BEM simulation for the pressing of glass

BEM Simulation for the Pressing of Glass

K. Wang, R.M.M. Mattheij & H.G. ter Morsche
EMail: wang@win.tue.nl

Abstract

The main objective of this research is to develop reliable numerical methods to simulate the pressing of glass. The glass can be modelled as a Stokes flow with three types of boundaries: free, fixed and "moving" boundaries. Because of axi–symmetry we actually have a 2-D problem. This problem is solved by rewriting it as a boundary integral equation. A boundary element method is then employed to solve the integral equation of the Stokes flow and time integration is carried out by a kind of predictor-corrector scheme.

1 Introduction

In Mattheij at el. [12] a model was given to describe the flow of glass in a mould with partially free and partially prescribed boundaries. Here we shall consider a numerical approach to simulate the actual solution by a boundary element method.

If we let $\mathbf{v} = (v_i)$ be the velocity vector, p the pressure and $\sigma = (\sigma_{ij})$ stress tensor, then the flow of the glass can be seen to be described by the *continuity equation*

$$\nabla \cdot \mathbf{v} = 0, \tag{1}$$

and the *momentum equation*

$$\Delta \mathbf{v} - \nabla p = 0. \tag{2}$$

Whereas boundary conditions are given by

$$\sigma n = -p_0 \mathbf{n} \quad \text{on the free boundary,} \tag{3}$$

$$\begin{cases} \mathbf{v} \cdot \mathbf{n} = 0 \\ \sigma \mathbf{n} \cdot \mathbf{t} = -\beta_m \mathbf{v} \cdot \mathbf{t} \end{cases} \text{on the mould boundary,} \tag{4}$$

and

$$\begin{cases} \mathbf{v} \cdot \mathbf{n} = \mathbf{v}_p \cdot \mathbf{n} \\ \sigma \mathbf{n} \cdot \mathbf{t} = -\beta_p (\mathbf{v} - \mathbf{v}_p) \cdot \mathbf{t} \end{cases} \text{on the plunger boundary.} \tag{5}$$

Here $\mathbf{n} = (n_i)$ is the outward unit normal, $\mathbf{t} = (t_i)$ is the unit tangential direction, $\mathbf{v}_p$ is the velocity of the plunger, β_m and β_p are parameters indicating the roughness of the mould and the plunger respectively.

The movement of a material fluid particle is described by

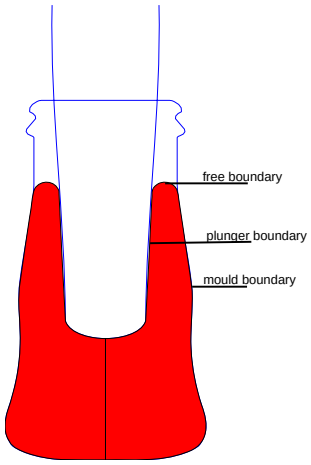

Figure 1: The boundaries

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x}(t)). \tag{6}$$

Now our objective is to solve problem (1) – (6). In section 2 we solve the Stokes problem in an axi-symmetric domain by a Boundary Element Method and In section 3 we further deal with equation (6), by developing a predictor-corrector scheme. Finally we will assess the method by considering some actual simulations in section 4.

2 BEM for an Axisymmetric Stokes Flow

There are two approaches for deriving the governing integral formulation for axi–symmetric problems based on hydrodynamic potentials of single– and double–layers. Both methods are leading to the same equation. The first one is to obtain the integral equation by using the axi–symmetric fundamental solution based on ring forces(cf. Becker[2]). The second approach is to apply the fundamental solution derived from a point force to obtain the a Cartesian version of the three–dimensional integral equation. Subsequently cylindrical coordinates are then substituted in this formulation. Here we use the latter method.

2.1 The Boundary Integral Equations

Let us denote the fluid region by Ω and its surface by S. The fundamental solution to problem (1) and (2) is given by

$$\begin{cases} u_{ki}(\mathbf{x},\mathbf{y}) &= \frac{1}{8\pi}\left[\frac{\delta_{ik}}{|\mathbf{x}-\mathbf{y}|}+\frac{(x_i-y_i)(x_k-y_k)}{|\mathbf{x}-\mathbf{y}|^3}\right], \\ q_k(\mathbf{x},\mathbf{y}) &= \frac{x_k-y_k}{4\pi|\mathbf{x}-\mathbf{y}|^3}, \end{cases} \tag{7}$$

i.e. $(\mathbf{u}_k, q_k)$ satisfy

$$\begin{cases} \nabla\cdot\mathbf{u}_k &= 0, \\ \Delta\mathbf{u}_k - \nabla q_k &= -\delta(\mathbf{x},\mathbf{y})\mathbf{e}_k. \end{cases}$$

where $\mathbf{e}_k$ is the k–th unit vector of an Cartesian coordinate system.

By using the divergence theorem, we can obtain the *Green's formula* for the Stokes problem, i.e., if $\mathbf{v}$ and $\mathbf{v}'$ are divergence free, p and p' are smooth scalar then

$$\begin{aligned} &\int_\Omega[(\Delta v_i - \tfrac{\partial p}{\partial x_i})v_i' - (\Delta v_i' - \tfrac{\partial p'}{\partial x_i})v_i]d\Omega \\ &= \int_\Gamma[\sigma_{ij}(\mathbf{v},p)n_j v_i' - \sigma_{ij}(\mathbf{v}',q')n_j v_i]d\Gamma, \end{aligned}$$

where $\sigma_{ij}(\mathbf{v},p) = -p\delta_{ij} + v_{i,j} + v_{j,i}$ and $\sigma_{ij}(\mathbf{v}',p') = -p'\delta_{ij} + v'_{i,j} + v'_{j,i}$.

By replacing v_i' and p' by the fundamental solutions u_{ki} and q_k, identifying v_i and p with the solution to Eqns. (1) and (2), and letting the source point approach the boundary, we obtain the following *boundary integral equation*(cf. Ladyzhenskaya[7], Pozrikidis[14] and Brebbia et al. [3])

$$c_{ij}(\mathbf{x})v_j(\mathbf{x}) + \int_S q_{ij}(\mathbf{x},\mathbf{y})v_j dS_\mathbf{y} = \int_S u_{ij}(\mathbf{x},\mathbf{y})b_j dS_\mathbf{y}, \tag{8}$$

where the kernels q_{ij} is equal to :

$$q_{ij}(\mathbf{x},\mathbf{y}) = \frac{3(x_i-y_i)(x_j-y_j)(x_k-y_k)n_k}{4\pi|\mathbf{x}-\mathbf{y}|^5}. \tag{9}$$

The coefficients c_{ij} depend on the position of the point $\mathbf{x}$, but in the BEM, we will show that it is not necessary to know the analytical expressions of c_{ij} although they are available (cf. Brebbia at el. [3]).

To obtain the integral equation for the axi-symmetric case, we reformulate the representation above by employing cylindrical coordinates (r,θ,z), i.e.

$$\mathbf{y} = (y_1,y_2,y_3)^T = (r\cos\theta, r\sin\theta, z)^T.$$

Because of the rotational symmetry, we only have to determine v_r and v_z at the intersection of the surface S and (say) the half–space $\theta = 0$. This intersection curve is denoted by Γ(see Figure 2), therefore $dS = rd\theta d\Gamma$. Let $\mathbf{x} = (R,0,Z)^T \in \Gamma$. After successive substitution of cylindrical coordinates and integration along the θ–direction of Eqn. (8) we obtain

$$c^c_{ij}(\mathbf{x}^c)v^c_j(\mathbf{x}^c) + \int_\Gamma r q^c_{ij}(\mathbf{x}^c,\mathbf{y}^c)v^c_j(\mathbf{y}^c)d\Gamma = \int_\Gamma r u^c_{ij}(\mathbf{x}^c,\mathbf{y}^c)b^c_j(\mathbf{y}^c)d\Gamma, \tag{10}$$

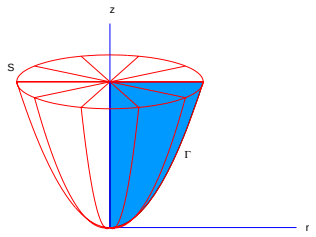

Figure 2: Γ(the solid lines) and S

where the superscript c stands for cylindrical and i, j are now either 1 or 2; source point $\mathbf{x}^c = (R, Z)^T$, field point $\mathbf{y}^c = (r, z)^T$; velocity $\mathbf{v}^c = (v_i^c) = (v_r, v_z)^T$, stress vector $\mathbf{b}_c = (b_i^c) = (b_r, b_z)^T$. the kernel u_{ij}^c can be written as

$$\begin{array}{rcl} u_{ij}^c & = & \frac{1}{2\pi\sqrt{a+b}}\left[\frac{E(k)}{a-b}\left(A_{ij}^0 + \frac{a}{b}A_{ij}^1 + \frac{2a^2-b^2}{b^2}A_{ij}^2\right)\right. \\ & & \left. - \frac{K(k)}{b}\left(A_{ij}^1 + \frac{2a}{b}A_{ij}^2\right)\right], \end{array} \tag{11}$$

where

$$a = r^2 + R^2 + c^2, \quad b = 2rR, \quad c = Z - z, \quad k = \sqrt{\frac{2b}{a+b}}.$$

$K(k)$ and $E(k)$ are the complete elliptic integrals of the first and second kind respectively, i.e.

$$K(k) := \int_0^{\frac{\pi}{2}} \frac{d\phi}{\sqrt{1 - k^2\sin^2\phi}}, \qquad E(k) := \int_0^{\frac{\pi}{2}} \sqrt{1 - k^2\sin^2\phi}\, d\phi. \tag{12}$$

The coefficient matrices $\mathbf{A}^n$ are defined by

$$\mathbf{A}^0 = \begin{pmatrix} -\frac{1}{2}b & cR \\ -cr & a + c^2 \end{pmatrix}, \mathbf{A}^1 = \begin{pmatrix} 2a - c^2 & -cr \\ cR & -b \end{pmatrix}, \mathbf{A}^2 = \begin{pmatrix} -\frac{3}{2}b & 0 \\ 0 & 0 \end{pmatrix}.$$

While the kernel q_{ij}^c can be represented by

$$\begin{array}{rcl} q_{ij}^c = & \frac{1}{\pi(a^2-b^2)\sqrt{a+b}}\left[K(k)\left(-B_{ij}^0 - \frac{a}{b}B_{ij}^1 + \frac{2a^2-3b^2}{b^2}B_{ij}^2 + \frac{a(8a^2-9b^2)}{b^3}B_{ij}^3\right)\right. \\ & \left. + \frac{E(k)}{a-b}\left(4aB_{ij}^0 + \frac{a^2+3b^2}{b}B_{ij}^1 + \frac{2a(3b^2-a^2)}{b^2}B_{ij}^2 - \frac{8a^4-15a^2b^2+3b^4}{b^3}B_{ij}^3\right)\right]. \end{array}$$

Here coefficient matrices $\mathbf{B}^n$ are defined by

$$\mathbf{B}^0 = \begin{pmatrix} -\frac{1}{2}db & Rdc \\ -rdc & dc^2 \end{pmatrix}, \quad \mathbf{B}^1 = \begin{pmatrix} de - \frac{1}{2}bRn_r & (R^2n_r - rd)c \\ (d - rn_r)Rc & Rc^2n_r \end{pmatrix},$$

$$\mathbf{B}^2 = \begin{pmatrix} (en_r - rd)R & -\frac{1}{2}bcn_r \\ R^2cn_r & 0 \end{pmatrix}, \quad \mathbf{B}^3 = \begin{pmatrix} -\frac{1}{2}bRn_r & 0 \\ 0 & 0 \end{pmatrix},$$

where $d = -rn_r + cn_z, e = R^2 + r^2$ and $\mathbf{n} = (n_r, n_z)$ is the unit outward normal of boundary Γ.

We remark that if the point $\mathbf{x}$ is lying at the z–axis, i.e. $R = 0$, the above formulae are not applicable because of $b = 0$. However, direct computation leads to

$$u_{ij}^c = \frac{1}{8a^{\frac{3}{2}}} \left[2(A_{ij}^0 + A_{ij}^2) \right],$$

and

$$q_{ij}^c = \frac{3}{4a^{\frac{5}{2}}} \left[2(B_{ij}^0 + B_{ij}^2) \right].$$

Note that if the domain includes z–axis then Γ is an open curve.

2.2 The Boundary Element Method

Although physically 3–D, the BEM implementation of Eqn. (10) is similar to a 2–D problem. To obtain a BEM formulation, we rewrite the boundary integral Eqn. (10) (we remove the superscript c for simplicity)

$$\mathbf{c}(x)v(x) + \int_\Gamma \mathbf{q}(x,y)v(y)d\Gamma = \int_\Gamma \mathbf{u}(x,y)b(y)d\Gamma. \tag{13}$$

The boundary is divided into segments Γ_k, i.e.

$$\Gamma = \sum_k \Gamma_k, \tag{14}$$

and the velocity and stress vector are represented by shape functions

$$\mathbf{v} = \sum_j \mathbf{v}^j \phi_j, \quad \mathbf{b} = \sum_j \mathbf{b}^j \phi_j. \tag{15}$$

By substituting Eqns. (14) and (15) into Eqn. (13) we have

$$\begin{array}{rcl} \mathbf{c}(x)v(x) & + & \sum_j \sum_k \int_{\Gamma_k} \mathbf{q}(x,y)\phi_j(\mathbf{y})d\Gamma \mathbf{v}^j \\ & = & \sum_j \sum_k \int_{\Gamma_k} \mathbf{u}(x,y)\phi_j(\mathbf{y})d\Gamma \mathbf{b}^j \end{array}$$

For a particular point $\mathbf{x}_i$, denoting $\mathbf{c}(\mathbf{x}_i)$, $\mathbf{v}(\mathbf{x}_i)$ by $\mathbf{c}^i$ and $\mathbf{v}^i$ respectively, we thus have

$$\begin{array}{rcl} \mathbf{c}^i\mathbf{v}^i & + & \sum_j \sum_k \int_{\Gamma_k} \mathbf{q}(x_i,\mathbf{y})\phi_j(\mathbf{y})d\Gamma \mathbf{v}^j \\ & = & \sum_j \sum_k \int_{\Gamma_k} \mathbf{u}(x_i,\mathbf{y})\phi_j(\mathbf{y})d\Gamma \mathbf{b}^j. \end{array} \tag{16}$$

Now define the matrix $\mathbf{H}$ by

$$\mathbf{H}_{ij} := \sum_k \int_{\Gamma_k} \mathbf{q}(x_i,\mathbf{y})\phi_j(\mathbf{y})d\Gamma + \mathbf{c}^i\delta_{ij}, \tag{17}$$

and the matrix $\mathbf{G}$ by

$$\mathbf{G}_{ij} := \sum_k \int_{\Gamma_k} \mathbf{u}(x_i, \mathbf{y}) \phi_j(\mathbf{y}) d\Gamma. \tag{18}$$

This results in the following square full rank system of linear algebraic equations, denoted by

$$\mathbf{H}\mathcal{V} = \mathbf{G}\mathcal{B}. \tag{19}$$

Here $\mathcal{V}$, $\mathcal{B}$ are vectors consisting of the velocity and stress vector respectively. Note that every element of $\mathbf{H}$ and $\mathbf{G}$ is a 2×2 submatrix, and every element of $\mathcal{V}$ and $\mathcal{B}$ is a $2N \times 1$ subvector. If there are N collocation points, and probably include M corners, then $\mathbf{H}$ and $\mathbf{G}$ are actually $2N \times 2N$ and $2N \times 2(N+M)$ matrices respectively, whereas $\mathcal{V}$ and $\mathcal{B}$ are $2N \times 1$ and $2(N+M) \times 1$ vectors.

In order to build $\mathbf{H}$ and $\mathbf{G}$ the elliptic integrals (12) that occur in the coefficients are approximated by using a series representation(cf. Becker[1] and Cody[5]), i.e.

$$\begin{array}{l} K(k) \approx \log(4) + \sum_{i=1}^n a_i \eta^i + \log(\frac{1}{\eta})[\frac{1}{2} + \sum_{i=1}^n b_i \eta^i] \\ E(k) \approx 1 + \sum_{i=1}^n c_i \eta^i + \log(\frac{1}{\eta}) \sum_{i=1}^n d_i \eta^i, \end{array} \tag{20}$$

where $\eta = 1 - k^2$.

When the element integrals become singular, we use Telles' transformation (cf. Partridge at al. [13] and Telles[17]) to remove the singularity. More precisely we consider the following integral

$$I = \int_{-1}^{1} f(\eta) d\eta,$$

in which $f(\eta)$ is singular at a point $\bar{\eta}$. One can choose a transformation

$$\eta = a\gamma^3 + b\gamma^2 + c\gamma + d,$$

such that

$$\begin{cases} \eta(-1) = -1 \\ \eta(1) = 1 \\ \frac{d\eta}{d\gamma}|_{\bar{\eta}} = 0 \\ \frac{d^2\eta}{d\gamma^2}|_{\bar{\eta}} = 0, \end{cases}$$

by which the coefficients a, b, c, d can be determined. Then I becomes

$$I = \int_{-1}^{1} f(\eta) \frac{d\eta}{d\gamma} d\gamma.$$

If $f(\eta)$ has only a logarithmic singularity at $\bar{\eta}$, then as a function of γ, $f(\eta)\frac{d\eta}{d\gamma}$ is a regular function, and standard Gaussian integration can be employed.

As for the diagonal submatrix of $\mathbf{H}$, we first use a rigid-body motion in the z–direction to obtain the elements that apply in this particular direction, i.e.

$$\begin{pmatrix} H_{rz} \\ H_{zz} \end{pmatrix}_{ii} = - \sum_{\substack{j=1 \\ i \neq j}}^{N} \begin{pmatrix} H_{rz} \\ H_{zz} \end{pmatrix}_{ij}, \quad \text{for } i = 1, 2, \cdots, N. \tag{21}$$

To determine the other two coefficients H_{rr} and H_{zr}, we employ a particular Stokes flow from which the velocity and tension can be computed for any arbitrarily shaped region. For example, we use the following axi–symmetric Stokes flow

$$\begin{aligned} \bar{\mathbf{v}}^c &= (r/6, -z/3)^T, \\ \bar{\mathbf{b}}^c &= (n_r, 0)^T. \end{aligned}$$

Substituting this solution into Eqn. (19) we obtain

$$\begin{aligned} \begin{pmatrix} H_{rr} \\ H_{zr} \end{pmatrix}_{ii} = & \left(\begin{pmatrix} H_{rz} \\ H_{zz} \end{pmatrix}_{ii} \left(\tfrac{z}{3}\right)_i \right. \\ & - \textstyle\sum_{\substack{j=1 \\ i \neq j}}^{N} \begin{pmatrix} H_{rr} & H_{rz} \\ H_{zr} & H_{zz} \end{pmatrix}_{ij} \begin{pmatrix} \frac{r}{6} \\ -\frac{z}{3} \end{pmatrix}_j \\ & \left. + \textstyle\sum_{j=1}^{M} \begin{pmatrix} G_{rr} & G_{rz} \\ G_{zr} & G_{zz} \end{pmatrix}_{ij} \begin{pmatrix} n_r \\ 0 \end{pmatrix}_j \right) \left(\tfrac{6}{r}\right)_i, \\ & \text{for } i = 1, 2, \cdots, N. \end{aligned} \tag{22}$$

Note, however, that if the point $\mathbf{x}^i$ is on the z–axis, r becomes zero and so it is impossible to calculate these diagonal terms from the above equation. Fortunately, there is no need to calculate these diagonal terms when the load point is on the z–axis, because the radial velocity and stress at the z–axis must be zero for all axi-symmetric problems. That means that if the load point is on the z-axis then its diagonal terms in the radial direction have no influence on the overall system of linear algebraic equations, and so we can give any non-zero values to them.

The final thing we need to consider is the application of the boundary conditions. If the stress vector is given at a boundary point then nothing has to be done; if the velocity vector is prescribed at boundary point $\mathbf{x}^i$ then we have to interchange the i–th column of $\mathbf{H}$ and the i–th column of $\mathbf{G}$ after reversing its sign. The mixed boundary condition, by which we mean a combination of the velocity and the stress vector is prescribed at a boundary point, can also be handled in a similar way. After this kind of rearrangement we have

$$\mathbf{H}^* \mathcal{Z}^{unknown} = \mathbf{G}^* \mathcal{Z}^{known} \tag{23}$$

where $\mathcal{Z}^{unknown}$ and $\mathcal{Z}^{known}$ stand for unknown boundary data and known boundary data. This is solved by Gaussian elimination.

2.3 Mesh Redistribution

In this problem, the shape of the mould and the plunger is given by a set of discrete coordinates in the (r, z) plane. Furthermore, after one time step, the free boundary is only known at a set of boundary nodes. An algorithm is needed to remesh the boundary of the glass domain.

We assume that the boundary Γ of the fluid region can be parameterized with respect to the arc length s, i.e.

$$\mathbf{x}(s) \in \Gamma, \quad 0 \leq s \leq L.$$

A grid can be described by

$$\{\mathbf{x}(s_0), \cdots, \mathbf{x}(s_{N-1})\},$$

or equivalently by

$$\{s_0, \cdots, s_{N-1}\}.$$

The quantity $h_i = s_i - s_{i-1}$ is called the *step-length*.

Let $\mathbf{x}_{i-1} = \mathbf{x}(s_{i-1})$ and $\mathbf{x}_i = \mathbf{x}(s_i)$ be two given successive nodal points. The next node $\mathbf{x}_{i+1}$ has to lie at a distance h_{i+1} from $\mathbf{x}_i$. Firstly the quasi–uniform condition(cf. Trevelyan[18]) is required, i.e. , for some $A > 0$,

$$\frac{h_i}{A} \leq h_{i+1} \leq A h_i, \text{ for } i = 0, \cdots, N-2. \tag{24}$$

Secondly we need a refined mesh on the free boundary, especially near the plunger; Note that the upper part of the mould has larger curvatures. To approximate this part of the mould boundary we divide the mould boundary into two parts: the upper part and the lower part. On the upper part smaller values for h are needed than on the lower part of the mould boundary. Of course at interfaces–between the mould and the free boundaries, between the plunger and the free boundaries, between the upper and the lower part of the mould boundaries–the condition (24) should be satisfied. A schematic grid is shown in Figure 3.

3 Time Integration

In this section, we discuss how the geometry of the glass is updated. Suppose we are at time t_0. From the Stokes problem we can determine the velocity field $\mathbf{v}^0$ at time t_0, which is used to determine the geometry of the glass in cylindrical coordinates at time $t_1 := t_0 + \Delta t$ by discretising the ordinary differential equation (6). We rewrite this initial value problem as follows

$$\begin{cases} \frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x}(t)), \\ \mathbf{x}(t_0) = \mathbf{x}^0. \end{cases} \tag{25}$$

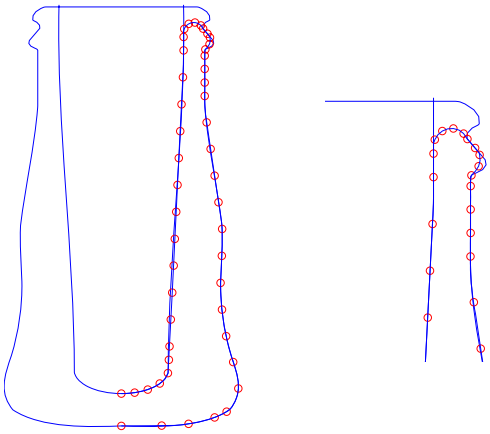

Figure 3: A schematic grid

We denote the area between the mould and the plunger at time t by $\mathcal{A}_t$. Of course the glass body at time t should be in $\mathcal{A}_t$ for all t. If the initial geometry of the glass, represented by its boundary points $\mathbf{x}^0$ satisfies

$$\mathbf{x}_i^0 \in \mathcal{A}_{t_0}, \qquad \text{for each} \qquad i,$$

the new geometry of the glass $\mathbf{x}^1$ at time $t_1 := t_0 + \Delta t$ has to satisfy the constraint

$$\mathbf{x}_i^1 \in \mathcal{A}_{t_1}, \qquad \text{for each} \qquad i. \tag{26}$$

Suppose we employ, say the Euler forward scheme, to discretize the problem (25)

$$\mathbf{y}^1 = \mathbf{x}_0 + \Delta t \mathbf{v}_0.$$

In general, constraint (26) will not be satisfied. That means we need a strategy to reposition those points which are not in $\mathcal{A}_{t_1}$: if $\mathbf{y}_i^1 \in \mathcal{A}_{t_1}$ then no further action is needed; otherwise they are redefined to be at the intersection of the boundary of $\mathcal{A}_{t_1}$ and the line which is from $\mathbf{x}_i^0$ to $\mathbf{y}_i^1$ (see Figure 4). We refer to this repositioning step as the *clipping algorithm*. Applying the clipping algorithm to $\mathbf{y}^1$ we obtain the new geometry of the glass at time t_1.

We prefer the mid–point rule as it is *symplectic*(cf. Stuart [16]). For the initial value problem (25) the midpoint rule reads

$$\mathbf{x}^1 = \mathbf{x}^0 + \Delta t \mathbf{v}(\mathbf{x}(t_{\frac{1}{2}})), \tag{27}$$

where $t_{\frac{1}{2}} := t_0 + \frac{\Delta t}{2}$. But the information $\mathbf{v}(\mathbf{x}(t_{\frac{1}{2}}))$ is unknown. To approximate $\mathbf{v}(\mathbf{x}(t_{\frac{1}{2}}))$, we first use the Euler forward scheme with time step $\frac{\Delta t}{2}$ to compute

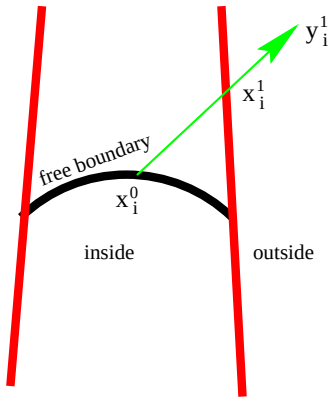

Figure 4: The clipping algorithm

$\mathbf{x}(t_{\frac{1}{2}})$, i.e.

$$\mathbf{x}(t_{\frac{1}{2}}) = \mathbf{x}^0 + \frac{\Delta t}{2}\mathbf{v}^0, \tag{28}$$

then $\mathbf{v}(\mathbf{x}(t_{\frac{1}{2}}))$ is achieved by solving a Stokes flow with a boundary represented by $\mathbf{x}(t_{\frac{1}{2}})$. In summary we have the following stages at each time step

- Solve a Stokes problem on the domain represented by $\mathbf{x}^0$ to get the velocity $\mathbf{v}^0$ at time t_0.
- Use the Euler forward scheme to predict the geometry of the glass $\mathbf{x}^{\frac{1}{2}}$ at time $t_{\frac{1}{2}}$.
- Solve a Stokes problem on the domain represented by $\mathbf{x}^{\frac{1}{2}}$ to predict the velocity $\mathbf{v}^{\frac{1}{2}}$ at time $t_{\frac{1}{2}}$.
- Use the midpoint rule to determine the new geometry of the glass $\mathbf{x}^1$ at time t_1.

Note that if necessary the clipping algorithm should be used before solving the Stokes problem.

4 Numerical Results

To show the efficiency and the accuracy of the method above, we give some examples here.

Some initial shape of the glass drop has to be chosen when the plunger starts moving. The moment the plunger starts moving is taken to be $t = 0$. The plunger

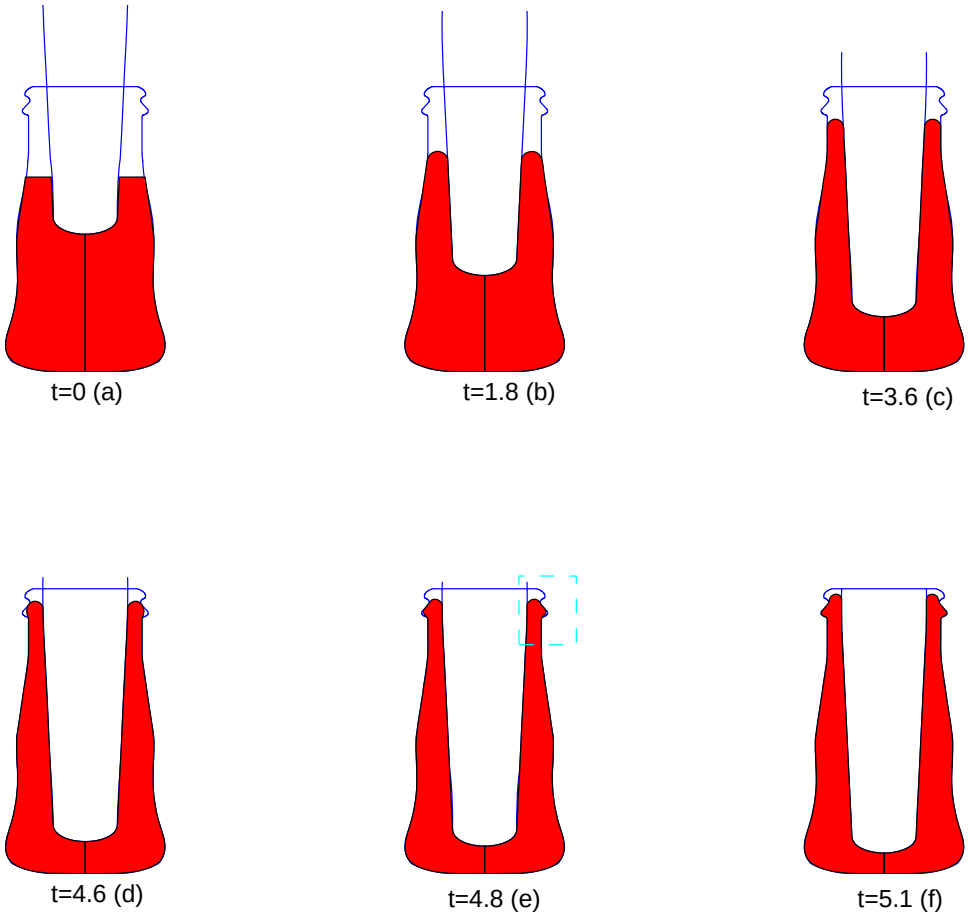

Figure 5: Evolution of the glass domain

is assumed to have a given velocity; the initial position of the plunger is a little bit in the glass drop(see Figure 5a).

The volume of the glass drop is determined by the volume of the parison since the glass is considered incompressible. The evolution of the glass domain is shown in Figure 5.

During the final stage of the pressing phase, one can see several free boundaries which are separated by the mould boundary, see also amplified version of Figure 5e, displayed in Figure 6.

The mass should be conserved. The mass as a function of time t is plotted in Figure 8. In this example only 0.55% of the mass is lost due to the clipping algorithm.

Intuitively, the velocity field near the plunger has larger gradients if the no–slip boundary condition is used. Figure 7 gives an example showing what happens

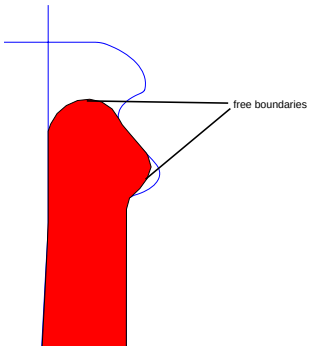

Figure 6: The free boundaries

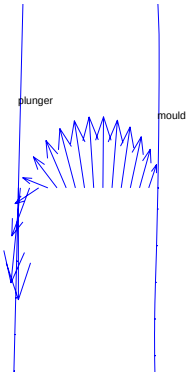

Figure 7: The velocity field on the free boundary

on the free boundary near the plunger. If a FEM is used then a very refined grid is needed. The BEM has the same problem on the free boundary near the plunger, but less serious. That means the BEM is cheaper although the FEM results in a sparse system. The BEM is more efficient in this situation.

References

[1] Becker, A.A., *The Boundary Element Method in Engineering*, McGRAW-HILL BOOK COMPANY, London, 1992.

[2] Becker, A.A., *The Boundary Integral Equation Method in Axisymmetric Stress Analysis Problems*, Springer–Verlag, 1986.

[3] Brebbia, C.A., Telles, J.C.F. & Wrobel, L.C., *Boundary Element Techniques*, Computational Mechanics Publications, Southampton, 1984.

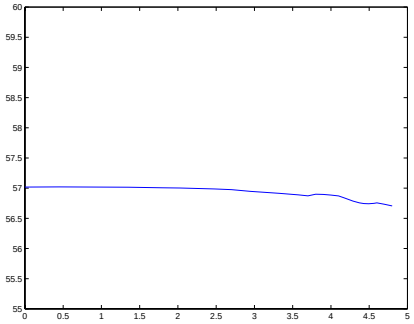

Figure 8: Mass as a function of time

[4] Chandra, T.D. & Rienstra, S.W., *Analytical Approximations to the Viscous Glass Flow Problem in the Mould–Plunger Pressing Process*, RANA 97–08, Eindhoven University of Technology, 1997.

[5] Cody, W.J., *Chebyshev Approximation for the Complete Elliptic Integrals K and E*, Math. of Computation, **19**, pp.105–112, 1965.

[6] Fung, Y.C., *A First Course in Continuum Mechanics*, Prentice–Hall, 1969.

[7] Ladyzhenskaya, O.A., *The Mathematical Theory of Viscous Incompressible Flow*, Gordon and Breach, New York–London, 1963.

[8] Lu, Pin & Huang, Mao–Kuang, *Some Problems in Boundary Element Programming of Axisymmetric Elasticity*, Applied Mathematics and Mechanics, **11**, No.1, pp.45–52, 1990.

[9] Knupp, P. & Steinbeg, S., *Fundamentals of Grid Generation*, CRC Press, 1993

[10] Laevsky, C. & Mattheij, R.M.M., *Development of a Simulation Model for Glass Flow in Bottle and Jar Manufacturing*, RANA 98–18, Eindhoven University of Technology, 1998.

[11] Mattheij, R.M.M. & Molenaar, J., *Ordinary Differential Equations in Theory and Practice*, Wiley, 1996.

[12] Mattheij, R.M.M., Wang, K. & ter Morsche, H.G., *Modelling Glass Parisons*, in this book, 1999.

[13] Partridge, P.W., Brebbia, C.A. & Wrobel, L.C., *The Dual Reciprocity Boundary Element Method*, Computational Mechanics Publications, Southampton, 1992.

[14] Pozrikidis, C., *Boundary Integral and Singularity Methods for Linearized Viscous Flow*, Cambridge University Press, 1992.

[15] Simons, P., *The Cooling of Molten Glass in a Mould*, Technique Report, Eindhoven University of Technology, 1996.

[16] Stuart, A.M. & Humphries, A.R., *Dynamical Systems and Numerical Analysis*, Cambridge University Press, 1996.

[17] Telles, J.C.F., *A Self–Adaptive Co–ordinate Transformation for Efficient Numerical Evaluation of General Boundary Element Integrals*, International Journal for Numerical Methods in Engineering, **24**, pp.959–973, 1987.

[18] Trevelyan, J., *Boundary Elements for Engineers–Theory and applications*, Computational Mechanics Publications, 1994.

[19] van de Vorst, G.A.L. & Mattheij, R.M.M., *A BEM–BDF scheme for Curvature Driven Moving Stokes Flows*, J. Comput. Phys. **No. 1**, 1995.

[20] van de Vorst, G.A.L., *Modelling and Numerical Simulation of Viscous, Sintering*, PH.D. Thesis, Eindhoven University of Technology, 1994.

[21] van de Vorst, G.A.L. & Mattheij, R.M.M., *Numerical analysis of a 2–D viscous sintering problem with non–smooth boundaries*, Computing, **49**, pp.239–263, 1992.

[22] van de Vorst, G.A.L. & Mattheij, R.M.M., *A Boundary Element Solution for Two–dimensional Viscous Sintering*, J. Comput. Phys., **100**, pp.50–63, 1992.