Министерство образования и науки Российской Федерации

КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

ИНСТИТУТ МАТЕМАТИКИ И МЕХАНИКИ ИМ. Н. И. ЛОБАЧЕВСКОГО

КАФЕДРА АЭРОГИДРОМЕХАНИКИ

Направление: 010800.68 — Механика и математическое моделирование

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

(Дипломная работа)

ВЛИЯНИЕ СИЛЫ БАССЭ НА НАПРАВЛЕНИЕ ДРЕЙФА ВКЛЮЧЕНИЯ В СТОЯЧЕЙ ВОЛНЕ

Работа завершена:
"___"_____2014 г**.** _____ (А.Н. Закиров)

Работа допущена к защите:
Научный руководитель
док. физ.-мат. наук,
профессор кафедры аэрогидромеханики
"___"_____2014 г. _____ (Д.А. Губайдуллин)

Научный консультант
док. физ.-мат. наук,
главный научный сотрудник
Института Механики и Машиностроения
Каз. НЦ РАН
"___"_____2014 г. _____ (П.П. Осипов)

Заведующий кафедрой
док. физ.-мат. наук, профессор
"___"_____2014 г. _____ (А.Г. Егоров)

Казань — 2014

Оглавление

Введение..... 5

Глава 1. Постановка задачи..... 10

Глава 2. Вычисление силы Бассэ..... 14

Глава 3. Влияние силы Бассэ на направление дрейфа включения 17

Заключение 21

Список литературы 22

Обозначения

L - длина резонатора;

x_2 - координата центра сферической частицы;

$v_1(x,t),\ v_2(t)$ - скорости среды (газа) и частицы;

r - радиус частицы;

$\rho_1 = \rho_1(x,t),\ \rho_1^0$ - плотность газа и её начальное значение;

$\rho_2 = \rho_2^0$ - плотность частицы;

$\tilde{\rho} = \dfrac{\rho_1}{\rho_2}$ - относительная плотность газа;

$p_1 = p_1(x,t),\ p_1^0$ - давление газа и его начальное значение;

$\mu, \nu = \dfrac{\mu}{\rho_1}$ - коэффициенты кинематической и динамической вязкости;

$\tau_\mu = \dfrac{r^2}{\nu}$ - время формирования стоксового режима обтекания включения;

$\tau_\nu = \dfrac{2\tau_\mu}{9\tilde{\rho}}$ - время релаксации частицы;

$\tau = (1 + 0.5\tilde{\rho})\tau_\nu$ - время релаксации частицы с учетом присоединенных масс;

ω - угловая частота;

$t, T = \dfrac{2\pi}{\omega}$ - время и период осцилляций;

X_0 - амплитуда осцилляций на левой границе;

$U_0 = \omega X_0$ - амплитуда скорости осцилляций на левой границе;

$\mathrm{Re}_0 = \dfrac{2rU_0}{\nu}$ - число Рейнольдса на левой границе;

$\mathrm{Re} = \rho_1 |v_1 - v_2| \frac{2r}{\mu}$ - относительное число Рейнольдса;

$K = \omega\tau_v$ - безразмерный критерий инерции включения,

$D = \frac{3\tilde{\rho}}{(2+\tilde{\rho})}$ - безразмерный критерий относительной плотности;

$Sr = \frac{\omega L}{U_0} = \frac{L}{X_0}$ - число Струхаля;

$C_x = \frac{24}{\mathrm{Re}}\left(1 + \frac{1}{6}\sqrt{\mathrm{Re}} + \frac{1}{60}\mathrm{Re}\right),\ 0 \le \mathrm{Re} \le 2\cdot 10^5$ - коэффициент лобового сопротивления сферы.

$\mu_p^2 = 1/(1+(\omega\tau)^2)$ - коэффициент увлечения частицы

Введение

В акустических полях на включение действуют различные силы: Стокса, силы Архимеда и присоединенных масс, а также сила Бассэ и другие. Осцилляции несущей среды могут вызвать дрейф включения. Этот эффект широко используется в волновых технологиях очистки и коагуляции. Общие вопросы дрейфа включений в акустических полях достаточно хорошо исследованы

Подробный обзор исследований динамики одиночного включения (частицы или пузырька) в гидродинамических полях флюида приведен в работах [1, 2]. В общем случае в волновых полях на включение действует средняя по периоду сила, называемая волновой силой [2], которая приводит к среднему по периоду ускорению (виброускорению) и дрейфу включения. Исследования дрейфа включения можно условно разделить на две большие группы: дрейф в однородном и неоднородном периодических полях скорости флюида. К первой группе относятся работы [3–8]. Из этой группы отметим лишь работу [3], в которой было впервые показано, что одной из причин дрейфа включения является наличие высших гармоник в решении уравнения Ланжевена вследствие нелинейности закона сопротивления включения. Ко второй группе относятся исследования дрейфа включения в неоднородных волновых полях, т. е. в полях стоячей волны $v_1 = U(x)\sin(\omega t)$, бегущей волны $v_1 = U(x)\sin(\omega t) + kx$ и периодической ударной волны резонатора. К этой группе относится одна из пионерских работ [9], в которой были получены, ставшие классикой, формулы Кинга для радиационного давления, испытываемого несжимаемой сферой в стоячей или бегущей волне. Длина волны предполагалась много больше радиуса сферы. Была учтена сила инерции флюида, а сила вязкого сопротивления полагалась равной нулю. Из формулы Кинга следует, что включение (например, частица аэрозоля), имеющее небольшую относительную плотность флюида $\rho < 2.5$, дрейфует к пучности, а включение (например, пузырек), имеющее большую относительную плотность флюида $\rho > 2.5$ — к ближайшему узлу

волны скорости флюида. В [10] формулы Кинга обобщены на случай сжимаемой сферы. Установлено, что если размер сферы меньше резонансного, она перемещается в узлы, а если больше — то в пучность скорости флюида. Если размер сферы равен резонансному значению, то она находится в состоянии безразличного равновесия. Эти выводы были подтверждены в экспериментальной работе [11].

В работах [3, 12, 13] учтена вязкость флюида и показано, что вязкие силы в приближении Осеена в десятки раз превосходят силу радиационного давления, определяемую формулой Кинга. В [14] предложен оригинальный метод расчета силы, приводящей к дрейфу частицы. В рамках потенциального обтекания сферы идеальной жидкостью потенциал рассеянной волны разложен по мультиполям с коэффициентами, взятыми из задачи обтекания несжимаемой жидкостью. Полученные формулы подтвердили формулы Кинга.

В статье [15] исследовано движение несжимаемых аэрозольных частиц в плоской стоячей волне для случая достаточно малых размеров частиц, малых чисел Рейнольдса и относительно невысоких частот. Была предложена теория дрейфа в стоячей волне с учетом силы Стокса, частично силы присоединенной массы и силы Бассэ. Получена формула для средней силы, вызывающей дрейф. Было показано, что для аэрозоля при невысоких частотах поля средняя сила, действующая на частицу, на несколько порядков превосходит силу радиационного давления, вычисляемую по формуле Кинга. При этом частица аэрозоля дрейфует не к пучности, как это предсказывает формула Кинга, а к ближайшему узлу волны. Учтено влияние силы Бассэ на фоновое периодическое колебание частицы, хотя полный учет этой силы на дрейф произвести не удалось ввиду сложного характера влияния предыстории движения включения на его среднее ускорение.

В работе [16] исследован дрейф пузырька в стоячей акустической волне в воде и в изопропиловом спирте с учетом силы Архимеда. Установлено, что изменением амплитуды волны можно вызвать всплытие, затопление или

левитацию пузырька. В последнем случае пузырек совершает колебания в окрестности некоторого положения равновесия. Скорость дрейфа зависит от положения пузырька в стоячей волне и его радиуса.

В статье [17] изучено движение частицы аэрозоля под действием асимметричных сил в стоячей волне и получена формула для среднего ускорения частицы. Проведены сравнения с известными формулами других авторов. В [18] исследовано уравнение движения частицы аэрозоля в стоячей волне в приближении Стокса и Осеена. Оценена возможность упрощения уравнения движения с учетом дрейфа из-за радиации, периодического изменения вязкости и асимметрии движения колеблющейся частицы в стоячей волне. Получена формула для среднего ускорения без учета инерции частицы.

В работе [19] исследован дрейф малой частицы, взвешенной в акустическом поле. Показано, что дрейф возникает из-за неоднородности амплитуды несущего поля. Этот тип дрейфа, названный авторами асимметричным, исследовался для бегущей и стоячей волн, а также для сферической расходящейся волны. В [20] в предположении несжимаемости флюида и включения получена формула для волновой силы, действующей на сферическое включение с учетом динамической составляющей силы Архимеда и силы присоединенных масс в полях типа стоячей волны. Применена декомпозиция движения включения на быструю и медленную составляющие. Установлено существование характерной частоты, при переходе через которую суммарная сила меняет направление. Предложена диаграмма направления дрейфа.

В работе [21] даются последующее уточнение формулы для среднего ускорения сферического включения и диаграммы направления волновой силы с учетом сжимаемости несущей фазы, включения и силы поверхностного натяжения. Методами малого параметра и Ван-дер-Поля исследован дрейф сферического включения в волновых гидродинамических полях. Изучено влияние на дрейф основных эффектов, вызванных силами Стокса и Архимеда и

силой присоединенной массы, а также сжимаемостью несущей фазы и включения. Для стоячей волны $v_1 = U(x)\sin(\omega t)$ амплитуда скорости несущей фазы аппроксимируется в окрестности включения линейной частью ряда Тейлора $U(x_2) = U(\xi) + U'(\xi)(\mathrm{x}_2 - \xi)$, где $\xi = x - 2$ — среднее по периоду положение центра включения. Выделение фоновой однородной составляющей $U(\xi)$ позволило оценить амплитуду и сдвиг фазы колебания включения. Получено обобщение формулы Брандта для стоксового инерционного режима. Неоднородная часть $U'(\xi)(\mathrm{x}_2 - \xi)$ дает ненулевое среднее ускорение включения. Получена формула для среднего ускорения включения. Показано, что даже при одинаковых плотностях и коэффициентах жесткостей волновая сила отлична от нуля при не слишком малых капиллярных числах. На основе приведенной формулы для несжимаемого включения обнаружена пороговая частота, при переходе через которую волновая сила меняет направление. Выделены пять режимов дрейфа. Предложена диаграмма направления волновой силы.

Следует отметить работу [22], в которой исследуется дрейф включения не на основе уравнений динамики включения в заданном поле флюида, а с помощью прямого моделирования динамики самого флюида на основе уравнений Навье-Стокса с учетом динамики незакрепленной сферы, находящейся во флюиде. При этом силы, действующие на сферу, вычисляются на основе тензора напряжений. В [23, 24] изучены силы, действующие на частицу в ударной волне. В [25] исследован дрейф включения в трубе-резонаторе при наличии периодической ударной волны. Показано, что под воздействием ударной волны малые частицы дрейфуют к концам резонатора.

В работе [26] показано, что при больших числах Рейнольдса также происходит смена направления результирующей волновой силы при превышении пороговой частоты. Для определения направления волновой силы удобно использовать диаграммы пороговых значений $\mu_p^2(D)$ (квадрат порогового коэффициента увлечения $\mu_p^2 = 1/(1+(\omega\tau)^2)$ в зависимости от

критерия относительной плотности включения D, при переходе через которые направление результирующей волновой силы меняется на противоположное. Построены универсальные диаграммы «пороговых» кривых $\mu_p^2(D)$ при различных числах Рейнольдса и Струхаля. При увеличении чисел Рейнольдса и Струхаля пороговые значения $\mu_p^2(D)$ заметно уменьшаются, т.е. пороговые частот возрастают. В [6] предложено использовать диаграммы ускорения дрейфа. В [7] исследовано изменение направление дрейфа включения в периодической ударной волне в резонаторе.

Несмотря на большое количество исследований дрейфа включения в стоячей волне, вопрос о влиянии силы Бассэ на направление дрейфа включения изучен слабо и является целью настоящей работы.

Глава 1. Постановка задачи.

На одиночное сферическое включение в акустическом поле действуют силы, приведенные в таблице 1.

Таблица 1.

Название	**Сила, F**
Сила Стокса	$F_S = 6\pi r \mu (v_1 - v_2)$
Сила вязкого сопротивления	$F_V = \frac{\mathrm{Re}}{24} C_x F_S$
Сила Бассэ	$F_B = 6r^2 \sqrt{\pi \rho_1 \mu} \int_{t_0}^{t} \left(\frac{d_2 v_1}{d\theta} - \frac{d v_2}{d\theta} \right) \frac{d\theta}{\sqrt{t-\theta}}$
Сила Архимеда	$F_A = -\frac{4}{3}\pi r^3 \rho_1 g$
Динамическая сила Архимеда	$F_{AD} = \frac{4}{3}\pi r^3 \rho_1 \frac{d_1 v_1}{dt}$
Вес включения	$F_g = \frac{4}{3}\pi r^3 \rho_2 g$
Сила присоединенных масс	$F_m = \frac{2}{3}\pi r^3 \rho_1 \left(\frac{d_1 v_1}{dt} - \frac{d v_2}{dt} \right)$

где $\frac{d_2}{dt} = \frac{\partial}{\partial t} + v_2 \frac{\partial}{\partial x}$, $\frac{d_1}{dt} = \frac{\partial}{\partial t} + v_1 \frac{\partial}{\partial x}$, r - радиус включения, ρ_1, ρ_2 – плотности жидкости и включения, v_1 v_2 - скорости жидкости и включения,

$\mathrm{Re} = \rho_1 | v_1 - v_2 | \frac{2r}{\mu}$ – относительное число Рейнольдса,

$C_x = \frac{24}{\mathrm{Re}} \left(1 + \frac{1}{6}\sqrt{\mathrm{Re}} + \frac{1}{60}\mathrm{Re} \right)$, $0 \le \mathrm{Re} \le 2 \cdot 10^5$ - коэффициент лобового сопротивления сферы.

Общее уравнение сохранения импульса сферического включения имеет вид

$$\frac{4}{3}\pi r^3\rho_2\frac{dv_2}{dt}=F_s+F_B+F_A+F_{AD}+F_g+F_m,$$

или

$$\frac{4}{3}\pi r^3\rho_2\frac{dv_2}{dt}=6\pi r\mu(v_1-v_2)\frac{C_x\,\mathrm{Re}}{24}+6r^2\sqrt{\pi\rho_1\mu}\int_{t_0}^{t}\left(\frac{d_2v_1}{d\theta}-\frac{dv_2}{d\theta}\right)\frac{d\theta}{\sqrt{t-\theta}}-\frac{4}{3}\pi r^3\rho_1 g+$$

$$\frac{4}{3}\pi r^3\rho_1\frac{d_1v_1}{dt}+\frac{4}{3}\pi r^3\rho_2 g+\frac{2}{3}\pi r^3\rho_1\left(\frac{d_1v_1}{dt}-\frac{dv_2}{dt}\right)$$

Введем обозначения:

$$\tilde{\rho}=\frac{\rho_1}{\rho_2},\ D=\frac{3\tilde{\rho}}{2+\tilde{\rho}},\ \tau_v=\frac{2}{9}\frac{r^2\rho_2}{\mu}=\frac{2}{9}\frac{\tau_\mu}{\tilde{\rho}},\ \tau=(1+0.5\tilde{\rho})\tau_v=\tau_\mu/(3D)$$

Воспользовавшись введенными обозначениями получим:

$$\frac{dv_2}{dt}=\frac{v_1-v_2}{\tau}\frac{C_x\,\mathrm{Re}}{24}+\frac{\sqrt{\tau_\mu}}{\tau}\frac{1}{\sqrt{\pi}}\int_{t_0}^{t}\left(\frac{d_2v_1}{d\theta}-\frac{dv_2}{d\theta}\right)\frac{d\theta}{\sqrt{t-\theta}}+D\frac{d_1v_1}{dt}+(1-D)g$$

Полная система уравнений будет выглядеть так:

$$\begin{cases}\dfrac{dv_2}{dt}=\dfrac{v_1-v_2}{\tau}\dfrac{C_x\,\mathrm{Re}}{24}+\dfrac{\sqrt{\tau_\mu}}{\tau}\dfrac{1}{\sqrt{\pi}}\displaystyle\int_{t_0}^{t}\left(\frac{d_2v_1}{d\theta}-\frac{dv_2}{d\theta}\right)\frac{d\theta}{\sqrt{t-\theta}}+D\frac{d_1v_1}{dt}+(1-D)g\\ \dfrac{dx_2}{dt}=v_2\\ x_2(0)=x_2^0\\ v_2(0)=v_2^0\end{cases}$$

где x_2 – положение включения.

Далее рассмотрим систему уравнений динамики включения без учёта сил гравитации:

$$\begin{cases}\dfrac{dx_2}{dt}=v_2\\ \dfrac{dv_2}{dt}=-\dfrac{v_2-v_1}{\tau}\left(1+\dfrac{1}{6}\sqrt{\mathrm{Re}}+\dfrac{1}{60}\mathrm{Re}\right)+D\left(\dfrac{\partial v_1}{\partial t}+v_1\dfrac{\partial v_1}{\partial x}\right)+\\ \qquad\qquad\dfrac{\sqrt{\tau_\mu}}{\tau}\dfrac{1}{\sqrt{\pi}}\displaystyle\int_0^t\left(\frac{\partial v_1}{\partial\theta}+v_2\frac{\partial v_1}{\partial x}-\frac{dv_2}{d\theta}\right)\frac{d\theta}{\sqrt{t-\theta}}\end{cases}$$

В настоящей работе ограничимся исследованием дрейфа в плоской стоячей волне.

Вводя безразмерные величины

$$v_1^* = \frac{v_1}{U_0},\ v_2^* = \frac{v_2}{U_0},\ t^* = \omega t,\ \theta^* = \omega\theta$$

и опуская звездочку, запишем

$$\begin{cases} L\omega\dfrac{dx_2}{dt} = U_0 v_2 \\ U_0\omega\dfrac{dv_2}{dt} = -\dfrac{1}{(1+0.5\rho)\tau_\nu}\left(1+\dfrac{1}{6}\sqrt{Re}+\dfrac{1}{60}Re\right)(v_2-v_1) + D\left(U_0\omega\dfrac{\partial v_1}{\partial t}+U_0^2\dfrac{v_1}{L}\dfrac{\partial v_1}{\partial x}\right)+ \\ \qquad\qquad \dfrac{\sqrt{\tau_\mu}3D}{\tau_\mu}\dfrac{1}{\sqrt{\pi}}\displaystyle\int_0^t\left(U_0\omega\dfrac{\partial v_1}{\partial\theta}+U_0^2\dfrac{v_2}{L}\dfrac{\partial v_1}{\partial x}-U_0\omega\dfrac{dv_2}{d\theta}\right)\dfrac{d\theta}{\sqrt{t-\theta}} \end{cases}$$

Перенесем все коэффициенты в правые части уравнений, распишем $\tau_\nu = 2\tau_\mu / 9\tilde{\rho}$

$$\begin{cases} \dfrac{dx_2}{dt} = \dfrac{U_0}{L\omega}v_2 \\ \dfrac{dv_2}{dt} = -\dfrac{U_0}{U_0\omega(1+0.5\rho)}\dfrac{9\rho}{2\tau_\mu}\left(1+\dfrac{1}{6}\sqrt{Re}+\dfrac{1}{60}Re\right)(v_2-v_1) + \dfrac{D}{U_0\omega}\left(\dfrac{\partial v_1}{\partial t}+U_0^2\dfrac{v_1}{L}\dfrac{\partial v_1}{\partial x}\right)+ \\ \qquad\qquad \dfrac{3D}{U_0\omega}\sqrt{\dfrac{\nu\rho_1}{r^2\rho_1}}\dfrac{1}{\sqrt{\pi}}\displaystyle\int_0^t\left(U_0\omega\dfrac{\partial v_1}{\partial\theta}+U_0^2\dfrac{v_2}{L}\dfrac{\partial v_1}{\partial x}-U_0\omega\dfrac{dv_2}{d\theta}\right)\dfrac{d\theta}{\sqrt{\omega}\sqrt{t-\theta}} \end{cases}$$

Собрав коэффициенты, получим задачу Коши в безразмерных переменных

$$\begin{cases} \dfrac{dx_2}{dt} = \dfrac{1}{Sr}v_2 \\ \dfrac{dv_2}{dt} = -\dfrac{3D}{K}\left(1+\dfrac{1}{6}\sqrt{\mathrm{Re}}+\dfrac{1}{60}\mathrm{Re}\right)(v_2-v_1) + D\left(\dfrac{\partial v_1}{\partial t}+\dfrac{v_1}{Sr}\dfrac{\partial v_1}{\partial x}\right)+ \\ \qquad\qquad \dfrac{3D}{\sqrt{K}}\dfrac{1}{\sqrt{\pi}}\displaystyle\int_0^t\left(\dfrac{\partial v_1}{\partial\theta}+\dfrac{v_2}{Sr}\dfrac{\partial v_1}{\partial x}-\dfrac{dv_2}{d\theta}\right)\dfrac{d\theta}{\sqrt{t-\theta}} \\ x_2(0)=0.75,\ v_2(0)=0,\ v_1(x,t)=\sin(\pi x)\sin t \end{cases}$$

где $Sr = \frac{\omega L}{U_0}$ – число Струхаля.

Целью данной работы является исследование дрейфа включения в стоячей синусоидальной волне при различных числах Рейнольдса и Струхаля под действием сил вязкости, присоединенных масс, Архимеда и Бассэ.

Одной из сложностей данной задачи является вычисление интеграла типа Дюамеля, выражающего силу Бассэ. Аналитически его подсчет невозможен, численный же расчёт занимает достаточно много времени.

В связи с этим ставилось три задачи:

- Построить схему численного решения задачи Коши.
- Распараллелить вычисление средствами MPI
- Оценить влияние силы Бассе на дрейф включения

Глава 2. Вычисление силы Бассэ

Сила Бассэ выражается в виде интеграла наследственности

$$I\langle f(t)\rangle = \frac{1}{\sqrt{\pi}}\int_0^t f(\theta)\frac{d\theta}{\sqrt{t-\theta}}$$

где, $f(\theta) = \frac{\partial v_1}{\partial \theta} + \frac{v_2}{Sr}\frac{\partial v_1}{\partial x} - \frac{dv_2}{d\theta}$.

Во избежание сингулярности в подынтегральном выражении проведем преобразование:

$$I\langle f(t)\rangle = \frac{1}{\sqrt{\pi}}\int_0^t f(\theta)\frac{d\theta}{\sqrt{t-\theta}} = -\frac{2}{\sqrt{\pi}}\int_0^t f(\theta)\,\mathrm{d}\sqrt{t-\theta}$$

Зададим равномерную сетку по времени $t_k = k\cdot\Delta t$ (k = 0, 1, 2 ... n) В узлах сетки известны значения подынтегральной функции $f_k = f(t_k)$. Для того, чтобы сосчитать интеграл на следующем слое времени, т.е. при $t_n = n\cdot\Delta t$ воспользуемся квадратурной формулой трапеций

$$I_n = -\frac{2}{\sqrt{\pi}}\int_0^{t_n} f(\theta)\,\mathrm{d}\sqrt{t_n-\theta} = -\frac{2}{\sqrt{\pi}}\sum_{k=1}^{k=n}\frac{f_{k-1}+f_k}{2}\left(\sqrt{t_n-t_k}-\sqrt{t_n-t_{k-1}}\right) =$$

$$\frac{\Delta t}{\sqrt{\pi}}\sum_{k=1}^{k=n}\frac{f_{k-1}+f_k}{\sqrt{t_n-t_k}+\sqrt{t_n-t_{k-1}}} = \sqrt{\frac{\Delta t}{\pi}}\sum_{k=1}^{k=n}\frac{f_{k-1}+f_k}{\sqrt{n-k}+\sqrt{n-(\mathrm{k}-1)}}$$

Численный подсчет интеграла был реализован на языке программирования C++(Приложение 1). В качестве проверки использовались два тождества

$$\frac{1}{\sqrt{\pi}}\int_o^{\varphi}\frac{d\theta}{\sqrt{\varphi-\theta}} = \frac{2}{\sqrt{\pi}}\sqrt{\varphi},\qquad \frac{1}{\sqrt{\pi}}\int_o^{\varphi}\sqrt{\theta}\frac{d\theta}{\sqrt{\varphi-\theta}} = \frac{\sqrt{\pi}}{2}\varphi$$

Графики численных расчетов соответствующих интегралов и их точных значений приведены на рисунке 1

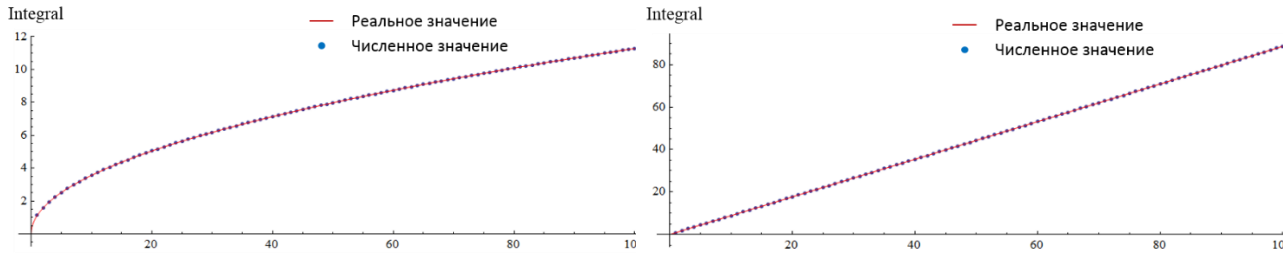

Рис. 1 Проверка интеграла наследственности.

Численное значение интегралов удовлетворительно совпадает с точными значениями.

Учет силы Бассэ резко увеличивает время численного расчета для больших интервалов времени. В связи с этим было проведено распараллеливание вычисления интеграла Дюамеля с помощью библиотеки MPI. Распараллеливание проводилось на трех четырех-ядерных компьютерах, объединенных локальной сетью. На каждом следующем слое $t_n = n\,\Delta t$ область интегрирования разбивалась по числу процессоров $\mathrm{np} = 12$.

$$-\frac{2}{\sqrt{\pi}}\int_{k_0\Delta t}^{k_1\Delta t} \mathrm{f}(\theta)\,\mathrm{d}\sqrt{\mathrm{t_n}-\theta} - \frac{2}{\sqrt{\pi}}\int_{k_1\Delta t}^{k_2\Delta t} \mathrm{f}(\theta)\,\mathrm{d}\sqrt{\mathrm{t_n}-\theta} - \ldots - \frac{2}{\sqrt{\pi}}\int_{k_{np-1}\Delta t}^{t_n} \mathrm{f}(\theta)\,\mathrm{d}\sqrt{\mathrm{t_n}-\theta}$$

где, $k_i = \left[\frac{\mathrm{n}}{\mathrm{np}}\right] i,\ (i = 0, \mathrm{np}-1),\ t_n = k_{np}\Delta t = n\Delta t$

Если число узлов не делилось без остатка на количество процессоров, то первым одиннадцати потокам доставались равные части, а остаток уходил к последнему. Каждый из интегралов вычислялся по формуле трапеций. Например:

$$-\frac{2}{\sqrt{\pi}}\int_{k_1\Delta t}^{k_2\Delta t} \mathrm{f}(\theta)\,\mathrm{d}\sqrt{\mathrm{t_n}-\theta} = -\frac{2}{\sqrt{\pi}}\sum_{\mathrm{k}=k_1}^{\mathrm{k}=k_2}\frac{\mathrm{f_{k-1}}+\mathrm{f_k}}{2}\left(\sqrt{\mathrm{t_n}-\mathrm{t_k}}-\sqrt{\mathrm{t_n}-\mathrm{t_{k-1}}}\right)$$

$$= \sqrt{\frac{\Delta\,\mathrm{t}}{\pi}}\sum_{\mathrm{k}=k_1}^{\mathrm{k}=k_2}\frac{\mathrm{f_{k-1}}+\mathrm{f_k}}{\sqrt{\mathrm{n}-\mathrm{k}}+\sqrt{\mathrm{n}-(\mathrm{k}-1)}}$$

Для расчета на следующем временном слое необходимо собрать значения с каждого процессора в первый и организовать один глобальный массив. Это

достигалось процедурами MPI_Send и MPI_Receive. Стоит отметить, что в случае, если число узлов не делится без остатка на количество процессов, остаток достается последнему процессу. Во избежание ошибок синхронизации потоков ставился барьер после вычислений (MPI_Barrier), заставляющий все остальные потоки дождаться окончания расчёта на последнем, перед тем как передать все данные в один глобальный массив.

В результате, вычисление интеграла на 12 процессорах ускорилось примерно в 9 раз.

Глава 3. Влияние силы Бассэ на направление дрейфа включения

В настоящей работе, определяется пороговое значение μ_p^2 в зависимости от числа плотности D. Преимуществом координат D, μ_p^2 является ограниченность области диаграммы ввиду условий $0 \le D \le 3$, $0 \le \mu_p^2 \le 1$. Для чисел $Re_0 = 0.01,\ 10,100$ и различных чисел Струхаля для каждого из значений D = 0.01, 0.02, ..., 2.99, 3.0 находилось пороговое значения μ_p^2, при переходе через которое включение меняло направление дрейфа. Решалось семейство задач о дрейфе включения из стандартного положения $x_2(0) = 0.75$, $v_2(0) = 0$ при различных частотах и по знаку средней скорости в течение первых десяти периодов определялось направление дрейфа. Сначала оно определялось для малых $\omega\tau = 0.01$. Далее безразмерная частота колебаний увеличивалась до тех пор, пока не происходила смена направления дрейфа. Таким образом, находились пороговые значения μ_p^2, при переходе через которые включение меняло направление дрейфа.

Расчеты показывают, что для заданной плотности включения при увеличении частоты стоячей волны ω достигается пороговое значение квадрата коэффициента увлечения μ_p^2, при переходе через которое волновая сила меняет свое направление на противоположное. На рис. 2 и 3 представлены семейства зависимостей порогового значения μ_p^2 от числа плотности D при различных числах Струхаля для стоксова режима без учета и с учетом сил Бассэ.

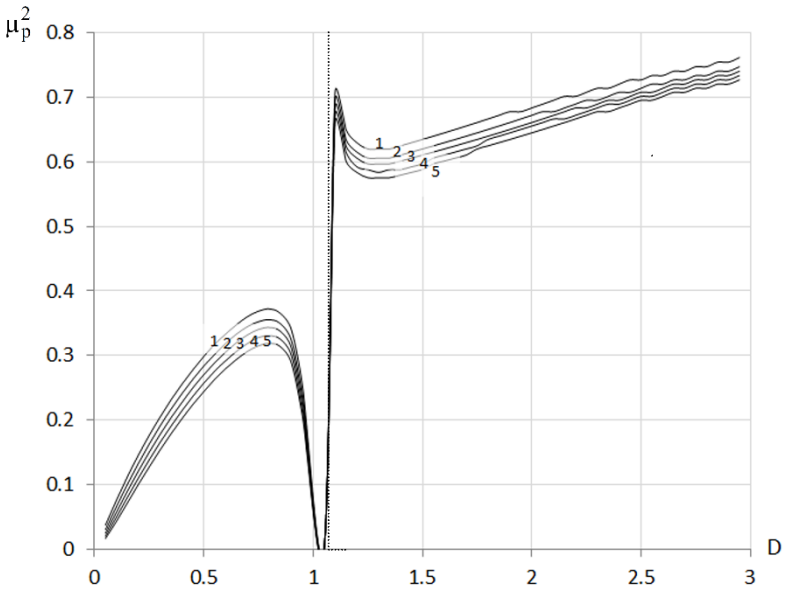

Рис. 2. Зависимость порогового значения от числа плотности. Без учета силы Бассэ. Re0=0.01, Sr=10(1), 20(2), 30(3), 40(4), 50(5).

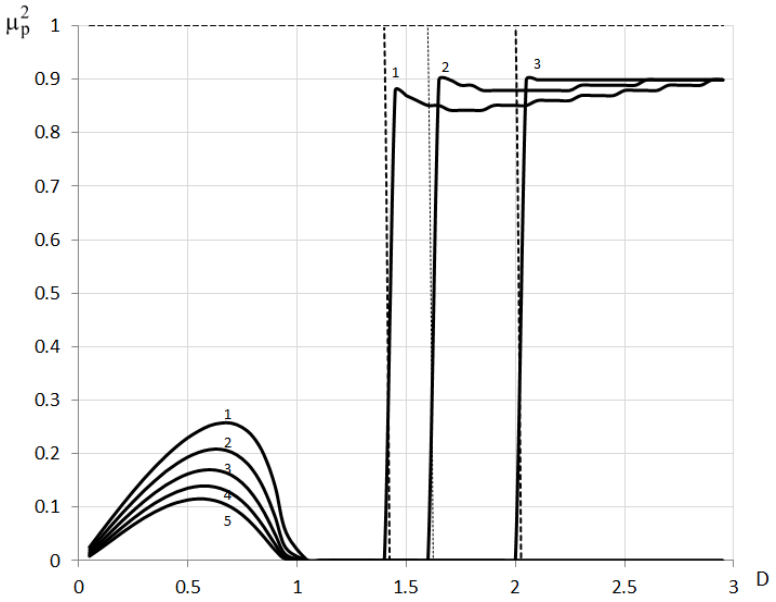

Рис. 3. Зависимость порогового значения от числа плотности. С учетом силы Бассэ. Re0=0.01, Sr=10(1), 20(2), 30(3), 40(4), 50(5).

Без учета силы Бассэ для всех пороговых кривых существует единое пороговое значение числа плотности, при переходе через которое меняется направление волновой силы. Учет силы Бассэ приводит к возникновению разных пороговых значений числа плотности.

На рис. 4 и 5 представлены семейства зависимостей порогового значения μ_p^2 от числа плотности D при различных числах Струхаля для нестоксова режима без учета и с учетом силы Бассэ.

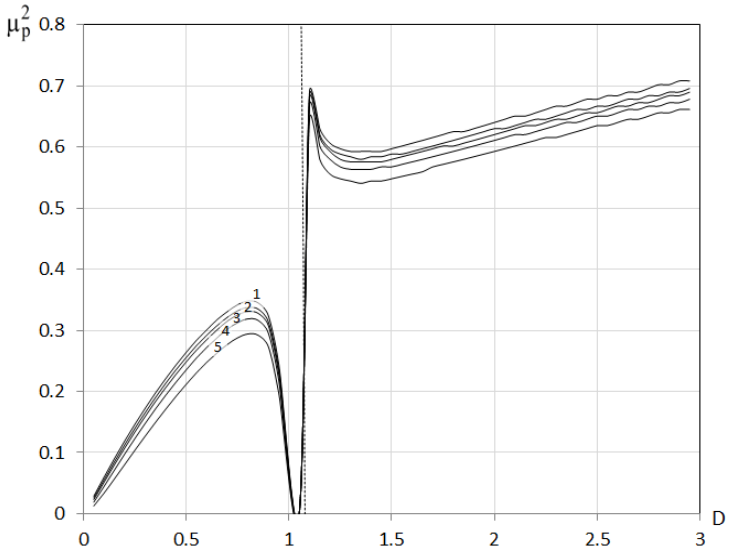

Рис. 4. Зависимость порогового значения от числа плотности. Без учета силы Бассэ. Re0=10, Sr=10(1), 20(2), 30(3), 40(4), 50(5).

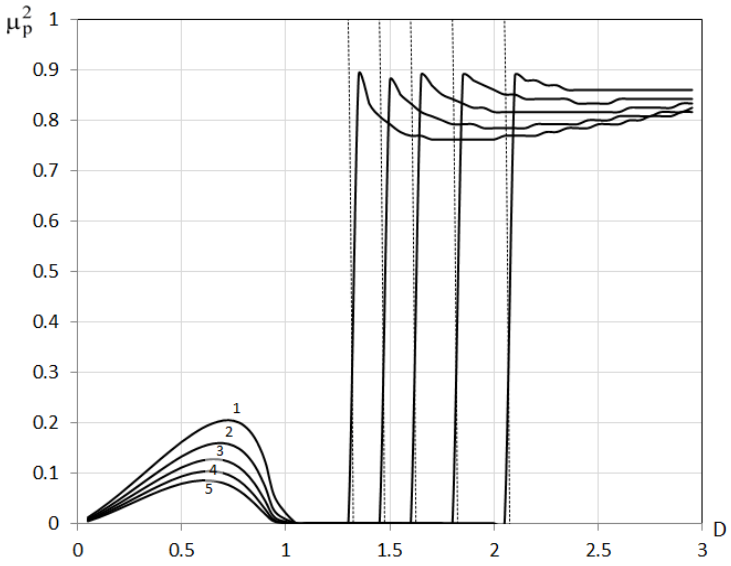

Рис. 5. Зависимость порогового значения от числа плотности. С учетом силы Бассэ. Re0=10, Sr=10(1), 20(2), 30(3), 40(4), 50(5).

На рис. 6 и 7 представлены семейства зависимостей порогового значения μ_p^2 от числа плотности D при различных числах Струхаля для нестоксова режима при числе Рейнольдса $\mathrm{Re}_0 = 100$ без учета и с учетом сил Бассэ.

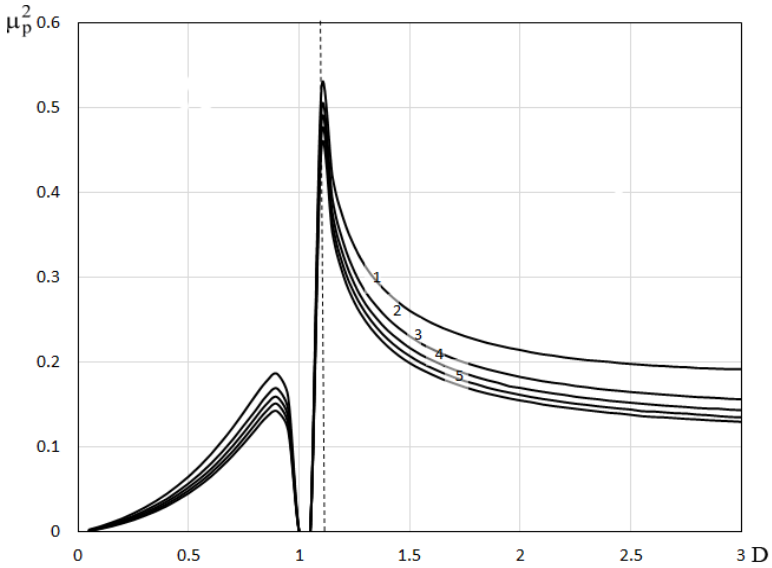

Рис. 6. Зависимость порогового значения от числа плотности. Без учета силы Бассэ. Re0=100, Sr=10(1), 20(2), 30(3), 40(4), 50(5).

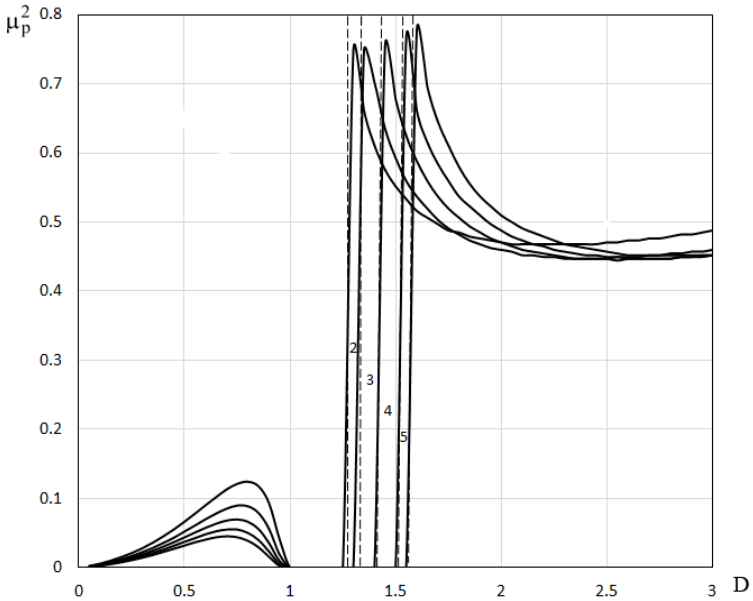

Рис. 7. Зависимость порогового значения от числа плотности. С учетом силы Бассэ. Re0=100, Sr=10(1), 20(2), 30(3), 40(4), 50(5).

Расчеты показывают, что учет силы Бассэ оказывает особенно сильное влияние на пороговые кривые для неплотных включений. При увеличении числа Струхаля пороговое значение μ_p^2 для плотных включений уменьшается. Пороговое значение плотности включения уменьшается (число D увеличивается).

Заключение

Исследован дрейф включения в стоячей синусоидальной волне при различных числах Рейнольдса и Струхаля под действием сил вязкости, присоединенных масс, Архимеда и Бассэ. Показано, что для заданной плотности включения при увеличении частоты стоячей волны рано или поздно достигается пороговое значение частоты, выше которого волновая сила меняет направление. Для различных чисел Рейнольдса и Струхаля найдены зависимости квадрата порогового коэффициента увлечения от относительной плотности включения, с учетом и без учета силы Бассэ. Установлено существование порогового значения относительной плотности включения. При переходе плотности включения через пороговое значение направление волновой силы меняется. Исследовано влияние силы Бассэ на пороговое значение относительной плотности включения. Показано, что учет силы Бассэ оказывает особенно сильное влияние на пороговые кривые для неплотных включений. При увеличении числа Струхаля пороговые значения плотности включения и коэффициент его увлечения уменьшаются.

Список литературы

1. **Нигматулин Р. И.** Динамика многофазных сред. Москва: Наука, 1987.

2. **Ганиев Р. Ф., Украинский Л. Е.** Нелинейная волновая механика и технология. Москва: R&C Dynamics, 2008.

3. **Westervelt P. J.** The theory of steady forces caused by sound waves. J. Acoust. Soc. Am. 1951. Vol. 23, No. 4. Pp. 312–315.

4. **Houghton G.** The behavior of particles in a sinusoidal velocity field. Proc. Royal Society of London, series A. 1962. Vol. 272, No. 3. Pp. 33–43.

5. **Schoeneborn P. R.** The interaction between a single particle and oscillating fluid. Int. J. Multiphase Flow. 1975. Vol. 2. Issue 3. Pp. 307–317.

6. **Sorokin V. S., Blekhman I. I., and Vasilkov V. B.** Motion of a gas bubble in fluid under vibration. Nonlinear Dynamics. 2011. DOI 10.1007/s11071-011-9966-9.

7. **Висицкий Е. В., Петров А. Г., Шундерюк М. М.** Движение частицы в вязкой жидкости под действием силы тяжести и вибрации при наличии силы Бассе. Прикл. матем. и механика. 2009. Т. 73, № 5. С. 763–775.

8. **Невский Ю. А., Осипцов А. Н.** О роли нестационарных и "наследственных" сил в задачах гравитационной конвекции суспензий. Вести Московского ун-та. Серия 1: Математика, механика. 2008. № 4. С. 37–44.

9. **King L.** On the acoustic radiation pressure on spheres. Proc. Roy. Soc. A. 1934. Vol. 147, No. 861. Pp. 212–240.

10. **Yosioka K. and Kawasima G.** Acoustic radiation pressure on compressible spheres. Acustica. 1955. Vol. 5. Pp. 167–173.

11. **Yosioka K., Kawasima G., and Hirano H.** Acoustic radiation pressure on bubbles and their logarithmic decrement. Acustica. 1955. Vol. 5. Pp. 173–178.

12. **Westervelt P. J.** The mean pressure and velocity in a plane acoustic wave in a gas. J. Acoust. Soc. Am. 1950. Vol. 22, No. 3. Pp. 319–327.

13. **Westervelt P. J.** Acoustic radiation pressure. J. Acoust. Soc. Am. 1957. Vol. 29, No. 1. Pp. 26–29.

14. **Горьков Л. П.** О силах, действующих на малую частицу в акустическом поле в идеальной жидкости. Докл. АН СССР.
1961. Т. 140, № 1. С. 88–91.

15. **Духин С. С.** Теория дрейфа аэрозольной частицы в стоячей звуковой волне. Коллоидный журнал. 1960. Т. 22, № 1. С. 128–130.

16. **Crum L. A. and Eller A. I.** Motion of bubbles in a stationary sound field. J. Acoust. Soc. Am. 1970. Vol. 48, No. 1(2). Pp. 181–189.

17. **Czyz H.** On the concentration of aerosol particles by means of drift forces in a standing wave field. Acustica. 1990. Vol. 70. Pp. 23–28.

18. **Czyz H.** The aerosol particle drift in a standing wave field. Arch. Acoust. 1987. Vol. 12, Nos. 3–4. Pp. 199–214.

19. **Czyz H. and Snakowski J. K.** Influence of acoustical field on small particles. Le Journal de physique. 1994. Vol. 4. Pp. 861–864.

20. **Губайдуллин Д. А., Осипов П. П.** Влияние гидродинамических сил на дрейф включений в волновых полях. Проблемы энергетики. 2010. № 1–2. С. 3–13.

21. **Губайдуллин Д. А., Осипов П. П.** О некоторых режимах дрейфа включений в акустических полях. Инженерно-физический журнал. 2011. Т. 84, № 2. С. 255–262.

22. **Zhe Cui, Li Yang, and Fan L. S.** Bubble modulation using acoustic standing waves in a bubbling system. Chem. Eng. Sci. 2005. Vol. 60. Pp. 5971–5981.

23. **Kutushev A. G.** Non-Stationary Shock Waves in Two-Phase Gas-Particle or Gas-Droplet Mixtures. St. Petersburg: Nedra, 2003.

24. **Boiko V. M. and Poplavskii S. V.** Particle and drop dynamics in the flow behind a shock wave. J. Fluid Dynamics. 2007. Vol. 42, No. 3. Pp. 433–441.

25. **Goldshtein A., Shuster K., Vainshtein P., Fichman M., and Gutfinger C**. Particle motion in resonance tubes. J. Fluid Mechanics. 1998. Vol. 360. Pp. 1–20

26. **Губайдуллин Д.А., Осипов П.П.** Влияние чисел Рейнольдса и Струхаля на направление волновой силы, действующей на включения в стоячей синусоидальной волне// ИФЖ, 2013. Т. 86. №1. С. 50-58

ПРИЛОЖЕНИЕ 1

Программа поиска пороговых значений коэффициента увлечения

```
#include <stdio.h>
#include <vector>
#include <math.h>
#include <time.h>
#include <mpi.h>


char time_str[80];

const int L_COUNT = 25;
const int M_COUNT = 60;

//output files
const char * cOutRo12 = "Ro12.txt ";
const char * cOutD = "D.txt";
const char * cOutWTau2 = "wTau2.txt";
const char * cOutMyP2 = "MyP2.txt";
const char * cOutDir1 = "Dir1.txt";

double dOutRo12[M_COUNT];
double dOutD[M_COUNT];
double dOutWTau2[M_COUNT];
double dOutMyP2[M_COUNT];
int iOutDir1[M_COUNT];

//time measurement function
char * get_local_time()
{
      time_t rawtime;
      struct tm timeinfo;
      time(&rawtime);
      localtime_s ( &timeinfo, &rawtime );
      strftime ( time_str,  80, "%H:%M:%S", &timeinfo );
      return time_str;
}

//Duhamel integral calculation function
double duhamel_int(double *t, double *v2, double *x2, unsigned int& n, const
double& delT, const double& St, unsigned int start, unsigned int stop)
{
      double s = 0.0;
      for(unsigned int k = start; k <= stop; k++)
      {
            double z1 = n - k;
            double z2 = n - (k - 1);
            s += (f(t[k], x2[k], v2[k], St) + f(t[k-1], x2[k-1], v2[k-1], St) -
2.0*(v2[k] - v2[k-1]) / delT) / (sqrt(z1) + sqrt(z2));
      };
      return s;
}

int main(int argc, char **argv)
{
      //MPI initialization
      int npes, mype, ierr;
      ierr = MPI_Init(&argc, &argv);
      MPI_Status status;
```

```
ierr = MPI_Comm_size(MPI_COMM_WORLD, &npes);//number of nodes
ierr = MPI_Comm_rank(MPI_COMM_WORLD, &mype);//current node number
printf("Reynolds + Inertia + Basset\n");

double delT(0), St(0), ro12(0), Re0(0),
            Re(0), kri(0),  del(0),
            wTau(0), wTau2(0), myP2(0), beta(0),
            v1(0), dv1dt(0), dv1dx(0), D(0), vAver(0), h(0);

unsigned int nt(0);
int directionNew(0), direction1(0);

FILE *outRo12, *outD, *outWTau2, *outMyP2, *outDir1;

vector <double> tt;
vector <double> x2;
vector <double> v2;

double duh = 0;
double K = 0;

const double pi = 4.0*atan(1.0);

try
{
      if (mype == 0)
      {
            printf("Start time: %s\n", get_local_time());
            if ( fopen_s(&outRo12, cOutRo12, "w") != 0)
                  throw std::exception("Cannot open file ro12.txt.");
            if ( fopen_s(&outWTau2, cOutWTau2, "w") != 0)
                  throw std::exception("Cannot open file wTau2.txt.");
            if ( fopen_s(&outMyP2, cOutMyP2, "w") != 0)
                  throw std::exception("Cannot open file outMyP2.txt.");
            if ( fopen_s(&outD, cOutD, "w") != 0)
                  throw std::exception("Cannot open file outD.txt.");
            if ( fopen_s(&outDir1, cOutDir1, "w") != 0)
                  throw std::exception("Cannot open file outDir1.txt.");
      }

      St = 50.0;
      Re0 = 100;
      beta = 1.0 + sqrt(Re0)/6.0 + Re0/60.0;

      kri = 2.0*pi/St;     // criterion of smallness of amplitude

      for (int m = 1; m <= M_COUNT; m++)
      {
            clock_t begin_time = clock();
            D = m*0.05;
            ro12 = 2.0*D/(3.0 - D);
            if (D == 1)
                  continue;

            h = 0.1;
            wTau2 = 0.0;
            for(int l = 1; l <= L_COUNT; l++)
            {
                  wTau2 = wTau2 + h;
                  myP2 = 1.0/(1.0+wTau2);
                  wTau = sqrt(wTau2);
                  K = wTau*3.0*D;
```

```
                        delT = 2.0*pi/200.0;

                        nt = static_cast<unsigned int>(2.0*pi*30.0/delT);

                        vAver = 0.0;

                        tt.resize(nt + 3); //new
                        x2.resize(nt + 3); //new
                        v2.resize(nt + 3); //new

                        x2[1] = 0.75;
                        v2[1] = 0.0;
                        //Diff. equation
                        for(unsigned int n = 1; n <= nt + 1; n++)
                        {
                              tt[n] = (n-1)*delT;

                              vAver = (vAver*(n - 1) + v2[n])/n;

                              v1 = sin(pi*x2[n])*sin(tt[n]);
                              dv1dt = sin(pi*x2[n])*cos(tt[n]);
                              dv1dx = pi*cos(pi*x2[n])*sin(tt[n]);

                              Re = Re0*fabs(v2[n] - v1);
                              del = 1.0 + sqrt(Re)/6.0 + Re/60.0;

                              x2[n + 1] = x2[n] + v2[n]/St*delT;
                              int cc = floor((double)(n/npes));
                              if (n > 1)
                              {
                                    if(n < npes)
                                    {
                                          if(mype == 0)
                                                duh = duhamel_int(tt, v2, x2, n,
delT, St, 1, n);
                                    }
                                    else
                                    {
                                          duh = 0;
                                          unsigned int remainder = n%npes;
                                          unsigned int start = mype*cc + 1;
                                          unsigned int stop = (mype + 1)*cc;
                                          if(mype == npes - 1)
                                                stop += remainder;
                                          duh = duhamel_int(tt, v2, x2, n, delT,
St, start, stop);
                                    }
                              }
                              duh1 = 0;
                              MPI_Barrier(MPI_COMM_WORLD);
                              int error1 = MPI_Reduce(&duh, &duh1, 1,
MPI_DOUBLE, MPI_SUM, 0, MPI_COMM_WORLD);
                              int error2 = MPI_Bcast(&duh1, 1, MPI_DOUBLE, 0,
MPI_COMM_WORLD);

                              v2[n + 1] = v2[n] - (v2[n] - v1)*3.0*D/K*del*delT
+ D*(dv1dt + v1*dv1dx/St)*delT + duh*3.0*D/sqrt(K)*delT;
                        };

                        if (l == 1)
                        {
                              if (vAver > 0.0) direction1 = 1;
```

```
                              if (vAver < 0.0) direction1 = -1;
                              if (vAver == 0.0) direction1 = 0;
                              // printf("dir1 = %d\n", direction1);
                        }
                        else
                        {
                              if (vAver > 0.0) directionNew = 1;
                              if (vAver < 0.0) directionNew = -1;
                              if (vAver == 0.0) directionNew = 0;
                        };

                        if (l > 1)
                        {
                              if ((directionNew*direction1) == -1)
                              {
                                    wTau2 = wTau2 - h;
                                    h = h/2.0;
                              };
                              if ((directionNew*direction1) == 1)
                              {
                                    h = h*2.0;
                              };
                              if (h < 0.01)
                              {
                                    break;
                              };
                        };
                  }; // for l
                  dOutRo12[m - 1] = ro12;
                  dOutWTau2[m - 1] = wTau2;
                  dOutMyP2[m - 1] = myP2;
                  dOutD[m - 1] = D;
                  iOutDir1[m - 1] = direction1;
                  printf("%d ; %d\n", m, ( clock () - begin_time ) /
CLOCKS_PER_SEC);
            };   // for m
            if (npes > 1)
            if (mype != 0)
            {
                  printf("Process %d sending.\n", mype);
                  MPI_Send(&dOutRo12[mype*cc], cc, MPI_DOUBLE, 0, 0,
MPI_COMM_WORLD);
                  MPI_Send(&dOutWTau2[mype*cc], cc, MPI_DOUBLE, 0, 1,
MPI_COMM_WORLD);
                  MPI_Send(&dOutMyP2[mype*cc], cc, MPI_DOUBLE, 0, 2,
MPI_COMM_WORLD);
                  MPI_Send(&dOutD[mype*cc], cc, MPI_DOUBLE, 0, 3,
MPI_COMM_WORLD);
                  MPI_Send(&iOutDir1[mype*cc], cc, MPI_INT, 0, 4,
MPI_COMM_WORLD);
                  printf("Process %d sent.\n", mype);
            }
            else
            {
                  for (unsigned int i = 1; i < npes; i++)
                  {
                        printf("Receiving %d.\n", i);
                        MPI_Recv(&dOutRo12[i*cc], cc, MPI_DOUBLE, i, 0,
MPI_COMM_WORLD, &status);
                        MPI_Recv(&dOutWTau2[i*cc], cc, MPI_DOUBLE, i, 1,
MPI_COMM_WORLD, &status);
```

```
                        MPI_Recv(&dOutMyP2[i*cc], cc, MPI_DOUBLE, i, 2,
MPI_COMM_WORLD, &status);
                        MPI_Recv(&dOutD[i*cc], cc, MPI_DOUBLE, i, 3,
MPI_COMM_WORLD, &status);
                        MPI_Recv(&iOutDir1[i*cc], cc, MPI_INT, i, 4,
MPI_COMM_WORLD, &status);
                        printf("Received %d.\n", i);
                  }
            }
            printf("Barrier reached by process %d.\n", mype);
            //MPI_Barrier(MPI_COMM_WORLD);
            printf("Barrier breached.\n", mype);
            if (mype == 0)
            {
                  printf("End time: %s\n", get_local_time());
                  for (int i = 0; i < M_COUNT; i++)
                  {
                        // // printf("      ro12=%f D=%f wTau2=%f delT=%f
dir1=%d  \n", ro12, D, wTau2, delT, direction1);
                        fprintf(outRo12,"%f\n", dOutRo12[i]);
                        fprintf(outD,"%f\n", dOutD[i]);
                        fprintf(outWTau2,"%f\n", dOutWTau2[i]);
                        fprintf(outMyP2,"%f\n", dOutMyP2[i]);
                        fprintf(outDir1,"%d\n", iOutDir1[i]);
                  }
                  fclose(outRo12);
                  fclose(outWTau2);
                  fclose(outMyP2);
                  fclose(outD);
                  fclose(outDir1);
            }

            // printf("St = %f  kri = %f D = %f \n", St, kri, D);

      }
      catch(std::exception& e)
      {
            printf("%s\n", e.what());
      }
      MPI_Finalize();
      if (mype != 0)
            return 0;
      else
            return getchar();
}
```