

МОДИФИКАЦИЯ ДВУМЕРНОГО АЛГОРИТМА РАСЧЕТА ИДЕАЛЬНОЙ ДЕТОНАЦИИ В МЕТОДИКЕ ЛЭГАК НА ОСНОВЕ ПРИНЦИПА ГЮЙГЕНСА

М. Н. Кирюхина, А. О. Наумов

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

В настоящее время метод контроля скорости детонационной волны (ДВ) для расчета детонации взрывчатых веществ (ВВ) используется практически во всех многомерных методиках. В методике ЛЭГАК [1] до текущего момента были реализованы геометрические методы контроля скорости ДВ и приближенный пошаговый метод, в котором размер «области влияния» задается пользователем. К недостаткам геометрических методов можно отнести, во-первых, постоянство скорости распространения ДВ, во-вторых, сложность реализации в случае сложной «геометрии» ВВ с большим количеством зон затенения. Применение же пошагового метода приводило в некоторых случаях к заметным ошибкам. Кроме того, этот метод не обладает сходимостью численного решения при уменьшении шага счетной сетки. В связи с вышесказанным, было принято решение разработать и реализовать более точный алгоритм.

Рассмотрим постановку проблемы расчета идеальной детонации с применением принципа Гюйгенса. Согласно этому принципу, каждая точка, до которой доходит волновое движение, служит источником вторичных волн; огибающая этих волн дает положение фронта волны.

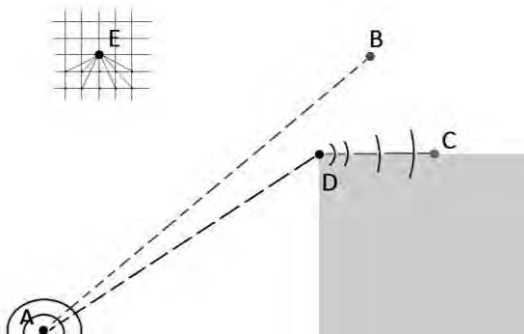


Рис. 1. Пример геометрии. Вещества: светлый – ВВ, темный – инерт

Рассмотрим точку А (рис. 1): пусть она является первичным источником ДВ. Каждая точка, до которой дошла волна от точки А, становится вторичным источником. До точки В (рис. 1) первичная волна идет без преград, а значит по кратчайшему расстоянию. Вычислить время прихода ДВ в точку В не составляет труда.

Рассмотрим теперь точку С (рис. 1). По прямой из точки А ДВ не дойдет, так как встретит преграду в

виде инертного вещества. Но волна дойдет до точки D, которая станет вторичным источником, и уже от нее – до точки С (именно по такому пути из соображения кратчайшего расстояния).

В методике ЛЭГАК стандартным является приближенный пошаговый алгоритм, который далее будем называть «fast deton». Он, в отличие от нового алгоритма (назовем его new), работает по другому принципу. Для расчета времени прихода ДВ, пользователь задает размеры области анализа выгорания (число ячеек). Например, для расчета времени прихода ДВ в узел E (рис. 1), задана окрестность в 2 ячейки. В пределах этой области проводится поиск узлов с уже заполненным временем. Каждый из таких узлов рассматривается как источник ДВ, от которого рассчитывается время в узле E (в случае прямой видимости узла). В качестве итогового времени – выбирается наименьшее из полученных. Соответственно, чем большая окрестность задается, тем точнее получается результат. Однако, при задании большой окрестности, возникают сложности с распараллеливанием в случае, когда узел лежит на границе фрагмента.

В новом алгоритме собирается и хранится информация о точках, которые являются источниками волн: в данном случае это точки А и D (рис. 1). С помощью этой информации происходит проверка прямой видимости между возможной точкой-источником и рассматриваемой точкой. В качестве источника выбирается точка с наименьшим расстоянием до рассматриваемого узла.

Изначально за основу была взята идея из методики ЭГАК [2] – определение и хранение центров иницирования для всех узлов. Все центры иницирования было решено хранить в специально созданной структуре. При расчете времени прихода ДВ проверялась прямая видимость между рассматриваемым узлом и точкой иницирования. Хотя алгоритм позволил практически точно определять время прихода ДВ, он был очень неэкономичен, а кроме того возникали большие сложности с его распараллеливанием.

Для оптимизации работы алгоритма было решено определять точку иницирования по локальным узлам, окружающим рассматриваемый узел при помощи дополнительной информации о точках иницирования. Такой алгоритм был довольно просто распараллелен. Однако в ходе тестирования выяснилось, что возникали ситуации, когда точка иници-

рования определялась некорректно в силу нехватки условий, по которым можно было бы отсеять неправильные варианты. Ошибки возникали при огибании ДВ препятствий в форме окружности.

В силу обозначившихся проблем было принято решение модифицировать алгоритм проверки прямой видимости. Новый алгоритм позволил учесть наличие инертного вещества не только в окрестности рассматриваемого узла, но и в окрестностях всех точек инициирования, которые могли оказать влияние на время прихода ДВ.

Сначала рассмотрим сам принцип работы новой программы. Пусть изначально в задаче заданы точки инициирования детонации, например, точка с индексом 1 (рис. 2). При считывании информации о задаче, мы заполняем структуру точек инициирования, а также в отдельную структуру вносим индексы узлов, для которых мы будем вычислять время прихода ДВ в первую очередь. В данном случае это будут узлы с индексом 2 (рис. 2). Эти узлы выбираются из предположения, что за один счетный шаг ДВ не может уйти на расстояние большее, чем длина стороны ячейки. Для каждого узла проверяется наличие прямой видимости до точки инициирования (в данном случае она одна), затем заполняется время. Если насчитанное время больше, чем время в задаче на следующем временном шаге, то время в массив не заносится. Если время в рассмотренном узле было заполнено, то вызывается подпрограмма, которая внесет в список узлов – новые узлы – соседей рассматриваемого. В данном случае – узлы с индексом 3 (рис. 2). Таким образом, мы расширяем массив узлов и постепенно заполняем времена прихода ДВ в них.

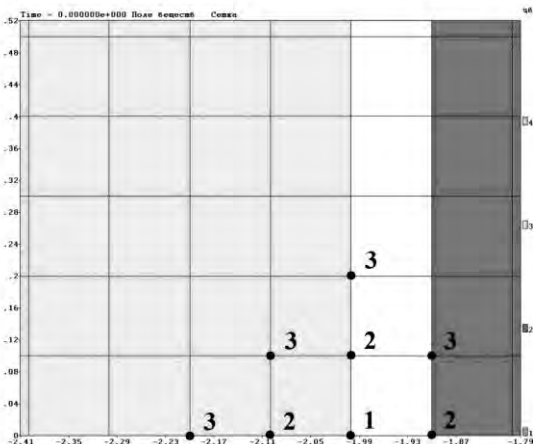


Рис. 2. Схема порядка расчета узлов сетки. Вещества: светлый – ВВ, темный – инерт, белый – смешанные ячейки

Теперь остановимся непосредственно на алгоритме проверки прямой видимости. Рассмотрим рис. 3 и на нем – узел с индексом А.

Пусть среди его соседей (ближайшие 8 узлов) есть узлы, проинициированные от точек 1, 2, 3. Следовательно, мы должны проверить, не является ли какая-нибудь из них и точкой инициирования для узла А.

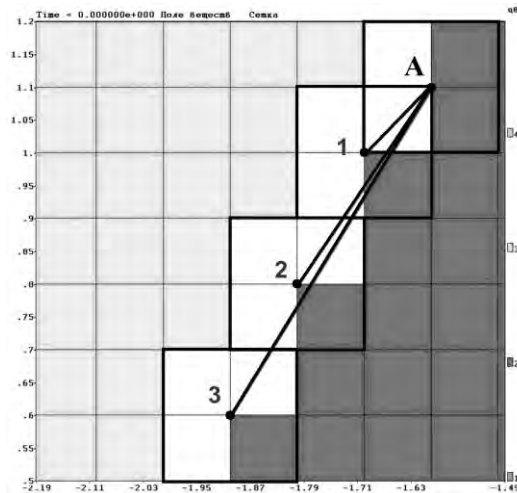


Рис. 3. Схема алгоритма проверки прямой видимости. Вещества: светлый – ВВ, темный – инерт, белый – смешанные ячейки

Рассмотрим 1 точку и отрезок, соединяющий ее и узел А. Теперь нам нужно проверить пересечение этого отрезка с инертном в окрестности одного слоя ячеек вокруг узла А и узла 1. Видим, что отрезок пересекает смешанную ячейку. На данный момент реализована возможность задания концентрации ВВ в ячейке, при которой она будет считаться «прозрачной» для ДВ. Будем считать, что в данном примере смешанные ячейки прозрачны и точка 1 нам подходит.

Рассмотрим теперь точку 2 и отрезок, соединяющий ее с узлом А. Проверяем пересечение отрезка: в окрестности узла А нет пересечения с инертном; в окрестности узла 2 нет пересечения. Казалось бы, что точка 2 нам подходит, но теперь мы рассмотрим известные нам точки инициирования, лежащие (геометрически) между узлами А и 2 – точка 1. Теперь проверим пересечение рассматриваемого отрезка с инертном в окрестности точки 1. Видно, что в данной окрестности отрезок пересекает инертное вещество, а значит, прямой видимости между узлами А и 2 нет. Аналогичная картина наблюдается в случае с точкой 3. Таким образом, дополнительное условие позволило избежать ошибки в определении точки инициирования для узла А.

В случае многопроцессорного счета алгоритм работает точно также, потому что вся необходимая для проверок информация хранится в структуре с точками инициирования, а обмены этой структурой выполняются на каждом временном шаге.

А теперь рассмотрим работу новой программы на примере расчетов нескольких методических задач. Результаты сравнивались с аналитически рассчитанными временами прихода ДВ в узлы, а также с алгоритмом пошагового контроля скорости детонации, являющимся стандартным в методике ЛЭГАК.

Задача 1

Рассматривается квадратная область $\{x \in [0; 10], y \in [0; 10]\}$, полностью заполненная ВВ. Детонация инициируется в центре области. Скорость движения

детонационной волны $V_{det} = 8,83$. Расчеты проводились на сетках 800×800 и 1000×500 ячеек. Для сравнения двух алгоритмов были исследованы времена прихода ДВ в 5 точках области.

На рис. 4 представлены поля веществ на момент времени $t = 0,5$ для двух алгоритмов детонации, а – new, б – fast deton. Расчет проводился на квадратной сетке. У стандартного алгоритма в исследуемых точках была получена погрешность в пределах 0,25 %.

В то же время, при расчетах на прямоугольной сетке, видно, что погрешность заметно увеличилась и в некоторых точках достигла 4 % (рис. 5). Новый же алгоритм позволил вычислить времена прихода с нулевой погрешностью на обеих сетках.

Задача 2

Начальная геометрия представлена на рис. 6. Область $\{x \in [-1,2; 1,9], y \in [-1,25; 1,75]\}$, скорость детонации $V_{det} = 10$, сетка 295×310 ячеек.

Результаты расчетов показали, что визуально фронт волны описывается обоими алгоритмами практически одинаково (рис. 7). Сравнение же времен прихода ДВ в указанные ранее узлы показало, что новый алгоритм позволил получить точное решение и во внутренних, и во внешних точках. Стандартный алгоритм точно рассчитал времена во внутренних точках, а во внешних погрешность составила порядка 1 %.

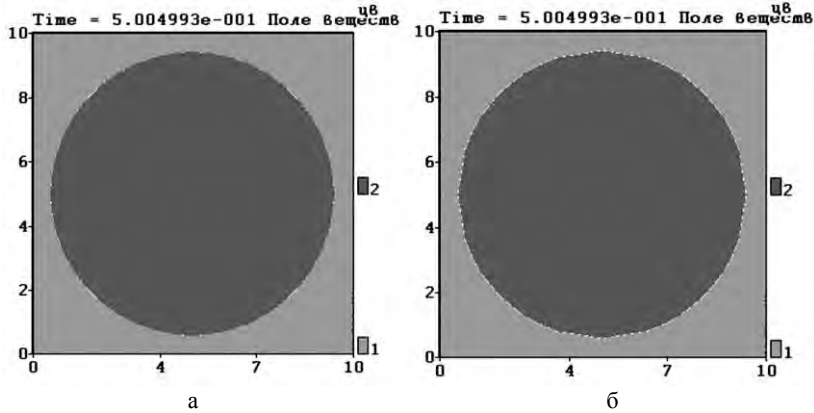


Рис. 4. Поле веществ на момент времени $t = 0,5$: а – new, б – fast deton. Квадратная сетка. Вещества: 1 – ВВ, 2 – продукты взрыва (ПВ)

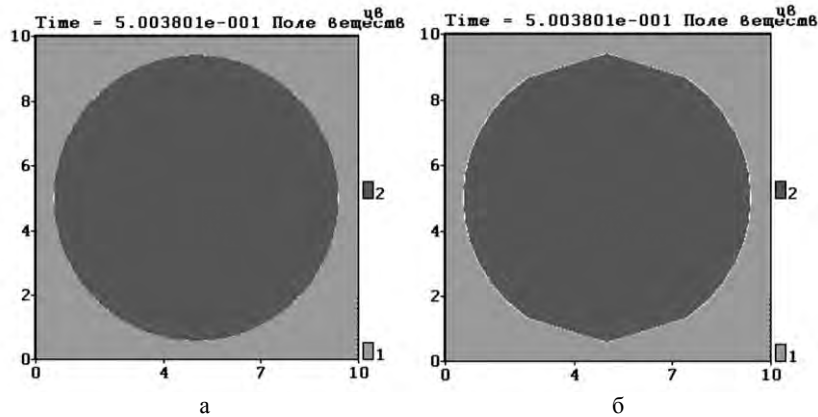


Рис. 5. Поле веществ на момент времени $t = 0,5$: а – new, б – fast deton. Прямоугольная сетка. Вещества: 1 – ВВ, 2 – ПВ

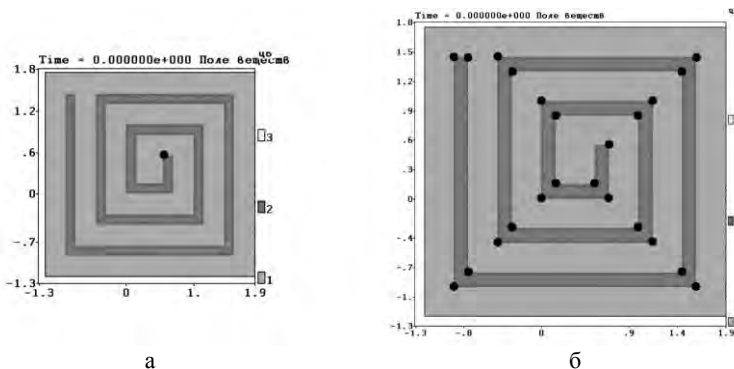


Рис. 6. Начальная геометрия задачи 2: а – маркер – точка иницирования, б – маркеры – исследуемые точки. Вещества: 1 (светлый) – инерт, 2 (темный) – ВВ, 3 (белый) – ПВ

Задача 3

Начальная геометрия представлена на рис. 8: а – квадратная сетка 240×400 ячеек, б – полярная сетка 100×200 ячеек. Область $\{x \in [-5; 5], y \in [0; 6]\}$, скорость детонации $V_{det} = 10$. В случае квадратной сетки был введен вакуум для описания сферической поверхности.

На рис. 9 представлено поле веществ на момент времени $t = 0,45$, расчет на квадратной сетке: а – new, б – fast deton. На рис. 10 представлено поле веществ на тот же момент времени, но на полярной сетке: а – new, б – fast deton. Видно, что положение фронта ДВ с хорошей точностью совпадает для всех расчетов. На рис. 11 приведен график погрешностей вре-

мени прихода ДВ в экваториальную и полюсную точки области. Видно, что на квадратной сетке погрешность заметно выше, что объясняется погрешностью аппроксимации окружности ломаной линией. В случае полярной сетки погрешности малы, но новый алгоритм дает более точный результат (на порядок).

Кроме того, на примере этой задачи было проведено исследование на сходимость решения при измельчении шага квадратной счетной сетки. Из рис. 12 видно, что новый алгоритм демонстрирует сходимость численного решения к точному решению в обеих точках, в то время как при использовании стандартного алгоритма сходимость нет вообще.

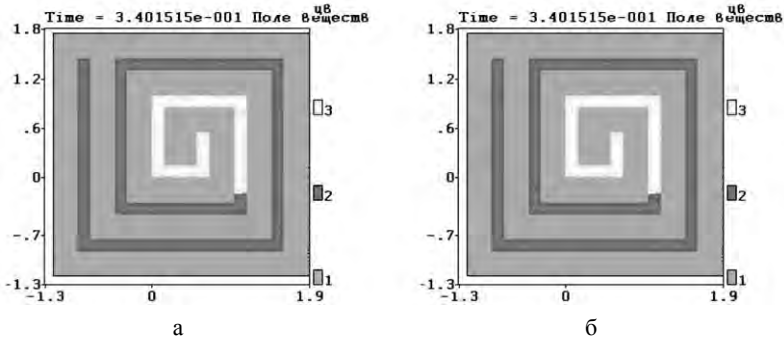


Рис. 7. Поле веществ на момент времени $t = 0,34$: а – new, б – fast deton. Вещества: 1 (светлый) – инерт, 2 (темный) – ВВ, 3 (белый) – ПВ

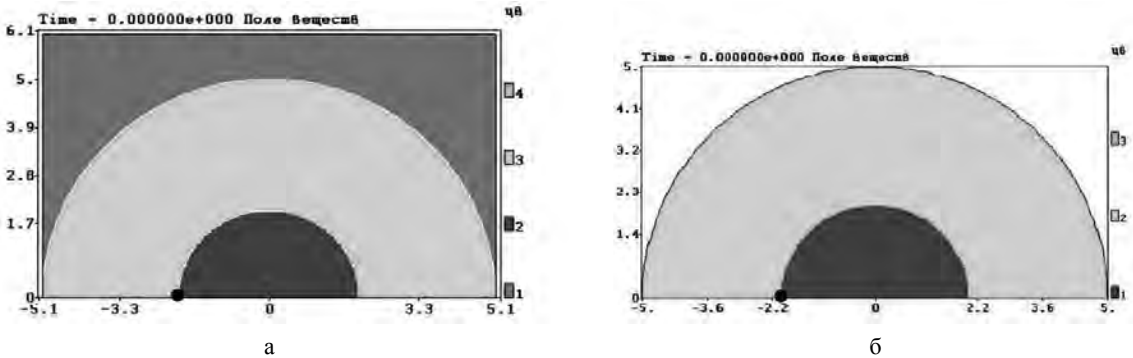


Рис. 8. Начальная геометрия задачи 3: а – квадратная сетка, б – полярная сетка. Маркер – точка иницирования. Вещества: внешний темно-серый (а) – вакуум, темный – инерт, светлый – ВВ

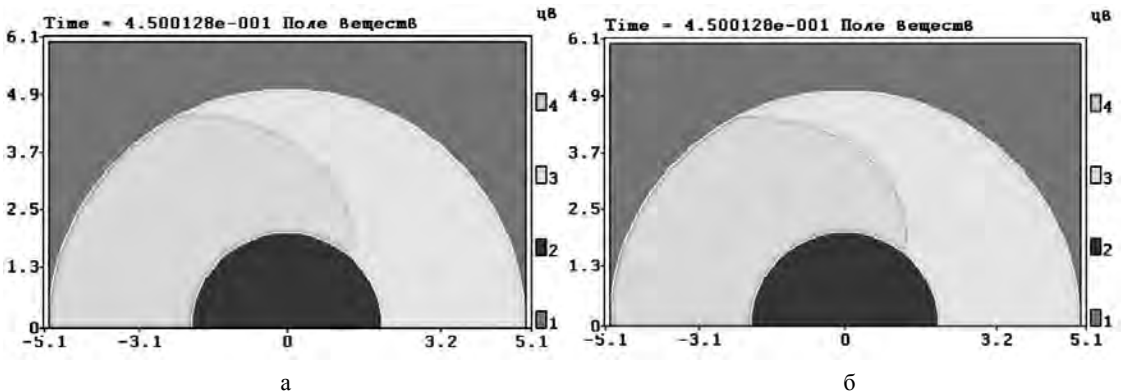


Рис. 9. Поле веществ на момент времени $t = 0,45$: а – new, б – fast deton. Квадратная сетка. Вещества: 1 – вакуум, 2 – инерт, 3 – ВВ, 4 – ПВ

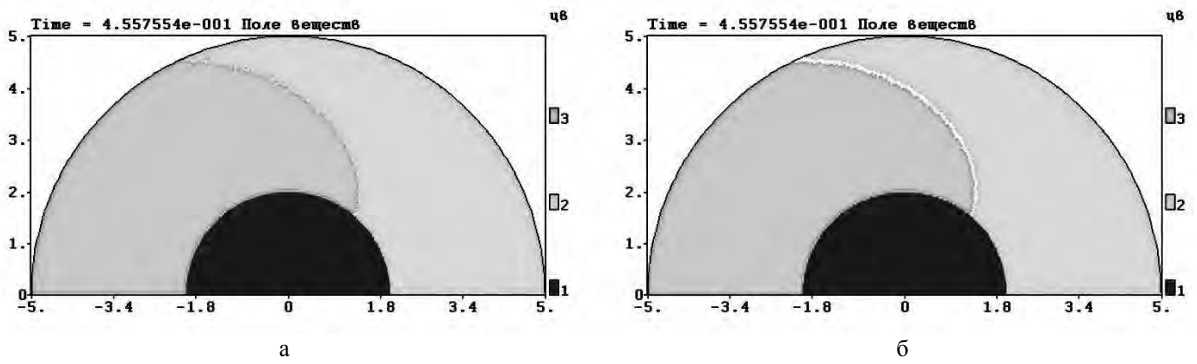


Рис. 10. Поле веществ на момент времени $t = 0,45$: а – new, б – fast deton. Полярная сетка.
Вещества: 1 – инерт, 2 – ВВ, 3 – ПВ



Рис. 11. График погрешности расчета времен в двух точках области при расчетах на двух типах сеток

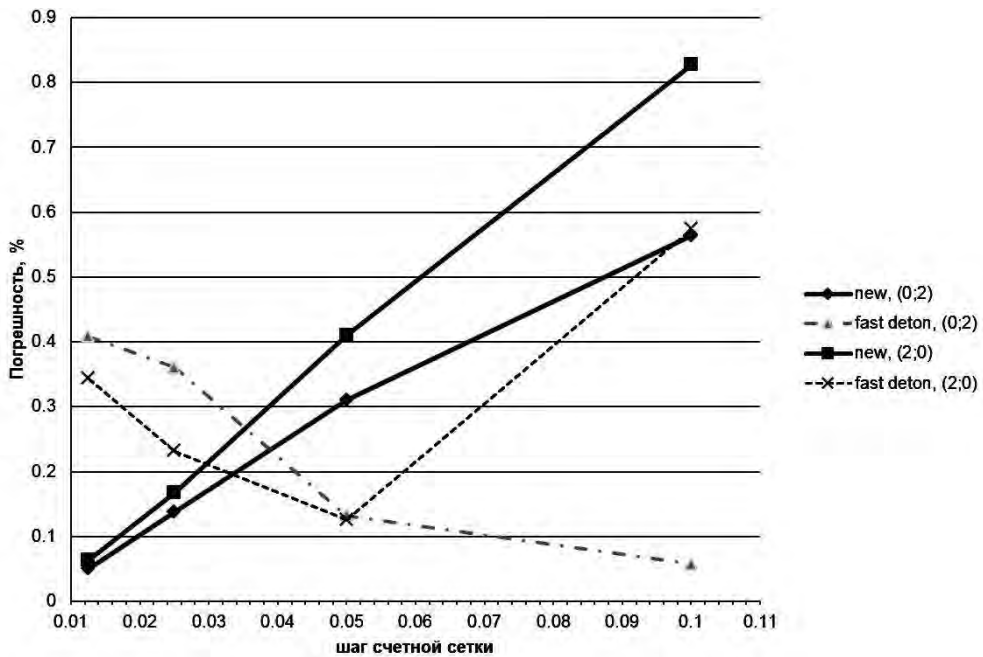


Рис. 12. График погрешностей в % для двух алгоритмов расчета детонации для двух точек в зависимости от шага счетной сетки

На примере этой же задачи также были проведены исследования эффективности распараллеливания. На рис. 13 приведен график, на котором отмечена эффективность работы двух алгоритмов при различном числе ядер. Видно, что новый алгоритм уступает стандартному по времени работы. В однопроцессорном режиме время работы новой программы детонации больше примерно на 1,5 %, в многопроцессорном – 24–37 %.

Задача 4

Начальная геометрия представлена на рис. 14: а – красный маркер – точка иницирования, б – синие маркеры – исследуемые точки. Область $\{x \in [-5; 5], y \in [-5; 5]\}$, скорость детонации $V_{det} = 10$, квадратная сетка 200×200 ячеек.

На рис. 15 представлены поля веществ на момент времени $t = 0,7$: а – new, б – fast deton. Видно хорошее согласие положения фронтов ДВ.

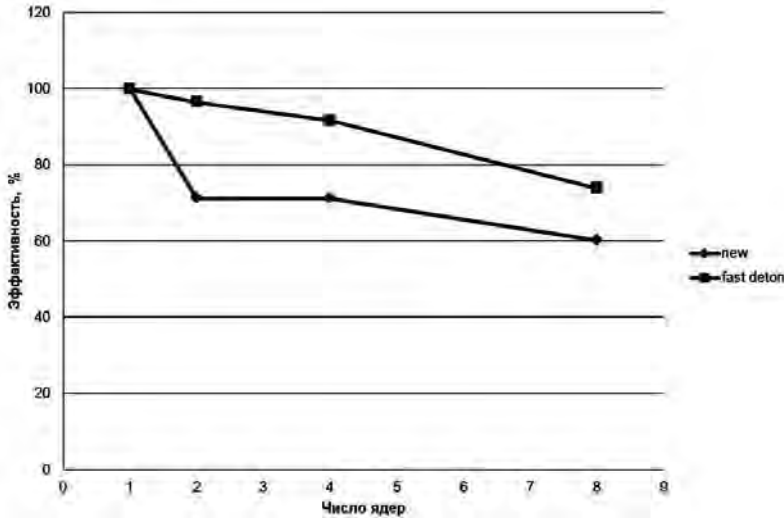


Рис. 13. Эффективность работы двух алгоритмов при различном числе узлов

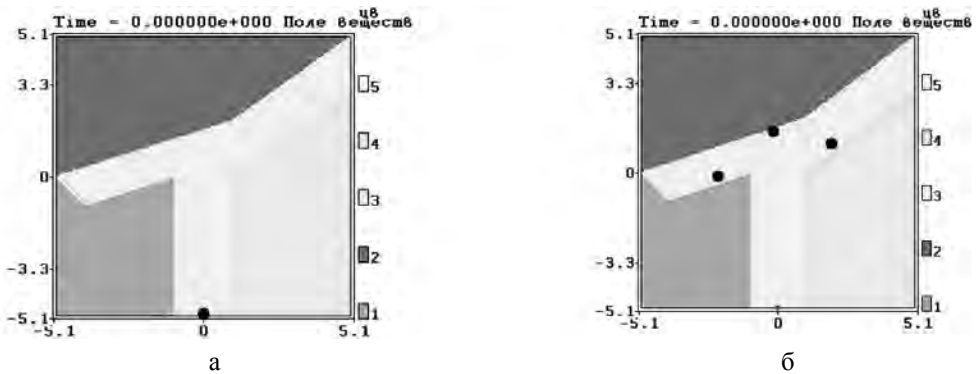


Рис. 14. Начальная геометрия задачи 4: а – маркер – точка иницирования, б – маркеры – исследуемые точки. Вещества: 1 – инерт, 2 – инерт, 3 – ВВ, 4 – инерт, 5 – ПВ

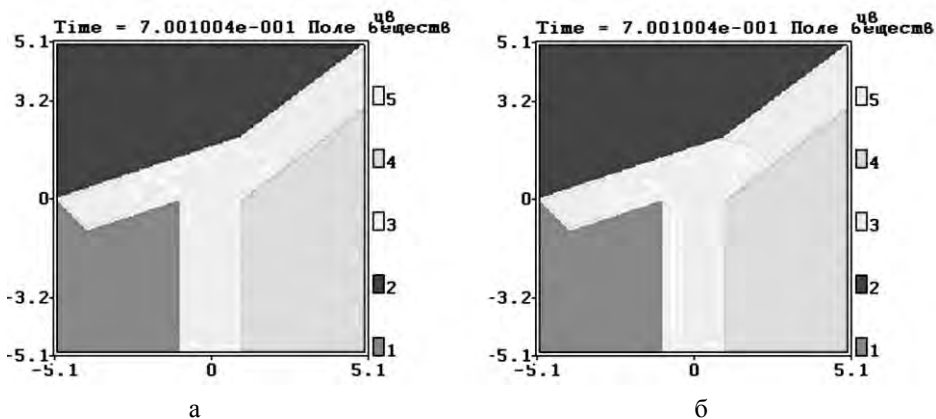


Рис. 15. Поле веществ на момент времени $t = 0,7$: а – new, б – fast deton. Вещества: 1 – инерт, 2 – инерт, 3 – ВВ, 4 – инерт, 5 – ПВ

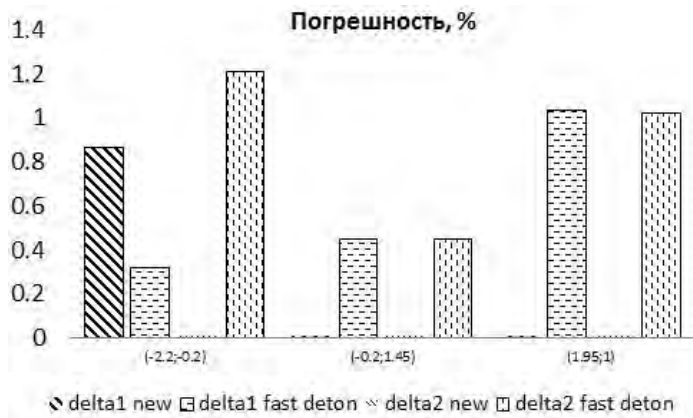


Рис. 16. График погрешностей расчета времени прихода ДВ в три точки для двух алгоритмов

На рис. 16 представлен график погрешности для трех исследуемых точек. Погрешность, обозначенная как δ_1 , рассчитывалась относительно аналитического решения для исходной геометрии. Погрешность δ_2 рассчитывалась относительно аналитического решения для сеточной геометрии.

В данном расчете смешанные ячейки являлись «прозрачными» для ДВ. В силу особенности построения счетной сетки острый угол «смазывается» (рис. 17). Это привело к росту погрешности для первой точки. Если же за «эталон» взять сеточную геометрию, то видим другую картину: новый алгоритм демонстрирует высокую точность (погрешности для сеточной геометрии δ_2 от 10^{-5} до 10^{-7}). Следовательно, можно сделать вывод, что в задачах расчета времен прихода ДВ большую роль играет точность аппроксимации исходной геометрии сеточной.

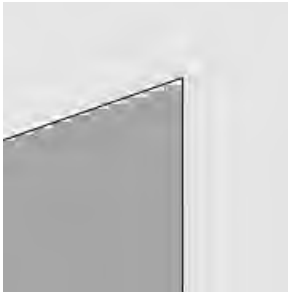


Рис. 17. Геометрия задачи, приближенный вид острого угла

Заключение

Таким образом, в методике ЛЭГАК разработан и реализован новый двумерный алгоритм расчета времен прихода ДВ в узлы счетной сетки, модификация которого описана в данном докладе.

Тестирование алгоритма проводилось на методических задачах, результаты расчетов которых продемонстрировали основные свойства и преимущества нового алгоритма перед стандартным алгоритмом «fast deton». Было показано, что новый алгоритм обладает высокой точностью, независимо от выбора счетной сетки (имеются ввиду ортогональные сетки). Показана сходимость к точному решению при измельчении шага счетной сетки в задаче об огибании волной сферической поверхности, в то время, как алгоритм «fast deton» этой сходимостью не обладает.

Однако стоит отметить, что новый алгоритм проигрывает стандартному алгоритму «fast deton» по времени работы на 20–40 %. Это можно объяснить усложненной логикой анализа выбора точки-источника детонационной волны.

Литература

1. Бахрах С. М., Величко С. В., Спиридонов В. Ф., Авдеев П. А., Артамонов М. В., Бакулина Е. А., Безрукова И. Ю., Борляев В. В., Володина Н. А., Огнева Н. Э., Резвова Т. В., Резяпов А. А., Стародубов С. В., Тарадай И. Ю., Тихонова А. П., Циберев К. В., Шанин А. А., Ширшова М. О., Шувалова Е. В. Методика ЛЭГАК-3D расчета трехмерных нестационарных течений многокомпонентной сплошной среды и принципы ее реализации на многопроцессорных ЭВМ с распределенной памятью // ВАНТ, серия Математическое моделирование физических процессов. 2004. Вып. 4. С. 41–50.
2. Янилкин Ю. В., Беляев С. П., Бондаренко Ю. А. и др. Эйлеравы численные методики ЭГАК и ТРЭК для моделирования многомерных течений многокомпонентной среды. Труды РФЯЦ-ВНИИЭФ. Научно-исследовательское издание, Саров: РФЯЦ-ВНИИЭФ. 2008. Вып. 12. 54–65.