Flow Simulations on Cartesian Grids involving Complex Moving Geometries

Hans Forrer, Marsha Berger

Abstract. We describe a method to solve the compressible time-dependent Euler equations using Cartesian grids for domains involving fixed or moving geometries. We describe the concept of a mirror flow extrapolation of a given solution over a reflecting wall which may be curved or moving at a fixed or varying speed. We use this mirror flow to develop a Cartesian grid method to treat the cells along a reflecting boundary avoiding the "small-cell" problem. Numerical Results are presented.

1. Introduction

We are developing numerical methods based on Cartesian grids for compressible inviscid time-dependent flows involving complex fixed or moving geometries. Cartesian grids offer great speed, robustness, and flexibility in dealing with complex industrial applications. In addition, they are relatively automated. However, to be able to use Cartesian grids we need to develop a treatment of the irregular boundary cells along reflecting walls of moving or fixed objects.

In previous work by the authors ([3], [6]) other boundary treatments for time-dependent flows described by the Euler equations have been developed. These boundary treatments are stable (without limiting the time step due to the arbitrarily small cut cells), accurate (more than first order along the boundary) and flexible (applicable for any finite-volume method).

Here we describe a numerical method to treat objects moving at a prescribed motion or in interaction with the fluid. As in [1] and in [5] we are using a fixed Cartesian grid and let the object move through it. Many of the same difficulties of accuracy and stability of Cartesian boundary treatments are also common to front tracking algorithms (cf. [2]).

2. Problem Description

The main idea of our approach is to use regular Cartesian grid cells as much as possible. To avoid the "small-cell" problem, we fill in the cut cells and a set of ghost cells, so that cell updates are performed on regular grid cells.

To obtain ghost cell values in our method, the flow is extrapolated beyond the boundary by a mirror flow reflection. The mirror flow is a smooth extrapolation of the flow variables beyond the boundary such that the extrapolated solution fulfills the governing equations.

The Euler equations in two dimensions are given as

$$\mathbf{U}_t + \mathbf{F}_x + \mathbf{G}_y = 0, \tag{1}$$

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho e \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ u(\rho e + p) \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ v(\rho e + p) \end{pmatrix},$$

$$p = (\gamma - 1)(\rho e - \frac{1}{2}\rho(u^2 + v^2)),$$

where ρ is the mass density, $\mathbf{u} = (u, v)^T$ is the velocity vector, ρe is the energy density, p is the pressure and $\gamma = 1.4$.

In the following the one-dimensional mirror flow extrapolation is described. The reflecting wall boundary condition at $x = x_w$ is

$$u(x_w, t) = 0. \tag{2}$$

To describe a mirror flow extrapolation at the wall boundary, we reflect the space coordinate x at x_w

$$\hat{x}(x) = 2x_w - x. \tag{3}$$

A mirror flow extrapolation is then given for $x \leq x_w$ as follows:

$$\begin{pmatrix} \hat{\rho}(x,t) \\ \hat{u}(x,t) \\ \hat{p}(x,t) \end{pmatrix} := \begin{pmatrix} \rho(\hat{x},t) \\ -u(\hat{x},t) \\ p(\hat{x},t) \end{pmatrix}. \tag{4}$$

This mirror flow defines a smooth extrapolation of the solution beyond the wall boundary fulfilling the governing equations.

If the reflecting wall is moving at constant speed, i.e.,

$$x_w = x_w(t), \quad \ddot{x}_w(t) = 0, \tag{5}$$

the boundary condition is

$$u(x_w(t), t) = \dot{x}_w. \tag{6}$$

The reflection of the space coordinate is time-dependent now:

$$\hat{x}(x,t) = 2x_w(t) - x. \tag{7}$$

A solution $\rho(x,t), u(x,t), p(x,t)$ for $x > x_w(t)$ fulfilling the boundary condition (6) can be smoothly extrapolated beyond the boundary by the following mirror flow:

$$\begin{pmatrix} \hat{\rho}(x,t) \\ \hat{u}(x,t) \\ \hat{p}(x,t) \end{pmatrix} := \begin{pmatrix} \rho(\hat{x},t) \\ 2\dot{x}_w - u(\hat{x},t) \\ p(\hat{x},t) \end{pmatrix}. \tag{8}$$

Consider now the case of a wall moving at varying velocity:

$$x_w = x_w(t), \ddot{x}_w \neq 0. \tag{9}$$

The boundary condition is the same as for the wall moving at constant speed, i.e., condition (6). In general a mirror flow extrapolation is only possible in a small neighborhood of the reflecting wall in this case. The solution has a non-zero pressure gradient at the wall according to

$$p_x(x_w(t), t) = -\rho(x_w(t), t)\, \ddot{x}_w(t). \tag{10}$$

Thus particles moving along the wall change their speed due to this pressure gradient. Not only pressure but also density has a non-zero gradient at the wall.

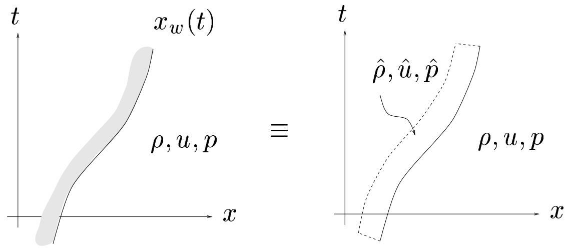

FIGURE 1. Mirror flow for a wall moving with changing speed

In two dimensions the inviscid boundary condition is

$$\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n} = 0, \quad \mathbf{x} \in \Gamma(t), \tag{11}$$

where $\Gamma(t)$ is the reflecting wall and $\mathbf{n}$ is the normal vector at the point $\mathbf{x}$. For a two-dimensional mirror flow extrapolation the normal velocity component u^n is treated as u in one dimension. The other variables as the tangential velocity component u^t, p, and ρ are treated as p and ρ in one dimension. Note that u^t, p, and ρ may have non-zero normal derivatives even at a fixed reflecting wall, namely if the wall curvature is non-zero, i.e.,

$$\frac{\partial p}{\partial n} = -\frac{u^t u^t}{R}, \quad \frac{\partial u^t}{\partial n} = -\frac{u^t}{R}, \tag{12}$$

where n is the normal coordinate and R is the curvature radius of the reflecting wall. The pressure gradient lets the particles move along the wall, the gradient of the tangential velocity is required by zero vorticity along the inviscid wall.

3. Numerical method

Here we describe how we incorporate a boundary treatment into the discretization of the interior flow on a regular Cartesian grid. For the computational results, the boundary treatment is coupled to the Clawpack method of LeVeque [7], a multi-dimensional shock-capturing finite-volume method for describing inviscid flows.

Let h be the grid parameter of a Cartesian grid. Then we set $x_i = x_0 + ih, \quad y_j = y_0 + jh, \quad i, j \in \mathbb{Z}$. The regular Cartesian grid cell C_{ij} is then given by:

$$C_{ij} = [x_i, x_{i+1}] \times [y_j, y_{j+1}]\,. \tag{13}$$

The numerical solution at time t_n is given by approximations of the cell averages of the exact solution $\mathbf{U}(x, y, t_n)$ over the grid cells:

$$\mathbf{U}^n_{ij} \approx \frac{1}{h^2} \int_{C_{ij}} \mathbf{U}(x, y, t_n) dx\, dy. \tag{14}$$

This numerical solution can then be updated using the Euler equations in integral form

$$\mathbf{U}^{n+1}_{ij} = \mathbf{U}^n_{ij} - \frac{\Delta t}{h} (\mathbf{F}^n_{i+1,j} - \mathbf{F}^n_{ij} + \mathbf{G}^n_{i,j+1} - \mathbf{G}^n_{ij}), \tag{15}$$

where we calculate the fluxes $\mathbf{F}_{ij}, \mathbf{G}_{ij}$ using Clawpack.

If at time t_n there is a reflecting wall along $\Gamma(t_n)$ going through the Cartesian grid, we divide the cells into regular cells, boundary cells and empty cells. Figure 2 on the left shows the regular and the boundary cells – all other cells are empty cells. The exact solution at time t_n can be extrapolated over the reflecting wall such that we can assign a numerical value also for boundary cells. If at time t_n the Cartesian grid cell C is a boundary cell, we split this cell into two parts, C^1 is the part lying in the fluid domain and C^2 is the part lying on the solid domain as sketched in Figure 2 on the right. The numerical solution for such a boundary cell C at time t_n is then given by $\mathbf{U}^n_C$ approximating the following cell average

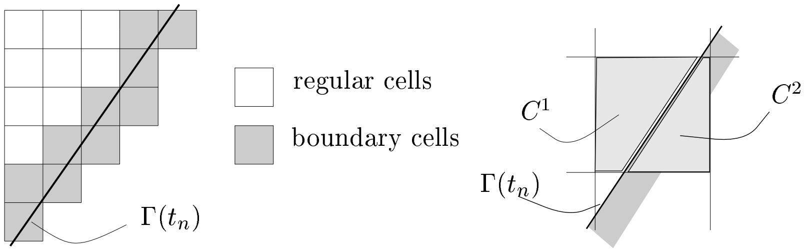

FIGURE 2. Different types of grid cells at time t_n (left), splitting of a boundary cell (right)

$$\mathbf{U}^n_C \approx \frac{1}{h^2} \left(\int_{C^2} \hat{\mathbf{U}}(x, y, t_n)\, dx\, dy + \int_{C^1} \mathbf{U}(x, y, t_n)\, dx\, dy \right). \tag{16}$$

Here $\hat{\mathbf{U}}(x, y, t_n)$ denotes the exact mirror flow solution at time t_n. For the regular cells the numerical solution is given by (14).

To advance the numerical solution of the regular cells and the boundary cells using (15), we fill in a set of ghost cells next to the boundary cells. These ghost cells C_{kl} are filled with flow variables $\hat{\mathbf{U}}^n_{kl}$ using a numerical mirror flow extrapolation of the numerical solution at time t_n. How many ghost cells are needed depends on the specific numerical method. Two ghost cells are needed for the second-order accurate Clawpack method, as shown in Figure 3 on the left.

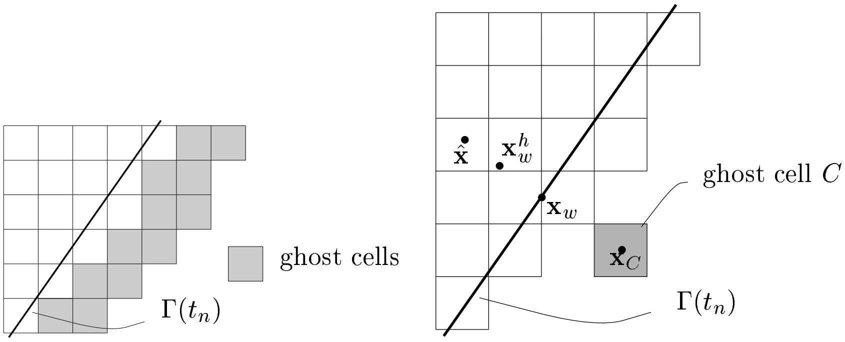

FIGURE 3. Two layers of ghost cells for a second-order accurate method (left), a ghost cell C lying beyond a reflecting wall (right)

In the following we describe how to calculate ghost cell values using a mirror flow extrapolation of the numerical solution. Suppose at time t_n there is a ghost cell C beyond a reflecting wall along $\Gamma(t_n)$ which may be moving (cf. Figure 3 on the right). The midpoint of cell C is denoted as $\mathbf{x}_C$. The point $\mathbf{x}_w$ is the point on $\Gamma(t_n)$ closest to $\mathbf{x}_C$. The point $\hat{\mathbf{x}}$ is the reflection of $\mathbf{x}_C$, i.e.,

$$\hat{\mathbf{x}} = 2\mathbf{x}_w - \mathbf{x}_C. \tag{17}$$

The normal vector $\mathbf{n}$ on $\Gamma(t_n)$ through $\mathbf{x}_w$ is given by

$$\mathbf{n} = \frac{\mathbf{x}_w - \mathbf{x}_C}{|\mathbf{x}_w - \mathbf{x}_C|}. \tag{18}$$

The flow variables of the ghost cell C are given by the pressure $\hat{p}$, the density $\hat{\rho}$, and the normal and tangential velocity components $\hat{u}^n$, $\hat{u}^t$ with respect to the $\mathbf{n}$ direction. The reflected point $\hat{\mathbf{x}}$ is lying between 4 cell centers of regular or boundary cells, such that $u^n(\hat{\mathbf{x}})$ can be obtained using a bilinear interpolation of the normal velocity at $\hat{\mathbf{x}}$. With $\dot{x}_w$ being the normal wall velocity at $\mathbf{x}_w$, the normal velocity of the ghost cell is obtained using

$$\hat{u}^n(\mathbf{x}_C) = 2\dot{x}_w - u^n(\hat{\mathbf{x}}). \tag{19}$$

We experimented with two different strategies to obtain the other ghost cell variables (as described only for pressure in the following). One is to use the bilinearly-interpolated values at the reflected point $\hat{\mathbf{x}}$, i.e.,

$$\hat{p}(\mathbf{x}_C) = p(\hat{\mathbf{x}}). \tag{20}$$

This is simpler than the volume weighted averaging used in the h-box method [3], and yields only a first-order boundary treatment for curved reflecting walls or walls moving with varying speed.

Another strategy is to extrapolate the corresponding values from the nearest boundary point $\mathbf{x}_w$. If $\mathbf{x}_w$ is lying between 4 cell centers of regular or boundary cells, a value $p(\mathbf{x}_w)$ is obtained using a bilinear interpolation, else $p(\mathbf{x}_w)$ is obtained by a linear interpolation from the nearest boundary cell center using finite-differences. By a bilinear interpolation we obtain also a value at the point $\mathbf{x}_w^h$ (cf. Figure 3, right), where

$$\mathbf{x}_w^h = \mathbf{x}_w + h\mathbf{n}. \tag{21}$$

Then the corresponding ghost cell values are obtained by

$$\hat{p}(\mathbf{x}_C) = p(\mathbf{x}_w) + |\mathbf{x}_w - \mathbf{x}_C| \frac{p(\mathbf{x}_w) - p(\mathbf{x}_w^h)}{h}. \tag{22}$$

Note that the boundary cells themselves can be updated using (15), so they do not need repeated application of (16); however the price of this is a lack of conservation (cf. [6] for more details).

4. Numerical results

First we look at a one-dimensional test case, namely a gas confined between two reflecting walls at $x_l = 0.5 + v_l t + \frac{a_l}{2}t^2$ and $x_r = 1.0$, with constants v_l and a_l. The initial conditions are

$$\begin{aligned} \rho(x,0) &= 1.0 + 0.2\cos(\pi\frac{x-0.5}{0.5}), && (23)\\ v(x,0) &= 2.0(1.0-x)v_l, && (24)\\ p(x,0) &= \rho(x,0)^\gamma, && (25) \end{aligned}$$

such that entropy is constant, i.e., $s(x,0) = p(x,0)/rho(x,0)^\gamma = 1.0$. As long as the solution stays smooth, the entropy stays constant, such that we can use this variable for a numerical error analysis. In the numerical experiment we study the following quantities at the final time t_e:

$$\begin{aligned} err_{bry} &= |s_{i_w}^{final} - 1.0|, && (26)\\ err_{tot} &= \frac{\sum_i |s_i^{final} - 1|\,|C_i|}{\sum_i |C_i|}, && (27)\\ \Delta m &= \left(\sum_i \rho_i^{initial}|C_i| - \sum_i \rho_i^{final}|C_i|\right) / \sum_i \rho_i^{initial}|C_i|, && (28) \end{aligned}$$

where i_w is the index of the left boundary cell. The cell volume $|C_i|$ is h for regular cells, the length in the fluid domain for the boundary cells and zero for empty cells.

For the first test case we set: $v0 = -0.5, a0 = 0.0, t_e = 0.5$. The results in Table 1 for err_{bry} suggest that the boundary treatment is only first-order accurate

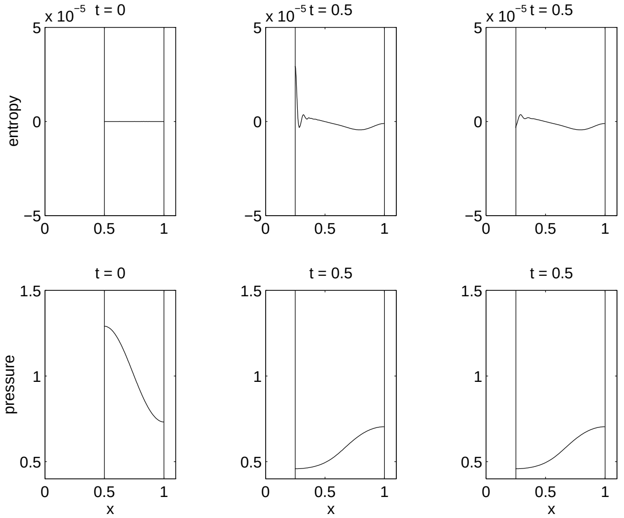

FIGURE 4. Wall moving with constant speed – ghost cell extrapolation using (20) (middle) or (22) (right), $h = 0.005$

	extrapolation using (20)			extrapolation using (22)		
$1/h$	Δm	err_{bry}	err_{tot}	Δm	err_{bry}	err_{tot}
200	4.27-6	2.93-5	2.00-6	3.19-5	3.35-6	1.67-6
400	9.71-7	1.16-5	5.31-7	8.08-6	7.31-7	4.14-7
800	2.39-7	5.57-6	1.41-7	2.04-6	1.70-7	1.03-7

TABLE 1. Wall moving with constant speed – error analysis using the two different ghost cell extrapolations

if pressure and density are extrapolated using (20) in this case of a wall moving with constant speed, whereas using (22) the err_{bry} values suggest second-order accuracy also at the boundary in this case. But the lack of conservation is smaller using (20). Figure 4 shows the results for $h = 0.005$.

For the second test case we set: $v0 = 0.0, a0 = -2.0, t_e = 0.5$. The results in Table 2 suggest that for this second test case of a wall moving at varying speed, using (22) for the pressure and density extrapolation yields a second-order accurate boundary treatment.

For the following two-dimensional test cases we are using (22) for the ghost cell values but in the second test case we we fix pressure and density to a small value (0.01), in case the ghost cell values drop below this small value.

$1/h$	Δm	err_{bry}	err_{tot}
400	6.65-5	1.90-5	3.22-6
800	2.29-5	4.55-6	8.71-7
1600	4.73-6	9.34-7	2.46-7

TABLE 2. Wall moving with varying speed – error analysis using (22) for the ghost cell extrapolations

The first test problem is taken from [4] for a numerical convergence study. It is a supersonic vortex in a channel formed by concentric circular arcs. The boundaries of the channel form one quarter of a circle, with inner radius r_i and outer radius r_o. A smooth analytic solution exists for this problem, so the errors in the computation can be evaluated. The density is

$$\rho(r) = \rho_i \left[1 + \frac{\gamma - 1}{2} M_i^2 (1 - (\frac{r_i}{r})^2)\right]^{\frac{1}{\gamma-1}}, \qquad (29)$$

and the velocity varies inversely with the radius. We use the same geometry and test parameters as [4], $\rho_i = 1.0, r_i = 1.0, r_o = 1.384, M_i = 2.25, p_i = 1.0/\gamma$. We take the exact solution as initial condition and run the calculation until time $t = 5.0$, where the analysed values have converged to the 4-th digit. In the numerical study we look at the following relative errors using a discretization of the continuous L_1-norm (cf. [4])

$$err_{tot} = \frac{\sum_k |\rho_{exact} - \rho_k|\,|C_k|}{\sum_k \rho_{exact}|C_k|}, \qquad (30)$$

$$err_{bry} = \frac{\sum_{k\in\partial} |\rho_{exact} - \rho_k|\,\sqrt{|C_k|}}{\sum_{k\in\partial} \rho_{exact}\sqrt{|C_k|}}, \qquad (31)$$

$$(32)$$

where $\sum_k$ is a summation over the regular cells and the boundary cells and $\sum_{k\in\partial}$ is a summation over the boundary cells only. $|C_k|$ is h^2 for regular cells and the area in the fluid domain for the boundary cells. Our boundary treatment is not strictly conservative. Therefore we look also at the difference of mass-inflow and mass-outflow in the final solution:

$$\Delta m = \sum_{k\in in} \rho_k|\mathbf{u}_k|l_k \;-\; \sum_{k\in out} \rho_k|\mathbf{u}_k|l_k, \qquad (33)$$

where $\sum_{k\in in}$ is a summation over the inflow cells and $\sum_{k\in out}$ over the outflow cells. l_k is the length of the inflow/outflow interface. Table 3 (left) shows an error analysis of the above errors. The boundary treatment is of order $\log_2 \frac{16.3}{5.49} = 1.57$.

For a moving-boundary example, we show the cylinder lift-off by a strong shock wave, an example found in [5]. The movement of the cylinder is induced by the flow-field. To describe this motion within second-order accuracy, we use a

h	err_{tot}	err_{bry}	Δm
2.68-2	1.52-3	1.83-3	2.49-3
1.32-2	3.58-4	4.77-4	4.13-4
6.70-3	8.95-5	1.63-4	7.09-5
3.35-3	1.96-5	5.49-5	1.89-5

h	X	Y	Δm
3.33-3	6.89-1	1.429-1	1.79-2
2.50-3	7.00-1	1.392-1	1.05-2
2.00-3	7.06-1	1.379-1	9.50-3

TABLE 3. Error analysis for the supersonic vortex (left) – convergence history for the cylinder liftoff (right)

staggered time grid. The center of the cylinder is given at full time-steps $\mathbf{X}_n$ and its velocity at staggered time-steps $\mathbf{V}_{n-\frac{1}{2}}$. At time t_n the force $\mathbf{F}_n$ on the cylinder is calculated by a numerical integration of the flow field pressure times the normal vector along the surface of the cylinder; the velocity is updated by

$$\mathbf{V}_{n+\frac{1}{2}} = \mathbf{V}_{n-\frac{1}{2}} + \frac{\Delta t}{M}\mathbf{F}_n, \tag{34}$$

where M is the mass of the cylinder. The position of the cylinder is updated by

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \Delta t \mathbf{V}_{n+\frac{1}{2}}. \tag{35}$$

We use the same parameters as in [5]. A cylinder with radius 0.05 is initially located at the lower wall at $x = 0.15$ of a channel with width 0.2. A Mach 3 shock wave starts at $x = 0.08$ moving towards the cylinder and lifting it off. The density and pressure of the resting gas are $\rho = 1.4$ and $p = 1.0$. The density of the cylinder is 10.77. The calculation is stopped at time 0.3282. In our calculation the cylinder hits the upper wall, opposed to the results of [5]. Table 3 (right) lists the final position of the center of the cylinder X, Y, and the final relative mass loss as a function of the grid parameter h. Figure 5 shows pressure contours of the initial condition, the solution at half-time and the final solution after the cylinder has hit the upper wall.

Acknowledgments

The authors were supported in part by DOE Grant De-FG02-92ER25139, and by AFOSR Grant F49620-97-1-0322.

References

[1] S.A. Bayyuk, K.G. Powell, and B. van Leer. *A Simulation Technique for 2-D Unsteady Inviscid Flows Around Arbitrarily Moving and Deforming Bodies of Arbitrary Geometry.* AIAA paper 93-3391, 1993.

[2] J.B. Bell, P. Colella, and M. Welcome. *Conservative Front Tracking for Inviscid Compressible Flow.* 10th AIAA Computational Fluid Dynamics Conference, Honolulu, pp. 814–822, 1991.

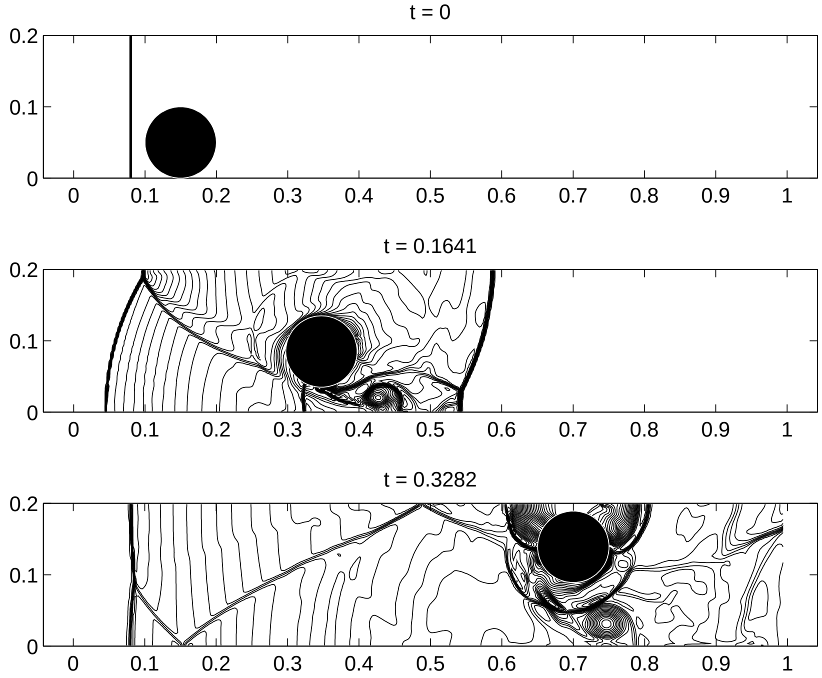

Figure 5. Cylinder lift-off, $h = 0.0025$

[3] M.J. Berger and R.J. LeVeque. *Stable Boundary Conditions for Cartesian Grid Calculations.* ICASE Report No. 90-37, May, 1990.

[4] M.J. Berger and J. Melton. *An Accuracy Test of a Cartesian Grid Method for Steady Flow in Complex Geometries.* Proceedings of 5th International Conference on Hyperbolic Problems, Stony brook, NY, 1994.

[5] J. Falcovitz, G. Alfandary, and G. Hanoch. *A Two-Dimensional Conservation Laws Scheme for Compressible Flows with Moving Boundaries.* Journal of Comp. Phys., 138, pp. 83–102, 1997.

[6] H. Forrer and R. Jeltsch. *A Higher-Order Boundary Treatment for Cartesian-Grid Methods.* To appear, J. Comp. Phys., 1998. Also, ETH Report No. 96-13, available via `netscape` in `http://www.sam.math.ethz.ch/Reports/1996-13.html`.

[7] R.J. LeVeque. *CLAWPACK software.* available from `http://www.amath.washington.edu/~rjl/clawpack.html`.

Courant Institute,
251, Mercer Street
New York, NY 10012
E-mail address: `forrer@cims.nyu.edu`