The Applications of Finite Volume Particle Method for Moving Boundary

Master Thesis

Submitted by
Bishnu Prasad Lamichhane

Supervisors:
Prof. Dr. Helmut Neunzert
Dr. Michael Junk

University of Kaiserslautern
Department of Mathematics
Germany

July 25, 2001

Acknowledgements

First, I want to thank my supervisor Prof. Dr. Neunzert who initiated and motivated me in applied mathematics. I would like to express my deep gratitude to my supervisor Dr. Michael Junk who initiated me in this interesting topic and devoted a lot of his time for giving me guidance for this thesis to be fulfilled. I am also grateful to DAAD for giving me financial support to study in Germany without which there was no chance for me to come here to study mathematics. I would also like to thank my family who give me moral support during my study. Finally, I would like to thank all of the members of Department of Mathematics and members of ITWM who directly and indirectly supported me in course of my study.

Contents

Outline

In this thesis we study the finite volume particle method (FVPM) presented in [3], [6] which has the generic features of particle methods and finite volume methods. Since the last decade, the particle methods for conservations laws are gaining more and more popularity in industrial applications. These particle methods are mesh-free and they are popular in handling the time-dependent problems, complicated geometries and moving domains since the mesh discretization becomes expensive and complicated for such problems. For such problems the scheme based on Lagrangian particles is a good choice, since they are completely mesh-free.

In fact, application of the finite volume methods or finite element methods for the problem which are time-dependent and based on complicated domains which may be changing with respect to time is really expensive and sometime complicated due to the necessity of handling the dynamic data-structures and adaptation of mesh-discretization. That is why in such cases, the particle methods are appropriate where we do not need to discretize the domain with some mesh at all. Our aim here is to derive the FVPM for time-dependent domains and test the scheme for some initial-boundary-value problems with moving boundary and analyze the numerical convergence.

1 Derivation of Finite Volume Particle Schemes in Bounded Domains

In the following we present the derivation of Finite Volume Particle Methods in a bounded domain with moving boundary. This method for Cauchy problem is first presented by Hietel, Steiner and Struckmeier in [3]. The consistency analysis in one dimension is done by Junk and Struckmeier in [6]. We follow almost the similar derivation as given in [6]. This method is mesh-free and based on general partition of unity and standard numerical flux function. In fact, classical finite volume schemes are recovered if we choose special kind of partition of unity. That is why this class of particle method can be thought as the generalization of finite volume schemes.

1.1 Derivation of the Method

We consider here the system of conservation laws in $\Omega(t)$ for the open, bounded and connected set $\Omega(t)$ (called a domain) in $\mathbf{R}^d$

$$\frac{\partial u}{\partial t} + \nabla.F(u) = \mathbf{0} \quad \text{in} \quad \Omega(t) \quad u(0,x) = u^0(x)$$

with suitable boundary conditions, where $u(t,x)$ is the vector of conservative quantities, $u(t,x) \in \mathbf{R}^p$ for $t \geq 0$ and $x \in \mathbf{R}^d$ and $F(u)(t,x)$ is the flux function of the conservation laws. We suppose that there is a continuously differentiable velocity-field $z : \mathbf{R}^+ \times \mathbf{R}^d \to \mathbf{R}$ such that $x_0 \in \partial\Omega(0)$ moves according to $\dot{x} = z(t,x)$, $x(0) = x_0$. We want to give a brief explanation of the basic symbols we have to use throughout this thesis. We assume $M = \{1, \ldots, m\}$ for some $m \in \mathbf{N}$ and take the set of points $\{x_i(t) : i \in M\}$ in $\bar{\Omega}(t)$. To each $x_i(t)$ we associate a function $\Psi_i(t,x) \geq 0$ defined in $\bar{\Omega}(t)$, which will be called particle placed at the point $x_i(t)$. Moreover, we set $\Omega = \{(t,x) : t \in \mathbf{R}^+, x \in \Omega(t)\}$. Now we define the partition of unity in $\bar{\Omega}$

Definition 1 *The set of functions $\{\Psi_i : i \in M\}$ where $\Psi_i : \bar{\Omega} \to \mathbf{R}^+$ will form a partition of unity if $\sum_{i=1}^m \Psi_i(t,x) = 1$ for all $x \in \bar{\Omega}(t)$ and for all $t \in \mathbf{R}^+$.*

These particles may be irregularly spaced and moving. They have overlapping support with one another and they are localized around $x_i(t)$. In the next subsection we will describe how the partition of unity is formed.

Construction of Partition of Unity

We take a Lipschitz continuous function $w : \mathbf{R}^d \to \mathbf{R}^+$ with compact support. We define

$$\Psi_i(t,x) = \frac{w_i(t,x)}{\sigma(t,x)}$$

where

$$\sigma(t,x) = \sum_{i=1}^{m} w_i(t,x).$$

In general, we can set

$$w_i(t,x) = w(A_i(t)(x - x_i(t)))$$

where $A_i(t)$ is symmetric, positive-definite matrix of size d by d for all $t \geq 0$. In the case of constant smoothing length in all directions, $A_i(t) = 1/h_i(t)I$ where I is the identity matrix of size d by d and $h_i(t) > 0$ for all t and for all $i \in M$. For simplicity we take here constant smoothing length for all spatial directions (circular patch) although $h_i(t)$ can be taken as vector giving smoothing length in different spatial directions. Thus

$$w_i(t,x) = w(x - x_i(t), h_i(t))$$

for all $i \in M$. The function $w : \mathbf{R}^d \to \mathbf{R}^+$ is known as the kernel function. For the simplicity of notation we also define the derivative of $w_i(t,x)$ with respect to $h_i(t)$

$$D_h w_i(t,x) = \frac{\partial}{\partial h_i(t)} w(x - x_i(t), h_i(t)).$$

For example, taking some special kernel function w in one dimension defined by

$$w(x) = \begin{cases} x+1 & \text{if } x \in (-1,0] \\ -x+1 & \text{if } x \in (0,1] \\ 0 & \text{else} \end{cases}$$

and some scalar smoothing length h, we get a particular partition of unity. The construction of the function $w_i(t,x)$ and the particle functions $\Psi_i(t,x)$ for irregularly spaced 12 particles are visualized in the Figures 1 and 2. On the x-axis are indicated particle positions x_i and around each point, the function

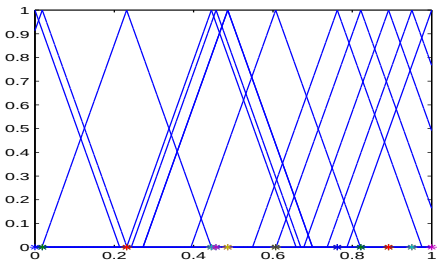

Figure 1: The function $w_i(t,x)$

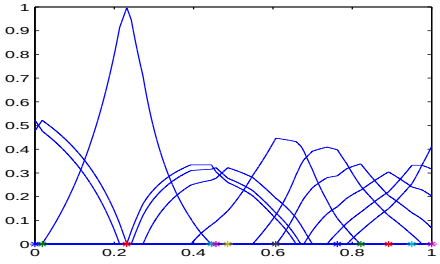

Figure 2: The particle function $\Psi_i(t,x)$

$w_i(t,x)$ are plotted. We have used this kernel in our numerical examples, although other choices are also possible. After dividing by $\sigma(t,x) = \sum_{i=1}^{m} w_i(t,x)$ we get the particle function $\Psi_i(t,x)$.

Remark 2 *:*

(1) The functions Ψ_i in general may be non-symmetric and irregularly spaced and moving.

(2) The supports of Ψ_i overlap, but we want to avoid overlapping of many particles and that is why we choose the compactly supported kernel function w.

(3) The function $\sigma(t,x) = \sum_{i=1}^{m} w_i(t,x)$ must be finite and non-zero for every $x \in \Omega(t)$ and for every $t \geq 0$. Thus $h_i(t)$ should be chosen in such a way that there is no gap in $\Omega(t)$. That means every point $x \in \Omega(t)$ should lie within the support of at least one $\Psi_i(t,x)$ for all $t \geq 0$.

(4) The kernel function w is to be chosen in such a way that $\Psi_i(t,.) \in \mathbf{C}^0(\Omega)$ and $\Psi_i(t,.)$ is piece-wise $\mathbf{C}^1$ in Ω, every piece being a regular domain.

We consider the following initial-boundary value problem of the system of conservation laws in $\Omega(t)$ for the open, bounded and connected set $\Omega(t)$ in $\mathbf{R}^d$

$$\frac{\partial u}{\partial t} + \nabla.F(u) = \mathbf{0} \quad \text{in} \quad \Omega(t) \quad u(0,x) = u^0(x) \tag{1}$$

together with the suitable boundary conditions. Here $u(t,x)$ is the vector of conservative quantities, $u(t,x) \in \mathbf{R}^p$ for $t \geq 0$ and $x \in \mathbf{R}^d$ and $F(u)(t,x)$ is the flux function of the conservation laws.
Now we get the weak formulation of above equation by multiplying both sides by the particle Ψ_i and integrating over the domain $\Omega(t)$

$$\int_{\Omega(t)} \frac{\partial u}{\partial t}\Psi_i(t,x)dx + \int_{\Omega(t)} \nabla.F(u)\Psi_i(t,x)dx = \mathbf{0}.$$

Including Ψ_i in the space and time derivative we obtain

$$\int_{\Omega(t)} \frac{\partial(u\Psi_i)}{\partial t}dx + \int_{\Omega(t)} \nabla.(F(u)\Psi_i)dx - \int_{\Omega(t)} \left(u\frac{\partial \Psi_i}{\partial t} + F(u).\nabla\Psi_i \right) dx = \mathbf{0}. \tag{2}$$

Now we define the discrete particle property

$$u_i(t) = \frac{1}{V_i(t)} \int_{\Omega(t)} \Psi_i(t,x)u(t,x)dx \tag{3}$$

where $V_i(t)$ is the volume of the particle $\Psi_i(t,x)$, given by

$$V_i(t) = \int_{\Omega(t)} \Psi_i(t,x)dx.$$

After multiplying both sides of the equation (3) by $V_i(t)$ and differentiating with respect to t, we get

$$\frac{d}{dt}(u_i(t)V_i(t)) = \frac{d}{dt}\int_{\Omega(t)} \Psi_i(t,x)u(t)dx. \tag{4}$$

We have velocity-field $z(t,x)$ such that $z(t,x)$ gives the velocity-field of the boundary if $x \in \partial\Omega(t)$, then we can use the transport theorem to get

$$\frac{d}{dt}(u_iV_i) = \int_{\Omega(t)} \left(\frac{\partial(u\Psi_i)}{\partial t} + \nabla.(u \otimes z)\Psi_i \right) dx \tag{5}$$

where $u \otimes z$ is the usual dyadic product of two vector-fields defined by $(a \otimes b) = (a_i b_j)_{i,j}$ and divergence applied to matrix A is defined by

$$\nabla.A = \begin{pmatrix} \nabla.A_1 \\ \nabla.A_2 \\ \cdots \\ \nabla.A_p \end{pmatrix}$$

where A_i represents the ith row of the matrix function A of size p by d. Now if we use the results (2) in the equation (5) and use Gauss theorem, we get

$$\frac{d}{dt}(u_i V_i) = \int_{\partial\Omega(t)} (u \otimes z - F(u))\Psi_i.n \, ds + \int_{\Omega(t)} \left(u \frac{\partial \Psi_i}{\partial t} + F(u).\nabla \Psi_i \right) dx. \quad (6)$$

Since generally we suppose that the smoothing length $h_i(t)$ can vary with respect to time and $w_i(t,x) = w(x - x_i(t), h_i(t))$ we have

$$\frac{\partial}{\partial t} w_i(t,x) = -\dot{x}_i(t).\nabla_x w_i(t,x) + D_h w_i(t,x)\dot{h}_i(t).$$

Now we want to prove the following:

Proposition 3 *With the notations as above*

$$\frac{\partial \Psi_i}{\partial t} = \sum_{j=1}^{m} \left[\left(\Psi_i \dot{x}_j . \frac{\nabla_x w_j}{\sigma} - \Psi_j \dot{x}_i . \frac{\nabla_x w_i}{\sigma} \right) - \left(\Psi_i \dot{h}_j \frac{D_h w_j}{\sigma} - \Psi_j \dot{h}_i \frac{D_h w_i}{\sigma} \right) \right].$$

Proof: Here

$$\begin{aligned} \frac{\partial \Psi_i}{\partial t} = -\dot{x}_i(t).\frac{\nabla w_i(t,x)}{\sigma(t,x)} + \frac{w_i(t,x)}{\sigma^2(t,x)} \sum_{j=1}^{m} \dot{x}_j(t).\nabla w_j(t,x) \\ + \dot{h}_i \frac{D_h w_i(t,x)}{\sigma(t,x)} - \frac{w_i(t,x)}{\sigma^2(t,x)} \sum_{j=1}^{m} \dot{h}_j D_h w_j(t,x). \end{aligned}$$

Now we use $\sigma(t,x) = \sum_{j=1}^{m} w_j(t,x)$ to get

$$\begin{aligned} \frac{\partial \Psi_i}{\partial t} = \sum_{j=1}^{m} \left(\Psi_i \dot{x}_j . \frac{\nabla w_j}{\sigma} - \Psi_j \dot{x}_i . \frac{\nabla w_i}{\sigma} \right) \\ - \sum_{j=1}^{m} \left(\Psi_i \dot{h}_j \frac{D_h w_j}{\sigma} - \Psi_j \dot{h}_i \frac{D_h w_i}{\sigma} \right). \end{aligned}$$

■

Proposition 4 *The term $\nabla\Psi_i(t,x)$ can be written as*

$$\nabla\Psi_i(t,x) = \sum_{j=1}^{m}\left(\Psi_j(t,x)\frac{\nabla w_i(t,x)}{\sigma(t,x)} - \Psi_i(t,x)\frac{\nabla w_j(t,x)}{\sigma(t,x)}\right).$$

Proof: Note that

$$\nabla\Psi_i(t,x) = \nabla\left(\frac{w_i(t,x)}{\sigma(t,x)}\right).$$

Now using the product rule, we get

$$\nabla\Psi_i(t,x) = \frac{\nabla w_i(t,x)}{\sigma(t,x)} - w_i(t,x)\frac{\nabla\sigma(t,x)}{\sigma^2(t,x)}.$$

The result follows from the relations $\sigma(t,x) = \sum_{j=1}^{m} w_j(t,x)$ and $\sum_{j=1}^{m}\Psi_j(t,x) = 1$. ■

Here, let us introduce the following notations:

$$\Gamma_{ij}(t,x) = \Psi_i(t,x)\frac{\nabla w_j(t,x)}{\sigma(t,x)}$$

and

$$\alpha_{ij}(t,x) = \Psi_i(t,x)\frac{D_h w_j(t,x)}{\sigma(t,x)}.$$

Using these symbols and the results from the propositions 3 and 4 in the equation (6) and suppressing the arguments t and x we have the equations of motion as

$$\frac{d}{dt}(u_i V_i) = \int_{\partial\Omega(t)}(u\otimes z - F(u))\Psi_i.n\,ds +$$
$$\sum_{j=1}^{m}\int_{\Omega(t)}\left[u\left(\dot{x}_j.\Gamma_{ij} - \dot{x}_i.\Gamma_{ji}\right) + F(u)\left(\Gamma_{ji} - \Gamma_{ij}\right) - u\left(\dot{h}_j\alpha_{ij} - \dot{h}_i\alpha_{ji}\right)\right]dx, \quad (7)$$

$$\frac{d}{dt}x_i(t) = z(t,x_i)$$

for $i = 1, \ldots, m$ where

$$V_i(t) = \int_{\Omega(t)} \Psi_i(t,x)dx. \tag{8}$$

Here we are interested in the particles which are moving with the velocity-field $z(t,x)$. These moving particles are called the Lagrangian particles. If we allow the particles to move, even the volume of the particles may change with respect to time and it is necessary to compute $V_i(t)$ for every particle at each time t. We can compute the volume $V_i(t)$ by using the formula (8). We have exactly used this formula for our numerical computation. However, we can get the additional equation for $V_i(t)$ by differentiating the equation (8) with respect to t, which yields

$$\frac{dV_i(t)}{dt} = \sum_{j=1}^{m} \int_{\Omega(t)} \left[\frac{w_i \nabla w_j}{\sigma^2} \dot{x}_j - \frac{w_j \nabla w_i}{\sigma^2} \dot{x}_i - (\dot{h}_j \alpha_{ij} - \dot{h}_i \alpha_{ji}) \right] dx + \int_{\partial\Omega(t)} \Psi_i z.n \, ds$$

and this equation can also be used to compute the volume of every particle at each time t. Rearranging the terms of the equation (7), we obtain the equations of motion as

$$\begin{aligned} \frac{d}{dt}(u_i V_i) = & \int_{\partial\Omega(t)} (u \otimes z - F(u)) \Psi_i . n ds + \\ & \sum_{j=1}^{m} \int_{\Omega(t)} [(F(u) - u \otimes \dot{x_i}) \Gamma_{ji} - (F(u) - u \otimes \dot{x_j}) \Gamma_{ij}] \, dx - \\ & \int_{\Omega(t)} u(\dot{h}_j \alpha_{ij} - \dot{h}_i \alpha_{ji}) dx, \end{aligned} \tag{9}$$

$$\frac{d}{dt} x_i(t) = z(t, x_i)$$

with the initial value

$$u_i^0 = \frac{1}{V_i(0)} \int_{\Omega(t)} \Psi_i(0,x) u^0(x) dx$$

and the prescribed boundary conditions which we would discuss later. To obtain the solution we use the interpolation formula

$$\tilde{u}(t,x) = \sum_{i=1}^{m} \Psi_i(t,x) u_i(t).$$

From now on we will call

$$B(t,i) = \int_{\partial\Omega(t)} (u \otimes z - F(u))\Psi_i . n ds$$

the boundary term and use the abbreviation FVPM for the Finite Volume Particle Method.

2 Approximation

Let us consider the Lagrangian flux function $G(t,x,u) = F(u) - u \otimes z$ and set $G_i = G(t, x_i, u_i)$ and $\gamma_{ij}(t) = \int_{\Omega(t)} \Gamma_{ij}(t,x)dx$ then, the equation (9) can approximately be written as

$$\frac{d}{dt}(u_i V_i) \approx \sum_{j=1}^{m} \left[(G_i\gamma_{ji} - G_j\gamma_{ij}) - \int_{\Omega(t)} u(\dot{h}_j\alpha_{ij} - \dot{h}_i\alpha_{ji})dx \right] + \int_{\partial\Omega(t)} (u \otimes z - F(u))\Psi_i . n \, ds.$$

Now we use the splitting $ac - bd = \frac{(a-b)(c+d)}{2} + \frac{(a+b)(c-d)}{2}$ which is valid even when a and b are matrices and c and d are vectors, we get

$$\sum_{j=1}^{m} (G_i\gamma_{ji} - G_j\gamma_{ij}) = \sum_{j=1}^{m} \frac{1}{2}[(G_i - G_j)(\gamma_{ij} + \gamma_{ji}) - (G_i + G_j)(\gamma_{ij} - \gamma_{ji})].$$

Suppose that $\mathrm{supp}(\Psi_i) \subset B_d(x_i(t), h_i(t))$ where $B_d(x,h)$ is the d-dimensional ball with radius h and center at x. Then if we take the smoothing length $h_i(t)$ and $h_j(t)$ sufficiently small we can assume that $G_i \approx G_j$ for $\gamma_{ij} + \gamma_{ji} \neq 0$ since $\gamma_{ij} + \gamma_{ji} \neq 0$ implies that x_i and x_j are nearby particles (precisely $\|x_i - x_j\| \leq h_i + h_j$, when $\gamma_{ij} \neq 0$), we can conclude that

$$\begin{aligned} \sum_{j=1}^{m} (G_i\gamma_{ji} - G_j\gamma_{ij}) &\approx \sum_{j=1}^{m} \frac{1}{2}(G_i + G_j)(\gamma_{ji} - \gamma_{ij}) \\ &= -\sum_{j=1}^{m} \|\beta_{ij}\| \frac{(G_i + G_j)}{2} n_{ij} \end{aligned}$$

where $\beta_{ij} = \gamma_{ij} - \gamma_{ji}$ and $n_{ij} = \beta_{ij}/\|\beta_{ij}\|$. Since β_{ij} carry geometrical information of the relative positions of the particles we will call them geometric coefficients. Here $\frac{G_i+G_j}{2}$ is the numerical flux function of central differencing.

A more general approach is obtained if we replace this particular expression by a general numerical flux function, $g_{ij} = g(t, x_i, u_i, x_j, u_j, n_{ij})$ for $G(t, x, u)$ which should be consistent with the Lagrangian flux function $F(u) - u \otimes z$. The general assumptions on the numerical flux function g_{ij} are presented in the appendix A. Let us introduce one more symbol

$$\phi_{ij}(t) = \int_{\Omega(t)} \alpha_{ij}(t, x)dx = \int_{\Omega(t)} \Psi_i \frac{D_h w_j}{\sigma} dx.$$

Since u_i represents the solution averaged with respect to the function Ψ_i, which is given by the equation (3), we can write

$$\int_{\Omega(t)} u(t, x)\alpha_{ij}(t, x)dx \approx u_i(t) \int_{\Omega(t)} \alpha_{ij}(t, x)dx.$$

Then

$$\int_{\Omega(t)} u(\dot{h}_j\alpha_{ij} - \dot{h}_i\alpha_{ji})dx = u_i\dot{h}_j \int_{\Omega(t)} \alpha_{ij}dx - u_j\dot{h}_i \int_{\Omega(t)} \alpha_{ji}dx$$

and this can be written as

$$\int_{\Omega(t)} u(\dot{h}_j\alpha_{ij} - \dot{h}_i\alpha_{ji})dx = u_i\dot{h}_j\phi_{ij} - u_j\dot{h}_i\phi_{ij}.$$

Now suppose that $B_i(t)$ be some discretization of the boundary term

$$B(t, i) = \int_{\partial\Omega(t)} (u \otimes z - F(u))\Psi_i.n\, ds$$

which will now depend on $\{u_i : i \in M\}$ and the prescribed boundary conditions. Thus we end up with a system of ordinary differential equations in vector form

$$\frac{d}{dt}(u_i V_i) = -\sum_{j=1}^{m} \Big[\|\beta_{ij}\| g_{ij} + (u_i\dot{h}_j\phi_{ij} - u_j\dot{h}_i\phi_{ji}) \Big] + B_i, \tag{10}$$

$$\frac{d}{dt}x_i(t) = z(t, x_i) \tag{11}$$

with initial condition

$$u_i^0 = \frac{1}{V_i(0)} \int_{\Omega(0)} \Psi_i(0, x)dx$$

along with the prescribed boundary conditions which should be computed by taking into account the interpolation formula,

$$\tilde{u}(t,x) = \sum_{i=1}^{m} \Psi_i(t,x) u_i(t)$$

and the given boundary conditions. The details about the boundary conditions will be treated later in section 5.

Proposition 5 *The FVPM defined by the equation* (10) *fulfills the discrete balance property*

$$\frac{d}{dt} \sum_{i=1}^{m} (u_i V_i) = \sum_{i=1}^{m} B_i$$

if the numerical flux function g fulfills the condition, $g(t, x_i, u_i, x_j, u_j, n_{ij}) = -g(t, x_j, u_j, x_i, u_i, n_{ji})$ *where* $B_i(t)$ *is a discretization of the boundary term* $B(t,i) = \int_{\partial\Omega(t)} (u \otimes z - F(u)) \Psi_i . n \, ds$.

Proof: We have

$$\frac{d}{dt}(u_i V_i) = -\sum_{j=1}^{m} \left[\|\beta_{ij}\| g_{ij} + (u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ji}) \right] + B_i.$$

Summing over i we find

$$\frac{d}{dt} \sum_{i=1}^{m} (u_i V_i) = -\sum_{i=1}^{m} \left[\sum_{j=1}^{m} \left(\|\beta_{ij}\| g_{ij} + u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ji} \right) - B_i \right].$$

Now using the property $\|\beta_{ij}\| g_{ij} = -\|\beta_{ji}\| g_{ji}$ we obtain

$$\sum_{i=1}^{m} \sum_{j=1}^{m} \|\beta_{ij}\| g_{ij} = 0,$$

and it is simple to observe that

$$\sum_{i=1}^{m} \sum_{j=1}^{m} (u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ji}) = 0$$

which concludes the proof. ■

Lemma 6 *The geometric coefficients can be given by the following formula*

$$\beta_{ij} = \int_{\Omega(t)} (\Psi_i \nabla \Psi_j - \Psi_j \nabla \Psi_i) dx.$$

or equivalently

$$\beta_{ij} = 2 \int_{\Omega(t)} \Psi_i \nabla \Psi_j dx - \int_{\partial\Omega(t)} \Psi_i \Psi_j n \, ds.$$

Proof: We have defined

$$\beta_{ij} = \int_{\Omega(t)} \left(\Psi_i \frac{\nabla w_j(t,x)}{\sigma(t,x)} - \Psi_j \frac{\nabla w_j(t,x)}{\sigma(t,x)} \right) dx.$$

Now if we use $\nabla w_j = \nabla(\sigma \Psi_j) = \Psi_j \nabla \sigma + \sigma \nabla \Psi_j$ in the above equation we get

$$\beta_{ij} = \int_{\Omega(t)} (\Psi_i \nabla \Psi_j - \Psi_j \nabla \Psi_i) dx.$$

We can write this formula for β_{ij} as

$$\beta_{ij} = \int_{\Omega(t)} (-\nabla(\Psi_i \Psi_j) + 2\Psi_j \nabla \Psi_i) dx.$$

The second result follows by using the Gauss divergence theorem. ■

Proposition 7 *The coefficients β_{ij} satisfy*

$$\beta_{ij} = -\beta_{ji} \quad \forall i, j \in M. \tag{12}$$

$$\sum_{j=1}^{m} \beta_{ij} = -\int_{\Omega(t)} \nabla \Psi_i dx \quad \forall i \in M. \tag{13}$$

$$\beta_{ij} = 0 \quad \textit{if} \quad supp(\Psi_i) \cap supp(\Psi_j) = \emptyset. \tag{14}$$

Proof: Properties (12) and (14) are obvious from the definition of β_{ij} and we prove only (13). In Lemma (6) we have proved

$$\beta_{ij} = \int_{\Omega(t)} (\Psi_i \nabla \Psi_j - \Psi_j \nabla \Psi_i) dx.$$

Summing over all j, we get

$$\sum_{j=1}^{m} \beta_{ij} = \sum_{j=1}^{m} \int_{\Omega(t)} (\Psi_i \nabla \Psi_j - \Psi_j \nabla \Psi_i) dx$$

Now we use

$$\sum_{j=1}^{m} \Psi_j = 1 \quad \text{and} \quad \sum_{j=1}^{m} \nabla \Psi_j = 0$$

to get

$$\sum_{j=1}^{m} \beta_{ij} = -\int_{\Omega(t)} \nabla \Psi_i dx.$$

■

Now we have the following corollary

Corollary 8 *If the velocity field $z(t,x) = 0$ in $\Omega(t)$ for all $t \geq 0$ and the smoothing length h_i does not vary with respect to time the property* (13) *in above proposition ensures the preservation of constant state if the discretization of the boundary term fulfills the consistency condition*

$$B_i = F(u_c). \int_{\partial\Omega(t)} \Psi_i(t,x) n \, ds$$

for the constant solution u_c of the initial-boundary-value problem given by (1)*.*

Proof: Assume that the constant u_c solves the given initial-boundary-value problem given by (1). Here we want to show that $u_i = u_c, i = 1, \ldots, m$, is a stationary solution of

$$\frac{d}{dt}(u_i V_i) = -\left[\sum_{j=1}^{m} \|\beta_{ij}\| g_{ij} + (u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ji}) - B_i\right].$$

Suppose

$$R(u_1, \ldots, u_m) = -\left[\sum_{j=1}^{m} \|\beta_{ij}\| g_{ij} + (u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ji}) - B_i\right].$$

To show that u_c is the stationary solution we have to prove $R(u_c, \ldots, u_c) = 0$. Since the particles are not moving and the smoothing length h_i is independent

of time $\phi_{ij} = 0$ for all $i, j \in M$ and due to the consistency of the numerical flux function g_{ij} with the Lagrangian flux function which we have already assumed we get $g_{ij} = F(u_c).n_{ij}$. Now we use the result (13) of the Proposition (7)

$$\sum_{j=1}^{m} \beta_{ij} = -\int_{\Omega(t)} \nabla \Psi_i dx$$

and the consistency condition for the discretization $B_i(t)$ of the boundary term to get

$$R(u_c, \ldots, u_c) = -\left(\int_{\Omega(t)} F(u_c) \nabla \Psi_i dx - \int_{\partial\Omega(t)} F(u_c) \Psi_i . n \, ds \right).$$

Recalling the use of Gauss theorem, we can write the above equation as

$$R(u_c, \ldots, u_c) = -\left(\int_{\Omega(t)} F(u_c) \nabla \Psi_i dx - \int_{\Omega(t)} \nabla . (F(u_c) \Psi_i) dx \right).$$

We rearrange the terms of this equation to obtain

$$R(u_c, \ldots, u_c) = -\left(\int_{\Omega(t)} F(u_c) \nabla \Psi_i - \nabla . (F(u_c) \Psi_i) dx \right).$$

Now using the product rule of differentiation, we get

$$R(u_c, \ldots, u_c) = -\int_{\Omega(t)} (-1)(\nabla . F(u_c)) \Psi_i dx \tag{15}$$

which leads to

$$R(u_c, \ldots, u_c) = 0. \tag{16}$$

■

2.1 Reduction into One Dimension

Since we take model problem in one dimension which we will discuss later, we want to reduce the equation of motion, we have derived, into one dimension. Suppose the domain in one dimension be $\Omega(t) = (a(t), b(t))$. The equations of motion in one dimension are

$$\frac{d}{dt}(u_i V_i) = -\sum_{j=1}^{m} \Big[|\beta_{ij}| g_{ij} + (u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ji}) \Big] + B_i, \tag{17}$$

$$\frac{d}{dt}x_i(t) = z(t, x_i)$$

with initial conditions

$$u_i^0 = \frac{1}{V_i(0)} \int_{\Omega(0)} \Psi_i(0, x) dx$$

for $i = 1, \ldots, m$ along with the prescribed boundary conditions which we will later discuss in details. Here $B_i(t)$ is the discretization of the boundary term $B(t, i) = \int_{\partial\Omega(t)} (uz - F(u))\Psi_i . n \, ds$.
Since ds is the element of the boundary of a one dimensional domain, we have $ds = d\delta_{a(t)} + d\delta_{b(t)}$ where $d\delta$ is the point measure in one dimension. Then the Gauss theorem in one dimension gives $B(t, i) = -\left[(F(u) - uz)\Psi_i\right]_{a(t)}^{b(t)}$, where $\left[(F(u) - uz)\Psi_i\right]_{a(t)}^{b(t)} = (F(u) - uz)\,\Psi_i|_{b(t)} - (F(u) - uz)\,\Psi_i|_{a(t)}$. Hence in one dimension $B_i(t)$ is just the discretization of $-\left[(F(u) - uz)\Psi_i\right]_{a(t)}^{b(t)}$. In the case of constant smoothing length which simply means $\dot{h}_i = 0, \quad \forall i \in M$, the equation of motion is

$$\frac{d}{dt}(u_i V_i) = -\sum_{j=1}^{m} |\beta_{ij}| g_{ij} + B_i$$

with the similar initial and boundary condition as in the general case. If we use the explicit Euler discretization for equation (17) for the time derivative, we get

$$(u_i V_i)^{n+1} = (u_i V_i)^n - \delta t \left[\sum_{j=1}^{m} \left(|\beta_{ij}^n| g_{ij}^n + u_i^n \dot{h}_j^n \phi_{ij}^n - u_j^n \dot{h}_i^n \phi_{ij}^n \right) - B_i^n \right].$$

Remark 9 *In one dimension we construct g_{ij} from a numerical flux function $\tilde{g}(t, x, u, y, v)$ which is consistent with Lagrangian flux function $G(t, x, u)$ for all $t \geq 0$ and $x, y \in \mathbf{R}$ and $u, v \in \mathbf{R}^p$. Then we define g_{ij} by*

$$g(t, x_i, u_i, x_j, u_j, n_{ij}) = \begin{cases} \tilde{g}(t, x_i, u_i, x_j, u_j) & \text{if } n_{ij} = 1 \\ -\tilde{g}(t, x_j, u_j, x_i, u_i) & \text{if } n_{ij} = -1. \end{cases}$$

3 Finite Volume Scheme as a Limit of FVPM

In this section we want to show how the special choice of a partition of unity leads to a finite volume scheme. Suppose the domain $\Omega(t) = (a(t), b(t))$

be subdivided into equal cells (intervals) $(x_0, x_1], (x_1, x_2], \ldots (x_m, x_{m+1})$ where $x_0 = a(t)$ and $x_{m+1} = b(t)$. We will take δx as the length of every subinterval. We choose the regular distribution of the particles, each particle placed in the middle of the cell and the partition of unity is induced by the family of the indicator functions $\{I_i : i \in M\}$ where I_i is the indicator function of the ith cell and $M = \{0, 1, \ldots, m+1\}$. Here by family we mean the set of functions. Thus

$$I_i(t,x) = \begin{cases} 1 & \text{if } x \in (x_{i-1} + \frac{\delta x}{2}, x_{i+1} - \frac{\delta x}{2}], \\ 0 & \text{else} \end{cases}$$

for $i = 1, \ldots, m$ and for $i = 0$ and $m+1$, we have,

$$I_0(t,x) = \begin{cases} 1 & \text{if } x \in (x_0, x_1 - \frac{\delta x}{2}], \\ 0 & \text{else} \end{cases}$$

and

$$I_{m+1}(t,x) = \begin{cases} 1 & \text{if } x \in (x_{m+1} - \frac{\delta x}{2}, x_{m+1}), \\ 0 & \text{else} \end{cases}$$

Our aim here is to compute the geometric coefficients β_{ij} and ϕ_{ij} for the partition of unity formed by the family of indicator functions with the help of partition of unity formed by the family of hat functions. We will prove some Lemmas concerning the computation of coefficients β_{ij} and ϕ_{ij} for the partition of unity formed by hat functions and we will show that the partition of unity formed by the family of regular indicator functions is the limiting case of the partition of unity formed by the family of regularly spaced hat functions when $h \to \delta x/2$. Here, by regular particles, we mean that the particles are placed regularly in $\Omega(t)$. To form the partition of unity from the hat functions we will take the kernel function w as in the first section, defined by

$$w(x) = \begin{cases} x+1 & \text{if } x \in (-1, 0], \\ -x+1 & \text{if } x \in [0, 1], \\ 0 & \text{else} \end{cases}$$

then, set

$$w_i(t,x) = w\left(\frac{x(t) - x_i(t)}{h(t)}\right).$$

The particles are then defined by

$$\Psi_i(t,x) = \frac{w_i(t,x)}{\sigma(t,x)}$$

where

$$\sigma(t,x) = \sum_{i=1}^{m} w_i(t,x).$$

Here we define the k-interaction of the particles

Definition 10 *If the k pairwise distinct particles $\Psi_{i_1}(t,x), \ldots, \Psi_{i_k}(t,x)$ has the property that $\mu(supp(\Psi_{i_1}) \cap \cdots \cap supp(\Psi_{i_k})) > 0$ then we will speak of k-interaction where μ is the Lebesgue measure in $\mathbf{R}^1$.*

Since in one dimension it is necessary that at least two neighboring particles should interact to form the partition of unity we define

Definition 11 *The family of partition of unity $\{\Psi_i(t,x) : i \in M\}$ will be called minimal overlapping family of partition of unity if there is no 3-interaction among the particles and $\Psi_i(t,x)\Psi_j(t,x) = 0$ for all $x \in \partial\Omega(t)$ and for all $i, j \in M, i \neq j$ and for all $t \geq 0$.*

The immediate consequence of Lemma 6 in one dimension is that

Lemma 12 *If the family of partition of unity is minimal overlapping then, for $i \neq j$, β_{ij} can be given by the formula,*

$$\beta_{ij} = 2\int_{\Omega(t)} \Psi_i \frac{\partial \Psi_j}{\partial x} dx. \tag{18}$$

Proof: Since the family of partition of unity is minimal overlapping, we get

$$\int_{\partial\Omega(t)} \Psi_i \Psi_j n \, ds = 0.$$

Now the result follows by using the formula of β_{ij} given in Lemma 6 ■

Definition 13 *A minimal overlapping family of partition of unity with particle independent smoothing length $h(t)$ formed by regularly spaced hat functions ordered according to their positions on the x-axis will be called a* **regular partition of unity**.

Since for the regular partition of unity the particles are ordered according to their positions in the x-axis, intersection of supports of two neighboring particles can be taken as an interval, if it is not empty. We prove the following assertions for the regular partition of unity we have just defined.

Lemma 14 *In the case of regular partition of unity the geometric coefficients β_{ij} are simply given by*

$$\beta_{ij} = \begin{cases} 1 & \text{if } j = i+1, \\ -1 & \text{if } j = i-1, \\ 0 & \text{else.} \end{cases}$$

Proof: Assume that two particles Ψ_i and Ψ_j interact which means $\mu(\text{supp}(\Psi_i) \cap \text{supp}(\Psi_j)) > 0$. If $i = j$, $\beta_{ii} = 0$ for all i by the definition of β_{ij}. If $i \neq j$ then $\Psi_i + \Psi_j = 1$ if $x \in \text{supp}(\Psi_i) \cap \text{supp}(\Psi_j)$. When the family of partition of unity is minimal, we can use the formula (18)

$$\beta_{ij} = 2 \int_{\Omega(t)} \Psi_i \frac{\partial \Psi_j}{\partial x} dx$$

Hence

$$\beta_{ij} = 2 \int_{\Omega(t)} (1 - \Psi_j) \frac{\partial \Psi_j}{\partial x} dx$$

Since the particles are ordered we can assume $[s_1, s_2] = \text{supp}(\Psi_i) \cap \text{supp}(\Psi_j)$, then

$$\beta_{ij} = 2[\Psi_j - \frac{1}{2}\Psi_j^2]_{s_1}^{s_2} = \begin{cases} 1 & \text{if } j = i+1 \\ -1 & \text{if } j = i-1 \\ 0 & \text{else.} \end{cases}$$

■

First we take the smoothing length $h = \delta x$ and then the partition of unity formed by the family of regular hat functions will tend to the partition of unity formed by the family of regular indicator functions when $h \to \frac{\delta x}{2}, h \leq dx$. For the following Lemma we will write $\Psi_i(t, x, h)$, $\beta_{ij}(t, h), w_i(t, x, h)$ and $\phi_{ij}(t, h)$ to represent the explicit dependence of Ψ_i, β_{ij}, w_i and ϕ_{ij} on h.

Lemma 15 *For the regular partition of unity*

$$\Psi_i(t, x, h) = \begin{cases} \frac{x - x_i + h}{2h - \delta x} & \text{if } x \in (x_i - h, x_{i-1} + h], \\ 1 & \text{if } x \in (x_{i-1} + h, x_{i+1} - h], \\ \frac{-x + x_i + h}{2h - \delta x} & \text{if } x \in (x_{i+1} - h, x_i + h], \\ 0 & \text{else.} \end{cases}$$

for $\frac{\delta x}{2} \leq h \leq \delta x$ *and*

$$\Psi_i(t, x, h) \to I_i(t, x)$$

almost everywhere when $h \to \frac{\delta x}{2}$, $h \leq \delta x$.

Proof: First we take the particle $\Psi_i(t, x, h)$ for $1 \leq i \leq m$ and $h \leq \delta x$

$$\Psi_i(t, x, h) = \begin{cases} \frac{w_i(t,x,h)}{w_{i-1}(t,x,h)+w_i(t,x,h)} & \text{if } x \in (x_i - h, x_{i-1} + h], \\ 1 & \text{if } x \in (x_{i-1} + h, x_{i+1} - h], \\ \frac{w_i(t,x,h)}{w_{i+1}(t,x,h)+w_i(t,x,h)} & \text{if } x \in (x_{i+1} - h, x_i + h], \\ 0 & \text{else.} \end{cases}$$

Now

$$w_{i-1}(t, x, h) + w_i(t, x, h) = \frac{2h - \delta x}{h} \quad \text{if } x \in (x_i - h, x_{i-1} + h]$$

and

$$w_{i+1}(t, x, h) + w_i(t, x, h) = \frac{2h - \delta x}{h} \quad \text{if } x \in (x_{i+1} - h, x_i + h].$$

Thus, the particle function Ψ_i is given by

$$\Psi_i(t, x, h) = \begin{cases} \frac{x - x_i + h}{2h - \delta x} & \text{if } x \in (x_i - h, x_{i-1} + h], \\ 1 & \text{if } x \in (x_{i-1} + h, x_{i+1} - h], \\ \frac{-x + x_i + h}{2h - \delta x} & \text{if } x \in (x_{i+1} - h, x_i + h], \\ 0 & \text{else.} \end{cases}$$

If $h \to \frac{\delta x}{2}$, the intervals, $(x_i - h, x_{i-1} + h]$ and $(x_{i+1} - h, x_i + h]$ are empty. Hence

$$\Psi_i(t, x, h) \to I_i(t, x)$$

almost everywhere when $h \to \frac{\delta x}{2}$. The similar reasoning holds for $i = 0$ and $i = m + 1$. ■

When we decrease the smoothing length, how the partition of unity formed by the family of regularly spaced hat functions with $\frac{\delta x}{2} \leq h \leq \delta x$, approximate the partition of unity formed by the family of indicator functions of each cell can be seen in the Figures (3).

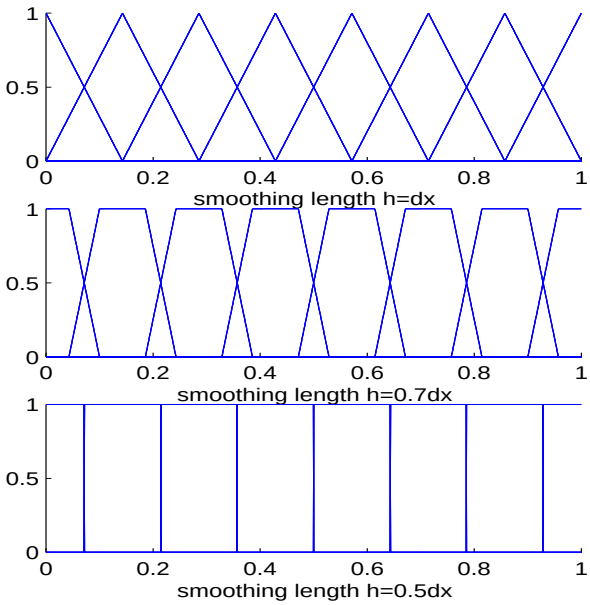

Figure 3: The partition of unity formed by hat functions with greater and lower smoothing length

We have already shown that

$$\beta_{ij}(t,h) = \begin{cases} 1 & \text{if } j = i+1, \\ -1 & \text{if } j = i-1, \\ 0 & \text{else.} \end{cases}$$

Hence $\beta_{ij}(t,h)$ are independent of h for regular partition of unity and so we can define

$$\beta_{ij}(t, \frac{\delta x}{2}) = \lim_{h \to \frac{\delta x}{2}} \beta_{ij}(t,h).$$

Lemma 16 *For the regular partition of unity*

$$\phi_{ij} = \begin{cases} \frac{h+\delta x}{6h} & \textit{if } j = i+1, \textit{and } j = i-1 \\ 0 & \textit{else} \end{cases}$$

for all $i, j \in M, i \neq j$ *and* $\delta x \geq h \geq \frac{\delta x}{2}$.

Proof: Here, $\phi_{ij}(t,h)$ are definded by

$$\phi_{ij}(t,h) = \int_{\Omega(t)} \frac{\Psi_i(t,x,h)}{\sigma(t,x,h)} \frac{\partial w_j(t,x,h)}{\partial h} dx$$

Suppose $j = i - 1$. Using the first result of Lemma 15

$$\phi_{ij}(t,h) = \int_{x_i-h}^{x_{i-1}+h} \frac{(x - x_i + h)}{(2h - \delta x)^2} \frac{(x - x_{i-1})}{h} dx.$$

A simple computation shows that

$$\phi_{ij}(t,h) = \frac{h + \delta x}{6h}.$$

We can use similar computation for $j = i + 1$ and the other cases are obvious. ■

Hence, using $h \to \delta x/2$ we get, $\phi_{ij} = \frac{1}{2}$, for the partition of unity formed by the family of indicator functions. We are interested here to change the smoothing length $h_i(t)$ according to the compression and the expansion of the domain due to the movement of the boundary but independent of all particles. Therefore, we can write $h(t)$ for $h_i(t)$ and thus $h(t) = (b(t) - a(t))/m$, where $m + 1$ is the number of particles we use for computation. Then $\dot{h}(t) = (v_b(t) - v_a(t))/m$. Here $b(t)$ and $a(t)$ are the positions and $v_b(t)$ and $v_a(t)$ are the velocities of the right and left boundary respectively. The particle velocity-field $z(t,x)$ is chosen in such a way that the particles are always equidistant and the function $z(t,x)$ satisfies

$$z(t,x) = \begin{cases} v_a(t) & \text{for } x = a(t) \text{ and} \\ v_b(t) & \text{for } x = b(t). \end{cases}$$

The equation of motion for variable smoothing length is given by

$$\frac{d}{dt}(u_i V_i) = -\sum_{j=0}^{m+1} \Big[|\beta_{ij}| g_{ij} + (u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ij}) - B_i \Big] \tag{19}$$

where $B_i(t)$ is the discretization of

$$\begin{aligned} B^{fv}(t,i) &= \lim_{h \to \frac{\delta x}{2}} \int_{\partial\Omega(t)} (uz - F(u)) \Psi_i(t,x,h).n \, ds \\ &= \lim_{h \to \frac{\delta x}{2}} [(uz - F(u)) \Psi_i(t,x,h)]_{a(t)}^{b(t)}. \end{aligned}$$

Here we can set $B_i(t) = 0$ for $i = 1, \ldots, m$ because we have

$$\lim_{h \to \frac{\delta x}{2}} [(F(u) - uz) \Psi_i(t,x,h)]_{a(t)}^{b(t)} = 0$$

for $i = 1, \ldots, m$. Now using the values of β_{ij} and ϕ_{ij} in the equation of motion (19) and using the construction of the numerical flux function as described in remark 9 the equation of motion for regular partition of unity is reduced to

$$\frac{d}{dt}(u_i V_i) = -[\tilde{g}_{i,i+1} - \tilde{g}_{i-1,i} + \frac{\dot{h}(t)}{2}(2u_i - u_{i-1} - u_{i+1})] + B_i$$
$$\frac{d}{dt}(x_i) = z(t, x_i)$$
$$V_i(t) = h(t)$$

for $i = 0, \ldots, m+1$, with initial condition $u_i^0 = u^0(x_i)$ and boundary conditions $v(t, x_0) = v_a(t)$ and $v(t, x_{m+1}) = v_b(t)$. Here, the term $\frac{\dot{h}(t)}{2}(2u_i - u_{i-1} - u_{i+1})$ in the scheme can work as the anti-diffusive or diffusive agent according as the sign of $\dot{h}$. However, this probably does not affect the stability of the scheme as long as the integral

$$\lim_{t \to \infty} \int_0^t \dot{h}(s) ds$$

stays close to zero which is certainly true for small h and slow periodic change of h.

4 Modeling and Model Problem

We have a cylindrical tube of length L filled with gas and a piston is moving backward and forward in the left side of the tube whereas the right side is kept fixed on the wall. (see Figure 4)

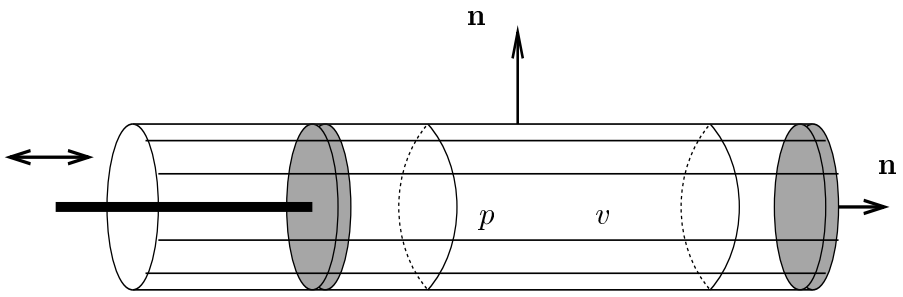

Figure 4: Model problem for the linear equations of acoustics

In the Figure 4, p, v and $\mathbf{n}$ represents pressure, velocity and outer normal respectively. When the piston will start moving forth the waves of pressure

will be propagated since the gas inside the tube will be compressed. Indeed, when the piston is moving back and forth the gas will be somewhere compressed and somewhere rarefied inside the tube giving the variation of pressure inside it. Because of axial symmetry, we just treat this problem as one-dimensional problem. To facilitate the exact solution, we have chosen that the piston is moving backward and forward with constant velocity. More natural choice would be a smooth motion. Then we will see the wave of velocity and pressure running to the right boundary and reflecting back after they hit the right boundary. The position and velocity of the piston with respect to time is visualized in the Figure 5.

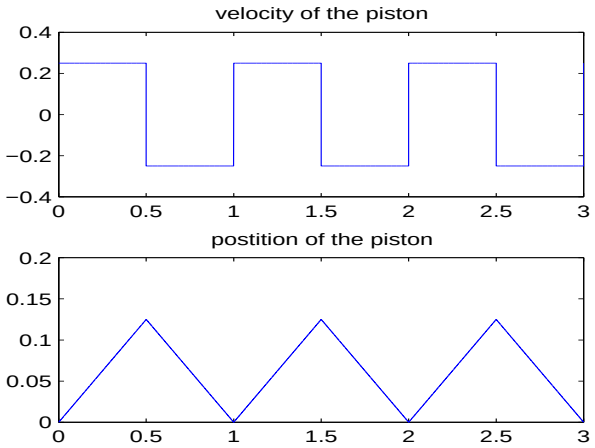

Figure 5: The position and velocity of the piston

This type of physical problem can be modeled by the linearized isentropic Euler systems in one dimension if the velocity of the piston is much less than the velocity of the sound. Our aim is to apply the numerical scheme we have derived (i.e. FVPM) for the isentropic Euler equations in one dimension linearized around some constant solution in the case of moving boundary to compute the velocity and pressure inside the cylinder at any time t and compare the result with finite volume case and finite difference case. The isentropic Euler system in one dimension is given by

$$\begin{pmatrix} \rho \\ \rho v \end{pmatrix}_t + \begin{pmatrix} \rho v \\ \rho v^2 + p \end{pmatrix}_x = \mathbf{0}. \tag{20}$$

Where ρ is the density, v is the velocity and p is the pressure. Now we use the constitutive relation $p = c\rho^k$ to write down the above equation in the form

$$\frac{\partial p}{\partial t} + v\frac{\partial p}{\partial x} + kp\frac{\partial v}{\partial x} = 0, \qquad \frac{\partial v}{\partial t} + v\frac{\partial v}{\partial x} + (\frac{p}{c})^{-1/\kappa}\frac{\partial p}{\partial x} = 0.$$

We wish to consider the equations satisfied by a small perturbation of the constant solution $p(t,x) = p_0$ and $u(t,x) = 0$. Thus we look for the perturbed pressure and the velocity of the form $p(t,x) = p_0 + \tilde{p}$ and $u(t,x) = \tilde{v} + 0$. Similarly the velocity of the piston is also much less than the velocity of sound. That is, $|z(t,x)|$ is much less than the velocity of the sound. We also introduce $c_0 = \sqrt{p'(\rho_0)}$ the sound speed in the gas. We have the relation $p'(\rho) = c\kappa\rho^{\kappa-1}$. Now if we linearize the isentropic Euler equation around $p(t,x) = p_0$ and $v(t,x) = 0$ we get the linear system

$$\frac{\partial \tilde{p}}{\partial t} + kp_0\frac{\partial \tilde{v}}{\partial x} = 0, \qquad \frac{\partial \tilde{v}}{\partial t} + (\frac{p_0}{c})^{-1/\kappa}\frac{\partial \tilde{p}}{\partial x} = 0.$$

If we remove the tilde and write the above equation as a system we get

$$\frac{\partial u}{\partial t} + A\frac{\partial u}{\partial x} = 0$$

where

$$A = \begin{bmatrix} 0 & kp_0 \\ (\frac{p_0}{c})^{-1/\kappa} & 0 \end{bmatrix} \quad \text{and} \quad u = \begin{pmatrix} p \\ v \end{pmatrix}.$$

We have the following initial and boundary conditions

$$\begin{aligned} v(t, a(t)) &= v_a(t) \\ v(t,1) &= 0 \\ v(0,x) &= 0 \\ p(0,x) &= 1 \end{aligned}$$

where $a(t)$ and $v_a(t)$ are the position and velocity of the piston respectively with respect to time. We want to make the equation dimensionless, and therefore, introduce the following dimensionless variables $\bar{t} = \frac{t}{T}, \bar{x} = \frac{x}{L}, \bar{p} = \frac{p}{\kappa p_0}, \bar{v} = \frac{Tv}{L}$. Then the above system becomes

$$\begin{aligned} \frac{\partial \bar{p}}{\partial \bar{t}} + \frac{\partial \bar{v}}{\partial \bar{x}} &= 0 \\ \frac{\partial \bar{v}}{\partial \bar{t}} + (\frac{p_0}{c})^{-1/\kappa}\frac{\kappa p_0 T^2}{L^2}\frac{\partial \bar{p}}{\partial \bar{x}} &= 0. \end{aligned}$$

But the sound speed is given by $c_0^2 = \kappa p_0(\frac{p_0}{c})^{-1/\kappa}$. Therefore, using $c_0^2 = \kappa p_0(\frac{p_0}{c})^{-1/\kappa}$ we get

$$\frac{\partial \bar{p}}{\partial \bar{t}} + \frac{\partial \bar{v}}{\partial \bar{x}} = 0$$
$$\frac{\partial \bar{v}}{\partial \bar{t}} + \frac{c_0 T^2}{L^2}\frac{\partial \bar{p}}{\partial \bar{x}} = 0.$$

If we choose T and L such that $\frac{c_0^2 T^2}{L^2} = 1$, that is , we scale the characteristic speed to one and remove the bar and use the same symbol p, v, x and t again, our system becomes

$$\frac{\partial u}{\partial t} + A\frac{\partial u}{\partial x} = 0$$

where

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{and} \quad u = \begin{pmatrix} p \\ v \end{pmatrix}. \tag{21}$$

We want to solve this equation in the case of moving boundary as described above with the given initial value and boundary values. The speed of the piston is taken to be much less than one.

4.1 Derivation of Numerical Flux

In this subsection, we want to derive the numerical flux function for FVPM and upwind scheme. In the case of FVPM, our numerical flux function which is given by $g_{ij} = g(t, x_i, u_i, x_j, u_j, n_{ij})$ should be consistent with the Lagrangian flux-function $G(t, x, u) = F(u) - uz(t, x)$ where $F(u) = Au$ and $z(t, x)$ is the velocity-field of the particles. For FVPM and finite difference scheme, we want to take the upwind numerical flux function consistent with $G(t, x, u)$ and $F(u)$ respectively. To derive the numerical flux function we need to analyze the eigenvalues of the matrix $(A - zI)$, where

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

and I is the 2 by 2 identity matrix. The velocity-field of the particles is chosen as follows:

$$z(t, x) = \begin{cases} v_a(t) & \text{for } x = a(t) \text{ and} \\ 0 & \text{for } x = 1 \end{cases}$$

and we choose the velocity-field of particles inside the domain by using some interpolation between these values. This is natural since one of our boundary is moving and we can imagine the particles to be compressed and expanded due to the movement of the boundary as the compression and expansion of a spring from one side. Now let us set $\tilde{A} = (A - zI)$ and hence the eigenvalues of $\tilde{A}$ are $\lambda_1 = 1 - z$ and $\lambda_2 = -1 - z$. Since we have scaled the characteristic speed to one, the velocity of the particles should be much less than one and thus the eigenvalues satisfy $1 - z > 0$ and $-1 - z < 0$. In the upwind method one-sided stencil points in the 'upwind' direction from which the characteristic information propagates. Thus when applying upwind scheme to the linear system with eigenvalues of mixed signs, we should take into account the correct direction of propagation and the stencil should also point to this direction. This is exactly the Godunov method applied to linear system which is discussed in [8]. Since the characteristic speed of the system is both negative and positive, we have to decompose the matrix $\tilde{A}$ into two matrices $\tilde{A}^+$ and $\tilde{A}^-$, so that $\tilde{A}^+$ has only positive eigenvalues and $\tilde{A}^-$ has only negative eigenvalues and $\tilde{A} = \tilde{A}^+ + \tilde{A}^-$. Here the characteristic lines go into both directions and hence the information is propagated towards both directions. That is why we can not prescribe both pressure and velocity on the same boundary. Only one can be prescribed on one boundary. Indeed, if we prescribe both variables on the boundary $x = 0$, the solution does not exist in the case of negative eigenvalues of the matrix $\tilde{A}$. We take the matrix R consisting of right eigenvectors of $\tilde{A}$ ordered in the same way as the eigenvalues. Then

$$R = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \text{and} \quad R^{-1} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

Now we get $R^{-1}\tilde{A}R = \Lambda = \text{diag}(1 - z, -1 - z)$. Multiplying the right side of this equation by R^{-1} and the left side by R, we get, $\tilde{A} = R\Lambda R^{-1}$. Assume that for the eigenvalues λ_1 and λ_2 of the matrix $\tilde{A}$, $\lambda_i^+ = \max(\lambda_i, 0)$ and $\lambda_i^- = \min(\lambda_i, 0)$. We define $\tilde{A}^+ = R\Lambda^+R^{-1}$ and $\tilde{A}^- = R\Lambda^-R^{-1}$ where $\Lambda^+ = \text{diag}(\lambda_1^+, \lambda_2^+)$ and $\Lambda^- = \text{diag}(\lambda_1^-, \lambda_2^-)$. Then

$$\Lambda^+ = \begin{bmatrix} 1 - z & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \Lambda^- = \begin{bmatrix} 0 & 0 \\ 0 & -1 - z \end{bmatrix}.$$

Hence we get,

$$\tilde{A}^+ = \frac{(1 - z(t,x))}{2} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad \tilde{A}^- = \frac{(1 + z(t,x))}{2} \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}.$$

The matrices $\tilde{A}^+$ and $\tilde{A}^-$ has the required properties as asserted above. Now we define $\tilde{g}(t, x_i, u_i, x_j, u_j) = \tilde{A}^+ u_i + \tilde{A}^- u_j$. Hence the numerical flux function for FVPM is defined by

$$g(t, x_i, u_i, x_j, u_j, n_{ij}) = \begin{cases} \tilde{g}(t, x_i, u_i, x_j, u_j) & \text{if } n_{ij} = 1 \\ -\tilde{g}(t, x_j, u_j, x_i, u_i) & \text{if } n_{ij} = -1. \end{cases}$$

This numerical flux function is consistent with Lagrangian flux function $F(u) - uz$. The consistency is clear since $g(t, x, u, x, u, n) = (F(u) - uz)n$. The numerical-flux for the upwind difference scheme can be obtained by simply setting $z(t, x) = 0$. Thus the numerical flux function for upwind difference scheme is given by

$$g(u_i, u_{i+1}) = (A^+ u_i + A^- u_{i+1})$$

where

$$\tilde{A}^+ = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad \tilde{A}^- = \frac{1}{2} \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}.$$

5 Treatment of Boundary Conditions

5.1 Method of Characteristics

The term $B_i(t)$ is an approximation of boundary term $B(t, i) = \int_{\partial\Omega(t)} (u \otimes z - F(u))\Psi_i . n \, ds$, i.e., the boundary term should be computed by using the value of the solution on the boundary at every time-step. For our model problem only the velocity is prescribed on the boundary and the pressure is not known. Therefore we have to use some techniques to extract the pressure on the boundary. We can use the information from characteristics to get the pressure on the boundary, which can be obtained by following the characteristic line one time-step backward. Since we know the solution on $\Omega(t)$ at time $t = T - \delta t$ we can trace back following the characteristic line and get the pressure on the boundary at time $t = T$. However, the numerical experiments shows that this method is quite sensitive to the approximation of the boundary values and for our model problem as soon as the particle distribution is quite irregular we see the strange behavior around the boundary which is due to the poor approximation of the pressure on the boundary by this method. The problem can be seen in Figure 6. In the Figure 6 the upper wave is the pressure wave and the lower wave is the velocity wave. The numerical experiment also shows

that this method is more sensitive with respect to time-step in comparison to other methods which we will discuss later. The value of pressure which is approximated by the method of characteristic on the boundary along with the prescribed velocity is given by the formula

$$\begin{pmatrix} p(t,1) \\ v(t,1) \end{pmatrix} = \begin{pmatrix} p(t-dt,1-dt)+v(t-dt,1-dt) \\ 0 \end{pmatrix} \tag{22}$$

for the boundary $x = 1$ and

$$\begin{pmatrix} p(t,a(t)) \\ v(t,a(t)) \end{pmatrix} = \begin{pmatrix} p(t-dt,a(t)+dt)-v(t-dt,a(t)+dt)+v_a(t) \\ v_a(t) \end{pmatrix} \tag{23}$$

for the boundary $x = a(t)$. Now the value of $B_i(t)$ is computed by using these values in the expression

$$\int_{\partial\Omega(t)} (u \otimes z - F(u))\Psi_i.nds.$$

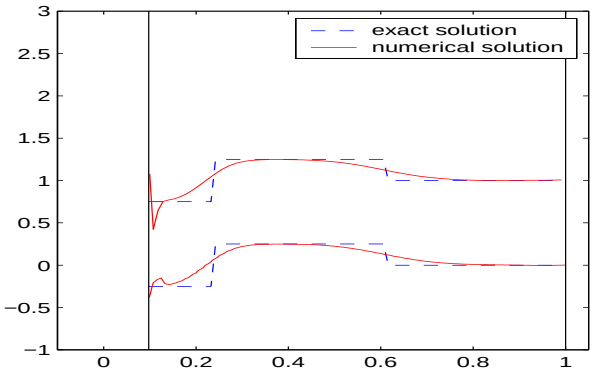

Figure 6: Problem on the boundary in the characteristic methods

5.2 Backward Method

Since in some cases the method of characteristic will not give the result with desired accuracy and may be difficult to apply, the boundary values of the variable which are not prescribed on the boundary can also approximated by the backward method. In this method , the boundary term is computed by using the values of the variables from one time-step back. In this case we propose to approximate the boundary term $B(t,i) = \int_{\partial\Omega(t)}(u\otimes z - F(u))\Psi_i.n\,ds$ (i.e. compute $B_i(t)$) in the following way:

- If the variable is not prescribed on the boundary, use the computed value of this variable on the boundary from one time-step back.
- If the variable is prescribed on the boundary, use the boundary value evaluating at time $t = T - \delta t$.

However, we get the same defect also in the backward method as in the characteristic method.

5.3 Boundary Particles or Ghost Particles

There is also another approach to treat the boundary value problem in hyperbolic partial differential equations which is more general than the method of characteristics and backward methods. We can introduce extra particles near the boundary to maintain the boundary flux and then the boundary term can be completely dropped. The particles which are placed nearest to the boundary are more important than the other particles for the treatment of the boundary conditions, since they carry the boundary information in the numerical schemes. We will call them either boundary particles if they are inside the domain or on the boundary, or the ghost particles if they are outside the domain. In FVPM, we can use either the ghost particles or the boundary particles to approximate the boundary term. We partition the collection of indices M of all particles into two disjoint sets N and D, $M = N \cup D$, such that $i \in D$ means the particle Ψ_i is the ghost particle or the boundary particle and $i \in N$ means that Ψ_i is inner particle (i.e. not ghost particle or boundary particle). One can prescribe the particle properties for the ghost particle or the boundary particles in such a way that the given boundary conditions are fulfilled. We have implemented this idea and this can easily be generalized to higher dimensional problems and even non-linear cases. We choose the particle properties for the ghost particles or the boundary particles in such a way that the reconstructed solution

$$\tilde{u}(t,x) = \sum_{i \in M} \Psi_i(t,x) u_i(t)$$

satisfies the boundary conditions. Thus after the introduction of the boundary particles or the ghost particles our numerical FVPM scheme in one dimension becomes

$$\frac{d}{dt}(u_i V_i) = -\sum_{j=0}^{m+1} \left[|\beta_{ij}| g_{ij} + u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ji} \right]$$

for $i = 1, \ldots, m$ where we introduce the new particles at x_0 and x_{m+1}. Here $M = \{0, \ldots, m+1\}$, $N = \{1, \ldots, m\}$ and $D = \{0, m+1\}$. In the case of ghost particles x_0 and x_{m+1} will lie outside the domain but near the boundary and in the case of boundary particles x_0 and x_{m+1} will lie exactly on the boundary, or inside the domain. We update the particle properties inside the domain for all particles except boundary particles or the ghost particles by the numerical scheme and we use the boundary conditions to get the particle properties for the ghost particles or for the boundary particles. In one dimension it is quite simple, however the same idea works also for higher dimensional cases. For the boundary particles or the ghost-particles we follow the following procedures:

- We get the values $u_1, \ldots, u_m$ by the numerical schemes.
- We have the interpolation formula for the solution

$$\tilde{u}(t, x) = \sum_{i=0}^{m+1} \Psi_i(t, x) u_i(t). \tag{24}$$

- For all boundary particles or the ghost particles we choose the particle properties by solving the linear system which comes from imposing the boundary conditions on (24).

In this modification of the scheme, if we introduce the ghost particles it is necessary to extend the partition of unity in the outer neighborhood of the boundary $\partial\Omega(t)$. Indeed, the partition of unity should cover also the ghost particles. However, in case of boundary particles it is not necessary to extend the partition of unity, since all the particles stay inside $\bar{\Omega}(t)$ in this case. Thus we get the linear system which should give particle properties for all boundary particles or for all ghost particles. Since generally in hyperbolic problems the boundary conditions for all variables can not be prescribed, we do not get the value of all variables on the boundary and the linear system may be under-determined. This problem can be overcome in some cases where the boundary conditions for the variables which are not prescribed on the boundary can be found by using the equation itself or where the boundary values of these variables can be approximated by the methods of characteristics which we have already discussed. However, in general case one has to recourse to the idea of extrapolation. We get the values of the variables which are not prescribed on the boundary by using the values of the variables we have inside the domain by using some extrapolation techniques. Also for our model problem we have to find the right boundary value for the pressure, since only the velocity is

prescribed on the both boundaries. However, we find the boundary conditions for the pressure by using the equation itself. Using the equation (21) we get,

$$\frac{\partial v}{\partial t} = -\frac{\partial p}{\partial x}$$

and thus

$$\frac{\partial v}{\partial t}|_{x=a(t)} = -\frac{\partial p}{\partial x}|_{x=a(t)} \quad \text{and} \quad \frac{\partial v}{\partial t}|_{x=1} = -\frac{\partial p}{\partial x}|_{x=1}.$$

Then after getting the values of all variables on the boundary, the linear system should be uniquely solvable. Now we have the following scheme:

$$\frac{d}{dt}(u_i V_i) = -\sum_{j=0}^{m+1} \left[|\beta_{ij}| g_{ij} + u_i \dot{h}_j \phi_{ij} - u_j \dot{h}_i \phi_{ji} \right] \tag{25}$$

for $i = 1, \ldots, m$ with the initial values $u_i^0 = \int_{\Omega(0)} \Psi_i(0, x) dx / V_i(0)$ along with the following boundary conditions

$$u_0 = f_l(u_1, \ldots, u_m)$$

$$u_{m+1} = f_r(u_1, \ldots, u_m)$$

where f_l and f_r depends on the boundary conditions prescribed for the variables. The values u_0 and u_{m+1} are obtained by solving the linear system. To write the linear system explicitly, let us suppose $u_i = (p_i, v_i)^T$, where p_i and v_i are the particle properties for the pressure and the velocity respectively. Thus we want to compute $p_b = (p_0, p_{m+1})^T$ and $v_b = (v_0, v_{m+1})^T$. Let us write $a_i = \Psi_i(t, x_1) - \Psi_i(t, x_0)$, $b_i = \Psi_i(t, x_{m+1}) - \Psi_i(t, x_m)$, $A_0 = -\sum_{i=1}^m p_i a_i$ and $B_0 = -\sum_{i=1}^m p_i b_i$. Using the numerical differentiation of the pressure at the boundary we get the conditions $p(t, x_0) = p(t, x_1)$ and $p(t, x_m) = p(t, x_{m+1})$ which gives

$$A_p p_b = (A_0, B_0)^T$$

where

$$A_p = \begin{bmatrix} a_0 & a_{m+1} \\ b_0 & b_{m+1} \end{bmatrix}.$$

To get the particle properties for the velocity at the boundary we use the symbols $A_1 = v_a(t) - \sum_{i=1}^m v_i \Psi_i(t, a(t))$ and $B_1 = -\sum_{i=1}^m v_i \Psi_i(t, 1)$, we have the linear system,

$$A_v v_b = (A_1, B_1)^T$$

where

$$A_v = \begin{bmatrix} \Psi_0(t, a(t)) & \Psi_{m+1}(t, a(t)) \\ \Psi_0(t, 1) & \Psi_{m+1}(t, 1) \end{bmatrix}.$$

The numerical experiment shows that this method is highly robust against the variation of time-step and the particle distribution. Indeed, we have always used $\delta t = h/2$, where h is the smoothing length. If we use the same time-step and same smoothing length with same distribution of the particles we see the oscillation in the characteristic method and backward method whereas we do not see any oscillation in the boundary particle or ghost particle case which can be seen in the Figure 7. The solution oscillates when the piston was returning back. The idea of putting the boundary particles or the ghost particles is a natural generalization of the similar idea in finite volume method to approximate the boundary term. However, putting the ghost particles may be problematic if the computational domain is complicated. If the two boundaries of the domain are very near and we have to put the ghost particles for both of them, we have to be very careful about the interaction of the ghost particles of the two boundaries.

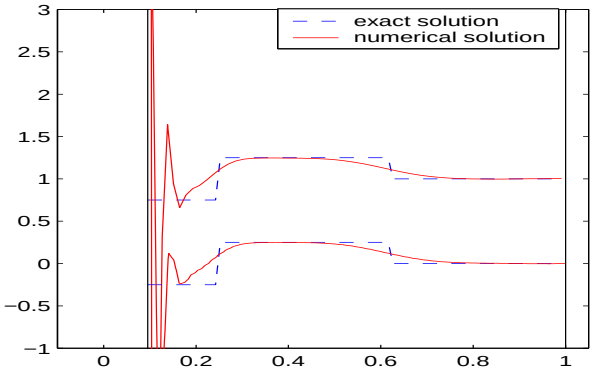

Figure 7: Oscillation in characteristic method

6 Numerical Results

In this section we will present all the numerical solutions we have computed by using different schemes. Indeed, we have compared the numerical results from finite difference scheme, finite volume schemes and FVPM. We have also compared various results from FVPM exploiting its flexibilities. In all of the

figures below the wave up is the pressure-wave and the wave down is the velocity-wave. Since finite difference scheme is the simplest of all the schemes, first of all, we implement the finite difference methods for our model problem.

6.1 Numerical Results from Finite Difference Scheme

We use the upwind numerical flux function that we derived in section 4.1. When applying the finite difference method in moving boundary one has to change the mesh at every time step. Here, instead of re-meshing whole domain at every time, we only change the mesh in the neighborhood of the moving boundary. When the boundary is moving forward or backward, we delete the mesh-point nearest to the boundary, if the distance of this point from the boundary is less than the half of the grid-size and otherwise, keep it. Since we fix the grid, we may need the information from some points behind the boundary. This we have obtained by some extrapolation. We also implement the other scheme, which is similar to the above method in all respects, except that it always uses one point exactly on the boundary and hence it is not necessary to use the extrapolation in this method. This method will deform the stencil around the moving boundary. Both of the schemes seem to be equally effective and we have presented the numerical results from these schemes in the following figures. In fact, we implemented also some other techniques, among which, re-meshing around the moving boundary seems to be the best choice. The pressure and the velocity waves run from the left moving boundary until they reach the right boundary and they reflect back. The Figure 8 shows how the stencil looks like around the boundary in both schemes. It is clear from the figure that we need to extrapolate the value in the first scheme at one point, since the stencil demands the value at that point.

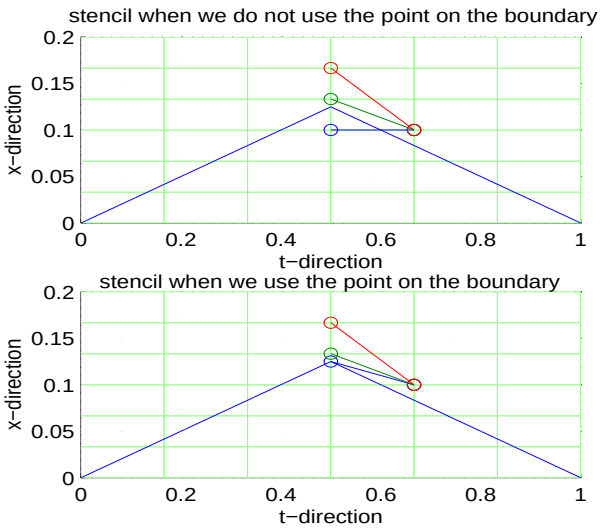

Figure 8: The stencil for both schemes

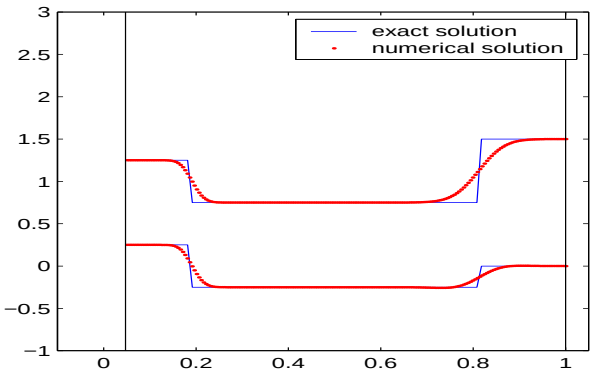

Figure 9: When we do not use the point exactly on the boundary

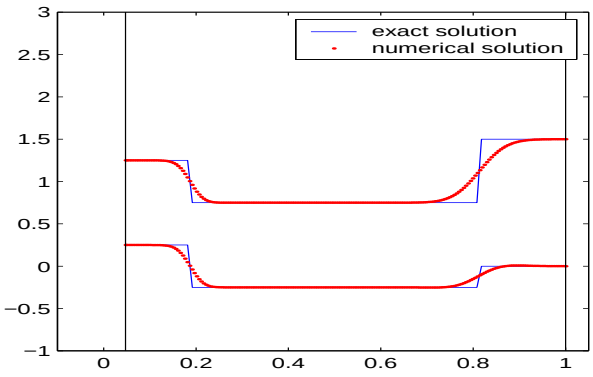

Figure 10: When we use the point exactly on the boundary

In the Figure 9, we have plotted the result of the scheme which does not use the point exactly on the boundary, but use the boundary-information by extrapolation. In the Figure 10, we have plotted the results of the scheme which always uses the point from the boundary and so we do not need to do extrapolation. The numerical results are derived using 200 points in space and CFL 0.5. The plotted solution is at the time $t = 1.186$. Although we do not see any difference in the results from these two different techniques here, if we plot the results at the time when the wave is just moving ahead of the piston or just returning back we can see that the extrapolation can cause small oscillation around the jump which can be seen from the Figures 11 and 12. That is why it is advisable to use the points on the boundary rather than fix the grid and use extrapolation. To show the difference more clearly we have used only 100 grid points here and we plot the solution at time $t = .6$ when the piston was just returning back. If we compare the Figure 11 with Figure 12 we see a small oscillation of the solution near the left jump in the Figure 11, which is due to the extrapolation. However, the oscillation will die down after some time.

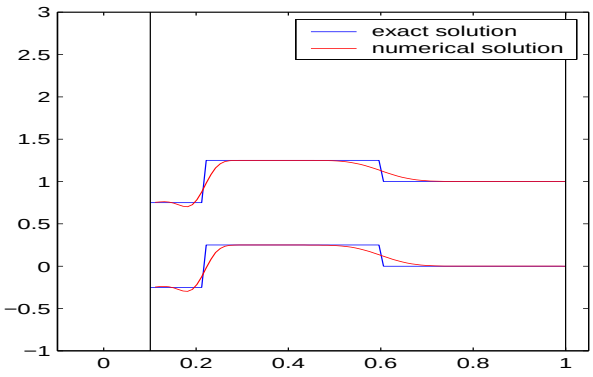

Figure 11: When we do not use the point exactly on the boundary

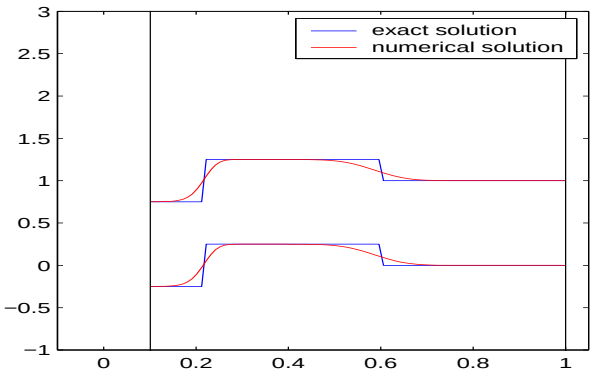

Figure 12: When we use the point exactly on the boundary

6.2 Numerical Results from FVPM

In this section we want to present the numerical results from FVPM. Since there are a lot of flexibilities one can implement in FVPM, we have also used several methods which we want to discuss here. Flexibilities in FVPM are due to the followings:

- Treatment of the boundary conditions

- Smoothing length
- Particle-distribution
- Velocity-field of the particles

Here we want to concentrate on the first three items. In fact, in our opinion the velocity-field of the particles does not influence the solution too much if one chooses a reasonable velocity-field. In section 4.1, we have already discussed how we choose our velocity-field. For all numerical results we have presented here we have used the explicit Euler discretization for the time derivative. Thus the numerical scheme we have derived in section 2 looks like

$$(u_iV_i)^{n+1} = (u_iV_i)^n - \delta t \left[\sum_{j=1}^{m} \left(|\beta_{ij}^n| g_{ij}^n + u_i^n \dot{h}_j^n \phi_{ij} - u_j^n \dot{h}_i^n \phi_{ij}^n \right) - B_i^n \right]$$

$$\frac{d}{dt} x_i(t) = z(t, x_i), \quad \text{and} \quad V_i(t) = \int_{\Omega(t)} \Psi_i(t, x) dx. \quad (26)$$

All the numerical simulations are performed by using the partition of unity formed by the hat function with the particle independent smoothing length. Even we have used constant smoothing for almost all simulations unless we state explcitly that we have used adaptive smoothing length, which is also depending only on time (not on particles). We have already discussed this partition of unity in the first section. In this case, we compute β_{ij}, V_i, ϕ_{ij} and x_i exactly. Therefore, it is not necessary to discretize these equations here. However, one can do similar discretization for them also. All numerical results presented in this section are obtained by using 200 regularly spaced particles in the domain $\Omega(t)$ and the result at time $t = 1.186$ is plotted in the figures. We have used the time-step $\delta t = h/2$ where h is the smoothing length. The smoothing length in the case of regularly distributed particles is taken to be the distance between two neighboring particles whereas in the case of irregularly distributed particles we choose $dm/2 < h < dm$ where dm is the maximal distance between two neighboring particles. In the Figure 14, we have plotted the difference in the solution between the characteristic method and the boundary particle method. Although we do not see the difference between these methods if we plot the solution as in the Figure 13, their difference is quite high in the neighborhood of the shock which we can see in the Figure 14. We will see which method is better when we present the plot of the errors in various methods in later subsection.

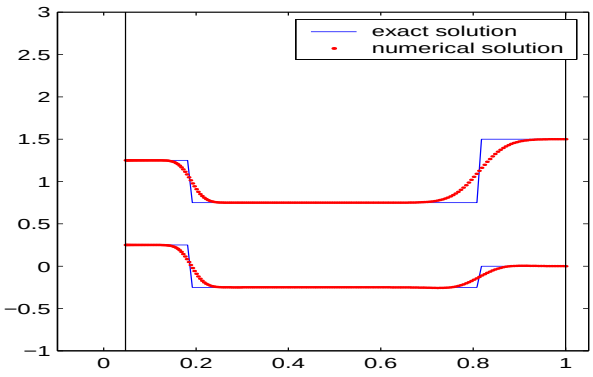

Figure 13: Exact and numerical solution from the method of boundary particles

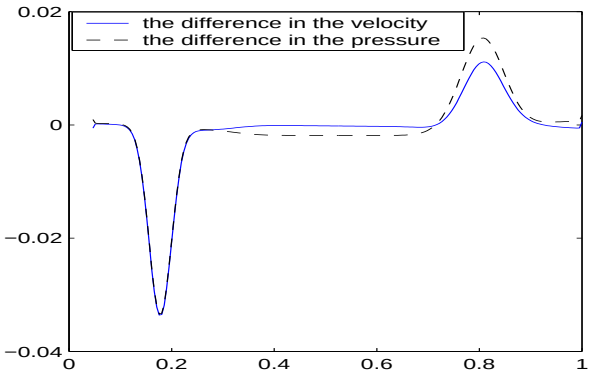

Figure 14: Difference in the solution between characteristic method and boundary particles method

We have presented the numerical result from the finite volume scheme we have derived in section 3 in the Figure 15. This is just the particular case of the FVPM when we change the smoothing length adaptively and use the partition of unity formed by regularly spaced indicator functions. The numerical simulation is performed using 200 spatial points and we plot the solution at time $t = 1.186$. All of the results we have presented here seem to be quite comparable with finite volume method. Although we do not see any difference in these methods we will see some difference when we analyze the errors between exact solution and the numerical solution in different cases.

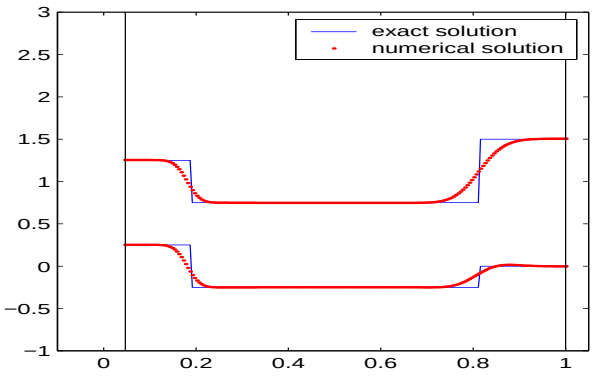

Figure 15: The exact solution and numerical solution from finite volume method

6.3 Comparison of Errors in Different Methods

Now we want to present some plot of the errors in various methods we have implemented so far. In fact, we plot $E = |u_{exact} - u_{numerical}|$ against the points where we compute them. For all numerical results we get the solution using 200 degrees of freedom. In the Figure 16, we have compared the errors between the finite volume method and finite difference method. The error plot shows that the finite volume scheme gives better resolution than the finite difference scheme. Therefore, from now on we will compare the FVPM only with finite volume methods.

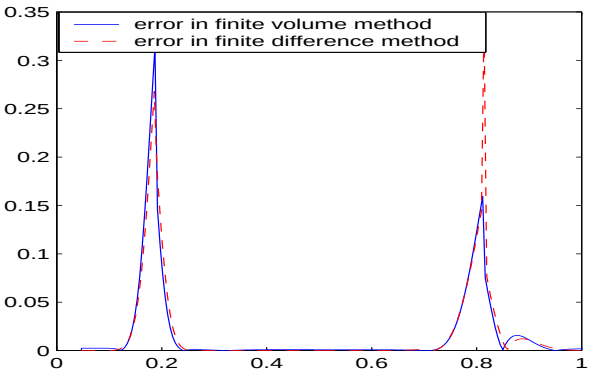

Figure 16: Error plots from finite volume method and finite difference method

We can see in the Figure 17 that the errors are large where there is jump in the solution and the error is almost zero in the constant part for different schemes.

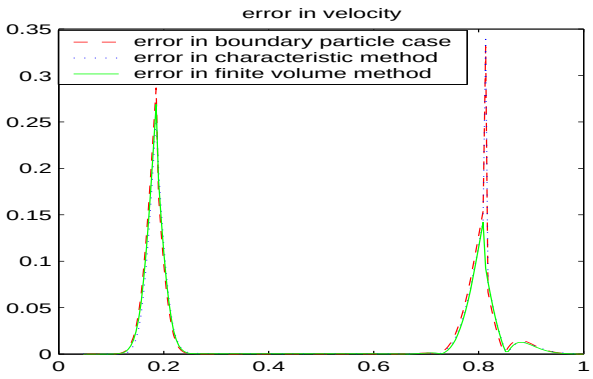

Figure 17: Error plots of different schemes

This error plot shows that the finite volume method gives the best shock resolution whereas the characteristic method and boundary particle cases are also quite comparable to finite volume method.

6.4 Comparison between Regular Particles and Irregular Particles

We do not get much difference in the case when the particles are regularly distributed between the finite volume methods and FVPM. The power of FVPM is that we can easily use irregularly spaced particles. Therefore, we want to compare the numerical results between the irregularly distributed particles and regularly distributed particles. The irregularly distributed particles are generated by the function $y = x^{5/4}$ initially, and thus the particles are dense near the moving boundary and sparse near the fixed boundary. The particle distribution in the irregular case for 20 particles is visualized in Figure 18.

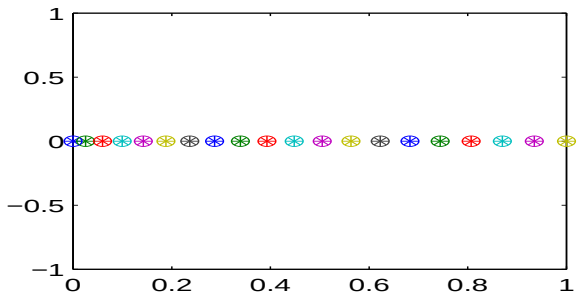

Figure 18: Particle distribution in irregular case

The irregularly distributed particles give a little bit better shock resolution near the moving boundary (i.e. where the particles are dense) and not better shock resolution near the fixed boundary (i.e. where the particles are sparse) which is quite interesting. This shows that we can probably increase the accuracy of the result locally by increasing the number of particles in this region. This can be seen from the Figure 19.

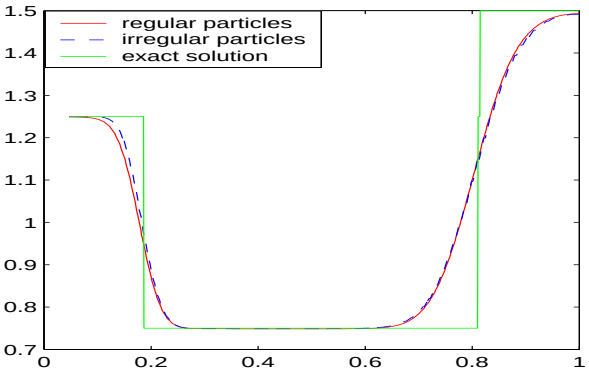

Figure 19: Comparison between regular and irregular particles

6.5 Constant Smoothing Length and Variable Smoothing Length

The choice of particle velocity is flexible in FVPM. If we take the particles to be spatially fixed, we get $\dot{x}_i = 0$ for $i = 1, \ldots, m$ and for all $t \geq 0$. Then even $\dot{V}_i = 0$ for each particle if the smoothing length is also taken to be

constant. However, we are interested in moving particles, that is, in the case when the particle-positions are changing with respect to time. When we move the particles either with the velocity-field of the fluid or any other way and if we adhere to the fixed smoothing length, the difficulty may arise if there are more particles concentrated on some parts of the domain or there are few particles in other parts. If there are few particles in some part, a gap may arise in the domain if the smoothing length is not big enough and this will cause the partition of unity to break down. The partition of unity will break down if there is some point x in the domain such that $x \notin \text{supp}(\Psi_i)$ for all $i \in M$. On the other hand if there are many particles concentrated on some part of the domain there will also be a many interactions of the particles in this part if the smoothing length is not small enough and this will increase the cost of computation (two particles will interact with each other if their supports intersect). This problem leads to the idea of changing the smoothing length in an adaptive way such that we can avoid the gap in the domain and the particle-interactions can be kept under a certain level. The idea of changing the smoothing length in an adaptive way can also increase the accuracy of the numerical solution which can be seen in Figure 20. In this Figure we have presented the comparison of errors between adaptive smoothing length and constant smoothing length. In adaptive case, we change the smoothing length in such a way that there is interaction of only two particles.

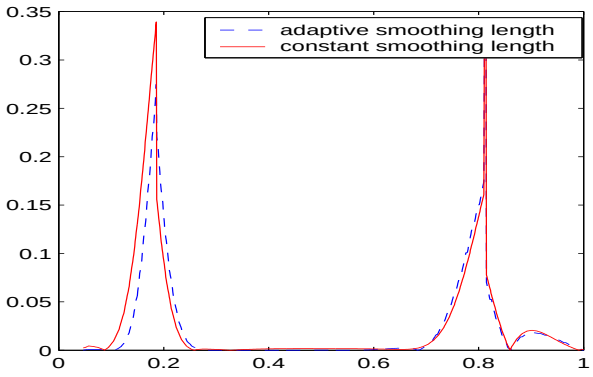

Figure 20: Comparison of errors between adaptive smoothing length and constant smoothing length

6.6 Longer Smoothing Length and Shorter Smoothing Length

Now we compare the results from shorter and longer smoothing length. Using shorter smoothing length will reduce the time-step, since the stability condition is related to the smoothing length here, and using the longer smoothing length will decrease the accuracy of the solution. That is why it is advisable to choose the smoothing length in a wise way. Now we compare the results from the shorter and longer smoothing length. To show the change in the results due the change in the smoothing length, we have used here 50 particles whose positions are generated by the function $y = x^{5/4}$ initially and the solution at time $t = 1.186$ is plotted in the Figure 21. Instead of giving the smoothing length to determine the time-step we use the same time step for both cases. That is, we use $\delta t = \frac{0.8dx}{2}$. Here dx means the maximum of distances between the two neighboring particles. We see the little steps in Figure 21 which is due to the reconstruction of the solution from the formula

$$\tilde{u}(t,x) = \sum_{i=0}^{m+1} \Psi_i(t,x) u_i(t) \tag{27}$$

and the structure of the particle $\Psi_i(t,x)$ which can be seen in the Figure 3.

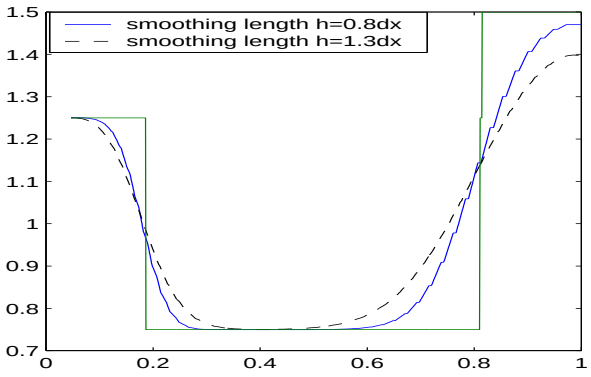

Figure 21: Comparison between longer and shorter smoothing length

7 Numerical Convergence Analysis

In this section we want to analyze the order of our numerical scheme by taking different initial and boundary values and different norms. We compute the

numerical solution for a sequence of particle distributions and expect that the numerical solution will approximate the exact solution in a better and better way, ultimately tending to the exact solution when the number of particles reaches to infinity. We want to see the difference among various cases, regular and irregular particles, smooth and non-smooth data and also want to compare the cases where both initial and boundary values are prescribed. We start with the convergence analysis for some simple initial value problems.

Generally, we want to analyse the numerical order of convergence in the L^p-norm, which is defined for general measurable functions $v(x)$ by

$$\|v\|_p = \left(\int_{\Omega(t)} |v(x)|^p dx \right)^{1/p}$$

so that the norm of the spatial error at the fixed time t is

$$\|E(.,t)\|_p = \left(\int_{\Omega(t)} |E(t,x)|^p dx \right)^{1/p}.$$

Since all the numerical methods for solving hyperbolic problems have the intrinsic numerical diffusion, jumps in the solution are smeared and the point-wise error in the neighborhood of the discontinuity does not go to zero as the grid is more and more refined. That is why we are not interested in point-wise error and thus L^p norm is appropriate for showing convergence for the conservation laws. In fact, we will see later that the solution of our model problem does not converge in L^∞-norm because of the discontinuity in the boundary value we have used on the left boundary. Suppose the exact solution is denoted by $u(t,x)$. We get the numerical solution by using the reconstruction from the interpolation formula,

$$\tilde{u}(t,x) = \sum_{i=1}^{m} \Psi_i(t,x) u_i(t)$$

and compute the point-wise error $E(t,x) = u(t,x) - \tilde{u}(t,x)$ and then we integrate the function $|E(t,x)|^p$ numerically by taking 1000 points and compute the L^p-norm by taking the pth root of the integral. Indeed, it is better to take at least $5n$ points to compute the numerical integral for the solution computed by n particles.

7.1 Convergence Analysis for an Initial Value Problem

In the beginning, we take a very simple initial value problem which is posed in the full space. Here we solve the linearized isentropic Euler equation given by

equation (21) posed in full space with smooth and non-smooth initial values where we do not need the boundary conditions. The positions of the particles are kept fixed. When simulating the full space problem in computer, we need to introduce artificial boundary. Therefore, we plot the solution before the artificial boundary affects our solution. For the case of smooth initial value, we take the velocity to be initially zero and the initial pressure is described by the function $y(x) = 100w(x - 0.5)$, where,

$$w(x) = \begin{cases} (x+h)^2 & \text{for } -h \leq x < -\frac{h}{2} \\ \frac{-2x^2+h^2}{2} & \text{for } -\frac{h}{2} \leq x < \frac{h}{2} \\ (x-h)^2 & \text{for } \frac{h}{2} \leq x < h \end{cases}$$

and $h = 0.08$. (see Figure 22) Indeed, the function $y(x)$ is in $\mathbf{C}^1(\mathbf{R})$. The numerical solution together with the exact solution at time T=0.25 are visualized in the Figure 23. The numerical solution is computed by using 400 irregularly spaced points whose positions are generated by the function $y = x^{5/4}$. Indeed, unless otherwise stated, by irregular we mean the particle-positions are generated by the function $y = x^{5/4}$ and by random we mean that the positions of the particles are generated by a random number generator. The initial wave gets separated into two waves, one of them moving forwards and other moving backwards with the same velocity. The logarithmic error-plot for different norms is plotted in the Figure 24. The slope of the line giving logarithmic error-plot in the case of regular and irregular particles for different norms are tabulated in the following tables.

The slopes in the case of regular particles				
types of norms	L^1	L^2	L^3	L^∞
slopes	-0.8798	-0.8658	-0.8726	-0.9493
The slopes in the case of irregular particles				
slopes	-0.8676	-0.8496	-0.8558	-0.9133
The slopes in the case of random particles				
slopes	-0.8263	-0.7880	-0.7972	-0.8663

This table shows that the order of convergence is almost one for different norms although the order is less for the irregular and random particles than for the regular particles.

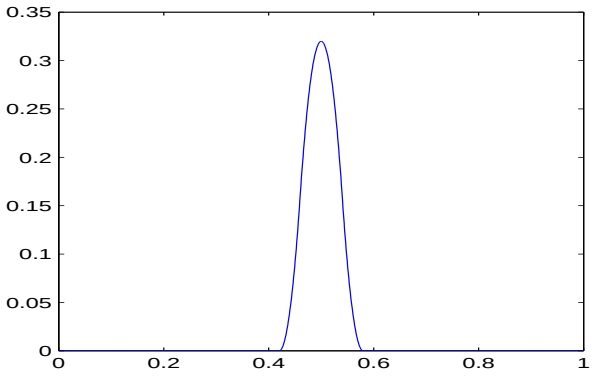

Figure 22: The initial wave

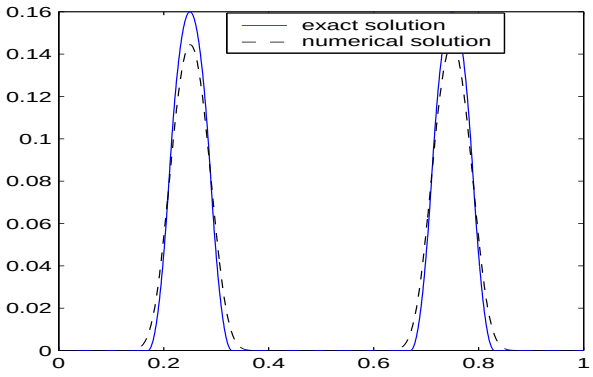

Figure 23: Exact and numerical solution for the problem with $\mathbf{C}^1$-initial value

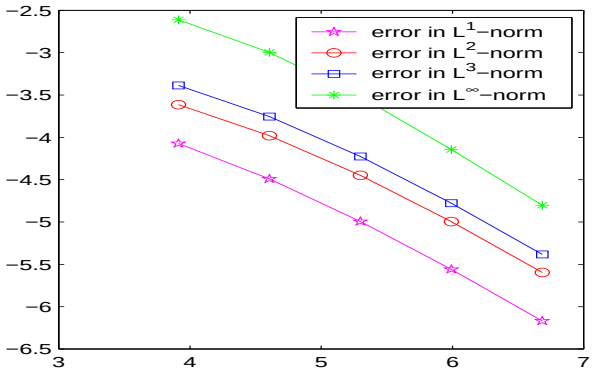

Figure 24: Logarithmic error-plot for smooth initial value

Now, we want to implement our scheme for the linearized isentropic Euler system posed in full space with discontinuous initial values. The initial profile is now a shock at the the point $x = 0.5$ and as time proceeds, the shock gets separated into two parts, moving in opposite direction. The exact solution and numerical solution using 400 irregularly spaced points are visualized in Figure 25. The distribution of points are generated as before. The logarithmic error-plot of the solution of the problem involving initial shock profile is plotted in the Figure 26. The slopes of the lines giving logarithmic error-plot for different norms and for different distributions of particles are tabulated in the following tables. This table shows that the order of convergence really depends on the regularity of the data we have prescribed.

The slopes in the case of regular particles				
types of norms	L^1	L^2	L^3	L^∞
slopes	-0.5019	-0.2534	-0.1714	-0.0333
The slopes in the case of irregular particles				
slopes	-0.5014	-0.2515	-0.1683	-0.0185
The slopes in the case of random particles				
slopes	-0.3624	-0.2110	-0.1548	-0.0134

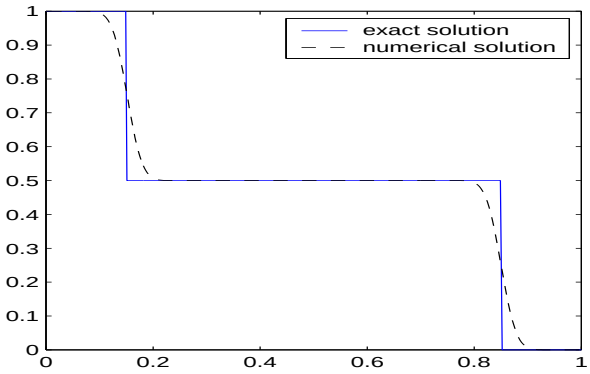

Figure 25: Exact and numerical solution involving initial shock profile

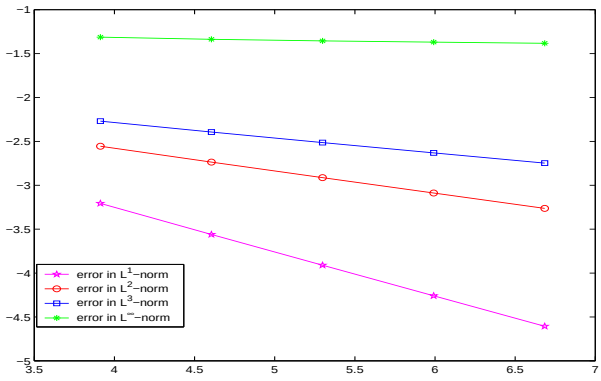

Figure 26: Logarithmic error-plot for the shock problem

7.2 Convergence Analysis in the Case of Smooth Boundary Values

To see the effect of the boundary value in the order of convergence of the scheme we implement our numerical scheme for the same linearized isentropic Euler equation given in equation 21 with smooth boundary values. In fact, we change the boundary value of our model problem only in the left boundary keeping other boundary value and initial values the same as before and get the solution in the case of non-moving boundary. That is, we do not move the boundary and impose new boundary conditions $v(t,0) = 1 - \cos(2\pi t), v(t,1) = 0$. Since the boundary is kept fixed we do not change the positions of the particles with respect to time. We use the increasing number of particles as 25,50,100,200 and 400 to compute the numerical solution. The numerical solution from FVPM which is computed by using 200 irregularly spaced particles and the exact solution at time $t = 1.186$ are visualized in the Figure 27. The distribution of particles is generated again by the function $y = x^{5/4}$. The logarithmic error plot for different norms which is visualized in the Figure 28 shows that we can achieve the convergence even in L^∞-norm even for irregular particles and the order of convergence in all types of norms is almost one. The slope of the line giving logarithmic error-plot in the case of various norms and various distribution of particles are tabulated in the following table.

The slopes in the case of regular particles				
types of norms	L^1	L^2	L^3	L^∞
slopes	-1.0094	-0.9715	-0.9506	-0.9351
The slopes in the case of irregular particles				
slopes	-0.9494	-0.9398	-0.9288	-0.8744

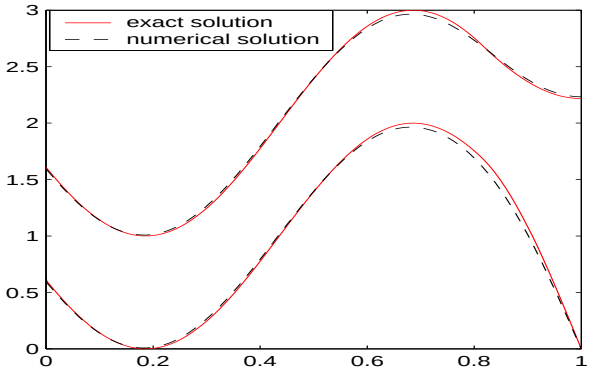

Figure 27: Exact and numerical solution of the problem with smooth initial and boundary data

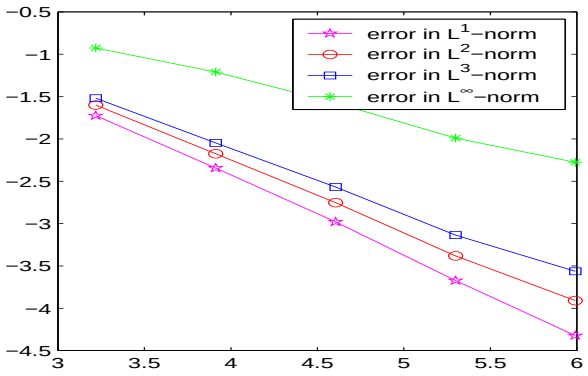

Figure 28: Logarithmic error plot for smooth boundary and initial data

7.3 Numerical Convergence Analysis for our Model Problem

In this subsection we want to give the numerical convergence analysis of FVPM applied to our model problem. As before, we use the increasing number of particles as 25,50,100,200 and 400 to compute the numerical solution and expect that the numerical solution will approximate the exact solution in a better and better way. The Figure 29 shows that the numerical solution converges to the exact solution when the number of particles reaches to infinity. In fact, in the Figure 29 we have plotted the solution computed with various numbers of irregularly spaced particles. Their initial positions are generated by the function $y = x^{5/4}$ as before. In the Figure 30 we have plotted the L^1-error in the logarithmic scale. This shows that although the numerical solution converges to the exact solution when the number of particles reaches to infinity, for the model problem, we can not achieve the convergence of the first order from this scheme. The slopes of the line giving the logarithmic error-plot in L^1-norm are -0.575 and -0.545 for regular and irregular particles respectively. However, this is not the defect of the FVPM. We can see that even the finite volume method, which gives the best numerical solution as we showed above, does not have the convergence of the first order for our model problem, which is due to the discontinuity in the boundary data. The logarithmic error-plot in L^1-norm from finite volume method is presented in Figure 31. The slope of the line in the case of finite volume method is -0.588. We have already seen that in the case of smooth boundary data we can achieve the convergence of the first order. By comparing the different cases we have presented so far, we can conclude that the order of convergence depends mainly on the regularity of the boundary and initial values while there is almost no difference between the regular and irregular particles.

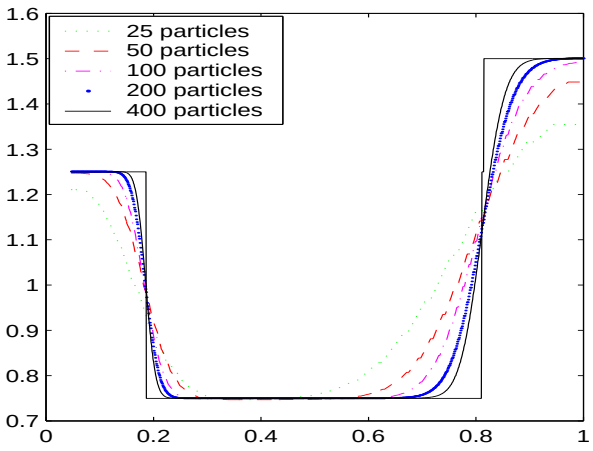

Figure 29: Numerical solution approaching the exact solutions

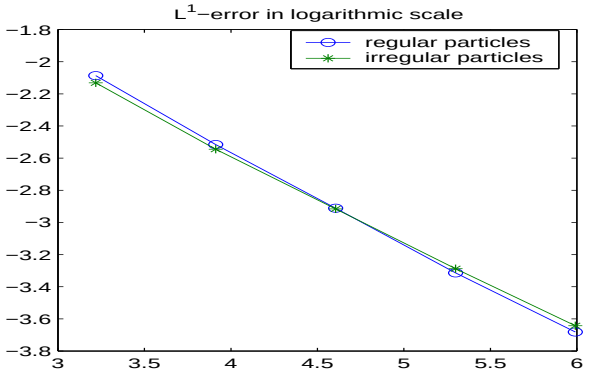

Figure 30: L^1-error in logarithmic scale for FVPM

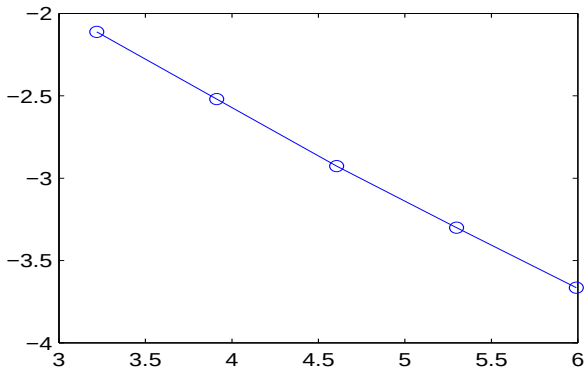

Figure 31: L^1-error in logarithmic scale for finite volume method

Finally, we want to compare the convergence of the numerical solution among different norms. As we told earlier, the L^1-norm is appropriate to analyze the convergence of any numerical scheme for conservation laws. Since even for our model problem, the boundary value which we have prescribed is discontinuous, the numerical solution does not converge to the exact solution in point-wise sense. Even the order of convergence depends upon the choice of the norms. We have presented the logarithmic error-plot for various norms in the Figures 32 and 33 for regularly spaced particles and irregularly spaced particles respectively. The slopes of the lines giving logarithmic error-plot in different norms are tabulated in the following tables. The figures in the table show that the convergence is almost of zeroth order for L^∞-norm and the order is gradually decreasing when we increase p in L^p-norm.

The slopes in the case of regular particles				
types of norms	L^1	L^2	L^3	L^∞
slopes	-0.5751	-0.3175	-0.2250	-0.0580
The slopes in the case of irregular particles				
slopes	-0.5449	-0.3011	-0.2137	-0.0510

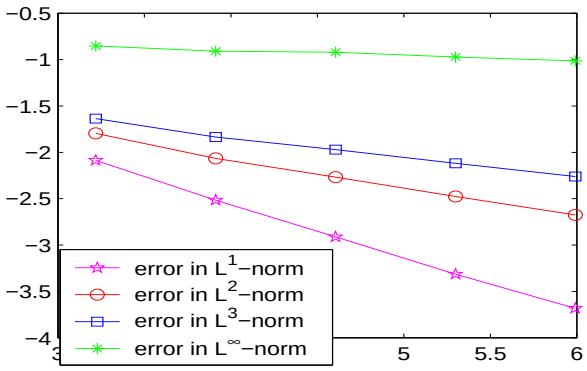

Figure 32: Logarithmic error plot in different norms for regular particles

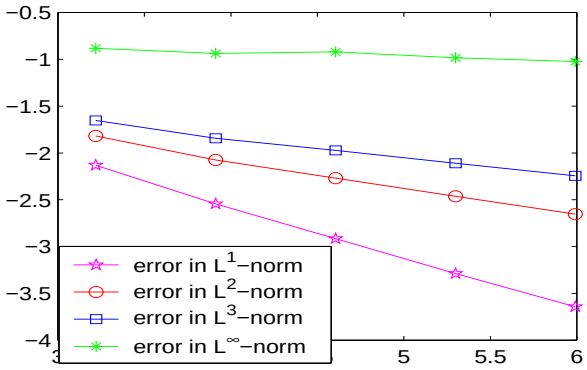

Figure 33: Logarithmic error plot in different norms for irregular particles

A Assumptions On the Flux Function g

We assume that $g(t, x_1, u_1, x_2, u_2, n)$ with $t \geq 0$, $x_1, x_2, n \in \mathbf{R}^d$ and $u_1, u_2 \in \mathbf{R}^p$ is a numerical flux function for G which satisfies
consistency

- $g(t, x, u, x, u, n) = G(t, x, u)n$

conservativity

- $g(t, x, u, y, v, -n) = g(t, y, v, x, u, n)$

continuity

- $\|g(t, x, u, y, v, n) - g(t, \bar{x}, \bar{u}, \bar{y}, \bar{v}, n)\| \leq L(\|x - \bar{x}\| + \|y - \bar{y}\| + \|u - \bar{u}\| + \|v - \bar{v}\|)$, where L depends monotonically on t and $\max\{\|u\|, \|\bar{u}\|, \|v\|, \|\bar{v}\|\}$. Also, g is assumed to be continuous in $t \in \mathbf{R}^+$.

B The Computation of β_{ij} and V_i

Here we want to present the computation of β_{ij} and V_i for the case of partition of unity induced by the family of hat functions in one dimensional domain. For this purpose we suppose that the particles are placed in the set $\bar{\Omega}(t)$ at the points $\{x_i : i \in M\}$ where $\Omega(t) = (\tilde{a}(t), \tilde{b}(t))$ and $M = \{1, \ldots, m\}$. Now let us take one particle Ψ_i, which is placed at the point x_i. Since all calculations are performed for a fixed $t \geq 0$, we drop the t-dependance for all functions. Then

$$\Psi_i(x) = \frac{w_i(x)}{\sigma(x)}$$

and

$$\sigma(x) = \sum_{i=1}^{m} w_i(x).$$

The partition of unity in this case is generated by the kernel function which is defined by

$$w(x) = \begin{cases} x + 1 & \text{if } x \in (-1, 0] \\ -x + 1 & \text{if } x \in (0, 1] \\ 0 & \text{else} \end{cases}$$

and thus

$$w_i(x) = w\left(\frac{x - x_i}{h}\right)$$

where h is the so-called smoothing length. Suppose 1_i^+ and 1_i^- be the indicator functions for the left- and right-half of the interval $(x_i - h, x_i + h]$. Now,

$$w_i(x) = \frac{x - x_i + h}{h} 1_i^-(x) + \frac{-x + x_i + h}{h} 1_i^+(x)$$

and hence

$$w_i(x) = x\frac{(1_i^-(x) - 1_i^+(x))}{h} + \left(\frac{x_i(1_i^+(x) - 1_i^-(x))}{h} + (1_i^+(x) + 1_i^-(x))\right)$$
$$= x\tilde{A}_i(x) + \tilde{B}_i(x)$$

where

$$\tilde{A}_i = \frac{(1_i^- - 1_i^+)}{h} \quad \text{and} \quad \tilde{B}_i = \left(\frac{x_i(1_i^+ - 1_i^-)}{h} + (1_i^+ + 1_i^-)\right)$$

Now defining,

$$\sum_{i=1}^{m} \tilde{A}_i = \tilde{A} \quad \text{and} \quad \sum_{i=1}^{m} \tilde{B}_i = \tilde{B}$$

we can write

$$\sigma(x) = x\tilde{A}(x) + \tilde{B}(x)$$

Thus

$$\Psi_i(x) = \frac{x\tilde{A}_i(x) + \tilde{B}_i(x)}{x\tilde{A}(x) + \tilde{B}(x)}$$

Here, $\tilde{A}_i, \tilde{B}_i, \tilde{A}$ and $\tilde{B}$ are piece-wise constant functions defined on the domain $\Omega(t)$. The idea of getting the exact integral of function $\Psi_i(x)$ over the domain $\Omega(t)$ is to partition the domain $\Omega(t)$ into small sub-intervals such that the function is a rational polynomial in each sub-interval and the functions $\tilde{A}_i, \tilde{B}_i, \tilde{A}$ and $\tilde{B}$ are constant in these sub-intervals. In fact, if we take the collection of points $\{x_i, x_i + h, x_i - h : i \in M\}$ and sort them out in ascending order, the particle function Ψ_i is rational polynomial in each piece of interval and $\tilde{A}_i, \tilde{B}_i, \tilde{A}$ and $\tilde{B}$ are constants in each piece. Hence, we integrate the particle function Ψ_i in every piece and sum the results to get the integral of Ψ_i in

$\Omega(t)$. For this purpose let us take a typical interval (a, b) in which the functions $\tilde{A}_i, \tilde{B}_i, \tilde{A}$ and $\tilde{B}$ are constant and get the integral of Ψ_i in this interval. Suppose that A_i, B_i, A and B be the values of the functions $\tilde{A}_i, \tilde{B}_i, \tilde{A}$ and $\tilde{B}$ in (a, b) respectively. Indeed

$$\int_a^b \Psi_i(x)dx = \int_a^b \frac{xA_i + B_i}{xA + B}dx.$$

By following the elementary calculations, if $A \neq 0$ in this piece, we get

$$\int_a^b \Psi_i(x)dx = \frac{(b-a)A_i}{A} + \frac{AB_i - BA_i}{A^2}\log\left|\frac{Ab+B}{Aa+B}\right|.$$

The case $A = 0$, can be treated easily, which yields

$$\int_a^b \Psi_i(x)dx = (b-a)\frac{(b+a)A_i + 2B_i}{2B}.$$

Since $\beta_{ij} = \gamma_{ij} - \gamma_{ji}$, we can compute γ_{ij} in similar way as we compute V_i. The γ_{ij} can be written as

$$\gamma_{ij} = \int_{\Omega(t)} \frac{w_i}{\sigma^2}\frac{\partial w_j}{\partial x}dx$$

and

$$\frac{\partial w_j}{\partial x} = \frac{1}{h}(1_j^-(x) - 1_j^+(x))$$

and hence

$$\gamma_{ij} = \int_{\Omega(t)} \frac{(1_j^-(x) - 1_j^+(x))}{h}\frac{(xA_i + B_i)}{(xA+B)^2}dx.$$

In a similar fashion as we did for computing V_i we can partition the domain $\Omega(t)$ into similar pieces also here and get γ_{ij} by summing the integral of each piece. Finally β_{ij} is be obtained by using the relation, $\beta_{ij} = \gamma_{ij} - \gamma_{ji}$. Indeed from the elementary calculation, if $A \neq 0$ in the piece (a, b), we get

$$\int_a^b \frac{w_i(x)}{\sigma(x)^2}\frac{\partial w_j(x)}{\partial x}dx = \frac{A_iA_j}{A^2}\log\left|\frac{Ab+B}{Aa+B}\right| + \frac{A_j(b-a)(AB_i - BA_i)}{A(Aa+B)(Ab+B)}$$

and if $A = 0$ in the piece (a, b), we have simply

$$\int_a^b \frac{w_i(x)}{\sigma(x)^2}\frac{\partial w_j(x)}{\partial x}dx = A_j(b-a)\frac{(b+a)A_i + 2B_i}{2B^2}.$$

Conclusions

In this thesis, we have studied the finite volume particle method (FVPM) presented in [3], [6] which combines the generic features of finite volume methods and particle methods. In fact, the scheme is the generalization of classical finite volume method and from the choice of the particular partition of unity the classical finite volume method can be recovered.

Here, we have derived the FVPM for time-dependent and bounded domains based on the Lagrangian particles following the similar techniques as in [6] in quite general settings. We have tested the scheme we have derived for the isentropic Euler system in one dimension in the case of moving boundary and fixed boundary for different initial and boundary data. We have shown that the scheme reasonably approximates the exact solution even for arbitrary particle distribution and even for discontinuous data, however the numerical convergence analysis shows that the order of the convergence depends on the regularity of the boundary and initial data we prescribe. We have also derived the finite volume method for moving boundary in one dimension by choosing the particular partition of unity. The next step would be to extend the scheme to higher orders and to higher dimensions. In these cases, an exact computation of the geometric coefficients and volumes is no longer possible. In fact, the efficient and accurate computation of the geometric coefficients and volumes for the general family of partition of unity in higher dimension is quite challenging task in this method.

References

[1] B. Ben Moussa, J.P. Vila, *Convergence of SPH Method for Scalar Nonlinear Conservation Laws*, SIAM J. Numer. Anal., **37**, (2000) 863-887

[2] M. Griebel, M.A. Schweitzer, *A Particle-Partition of Unity Method for the Solution of Elliptic, Parabolic and Hyperbolic PDEs*, SIAM Journal on Scientific Computing, to appear

[3] D. Hietel, K. Steiner, J. Struckmeier, *A Finite-Volume Particle Method for Compressible Flows*, Math. Models Methods Appl. Sci

[4] D. Kröner, *Numerical Schemes for Conservation Laws*, Wiley & Teubner, Chichester/Stuttgart,1997

[5] J.J. Monaghan, *Smoothed Particle Hydrodynamics*, Annu. Rev. Astron. Astrophys., **30** (1992) 543–574

[6] M. Junk, J. Struckmeier, *Consistency analysis of mesh-free methods for conservation laws*, Preprint,Universität Hamburg, 2000

[7] W. Rudin, *Real and Complex Analysis*, McGraw-Hill, 1987

[8] R. Leveque, *Numerical Methods for Conservation Laws*, Birkhähuser, Basel, 1990

[9] D. Teleaga, *Numerical studies of a finite-volume particle method for conservartion laws*, Master Thesis, University of Kaiserslautern, 2000

[10] J.P. Vila, *On particle weighted methods and smooth particle hydrodynamics*, Math. Models Methods Appl. Sci., **9** (1999) 161-209

I hereby declare that I am the only author of this work and I have used the literature listed above exclusively.
Bishnu Prasad Lamichhane
Kaiserslautern, July 25, 2001