Accepted Manuscript

Computer methods in applied mechanics and engineering

Isogeometric Boundary Element Analysis of steady incompressible viscous flow, Part 2: 3-D problems

Gernot Beer, Vincenzo Mallardo, Eugenio Ruocco, Christian Dünser

PII: S0045-7825(18)30009-4
DOI: https://doi.org/10.1016/j.cma.2018.01.007
Reference: CMA 11727

To appear in: *Comput. Methods Appl. Mech. Engrg.*

Received date : 21 August 2017
Revised date : 1 January 2018
Accepted date : 4 January 2018

Please cite this article as: G. Beer, V. Mallardo, E. Ruocco, C. Dünser, Isogeometric Boundary Element Analysis of steady incompressible viscous flow, Part 2: 3-D problems, *Comput. Methods Appl. Mech. Engrg.* (2018), https://doi.org/10.1016/j.cma.2018.01.007

Isogeometric Boundary Element Analysis of steady incompressible viscous flow, Part 2: 3-D problems

Gernot Beer[a,*], Vincenzo Mallardo[b], Eugenio Ruocco[c], Christian Dünser[a]

[a] *Institute of Structural Analysis, Graz University of Technology, Lessingstraße 25/II, 8010 Graz, Austria*
[b] *Department of Architecture, University of Ferrara, Via Quartieri 8, 44121 Ferrara, Italy*
[c] *Department of Civil Engineering, Design, Building and Environment, University of Campania "L. Vanvitelli", Via Roma 28, 81031 Aversa, Caserta, Italy*

Abstract

This is a sequel to a previous paper [1] where a novel approach was presented to the 2-D Boundary Element analysis of steady incompressible viscous flow. Here the method is extended to three dimensions. NURBS basis functions are used for describing the geometry of the problem and for approximating the unknowns. In addition, the arising volume integrals are treated differently to published work and volumes are described by bounding NURBS surfaces instead of cells and only one mapping is used. The advantage of the present approach is that complex boundary shapes can be described with very few parameters and that no generation of cells is required. For the solution of the non-linear equations full and modified Newton-Raphson methods are used. A comparison of the two methods is made on the classical example of a forced cavity flow, where accurate two-dimensional solutions are available in the literature. Finally, it is shown on a practical example of an airfoil how more complex boundary shapes can be approximated with few parameters and a solution obtained with a small number of unknowns.

Keywords: BEM, isogeometric analysis, flow, incompressible

1. Introduction

Numerous approaches to numerically solve incompressible viscous flow problems can be found in the literature. Most publications use domain methods such as Finite Difference, Finite Elements or Finite Volumes (see for example [2]). A classical example to test published numerical methods is the forced flow in a cavity and very accurate solutions are available for comparison. In [3] for example an extremely fine finite difference mesh is used for the solution. We will use these solutions to compare our results.

Here we use the Boundary Element method (BEM). The advantage of this method is that, for linear problems, unknowns only exist on the boundary and that the solution

[*] Corresponding author. Tel.: +43 316 873 6181, fax: +43 316 873 6185, mail: *gernot.beer@tugraz.at*, web: *www.ifb.tugraz.at*

inside the domain satisfies the governing differential equations exactly. For nonlinear problems, such as the one discussed here, volume integrals arise which need to be dealt with and this will be discussed in more detail in the successive sections.

1.1. The Boundary Element method for viscous flow

The governing differential equations for steady incompressible viscous flow can be developed from the laws governing the conservation of mass and momentum and assume the following differential forms:

$$\begin{aligned} \frac{\partial u_j}{\partial x_j} &= 0 \\ \mu \frac{\partial^2 u_i}{\partial x_j \partial x_j} - \frac{\partial p}{\partial x_i} - \rho u_j \frac{\partial u_i}{\partial x_j} &= 0 \end{aligned} \tag{1}$$

where x_i is an Eulerian coordinate, u_i is a velocity vector, p is the pressure, ρ the mass density and μ the viscosity.

The requirement for the BEM is the existence of fundamental solutions of the differential equations. These solutions can be found for the nonlinear equations (1) only if we consider the non-linear terms as body forces. We rewrite the equations as:

$$\begin{aligned} \frac{\partial u_j}{\partial x_j} &= 0 \\ \mu \frac{\partial^2 u_i}{\partial x_j \partial x_j} - \frac{\partial p}{\partial x_i} + f_i &= 0 \end{aligned} \tag{2}$$

with

$$f_i = -\rho u_j \frac{\partial u_i}{\partial x_j} \tag{3}$$

Fundamental solutions of equations (2) can now be obtained for an infinite domain by substituting the Dirac-Delta function for the body force.

We define fluid stresses as:

$$\sigma_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \tag{4}$$

and the resulting tractions on boundary S:

$$t_i = \sigma_{ij} n_j - p n_i \tag{5}$$

where n_i is the unit vector normal to the boundary. Using the reciprocal theorem, the following integral equation is obtained (see for example [4] and [5]):

$$\begin{aligned} c_{ij}(y)\, \dot{u}_j(y) = \int_S \left[U_{ij}(y,x)\, t_j(x) - T_{ij}(y,x)\, \dot{u}_j(x) \right] dS(x) \\ + \int_{V_0} U_{ij}(y,\bar{x})\, f_j(\bar{x}) dV_0(\bar{x}) \end{aligned} \tag{6}$$

where $c_{ij}(y)$ is an integral free term, depending on the shape of the boundary, V_0 is the subdomain where body forces are present and S_0 is its boundary. $\bar{x}$ specifies a point inside V_0 and $\dot{u}_i$ is the velocity perturbation, i.e. the total velocity can be written as:

$$u_i(x) = \dot{u}_i(x) + u_i^0(x) \tag{7}$$

with u_i^0 the free stream velocity. The values associated with body forces are given by:

$$\begin{aligned} b_{ik}^0 &= \rho\, u_k(\bar{x})\, \dot{u}_i(\bar{x}) \\ t_i^0 &= b_{ik}^0\, n_k \end{aligned} \tag{8}$$

$U_{ij}(y,x)$ and $T_{ij}(y,x)$ are fundamental solutions for the velocity and traction at point x due to a source at point y.

In Equation (6) f_j involves derivatives of velocities. As has been shown in [6] these derivatives can be computed by using finite differences or by taking derivatives of an approximation of the velocity field. In both cases additional computational work needs to be done and errors are introduced.

Alternatively, the requirement of computing derivatives can be eliminated by applying the divergence theorem to the volume integral of Equation (6) resulting in [1]:

$$\begin{aligned} c_{ij}(y)\, \dot{u}_j(y) = &\int\limits_S \left[U_{ij}(y,x)\, t_j(x) - T_{ij}(y,x)\, \dot{u}_j(x)\right] dS \\ &- \int\limits_{S_0} U_{ij}(y,x)\, t_j^0(x) dS_0 + \int\limits_{V_0} U_{ij,k}(y,\bar{x})\, b_{jk}^0(\bar{x}) dV_0 \end{aligned} \tag{9}$$

where $U_{ij,k}(y,x)$ is the k-th derivative of $U_{ij}(y,x)$ listed in the Appendix. This is the approach used here.

1.2. Previous work and novelty of our approach

Internal cells were used in most published methods for the evaluation of the volume integrals. Cells are basically like finite elements with the subtle difference that they are only used for evaluating integrals and not for approximating the unknowns (fundamental solutions, that satisfy the linear differential equation are used to approximate the unknowns inside the domain). In [5] Equation (9) was used and therefore the computation of derivatives of velocities was avoided. Solving the forced cavity flow problem it was found that a very accurate integration scheme and a fine mesh of cells had to be used to obtain good results. For higher Reynolds numbers it was necessary to use a full Newton-Raphson method, where the left hand side is updated at every iteration step, to get results that converge to the right solution. Results for Reynolds numbers up to 1000 are presented, but details on how the tangent operator was determined are missing. In [7] the same approach as in [5] is used. A solution of the problem with the BEM can also be found in [6], where Equation (6) is used and the derivatives of velocities are

[1] Details are presented in [4]

computed using either finite differences or by taking the derivatives of basis functions that approximate the solution inside cells. A modified Newton-Raphson method with relaxation is used and good results are obtained for Reynolds numbers up to 1000. All these solutions are for plane problems only. In [8] the method is extended to three dimensions. Results for the forced cavity flow problem are only presented for Reynolds numbers up to 100.

Isogeometric analysis [9] has gained significant popularity in the last decade because of the fact that geometry data can be taken directly from Computer Aided Design (CAD) programs, potentially eliminating the need for mesh generation. The novelty of our approach is that instead of Lagrange polynomials, that are used in the quoted published work, NURBS basis functions are used for describing the geometry and the variation of the unknowns. The use of these functions means that fewer parameters are needed to describe complex geometries accurately. Using NURBS to approximate values at the boundary also provides greater flexibility with respect to refinement options. The geometry independent field approximation method, that has already been used successfully in [10] and [11], means that approximation of the unknown is uncoupled from the geometry definition. Another novelty is that instead of cells, a mapping method that was first introduced in [12, 13], is used, abolishing the requirement of generating a cell mesh. Finally, it is shown how the tangent operator is computed and a comparison between the modified and full Newton-Raphson methods is presented.

2. Numerical implementation

As in majority of previous work on the isogeometric BEM [10, 11, 14, 15, 16, 17, 18, 19, 20] we use the collocation method, i.e. we write the integral equations for a finite number (N) of source points at locations $\mathbf{y}_n$. Changing to matrix notation, the integral equations are re-written as:

$$\begin{aligned}\mathbf{c}\left(\mathbf{y}_n\right)\dot{\mathbf{u}}\left(\mathbf{y}_n\right) = &\int_S \mathbf{U}\left(\mathbf{y}_n,\mathbf{x}\right)\mathbf{t}\left(\mathbf{x}\right)\mathrm{d}S - \int_S \mathbf{T}\left(\mathbf{y}_n,\mathbf{x}\right)\dot{\mathbf{u}}\left(\mathbf{x}\right)\mathrm{d}S \\ &- \int_{S_0} \mathbf{U}\left(\mathbf{y}_n,\bar{\mathbf{x}}\right)\mathbf{t}_0\left(\bar{\mathbf{x}}\right)\mathrm{d}S_0 + \int_{V_0} \mathbf{U}'\left(\mathbf{y}_n,\bar{\mathbf{x}}\right)\mathbf{b}_0\left(\bar{\mathbf{x}}\right)\mathrm{d}V_0\end{aligned} \tag{10}$$

with $n = \{1,\ldots,N\}$. In the above $\mathbf{c}\left(\mathbf{y}_n\right)$ is a matrix containing integral free terms, $\dot{\mathbf{u}}(\mathbf{x})$ and $\mathbf{t}(\mathbf{x})$ are vectors containing velocity perturbations and tractions at point $\mathbf{x}$ on the boundary. $\mathbf{U}\left(\mathbf{y}_n,\mathbf{x}\right)$ and $\mathbf{T}\left(\mathbf{y}_n,\mathbf{x}\right)$ are matrices containing fundamental solutions listed in the Appendix. $\mathbf{b}_0\left(\bar{\mathbf{x}}\right)$ is a body force vector at points $\bar{\mathbf{x}}$ inside V_0.

Using the Einstein summation convention we can write for the last integrand in Equation (9):

$$\begin{aligned}U_{ij,k}(y,\bar{x})\,b^0_{jk}(x) = &\,U'_{i11}\,b^0_{11} + U'_{i12}\,b^0_{12} + U'_{i13}\,b^0_{13} \\ &+U'_{i21}\,b^0_{21} + U'_{i22}\,b^0_{22} + U'_{i23}\,b^0_{23} \\ &+U'_{i31}\,b^0_{31} + U'_{i32}\,b^0_{32} + U'_{i33}\,b^0_{33}\end{aligned} \tag{11}$$

With the fundamental solution U'_{ijk} listed in the Appendix this can be converted into a matrix multiplication involving the matrix

$$\mathbf{U}' = \begin{pmatrix} U'_{111} & U'_{112} & U'_{113} & U'_{121} & U'_{122} & U'_{123} & U'_{131} & U'_{132} & U'_{133} \\ U'_{211} & U'_{212} & U'_{213} & U'_{221} & U'_{222} & U'_{223} & U'_{231} & U'_{232} & U'_{233} \\ U'_{311} & U'_{312} & U'_{313} & U'_{321} & U'_{322} & U'_{323} & U'_{331} & U'_{332} & U'_{333} \end{pmatrix} \tag{12}$$

and the vector

$$\mathbf{b}_0(\bar{x}) = \begin{pmatrix} b^0_{11} & b^0_{12} & b^0_{13} & b^0_{21} & b^0_{22} & b^0_{23} & b^0_{31} & b^0_{32} & b^0_{33} \end{pmatrix}^T \tag{13}$$

The initial traction vector is given by:

$$\mathbf{t}_0 = \mathbf{N}\,\mathbf{b}_0 \tag{14}$$

where

$$\mathbf{N} = \begin{pmatrix} n_x & n_y & n_z & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & n_x & n_y & n_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & n_x & n_y & n_z \end{pmatrix} \tag{15}$$

The integrals over the domain V_0 and boundary S_0 need only be evaluated if the body force term is not zero. In practice this means that volume integration can be avoided if the body force is negligible, i.e. the integration will usually be restricted to an area near the boundary.

2.1. Discretization

For the discretization of the surface integrals over S we divide the boundary into patches and use a geometry independent field approximation approach for each patch, i.e. we use different basis functions for the description of the geometry and for the field values.

$$\begin{aligned} \mathbf{x}^e &= \sum_{i=1}^{I}\sum_{j=I}^{J} R_{ij}(s,t)\,\mathbf{x}^e_{ij} \\ \dot{\mathbf{u}}^e &= \sum_{i=1}^{I^u}\sum_{j=I}^{J^u} R^u_{ij}(s,t)\,\dot{\mathbf{u}}^e_{ij} \\ \mathbf{t}^e &= \sum_{i=1}^{I^t}\sum_{j=I}^{J^t} R^t_{ij}(s,t)\,\mathbf{t}^e_{ij} \end{aligned} \tag{16}$$

In above equations the superscript e refers to the number of the patch, R_{ij}, R^u_{ij} and R^t_{ij} are NURBS basis functions[2] with respect to the local coordinates s,t for defining the geometry and for approximating velocities and tractions respectively. $\mathbf{x}^e_{ij}$ specify the location of control points and $\dot{\mathbf{u}}^e_{ij}$ and $\mathbf{t}^e_{ij}$ are the parameters for velocities and tractions.

[2] Considering the vastly increasing literature on isogeometric analysis we refrain from presenting the equations for NURBS. The interested reader is referred to [21]

I, I^u, I^t are the number of parameters for each patch in s-direction and J, J^u, J^t are the number of parameters in t-direction.

2.2. Discretized integral equations

For establishing the system of equations it is convenient to simplify the description of the boundary values by introducing one subscript instead of two:

$$\begin{aligned}\dot{\mathbf{u}}^e &= \sum_{k=1}^{K^u} R^u_k(s,t)\,\dot{\mathbf{u}}^e_k \\ \mathbf{t}^e &= \sum_{k=1}^{K^t} R^t_k(s,t)\,\mathbf{t}^e_k\end{aligned} \tag{17}$$

where the parameter points are numbered consecutively first in s and then t direction and K^u, K^t refers to the total number of parameters.

Inserting the approximations into the integral equations and applying the rigid body trick for eliminating the free term and singular integration involving Kernel $\mathbf{T}$ as outlined in detail in [21] the following discretized integral equations can be obtained:

$$\begin{aligned}\sum_{e=1}^{E}\sum_{k=1}^{K^t} \triangle\mathbf{U}^e_{nk}\mathbf{t}^e_k &= \sum_{e=1}^{E}\sum_{k=1}^{K^u} \triangle\mathbf{T}^e_{nk}\dot{\mathbf{u}}^e_k - \mathbf{T}_n \sum_{k=1}^{K^u} R^u_k(s_n,t_n)\,\dot{\mathbf{u}}^{en}_k \\ &+ \int_{S_0} \mathbf{U}(\mathbf{y}_n,\bar{\mathbf{x}})\,\mathbf{t}_0(\bar{\mathbf{x}})\,\mathrm{d}S_0 - \int_{V_0} \mathbf{U}(\mathbf{y}_n,\bar{\mathbf{x}})\,\mathbf{b}_0(\bar{\mathbf{x}})\,\mathrm{d}V_0 \\ &\qquad\qquad \text{for} \quad n = 1,2,3\cdots N\end{aligned} \tag{18}$$

with en specifying the patch that contains the collocation point n and E the total number of patches. Equation (18) can be re-written as:

$$\begin{aligned}\sum_{e=1}^{E} \triangle\mathbf{U}^e_n\mathbf{t}^e &= \sum_{e=1}^{E} [\triangle\mathbf{T}^e_n\dot{\mathbf{u}}^e - \mathbf{T}^e_{n0}]\,\dot{\mathbf{u}}^e \\ &+ \int_{S_0} \mathbf{U}(\mathbf{y}_n,\bar{\mathbf{x}})\,\mathbf{t}_0(\bar{\mathbf{x}})\,\mathrm{d}S_0 - \int_{V_0} \mathbf{U}(\mathbf{y}_n,\bar{\mathbf{x}})\,\mathbf{b}_0(\bar{\mathbf{x}})\,\mathrm{d}V_0\end{aligned} \tag{19}$$

where

$$\begin{array}{rcl}\triangle\mathbf{U}^e_n &=& \begin{pmatrix} \triangle\mathbf{U}^e_{n1} & \triangle\mathbf{U}^e_{n2} & \cdots \end{pmatrix} \\ \triangle\mathbf{T}^e_n &=& \begin{pmatrix} \triangle\mathbf{T}^e_{n1} & \triangle\mathbf{T}^e_{n2} & \cdots \end{pmatrix}\end{array} \tag{20}$$

and

$$\begin{array}{lcl}\mathbf{T}^e_{n0} &=& \begin{pmatrix} \mathbf{T}_n R^u_1(s_n,t_n) & \mathbf{T}_n R^u_2(s_n,t_n) & \cdots \end{pmatrix} \\ \quad\text{if} & n \in e & \\ \mathbf{T}^e_{n0} &=& \begin{pmatrix} 0 & 0 & \cdots \end{pmatrix} \\ & \text{otherwise} & \end{array} \tag{21}$$

The coefficients of the sub-matrices are given by:

$$\triangle \mathbf{U}^{e}_{nk} = \int_{0}^{1}\int_{0}^{1} \mathbf{U}\left(\mathbf{y}_n, \mathbf{x}^e(s,t)\right) R^t_k(s,t)\, J\, ds\, dt \tag{22}$$

$$\triangle \mathbf{T}^{e}_{nk} = \int_{0}^{1}\int_{0}^{1} \mathbf{T}\left(\mathbf{y}_n, \mathbf{x}^e(s,t)\right) R^u_k(s,t)\, J\, ds\, dt$$

$$\mathbf{T}_n = \sum_{e=1}^{E}\int_{0}^{1}\int_{0}^{1} \mathbf{T}\left(\mathbf{y}_n, \mathbf{x}^e(s,t)\right) J\, ds\, dt \tag{23}$$

In the above J is the Jacobian of the transformation from local s,t to global x,y,z coordinate systems.

After assembly the following system of equations

$$[\mathbf{U}]\{\mathbf{t}\} = ([\mathbf{T}] - [\mathbf{T}]_0)\{\dot{\mathbf{u}}\} + \{\mathbf{F}\}_0 \tag{24}$$

is obtained, where $[\mathbf{U}]$, $[\mathbf{T}]$ are matrices assembled from patch contributions (22) and $\{\mathbf{t}\}$, $\{\dot{\mathbf{u}}\}$ are vectors that collect all traction and perturbation velocity components on points $\mathbf{y}_n$. $[\mathbf{T}]_0$ is a matrix relating to rigid body modes and $\{\mathbf{F}\}_0$ relates to the integrals involving body forces. Either $\mathbf{t}$ or $\mathbf{u}$ must be known on the boundary, so for a mixed boundary problem we have

$$[\mathbf{L}]\{\mathbf{a}\} = \{\mathbf{F}\} + \{\mathbf{F}\}_0 \tag{25}$$

where $[\mathbf{L}]$ contains a mixture of $[\mathbf{U}]$, $[\mathbf{T}]$ coefficients and $\{\mathbf{a}\}$ contains a mixture of unknown tractions and velocities. $\{\mathbf{F}\}$ is a vector computed with known boundary values.

2.3. Evaluation of boundary integrals

The boundary integrals are evaluated efficiently using Gauss Quadrature. To reduce the number of Gauss points a Quadtree method is used. Details can be found in [13].

3. Basic approach for dealing with volume terms

The basic approach is to solve the problem in an iterative way. First the linear problem is solved. Then the solution is modified to account for the presence of body forces.

The procedure can be summarized as follows:

1. Solve the linear problem and determine the velocity perturbation $\dot{\mathbf{v}}$ inside V_0.
2. Determine the increment in body force $\mathbf{b}_0$ from Equation (8).
3. Compute new right hand side $\mathbf{F}_0$ by evaluating the arising volume integral.
4. If full Newton-Raphson, compute a new left hand side
5. Solve for the new right hand side and compute a new increment of velocity perturbation $\dot{\mathbf{v}}$.
6. Repeat 2. to 5. until $\mathbf{b}_0$ is sufficiently small.

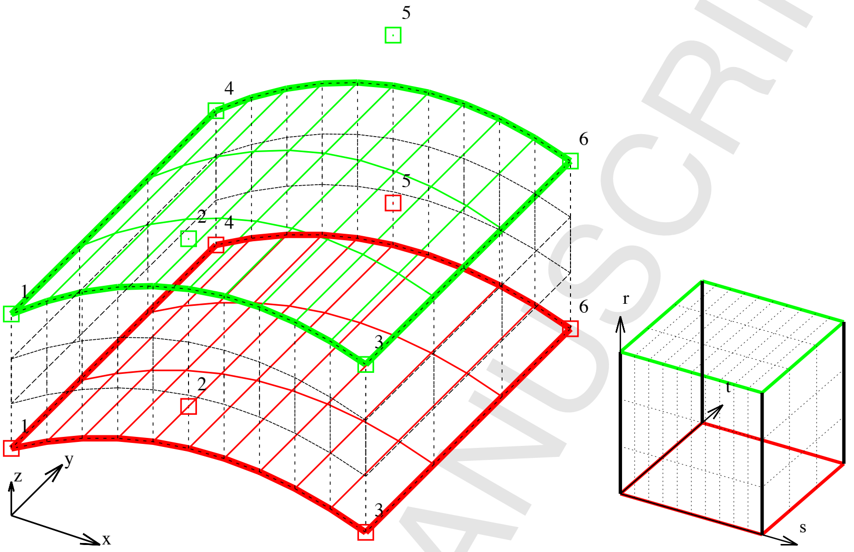

Figure 1: Example of the geometry definition of V_0 with the proposed approach. Left: V_0 in the global coordinate system. The bounding surfaces are colour coded and the associated control points depicted by coloured and numbered hollow squares. Right: the local map. Note that in contrast to the cell based approach no discretization and only one map is involved.

4. Geometry definition of V_0

The first task is the description of the geometry of the subdomain V_0. The geometry of the domain is defined by two NURBS surfaces and a linear interpolation between them as explained in detail in [13].

We establish a local coordinate system $\mathbf{s} = (s,t,r)^\intercal = [0,1]^3$ as shown in Figure 1 and perform the integration in this system and then map it to the global (x,y,z)-system. The global coordinates of a point $\mathbf{x}$ with the local coordinates $\mathbf{s}$ are given by

$$\mathbf{x}(s,t,r) = (1-r)\,\mathbf{x}^I(s,t) + r\,\mathbf{x}^{II}(s,t) \tag{26}$$

where

$$\mathbf{x}^I(s,t) = \sum_{k=1}^{K^I} R_k^I(s,t)\,\mathbf{x}_k^I \quad \text{and} \quad \mathbf{x}^{II}(s,t) = \sum_{k=1}^{K^{II}} R_k^{II}(s,t)\,\mathbf{x}_k^{II}. \tag{27}$$

The superscript I relates to the bottom (red) surface and II to the top (green) surface and $\mathbf{x}_k^I$, $\mathbf{x}_k^{II}$ are control point coordinates. K^I and K^{II} represent the number of control points, $R_k^I(s,t)$ and $R_k^{II}(s,t)$ are NURBS basis functions.

Remark 1: It should be noted that in this mapping approach the bottom and top surfaces may have a different number of control points.

The derivatives are given by

$$\begin{aligned}
\frac{\partial \mathbf{x}(s,t,r)}{\partial s} &= (1-r)\frac{\partial \mathbf{x}^{I}(s,t)}{\partial s} + r\frac{\partial \mathbf{x}^{II}(s,t)}{\partial s} \\
\frac{\partial \mathbf{x}(s,t,r)}{\partial t} &= (1-r)\frac{\partial \mathbf{x}^{I}(s,t)}{\partial t} + r\frac{\partial \mathbf{x}^{II}(s,t)}{\partial t} \\
\frac{\partial \mathbf{x}(s,t,r)}{\partial r} &= -\mathbf{x}^{I}(s,t) + \mathbf{x}^{II}(s,t)
\end{aligned} \tag{28}$$

where for example:

$$\frac{\partial \mathbf{x}^{I}(s,t)}{\partial s} = \sum_{k=1}^{K^I} \frac{\partial R_k^I(s,t)}{\partial s}\,\mathbf{x}_k^I \quad \text{and} \quad \frac{\partial \mathbf{x}^{II}(s,t)}{\partial s} = \sum_{k=1}^{K^{II}} \frac{\partial R_k^{II}(s,t)}{\partial s}\,\mathbf{x}_k^{II}. \tag{29}$$

The Jacobian matrix of this mapping is

$$\mathbf{J} = \begin{pmatrix} \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} & \frac{\partial z}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} & \frac{\partial z}{\partial t} \\ \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} & \frac{\partial z}{\partial r} \end{pmatrix} \tag{30}$$

and the Jacobian is $J(\mathbf{s}) = |\mathbf{J}|$.

5. Computation of $\{\mathbf{F}\}_0$

Here we discuss the computation of the right hand side during iteration. This involves the evaluation of the integrals in Equation (10) over S_0 and V_0. In order to limit the number of internal point evaluations we propose to compute results on grid points inside V_0 and to interpolate to the required Gauss point locations using either linear or quadratic interpolation (for details see [13]).

5.1. Computation of the surface integral over S_0

For the computation of the surface integral the surface is divided into integration regions, based on the location of the grid points. For it's evaluation the same procedures as explained in 2.3 are used. Gauss point values on the surface are extrapolated from internal points.

5.2. Computation of the volume integral over V_0

Here the volume is divided into N_s integration regions. Two adjacent grid points define the edges of an integration region. For integration region n_s the transformation from $\mathbf{s}$ coordinates to $\boldsymbol{\xi} = (\xi, \eta, \zeta)^{\mathsf{T}} = [-1,1]^3$ is given by

$$\begin{aligned} s &= \tfrac{\Delta s_n}{2}(1+\xi) + s_{n_s} \\ t &= \tfrac{\Delta t_n}{2}(1+\eta) + t_{n_s} \\ r &= \tfrac{\Delta r_n}{2}(1+\zeta) + r_{n_s} \end{aligned} \tag{31}$$

where $\Delta s_n \times \Delta t_n \times \Delta r_n$ denotes the size of the integration region and s_n, t_n, r_n are the edge coordinates. The Jacobian of this transformation is $J^n_\xi = \frac{1}{8}\Delta s_n \Delta t_n \Delta r_n$.

The sub vector of $\{\mathbf{F}\}^{V_0}_0$ related to collocation point n can be written as:

$$\mathbf{F}^{V_0}_{0n} = \sum_{n_s=1}^{N_s} \int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} \mathbf{U}'(\mathbf{y}_n, \bar{\mathbf{x}}(\xi,\eta,\zeta))\, \dot{\mathbf{b}}_0(\bar{\mathbf{x}}(\xi,\eta,\zeta))\, J(\mathbf{s})\, J^{n_s}_\xi \, \mathrm{d}\xi\, \mathrm{d}\eta\, \mathrm{d}\zeta \tag{32}$$

where $J(\mathbf{s})$ is the Jacobian of the mapping between $\mathbf{s}$ and $\mathbf{x}$ coordinate systems.

Applying Gauss integration we have:

$$\mathbf{F}^{V_0}_{0n} \approx \sum_{n_s=1}^{N_s}\sum_{m=1}^{M}\sum_{l=1}^{L}\sum_{k=1}^{K} \mathbf{U}'(\mathbf{y}_n, \bar{\mathbf{x}}(\xi_m,\eta_l,\zeta_k))\, \dot{\mathbf{b}}_0(\bar{\mathbf{x}}(\xi_m,\eta_l,\zeta_k))\, J(\mathbf{s})\, J^{n_s}_\xi\, W_m W_l W_k \tag{33}$$

where M, L and K are the number of integration points in ξ, η and ζ directions respectively. To determine the number of Gauss points necessary for an accurate integration we consider that, whereas there is usually a moderate variation of body force, the Kernel $\mathbf{U}$ is $\mathcal{O}(r^{-1})$ so the number of integration points has to be increased if $\mathbf{y}_n$ is close to V_0.

If the integration region includes the collocation point $\mathbf{y}_n$, then the integrand tends to infinity as the point is approached. To deal with the integration involving the weakly singular Kernel we perform the integration in a local coordinate system, where the Jacobian tends to zero as the singularity point is approached. For this we divide the integration region into tetrahedral sub-regions. The transformation from the local $\boldsymbol{\xi}$ coordinate system, in which the Gauss coordinates are defined, to global coordinates involves the following transformation steps:

1. from $\boldsymbol{\xi}$ to a local system $(\sigma, \tau, \rho)^{\mathsf{T}} = [0,1]^3$
2. from (σ, τ, ρ) to $\mathbf{s}$
3. from $\mathbf{s}$ to $\mathbf{x}$

Steps 1 and 3 have already been discussed, so we concentrate on explaining the second step. Referring to Figure 2 we assume that the singular point is an edge point of the integration region.

For this case the transformation is as follows: First we determine the local coordinates $\mathbf{s}_1$ to $\mathbf{s}_5$ of the edge points of the tetrahedron, with 5 being the singularity point. Next we define the bottom surface using NURBS using points 1 to 4 as control points

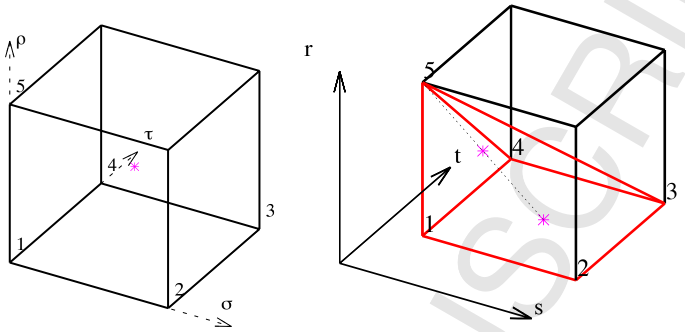

Figure 2: Singular volume integration, showing a tetrahedral subregion and the mapping from the local σ, τ, ρ system to the $\mathbf{s}$ coordinate system. A point with the local coordinates $\sigma = \tau = \rho = 0.5$ (i.e. $\xi = \eta = \zeta = 0$) is shown.

and map the coordinates of the point (σ, τ) onto this surface:

$$\mathbf{s}_0(\sigma,\tau) = \sum_{i=1}^{4} R_i(\sigma,\tau)\,\mathbf{s}_i \tag{34}$$

where $R_i(\sigma,\tau)$ are linear basis functions. The final map is obtained by interpolation in the ρ direction:

$$\mathbf{s}(\sigma,\tau,\rho) = (1-\rho)\,\mathbf{s}_0(\sigma,\tau) + \rho\,\mathbf{s}_5 \tag{35}$$

The Jacobian matrix of this transformation is given by:

$$\mathbf{J} = \begin{pmatrix} (1-\rho)\frac{\partial \mathbf{s}_0}{\partial \sigma} \\ (1-\rho)\frac{\partial \mathbf{s}_0}{\partial \tau} \\ \mathbf{s}_5 - \mathbf{s}_0 \end{pmatrix} \tag{36}$$

The Jacobian of this transformation tends to zero as the singular point is approached.

Since the right hand side has to be evaluated for every iteration step it is convenient to precompute matrices that multiply with the values of $\mathbf{b}_0$ at grid points. The value of body force at any location s,t,r can then be computed using:

$$\mathbf{b}_0(s,t,r) = \sum_{k=1}^{K} M_k(s,t,r)\,\mathbf{b}_{0k} \tag{37}$$

where $M_k(s,t,r)$ are either linear or quadratic interpolation functions, K is the number of grid points and $\mathbf{b}_{0k}$ are values of body force at grid points. With this the following matrix equation can be written:

$$\{\mathbf{F}_0\} = [\mathbf{B}]\{\mathbf{b}_0\} \tag{38}$$

where $\{\mathbf{b}_0\}$ is a vector of body force values at grid points and $[\mathbf{B}] = \left([\mathbf{B}]^S - [\mathbf{B}]^V\right)$, the sub-matrices of which are given by

$$\mathbf{B}_{nk}^{S} = \int_{S_0} \mathbf{U}(\mathbf{y}_n, \bar{\mathbf{x}}) M_k \mathbf{N} \,\mathrm{d}S_0 \tag{39}$$

and

$$\mathbf{B}_{nk}^{V} = \int_{V_0} \mathbf{U}'(\mathbf{y}_n, \bar{\mathbf{x}}) M_k \,\mathrm{d}V_0 \tag{40}$$

6. Computation of results inside V_0

The solution algorithm requires the evaluation of velocity perturbations inside the inclusion V_0. The perturbation velocity vector $\dot{\mathbf{v}}$ at any internal point $\mathbf{y}_i$ can be computed by[3]

$$\begin{aligned} \dot{\mathbf{v}}(\mathbf{y}_i) = & \int_{S} \mathbf{U}(\mathbf{y}_i, \mathbf{x}) \mathbf{t}(\mathbf{x}) \,\mathrm{d}S - \int_{S} \mathbf{T}(\mathbf{y}_i, \mathbf{x}) \dot{\mathbf{u}}(\mathbf{x}) \,\mathrm{d}S \\ & - \int_{S_0} \mathbf{U}(\mathbf{y}_i, \bar{\mathbf{x}}) \mathbf{t}_0(\bar{\mathbf{x}}) \,\mathrm{d}S_0 + \int_{V_0} \mathbf{U}'(\mathbf{y}_i, \bar{\mathbf{x}}) \mathbf{b}_0(\bar{\mathbf{x}}) \,\mathrm{d}V_0 \end{aligned} \tag{41}$$

The above equation can be written in matrix notation as:

$$\{\dot{\mathbf{v}}\} = [\mathbf{A}]\{\mathbf{t}\} - [\mathbf{C}]\{\dot{\mathbf{u}}\} + [\mathbf{D}]\{\mathbf{b}_0\} \tag{42}$$

where matrices $[\mathbf{A}]$ and $[\mathbf{C}]$ are assembled from element contributions of Kernel basis function products and $[\mathbf{D}] = \left([\mathbf{D}]^V - [\mathbf{D}]^S\right)$ where $[\mathbf{D}]^V$ and $[\mathbf{D}]^S$ are computed as shown in equation (39) and (40) and replacing $\mathbf{y}_n$ with $\mathbf{y}_i$.

7. Iterative procedure

There are two possibilities for the iterative procedure: modified Newton-Raphson or full Newton-Raphson. In the former the left hand side of the system of equations is not changed and only a new right hand side is computed at each iteration, whereas in the latter the left hand side is changed at every iteration.

7.1. Modified Newton-Raphson

The iterative procedure for modified Newton-Raphson is shown in Algorithm 1.

For the first iteration the unknown are computed by

$$[\mathbf{L}]\{\mathbf{a}\}^0 = \{\mathbf{F}\} \tag{43}$$

[3] The velocities at internal points are referred to as $\dot{\mathbf{v}}$ to distinguish them from the boundary velocities $\dot{\mathbf{u}}$.

```
INITIALIZATION;
solve for unknowns ${\bf a}\}^0$ using Equation (43);
compute velocities at internal points $\{\dot{\bf v}\}^0$ using Equation (45);
compute body forces $\{{\bf b}_0\}^0$ using Equation (8);
compute vector $\{{\bf F}\}_0^0$ using Equation (38);
$k = 1$;
ITERATION;
while residual > Tolerance do
    solve for unknowns $\{{\bf a}\}^k$ using Equation (44);
    set $\{{\bf a}\} = \beta\,\{{\bf a}\}^k + (1-\beta)\,\{{\bf a}\}^{k-1}$;
    compute velocities at internal points $\{\dot{\bf v}\}^k$ using Equation (46);
    set $\{{\bf v}\} = \beta\,\{{\bf v}\}^k + (1-\beta)\,\{{\bf v}\}^{k-1}$;
    compute body forces $\{{\bf b}_0\}^k$ using Equation (8);
    compute vector $\{{\bf F}\}_0^k$ using Equation (38);
    compute residual;
    $k = k+1$
end
```

Algorithm 1: Modified Newton-Raphson

For the subsequent iteration we have

$$[\mathbf{L}]\,\{\mathbf{a}\}^k = \{\mathbf{F}\} + \{\mathbf{F}\}_0^{k-1} \tag{44}$$

where k is an iteration counter.

The velocities at internal points are computed by:

$$\{\mathbf{v}\}^0 = [\mathbf{A}]\ \{\mathbf{t}\}^0 - [\mathbf{C}]\ \{\dot{\mathbf{u}}\} \tag{45}$$

for the first iteration and

$$\{\mathbf{v}\}^k = [\mathbf{A}]\ \{\mathbf{t}\} - [\mathbf{C}]\ \{\dot{\mathbf{u}}\} + [\mathbf{D}]\ \{\mathbf{b}_0\}^{k-1} \tag{46}$$

for the subsequent iterations.

To ensure convergence for higher Reynolds numbers we apply a relaxation scheme:

$$\begin{aligned} \{\mathbf{a}\} &= \beta\,\{\mathbf{a}\}^k + (1-\beta)\,\{\mathbf{a}\}^{k-1} \\ \{\mathbf{b}\}_0 &= \beta\,\{\mathbf{b}\}_0^k + (1-\beta)\,\{\mathbf{b}\}_0^{k-1} \end{aligned} \tag{47}$$

where β is a relaxation coefficient ($0 < \beta \leq 1$).

7.2. *Full Newton-Raphson*

The iterative procedure for full Newton-Raphson is shown in Algorithm 2.

For the first iteration the unknowns and the velocities are computed by the same equations of the modified Newton Raphson approach, that is by Equation (43) and

```
INITIALIZATION;
solve for unknowns ${\mathbf{a}}^0$ using Equation (43);
compute velocities at internal points ${\mathbf{v}}^0$ using Equation (45);
compute body forces ${\mathbf{b}_0}^0$ using Equation (8);
compute vector ${\mathbf{F}}_0^0$ using Equation (38);
$k = 1$;
ITERATION;
while residual > Tolerance do
    compute $\left[\frac{\partial\{\mathbf{b}_0\}}{\partial\{\mathbf{a}\}}\right]^k$ and $\left[\frac{\partial\{\mathbf{b}_0\}}{\partial\{\mathbf{v}\}}\right]^k$ and update the tangent operator in Equation (49);
    solve the system of equations (49) to compute $\{\Delta\mathbf{a}\}^{k+1}$ and $\{\Delta\mathbf{v}\}^{k+1}$;
    update the unknowns by Equation (52a);
    update the velocities by Equation (52b);
    compute body forces $\{\mathbf{b}_0\}^k$ using Equation (8);
    compute vector $\{\mathbf{F}\}_0^k$ using Equation (38);
    compute the residuals $\{\mathbf{R_a}\}$ and $\{\mathbf{R_v}\}$ by Equations (48);
    $k = k + 1$
end
```

Algorithm 2: Full Newton-Raphson

Equation (45), respectively. For the subsequent iterations we introduce the following residuals:

$$\{\mathbf{R_a}\}^k = -[\mathbf{L}]\{\mathbf{a}\}^k + \{\mathbf{F}\} + \{\mathbf{F}\}_0^{k-1} \tag{48a}$$

$$\{\mathbf{R_v}\}^k = -\{\mathbf{v}\}^k + [\mathbf{A}]\{\mathbf{t}\}^k - [\mathbf{C}]\{\dot{\mathbf{u}}\} + [\mathbf{D}]\{\mathbf{b}_0\}^{k-1} \tag{48b}$$

The boundary unknown and velocity increments are computed by solving the following first-order Taylor expansions:

$$-\begin{Bmatrix}\{\mathbf{R_a}\}\\ \\ \{\mathbf{R_v}\}\end{Bmatrix}^k = \begin{bmatrix} -[\mathbf{L}]+[\mathbf{B}]\left[\frac{\partial\{\mathbf{b}_0\}}{\partial\{\mathbf{a}\}}\right] & [\mathbf{B}]\left[\frac{\partial\{\mathbf{b}_0\}}{\partial\{\mathbf{v}\}}\right] \\ \\ [\mathbf{A}]+[\mathbf{D}]\left[\frac{\partial\{\mathbf{b}_0\}}{\partial\{\mathbf{a}\}}\right] & [\mathbf{D}]\left[\frac{\partial\{\mathbf{b}_0\}}{\partial\{\mathbf{v}\}}\right]-[\mathbf{I}] \end{bmatrix}^k \begin{Bmatrix}\{\Delta\mathbf{a}\}\\ \\ \{\Delta\mathbf{v}\}\end{Bmatrix}^{k+1} \tag{49}$$

where

$$\left[\frac{\partial\{\mathbf{b}_0\}}{\partial\{\alpha\}}\right] = \begin{bmatrix} \frac{\partial\{\mathbf{b}_0\}}{\partial\{\alpha\}}(\mathbf{X}_1) & \{\mathbf{0}\} & \cdots & \{\mathbf{0}\} \\ \{\mathbf{0}\} & \frac{\partial\{\mathbf{b}_0\}}{\partial\{\alpha\}}(\mathbf{X}_2) & \cdots & \{\mathbf{0}\} \\ \vdots & \vdots & \vdots & \\ \{\mathbf{0}\} & \cdots & \cdots & \frac{\partial\{\mathbf{b}_0\}}{\partial\{\alpha\}}(\mathbf{X}_{\bar{N}}) \end{bmatrix} \tag{50}$$

$\bar{N} = N_S =$ Number of boundary nodes if $\alpha = \mathbf{a}$ (and the remaining rows are all zero) and $\bar{N} = N_{V_0} =$ Number of internal nodes if $\alpha = \mathbf{v}$.

The derivatives of $\mathbf{b}_0$ are given by:

$$\frac{\partial\{\mathbf{b}_0\}}{\partial\{\alpha\}}(\mathbf{X}_i) = \begin{bmatrix} \lambda_1(2\dot{u}_1(\mathbf{X}_i)+u_1^0(\mathbf{X}_i)) & 0 & 0 \\ \lambda_1 u_2(\mathbf{X}_i) & \lambda_2\dot{u}_1(\mathbf{X}_i) & 0 \\ \lambda_1 u_3(\mathbf{X}_i) & 0 & \lambda_3\dot{u}_1(\mathbf{X}_i) \\ \lambda_1\dot{u}_2(\mathbf{X}_i) & \lambda_2 u_1(\mathbf{X}_i) & 0 \\ 0 & \lambda_2(2\dot{u}_2(\mathbf{X}_i)+u_2^0(\mathbf{X}_i)) & 0 \\ 0 & \lambda_2 u_3(\mathbf{X}_i) & \lambda_3\dot{u}_2(\mathbf{X}_i) \\ \lambda_1\dot{u}_3(\mathbf{X}_i) & 0 & \lambda_3 u_1(\mathbf{X}_i) \\ 0 & \lambda_2\dot{u}_3(\mathbf{X}_i) & \lambda_3 u_2(\mathbf{X}_i) \\ 0 & 0 & \lambda_3(2\dot{u}_3(\mathbf{X}_i)+u_3^0(\mathbf{X}_i)) \end{bmatrix} \tag{51}$$

For $\alpha = \mathbf{a}$, $\lambda_i = 1$ if the i^{th} boundary condition component is of Neumann type, 0 else, and $\mathbf{X}_i = \mathbf{x} \in S$.

For $\alpha = \mathbf{v}$, $\lambda_i = 1$ always, and $\mathbf{X}_i = \bar{\mathbf{x}} \in V_0$ but with $\mathbf{X}_i \notin S$.

The total updated unknowns and velocities are:

$$\{\mathbf{a}\}^{k+1} = \{\mathbf{a}\}^k + \{\Delta\mathbf{a}\}^{k+1} \tag{52a}$$
$$\{\mathbf{v}\}^{k+1} = \{\mathbf{v}\}^k + \{\Delta\mathbf{v}\}^{k+1} \tag{52b}$$

8. Numerical results

The implementation of the theory is tested here on the classical example of a driven cavity which has become a standard test problem for fluid dynamics codes. However, nearly all solution published in the literature are for plane geometry.

The problem is described as follows: An incompressible fluid of uniform viscosity ($\mu = 1$) is confined within a square region of dimension $H = 1 \times 1$. The fluid velocities on the bottom, left and right are fixed at zero, while a unit uniform horizontal velocity (U) is specified at the top, which is tapered off to zero very near the corners. The Reynolds number is defined as $Re = \rho\, U\, H/\mu$. The example is tested for three different Reynolds numbers (0,100, 400) by changing the value of ρ. We compare the variation of the horizontal velocity component along a vertical line at y=0.5 with an extremely accurate plane solution published in [3] (termed reference solution).

To be able to compare with the plane solution there are two possibilities for the definition of the 3-D problem. One is to extend the discretisation into the third dimension for a distance and then truncate it. In this case errors will be introduced due to the truncation. The other possibility is to define a closed box of dimension $1 \times 1 \times 0.25$ and to apply slip boundary conditions on the two surfaces in the third direction. This approach was taken in [8] and has the advantage of not producing any truncation error. However, this approach results in a significantly higher number of degrees of freedom (d.o.f) as compared with the first approach. In the following we compare the two approaches with respect to accuracy and d.o.f.

8.1. Truncated geometry

8.1.1. Description of geometry

In this approach we extend the geometry in the x-direction and truncate it without producing a closed surface. The geometry is defined by 4 linear patches with 4 control points each as shown in Figure 3. We investigate the error introduced by the truncation by extending the truncation distance.

Using a geometry independent field approximation, the non-zero Dirichlet boundary condition along the top NURBS patch was defined using the following Knot vectors for the basis function $R^u_{ij}(s,t)$

$$\begin{aligned} \Xi &= [0,0,0.05,0.95,1,1] \\ \mathrm{H} &= [0,1] \end{aligned} \tag{53}$$

with all weights equal to 1. The parameters were specified as:

$$\dot{\mathbf{u}}^e = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{54}$$

This means that the velocity vector at the top is constant in the x-direction and tapered off to zero very near to the corners.

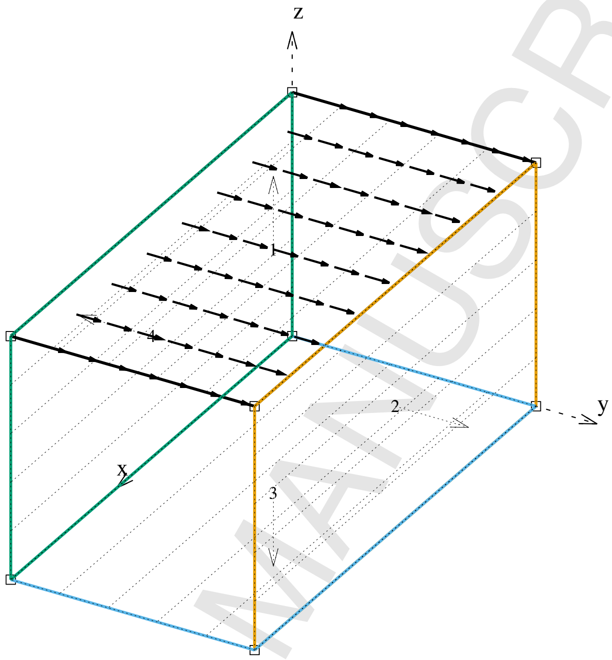

Figure 3: Forced cavity flow: definition of the geometry and boundary conditions for the truncated mesh

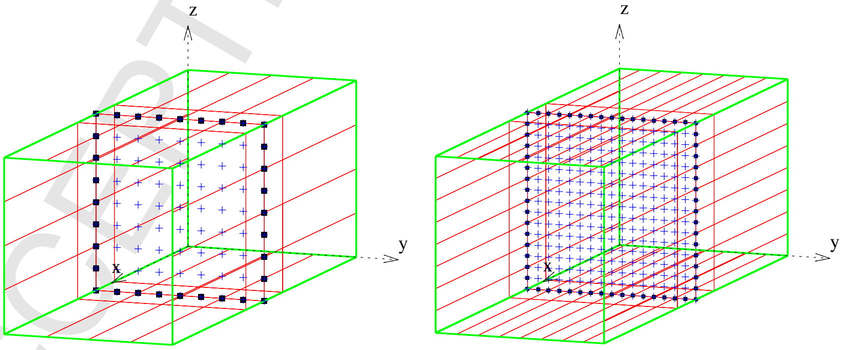

Figure 4: Approximation/refinement of the unknown: Location of collocation points (filled squares) achieved by order elevation (from linear to quadratic) and by 3 (mesh1) and 7 (mesh 2) knot insertions in the local s-direction for each patch. Also shown are the internal points (crosses) for computing the body forces. Red lines indicate the limits of the integration regions.

8.1.2. Approximation of the unknown and refinement

The approximation of the boundary unknown (in this case **t**) was achieved by inserting knots and by order elevating the basis functions for describing the geometry (from linear to quadratic) in the local s-direction. In the t-direction (in the direction of truncation) the order was reduced (from linear to constant). Two different refinements were investigated and the resulting locations of collocation points computed using Greville abscissa [22] are shown in Figure 4.

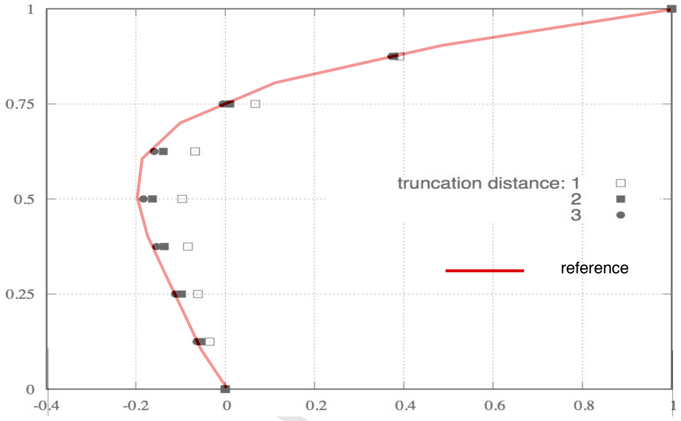

Figure 5: Convergence of the solution for Re=0 for different distances of the truncated boundary. Plotted is the magnitude of the horizontal velocity component along a vertical line through the middle.

8.1.3. Approximation of body forces inside domain

The domain for the volume integration was defined by 2 NURBS surfaces which were identical to the top and bottom patches for defining the problem geometry. The refinement of the boundary values was accompanied by an increased number of internal points as shown in Figure 4. Quadratic interpolation between the points was assumed.

8.1.4. Results for Re=0

To ascertain the errors introduced by truncation we compute the results for Re=0 first and compare with a reference plane solution [3]. The convergence of the solution is shown for mesh 1 as a function of the distance of the truncation. It can be seen that for truncating the mesh at a distance of 3 a fairly good agreement can be obtained. Next we investigate the influence of the discretisation on the results. It can be seen in Figure 6 that the results for mesh 2 (7 knot insertions) agree well with the reference curve. Mesh 1 has 96 and mesh 2 has 192 d.o.f.

8.1.5. Results for Re=100

In Figure 7 we show the results for the mesh truncated at a distance of 3 and for mesh 1 and mesh 2. A fairly good agreement can be seen with the reference solution for all meshes and either modified or full Newton-Raphson (referred to as NR in the Figure).

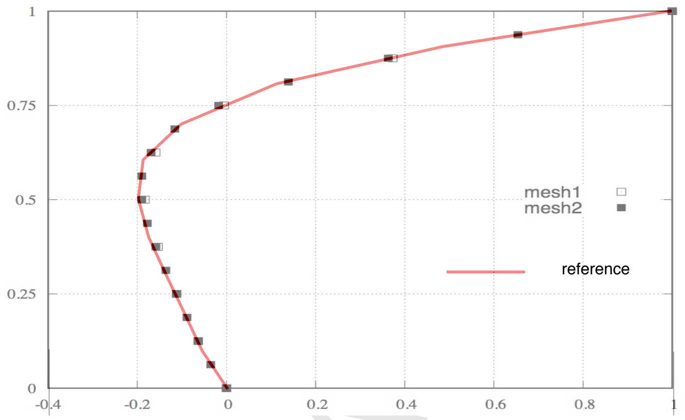

Figure 6: Convergence of the solution for Re=0 for two different discretisations.

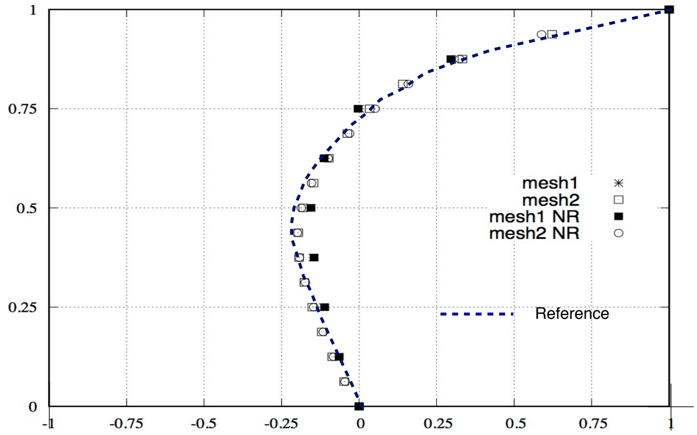

Figure 7: Vertical velocity profile for Re=100 with relaxation and full Newton-Raphson (NR)

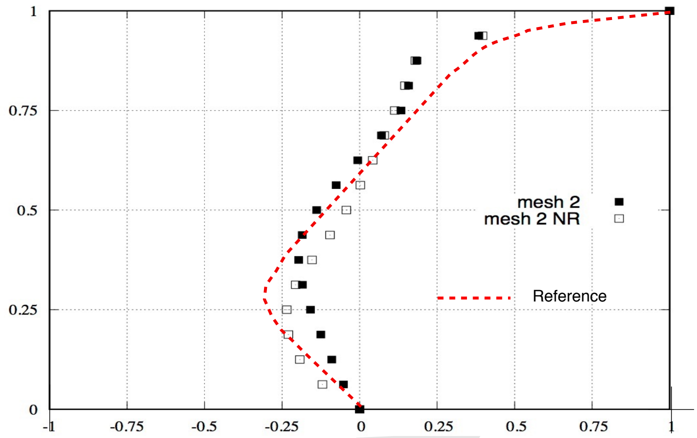

Figure 8: Vertical velocity profile for Re=400 using relaxation and full Newton-Raphson (NR)

8.1.6. *Results for Re=400*

In Figure 8 we show the results for the mesh truncated at a distance of 3 and for mesh 2. It can be seen that the results do not agree well with the reference solution, with the full Newton-Raphson being closer to it.

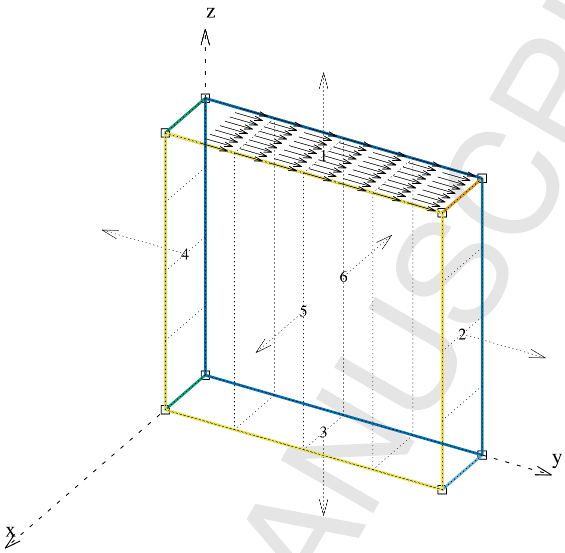

Figure 9: Geometry description for non-truncated problem.

8.2. Non-truncated geometry

In this approach the geometry is not truncated and 2 surfaces are included parallel to the y-z plane at a distance of 0.25.

8.2.1. Geometry description and boundary conditions

The geometry is described by 6 linear NURBS patches as shown in Figure 9.

The following Boundary conditions are applied:
Patch 1: Dirichlet BC with $u_x = 0, u_y = 1, u_z = 0$
Patches 2 to 4: Dirichlet BC with $u_x = 0, u_y = 0, u_z = 0$
Patches 5 and 6: Mixed BC with $u_x = 0, t_y = 0, t_z = 0$

For patch 1 the velocities were tapered off towards the corner as shown for the truncated mesh.

8.2.2. Approximation of the unknown and refinement

The approximation of the boundary unknowns was achieved by inserting knots and by order elevating the basis functions for describing the geometry (from linear to quadratic) in the *s*-direction. In the *t*-direction the basis functions for describing the geometry were used for patches 1 to 4. For patches 5 and 6 the same refinement as in *s*-direction was used for the *t*-direction. Two different refinements were investigated and the resulting locations of collocation points computed using Greville abscissa are shown in Figure 10. The meshes 1 and 2 have 486 and 1734 d.o.f respectively. Mesh 3, that has one additional knot inserted, has 2166 d.o.f.

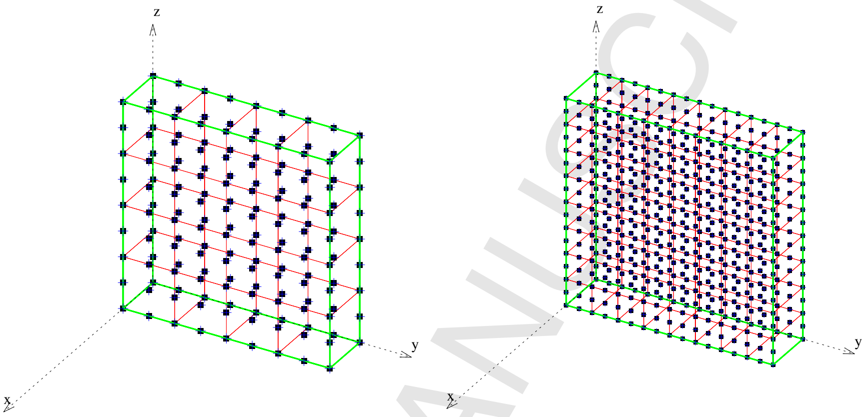

Figure 10: Location of collocation points for the two refinements. Left: mesh 1, right: mesh 2

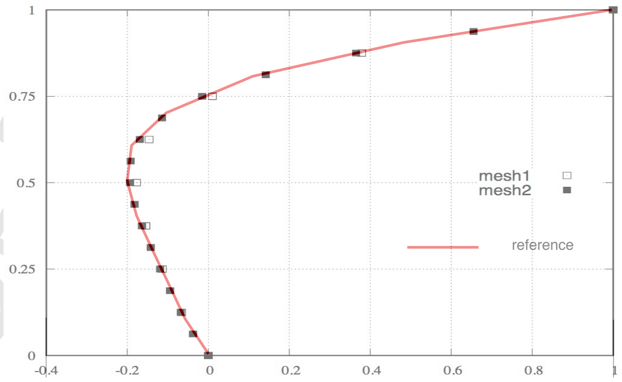

Figure 11: Vertical velocity profile for Re=0 and comparison with plane reference solution.

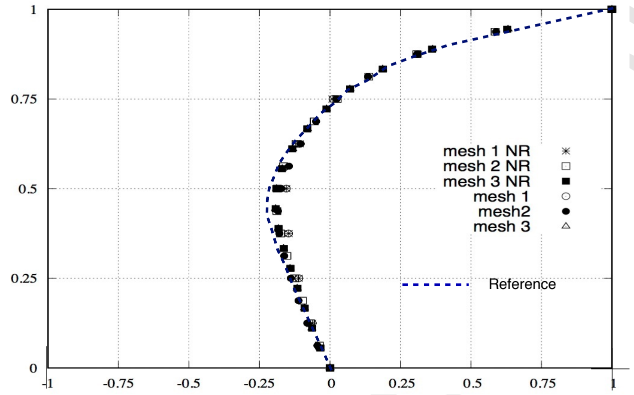

Figure 12: Vertical velocity profile for Re=100 and comparison with plane reference solution.

8.2.3. Results for Re=0

The results for Re=0 are compared with the plane reference solution in Figure 11 and good agreement can be found .

8.2.4. Results for Re=100

In Figure 12 we show the results for Re=100 for the modified and full Newton-Raphson method. All results are in good agreement.

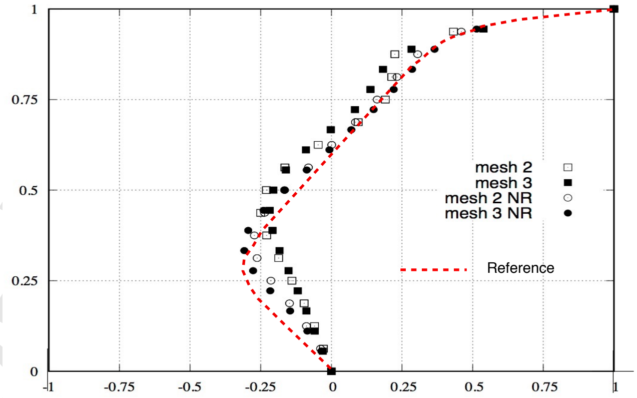

Figure 13: Vertical velocity profile for Re=400 and comparison with plane reference solution.

8.2.5. Results for Re=400

In Figure 13 we show the results for Re=400. For this Reynolds number the modified Newton Raphson method requires a high number of iterations with a very low

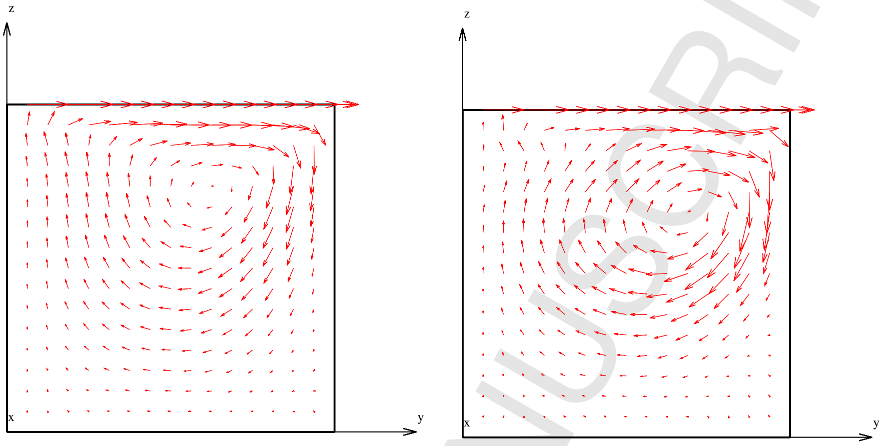

Figure 14: Velocity vectors for (left) Re=100 and (right) Re=400.

relaxation factor. As can be seen the results are not good. For meshes 2 and 3 with full Newton-Raphson convergence is achieved with few iterations and the results are much better. It can be seen that as the number of unknowns is increased the reference solution is approached. The resulting velocity vectors are shown in Figure 14.

9. Example

The example shown here is mainly designed to demonstrate that the method also works with more complex geometries and that these can be described with few parameters if NURBS functions are used. It relates to an airfoil in an infinite domain subjected to a horizontal flow of 1. The geometry of the NACA0018 airfoil was taken from the internet (www.airfoiltools.com). It was found that a NURBS function of order 2 and only 5 control points can represent this geometry with good accuracy.

9.1. Description of geometry

Figure 15 shows the description of the boundary with 2 NURBS patches with order 2 in s direction and order 1 in t direction. The mesh is truncated at a distance of 2, to simulate plane conditions.

9.2. Description of the unknown

Following the geometry independent field approximation philosophy we refine the NURBS basis functions in the s-direction by inserting 7 knots at (0.25, 0.375, 0.65, 0.75, 0.85, 0.91, 0.95). In the t direction we decrease the order from 1 to 0 (constant variation). The resulting collocation points are shown in Figure 16. The model has only 66 d.of.

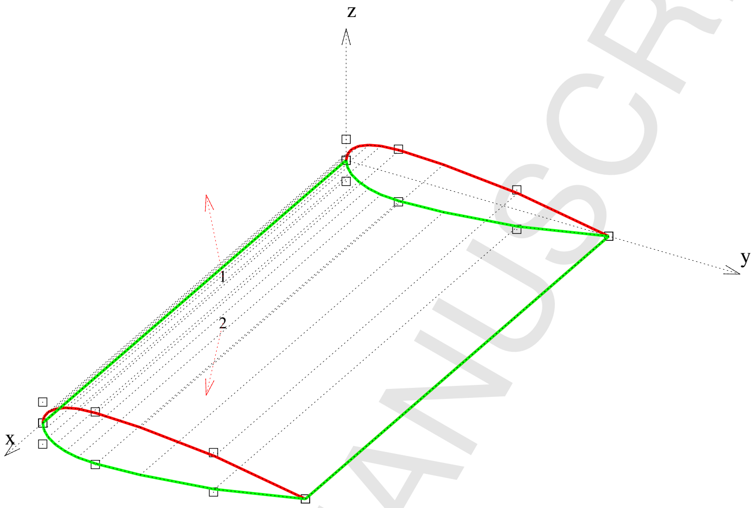

Figure 15: Example: Description of the geometry with 2 NURBS patches, showing control points.

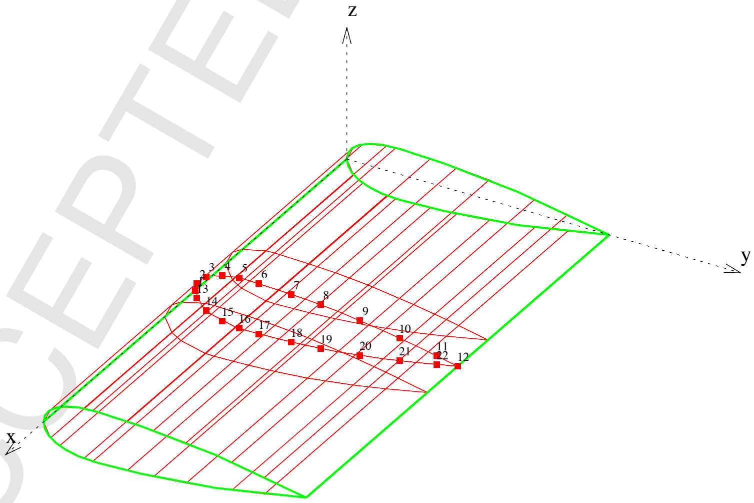

Figure 16: Example: Location of collocation points.

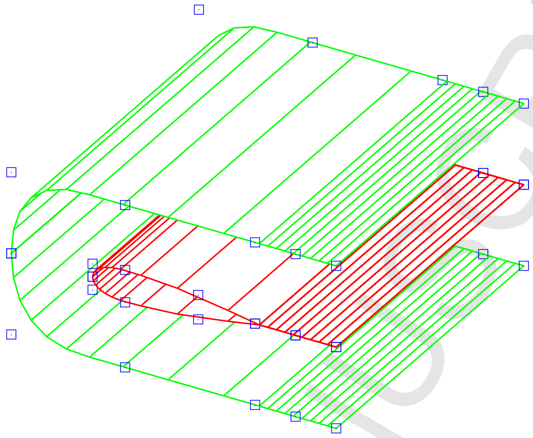

Figure 17: Example: Definition of V_0 with 2 NURBS surfaces.

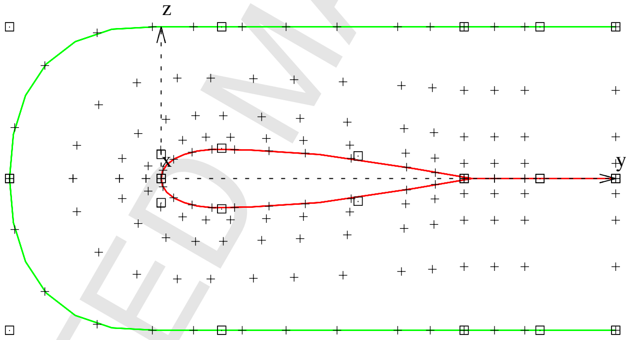

Figure 18: Example: Location of grid points in V_0.

9.3. Description of V_0

The volume near the airfoil, where we assume nonlinear effects are significant, is described by NURBS surfaces as shown in Figure 17. The location of the grid points that were used to determine the Gauss point values with a linear interpolation are shown in Figure 18.

9.4. Results

For this relatively coarse discretisation of V_0, the simulation only converges for a low Reynolds number. The results velocity vectors for Re=10 are shown in Figure 19.

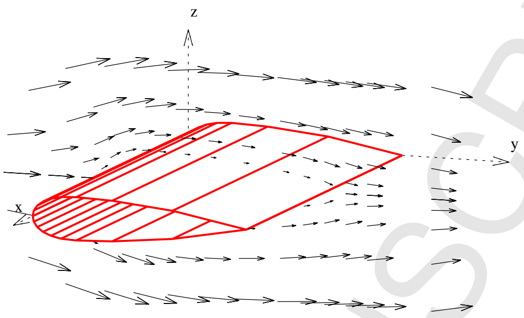

Figure 19: Example: Flow vectors for Re=10.

10. Summary and Conclusions

A novel approach to the simulation of steady incompressible viscous flow problems with the Boundary Element method was presented. For the definition of the geometry of the problem, NURBS basis functions were used. These functions were refined for the approximation of the unknowns by order elevation and knot insertion. A unique feature of our approach is that this refinement is independent of the definition of the geometry. For the evaluation of the arising volume integrals only one mapping is used and this contrasts with published methods using cells.

To test the accuracy of the method it was applied to a standard test example, where accurate plane solutions are available in the literature. Two approaches to simulate the plane conditions in 3-D were used. One where the mesh is truncated and one where the domain is closed in the third dimension. For the solution of the non-linear equations a modified Newton-Raphson (NR) scheme (where the left hand side is not changed) and a full NR method is used. Good agreement with the reference solution was found for moderate Reynolds numbers (up to 100) but it was found that for higher Reynolds numbers the modified NR requires a relaxation scheme to converge and a large number of iterations. For a Reynolds number of 400 a full NR, using the non-truncated geometry, was the only method that gave results that were converging to the reference solution. On an example it was shown that the method also works for more complex geometry. The main features of our approach are: An accurate geometry description can be attained with few parameters and no mesh. With the geometry independent field approximation results can be obtained with a small number of unknowns. The description of the volume, where non-linear effects are assumed to be significant, is simplified. Finally,it should be noted that computation times for large problems can be significant but can be reduced considerably using fast methods (see for example [23])

Appendix A. Fundamental solutions

The fundamental solutions for equations (2) for the velocity at point **x** due to point sources or forces at **y** is

$$\mathbf{U}(\mathbf{y},\mathbf{x}) = c(\mathbf{R}+\mathbf{I}) \tag{A.1}$$

with $c = \frac{1}{32\pi\mu r}$, **I** is the identity matrix and

$$\mathbf{R} = \begin{pmatrix} r_1 r_1 & r_1 r_2 & r_1 r_3 \\ r_2 r_1 & r_2 r_2 & r_2 r_3 \\ r_3 r_1 & r_3 r_2 & r_3 r_3 \end{pmatrix} \tag{A.2}$$

For the tractions acting on a boundary S:

$$\mathbf{T}(\mathbf{y},\mathbf{x}) = -c_2\ \mathbf{R}\ \cos\theta \tag{A.3}$$

where with $c_2 = \frac{3}{4\pi r^2}$. In the above r is the distance between **x** and **y**, n_i is a unit vector normal to S and

$$\begin{aligned} r_i &= \frac{1}{r}(x_i - y_i) \\ \cos\theta &= r_i n_i \end{aligned} \tag{A.4}$$

The derived fundamental solution is

$$U'_{ijk} = \frac{1}{8\pi\mu r^2}\left(-\delta_{ij} r_k + \delta_{jk} r_i + \delta_{ik} r_j - 3 r_i r_j r_k\right) \tag{A.5}$$

where δ_{ij} is the Kronecker Delta.

References

[1] G. Beer, V. Mallardo, E. Ruocco, C. Duenser, Isogeometric Boundary Element Analysis of steady incompressible viscous flow, Part 1: Plane problems, Computer Methods in Applied Mechanics and Engineering 326C (2017) 51–69.

[2] M. D. Gunzburger, Finite Element analysis for viscous incompressible flows, Academic Press, 1989.

[3] U. Ghia, K. N. Ghia, C. T. Shin, High-Re Solutions for incompressible flow using the Navier-Stokes equations and Multigrid Method, Journal of computational physics 48 (1982) 387–411.

[4] G. Dargush, P. Banerjee, Boundary Element Methods in Nonlinear Fluid Dynamics, Vol. 6 of Developments in boundary element methods, Elsevier, 1990, Ch. Advanced boundary element methods for steady incompressible thermoviscous flow.

[5] G. Dargush, P. K. Banerjee, A Boundary Element method for steady incompressible thermoviscous flow, International Journal for Numerical Methods in Engineering 31 (1991) 1605–1626.

[6] M. Aydin, R. T. Fenner, Boundary element analysis of driven cavity flow for low and moderate Reynolds numbers, International Journal for Numerical Methods in Fluids 37 (2000) 45–64.

[7] X.-W. Gao, A boundary-domain integral equation method in viscous fluid flow, International Journal for Numerical Methods in Fluids 45 (2004) 463–484.

[8] X.-W. Gao, A promising boundary element formulation for three-dimensional viscous flow, International Journal for Numerical Methods in Fluids 47 (2004) 19–43.

[9] T. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering 194 (39–41) (2005) 4135–4195.

[10] B. Marussig, J. Zechner, G. Beer, T.-P. Fries, Fast isogeometric boundary element method based on independent field approximation, Computer Methods in Applied Mechanics and Engineering 284 (2015) 458 – 488, isogeometric Analysis Special Issue.

[11] G. Beer, B.Marussig, J. Zechner, A simple approach to the numerical simulation with trimmed CAD surfaces, Computer Methods in Applied Mechanics and Engineering 285 (2015) 776–790.

[12] G. Beer, B. Marussig, J. Zechner, C. Duenser, T.-P. Fries, Isogeometric boundary element analysis with elasto-plastic inclusions. part 1: plane problems, Computer Methods in Applied Mechanics and Engineering 308 (2016) 552–570.

[13] G. Beer, V. Mallardo, E. Ruocco, B. Marussig, J. Zechner, C. Duenser, T. P. Fries, Isogeometric Boundary Element Analysis with elasto-plastic inclusions. Part 2: 3-D problems, Computer Methods in Applied Mechanics and Engineering 315 (2017) 418–433.

[14] V. Mallardo, E. Ruocco, An improved isogeometric Boundary Element Method approach in two dimensional elastostatics, CMES - Computer Modeling in Engineering and Sciences 102 (2014) 373–391.

[15] R. Simpson, S. Bordas, J. Trevelyan, T. Rabczuk, A two-dimensional isogeometric boundary element method for elastostatic analysis, Computer Methods in Applied Mechanics and Engineering 209–212 (0) (2012) 87–100.

[16] M. Scott, R. Simpson, J. Evans, S. Lipton, S. Bordas, T. Hughes, T. Sederberg, Isogeometric boundary element analysis using unstructured T-splines, Computer Methods in Applied Mechanics and Engineering 254 (0) (2013) 197 – 221.

[17] B. Marussig, G. Beer, C. Duenser, Isogeometric boundary element method for the simulation in tunneling, Applied Mechanics and Materials 553 (2014) 495–500.

[18] G. Beer, B. Marussig, J. Zechner, C. Duenser, T.-P. Fries, Boundary element analysis with trimmed NURBS and a generalized IGA approach, in: E. Oñate, J. Oliver, A. Huerta (Eds.), 11th World Congress on Computational Mechanics (WCCM XI), 2014, pp. 2445–2456.

[19] G. Beer, Mapped infinite patches for the NURBS based boundary element analysis in geomechanics, Computers and Geotechnics 66 (2015) 66–74.

[20] V. Mallardo, E. Ruocco, A NURBS boundary-only approach in elasticity, European Journal of Computational Mechanics 25 (2016) 71–90.

[21] G. Beer, Advanced numerical simulation methods - From CAD Data directly to simulation results, CRC Press/Balkema, 2015.

[22] R. W. Johnson, Higher order B-spline collocation at the Greville abscissae, Applied Numerical Mathematics 52 (1) (2005) 63 – 75.

[23] T. Takahashi, T. Matsumoto, An application of fast multipole method to isogeometric boundary element method for Laplace equation in two dimensions, Engineering Analysis with Boundary Elements 36 (12) (2012) 1766–1775.