AN AUTOMATIC ANISOTROPIC MESHING ALGORITHM ADAPTED TO THE MULTIPHYSICS SIMULATION OF A TUNGSTEN INERT GAS WELDING ARC

S. GOUNAND*

*Université Paris-Saclay, CEA, Service de Recherche en Matériaux et procédés Avancés (SRMA), 91191, Gif-sur-Yvette, France, https://orcid.org/0000-0003-3058-5735, stephane.gounand@cea.fr

DOI 10.3217/978-3-99161-089-2-036, license CC BY 4.0

ABSTRACT

In the context of the multiphysics numerical simulation of arc welding using finite elements, meshing difficulties are often encountered. Indeed, the geometry of welding tools and of the parts to be joined can be quite intricate. Also, the multiphysics model can introduce more constraints on the mesh such as maximal layer thickness for cathodic and anodic layers. Fluid mechanics may also trigger boundary layers in the physical solution. Moreover, as multiphysics frequently involve many physical unknowns per mesh node, it is important to keep the number of mesh elements under control to ensure feasible computations. This contribution presents a meshing algorithm based on the construction of a tensor field of the desired mesh size by metric interpolation, which is then fed into an automatic anisotropic mesher and a multiphysics model of the Tungsten Inert Gas (TIG) welding arc.

Keywords: Multiphysics simulation of TIG welding, Anisotropic meshing, Metric interpolation, Cast3M software

INTRODUCTION

In the context of the multiphysics numerical simulation of arc welding using finite elements, meshing difficulties are often encountered due to the following facts:

- welding tools can have complex geometric shapes, as can the parts to be joined;
- the multiphysics model may impose maximum size constraints on certain parts of the mesh, particularly at the interfaces. For example, the surfaces of the electrodes may

host a cathodic and anodic layer model of a certain thickness and, more generally, thin boundary layers can be present in the fluid model's solutions;

- the multiphysics model may involve many physical unknowns. In order to ensure a reasonable computation time, it is important to control the number of mesh elements.

Thus, we wish to devise a meshing algorithm with the following properties:

- Simplicity: the mesh construction should be automated and necessitate few user input data;
- Size control: particularly in the vicinity of the boundaries and interfaces;
- Anisotropy: in order to limit the number of mesh elements.

In order to meet these requirements, we adopt the following strategy for a typical computation:

1. Initial volume and surface meshing;
2. **Computation of a suitable anisotropic metric tensor M via interpolation**;
3. Anisotropic remeshing of the initial mesh according to the M-metric;
4. Loop to 2. if necessary;
5. Numerical simulation;
6. Post-treatment.

The focus of this contribution is step 2. The other steps are prerequisites that we do not discuss at length here. In particular, anisotropic remeshing is a rather non-standard and delicate step. The particular algorithm we use is due to Coupez and co-workers [1]. It works with simplex (triangular or tetrahedral) meshes. Another possibility is to use the open-source MMG3D library [2].

The essence of anisotropic remeshing is to optimize the mesh elements' shape according to a quality criterion depending on a given metric M. Herein, we use the following quality criterion Q for a particular simplex T:

$$Q(T, M) = c_0 \parallel T \parallel_M / h_M^n$$

where c_0 is a normalization constant, $\parallel T \parallel_M$ is the surface (volume) of the simplex T in the given metric M, h_M is the mean edge length of T and n is the space dimension. The Q criterion takes value 1 for a regular element and 0 for a degenerate flat element. See [1] for details.

In the following, we first detail the computation of the metric tensor M and then provide examples, first on simple test cases and then on more representative examples from the field of multiphysics welding simulations.

METRIC TENSOR COMPUTATION

Given a n-dimensional domain and its interior and boundary mesh, we want to compute a metric tensor field in the interior of the domain following the requirements:

1. the field variation is smooth on the domain;

2. the field is consistent with the tangential metric tensor (size) of the boundary elements, except when these are allowed to be remeshed;
3. one can prescribe the mesh size in the direction normal to the boundary if desired.

We recall that our goal here is not to describe a complete meshing algorithm. We suppose that we have an anisotropic meshing algorithm at our disposal, cite Coupez, called the mesher in the following. Here we wish to determine a metric, which is a symmetric positive definite (SPD) tensor field, typically defined on the mesh vertices, as a suitable input parameter for the mesher.

METRIC TENSOR INTERPOLATION

Our problem can be categorized as an SPD tensor interpolation problem. This problem is not trivial. Indeed, it is known [3] that just interpolating the Cartesian components of the tensor is generally not advisable: the interpolate is still a symmetric tensor but it can lose positivity and its invariants (trace, determinant) can vary wildly. More generally, this interpolation is not frame-invariant which goes against tensoriality.

Here we use the so-called Log-Euclidean framework which was initially devised in the context of MRI imaging for interpolation of SPD diffusion tensor [3]. It consists in taking the log of the SPD tensor, interpolating the Cartesian components of this symmetric, but not positive, log tensor, and then going back to SPD via exponentiation. The log of an SPD tensor is a well-defined operation which consists in:

- compute an eigen decomposition of the tensor: $M = ODO^t$;
- take the log of its positive eigenvalues: $\widehat{D}_{ii} = \log D_{ii}$;
- recompose: $\widehat{M} = O\widehat{D}O^t$.

Exponentiation of a symmetric tensor is the same process with exp instead of log.

In this decomposition, O is a unit orthogonal tensor the columns of which are the principal directions (unit vectors). D is a diagonal tensor whose elements are the eigenvalues, which are the corresponding directional size squared. $\widehat{D}$ are the log of these sizes. $\widehat{M}$ and M thus share the same principal directions.

The Log-Euclidean framework has the advantage of being simple to work with, necessitating only local eigen decompositions of tensor besides linear interpolation, but it is still not totally frame-invariant [3]. However, beyond simplicity, it has a number of desirable properties like invariance with respect to rotation and contraction-dilation and generally gives results that are close to fully frame-invariant (Riemannian) non-linear frameworks. These properties are deemed "good enough" for our meshing purposes.

Then, interpolating in the Log-Euclidean framework consists in solving $n(n+1)/2$ Laplace equations for the $n(n+1)/2$ components of the n-dimensional symmetric Log tensor. Solving Laplace equations ensures requirement 1 (tensor field variation smoothness) due to its gradient-minimization property [4].

We still have to devise suitable boundary conditions that comply with requirements 2 and 3 for these Laplace problems: this is the focus and main contribution of this work.

BOUNDARY CONDITIONS

We partition the boundary according to the following boundary conditions (BC):

- **Neumann BC (NBC)**: no size requirements are given. On this part of the boundary, the mesh can be modified (refined or coarsened) by the mesher;
- **Normal Dirichlet BC (NDBC)**: only the normal component of the metric tensor is constrained in accordance with a normal size scalar field $d_{\mathbf{n}}$ **prescribed by the user**. It reads: $\boldsymbol{n}^t M \boldsymbol{n} = d_{\boldsymbol{n}}$. This leads to a linear constraint on the tensor components at each node on this part of the boundary. This linear constraint is prescribed using the Lagrange multiplier method. On this part of the boundary, the mesh can also be modified by the mesher;
- **Tangential Dirichlet BC (TDBC)**: only two tangential components of the metric tensor are constrained in accordance with the boundary's surface metric tensor eigen decomposition. This leads to $(n-1)$ linear constraints on the tensor components at each node on this part of the boundary. On this part of the boundary, no input other than the boundary mesh is needed. We choose that this part of the boundary mesh will not be modified by the mesher;
- **Normal and Tangential Dirichlet BC (NTDBC)**: is a combination of the previous two. This leads to n constraints per node. We choose not to modify this part of the boundary.

As an alternative to NTDBC, we can also use the more stringent:

- **Full Dirichlet BC (FDBC)**: all the components of the metric tensor (principal directions and corresponding principal eigenvalues, i.e., directional size squared) are prescribed. This corresponds to prescribing all the $n(n+1)/2$ components.

IMPLEMENTATION

We have implemented the given tensor interpolation algorithm in our in-house open-source Finite Element Toolbox Cast3M [5]. Note that the Cast3M software is free to download for research and education purposes.

In implementing our algorithm for metric tensor computation, we remarked that the computed tensor field gives correctly sized elements but their principal directions are not necessarily parallel to the boundaries (see figure 1 left). This is expected since the BCs other than Full Dirichlet do not enforce this property. However, alignment is a desirable mesh property to have. We thus implement the algorithm as a two-pass process:

- Pass 1: tensor interpolation with loose BCs (all except Full DBC);
- Pass 2: using the result of pass 1, redo the tensor interpolation with Full Dirichlet BCs on the entire boundary.

Loosely speaking, the first pass builds a metric tensor that conforms to the BCs given by the user and correctly interpolates the element sizes. The second pass then enforces all the principal directions respecting the sizes obtained in pass 1.

We then obtain the result of Fig. 1 (right) where only TDBC were used in pass 1 (left). This result is deemed satisfactory.

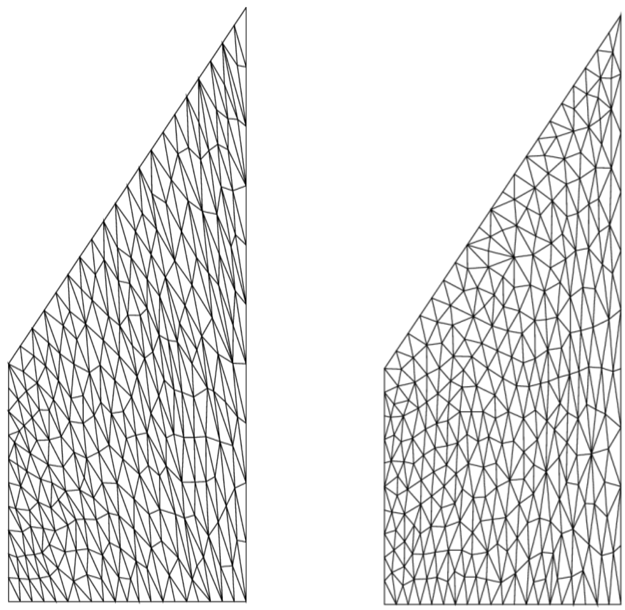


Fig. 1 Mesh of a quadrilateral non-regular object. The boundary conditions are TDBC only which means that only the meshed contour was provided by the user. Mesh after pass 1 of the tensor interpolation process (left). Mesh after pass 2 of the tensor interpolation process (right).

Note that the algorithm has the advantage of requiring very few inputs from the user: an initial mesh and an optional normal size scalar field on the boundary. Thus, complex anisotropic 3D meshes could be built from the main outline of the mesh (in the form of meshed edges) together with some well-chosen scalar normal-size fields on (part of) the edges and faces which is quite convenient.

On the minus side, the algorithm is a bit costly because we have to solve a tensorial PDE with $n(n+1)/2$ unknowns and complex from the programmer's point of view because we rely on anisotropic remeshers.

TEST CASES

We now demonstrate the use of the aforementioned anisotropic meshing algorithm in simple 2D and 3D case.

2D

This first test case illustrates the use of the algorithm for anisotropic meshing of a 1×1 square. Figure 2 illustrates the boundary conditions that were prescribed:

- the left boundary is regularly meshed with 10 elements. It is subject to a Neumann BC and will be remeshed;
- the top and right boundary are regularly with 10 elements. On these boundaries, we ask for a size in the normal direction of 0.1. They are subject to a full Dirichlet boundary conditions and will not be remeshed;
- the bottom boundary is meshed in a geometrical fashion from a size of 5.10^{-3} (left edge) to a size of 0.1 (right edge). On the first left half of the boundary, we apply a full DBC with a prescribed normal size of 5.10^{-3}. On the second right half, a tangential DBC is kept.

Figure 3 illustrates the mesh we obtain. It satisfies all prescribed boundary conditions and its quality according to the anisotropic Q criterion given before is good and above 0.62 for all elements.

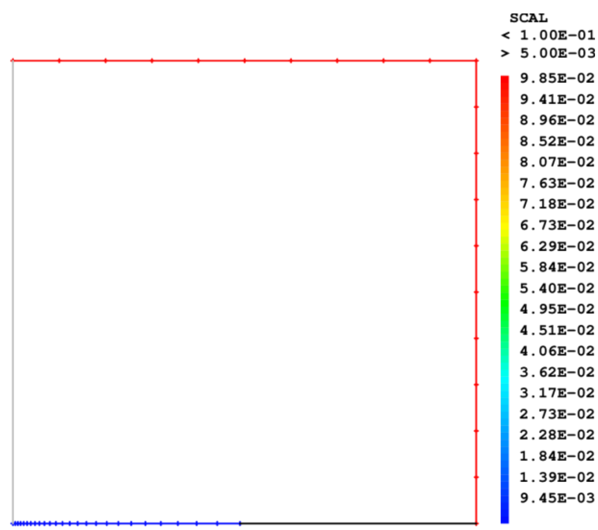


Fig. 2 Square case: boundary conditions for the tensor interpolation problem

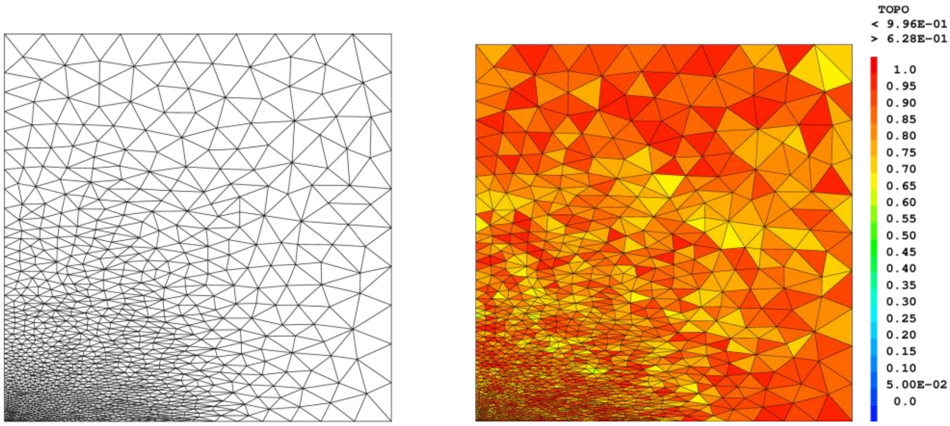


Fig. 3 Square case: mesh (2000 elements) and element quality

3D

This case is similar to the previous one and illustrates the anisotropic meshing of a $1 \times 1 \times 1$ cube. Figure 4 illustrates the boundary conditions that were applied:

- the two front faces are subjected to a Neumann BC and will be remeshed;
- the top and two rear faces are regularly 5×5 meshed. On these boundaries, we ask for a size in the normal direction of 0.2. They are subject to a full Dirichlet boundary conditions and will not be remeshed;
- the bottom face is meshed in a geometrical fashion from a size of 5.10^{-3} (bottom front point) to a size of 0.2 (other points). Points at a distance less than 0.75 from the bottom front point are subject to a full DBC with a prescribed normal size of 10^{-2} in the up direction. On the rest of the face, a tangential DBC is kept.

Figure 5 illustrates the mesh we obtain. It satisfies all prescribed boundary conditions and its quality according to the anisotropic Q criterion given before is above 0.44 for all elements. Such a value is lower than the one obtained in 2D but it is typical for 3D.

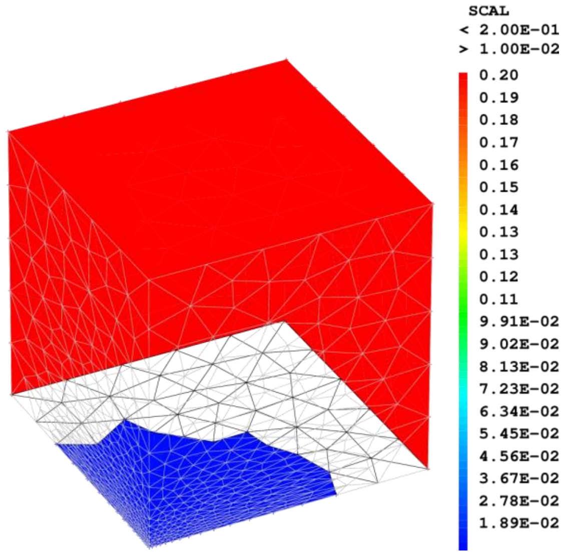


Fig. 4 Cube case: boundary conditions for the tensor interpolation problem

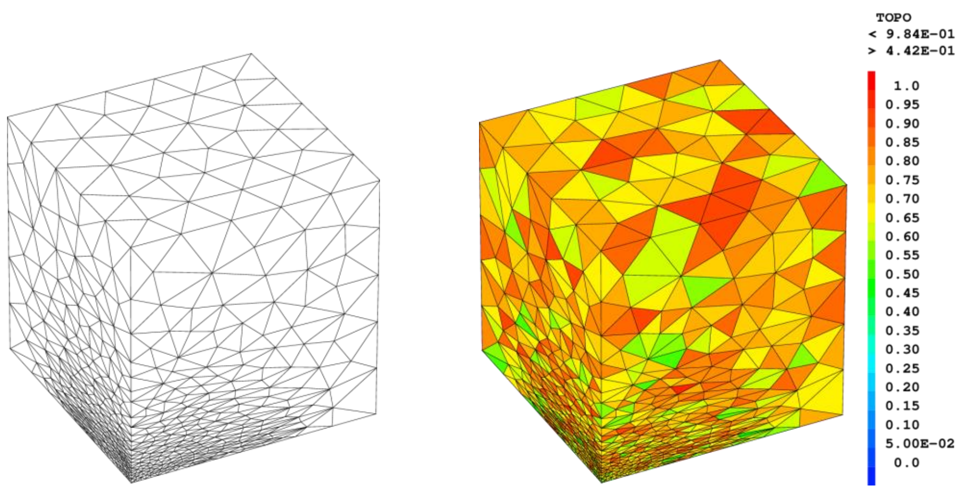


Fig. 5 Cube case: mesh (8646 elements) and element quality

APPLICATION TO TIG PLASMA COMPUTATIONS

We now demonstrate the use of the anisotropic meshing algorithm in the context of multiphysics plasma TIG simulation in typical 2D axisymmetric and 3D case.

2D AXISYMMETRIC

On figure 6, we display a typical 2D axisymmetric TIG spot situation. On the left side is the axisymmetric TIG tungsten cathode inside a hollow welding nozzle. The bottom side is the anode (work piece).

The boundary conditions are as follows:

- the right and top right boundary are meshed as shown and subject to full DBC with a desired normal size of 1 cm;
- the bottom boundary is subject to full DBC with a desired normal size of 0.4 mm;
- the cathode tip is subject to full DBC with a desired normal size of 0.1 mm;
- the part of the axis between the cathode and the anode is subject to a Neumann BC and will be remeshed;
- the rest of the boundary is given and will not be remeshed (tangential DBC).

The resulting mesh is shown in figure 7. It satisfies all the requirements while keeping a low number of elements (3500). The quality is good with minimum of the criterion similar to the simple square case.

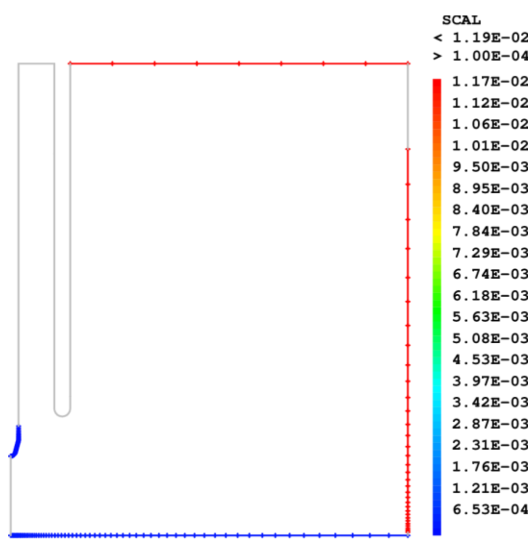


Fig. 6 Plasma TIG spot mesh: boundary conditions for the tensor interpolation problem

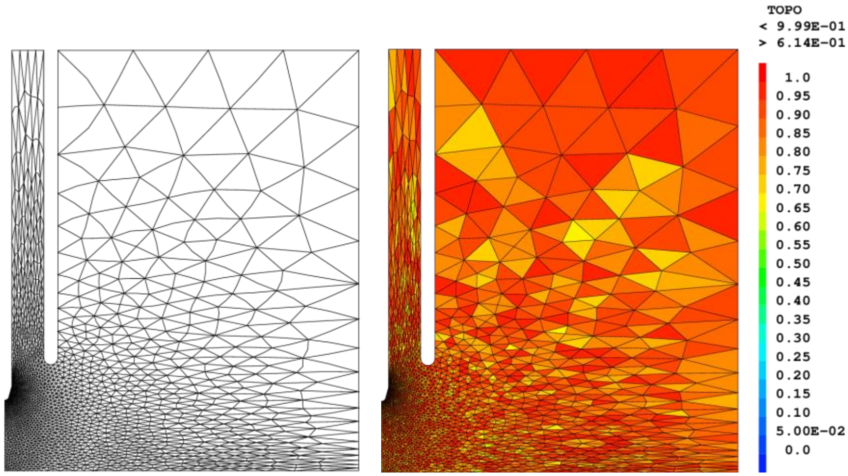
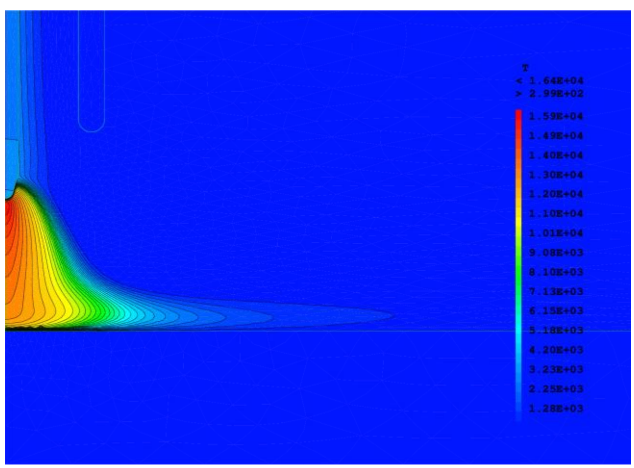


Fig. 7 Plasma mesh: 3529 elements and element quality

Next, in order to confirm that the obtained mesh is suitable for multiphysics plasma TIG Spot simulations, we carry out a typical computation. We do not describe the plasma model in any detail, referring to the literature instead [6], [7], [8]. Figure 8 shows typical results for the temperature and velocity field. More detailed post-treatment, not shown here, confirm that the field are smoothly interpolated all over the domain, even in the high-gradient regions.



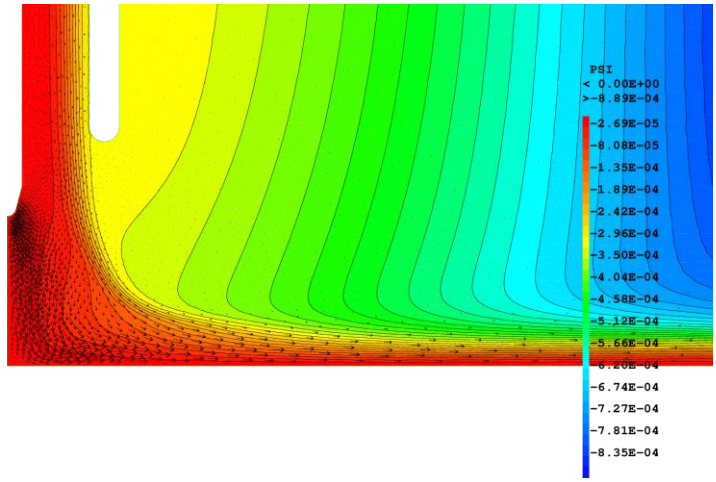


Fig. 8 Plasma computations: temperature (top, in K) and velocity (bottom, max=148 m/s)

3D MESH ENHANCEMENT

Our final example illustrates the use of our anisotropic meshing methodology for 3D case of a work piece with a chamfer. The initial mesh of 54000 elements is shown on figure 9 (left). It is an isotropic mesh designed with a commercial software. We see that the mesh is quite regular. However, near the cathode tip and the facing part of the work piece, it is still too coarse.

Thus, we use our anisotropic method to remesh the plasma part of the mesh in order to ensure the boundary conditions pictured on the right of figure 9:

- the two front faces are subjected to a Neumann BC and will be remeshed;
- the top and two rear faces are subject to a Tangential Dirichlet BC and will not be remeshed;
- on the cathode tip, a normal size of 0.1 mm is required with full Dirichlet BC while on the anode part facing the tip, a normal size of 0.4 mm is thought for. The remaining part of the bottom face is kept with a TDBC.

The resulting mesh is shown in figure 10. It satisfies the given requirements near the cathode tip while keeping the number of elements to an acceptable figure (+17000). The quality is good with a minimum of the criterion similar to the simple cube case: 0.46.

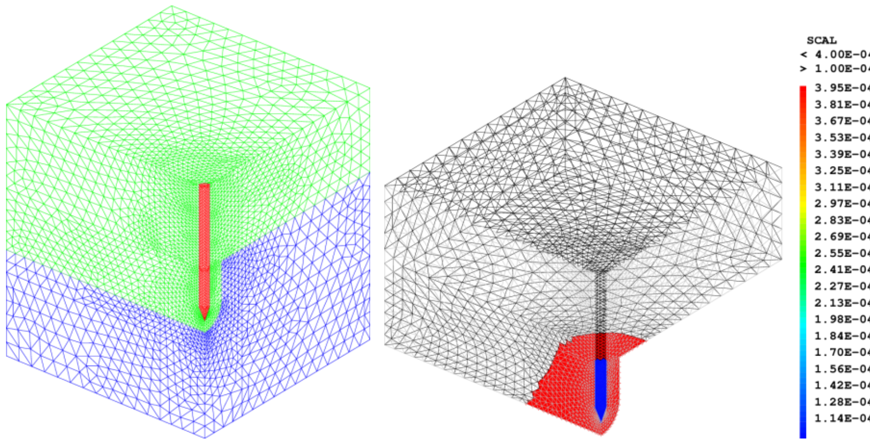


Fig. 9 Chamfer case: initial mesh (left: 54000 elements) and boundary conditions for the tensor interpolation problem. (right)

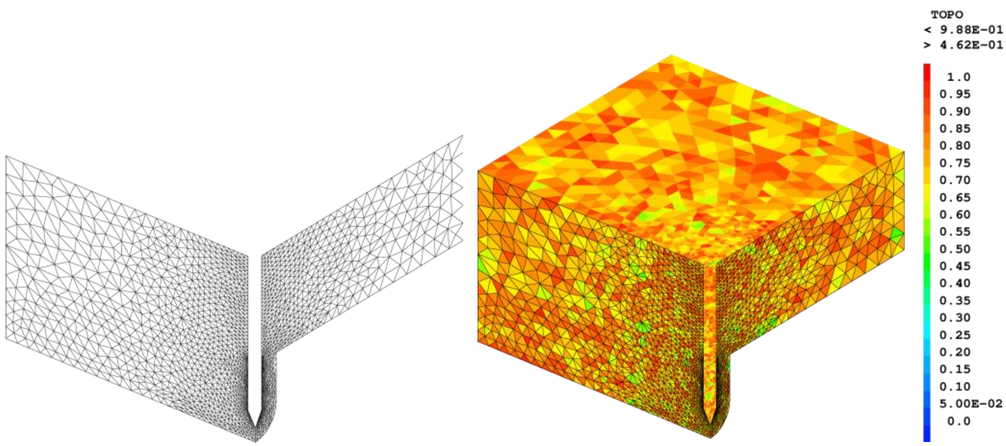


Fig. 10 Chamfer case: front faces of the refined anisotropic mesh (left: 71000 elements) and element quality

Note that, due to the boundary conditions that we used on the plasma part, the cathode and anode meshes are still conforming with the plasma mesh and need not be remeshed. TIG plasma simulations on the refined mesh are left for future work.

CONCLUSION

In this contribution, we have described a generic user-friendly anisotropic meshing tool. It is implemented in a software, Cast3M, which is open-source and free to use for educational and research purposes. It was demonstrated on some examples that the method is useful for dealing with problems in the field of the numerical multiphysics modeling of welding: anisotropy allows to keep the number of mesh elements under control, while taking into account the geometric constraints of the TIG plasma MHD model.

We have shown here that our methodology is most useful for creating a one-shot anisotropic mesh in an easy manner. However, some of the ideas concerning geometric constraints and tensor interpolation carry out to the adaptive mesh refinement case and we wish to pursue research in this direction.

ACKNOWLEDGMENTS

I thank Morgane Guilbert, International Welding Technologist, who provided some of the momentum needed for this research work.

References

[1] C. GRUAU and T. COUPEZ: '3D tetrahedral, unstructured and anisotropic mesh generation with adaptation to natural and multidomain metric', *Computer Methods in Applied Mechanics and Engineering*, Vol. 194, No. 48, pp. 4951-4976, 2005, available: https://www.sciencedirect.com/science/article/pii/S0045782505000745.

[2] C. DAPOGNY, C. DOBRZYNSKI and P. FREY: 'Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems', *Journal of Computational Physics*, Vol. 262, pp. 358-378, 2014, available: https://www.sciencedirect.com/science/article/pii/S0021999114000266.

[3] V. ARSIGNY, P. FILLARD, X. PENNEC and N. AYACHE: 'Log-euclidean metrics for fast and simple calculus on diffusion tensors', *Magnetic Resonance in Medicine*, Vol. 56, No. 2, pp. 411-421, Jun. 2006, available: http://dx.doi.org/10.1002/mrm.20965.

[4] S. GOUNAND and S. KUDRIAKOV: 'Introduction to the finite element method applied to incompressible fluid mechanics', *ENSTA ParisTech*, France, p. 132, Jan. 2016, available: https://hal.science/hal-04108255.

[5] *Cast3M Website*, 2025, available: https://www-cast3m.cea.fr/.

[6] M. TANAKA, H. TERASAKI, M. USHIO and J. J. LOWKE: 'A unified numerical modeling of stationary tungsten-inert-gas welding process', *Metallurgical and Materials Transactions A*, Vol. 33, No. 7, pp. 2043-2052, Jul. 2002, available: http://dx.doi.org/10.1007/s11661-002-0036-2.

[7] M. BROCHARD: *Modèle couplé cathode-plasma-pièce en vue de la simulation du procédé de soudage à l'arc TIG*, PhD thesis, 2009, available: http://www.theses.fr/2009AIX11006.

[8] C. NAHED, S. GOUNAND and M. MEDALE: 'A numerical study of the effects of cathode geometry on tungsten inert gas type electric arcs', *International Journal of Heat and Mass Transfer*, Vol. 182, p. 121923, 2022, available: https://www.sciencedirect.com/science/article/pii/S0017931021010280.