

# THREE DIMENSIONAL FINITE ELEMENT ANALYSIS OF ALUMINUM EXTRUSION

**B.J.E. van Rens, W.A.M. Brekelmans and F.P.T Baaijens**

**Faculty of Mechanical Engineering, Technical University of Eindhoven, The Netherlands**

## Abstract

The 3D finite element analysis of aluminum extrusion is challenging due to, amongst others, the advanced geometries, the large number of unknowns and the complex boundary conditions involved. In the present paper a mesh generator is presented that is capable of generating meshes for complex geometries while reducing the resulting number of unknowns with directional refinement. With the meshes 3D thermo-mechanical computations are performed to determine the influence of the die geometry, in particular the bearing length, on the velocity with which the aluminum exits the die. Comparison of the computations with experiments reveals that, even though the trends are correct, the influence of the bearing on the exit velocity is overestimated.

## 1 Introduction

In aluminum extrusion the quality of the produced profile is mainly determined by the shape of the die. The die shape depends on the shape of the pocket and the bearing (see Figure 1). The pocket serves to streamline the flow towards the die opening. The bearing balances the flow at the exit of the die and forms the aluminum into an intermediate shape that closely resembles the final shape. After cooling the profile obtains the final cross sectional shape. The shape of the pocket and the bearing are determined by the pocket opening and length, and the bearing opening and length, respectively. Moreover, the bearing length may vary along the circumference of the bearing opening for flow balancing purposes.

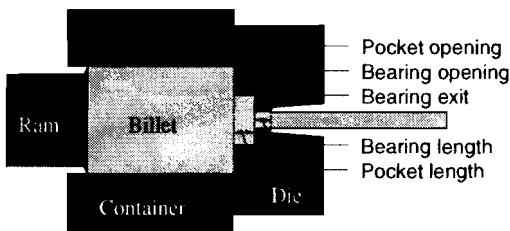


Figure 1: Extrusion

For extrusion the flow is optimally balanced if the velocity of the aluminum in extrusion direction is constant over the profile cross section, and if the velocity

perpendicular to the extrusion direction is negligible. The first condition ensures that the profile will be straight and the second that the cross section has the appropriate intermediate shape.

Due to the complexity of the shape and flow balancing issues, the dies are designed in an iterative process. In this process the die design is adjusted, based on the results of trial-pressings and a re-tested. This is an expensive and time-consuming process. Therefore, numerical simulations of flow through dies are of great interest since these can improve the design and reduce costs and time.

### 1.1 Modeling Extrusion

Extrusion is a thermo-mechanical process where plastic, incompressible deformations with low Reynolds numbers predominate. The differential equations describing the conservation of momentum, of mass, and of energy can be transformed into a (mechanical) Stokes problem (1)-(2) coupled with a (thermal) convection-diffusion problem (3).

$$-\vec{\nabla}p + \vec{\nabla} \cdot 2\eta\mathbf{D} = \vec{0} \quad (1)$$

$$\vec{\nabla} \cdot \vec{u} = 0 \quad (2)$$

$$\rho c_p \vec{u} \cdot \vec{\nabla}T - \vec{\nabla} \cdot (\lambda \vec{\nabla}T) - 2\eta\mathbf{D} : \mathbf{D} = 0, \quad (3)$$

where  $\vec{u}$  represents the velocity,  $p$  the pressure,  $\vec{\nabla}$  the gradient operator,  $\eta$  the viscosity,  $\mathbf{D} = \frac{1}{2}(\vec{\nabla}\vec{u} + (\vec{\nabla}\vec{u})^c)$  the rate of deformation tensor,  $\rho$  the density,  $c_p$  the specific heat,  $T$  the temperature and  $\lambda$  the Fourier constant.

The thermal and the mechanical problem are coupled through the viscosity of the aluminum  $\eta$  which is dependent on the strain rate  $\dot{\gamma} = \sqrt{2\mathbf{D} : \mathbf{D}}$  and the temperature  $T$ . This dependency is generally expressed using a Zener-Hollomon parameter  $Z$  [1]:

$$\eta = (CZ^m)/(\dot{\gamma}) \quad \text{with} \quad Z = \dot{\gamma} \exp\left(\frac{Q}{RT}\right) \quad (4)$$

with  $C$  and  $m$  material parameters,  $Q$  the specific heat, and  $R$  the gas constant.

The Stokes problem is discretized using the MINI-element [2, 3]. The MINI-element is a tetrahedral element which can be characterized by a linear pressure field combined with a linear velocity field that is enriched with a bubble function. For the discretization of the temperature problem a linear tetrahedral element will be

used. The system of coupled mechanical and thermal systems of equations that result after discretization are solved decoupled using iterative solvers.

Many methods exist to compute the flow of aluminum through dies. For instance, one can consider the profile to be a collection of simple geometries and determine analytical solutions for the flow field in the simple geometries [4], while possibly taking into account material interaction between the geometry parts. The solutions for the simple geometries could also be obtained from 2D finite element calculations [5]. However, the proposed methods fail for complex profiles because the interaction between the various geometry parts becomes too complex. This deficiency necessitates the use of 3D finite element calculations but these have been restricted to relatively simple geometries such as tubes [6] and squares [7].

Many challenges arise from the 3D finite element of complex profiles. First, the complex die geometries are difficult to mesh. Second, the complexity of the die geometries necessitates the use of fine meshes with a large number of elements, which implies computational problems due to the large number unknowns. Third, the boundary conditions in the die are complex and it is difficult to assess how these should be modeled.

## 2 Mesh Generation

The meshing can be simplified by using an expansion technique. Within the pocket and the bearing, the cross section perpendicular to the extrusion direction does not change. Therefore, after a 2D mesh has been generated for these cross sections, the 3D volume mesh can be obtained by expansion in extrusion direction. This way the 3D meshing of a complex volume is reduced to the 2D meshing of complex surfaces.

Furthermore, the cross section area of the aluminum is reduced to about one percent from billet to bearing which causes the length scales near the ram to be much larger than those close to the die. Therefore it is desirable to have a fine mesh in the billet near the pocket and a coarse mesh near the ram. This is achieved by using an unstructured mesh generator.

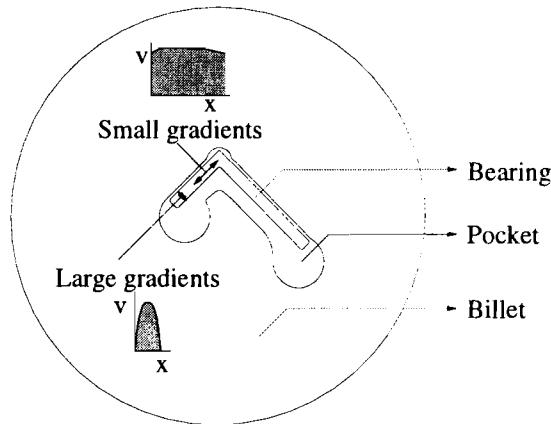


Figure 2: Relevant perpendicular cross sections for simple profile

### 2.1 Generating the 2D mesh

In extrusion the flow field is characterized by varying length scales. The profiles are often thin walled which implies that the length scale along the profile wall is much larger than perpendicular to the wall. This property of the flow field is exploited in the 2D mesh by using directional refinement.

Several methods exist to generate directionally refined meshes for the domains in Figure 2. These methods can be split in mapped and unstructured procedures. Here an unstructured mesh generator is applied because it is better suitable for complex domains.

Unstructured mesh generators can globally be split up into two classes, Delaunay triangulation type generators [8] and paving or plastering generators [9]. Delaunay type generators construct grids between just the boundary nodes of the domain. Since this often generates very low-quality elements, points are added to the interior of the domain in order to meet quality criteria for the mesh. Gobea et al. [10] introduced a version of such a generator that rendered meshes with a directional refinement, by defining different quality criteria for different directions. However, for complex geometries the direction of the refinement varies throughout the domain, which causes complications.

Therefore, here a paving generator [9] is used. Paving consists of proceeding along the boundary between the gridded and the ungridded part of the domain, adding one element layer at a time. An advantage of paving over a Delaunay type generator [8] is that the thickness of each layer can be controlled to render meshes that are, for instance, more refined in a direction perpendicular to the boundary than parallel. This makes it possible to deal with a varying direction of refinement in a straightforward manner.

#### 2.1.1 The Paving Algorithm

The paving algorithm consists of the following steps (see also Figure 3(a)-(f)):

- (a) Process the input data
- (-) Repeat
  - (b) Generate triangles using existing nodes on the current boundary
  - (c) Add a layer of quadrilaterals that are split into triangles
  - (d) Merge nodes on the new boundary that are close
  - (e) Until entire domain is meshed
  - (f) Smooth the mesh to improve the element geometry

#### 2.1.2 Paving the Bearing, Pocket and Billet Surface

To reduce meshing effort and achieve connectivity between the volumes of the mesh after expansion, the pocket opening surface is constructed out of the bearing opening

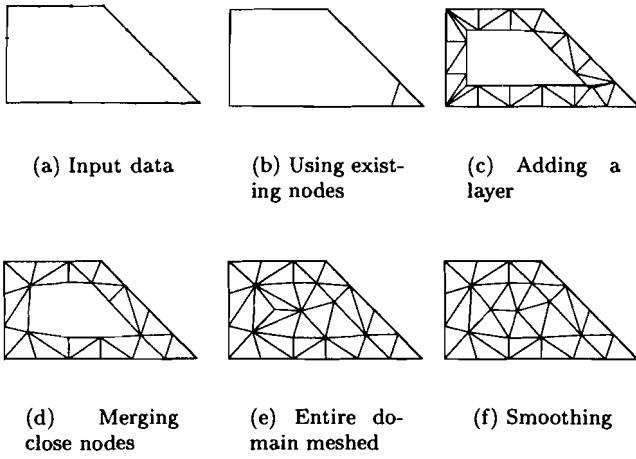


Figure 3: The principle of paving

surface and the pocket remainder surface. Similarly the billet surface is constructed out of the pocket opening surface and the billet remainder surface. The (compatible) meshes filling the three areas distinguished in Figure 2 are depicted in Figure 4.

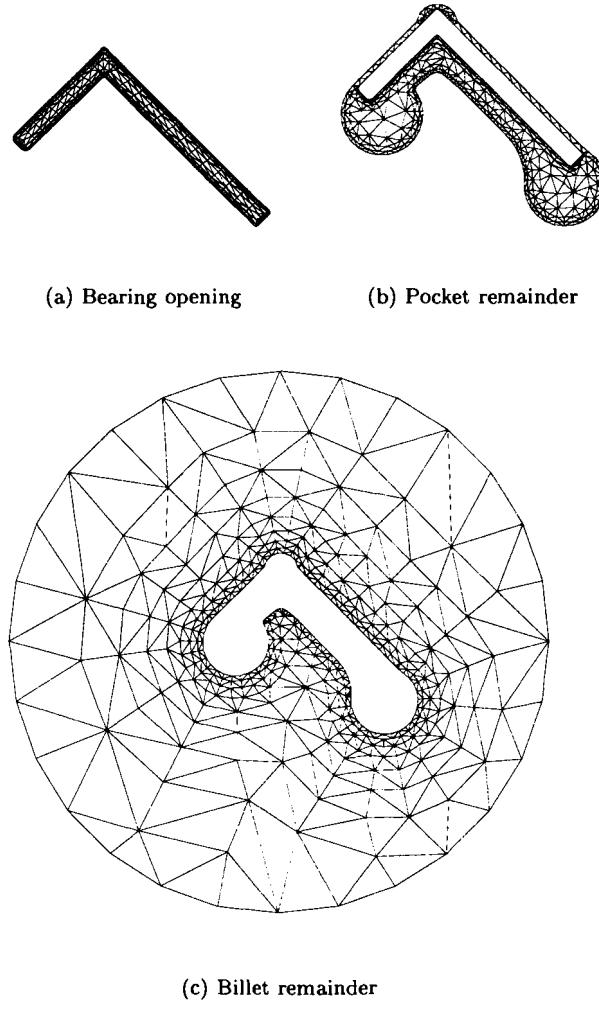


Figure 4: Meshes of billet, pocket opening and die opening

### 2.1.3 Example of a Complex Profile

An example of a more complex profile can be seen in Figure 5. The profile that is meshed is used in electronic devices for cooling purposes.

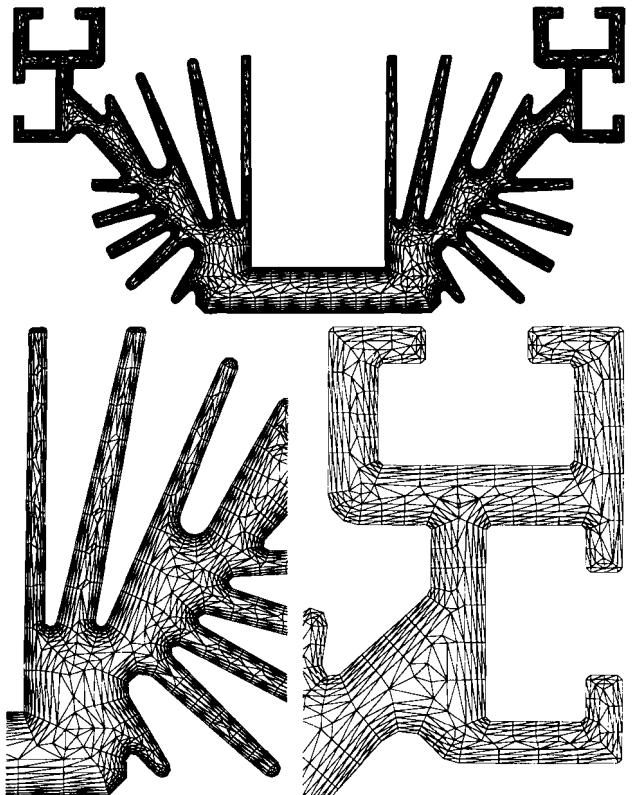


Figure 5: Directionally refined mesh of a complex profile

## 2.2 Expanding the 2D Mesh to a 3D mesh

The 2D surface meshes of the bearing and the pocket in Figure 4 are expanded in the extrusion direction to obtain 3D volumes of the pocket, and the bearing. The expansion is performed in two steps. First prisms are created from the triangles. Then the prisms are subdivided into tetrahedra.

### 2.2.1 Generating Prisms

The prisms are created by copying the triangles from the surface meshes and translating them over a prescribed distance to render a new layer of triangles. By varying the translation distance of each layer longitudinal refinement can be achieved. The triangles from the two layers are combined to render prisms (see Figure 6). This is repeated until the complete volume is filled.

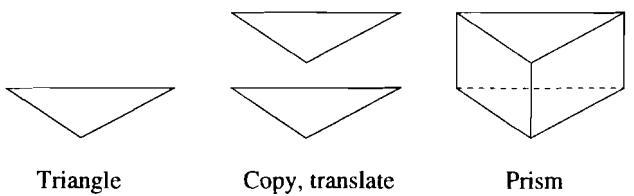


Figure 6: Generating prisms

## 2.2.2 Subdividing Prisms into Tetrahedra

Next the prisms are split into tetrahedra. This split has to be performed such that the diagonals introduced on the rectangular faces of the prisms match for adjacent prisms. The split of a prism into three tetrahedra can only be performed if the diagonals of these faces are not all oriented in the same way (see also Figure 7).

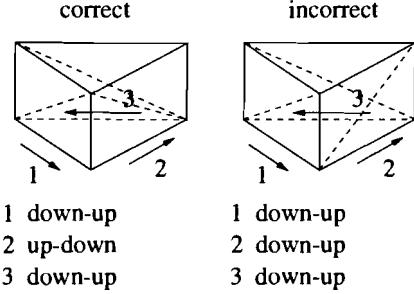


Figure 7: Diagonal definition on prisms

An algorithm is used to set the orientation of the diagonals in a pattern that meets the requirements. First, the diagonals of the faces that lie on the boundary are set. Then a loop over the prisms is performed and for the prisms with the orientation of the diagonals defined on one or two faces an additional face diagonal is selected. The orientation is chosen such that the diagonals of this prism and that of the prism that shares the face do not violate the orientation criterion. If it is not possible to satisfy the orientation criterion for both prisms (if both prisms already have two faces set and the criterion requires different orientations on their mutual face) the orientation of the diagonal is set such that the current prism is adequate. This loop is repeated until the diagonals of all the faces are set. After this, those prisms which do not meet the orientation criterion are removed with an algorithm similar to that of Löhner [11]. This algorithm locates these prisms and changes the orientation of one of the diagonals. To avoid infinite loops the orientation of each diagonal can be altered only once.

Given the orientation of the diagonals the prism can be subdivided into three tetrahedra (see Figure 8).

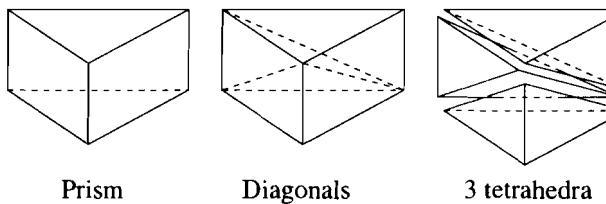


Figure 8: Splitting a prism into 3 tetrahedra

## 2.3 Stretching the Bearing

In the mesh generated in the previous section the bearing has a constant length however, in reality the bearing length may vary over the bearing boundary. The varying bearing length will influence the mesh in both the bearing

and the outflow volume. Here the adaptation of the bearing volume is discussed but the same method is applied to the free flow volume.

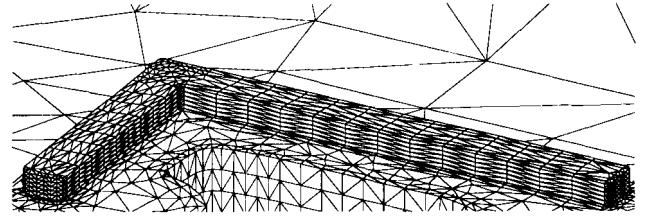


Figure 9: Bearing exit surface for constant bearing length

The nodes on the boundary of the bearing exit surface are translated over the distance between the current (constant) bearing length and the actual bearing length. The positions of the other nodes in the surface are obtained by interpolation between the nodes on the boundary (see Figure 10). It is beyond the scope of this paper to discuss the interpolation method, therefore the interested reader is referred to Tezduyar et al. [12] and Johnson and Tezduyar [13].

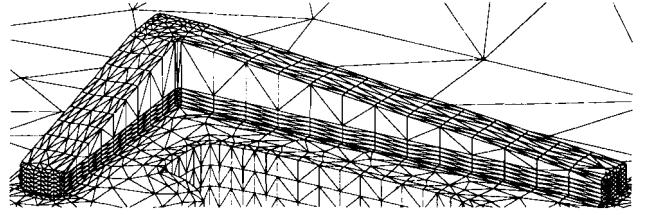


Figure 10: Bearing exit surface for varying bearing length

The bearing mesh is smoothed by repositioning the nodes between the the bearing opening surface and the bearing exit surface. The new positions of the nodes are obtained by interpolation between the two surfaces. The resulting mesh is depicted in Figure 11.

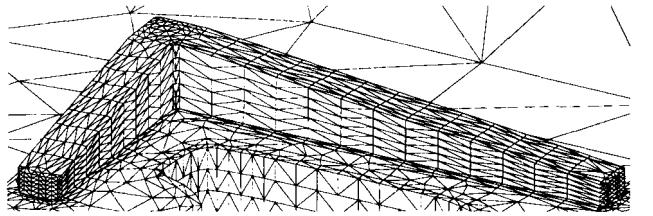


Figure 11: Smoothed bearing mesh

## 2.4 The Resulting 3D Mesh

The 3D mesh in Figure 12 is constructed out of the 3D meshes of the free flow, the bearing, the pocket, and the billet volumes. The free flow, the bearing, and the pocket meshes are obtained by expanding the paved meshes in extrusion direction. Because the flow field has the shortest length scales in the bearing, the thickness of the layers is selected to be the smallest there. After expansion the bearing mesh is stretched to the actual bearing length.

Clearly, the free flow mesh has to be stretched as well to ensure connectivity of the mesh on the bearing exit surface. The billet mesh is obtained using an unstructured mesh generator which is capable of generating prisms with high aspect ratio's near the pocket opening and coarsening the mesh away from the pocket opening.

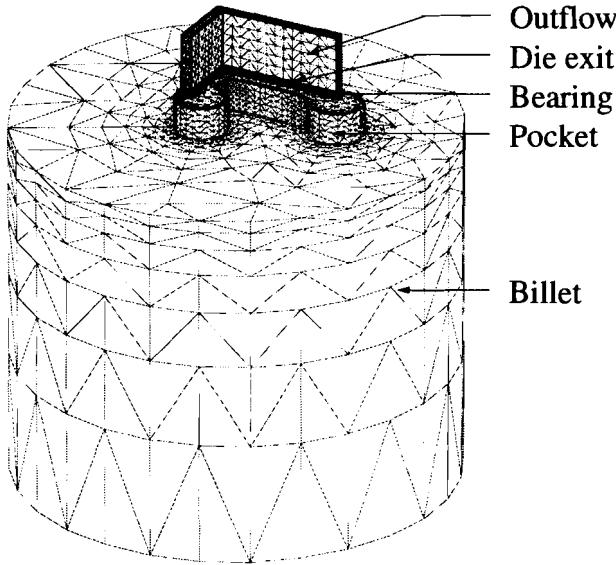


Figure 12: The surface meshes used to generate a 3D mesh for FE analysis of extrusion

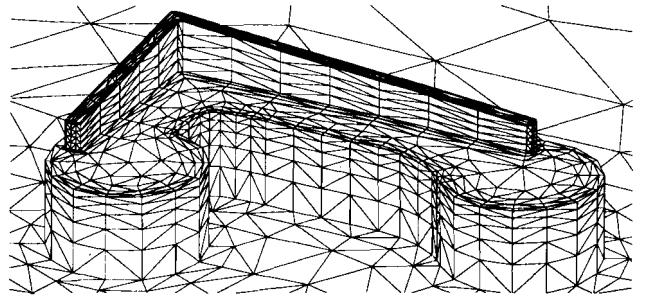
### 3 Flow Calculations

A 3D mesh similar to the mesh in Figure 12 is used to perform coupled thermo-mechanical computations. Three different bearing configurations are modeled (see also Figure 13). For the first (a) the bearing is longer in the center of the profile, for the second (b) the bearing length is constant, and for the third (c) the bearing is shorter in the center.

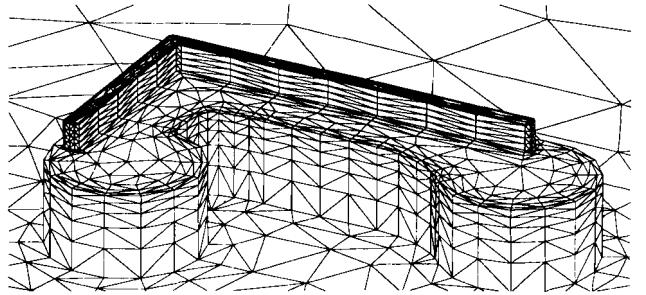
For each of the bearing configurations the velocity field and the temperature field are determined. The results of the computations are used to investigate the influence of the bearing length on the temperature and the velocity field. Additionally, the exit velocity distributions are compared to physical experiments that were performed on an industrial extrusion press with a die that contained a bearing configuration resembling the one depicted in Figure 13(a).

#### 3.1 Boundary Conditions

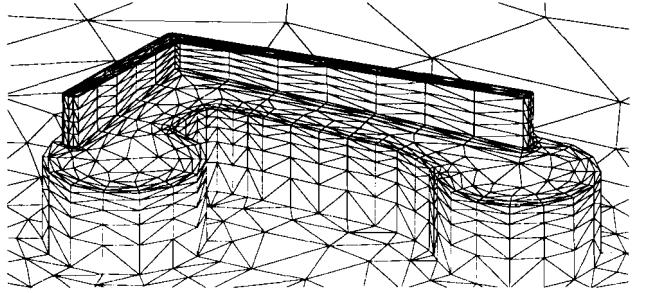
For the boundary conditions in the mechanical problem a constant velocity is defined on the inflow surface (i.e. the ram surface). Also it was assumed that the aluminum would stick to the container and the die surface while at the outflow boundaries the surface stress was assumed to be zero. For the thermal problem a constant temperature was prescribed on the inflow surface while the other surfaces were modeled to be thermally insulated.



(a) Longer in center



(b) Constant length



(c) Shorter in center

Figure 13: Bearing configuration meshes (mesh of the outflow is omitted for clarity)

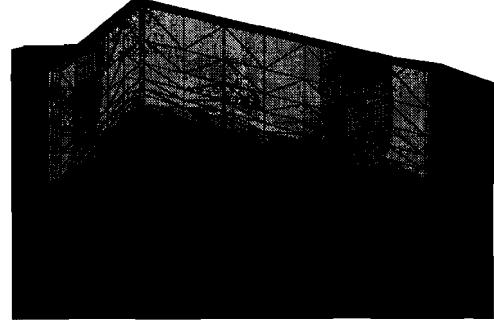
#### 3.2 Computing the Velocity and Temperature Field

The mesh contained 60.000 d.o.f. for the mechanical and 15.000 d.o.f for the thermal problem. For both problems an iterative BiCGStab [14] solver was employed. On average 4000 BiCGStab iterations were needed for the mechanical problem. Around 200 BiCGStab iterations were needed for the thermal problem.

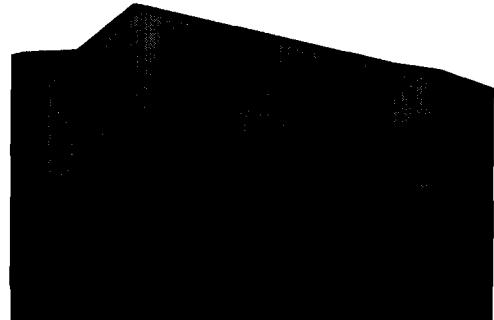
The thermal and mechanical problem were solved decoupled in an iterative loop. In this loop first the viscosity was determined in every point of the domain for a given velocity and temperature field. Then the Stokes problem was solved with this viscosity distribution. Next, the resulting velocity field was used to determine the viscous dissipation in every point of the domain. Last, the viscous dissipation was used to solve the thermal convection-diffusion equation. On average ten thermo-mechanical

loops were needed to converge to a stable velocity and temperature field.

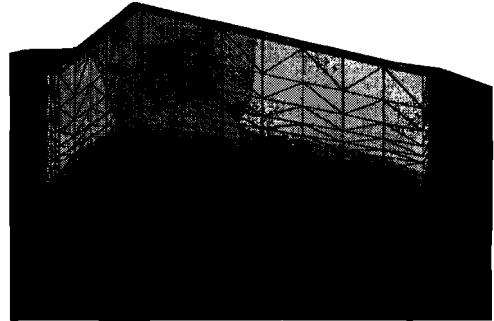
### 3.3 Results



(a) Longer in center



(b) Constant length



(c) Shorter in center

Figure 14: Temperature contours for different bearing configurations

In Figure 14 the temperatures are plotted for the different bearing configurations. In Figure 15 the exit velocities in the extrusion direction are plotted inside the bearing.

In Figure 14 it can be observed that the bearing length has a negligible influence on the temperature in the profile ( $T_{\max} - T_{\min} \approx 150^{\circ}\text{C}$ ). It should be noted that the average bearing length in configuration (a) and (c) is

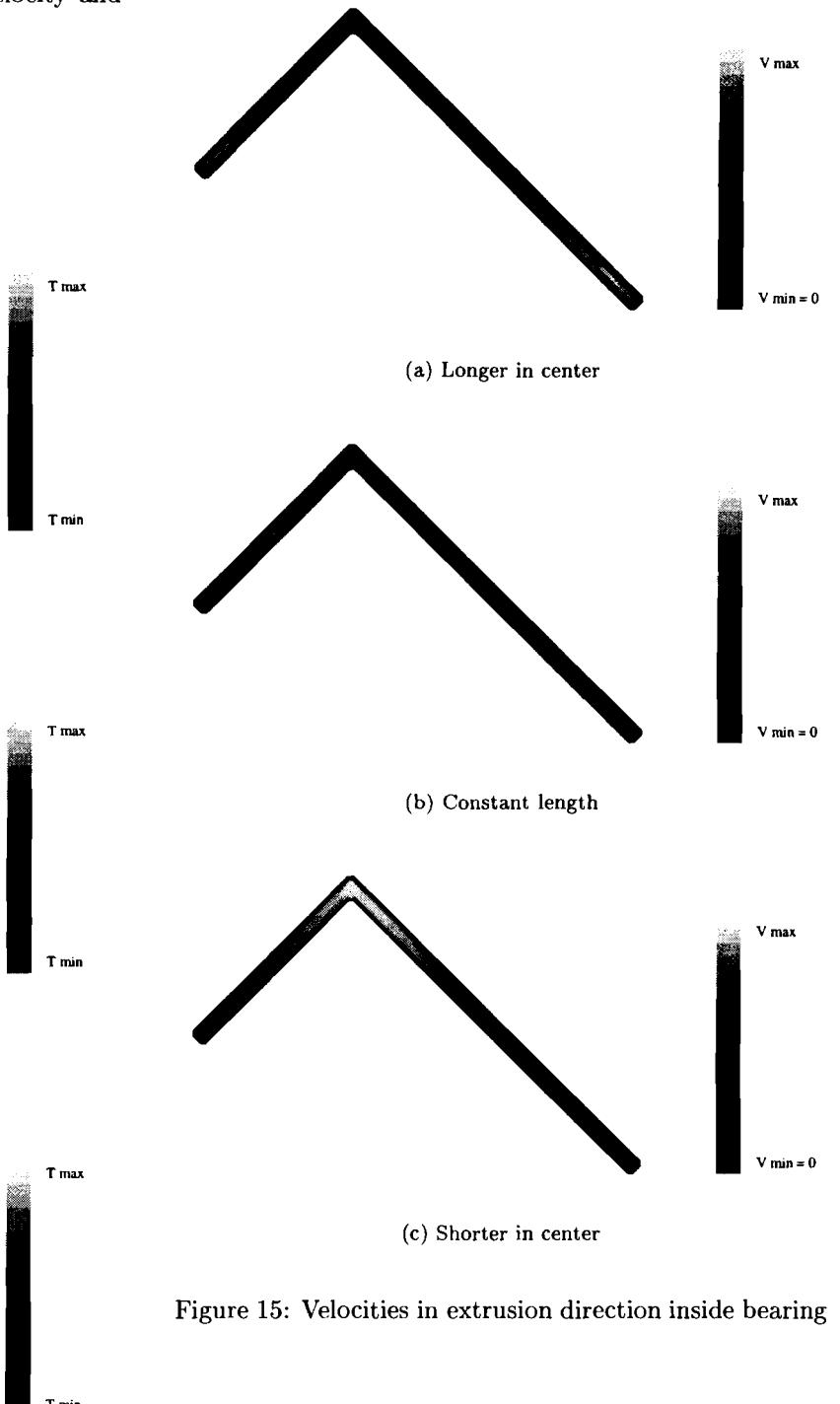


Figure 15: Velocities in extrusion direction inside bearing

larger than that in (b), which causes the overall temperature to be slightly higher in (a) and (c).

From Figure 15 it can be concluded that the bearing length influences the exit velocity distribution considerably. The velocities at the boundary of the profiles are equal (zero) for every bearing configuration due to the stick assumption, however in the middle of the profile the velocity distribution is influenced by the length of the bearing. For the configurations (b) and (c) the exit velocities are higher in the center of the profile than at the ends. For configuration (a) the velocities are higher at the ends. These results concur with the observed trends in practice.

The influence of the bearing length on the exit velocities is larger than what was observed in the exper-

iment. In the experiment, where a die with a bearing length variation as indicated in Figure 13(a) was used, the aluminum exited the die in a straight line which indicates a constant exit velocity. However, in the simulations the exit velocity is higher at the ends of the profile than in the center for configuration(a). Since the opposite is true for configuration(b) this suggests that the optimal bearing configuration is in between (a) and (b). Actually, since configuration (b) gives a more uniform exit velocity profile this would indicate that the optimal bearing configuration is closer to (b) than (a). This is does not in agreement with the experiments. This discrepancy between the experiments and the computations is most likely caused by the assumption of pure stick that was made when modeling the aluminum die interaction.

## 4 Conclusion

With the combination of the mesh generator, the MINI-element and the iterative solver it is possible to analyze complex profiles using the finite element method. However, several issues remain.

One issue is the adequate modeling of the aluminum-die interaction in the bearing area. The current assumption that the aluminum sticks to the bearing renders unsatisfactory results. Therefore a more elaborate model is necessary that incorporates this interaction adequately. One possibility that is currently being investigated is the modeling of Coulomb friction.

Another issue is associated with the large number of iterations in the BiCGStab solver to obtain the velocity field. These iterations are caused by ill-conditioning of the matrix for which the matrix-vector equation is being solved. The ill-conditioning is caused by the high aspect ratio's of the tetrahedra in the mesh. Additionally, the viscosity can range over 6 to 8 decades from very high viscosity near the ram to a very low viscosity in the shear bands near the bearing. These large differences in viscosity throughout the domain worsen the conditioning of the matrix.

The ill-conditioning of the matrix is not a problem for direct solvers. However, for direct solvers it is impossible to solve very large systems due to memory limitations. This can be solved by using a domain decomposition technique because this makes it possible to apply a direct solver on the smaller systems associated with each of the sub-domains. Currently such an approach is being implemented.

## References

- [1] R.N. Wright, G.G. Lea, and F.F. Kraft. Constitutive equations and flow stress characterization concepts for aluminum extrusion. In *Extrusion Technology Vol. I*, pages 259–268, 1996.
- [2] R. Pierre. Optimal selection of the bubble function in the stabilization of the P1-P1 element for the Stokes problem. *SIAM J. Numer. Analysis*, 32(4): 1210–1224, 1995.
- [3] L.P. Franca and C. Farhat. Bubble functions prompt unusual stabilized finite element methods. *Comput. Methods Appl. Mech. Engrg.*, 123:299–308, 1995.
- [4] S. Støren. Theory of extrusion-advances and challenges. *Int. J. Mech. Sci.*, 35:1007–1020, 1993.
- [5] H.G. Mooi, A.J. den Bakker, K.E. Nilsen, and J. Huétink. Simulation of aluminium extrusion based on a finite element method. In *Extrusion Technology 96 II*, pages 67–73, 1996.
- [6] H.G. Mooi. *Finite Element Simulations of Aluminum Extrusion*. PhD thesis, University of Twente, the Netherlands, 1996.
- [7] V. Legat and J.M. Marchal. Die design: An implicit formulation for the inverse problem. *Int. J. Numer. Methods Fluids*, 16:29–42, 1993.
- [8] Y. Zheng, R.W. Lewis, and D.T. Gethin. Three-dimensional unstructured mesh generation : Part 1. Fundamental aspects of triangulation and points creation. *Comput. Methods Appl. Mech. Engrg.*, 134: 249–268, 1996.
- [9] T.D. Blacker and M.B. Stephenson. Paving : A new approach to automated quadrilateral mesh generation. *Int. J. Numer. Methods Engrg.*, 32:811–847, 1991.
- [10] J.F. Gobéau, T. Coupez, B. Vergnes, and J.F.A. Agassant. Computation of profile dies for thermoplastic polymers using anisotropic meshing. In *Simulation of Materials Processing: Theory, Methods and Applications*, pages 59–66, 1995.
- [11] R. Löhner. Progress in grid generation via the advancing front technique. *Engrg. with Computers*, 12:186–210, 1996.
- [12] T.E. Tezduyar, M. Behr, S. Mittal, and J. Liou. A new strategy for finite element computations involving moving boundaries and interfaces. *Comput. Methods Appl. Mech. Engrg.*, 94:353–371, 1992.
- [13] A.A. Johnson and T.E. Tezduyar. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Comput. Methods Appl. Mech. Engrg.*, 119: 73–94, 1994.
- [14] Y. Saad. *Iterative Methods for Sparse Linear Systems*. PWS Publishing company, Boston, USA, 1996.