

# Тестирование вычислительных возможностей метода MAC для моделирования динамики жидкости и газа в компьютерной графике

Спижевой А.С., Турлапов В.Е.

Нижегородский Государственный Университет им. Н.И. Лобачевского

[alexey.spizhevoj@gmail.com](mailto:alexey.spizhevoj@gmail.com), [vadim.turlapov@gmail.com](mailto:vadim.turlapov@gmail.com)

## Abstract

Paper describes new interactive implementation of known MAC-method algorithms. Performances of this implementation are investigated on the popular tasks: fluid-solid coupling; gas-solid coupling; and melting under heat propagation. Implicit surface function for quality visualization is built. Real time of these simulations on one-core CPU is achieved for 32x32x64 cells and 30000 markers.

Key words: computer graphics, fluid modeling, Marker-And-Cell, CPU, real time

## Математическая модель для жидкости

Цель данной работы: реализация на центральном процессоре метода Marker-And-Cell (MAC) и тестирование его вычислительных возможностей на популярных в компьютерной графике задачах моделирования жидкости и газа со свободными границами и их взаимодействия с твердыми телами. В литературе имеются алгоритмы и примеры решения таких задач [1-4], однако, как правило, не в реальном времени. Наша цель – интерактивные модели. При реализации мы следовали алгоритмам работ [1-3].

Основой моделирования динамики несжимаемой жидкости является уравнения Навье-Стокса:

$$\mathbf{u}_t = -(\mathbf{u} \cdot \nabla)\mathbf{u} + \nu \nabla^2 \mathbf{u} - \frac{1}{\rho} \nabla p + \mathbf{g} \quad (1.1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (1.2)$$

где  $\mathbf{u} = (u, v, w)$  – вектор скорости в точке.

**Граничные условия:**  $\mathbf{u} \cdot \mathbf{n} = 0, (x, y, z) \in \partial ModelingArea$

В качестве области моделирования принят прямоугольный параллелепипед, со сторонами кратными размеру ячейки  $h$ . Следуя работам [1-2], используем регулярную MAC-сетку.

Для соблюдения граничных условий, как и в работе [1], вводится дополнительный слой ячеек (пристеночный слой). На тех гранях ячеек сетки, которые граничат со стенами, устанавливаются скорости равными нулю. На остальных гранях ячеек сетки значения устанавливаются таким образом, чтобы обеспечивалось свободное скольжение вдоль стен. Для этого необходимо скопировать в грани ячеек приграничного слоя, расположенные перпендикулярно стенам, значения из граней ближайших ячеек.

## Процесс интегрирования

1. Поле скоростей задано из начальных условий или из предыдущего шага. Шаг интегрирования выбирается из условия Куранта-Фридрихса-Леви  $\Delta t < \frac{h}{\max(|u|, |v|, |w|)}$ , и сходимости численных методов  $\Delta t < \frac{h^2}{6\nu}$ .

2. Осуществляем обновление координат маркеров методом Рунге-Кутты второго порядка (для дыма подходит и метод Эйлера). При интерполяции скорости в точках близких к поверхности жидкости, возникает необходимость использовать

скорости в ячейках, которые не были вычислены при интегрировании на предыдущем шаге. В таком случае в ячейке вычисляется среднее значение необходимой компоненты скорости по соседним, содержащим маркеры ячейкам.

3. Во вновь заполненных ячейках скорости в центре каждой грани вычисляются на основе скоростей маркеров содержащихся в них. Для этого берется среднее арифметическое соответствующих компонент скоростей всех частиц, попавших в кубик со стороной  $h$ , центр которого находится в той точке, скорость в которой определяется. Если для не найдется частиц попавших в кубик, то берется ближайшая частица.

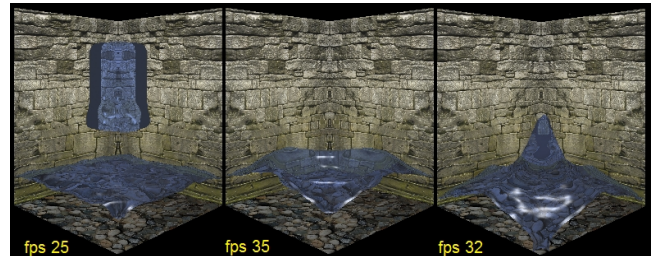
4. Решение уравнений (1.1) производится отдельно для каждого члена. Скорости обновляются на гранях ячеек, имеющих жидкость по обе стороны. Вычисляется  $\mathbf{u}^1 = \mathbf{u} + \Delta t(-(\mathbf{u} \cdot \nabla)\mathbf{u} - \nu \nabla^2 \mathbf{u} + \mathbf{g})$  с использованием semi-Lagrangian техники для конвекции [3] и метода Эйлера для остальных членов.

5. Обеспечиваем выполнение граничных условий у стен.

6.  $\mathbf{u}^1$  не удовлетворяет уравнению несжимаемости (1.2). Для компенсации этого вычисляется  $\mathbf{u}^2 = \mathbf{u}^1 - \frac{\Delta t}{\rho} \nabla p$ , где  $p$  находится из условия (1.2). На данном шаге, при нахождении давления, возникает уравнение Пуассона, которое в дискретной форме представляет собой систему линейных уравнений с разреженной, симметричной и положительно определенной матрицей. Решение может быть найдено с помощью метода сопряженных градиентов [1].  $\mathbf{u}^2$  – есть новое корректное поле скоростей.

7. Обрабатываем поверхность жидкости (границу жидкость-воздух). В таких точках мы не можем воспользоваться уравнением (1.1), поэтому скорости в ячейках на поверхности устанавливаются непосредственно из условия (1.2) в разностной форме. Его решение не единственно, когда рассматриваемая ячейка (содержащая хотя бы один маркер, но граничащая с воздухом) имеет более чем одну грань, граничащую с воздухом. В таком случае используются эвристики [1].

8. Повторяем все сначала.



## Моделирование дыма

Дополнительно вводятся два скалярных поля: плотность дыма и его температура, динамика которых в простейшем случае (только конвекция, без диффузии) описывается следующими уравнениями:

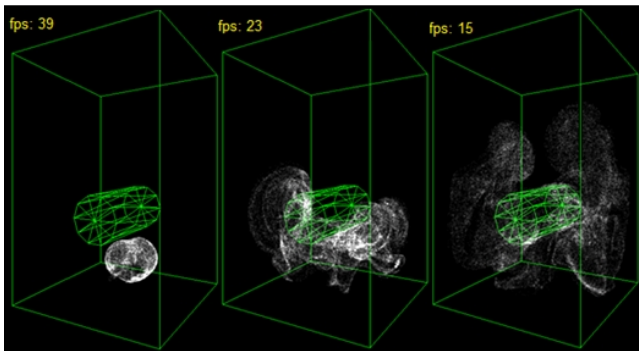
$$d_t = -(\mathbf{u} \cdot \nabla)d, T_t = -(\mathbf{u} \cdot \nabla)T$$

Для учета влияния плотности и температуры на динамику дыма используется следующая модель:

$$\mathbf{f}_{smoke} = (0, -\alpha d + \beta(T - T_{ambient}), 0)$$

Как и давление, данные величины хранятся в центрах ячеек. Для их обновления полей на каждом шаге используется semi-Lagrangian техники. Как известно semi-Lagrangian техника может приводить к возникновению диссипативных эффектов [2]. В частности при моделировании дыма было отмечено, что суммарная плотность и температура со временем уменьшаются, чего не должно быть согласно математической модели. Предполагая, что это изменение невелико, мы просто умножаем значение плотности и температуры в ячейках на отношение заданного суммарного значения к текущему суммарному значению. Артефакты в результате использования предложенного алгоритма замечены не были. Для компенсации диссипации скорости вводится дополнительная сила [2], не дающая возникающим вихрям быстро затухать:

$$\mathbf{f}_{vorticity} = \gamma h(\text{normalize}(\nabla(\boldsymbol{\omega})) \times \boldsymbol{\omega}), \text{ где } \boldsymbol{\omega} = \nabla \times \mathbf{u}.$$



## Твердые тела

Мы рассматриваем одностороннее взаимодействие: когда твердое тело влияет на жидкость или газ, а они на твердое тело не влияют.

Множество ячеек, которые пересекаются границей твердого тела, обрабатываются также как и другие, но перед передвижением маркеров в них производится коррекция скоростей. Она производится таким образом, чтобы нормальная составляющая относительной скорости в центре граничных ячеек была направлена от границы твердого тела:

1. Вычисляется вектор скорости  $\mathbf{u}_c$  в центре ячейки.
2.  $\mathbf{u}_c^{new} = \mathbf{u}_c + \max(0, (\mathbf{u}_0 - \mathbf{u}_c) \cdot \mathbf{n}_c) \mathbf{n}_c$ , где  $\mathbf{u}_0$  – скорость тела,  $\mathbf{n}_c$  – усредненная внешняя нормаль.
3. Во все шесть граней данной ячейки записывается вычисленное значение  $\mathbf{u}_c^{new}$ .

Для эффективного поиска ячеек пересекаемых телом, мы используем следующий алгоритм: перед началом вычислений мы набрасываем на каждый полигон (используются треугольники) тела случайным образом точки. Их число в одной ячейке ограничиваем сверху некоторой константой, далее эти точки считаются “приклеенными” к твердому телу и движутся вместе с ним.

Пусть нам дан треугольник  $(\mathbf{a}, \mathbf{b}, \mathbf{c})$ . Точка, лежащая на нем, может быть представлена в виде:

$$\mathbf{p}_{triangle} = \alpha_1 \cdot (\mathbf{b} - \mathbf{a}) + \alpha_2 \cdot (\mathbf{c} - \mathbf{a}),$$

$$\alpha_1 \geq 0, \alpha_2 \geq 0, \alpha_1 + \alpha_2 \leq 1.$$

Таким образом, остается сгенерировать необходимое число пар  $(\alpha_1, \alpha_2)$  удовлетворяющих вышеприведенному условию.

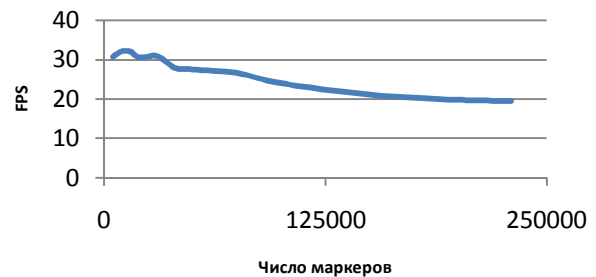
Пусть тело пересекает только те ячейки, в которых есть его представители – набросанные точки. На регулярной сетке индекс ячейки, в которой лежит заданная точка, может быть вычислен следующим образом:  $(i, j, k) = \left(\left\lfloor \frac{x}{h} \right\rfloor, \left\lfloor \frac{y}{h} \right\rfloor, \left\lfloor \frac{z}{h} \right\rfloor\right)$ , где  $i, j, k$  – индексы ячейки,  $x, y, z$  – координаты точки,  $h$  – размер ячейки.

Нормаль  $\mathbf{n}_c$  вычисляется, как средняя по нормальям исходящим из точек-представителей (в них нормаль определяется однозначно по треугольникам, которым они принадлежат).

Весь нижний уровень вычислений векторизован.

## Результаты<sup>1</sup>

Достигнута интерактивность реализации MAC-метода для 32x32x64 ячеек и 30000 маркеров, т.е. на уровне и выше [4]. Для визуализации жидкости использовался ray tracing и photon mapping на GPU (предоставленные на Graphicon2009 Денисом Боголеповым). Дым визуализировался сглаженными точками большого размера с использованием OpenGL.



## Литература

- [1] Carlson, M. Rigid, Melting, and Flowing Fluid / PhD thesis, Georgia Institute of Technology. 2004.
- [2] Fedkiw, R., Stam, J., and Jensen, H. W. Visual simulation of smoke. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, ACM Press, p.15–22. 2001.
- [3] David, C., David C. and Parris, K. E. Fluid Flow for the Rest of Us: Tutorial of the Marker and Cell Method in Computer Graphics / Brigham Young University.
- [4] Batty, Ch. , Bertails, F., Bridson, R. A Fast Variational Framework for Accurate Solid-Fluid Coupling / University of British Columbia, p.1-7, In Proceeding of SIGGRAPH 2007.

<sup>1</sup> Intel® Core™ 2 Duo CPU P8400, RAM 3.0GB, NVIDIA GeForce 9600M GS, Windows Vista™ Home Basic