**Sci. Rep. Kanazawa Univ.**
Vol. 62, pp. 1-22, 2018 **ARTICLES**

# Hyperbolic mean curvature flow with an obstacle

Vita KUSUMASARI*

Graduate School of Natural Science and Technology, Kanazawa University
Kanazawa, 920-1192, Japan

(Received June 26, 2017 and accepted in revised form February 28, 2018)

**Abstract** We treat an interface motion with an obstacle according to the hyperbolic mean curvature flow. In order to realize this motion, we follow the approximation method that is the so-called Hyperbolic MBO (HMBO) algorithm. In this work, we modify the scheme to treat the obstacle problem. Then, we investigate the behaviour when the interface touches the obstacle. We consider two cases of the interface motion. In the first case, the interface stops moving and lies on the obstacle after touching it. For the second case, the interface reflects after touching the obstacle. We also plot the points when the interface contacts with the obstacle and call it the free boundary shape.

**Keywords.** hyperbolic mean curvature flow, obstacle problem.

## 1 Introduction

In this paper, we consider an interfacial motion problem called the hyperbolic mean curvature flow [3]. We suppose that the interfaces are given by a parametrized family of curves $\gamma : I \times [0,T) \to \mathbb{R}^2$, where $I = [a,b]$ and $T > 0$. Let us consider the region $P \subset \mathbb{R}^2$ such that the boundary of $P$ coincides with the interface. Also, the interface is considered as an oriented curve, such that the region $P$ is on the left side of the curve. Then, the problem of the hyperbolic mean curvature flow is to find $\gamma(s,t)$ satisfying

$$\begin{cases} \dfrac{\partial^2 \gamma}{\partial t^2}(s,t) = -\kappa(s,t)\boldsymbol{n}(s,t), \\ \gamma(s,0) = \gamma_0(s), \\ \dfrac{\partial \gamma}{\partial t}(s,0) = v_0(s)\boldsymbol{n}_0(s), \end{cases} \tag{1.1}$$

with $s \in I$ and $t \in [0,T)$. Here, $\kappa(s,t)$ is the curvature, $\boldsymbol{n}(s,t)$ is the unit outer normal vector to the curve $\gamma$ at a point $(s,t)$ and the unit outer normal vector at $t = 0$ is denoted by $\boldsymbol{n}_0(s)$. This geometric evolution says that the normal acceleration of the interface is proportional to its curvature [4] and given by

$$a = -\kappa. \tag{1.2}$$

**Current address:* Department of Mathematics, Faculty of Mathematics and Science, Universitas Negeri Malang, Jalan Semarang No. 5 Malang, Jawa Timur, Indonesia. Email: vitakusumasari@gmail.com

In order to realize the interface motion according to (1.1), we follow the approximation method introduced by Ginder and Svadlenka [2, 3] that is the so-called Hyperbolic MBO (HMBO) algorithm. In those papers, the authors construct the numerical scheme related to the MBO algorithm. The MBO algorithm was developed by Merriman, Bence, and Osher [1]. This algorithm applies a level set approach for realizing the interfacial motion according to the mean curvature flow. Furthermore, [2, 3] develop a modified scheme that can apply to the hyperbolic mean curvature flow and call it the HMBO algorithm. Also, [3] explains the numerical analysis of this algorithm and presents the computational results, including multiphase and volume-preserving motions.

In this work, we are interested in the obstacle problem when the interface touches the obstacle. We use the level set method, so that the interface and the obstacle are expressed as a zero level set of an auxiliary function [4]. The interface model with an obstacle is given in Section 2. The aims of this paper are to modify the scheme for the obstacle problem using the HMBO algorithm and investigate the behaviour of the interface when it touches the obstacle.

We consider two cases of the interface motion based on the choice of the initial curve. In the first case, we consider a closed curve as the initial curve and a circle as the obstacle located inside the curve. The curve stops moving and lies on the obstacle after touching it. In this case, we compare the result of the HMBO algorithm with the solution of a differential equation describing circle motion by (1.1) before touching the obstacle. We do this comparison for several mesh sizes and obtain the convergence order of the HMBO algorithm for the case of a circle evolving by the hyperbolic mean curvature flow.

For the second case, the initial curve is attached to the boundary of the domain and the obstacle is below the curve. The curve reflects after touching the obstacle. We make a comparison between the result of the HMBO algorithm for the obstacle problem and the solution of the hyperbolic obstacle problem based on [7]. At every time we plot the points where the curve contacts the obstacle, we call it the free boundary shape. Then, we find the slope of the free boundary shape and present this result for both schemes.

The paper is organized as follows. Section 2 is about the interface model with an obstacle. The numerical method to simulate this interface motion is given in Section 3. In this section, we explain the level set method and the HMBO algorithm. Also, we give the scheme to treat the obstacle problem using the HMBO algorithm. Then, we apply the numerical method and present the numerical results in Section 4. Finally, the results are summarized in Section 5.

## 2 Interface model with an obstacle

In this paper, we treat the interface motion with an obstacle. We assume that the obstacle $O \subset \mathbb{R}^2$ is a convex open set. As we mentioned before, the interface is expressed by a parametrized curve $\gamma(s,t)$. Here, we suppose that $\gamma$ is a Lipschitz function and $\gamma \in C^2(\{\gamma \in \bar{O}^c\})$. According to (1.1), we formally solve the following obstacle problem

$$\begin{cases} \dfrac{\partial^2 \gamma}{\partial t^2} = -\kappa \boldsymbol{n} & \text{if } \gamma \in \bar{O}^c, \\ \dfrac{\partial \gamma}{\partial t} \cdot \nu_{\partial O} \leq 0 & \text{if } \gamma \in \bar{O}, \\ \gamma \in O^c, & \\ \gamma(s,0) = \gamma_0(s), & \\ \dfrac{\partial \gamma}{\partial t}(s,0) = v_0(s)\boldsymbol{n}_0(s), & \end{cases} \tag{2.1}$$

where $\nu_{\partial O}$ is the unit outer normal vector to the obstacle. To simulate the interface motion with an obstacle, we apply the HMBO algorithm using the level set method. We will explain this numerical method in the next section.

# 3 Numerical method

## 3.1 The level set method

In this paper, we use the level set method to represent the interface motion (1.1). By this method, we consider the interface described by the zero level set of a function $\varphi : \mathbb{R}^2 \times [0,T) \to \mathbb{R}$. Let the interface at a time $t \in [0,T)$ be given by

$$\Gamma_t = \{x \in \mathbb{R}^2 \mid \varphi(x,t) = 0\}.$$

By the level set method, we can derive the equation for time evolution of $\Gamma_t$ as the zero level set of $\varphi(x,t)$. Then, we can express equation (1.2) by the level set representation in terms of $\varphi(x,t)$. The motion in normal direction by the normal acceleration is

$$a = \frac{d}{dt}(v), \tag{3.1}$$

where $v$ is the normal velocity. The level set representation of the normal velocity is

$$\varphi_t + v|\nabla\varphi| = 0.$$

So, the normal acceleration (3.1) is

$$a = -\frac{d}{dt}\left(\frac{\varphi_t}{|\nabla\varphi|}\right) = -\frac{\varphi_{tt}|\nabla\varphi| - \varphi_t|\nabla\varphi|_t}{|\nabla\varphi|^2}.$$

Further, the level set formulation for curvature is

$$\kappa = \operatorname{div}\left(\frac{\nabla\varphi}{|\nabla\varphi|}\right).$$

According to equation (1.2), we get

$$\frac{\varphi_{tt}|\nabla\varphi| - \varphi_t|\nabla\varphi|_t}{|\nabla\varphi|^2} = \operatorname{div}\left(\frac{\nabla\varphi}{|\nabla\varphi|}\right). \tag{3.2}$$

In this case, we take the level set function to be the signed distance to the $\Gamma_t$. We define the signed distance function from a point $x$ to $\Gamma_t$ as

$$d(x,t) = \begin{cases} \inf_{y\in\Gamma_t}\|x-y\| & \text{if } x \in E_t, \\ -\inf_{y\in\Gamma_t}\|x-y\| & \text{otherwise}, \end{cases} \tag{3.3}$$

with $E_t = \{x \in \mathbb{R}^2 | \varphi(x,t) > 0\}$ and $\Gamma_t = \partial E_t$. Since the signed distance function has $|\nabla d| = 1$, we solve (3.2) approximated by [3]

$$\varphi_{tt} = \Delta\varphi.$$

Therefore, we consider that $t_k = k\tau$ is $k$-th time step and $\tau = \frac{T}{M}$, $M$ is a positive integer. Let $d_k(x)$ denote the signed distance to the interface at time $t = t_k$. According to the HMBO algorithm [2, 3], the implementation of this interfacial motion is to solve the initial problem

$$\begin{cases} \varphi_{tt}(x,t) = \Delta\varphi(x,t) & \text{in } \mathbb{R}^2 \times (0,\tau), \\ \varphi(x,0) = d_k(x) & \text{in } \mathbb{R}^2, \\ \varphi_t(x,0) = v_0(x) & \text{in } \mathbb{R}^2 \end{cases} \tag{3.4}$$

for $k = 0, \dots, M-1$ with $M$ being the number of time steps. But, we solve equation (3.4) only for the first step, $k = 0$. For the further steps, $k = 1, \dots, M-1$, we modify this initial problem. If we solve it for all steps, we need to know the velocities along the interface. This can complicate the numerical solution [3]. Therefore, this difficulty is overcome by using the solution to the following problem

$$\begin{cases} \varphi_{tt}(x,t) = 2\Delta\varphi(x,t) & \text{in } \mathbb{R}^2 \times (0,\tau), \\ \varphi(x,0) = 2d_k(x) - d_{k-1}(x) & \text{in } \mathbb{R}^2, \\ \varphi_t(x,0) = 0 & \text{in } \mathbb{R}^2. \end{cases} \tag{3.5}$$

We explain the intuitive idea of this modification. Let us have a very small time step size and assume that the interface moves smoothly. Therefore, the interface does not change much, so that $\kappa$ is approximately constant. Then we can rewrite equation (3.4) depending on time only. Let the initial value be denoted by $\varphi_0$ and let the initial velocity be $v_0$. The equation becomes

$$\begin{cases} \varphi_{tt}(t) = -\kappa & \text{in } (0,\tau), \\ \varphi(0) = \varphi_0, \\ \varphi_t(0) = v_0. \end{cases} \tag{3.6}$$

The solution of equation (3.6) is

$$\varphi(t) = \varphi_0 + v_0 t - \frac{1}{2}\kappa t^2. \tag{3.7}$$

We denote $\varphi(-t)$ as $\varphi_{-1}$ and we get

$$\varphi_{-1} = \varphi_0 - v_0 t - \frac{1}{2}\kappa t^2.$$

Then

$$v_0 = \frac{\varphi_0 - \varphi_{-1}}{t} - \frac{1}{2}\kappa t.$$

Substituting the approximation of $v_0$ above into (3.7) yields

$$\varphi(t) = 2\varphi_0 - \varphi_{-1} - \kappa t^2. \tag{3.8}$$

Now, the solution (3.8) does not contain the term $v_0$. We see that function $\varphi(t)$ in (3.8) is the solution of the modified equation

$$\begin{cases} \varphi_{tt}(t) = -2\kappa & \text{in } (0,\tau), \\ \varphi(0) = 2\varphi_0 - \varphi_{-1}, \\ \varphi_t(0) = 0. \end{cases}$$

From this intuitive idea, we deduce the modified equation (3.5) for the further steps with zero initial velocity by making the following modifications:

1. Use constant $\sqrt{2}$ as the wave speed.
2. Combine the initial values from both previous and current time steps, that is $2d_k(x) - d_{k-1}(x)$.

The detailed explanation of the analysis of this method is given in [3]. Following this idea, we will describe the HMBO algorithm adopted from [3] in the next section.

## 3.2 The HMBO algorithm

### 3.2.1 The original HMBO algorithm

The HMBO algorithm is a numerical approximation for the interface motion (1.1) [3]. It means that we find $\Gamma_t = \{\gamma(s,t) | s \in I\}$ satisfying (1.1). In this case, we consider that the interface will evolve up to a time $T$. We take the time step $\Delta t = \frac{T}{M}$, where $M$ is a positive integer and $0 < \Delta t \ll 1$. Then, the approximation method is as follows:

1. For $k = 0$, we assume that the initial curve is $\Gamma_0$. Then, construct the signed distance function to $\Gamma_0$ as defined in (3.3) and denote it by $d_0(x)$. We find $u : \Omega \times (0,\Delta t) \to \mathbb{R}, \Omega \subset \mathbb{R}^2$, satisfying
$$\begin{cases} u_{tt}(x,t) = \Delta u(x,t) & \text{in } \Omega \times (0,\Delta t), \\ u(x,t) = d_0(x) & \text{on } \partial\Omega \times (0,\Delta t), \\ u(x,0) = d_0(x) & \text{in } \Omega, \\ u_t(x,0) = 0 & \text{in } \Omega. \end{cases} \tag{3.9}$$
For simplicity, we restrict the initial velocity $u_t(x,0) = 0$. Then, we define the zero level set of $u(x,\Delta t)$ as $\Gamma_1$ and compute the signed distance function to $\Gamma_1$.
2. For $k = 1,2,\ldots,M-1$, repeat the following steps
   (a) Solve the equation
$$\begin{cases} u_{tt}(x,t) = 2\Delta u(x,t) & \text{in } \Omega \times (0,\Delta t), \\ u(x,t) = d_k(x) & \text{on } \partial\Omega \times (0,\Delta t), \\ u(x,0) = 2d_k(x) - d_{k-1}(x) & \text{in } \Omega, \\ u_t(x,0) = 0 & \text{in } \Omega. \end{cases} \tag{3.10}$$
   (b) Update the surrounded region and the interface using the zero level set of the solution to (3.10):
$$E_{k+1} = \{x \in \Omega \,|\, u(x,\Delta t) > 0\},$$
$$\Gamma_{k+1} = \partial E_{k+1}.$$
   (c) Compute the signed distance function to $\Gamma_{k+1}$.

We apply the finite difference approximation to solve the wave equation in the HMBO algorithm. Hence, $u_{tt}$ and $\Delta u$ are approximated by the central difference. Here, the domain $\Omega$ is a subset of $\mathbb{R}^2$, $\Omega = (a,b) \times (a,b)$. Each $(a,b)$ is divided into $N$ equal intervals, that is $h = \Delta x = \Delta y = \frac{b-a}{N}$. So, we have $x_i = a + ih, y_j = a + jh$ for $i, j = 0,\ldots,N$. Also, the time interval $(0,\Delta t)$ is divided into $m$ equal intervals, that is $\Delta\tau = \frac{\Delta t}{m}$ and $\tau_l = l\Delta\tau, l = 0,\ldots,m$. In this algorithm, we use Dirichlet boundary condition. If the interface touches the boundary of the domain, then the interface rests at the boundary [6].

### 3.2.2 The modification of the HMBO algorithm for the obstacle problem

Since the HMBO algorithm requires us to solve the wave equation, we need to modify the wave equation to treat the obstacle problem. Then, we follow the idea from the hyperbolic obstacle problem [7].

**(1) The hyperbolic obstacle problem**

In [7], the author describes the hyperbolic obstacle problem as the string vibration with an obstacle. In this phenomenon, we can consider two conditions for the string vibration near the obstacle. First, when the string goes up from the obstacle, the energy conservation law holds. Second, if the string goes down, the energy is not preserved. Also, we assume that the reflection constant is zero when the string hits the obstacle. It means that the string stops if it collides with the obstacle. Here, the shape of the string is described by the graph of a scalar function $v : \hat{\Omega} \times [0,T) \equiv \hat{\Omega}_T \to \mathbb{R}$, where $\hat{\Omega} \subset \mathbb{R}^n$ and the obstacle is the graph of a fixed function $\psi : \hat{\Omega} \to \mathbb{R}$. In [7], the author considered that $\psi$ is the zero function. Then, we derive the equation when the energy conservation law holds. We suppose that the tension energy of the string is $\int_{\hat{\Omega}} |\nabla v|^2 dx$ and the kinetic energy is $\int_{\hat{\Omega}} v_t^2 \chi_{\{v>\psi\}} dx$.

Therefore, the stationary points of the following action functional describe the motion of the string:

$$J(v) = \int_0^T \int_{\hat{\Omega}} ((v_t)^2 - |\nabla v|^2) \chi_{\{v>\psi\}} \, dxdt,$$

where $\chi_{\{v>\psi\}}$ is the characteristic function of the set $\{(x,t) \in \hat{\Omega}_T \,|\, v(x,t) > \psi(x)\}$. We calculate the first variation $\frac{d}{d\varepsilon} J(v + \varepsilon\phi)|_{\varepsilon=0} = 0$, with $\phi \in C_0^\infty(\hat{\Omega}_T \cap \{v > \psi\})$. Then, we get the weak formulation for the wave-type equation

$$v_{tt} = \Delta v \quad \text{in} \quad \hat{\Omega}_T \cap \{v > \psi\}. \tag{3.11}$$

From the inner variation $\frac{d}{d\varepsilon} J(v \circ \tau_\varepsilon^{-1})|_{\varepsilon=0} = 0$, where $\tau_\varepsilon = \mathrm{Id} + \varepsilon\eta$ and $\eta \in C_0^\infty(\hat{\Omega}_T; \mathbb{R}^n \times \mathbb{R})$, we obtain the free boundary condition ([7])

$$|\nabla v|^2 - (v_t)^2 = 0 \quad \text{on} \quad \hat{\Omega}_T \cap \partial\{v > \psi\}. \tag{3.12}$$

From (3.11) and (3.12), we can derive the equation [5]

$$\chi_{\overline{\{v>\psi\}}} v_{tt} = \Delta v \quad \text{in} \quad \hat{\Omega}_T.$$

Hence, we introduce the problem as

$$\begin{cases} \chi_{\overline{\{v>\psi\}}} v_{tt} = \Delta v & \text{in } \hat{\Omega}_T, \\ v(x,0) = f_0(x) & \text{on } \hat{\Omega}, \\ v_t(x,0) = g_0(x) & \text{on } \hat{\Omega}, \\ v(x,t) = p(x,t) & \text{on } \partial\hat{\Omega}_T, \text{ with } p(x,0) = f_0(x) \text{ on } \partial\hat{\Omega}, \end{cases} \tag{3.13}$$

where the first equation is understood in the sense of distributions. When the string touches the obstacle, the solution $v$ also satisfies

$$\begin{cases} v \geq \psi, \quad \Delta v \geq v_{tt} & \text{in } \hat{\Omega}_T, \\ \Delta v = v_{tt} & \text{on } \hat{\Omega}_T \cap \{v > \psi\} \end{cases}$$

in the sense of distributions.

We solve (3.13) using a finite difference approximation. Consider $\hat{\Omega} = (a,b) \subset \mathbb{R}$ to be divided into $N$ equal intervals, so we have $h = \frac{b-a}{N}$ and $x_i = a + ih, i = 0, \ldots, N$. For $t_k = k\Delta t, k = 0, \ldots, M$, we approximate $v_{tt}$ and $\Delta v$ by central difference. Also, the characteristic function is defined ([5]) by

$$\chi_{\{v>\psi\}}(x_i, t_k) = \begin{cases} 1 & \text{if } v_{i-1}^k > \psi_{i-1} \text{ or } v_i^k > \psi_i \text{ or } v_{i+1}^k > \psi_{i+1}, \\ 0 & \text{otherwise,} \end{cases}$$

where $v_i^k = v(x_i, t_k)$ and $\psi_i = \psi(x_i)$. Hence, we get the scheme

$$\begin{cases} v_i^{k+1} = 2v_i^k - v_i^{k-1} + \frac{\Delta t^2}{h^2}(v_{i+1}^k + v_{i-1}^k - 2v_i^k) & \text{if } \chi_{\{v>\psi\}}(x_i, t_k) = 1, \\ v_i^{k+1} = \psi_i & \text{if } \chi_{\{v>\psi\}}(x_i, t_k) = 0 \end{cases} \tag{3.14}$$

for $k = 0, \ldots, M-1$ and $i = 1, \ldots, N-1$.

**(2) The HMBO algorithm for the obstacle problem**

In the HMBO algorithm, the interface is expressed as the zero level set of a function $u\colon \Omega \times (0, \Delta t) \to \mathbb{R}$. Similarly, the obstacle is also represented by the zero level set of a fixed function. We define $w : \Omega \to \mathbb{R}$ such that $\{x \in \Omega \,|\, w(x) = 0\}$ is the obstacle. Let $\mu : \Omega \to \mathbb{R}$ be the signed distance function to the obstacle. To treat the obstacle problem, we follow the discretization of the hyperbolic obstacle problem given in scheme (3.14) for solving equations (3.9) and (3.10). Let $u_{i,j}^l = u(x_i, y_j, \tau_l)$ and $\mu_{i,j} = \mu(x_i, y_j)$. Then, we have the following scheme

$$\begin{cases} u_{i,j}^{l+1} = 2u_{i,j}^l - u_{i,j}^{l-1} & \\ \qquad\qquad + c^2\frac{\Delta\tau^2}{h^2}(u_{i+1,j}^l + u_{i-1,j}^l + u_{i,j+1}^l + u_{i,j-1}^l - 4u_{i,j}^l) & \text{if } \chi_{\{u>\mu\}}(x_i, y_j, \tau_l) = 1, \\ u_{i,j}^{l+1} = \mu_{i,j} & \text{if } \chi_{\{u>\mu\}}(x_i, y_j, \tau_l) = 0, \end{cases} \tag{3.15}$$

for $l = 0, \ldots, m-1$ and $i, j = 1, \ldots, N-1$. The constant $c^2 = 1$ for equation (3.9) and $c^2 = 2$ for equation (3.10). Here, we define

$$\chi_{\{u>\mu\}}(x_i, y_j, \tau_l) = \begin{cases} 1 & \text{if } u_{i,j}^l > \mu_{i,j}, \\ 0 & \text{otherwise.} \end{cases} \tag{3.16}$$

By implementing this scheme, we obtain

$$u(x,t) \geq \mu(x) \quad \text{for} \quad (x,t) \in \Omega \times (0,T).$$

However, scheme (3.15) is still developed. We will investigate the results using this scheme.

**Remark.** If we follow the characteristic function for the hyperbolic obstacle problem, then the characteristic function for the HMBO algorithm should be

$$\chi_{\{u>\mu\}}(x_i, y_j, \tau_l) = \begin{cases} 1 & \text{if } u_{i\pm1,j}^l > \mu_{i\pm1,j} \text{ or } u_{i,j}^l > \mu_{i,j} \text{ or } u_{i,j\pm1}^l > \mu_{i,j\pm1}, \\ 0 & \text{otherwise.} \end{cases} \tag{3.17}$$

However, using (3.17), the interface motion bounces after touching the obstacle. This behaviour does not agree with the expected motion. The expected motion based on problem (2.1) is that the interface can rest on the obstacle after touching it. Hence, to match the expected motion, we use the characteristic function (3.16). We will present these results in the next section.

## 4 Numerical results

In this part, we consider two cases of the interface motion. Both cases are based on the choice of the initial curve. In the first case, we consider a closed curve as the initial curve and the obstacle is a circle located inside the curve. Applying the HMBO algorithm for the obstacle problem, the curve stops moving and lies on the obstacle after touching it. For the second case, the initial curve is attached to the boundary of the domain. Also, we position the obstacle so that it is below the curve. In this case, the curve reflects after touching the obstacle.

### 4.1 First case

For the numerical test, we consider a circle evolving by (1.1) with initial radius $r_0$ and initial velocity $v_0$. We give a fixed circle with a smaller radius as an obstacle. The circle will shrink before touching the obstacle and stop after touching it. Before touching the obstacle, the curve is the circle with radius $r(t)$ satisfying

$$\begin{cases} r''(t) = -\dfrac{1}{r(t)}, \\ r(0) = r_0, \\ r'(0) = v_0. \end{cases} \tag{4.1}$$

Consider a circle with initial radius $r_0 = 0.35$, centered at $(x,y) = (0.5, 0.5)$ on a domain $(0,1)\times(0,1)$, and zero initial velocity. A circle with radius $r = 0.1$ at the center of the initial circle is given as the obstacle. We use mesh size $h = \frac{1}{N}$ where $N$ is grid resolution, time step $\Delta t = \frac{t_e}{2^9}$ with $t_e$ being the extinction time given by $t_e = 0.35\sqrt{\frac{\pi}{2}}$ [3]. Then, this gives the time discretization for finite difference approximation to the wave equation $\Delta\tau = \frac{\Delta t}{2^6}$. In this part, we take the condition that is close to the case of a circle evolving by the hyperbolic mean curvature flow in [3], so we can compare both results.

The error of the radius of the circle before touching the obstacle is obtained by the comparison between the result from the HMBO algorithm and the solution of (4.1) using Runge Kutta fourth order method. The $L^2$ error is

$$e = \sqrt{\Delta t \sum_{k=1}^{2^9} (r_r^k - r_n^k)^2},$$

where $r_n$ is the maximum distance to the center from the HMBO algorithm result and $r_r$ is the solution to (4.1). Furthermore, we show the convergence order related to $L^2$ error. The convergence order is computed by

$$\frac{\log e_{h_2} - \log e_{h_1}}{\log h_2 - \log h_1},$$

where $h_2 > h_1$ are two different mesh sizes. The error and the convergence order are shown in the table below.

Table 1: Error and convergence order using the HMBO algorithm

| $N$ | $e$ | convergence order |
|---|---|---|
| 16 | 0.107927 | - |
| 32 | 0.0966846 | 0.159 |
| 64 | 0.0813308 | 0.249 |
| 128 | 0.0573378 | 0.504 |
| 256 | 0.0324356 | 0.822 |
| 512 | 0.0181595 | 0.837 |

From Table 1, as the mesh size decreases, the error value also decreases. Moreover, the $L^2$ error and its convergence order agree with the result of the circle case given in [3].

We also tried other conditions when the obstacle position is not at the center of the initial curve. From the results of this trial, the interfaces shrink and stop moving after touching the obstacle. Then, the interfaces lie on the obstacle and follow the shape of obstacle. The numerical results of the interface motion with an obstacle for several times are shown in the figures below.

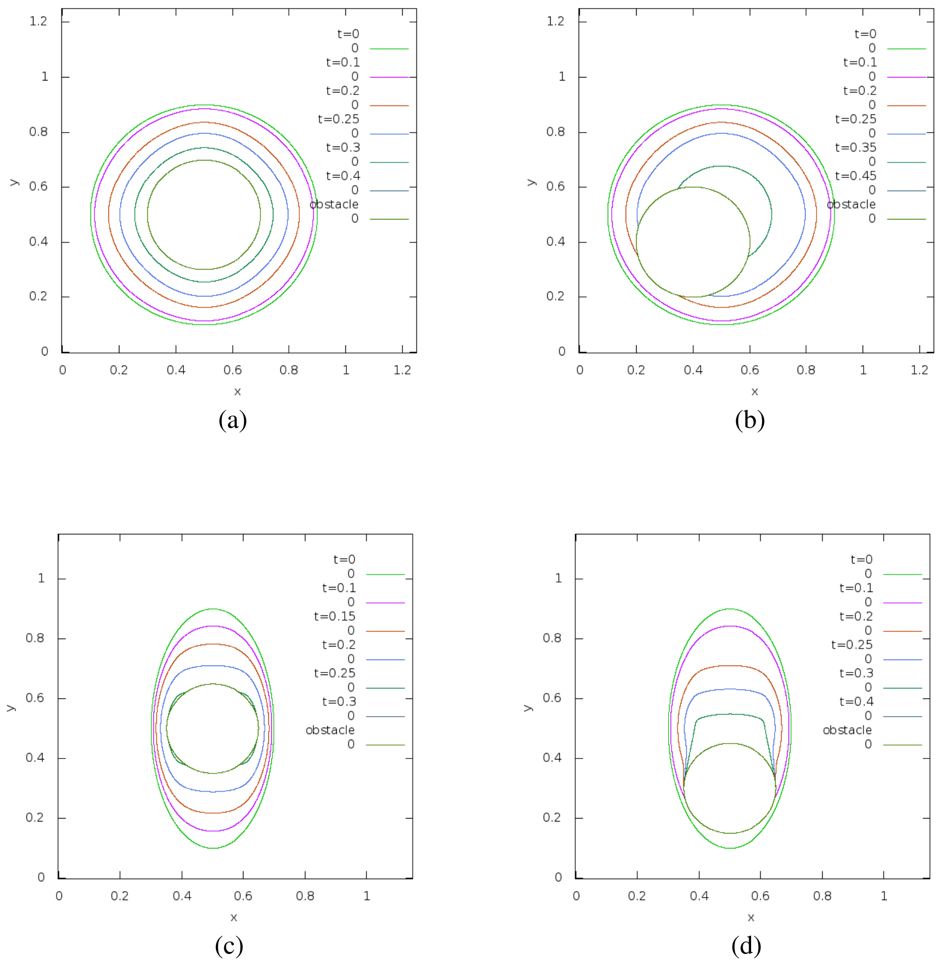



Figure 1: The interface motion with an obstacle

## 4.2 Second case

In this case, we consider the curve having small displacement, such that the interface motion by the hyperbolic mean curvature flow coincides with the wave equation. Therefore, we compare the results of the HMBO algorithm for the obstacle problem and the hyperbolic obstacle problem using scheme (3.14). For the HMBO algorithm, we consider $\Omega = (-1,1) \times (0,2)$, the mesh size $h = \frac{2}{N}, N = 128, 256, 512, 1024$, the time step $\Delta t = 0.001$, and time discretization for finite difference approximation $\Delta\tau = \frac{\Delta t}{m}, m = 10$. Then, for scheme (3.14), we use $\hat{\Omega} = (-1,1)$, the mesh size $h = \frac{2}{N}, N = 128, 256, 512, 1024$, and the time step $\Delta t = 0.001$.

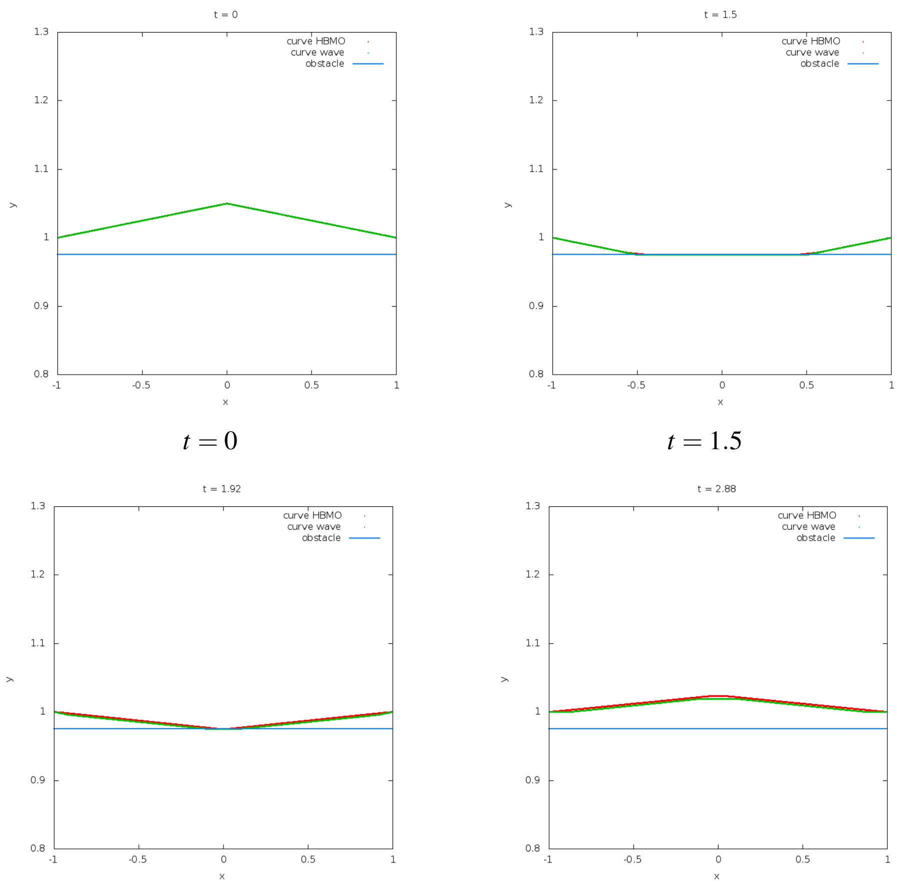
In this trial, the initial curves are given by a piece-wise linear function or a quadratic function with small displacement. Also, we set the position of an obstacle below the curve such that it is not too close to the curve but the curve motion can touch it. Here, we use more general function for the obstacle besides the zero function as in [7], namely constant function, linear function, and quadratic function. Then, we will describe the numerical results for each obstacle function.

### 4.2.1 Constant function

In this part, the obstacle is given by a constant function. The initial conditions and the obstacle for both schemes are described below. Here, $u(x,y,0)$ and $w(x,y)$ are the initial value and the obstacle function for the HMBO algorithm, respectively. For scheme (3.14), the initial conditions are $v(x,0)$ and $v_t(x,0)$ with the boundary conditions being $v(-1,t) \equiv v(-1,0)$ and $v(1,t) \equiv v(1,0)$. Also, $\psi(x)$ represents the obstacle function. More precisely, we have

- Case 1 (Initial curve is a piecewise linear function)
  $u(x,y,0) = -0.05|x| - y + 1.05$
  $w(x,y) = 0.975 - y$ on $\Omega$,
  $v(x,0) = -0.05|x| + 1.05, v_t(x,0) = 0$
  $\psi(x) = 0.975$ on $\hat{\Omega}$.
- Case 2 (Initial curve is a quadratic function)
  $u(x,y,0) = -0.05x^2 - y + 1.05$
  $w(x,y) = 0.975 - y$ on $\Omega$,
  $v(x,0) = -0.05x^2 + 1.05, v_t(x,0) = 0$
  $\psi(x) = 0.975$ on $\hat{\Omega}$.

The figures below show the curve motions with an obstacle using the HMBO algorithm and




$t = 0$ $t = 1.5$

$t = 1.92$ $t = 2.88$

Figure 2: Curve motion with a constant obstacle for Case 1

scheme (3.14). For instance, we plot each case with the mesh size $N = 1024$. Since the curve has small displacement, we use different scales for $y-$axis and $x-$axis to make the figure of the curve motion clearer. In these figures, the curve motion using the HMBO algorithm is indicated by curve HBMO and curve wave is for the hyperbolic obstacle problem using scheme (3.14).

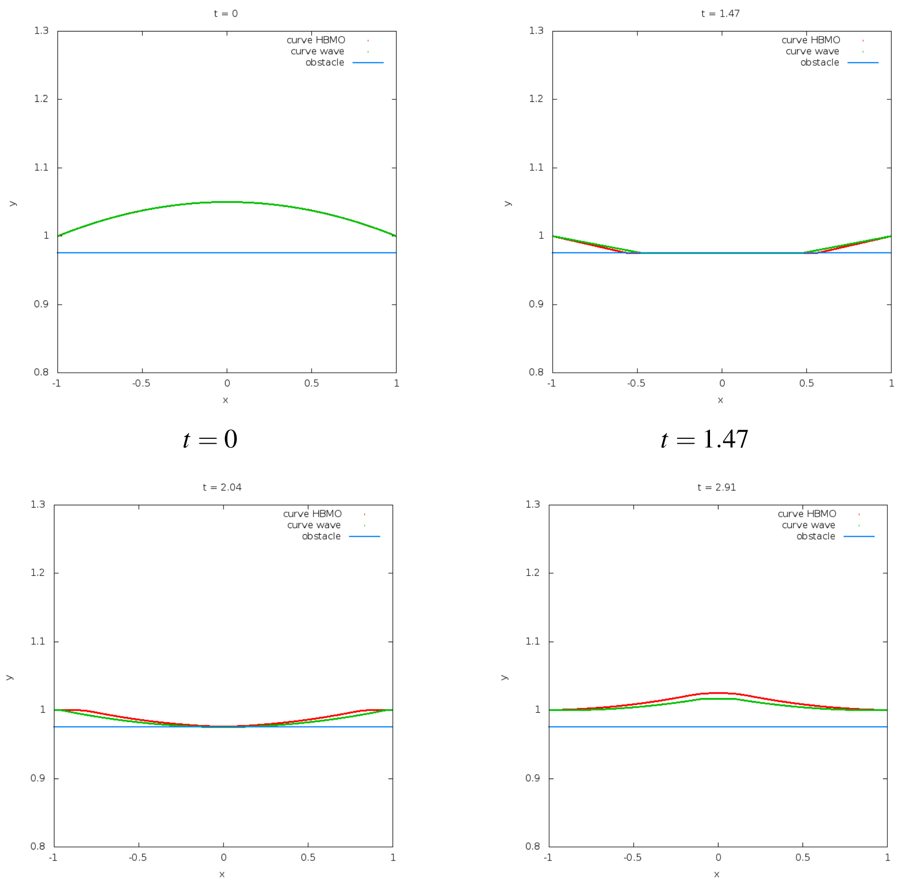
From Figure 2 and Figure 3, the curve reflects after touching the obstacle and vibrates above the obstacle. We plot the points when the curve contacts with the obstacle at every time step. To get the contact points with the obstacle, we find the end points in both sides when the curve is close enough to the obstacle. We find the left contact point $x_L$ as

$$x_L = \min_{j \in \mathbb{N}} \{x_j \mid |f(x_j) - g(x_j)| < \varepsilon\} \tag{4.2}$$

and the right contact point $x_R$ as

$$x_R = \max_{j \in \mathbb{N}} \{x_j \mid |f(x_j) - g(x_j)| < \varepsilon\}, \tag{4.3}$$

where $0 < \varepsilon \ll 1$. Then, $f(x_j)$ and $g(x_j)$ are the values of functions describing the curve and the obstacle, respectively, for every grid point at a certain time.

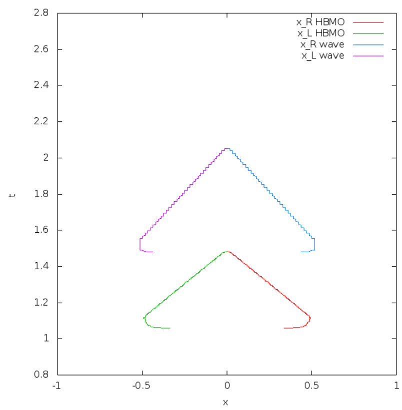



$t = 0$ $t = 1.47$

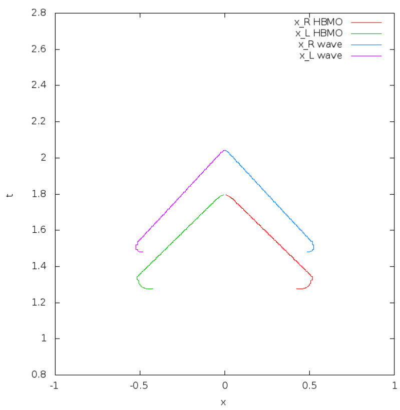
$t = 2.04$ $t = 2.91$

Figure 3: Curve motion with a constant obstacle for Case 2

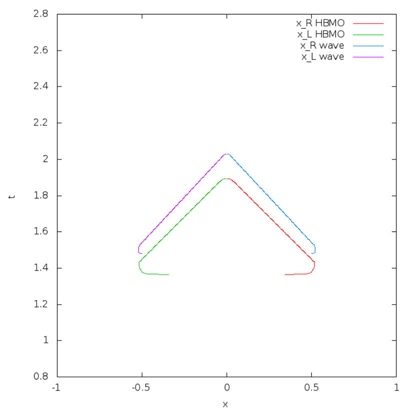
For the HMBO algorithm, $f(x)$ and $g(x)$ are the functions whose graphs represent the zero level set of $u(x,y)$ and $w(x,y)$, respectively. Meanwhile, for scheme (3.14), $f(x)$ represents the function $v(x)$ and $g(x)$ is the obstacle function $\psi(x)$. For the range that determines where the curve touches the obstacle, that is $\{x \mid |f(x)-g(x)| < \varepsilon\}$, we take $\varepsilon = 0.001$ for both schemes. Furthermore, we plot the contact points, $x_L$ and $x_R$, at every time step for each mesh size using both schemes. These graphs are called the free boundary shape [7] and are shown in the figures below. In these figures, the free boundary shape from the curve motion using the HMBO algorithm is formed by x_R HBMO and x_L HBMO. Further, x_R wave and x_L wave denote the free boundary shape for the hyperbolic obstacle problem.



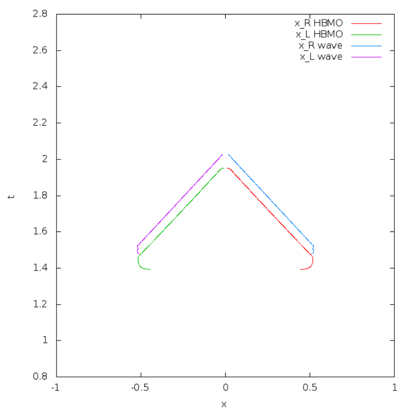

$N = 128$




$N = 256$

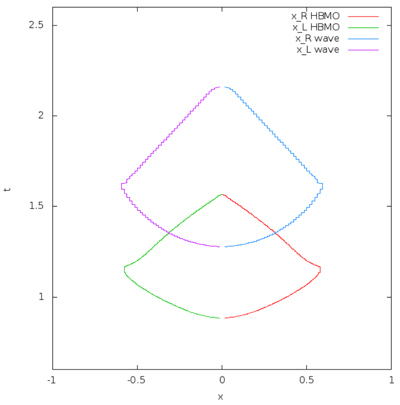



$N = 512$

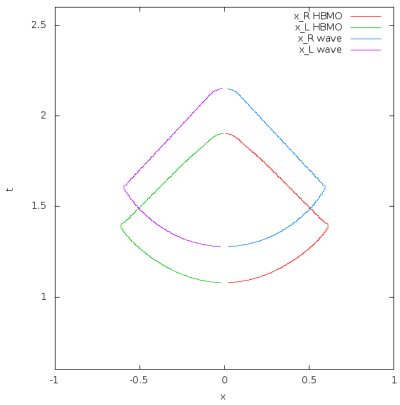



$N = 1024$

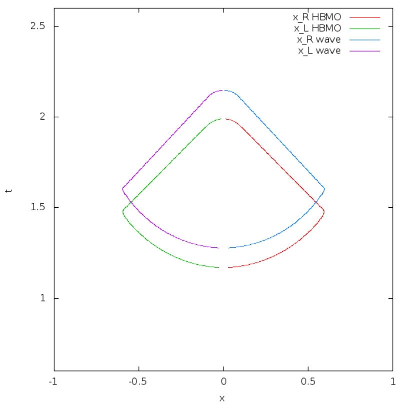
Figure 4: The free boundary shape for Case 1



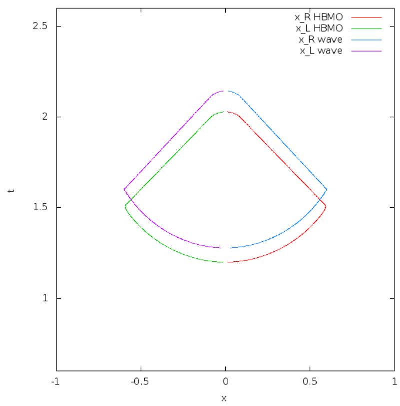

$N = 128$




$N = 256$




$N = 512$




$N = 1024$

Figure 5: The free boundary shape for Case 2

Moreover, we find the slope of the free boundary shape when the curve is going up for both sides. This slope represents the free boundary condition (3.12). According to this free boundary condition, we expect that the slope of the free boundary shape should be $\pm 1$. The slopes of the free boundary shape using both schemes for each mesh size are given in the following tables. By the curve motion, we consider that $t^*$ denotes the time when the curve starts going up.

Table 2: The slope of the free boundary shape for Case 1

| $N$ | Slope | | | |
|---|---|---|---|---|
| | HMBO algorithm | | scheme (3.14) | |
| | right and left sides | $t^*$ | right and left sides | $t^*$ |
| 128 | $\pm$ 0.76 | 1.119 | $\pm$ 1.01 | 1.556 |
| 256 | $\pm$ 0.91 | 1.338 | $\pm$ 1.01 | 1.521 |
| 512 | $\pm$ 0.93 | 1.425 | $\pm$ 1 | 1.518 |
| 1024 | $\pm$ 0.99 | 1.454 | $\pm$ 1.01 | 1.493 |

Table 3: The slope of the free boundary shape for Case 2

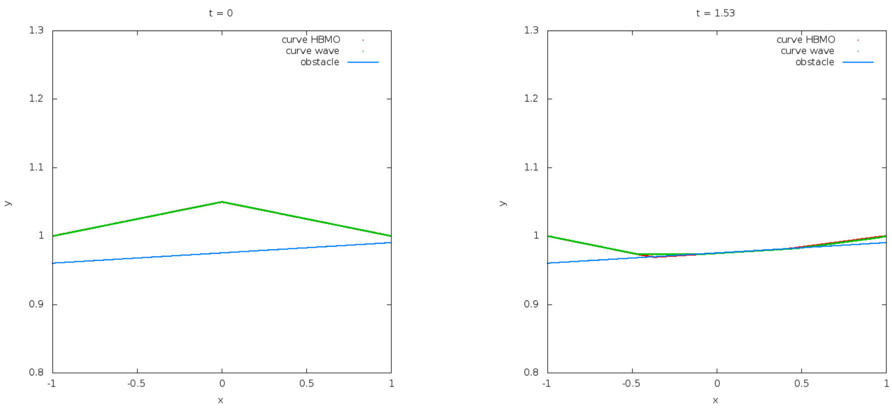
| $N$ | Slope | | | |
|---|---|---|---|---|
| | HMBO algorithm | | scheme (3.14) | |
| | right and left sides | $t^*$ | right and left sides | $t^*$ |
| 128 | $\pm\, 0.75$ | 1.167 | $\pm\, 1$ | 1.628 |
| 256 | $\pm\, 0.9$ | 1.392 | $\pm\, 0.99$ | 1.628 |
| 512 | $\pm\, 0.93$ | 1.478 | $\pm\, 0.99$ | 1.61 |
| 1024 | $\pm\, 0.95$ | 1.509 | $\pm\, 0.99$ | 1.606 |

### 4.2.2 Linear function

We simulate the curve motion with the obstacle given by a linear function using the HMBO algorithm and scheme (3.14). Here, we use the same domain and discretization in space and time as in the previous part. Also, the initial and boundary conditions are the same as before. We take

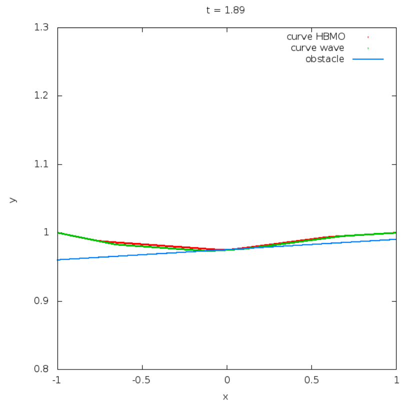
- Case 3 (Initial curve is a piecewise linear function)
  $u(x,y,0) = -0.05|x| - y + 1.05$, $w(x,y) = 0.015x - y + 0.975$ on $\Omega$,
  $v(x,0) = -0.05|x| + 1.05, v_t(x,0) = 0$, $\psi(x,y) = 0.015x + 0.975$ on $\hat{\Omega}$.
- Case 4 (Initial curve is a quadratic function)
  $u(x,y,0) = -0.05x^2 - y + 1.05$, $w(x,y) = 0.015x - y + 0.975$ on $\Omega$,
  $v(x,0) = -0.05x^2 + 1.05, v_t(x,0) = 0$, $\psi(x,y) = 0.015x + 0.975$ on $\hat{\Omega}$.

The figures below show the curve motions for each case using both schemes with $N = 1024$.

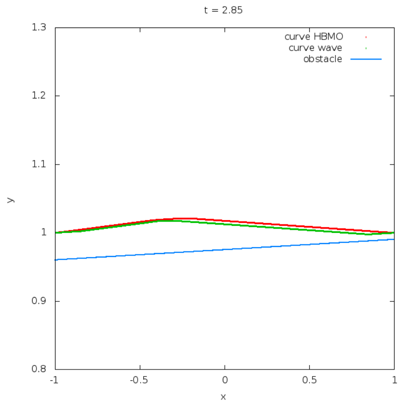



$t = 0$ $t = 1.53$

Figure 6: Curve motion with a linear obstacle for Case 3 ($t = 0, 1.53$)

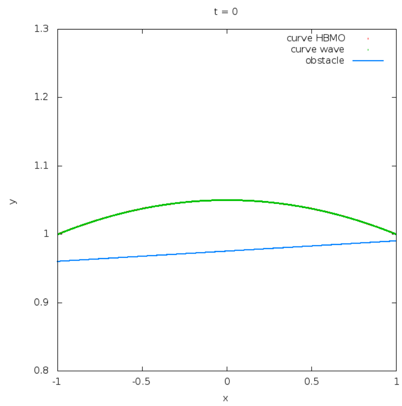



$t = 1.89$

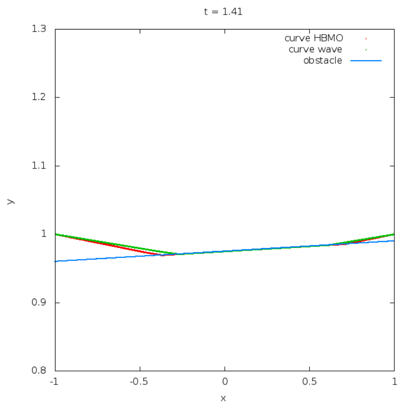



$t = 2.85$

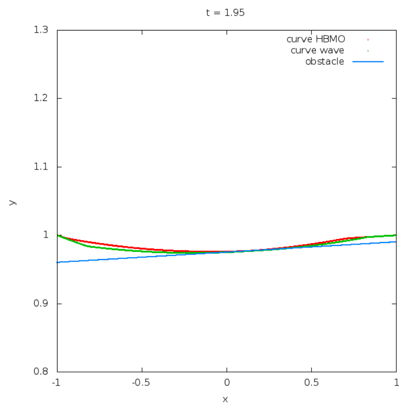
Figure 7: Curve motion with a linear obstacle for Case 3 ($t = 1.89, 2.85$)



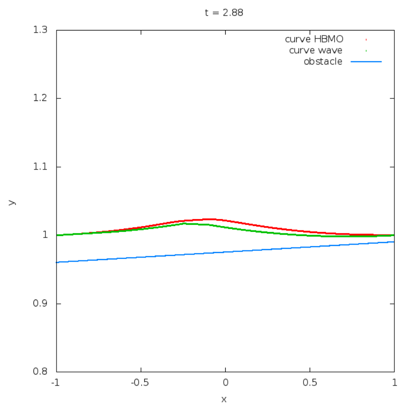

$t = 0$




$t = 1.41$



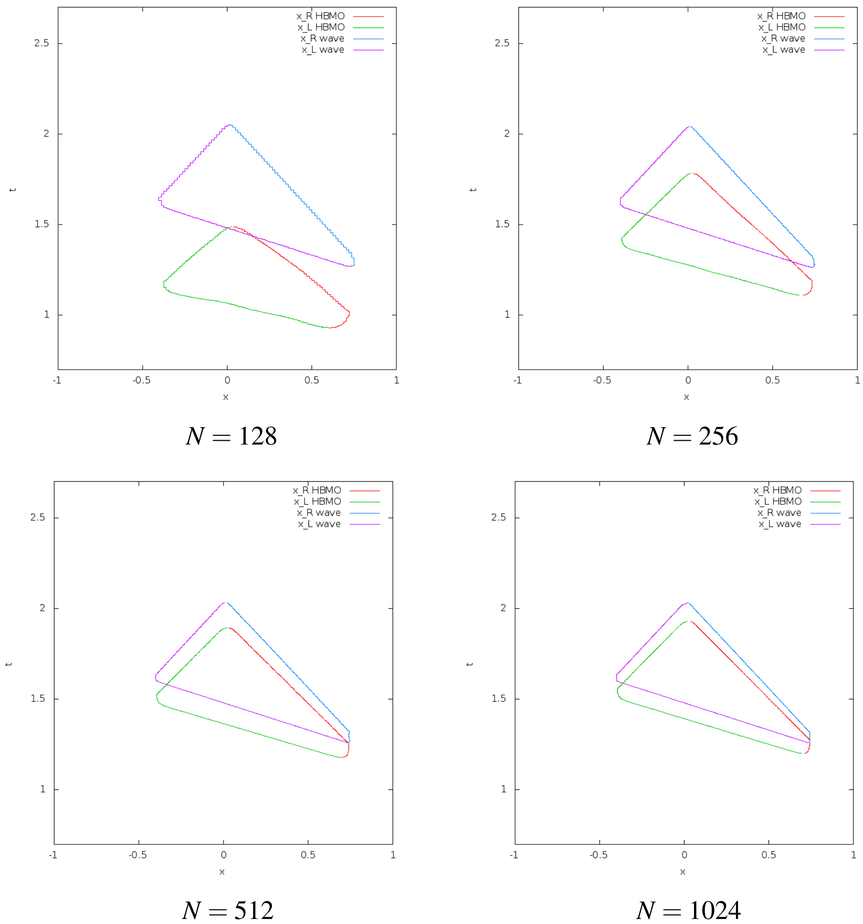

$t = 1.95$




$t = 2.88$

Figure 8: Curve motion with a linear obstacle for Case 4

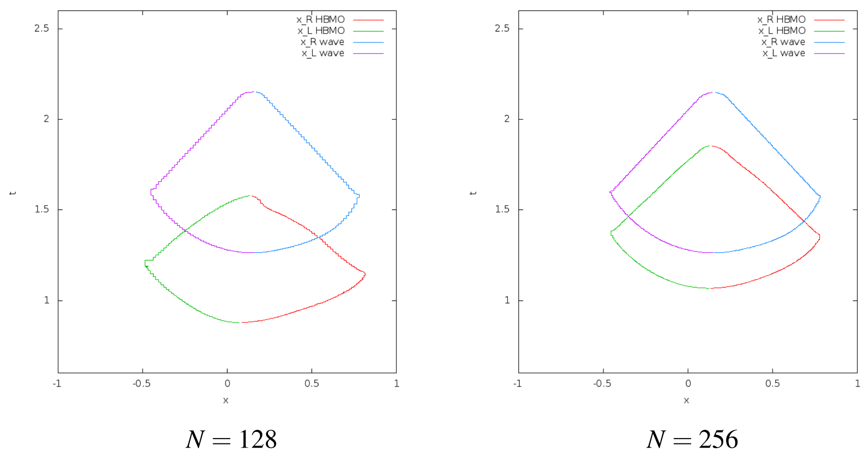
Also, we plot the free boundary shape using both schemes for each case. Hence, the free boundary shapes for each mesh size are shown in the figures below.




$N = 128$ $N = 256$

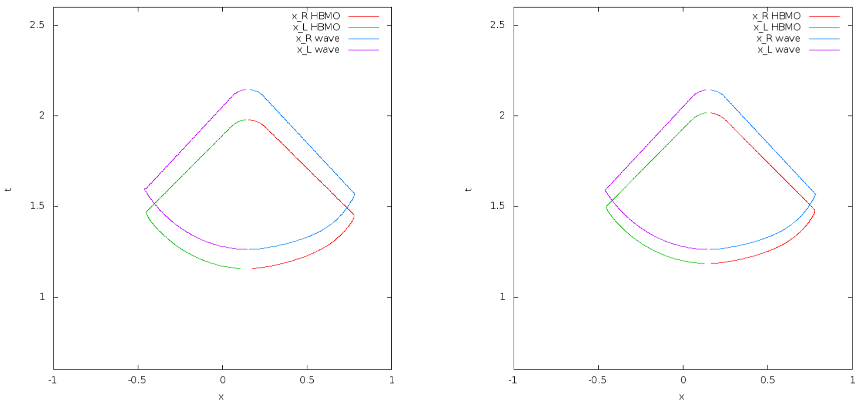
$N = 512$ $N = 1024$

Figure 9: The free boundary shape for Case 3




$N = 128$ $N = 256$

Figure 10: The free boundary shape for Case 4 $(N = 128, 256)$




$N = 512$ $N = 1024$

Figure 11: The free boundary shape for Case 4 ($N = 512, 1024$)

Moreover, we find the slope of the free boundary shape when the curve is going up. In order to obtain the left contact point, $x_L$, and the right contact point, $x_R$, with the obstacle, we use equations (4.2) and (4.3). The tables below represent the slopes of the free boundary shape for both schemes.

Table 4: The slope of the free boundary shape for Case 3

| $N$ | Slope | | | | | |
|---|---|---|---|---|---|---|
| | HMBO algorithm | | | scheme (3.14) | | |
| | right side | left side | $t^*$ | right side | left side | $t^*$ |
| 128 | -0.66 | 0.8 | 1.187 | -1 | 1 | 1.652 |
| 256 | -0.85 | 0.91 | 1.427 | -1.01 | 1 | 1.65 |
| 512 | -0.92 | 0.93 | 1.524 | -0.99 | 1 | 1.621 |
| 1024 | -0.94 | 0.94 | 1.557 | -1 | 1 | 1.619 |

Table 5: The slope of the free boundary shape for Case 4

| $N$ | Slope | | | | | |
|---|---|---|---|---|---|---|
| | HMBO algorithm | | | scheme (3.14) | | |
| | right side | left side | $t^*$ | right side | left side | $t^*$ |
| 128 | -0.62 | 0.66 | 1.223 | -0.99 | 0.99 | 1.616 |
| 256 | -0.82 | 0.87 | 1.385 | -0.99 | 0.99 | 1.604 |
| 512 | -0.91 | 0.93 | 1.473 | -0.99 | 0.99 | 1.598 |
| 1024 | -0.93 | 0.94 | 1.501 | -0.99 | 0.99 | 1.594 |

### 4.2.3 Quadratic function

Next, we use a quadratic function as the obstacle that is given below

- Case 5 (Initial curve is a piecewise linear function)

$$\left.\begin{array}{l} u(x,y,0) = -0.05|x| - y + 1.05 \\ w(x,y) = -0.015x^2 - y + 0.975 \end{array}\right. \quad \text{on } \Omega,$$

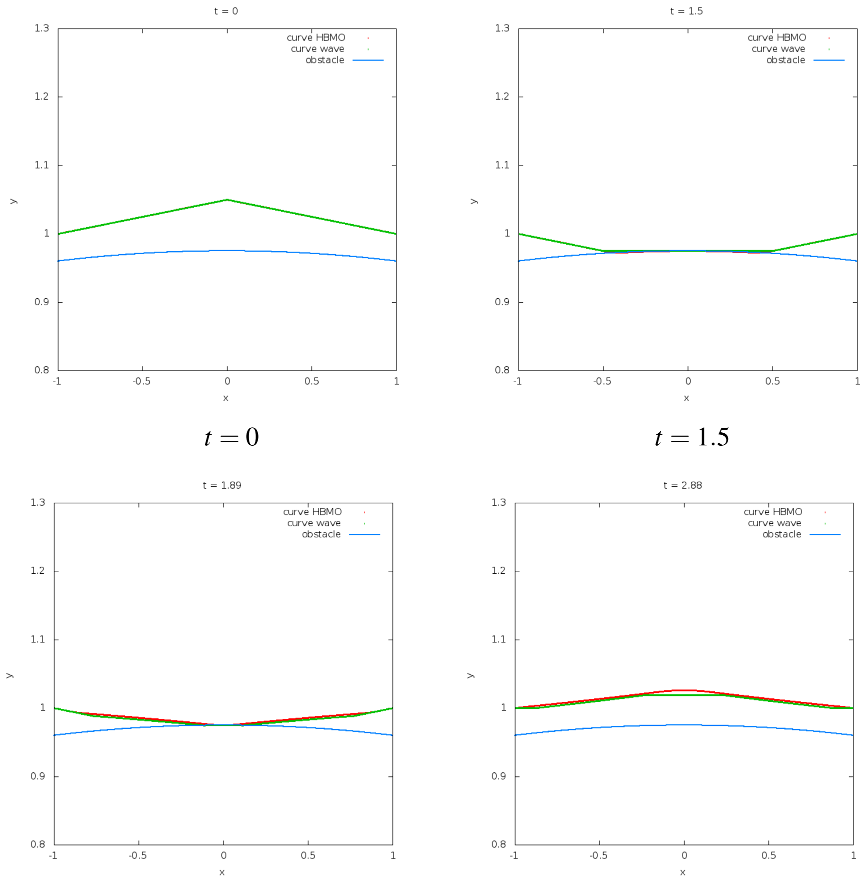
$$\left.\begin{array}{l} v(x,0) = -0.05|x| + 1.05, v_t(x,0) = 0 \\ \psi(x,y) = -0.015x^2 + 0.975 \end{array}\right. \quad \text{on } \hat{\Omega}.$$

- Case 6 (Initial curve is a quadratic function)

$$\left.\begin{array}{l} u(x,y,0) = -0.05x^2 - y + 1.05 \\ w(x,y) = -0.015x^2 - y + 0.975 \end{array}\right. \quad \text{on } \Omega,$$

$$\left.\begin{array}{l} v(x,0) = -0.05x^2 + 1.05, v_t(x,0) = 0 \\ \psi(x,y) = -0.015x^2 + 0.975 \end{array}\right. \quad \text{on } \hat{\Omega}.$$

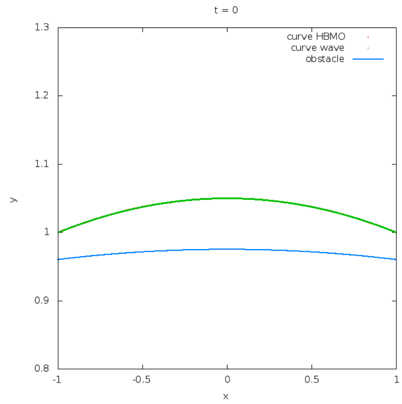
The figures below show the curve motions for each case using both schemes with $N = 1024$.



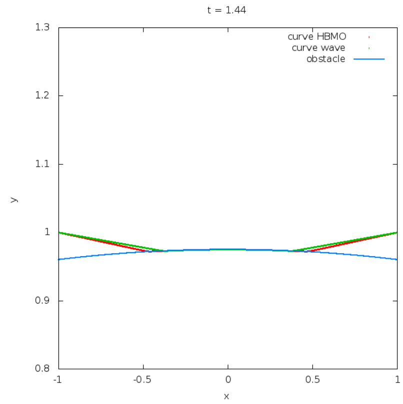

$t = 0$ $t = 1.5$

$t = 1.89$ $t = 2.88$

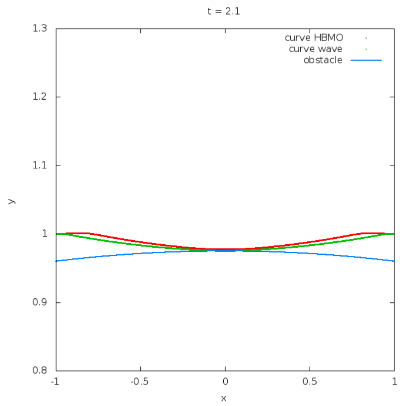
Figure 12: Curve motion with a quadratic obstacle for Case 5



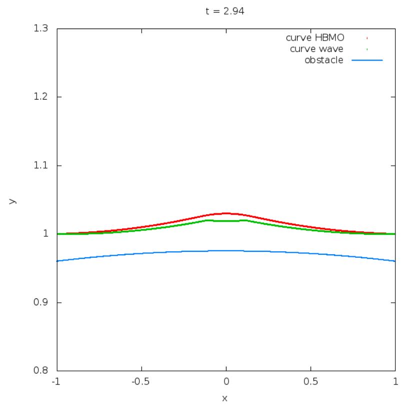

$t = 0$




$t = 1.44$



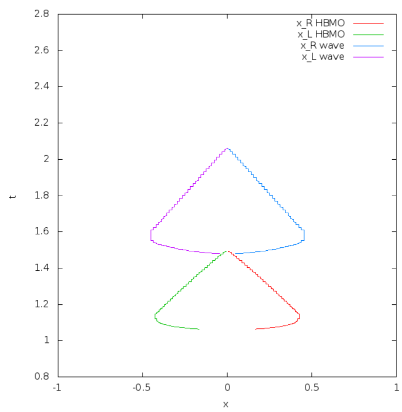

$t = 2.1$



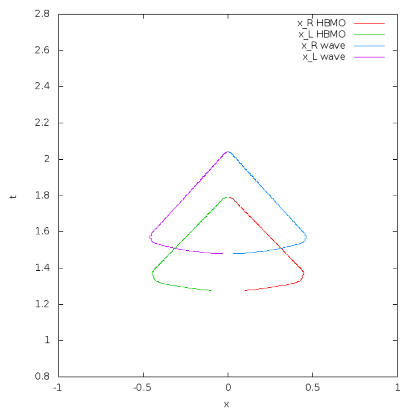

$t = 2.94$

Figure 13: Curve motion with a quadratic obstacle for Case 6

Also, the free boundary shapes for each mesh size are shown in the figures below.

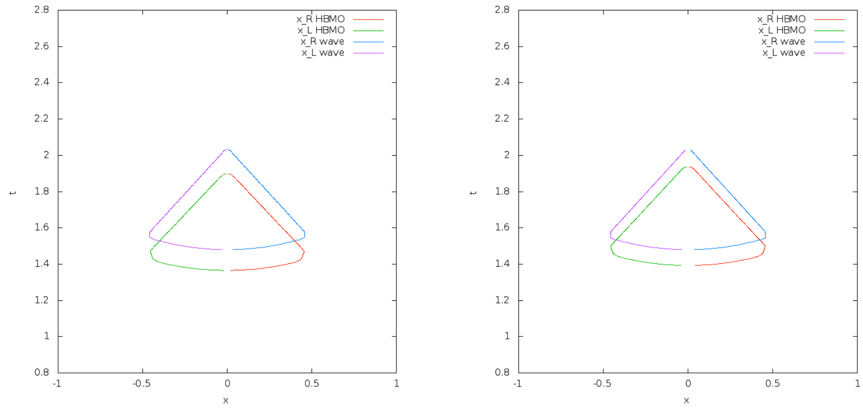



$N = 128$



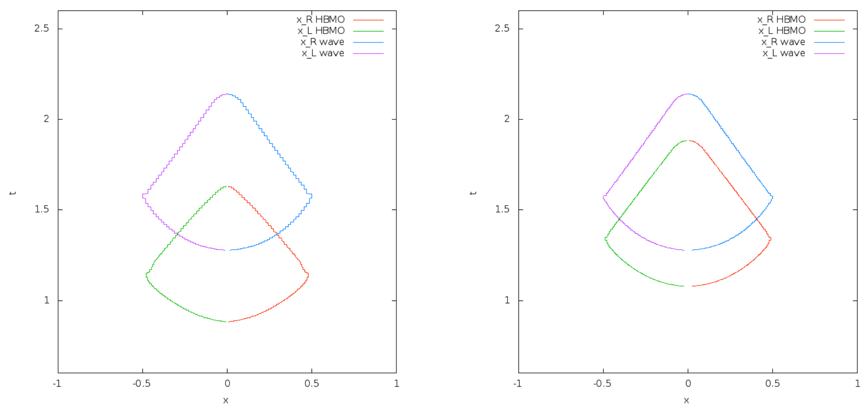

$N = 256$

Figure 14: The free boundary shape for Case 5 $(N = 128, 256)$

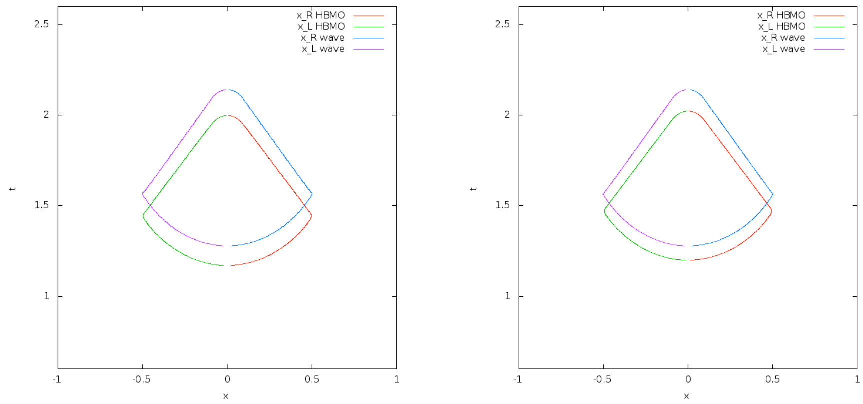



$N = 512$ $N = 1024$

Figure 15: The free boundary shape for Case 5 $(N = 512, 1024)$




$N = 128$ $N = 256$




$N = 512$ $N = 1024$

Figure 16: The free boundary shape for Case 6

Moreover, we find the slopes of free boundary shape when the curve is going up and they are given in the tables below.

Table 6: The slope of the free boundary shape for Case 5

| $N$ | Slope | | | |
|---|---|---|---|---|
| | HMBO algorithm | | scheme (3.14) | |
| | right and left sides | $t^*$ | right and left sides | $t^*$ |
| 128 | $\pm$ 0.84 | 1.149 | $\pm$ 1.04 | 1.611 |
| 256 | $\pm$ 0.96 | 1.378 | $\pm$ 1.02 | 1.579 |
| 512 | $\pm$ 0.98 | 1.475 | $\pm$ 1.02 | 1.579 |
| 1024 | $\pm$ 1 | 1.504 | $\pm$ 1.03 | 1.565 |

Table 7: The slope of the free boundary shape for Case 6

| $N$ | Slope | | | |
|---|---|---|---|---|
| | HMBO algorithm | | scheme (3.14) | |
| | right and left sides | $t^*$ | right and left sides | $t^*$ |
| 128 | $\pm$ 1.07 | 1.14 | $\pm$ 1.26 | 1.59 |
| 256 | $\pm$ 1.22 | 1.341 | $\pm$ 1.27 | 1.578 |
| 512 | $\pm$ 1.22 | 1.445 | $\pm$ 1.28 | 1.572 |
| 1024 | $\pm$ 1.22 | 1.473 | $\pm$ 1.28 | 1.568 |

From Case 1 - Case 6, the slopes of free boundary shape obtained by using the HMBO algorithm and scheme (3.14) coincide as the mesh size becomes smaller. It means that the curve motion using both schemes gives similar results. Also, the slope of free boundary shape approaches the free boundary condition.

# 5 Conclusion

We considered the interface motion with an obstacle according to the hyperbolic mean curvature flow. In this work, we formally solved the obstacle problem given in equation (2.1). In order to realize this motion, we used the HMBO algorithm as an approximation method. Then, we modified the HMBO algorithm to treat the obstacle problem. The HMBO algorithm requires us to solve the wave equation. Therefore, we modified the wave equation based on the hyperbolic obstacle problem and we got scheme (3.15).

Moreover, we investigated the behaviour of the interface when it hit the obstacle. We considered two cases of the interface motion based on the choice of the initial curve. In the first case, we considered that the initial curve was a closed curve and the obstacle was located inside the curve. In this case, the interface stopped moving and lied on the obstacle after touching it, following the shape of the obstacle. We compared the result from the HMBO algorithm with the solution of equation (4.1) describing circle motion by (1.1) before touching the obstacle. The $L^2$ error and its convergence order agreed with the result of the case of a circle evolving by the hyperbolic mean curvature flow given in [3].

For the second case, the initial curve was fixed at the boundary of the domain and the obstacle was below the curve. After touching the obstacle, the interface reflected and vibrated above the obstacle. At every time we plotted the points where the interface contacted the obstacle, we called it the free boundary shape. We compared the results of the HMBO algorithm for the obstacle problem with the solution of the hyperbolic obstacle problem using scheme (3.14). The slope of the free boundary shape obtained by using the HMBO algorithm coincided with the one using scheme (3.14) as the mesh size becomes smaller. This indicated that the curve motion using both schemes converged to the same result. Also, the slope of the free boundary shape approached the free boundary condition.

## Acknowledgement

The author would like to express her gratitude to Professor Seiro Omata for his support and useful discussions during preparation of this paper. Also, the author would like to thank the Ministry of Research, Technology and Higher Education (RISTEKDIKTI) of the Republic of Indonesia for the support of scholarship.

## References

[1] B. Merriman, J. Bence, S. Osher. *Motion of multiple junctions: a level set approach*. Journal of Computational Physics **112** (1994), 334-363.

[2] E. Ginder, K. Svadlenka. *On an algorithm for curvature-dependent interfacial acceleration*. Proceedings of Computational Engineering Conference, JSCES, Vol.19 (2014).

[3] E. Ginder, K. Svadlenka. *Wave-type threshold dynamics and the hyperbolic mean curvature flow*. Japan Journal of Industrial and Applied Mathematics **33** No.2 (2016), 501-523.

[4] E. Ginder, A. Katayama, K. Svadlenka. *On an approximation method for hyperbolic mean curvature flow*, RIMS Kôkyûroku **1995** (2016), 1–8.

[5] F. Makhrus, Y. Akagawa, A. Syahrini. *Numerical methods for 1-D hyperbolic-type problems with free boundary*. The Science Reports of Kanazawa University **59** (2015), 27-50.

[6] K. Svadlenka, E. Ginder, S. Omata. *A variational method for multiphase volume-preserving interface motions*. Journal of Computational and Applied Mathematics **257** (2014), 157-179.

[7] H. Yoshiuchi. *A numerical solution of film vibration with obstacle*. Dissertation, Kanazawa University (2007).