93

A Domain Decomposition Solver for Three-Dimensional Steady Free Surface Flows

Einar M. Rønquist

1 Introduction

The prediction of the free surface shape is an important aspect of many materials processing applications. It poses great challenges in terms of the design of efficient solution algorithms for the governing equations, in particular, for three-dimensional problems. The governing equations are highly coupled, and there is a great need for fast, robust and memory efficient methods. Even though recent improvements in solution strategies have made three-dimensional simulations tractable [HP91, CMBBC91, EPW92, RE96], there is still room for significant improvement.

In [Røn96], a domain decomposition method was proposed for the solution of the steady, incompressible Navier-Stokes equations, but in the context of imposing velocity boundary conditions. This paper represents an extension of the method in [Røn96] to general stress boundary conditions. In particular, we consider steady free surface problems in which surface tension plays an important role. The segregated solution approach used in [HR94] for two-dimensional problems is here extended to the three-dimensional case. The update of the geometry is decoupled from the solution of the steady Navier-Stokes equations; this paper emphasizes the latter step.

In terms of providing insight into why the method in [Røn96] works as well as it does, we refer to [Cas96]. In [Cas96] other solution approaches are also proposed for the steady Navier-Stokes equations. However, no numerical results exist yet for these alternative approaches.

Ninth International Conference on Domain Decomposition Methods
Editor Petter E. Bjørstad, Magne S. Espedal and David E. Keyes

2 Governing Equations

We consider here steady, incompressible, Newtonian, viscous fluid flow in a three-dimensional domain Ω, as governed by the incompressible Navier-Stokes equations

$$\begin{aligned} \rho u_j u_{i,j} &= \tau_{ij,j} + f_i && \text{in } \Omega\,, && (2.1)\\ u_{i,i} &= 0 && \text{in } \Omega\,. && (2.2) \end{aligned}$$

Here, the stress tensor τ_{ij} can be expressed as

$$\tau_{ij} = -p\delta_{ij} + \mu(u_{i,j} + u_{j,i})\,. \tag{2.3}$$

In (2.1)-(2.3), u_i is the velocity, p is the pressure (relative to zero ambient), f_i is the body force, ρ is the density, μ is the viscosity and δ_{ij} is the Kronecker delta symbol. We use indicial notation, with subscript indicial comma denoting differentiation, e.g., $u_{i,j} = \partial u_i/\partial x_j$. Summation over repeated indices is assumed.

The domain boundary $\partial\Omega$ is decomposed as $\partial\Omega = \partial\Omega_v \cup \partial\Omega_\sigma \cup \partial\Omega_s \cup \partial\Omega_o$, with velocity boundary conditions imposed on $\partial\Omega_v$, surface tension traction boundary conditions imposed on $\partial\Omega_\sigma$, symmetry conditions imposed on $\partial\Omega_s$ and outflow boundary conditions imposed on $\partial\Omega_o$. The appropriate boundary conditions can then be expressed as

$$\begin{aligned} u_i &= \overline{u}_i && \text{on } \partial\Omega_v && (2.4)\\ n_i\tau_{ij}n_j &= \sigma\kappa && \text{on } \partial\Omega_\sigma && (2.5)\\ t_i\tau_{ij}n_j &= t_i\sigma_{,i} && \text{on } \partial\Omega_\sigma && (2.6)\\ t_i\tau_{ij}n_j &= 0 && \text{on } \partial\Omega_s && (2.7)\\ n_i\,u_i &= 0 && \text{on } \partial\Omega_s && (2.8)\\ n_i\tau_{ij}n_j &= -\overline{p} && \text{on } \partial\Omega_o && (2.9)\\ t_i\,u_i &= 0 && \text{on } \partial\Omega_o && (2.10) \end{aligned}$$

Here, $\overline{u}_i$ is the prescribed velocity on $\partial\Omega_v$ (this specification also includes the specification of no-slip conditions), n_i is the outward unit normal on the surface, t_i is any tangent vector on the surface, σ is the surface tension, κ is twice the mean curvature, and $\overline{p}$ is the ambient pressure. Note that the $t_i\,\sigma_{,i}$ in (2.6) must be interpreted as the surface gradient of the surface tension on $\partial\Omega_\sigma$. From (2.7)-(2.10) we also note that the symmetry conditions represent zero normal velocity and zero tangential stress, while the outflow conditions represent zero tangential velocity and normal stress equal to minus the ambient pressure.

Finally, the kinematic condition at steady state is

$$u_i n_i = 0 \quad \text{on } \partial\Omega_\sigma\,. \tag{2.11}$$

3 Variational Formulation

From (2.4)-(2.10) we note that all the three velocity components are specified on $\partial\Omega_v$, two components are specified on $\partial\Omega_o$, one component is specified on $\partial\Omega_s$ and no component is specified on $\partial\Omega_\sigma$. We first define the space $H^1_D(\Omega)$ to be the usual H^1 space, but where the (scalar) members of the space must be compatible with both the non-homogeneous and homogeneous *Dirichlet* (vector) velocity boundary conditions on the domain boundary $\partial\Omega$. In an analogous fashion, we define the space $H^1_0(\Omega)$ to be similar to the space $H^1_D(\Omega)$ except that *homogeneous* Dirichlet conditions are imposed wherever Dirichlet velocity boundary conditions are specified in $H^1_D(\Omega)$.

The equivalent weak form of (2.1)-(2.10) can then be expressed as: Find $u_i \in H^1_D(\Omega)$ and $p \in L^2(\Omega)$ such that $\forall v_i \in H^1_0(\Omega)$,

$$\int_\Omega \{ \rho v_i u_j u_{i,j} + v_{i,j}[-p\delta_{ij} + \mu(u_{i,j} + u_{j,i})] - v_i f_i \} d\Omega - I_\sigma(v_i) = 0 \ , \tag{3.12}$$

$$\forall q \in L^2(\Omega) \, , \qquad \int_\Omega q\, u_{i,i}\, d\Omega = 0 \qquad , \tag{3.13}$$

where

$$I_\sigma(v_i) = \int_{\partial\Omega_\sigma} v_i\, \tau_{ij} n_j\, dS = \int_{\partial\Omega_\sigma} v_i\, \overline{g}_i\, dS \quad . \tag{3.14}$$

The surface integral given in (3.14) corresponds to the natural imposition of stress boundary conditions along the free surface. In [HP91] a very elegant form is derived for the imposed surface traction $\overline{g}_i$, where only surface-intrinsic co-ordinates are needed, and where a single form automatically generates both the normal and tangential boundary conditions required for viscous analysis.

In this study, our variational form is based on the results derived in [HP91]. Even though the numerical results in [HP91] put a restriction on the free surface to be either closed or periodic, the variational results derived in [HP91] are also appropriate for the more general case where the free surface intersects an outflow boundary or a symmetry boundary. For example, the variational form automatically provides a term which allows for the imposition of contact angles in three dimensions; this term is needed in this study in order to impose a 90° angle along the line where the free surface intersects the symmetry boundary or the outflow boundary.

For free surface problem, the geometry along the free surface is an unknown in addition to the velocity and pressure. The variational form derived in [HP91] only requires the geometry space to include C^0-surfaces, even though the curvature term in (2.5) (strong form) suggests that C^1-surfaces may be required. The lowering of the continuity requirement for the geometry description is here in line with finite element analysis for second-order partial differential equations.

The kinematic condition given in (2.11) can be used directly in order to update the free surface shape. Here, however, we shall use an approach similar to the approach studied in [HR94] for two-dimensional problems. In particular, we shall consider an extension to three dimensions for the case when surface tension plays an important role; this is described in the next section.

4 Discretization and Solution Strategy

Our discrete equations are generated based upon the weak form. Following a spectral element discretization procedure [MP89], the domain is broken up into K hexahedral elements. Within each element, and along each (local) spatial direction, the geometry and the velocity are approximated as N^{th} order polynomials, while the pressure is approximated as a polynomial of degree $N-2$ [MPR92].

Following a similar procedure as described in [Røn96] (non-staggered grid), we arrive at a set of discrete equations which can be expressed in matrix form as

$$\underline{\mathbf{A}}\,\underline{\mathbf{u}} + \underline{\mathbf{C}}(\underline{\mathbf{u}})\,\underline{\mathbf{u}} - \underline{\mathbf{D}}^T \underline{p} \;=\; \underline{\mathbf{f}} \;, \tag{4.15}$$

$$\underline{\mathbf{D}}\,\underline{\mathbf{u}} \;=\; \underline{0} \;. \tag{4.16}$$

Here, $\underline{\mathbf{A}}$ is the discrete (coupled) viscous operator, $\underline{\mathbf{C}}(\underline{\mathbf{u}})$ is the discrete, nonlinear, nonsymmetric advection operator. $\underline{\mathbf{D}}$ is the discrete divergence operator, and its transpose $\underline{\mathbf{D}}^T$ is the discrete gradient operator. The vector $\underline{\mathbf{u}}$ contains the nodal velocity values, $\underline{p}$ represents the nodal pressure values, and the components of $\underline{\mathbf{f}}$ are the nodal forces.

We now give a brief summary of the entire solution procedure:

1. Similar to the approach used in [HR94], we first solve the discrete Navier-Stokes equations imposing the *kinematic* condition along the free surface (zero normal component) instead of the original *free surface* boundary conditions.
2. Next, we evaluate the residual in the discrete momentum equations imposing the *original* free surface boundary conditions, and restrict this residual to the free surface. We remark that, since we assume that surface tension plays an important role (large Weber number), we can *interpret* the normal component of this free surface residual as being due to an incorrect local curvature used in (3.14).
3. The surface term (3.14), represented in the form derived in [HP91], is linearized with respect to the normal surface displacement. From this linearization, we create a system of equations for the normal free surface displacement, with a right hand side equal to the residual in the momentum equations, restricted to the free surface, and restricted to the normal component.
4. We now solve the system for the normal free surface displacement.
5. The free surface displacement is extended smoothly to the interior of the computational domain by solving the three-dimensional, discrete elasticity equations, imposing the free surface displacement as Dirichlet data.
6. The geometry is now updated, taking into account the computed deformation in the entire domain.
7. Steps 1-6 are repeated until the residual in the momentum equations and the free surface displacement are below prescribed tolerances.

5 A Domain Decomposition Navier-Stokes Solver

In the following, we shall focus our attention on Step 1 in the algorithm described in the previous section. We start by first expressing the original, nonsymmetric, nonlinear, discrete steady Navier-Stokes system (4.15)-(4.16) in the compact form

$$\underline{\mathbf{F}}\,\underline{\mathbf{x}} = \underline{\mathbf{g}} \;, \tag{5.17}$$

where

$$\underline{\mathbf{F}} = \begin{pmatrix} (\underline{\mathbf{A}} + \underline{\mathbf{C}}(\mathbf{u})) & -\underline{\mathbf{D}}^T \\ \underline{\mathbf{D}} & \underline{\mathbf{0}} \end{pmatrix} ,$$

$$\underline{\mathbf{x}} = [\underline{\mathbf{u}}, \underline{p}]^T ,$$

$$\underline{\mathbf{g}} = [\underline{\mathbf{g}}_u, \underline{g}_p]^T \equiv [\underline{\mathbf{f}}, \underline{0}]^T .$$

As usual we linearize the system (5.17) and perform a Newton iteration. For each iteration we have to solve a system of the form

$$\underline{\mathbf{N}}\,\delta\underline{\mathbf{x}}^n = \underline{\mathbf{g}} - \underline{\mathbf{F}}\,\underline{\mathbf{x}}^{n-1} \tag{5.18}$$

where $\underline{\mathbf{N}}$ represents the linearized Navier-Stokes operator, $\underline{\mathbf{x}}^n$ is the solution after n Newton iterations, and $\delta\underline{\mathbf{x}}^n = \underline{\mathbf{x}}^n - \underline{\mathbf{x}}^{n-1}$.

The iterative substructuring algorithm we now present is for the system (5.18). Our method can therefore be described as a Newton-Krylov method. We proceed by directly considering the preconditioned, linearized, steady Navier-Stokes system

$$\underline{\mathbf{M}}^{-1}\underline{\mathbf{N}}\,\delta\underline{\mathbf{x}}^n = \underline{\mathbf{M}}^{-1}(\underline{\mathbf{g}} - \underline{\mathbf{F}}\,\underline{\mathbf{x}}^{n-1}) , \tag{5.19}$$

where $\underline{\mathbf{M}}^{-1}$ represents the Navier-Stokes preconditioner.

Following the approach given in [Røn96], the Navier-Stokes preconditioner can be expressed in the additive form

$$\underline{\mathbf{M}}^{-1} = \underline{\mathbf{M}}_0^{-1} + (\mathbf{I} + \underline{\mathbf{M}}_\Gamma^{-1}\underline{\mathbf{G}})\,[\sum_{k=1}^{K} \underline{\mathbf{M}}_k^{-1}\,] + \underline{\mathbf{M}}_\Gamma^{-1} \tag{5.20}$$

where

$$\underline{\mathbf{G}} = \begin{pmatrix} \underline{\mathbf{0}} & +\underline{\mathbf{D}}^T \\ \underline{\mathbf{0}} & \underline{\mathbf{0}} \end{pmatrix} . \tag{5.21}$$

We now briefly comment on the individual components in this preconditioner. Most of the details are given in [Røn96] in the context of the velocity formulation of the Navier-Stokes equations. We shall here emphasize the extension of this preconditioner to solving problems with general boundary conditions, for which the stress formulation of the Navier-Stokes equations is required. Our specific application will be the study of three-dimensional free surface flow. The original work in [Røn96] was inspired by progress in the understanding and application of domain decomposition methods over the past years [DW90, SBG96].

The first term in the preconditioner, $\underline{\mathbf{M}}_0^{-1}$, involves the restriction of the velocity-pressure residual to a coarse grid. We consider here a coarse grid based upon quadratic Q_2/P_1 finite elements induced by the original spectral element decomposition. Note that, in this study, each N^{th}-order spectral element represents an individual subdomain. The quadratic finite element grid is used in order to construct a coarse discretization of the original, linearized Navier-Stokes operator. We also add streamline diffusion on the coarse grid if the grid Peclet number is sufficiently large. The coarse residual represents the right-hand-side for this coarse system, which is solved using a direct, banded solver. Note that the coarse, linearized Navier-Stokes system here couples all the velocity components, even for the linear Stokes case. The coarse system is also compatible with all the original boundary conditions, in which zero, one, two or three velocity components are specified on the various surface segments. After the coarse solution has been computed, the solution is extended to the original spectral element mesh.

The term $\underline{\mathbf{M}}_k^{-1}$ in the preconditioner represents the solution of a local Stokes problem defined in subdomain $\Omega_k, k = 1, ..., K$, with homogeneous velocity boundary conditions prescribed on the subdomain boundary $\partial\Omega_k$. We now make several remarks regarding the local, discrete Stokes operator. First, the stress formulation is replaced by the velocity formulation. This is a valid replacement due to the homogeneous velocity boundary conditions. Thus, the coupled viscous operator is replaced by the (vector) Laplacian operator. Second, the scalar, spectral, elemental Laplacian operator is replaced by a spectrally equivalent finite element operator [DM90]; in three dimensions, this operator represents linear tetrahedral elements based on the tensor-product Gauss-Lobatto Legendre points. The local, scalar, discrete finite element Laplacian operator is explicitly constructed and inverted using a banded, direct solver. In order to proceed, this operator is then used to explicitly construct and invert the local, Uzawa pressure operator (the local pressure is defined to have a zero average). Hence, only back substitution is needed during each preconditioning step, one to find the local pressure, and one for each velocity component. For more details, see [Røn96].

The term $\underline{\mathbf{M}}_\Gamma^{-1}$ represents the inversion of the diagonal of the viscous operator for all the velocity degrees-of-freedom on the subdomain interfaces, Γ. Note that this operator appears in two places in the preconditioner (5.20). This is due to the fact that there are two contributions to the right-hand-side for this interface system. One is the initial residual in the momentum equations, restricted to the subdomain interfaces. The second is the result of operating with the gradient operator $\underline{\mathbf{D}}^T$ on the pressure found from solving all the local Stokes problems, and then restricting the result to the subdomain interfaces; this second contribution gives rise to the gradient operator $\underline{\mathbf{G}}$ in (5.20). Finally, the prolongation operator associated with this interface system represents the identity operator for the velocity degrees-of-freedom along Γ, and the zero operator for the remaining degrees-of-freedom in the domain.

In summary, our Navier-Stokes preconditioner is based upon a *hierarchy* of discrete, spatial operators, starting with a linearized Navier-Stokes operator for the coarse, global problem, a steady Stokes operator for each individual, local problem, and finally, an elliptic (Poisson type) operator for the interface problem.

Finally, we remark that all the individual components in the preconditioner (5.20) are here updated after each geometry update. This is perhaps a conservative approach, and future work will include an investigation into finding a more optimal updating strategy for free surface problems.

6 Numerical Results

We now present numerical results for the extrusion of a three-dimensional die through an extruder with a square cross-section. Symmetry conditions allows us to only consider a quarter of the die. The boundary conditions consist of: (i) the specification of an inlet velocity; (ii) no-slip velocity conditions along the extruder walls; (iii) free surface boundary conditions along the surface of the die that is exposed to ambient pressure; (iv) outflow conditions to limit the computational domain; and (v) two orthogonal symmetry planes resulting from the fact that we only consider a quarter of the original, square die.

The computational domain is broken up into $K = 33$ spectral elements, each of order $N = 5$. The Reynolds number is $Re = \rho UL/\mu = 20$ and the Weber number is $We = \sigma/(\rho LU^2) = 2$. Here, the characteristic length, L, is based upon the thickness of the die, and the characteristic velocity, U, is based upon the inlet velocity.

Figure 1 shows the initial surface mesh along the (fixed) extruder walls (right half of the domain), along the free surface (left half of the domain), and along one of the symmetry planes (bottom surface). The flow goes from right to left. The fixed, square extruder exit is represented by the square cross-section in the middle of the domain.

Figure 2 shows the final surface mesh along the fixed extruder walls (identical to the initial mesh), along one of the symmetry planes (the view is identical to the initial mesh), along the free surface, and along the outflow boundary. As the die exits from the extruder (in the middle of the domain), and a relaxation of the stresses occur, we see that the die swells away from the corners of the square extruder exit, while the die contracts near the corners of the extruder (only a single corner is present in this computational model due to the two orthogonal symmetry planes). The die shape also undergoes a dramatic change in shape from having a square cross section at the extruder exit to having a circular cross section at the outflow boundary.

We now comment on the outflow conditions used during this simulation. A priori, the pressure at the outflow is not known. However, we made the *assumption* that the final die shape at the outflow would, in fact, have a circular shape. In this case, the pressure can readily be calculated from knowing the radius of the cross-section and the surface tension. For each update of the geometry, this model was used in order to update the pressure at the outflow boundary. Note that only a *single node* along the intersection of the free surface and the outflow boundary was used for the purpose of estimating the radius. At convergence, we see that our assumption was correct, and hence, our strategy is valid for this particular case.

In terms of the performance of the steady Navier-Stokes algorithm, our experience so far suggests that the behavior is quite similar to the one reported in [Røn96]. For simple, three-dimensional model problems, the introduction of the more general stress formulation as well as more general stress boundary conditions, do not seem to significantly change the earlier performance results.

Acknowledgement

The author would like to thank NASA Goddard Space Flight Center for the financial support of most of this work (contract NAS5-38075). The author would also like to thank Dr. Lee W. Ho for many helpful discussions and suggestions during the course of this work. Finally, the author would like to thank Prof. Mario A. Casarin and Prof. Olof B. Widlund for their comments on an earlier version of the manuscript.

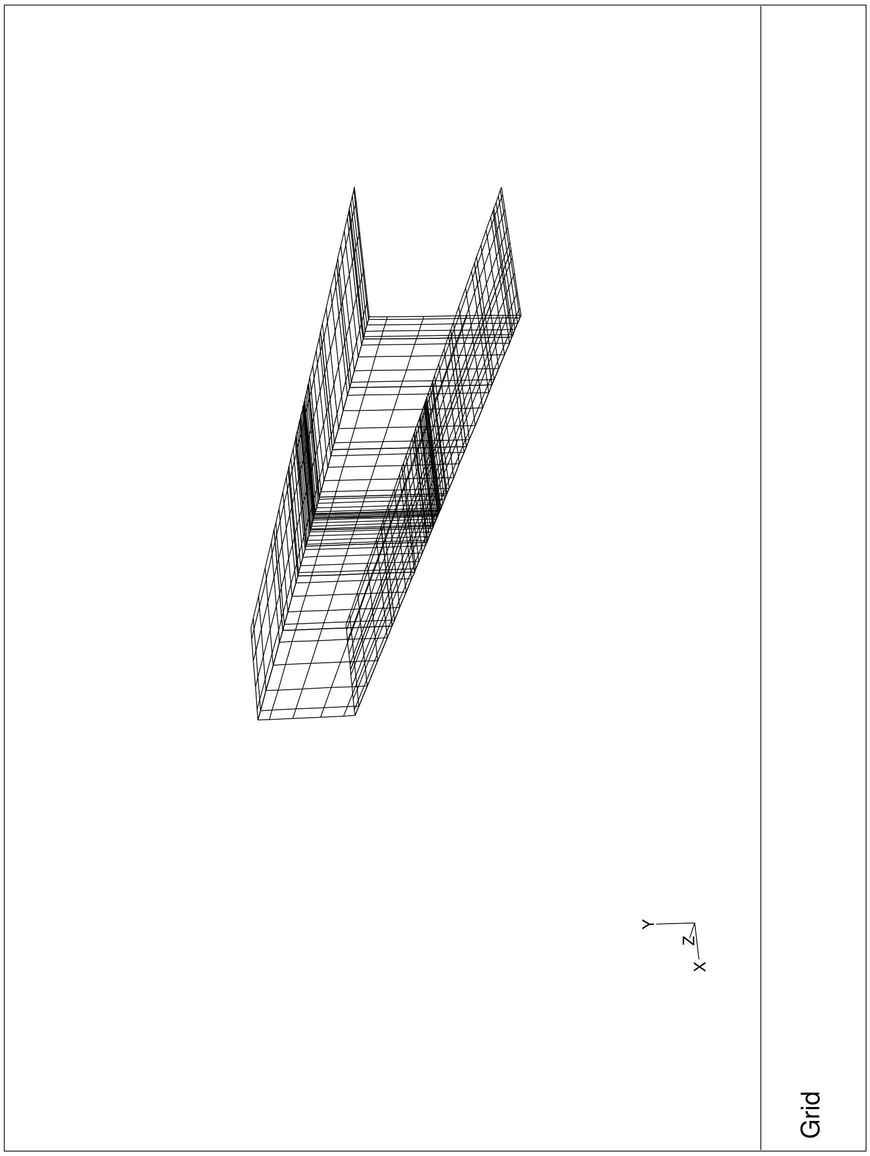

Figure 1 Extrusion of a square die; initial mesh

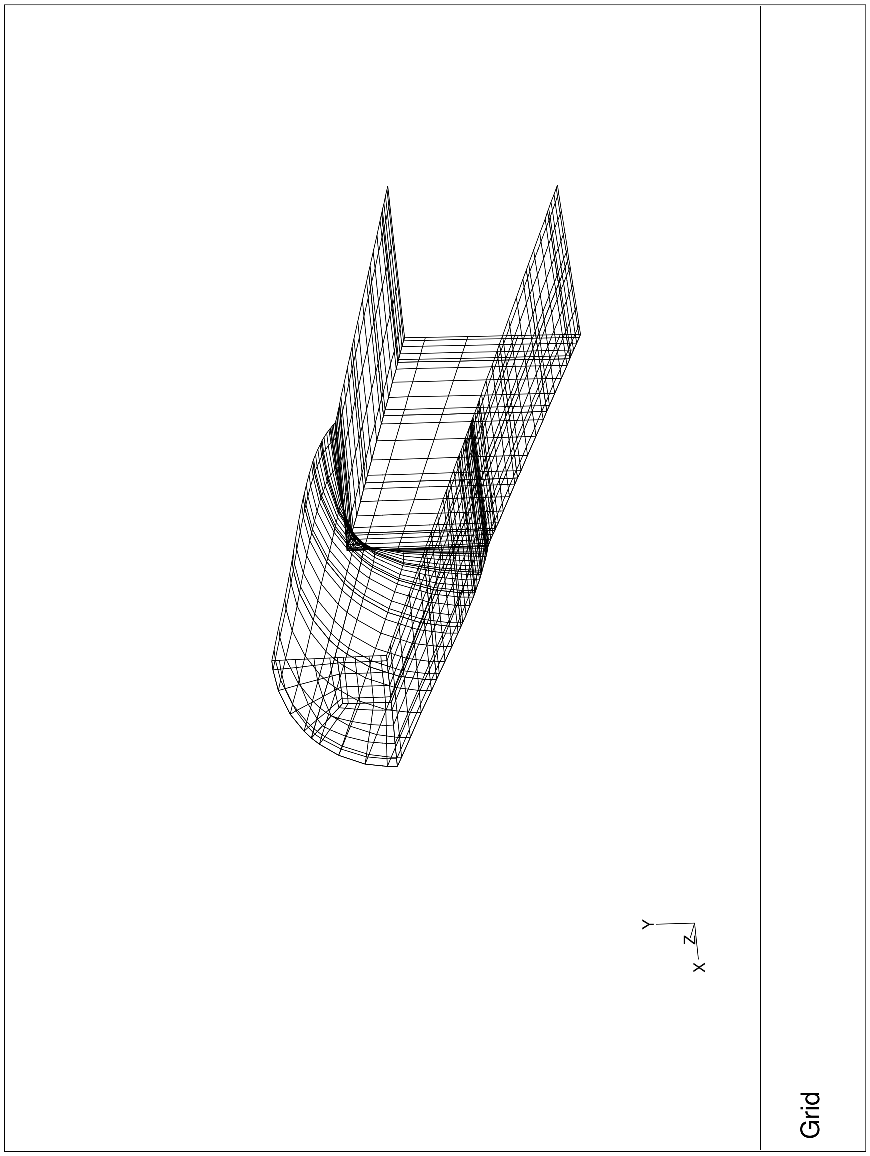

Figure 2 Extrusion of a square die; final mesh; Re=20 and We=2

REFERENCES

[Cas96] Casarin M. (1996) *Schwarz preconditioners for spectral and mortar finite element methods with applications to incompressible fluids.* PhD thesis, New York University.

[CMBBC91] Chenot J., Montmitonnet P., Bern A., and Bertrand-Corsini C. (1991) A method for determining free surfaces in steady state finite element computations. *Computer Methods in Applied Mechanics and Engineering* 92: 245–260.

[DM90] Deville M. and Mund E. (1990) Finite-element preconditioning for pseudospectral solutions of elliptic problems. *SIAM J. Sci. Stat. Comput.* 11(2): 311–342.

[DW90] Dryja M. and Widlund O. (1990) Towards a unified theory of domain decomposition algorithms for elliptic problems. In Chan T., Glowinski R., Periaux J., and Widlund O. (eds) *Proceedings of the Third International Symposium on Domain Decomposition Methods for PDE's.* SIAM.

[EPW92] Ellwood K., Papanastasiou T., and Wilkes J. (1992) Three-dimensional streamlined finite elements: design of extrusion dies. *Int. J. Numer. Meth. in Fluids* 14: 13–24.

[HP91] Ho L. and Patera A. (1991) Variational formulation of three-dimensional viscous free-surface flows: natural imposition of surface tension boundary conditions. *International Journal for Numerical Methods in Fluids* 13: 691–698.

[HR94] Ho L. and Rønquist E. (1994) Spectral element solution of steady incompressible viscous free-surface flows. *Finite Elements in Analysis and Design* 16: 207–227.

[MP89] Maday Y. and Patera A. (1989) Spectral element methods for the Navier-Stokes equations. In Noor A. (ed) *State of the Art Surveys in Computational Mechanics*, pages 71–143. ASME, New York.

[MPR92] Maday Y., Patera A., and Rønquist E. (1992) The $P_N \times P_{N-2}$ method for the approximation of the Stokes problem. Technical Report 92009, Department of Mechanical Engineering, Massachusetts Institute of Technology.

[RE96] Ramanan N. and Engelman M. (1996) An algorithm for simulation of steady free surface flows. *Int. J. Numer. Meth. in Fluids* 22: 103–120.

[Røn96] Rønquist E. (1996) A domain decomposition solver for the steady Navier-Stokes equations. In Ilin A. and Scott L. (eds) *Proceedings of the Third International Conference on Spectral and High Order Methods.* Houston Journal of Mathematics. Conference held in Houston, Texas, 5-9 June 1995.

[SBG96] Smith B. F., Bjørstad P., and Gropp W. (1996) *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations.* Cambridge University Press.