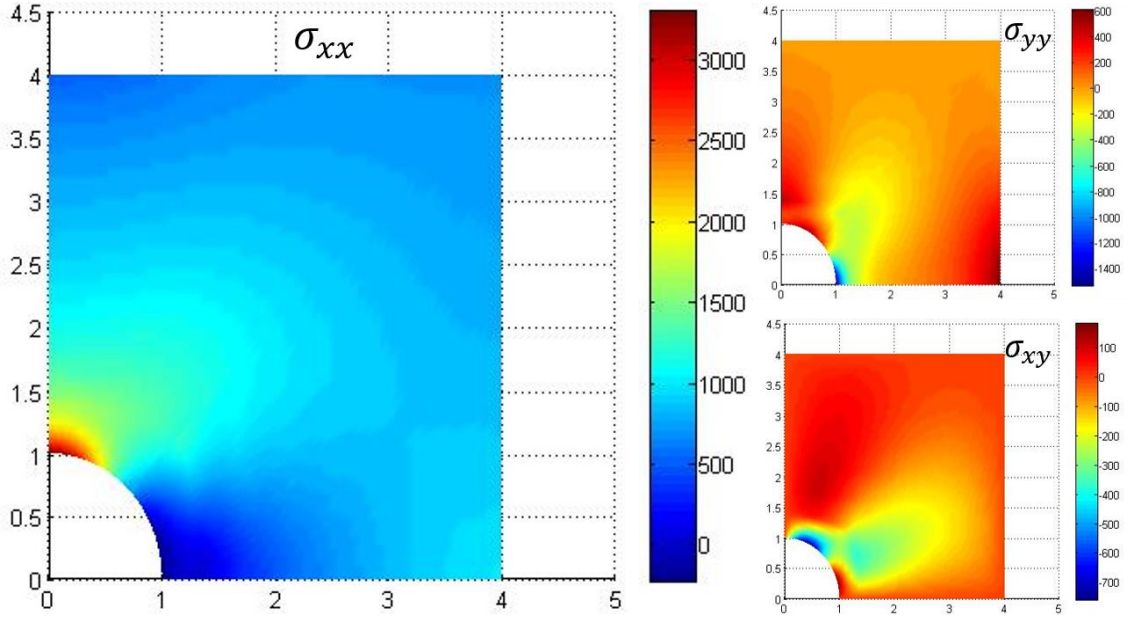


Figure 37. Elastic hole plate: solutions for stress components $\boldsymbol{\sigma_{xx}}$, $\boldsymbol{\sigma_{yy}}$ and $\boldsymbol{\sigma_{xy}}$.

In the Figure 38 and Figure 39, the variation of control net along with the refinement algorithm for NURBS: knot insertion and degree elevation introduced in 1.1.2 has been demonstrated. As expected, when elevating the degree of the patch, the number of control point grows less rapidly than inserting knots.

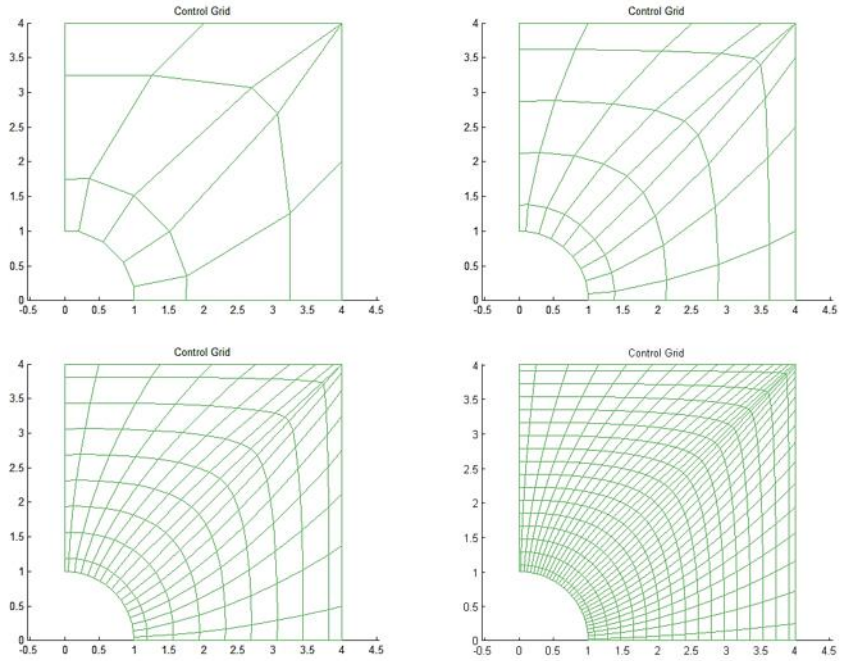


Figure 38. Elastic hole plate: control mesh evolution with knot insertion refinement.

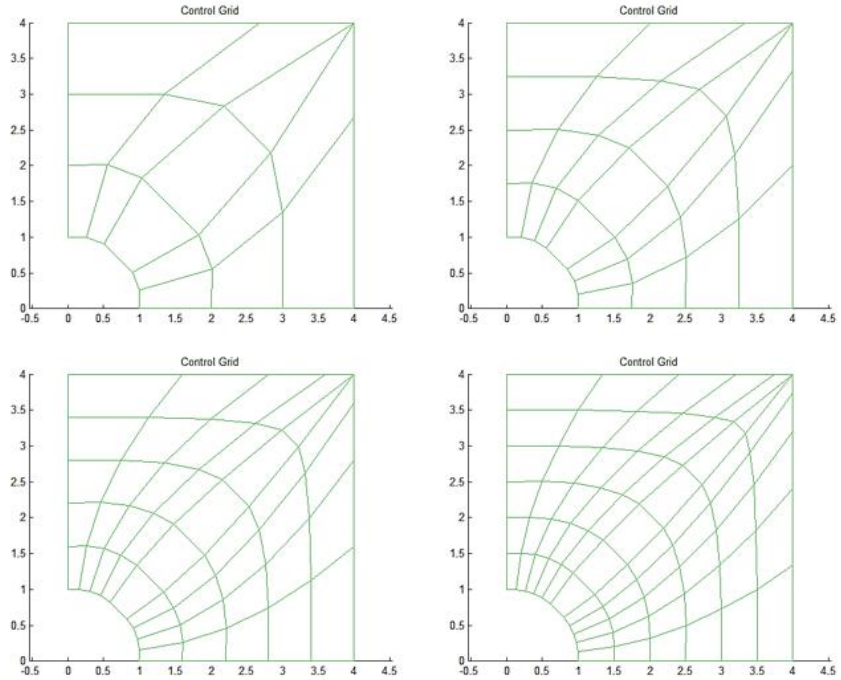


Figure 39. Elastic hole plate: control mesh evolution with degree elevation refinement.

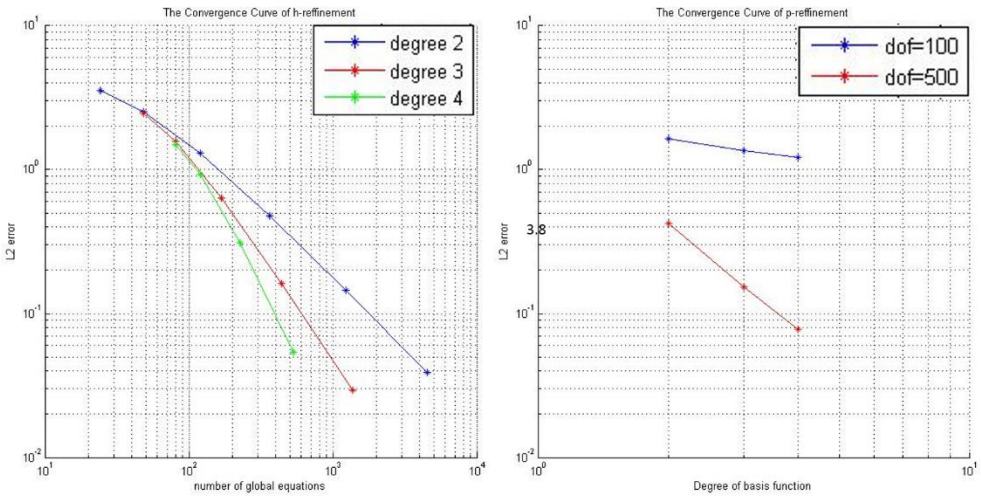


Figure 40. Elastic hole plate: Convergence of L^2-norm of σ with h-refinement for different degrees with respect to degrees of freedom and interpolation degrees.

Convergence results in the L^2-norm of stresses with respect to problem's number of unknowns and to degree of NURBS basis function are shown in Figure 40. The cubic and quartic NURBS are obtained by order elevation of the quadratic NURBS on the coarsest mesh. Since the parameterization of the geometrical mapping does not change, the h-refinement algorithm (knot insertion) generates identical meshes for all polynomial orders.

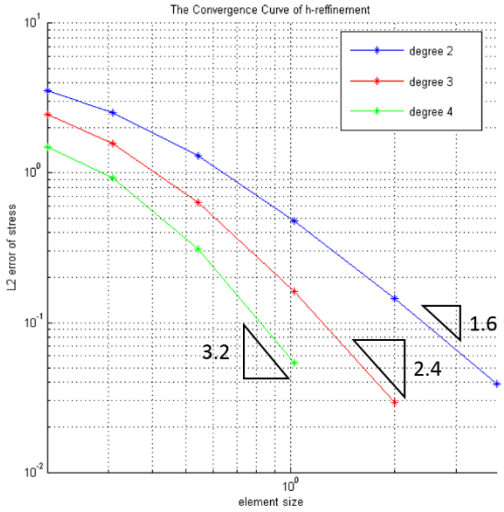


Figure 41. Elastic hole plate: Convergence of L^2-norm of σ with h-refinement for different degrees with respect to element size.

The L^2-error of stress which is somehow equivalent to H^1- norm displacement error is plotted in Figure 41 It shows that the rate of convergence is close to the optimal rate that equals to p.

2.4.4. A 3D spanner

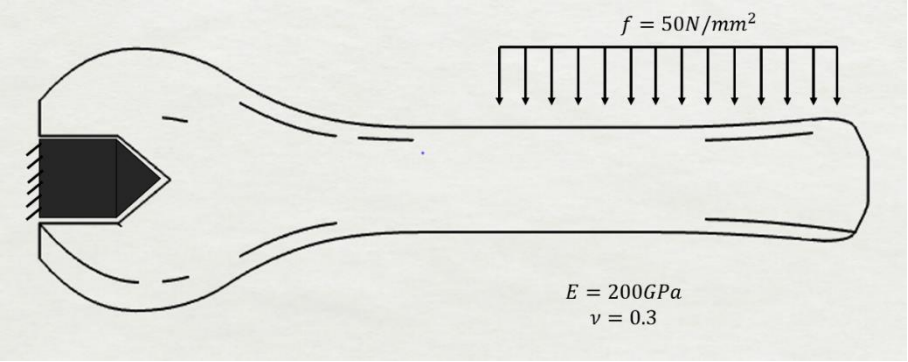


Figure 42. Elastic Spanner: Problem definition

For the second numerical example, we have chosen a more complex 3D geometry to validate the 3D linear elasticity code. Note that the same class for formulation in the code holds for 2D and 3D, only the kinematics of the geometry is changing. The boundary condition and material parameters are given in Figure 42. A vertical surface load $\boldsymbol{f} = 50N/mm^2$ is applied at the top surface as shown in Figure 42. The inside face of jaw is clamped.

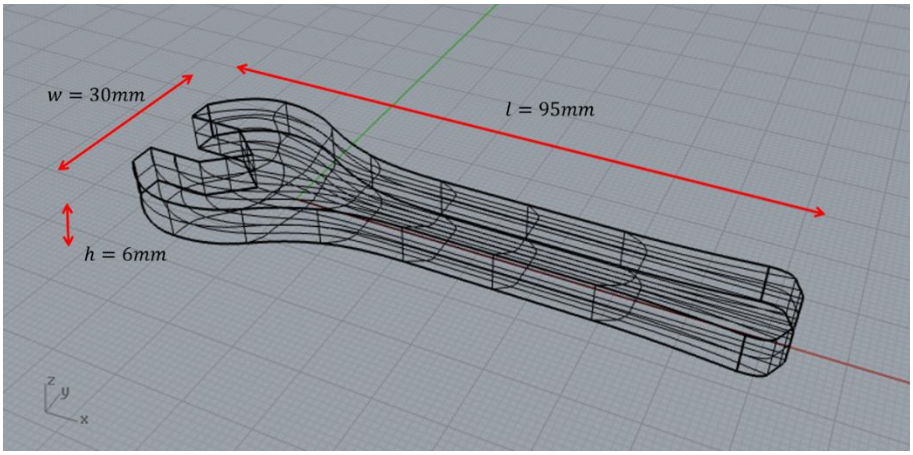


Figure 43. Elastic Spanner: Geometry and dimensions

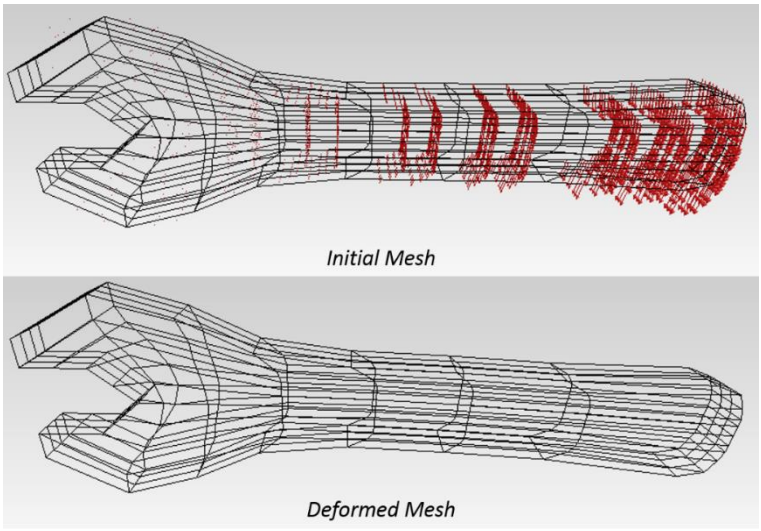


Figure 44. Elastic Spanner: Initial (top) and deformed mesh (bottom)

The NURBS based 2D profile is built in Rhino with a single patch. Then, the 3D patch is obtained in FEMJava by extruding the 2D patch. The detailed geometric dimensions are depicted in Figure 43.

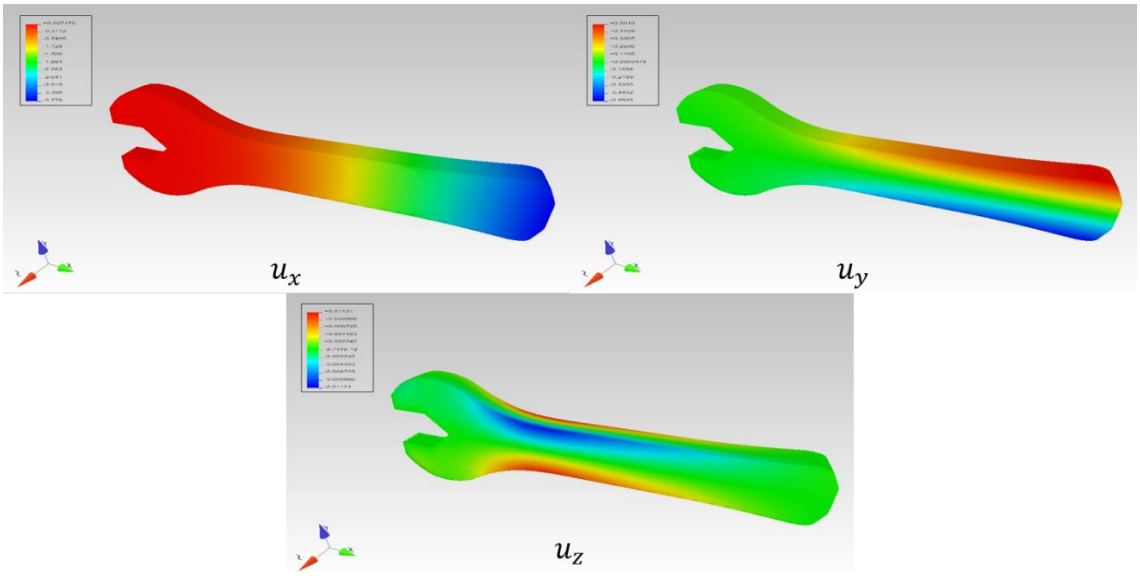


Figure 45. Elastic Spanner: displacement components

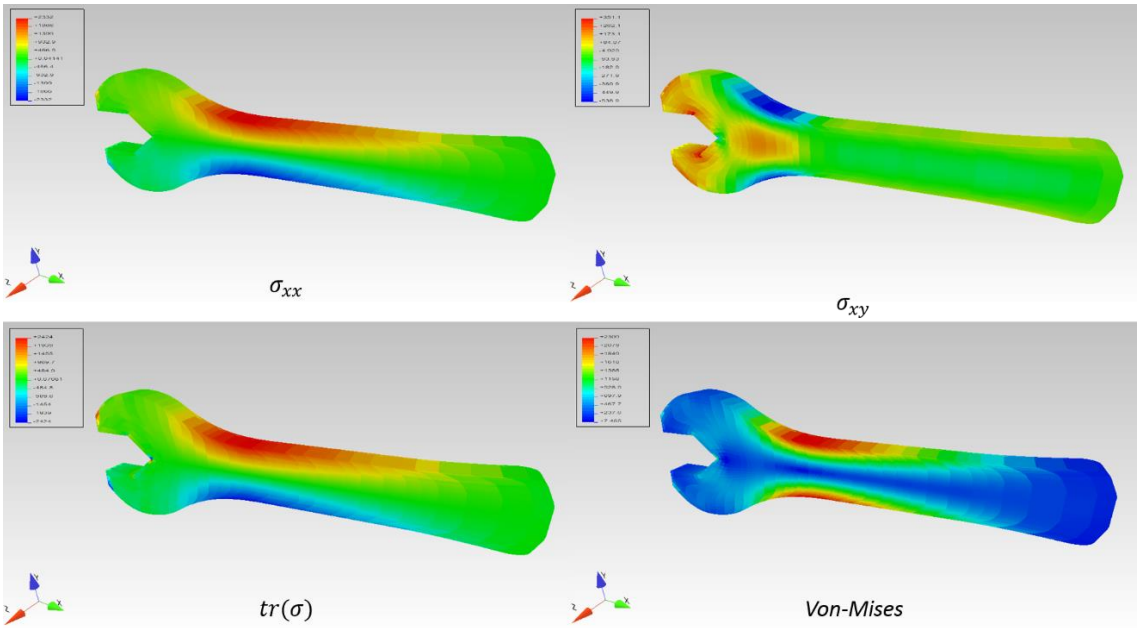


Figure 46. Elastic Spanner: Stress's components and trace, Von-Mises

Figure 44 demonstrated the initial and deformed mesh. We observe a flexion along the handle. The displacement components in all three directions $\{u_x, u_y, u_z\}$ has been shown in Figure 45 in form of isochromatic diagram. The stress components and a post-processed Von-Mises stress has been displayed as well in Figure 46. We can see clearly a stress concentration at the connection the fixed jaw and charged handle. The solution obtained here is qualitatively correct. This example shows that FEMJava has all the necessary tools to perform a isogeometric analysis for problems with complex geometries, including post-processing facilities we have developed.

3. Modeling and implementation of elastomers

In this chapter, we focus on the modeling of rubber like materials with isogeometric analysis. After a short introduction to the behavior of elastomers, we present different material frameworks and models from the simplest to the more complicated: hyperelastic, thermomechanical, thermos-chemo-mechanical.

3.1. A brief introduction to elastomers

Nowadays, the industrial use of rubber-like products is mainly due to the properties of elastomers. They offer the capability to be deformed to quite large deformations, and then spring back to their original form (for filled materials this can take some time). This results from crosslinks in the polymer that provides a force to restore the chains to stress-free conformations. Rubber like materials belong to the family of polymers. It is a kind of materials which have been used for centuries. These materials include firstly the naturally occurring polymers, such as rubber, cellulose, proteins. Modern scientific developments make it possible to artificially synthesize numerous polymers which can be produced less expensively and with superior physical properties than their natural counterparts. The growing industrial demand is accompanied by a need of simulation tools suited to accuratly describe their behaviors. The simulation of polymer like materials poses challenges on the constitutive side. The molecules of polymers are generally gigantic hydrocarbon molecules, and they are often referred to as macromolecules. Within each molecule, the atoms are bound together by covalent interatomic bonds. As the backbone of polymers, the long carbon chains are responsible for much of polymer's constitutive characteristics. These physical characteristics of a polymer depend principally on its macromolecular mass, molecular shape, and more importantly on the differences in the structure of the molecular chains which is also referred as molecular structure.

Polymer's molecular structure mainly includes the types: linear, branched, crosslinked and network, as illustrated in Figure 47. In crosslinked polymers, adjacent linear molecular chains are jointed one to another at various positions by covalent bonds. The processing of crosslinking is achieved during either synthesis or nonreversible chemical reaction. In general, the crosslinking is accomplished by adding atoms or molecules that are covalently

bonded to the chains. Rubbers are generally crosslinked. For rubbers, this process is called vulcanization.

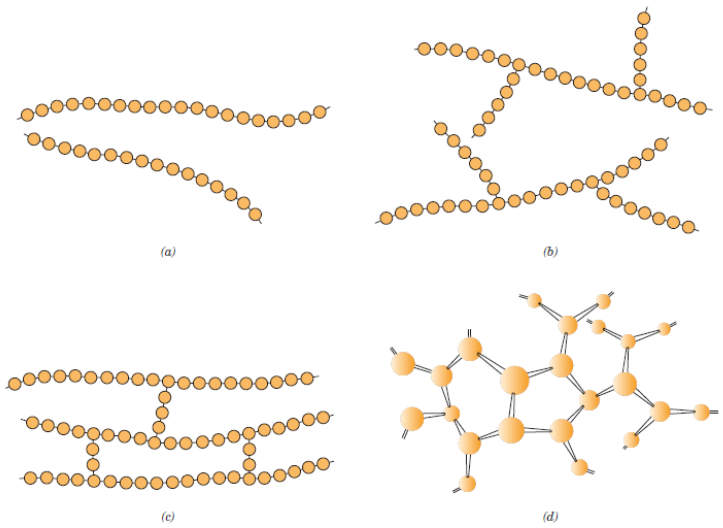


Figure 47. Schematic representations of polymer molecule structures: (a) linear (b) branched (c) crosslinked and (d) network from [238]

The mechanical response of a polymer at elevated temperatures depends on the dominant molecular structure of it. Thus, polymers are classified into two categories: thermoplastic polymers and thermosetting polymers. They are also respectively called thermoplastics and thermosets. Thermoplastics usually exhibits softening when heated and hardening when cooled. This kind of process is totally reversible. As most thermoplastics predominantly possess the linear or branched molecular structure, with the rising of temperature, secondary bond forces such as Van der Waals interactions or hydrogen bond are significantly reduced. The relative movement of chains is facilitated when the material is submitted to mechanical loading. As the temperature decreases, these secondary bonds inter-chains can be reestablished. Thermosetting polymers, with predominantly crosslinked or network molecular structure, remain permanently hard during their formation, and do not soften upon heating. They develop covalent crosslinks between macromolecular chains. These are stronger than secondary bonds. Therefore, these covalent bonds keep the chains together when the material suffers from heating. Only heating to excessive temperatures will cause severance of these crosslink bonds and polymer degradation. Thermoset polymers are generally harder than thermoplastics and have better dimensional stability.

Most of the crosslinked and networked polymers, which include vulcanized rubbers and some polyester resins, are thermosetting. The terminology elastomer indicated a polymer that possesses viscoelastic characteristics and relatively weak inter-molecular interaction forces. A typical instance of elastomer is the natural rubber. The mechanical properties of elastomers such as elasticity modulus are relativity small and vary nonlinearly with strain. In a relaxed state, an elastomer will be amorphous and composed of crosslinked molecular chains that are highly twisted, kinked, and coiled. Elastic deformation, e.g. upon application of a tensile load, is simply the partial uncoiling, untwisting, and straightening of these chains as illustrated in Figure 48**Error! Reference source not found.**. Upon release of the stress, the chains spring back to stress-free configurations, and the macroscopic piece returns to its original shape.

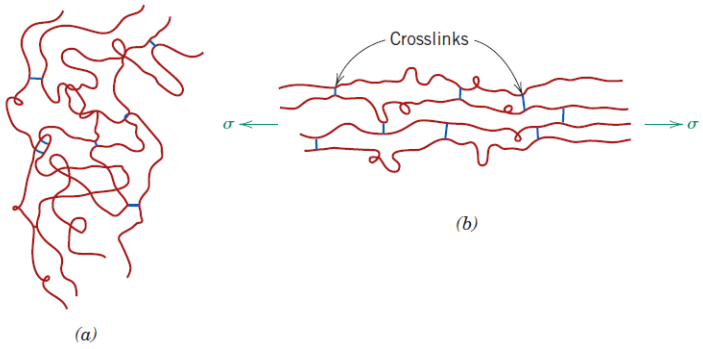


Figure 48. Schematic representation of crosslinked elastomer chain molecules in relaxed and stressed states from [238]

3.1.1. Hyperelastic models for elastomers

In order to model the nonlinear elastic behavior of elastomer, hyperelastic models has been developed. Two main approaches of hyperelastic models exist: the phenomenological models and the statistical models. The phenomenological approach seeks to reproduce the experimental data from a purely mathematical point of view. It consists in searching a strain energy density function, from which the stress-strain relationship derives. The most popular phenomenological models are: Mooney (1940), Mooney-Rivlin (Rivlin, 1948), Ogden (1972) and Gent (1996). The second branch is based on the Blotzmann law that

relates the entropy of a single chain to a probability function. This function describes the probability of presence of the endpoint of a macromolecular chain (the start point located at the origin) in a spherical annulus of radius r, one can derive the free energy of an elastomer, such as Neo-Hookean model and Arruda-Boyce model. Some models of this type can also take into account the topological constraints between the macromolecular chains. The models cited are all based on the same thermodynamic framework.

3.1.2. Thermo-mechanical behavior of elastomer

Because of the specific characteristics of their microstructures, the cured elastomers can suffer large deformations even up to 700% in a quasi-reversible way. A curve of uniaxial traction test of elastomer to the rupture exhibits a highly nonlinear elastic behavior. In Figure 49, the curve consists of three steps with respect to the elongation. The first step features a linear elastic behavior for a less than 10% stretching. Then, there comes a softening of the specimen corresponding to disentanglement of macromolecular chains. This softening is followed by a strong stiffening that identifies the scalability limits of macromolecular chains. This stiffening effect may also be reinforced by the crystallization of chains. For rubber like materials, the equilibrium strain energy is directly related to the change of entropy. That's why it is called entropic elasticity.

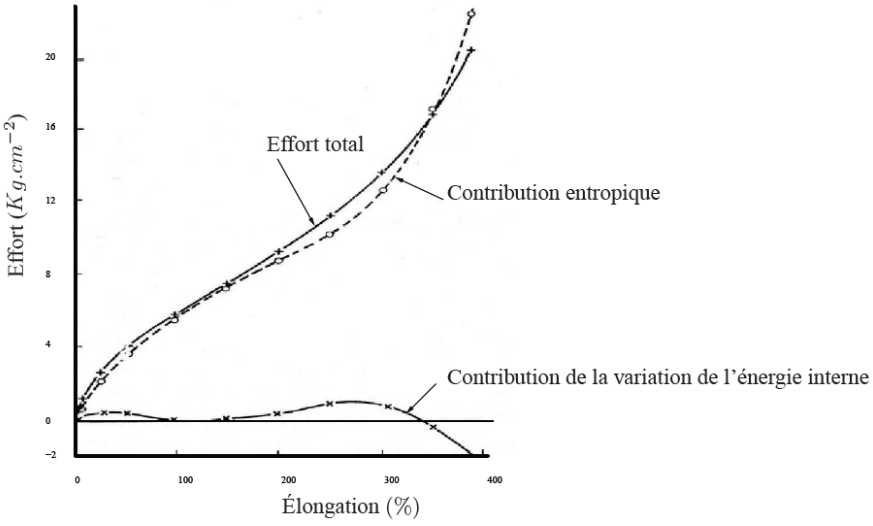


Figure 49 Uniaxial traction on elastomer from [239]

In most applications for elastomers, changes in volume are very small. Therefore, they are usually assumed to be weakly compressible or even incompressible (e.g. see the modeling framework adapted in Lejeunes et al [240] and references therein). It is important to note that the bulk modulus of rubber (1000-5000MPa) is small compared to other materials but high compare to the its shear modulus (0.1-10 MPa).

The thermo-elastic inversion phenomena during an adiabatic stretching on rubber is illustrated by the curve of temperature relating to deformation. This fact was discovered by Joule.

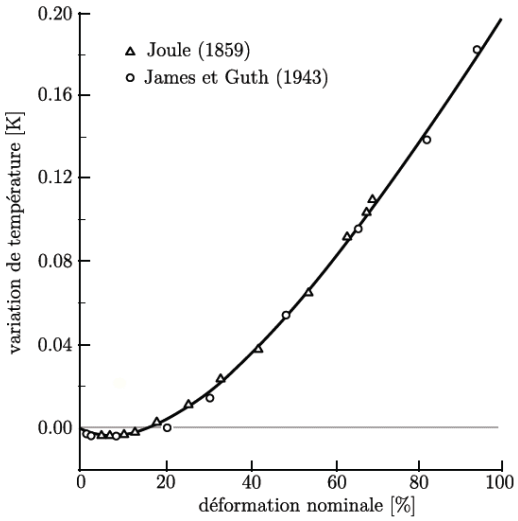


Figure 50. Thermo-elastic inversion point during adiabatic stretching on rubber from [241]

For a small stretch ratio, the rubber denotes an initial cooling effect. As the deformation increasing, it comes a heating effect at a certain minimum point, which is the so-called thermo-elastic inversion point. This effect is still a signature of strong thermo-mechanical coupling of elastomers: it can be explained by a competition between thermal dilatation and entropic elasticity. The mechanical behavior of filled elastomer depends on the temperature, the amplitude of the loading but also the history of loading. One can find many experimental studies on these topics in the literatures. For instance, we can refer to Mullins et al [242], Lion et al [243] and Robisson et al [244]. Consider the results in Martinez et al [245] as an example for the cyclic dynamic loading on elastomers. In the

latter, the mechanical responses of non-filled and filled rubber band are compared. We notice an increase in dynamic stiffness and loss angle (energy dissipation) by the comparison on stabilized responses shown in Figure 51.

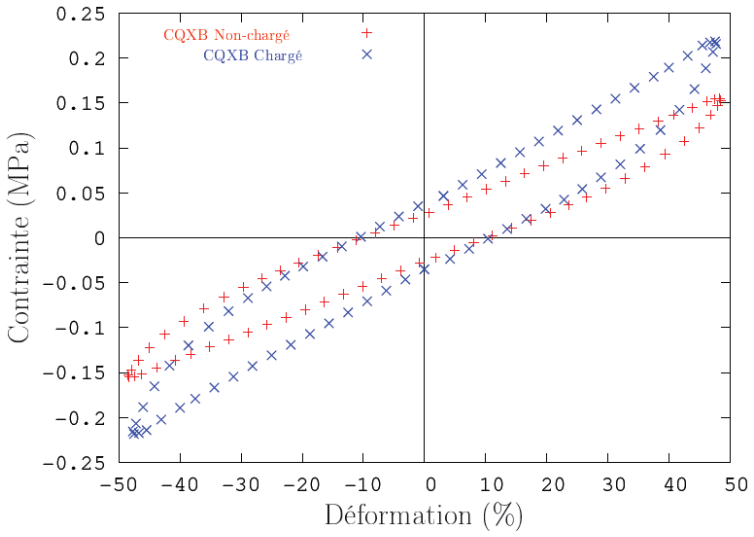


Figure 51. Rubber-like materials responses under cyclic dynamic loading from [245]

From a simple microscopic view, the reinforcements emphasize the interactions like sliding or viscous friction between reinforcements/macromolecular chains and also between reinforcements and reinforcements. This dissipation phenomenon is very sensitive to the stress velocity. And generally, there exists no macroscopic permanent deformation in regard to a charge-discharge test. Therefore, they are considered as viscoelastic elastomers.

As presented above filled elastomers exhibit a highly dissipative behavior. because of their low thermal conductivity, the mechanical dissipation is partially transformed into heat and lead to a strong temperature growth inside the matter. This phenomenon is named self-heating. If we consider the experimental results of Figure 52, it can be observed at the center of the cylinder, the temperature is much higher than at the boundary, and get even get close to the one necessary in the vulcanization process. Therefore, one can image that the thermal energy may lead to a chemical aging (new chemical reactions together with a physical reordering of the chains and fillers of network) within the material.

The elastomer material coefficients, such as bulk modulus, thermic dilatation coefficient and calorific capacity depends closely on the temperature. The self-heating phenomenon will make the practical problem more complicate.

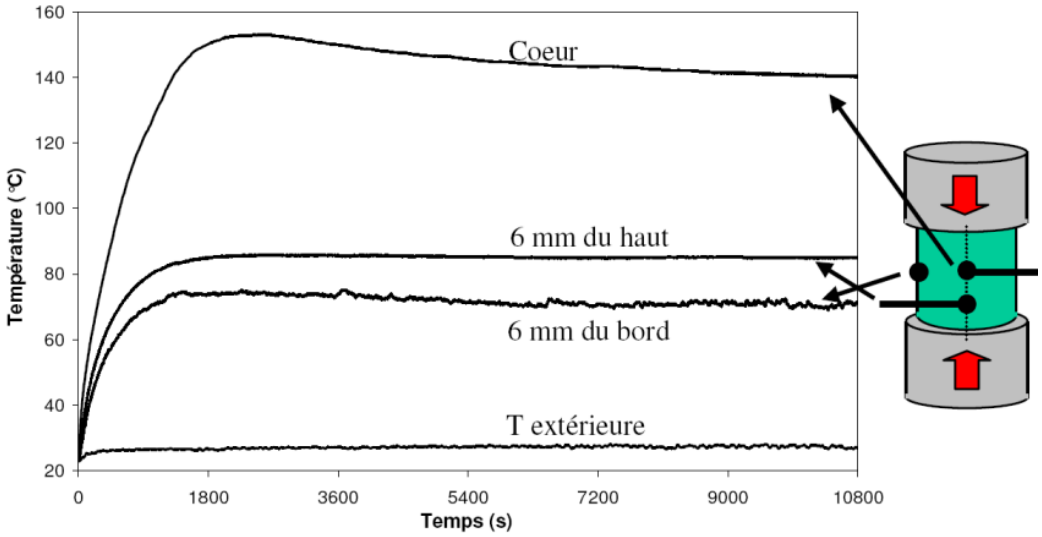


Figure 52. Self-heating during cyclic solicitation from [246]

In this subsection, we have introduced some important phenomenological features in elastomers. It should be note that additional aspects such as damage and fatigue, crystallization, Mullins and Payne effects exist in elastomer behavior.

3.1.3. Thermo-mechanical models for elastomer

Strong thermo-mechanical coupling exits in elastomer. The large deformation modeling of elastomers started with the rise of modern experimental facilities and the development of viscoelastic and viscoplastic models. One can refers to the works: Miehe [247], Holzapfel et Simo [248], Reese et Govindjee [249], Boukamel et Méo [250], Resse et al [251] and Behnke et al [252]. The level of thermo-mechanical coupling in the models relies on the following factors:

- the coupling terms in the heat equation: mechanical dissipation, the latent heat (thermos-elastic coupling);

- the dependence of thermal parameters e.g. heat capacity, thermal conductivity and thermal dilatation coefficient on temperature and deformation for hydrostatic pressure;

For thermo-elastic models, the deformation gradient is usually split into a purely mechanical part and a thermal dilatation part. This thermal expansion part of free energy is assumed to depend nonlinearly on the temperature. The isochoric mechanical part is often assumed to be described by entropic elastic models with a linear temperature dependency. As mentioned previously, these models can reproduce the thermos-elastic inversion phenomenon (see Miehe [247] , Holzapfel et Simo [248] and references therein).

These previously mentioned models conserve energy during a loading process. Which means they are not capable of simulating essential behaviors of elastomer such as the viscosity and self-heating under cyclic solicitation. Some extension were accomplished by Reese et al [249] [251], Boukamel et al [250] and Méo [253]. The proposed models differ from one to another by the coupling level. In the works of Reese et Govindjee [249] and Reese [251], a thermo-viscoelastic model was developed with a nonlinear evolution law for viscous behaviors. Two main assumptions were made for this model: the multiplicative decomposition of the deformation gradient into volumetric, thermal and isochoric parts; and a nonlinear dependence of the non-isothermal free energy on the temperature (that leads to a temperature dependency for heat). The mechanical dissipation and the latent heat were taken into consideration in the heat balance equation by Méo [253] and Boukamel et al [250]. It was aasumed in the previous work that only part of the mechanical dissipation is transformed into heat. The rest of the dissipation is assumed to produce microstructural changes within elastomers. At last, Yang et al [254] and Stainier et al [255] proposed a variational approach for coupling problems. This makes it possible to rewrite the thermal and mechanical balance equations as an optimization problem of a scalar valued function. The variational approach has the advantage of leading to a numerical formulation with a symmetric structure. Moreover, the authors have proved that this formalism offers a more interesting computational cost than the traditional way.

3.1.4. Thermo-chemo-mechanical models for elastomer

The thermo-chemo-mechanical coupling modeling for elastomer is a relatively new research topic, consequently there are few models discussed in the literature. By enlarging the search field to polymer, we perceive that this problematic has been addressed in recent years. For polymers, the chemical phenomena taken into consideration are vulcanization, chemical or thermal ageing of materials. These models are divided into infinitesimal strain regime and finite strain regime (see Nguyen [241]).

3.2. Principle of virtual work

In this section, we discuss at the outset the most important variational principles that can be used within the finite element method and isogeometric analysis. The so-called single-field variational principle is presented in this subsection.

3.2.1. Spatial description of variational principle

In current configuration at finite strain regime, with the definition of boundary value problem of a continuum body as the start point, the principle of virtual work in spatial description is defined as:

Find $\boldsymbol{u} \in V_u$ that for all $\delta\boldsymbol{u} \in V_u^0$,

$$\begin{aligned} \delta W(\boldsymbol{u}, \delta\boldsymbol{u}) &= \delta W_{int} - \delta W_{ext} = 0 \\ \delta W_{int}(\boldsymbol{u}, \delta\boldsymbol{u}) &= \int_\Omega \boldsymbol{\sigma} : \delta\boldsymbol{e}\, dV \\ \delta W_{ext}(\boldsymbol{u}, \delta\boldsymbol{u}) &= \int_\Omega \boldsymbol{b} \cdot \delta\boldsymbol{u}\, dV + \int_{\partial\Omega_\sigma} \bar{\boldsymbol{t}} \cdot \delta\boldsymbol{u}\, dS \end{aligned} \tag{3-1}$$

where $\delta\boldsymbol{u}$ represent the virtual displacement, and δW_{int} and δW_{ext} are known as internal virtual work and external virtual work. Then $\delta\boldsymbol{e}$ denoted the first variation of the Euler-Almansi strain tensor $\boldsymbol{e}$, and it equals to the symmetric part of $grad(\delta\boldsymbol{u})$.

$$\delta \boldsymbol{e} = \frac{1}{2}(grad^T \delta \boldsymbol{u} + grad \delta \boldsymbol{u}) \tag{3-2}$$

The smooth virtual displacement field $\delta \boldsymbol{u}$ is arbitrary over the current region Ω and over the boundary surface $\partial\Omega_\sigma$ where the traction vector $\bar{\boldsymbol{t}}$ is prescribed. And the boundary surface $\partial\Omega$ is partitioned into disjoint part:

$$\partial\Omega = \partial\Omega_u \cup \partial\Omega_\sigma \text{ with } \partial\Omega_u \cap \partial\Omega_\sigma = \emptyset \tag{3-3}$$

3.2.2. Material description of variational principle

Similarly, from the material balance equations, we can derive the principle of virtual work in mixed description.

Find $\boldsymbol{u} \in S_u$ that for all $\delta \boldsymbol{u} \in S_u^0$,

$$\begin{aligned} \delta W(\boldsymbol{u}, \delta \boldsymbol{u}) &= \delta W_{int} - \delta W_{ext} = 0 \\ \delta W_{int}(\boldsymbol{u}, \delta \boldsymbol{u}) &= \int_{\Omega_0} \boldsymbol{P} : Grad(\delta \boldsymbol{u})\, dV \\ \delta W_{ext}(\boldsymbol{u}, \delta \boldsymbol{u}) &= \int_{\Omega_0} \boldsymbol{B} \cdot \delta \boldsymbol{u} dV + \int_{\partial\Omega_{0\sigma}} \overline{\boldsymbol{T}} \cdot \delta \boldsymbol{u}\, dS \end{aligned} \tag{3-4}$$

where $\delta \boldsymbol{u}$ represent the virtual displacement, and δW_{int} and δW_{ext} are known as internal virtual work and external virtual work. S_u indicates the Soblev space on region Ω_0.

By a pull-back operation on the internal virtual work δW_{int}, it allows to obtain the principle of virtual work in material description that was implemented in the code.

Find $\boldsymbol{u} \in S_u$ that for all $\delta \boldsymbol{u} \in S_u^0$,

$$\begin{aligned} \delta W(\boldsymbol{u}, \delta \boldsymbol{u}) &= \delta W_{int} - \delta W_{ext} = 0 \\ \delta W_{int}(\boldsymbol{u}, \delta \boldsymbol{u}) &= \int_{\Omega_0} \boldsymbol{S} : \delta \boldsymbol{E}\, dV \\ \delta W_{ext}(\boldsymbol{u}, \delta \boldsymbol{u}) &= \int_{\Omega_0} \boldsymbol{B} \cdot \delta \boldsymbol{u} dV + \int_{\partial\Omega_{0\sigma}} \overline{\boldsymbol{T}} \cdot \delta \boldsymbol{u}\, dS \end{aligned} \tag{3-5}$$

where $\boldsymbol{S}$ and $\delta\boldsymbol{E}$ represent the second Piola-Kirchhoff stress tensor and Green-Lagrange strain tensor in terms of virtual displacement $\delta\boldsymbol{u}$ and the deformation gradient $\boldsymbol{F}$ explicated as:

$$\delta\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^T Grad(\delta\boldsymbol{u}) + (\boldsymbol{F}^T Grad(\delta\boldsymbol{u}))^T) \quad (3\text{-}6)$$

These three principles of virtual works are all usable to modeling a pure mechanical problem at the finite strain regime. However, in our works, for the reason of simplicity, we choose the total Lagrangian description. Consequently, all the formulations implemented in FEMJava are based on the first or second Piola-Kirchhoff variational principles.

3.3. A Java object-oriented implementation of hyperelastic models and formulations

Many free energy functions for hyperelastic materials have been proposed in the literatures since the end of 1940s. Initially focusing on natural rubbers and elastomers (vulcanized elastomer, synthetic elastomer, etc.), the models have extended to many other domains of material science, such as biological tissues. In this section, we focus on isotropic hyperelastic models and their implementation in an object-oriented framework. The hyperelastic models can be decomposed in two sets: incompressible and compressible. This feature has a consequence on both the form of the free energy form and the variational statement. We propose a general object-oriented implementation for decoupled compressible isotropic hyperelastic models in a compatible formulation. This framework is also capable to model weakly compressible materials. At last, we illustrate the section with two classical compressible models: a Neo-Hookean model and a Mooney-Rivlin model.

3.3.1. Incompressible and quasi-incompressible materials

For incompressible materials, the internal constraint of incompressibility is stated as follows:

$$J = 1 \tag{3-7}$$

The free energy function can be expressed with a Lagrange multiplier p. A general form of the constitutive law is:

$$\psi = \psi(\boldsymbol{F}) - p(J-1),\ \ J = det(\boldsymbol{F}) \tag{3-8}$$

The multiplier p can be identified as hydrostatic pressure. Volumetric changes entirely vanish when p tends to positive infinity.

At finite strains, the materials allowing volumetric changes are modeled by compressible hyperelastic models. Bulk and shear deformations have significant different contributions in the free energy. A decoupling of free energy into isochoric and volumetric part has been proposed as follow:

$$\psi = \psi_{iso} + \psi_{vol} \tag{3-9}$$

Mathematically, the decoupling results from a multiplicative decomposed of the deformation gradient $\boldsymbol{F}$,

$$\boldsymbol{F} = (J^{1/3}\boldsymbol{I})\overline{\boldsymbol{F}} \tag{3-10}$$

where J is the determinant $\det(\boldsymbol{F})$ and $\boldsymbol{I}$ the second order identity tensor. In this context, the right Cauchy-Lagrange strain tensor can be decomposed in the same way:

$$\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F} = \left(\left(J^{\frac{1}{3}}\boldsymbol{I}\right)\overline{\boldsymbol{F}}\right)^T\left(\left(J^{\frac{1}{3}}\boldsymbol{I}\right)\overline{\boldsymbol{F}}\right) = \left(J^{\frac{2}{3}}\right)\overline{\boldsymbol{C}} \tag{3-11}$$

where $\overline{\boldsymbol{C}} = \overline{\boldsymbol{F}}^T\overline{\boldsymbol{F}}$.

These two isochoric strain measures $\overline{\boldsymbol{F}}$ and $\overline{\boldsymbol{C}}$ are respectively called modified deformation gradient and the modified right Cauchy-Lagrange strain tensor. The volumetric part ψ_{vol} depends on the gradient of the deformation gradient, i.e. $\psi_{vol}(J)$, and the isochoric part ψ_{iso} depends only on modified strain tensors, i.e. either $\psi_{iso}(\overline{\boldsymbol{F}})$ or $\psi_{iso}(\overline{\boldsymbol{C}})$.

3.3.2. Material formulation and iterative algorithm

The implementation of compatible formulation is based on material description of finite strain motion in the form of variational formulation and Galerkin method for discretization. In the code FEMJava, the formulations that have been implemented are based on the principle of virtual work in material description as defined in equations (3-4) and (3-5).

Since the principle of virtual work at the regime of finite strain is nonlinear equation which is caused by geometrical nonlinearity and/or material nonlinearity. Thus, it cannot be solved directly, but demands an iterative scheme such as Newton-Raphson algorithm that was implemented in the code. To adapt this method, it obliges us to linearize the equation (3-4) and (3-5). In this paragraph, we take the consistent linearization of equation (3-5) as an example.

Here we derive the linearized form of the principle of virtual work in Lagrangian description:

$$\begin{aligned} D_{\Delta u}\delta W(\vec{u},\delta\vec{u}) &= \frac{d}{d\varepsilon}\,\delta W_{int}(\vec{u}+\varepsilon\Delta\vec{u},\delta\vec{u})|_{\varepsilon=0} \\ &= \int_{\Omega_0}(\boldsymbol{S}:D_{\Delta u}\delta\boldsymbol{E}(\vec{u})+\delta\boldsymbol{E}(\vec{u}):\mathbb{C}(\vec{u}):D_{\Delta u}\boldsymbol{E}(\vec{u}))dV \end{aligned} \tag{3-12}$$

where the $\mathbb{C}(\vec{u})$ represents the elasticity tensor that is a forth order tensor.

$$\mathbb{C}=\frac{\partial\boldsymbol{S}}{\partial\boldsymbol{E}}=2\frac{\partial\boldsymbol{S}}{\partial\boldsymbol{C}}=4\frac{\partial^2\psi}{\partial\boldsymbol{C}\partial\boldsymbol{C}} \tag{3-13}$$

To conclude the numerical implementation of the material formulation, the most demanding effort lies in the computing of the second Piola-Kirchhoff stress tensor $\bar{\bar{S}}$ and the elasticity tensor $\mathbb{C}$.

3.3.3. A tentative object-oriented implementation

The implemented models are all based on the decoupled isotropic compressible hyperelastic free energy in the form $\psi(\bar{I}_1,\bar{I}_2,J)$. In FEMJava, an abstract super class for all

these types of models are created and named by **Hyperelasticity**, as shown by the UML class diagram in Figure 53.

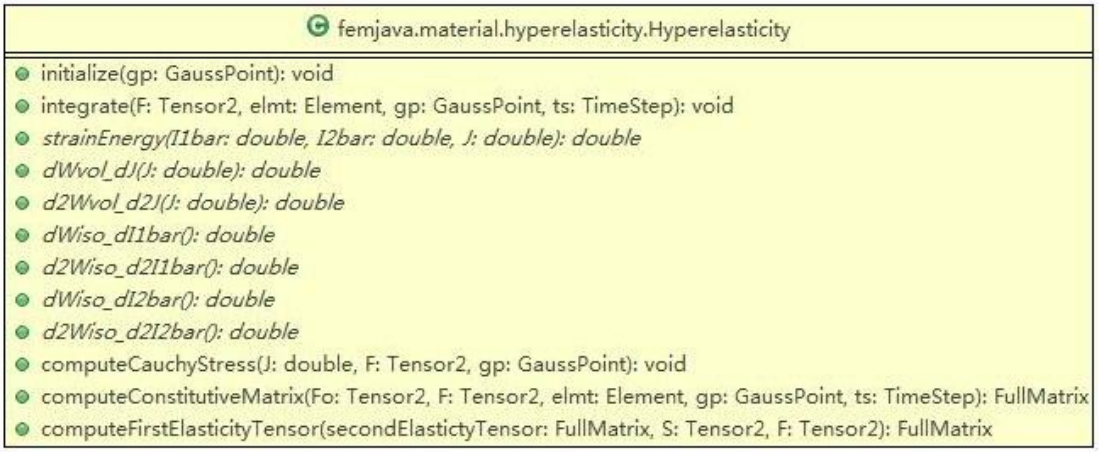


Figure 53. The abstract super class: *Hyperelasticity*

In order to calculate the tangent matrix for iterative algorithm, the most important tasks of the class *Hyperelasticity* consist in evaluating the second Piola-Kirchhoff stress tensor $\bar{\bar{S}}$ and the elasticity tensor $\mathbb{C}$. Since these two tensors stem from the first and second order partial derivation of isochoric and volumetric parts of free energy that denoted by $\psi_{iso}(\bar{I}_1, \bar{I}_2)$ and $\psi_{vol}(J)$ respectively. Thus, the intermediate terms need to be computed first:

$$\frac{\partial \psi_{iso}(\bar{\boldsymbol{C}})}{\partial \bar{I}_1}, \frac{\partial \psi_{iso}(\bar{\boldsymbol{C}})}{\partial \bar{I}_2} \text{ and } \frac{\partial^2 \psi_{iso}(\bar{\boldsymbol{C}})}{\partial \bar{I}_1^{\,2}}, \frac{\partial^2 \psi_{iso}(\bar{\boldsymbol{C}})}{\partial \bar{I}_2^{\,2}}$$

along with:

$$\frac{\partial \psi_{vol}(J)}{\partial J} \; and \; \frac{\partial^2 \psi_{vo}(J)}{\partial J^2}$$

The evaluation of these intermediate terms is carried out by the methods:

```
public abstract double dWvol_dJ(double J);

public abstract double d2Wvol_d2J(double J);

public abstract double dWiso_dI1bar();

public abstract double d2Wiso_d2I1bar();

public abstract double dWiso_dI2bar();

public abstract double d2Wiso_d2I2bar();
```

Note that these methods are abstract. The implementations will be provided in subclasses such as the Neo-Hookean model.

According to the chain rule, the calculated of $\frac{\partial \psi(\boldsymbol{C})}{\partial \boldsymbol{C}}$ and $\frac{\partial^2 \psi}{\partial \boldsymbol{C} \partial \boldsymbol{C}}$ can be completed by evaluating the partial derivative of modified invariants $\overline{I}_1$, $\overline{I}_2$ and determinant of deformation gradient J with respect to, either right Cauchy-Green strain tensor $\boldsymbol{C}$, or Green-Lagrange strain tensor $\boldsymbol{E}$. From a practical point of view, these computations are implemented in the methods *integrate()* and *computeConstitutiveMatrix()* which are the same for all the isotropic decoupled hyperelastic models.

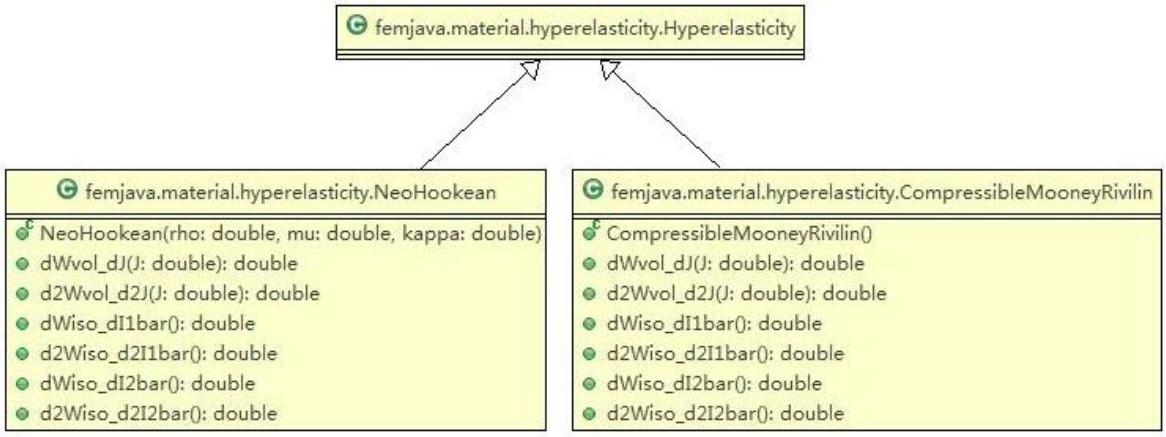


Figure 54. Subclassed of *Hyperelasticity* for Neo-Hookean and Mooney-Rivilin models

At last, two subclasses of **Hyperelasticity** for a Neo-Hookean model and a Mooney-Rivlin model are shown in Figure 54 as typical examples of free energy classes. They respectively implement the two following models:

$$\text{Neo} - \text{Hookean:}$$

$$\psi(\bar{I}_1, J) = C_1(\bar{I}_1 - 3) + \frac{1}{\beta}(J^\beta - 1) \tag{3-14}$$

$$\text{Mooney} - \text{Rivlin:}$$

$$\psi(\bar{I}_1, \bar{I}_2, J) = C_{10}(\bar{I}_1 - 3) + C_{01}(\bar{I}_2 - 3) + \frac{1}{\beta}(J^\beta - 1)$$

3.3.4. Multi-field formulations implementation

The basic formulation introduced in 3.2 has just one displacement field, as shown in Figure 55. In this situation, the application of NURBS based isogeometric analysis requires only one patch instance.

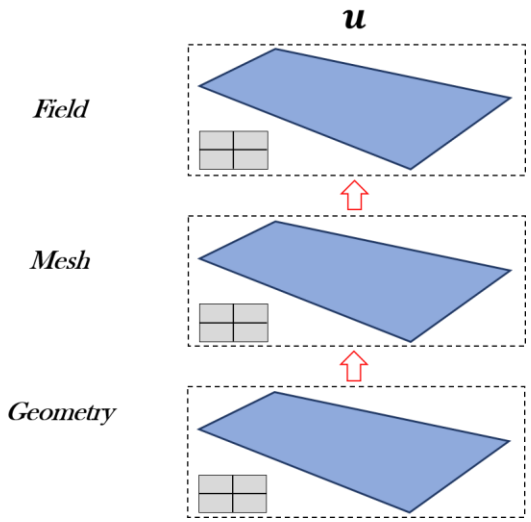


Figure 55 One field standard formulation

However, numerous models for elastomer include multi-physical phenomenon, such as those thermos-mechanical and thermos-chemo-mechanical approaches. A formulation to simulate this kind of models holds generally multiple fields. Furthermore, those

formulations for materials with internal constraint such as mixed formulations, have also more than one fields.

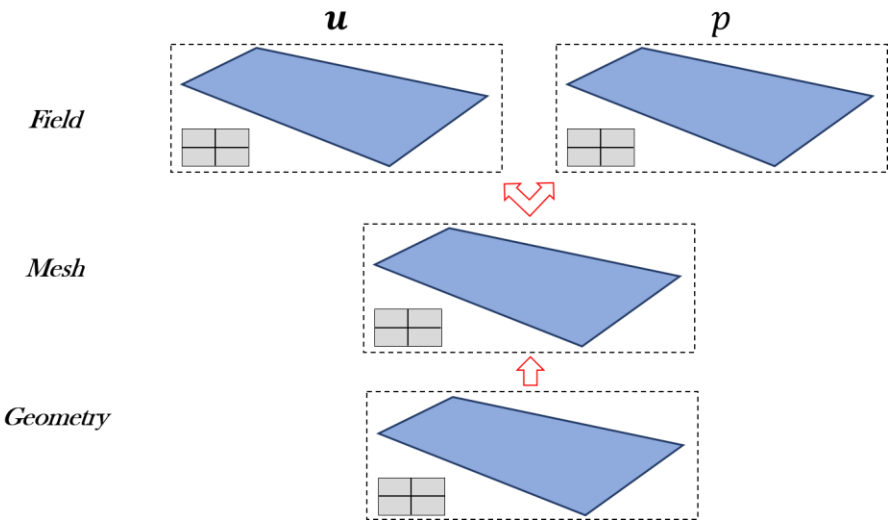


Figure 56. Two fields formulation with identical interpolation

Recalling the strategy of FEMJava to create multi-fields, we can make it clear by the schema in Figure 56 for an example where there are a field for displacement $\boldsymbol{u}$ another field for hydrostatic pressure p (more detailed explications will be given in the following chapter). It shows that these two field hold each an interpolation instance, but exactly the same topologically structure and geometrical description. That limits the choices for fields interpolation.

In order to cover the general case, where user of the code can specify the approach for every single field in multi-field problems, we have extended the field generation by overriding those concerned methods of the class **Subdomain** in the subclass **SubdomainNURBS** for NURBS based isogeometric analysis. As shown in following code segment, the method *diverseInterpolationForScalarVectorFields()* take charge of generating distinct patches for scalar and vector fields. Then they are used as the support for fields.

```
public class SubdomainNURBS extends Subdomain {

protected Patch patch;

protected Patch scalarPatch;

protected Patch vectorPatch;

public void diverseInterpolationForScalarVectorFields(int
mU, int pU, int mP, int pP, int subdivisionTime) {…}

@Override
public SubdomainField generateAScalarField(Field f) {…}

@Override
public SubdomainField generateAVectorField(Field f) {…}
```

Two examples of distinct patches for two fields problem and three fields problem are schemed in Figure 57 and Figure 58. In the latter, the third field could be a temperature field for thermal physics as an illustration. It is noted that the displacement field's support patch has a finer mesh than that of pressure field or temperature field.

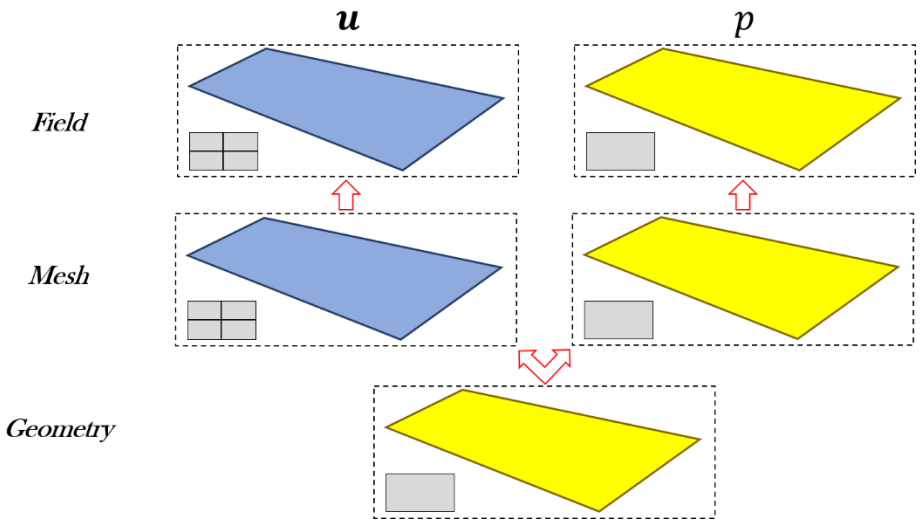


Figure 57 Two fields formulation with distinct interpolations

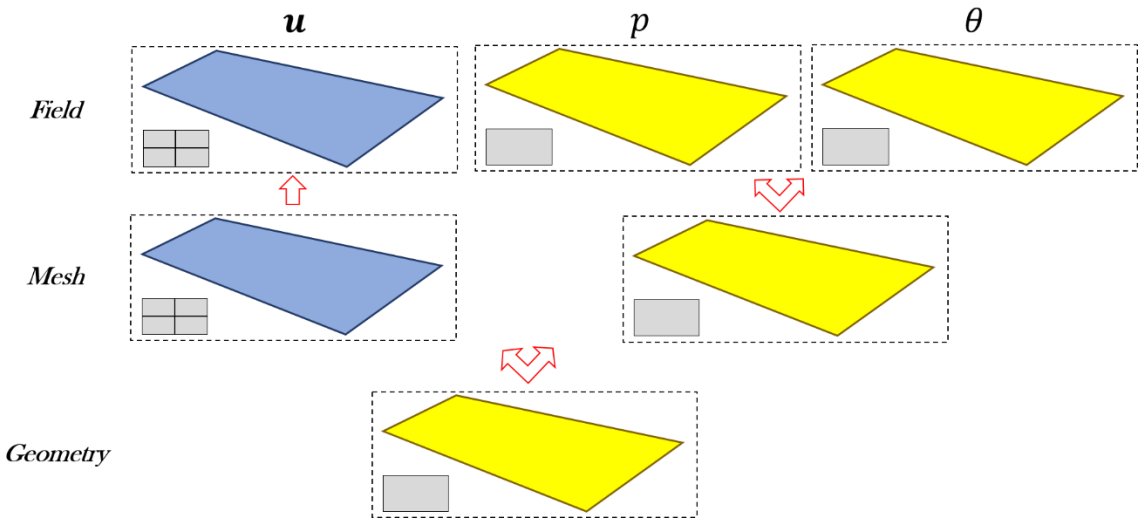


Figure 58. Three fields formulation with diverse interpolations

This special implementation for field construction gives the users the flexibility in choosing interpolations so as to fulfil certain stability condition (LBB) or for other purposes. Meanwhile, since the distinction in topological structure changes the mapping from reference space to parametric space, we are obliged to recalculate the numerical integration point for those slave patches from the primary patch.

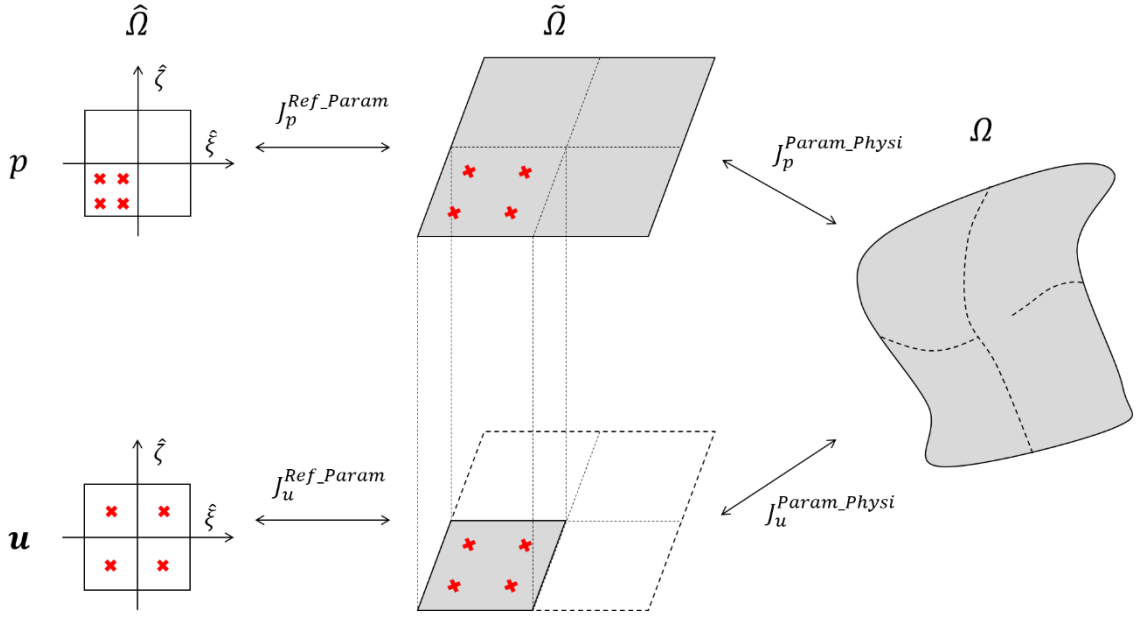


Figure 59. Numerical integration scheme for distinct topological parametric space patches

Considering the example of an approach with displacement field $\boldsymbol{u}$ and pressure field p where the patch for $\boldsymbol{u}$ holds a halved parametric topology as shown in Figure 59. Gauss points indicated by red crossed originate in displacement reference element. The mapping to the parametric space of patch $\boldsymbol{u}$ infers the images of Gauss points. With the assumption that all the patch constructed for different fields share the same parametrization from parametric space to physical space $(\xi, \zeta) \rightarrow (x, y)$, the numerical integration points in pressure reference element can be located by inversing several simple linear equations in each parametric direction.

The diverse interpolation approach causes the loss of the isoparametric property. It may have negative influence on the discretization consistence, therefore probably recedes the convergence rate of error estimation. On the contrary, its ability to satisfy the LBB condition make it possible to improve the stability of solutions. These potential advantages and shortcomings will be demonstrated through several applications in the following chapters.

4. Isogeometric analysis for incompressible material at small and large strains

Most rubber-like materials such as elastomers undergo large deformations without significant volumetric changes. They are modeled as incompressible or weakly compressible materials. In this section, we address the problem of incompressibility or weak compressibility in the context of isogeometric analysis with references to finite elements. At first, we present a brief overview of methods that have been developed to overcome stability problems occurring in a such situation focusing on methods based on projections schemes and mixed methods. Secondly, we draw an overview of IGA methods developed for incompressible or nearly incompressible materials. We then compare the performances of several IGA schemes for incompressibility or weakly compressibility at small and large strains on various numerical tests.

4.1. Incompressibility/weak compressibility in finite element methods

Standard finite elements displacement formulations fail when simulating incompressible or weakly compressible behaviors. E.g. in linear elasticity, when the Poisson's ratio gets close to 0.5, the simple linear approximation with triangle element produces highly oscillatory solutions. In the same way, volumetric locking is a phenomenon that occurs when a confined incompressible or weakly compressible media is submitted to a given load. The numerical solution stems from standard finite element is too small compared with the real solution. In such a case, a reduced rate of convergence or even an incorrect convergence to zero is observed. Volumetric locking is caused by the fact that there are too many incompressibility constraints imposed on the discretized finite element solution, compared to the global number of degrees of freedom. Since the origin of the finite element method a wide range of numerical methods have been proposed to build stable discrete formulations. Similarly, mixed Galerkin formulations for linear incompressible elasticity are stable only for convenient combination of displacement/pressure interpolation. The lack of stability for the pressure field is explained by the Ladyzhenskaya-Babuˇska-Brezzi (LBB) stability condition (or inf-sup stability condition). Stablity can checked numerically either by means of patch tests or by explicitly numerically evaluate the inf-sup stability condition (see Bathe [256])

In the scope of pure displacement formulations, selective and reduced integration method consisting in decreasing the number of numerical integration points for the mean stress contribution was firstly proposed in the work of Naylor [24]. The equivalence of the selective reduced displacement formulations with some mixed finite element methods has been shown e.g. in Malkus et Hughes [25] but are not valid in the general case (anisotropy,…). They also derived the projection method based on the modification of the discrete gradient operator $\boldsymbol{B}$. This method, named $\overline{\boldsymbol{B}}$ method was originally proposed in Hughes [81] by generalization from selective integration and mean-dilatation formulations.

Two fields mixed formulations method were initially proposed in Hermann et al [39] and Taylor & al [40] for a linear incompressible or quasi-incompressible elasticity problem. This method has been extended to nonlinear problems at large strains e.g. in Argyris & al [41]. Simo & Taylor [42] developed a three-field mixed formulation based on the Hu-Washizu variational principle to handle volume-preserving plastic flow. In Jankovich [43], the penalty method come into view by simulating the behavior of rubber parts. A combination of penalty method with reduced integration method was then studied and applied to incompressible elasticity and contact problems by Oden et al [44]. Volumetric locking was overcome through the Enhanced Assumed Strain (EAS) method in Simo et Rifai [45]. More recently, the p-finite elements method for incompressible materials was studied in Wells & al [257], and in Heisserer & al [258]. Its efficiency in alleviating volumetric locking was demonstrated for finite strain problems.

In the scope of mixed methods, many attempts with intention of stabilizing such elements have been introduced. The so-called the Galerkin least squares method was first proposed in Taylor et al [40], and then progressed by Hughes et al [26]–[28]. This kind of approach allows the use of simple low-order and equal order element, but a good selection of the mesh dependent stabilization parameter is necessary. These methods allow to circumvent the *inf-sup* stability condition. Alternative proposal of achieving similar answers has been proposed in Oñate [29] which gains the addition of diagonal terms by the introduction of the so-called finite increment calculus to the formulation. In Another class of stabilized methods based on the enrichment of the interpolation for the Galerkin method with a virtual bubble functions was introduced by Brezzi and coworkers in [30]–[33]. In the mid-90s

Hughes revisited the origins of the stabilization schemes from a variational multiscale view point and presented the variational multiscale method in [34], [35]. The different stabilization techniques come together as special cases of the underlying subgrid scale modeling concept (see e.g. Nakshatrala et al [37] for incompressible linear elasticity, Masud et al [36] for incompressible Navier-Stokes, Cervera et al [38] for plasticity…)

Preliminary remark for IGA

For the sake of simplicity, let introduce a notation proposed by Nielsen et al [49], The discretization combinations for $\boldsymbol{u} - p$ fields is shown in Figure 60. The order of basis function is indicated for each field. Then, for each field, the superscript indicates the multiplicity of inner knots (the continuity order can be deduced from it). The subscript designates the time of h-refinement by halving knot span (subdivision property of NURBS). On the example of Figure 60, the velocity $\boldsymbol{u}$ is of degree 3 and inner knots are repeated twice (no h-refinement), and the pressure p is of degree 3 and inner knots are repeated once (no h-refinement).

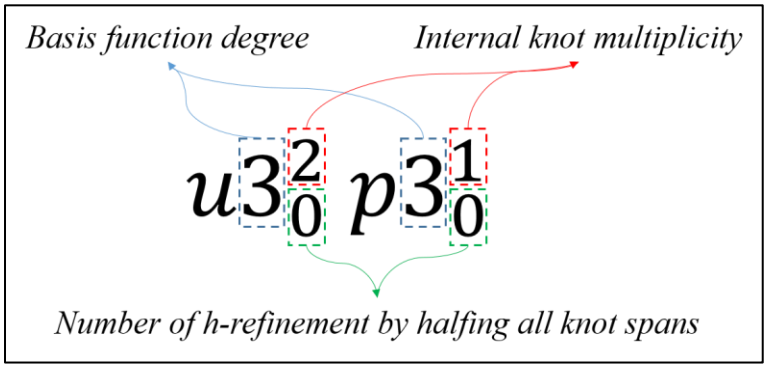


Figure 60. Nomenclature for discretization combinations.

In the context of isogeometric analysis, as already mentioned in the state of the art in section 1.3.1, the treatment of incompressibility and quasi-incompressibility has followed mains tracks.

In the scope of pure displacement formualtions, an extension of the $\bar{B}$ and $\bar{F}$ have been proposed by Elguedj and al [46]. The volumetric part of the strain is projected on patch one order lower than the original one (application quasi-incompressible elasticity at

infinitesimal strains and quasi-incompressible hyperelasticity at finite strains).Adams et al [47] have proposed selective and reduced integration approach applied in the context of NURBS interpolations (application to quasi-incompressible 2D-elasticity and Reissner-Mindlin shell). A first family of mixed formulation concerns two-fields mixed formulations velocity/pressure fields (application to Stokes flow that can be interpreted as incompressible linear elasticity for displacement/pressure fields). The underlying idea for mixed formulations is that two-fields velocity/pressure $\boldsymbol{u} - p$ are stable only for suitable combinations of discretization for $\boldsymbol{u} - p$. The so-called Ladyzenskaia-Babushka-Brezzi (LBB) stability condition or inf-sup condition must be fulfilled to guaranty the stability:

$$inf_p sup_{\boldsymbol{u}} \frac{\int_\Omega p \nabla \cdot \boldsymbol{u} \, d\Omega}{\|p\| \, \|\boldsymbol{u}\|} \geq \beta > 0 \qquad (4\text{-}1)$$

where β is a constant.

For discrete fields, the condition to avoid the volumetric locking is $n_u \geq n_p$ where n_u is the number of unknowns for displacement and n_p is the number of unknowns for pressure parameter. The ideal value of the constraint ration $r = \frac{n_u}{n_p}$ is the number of equilibrium equations divided by the number of incompressibility conditions. The key idea for mixed formulation is to choose discretizations combination for which $n_u \geq n_p$, and the possibilities to enforce it with NURBS is rich: increase the degree of basis function for velocity, proceed to a h-refinement for velocity, repeat inner knots for velocity. Recall that for Lagrange finite elements, the only way to achieve it, is the increase the degree of basis functions for velocity. Ensuing this basic idea, in Nielsen & al [49], different kind of discretizations are tested combinations of type $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$, $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$, $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p - 1_0^1)$, $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p - 1_0^1)$, $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p - 2_0^1)$, $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p - 2_0^1)$, $(\boldsymbol{u}\, p_0^1, \boldsymbol{p}\, p - 2_0^1)$ where p is the degree of basis functions, Raviart-Thomas and Nédélec like elements. We give in Table 4-1 an example of 1D discretization for these different elements for cubic interpolations ($p = 3$). The *inf-sup* condition is tested numerically to prove the stability of the proposed elements in the context of IGA. Note that the stable elements are applied for fluid (Navier-Stokes equations). In Buffa et al [48], the approach is similar. The elements proposed are: Raviart-Thomas, Nédélec $(\boldsymbol{u}\, p + 1_0^\alpha, \boldsymbol{p}\, p_0^\alpha)$ where p is the degree of basis functions and $0 \leq \alpha \leq p - 1$. The numerical evaluation of the constant of the inf-sup condition is done

to prove the stability of the elements. In Kadapa et al [62], the approach is similar in the context of solid mechanics (quasi-incompressible elasticity, plasticity at small and large strains). The second family of mixed formulations are three field formulation based on the Hu-Washizu variational formulation or equivalent is proposed in Taylor et al [50]. In Cardoso et al [51] based on the Hu-Washizu variational principle, an enhanced assume strain is developed.

Table 4-1. Velocity and Pressure discretization combinations

Type	Name	Velocity knot vector	Pressure knot vector
a	$(\boldsymbol{u}3_1^1, p3_0^1)$	[0 0 0 0 0.5 1 1.5 2 2 2 2]	[0 0 0 0 1 2 2 2 2]
b	$(\boldsymbol{u}3_0^2, p3_0^1)$	[0 0 0 0 1 1 2 2 2 2]	[0 0 0 0 1 2 2 2 2]
c	$(\boldsymbol{u}3_1^1, p2_0^1)$	[0 0 0 0 0.5 1 1.5 2 2 2 2]	[0 0 0 1 2 2 2]
d	$(\boldsymbol{u}3_0^2, p2_0^1)$	[0 0 0 0 1 1 2 2 2 2]	[0 0 0 1 2 2 2]
e	$(\boldsymbol{u}3_0^1, p3_0^1)$	[0 0 0 0 1 2 2 2 2]	[0 0 0 0 1 2 2 2 2]

The main drawback of the selective and reduced integration scheme is that the generalization to anisotropic formulations, and the strongly nonlinear applications we target may exhibit strong anisotropy (non-symmetric anisotropic tangent operator). Three fields formulation are quite heavy in term of complexity and computational cost. We prefer to address simpler formulations from a practical point of view, to be able to more easily take into account additional physics. Simple two-field formulations have proven to be accurate to deal with incompressibility. In the following, we will systematically compare several elements focusing on IGA mixed elements of type $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p-1_0^1)$ $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p-1_0^1)$ $(\boldsymbol{u}\, p_0^1, \boldsymbol{p}-\boldsymbol{1}\, p_0^1)$ at small and finite strain. We include in this comparative study the strain projection method ($\bar{B}$ and $\bar{F}$). In Table 4-2, the method that will be tested and compared at small and finite strain are summarized.

Table 4-2. Implemented Locking-Free methods

	Infinitesimal Strain	*Finite Strain*
Strain projection method	$\bar{B}$ projection	$\bar{F}$ projection
Mixed formulation	Type Stokes equations	Type H-R

As shown in Table 4-2 these two categories method will be tested and compared at small and finite strain.

4.2. A numerical comparison of stability at small strains

In this section, we compare the numerical performance of different patches on the example of a simple mixed formulation for incompressible linear elasticity. For the sake of simplicity, we consider a form known as the homogeneous Stokes problem. This equation model that can represent either the Stokes flow ($\boldsymbol{u}$ velocity/ $\boldsymbol{p}$ pressure) or the incompressible linear elasticity equation ($\boldsymbol{u}$ displacement/ $\boldsymbol{p}$ pressure parameter). These equations of the problem are:

$$\begin{cases} -\mu\nabla^2\boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega \\ \nabla \cdot \boldsymbol{u} = \boldsymbol{0} \quad \text{in } \Omega \\ \boldsymbol{u} = \bar{\boldsymbol{u}} \quad \text{on } \partial_1\Omega \\ (2\mu(\nabla\, \boldsymbol{u} + \nabla^t\boldsymbol{u}) + p\boldsymbol{I}).\boldsymbol{n} = \boldsymbol{0} \quad \text{on } \partial_2\Omega \end{cases} \tag{4-2}$$

where μ is the Lamé's coefficient, $\boldsymbol{u}$ is the displacement, $\boldsymbol{f}$ represents the body loads. The incompressibility constraint is expressed $\nabla \cdot \boldsymbol{u} = 0$. The weak form of the problem is given as follows:

Find $\boldsymbol{u} \in S_u$ and $p \in S_p^0$ that $\forall\delta\boldsymbol{u} \in S_u^0$ and $\forall\delta p \in S_p^0$

$$\begin{cases} \mu \int_\Omega \nabla\delta\boldsymbol{u} : \nabla\boldsymbol{u}\, d\Omega - \int_\Omega \nabla \cdot \delta\boldsymbol{u}\, p\, d\Omega = \int_\Omega \delta\boldsymbol{u} \cdot \boldsymbol{f}\, d\Omega \\ \int_\Omega \nabla \cdot \boldsymbol{u}\, \delta p\, d\Omega = 0 \end{cases} \tag{4-3}$$

The matrix form of the problem is given by:

Discretizing the variational equations with the elemental expressions $\boldsymbol{u}^{h,e} = \boldsymbol{N_u}\widetilde{\boldsymbol{d^e}}$ and $p^h = \boldsymbol{N_p}\widetilde{\boldsymbol{p}}$, we obtain the matrix form of the problem:

$$\begin{bmatrix} \boldsymbol{K_{uu}} & \boldsymbol{K_{up}} \\ \boldsymbol{K_{pu}} & 0 \end{bmatrix} \begin{Bmatrix} \widetilde{\boldsymbol{d}} \\ \widetilde{\boldsymbol{p}} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{f_u} \\ \boldsymbol{f_p} \end{Bmatrix} \tag{4-4}$$

that

$$\boldsymbol{K}_{\boldsymbol{uu}}^{\boldsymbol{e}} = \int_{\Omega} \boldsymbol{B}^{T} \boldsymbol{D}^{dev} \boldsymbol{B} dV$$

$$\boldsymbol{K}_{\boldsymbol{up}}^{\boldsymbol{e}} = \boldsymbol{K}_{\boldsymbol{pu}}^{\boldsymbol{e}\,\boldsymbol{T}} = \int_{\Omega} \boldsymbol{G}^{T} \boldsymbol{N}_{\boldsymbol{p}} dV \tag{4-5}$$

With $\boldsymbol{K}_{\boldsymbol{uu}} = \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{K}_{\boldsymbol{uu}}^{\boldsymbol{e}}$, $\boldsymbol{K}_{\boldsymbol{up}} = \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{K}_{\boldsymbol{up}}^{\boldsymbol{e}}$

$$\boldsymbol{K}_{\boldsymbol{pu}} = \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{K}_{\boldsymbol{pu}}^{\boldsymbol{e}} \text{ , } \boldsymbol{f}_{\boldsymbol{u}} = \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{f}_{\boldsymbol{u}}^{\boldsymbol{e}}$$

In this subsection, the strain projection methods and the mixed formulation methods are applied to treat incompressible problems within the framework presented in the previous paragraph.

In the following, the L^2-norm of both the displacement error and the pressure error are numerically evaluated as:

$$\epsilon_u = \sqrt{\int_{\Omega} \|\boldsymbol{u}(x,y) - \bar{\boldsymbol{u}}(x,y)\|^2 \, d\,\Omega} \tag{4-6}$$

$$\epsilon_p = \sqrt{\int_{\Omega} \|p(x,y) - \bar{p}(x,y)\|^2 \, d\,\Omega} \tag{4-7}$$

4.2.1. Body load driven problem

As a first numerical experiment, inspired by the problem studied in Buffa & al [48], we consider an homogeneous Dirichlet and Neumann problem in unit cavity. The equation model is given in (4-2) with $\mu = 1$.

The definition of problem is given in Figure 61. The solution of this problem is chosen by $(\boldsymbol{u}, p) = (\bar{\boldsymbol{u}}, \bar{p})$ as follows:

$$\bar{\boldsymbol{u}} = \begin{bmatrix} 2e^{x}(-1+x)^2 x^2 (-1+y) y (-1+2y) \\ -e^{x}(-1+x) x (-2+3x+x^2)(-1+y)^2 y^2 \end{bmatrix} \tag{4-8}$$

and

$$\bar{p} = sin(2\pi x) sin(2\pi y) \tag{4-9}$$

We impose the following body loads $\boldsymbol{f_u} = -\mu\nabla^2\bar{\boldsymbol{u}} + \nabla\bar{p}$ for the mechanical problem, and $\boldsymbol{f_p} = -\nabla\cdot\bar{\boldsymbol{u}}$ for the constraint equation. We plot an example of solution in Figure 62 for the combination $(\boldsymbol{u}4_0^2, p4_0^1)$ for a 20×20 mesh. The divergence of displacement is of magnitude 10^{-7} as shown in Figure 63.a. The distribution of pressure parameter in Figure 63.b does not exhibit oscillations. When we chose interpolation of type $(\boldsymbol{u}1_0^1, p1_0^1)$ and $(\boldsymbol{u}3_0^1, p3_0^1)$ which are known to be unstable, the numerical solutions for displacement seem to be truthful (Figure 64.a and c), but the pressure distributions turn out to be erroneous as shown in Figure 64.b and d. The solution exhibits a checkboard pattern such as in finite elements. This shows that equal order patches should definitely be avoided such as it should be for equal order interpolations in finite elements.

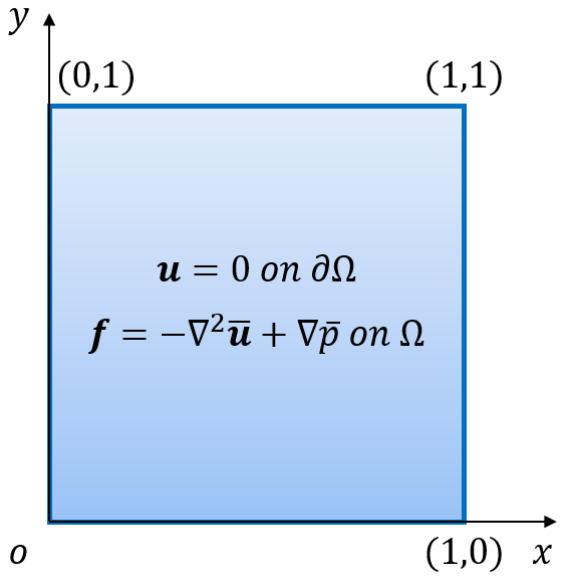


Figure 61. Problem definition of Stokes flow

We study the convergence of –norm of displacement and pressure errors for different types of interpolations $(\boldsymbol{u}\,p_0^1, \boldsymbol{p}\,p_0^1)$, $(\boldsymbol{u}\,p_0^2, \boldsymbol{p}\,p_0^1)$, $(\boldsymbol{u}\,p_1^1, \boldsymbol{p}\,p_0^1)$, $(\boldsymbol{u}\,p + 1_0^1, \boldsymbol{p}\,p_0^1)$ and $(\boldsymbol{u}\,p + 1_0^2, \boldsymbol{p}\,p_0^1)$ for $p = 2, \ldots, 5$. The mesh varies from 2×2 to 32×32 knot spans. And at the same time, we ascend the degree for each discretization combination from $p = 2$ to $p = 4$.

The results for convergence study of L^2-norm of displacement and pressure errors are plotted in Figure 65 to Figure 69. In Figure 65, the discretization of $(\boldsymbol{u}\,p_0^1, \boldsymbol{p}\,p_0^1)$ gives the

optimal convergence rate $(p+1)$ for the L^2-norm of displacement error. However, the test fails for the pressure. This confirms the qualitative remark made for Figure 64. We observe for any other combination i.e. $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$, $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$, $(\boldsymbol{u}\, p+1_0^1, \boldsymbol{p}\, p_0^1)$ and $(\boldsymbol{u}\, p+1_0^2, \boldsymbol{p}\, p_0^1)$ an optimal convergence of the L^2-norm of displacement and pressure errors as shown in Figure 66, Figure 67, Figure 68 and Figure 69. However, the convergence of the L^2-norm of displacement error for $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ is one order lower than for $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ for any order of interpolation.

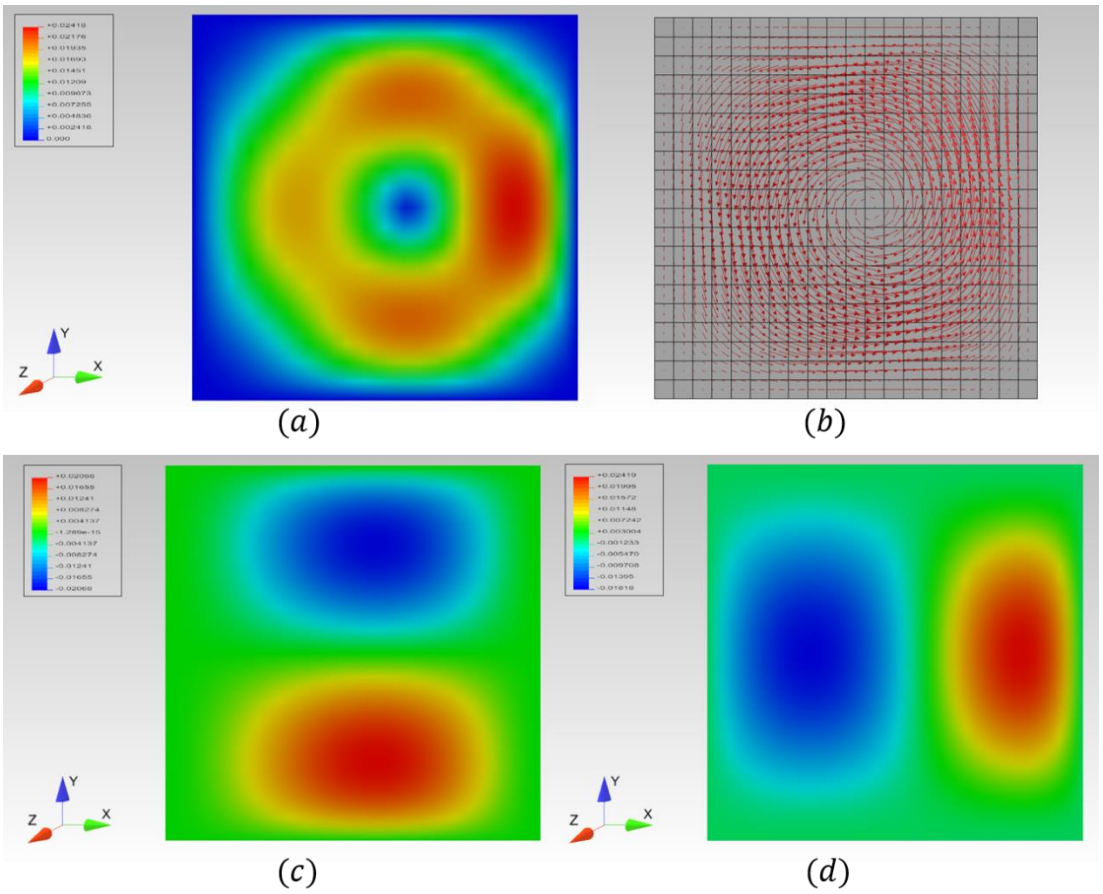
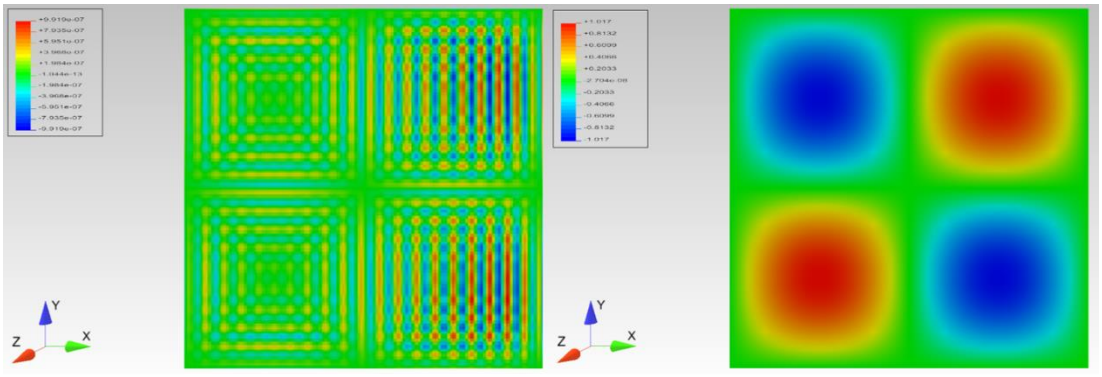
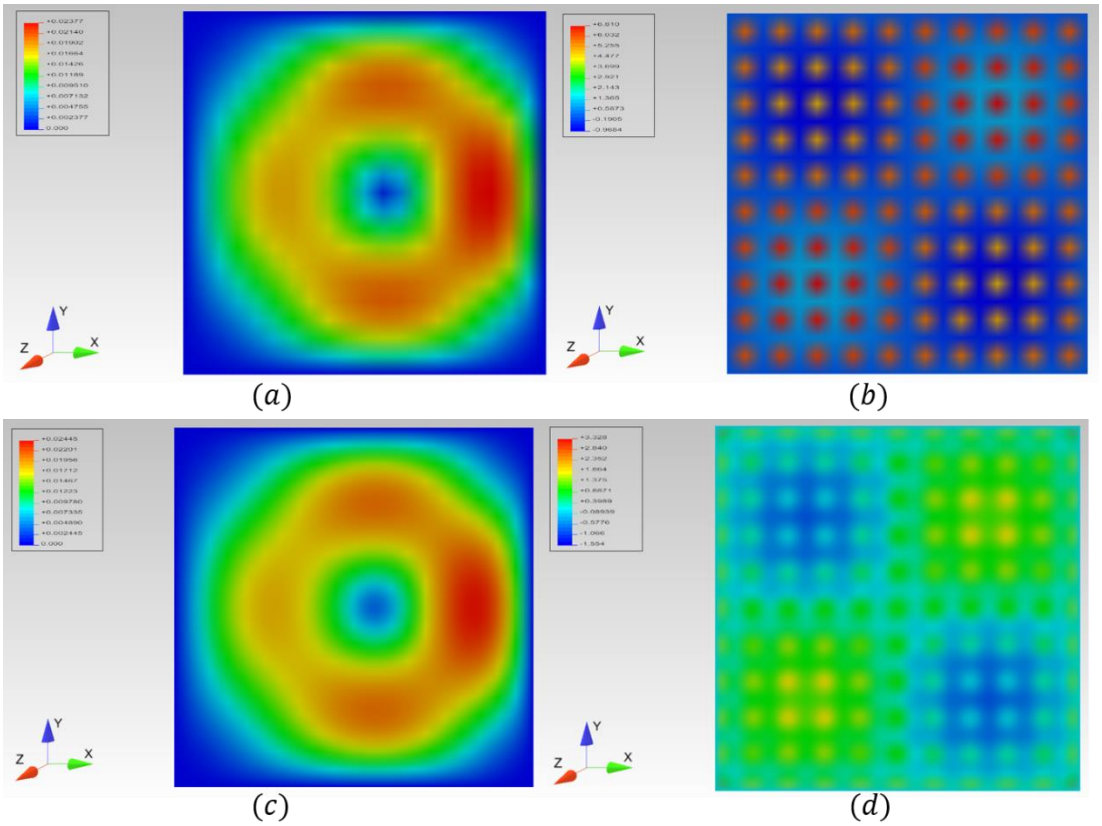


Figure 62. Numerical solution for velocity $\boldsymbol{u}$ in FEMJava: (a) the nom of velocity $||\boldsymbol{u}||$; (b) mesh and flow distribution; (c) $\boldsymbol{u}_x$; (d) $\boldsymbol{u}_y$



(a) (b)

Figure 63. Numerical solution for velocity and pressure: (a) $\boldsymbol{\nabla \cdot u}$; (b) Pressure distribution $\boldsymbol{p}$



(a) (b)

(c) (d)

Figure 64. Numerical solution with instable discretization combinations of type (e) for linear and cubic interpolations: (a) velocity norm $(\boldsymbol{p = 1})$; (b) pressure distribution $(\boldsymbol{p = 1})$; (c) velocity norm $(\boldsymbol{p = 3})$; (d) pressure distribution $(\boldsymbol{p = 3})$.

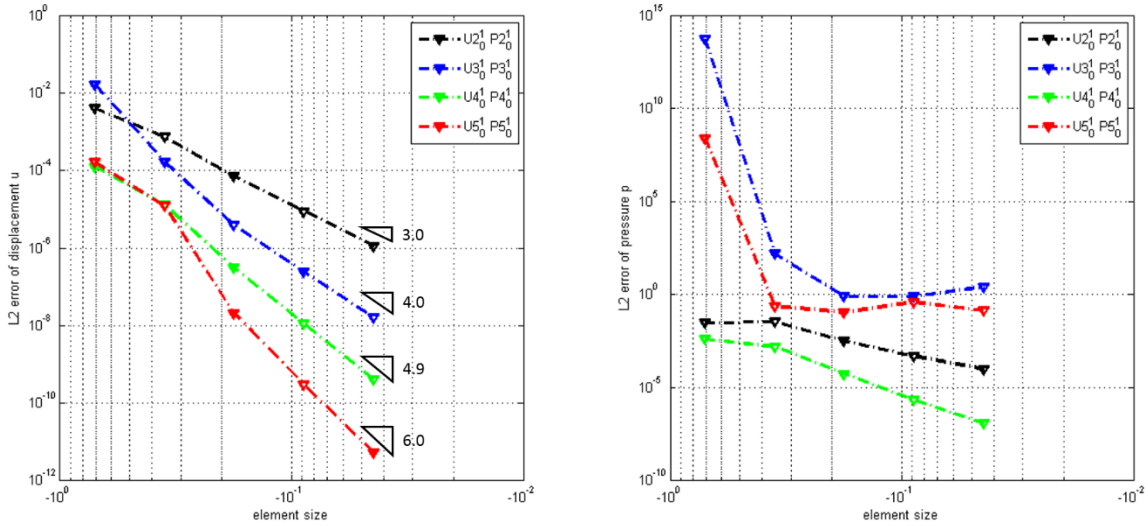


Figure 65. Convergence of L^2-norm of displacement error (Left) and pressure error (right) for $(\boldsymbol{u}\, \boldsymbol{p}_0^1, \boldsymbol{p}\, _0^1)$.

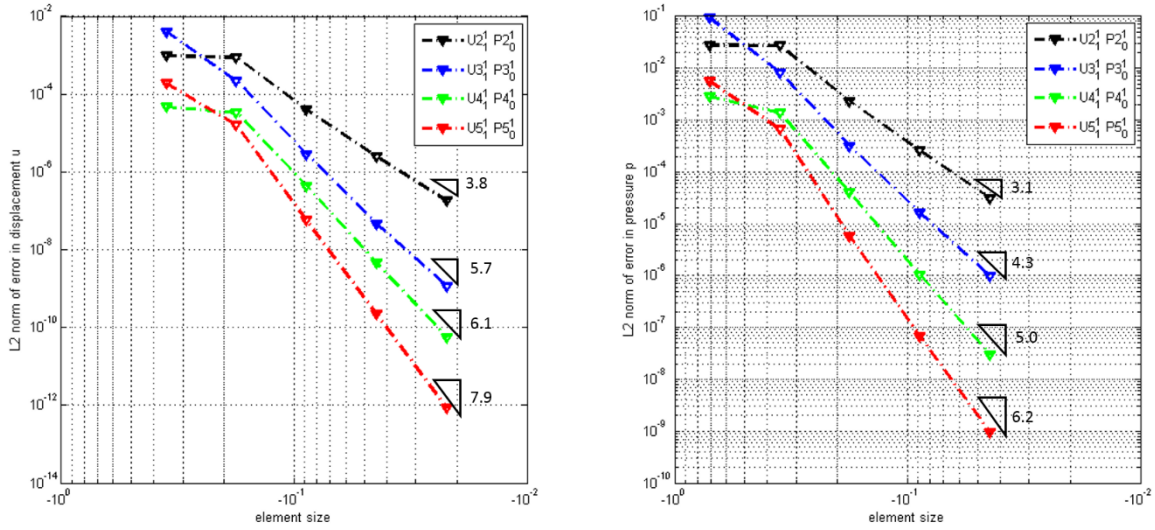


Figure 66. Convergence of L^2-norm of displacement error (Left) and pressure error (right) for $(\boldsymbol{u}\, \boldsymbol{p}_1^1, \boldsymbol{p}\, \boldsymbol{p}_0^1)$

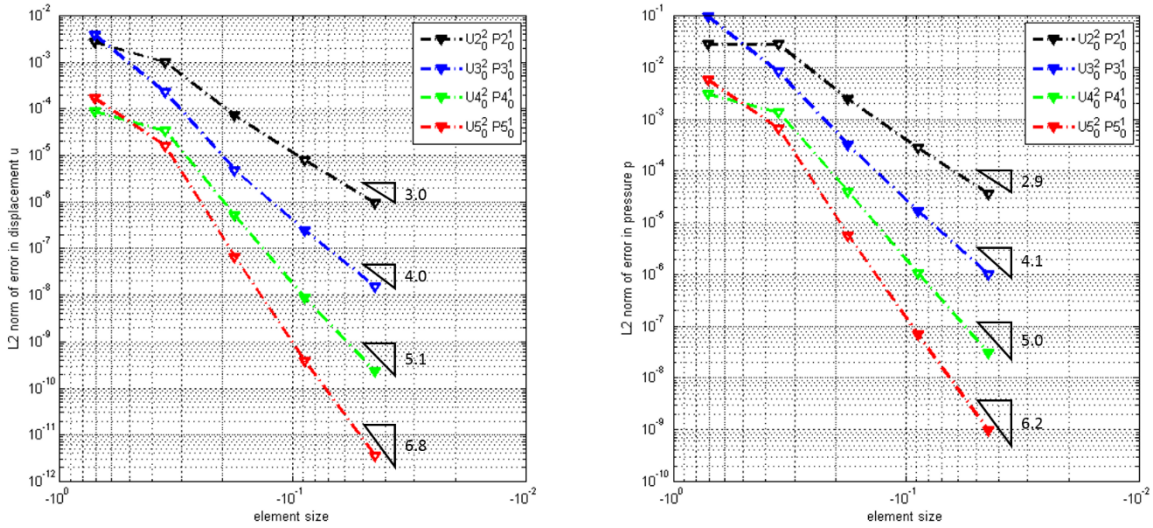


Figure 67. Convergence of L^2-norm of displacement error (Left) and pressure error (right) for $(\boldsymbol{u}\ \boldsymbol{p}_0^2, \boldsymbol{p}\ \boldsymbol{p}_0^1)$.

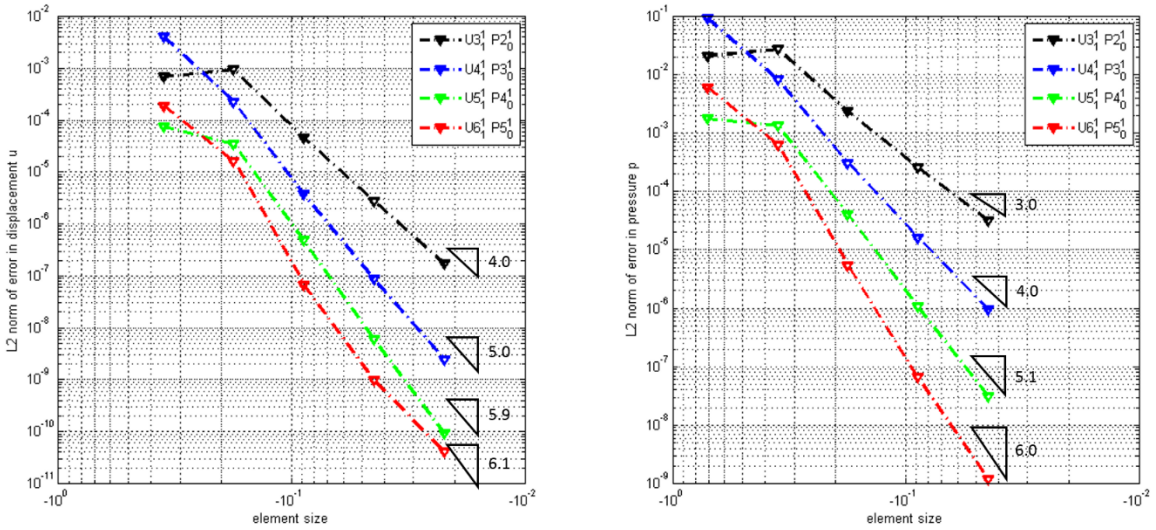


Figure 68. Convergence of L^2-norm of displacement error (left) and pressure error (right) for $(\boldsymbol{u}\ \boldsymbol{p} + \mathbf{1}_1^1, \boldsymbol{p}\ \boldsymbol{p}_0^1)$.

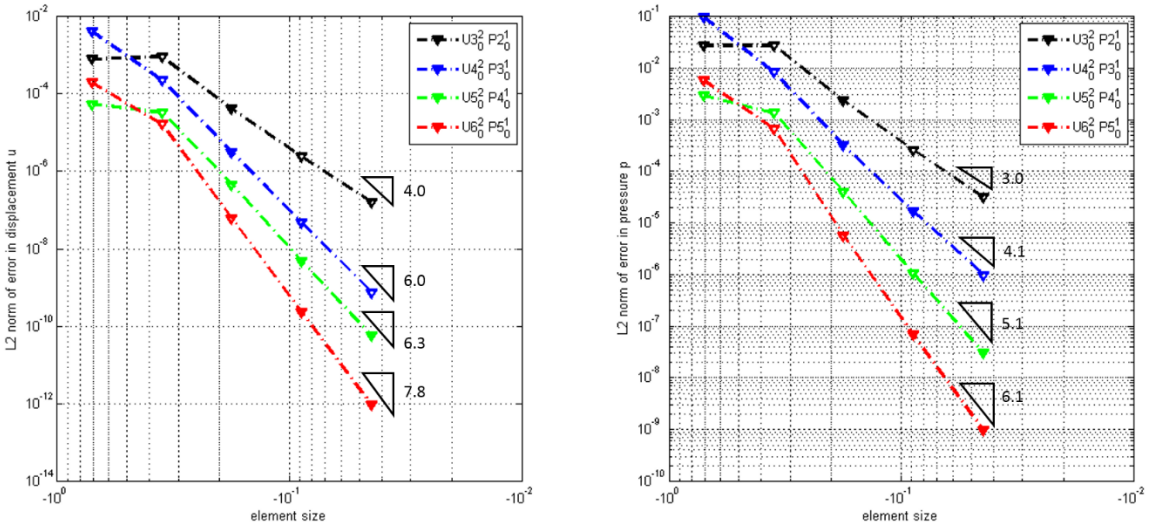


Figure 69. Convergence of L^2-norm of displacement error (left) and pressure error (right) for $(\boldsymbol{u}\ \boldsymbol{p} + \mathbf{1}_0^2, \boldsymbol{p}\ \boldsymbol{p}_0^1)$.

4.2.2. Wall-driven Square Cavity flow

We consider the wall-driven square cavity problem outlined in Figure 70. This problem is a classical problem for Stokes flow known as cavity flow problem. The square cavity is full of an incompressible material and a shear displacement is imposed at the upper wall moves. The three sides of the cavity are fixed. This problem which is highly confined is known to fail with unstable mixed formulation which can lead either to volumetric locking or checker board pressure instabilities depending of the formulation. The pressure singularities at the top corners make it a rather crude problem. In Figure 71**,** the contour plot of displacement components u_x and u_y, and displacement field (arrows field).

The profile of u_x along vertical axis and the profile of u_y along horizontal axis is respectively plotted in Figure 72 and Figure 73. We observe no spurious oscillation in numerical solution of displacement. Similar profile for displacement are obtained for discretization combinations $(\boldsymbol{u}2_1^1, p2_0^1)$, $(\boldsymbol{u}2_0^2, p2_0^1)$, $(\boldsymbol{u}2_0^1, p1_0^1)$ and $(\boldsymbol{u}2_0^1, p1_0^1)$. The results obtained are also similar to the one obtained for mixed finite elements displacement/pressure (Q2/Q2 known unstable and Q2/Q1 known stable).

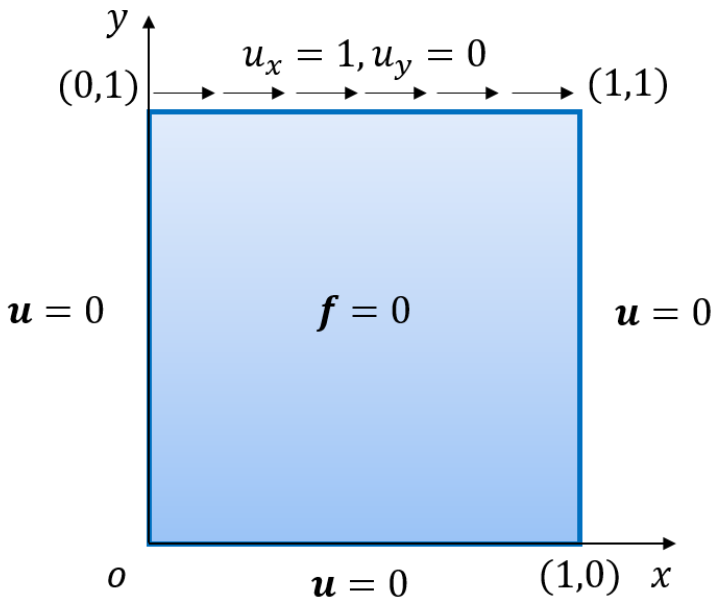


Figure 70. Cavity Flow: problem definition

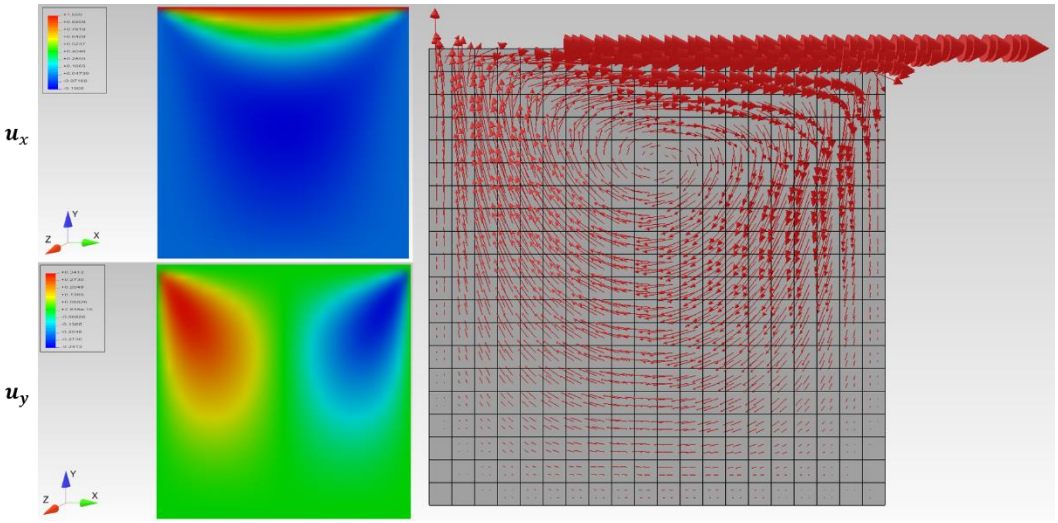


Figure 71. Cavity Flow: u_x, u_y distributions and stream field.

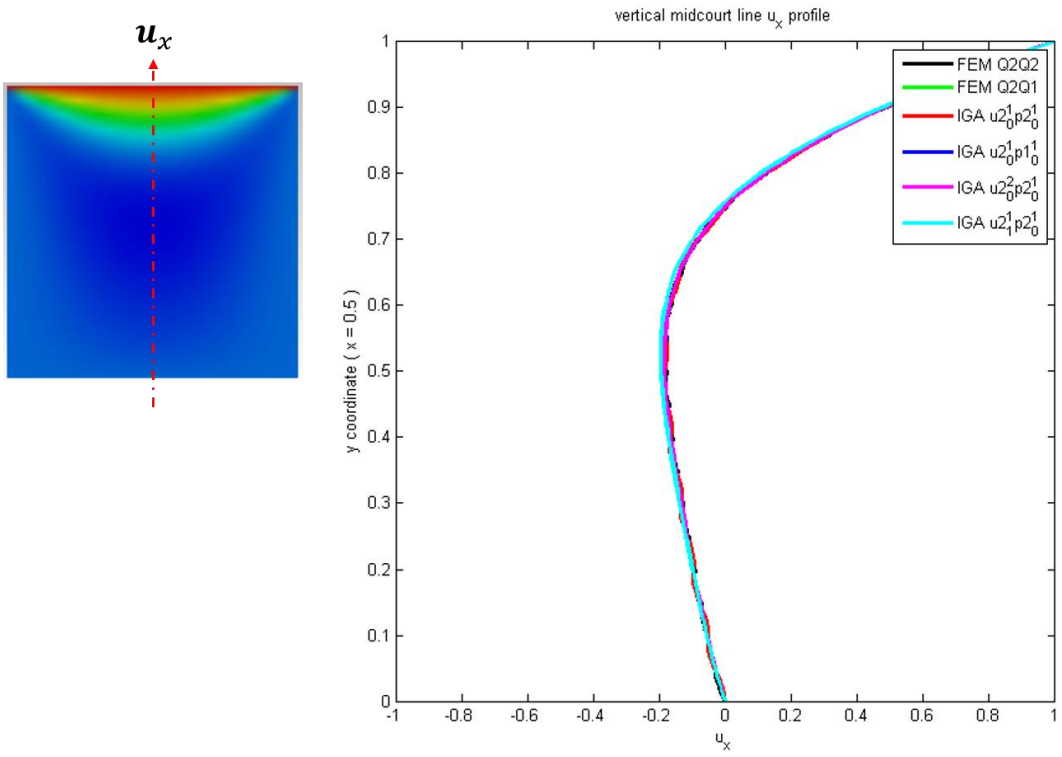


Figure 72. Cavity Flow: u_x profiles along vertical midline

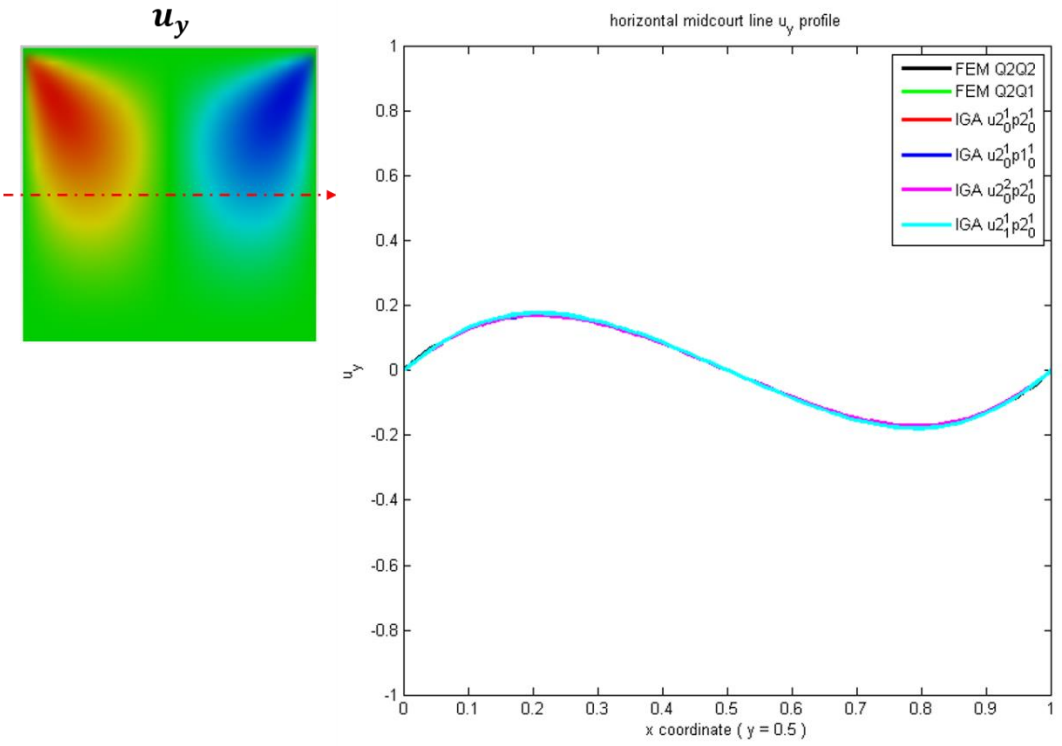


Figure 73. Cavity Flow: u_y profiles along horizontal midline

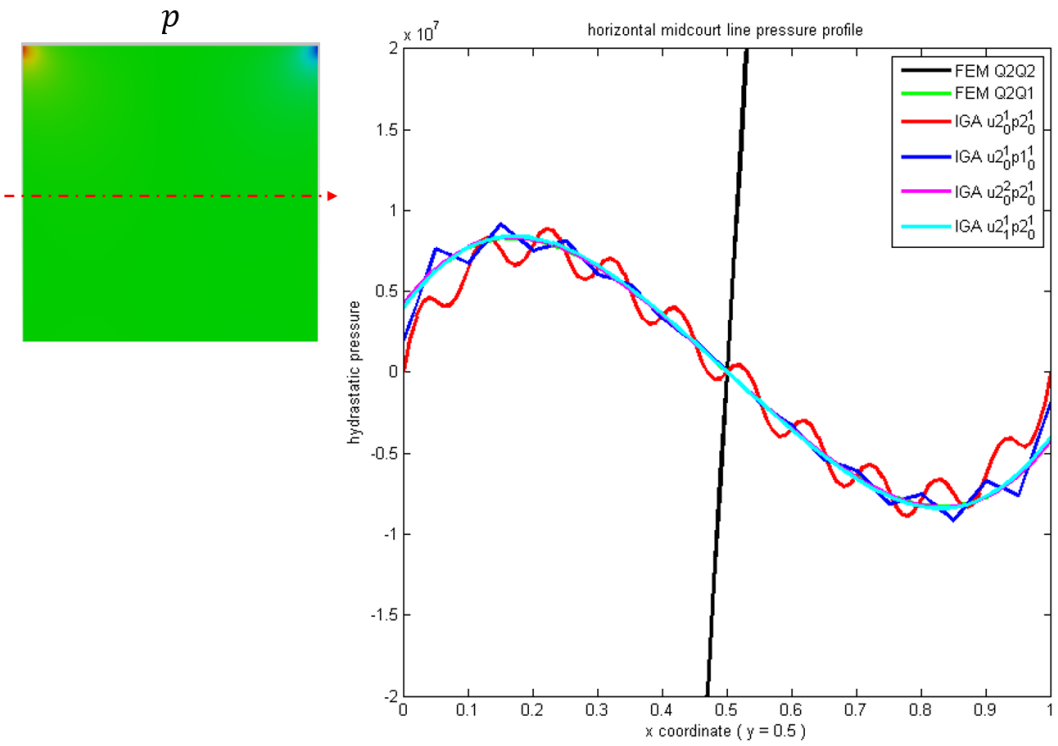


Figure 74. Cavity Flow: pressure profiles along horizontal midline

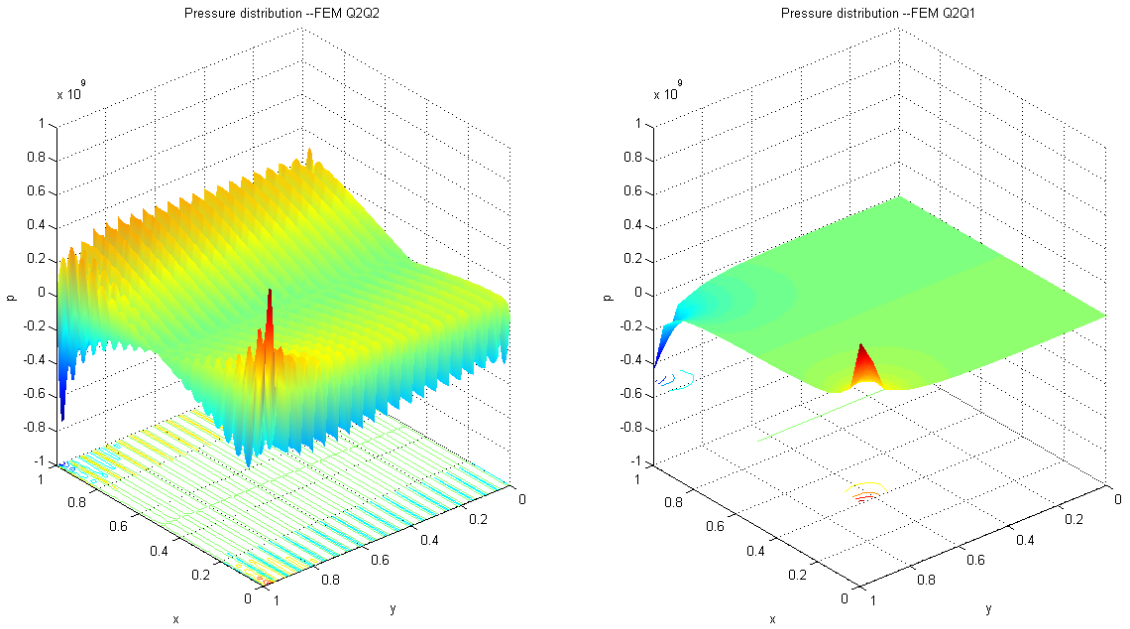


Figure 75. Cavity Flow: pressure surface plots for FEM $\boldsymbol{Q2Q2}$ and FEM $\boldsymbol{Q2Q1}$

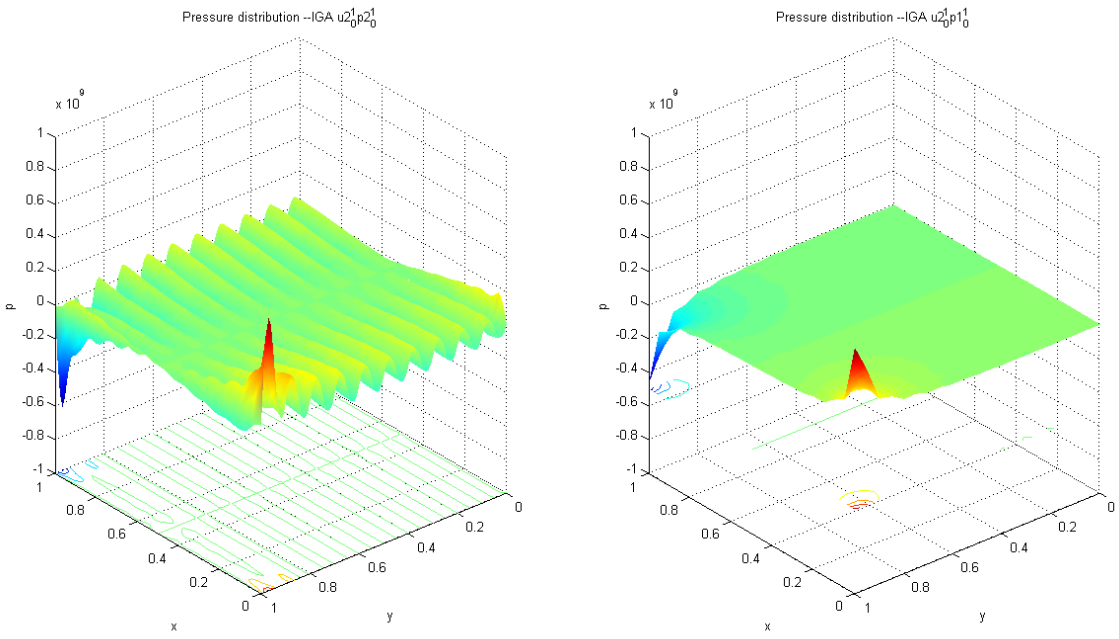


Figure 76. Cavity Flow: pressure surface plots for IGA $\boldsymbol{u2_0^1p2_0^1}$ and IGA $\boldsymbol{u2_0^1p1_0^1}$

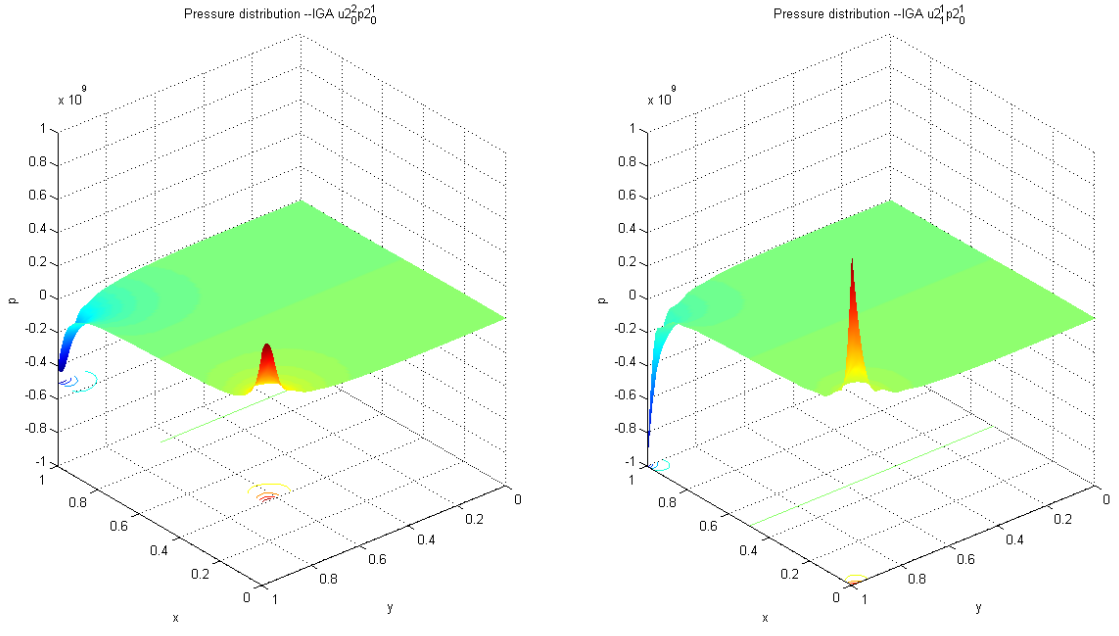


Figure 77. Cavity Flow: pressure surface plots for IGA $\boldsymbol{u2_0^2p2_0^1}$ and IGA $\boldsymbol{u2_1^1p2_0^1}$

Oscillations for pressure are observed in Figure 75 and Figure 76 for finite elements Q2/Q2 and IGA $(\boldsymbol{u}\ 2_0^1, \boldsymbol{p}\ 2_0^1)$. In this test, IGA elements $(\boldsymbol{u}\ 2_0^2, \boldsymbol{p}\ 2_0^1)$, subdivided element $(\boldsymbol{u}\ 2_1^1, \boldsymbol{p}\ 2_0^1)$ and IGA element $(\boldsymbol{u}\ 2_0^1, \boldsymbol{p}\ 1_0^1)$ offer nice plots for pressure. In Figure 74, we observe no oscillation for discretizations FEM $Q2/Q1$. Note that in Figure 76 and Figure 77 the pressure plot elevation is more diffusive for $(\boldsymbol{u}2_0^2, \boldsymbol{p}2_0^1)$ and $(\boldsymbol{u}2_0^2, \boldsymbol{p}1_0^1)$ than for $(\boldsymbol{u}2_1^1, \boldsymbol{p}2_0^1)$ for which the pressure elevation at both top corners is sharper.

4.2.3. Cook's Membrane – infinitesimal strain

In this subsection, the strain projection methods (see Elguedj et al [46]) and the mixed formulation methods are applied to treat incompressible problems (see The main drawback of the selective and reduced integration scheme is that the generalization to anisotropic formulations, and the strongly nonlinear applications we target may exhibit strong anisotropy (non-symmetric anisotropic tangent operator). Three fields formulation are quite heavy in term of complexity and computational cost. We prefer to address simpler formulations from a practical point of view, to be able to more easily take into account additional physics. Simple two-field formulations have proven to be accurate to deal with incompressibility. In the following, we will systematically compare several elements focusing on IGA mixed elements of type $(\boldsymbol{u}\ p_1^1.\boldsymbol{p}\ p_0^1)$ $(\boldsymbol{u}\ p_0^2.\boldsymbol{p}\ p_0^1)$ $(\boldsymbol{u}\ p_1^1.\boldsymbol{p}\ p-1_0^1)$ $(\boldsymbol{u}\ p_0^2.\boldsymbol{p}\ p-1_0^1)$ $(\boldsymbol{u}\ p_0^1.\boldsymbol{p}-\boldsymbol{1}\ p_0^1)$ at small and finite strain. We include in this comparative study the strain projection method ($\bar{B}$ and $\bar{F}$). In Table 4-2, the method that will be tested and compared at small and finite strain are summarized.

Table 4-2) within the framework presented in the previous paragraph. Cook's membrane is a standard reference problem to evaluate the quality for nearly incompressible models with dominant bending and shearing deformations (for finite element method see e.g Cook [259], Kasper et Taylor [260] and for IGA see Elguedj [46], Kadapa [62] and Mathisen [261]). The material parameters are given in Figure 78 along with its geometry and boundary conditions. A clamped tapered panel is subjected to a uniform shear load on its right side. The quantity of interest is the vertical displacement of top right corner point $\boldsymbol{C}$. Analysis are performed on successively and uniformly h-refined meshes, for different

patches combinations of NURBS, including standard displacement element, and for $\bar{\boldsymbol{B}}$ projection formulation and mixed formulations proposed in Elguedj et al [46].

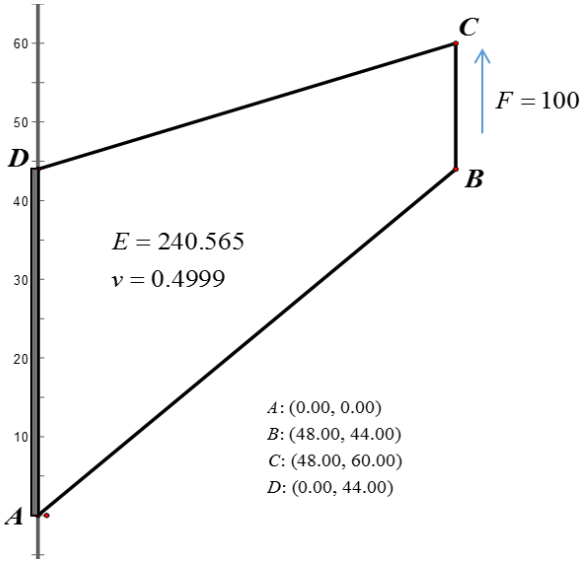


Figure 78. Problem definition for Cook's Membrane at infinitesimal strain regime.

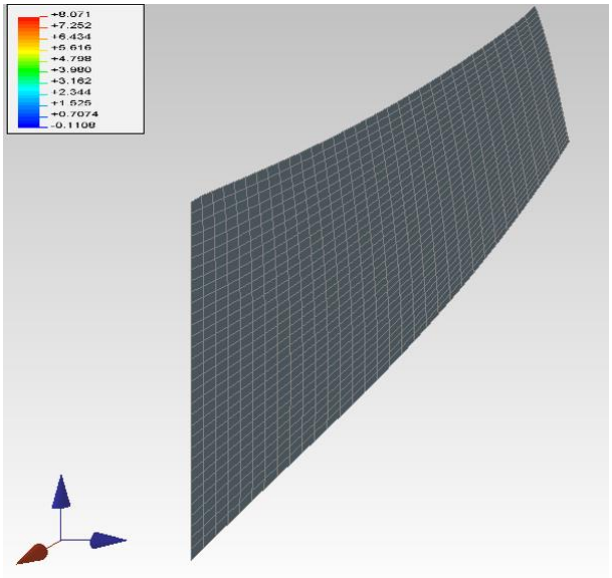


Figure 79. Cook's Membrane at infinitesimal strain regime: Deformed mesh.

The amplified deformed mesh panel is shown in Figure 79. In Figure 80, the standard displacement formulation and the $\bar{\boldsymbol{B}}$ projection method are compared. Note that the NURBS element of degree 1 is equivalent to the $Q1$ Lagrange finite element. It suffers from volumetric locking. If p or k refinement is applied, the result can be improved for the

NURBS based isogeometric element. However, the volumetric locking stays bothering for low degree of shape functions. The $\bar{\boldsymbol{B}}$ projection method allows to significantly improve the displacement formulation's performance facing incompressibility constraints. Even with a lower degree projection element such as $Q1/Q0$, the locking issue has been notably alleviated. With a rather coarse mesh of 16×16 elements, the $\bar{\boldsymbol{B}}$ projection formulation of order p (p from 1 to 4) exhibit solutions close enough to converged position. Similar outcomes are shown and $\bar{\boldsymbol{B}}$ projection element superior is superior to standard displacement elements. Moreover, since the projection allows to reduce the degree of freedom of problem, the advantage of $\bar{\boldsymbol{B}}$ projection method becomes more evident.

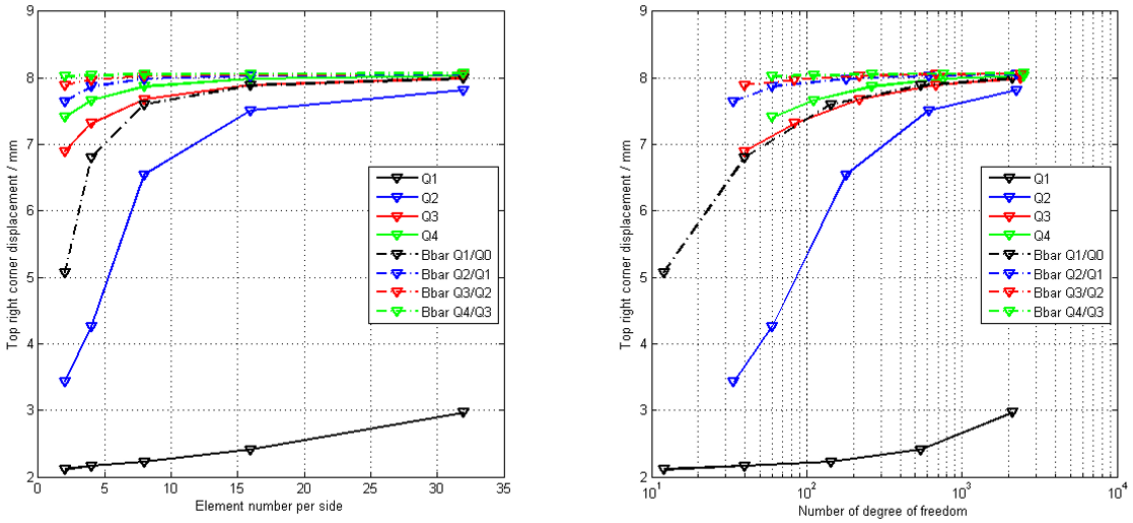


Figure 80. Cook's Membrane at infinitesimal strain regime: Vertical displacement of top right corner with respect to element number per side or degree of freedom for - $\bar{\boldsymbol{B}}$ projection method and Q1, Q2, Q3 and Q4 displacement elements.

The performances of mixed formulation methods with discretization $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ and $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$, for degree 2, 3, 4 can be compared in Figure 81. The results proved that these two combinations efficiently handle incompressible linear elasticity as expected. Moreover, the discretization of type $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ seems to be more performant than $(\boldsymbol{u}\, p_0^2, \boldsymbol{p}\, p_0^1)$ on this test, for a given number of elements per side or the total number degrees of freedom. This can partly be explained by the fact that the discretization combination with subdivision $(\boldsymbol{u}\, p_1^1, \boldsymbol{p}\, p_0^1)$ has a coarser mesh for pressure fields, and thus, have a smaller number of unknowns.

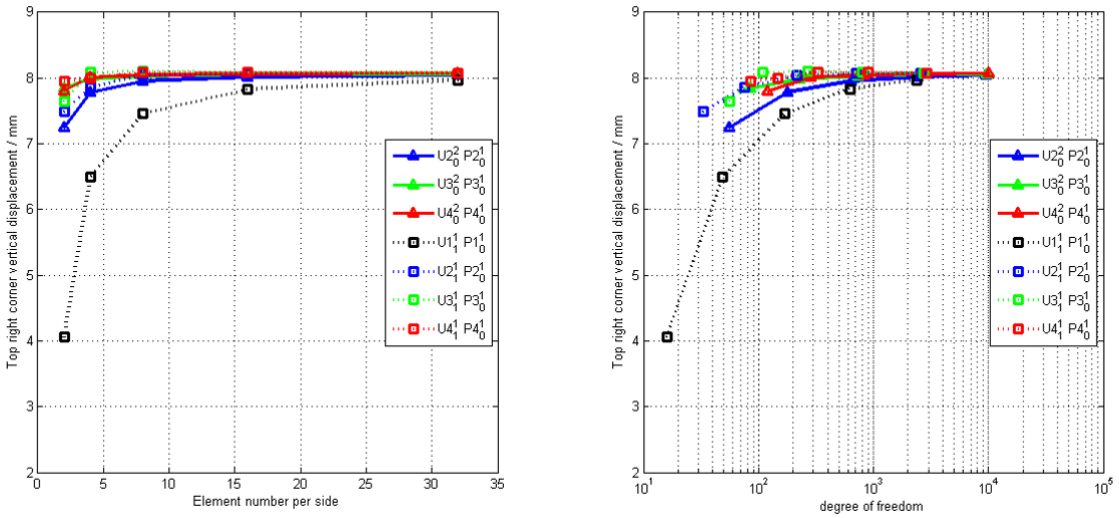


Figure 81. Cook's Membrane at infinitesimal strain regime: Vertical displacement of top right corner with respect to element number per side or degree of freedom for - mixed formulation with discretizations $(\boldsymbol{u}\ \boldsymbol{p}_0^2, \boldsymbol{p}\ \boldsymbol{p}_0^1)$ and $(\boldsymbol{u}\ \boldsymbol{p}_1^1, \boldsymbol{p}\ \boldsymbol{p}_0^1)$.

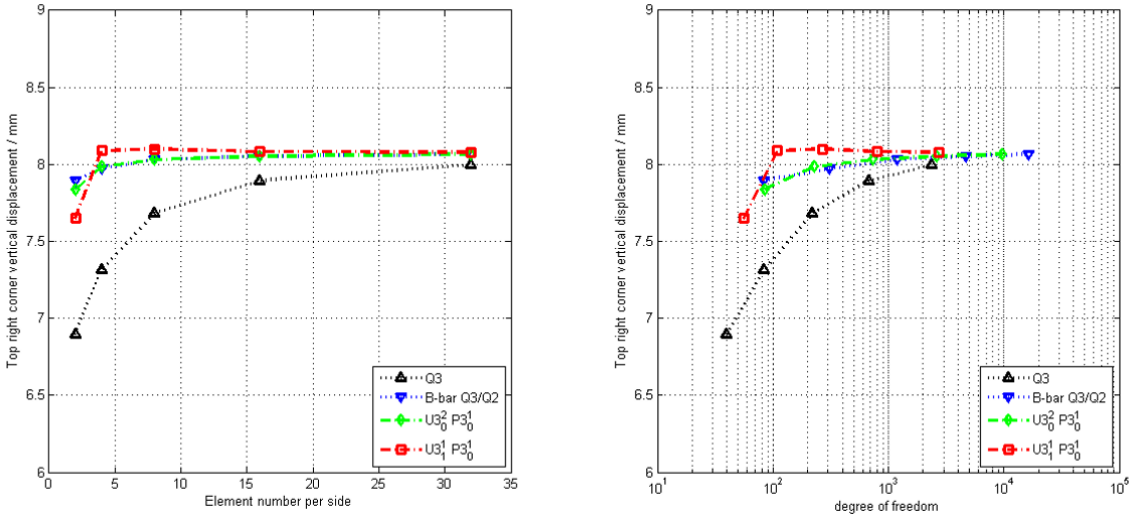


Figure 82. Cook's Membrane at infinitesimal strain regime: Vertical displacement of top right corner with respect to element number per side or degree of freedom for - cubic displacement formulation, cubic $\bar{\boldsymbol{B}}$ projection method and mixed formulation with discretizations $(\boldsymbol{u}\ \boldsymbol{p}_0^2, \boldsymbol{p}\ \boldsymbol{p}_0^1)$ & $(\boldsymbol{u}\ \boldsymbol{p}_1^1, \boldsymbol{p}\ \boldsymbol{p}_0^1)$.

At last, we compared the different formulations with cubic interpolations *as* shown in Figure 82. The standard Q3 displacement element is locking whereas locking is avoided

for coarse mesh with $\bar{B}$ Q3/Q2 (projection on a Q2 mesh) and $(\boldsymbol{u}\ 3_0^2, \boldsymbol{p}\ 3_0^1)$ & $(\boldsymbol{u}\ 3_1^1, \boldsymbol{p}\ 3_0^1)$. On this test, the IGA element $(\boldsymbol{u}3_1^1, \boldsymbol{p}3_0^1)$ seems a little bit performant.

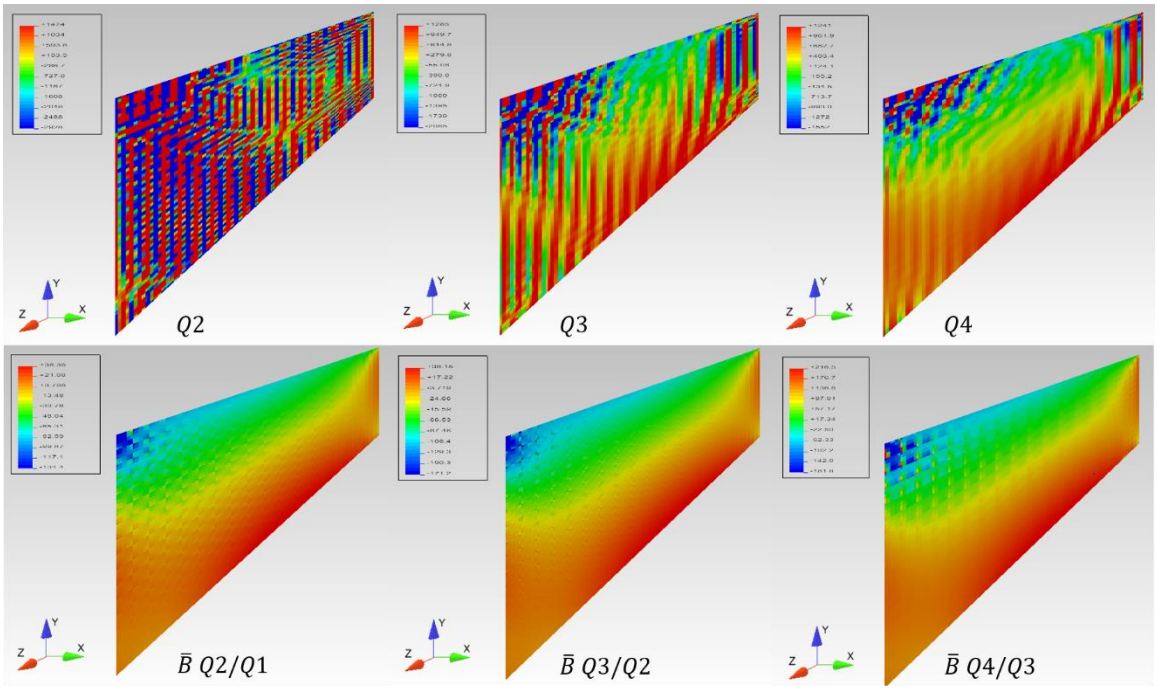


Figure 83. Cook's Membrane at infinitesimal strain regime: trace of stress for standard displacement element (top) and $\bar{\boldsymbol{B}}$ projection elements (bottom).

At last, in Figure 84, we focus on the contour plots of hydrostatic pressure for mixed formulations. In Figure 83, we focus for $\bar{\boldsymbol{B}}$ projection method, on the contour plot of the trace of the Cauchy stress $tr(\sigma)$. We observe that the standard quadratic NURBS element $(\boldsymbol{u}\ 2_0^1, \boldsymbol{p}\ 2_0^1)$ and $(\boldsymbol{u}\ 2_0^1, \boldsymbol{p}\ 1_0^1)$ exhibit spurious oscillation for stress trace. However, these spurious oscillations have been slightly alleviated with a higher degree. The trace obtained by $\bar{\boldsymbol{B}}$ projection method appears to be correct except for some tiny local perturbations. And these undesired checkerboard-like effects can also be relieved with a high degree interpolation for NURBS. Note that in Figure 84 the mixed formulation with equal order interpolation for $(\boldsymbol{u}, p)$ exhibit spurious oscillations. Even the combination of type $(\boldsymbol{u}\ p+1_0^1, \boldsymbol{p}\ p_0^1)$ do not work properly.

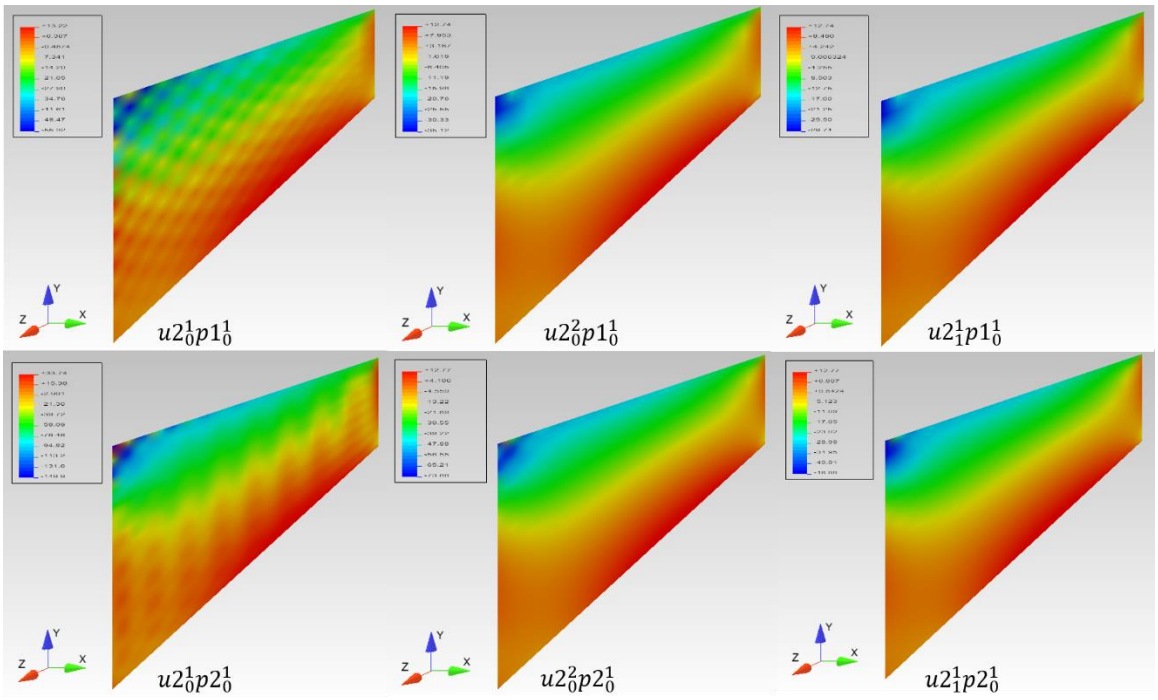


Figure 84. Cook's Membrane at infinitesimal strain regime: hydrostatic pressure for mixed formulations.

To conclude this section, the better choice for this test seems are elements of type $(\boldsymbol{u}2_0^2, \boldsymbol{p}1_0^1)$, $(\boldsymbol{u}2_1^1, \boldsymbol{p}1_0^1)$ and for equal order interpolation $(\boldsymbol{u}2_0^2, \boldsymbol{p}2_0^1)$, $(\boldsymbol{u}2_1^1, \boldsymbol{p}2_0^1)$.

4.3. A numerical comparison of stability at large strains

In this section, we consider, two-fields mixed formulations for hyperelastic materials. In the following, the free energy is decomposed into volumetric and isochoric contributions:

$$\psi(\boldsymbol{u}) = \psi_{iso}\big(\bar{I}_1(\boldsymbol{u}), \bar{I}_2(\boldsymbol{u})\big) + \frac{1}{2}\kappa(J-1)^2 \tag{4-10}$$

Varying the expression of $\psi_{iso}\big(\bar{I}_1(\boldsymbol{u}), \bar{I}_2(\boldsymbol{u})\big)$ allows us to adopt different types of hyperelastic models, for instance Neo-Hookean or Mooney-Rivilin.

This expression allows us to adopt different types of hyperelastic models, e.g. Neo-Hookean or Mooney-Rivilin depending on the form of $\psi_{iso}\big(\bar{I}_1(\boldsymbol{u}), \bar{I}_2(\boldsymbol{u})\big)$. The mixed formulation is deduced by applying the Legendre transformation to the volumetric contribution $\psi_{vol}(J) = \frac{1}{2}\kappa(J-1)^2$. As matter of fact, the variable $p = \frac{d\psi_{vol}(J)}{dJ}$ represents

a Lagrange multiplier that can be identified as the hydrostatic pressure ($\frac{1}{3}$ trace of the Cauchy stress) and the free energy states as follows:

$$\psi(\boldsymbol{u}, p) = \psi_{iso}\big(\bar{I}_1(\boldsymbol{u}), \bar{I}_2(\boldsymbol{u})\big) + p(J-1) - \frac{1}{2}\frac{p^2}{\kappa} \tag{4-11}$$

The potential energy of material domain can be computed by integrating the free energy density on the reference domain. This expression for the potential can be considered as a particular case of the Hellinger-Reissner variational principle:

$$\begin{aligned} W(\boldsymbol{u}, p) = \int_{\Omega_0} (\psi_{iso}\big(\bar{I}_1(\boldsymbol{u}), \bar{I}_2(\boldsymbol{u})\big) + p(J-1) - \frac{1}{2}\frac{p^2}{\kappa})\mathrm{dv} \\ - \int_{\Omega_0} \boldsymbol{B} \cdot \delta\boldsymbol{u} dV - \int_{\partial\Omega_{0\sigma}} \overline{\boldsymbol{T}} \cdot \delta\boldsymbol{u}\, dS \end{aligned} \tag{4-12}$$

We minimize the potential energy with respect to the variations $(\delta\boldsymbol{u}, \delta p)$. The variational formulation states as follows:

Find $\boldsymbol{u} \in V_u$ that for all $\delta\boldsymbol{u} \in V_u^0$, and $p \in V_p$ that for all $\delta p \in V_p^0$,

$$\begin{cases} D_{\delta\boldsymbol{u}} W(\boldsymbol{u}, p) = \int_{\Omega_0} \left(2\frac{\partial \psi_{iso}}{\partial \boldsymbol{C}} + Jp\boldsymbol{C}^{-1}\right) : \delta\boldsymbol{E}\, dV - \delta W_{ext} = 0 \\ D_{\delta p} W(\boldsymbol{u}, p) = \int_{\Omega_0} \delta p \left(J - 1 - \frac{p}{\kappa}\right) dV = 0 \end{cases} \tag{4-13}$$

This method is referred as a Perturbed Lagrange-Multiplier method. The last term of the potential energy can be seen as a penalty term for which the penalty coefficient κ measures the stiffness of volumetric deformation. When $\kappa \to \infty$, a perfectly incompressible model is obtained which correspond to the classical Lagrange-Multiplier method. Alternatively, the level of compressibility of a perturbed Lagrange-Multiplier model is fully determined by the volumetric contribution depending on J (determinant of the gradient deformation). This part of free energy is usually formulated by:

$$\psi_{vol}(J) = \kappa \mathcal{G}(J) \tag{4-14}$$

where $\mathcal{G}(J)$ denotes the incompressibility function which has to be a convex function for which the minimum is set at $J = 1$ i.e.:

$$\mathcal{G}(J) = 0 \text{ if and only if } J = 1 \tag{4-15}$$

For example, the free energy given in equation (4-10), $\mathcal{G}(J) = \frac{1}{2}(J-1)^2$ obviously meet the above requirements. Many forms of incompressibility functions exist in the literature (even in commercial codes such as ANSYS, ABAQUS). We list the one we have implemented in the code in Table 4-3. The shapes of these different model are plot in in Figure 85. For the model of Ogden, we have chosen the parameter $\beta = \{1,2,3\}$.

Table 4-3. Incompressibility functions

Model	$\mathcal{G}(J)$	$d\mathcal{G}(J)/dJ$	$d^2\mathcal{G}(J)/dJ^2$	Reference
1	$\frac{1}{2}(J-1)^2$	$J-1$	1	
2	$\frac{1}{4}(J^2-1-2\ln J)$	$\frac{1}{2}\left(J-\frac{1}{J}\right)$	$\frac{1}{2}\left(1+\frac{1}{J^2}\right)$	Simo&Miehe [262]
3	$\frac{1}{\beta^2}\left(\beta \ln J + \frac{1}{J^\beta} - 1\right)$	$\frac{1}{\beta}\left(\frac{1}{J} - \frac{1}{J^{\beta+1}}\right)$	$\frac{1}{J^{\beta+2}}\left(1+\frac{1}{\beta}-\frac{J^\beta}{\beta}\right)$	Ogden [263]

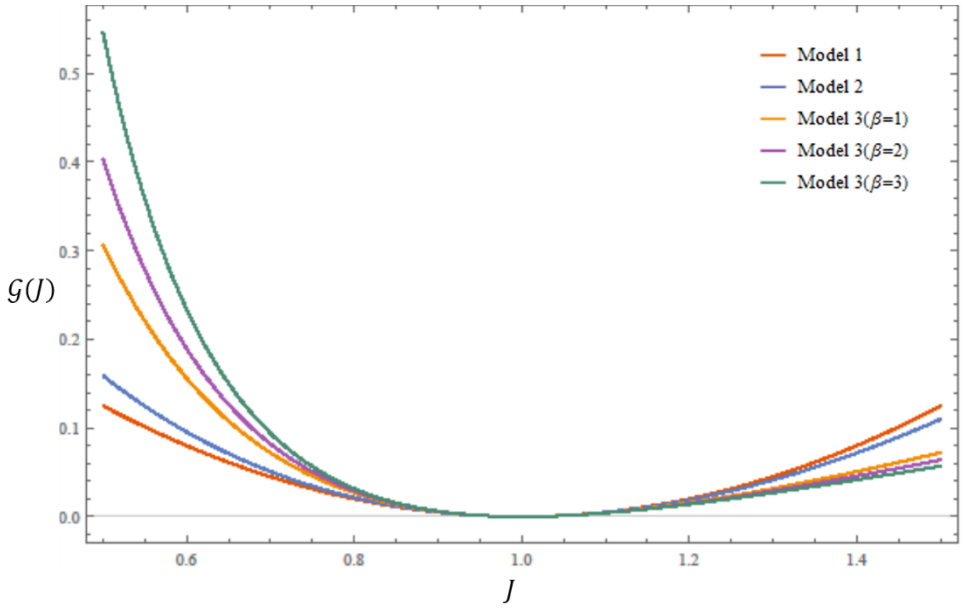


Figure 85. Incompressibility Functions $\mathcal{G}(J)/J$

For these incompressibility functions, the Legendre transformation is quite difficult to apply. Thus, we enforce the constraints coming from the incompressibility function in a weak sense. It consists in weakly imposing the following equality into the free energy (4-10).

$$p = \frac{d\psi_{vol}(J)}{dJ} = \kappa \frac{d\mathcal{G}(J)}{dJ} \tag{4-16}$$

As matter of fact, the alternative mixed formulation can be written as:

Find $\boldsymbol{u} \in V_u$ that for all $\delta\boldsymbol{u} \in V_u^0$, and $p \in V_p$ that for all $\delta p \in V_p^0$,

$$\begin{cases} \int_{\Omega_0} \left(2\frac{\partial \psi_{iso}}{\partial \boldsymbol{C}} + Jp\boldsymbol{C}^{-1}\right) : \delta\boldsymbol{E}\, dV - \delta W_{ext} = 0 \\ \int_{\Omega_0} \delta p \left(\frac{d\mathcal{G}(J)}{dJ} - \frac{p}{\kappa}\right) dV = 0 \end{cases} \tag{4-17}$$

It is important to note that the tangent operator obtained from the system is no longer symmetric.

4.3.1. Cook's Membrane – finite strain

The Cooks' membrane problem has been defined in section 4.2.3 (see Figure 78). The loading is chosen in order to reach the regime of finite strain. The aim of this test is to evaluate the performance of both the $\bar{\boldsymbol{F}}$ projection method proposed by Elguedj et al [46] and the mixed formulation method in the nonlinear case. The material is hyperelastic and a Neo-Hookean model is adopted. The free energy ψ is decoupled into isochoric and volumetric parts such as:

$$\psi(\bar{I}_1, J) = \frac{1}{2}\mu(\bar{I}_1 - 3) - \mu \ln J + \frac{1}{4}\kappa(J^2 - 1 - 2\ln J) \tag{4-18}$$

The parameters of the model are $\mu = 80.1938MPa$ and $\kappa = 400889.806MPa$. We focus on the vertical displacement of the top right corner of the plate and the convergence is studied with respect to both the number of elements per edge and the total number of degrees of freedom.

In Figure 86, the vertical displacement of the top right corner is given for the $\bar{F}$ projection method for various degrees of basis function. We can see that the low-order based projection element still suffers from volumetric locking when the mesh is not fine enough. But with h-refinement, the $\bar{F}$ projection elements rapidly converge to the final position. The vertical displacement results for the mixed formulations with discretization

combinations of type $(\boldsymbol{u}\ 3_0^2, \boldsymbol{p}\ 3_0^1)$ and $(\boldsymbol{u}\ 3_1^1, \boldsymbol{p}\ 3_0^1)$ are shown in Figure 87. We can see that mixed formulation substantially improves the accuracy of the results as expected and already observed at small strains. Higher order NURBS elements with mixed formulation gives almost the converged solution even for coarse meshes. Compared to the $\bar{F}$ projection method the rate of convergence for the vertical displacement seems to be better. In order to confirm this observation, we plot on the same diagram the results for cubic interpolations for both the $\bar{F}$ projection method and the mixed formulation (interpolation $(\boldsymbol{u}3_1^1, p3_0^1)$ and $(\boldsymbol{u}3_0^2, p3_0^1)$) in Figure 88. At a given degree for interpolation, the mixed formulations seem to provide a better solution than $\bar{F}$ projection method with respect the number of elements and the total number of nodes. The mixed formulation with the subdivided element for displacement of type $(\boldsymbol{u}3_1^1, p3_0^1)$ is better than the $(\boldsymbol{u}\ 3_0^2, \boldsymbol{p}\ 3_0^1)$.

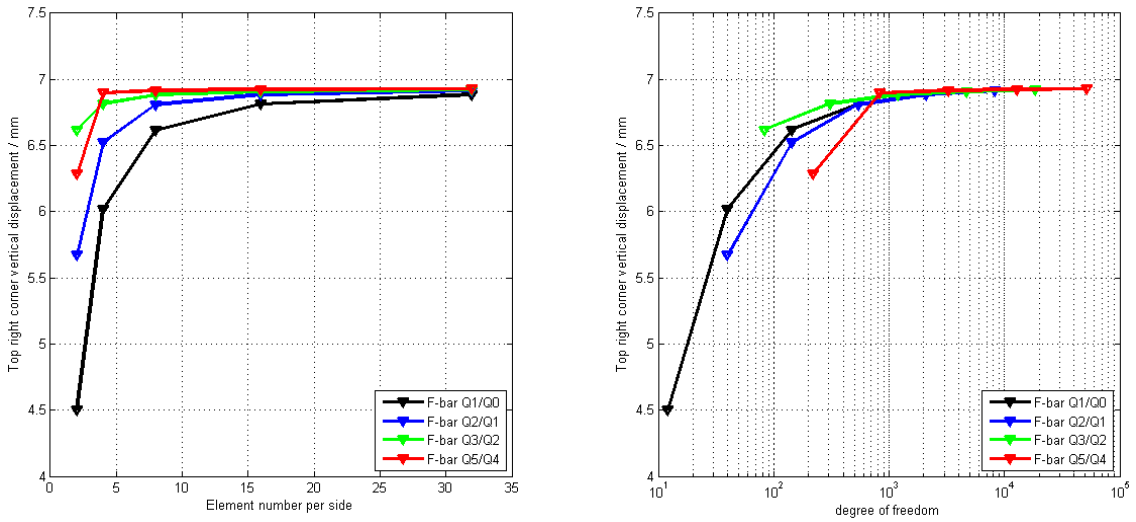


Figure 86. Cook's Membrane at finite strain regime: Vertical displacement of top right corner with respect to element number per side or degree of freedom for - $\bar{\boldsymbol{F}}$ projection method.

An important feature for the application in elastomers we are targeting to obtain good quality of stresses and/or pressure parameter. Figure 89 gives the color maps for the trace of the Cauchy stress $tr(\sigma)$ post-processed from the solutions (standard quadrilateral Q1 element and $\bar{F}$ projection method with $Q1/Q0$). The standard displacement method with

Q1 fails. The $\bar{F}\ Q1/Q0$ gives a globally reasonable result but some tiny local oscillations remain.

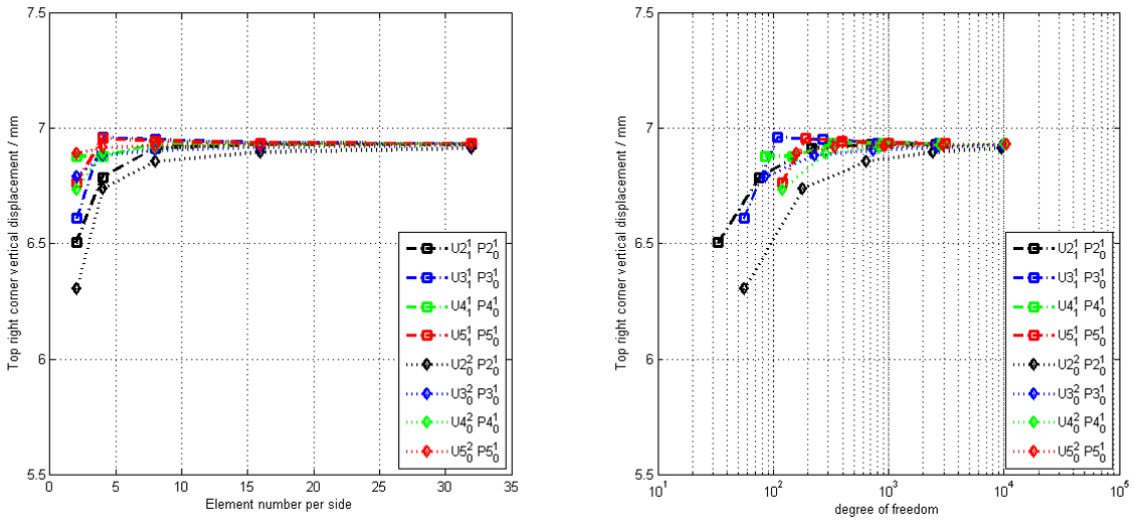


Figure 87. Cook's Membrane at finite strain regime: Vertical displacement of top right corner with respect to element number per side or degree of freedom for - mixed formulation with discretizations $(\boldsymbol{u}\ \boldsymbol{p}_{\boldsymbol{0}}^{\boldsymbol{2}}, \boldsymbol{p}\ \boldsymbol{p}_{\boldsymbol{0}}^{\boldsymbol{1}})$ and $(\boldsymbol{u}\ \boldsymbol{p}_{\boldsymbol{1}}^{\boldsymbol{1}}, \boldsymbol{p}\ \boldsymbol{p}_{\boldsymbol{0}}^{\boldsymbol{1}})$.

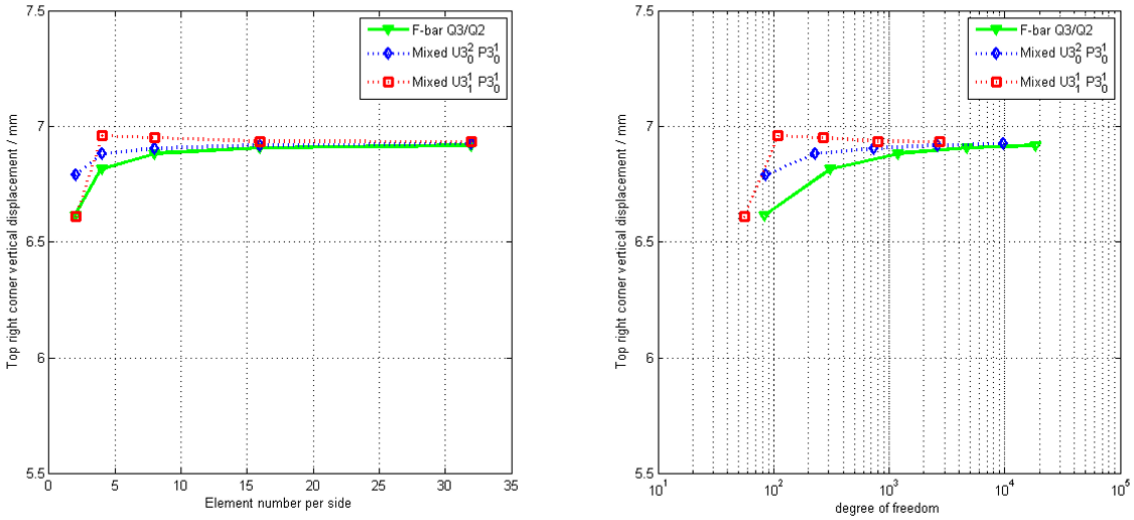


Figure 88. Cook's Membrane at finite strain regime: Vertical displacement of top right corner with respect to element number per side or degree of freedom for - cubic $\bar{F}$ projection method and mixed formulation with discretizations $(\boldsymbol{u3}_{\boldsymbol{0}}^{\boldsymbol{2}}, \boldsymbol{p3}_{\boldsymbol{0}}^{\boldsymbol{1}})$ & $(\boldsymbol{u3}_{\boldsymbol{1}}^{\boldsymbol{1}}, \boldsymbol{p3}_{\boldsymbol{0}}^{\boldsymbol{1}})$.

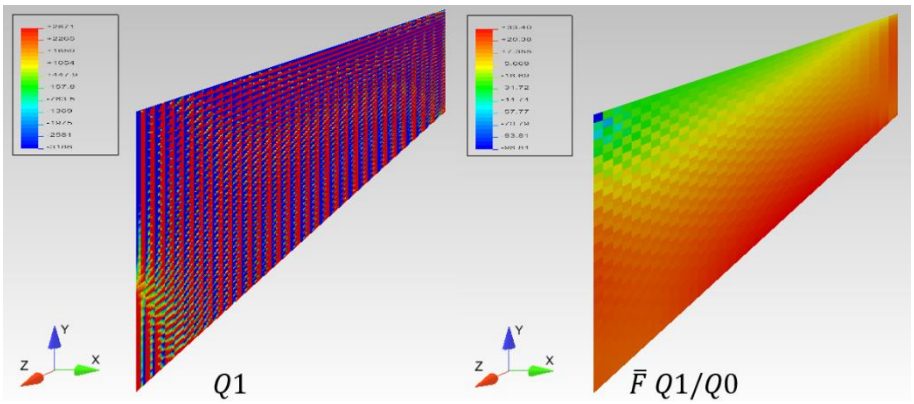


Figure 89. Cook's Membrane at finite strain regime: trace of stress for standard displacement element (top) and $\overline{\boldsymbol{F}}$ projection elements (bottom).

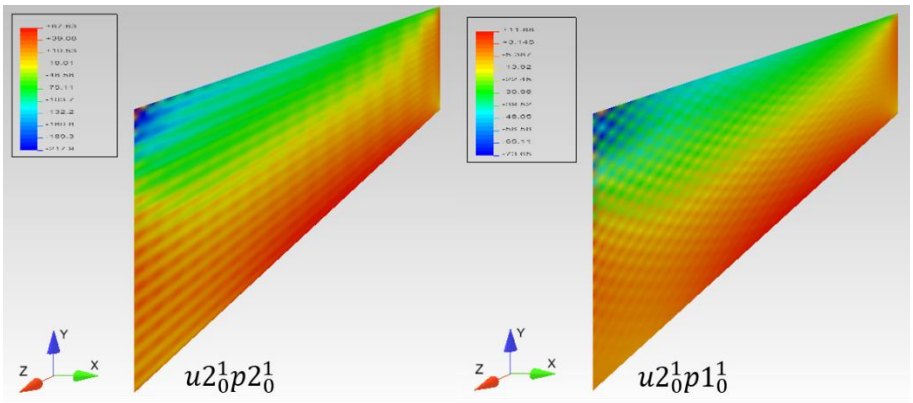


Figure 90. Cook's Membrane at finite strain regime: hydrostatic pressure for mixed formulations with instable discretizations.

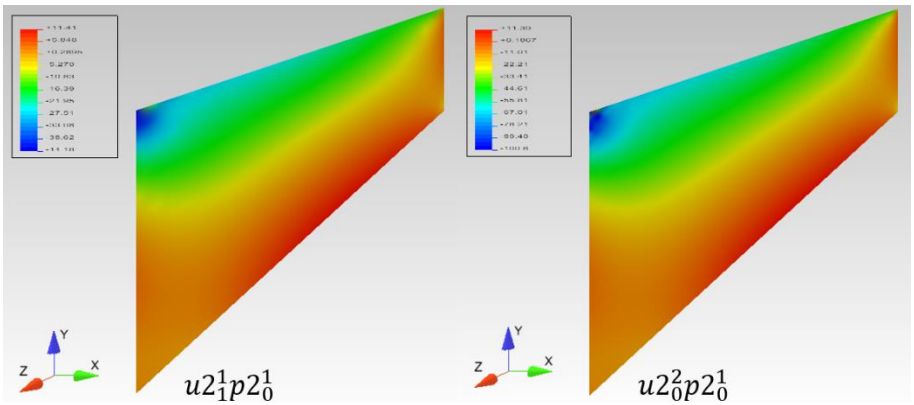


Figure 91. Cook's Membrane at finite strain regime: hydrostatic pressure for mixed formulations with stable discretizations

The mixed formulation solutions based on four discretizations combinations $(\boldsymbol{u}2_0^1, \boldsymbol{p}2_0^1)$, $(\boldsymbol{u}2_0^1, \boldsymbol{p}1_0^1)$ are given in Figure 90. The results exhibit of course oscillations but are much better than the one for the Q1 element. The mixed formulation solutions based on four discretizations combinations $(\boldsymbol{u}2_1^1, \boldsymbol{p}2_0^1)$ and $(\boldsymbol{u}2_0^2, \boldsymbol{p}2_0^1)$ are given in Figure 91 The stability for hydrostatic pressure is achieved these 2 formulations.

Table 4-4. Cook's membrane at finite strain regime: evolution of norm of residual for the last step for $\mathbf{8}\times\mathbf{8}$ mesh with quadratic NURBS.

Iteration number	Norm of residue		
	$\bar{F}$ projection Q2/Q1	Mixed $(\boldsymbol{u}2_0^2, p2_0^1)$	Mixed $(\boldsymbol{u}2_1^1, p2_0^1)$
1	1.8261753737×10^{3}	$2.1648608997\times10^{-1}$	$2.29397608178\times10^{-1}$
2	2.7469647446×10^{0}	$5.0330858006\times10^{-4}$	$5.94460382817\times10^{-4}$
3	1.5847529783×10^{0}	$2.6129905133\times10^{-9}$	$5.08705436624\times10^{-9}$
4	$5.39329484307\times10^{-3}$		
5	$3.30934230885\times10^{-6}$		

At last. The evolution of the residual norm over the iterations for the last loading step of total 5 steps with an 8×8 elements mesh is explicated in Table 4-4. We see that the Newton iteration for the mixed formulation converges with substantially less number of load steps compared to the strain projection displacement formulation for a given load. Fewer iterations are mandatory to achieve the targeted convergence criterion.

4.3.2. Compression Test

To evaluate the ability of the mixed formulation to undergo large deformations in the quasi-incompressible finite strain regime we study the compression of a block with plain strain kinematics.

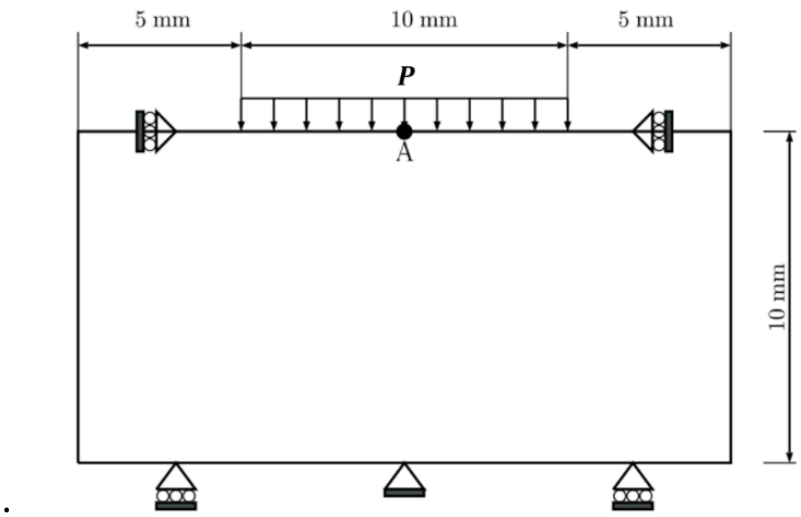


Figure 92. Compression Test: Definition of Problem

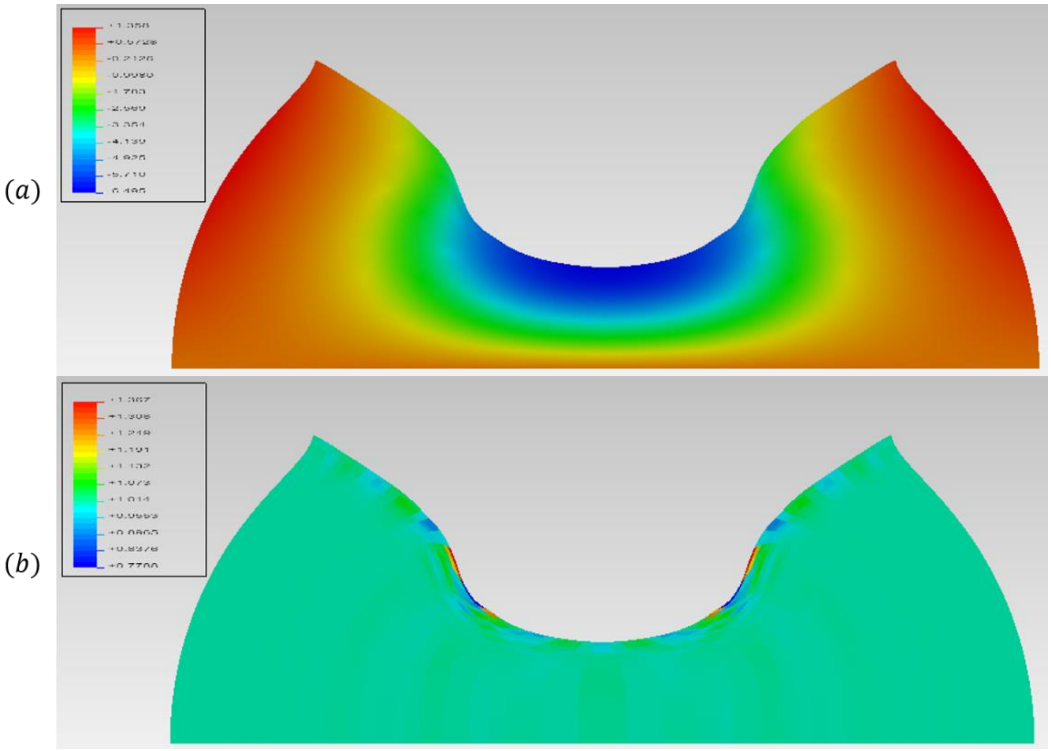


Figure 93. Compression Test: (a) vertical displacement $\boldsymbol{u_y}$; (b) deformation gradient determinant $\boldsymbol{J = det(F)}$

This problem has already been studied in numerous works in context of finite element method (see Reese et al. [264]) and also in context of isogeometric analysis (see Elguedj et al. [46] and Kadapa et al. [62]). The definition of problem along with the geometry are given in Figure 92. The block is submitted to a vertical surface density of effort at the center

of the top side. At the top side, the boundary conditions are defined such as the horizontal displacement is forbidden.

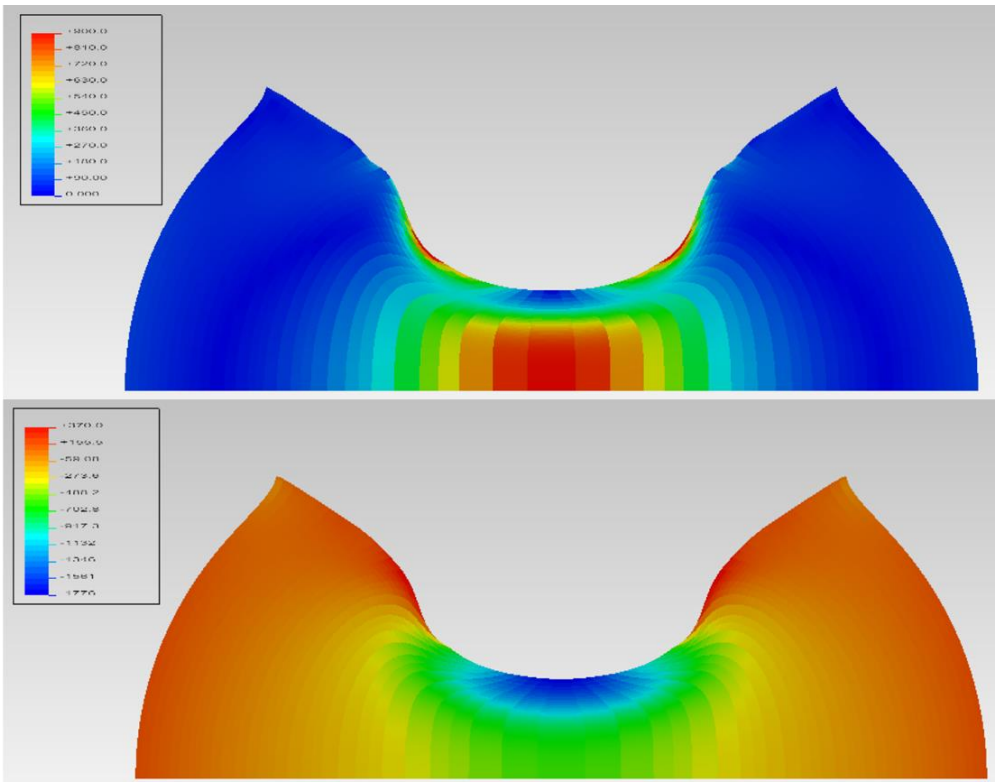


Figure 94. Compression Test: (a)Von-Mises stress; (b)Trace of Cauchy stress $\mathbf{tr}(\boldsymbol{\sigma})$

The Neo-Hookean type free energy is defined as:

$$\psi(\bar{I}_1, J) = \frac{1}{2}\mu(\bar{I}_1 - 3) - \mu \ln J + \frac{1}{4}\kappa(J^2 - 1 - 2\ln J) \qquad (4\text{-}19)$$

Where $\bar{I}_1$ is the first modified invariant of right Cauchy-Green strain tensor $\boldsymbol{C}$. Note that the volumetric contribution of the free energy is different from the one used in the previous example. This corresponds to the 3rd model listed in Table 4-3. The incompressibility constraint is stronger than the one of the previous model. The material parameters are $\mu = 80.1938MPa$ and $\kappa = 400889.806MPa$. We focus on the compression level of the top middle point which is defined as the quotient of vertical displacement of the top central point and the block thickness.

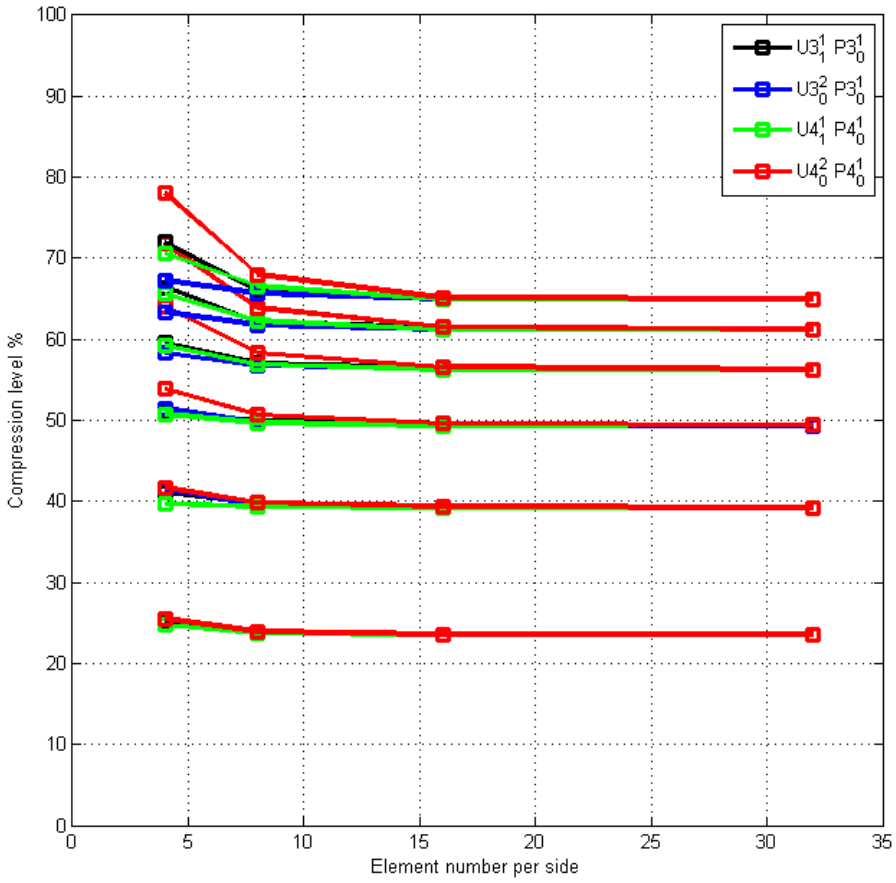


Figure 95. Compression Test: compression level for mixed formulation with discretization $(\boldsymbol{u3_0^2}, \boldsymbol{p3_0^1})$ and $(\boldsymbol{u3_1^1}, \boldsymbol{p3_0^1})$ under different loading intensities.

We analyze the convergence of $(u\,\boldsymbol{p}_0^2, p\,\boldsymbol{p}_0^1)$ and $(u\,\boldsymbol{p}_1^1, p\,\boldsymbol{p}_0^1)$ element for various compression level for various surface load P. The load level is characterized by the ratio P/P_0 where $P_0 = 20MPa$. Figure 93 shows the contour plots of vertical displacement $\boldsymbol{u}_x$ and the determinant of deformation gradient $J = \det(\boldsymbol{F})$ based on a converged mesh with the combination $(\boldsymbol{u}3_0^2, \boldsymbol{p}3_0^1)$ for $P/P_0 = 60$. The Von-Mises and trace of Cauchy stress are shown in Figure 94.

The results for $(u\,\boldsymbol{p}_0^2, p\,\boldsymbol{p}_0^1)$ and $(u\,\boldsymbol{p}_1^1, p\,\boldsymbol{p}_0^1)$ interpolations are compared in Figure 95 for various loading conditions $P/P_0 \in \{10,20,30,40,50,60\}$ for a 16×16 mesh. From the results in Figure 96, we can draw the same conclusion for $\overline{\boldsymbol{F}}$ projection method with Q1/Q0 and Q2/Q1 basis. The mixed formulation with stable discretizations appear to get a superiority on the rate of convergence than the $\overline{\boldsymbol{F}}$ projection method, typically when the

load is rather important, i.e. $P/P_0 = 50$ or 60. The mixed formulation substantially improves the accuracy of the solution for coarse meshes. Once again, we observed that mixed formulation converges with substantially less number of load steps compared to the pure displacement formulation or the strain projection displacement formulation, and for each load step, the number of iteration is smaller. Therefore, the use of mixed formulation reduces the overall computational cost and also gives more accurate results and smooth variation of stresses.

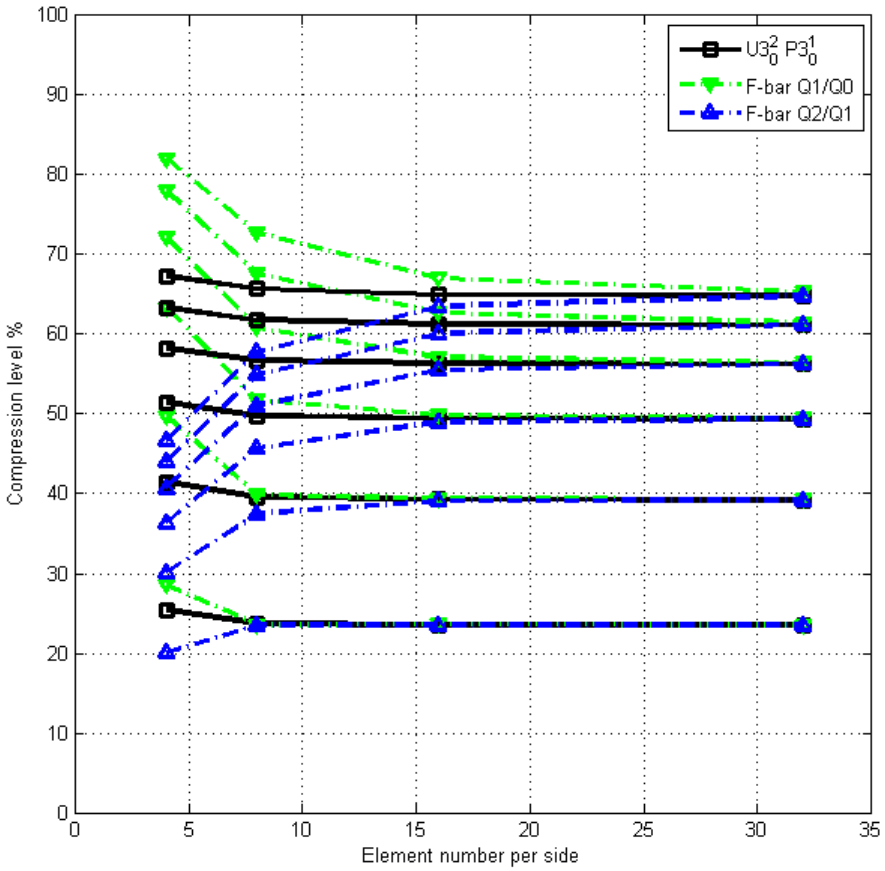


Figure 96. Compression Test: compression level for mixed formulation with discretization $(\boldsymbol{u3_0^2, p3_0^1})$ and $\overline{\boldsymbol{F}}$ projection $\mathbf{Q1\backslash Q0}$ and $\mathbf{Q2\backslash Q1}$ under different loading intensities.

4.4. Conclusion

From these works it is remarked that Standard displacement element of NURBS based Galerkin isogeometric analysis still suffers from mesh locking issue caused by incompressible constraints. But compared to FEM, the high order continuity of NURBS allows IGA to relieve slightly volumetric locking.

Both mixed formulation and strain projection methods are efficient to cure volumetric locking specially under the circumstances of h-refined mesh. And according to our numerical tests for problems at small and finite strain, the mixed formulations of type $(u\,\boldsymbol{p}_0^2, p\,\boldsymbol{p}_0^1)$ and $(u\,\boldsymbol{p}_1^1, p\,\boldsymbol{p}_0^1)$ have a better accuracy for equal mesh size or the equal degree of freedom.

Compared to projection method, that mixed formulation converges with substantially less number of load steps (larger step). And for each load step, the mixed formulation with stable discretizations needs fewer iterations than $\bar{F}$ projection method to convergence.

From the viewpoint of code implementation, the mixed formulation appears to be more straightforward to derive than the strain projection method, especially for nonlinear problems for which the linearization step can become cumbersome for highly coupled multi-fields problems (e.g. with a thermal contribution or any other field in the strain).

5. Thermoelasticity and IGA

As seen in section 3, the temperature plays a crucial role in engineering problems with elastomers supplies. During a mechanical solicitation, the deformation of elastomer is frequently accompanied by important thermal changes. In general, mechanical problem and thermal problem are fully coupled. To our knowledge, thermoelasticity have not clearly been addressed in IGA.

In finite elements, it is well known that the interpolation of the temperature field and the displacement field must fulfill a consistency requirement. The consistency between the prescribed temperature and the assumed representation for the approximations is essential to get accurate solutions. A simple example of incompatibility on a 2-nodes bar element with linear interpolations for both temperature and displacement is shown in Prathap [53]. In the context 2-fields formulation, displacement temperature, the Ladyzhenskaya-Babuska-Brezzi stability condition must be fulfilled to guaranty stability. However, in the literature, one can find many publications in which this consistency condition is violated. In practice, the stresses computed from temperature and displacement does not necessary exhibit oscillations. It mainly depends on the shape of the domain, the mesh size, the magnitude of the coupling, etc. Several schemes of this problem have been proposed in the literature. Since the 70's, many authors proposed variational approaches and derived finite element model for thermoelasticity. In the abundant literature, we will find the same kind of approaches as the one developed in the context of incompressibility. The simplest and most used in practice kind of formulation consists in considering two-field variational formulation choosing various combinations for interpolating displacement and temperature (see e.g. Keramidas et al [54], Carter et al [55], Rao et al [56]). Some authors have proposed to stabilize thermomechanical formulation using reduced integration techniques (see e.g. Juhre et al [57]). In order to alleviate oscillations in stresses and have a better accuracy for both stresses and heat fluxes, a wide variety of mixed formulations have been proposed based on Hu-Washizu variational principle or equivalent: see Miranda et al [58] for discussion on consistency of finite element models in thermoelasticity, Prathap [53] for consistency requirement established through a mixed Hu-Washizu variational formulation, Zhu et al [59] mixed approach, Cannarozzi et al [60] et al for hybrid stress formulation, etc.

At last, in Dittmann et al [61] investigates a thermomechanical model for a contact problem. A mortar algorithm is applied to both mechanical and thermal problem in the context of nonlinear elasticity. An NURBS based isogeometric analysis is investigated for a two-fields formulation. Displacement and temperature are interpolated with same or patches and convenient numerical results are obtained including that for stresses.

In this section, in a similar manner as we did for the incompressibility problem, we investigate the choices of patches for two-fields formulation displacement/temperature fields for IGA applied to thermoelasticity. This choice is driven first by the sake of simplicity, and second, by the study on incompressibility in the previous section. The reader should keep in mind that the final aim of this work is to develop thermomechanical model for incompressible/quasi-incompressible medias. As for incompressibility, for two-fields thermomechanical formulation, stability is also driven by the *inf-sup* condition. In this context, it seems natural to develop simple two-fields IGA formulation with adequate choices of patches for displacement/temperature fields. In this study, we will focus on patches for displacement U and temperature T of type $(\boldsymbol{U}\ p_1^1, \boldsymbol{T}\ p_0^1)$ $(\boldsymbol{T}\ p_0^2, \boldsymbol{T}\ p_0^1)$ $(\boldsymbol{U}\ p_1^1, \boldsymbol{T}\ p-1_0^1)$ $(\boldsymbol{U}\ p_1^1, \boldsymbol{T}\ p-1_0^1)$ $(\boldsymbol{U}\ p_0^2, \boldsymbol{T}\ p-1_0^1)$ which enforce the condition $n_u \geq n_T$(number of degree of freedom for displacement with respect to number of degree of freedom for temperature). Thus, we can except to be able to develop consistent thermomechanical incompressible/quasi-incompressible formulations if similar combination of patches provide stability for both problems.

5.1. Linear thermoelasticity

In this section, we numerically study the convergence of isotropic linear thermoelasticity problems. The coupling scheme for this simple coupled problem is illustrated in Figure 97. The aim of this section is to provide convenient choices of interpolations for coupled multi-field problem, the displacements and temperature.

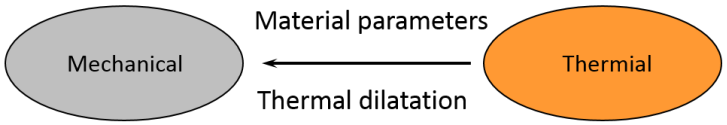


Figure 97. Thermoelastic coupling scheme

The equation of the linear thermoelasticity are:

$$\begin{aligned} \operatorname{div} \boldsymbol{\sigma} + \boldsymbol{b} &= 0 && \text{on } \Omega \\ \boldsymbol{u} &= \bar{\boldsymbol{u}} && \text{on } \partial_1\Omega^u \\ \boldsymbol{\sigma} \cdot \boldsymbol{n} &= \bar{\boldsymbol{t}} && \text{on } \partial_2\Omega^u \\ \operatorname{div} \boldsymbol{q} + r &= 0 && \text{in } \Omega \\ \theta &= \bar{\theta} && \text{on } \partial_1\Omega^\theta \\ \boldsymbol{q} \cdot n &= \bar{q} && \text{on } \partial_2\Omega^\theta \end{aligned} \tag{5-1}$$

Where:

- the boundary is decomposed such as $\partial\Omega = \partial_1\Omega^u \cup \partial_2\Omega^u$ with $\partial_1\Omega^u \cap \partial_2\Omega^u = \emptyset$ and $\partial\Omega = \partial_1\Omega^\theta \cup \partial_2\Omega^\theta$ with $\partial_1\Omega^\theta \cap \partial_2\Omega^\theta = \emptyset$

- $\boldsymbol{u}$ is the displacement field

- $\boldsymbol{\sigma}$ is the Cauchy stress tensor

- $\boldsymbol{b}$ are the body loads

- $\boldsymbol{q} = -k\nabla\theta$ is the heat flux density linearly dependent of the gradient of the temperature (Fourier's law)

- $\bar{\boldsymbol{u}}$ and $\bar{\boldsymbol{t}}$ denote the imposed displacements on the boundary $\partial_1\Omega^u$ and boundary traction on $\partial_2\Omega^u$

- $\bar{\theta}$ and $\bar{q}$ denote the imposed temperature on the boundary $\partial_1\Omega^\theta$ and the prescribed normal heat flux on the boundary $\partial_2\Omega^\theta$, respectively. The definition of the thermoelastic stress is given as:

$$\begin{aligned} \sigma &= \mathbb{C}(\varepsilon - \alpha\Delta\theta\, \boldsymbol{I}) \\ \varepsilon(\boldsymbol{u}) &= (\nabla\boldsymbol{u})_{sym} \end{aligned} \tag{5-2}$$

The weak formulation is stated as follows:

Find $u \in S_u$ and $\theta \in S_\theta$ such as $\forall \delta u \in S_u^0$ and $\forall \delta\theta \in S_\theta^0$.

$$\begin{cases} \mathcal{H}^u(u,\theta,\delta u,\delta\theta) = \int_\Omega \varepsilon(\delta u) : C : \varepsilon(u) d\Omega - \\ \int_\Omega \alpha\, \Delta\theta\, I : C : \varepsilon(\delta u)\, d\Omega - \int_\Omega \delta u\, f\, d\Omega - \int_{\partial_2\Omega^u} \delta u\, F d\Omega = 0 \\ \\ \mathcal{H}^\theta(\theta,\delta\theta) = \int_\Omega k \nabla\delta\theta\, \nabla\theta d\Omega - \\ \int_\Omega \delta\theta\, r d\Omega - \int_{\partial_2\Omega^\theta} \delta\theta\, \bar{q} d\Omega = 0 \end{cases} \tag{5-3}$$

5.1.1. Heat source driven problem on a square domain

The numerical test is a simple heat source driven problem. The displacement depends on the solution of temperature and imposed body loads. A square geometry domain is considered as schemed in Figure 98. Homogeneous boundary conditions in displacements and temperature are imposed.

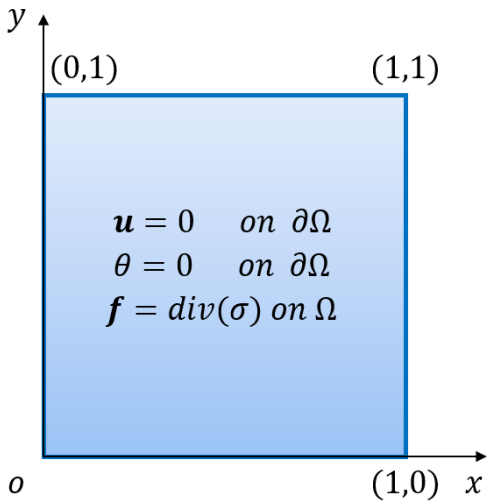


Figure 98. Thermoelastic 2D Square: Problem definition.

The imposed body loads and the volumetric heat source are computed are computed with the predefined solution for the displacement $\boldsymbol{u}$ and temperature θ such that $(\boldsymbol{u},\theta) = (\bar{\boldsymbol{u}},\bar{\theta})$:

$$\bar{\boldsymbol{u}} = \begin{bmatrix} sin(4\pi x)\, sin(4\pi y) \\ sin(4\pi x)\, sin(4\pi y) \end{bmatrix} \tag{5-4}$$

and

$$\bar{\theta} = sin(2\pi x)\, sin(2\pi y) \tag{5-5}$$

The parameters are first defined as follows:

$$\begin{aligned} E &= 1.0, \qquad \nu = 0.3 \\ \kappa &= 1.0, \qquad \alpha = 0.0001 \end{aligned} \tag{5-6}$$

An example of results obtained with a fine mesh (discretization $\boldsymbol{U3_0^2T3_0^1}$ and 20×20 elements) are plotted in Figure 99. From a qualitative point of view, the solution is in agreement with the analytical one.

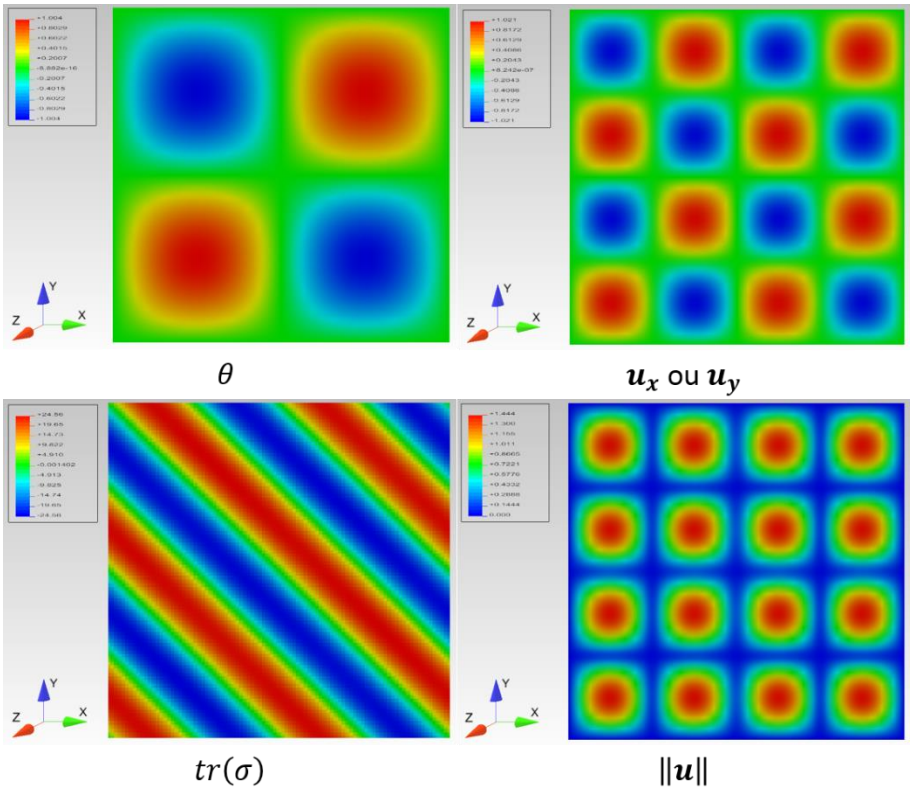


Figure 99. Thermoelastic 2D Square: numerical solution for temperature, displacement and stress trace.

The purpose of this numerical test stays on assessing the two groups of discretizations previously described in the context of incompressibility:

- equal order interpolation for displacement (denoted $\boldsymbol{U}$) and temperature (denoted $\boldsymbol{T}$): $\left(\boldsymbol{U}n_{\boldsymbol{0}}^{\boldsymbol{1}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$, $\left(\boldsymbol{U}n_{\boldsymbol{1}}^{\boldsymbol{1}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$ and $\left(\boldsymbol{U}n_{\boldsymbol{0}}^{\boldsymbol{2}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$
- one order higher for displacements $\left(\boldsymbol{U}n + 1_{\boldsymbol{0}}^{\boldsymbol{1}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$, $\left(\boldsymbol{U}n + 1_{\boldsymbol{1}}^{\boldsymbol{1}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$ and $\left(\boldsymbol{U}n + 1_{\boldsymbol{0}}^{\boldsymbol{2}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$.

The convergence of L^2-norm of both the displacement error and the temperature error are computed. The Figure 100 shows all the convergence curves for equal order discretization combinations. As expected, the L^2-norm of the temperature error decreases with the optimal rate equal to $n + 1$ for n the interpolation order of temperature. The convergence rates of the L^2-norm of displacement error is also really close to the optimal one as well.

The same results have been obtained for higher order interpolation for displacements compared to temperature.

From Figure 102 to Figure 105, the thermal expansion coefficient α is tuned from 0 to 1.0 $\{0,\ 0.0001,\ 0.1, 1\}$. This test is held for (Un_0^2, Tn_0^1) where *n* is the order of interpolation. Recall that inner knots are repeated once (superscript 2 for displacement) reducing continuity at each knot for displacement. The coupling between both equation is getting stronger when α grows. For $\alpha = 0$, the problem is fully decoupled and, in Figure 102, expected error convergence results for linear elasticity and linear thermal problems are obtained (optimal convergence for both displacement and temperature). For Figure 102 to Figure 105, we show that the stronger le coupling is for thermoelasticity, the weaker the convergence is. From a practical point of view, for strongly coupled problems, this formulation may fail. It is important to note that in most application the coupling due to thermal expansion is rather weak.

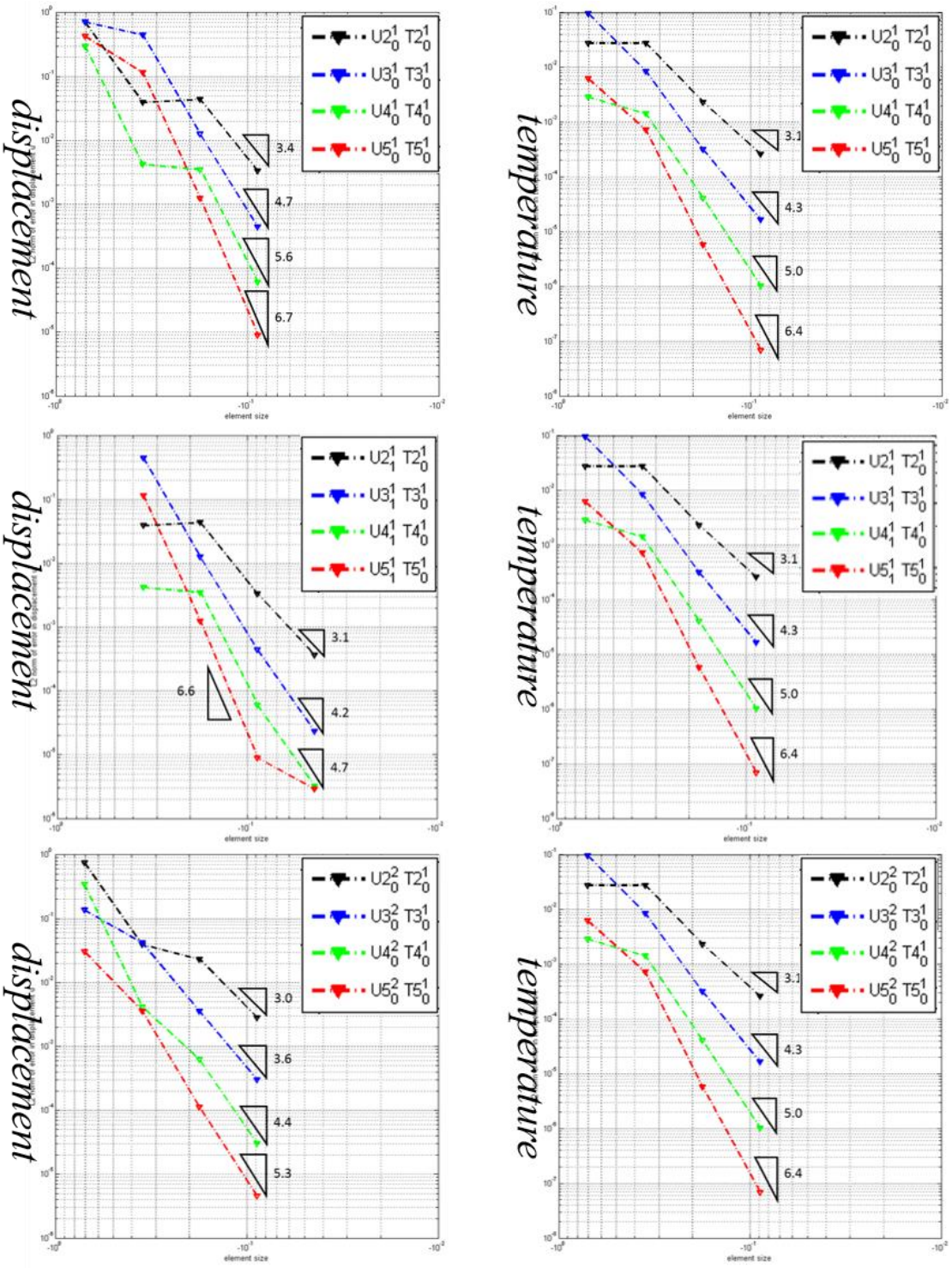


Figure 100. Convergence of L^2-norm of displacement and temperature errors for (Un_0^1, Tn_0^1), (Un_1^1, Tn_0^1) and (Un_0^2, Tn_0^1).

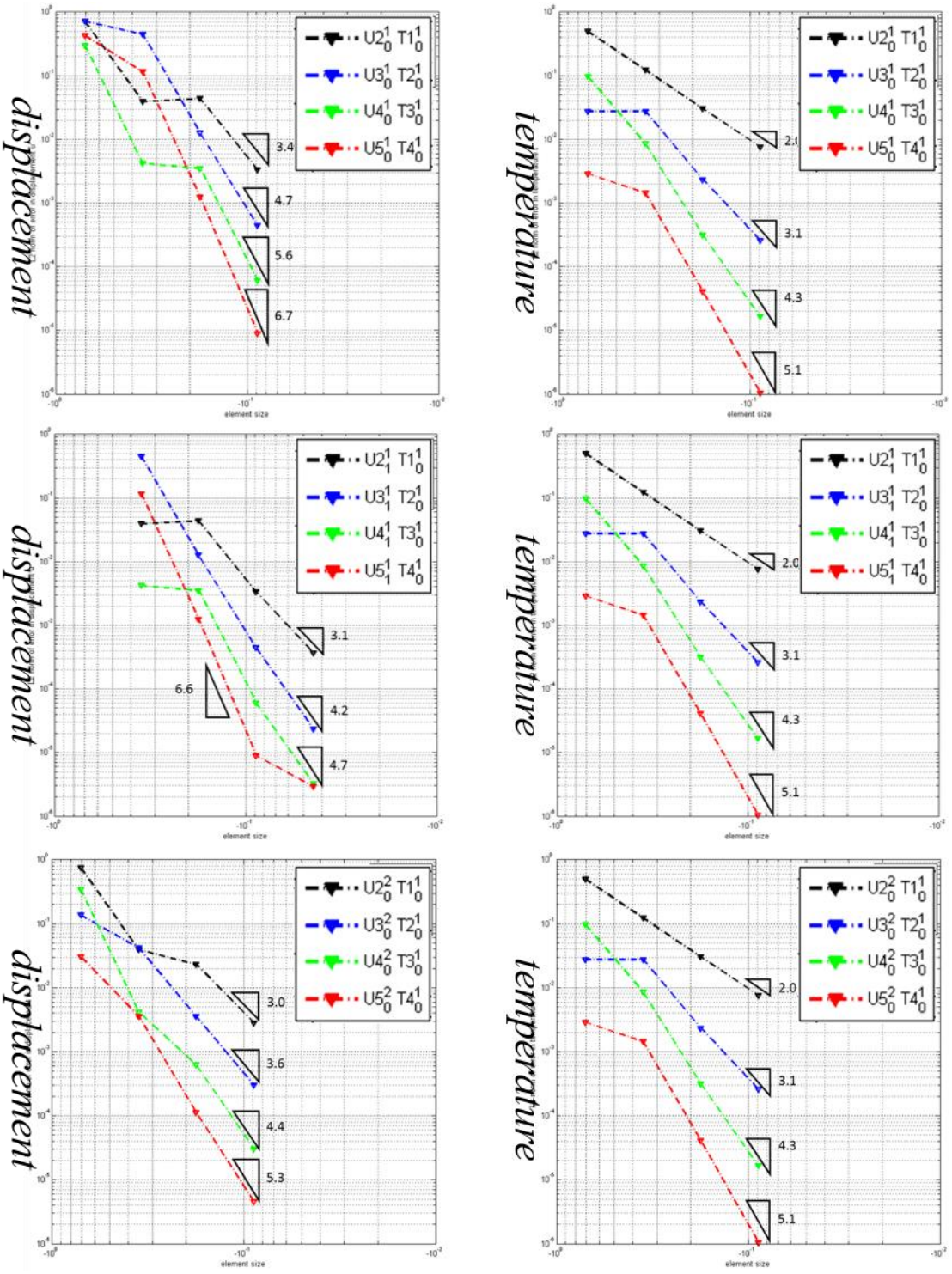


Figure 101. Convergence of L^2-norm of displacement and temperature errors for$(Un + 1_0^1, Tn_0^1)$, $(Un + 1_1^1, Tn_0^1)$ and $(Un + 1_0^2, Tn_0^1)$

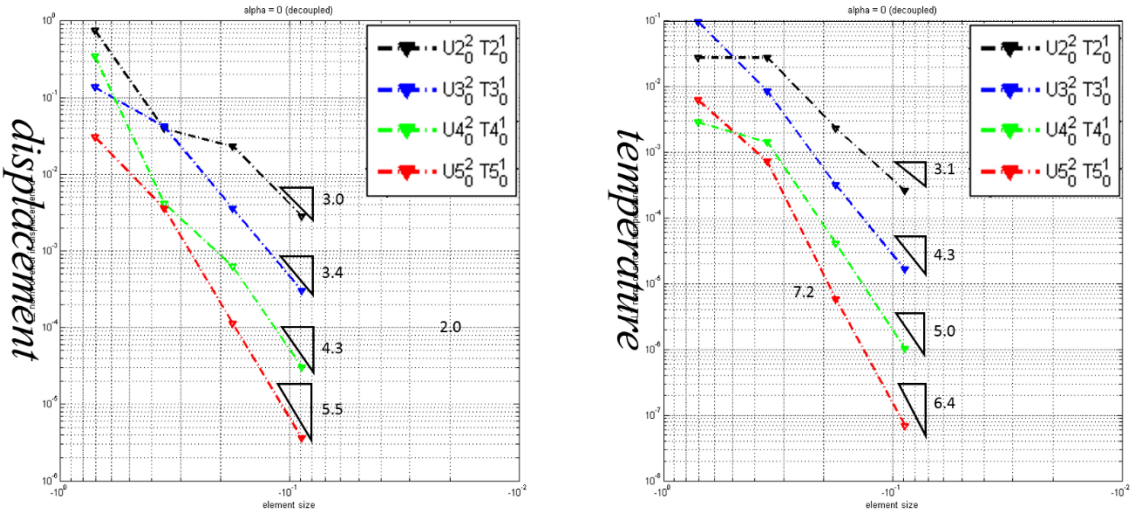


Figure 102. Convergence of L^2-norm of displacement and temperature errors for$\left(\boldsymbol{Un_0^2}, \boldsymbol{Tn_0^1}\right)$ with $\boldsymbol{\alpha = 0}$.

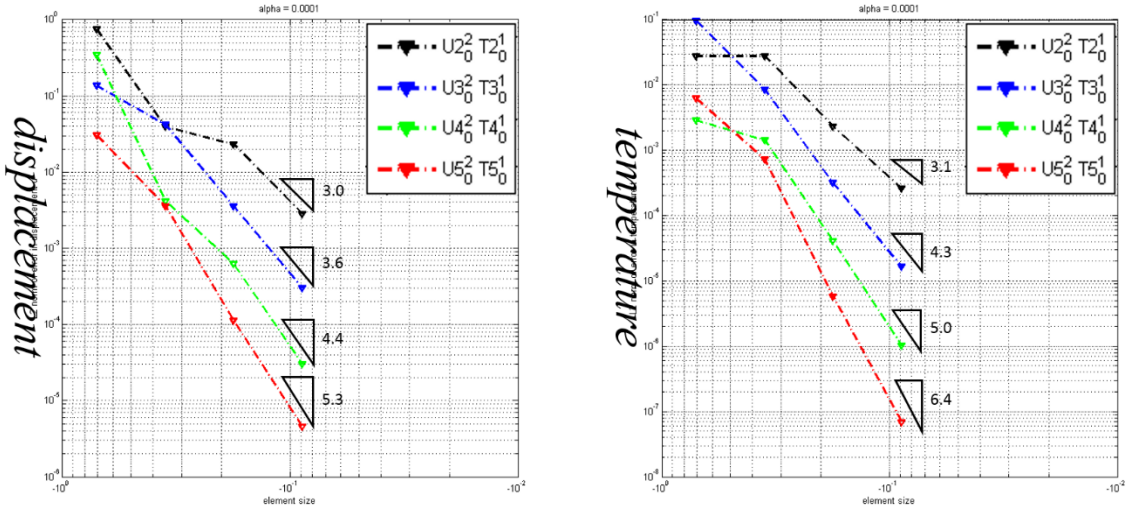


Figure 103. Convergence of L^2-norm of displacement and temperature errors for$\left(\boldsymbol{Un_0^2}, \boldsymbol{Tn_0^1}\right)$ with

$\boldsymbol{\alpha = 0.0001}$.

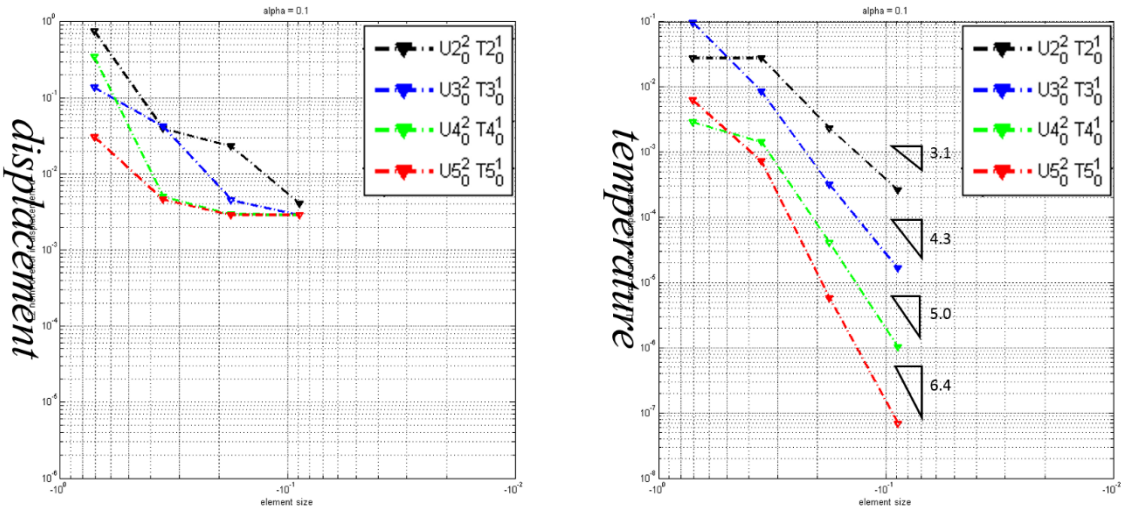


Figure 104. Convergence of L^2-norm of displacement and temperature errors for$\left(\boldsymbol{Un_0^2}, \boldsymbol{Tn_0^1}\right)$ with $\boldsymbol{\alpha = 0.1}$.

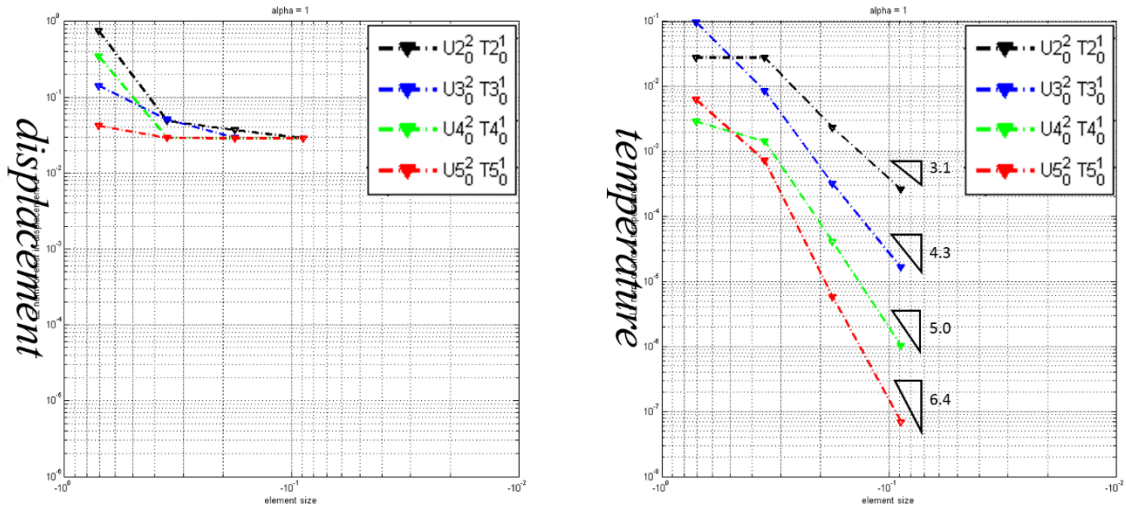


Figure 105. Convergence of L^2-norm of displacement and temperature errors for$\left(\boldsymbol{Un_0^2}, \boldsymbol{Tn_0^1}\right)$ with $\boldsymbol{\alpha = 1}$.

5.1.2. Heat conduction in a thick cylinder

In this test, our intention is to further investigate converge rate of errors on a problem for which geometry is more complex than the square of previous section (see Zander et al [265] for the definition of the problem). In this case, the mapping from parameter space to physical space is not a simple stretching and the coupling between thermal problem and elasticity is strong ($\alpha = 1.0$). The element shape is more complex. The geometric domain,

boundary conditions have described in Figure 106. On the thick cylinder, the temperature is imposed on both inner and outer surface, respectively $3K$ and $1K$. Once the material parameters given as:

$$\begin{aligned} E &= 1.0, \qquad \nu = 0.0 \\ k &= 1.0, \qquad \alpha = 1.0 \end{aligned} \tag{5-7}$$

The analytical solution temperature distribution is known:

$$\theta(r) = 1 - \frac{ln\, r}{ln\, 2} \tag{5-8}$$

where r denotes the radial distance.

A radial imposed displacement of 0.25 is applied on the inner surface of the cylinder and the outer surface is fixed. As the problem is axisymmetric, we consider only a quarter of the cylinder as shown in Figure 106. The analytical solutions for displacement is given as follows:

$$\begin{aligned} u_r(r) &= -\frac{r}{2}\frac{ln\, r}{ln\, 2} \\ u_t(r) &= 0 \end{aligned} \tag{5-9}$$

An example of solution obtained with combination $(\boldsymbol{U}3_0^2, T3_0^1)$ is plotted in Figure 107. Roughly speaking, the temperature, the displacement and the stress are captured without any oscillations. Similar results are obtained with $(\boldsymbol{U}3_0^1, T3_0^1)$, which contrasts with mixed formulation for incompressible problems for which spurious oscillations for stresses were observed (see Figure 108).

As for the mixed formulation for the Stokes problem in section 4.2, all the discretizations listed in Table 4-1 with different order have been assessed for this problem. The L^2-norms of the temperature and displacement errors are given in Figure 109 for $\left(\boldsymbol{U}n_{\boldsymbol{0}}^{\boldsymbol{1}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$, $\left(\boldsymbol{U}n_{\boldsymbol{1}}^{\boldsymbol{1}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$ and $\left(\boldsymbol{U}n_{\boldsymbol{0}}^{\boldsymbol{2}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$, and in Figure 110 for $\left(\boldsymbol{U}n + 1_{\boldsymbol{0}}^{\boldsymbol{1}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$, $\left(\boldsymbol{U}n + 1_{\boldsymbol{1}}^{\boldsymbol{1}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$ and $\left(\boldsymbol{U}n + 1_{\boldsymbol{0}}^{\boldsymbol{2}}, Tn_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$. In Figure 109, the L^2-norm of temperature error has an optimal convergence rate as expected (the thermal equation is fully decoupled in this problem), i.e. convergence of order *n+1* for interpolation of order *n*. However, the L^2-norm of displacement error converges but not optimally. However, we can remark that for

$\left(\boldsymbol{Un_0^2}, \boldsymbol{Tn_0^1}\right)$ the error is smaller than for alternate interpolation choices. In Figure 110, similar numerical results are obtained for displacements. This might come from a strong thermoelasticity coupling as shown in the previous section.

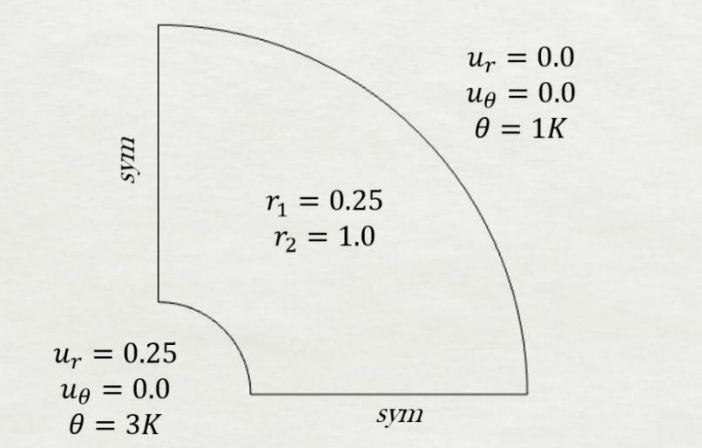


Figure 106. Thermoelastic thick cylinder: Problem definition

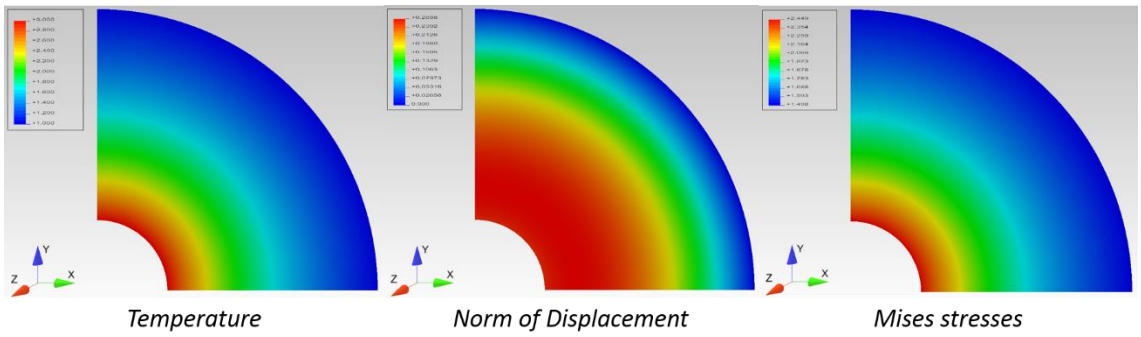


Figure 107. Typical solution for temperature, displacement norm and Von-Mises stresses for $\left(\boldsymbol{U3_0^2}, \boldsymbol{T3_0^1}\right)$

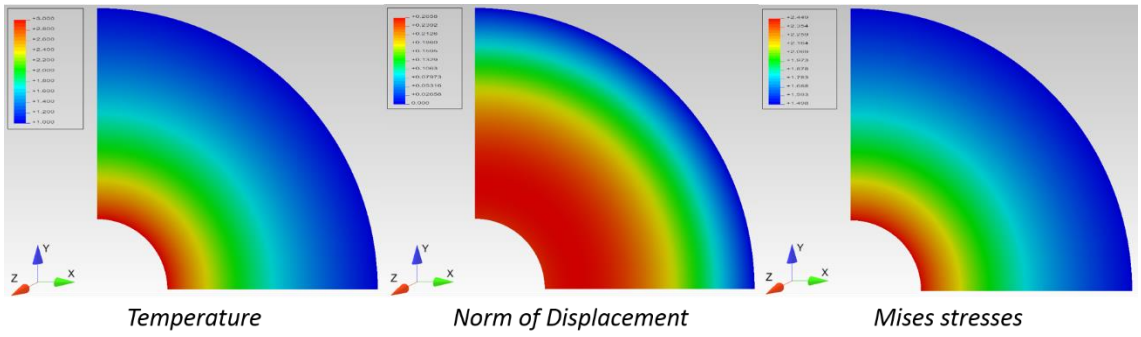


Figure 108. Typical solution for temperature, displacement norm and Von-Mises stresses for $\left(\boldsymbol{U3_0^1}, \boldsymbol{T3_0^1}\right)$

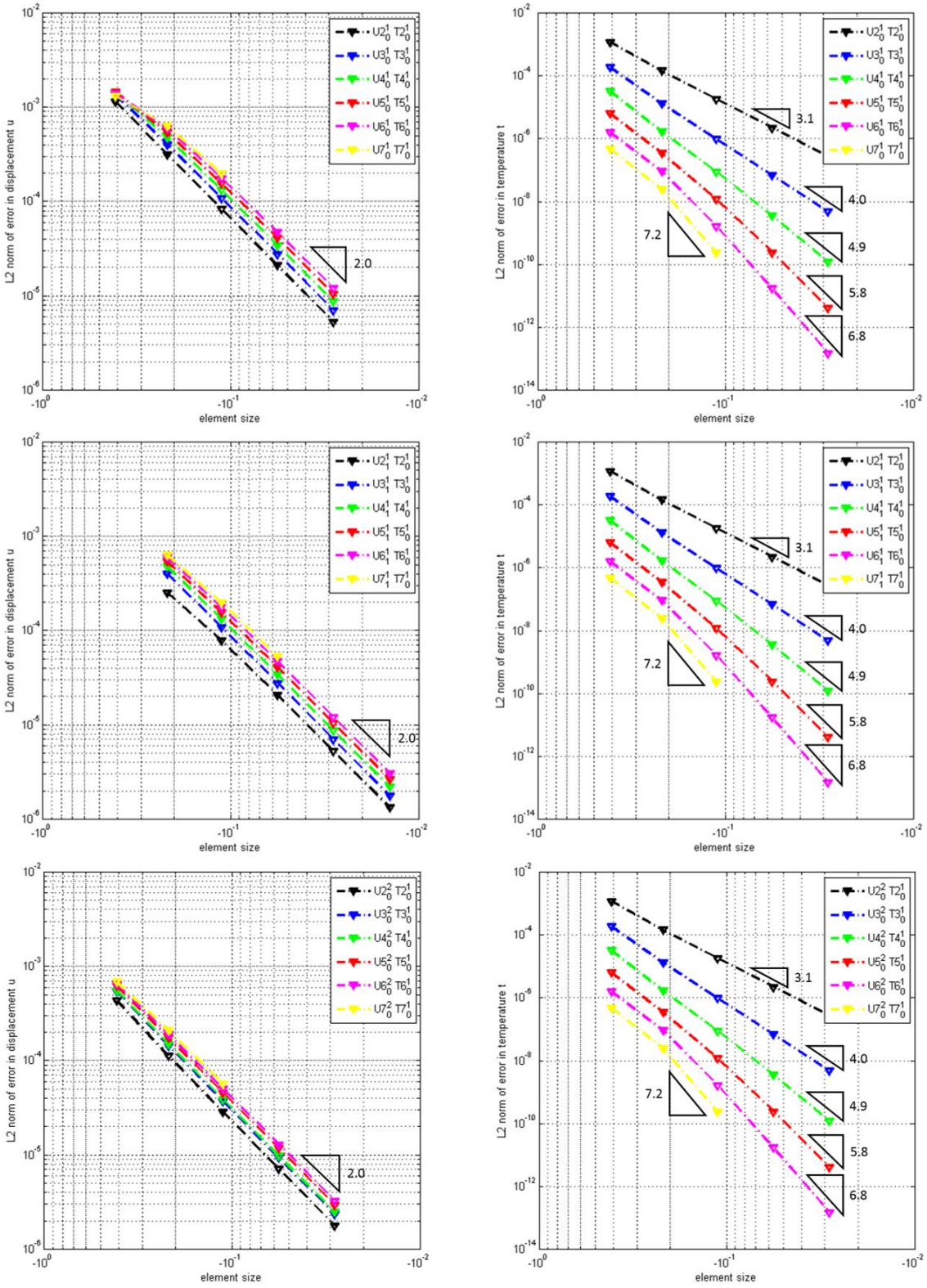


Figure 109. Convergence of L^2-norm of displacement and temperature errors for (Un_0^1, Tn_0^1), (Un_1^1, Tn_0^1) and (Un_0^2, Tn_0^1).

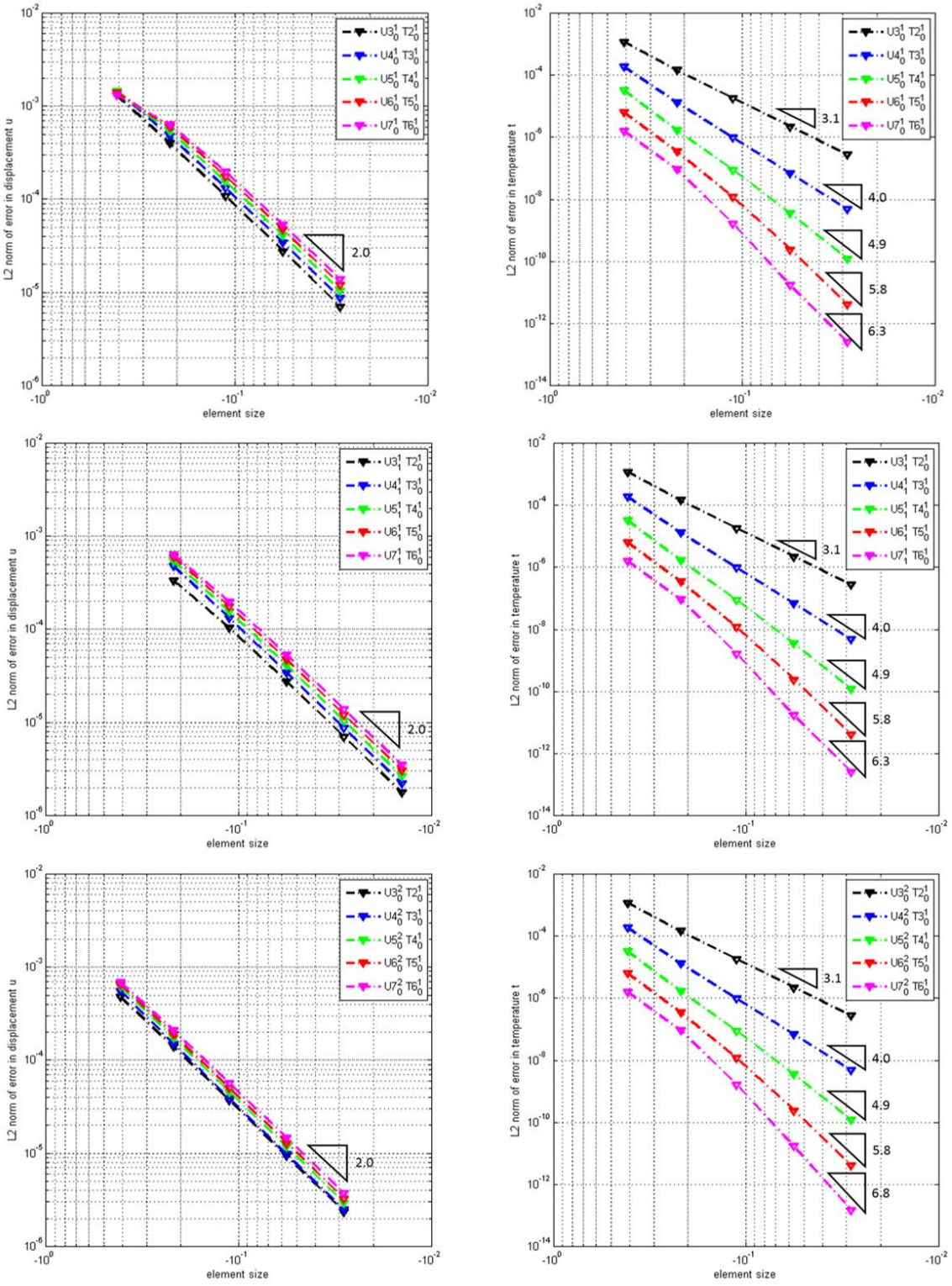


Figure 110. Convergence of L^2-norm of displacement and temperature errors for $(Un + 1_0^1, Tn_0^1)$, $(Un + 1_1^1, Tn_0^1)$ and $(Un + 1_0^2, Tn_0^1)$.

5.1.3. Patches' coupling: A preliminary study for heat conduction in a thick cylinder

As preliminary test for patches coupling, we investigate the same problem solved with two patches as shown in Figure 111. The patches are coupled by using the Nitsche's method. The detail of the implementation and references are given in Appendix A.

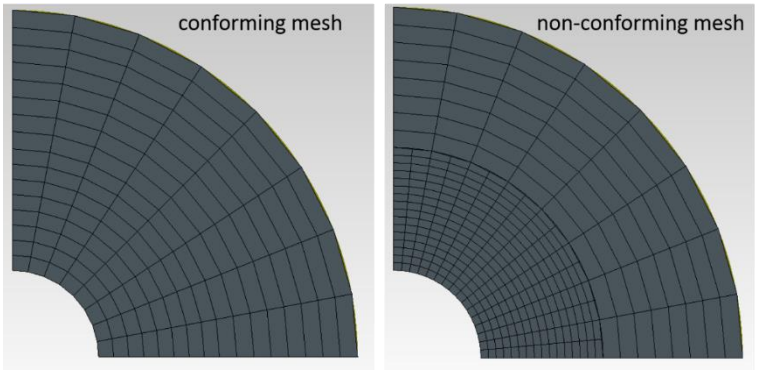


Figure 111. Thermoelastic 2D ring plate: two patches with conforming and non-conforming mesh

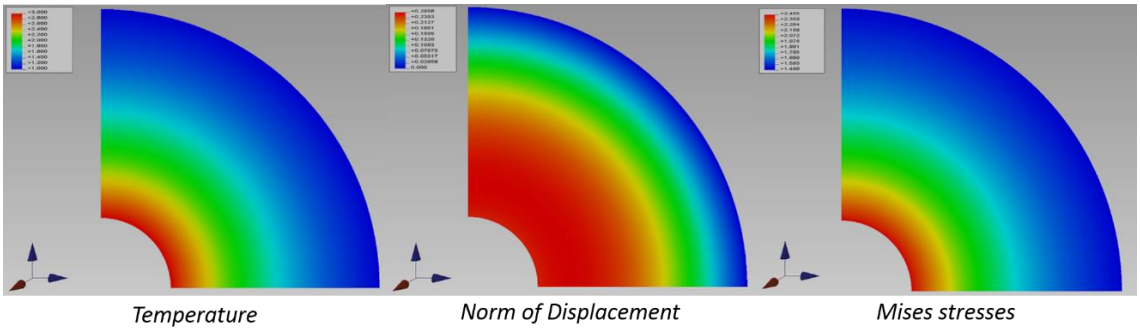


Figure 112. Thermoelastic 2D ring plate: Numerical solutions for temperature, displacement norm and Von-Mises stress derived with Nitsche's Method.

The quarter of the thick cylinder is decomposed into two subdomains for both conforming and non-conforming meshes as shown in Figure 111. The numerical results for temperature, displacement norm and Von-Mises stress with these two type mesh are shown in Figure 112. No spurious local stress oscillations appear from a qualitative point (see for further investigations).

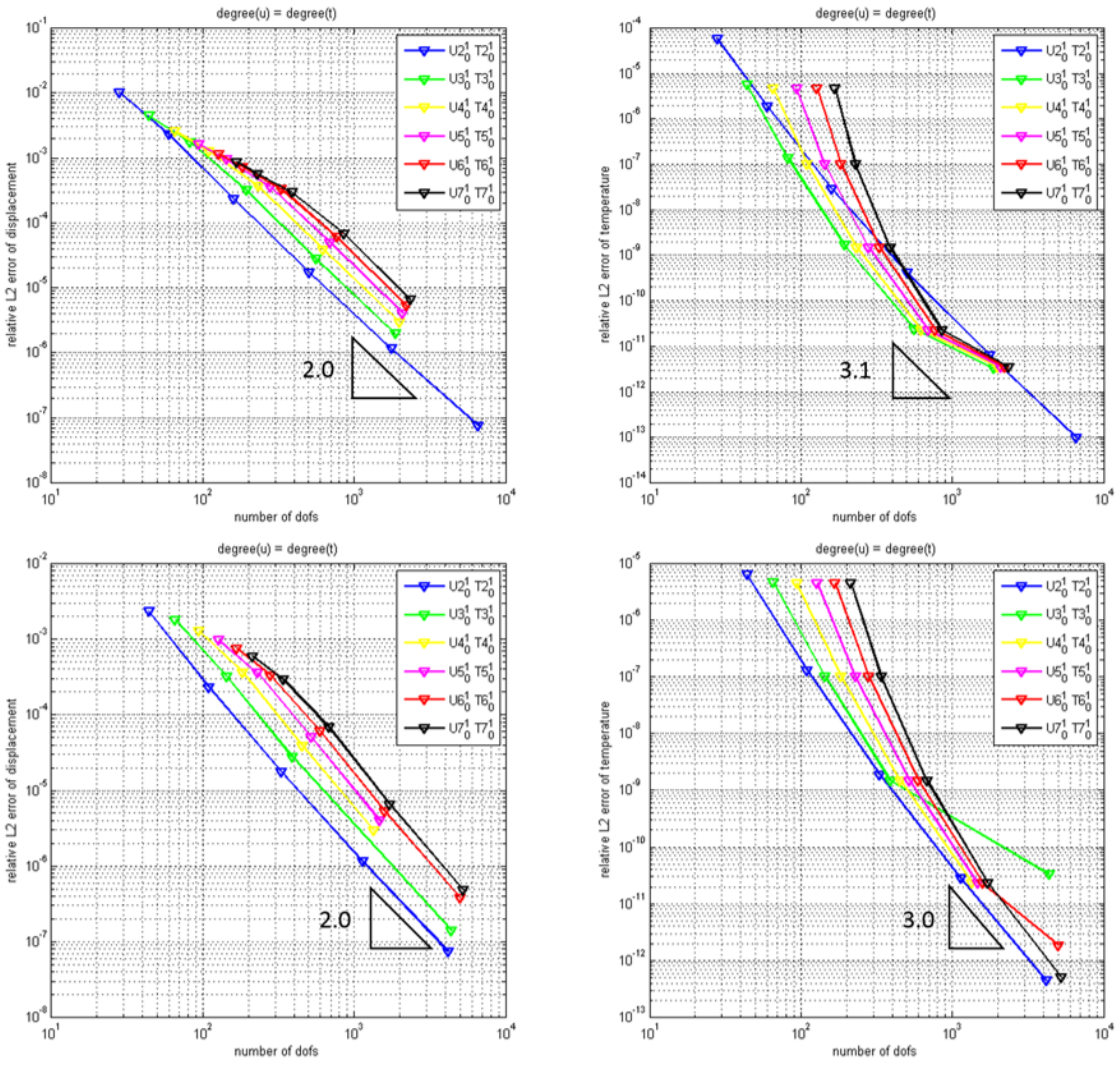


Figure 113. Convergence of L^2-norm of displacement and temperature errors for conforming (top) and non-conforming (bottom) mesh with Nitsche's method.

Nevertheless, as observed in Figure 113, the convergence rates of the L^2-norm of displacement and temperature errors is affected by the coupling method. The convergence is good but does not achieve the optimal rate expected for such choices of patches. The C^1 continuity imposed by Nitsche's method probably affect this convergence. Note that on this example, the convergence rate of the L^2-norm of displacement error is limited to 2, and the convergence rate of the L^2-norm of temperature errors is limited to 3.

5.2. Nonlinear thermoelastic applications

In this section, we investigate several thermoelastic problem at the finite strain regime. Compared to problems developed in section 5.1, these problems may have both geometric nonlinearity and material nonlinearity. Note that the coupling between mechanical and thermal fields turns out to be bidirectional.

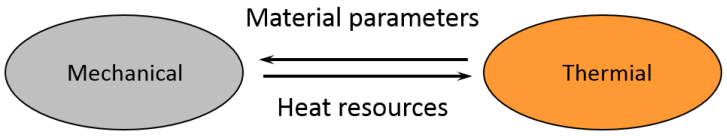


Figure 114. Full coupling scheme for nonlinear thermomechanics.

The local form of the balance of momentum for the quasi-static mechanical problem and the balance of energy in the reference configuration Ω_0 within the time period $[0, T]$ for a homogeneous isotropic material (the notations are not recalled here) are given by:

$$\begin{aligned} Div\boldsymbol{P} + \boldsymbol{B} &= 0 \;\; in\, \Omega_0 \times [0,T] \\ \boldsymbol{u} &= \bar{\boldsymbol{u}} \quad on\, \partial\Omega_{0u} \\ \boldsymbol{P}\cdot\boldsymbol{N} &= \bar{\boldsymbol{T}}\, on \;\; \partial\Omega_{0\sigma} \end{aligned} \tag{5-10}$$

$$\begin{aligned} \theta\dot{\eta} &= -Div\,\boldsymbol{Q} + R \quad in\, \Omega_0 \times [0,T] \\ \theta &= \bar{\theta} \quad on\, \partial\Omega_{0\theta} \\ \boldsymbol{Q}\cdot\boldsymbol{N} &= \bar{Q} \quad on\, \partial\Omega_{0q} \\ \theta(t=0) &= \theta_0 \end{aligned} \tag{5-11}$$

A nonlinear initial boundary value problem of coupled partial differential equations is defined in equations (5-10) and (5-11). The thermos-hyperelastic free energy $\psi(\boldsymbol{u}, \theta)$ depends on both the displacement and temperature. The constitutive law can be expressed as follows:

$$\boldsymbol{P} = \frac{\partial\psi(\boldsymbol{F}(\boldsymbol{u}),\theta)}{\partial\boldsymbol{F}} \quad and \quad \eta = \frac{\partial\psi(\boldsymbol{u},\theta)}{\partial\theta} \tag{5-12}$$

5.2.1. A fully coupled time-dependent thermomechanical problem

The problem chosen here is taken from Erbts et Düster [266]. We extend the linear thermoelasticity model to nonlinear case to obtain the free energy $\psi(\boldsymbol{u},\theta)$. Thus, the energy can be obtained in a straightforward manner by replacing the infinitesimal strain tensor ε with Green-Lagrange strain tensor E as follows:

$$\psi(E(\boldsymbol{u}),\theta) = \frac{1}{2\rho_0}(E:\mathbb{C}:E) - 3\kappa\,\alpha\,tr(E)\frac{\theta-\theta_0}{\theta} - \frac{1}{2}c(\theta-\theta_0)^2 \qquad (5\text{-}13)$$

where c denotes the heat capacity, and $\mathbb{C}$ indicates the elasticity tensor.

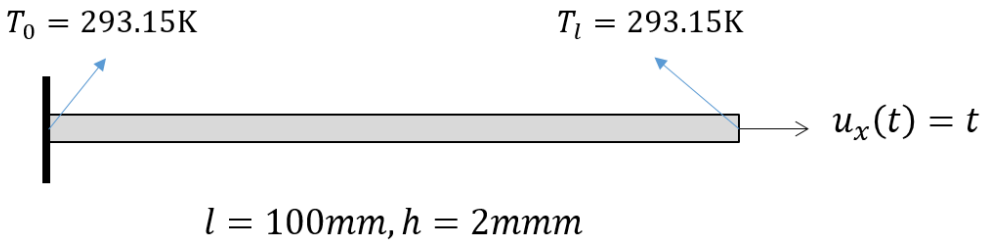


Figure 115. 2D clamped beam.

On this example, we focus on convergence properties of FEM and IGA. The self-cooling of a long beam with a square cross-sectional area under tensile loading is computed as defined in Figure 115. The material properties are summarized in Table 5. For the FEM, the domain is discretized with hexahedral elements. The boundary conditions are defined as follows:

- the left end of the beam is clamped
- at both ends, the temperature is imposed and corresponds to the reference temperature ($\theta_0 = 293.15K$), and adiabatic conditions are considered on other faces
- on the right end, a time dependent displacement is imposed

Due to the thermoelastic coupling, the temperature of the beam will evolve due to the deformation of the beam. Note, first, that the formulation is defined at finite strain and the test is at small strain and second, the exact solution is a 1D solution for temperature, but a full 3D computation.

Table 5. Thermoelastic Rod: Material parameters and domain dimensions

Bulk modulus	κ	164206	N/mm^2
Shear modulus	μ	80194	N/mm^2
Density	ρ_0	7.8×10^{-9}	Ns/mm^4
Conductivity	k	45	N/sK
Heat capacity	c	4.6×10^{8}	N/sK
Thermal expansion	α	1.5×10^{-5}	$1/K$
Length of rod	l	100	mm
Quadratic area	A	4	mm^2
Rate of displacement	$\dot{u}_0$	1	mm/s
Reference temperature	θ_0	293.15	K

The 1D analytical solution for temperature of the problem is given by (e.g. see [266]):

$$\theta(x,t) = \frac{4\gamma l^2}{\pi^3\delta^2}\sum_{n=1}^{\infty}\frac{\sin\left(\frac{n\pi x}{l}\right)}{n^3}\left[1-e^{-\left(\frac{n\pi\delta}{l}\right)^2 t}\right], \qquad n = 1,3,5,\dots$$
$$with \quad \delta = \sqrt{\frac{\kappa}{(\rho c + 3\alpha^2 k\theta_0)}} \quad and \quad \gamma = \frac{3\alpha k\theta_0 \dot{u}_0}{l(\rho c + 3\alpha^2 k\theta_0)} \tag{5-14}$$

This analytical solution is only valid for small strains. Remember that the formulation we are using here is finite strain formulation to preserve the small strains hypothesis. Thus, we perform the convergence analysis at $t = 1s$ for an elongation ${}^{u}/_{l} = 0.01$. The temperature isovalues and vertical displacement isovalues distribution at $t = 1s$ are shown in Figure 116. Discretizations with equal degree $\left(\boldsymbol{U}p_{\boldsymbol{0}}^{\boldsymbol{1}}, Tp_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$, $\left(\boldsymbol{U}p_{\boldsymbol{1}}^{\boldsymbol{1}}, Tp_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$ and $\left(\boldsymbol{U}p_{\boldsymbol{0}}^{\boldsymbol{2}}, Tp_{\boldsymbol{0}}^{\boldsymbol{1}}\right)$ for $p \in \{2,3,4\}$ are considered in this example. Such as the linear case, the three combinations give similar results for the displacement, the temperature and stresses.

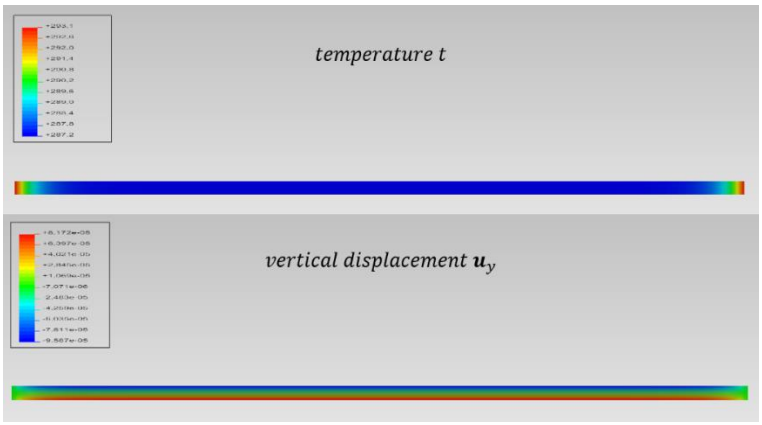


Figure 116. Temperature and vertical displacement color maps.

Figure 117 shows the temperature profiles along the horizontal midline with quadratic IGA and linear FEM for different meshes. The plain black curve represents the analytical solution. It is observed that IGA qualitatively captures the solution with 7 times less degree of freedom than FEM.

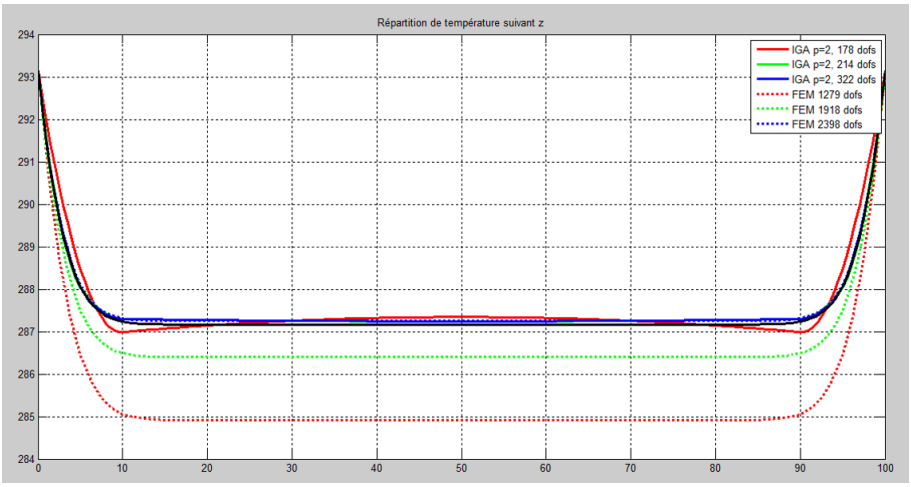


Figure 117. Thermoelastic Rod: temperature profiles with IGA and FEM.

In the Figure 118, we compare the convergence rate of L^2-norm of the temperature error for IGA and FEM (linear and quadratic discretizations, for equal order interpolation or linear/quadratic interpolations for temperature/displacement). For IGA, equal order patches for displacement and temperature are chosen (for order $p \in \{2, 3, 4\}$). For finite elements, equal order interpolations of degree 1 and 2 (elements H1 and H2) for hexahedron elements) and linear/quadratic combination for temperature/displacement (element H1/H2) are chosen. Roughly speaking, IGA produces better results than FEM, i.e. for a given number of elements the L^2-norm of temperature error is smaller, except for the

equal order H2 finite element compared to the IGA of order 2 patch. In this case, it should be note that the number of degree of freedom is smaller for IGA. However, note that the convergence rate of L^2-norm of temperature error seem not to be optimal here. Note that here the reference solution taken for the temperature is a 1D analytical solution while the problem solved here is a full 2D problem. From our point of view, the 2D effects at both end of the beam cannot be neglected. Despite this fact, the test gives a meaningful comparison for various schemes.

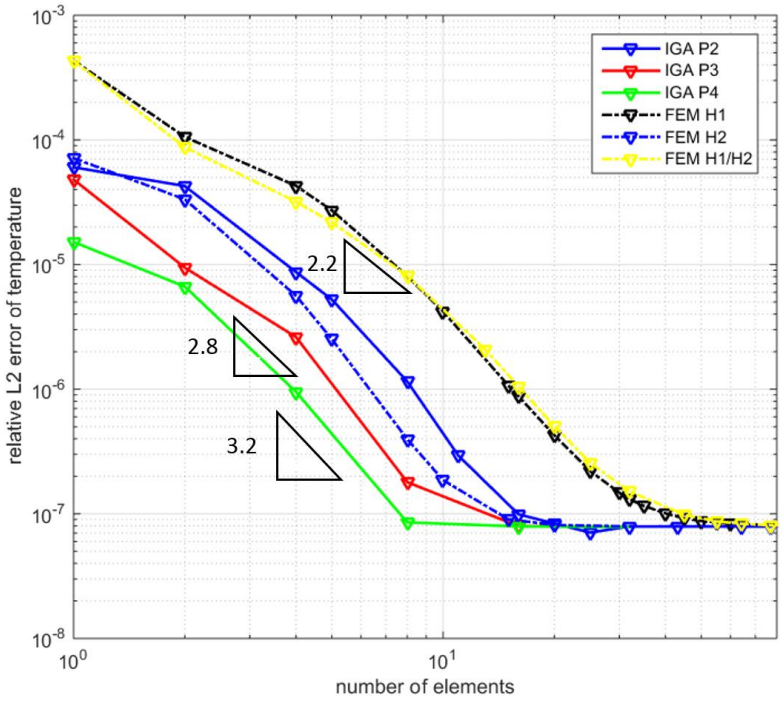


Figure 118. Convergence of L^2-norm of temperature error with IGA and FEM.

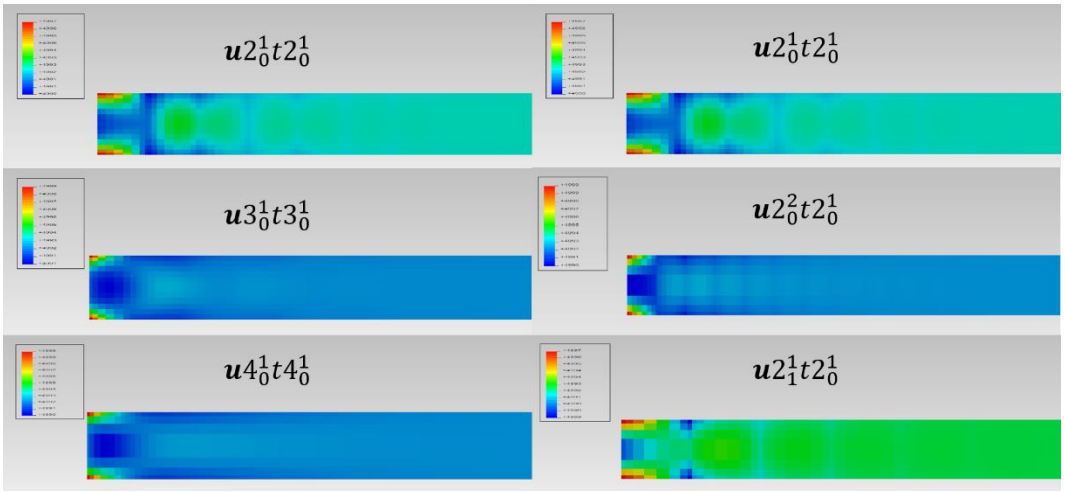


Figure 119. Thermoelastic Rod: Von-Mises stress on the left for diverse discretizations.

At last, the Von-Mises stresses at both ends are shown in Figure 119 for various patches combinations. Due to the Dirichlet boundary conditions at both ends, we observed a concentration of stresses with spurious oscillations for some combinations of patches. When the order of the patch is less than 2, some oscillations are observed for all the discretization combinations. These oscillations disappear when elevating the order of the patch whatever the combination is for displacement and temperature. However, note that the combination $\left(\boldsymbol{Up_1^1}, \boldsymbol{Tp_0^1}\right)$ and $\left(\boldsymbol{Up_0^2}, \boldsymbol{Tp_0^1}\right)$ exhibit much less oscillation. For that reason, we will prefer this kind of element for thermoelasticity.

5.2.2. Thermomechanical entropic elasticity

we adopt a form for the Helmholtz free energy function which is similar to the proposed in the Holzapfel et Simo [267]. On this example, we want to evaluate the capability of the IGA to capture the physics for a complex thermomechanical coupling capable to reproduce thermoelastic inversion.

$$\psi(F,\theta) = \psi_0(F)\frac{\theta}{\theta_0} - e_0(F)\frac{\theta-\theta_0}{\theta_0} + T(\theta) \tag{5-15}$$

where $\psi_0(F)$ is set to be the isothermal weakly compressible Neo-Hookean type model:

$$\begin{gathered}\psi_0(F) = \overline{\psi}_0(\bar{I}_1, J) = C_1(\bar{I}_1 - 3) + \chi\, G(J) \\ G(J) = (J-1)\end{gathered} \tag{5-16}$$

The volumetric coupling term is defined as follows:

$$e_0(F) = 3\alpha_0\, \chi(\theta_0) G(J) \tag{5-17}$$

and the purely thermal contribution is:

$$T(\theta) = c_0\left(\theta - \theta_0 - \theta \ln\frac{\theta}{\theta_0}\right) \tag{5-18}$$

For thermal equation, a Piola heat flux derives the constitutive law as follows:

$$Q(F,\theta,\nabla\theta) = -kC^{-1}\mathrm{Grad}(\theta) \tag{5-19}$$

The same material parameters as in Holzapfel et Simo [267] used in this example:

$$\begin{gathered} C_1 = 2.1125\times10^5 \\ \chi = 4.0\times10^5 \\ c_0 = 183.0 \\ k = 20.15 \\ \alpha = 22.333\times10^{-5} \\ \theta_0 = 293.15 \end{gathered} \tag{5-20}$$

We assume a unit cube which is stretched in one direction from the initial no deformed state up to a stretching ratio of $\lambda = {}^{l}/_{l_0} = 1.4$. This is a classical trial example in rubber-elasticity to show a difference of behavior between rubber-like materials and metal-like solids. Both materials type exhibits different response in temperature.

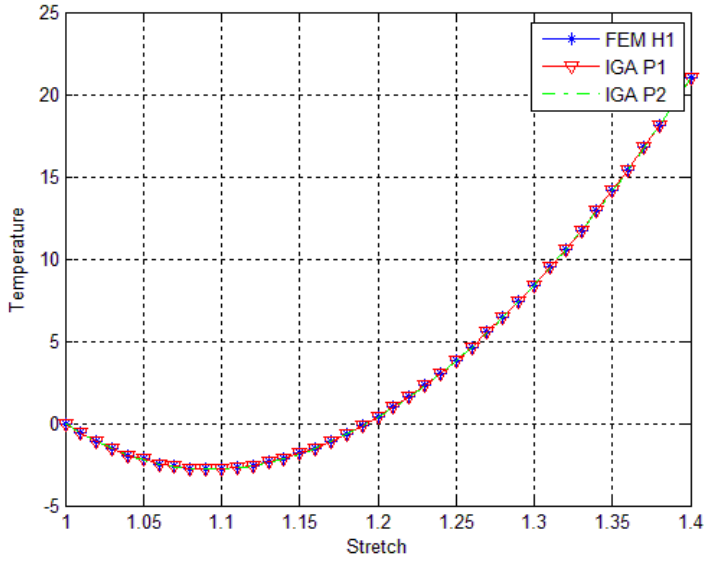


Figure 120. Temperature variations along with one-directional stretching.

In Figure 120 , we show the evolution of the temperature with respect to stretching for a single linear hexahedron element in FEM and for a single quadratic NURBS element in IGA. As observed, they provide the same results for the temperature. We can observe the

so-called thermoelastic inversion, which corresponds to an initial cooling of the material followed by a temperature elevation during stretching.

5.3. Applications to industrial interest constitutive laws

The section presents a new formalism to revisit thermo-mechanical coupling of nearly-incompressible materials at finite strain in a robust multi-fields context. The model is an evolution of Nguyen et al. [268] and was developed by Lejeunes [269]. Some numerical examples are shown to illustrate that this formalism can consider well-known phenomena such as the thermoelastic inversion of unfilled rubber or the self-heating phenomenon in filled rubber. We evaluate this new form of constitutive law through IGA.

5.3.1. Thermodynamic framework

Following the work of Flory [270], the transformation is split into volumetric and isochoric part is given by:

$$\boldsymbol{F} = (J^{1/3}\boldsymbol{I}) \cdot \overline{\boldsymbol{F}} \tag{5-21}$$

where $J = \det(\boldsymbol{F})$is the volumetric variation and $\boldsymbol{I}$ the identity tensor.

We assume that the volume variation can result from two independent terms: the mechanical compressibility J_m, the thermal dilatation J_θ, such that:

$$J = J_m J_\theta \tag{5-22}$$

The volumetric variation terms are defined through the following relations:

$$\begin{aligned} J_\theta &= 1 + \alpha(\theta - \theta_0) \\ J_m &= J(J_\theta)^{-1} \end{aligned} \tag{5-23}$$

where α is the thermal expansion coefficient, θ_0 is the reference temperature. We assume a linear dilatation evolution. This must be validated by experimental investigation. Note that any of relationship can be adopted at this stage without compromising the genericity of the approach.

To describe the inelastic effects, we introduce an intermediate configuration and the isochoric transformation gradient is multiplicatively decomposed into:

$$\overline{\boldsymbol{F}} = \overline{\boldsymbol{F}}_{\boldsymbol{e}} \cdot \overline{\boldsymbol{F}}_{\boldsymbol{i}} \tag{5-24}$$

where $\overline{\boldsymbol{F}}_{\boldsymbol{i}}$ represents the inelastic transformation and $\overline{\boldsymbol{F}}_{\boldsymbol{e}}$ indicates the elastic counterpart. This decomposition implies that inelastic flows are incompressible.

In the following, we consider the Helmoltz free specific energy ψ to characterize thermodynamic states defined such as:

$$\psi = e - \theta s \tag{5-25}$$

where e is the specific internal energy and s is the specific entropy. Combining the first and second laws of thermodynamics together with the previous expression, on can obtain the Clausius-Duhem inequality. The following relationship holds in an Eulerian configuration:

$$\phi = \sigma : \boldsymbol{D} - \rho\dot{\psi} - \rho s\dot{\theta} - \boldsymbol{q} \cdot \frac{grad\ \theta}{\theta} \geq \theta \ \ \forall \boldsymbol{D}, \dot{\theta}, \boldsymbol{q} \tag{5-26}$$

where ϕ stands for the dissipation, $\sigma(\boldsymbol{x}, t)$ is the Cauchy stress, $\boldsymbol{D} = \dot{\boldsymbol{F}} \cdot \boldsymbol{F}^{-1}$ the Eulerian rate of deformation, $\boldsymbol{q}(\boldsymbol{x}, t)$ is the Eulerian heat flux, $grad$ indicates the gradient operator in current configuration. Introducing the left Cauchy-Green deformation $\overline{\boldsymbol{B}} = \overline{\boldsymbol{F}}\overline{\boldsymbol{F}}^{t}$ and $\overline{\boldsymbol{B}}_{e} = \overline{\boldsymbol{F}}_{e}\overline{\boldsymbol{F}}_{e}^{\ t}$ and assuming that the free energy is a function of $\overline{\boldsymbol{B}}, \theta, J, \overline{\boldsymbol{B}}_{e}$, it can be obtained:

$$\dot{\psi} = \frac{\partial \psi}{\partial \overline{\boldsymbol{B}}} : \dot{\overline{\boldsymbol{B}}} + \frac{\partial \psi}{\partial \overline{\boldsymbol{B}}_{e}} : \dot{\overline{\boldsymbol{B}}}_{e} + \frac{\partial \psi}{\partial J}\dot{J} + \frac{\partial \psi}{\partial \theta}\dot{\theta} \tag{5-27}$$

where:

$$\begin{gathered} \dot{J} = J(\boldsymbol{I} : \boldsymbol{D}) \\ \dot{\overline{\boldsymbol{B}}} = \boldsymbol{L} \cdot \overline{\boldsymbol{B}} + \overline{\boldsymbol{B}} \cdot \boldsymbol{L}^{t} - \frac{2}{3}(\boldsymbol{I} : \boldsymbol{D})\overline{\boldsymbol{B}} \\ \dot{\overline{\boldsymbol{B}}}_{e} = \boldsymbol{L} \cdot \overline{\boldsymbol{B}}_{e} + \overline{\boldsymbol{B}}_{e} \cdot \boldsymbol{L}^{t} - 2\overline{\boldsymbol{V}}_{e} \cdot \overline{\boldsymbol{D}}_{i}^{o} \cdot \overline{\boldsymbol{V}}_{e} - \frac{2}{3}(\boldsymbol{I} : \boldsymbol{D})\overline{\boldsymbol{B}}_{e} \end{gathered} \tag{5-28}$$

with $\overline{\boldsymbol{V}}_{e}$ is the pure deformation coming from the polar decomposition of $\overline{\boldsymbol{F}}_{\boldsymbol{e}} = \overline{\boldsymbol{V}}_{e} \cdot \overline{\boldsymbol{R}}_{e}$, $\overline{\boldsymbol{D}}_{i}^{o}$ is the objective rate of inelastic deformation, defined by: $\overline{\boldsymbol{D}}_{i}^{o} = \overline{\boldsymbol{R}}_{e} \cdot \overline{\boldsymbol{D}}_{i} \cdot \overline{\boldsymbol{R}}_{e}^{\ t}(\overline{\boldsymbol{D}}_{i})$. $\overline{\boldsymbol{D}}_{i}^{o}$ is not an objective rate as the decomposition of equation (5-29) is not unique but defined upon an arbitrary rotation.

Inserting equations (5-22) into equation (5-21) and introducing the results in (5-20) to develop the dissipation as:

$$\begin{aligned}\phi = &\left(\sigma - 2\rho\left(\overline{\boldsymbol{B}}\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}} + \overline{\boldsymbol{B}}_e\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}_e}\right)^D - \rho J\frac{\partial\psi}{\partial J}\boldsymbol{I}\right):\boldsymbol{D}\\ &+2\rho\left(\overline{\boldsymbol{B}}_e\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}_e}\right)^D:\overline{\boldsymbol{D}}_i^o - \rho\left(s+\frac{\partial\psi}{\partial\theta}\right)\dot{\theta} - \frac{grad\,\theta}{\theta}\cdot\boldsymbol{q}\geq 0\end{aligned} \tag{5-30}$$

To proceed further, an additional hypothesis is done: entropy is fully defined from the free specific energy variation and there is no dissipation for the thermodynamic force associated with the thermodynamic flux $\boldsymbol{D}$.

$$s = -\frac{\partial\psi}{\partial\theta} \tag{5-31}$$

$$\begin{aligned}\sigma &= 2\rho\left(\overline{\boldsymbol{B}}\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}} + \overline{\boldsymbol{B}}_e\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}_e}\right)^D + \rho J\frac{\partial\psi}{\partial J}\boldsymbol{I}\backslash\\ \sigma_{eq} &= 2\rho\left(\overline{\boldsymbol{B}}\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}}\right)^D, \sigma_{neq} = 2\rho\left(\overline{\boldsymbol{B}}_e\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}_e}\right)^D, \sigma_{vol} = \rho J\frac{\partial\psi}{\partial J}\end{aligned} \tag{5-32}$$

It remains the following terms for the dissipation:

$$\begin{aligned}\phi &= \phi_m + \phi_\theta \geq 0\\ \phi_m &= 2\rho\left(\overline{\boldsymbol{B}}_e\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}_e}\right)^D:\overline{\boldsymbol{D}}_i^o\\ \phi_\theta &= \left(-\frac{grad\,\theta}{\theta}\right)\cdot\boldsymbol{q}\end{aligned} \tag{5-33}$$

where ϕ_m is the intrinsic dissipation and ϕ_θ denotes the thermal dissipation. It is assumed that ϕ_m and ϕ_θ are independently positives. From equation (5-27), it is possible to define a mechanical thermodynamic force as $\sigma_i = 2\rho\left(\overline{\boldsymbol{B}}_e\cdot\frac{\partial\psi}{\partial\overline{\boldsymbol{B}}_e}\right)^D$ and a heat force as $\mathcal{A}_\theta = -\frac{grad\,\theta}{\theta}$.

The heat equation can be obtained from the first thermodynamic principle (energy conservation), which takes the following local form in the Eulerian configuration:

$$\rho\dot{e} = \sigma:\boldsymbol{D} + \rho r - div\,\boldsymbol{q} \tag{5-34}$$

where r is a volumetric heating source term (defined by unit of volume). The material time derivation of equation (5-19) leads to:

$$\dot{e} = \dot{\psi} + s\dot{\theta} + \dot{s}\theta \tag{5-35}$$

Using equation (5-29) in equation (5-28), one can obtain:

$$\rho\dot{s}\theta = \sigma : \boldsymbol{D} + \rho r - div\, \boldsymbol{q} - \rho\dot{\psi} - \rho s\dot{\theta} \tag{5-36}$$

The material time derivative of entropy is derived as:

$$\begin{aligned} \dot{s} = -\frac{\partial^2\psi}{\partial\theta^2}\dot{\theta} - 2\left(\overline{\boldsymbol{B}}\cdot\frac{\partial^2\psi}{\partial\theta\partial\overline{\boldsymbol{B}}} + \overline{\boldsymbol{B}}_e\cdot\frac{\partial^2\psi}{\partial\theta\partial\overline{\boldsymbol{B}}_e}\right)^D : \boldsymbol{D} \\ + 2\left(\overline{\boldsymbol{B}}_e\cdot\frac{\partial^2\psi}{\partial\theta\partial\overline{\boldsymbol{B}}_e}\right)^D : \overline{\boldsymbol{D}}_i^o - J\frac{\partial^2\psi}{\partial\theta\partial J}(\boldsymbol{I}:\boldsymbol{D}) \end{aligned} \tag{5-37}$$

Finally, the heat equation is obtained by replacing the equation (5-29) with (5-30) and (5-31):

$$\rho C\dot{\theta} = \phi_m + l_m + \rho r - div\boldsymbol{q} \tag{5-38}$$

when C is the specific heat capacity which is defined as:

$$C = -\theta\frac{\partial^2\psi}{\partial\theta^2} \tag{5-39}$$

At last, the latent heat l_m which introduce a mechanical coupling into the heat equation is defined as:

$$l_m = \theta\frac{\partial\sigma}{\partial\theta} : \boldsymbol{D} - \theta\frac{\partial\sigma_{neq}}{\partial\theta} : \overline{\boldsymbol{D}}_i^o \tag{5-40}$$

5.3.2. Application to a Zener thermomechanical model

We first consider the specific free energy spitted into isochoric, volumetric and purely thermal parts as:

$$\psi = \psi_{eq}(\overline{\boldsymbol{B}},\theta) + \psi_{neq}(\overline{\boldsymbol{B}}_e,\theta) + \psi_{vol}(J,\theta) + \psi_\theta(\theta) \tag{5-41}$$

We adopt the following potentials

$$\rho_0\psi_{eq} = C_{10}(\theta)(I_1(\overline{\boldsymbol{B}}) - 3) \tag{5-42}$$

$$\rho_0\psi_{neq} = G(\theta)(I_1(\bar{\boldsymbol{B}}_e) - 3)$$

$$\rho_0\psi_{vol} = \frac{K_v}{2}(J_m - 1)^2$$

$$\rho_0\psi_\theta = C_0\left(\theta - \theta_0 - \theta\log\left(\frac{\theta}{\theta_0}\right)\right) - C_1\frac{(\theta-\theta_0)^2}{2\theta_0}$$

The stress components are therefore expressed as:

$$\begin{aligned}\sigma_{eq} &= 2C_{10}(\theta)J^{-1}\bar{\boldsymbol{B}}^D\\ \sigma_{neq} &= 2G(\theta)J^{-1}\bar{\boldsymbol{B}}_e^D\\ \sigma_{vol} &= K_v(J_m - 1){J_\theta}^{-1}\boldsymbol{I}\end{aligned} \tag{5-43}$$

The specific entropy can also be derived as follows:

$$\begin{aligned}s = -\frac{1}{\rho_0}\Bigg(&\frac{\partial C_{10}}{\partial\theta}(I_1(\bar{\boldsymbol{B}}) - 3) + \frac{\partial G}{\partial\theta}(I_1(\bar{\boldsymbol{B}}_e) - 3) - \alpha K_v\frac{J(J_m-1)}{{J_\theta}^2}\\ &- C_0\log\left(\frac{\theta}{\theta_0}\right) - C_1\frac{(\theta-\theta_0)}{\theta_0}\Bigg)\end{aligned} \tag{5-44}$$

The visco-elastic flow rule is assumed to be as (Maxewell viscosity: $\bar{\boldsymbol{D}}_i^o = (2\tau(\theta))^{-1}\bar{\boldsymbol{B}}_e^D$):

$$\dot{\bar{\boldsymbol{B}}}_e = \boldsymbol{L}\cdot\bar{\boldsymbol{B}}_e + \bar{\boldsymbol{B}}_e\cdot\boldsymbol{L}^t - \frac{2}{3}(\boldsymbol{I}:\boldsymbol{L})\bar{\boldsymbol{B}}_e - \frac{1}{\tau(\theta)}\bar{\boldsymbol{B}}_e^D\cdot\bar{\boldsymbol{B}}_e \tag{5-45}$$

where $\tau(\theta) = \frac{\eta(\theta)}{4G(\theta)}$ is a characteristic time of viscosity.

The heat capacity is obtained using definition of equation (5-33):

$$\begin{aligned}\rho C = \theta K_v\alpha^2(2 - 3J_m){J_\theta}^{-3} &- \frac{\theta}{J}\frac{\partial^2 C_{10}}{\partial\theta^2}(I_1(\bar{\boldsymbol{B}}) - 3)\\ &- \frac{\theta}{J}\frac{\partial^2 G}{\partial\theta^2}(I_1(\bar{\boldsymbol{B}}_e) - 3) + \frac{C_0}{J} + \frac{C_1}{J}\frac{\theta}{\theta_0}\end{aligned} \tag{5-46}$$

And the latent heat term can be expresses be recalling its definition in equation (5-34):

$$\begin{aligned}l_m = 2\frac{\theta}{J}\frac{\partial C_{10}}{\partial\theta}\bar{\boldsymbol{B}}^D:\boldsymbol{D} + \frac{\alpha K_v\theta}{{J_\theta}^2}(1 - 2J_m):\mathbf{D} + 2\frac{\theta}{J}\frac{\partial G}{\partial\theta}\bar{\boldsymbol{B}}^D:\boldsymbol{D}\\ - \frac{\theta}{J\tau(\theta)}\frac{\partial G}{\partial\theta}\bar{\boldsymbol{B}}_e^D:\bar{\boldsymbol{B}}_e\end{aligned} \tag{5-47}$$

The intrinsic dissipation is thus defined as;

$$\phi_m = \frac{1}{J\tau(\theta)}(\bar{\boldsymbol{B}}_e^D : \bar{\boldsymbol{B}}_e^D) \tag{5-48}$$

When $J_m = 0$ or $K_v \to \infty$ the previous formulation need to be reformulated to take into account of quasi-incompressible constraint. Similarly, as done for pure mechanical problems, a Lagrange multiplier p, homogenous to a hydrostatic pressure, is introduced as an additional unknown field. The volumetric free energy turns into:

$$\rho_0\psi_{vol} = \rho_0\tilde{\psi}_{vol}(\theta, J) = (J_m - 1)p - \frac{1}{2}p^2\frac{J_\theta}{K_v} \tag{5-49}$$

The compressibility modulus $K_v$can be viewed as an inverse of a perturbation parameter to enforce the condition $J_m = 1$. The volumetric stress component is therefore redefined:

$$\sigma_{vol} = \frac{p}{J_\theta}\boldsymbol{I} \tag{5-50}$$

The volumetric part of the specific energy, the heat capacity and the latent heat are obtained as follows:

$$\begin{aligned} s_{vol} &= -\frac{1}{\rho_0}\left(-\frac{\alpha p J}{{J_\theta}^2} - \frac{\alpha p^2}{2K_v}\right) \\ \rho C_{vol} &= -\frac{2\alpha^2 p\theta}{{J_\theta}^3} \\ l_{mvol} &= -\frac{\alpha p J}{{J_\theta}^2}\boldsymbol{I} : \boldsymbol{D} \end{aligned} \tag{5-51}$$

We consider a case of a homogenous extension of an elastic volume elemental material. This example intends to validate the constitutive law on a trail test. The deformation gradient is defined as:

$$\boldsymbol{F} = \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \sqrt{J_\theta/\lambda} & 0 \\ 0 & 0 & \sqrt{J_\theta/\lambda} \end{bmatrix} \tag{5-52}$$

We pose for this example: $G(\theta) = 0$, $\tau(\theta) = \infty$, $C_{10}(\theta) = \mu\frac{\theta}{\theta_0}$. Thus, we can derive the following expression for the entropy:

$$\rho_0 s = -\frac{\mu}{\theta_0}\left(J_\theta^{-2/3}\lambda^2 + 2\frac{J_\theta^{1/3}}{\lambda} - 3\right) + \frac{\alpha p}{J_\theta} + \frac{\alpha p^2}{2K_v} \tag{5-53}$$

$$+C_0 \log\left(\frac{\theta}{\theta_0}\right) + C_1 \frac{(\theta - \theta_0)}{\theta_0}$$

Neglecting thermal diffusion and assuming an isentropic (adiabatic) process, equilibrium leads to homogenous thermal and Lagrange multiplier fields. The Lagrange multiplier can be expressed from the boundary conditions $\boldsymbol{\sigma}_{22} = \boldsymbol{\sigma}_{33} = 0$:

$$p = -\frac{2}{3}\mu \frac{\theta}{\theta_0} J_\theta^{1/3} \left(\frac{J_\theta}{\lambda} - \lambda^2\right) \tag{5-54}$$

Replacing the multiplier p in equation (5-47) by (5-48), the isentropic condition leads to a nonlinear function $\theta(\lambda)$. A typical solution that illustrates the thermoelastic inversion is given in Figure 121. The material parameters are set as follows:

$$\begin{gathered} \mu = 1.54\times10^4 \ \ Pa \\ \theta_0 = 293 \ \ K \\ \alpha = 6.7\times10^{-4} \ \ K^{-1} \\ C_0 = 1\times10^3 \ \ J/(m^3K) \\ C_1 = 1\times10^4 \ \ J/(m^3K) \end{gathered} \tag{5-55}$$

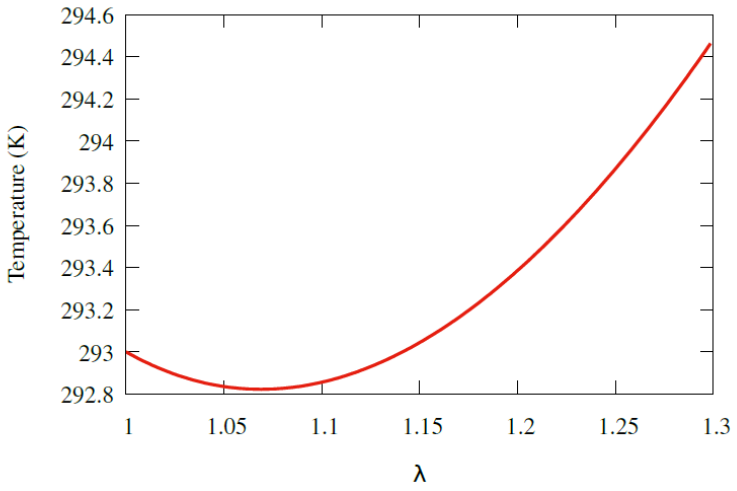


Figure 121. Thermoelastic inversion for an adiabatic stretching of a rubber band.

An alternative of using directly equation (5-47) could be to compute the temperature from the thermal equilibrium reduced equation $\rho C\dot{\theta} = l_m$. We therefore express the Eulerian rate of deformations:

$$D = \begin{bmatrix} \frac{\dot{\lambda}}{\lambda} & 0 & 0 \\ 0 & -\frac{\dot{\lambda}}{2\lambda} + \frac{\alpha\dot{\theta}}{2\sqrt{\lambda J_\theta}} & 0 \\ 0 & 0 & -\frac{\dot{\lambda}}{2\lambda} + \frac{\alpha\dot{\theta}}{2\sqrt{\lambda J_\theta}} \end{bmatrix} \tag{5-56}$$

The latent heat and heat capacity are derived as follows:

$$\begin{aligned} \rho C &= \frac{C_0}{J_\theta} + \frac{C_1\theta}{J_\theta\theta_0} - 2\alpha^0 p\frac{\theta^3}{J_\theta} \\ l_m &= 2\frac{\theta}{J_\theta}\frac{\mu}{\theta_0}(\bar{\boldsymbol{B}}^D : \boldsymbol{D}) - \frac{\alpha p\theta}{{J_\theta}^2}(\boldsymbol{D} : \boldsymbol{I}) \end{aligned} \tag{5-57}$$

One can therefore exhibit a differential equation in θ. With the same material parameters depicted in equation (5-49), this equation can be numerically solved for a given time-dependent $\lambda(t)$, and it provides the results in Figure 122.

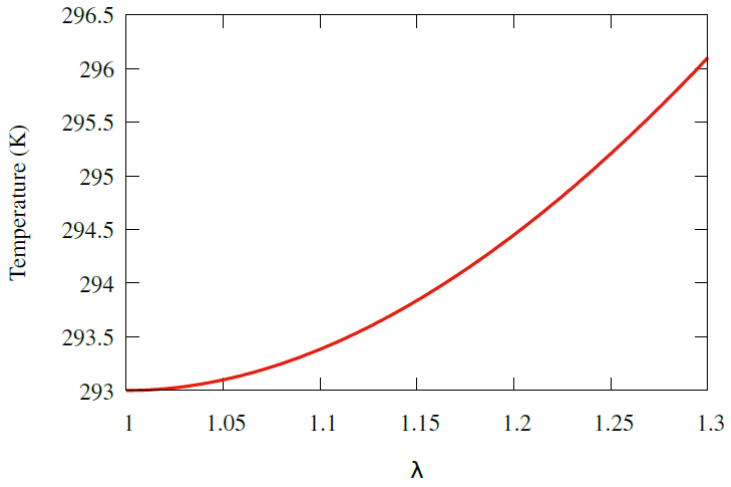


Figure 122. Thermoelastic inversion for an adiabatic stretching of a rubber band.

5.3.3. Numerical applications

We presented several numerical test based on the previous model derived from NURBS based IGA.

✧ **3D shear test problem**

For the first test, we remove the dependence of the free energy in equation (5-41) to the temperature. A simple hyperelastic model is obtained with standard linear viscosity. Our objective is to assess the NURBS based isogeometric analysis on simulating the dissipative behavior of rubber-like materials with a simple hyperelastic Zener model. The material parameters are given:

$$C_{10} = 0.1, C_{01} = 0.2, K_v = 1500$$
$$\eta_0 = 0.8, G_0 = 0.4 \tag{5-58}$$

The computational domain a thin bloc of elastomer defined with quadratic NURBS. The dimensions are given in Figure 123. Its lower surface is clamped. And the upper surface is submitted to a combination of, in a first stage, an imposed vertical displacement, and in a second phase, to a cyclic imposed shear. Both displacements are defined by the time-dependent functions $u_x(t)$ and $u_y(t)$ in Figure 123.

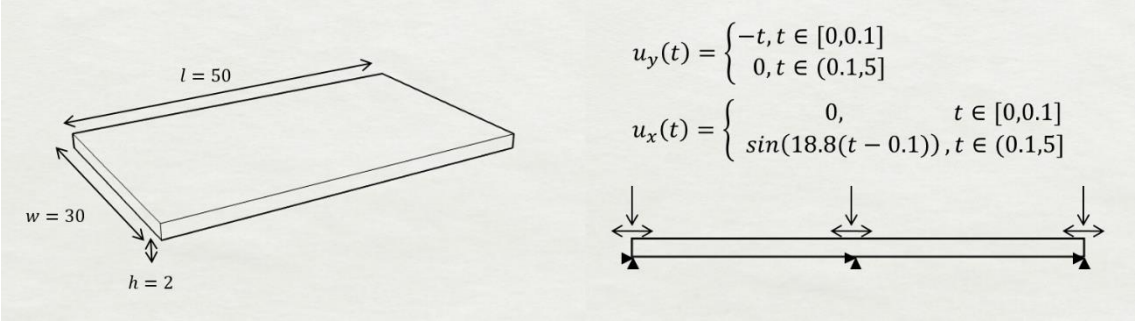


Figure 123. 3D shear test problem definition.

In Figure 124, the mesh for the patch used here is a $8\times2\times8$ elements. The displacement vectors are plotted in the same figure on the deformed configuration.

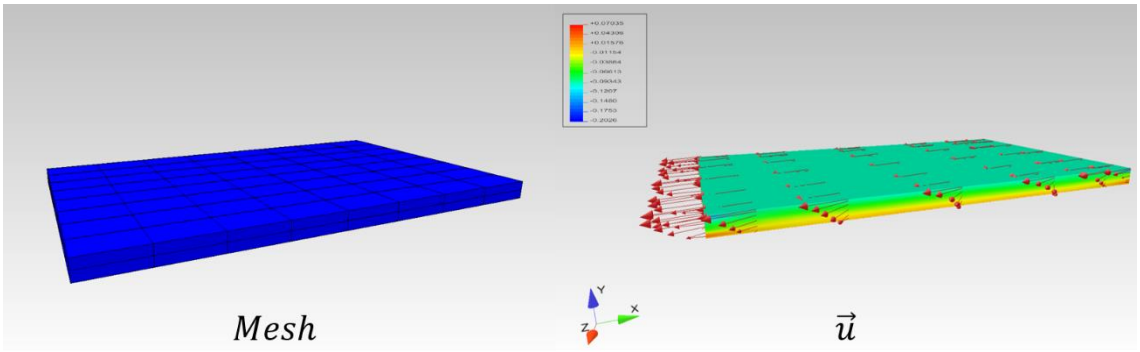


Figure 124. Mesh and displacement vector field at control points.

In Figure 125, we plot the stress components and Von-Mises stresses on the final configuration. Since the material is modeled by a Zener type viscosity, the system is not conservative. A portion of mechanical energy coming from the loading is stored by the structure as potential energy. The complement is dissipated through material's viscosity. The dissipation is exhibited by plotting the hysteresis of the main stress component σ_{xy} during the shear loading with respect to the horizontal displacement u_x for the upper center as shown in Figure 126. The mean slope of the hysteresis curve decreases during the first pseudo periods to attenuate the structure's reactions to the compression stabilizes. Thus, the energy lost by material's viscosity can be clearly observed.

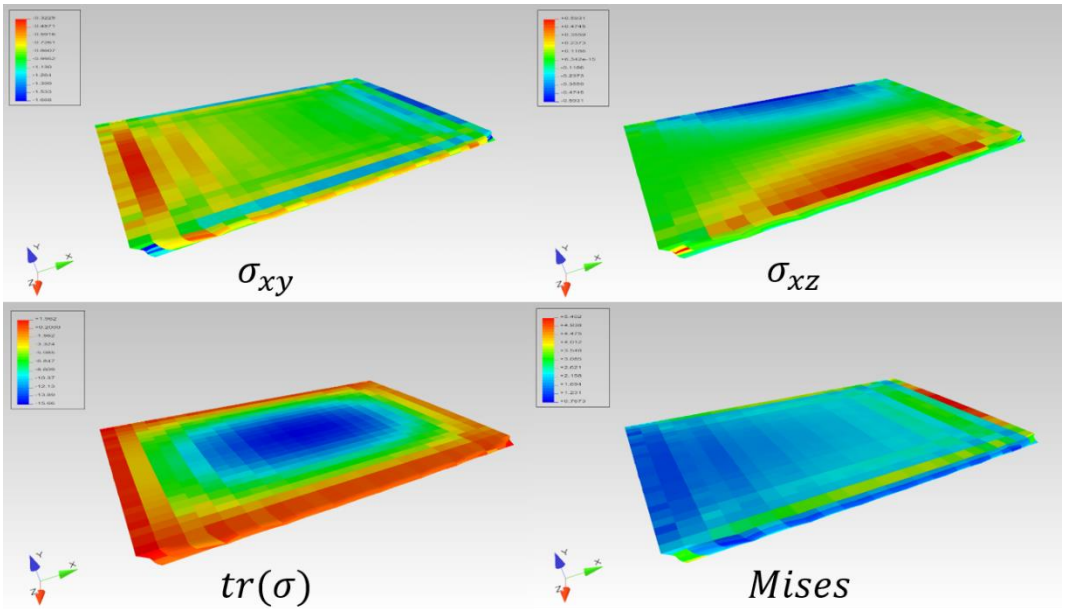


Figure 125. stress distributions on deformed configuration.

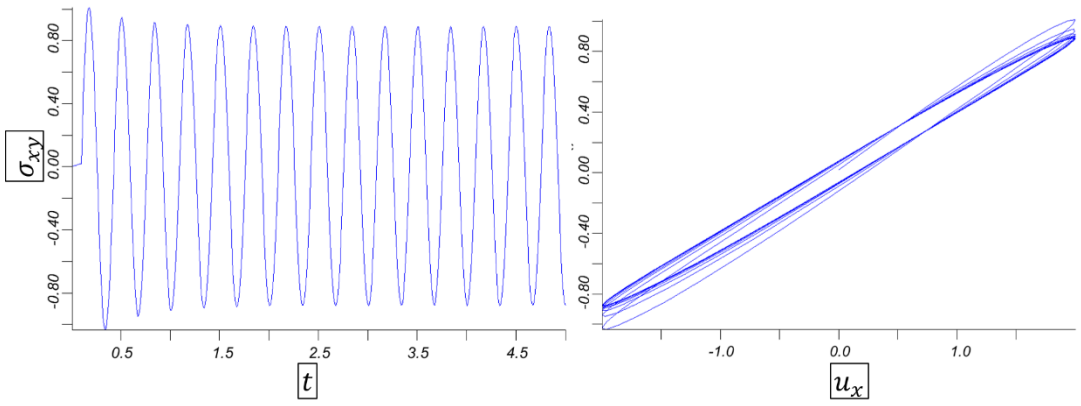


Figure 126. $\boldsymbol{\sigma_{xy}}$ at upper surface center variation with $\boldsymbol{t}$ (left) and $\boldsymbol{u_x}$(right).

✧ 2D dumbbell sample under time varying loading

This application target at simulating the self-heating effect of filled-elastomer. Thus, the model we propose here is fully developed. The 2D geometry and the boundary conditions are described in Figure 127. No heat exchanges are allowed at the boundaries (adiabatic case).

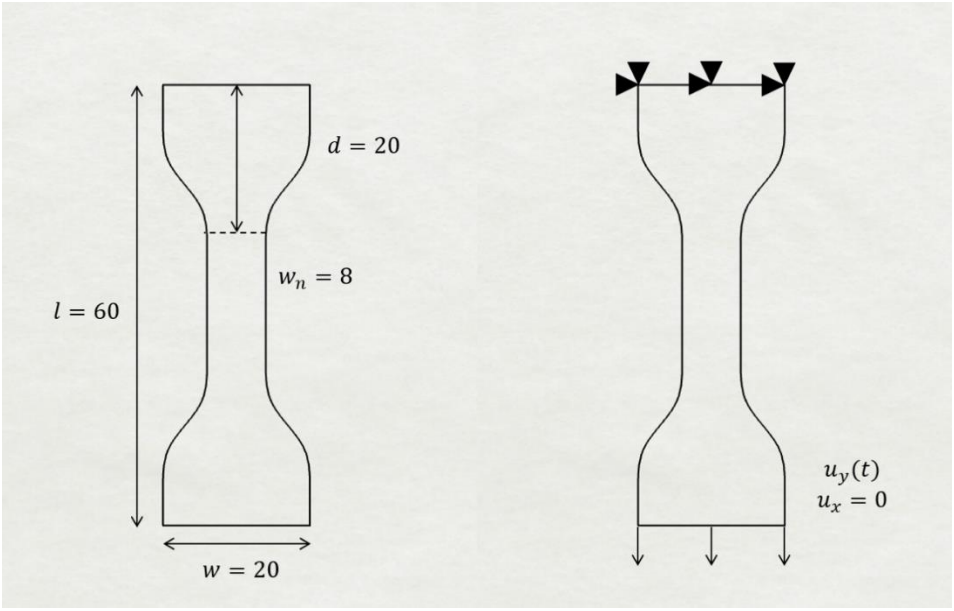


Figure 127. Thermo-elastic viscous model: Problem definition.

The top boundary of the dumbbell sample is clamped and a vertical time dependent displacement $u_y(t)$ is imposed on the lower side.

$$u_y(t) = \begin{cases} -0.5t,\ t \in [0,10] \\ -5 + sin(18.8(t-10)),\ t \in (10,13] \end{cases} \quad (5\text{-}59)$$

Basically, the dumbbell sample is first stretched, and then submitted to a high frequency cyclic loading. The vertical displacement of the middle point on the bottom face is plotted in Figure 128. The material parameters are the same as the one defined in equation (5-55) and the coefficient for viscosity is set to $G(\theta) = 8.0\times10^3$.

The mesh and the displacement vectors are shown in Figure 129. The temperature and stresses are given in Figure 130. The central part of the specimen undergoes a nearly uniform deformation. We observe a stress concentration at both end of the dumbbell sample because of the boundary condition (clamped). No spurious oscillations for stresses can be observed for σ_{yy} and σ_{xy}.

The classical temperature inverse phenomenon produced by entropic thermoelasticity is observed during the first stage of loading. Figure 131 demonstrates a result coherent with that in Figure 121. Given the boundaries are adiabatic, the dissipation (due to viscosity) turns into thermal energy and the temperature goes up during the second phase of loading (high frequency cyclic solicitations). We can see in Figure 131 that the temperature evolution follows the cyclic loading one. (see Figure 128). The mean value of the temperature keeps rising. This is the so-called self-heating phenomenon of filled elastomer.

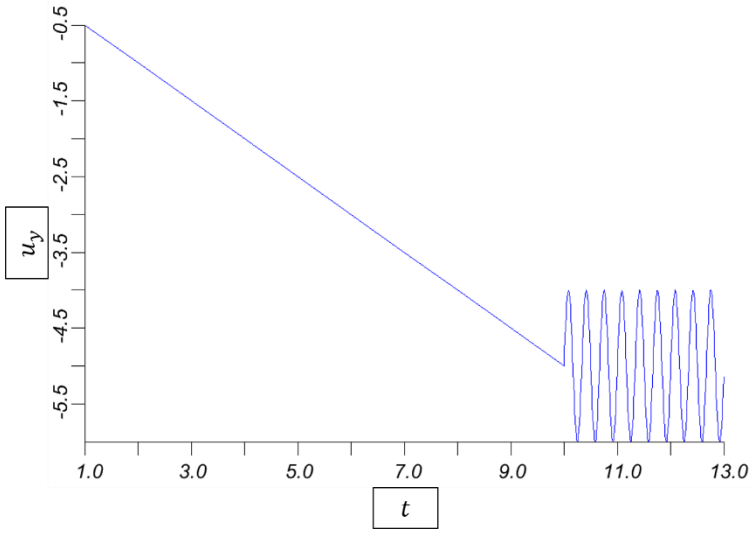


Figure 128. Imposed vertical displacement $\boldsymbol{u_y}$with respect to time $\boldsymbol{t}$.

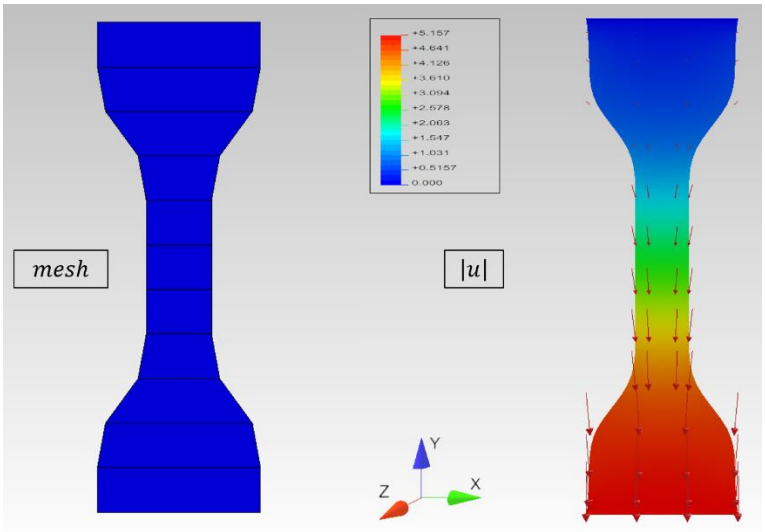


Figure 129. Mesh and displacement vector fields at control points on deformed configuration.

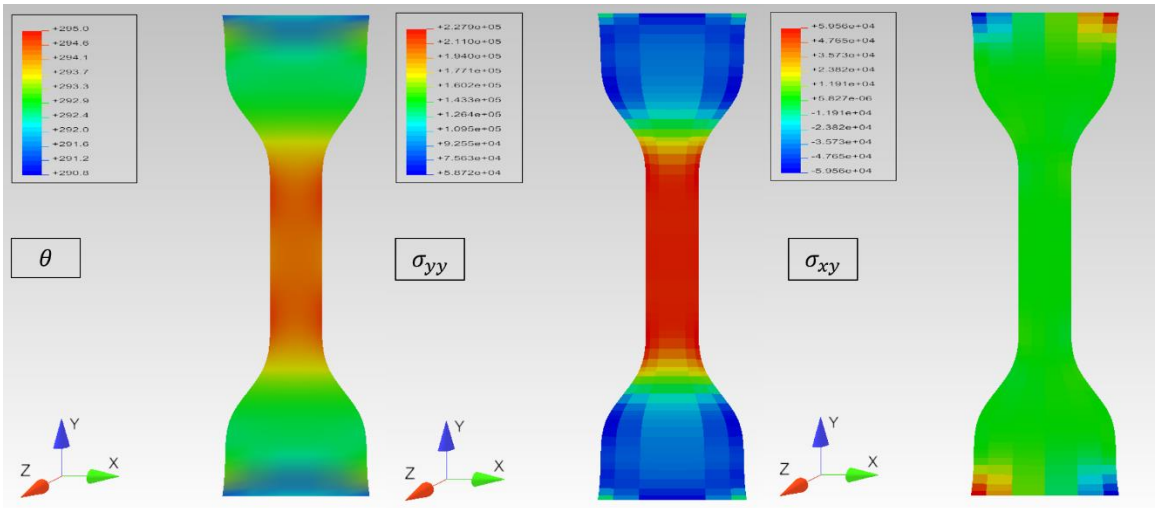


Figure 130. Temperature and stress components on deformed configuration.

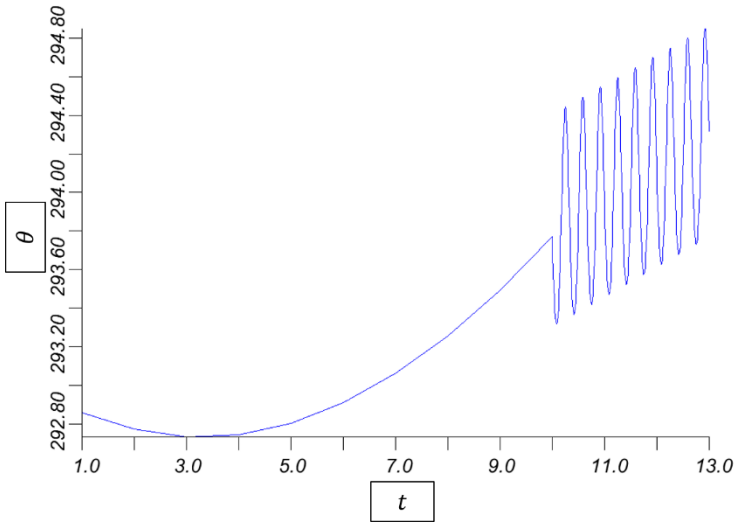


Figure 131. Center point temperature increases with the periodic solicitation.

✧ 3D dumbbell sample

We consider a 3D dumbbell sample of thickness equals to 2 (H2) described in Figure 132 (it is the 3D version of the specimen used in the previous subsection). The top surface is clamped and a vertical displacement is imposed at the bottom surface.

The thermoelastic model with Zener type viscosity is chosen, and the material parameters have been set as in equation (5-55).

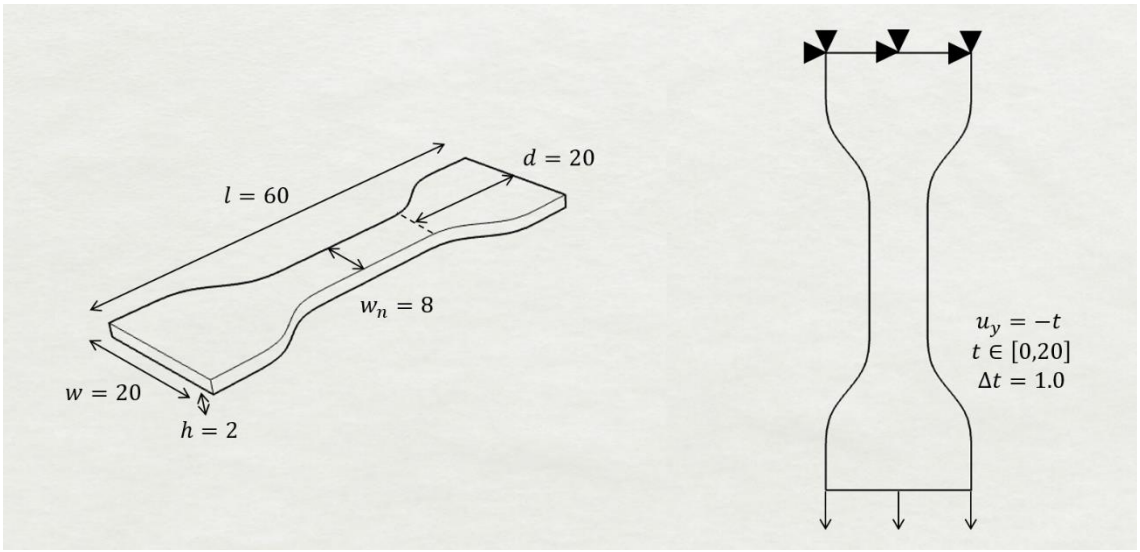


Figure 132. Stretching 3D: geometry dimensions and problem definition

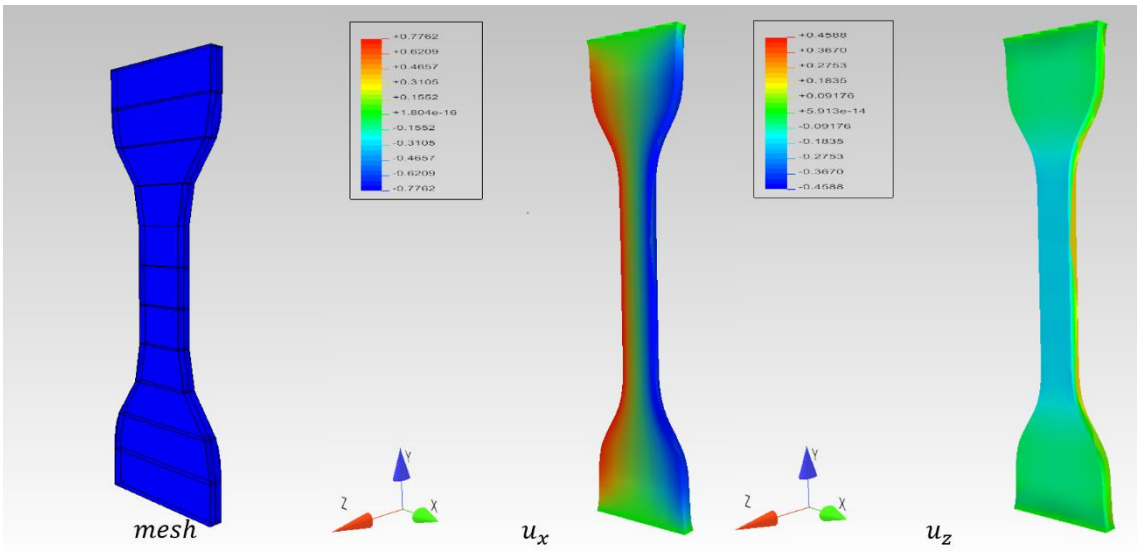


Figure 133. Mesh and displacement components on deformed configuration.

The mesh and displacement are plotted in Figure 133 and the temperature and stresses in Figure 134. the deformation is uniform on the central part of the specimen. As expected, the temperature varies averagely at sample's narrow part. The temperature's variation with respect to time is given in Figure 135. It increases during the elongation which is in good agreement with the simplified analytical solution depicted in Figure 122.

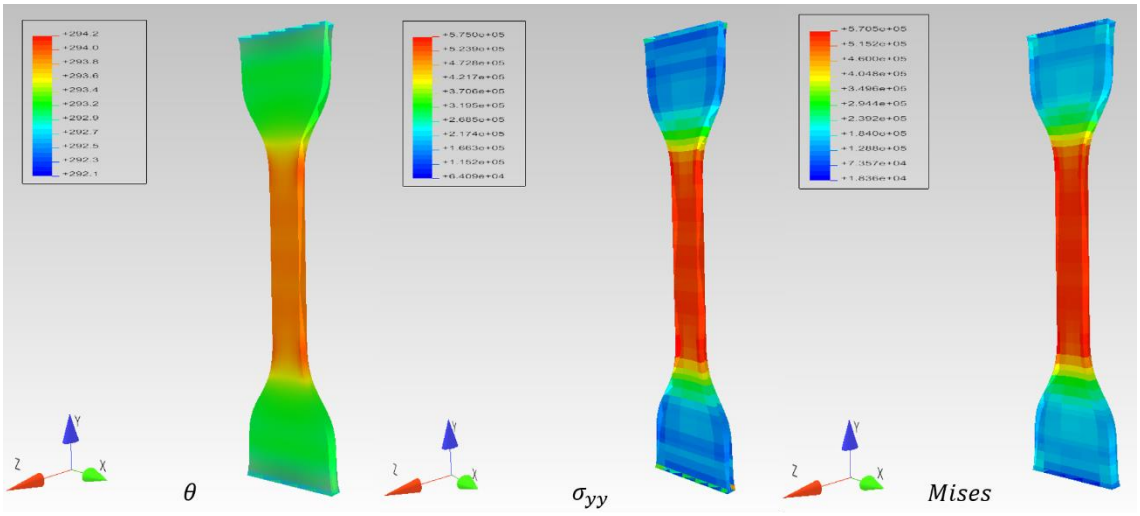


Figure 134. Temperature and stress on deformed configuration.

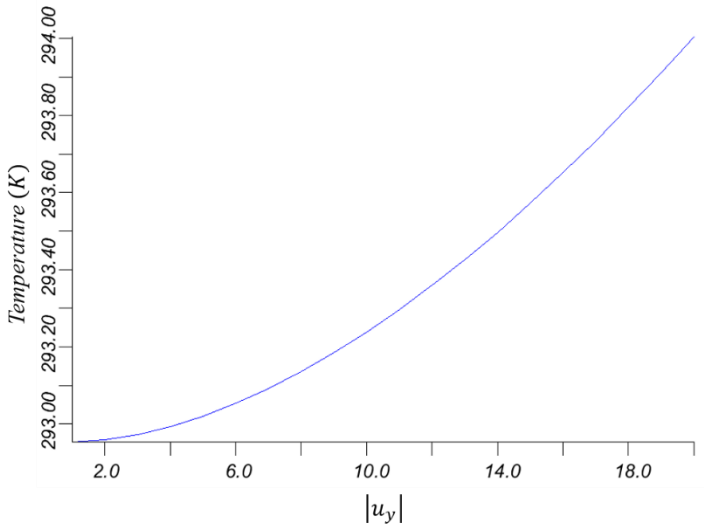


Figure 135. Variation of temperature along with vertical displacement at the center of specimen.

5.4. Conclusion

In this section, we have presented several numerical results for thermomechanical problems at small and finite strains, linear and nonlinear. We will adopt simple equal order interpolation for displacement and temperature with either by increasing the multiplicity of inner knots by 1 for displacement, or by halving the knot span for displacement. These elements respectively correspond to $\left(\boldsymbol{Up_0^2}, \boldsymbol{Tp_0^1}\right)$ and $\left(\boldsymbol{Up_1^1}, \boldsymbol{Tp_0^1}\right)$ where p is the order of the patch. As shown in this section (thick cylinder test), in some cases, the combination $\left(\boldsymbol{Up_1^1}, \boldsymbol{Tp_0^1}\right)$ may be less efficient than $\left(\boldsymbol{Up_0^2}, \boldsymbol{Tp_0^1}\right)$. An industrial level application: incompressible viscous thermo-hyperelastic model is adopted and evaluated with the preferred approach. Some well-known phenomena of elastomer such as the thermo-elastic inversion and the self-heating are correctly reproduced through the numerical experiment.

Conclusion

In this work, we targeted to study numerical performances of IGA in the context of the simulation of rubber-like materials for fully coupled multiphysics problems, and to provide users with a flexible simulation tool for IGA and FEM.

First, we propose, a programming paradigm of the IGA in an existing Java object-oriented hierarchy initially designed for solving multi-fields coupled problems at finite strains. We develop an approach that fully take benefit of the original architecture to reduce developments for both FEM and IGA (one problem developed in FEM can be run in IGA and vice versa). The integration of IGA has been done by remarking that the element in the finite element sense and the element in the IGA sense are both support of the integration scheme despite they do not belong to the same space (physical space for FEM and parameter space for IGA). The multilayer architecture of the code FEMJava has been adapted the patch oriented definition of both the geometry and the fields (objet **Field**). The wide variety of 2D and 3D problems solved within FEMJava in this report, including the appendices, (i.e. linear elasticity, hyperelasticity, incompressible and quasi compressible material, thermomechanical coupling, small and large strains with highly deformable media, patches coupling with the Nitsche's method in thermomechanics at large strains…) show the pertinence of the approach.

Second, we investigate volumetric locking issues one of the main numerical problem coming from the incompressible or quasi-incompressible character of rubber-like material. Locking is persisting for low order NURBS element observed with standard displacement formulation as finite elements. To cure the problem, we adopt two-fields mixed formulations (displacement/pressure) for the sake of simplicity and target at assessing different discretizations in stability (*inf-sup* condition). In the context of IGA, a wide range of strategies to relaxing the incompressible constraint for displacement can be developed based on basic operations that are knot insertion and degree elevation. The basic idea we followed is to first to increase the internal knot's multiplicity or to subdivide the patch for displacements keeping the highest inter-element continuity for accuracy purposes. These

ideas that are directly inspired from patches properties, have been found in the literature for the Stokes problem and extended to large strain in solid mechanics. The comparison between the two-fields mixed formulation and a strain projection method is lead at small and large strains. For incompressible linear elasticity at small strain, we have shown that a wide range of combination of patches for displacement and pressure exhibit an optimal rate of convergence for the $\boldsymbol{L^2}$-norm of displacement and pressure errors: $\left(\boldsymbol{u\ p_1^1, p\ p_0^1}\right)$ $\left(\boldsymbol{u\ p_0^2, p\ p_0^1}\right)$ $\left(\boldsymbol{u\ p+1_1^1, p\ p_0^1}\right)$. $\left(\boldsymbol{u\ p+1_0^2, p\ p_0^1}\right)$. For the sake of simplicity, we can adopt equal interpolation discretization. At large strains for quasi incompressible material, locking is avoided with the chosen equal order interpolations: $\left(\boldsymbol{u\ p_1^1, p\ p_0^1}\right)$ $\left(\boldsymbol{u\ p_0^2, p\ p_0^1}\right)$ such as with projection techniques but convergence of the nonlinear algorithm is twice faster (number of iterations) compared to projection technique. A test on a highly deformed bloc in compression confirm the adequate choice for elements choice. Even if stability is not mathematically proved here the check of the convergence rate or errors are provide a trustable information. They should probably be fulfilled by equivalent convergence tests for the stresses or H_1 norm of displacement error. But, we are confident given the plot of stresses for the different tests does not exhibits spurious oscillations.

At last, we adopt a similar strategy for thermomechanical problem at small and large strains which is original. In the context two-fields formulation, displacement/temperature, the LBB stability condition must be fulfilled to guaranty stability and a similar analysis lead us to adopt this simple choice. We obtained optimal convergence for a linear thermoelasticity at small strains test on a square domain for the $\boldsymbol{L^2}$-norm of displacement and temperature errors for elements $\left(\boldsymbol{U\ 1_1^1, T\ p_0^1}\right)$ $\left(\boldsymbol{U\ p_0^2, T\ p_0^1}\right)$. We note that if the thermomechanical coupling becomes too strong, the formulation fails to converge, this must be kept in mind when computing. For a similar test on a thick cylinder, the optimal convergence is not achieved for displacement. This might be cause either by the rather strong thermomechanical coupling or by the mapping from the parameter space to the physical one. Additional investigations are necessary at this stage to get confirmation. Despite this, the results are globally satisfactory. A preliminary study for patches coupling in the context of nonlinear thermomechanics exhibit promising results (see complements in Appendix A). Additional qualitative tests show quite good results for displacement,

temperature and stresses: fully coupled time-dependent thermomechanical formulation at large strains (tested at small strain), thermomechanical entropic elasticity. Note that both reproduce expected physical results with coarser meshes as for the FEM. At last, an incompressible viscous thermo-hyperelastic model is evaluated in the IGA framework with the proposed approach. Preliminary results are very promising compared to what we were used to obtain with the FEM in similar situation (see e.g. Nguyen et al. . [268]).

Finally, the global aim of our work is achieved in the sense we do have developed trustable IGA mixed formulations for thermomechanical problems at large strains. The next step would be to extend the framework to chemo-thermo-mechanical formulations at large strain such as in Nguyen et al. . [268].

From our point of view, we are convinced that IGA approaches open promising tracks in the design of modern simulation tools for fully coupled multiphysics problems. The exact geometric representation guaranties the numerical solution won't be disturbed by the approximations at the geometry level. With the exact geometry at the analysis step, the refinement procedure could be facilitated since redundant communication with CAD software is not mandatory. The second important characteristic of IGA is the local refinement ability that could improve a lot the quality of solution with really coarse meshes**.** It is an important capability for which the NURBS are short. The realistic engineering level simulation usually needs to solve problems defined on complex geometries, such as a ship propeller in Figure 14 which can be model with only one T-splines patch. With NURBS based isogeometric analysis, this kind of problems can only be solved with multi-patch. That means additional computational cost is inevitably paid to impose the solution's inter-patch continuity. The division caused by geometry modeling may cause errors in the numerical solution, thus degrade the accuracy of the solution. The most important challenge at this remains the volume parametrization. As today's CAD software model, a three-dimensional part by define its closed exterior surfaces. The most significant challenge facing isogeometric analysis is developing three-dimensional spline parameterizations from those surfaces (see e.g. H. Al Akhrasa et al. [154]).

Appendix A. Nitsche's method for IGA of thermo-elastic problem and finite strain hyperelastic problem

Introduction

As presented in the state of the art on computational geometry technologies for isogeometric analysis in section 1.3.2, the objects of complex topologies are usually represented with multiple patches of NURBS in the CAD world. That makes the implementation of IGA with high order inter-patch continuity an important research topic.

Under this topic, there are two main kinds of solution proposed. The first consist in employing more advanced computational geometry technology which allows to describe objects with complex topological structures through only one patch. Thus no more inter-patch coupling is necessary. We can cite those previously introduced works: Sederberg et al [136], [137], Li et al [138], [140] and Scott et al [139], [143], [271] as examples for T-splines; and also Deng et al [152] for PHT-splines. For a more detailed review, please refer to the section 1.3.2.

Another solution focus on conserving the NURBS for geometric technology (commonly used in CAD software), but improving the formulation to maintain inter-patch continuity. When the patches have matching mesh at interfaces, un exact multi-constraint method introduced in section 2.3 of book [7] has been directly applied by superposing theses interfaces related control points. These border control points are interpolant as the knot vectors are open. This method generally results in a c^0 order continuity across the patch boundaries, thus may lose the interesting property of high inter-element continuity of NURBS at the patch level. Rather than gluing patches strongly, the application of Nitsche's method to coupling conforming or non-conforming NURBS patches in a weak sense has been studied in Apostolatos et al [272], Nguyen et al [273] and Ruess et al [274] for standard displacement formulation in the context of linear elasticity. Additionally, the

works of Du et al [275] and Guo [276] tried to attach shell and plate patches for NURBS based IGA. Compared to the strong coupling method, the great advantage of Nitsche's method is its suitability for non-conforming discretization.

Through this paragraph, we present some initial endeavors to extend Nitsche's method to the multi-field formulation (linear thermoelasticity) and to non-linear purely mechanical formulation (finite elasticity) in the context of NURBS based IGA. A brief discussion about the influence of this method on the convergence rate of solution's error estimation has already been opened in section 5.1.3.

Structure of this paragraph is as follows: Part 2 presents the extension from linear elastostatic problem to linear static thermoelastic problem for NURBS based IGA along with two numerical tests. Part 3 introduces the finite strain version of Nitsche's method for hyperelastic models.

Nitshce's method for thermoelasticity at small strain

We reclaim the problem studied in section 5.1.1 but with two patches of NURBS to describe the quarter of cylinder. Figure 136 recalls the geometric dimensions, boundary condition, and the supplementary patch interface designated by Γ_c. The material parameters are identical as those in equation (5-7).

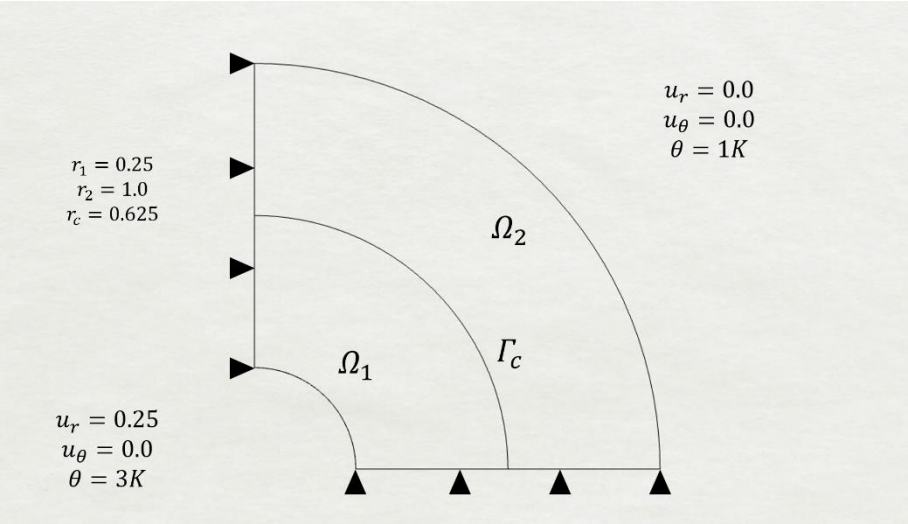


Figure 136. Thermoelastic cylinder in two patches: Problem definition

In addition to the governing equations (5-1) for linear thermoelasticity, the coupling of this two-fields problem on common boundary Γ_c through Nitsche's Method is formulated by the followings equations:

$$\begin{aligned} \boldsymbol{u}^1 - \boldsymbol{u}^2 &= 0 && \text{on } \Gamma_c \\ (\sigma^1 + \sigma^2) \cdot \boldsymbol{n}^1 &= 0 && \text{on } \Gamma_c \\ \theta^1 - \theta^2 &= 0 && \text{on } \Gamma_c \\ (\boldsymbol{q}^1 + \boldsymbol{q}^2) \cdot \boldsymbol{n}^1 &= 0 && \text{on } \Gamma_c \end{aligned} \tag{0-1}$$

$\boldsymbol{n}^1$ denotes the outward normal vector for the subdomain Ω_1 on interface Γ_c. It can be replaced by the normal vector of the subdomain Ω_2, but this does not change the equation since we have $\boldsymbol{n}^1 = -\boldsymbol{n}^2$ on Γ_c.

By defining the jump operator for displacement $\boldsymbol{u}$ and temperature θ, and the average operator for stress σ and heat flux $\boldsymbol{q}$ as:

$$\begin{aligned} \boldsymbol{u}^1 - \boldsymbol{u}^2 &= \chi^u \\ \frac{1}{2}(\sigma^1(\boldsymbol{u}) + \sigma^2(\boldsymbol{u})) \cdot \boldsymbol{n}^1 &= \xi^u \\ \theta^1 - \theta^2 &= \chi^\theta \\ \frac{1}{2}(\boldsymbol{q}^1(\theta) + \boldsymbol{q}^2(\theta)) \cdot \boldsymbol{n}^1 &= \xi^\theta \end{aligned} \tag{0-2}$$

The Nitsche's method is formulated in the variational form as following:

Find $u \in S_u$ and $\theta \in S_\theta$ such as $\forall \delta u \in S_u^0$ and $\forall \delta\theta \in S_\theta^0$

$$\begin{cases} \mathcal{H}^u(\boldsymbol{u}, \theta, \delta\boldsymbol{u}, \delta\theta) - \int_\Gamma \xi^u \chi^{\delta u} d\Omega - \int_\Gamma \chi^u \xi^{\delta u} d\Omega + \int_\Gamma \beta^u \chi^u \chi^{\delta u} d\Omega = 0 \\ \mathcal{H}^\theta(\boldsymbol{u}, \theta, \delta\boldsymbol{u}, \delta\theta) - \int_\Gamma \xi^\theta \chi^{\delta\theta} d\Omega - \int_\Gamma \chi^\theta \xi^{\delta\theta} d\Omega + \int_\Gamma \beta^\theta \chi^\theta \chi^{\delta\theta} d\Omega = 0 \end{cases} \tag{0-3}$$

where $\chi^{\delta u}$, $\chi^{\delta\theta}$ and $\xi^{\delta u}$, $\xi^{\delta\theta}$ represent the variational version of operators defined in equation (7-2).

The coefficients of the last terms in the variational equations β^u and β^θ are the stabilization parameter. It can be shown there exists an interval of choice for the

stabilization parameters such that the bilinear form is coercive. In the work of Apostolatos et al [272], this parameter has been determinated for linear elastic problem by the relation:

$$\|\xi^u(\delta\boldsymbol{u}^1, \delta\boldsymbol{u}^2)\|_{0,\Gamma_c} \leq c^u(a(\delta\boldsymbol{u}, \delta\boldsymbol{u}))^{1/2} \quad (0\text{-}4)$$

and β^u is required to be $\beta^u \geq 2{c^u}^2$where $c^u > 0$. This parameter can be derived by solve the discretized version of the system in equation (7-18). The c^u is set as the maximal eigenvalue of the discretized system. For temperature, the same process is necessary to be done to find the appropriate stabilization parameter β^θ.

The first numerical with this method has already been discussed in section 5.1.3 for conforming and non-conforming mesh. For the results and conclusions, please refer to the end of section 5.1.3. Briefly speaking, the numerical solutions for displacement and temperature are correct as well as the post-processed thermoelastic stress. However, the convergence rates of the $\boldsymbol{L^2}$-norm of displacement and temperature errors is affected by the coupling method. The convergence is good but does not achieve the optimal rate expected for such choices of patches. The C^1 continuity imposed by Nitsche's method probably affect this convergence. Note that on this example, the convergence rate of the $\boldsymbol{L^2}$-norm of displacement error is limited to 2, and the convergence rate of the $\boldsymbol{L^2}$-norm of temperature errors is limited to 3.

Here, a more realistic problem originally solved in Margonari [277] with the same and different material models has been studied. Considering a vessel containing a fluid at high temperature and pressure, the geometric dimension and boundary conditions are depicted in Figure 138. We reuse the same material parameter for steel and insulation as the in the paper Margonari [277] (see the Figure 137).

Material	Density [kg/m^3]	Specific heat [J/kg°C]	Thermal conductivity [W/m°C]	Young modulus [N/m^2]	Poisson ratio [---]	Thermal expansion coeff. [1/°C]
Steel	7850	434	60.5	$2.0\cdot10^{11}$	0.30	$1.2\cdot10^{-5}$
Insulation	937	303	0.5	$1.1\cdot10^{9}$	0.45	$2.0\cdot10^{-4}$

Figure 137. thermal and mechanical material parameters.

Firstly, the vessel is assumed been entirely constructed with steel. Within FEMJava, we compute this problem through one subdomain of FEM and 6 patches of NURBS based IGA the same material, formulation and algorithm, i.e. here with the same software (the

same piece of code for the formulation). Thus, this allows us to compare the results of FEM and IGA on a relatively complex problem.

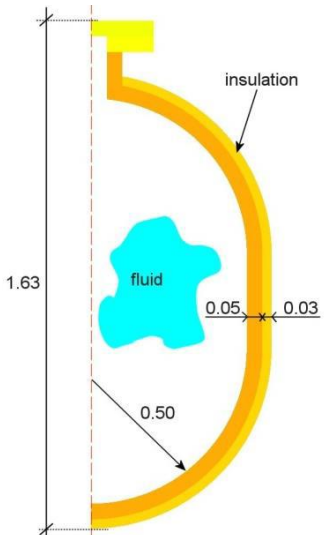


Figure 138. Thermoelastic vessel problem definition from [277]

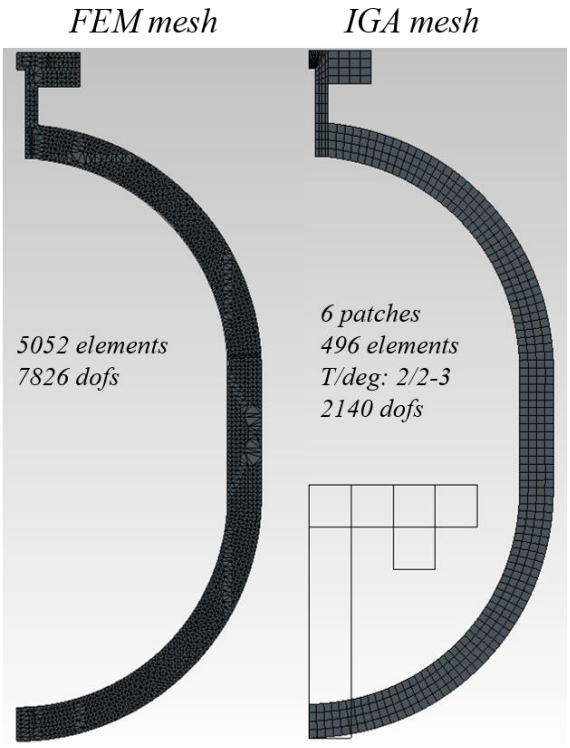


Figure 139. Steel vessel meshes with FEM and IGA.'

The meshes of FEM and IGA models have both been described in Figure 139. There is only one subdomain with a 5052 linear triangular elements mesh for the FEM model. The number of degrees of freedom is 7826. For the IGA model, quadratic elements for temperature and quadratic/cubic element for displacement are set. This mesh has totally 6 patches, 5 inter-patch interfaces, 496 elements and 2140 degrees of freedoms.

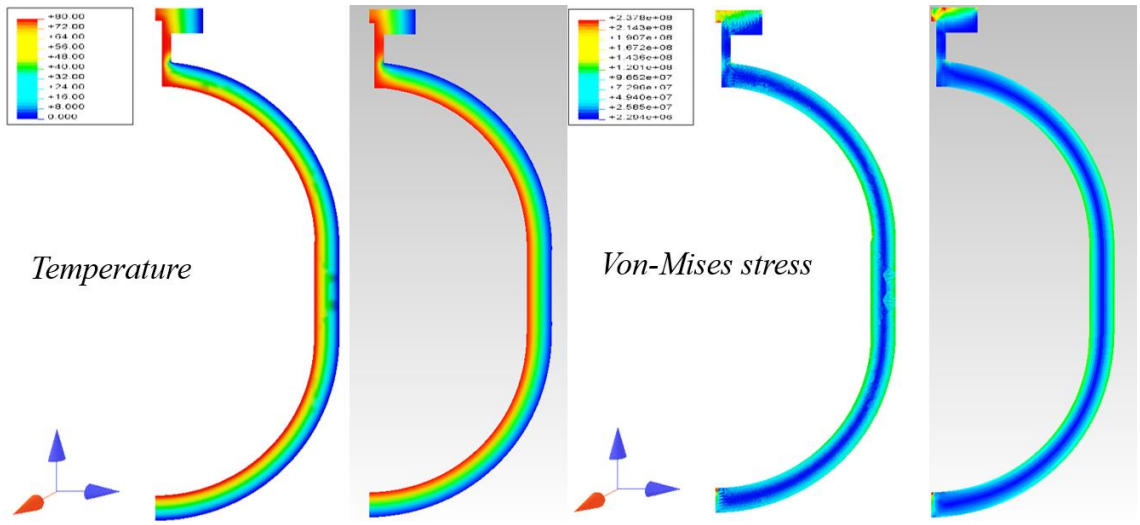


Figure 140. Steel vessel solutions for temperature and Von-Mises stress.

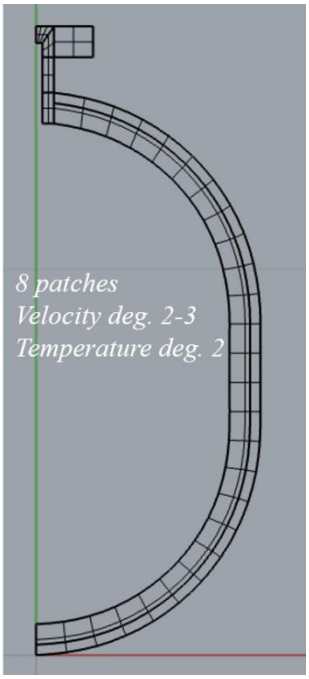


Figure 141. Steel and insulation vessel: multi-patch NURBS mesh for IGA.

The numerical results of temperature and Von-Mises stress with FEM and IGA have been displayed in Figure 140. We observe that NURBS based IGA gives a better displacement field than that of FEM. The results of IGA presents a higher concentration of stress close to the head part of the vessel.

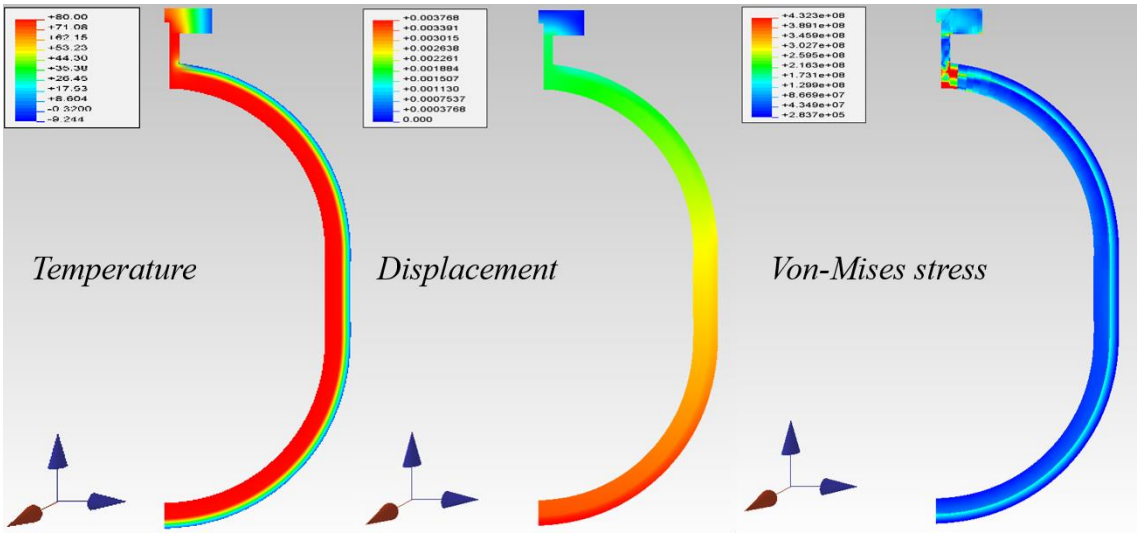


Figure 142. Steel and Insulation vessel: solutions for temperature, displacement and Von-Mises stress.

The second numerical example respect exactly as the problem definition described in Figure 138. The inner layer of the vessel is made of steel, the support and outer part layer are made of thermal insulation whose parameters have been given in Figure 137.

A 8 patches mesh is shown in Figure 141. The interpolation degrees for displacement and temperature are chosen identical to the first test. Figure 142 demonstrates the numerical solutions for temperature, displacement norm and Von-Mises stress. It is observed that the type of Nitsche's method that we have implemented for linear thermoelasticity gives smooth results for temperature and displacement comparing to the results in Margonari [277]. Nevertheless, there are important non-physical oscillation on the stress evaluation, especially on the intersection part where inter-patch coupling is concentrated.

We believe that the method applied to coupling patches in the context of a multi-field formulation succeeds partially for these primary unknown fields such as temperature and displacement in the linear thermoelasticity case. The stress evaluation fails, mainly in the case where the materials parameters between two patches are too different. This drawback

may be overcome by the befitting choice of stabilization parameters β^{θ} and β^{u}. Thus it demands a thorough study on this topic.

Nitsche's method for finite elasticity

The finite deformation version of the Nitsche's method has been implemented in a similar way as the one of small strain. The strong form of balance equations is known as:

$$\begin{aligned} Div\,\boldsymbol{P} + \boldsymbol{B} = \boldsymbol{0}\ \ in\ \Omega_{01} \cup\ \Omega_{02} \\ \varphi = \bar{\varphi}\ \ on\ \partial\Omega_{0D} \\ \boldsymbol{P}\cdot\boldsymbol{N} = \overline{\boldsymbol{T}}\ on\ \partial\Omega_{0N} \end{aligned} \tag{0-5}$$

The continuity of displacement and traction vector on the common boundary Γ_c is formulated as follows:

$$\begin{aligned} \varphi^1 = \varphi^2\ \ on\ \Gamma_{0c} \\ (\boldsymbol{P}^1 + \boldsymbol{P}^2)\cdot\boldsymbol{N}^1 = \boldsymbol{0}\ \ on\ \Gamma_{0c} \end{aligned} \tag{0-6}$$

Weak formulation of the balance equation with continuity conditions is thus derived as:

$$\begin{aligned} \int_{\Omega_{01}\cup\,\Omega_{02}} \boldsymbol{P} : Grad(\delta\varphi)\,dV - \int_{\Omega_{01}\cup\,\Omega_{02}} \boldsymbol{B}\cdot\delta\varphi dV - \int_{\partial\Omega_{0N}} \overline{\boldsymbol{T}}\cdot\delta\varphi dS \\ + \int_{\Gamma_{0c}} \langle\boldsymbol{P}\rangle\cdot\boldsymbol{N}^1\cdot[\![\delta\varphi]\!]\,dS \\ + \int_{\Gamma_{0c}} [\![\varphi]\!]\cdot\langle\bar{\mathbb{C}}\colon Grad(\delta\varphi)\rangle\cdot\boldsymbol{N}^1\,dS \\ + \int_{\Gamma_{0c}} [\![\delta\varphi]\!]\otimes\boldsymbol{N}^1 \colon \langle\frac{\beta}{h_{\boldsymbol{s}}}\bar{\mathbb{C}}\,\rangle : [\![\delta\varphi]\!]\otimes\boldsymbol{N}^1\,dS \end{aligned} \tag{0-7}$$

where $[\![\varphi]\!]$ and $\langle\boldsymbol{P}\rangle$ denote the jump and average for displacement and first Piola-Kirchhoff stress tensor. $\bar{\mathbb{C}}$ indicates the first elasticity tensor with definition $\bar{\mathbb{C}} = \frac{\partial\boldsymbol{P}}{\partial\boldsymbol{F}}$.

The last line is the stabilization term inspired by the work of Noels et Radovitzky [278] for the discontinuous Galerkin method for FEM. In the latter, $h_{\boldsymbol{s}}$ is a characteristic length of the mesh and β is a stabilization parameter. The method is stable if β is larger than a

constant depending on the degree of the basis functions and the material parameters of the subdomains which share the common boundary.

We tested this formulation on a primary problem described in Figure 143. The entire domain is made up of homogeneous elastomer material (type Neo-Hookean), thus the results could be compared with the one patch solution.

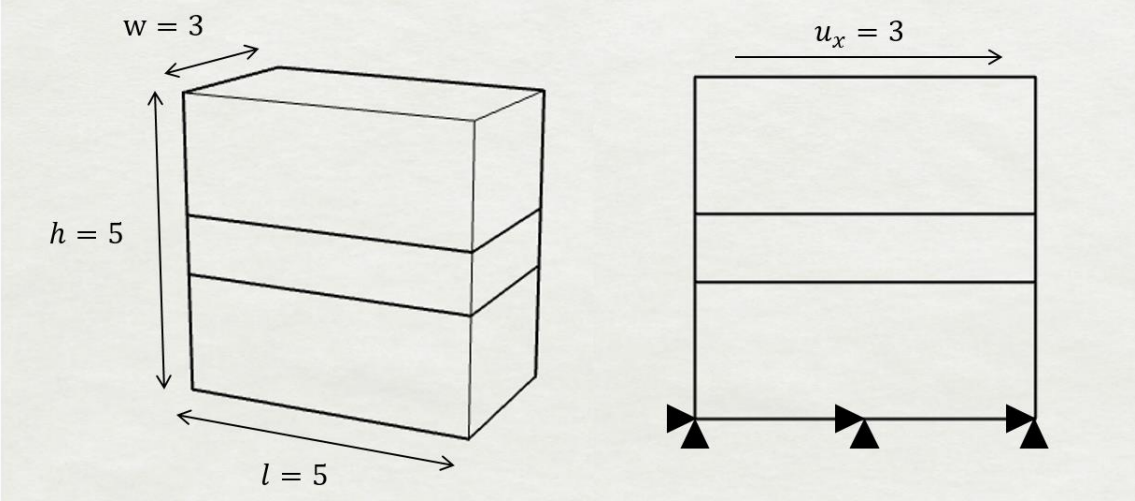


Figure 143. Block shearing: Problem definition.

All the numerical results in the following are resolved with cubic NURBS basis functions on a $5\times5\times3$ elements domain. The nonlinear system has been solved iteratively with 6 steps. The constitutive model for both elastomer and metal are given by:

$$\text{Neo-Hookean } \psi(\bar{I}_1, J) = \frac{1}{2}\mu(\bar{I}_1 - 3) + \frac{1}{2}\kappa(J^2 - 1) \qquad (0\text{-}8)$$

the parameters have been set as $\mu = 4.225\times10^5$ and $\kappa = 4.0\times10^5$.

In Figure 144, the solutions for displacement norm $\|\boldsymbol{u}\|$, the Cauchy stress component σ_{xy} and the determinant of deformation gradient $J = \det(\boldsymbol{F})$ of multi-patch geometry (left plots) and single patch geometry (right plots) are given. From a qualitative point of view, we observe no evident instability as that in the linear thermoelasticity case. Even the distributions of stress and deformation gradient distribution appear to be smooth at the inter-patch interfaces.

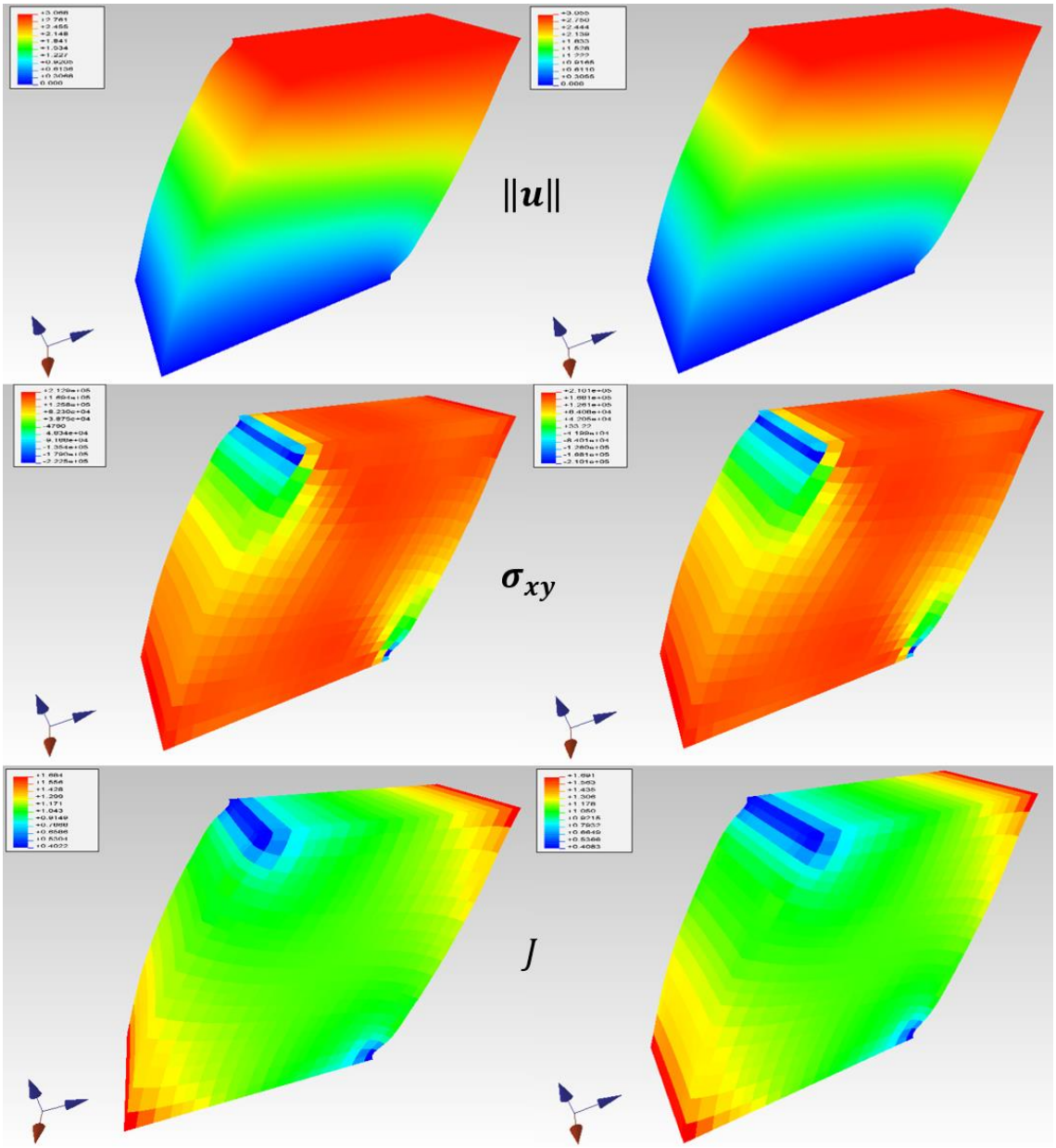


Figure 144. Multi-patch with homogenous material solutions (left) compared to single patch solutions (right).

Conclusion

The application of Nitsche's method in the context of NURBS based isogeometric analysis has been studied. The extensions of this patch coupling method to linear thermoelasticity and finite thermoelasticity are proposed and tested with weak gradient for solutions. It is observed that the proposed Nitsche's methods seems to be able to derive truthful primary solutions for the primary unknown fields (temperature and displacement) assuming that the

stabilization parameter is appropriately measured. But it demands a further study to eliminate the spurious oscillations which exits generally in these secondary fields such as stress and deformation gradients.

Appendix B. $\overline{\boldsymbol{B}}$ and $\overline{\boldsymbol{F}}$ projection method

Mesh-locking occurs in nearly incompressible model can be relevantly relieved by projection method. The essential idea is to treat separately and differently the volumetric and isochoric parts of deformation. Alleviating the incompressible constraint by projecting the volumetric deformation to a lower-order space for the purpose of reducing constraint numbers.

$\overline{\boldsymbol{B}}$ projection

The linear elastic problems lie at the infinitesimal regime, where the initial and current configuration are regarded as identic. Thus the distinction between spatial and material descriptions becomes unnecessary. We complete the problem's definition with the generalized Hooke's Law:

$$\boldsymbol{\sigma} = \mathbb{C}: \boldsymbol{\epsilon} \tag{0-1}$$

The Cauchy stress tensor denoted by $\boldsymbol{\sigma}$, and $\boldsymbol{\epsilon}$indicates the deformation tensor at small strain regime defined as the symmetric part of displacement $\boldsymbol{u}$ gradient:

$$\boldsymbol{\epsilon} = \nabla^s \boldsymbol{u} = \frac{1}{2}(\nabla \boldsymbol{u} + \nabla^T \boldsymbol{u}) \tag{0-2}$$

For isotropic material, the elasticity tensor of Hooke's law can be expressed in terms of the Lamé parameters λ andμ as:

$$\mathbb{C} = \lambda \boldsymbol{I} \otimes \boldsymbol{I} + 2\mu \mathbb{I} \tag{0-3}$$

where$\boldsymbol{I}$is the second-order identity tensor, and $\mathbb{I}$ is the symmetric part of the fourth-order identity tensor $\mathbb{I}_{ijkl} = (\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})/2$.In addition, the Lamé parameters are functions of Young's modulus E and Poisson ratio υ:

$$\mu = \frac{E}{2(1+\nu)}, \quad \lambda = \frac{2\mu\nu}{(1-2\upsilon)} \tag{0-4}$$

The incompressibility of linear elastic materials can be represented in the case where $\upsilon \rightarrow 0.5$. For the reason that the volumetric stress is originated only by λ, and $\lambda \rightarrow \infty$ when$\upsilon \rightarrow$

0.5. The so-called nearly incompressible or weakly compressible linear elastic models are characterized when υtakes a value close to 0.5 such as 0.49 or even 0.4999.

For the case of small strain, the projection method named $\bar{\boldsymbol{B}}$ demands to decompose the strain tensor $\boldsymbol{\epsilon}$into its deviatoric and dilatational parts.

$$\begin{gathered}\boldsymbol{\varepsilon}(\boldsymbol{u}) = \boldsymbol{\varepsilon}^{dil}(\boldsymbol{u}) + \boldsymbol{\varepsilon}^{dev}(\boldsymbol{u}) \\ \boldsymbol{\varepsilon}^{dil} = \frac{1}{3} tr(\boldsymbol{\varepsilon}) \boldsymbol{I}, \qquad \boldsymbol{\varepsilon}^{dev} = \boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^{dil}\end{gathered} \tag{0-5}$$

The incompressible constraint can be relieved by projecting the dilatational part of strain $\boldsymbol{\epsilon}^{dil}(\boldsymbol{u})$to a low-order space, and we obtain a modified strain tensor as:

$$\begin{gathered}\bar{\boldsymbol{\varepsilon}}(\boldsymbol{u}) = \overline{\boldsymbol{\varepsilon}^{dil}}(\boldsymbol{u}) + \boldsymbol{\varepsilon}^{dev}(\boldsymbol{u}) \\ \overline{\boldsymbol{\varepsilon}^{dil}}(\boldsymbol{u}) = \boldsymbol{\pi}(\boldsymbol{\varepsilon}^{dil}(\boldsymbol{u}))\end{gathered} \tag{0-6}$$

where $\boldsymbol{\pi}$ represents a linear projection operator.

The expression of strain-displacement matrix $\boldsymbol{B}$is derived therefrom with Voigt notation and discretization of displacement $\boldsymbol{u} \to \boldsymbol{u}^h = \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} N_{\boldsymbol{i}}(\boldsymbol{x})\, \boldsymbol{e_i}$. We perform the decomposition and projection on strain-displacement matrixto find its modified version $\bar{\boldsymbol{B}}$:

$$\bar{\boldsymbol{B}} = \overline{\boldsymbol{B}^{dil}} + \boldsymbol{B}^{dev} \tag{0-7}$$

Finally, the $\bar{\boldsymbol{B}}$ projection variational formulation for linear elasticity can be written as:

Find $\boldsymbol{u} \in V_u$ that for all $\delta\boldsymbol{u} \in V_u^0$

$$\begin{gathered}\bar{a}(\delta\boldsymbol{u}, \boldsymbol{u}) = (\delta\boldsymbol{u}, \boldsymbol{b}) + (\delta\boldsymbol{u}, \bar{\boldsymbol{t}})_{\partial\Omega_\sigma} \\ \text{with } \bar{a}(\delta\boldsymbol{u}, \boldsymbol{u}) = \int_\Omega \bar{\boldsymbol{\varepsilon}}(\delta\boldsymbol{u}) : \mathbb{C} : \bar{\boldsymbol{\varepsilon}}(\boldsymbol{u})\, d\Omega\end{gathered} \tag{0-8}$$

$\bar{\boldsymbol{F}}$ projection

For the strain measure at finite strain regime, the deformation gradient $\boldsymbol{F}$is chosen to be modified by the projection method. In order to achieve that, $\boldsymbol{F}$isas well split into isochoric part and volumetric part. Conversely, this decomposition is multiplicative rather than the addictive decomposition that has been performed for strain tensor of linear case.

$$\boldsymbol{F} = \boldsymbol{F}^{vol}\boldsymbol{F}^{iso} \tag{0-9}$$

The isochoric part $\boldsymbol{F}^{iso}$ preserves volume during deformation so that:

$$\begin{gathered} det\left(\boldsymbol{F}^{iso}\right) = 1 \\ J = det(\boldsymbol{F}) = det(\boldsymbol{F}^{vol}) \cdot det\left(\boldsymbol{F}^{iso}\right) \end{gathered} \tag{0-10}$$

We can thus derive that:

$$\begin{gathered} \boldsymbol{F}^{vol} = J^{\mathbf{1/3}}\boldsymbol{I}, \qquad \boldsymbol{F}^{iso} = J^{\mathbf{-1/3}}\boldsymbol{F} \\ det(\boldsymbol{F}^{vol}) = J, \qquad det\left(\boldsymbol{F}^{iso}\right) = 1 \end{gathered} \tag{0-11}$$

This decomposition has been studied previously by Hughes et al. [279] and Simo et Taylor [42]. Next, the modified deformation gradient tensor $\overline{\boldsymbol{F}}$ is constructed by multiply the isochoric part $\boldsymbol{F}^{iso}$ with a projected volumetric component $\overline{\boldsymbol{F}^{vol}}$.

$$\overline{\boldsymbol{F}} = \boldsymbol{F}^{iso}\overline{\boldsymbol{F}^{vol}} \tag{0-12}$$

Similar as for $\overline{\boldsymbol{B}}$ projection, the modified volumetric part of deformation gradient $\overline{\boldsymbol{F}^{vol}}$ stems from a linear projection operation $\boldsymbol{\pi}$ on $\boldsymbol{F}^{vol}$

$$\overline{\boldsymbol{F}^{vol}} = \boldsymbol{\pi}(\boldsymbol{F}^{vol}) = \boldsymbol{\pi}(J^{\mathbf{1/3}})\boldsymbol{I} \tag{0-13}$$

By recalling the variational principle in terms of second Piola-Kirchhoff stress tensor $\boldsymbol{S}$ introduced in equation (3-13), the modified potential energy can be formulated as:

$$\bar{W}(\boldsymbol{u}) = \int_{\Omega_0} \Psi(\overline{\boldsymbol{E}}(\boldsymbol{u}))\ dV - \int_{\Omega_0} \boldsymbol{B} \cdot \boldsymbol{u}\ dV - \int_{\partial\Omega_{0\sigma}} \overline{\boldsymbol{T}} \cdot \boldsymbol{u}\ dS \tag{0-14}$$

where the modified Green-Lagrange strain tensor $\overline{\boldsymbol{E}}(\boldsymbol{u})$ is defined in terms of the modified deformation gradient:

$$\overline{\boldsymbol{E}} = \frac{1}{2}(\overline{\boldsymbol{F}}^T\overline{\boldsymbol{F}} - \boldsymbol{I}) \tag{0-15}$$

The stationary point of modified potential energy (8-14) is derived by its directional derivation with respect to $\boldsymbol{\delta u}$. Moreover, since its stationarity equations are typically nonlinear, we are obliged to solve them through iterative algorithms such Newton-Raphson method. Consequently, the constituent linearization of the stationarity equations will be accomplished. The calculation details have been originally presented in the work of Elguedj et al [46].

Appendix C. Mixed formulation for linear incompressible and quasi-incompressible elasticity

For small deformations, to overcome the volumetric locking of incompressible linear elastic problem, the main problem lies in the determination of the mean stress which is related to the volumetric part of the strain for isotropic materials. Therefore, the mean part is habitually split from the stress tensor and regarded as an independent variable p, which has a physical meaning of hydrostatic pressure or parameter pressure.

$$p = \frac{1}{3} tr(\boldsymbol{\sigma}) \tag{0-1}$$

The hydrostatic pressure is proportional to dilatational part of deformation tensor $\boldsymbol{\epsilon}^{dil}$:

$$p = K\ tr(\boldsymbol{\varepsilon}) = \ K\ \mathrm{div}(\boldsymbol{u}) \tag{0-2}$$

The deviatoric part of stress $\boldsymbol{\sigma}^{dev}$ can be written by introducing the deviatoric form for the elastic moduli of an isotropic material as follows:

$$\boldsymbol{\sigma}^{dev} = \boldsymbol{D}^{dev}\boldsymbol{\varepsilon} \tag{**0**-3}$$

We can derive the variational mixed formulation for quasi-incompressible linear elasticity problem:

Find $\boldsymbol{u} \in V_u$ that for all $\delta\boldsymbol{u} \in V_u^0$, and $p \in V_p$ that for all $\delta p \in V_p^0$,

$$\begin{cases} \int_\Omega \delta\boldsymbol{\varepsilon}^T \boldsymbol{D}^{dev}\boldsymbol{\varepsilon} dV + \int_\Omega \mathbf{div}(\delta\boldsymbol{u})\ p\ dV - \int_\Omega \boldsymbol{b}\cdot\delta\boldsymbol{u}\ dV - \int_{\partial\Omega_\sigma} \bar{\boldsymbol{t}}\cdot\delta\boldsymbol{u}\ dS = 0 \\ \int_\Omega \delta p \left(\mathbf{div}(\boldsymbol{u})\ - \frac{p}{K}\right) dV \end{cases} \tag{0-4}$$

Discretizing the variational equations with $\boldsymbol{u} \cong \boldsymbol{u}^h = \boldsymbol{N_u}\tilde{\boldsymbol{u}}$ and $p \cong p^h = \boldsymbol{N_p}\tilde{\boldsymbol{p}}$, we obtain the matrix form of the problem:

$$\begin{bmatrix} \boldsymbol{K_{uu}} & \boldsymbol{K_{up}} \\ \boldsymbol{K_{pu}} & \boldsymbol{K_{pp}} \end{bmatrix} \begin{Bmatrix} \tilde{\boldsymbol{u}} \\ \tilde{\boldsymbol{p}} \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{f_u} \\ \boldsymbol{f_p} \end{Bmatrix} \tag{0-5}$$

Where:

$$\boldsymbol{K}_{\boldsymbol{uu}}^{\boldsymbol{e}} = \int_{\Omega} \boldsymbol{B}^{T} \boldsymbol{D}^{dev} \boldsymbol{B} dV$$

$$\boldsymbol{K}_{\boldsymbol{up}}^{\boldsymbol{e}} = \boldsymbol{K}_{\boldsymbol{pu}}^{\boldsymbol{e}\,\boldsymbol{T}} = \int_{\Omega} \boldsymbol{G}^{T} \boldsymbol{N}_{\boldsymbol{p}} dV \tag{0-6}$$

$$\boldsymbol{K}_{\boldsymbol{pp}}^{\boldsymbol{e}} = -\frac{1}{K} \int_{\Omega} \boldsymbol{N}_{\boldsymbol{p}}{}^{T} \boldsymbol{N}_{\boldsymbol{p}} dV$$

with

$$\begin{aligned} \boldsymbol{K}_{\boldsymbol{uu}} &= \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{K}_{\boldsymbol{uu}}^{\boldsymbol{e}} \quad \boldsymbol{K}_{\boldsymbol{up}} = \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{K}_{\boldsymbol{up}}^{\boldsymbol{e}} \\ \boldsymbol{K}_{\boldsymbol{pu}} &= \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{K}_{\boldsymbol{pu}}^{\boldsymbol{e}} \quad \boldsymbol{K}_{\boldsymbol{pp}} = \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{K}_{\boldsymbol{pp}}^{\boldsymbol{e}} \quad \boldsymbol{f}_{\boldsymbol{u}} = \mathbf{A}_{\boldsymbol{e}}\, \boldsymbol{f}_{\boldsymbol{u}}^{\boldsymbol{e}} \end{aligned} \tag{0-7}$$

and with classical definitions of matrices $\boldsymbol{B}$, $\boldsymbol{G}$ and $\boldsymbol{N}_{\boldsymbol{p}}$.

For incompressible materials, the equations degenerate to $\boldsymbol{K}_{\boldsymbol{pp}} = 0$ when $K \to \infty$. The formulation is useful in practice for nearly incompressible i.e. $\nu \to 0.5$.

Considering the discretized form of equation (0-5) and removing the term $\boldsymbol{K}_{\boldsymbol{pp}}$ for a perfectly incompressible material model in which $\tilde{\boldsymbol{u}}$ is the primary variable, and $\tilde{\boldsymbol{p}}$ is the constraint variable. The pressure parameter $\tilde{\boldsymbol{p}}$ can be computed by eliminating $\tilde{\boldsymbol{u}}$ from the first equation and substituting it into the second to obtain:

$$\left(\boldsymbol{K}_{\boldsymbol{pu}} \boldsymbol{K}_{\boldsymbol{uu}}{}^{-1} \boldsymbol{K}_{\boldsymbol{up}}\right) \tilde{\boldsymbol{p}} = -\boldsymbol{f}_{\boldsymbol{p}} + \boldsymbol{K}_{\boldsymbol{pu}} \boldsymbol{K}_{\boldsymbol{uu}}{}^{-1} \boldsymbol{f}_{\boldsymbol{u}} \tag{0-8}$$

To calculate $\tilde{\boldsymbol{p}}$ it is necessary to ensure that the matrix $\left(\boldsymbol{K}_{\boldsymbol{pu}} \boldsymbol{K}_{\boldsymbol{uu}}{}^{-1} \boldsymbol{K}_{\boldsymbol{up}}\right)$ is non-singular. Thus, the number of unknowns n_u for $\tilde{\boldsymbol{u}}$ has to be equal or greater to n_p which is the number of unknowns for $\tilde{\boldsymbol{p}}$:

$$n_u \geq n_p \tag{0-9}$$

The reason for this is evident as the matrix $\left(\boldsymbol{K}_{\boldsymbol{pu}} \boldsymbol{K}_{\boldsymbol{uu}}{}^{-1} \boldsymbol{K}_{\boldsymbol{up}}\right)$ needs to be full rank for its reversibility.

Bibliography

[1] S. Cabay, “Spline Analysis (Martin H. Schultz),” *SIAM Rev.*, vol. 16, no. 1, pp. 105–106, Jan. 1974.

[2] K. Höllig, “Finite Element Methods with B-Splines,” p. 145, 2003.

[3] D. Natekar, X. Zhang, and G. Subbarayan, “Constructive solid analysis: A hierarchical, geometry-based meshless analysis procedure for integrated design and analysis,” *CAD Comput. Aided Des.*, vol. 36, no. 5, pp. 473–486, 2004.

[4] M. S. Casale and J. E. Bobrow, “The analysis of solids without mesh generation using trimmed patch boundary elements,” *Eng. with Comput. An Int. J. Simulation-Based Eng.*, vol. 5, no. 3, pp. 249–257, 1989.

[5] G. Celnikera and D. Gossard, “Deformable Curve and Surface Finite-Elements for Free-Form Shape Design,” *Siggraph*, vol. 25, no. 4, pp. 257–366, 1991.

[6] D. Terzopoulos and H. Qin, “Dynamic NURBS with geometric constraints for interactive sculpting,” *ACM Trans. Graph.*, vol. 13, no. 2, pp. 103–136, 1994.

[7] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs, “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement,” *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 39–41. pp. 4135–4195, 2005.

[8] D. R. Rehak and J. W. Baugh, “Alternative Programming Techniques for Finite Element Programming Development,” *Proc. IABSE Colloq. Expert Syst. Civ. Eng.*, 1989.

[9] G. R. Miller, “A LISP-based object-oriented approach to structural analysis,” *Eng. Comput.*, vol. 4, no. 4, pp. 197–203, Dec. 1988.

[10] T. Zimmermann, Y. Dubois-Pèlerin, and P. Bomme, “Object-oriented finite element programming: I. Governing principles,” *Comput. Methods Appl. Mech. Eng.*, vol. 98, no. 2, pp. 291–303, Jul. 1992.

[11] J. Besson and R. Foerch, “Large scale object-oriented finite element code design,” *Comput. Methods Appl. Mech. Eng.*, vol. 142, no. 1–2, pp. 165–187, Mar. 1997.

[12] D. Eyheramendy, “Object-Oriented parallel CFD with JAVA,” in *Parallel Computational Fluid Dynamics 2003*, Elsevier, 2004, pp. 409–416.

[13] D. Eyheramendy, S. Lejeunes, R. Saad, and L. Zhang, “Advances in Symbolic and Numerical Approaches in Computational Mechanics,” in *Computational Methods for Engineering Technology*, Stirlingshire, UK: Saxe-Coburg Publications, 2014, pp. 61–88.

[14] R. Saad, D. Eyheramendy, and L. Zhang, “An object-oriented symbolic approach to the automated derivation of finite element contributions,” *Adv. Eng. Softw.*, vol. 94, pp. 1–13, 2016.

[15] J. Korelc, “Automatic generation of finite-element code by simultaneous optimization of expressions,” *Theor. Comput. Sci.*, vol. 187, no. 1–2, pp. 231–248, 1997.

[16] A. Logg, “Automating the finite element method,” *Arch. Comput. Methods Eng.*, vol. 14, no. 2, pp. 93–138, 2007.

[17] A. V. Vuong, C. Heinrich, and B. Simeon, “ISOGAT: A 2D tutorial MATLAB code for Isogeometric Analysis,” in *Computer Aided Geometric Design*, 2010, vol. 27, no. 8, pp. 644–655.

[18] V. P. Nguyen, C. Anitescu, S. P. A. Bordas, and T. Rabczuk, “Isogeometric analysis: An overview and computer implementation aspects,” *Math. Comput. Simul.*, vol. 117, pp. 89–116, 2015.

[19] C. De Falco, A. Reali, and R. Vázquez, “GeoPDEs: A research tool for Isogeometric Analysis of PDEs,” *Adv. Eng. Softw.*, vol. 42, no. 12, pp. 1020–1034, 2011.

[20] L. Dalcin, N. Collier, P. Vignal, A. M. A. Côrtes, and V. M. Calo, “PetIGA: A framework for high-performance isogeometric analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 308, pp. 151–181, 2016.

[21] A. Duval, F. Maurin, and T. Elguedj, “Abaqus user element implementation of NURBS based isogeometric analysis,” in *ECCOMAS*, 2012.

[22] K. A. A. M. Kvarving, K. Kvamsdal, K.M. Okstad, K.M. Mathisen, E. Fonn and A. R. and R. H. Johannessen, “IFEM – an isogeometric toolbox for the solution of PDEs,” in *III International Conference on Isogeometric Analysis*, 2015.

[23] D. Rypl and B. Patzák, “Object oriented implementation of the T-spline based isogeometric analysis,” *Adv. Eng. Softw.*, vol. 50, no. 1, pp. 137–149, 2012.

[24] D. J. Naylor, “Stresses in Nearly Incompressible Materials by Finite Elements with Application to the Calculation of Excess Pore Pressures,” *Int. J. Numer. Methods Eng.*, vol. 8, no. 3, pp. 443–460, 1974.

[25] D. S. Malkus and T. J. R. Hughes, “Mixed finite element methods — Reduced and selective integration techniques: A unification of concepts,” *Comput. Methods Appl. Mech. Eng.*, vol. 15, no. 1, pp. 63–81, Jul. 1978.

[26] T. J. R. Hughes, L. P. Franca, and M. Balestra, “A new finite element formulation for computational fluid dynamics: V. Circumventing the babu??ka-brezzi condition: a stable Petrov-Galerkin formulation of the stokes problem accommodating equal-order interpolations,” *Comput. Methods Appl. Mech. Eng.*, vol. 59, no. 1, pp. 85–99, 1986.

[27] T. J. R. Hughes and L. P. Franca, “A new finite element formulation for computational fluid dynamics: VII. The stokes problem with various well-posed boundary conditions: Symmetric formulations that converge for all velocity/pressure spaces,” *Comput. Methods Appl. Mech. Eng.*, vol. 65, no. 1, pp. 85–96, 1987.

[28] T. J. R. Hughes, L. P. Franca, and G. M. Hulbert, “A new finite element formulation for computational fluid dynamics: VIII. The galerkin/least-squares method for advective-diffusive equations,” *Comput. Methods Appl. Mech. Eng.*, vol. 73, no. 2, pp. 173–189, 1989.

[29] E. Oñate, “Derivation of stabilized equations for numerical solution of advective-diffusive transport and fluid flow problems,” *Comput. Methods Appl. Mech. Eng.*, vol. 151, no. 1–2, pp. 233–265, 1998.

[30] F. Brezzi, L. P. Franca, and A. Russo, “Further considerations on residual-free bubbles for advective-diffusive equations,” *Comput. Methods Appl. Mech. Eng.*, vol. 166, no. 1–2, pp. 25–33, 1998.

[31] F. Brezzi, L. P. Franca, T. J.R.Hughes, and A. Russo, “B Identical With Integral of G,” *Comput. Methods Appl. Mech. Eng.*, vol. 145, no. 3, pp. 329–339, 1997.

[32] F. Brezzi, P. Houston, and D. Marini, “Modeling subgrid viscosity for advection ± di usion problems,” vol. 190, 2000.

[33] F. Brezzi, M. O. Bristeau, L. P. Franca, M. Mallet, and G. Rogé, “A relationship between stabilized finite element methods and the Galerkin method with bubble functions,” *Comput. Methods Appl. Mech. Eng.*, vol. 96, no. 1, pp. 117–129, 1992.

[34] T. J. R. Hughes, “Multiscale phenomena: Green’s functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods,” *Comput. Methods Appl. Mech. Eng.*, vol. 127, no. 1–4, pp. 387–401, 1995.

[35] T. J. R. Hughes, G. R. Feijóo, L. Mazzei, and J.-B. Quincy, “The variational multiscale method—a paradigm for computational mechanics,” *Comput. Methods Appl. Mech. Eng.*, vol. 166, no. 1–2, pp. 3–24, 1998.

[36] A. Masud and R. A. Khurram, “A multiscale finite element method for the incompressible Navier-Stokes equations,” *Comput. Methods Appl. Mech. Eng.*, vol. 195, no. 13–16, pp. 1750–1777, 2006.

[37] K. B. Nakshatrala, A. Masud, and K. D. Hjelmstad, “On finite element formulations for nearly incompressible linear elasticity,” *Comput. Mech.*, vol. 41, no. 4, pp. 547–561, 2008.

[38] M. Cervera, N. Lafontaine, R. Rossi, and M. Chiumenti, “Explicit mixed strain–displacement finite elements for compressible and quasi-incompressible elasticity and plasticity,” *Comput. Mech.*, vol. 58, no. 3, pp. 511–532, Sep. 2016.

[39] L. R. Herrmann and R. M. Toms, “A Reformulation of the Elastic Field Equation, in Terms of Displacements, Valid for all Admissible Values of Poisson’s Ratio,” *J. Appl. Mech.*, vol. 31, no. 1, pp. 140–141, Mar. 1964.

[40] R. L. Taylor, K. S. Pister, and L. R. Herrmann, “On a variational theorem for incompressible and nearly-incompressible orthotropic elasticity,” *Int. J. Solids Struct.*, vol. 4, no. 9, pp. 875–883, Sep. 1968.

[41] J. H. Argyris, P. C. Dunne, T. Angelopoulos, and B. Bichat, “Large natural strains and some special difficulties due to non-linearity and incompressibility in finite elements,” *Comput. Methods Appl. Mech. Eng.*, vol. 4, no. 2, pp. 219–278, Sep. 1974.

[42] J. C. Simo, R. L. Taylor, and K. S. Pister, “Variational and projection methods for the volume constraint in finite deformation elasto-plasticity,” *Comput. Methods Appl. Mech. Eng.*, vol. 51, no. 1–3, pp. 177–208, 1985.

[43] E. Jankovich, F. Leblanc, M. Durand, and M. Bercovier, “A finite element method for the analysis of rubber parts, experimental and analytical assessment,” *Comput. Struct.*, vol. 14, no. 5–6, pp. 385–391, Jan. 1981.

[44] J. T. Oden and N. Kikuchi, “Finite element methods for constrained problems in elasticity,” *Int. J. Numer. Methods Eng.*, vol. 18, no. May 1980, pp. 701–725, 1982.

[45] J. C. Simo and M. S. Rifai, “A class of mixed assumed strain methods and the method of incompatible modes,” *Int. J. Numer. Methods Eng.*, vol. 29, no. 8, pp. 1595–1638, Jun. 1990.

[46] T. Elguedj, Y. Bazilevs, V. M. Calo, and T. J. R. Hughes, “B and F projection methods for nearly incompressible linear and non-linear elasticity and plasticity using higher-order NURBS elements,” *Comput. Methods Appl. Mech. Eng.*, vol. 197, no. 33–40, pp. 2732–2762, Jun. 2008.

[47] C. Adam, T. J. R. Hughes, S. Bouabdallah, M. Zarroug, and H. Maitournam, “Selective and reduced numerical integrations for NURBS-based isogeometric analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 284, pp. 732–761, Feb. 2015.

[48] A. Buffa, C. de Falco, and G. Sangalli, “IsoGeometric Analysis: Stable elements for the 2D Stokes equation,” *Int. J. Numer. Methods Fluids*, vol. 65, no. 11–12, pp. 1407–1422, Apr. 2011.

[49] P. N. Nielsen, A. R. Gersborg, J. Gravesen, and N. L. Pedersen, “Discretizations in isogeometric analysis of Navier–Stokes flow,” *Comput. Methods Appl. Mech. Eng.*, vol. 200, no. 45–46, pp. 3242–3253, Oct. 2011.

[50] R. L. Taylor, “Isogeometric analysis of nearly incompressible solids,” *Int. J. Numer. Methods Eng.*, vol. 87, no. 1–5, pp. 273–288, Jul. 2011.

[51] R. P. R. Cardoso and J. M. A. Cesar de Sa, “The enhanced assumed strain method for the isogeometric analysis of nearly incompressible deformation of solids,” *Int. J. Numer. Methods Eng.*, vol. 92, no. 1, pp. 56–78, Oct. 2012.

[52] W.-M. Xue and S. N. Atluri, “Existence and stability, and discrete BB and rank conditions, for general mixed-hybrid finite elements in elasticity,” 1985.

[53] G. Prathap and B. P. Naganarayana, “Consistent thermal stress evaluation in finite elements,” *Comput. Struct.*, vol. 54, no. 3, pp. 415–426, 1995.

[54] G. A. Keramidas and E. C. Ting, “A finite element formulation for thermal stress analysis. Part II: Finite element formulation,” *Nucl. Eng. Des.*, vol. 39, no. 2–3, pp. 277–287, 1976.

[55] J. P. Carter and J. R. Booker, “Finite element analysis of coupled thermoelasticity,” *Comput. Struct.*, vol. 31, no. 1, pp. 73–80, 1989.

[56] D. M. Rao and P. K. Sinha, “Finite element coupled thermostructural analysis of composite beams,” *Comput. Struct.*, vol. 63, no. 3, pp. 539–549, May 1997.

[57] D. Juhre and S. Reese, “A reduced integration finite element technology based on a thermomechanically consistent stabilisation for 3D problems,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 29, pp. 2050–2058, 2010.

[58] S. de Miranda and F. Ubertini, “On the consistency of finite element models in thermoelastic analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 190, no. 18–19, pp. 2411–2427, Jan. 2001.

[59] Y. Y. Zhu and S. Cescotto, “Unified and mixed formulation of the 8-node hexahedral elements by assumed strain method,” *Comput. Methods Appl. Mech. Eng.*, vol. 129, no. 95, pp. 177–209, 1996.

[60] A. A. Cannarozzi and F. Ubertini, “A mixed variational method for linear coupled thermoelastic analysis,” *Int. J. Solids Struct.*, vol. 38, no. 4, pp. 717–739, Jan. 2001.

[61] M. Dittmann, M. Franke, I. Temizer, and C. Hesch, “Isogeometric Analysis and thermomechanical Mortar contact problems,” *Comput. Methods Appl. Mech. Eng.*, vol. 274, pp. 192–212, 2014.

[62] C. Kadapa, W. G. Dettmer, and D. Perić, “Subdivision based mixed methods for isogeometric analysis of linear and nonlinear nearly incompressible materials,” *Comput. Methods Appl. Mech. Eng.*, vol. 305, pp. 241–270, Jun. 2016.

[63] L. Piegl and W. Tiller, “The NURBS Book (Monographs in Visual Communication),” *Springer-Verleg*, 1996.

[64] T. J. R. Hughes, *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, vol. 682. 2000.

[65] V. P. Nguyen and S. Bordas, “Isogeometric Methods for Numerical Simulation,” G. Beer and S. Bordas, Eds. Vienna: Springer Vienna, 2015, pp. 21–120.

[66] W. Y. Kwok, R. D. Moser, and J. Jiménez, “A Critical Evaluation of the Resolution Properties of {B-Spline} and Compact Finite Difference Methods,” *J. Comput. Phys.*, vol. 174, pp. 510–551, 2001.

[67] F. P. Renken and G. Subbarayan, “NURBS-based solutions to inverse boundary problems in droplet shape prediction,” *Comput. Methods Appl. Mech. Eng.*, vol. 190, no. 11–12, pp. 1391–1406, 2000.

[68] F. Cirak, M. Ortiz, and P. Schroder, “Subdivision surfaces: a new paradigm for thin-shell {finite -element analysis,” *Int. J. Numer. Meth. Engng*, vol. 47, no. July 1999, pp. 2039–2072, 2000.

[69] P. Kagan, A. Fischer, and P. Z. Bar-Yoseph, “New B-Spline Finite Element Approach for Geometric Design and Mechanical Analysis,” *Int. J. Numer. Methods Eng.*, vol. 41, no. March 1997, pp. 435–458, 1998.

[70] P. Kagan and a. Fischer, “Integrated mechanically based CAE system using B-Spline finite elements,” *Comput. Des.*, vol. 32, pp. 539–552, 2000.

[71] D. F. Rogers, *An introduction to NURBS: with historical perspective*. 2001.

[72] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, *Isogeometric Analysis: Toward Integration of CAD and FEA*. 2009.

[73] J. A. Cottrell, T. J. R. Hughes, and A. Reali, “Studies of refinement and continuity in isogeometric structural analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 196, no. 41–44, pp. 4160–4183, 2007.

[74] J. Kiendl, K. U. Bletzinger, J. Linhard, and R. Wüchner, “Isogeometric shell analysis with Kirchhoff-Love elements,” *Comput. Methods Appl. Mech. Eng.*, vol. 198, no. 49–52, pp. 3902–3914, 2009.

[75] J. Kiendl, Y. Bazilevs, M. C. Hsu, R. Wüchner, K. U. Bletzinger, R. Wuchner, and K. U. Bletzinger, “The bending strip method for isogeometric analysis of Kirchhoff-Love shell structures comprised of multiple patches,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 37–40, pp. 2403–2416, 2010.

[76] D. J. Benson, Y. Bazilevs, M. C. Hsu, and T. J. R. Hughes, “Isogeometric shell analysis: The Reissner-Mindlin shell,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 5–8, pp. 276–289, 2010.

[77] D. J. Benson, Y. Bazilevs, M. C. Hsu, and T. J. Hughes, “A large deformation, rotation-free, isogeometric shell,” *Comput. Methods Appl. Mech. Eng.*, vol. 200, no. 13–16, pp. 1367–1378, 2011.

[78] D. J. Benson, S. Hartmann, Y. Bazilevs, M. C. Hsu, and T. J. R. Hughes, “Blended isogeometric shells,” *Comput. Methods Appl. Mech. Eng.*, vol. 255, pp. 133–146, 2013.

[79] W. Dornisch, S. Klinkel, and B. Simeon, “Isogeometric Reissner–Mindlin shell analysis with exactly calculated director vectors,” *Comput. Methods Appl. Mech. Eng.*, vol. 253, pp. 491–504, Jan. 2013.

[80] R. Echter, B. Oesterle, and M. Bischoff, “A hierarchic family of isogeometric shell finite elements,” *Comput. Methods Appl. Mech. Eng.*, vol. 254, pp. 170–180, Feb. 2013.

[81] T. J. R. Hughes, “Generalization of selective integration procedures to anisotropic and nonlinear media,” *Int. J. Numer. Methods Eng.*, vol. 15, pp. 1413–1418, 1980.

[82] R. Echter and M. Bischoff, “Numerical efficiency, locking and unlocking of NURBS finite elements,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 5–8, pp. 374–382, 2010.

[83] R. Bouclier, T. Elguedj, and A. Combescure, “Locking free isogeometric formulations of curved thick beams,” *Comput. Methods Appl. Mech. Eng.*, vol. 245–246, pp. 144–162, Oct. 2012.

[84] R. Bouclier, T. Elguedj, and A. Combescure, “An isogeometric locking-free NURBS-based solid-shell element for geometrically nonlinear analysis,” *Int. J. Numer. Methods Eng.*, vol. 101, no. 10, pp. 774–808, 2015.

[85] L. Beirão da Veiga, C. Lovadina, and A. Reali, “Avoiding shear locking for the Timoshenko beam problem via isogeometric collocation methods,” *Comput. Methods Appl. Mech. Eng.*, vol. 241–244, pp. 38–51, 2012.

[86] F. Auricchio, L. Beir??o da Veiga, J. Kiendl, C. Lovadina, and A. Reali, “Locking-free isogeometric collocation methods for spatial Timoshenko rods,” *Comput. Methods Appl. Mech. Eng.*, vol. 263, pp. 113–126, 2013.

[87] J. F. Caseiro, R. A. F. Valente, A. Reali, J. Kiendl, F. Auricchio, and R. J. Alves de Sousa, “On the Assumed Natural Strain method to alleviate locking in solid-shell NURBS-based finite elements,” *Comput. Mech.*, vol. 53, no. 6, pp. 1341–1353, Jun. 2014.

[88] T. Elguedj and T. J. R. Hughes, “Isogeometric analysis of nearly incompressible large strain plasticity,” *Comput. Methods Appl. Mech. Eng.*, vol. 268, pp. 388–416, Jan. 2014.

[89] J. F. Caseiro, R. J. Alves de Sousa, and R. A. F. Valente, “A systematic development of EAS three-dimensional finite elements for the alleviation of locking phenomena,” *Finite Elem. Anal. Des.*, vol. 73, pp. 30–41, 2013.

[90] R. P. R. Cardoso and J. M. A. Cesar de Sa, “Blending moving least squares techniques with NURBS basis functions for nonlinear isogeometric analysis,” *Comput. Mech.*, vol. 53, no. 6, pp. 1327–1340, Jun. 2014.

[91] C. Kadapa, W. G. Dettmer, and D. Perić, “NURBS based least-squares finite element methods for fluid and solid mechanics,” *Int. J. Numer. Methods Eng.*, vol. 101, no. 7, pp. 521–539, Feb. 2015.

[92] Y. Bazilevs, V. M. Calo, J. A. Cottrell, T. J. R. Hughes, A. Reali, and G. Scovazzi, “Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows,” *Comput. Methods Appl. Mech. Eng.*, vol. 197, no. 1–4, pp. 173–201, Dec. 2007.

[93] Y. Bazilevs and I. Akkerman, “Large eddy simulation of turbulent Taylor-Couette flow using isogeometric analysis and the residual-based variational multiscale method,” *J. Comput. Phys.*, vol. 229, no. 9, pp. 3402–3414, 2010.

[94] Y. Bazilevs and T. J. R. Hughes, “Weak imposition of Dirichlet boundary conditions in fluid mechanics,” *Comput. Fluids*, vol. 36, no. 1, pp. 12–26, Jan. 2007.

[95] Y. Bazilevs, C. Michler, V. M. Calo, and T. J. R. Hughes, “Weak Dirichlet boundary conditions for wall-bounded turbulent flows,” *Comput. Methods Appl. Mech. Eng.*, vol. 196, no. 49–52, pp. 4853–4862, 2007.

[96] K. Chang, T. J. R. Hughes, and V. M. Calo, “Isogeometric variational multiscale large-eddy simulation of fully-developed turbulent flow over a wavy wall,” *Comput. Fluids*, vol. 68, no. 4798, pp. 94–104, Sep. 2012.

[97] O. Colomés, S. Badia, R. Codina, and J. Principe, “Assessment of variational multiscale models for the large eddy simulation of turbulent incompressible flows,” *Comput. Methods Appl. Mech. Eng.*, vol. 285, pp. 32–63, 2015.

[98] R. Golshan, A. E. Tejada-Martínez, M. Juha, and Y. Bazilevs, “Large-eddy simulation with near-wall modeling using weakly enforced no-slip boundary conditions,” *Comput. Fluids*, vol. 118, pp. 172–181, Sep. 2015.

[99] G. Bauer, P. Gamnitzer, V. Gravemeier, and W. A. Wall, “An isogeometric variational multiscale method for large-eddy simulation of coupled multi-ion transport in turbulent flow,” *J. Comput. Phys.*, vol. 251, pp. 194–208, Oct. 2013.

[100] T. Rüberg and F. Cirak, “Subdivision-stabilised immersed b-spline finite elements for moving boundary flows,” *Comput. Methods Appl. Mech. Eng.*, vol. 209–212, pp. 266–283, Feb. 2012.

[101] B. S. Hosseini, M. Möller, and S. Turek, “Isogeometric Analysis of the Navier-Stokes equations with Taylor-Hood B-spline elements,” *Appl. Math. Comput.*, vol. 267, pp. 264–281, Sep. 2015.

[102] A. H. Niemi, N. O. Collier, and V. M. Calo, “Discontinuous Petrov-Galerkin method based on the optimal test space norm for one-dimensional transport problems,” *Procedia Comput. Sci.*, vol. 4, pp. 1862–1869, 2011.

[103] C. Manni, F. Pelosi, and M. L. Sampoli, “Isogeometric analysis in advection-diffusion problems: Tension splines approximation,” *J. Comput. Appl. Math.*, vol. 236, no. 4, pp. 511–528, 2011.

[104] S. Dhawan, S. Kapoor, and S. Kumar, “Numerical method for advection diffusion equation using FEM and B-splines,” *J. Comput. Sci.*, vol. 3, no. 5, pp. 429–437, 2012.

[105] H. Gomez, T. J. R. Hughes, X. Nogueira, and V. M. Calo, “Isogeometric analysis of the isothermal Navier-Stokes-Korteweg equations,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 25–28, pp. 1828–1840, 2010.

[106] H. Casquero, C. Bona-Casas, and H. Gomez, “A NURBS-based immersed methodology for fluid-structure interaction,” *Comput. Methods Appl. Mech. Eng.*, vol. 284, pp. 943–970, 2015.

[107] J. Bueno and H. Gomez, “Liquid-vapor transformations with surfactants. Phase-field model and Isogeometric Analysis,” *J. Comput. Phys.*, vol. 321, pp. 797–818, Sep. 2016.

[108] P. Vignal, A. Sarmiento, A. M. A. Côrtes, L. Dalcin, V. M. Calo, and K. Abdullah, “Coupling Navier-Stokes and Cahn-Hilliard equations in a two-dimensional annular flow configuration,” *Procedia - Procedia Comput. Sci.*, vol. 51, pp. 934–943, 2015.

[109] D. Schillinger, L. Ded??, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, and T. J. R. Hughes, “An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces,” *Comput. Methods Appl. Mech. Eng.*, vol. 249–252, pp. 116–150, Dec. 2012.

[110] M. C. Hsu, C. Wang, F. Xu, A. J. Herrema, and A. Krishnamurthy, “Direct immersogeometric fluid flow analysis using B-rep CAD models,” *Comput. Aided Geom. Des.*, vol. 43, pp. 143–158, Mar. 2016.

[111] C. Heinrich, B. Simeon, and S. Boschert, “A finite volume method on NURBS geometries and its application in isogeometric fluid-structure interaction,” *Math. Comput. Simul.*, vol. 82, no. 9, pp. 1645–1666, May 2012.

[112] I. Akkerman, Y. Bazilevs, C. E. Kees, and M. W. Farthing, “Isogeometric analysis of free-surface flow,” *J. Comput. Phys.*, vol. 230, no. 11, pp. 4137–4152, 2011.

[113] C. E. Kees, I. Akkerman, M. W. Farthing, and Y. Bazilevs, “A conservative level set method suitable for variable-order approximations and unstructured meshes,” *J. Comput. Phys.*, vol. 230, no. 12, pp. 4536–4558, 2011.

[114] Y. Bazilevs, V. M. Calo, Y. Zhang, and T. J. R. Hughes, “Isogeometric fluid-structure interaction analysis with applications to arterial blood flow,” *Comput. Mech.*, vol. 38, no. 4–5, pp. 310–322, 2006.

[115] Y. Bazilevs, V. M. Calo, T. J. R. Hughes, and Y. Zhang, “Isogeometric fluid-structure interaction: theory, algorithms, and computations,” *Comput. Mech.*, vol. 43, no. 1, pp. 3–37, 2008.

[116] Y. Bazilevs, J. R. Gohean, T. J. R. Hughes, R. D. Moser, and Y. Zhang, “Patient-specific isogeometric fluid-structure interaction analysis of thoracic aortic blood flow due to implantation of the Jarvik 2000 left ventricular assist device,” *Comput. Methods Appl. Mech. Eng.*, vol. 198, no. 45–46, pp. 3534–3550, 2009.

[117] Y. Bazilevs and T. J. R. Hughes, “NURBS-based isogeometric analysis for the computation of flows about rotating components,” *Comput. Mech.*, vol. 43, no. 1, pp. 143–150, 2008.

[118] Y. Bazilevs, M. Hsu, and M. A. Scott, “Isogeometric Fluid – Structure Interaction Analysis with Emphasis on Non-Matching Discretizations , and with Application to Wind Turbines,” *Comput. Methods Appl. Mech. Eng.*, vol. 252, pp. 28–41, 2012.

[119] İ. Temizer, P. Wriggers, and T. J. R. Hughes, “Contact treatment in isogeometric analysis with NURBS,” *Comput. Methods Appl. Mech. Eng.*, vol. 200, no. 9–12, pp. 1100–1112, Feb. 2011.

[120] J. Lu, “Isogeometric contact analysis: Geometric basis and formulation for frictionless contact,” *Comput. Methods Appl. Mech. Eng.*, vol. 200, no. 5–8, pp. 726–741, 2011.

[121] R. E. Jones and P. Papadopoulos, “A novel three-dimensional contact finite element based on smooth pressure interpolations,” *Int. J. Numer. Methods Eng.*, vol. 51, no. 7, pp. 791–811, 2001.

[122] L. De Lorenzis, İ. Temizer, P. Wriggers, and G. Zavarise, “A large deformation frictional contact formulation using NURBS-based isogeometric analysis,” *Int. J. Numer. Methods Eng.*, vol. 87, no. 13, p. n/a-n/a, 2011.

[123] İ. Temizer, P. Wriggers, and T. J. R. Hughes, “Three-dimensional mortar-based frictional contact treatment in isogeometric analysis with NURBS,” *Comput. Methods Appl. Mech. Eng.*, vol. 209–212, pp. 115–128, Feb. 2012.

[124] M. Dittmann, M. Franke, İ. Temizer, and C. Hesch, “Isogeometric Analysis and thermomechanical Mortar contact problems,” *Comput. Methods Appl. Mech. Eng.*, vol. 274, no. April 2016, pp. 192–212, Jun. 2014.

[125] J. A. Cottrell, A. Reali, Y. Bazilevs, and T. J. R. Hughes, “Isogeometric analysis of structural vibrations,” *Comput. Methods Appl. Mech. Eng.*, vol. 195, no. 41–43, pp. 5257–5296, 2006.

[126] T. J. R. Hughes, A. Reali, and G. Sangalli, “Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: Comparison of p-method finite elements with k-method NURBS,” *Comput. Methods Appl. Mech. Eng.*, vol. 197, no. 49, pp. 4104–4124, 2008.

[127] D. Wang, W. Liu, and H. Zhang, “Novel higher order mass matrices for isogeometric structural vibration analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 260, pp. 92–108, 2013.

[128] W. A. Wall, M. A. Frenzel, and C. Cyron, “Isogeometric structural shape optimization,” *Comput. Methods Appl. Mech. Eng.*, vol. 197, no. 33, pp. 2976–2988, 2008.

[129] X. Qian, “Full analytical sensitivities in NURBS based isogeometric shape optimization,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 29, pp. 2059–2071, 2010.

[130] N. D. Manh, A. Evgrafov, A. R. Gersborg, and J. Gravesen, “Isogeometric shape optimization of vibrating membranes,” *Comput. Methods Appl. Mech. Eng.*, vol. 200, no. 13, pp. 1343–1353, 2011.

[131] D. R. Forsey and R. H. Bartels, “Hierarchical B-spline refinement,” *ACM SIGGRAPH Comput. Graph.*, vol. 22, no. 4, pp. 205–212, Aug. 1988.

[132] D. R. Forsey and R. H. Bartels, “Surface fitting with hierarchical splines,” *ACM Trans. Graph.*, vol. 14, pp. 134–161, 1995.

[133] D. Schillinger and E. Rank, “An unfitted hp-adaptive finite element method based on hierarchical B-splines for interface problems of complex geometry,” *Comput. Methods Appl. Mech. Eng.*, vol. 200, no. 47–48, pp. 3358–3380, Nov. 2011.

[134] A.-V. Vuong, C. Giannelli, B. Jüttler, and B. Simeon, “A hierarchical approach to adaptive local refinement in isogeometric analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 200, no. 49–52, pp. 3554–3567, 2011.

[135] D. Schillinger, L. Ded??, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, and T. J. R. Hughes, “An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces,” *Comput. Methods Appl. Mech. Eng.*, vol. 249–252, pp. 116–150, 2012.

[136] T. W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri, “T-splines and T-NURCCs,” *ACM Trans. Graph.*, vol. 22, no. 3, p. 477, Jul. 2003.

[137] T. W. Sederberg, D. L. Cardon, G. T. Finnigan, N. S. North, J. Zheng, and T. Lyche, “T-spline simplification and local refinement,” *ACM Trans. Graph.*, vol. 23, no. 3, p. 276, Aug. 2004.

[138] X. Li, J. Zheng, T. W. Sederberg, T. J. R. Hughes, and M. A. Scott, “On linear independence of T-spline blending functions,” *Comput. Aided Geom. Des.*, vol. 29, no. 1, pp. 63–76, 2012.

[139] M. A. Scott, X. Li, T. W. Sederberg, and T. J. R. Hughes, “Local refinement of analysis-suitable T-splines,” *Comput. Methods Appl. Mech. Eng.*, vol. 213–216, pp. 206–222, 2012.

[140] X. Li and M. A. Scott, “On the Nesting Behavior of T-splines,” *Univ. Texas Austin*, vol. ICES repor, no. May, pp. 11–13, 2011.

[141] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, and T. W. Sederberg, “Isogeometric analysis using T-splines,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 5–8, pp. 229–263, 2010.

[142] M. R. Dörfel, B. Jüttler, and B. Simeon, “Adaptive isogeometric analysis by local h-refinement with T-splines,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 5–8, pp. 264–275, Jan. 2010.

[143] M. A. Scott, M. J. Borden, C. V. Verhoosel, T. W. Sederberg, and T. J. R. Hughes, “Isogeometric finite element data structures based on Bézier extraction of T-splines,” *Int. J. Numer. Methods Eng.*, vol. 88, no. 2, pp. 126–156, Oct. 2011.

[144] M. J. Borden, M. A. Scott, J. A. Evans, and T. J. R. Hughes, “Isogeometric finite element data structures based on Bézier extraction of NURBS,” *Int. J. Numer. Methods Eng.*, vol. 87, no. 1–5, pp. 15–47, Jul. 2011.

[145] “T-Splines.” [Online]. Available: http://www.tsplines.com/.

[146] D. Burkhart, B. Hamann, and G. Umlauf, “Iso-geometric finite element analysis based on Catmull-Clark subdivision solids,” *Eurographics Symp. Geom. Process.*, vol. 29, no. 5, pp. 1575–1584, 2010.

[147] H. Speleers, C. Manni, F. Pelosi, and M. L. Sampoli, “Isogeometric analysis with Powell–Sabin splines for advection–diffusion–reaction problems,” *Comput. Methods Appl. Mech. Eng.*, vol. 221–222, pp. 132–148, May 2012.

[148] B. Jüttler, A. Mantzaflaris, R. Perl, and M. Rumpf, “On numerical integration in isogeometric subdivision methods for PDEs on surfaces,” *Comput. Methods Appl. Mech. Eng.*, vol. 302, pp. 131–146, Apr. 2016.

[149] A. Riffnaller-Schiefer, U. H. Augsdörfer, and D. W. Fellner, “Isogeometric shell analysis with NURBS compatible subdivision surfaces,” *Appl. Math. Comput.*, vol. 272, pp. 139–147, Jan. 2016.

[150] Q. Pan, G. Xu, G. Xu, and Y. Zhang, “Isogeometric analysis based on extended Catmull–Clark subdivision,” *Comput. Math. with Appl.*, vol. 71, no. 1, pp. 105–119, Jan. 2016.

[151] M. A. Scott, D. C. Thomas, and E. J. Evans, “Isogeometric spline forests,” *Comput. Methods Appl. Mech. Eng.*, vol. 269, pp. 222–264, 2014.

[152] J. Deng, F. Chen, X. Li, C. Hu, W. Tong, Z. Yang, and Y. Feng, “Polynomial splines over hierarchical T-meshes,” *Graph. Models*, vol. 70, no. 4, pp. 76–86, 2008.

[153] T. Dokken, T. Lyche, and K. F. Pettersen, “Polynomial splines over locally refined box-partitions,” *Comput. Aided Geom. Des.*, vol. 30, no. 3, pp. 331–356, 2013.

[154] H. Al Akhras, T. Elguedj, A. Gravouil, and M. Rochette, “Isogeometric analysis-suitable trivariate NURBS models from standard B-Rep models,” *Comput. Methods Appl. Mech. Eng.*, vol. 307, pp. 256–274, 2016.

[155] T. J. R. Hughes, A. Reali, and G. Sangalli, “Efficient quadrature for NURBS-based isogeometric analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 199, no. 5–8, pp. 301–313, Jan. 2010.

[156] F. Auricchio, F. Calabr??, T. J. R. Hughes, A. Reali, and G. Sangalli, “A simple algorithm for obtaining nearly optimal quadrature rules for NURBS-based isogeometric analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 249–252, pp. 15–27, 2012.

[157] D. Schillinger, S. J. Hossain, and T. J. R. Hughes, “Reduced Bézier element quadrature rules for quadratic and cubic splines in isogeometric analysis,” *Comput. Methods Appl. Mech. Eng.*, vol. 277, pp. 1–45, Aug. 2014.

[158] F. AURICCHIO, L. B. DA VEIGA, T. J. R. HUGHES, A. REALI, and G. SANGALLI, “ISOGEOMETRIC COLLOCATION METHODS,” *Math. Model. Methods Appl. Sci.*, vol. 20, no. 11, pp. 2075–2107, Nov. 2010.

[159] F. Auricchio, L. Beirão da Veiga, T. J. R. Hughes, A. Reali, and G. Sangalli, “Isogeometric collocation for elastostatics and explicit dynamics,” *Comput. Methods Appl. Mech. Eng.*, vol. 249–252, pp. 2–14, Dec. 2012.

[160] D. Schillinger, J. a. Evans, A. Reali, M. a. Scott, and T. J. R. Hughes, “Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations,” *Comput. Methods Appl. Mech. Eng.*, vol. 267, pp. 170–232, Dec. 2013.

[161] J. Kiendl, F. Auricchio, L. Beir??o da Veiga, C. Lovadina, and A. Reali, “Isogeometric collocation methods for the Reissner-Mindlin plate problem,” *Comput. Methods Appl. Mech. Eng.*, vol. 284, pp. 489–507, 2015.

[162] L. De Lorenzis, J. A. Evans, T. J. R. Hughes, and A. Reali, “Isogeometric collocation: Neumann boundary conditions and contact,” *Comput. Methods Appl. Mech. Eng.*, vol. 284, pp. 21–54, 2014.

[163] F. Salmoiraghi, F. Ballarin, L. Heltai, and G. Rozza, “Isogeometric analysis-based reduced order modelling for incompressible linear viscous flows in parametrized shapes,” *Adv. Model. Simul. Eng. Sci.*, vol. 3, no. 1, p. 21, 2016.

[164] A. Manzoni, F. Salmoiraghi, and L. Heltai, “Reduced Basis Isogeometric Methods (RB-IGA) for the real-time simulation of potential flows about parametrized NACA airfoils,” *Comput. Methods Appl. Mech. Eng.*, vol. 284, pp. 1147–1180, 2015.

[165] A. Buffa and R. Vazquez, “Isogeometric analysis for electromagnetic scattering problems,” in *2014 International Conference on Numerical Electromagnetic Modeling and Optimization for RF, Microwave, and Terahertz Applications (NEMO)*, 2014, pp. 1–3.

[166] K. Takizawa, T. E. Tezduyar, J. Boben, N. Kostov, C. Boswell, and A. Buscher, “Fluid-structure interaction modeling of clusters of spacecraft parachutes with modified geometric porosity,” *Comput. Mech.*, vol. 52, no. 6, pp. 1351–1364, 2013.

[167] W. Liu, Z. Huang, Q. Liu, and J. Zeng, “An isogeometric analysis approach for solving the Reynolds equation in lubricated piston dynamics,” *Tribol. Int.*, vol. 103, pp. 149–166, 2016.

[168] W. Habchi, D. Eyheramendy, P. Vergne, and G. Morales-Espejel, “Stabilized fully-coupled finite elements for elastohydrodynamic lubrication problems,” *Adv. Eng. Softw.*, vol. 46, no. 1, pp. 4–18, 2012.

[169] M. Shahrbanozadeh, G.-A. Barani, and S. Shojaee, “Simulation of flow through dam foundation by isogeometric method,” *Eng. Sci. Technol. an Int. J.*, vol. 18, pp. 1–9, 2015.

[170] A.-T. Vuong, C. Ager, and W. A. Wall, “Two finite element approaches for Darcy and Darcy-Brinkman flow through deformable porous media — Mixed method vs. NURBS based (isogeometric) continuity,” *Comput. Methods Appl. Mech. Eng.*, vol. 305, pp. 634–657, Jun. 2016.

[171] A. R. Owens, J. A. Welch, J. K??ph??zi, and M. D. Eaton, “Discontinuous isogeometric analysis methods for the first-order form of the neutron transport equation with discrete ordinate (SN) angular discretisation,” *J. Comput. Phys.*, vol. 315, pp. 501–535, Jun. 2016.

[172] R. Rasool, C. J. Corbett, and R. A. Sauer, “A strategy to interface isogeometric analysis with Lagrangian finite elements-Application to incompressible flow problems,” *Comput. Fluids*, vol. 127, pp. 182–193, 2016.

[173] G. L. Fenves, “Object-oriented programming for engineering software development,” *Eng. Comput.*, vol. 6, no. 1, pp. 1–15, Dec. 1990.

[174] J. W. Baugh and D. R. Rehak, “Data Abstraction in Engineering Software Development,” *J. Comput. Civ. Eng.*, vol. 6, no. 3, pp. 282–301, Jul. 1992.

[175] S.-P. Scholz, “Elements of an object-oriented fem++ program in c++,” *Comput. Struct.*, vol. 43, no. 3, pp. 517–529, May 1992.

[176] P. R. B. Devloo, “Efficiency issues in an object oriented programming environment,” *Artif. Intell. Object Oriented Approaches Struct. Eng. eds Topping BHV, Papadrakakis M. ISBN 0-948749-29-6. Civil-Comp Press. Edinburgh*, pp. 147–151, 1994.

[177] Y. Dubois-Pe`lerin and T. Zimmermann, “Object-oriented finite element programming: III. An efficient implementation in C++,” *Comput. Methods Appl. Mech. Eng.*, vol. 108, no. 1–2, pp. 165–183, Jan. 1993.

[178] P. H. Menbtrey, P. Menétrey, and T. Zimmermann, “Object-oriented non-linear finite element analysis: Application to J2 plasticity,” *Comput. Struct.*, vol. 49, no. 5, pp. 767–777, 1993.

[179] R. Foerch, J. Besson, P. Pilvin, and G. Cailletaud, “Formulation des relations de comportement dans les calculs par éléments finis : approche C++,” 1995, pp. 547–552.

[180] F. Golay, P. Breitkopf, and G. Touzot, “SIC: Système Interactif de Conception,” 2002.

[181] P.VERPEAUX, T.CHARRAS, and A.MILLARD, “Castem2000 : une approche moderne du calcul des structure,” *Calc. des Struct. Intell. Artif.*, vol. Pluralis, p. p261, 1988.

[182] J. Mackerle, “Object-oriented programming in FEM and BEM: A bibliography (1990-2003),” *Adv. Eng. Softw.*, vol. 35, no. 6, pp. 325–336, 2004.

[183] G. W. Zeglinski, R. P. S. Han, and P. Aitchison, “Object oriented matrix classes for use in a finite element code using C++,” *Int. J. Numer. Methods Eng.*, vol. 37, no. 22, pp. 3921–3937, Nov. 1994.

[184] R. I. Mackie, “An object-oriented approach to calculation control in finite element programs,” *Comput. Struct.*, vol. 77, no. 5, pp. 461–474, Jul. 2000.

[185] P. Bomme and T. Zimmermann, “Towards Intelligent Objects in Finite Element Programming,” pp. 107–114.

[186] Y. Fang, K. Ravi-Chandar, and K. W. White, “Influence of Surface Residual Stress State on Crack Path Evolution in Polycrystalline Alumina,” *J. Am. Ceram. Soc.*, vol. 85, no. 7, pp. 1783–1787, Dec. 2004.

[187] H. Adeli and H. Kim, “Web-Based Interactive Courseware for Structural Steel Design Using Java,” *Comput. Civ. Infrastruct. …*, vol. 15, no. 2, pp. 158–166, Mar. 2000.

[188] G. Rio, H. Laurent, and G. Blès, “Asynchronous interface between a finite element commercial software ABAQUS and an academic research code HEREZH++,” *Adv. Eng. Softw.*, vol. 39, no. 12, pp. 1010–1022, 2008.

[189] D. Eyheramendy, “High Abstraction Level Frameworks for the Next Decade in Computational Mechanics,” B. H. V. Topping and G. R. M. Montero, Eds. Saxe-Cobourg Publications, 2006, pp. 41–61.

[190] P. Dadvand, R. Rossi, and E. Oñate, “An object-oriented environment for developing finite element codes for multi-disciplinary applications,” *Arch. Comput. Methods Eng.*, vol. 17, no. 3, pp. 253–297, 2010.

[191] M. R. Tonks, D. Gaston, P. C. Millett, D. Andrs, and P. Talbot, “An object-oriented finite element framework for multiphysics phase field simulations,” *Comput. Mater. Sci.*, vol. 51, no. 1, pp. 20–29, 2012.

[192] M. Yong-qi and F. Wei, “Object-oriented finite element analysis and programming in VC++,” *Appl. Math. Mech.*, vol. 23, no. 12, pp. 1437–1443, Dec. 2002.

[193] M. Nuggehally, Y. . Liu, S. . Chaudhari, and P. Thampi, “An Internet-based computing platform for the boundary element method,” *Adv. Eng. Softw.*, vol. 34, no. 5, pp. 261–269, May 2003.

[194] G. P. Nikishkov and H. Kanda, “Development of a Java engineering application for higher-order asymptotic analysis of crack-tip fields,” *Adv. Eng. Softw.*, vol. 30, no. 7, pp. 469–477, 1999.

[195] G. R. Miller, P. Arduino, J. Jang, and C. Choi, “Localized tensor-based solvers for interactive finite element applications using C++ and Java,” *Comput. Struct.*, vol. 81, no. 7, pp. 423–437, 2003.

[196] J. M. Bull, L. A. Smith, L. Pottage, and R. Freeman, “Benchmarking Java against C and Fortran for scientific applications,” in *Proceedings of the 2001 joint ACM-ISCOPE conference on Java Grande - JGI ’01*, 2001, pp. 97–105.

[197] J. Häuser, T. Ludewig, R. . Williams, R. Winkelmann, T. Gollnick, S. Brunett, and J. Muylaert, “A test suite for high-performance parallel Java,” *Adv. Eng. Softw.*, vol. 31, no. 8–9, pp. 687–696, Aug. 2000.

[198] R. Marchand, M. Charbonneau-Lefort, M. Dumberry, and B. Pronovost, “ARANEA, a program for generating unstructured triangular meshes with a JAVA graphics user interface,” *Comput. Phys. Commun.*, vol. 139, no. 2, pp. 172–185, 2001.

[199] N. T. Padial-Collins, W. B. VanderHeyden, D. Z. Zhang, E. D. Dendy, and D. Livescu, “Parallel operation of CartaBlanca on shared and distributed memory computers,” *Concurr. Comput. Pract. Exp.*, vol. 16, no. 1, pp. 61–77, Jan. 2004.

[200] W. B. VanderHeyden, E. D. Dendy, and N. T. Padial-Collins, “CartaBlanca—a pure-Java, component-based systems simulation tool for coupled nonlinear physics on unstructured grids—an update,” *Concurr. Comput. Pract. Exp.*, vol. 15, no. 35, pp. 431–458, Mar. 2003.

[201] L. Baduel, F. Baude, D. Caromel, C. Delbe, N. Gama, S. El Kasmi, and S. Lanteri, “A parallel object-oriented application for 3D electromagnetism,” in *18th International Parallel and Distributed Processing Symposium, 2004. Proceedings.*, 2004, vol. 0, no. C, pp. 7–16.

[202] G. P. Nikishkov, “Generating contours on FEM/BEM higher-order surfaces using Java 3D textures,” *Adv. Eng. Softw.*, vol. 34, no. 8, pp. 469–476, Aug. 2003.

[203] C. J. Riley, S. Chatterjee, and R. Biswas, “High-performance Java codes for computational fluid dynamics,” *Concurr. Comput. Pract. Exp.*, vol. 15, no. 35, pp. 395–415, Mar. 2003.

[204] G. P. Nikishkov, “Object oriented design of a finite element code in java,” *Comput. Model. Eng. Sci.*, vol. 11, no. 2, p. 81, 2006.

[205] G. Nikishkov, *Programming Finite Elements in Java™*. London: Springer London, 2010.

[206] D. Eyheramendy, C. Istil, I. Camille, J. Umr, T. De, I. De Lyon, U. Claude, and B. Lyon, “An advanced object model for an operator split method Application to elastoplasticity,” 2005, vol. 2005, pp. 7–12.

[207] B. C. P. Heng and R. I. Mackie, “Using design patterns in object-oriented finite element programming,” *Comput. Struct.*, vol. 87, no. 15–16, pp. 952–961, Aug. 2009.

[208] R. I. Mackie, “Object-Oriented Design of Pre-conditioned Iterative Equation Solvers using .NET.”

[209] R. W. Luft, J. M. Roesset, and Jerome J. Connor, “Automatic Generation of Finite Element Matricesle,” in *Electronic Computation*, 1970.

[210] R. H. Gunderson and A. Cetiner, “Element Stiffness Matrix Generator,” *J. Struct. Div.*, vol. 97, no. 1, pp. 363–375, 1971.

[211] A. K. Noor and C. M. Andersen, “Computerized symbolic manipulation in nonlinear finite element analysis,” *Comput. Struct.*, vol. 13, no. 1–3, pp. 379–403, Jun. 1981.

[212] D. Eyheramendy and T. Zimmermann, “Object-oriented finite elements III. Theory and application of automatic programming,” *Comput. Methods Appl. Mech. Eng.*, vol. 154, no. 97, pp. 41–68, 1998.

[213] D. K. Choi and S. Nomura, “Application of symbolic computation to two-dimensional elasticity,” *Comput. Struct.*, vol. 43, no. 4, pp. 645–649, 1992.

[214] N. I. Ioakimidis, “Elementary applications of MATHEMATICA to the solution of elasticity problems by the finite element method,” *Comput. Methods Appl. Mech. Eng.*, vol. 102, no. 1, pp. 29–40, 1993.

[215] G. Yagawa, G.-W. Ye, and S. Yoshimura, “A numerical integration scheme for finite element method based on symbolic manipulation,” *Int. J. Numer. Methods Eng.*, vol. 29, no. 7, pp. 1539–1549, May 1990.

[216] C.-Y. Yang, “An algebraic-expressed finite element model for symbolic computation,” *Comput. Struct.*, vol. 52, no. 5, pp. 1069–1077, 1994.

[217] P. P. Silvester and S. V. Chamlian, “Symbolic Generation of Finite Elements for Skin-Effect Integral Equations,” *IEEE Trans. Magn.*, vol. 30, no. 5, pp. 3594–3597, 1994.

[218] D. Eyheramendy and T. Zimmermann, “Object-Oriented Symbolic Derivation and Automatic Programming of Finite Elements in Mechanics,” *Eng. Comput.*, vol. 15, no. 1, pp. 12–36, Apr. 1999.

[219] D. Eyheramendy and T. Zimmermann, “Object-oriented finite elements II. A symbolic environment for automatic programming,” *Comput. Methods Appl. Mech. Eng.*, vol. 132, no. 96, pp. 277–304, 1996.

[220] J. Korelc, “Multi-language and multi-environment generation of nonlinear finite element codes,” *Eng. Comput.*, vol. 18, no. 4, pp. 312–327, 2002.

[221] Comsol, *Multiphysics COMSOL*, vol. 10. 2010.

[222] F. Hecht, “New development in freefem+,” *J. Numer. Math.*, vol. 20, no. 3–4, pp. 251–265, 2012.

[223] R. Vázquez, “A new design for the implementation of isogeometric analysis in Octave and Matlab: GeoPDEs 3.0,” *Comput. Math. with Appl.*, vol. 72, no. 3, pp. 523–554, Aug. 2016.

[224] S. Balay, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. C. Mcinnes, B. Smith, and H. Zhang, “PETSc Users Manual,” *ReVision*, vol. 2, no. March, pp. 1–211, 2010.

[225] L. M. M. Bernal, V. M. M. Calo, N. Collier, G. A. a. Espinosa, F. Fuentes, and J. C. C. Mahecha, “Isogeometric analysis of hyperelastic materials using petiGA,” *Procedia Comput. Sci.*, vol. 18, pp. 1604–1613, Jan. 2013.

[226] P. Vignal, L. Dalcin, N. O. Collier, and V. M. Calo, “Modeling phase-transitions using a high-performance, isogeometric analysis framework,” in *Procedia Computer Science*, 2014, vol. 29, pp. 980–990.

[227] A. F. Sarmiento, A. M. A. Cortes, D. A. Garcia, L. Dalcin, N. Collier, and V. M. Calo, “PetIGA-MF: a multi-field high-performance toolbox for structure-preserving B-splines spaces,” Feb. 2016.

[228] K. Sorli, M. Floater, K. O. Okstad, T. Kvamsdal, and A. F. Rasmussen, “Integration of CAD and FEM for three-dimensional problems using splines and Object-oriented programming, Finite element methods. Three-dimensional problems,” in *Proceedings of international conference,* 2000.

[229] B. Jüttler, U. Langer, A. Mantzaflaris, S. E. Moore, and W. Zulehner, “Geometry + Simulation Modules: Implementing Isogeometric Analysis,” *Pamm*, vol. 14, no. 1, pp. 961–962, Dec. 2014.

[230] M. S. Pauletti, M. Martinelli, N. Cavallini, P. Antolín, and P. Antolin, “Igatools: An Isogeometric Analysis Library,” *SIAM J. Sci. Comput.*, vol. 37, no. 4, pp. C465–C496, Jan. 2015.

[231] B. Patzák and Z. Bittnar, “Design of object oriented finite element code,” *Adv. Eng. Softw.*, vol. 32, no. 10–11, pp. 759–767, 2001.

[232] R. Cimrman, “SfePy -Write Your Own FE Application,” *PROC. 6th EUR. CONF. PYTHON Sci.*, 2013.

[233] R. Cimrman, “Enhancing SfePy with Isogeometric Analysis,” no. Euroscipy, pp. 65–72, 2014.

[234] “MFEM,” 2016. [Online]. Available: http://mfem.org/.

[235] A. Karatarakis, P. Karakitsios, and M. Papadrakakis, “GPU accelerated computation of the isogeometric analysis stiffness matrix,” *Comput. Methods Appl. Mech. Eng.*, vol. 269, pp. 334–355, 2014.

[236] M. Woźniak, “Fast GPU integration algorithm for isogeometric finite element method solvers using task dependency graphs,” *Journal of Computational Science*, 2014.

[237] D. Eyheramendy, “Advanced object-oriented techniques for coupled multiphysics,” in *Civil Engineering Computation: Tools and Techniques*, B. H. V. Topping, Ed. Saxe-Cobourg Publications, 2007, pp. 37–60.

[238] W. Callister and D. Rethwisch, *Materials science and engineering: an introduction*, vol. 94. 2007.

[239] L. R. G. Treloar, *The physics of rubber elasticity*. Clarendon Press, 2005.

[240] S. Lejeunes, A. Boukamel, and S. Méo, “Finite element implementation of nearly-incompressible rheological models based on multiplicative decompositions,” *Comput. Struct.*, vol. 89, no. 3–4, pp. 411–421, Feb. 2011.

[241] T. A. N. Van, “Sur la modélisation et la simulation du couplage thermo-chimio-mécanique au sein des élastomères chargés,” Aix-Marseille Université, 2012.

[242] L. Mullins, “Effect of Stretching on the Properties of Rubber,” *Rubber Chem. Technol.*, vol. 21, no. 2, pp. 281–300, 1948.

[243] A. Lion, “A constitutive model for carbon black filled rubber: Experimental investigations and mathematical representation,” *Contin. Mech. Thermodyn.*, vol. 8, no. 3, pp. 153–169, 1996.

[244] A. ROBISSON, “Comportement visco-hyperelastique endommageable d’elastomeres (sbr et pu) : prevision de la duree de vie en fatigue,” ENSM PARIS, 2000.

[245] J.-M. Martinez, “Modélisation et caractérisation du comportement hyper-visco-plastique d’un élastomère sous sollicitations multi-harmoniques et à différentes températures,” Université de la Méditerranée - Aix-Marseille II, 2005.

[246] P. G.Pichon, “Fatigue thermomécanique des élastomères polyuréthane : caractérisation expérimentale de l’évolution des microstructures et modélisation des échanges thermiques.,” 2010.

[247] C. Miehe, “Discontinuous and continuous damage evolution in Ogden-type large-strain elastic materials,” *Eur. J. Mech. A. Solids*, vol. 14, no. 5, pp. 697–720, 1995.

[248] G. A. Holzapfel and J. C. Simo, “Entropy elasticity of isotropic rubber-like solids at finite strains,” *Comput. Methods Appl. Mech. Eng.*, vol. 132, no. 1–2, pp. 17–44, May 1996.

[249] S. Reese and S. Govindjee, “Theoretical and Numerical Aspects in the Thermo-Viscoelastic Material Behaviour of Rubber-Like Polymers,” *Mech. TimeDependent Mater.*, vol. 1, no. 4, pp. 357–396, 1998.

[250] A. Boukamel, S. Méo, O. Débordes, and M. Jaeger, “A thermo-viscoelastic model for elastomeric behaviour and its numerical application,” *Arch. Appl. Mech.*, vol. 71, no. 12, pp. 785–801, 2001.

[251] S. Reese, “A micromechanically motivated material model for the thermo-viscoelastic material behaviour of rubber-like polymers,” *Int. J. Plast.*, vol. 19, no. 7, pp. 909–940, 2003.

[252] R. Behnke, H. Dal, and M. Kaliske, “An extended tube model for thermo-viscoelasticity of rubberlike materials: Parameter identification and examples,” *PAMM*, vol. 11, no. 1, pp. 353–354, Dec. 2011.

[253] S. Meo, “Modélisation numérique du comportement mécanique de structures en élastomère : de l’élasticité à la thermo-visco-hyperélasticité,” Université de la Méditerranée - Aix-Marseille II, 2000.

[254] Q. Yang, L. Stainier, and M. Ortiz, “A variational formulation of the coupled thermo-mechanical boundary-value problem for general dissipative solids,” *J. Mech. Phys. Solids*, vol. 54, no. 2, pp. 401–424, 2006.

[255] L. Stainier, “Approche énergétique variationnelle des problèmes thermomécaniques couplés.”

[256] K. J. Bathe, “The inf -- sup condition and its evaluation for mixed finite element methods,” *Comput. Struct.*, vol. 79, no. 2, pp. 243–252, 2001.

[257] G. N. Wells, L. J. Sluys, and R. De Borst, “A p-adaptive scheme for overcoming volumetric locking during plastic flow,” *Comput. Methods Appl. Mech. Eng.*, vol. 191, no. 29–30, pp. 3153–3164, 2002.

[258] U. Heisserer, S. Hartmann, A. Düster, and Z. Yosibash, “On volumetric locking-free behaviour of p-version finite elements under finite deformations,” *Commun. Numer. Methods Eng.*, vol. 24, no. 11, pp. 1019–1032, Jun. 2007.

[259] R. D. Cook and H. Saunders, “Concepts and Applications of Finite Element Analysis (2nd Edition),” *J. Press. Vessel Technol.*, vol. 106, no. 1, p. 127, 1984.

[260] E. P. Kasper and R. L. Taylor, “Mixed-enhanced strain method. Part I: Geometrically linear problems,” *Comput. Struct.*, vol. 75, no. 3, pp. 237–250, 2000.

[261] K. M. Mathisen, K. M. Okstad, T. Kvamsdal, and S. B. Raknes, “Isogeometric analysis of finite deformation nearly incompressible solids,” *Raken. Mek. (Journal Struct. Mech.*, vol. 44, no. 3, pp. 260–278, 2011.

[262] J. C. Simo and C. Miehe, “Associative coupled thermoplasticity at finite strains: Formulation, numerical analysis and implementation,” *Comput. Methods Appl. Mech. Eng.*, vol. 98, no. 1, pp. 41–104, Jul. 1992.

[263] R. W. Ogden, “Large Deformation Isotropic Elasticity: On the Correlation of Theory and Experiment for Compressible Rubberlike Solids,” *Proc. R. Soc. London A Math. Phys. Eng. Sci.*, vol. 328, no. 1575, 1972.

[264] S. Reese, M. Küssner, and B. D. Reddy, “A new stabilization technique for finite elements in non-linear elasticity,” *Int. J. Numer. Methods Eng.*, vol. 44, no. February 1998, pp. 1617–1652, 1999.

[265] N. Zander, S. Kollmannsberger, M. Ruess, Z. Yosibash, and E. Rank, “The Finite Cell Method for linear thermoelasticity,” *Comput. Math. with Appl.*, vol. 64, no. 11, pp. 3527–3541, Dec. 2012.

[266] P. Erbts and A. Düster, “Accelerated staggered coupling schemes for problems of thermoelasticity at finite strains,” *Comput. Math. with Appl.*, vol. 64, no. 8, pp. 2408–2430, 2012.

[267] G. A. Holzapfel and J. C. Simo, “Entropy elasticity of isotropic rubber-like solids at finite strains,” *Comput. Methods Appl. Mech. Eng.*, vol. 132, no. 96, pp. 17–44, 1996.

[268] T. A. N’Guyen, S. Lejeunes, D. Eyheramendy, and A. Boukamel, “A thermodynamical framework for the thermo-chemo-mechanical couplings in soft materials at finite strain,” *Mech. Mater.*, vol. 95, pp. 158–171, 2016.

[269] S. Lejeunes, “Private communications,” 2016.

[270] P. J. Flory, “Thermodynamic relations for high elastic materials,” *Trans. Faraday Soc.*, vol. 57, no. 0, p. 829, 1961.

[271] M. A. Scott, R. N. Simpson, J. A. Evans, S. Lipton, S. P. A. Bordas, T. J. R. Hughes, and T. W. Sederberg, “Isogeometric boundary element analysis using unstructured T-splines,” *Comput. Methods Appl. Mech. Eng.*, vol. 254, pp. 197–221, Feb. 2013.

[272] R. W. and K.-U. B. Andreas Apostolatos*,†, Robert Schmidt, “A Nitsche-type formulation and comparison of the most common domain decomposition methods in isogeometric analysis,” *Int. J. Numer. Methods Eng.*, no. 97, pp. 473–504, 2014.

[273] V. P. Nguyen, P. Kerfriden, M. Brino, S. P. A. Bordas, and E. Bonisoli, “Nitsche’s method for two and three dimensional NURBS patch coupling,” *Comput. Mech.*, vol. 53, no. 6, pp. 1163–1182, 2014.

[274] M. Ruess, D. Schillinger, A. I. Özcan, and E. Rank, “Weak coupling for isogeometric analysis of non-matching and trimmed multi-patch geometries,” *Comput. Methods Appl. Mech. Eng.*, vol. 269, pp. 46–71, 2014.

[275] X. Du, G. Zhao, and W. Wang, “Nitsche method for isogeometric analysis of Reissner-Mindlin plate with non-conforming multi-patches,” *Comput. Aided Geom. Des.*, vol. 35–36, pp. 121–136, 2015.

[276] Y. Guo and M. Ruess, “Nitsche’s method for a coupling of isogeometric thin shells and blended shell structures,” *Comput. Methods Appl. Mech. Eng.*, vol. 284, pp. 881–905, 2015.

[277] M. Margonari, “A finite element solver for thermo-mechanical problems,” p. 10, 2010.

[278] L. Noels and R. Radovitzky, “A general discontinuous Galerkin method for finite hyperelasticity. Formulation and numerical applications,” *Int. J. Numer. Methods Eng.*, vol. 68, no. 1, pp. 64–97, 2006.

[279] T. J. Hughes, R. L. Taylor, J. L. Sackman, C. U. B. D. I. V. O. F. S. ENGINEERING, and S. MECHANICS., *Finite Element Formulation and Solution of Contact/Impact Problems in Continuum Mechanics. III.* Defense Technical Information Center, 1975.