Journal of
Mechanics of
Materials and Structures

REMARKS ON THE ACCURACY OF ALGORITHMS FOR MOTION BY MEAN CURVATURE IN BOUNDED DOMAINS

Simon Cox and Gennady Mishuris

Volume 4, Nº 9

November 2009

mathematical sciences publishers

JOURNAL OF MECHANICS OF MATERIALS AND STRUCTURES
Vol. 4, No. 9, 2009

REMARKS ON THE ACCURACY OF ALGORITHMS FOR MOTION BY MEAN CURVATURE IN BOUNDED DOMAINS

SIMON COX AND GENNADY MISHURIS

Simulations of motion by mean curvature in bounded domains, with applications to bubble motion and grain growth, rely upon boundary conditions that are not necessarily compatible with the equation of motion. Three closed form solutions for the problem exist, governing translation, rotation, and expansion of a single interface, providing the only benchmarks for algorithm verification. We derive new identities for the translation solution. Then we estimate the accuracy of a straightforward algorithm to recover the analytical solution for different values of the velocity V given along the boundary. As expected, for large V the error can reach unacceptable levels especially near the boundary. We discuss factors influencing the accuracy and propose a simple modification of the algorithm which improves the computational accuracy.

1. Introduction

Motion by mean curvature and the dynamics of two-dimensional foams are closely related subjects in the study of materials [Smith 1952]. In the ideal model of the evolution of crystalline grains in a polycrystalline metal, known as normal grain growth, the size of each grain evolves due to the normal motion of each of its boundaries [Weaire and McMurry 1996]. Each boundary has a certain mobility λ, and moves in such a way as to reduce the total perimeter of the pattern. The ideal soap froth is a model of a two-dimensional foam [Weaire and Hutzler 1999], such as the one studied by Bragg and Nye [1947] and which has recently enjoyed a renaissance, in which interface curvature and pressure differences are balanced, again due to minimization of the total perimeter. Both ideal models arise naturally as limits of the following viscous froth model (VFM) [Kern et al. 2004], derived as a force balance per unit length of the interface:

$$\Delta p = \gamma \kappa + \lambda v_n. \tag{1-1}$$

Here Δp is the pressure difference across the interface and γ its surface tension (assumed constant); κ is the local curvature and v_n the velocity normal to the interface.

The aim of this paper is to investigate the accuracy of algorithms for (1-1), and for this reason we restrict ourselves to the simplest case, when $\Delta p = 0$. In the case of grain growth, there are no area (volume in 3D) constraints, and pressure differences between cells are negligible. This limit is also appropriate to ordered (hexagonal) foams and single soap films.

We ask how a solution to the equations of motion by mean curvature can be commensurate with the boundary of the domain. That is, even though we can find an interface *shape* at a particular instant in time, by solving the governing equation, it is not always possible to match this with the imposed

Keywords: motion by mean curvature, grain growth, foam rheology, algorithms, measures of accuracy.
Cox was financially supported by EPSRC (EP/D048397/1, EP/D071127/1).

boundary condition on *velocity*. We then ask how well solutions that satisfy both the governing equation and boundary condition can be calculated numerically.

The most stringent test of a numerical algorithm here is an initial interface shape that is far from satisfying the boundary conditions. It is this that we use for our preliminary numerical tests. We develop a general class of less severe solutions, that can be used to test numerical algorithms, based upon this simple interface shape (rather than the more complicated shapes found in a real foam or metal). We propose three identities which provide measures of accuracy and, further, can be implemented within an existing algorithm to improve its performance.

2. Curvature-driven motion of a bounded interface

2A. ***Problem formulation.*** In vector form, the motion of an interface in the model of ideal grain growth can be described by

$$\boldsymbol{v} = \kappa \boldsymbol{n}, \tag{2-1}$$

where $\boldsymbol{n}$ and $\boldsymbol{s}$ are the normal and tangential unit vectors to the interface (see Figure 1):

$$\boldsymbol{n} = [n_1, n_2], \qquad \boldsymbol{s} = [n_2, -n_1]. \tag{2-2}$$

If the representation of the interface is taken in the form

$$x = x(y, t), \qquad y \in [\underline{y}(t), \overline{y}(t)], \tag{2-3}$$

then the vector components n_1, n_2 are calculated as follows:

$$n_1 = -\frac{dy}{ds} = -\sin\theta = -\frac{1}{\sqrt{1 + (x_y)^2}}, \qquad n_2 = \frac{dx}{ds} = \cos\theta = \frac{x_y}{\sqrt{1 + (x_y)^2}}, \tag{2-4}$$

where $x_y = dx/dy$, $ds = \sqrt{(dx)^2 + (dy)^2}$, and θ is the tangential angle to the interface (see Figure 1). Finally, the vector $\boldsymbol{v} = [v_1, v_2]$ is the instantaneous velocity of the point (x, y) lying on the interface at time t and κ is the curvature of the interface at that point:

$$\kappa = \frac{d\theta}{ds} = \frac{-x_{yy}}{\sqrt{(1 + (x_y)^2)^3}}. \tag{2-5}$$

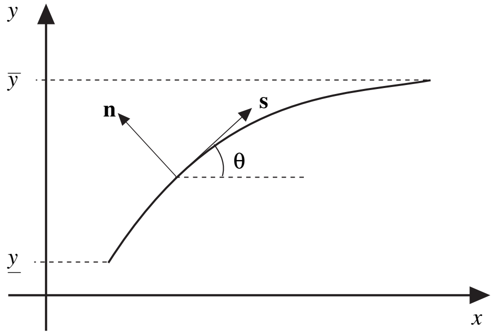

Figure 1. The bounded interface considered here.

Equation (2-1) can also be written in component form:

$$v_n = \boldsymbol{v} \cdot \boldsymbol{n} = v_1 n_1 + v_2 n_2 = \kappa, \tag{2-6}$$

$$v_s = \boldsymbol{v} \cdot \boldsymbol{s} = v_1 n_2 - v_2 n_1 = 0. \tag{2-7}$$

In this paper we will consider only Mullins' translational solution [Mullins 1956] (see below), also known as the grim reaper because of the way in which it scythes through space without change of shape, which is symmetrical with respect to the x-axis. Invariant solutions for rotation have been considered elsewhere [Mullins 1956; Wood 1996]. Taking into account the direction of the interface motion, we can assume that

$$n_1 < 0, \quad n_2 > 0, \quad x_y > 0, \quad 0 < \theta < \pi/2, \quad x_{yy} > 0, \quad \kappa < 0, \tag{2-8}$$

$$v_n < 0 \quad (v_2 < 0,\ v_1 > 0). \tag{2-9}$$

Equation (2-6) is widely discussed in the literature [Mullins 1956; Peleg et al. 2001], while (2-7) is somehow usually forgotten in this context. If one is only interested in reconstructing the interface position at any time step an approach based only on (2-6) is sufficient. However, if it is required that the position of each material point along the interface is controlled, as in the case of numerical computation, then both equations are equally important. Note that there has previously been an attempt to control both the velocity components in a specific way [Green et al. 2006]. Equation (2-7) allows us to find a relation between the two unknown components of the velocity vector $\boldsymbol{v}$ and the normal vector $\boldsymbol{n}$ in the form

$$n_2 = \frac{v_2}{v_1} n_1. \tag{2-10}$$

This allows us to eliminate components of the normal vector $\boldsymbol{n}$ from (2-6) to give

$$-\left(v_1 + \frac{v_2^2}{v_1}\right) = \frac{d\theta}{dy}. \tag{2-11}$$

2B. ***Mullins' solution for translation revisited.*** Let us assume that the interface conserves its shape but moves in the x-direction with a constant speed V. We consider two points A and C having the same y-coordinate $y = y_0$ at two consecutive time steps t_0 and $t_0 + dt$ (see Figure 2). It is clear that these two points correspond to two different material points. Namely, there exists a point B on the interface at

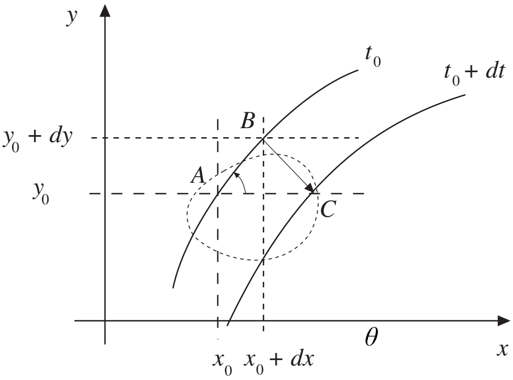

Figure 2. Interface under translational motion at two consecutive instants in time.

time t_0 which moves according to the curvature law (2-1) to the point C for an infinitesimally small time step dt. If the coordinates of the point A are (x_0, y_0) then the coordinates of B and C can be written $(x_0 + dx, y_0 + dy)$ and $(x_0 + V dt, y_0)$.

Note that

$$\overline{BC} = \boldsymbol{v}(x_0 + dx, y_0 + dy)\,dt = \boldsymbol{v}(x_0, y_0)\,dt + O(dt\,ds).$$

On the other hand, $|BC| = V \sin\theta\, dt$, and $\tan\theta = v_1/|v_2|$. As a result one can conclude:

$$V = \frac{1}{v_1(y)}\big(v_1^2(y) + v_2^2(y)\big), \tag{2-12}$$

in some interval $y \in [0, h]$. This relation follows immediately from (2-11) in the case of translation of the interface in the x-direction.

Now, to reconstruct the solution obtained by Mullins [1956] it is sufficient to substitute (2-12) into (2-11) to give

$$\frac{\pi}{2} - Vy = \theta. \tag{2-13}$$

Here we have taken into account the second of the two symmetry conditions at the point $y = 0$:

$$v_2(0) = 0, \qquad \theta(0) = \frac{\pi}{2}. \tag{2-14}$$

Equation (2-13) can be written in the form

$$\cot Vy = y_x, \tag{2-15}$$

which after direct integration leads to Mullins' solution:

$$x(y) = x(0) + Vt - \frac{1}{V}\log\cos(Vy), \qquad y \in [0, h], \tag{2-16}$$

where $x(0)$ is the arbitrary initial position of the centre of the interface. This solution exists only under the condition $h < h_{\max}$, where

$$h_{\max} = \frac{\pi}{2V}. \tag{2-17}$$

Note also that the angle θ defined by such a solution monotonically decreases in the interval $y \in (0, h)$ $(h < h_{\max})$, taking values

$$\theta(y) \in \left(\theta_{\min}, \frac{\pi}{2}\right), \qquad \theta_{\min} = \theta_{\min}(h) = \frac{\pi}{2} - Vh. \tag{2-18}$$

It is now possible to write analytical representations of all problem variables in the interval $y \in (0, h)$:

$$v_1 = V\cos^2 Vy, \quad v_2 = -\tfrac{1}{2}V\sin 2Vy, \quad n_1 = \sin Vy, \quad n_2 = \cos Vy, \quad \kappa = V\cos Vy. \tag{2-19}$$

Note that the first symmetry condition (2-14) has not been used but the reconstructed Mullins solution (2-16) satisfies it automatically, by (2-4) and (2-10). The solution exhibits the following asymptotics near the symmetry axis:

$$x(y) = x(0) + Vt + \tfrac{1}{2}Vy^2 + O(y^4), \quad y \to 0; \tag{2-20}$$

thus near $y = 0$ the interface is close to a parabola. Close to the other end of the reaper (in the case of the maximal thickness $h = h_{\max}$), the following asymptotic estimate can be obtained:

$$x(y) = -\frac{1}{V}\log(h_{\max} - y) + O(1), \quad y \to h_{\max}, \qquad \text{or} \qquad y - h_{\max} \sim -de^{-Vx}, \quad x \to +\infty, \tag{2-21}$$

where $d = (1/V)\exp\{V(x(0) + Vt)\}$ is a positive constant.

Note that relation (2-12) also represents the boundary condition for any moving interface whose upper point lies on the line $y = \overline{y} = h$ and which moves in the x-direction with velocity V. Moreover, the velocity can be in that case a function of time $V = V(t)$:

$$V(t)v_1(h, t) = v_1^2(h, t) + v_2^2(h, t). \tag{2-22}$$

Substituting (2-1) into (2-22) such a boundary condition can be equivalently rewritten in other forms:

$$\kappa(h, t) = V(t)n_1(h, t), \qquad \text{or} \qquad x_{yy}(h, t) = V(t)\big(1 + (x_y(h, t))^2\big). \tag{2-23}$$

Note that we have not used (2-16) to define (2-23).

2C. ***Arbitrary instantaneous solution of equations* (2-6) *and* (2-7) *in a bounded domain.*** Let us consider any instantaneous solution of the equations (2-6) and (2-7) with the prescribed boundary condition (2-22), by which we mean an instantaneous state of the film which may or may not be commensurate with the boundary conditions and could or could not be a steady state. Effectively this means that, for a particular time t, the end points of the interface $y = \underline{y}(t)$ and $y = h$ are defined and the velocity components $v_1(y, t)$ and $v_2(y, t)$ are known functions of the variable y, while the problem is now to determine, using this information, the position of the interface in space variables (y, x).

We introduce a function which in what follows is considered known:

$$F(y) = v_1(y) + \frac{v_2^2(y)}{v_1(y)} > 0, \qquad y \in (\underline{y}, h). \tag{2-24}$$

Note that the condition (2-12) may be not valid at all inside the interval $y \in (\underline{y}, h)$ as it was for Mullins' solution; as a result, $F(y)$ is not a constant, in general. Equation (2-11) can be integrated to give

$$-\int_{\underline{y}}^{y} F(\xi)\,d\xi = \theta(y) - \underline{\theta}. \tag{2-25}$$

Here, recall that F depends upon time t, so that $\underline{y} = \underline{y}(t)$ and the constant of integration is $\underline{\theta} = \underline{\theta}(t)$. Equation (2-25) should be considered together with (2-10) which, in this case, takes the form

$$\frac{dx}{dy} = -\frac{v_2}{v_1} = \cot\theta, \tag{2-26}$$

or

$$x(y) = \underline{x} - \int_{\underline{y}}^{y} \frac{v_2(\xi)}{v_1(\xi)}\,d\xi. \tag{2-27}$$

Equations (2-25) and (2-26) together indicate that the functions v_1 and v_2 cannot be chosen arbitrarily to satisfy the vectorial (2-1) as one might expect. Instead, the following identity has to be satisfied:

$$\arctan\frac{v_1}{v_2} = \int_{\underline{y}}^{y} F(\xi)\,d\xi - \underline{\theta}, \tag{2-28}$$

or writing $w = v_1/v_2$ and using (2-24), the derivative of (2-28) becomes

$$\frac{w_y}{1+w^2} = \frac{(1+w^2)v_2}{w}, \tag{2-29}$$

where the subscript y denotes differentiation, which leads to the following identity valid within the entire interval $y \in (\underline{y}, h)$:

$$v_1^2(y) = -v_2^2(y)\left(\frac{1}{2\int_{\underline{y}}^{y} v_2(\xi)\,d\xi + c} + 1\right), \tag{2-30}$$

where the constant of integration clearly depends on time too: $c = c(t) = 1/(1+w^2)|_{y=\underline{y}}$. Note that any solution of equations (2-6) and (2-7) satisfies this additional relation, which makes sense only under the constraint

$$-1 \le 2\int_{\underline{y}}^{y} v_2(\xi)\,d\xi + c \le 0. \tag{2-31}$$

As $0 < \theta < \underline{\theta}$, one can also deduce another condition which has to be true for any admissible velocities:

$$0 \le \int_{\underline{y}}^{y} F(\xi)\,d\xi \le \underline{\theta}. \tag{2-32}$$

Note that the constant $\underline{x}$ in (2-27) is arbitrary (it changes only the position of the interface in the x-direction and does not influence any other variables). To determine the other constants $\underline{\theta}(t)$ and $c(t)$, we need to use the boundary conditions at the ends of the interface. Thus, condition (2-22) together with (2-30) leads to

$$v_1(h) = V(1+c+2I_2), \qquad v_2(h) = -V\sqrt{-(1+c+2I_2)(c+2I_2)}. \tag{2-33}$$

where we have set

$$I_2 \equiv I_2(t) = \int_{\underline{y}(t)}^{h} v_2(\xi)\,d\xi.$$

If the boundary condition on the other end is given in the form

$$\theta(\underline{y}(t)) = \underline{\theta}(t), \tag{2-34}$$

then all the constants have been defined. Such an instantaneous solution, assuming that the functions $v_1(y)$ and $v_2(y)$ satisfy (2-28), conditions (2-33), and restrictions (2-31) and (2-32), can be realized during the interface evolution at some step.

In the case of the symmetrical solution, where both symmetry conditions (2-14) and the additional condition $v_1(0) = W > 0$ have to be satisfied, one can show that

$$c(t) \equiv 0, \qquad v_2(y) \sim -W^2 y, \quad y \to 0. \tag{2-35}$$

Note here that the value $W = W(V, h)$ should be found from the constructed solution and is not an additional (arbitrary or given) constant.

Finally, both restrictions (2-31) and (2-32) should be valid for the symmetrical interface:

$$\int_0^h v_2(\xi)\,d\xi \ge -\frac{1}{2}, \qquad \int_0^h F(\xi)\,d\xi \le \frac{\pi}{2}. \tag{2-36}$$

Note that the tangential angle θ for this solution is a monotonically decreasing function in the interval $y \in (0, h)$ so that

$$\theta(y) \in \left(\theta_{\min}, \frac{\pi}{2}\right), \qquad \theta_{\min} = \frac{\pi}{2} - \int_0^h F(\xi)\,d\xi. \tag{2-37}$$

It is straightforward to see that Mullins' solution (2-19) satisfies all these relationships with $c(t) = 0$, $\underline{\theta}(t) = \pi/2$, and $W = V$, as expected. In the next section, we construct analytical examples of symmetrical instantaneous solutions which are different from Mullins'.

3. A family of symmetrical instantaneous solutions

In this section we present analytical representations of some instantaneous symmetrical solutions for the interface satisfying the same boundary (2-22) and symmetry (2-14) conditions as Mullins' solution. Those solutions are not, generally speaking, steady-state ones. This means that they can be reached at some time step t, given the interface boundary velocity $V(t)$ and the position of the ends $h(t)$, but all these parameters may later change with time. What is extremely interesting about these solutions is that some of them are well-defined for any velocity $V > 0$ and an arbitrary position of the boundary $y = h$. This shows a rich behaviour of possible instantaneous solutions. It is also clear that there is an infinite number of admissible instantaneous solutions. Some of them can be realized during some specific non-steady-state interface motion. For example, *any* instantaneous solution obtained during a numerical computation, for any particular time step, boundary velocity, and topology, has to satisfy all the relations (2-24)–(2-34). This will allow us to use the relations as indicators of the accuracy of computations. Moreover, they could provide a means to improve the accuracy of the algorithms.

(One could also consider the family of arbitrary, not necessarily symmetric, instantaneous solutions, which is even larger than the symmetric case. In fact, the family of symmetrical solutions has one degree of freedom — since $c(t)$ vanishes in this case — and correspondingly one less boundary condition; compare (2-34) and (2-14). In the context of further applications of this result to a given algorithm, where the angle-type boundary condition has to be preserved at the interface intersection point, it is worth mentioning that the boundary condition (2-34) is therefore more important for application than the symmetry condition. On the other hand, symmetrical instantaneous solutions can also be considered a subset of the asymmetric solutions if one considers the interval (h_0, h) instead of $(0, h)$ $(0 < h_0 < h)$. This idea has been exploited previously in [Green et al. 2009].)

3A. ***First example.*** We consider the following simple combination of compatible velocities

$$v_2(y) = -W^2 y, \qquad v_1(y) = W\sqrt{1 - W^2 y^2}, \qquad W = \frac{V}{\sqrt{1 + V^2 h^2}}, \tag{3-1}$$

which satisfy (2-30) with $c = 0$ and $\underline{y} = 0$ and, as a result, can be used to construct a symmetrical instantaneous solution. Here W is the same constant as in (2-35). Natural restrictions (2-36) for the existence of such a solution give the same estimate:

$$W < \frac{1}{h}, \qquad \text{or} \qquad \frac{V}{\sqrt{1 + V^2 h^2}} < \frac{1}{h}, \tag{3-2}$$

which holds true for any values of V and h. The shape of the interface is an ellipse described by (2-27):

$$x(y,t)=x(0,t)-\sqrt{1-W^2y^2}. \tag{3-3}$$

The tangential angle θ for this solution is a monotonically decreasing function in the interval $y\in(0,h)$ and

$$\theta(y)\in\Big(\theta_{\min},\frac{\pi}{2}\Big),\qquad \theta_{\min}=\frac{\pi}{2}-\arcsin(Wh)>0. \tag{3-4}$$

3B. ***Second example.*** We now consider another specific instantaneous solution assuming that $v_1(y)=W<V$. Then the second component of the velocity satisfies the equation

$$\frac{W^2}{v_2^2(y)}=-\frac{1}{2\int_0^y v_2(\xi)\,d\xi}-1. \tag{3-5}$$

To find $v_2(y)$ it is more convenient to return to the differential equation (2-29) rather than working with the nonlinear integral equation (3-5). After integration it takes the form

$$\Phi\Big(\frac{v_2}{W}\Big)=-Wy, \tag{3-6}$$

where the odd function Φ is defined as

$$\Phi(\xi)=\frac{1}{2}\Big(\arctan\xi+\frac{\xi}{1+\xi^2}\Big),\qquad \Phi'(\xi)=\frac{1}{(1+\xi^2)^2}. \tag{3-7}$$

Note that $\Phi:\mathbb{R}_+\to[0,\pi/4)$ is a monotonic function. Moreover, one can easily obtain the constraint $Wh<\pi/4$, which is similar to (2-36) and (3-2). Then the required velocity component v_2 can be found from

$$v_2=-W\Phi^{-1}(Wy), \tag{3-8}$$

and we can finally find the complete solution using (2-26):

$$x(y,t)=x(0,t)+\frac{1}{W}\int_0^{Wy}\Phi^{-1}(\xi)\,d\xi. \tag{3-9}$$

Finally, note that the tangential angle θ for this solution is a monotonically decreasing function in the interval $y\in(0,h)$:

$$\theta(y)\in\Big(\theta_{\min},\frac{\pi}{2}\Big),\qquad \theta_{\min}=\frac{\pi}{2}-\int_0^{Wh}\Big(1+\big(\Phi^{-1}(\xi)\big)^2\Big)\,d\xi. \tag{3-10}$$

It remains only to find possible values of the unknown constant W in order to satisfy the boundary condition (2-22). The relevant equation takes the form

$$\Phi^{-1}(Wh)=\sqrt{\frac{Vh}{Wh}-1}. \tag{3-11}$$

This equation has the unique solution $W=W_*(V,h)<V$. In fact, the left-hand side is an increasing function from zero to infinity as $Wh\to\pi/4$, whereas the right-hand side is a decreasing function taking values between ∞ when $Wh\to 0$ and 0 when $Wh\to Vh$. Additionally one can conclude from this that $Wh<\min\{Vh,\pi/4\}$, so the restriction defined after (3-7) always holds. In other words, this solution, as well as that of the first example, is well-defined for arbitrary velocity V and position of the boundary $y = h$. We can also show that $\theta_{\min}$ is always positive:

$$\theta_{\min} > \frac{\pi}{2} - \int_0^{\pi/4} \bigl(1 + \bigl(\Phi^{-1}(\xi)\bigr)^2\bigr)\,d\xi = \frac{\pi}{2} - \int_0^{\infty} (1+\eta^2)\Phi'(\eta)\,d\eta = \frac{\pi}{2} - \int_0^{\infty} \frac{d\eta}{1+\eta^2} = 0. \tag{3-12}$$

It is interesting to note that in the case $V \ll 1$ both of the instantaneous solutions constructed above coincide with Mullins' solution to within an accuracy of $O(V^2)$ for any fixed value of h. On the other hand, in this case the solution is practically (with the same accuracy) a straight line (or at the next order of accuracy, a parabola).

3C. ***General case.*** The previous example indicates how to build a wider class of symmetrical instantaneous solutions. Let us introduce the set $\mathfrak{A} \subset C^2([-a,a])$ $(a > 0)$ of even functions ψ satisfying the following four conditions:

$$\psi(\xi) = \frac{1}{2}\xi^2 + O(\xi^4),\ \ \xi \to 0; \qquad \psi(a) \le \frac{1}{2}; \qquad \psi' > 0; \qquad \left(\frac{\psi'}{\sqrt{\psi(1-2\psi)}}\right)' \ge 0,\ \ \xi \in (0,a). \tag{3-13}$$

Note that a may differ from function to function, but it is necessary that for every function there exists some $a > 0$ for which all four conditions hold.

For example, the following three functions belong to the set $\mathfrak{A}$:

$$\psi_1(\xi) = \frac{1}{2}\sin^2\xi, \qquad \psi_2(\xi) = \frac{1}{2}\xi^2, \qquad \psi_3(\xi) = \int_0^{\xi} \Phi^{-1}(\zeta)\,d\zeta = \frac{1}{2}\left(1 - \frac{1}{1+\bigl(\Phi^{-1}(\xi)\bigr)^2}\right), \tag{3-14}$$

with $a = \pi/2$, 1, and $\pi/4$, respectively. These three functions have been collected from Mullins' solution and the two previous examples. Thus, the set $\mathfrak{A}$ is not empty.

Using any function from this set we can construct a symmetrical instantaneous solution with velocity components in the form

$$v_2(y) = -W\psi'(Wy), \qquad v_1(y) = W\psi'(Wy)\sqrt{\frac{1-2\psi(Wy)}{2\psi(Wy)}}, \tag{3-15}$$

that identically satisfies (2-30) with $c = 0$. Then the unknown constant W should be taken to be of the form $W = W_*(Vh)/h$ where $W_*(Vh) > 0$ is a solution of the implicit equation

$$\frac{\psi'(W_*)}{\sqrt{2\psi(W_*)(1-2\psi(W_*))}} = \frac{Vh}{W_*}, \tag{3-16}$$

which follows from (2-22).

Because of the last condition in (3-13), there may exist only one solution of this equation. If, in addition, the left-hand side of (3-16) tends to infinity as $W_* \to a$, then the solution always exists and $W_* < a$. However, if the left-hand side of (3-16) tends to a finite value $L_* > 0$ as $W_* \to a$, then the solution exists only under the additional condition

$$Vh < L_* a. \tag{3-17}$$

One can check that for Mullins' solution $L_* = 1$ and (3-17) coincides with (2-17). For the other two cases previously discussed above, we have $L_* = \infty$ so the solution of the implicit equation (3-16) always exists and no solvability condition (3-17) is needed in these cases.

To reconstruct the complete symmetrical instantaneous solution based on (3-15) it is enough to substitute it in (2-24), (2-25), and (2-27).

Note that in the case $V \ll 1$, the solution to (3-16) gives $W_* \sim V$, as one can conclude from the first part of (3-13). This means that any constructed instantaneous solution differs negligibly from the Mullins' solution for small values of the velocity V.

It remains to investigate two important constraints (2-36). Taking into account that

$$\int_0^h \upsilon_2(\xi)\,d\xi = -\psi(Wh), \qquad \int_0^h F(\xi)\,d\xi = \frac{1}{2}\Big(\arcsin\big(4\psi(Wh)-1\big)+\frac{\pi}{2}\Big),$$

then the two constraints (2-36) are equivalent in this case and correspond to $\psi(Wh) \le 1/2$, which coincides with the second part of (3-13).

In fact, the third condition, $\psi' > 0$, from (3-13) is not required: it guarantees that the instantaneous solution is convex but without it we can construct nonconvex interfaces.

4. Numerical simulations

To indicate the computational inaccuracy of the algorithms, we discuss Mullins' solution for a symmetrical reaper, for which all quantities are known in closed form (see (2-13) and remarks thereafter), and compare it with the result of a numerical computation using a simple algorithm, implemented in the Surface Evolver [Brakke 1992]. This takes the form of a single interface separating two cells of equal pressure being pulled at a velocity V at each boundary (see Figure 3).

The numerical procedure can be briefly described as follows. We start from a straight (vertical) line joining the two walls a distance $2h = 2$ apart. This is subdivided into 2^5 short elements (edges which

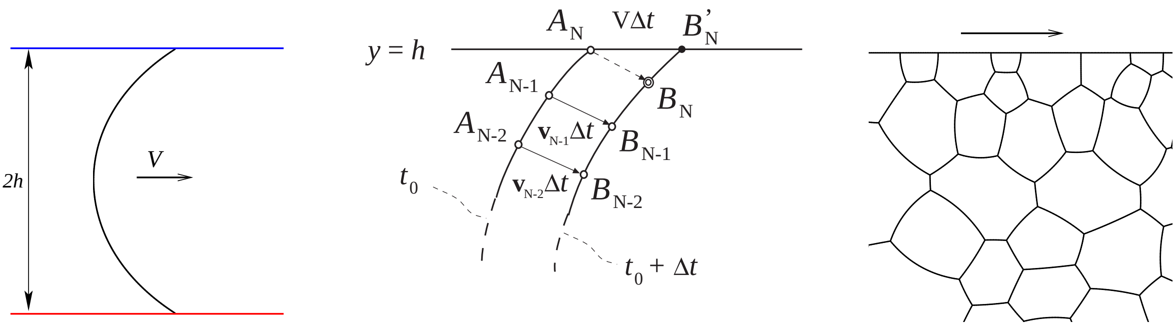

Figure 3. Left: The test problem considered here consists of a single interface that is sheared symmetrically by translating the boundaries. The shape should correspond to Mullins' solution. Middle: Standard algorithm for implementation of the boundary condition at each time step Δt. A_j and B_j are the respective material points on the interface at times t_0 and $t_0 + \Delta t$. Right: Example of a multi-bubble foam simulation, for which the numerical procedure developed here will be of use.

meet at points). A time step $\Delta t = 1 \times 10^{-5}$ is chosen for the computations and bounds on the possible length L of each edge $(0.01 \leq L \leq 0.05)$. The algorithm proceeds as follows:

(i) Each boundary point is moved in the x-direction a distance $V \Delta t$.

(ii) The curvature at each point that is not on the boundary is calculated from $\kappa = \boldsymbol{F} \cdot \boldsymbol{n}/\bar{L}$, where $\bar{L}$ is the average length of the neighbouring edges, $\boldsymbol{F}$ is the negative energy (perimeter) gradient [Brakke 1992], and $\boldsymbol{n}$ is the normal to the line joining the other ends of the neighbouring edges.

(iii) Each point that is not on the boundary is then moved according to $\Delta \boldsymbol{x} = \Delta t \kappa \boldsymbol{n}$ (see Figure 3, middle).

This procedure (one step) is repeated until the difference in the x component of velocity between the centre-point of the interface and the boundary is less than a critical value (1×10^{-8}). Every 20 steps we check the edge-length bounds and add or remove edges as necessary. Thus, each tessellation point corresponds to the same material point within the step; nonetheless from step to step the algorithm may use different material points because of the refinement of the tessellation. Note that this standard algorithm preserves a reasonable restriction on the length of the edges, mainly following the initial distribution of material points; however, it works in a way that does not guarantee equal-length edges.

Note that this choice of the parameters for numerical simulation is standard and allows us to obtain acceptable accuracy in reasonable computational time [Cox 2005]. On the other hand, when one computes the dynamics of foams with many bubbles, the total computational error accumulates. Therefore information about the error is crucial, since it gives us a lower bound for the total computational error.

Two important observations illustrating the weakness of the algorithm should be noted here:

- The density of the tessellation points near the symmetry axis $(y = 0)$ increases with each time step (see Figure 3, middle). Since the time step is constant, this may lead to failure of the stability condition for the linearized finite difference (FD) scheme applied to the nonlinear parabolic equation (2-1) due to this algorithm.
- The opposite effect occurs near the external boundary $y = h$. However, the situation here is even worse. In fact, there is not enough information to reconstruct the curvature and the unit vector at a point A_N lying on the boundary, and the algorithm, in fact, simply eliminates it. It creates instead the point B'_N along the boundary which should be the next point B_{N+1} (see again Figure 3, middle). See [Green et al. 2006] for more details.

We stress that the computations were stable (the stable steady-state regime has been reached) for every value of the external velocity V under consideration. Note that in our computations at high V, the number of tessellation points has increased from 2^5 to about 220 at the steady state. As expected, the worst situation in the sense of computational time, as shown in Figure 4, occurs for the largest value of the velocity, $V = 1.560796$, which is slightly less than the critical value $V_{\text{cr}} = \pi/(2h)$ predicted by the analytical solution [Mullins 1956]. The algorithm could not reach the steady-state regime at all for $V > 1.560796$; physically this is because films are being stretched indefinitely until they burst, which is manifested in the computations by the interface developing a branch lying outside the external boundary $y = h$. All this illustrates that the existing algorithm is well organized and works according to expectations but it is naturally sensitive to the value of the boundary velocity V. Thus it makes sense to ask about algorithm accuracy for a specific velocity versus space and time steps.

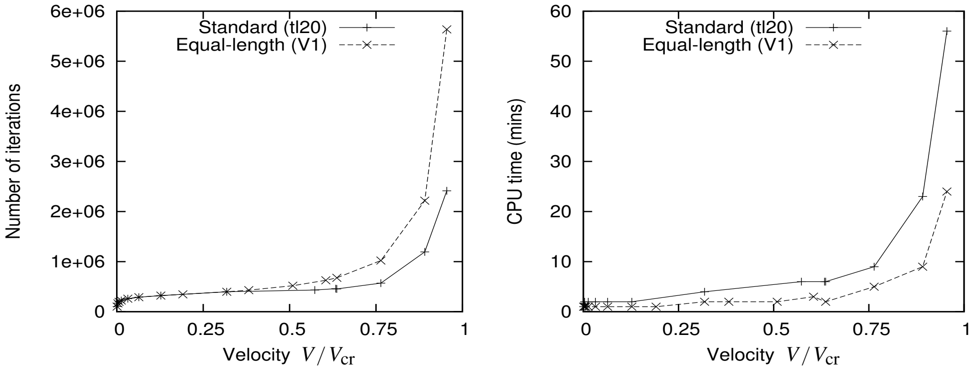

Figure 4. The number of iterations and the computational time both increase as the driving velocity approaches the critical value. The equal-length algorithm requires more iterations to converge, but does so in a faster time. The calculations were performed on a 2.66 GHz desktop PC; error bars are ±1 minute.

As the exact analytical solution to the Mullins' problem is known, we can estimate errors in the computations for all the physical and geometrical quantities: position of the interface $x(y)$, curvature $\kappa(y)$, and the velocity components $v_1(y)$ and $v_2(y)$. Corresponding relative errors for all solution parameters are presented in Figure 5 for different applied velocities.

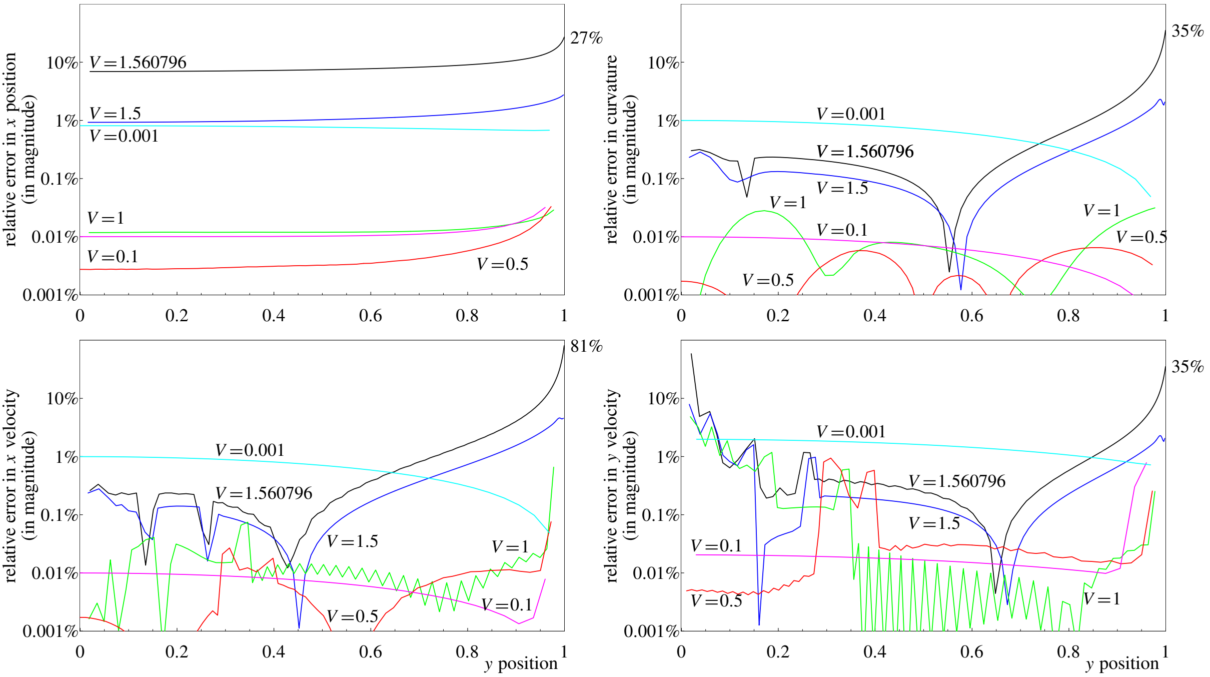

Figure 5. Relative error, compared to Mullins' solution (2-16) and (2-19), in the position $x(y)$ of the interface (top left), its curvature κ (top right), and the x- and y- velocity components v_1 and v_2 (bottom row), for different velocities V of the external boundary.

As expected, the least accurate solutions are those for the largest velocity $V = 1.560796$. The error can be as large as 90% near the boundary ($y = h = 1$) in the x component of velocity v_1 and 60% near the symmetry axis for the y component. The former error is due to the fact that the interface is almost parallel to the boundary, while the latter is naturally related to the fact that the value of the velocity is equal to zero at the symmetry point. However, the drastic difference in value for this *numerical noise*, and its distance from the axis, indicates that at the larger values of V it exceeds reasonable expectations and is really related to the computational accuracy.

One can also consider that the error near the external boundary is due to parametrization of the numerical solution in y rather than x, but the standard numerical algorithm tries to preserve edge lengths. Moreover, the algorithm introduces new points in a regular fashion.

Thus both the errors (near the interface ends) are a consequence of the phenomena discussed in the two observations above. As V decreases, the accuracy increases for given bounds on the edge lengths L.

At first glance, it would appear that the position of the interface, $x(y)$, should be computed with better accuracy than all other solution parameters, which are, in fact, the results of some derivative procedure. However, our computations show that this is not the case and the relative error for $x(y)$ varies from 6% to 28% for the velocity $V = 1.560796$ while the curvature error is lower. Note also that the error for smaller velocities reaches a few percent near the boundary or symmetry axis.

The maximal absolute errors for all solution parameters mostly appear near the external boundary $y = h(= 1)$. This highlights that the implementation of the boundary condition in the existing algorithm cannot be considered as sufficient and should be improved.

Moreover, in the case of Mullins' solution an additional simple local indicator defined by identity (2-12) (independent of the integration of the solution variables) could equally be considered. It is clear from the results presented in Figure 6, left, that the error in this condition is not localized near the ends of the interface, as one might expect from the above. This so-called *internal* error is present for all values of V and is comparable with that near the interface ends.

To investigate accurately this internal error we repeated the computations for a specific velocity $V = 1$ and decreased both the time step, Δt, and the minimum edge length, $L_{\min}$ (see Figure 6, right). This

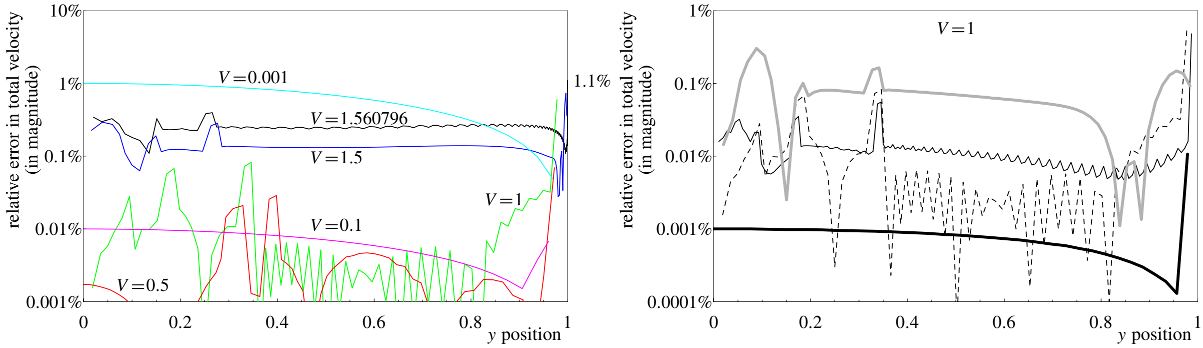

Figure 6. Relative error for the characteristic relation (2-12) for varying velocity V (left diagram) and varying numerical parameters Δt, L (right diagram: thick black line = equal length; dashed line = standard algorithm; thin black line = halve timestep and minimum length; grey line = decrease timestep by factor of 10). The best accuracy is obtained by keeping the line segments of equal length.

has improved the quality of the computations, but there is still an error at some internal points of the interval that is comparable with the error near the ends. Unfortunately the computational time increases considerably. The obvious route of decreasing Δt but fixing $L_{\min}$, to ameliorate this effect, leads to greater error in the solution (see again Figure 6, right). The accuracy of the solution can be improved internally by making the line segments of equal length [Green et al. 2006]; although this doesn't affect the error at the boundary, it does make the calculations faster (see Figure 4).

Possible sources of internal error include: (a) nonoptimal distribution of the tessellation points along the interface after some time; (b) imperfections in the correction procedure (which adds and eliminates points from the interface at some prescribed time); and (c) point-to-point error variation related to the fact that the diffusion-type coefficient changes from point to point along the interface (recall that (2-1) is a nonlinear parabolic equation which is solved by a direct FD scheme with a fixed time step).

In the last computation in Figure 6, right, for the line segments of equal length, we have redistributed points to make the segments equal at every time step (note that this length may change in time). Apart from the fact that the number of tessellation points is smaller than for the standard algorithm, such a comparison is not absolutely fair as the redistribution in the standard algorithm (removing or subdividing edges with lengths outside the range $0.01 \le L \le 0.05$) was done every 20 time steps. To discover if there

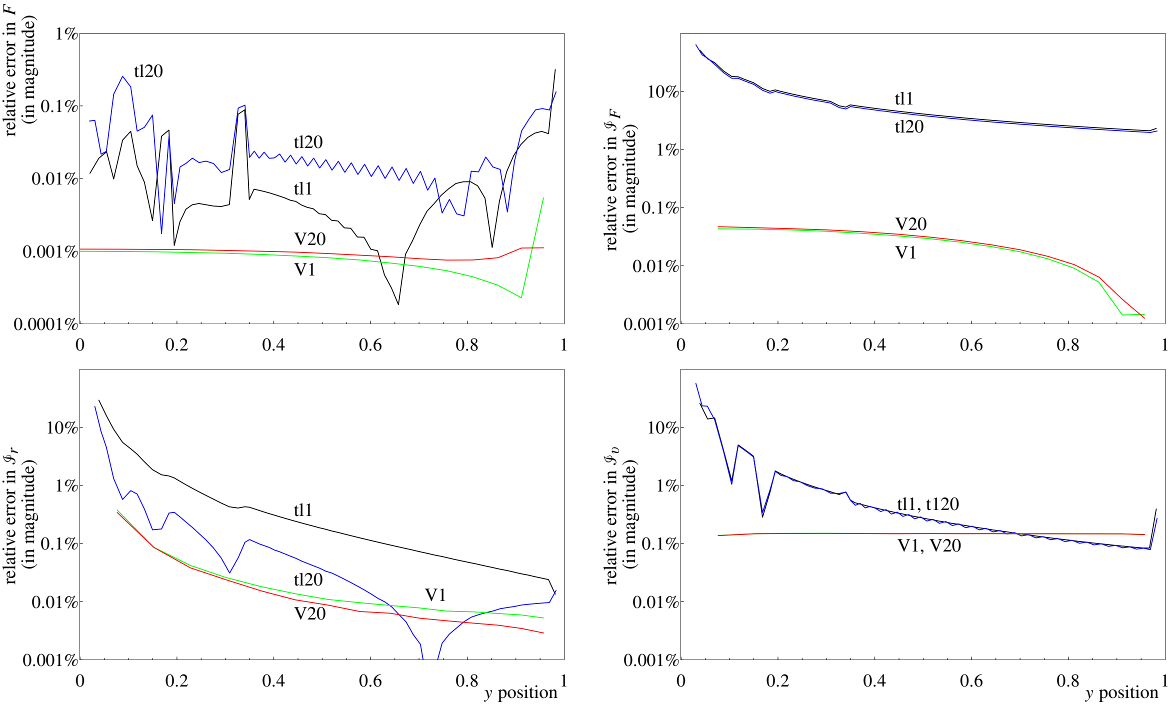

Figure 7. Relative error in the computations shown by the integral measures for unit velocity of the external boundary, $V = 1.0$, obtained with four different computational strategies: the standard redistribution of the tessellation points, tl1 and tl20; and uniform length segment redistribution, V1 and V20, at every time step and every twentieth step, respectively. The top left graph shows the function F of (2-24) (as in Figure 6); the other three graphs correspond to the three error indicators defined in (4-1)–(4-3). All integrals were computed with the trapezium rule.

is an effect of redistribution frequency on the accuracy, we have tested these two algorithms under the same strategy by redistributing the points (in a different way) every time step and every twentieth time step. The error in the function F defined in (2-24), which is a constant in the case of the Mullins solution, are presented in Figure 7, top left. It is evident that the standard algorithm is quite sensitive to the chosen strategy. For a given position of the points on the interface, the relative error may differ by as much as two orders of magnitude, whereas this is not the case for the equal segment strategy. In this case, only near the external boundary is there some small fluctuation in the accuracy. Comparing the two redistribution algorithms for the same frequency of redistribution, the largest error always appears in the case of the standard algorithm — by up to two orders of magnitude — despite the fact that the number of tessellation points was greater. Moreover, in the standard algorithm this error is irregularly distributed along the interface. Recall that the number of tessellation points in the standard algorithm changes during the computations from 2^5 initially to around 150 (for $V = 1$) in the steady-state regime, while the number of the points in the second (equal length) algorithm remains constant. Therefore the computational time for the second algorithm was less by a factor of approximately two. On the other hand, the difference in the computational time between the different frequencies for redistribution for the equal length algorithm was only a few percent. This indicates the further possibility of optimizing this algorithm by redistributing points every M_T time steps. It is clear that $M_T = M_T(V)$ and this needs further investigation [Green et al. 2006].

Note that the function F from (2-24) can be used as an indicator of the accuracy of the computation only for the Mullins' solution. However, there are three universal indicators which can be helpful to estimate the accuracy for any computations, namely, the relative errors of the numerical representations of the identities (2-25), (2-27), and (2-30), embodied in the following explicit definitions:

$$\text{After (2-25):}\qquad \mathcal{I}_F = -\int_0^y F(\xi)\,d\xi; \qquad \text{relative error} = \frac{\mathcal{I}_F}{\theta(y) - \pi/2} - 1. \tag{4-1}$$

$$\text{After (2-27):}\qquad \mathcal{I}_r = \int_0^y \frac{v_2(\xi)}{v_1(\xi)}\,d\xi; \qquad \text{relative error} = \frac{\mathcal{I}_r}{x(0) - x(y)} - 1. \tag{4-2}$$

$$\text{After (2-30):}\qquad \mathcal{I}_v = \int_0^y v_2(\xi)\,d\xi; \qquad \text{relative error} = \mathcal{I}_r \Big/ \left(\frac{v_2^2(y)}{2(v_2^2(y) + v_1^2(y))} \right) - 1. \tag{4-3}$$

The respective results are shown in the remaining three parts of Figure 7. All four indicators suggest that the equidistant distribution of the tessellation points is much better than the standard algorithm, regardless of the chosen strategy, as recommended in [Green et al. 2006]. Moreover, even near the symmetry point, $y = 0$, where the value of the indicators all tend to zero and have a large influence on the relative errors, the accuracy of the computations for the second algorithm is extremely high. However, this is not the case for the standard algorithm.

In Figure 8, the relative errors of the solution variables are presented for both algorithms: the standard one and the equidistant distribution. The result for $x(y)$, shown in the top left part of the figure, looks surprising at first glance; although the accuracy of the computations performed with these two algorithms is of the same order and the error related to the new algorithm is distributed more uniformly, it appears that the accuracy of the standard algorithm is better than the equal-segment-length algorithm, at least with respect to the accuracy of the position of the interface. However, this is not the case. In fact, as was shown above, the computational error for the standard algorithm is redistributed along the interface irregularly whereas that for the equal-segment algorithm is practically uniform. As a result, the criterion to stop the iteration process to find the steady-state solution works differently for the two algorithms. The prescribed maximal growth 10^{-8} in each time-step, measured on the axis of symmetry, is reached more quickly for the new algorithm. This is the second reason (together with number of tessellation points) why this algorithm is faster. If one were to run both algorithms for the same time, or for the same number of iterations, and compare the corresponding results, the discussed paradox should not appear and the new algorithm always provides better accuracy by as much as two orders of magnitude.

For the accuracy of other problem variables — the interface curvature, κ, and the interface velocities, v_1 and v_2 — the new algorithm is more accurate, notwithstanding the above argument, as can be seen in the last three parts of Figure 8.

Finally, we stress again that the proposed three indicators are more versatile measures than a comparison of the numerical steady-state solution with the analytical one, since the latter comparison includes an additional error related to the determination of the steady-state regime, while the indicators show us accuracy of the solution even if the steady state has not been reached.

5. Discussion and conclusions

All these results clearly indicate that the existing algorithm should be used with caution, especially when investigating the behaviour of a many-bubble foam near the critical velocity. Moreover, when there

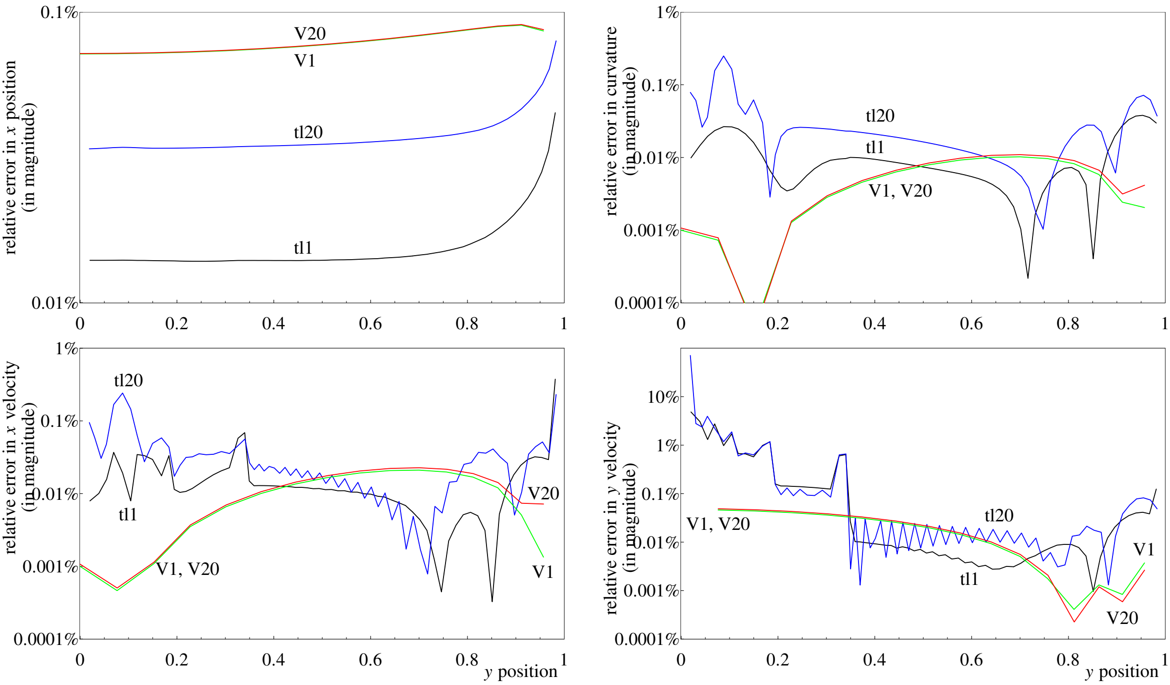

Figure 8. Relative error in the position $x(y)$ of the interface (top left), its curvature κ (top right), and the x- and y- velocity components v_1 and v_2 (bottom row), for different numerical algorithms with velocity of the external boundary $V=1$.

are many cells in a simulation (see Figure 3, right), the user is restricted to some critical number of tessellation points M_P, which gives a limitation on accuracy even for low velocity. In fact, the material structure is highly nonuniform, in the sense that the cells may have different sizes. Effectively this means that every interface has its own critical velocity and the bigger cells thus introduce larger errors. In addition, it takes longer for curvature to diffuse along a longer interface and set it in motion. This creates the following duality: to accurately describe the process of cell motion it is necessary to have the computational error as small as possible, while the error generated near the critical velocity takes its greatest value. This is true for boundary or internal cells equally. Problems requiring high accuracy of the solution near the external boundary are related to the investigation of the boundary effects describing the total phenomenological behaviour of the foam structures.

Another important remark is that the choice of the initial condition for testing the numerical procedure here (a straight-line interface) is much more severe than any of the instantaneous solutions reported in Section 3. One could even think of worse situations to test the algorithm, for example if the initial interface is not convex or even not smooth. This may lead to supercritical velocities and so on.

To revise and improve the existing numerical algorithm, we propose to use another strategy for the redistribution of the tessellation points: an equal-segment-length distribution of the tessellation points is much more favourable [Green et al. 2006]. However, this strategy is not sufficient to eliminate inaccuracy in the computation near the maximum velocity in the steady-state regime. The reason for this is the behaviour of the steady-state solution near the wall (2-21). In fact, there exist two possible realizations of this algorithm. One, which we have used in our computations, is to keep the same number of tessellation points, M_P, then with time the length between consecutive points, L, will increase significantly when V is near $V_{\rm cr}$. This leads to an effective loss of accuracy. Another strategy would be to keep the same distance between points during the computations. However, M_P will then increase to infinity as V approaches $V_{\rm cr}$. Formally this should preserve computational accuracy but will lead to an unacceptable increase in computational time and memory use. Thus, further adaptations to the algorithm are required if high accuracy is required, for example in the steady-state regime with velocities near the critical value [Cox 2005].

Taking advantage of the auxiliary identities (2-25), (2-27), and (2-30), we may correct the instantaneous solution obtained within any algorithm at any or even every time step without time-consuming computations, as the identities are valid for any instantaneous solution. They also make possible further investigation of the asymptotic behaviour of the bounded interface solution near the ends. For example, any possible solution behaves at the symmetry axis according to (2-35), which allows us to tackle the error in the solution near the symmetry axis. On the other hand, the results obtained in Section 2C may allow us to construct and implement a new numerical procedure/elements tackling the boundary condition in a more accurate way (without losing any near-boundary points, using, for example, a boundary layer result [Grassia et al. 2008]).

Finally, as we have shown, the identities (2-25), (2-27), and (2-30) may be used to probe the accuracy of computations. These indicators are extremely helpful as they are not based on information about the exact solution and can therefore illuminate inaccuracy of the numerical solution without any preliminary knowledge about the exact solution itself.

Summarizing, we have shown that an improvement of the numerical algorithm is highly desirable and possible. Apart from the fact that some of the improvements have been indicated and proven in this paper, there is still an open question how to deal with the accuracy of the computations near the critical velocities and near the external boundaries. We have also suggested possible directions for future investigation: improved implementation of the boundary condition and creation of additional near-boundary points. Further, to check new results related to the numerical algorithm we need a larger set of analytical benchmarks.

Acknowledgements

The authors thank P. Grassia for his many helpful remarks on an earlier version, and the unknown referees for their insightful comments.

References

[Bragg and Nye 1947] L. Bragg and J. F. Nye, "A dynamical model of a crystal structure", *Proc. R. Soc. Lond. A* **190**:1023 (1947), 474–481.

[Brakke 1992] K. Brakke, "The Surface Evolver", *Exp. Math.* **1**:2 (1992), 141–165.

[Cox 2005] S. J. Cox, "A viscous froth model for dry foams in the Surface Evolver", *Colloid. Surface. A* **263**:1–3 (2005), 81–89.

[Grassia et al. 2008] P. Grassia, G. Montes-Atenas, L. Lue, and T. E. Green, "A foam film propagating in a confined geometry: analysis via the viscous froth model", *Eur. Phys. J. E* **25**:1 (2008), 39–49.

[Green et al. 2006] T. E. Green, A. Bramley, L. Lue, and P. Grassia, "Viscous froth lens", *Phys. Rev. E* **74**:5 (2006), 051403.

[Green et al. 2009] T. E. Green, P. Grassia, L. Lue, and B. Embley, "Viscous froth model for a bubble staircase structure under rapid applied shear: an analysis of fast flowing foam", *Colloid. Surface. A* **348** (2009), 49–58.

[Kern et al. 2004] N. Kern, D. Weaire, A. Martin, S. Hutzler, and S. J. Cox, "Two-dimensional viscous froth model for foam dynamics", *Phys. Rev. E* **70**:4 (2004), 041411.

[Mullins 1956] W. W. Mullins, "Two-dimensional motion of idealized grain boundaries", *J. Appl. Phys.* **27**:8 (1956), 900–904.

[Peleg et al. 2001] A. Peleg, B. Meerson, and A. Vilenkin, "Area-preserving dynamics of a long slender finger by curvature: a test case for globally conserved phase ordering", *Phys. Rev. E* **63**:6 (2001), 066101.

[Smith 1952] C. S. Smith, "Grain shapes and other metallurgical applications of topology", pp. 65–108 in *Metal interfaces* (Detroit, MI, 1951), American Society for Metals, Cleveland, OH, 1952.

[Weaire and Hutzler 1999] D. Weaire and S. Hutzler, *The physics of foams*, Clarendon, Oxford, 1999.

[Weaire and McMurry 1996] D. Weaire and S. McMurry, "Some fundamentals of grain growth", *Solid State Phys.* **50** (1996), 1–36.

[Wood 1996] G. P. Wood, *Some problems in nonlinear diffusion*, Ph.D. Thesis, University of Nottingham, 1996.

Received 14 Jan 2009. Revised 11 Mar 2009. Accepted 8 May 2009.

SIMON COX: foams@aber.ac.uk
Institute of Mathematics and Physics, Aberystwyth University, Aberystwyth SY23 3BZ, United Kingdom

GENNADY MISHURIS: ggm@aber.ac.uk
Institute of Mathematics and Physics and Wales Institute of Mathematical and Computational Sciences,
Aberystwyth University, Aberystwyth SY23 3BZ, United Kingdom