

Параллельный расчет специальных функций волновых катастроф

С.В. Рогачев

НОУ ВПО Российский Новый Университет
105005, г. Москва, ул. Радио, 22. rogachevsergei@gmail.com

Волновая теория катастроф, развитие пространственно-временной теории дифракции и равномерных асимптотических методов позволили решать задачи распространения, дифракции и рассеяния электромагнитных волн вблизи огибающих лучевых семейств и границ «свет-тень». Равномерные асимптотические решения были построены в областях совместной фокусировки основных ГО-лучей, краевых и угловых лучей.

СВК – специальные функции волновых катастроф, вычисление которых требуется при поиске равномерного асимптотического решения. Их аналитический расчет не возможен, а численный, в зависимости от метода, требует значительных временных затрат. Таким образом, эффективное решение задачи расчета СВК возможно только при помощи ЭВМ. Но и использование компьютеров для расчета СВК не гарантирует быстрого получения точного результата. Ряд методов становится малоэффективным при работе с многопараметрическими катастрофами. Ряд нуждается в дополнительной доработке с учетом современных возможностей вычислительных машин.

Актуальной задачей становится создание программной инфраструктуры для эффективного расчета СВК в различных точках или при дискретном изменении параметров, изучения спецфункций в динамике, наглядного графического представления результатов.

Важной характеристикой современных вычислительных программ является возможность использовать многопроцессорные системы или выполнять вычисления параллельно при помощи иных механизмов (вычислительные сети, векторные процессоры, GPU). А реализация старых методов в параллельных системах становится необходимой работой при создании эффективных вычислительных систем.

Материалы данного доклада посвящены, в свою очередь, вопросам вычисления СВК различных особенностей в параллельных системах и конкретному программному комплексу, воплощающему эти идеи.

The wave catastrophe theory, the development of spatiotemporal theory of diffraction and uniform asymptotic methods allowed to resolve the problems of propagation, diffraction and dispersion of electromagnetic waves nearly caustics and the edges “light-shadow”. Uniform asymptotic solutions were made in the fields of joint focus of GO-, edge and corner rays.

SWC is an acronym for special functions of wave catastrophes, calculation of which is necessary during building of uniform asymptotic solution. Their analytic calculation is impossible and the numerical one, depending on the applied method, requires huge time expenses. Thus, effective solution of SWC calculation problem lies in the field of computer computation.

But usage of a computer for SWC calculation is not a guarantee of quick and exact resolution of the problem. A set of methods used for numeric computation of SWC is not so efficient (and non-scalable) in the case of multi-parametric catastrophes. Some of them lacks the support of modern SMP systems and need additional work for adaptation to use the algorithms in parallel environment.

The important task is a development of a program infrastructure for efficient computation of SWC in various points or with discrete changing of parameters in some range, for visualization the solutions in the form of contour plots, etc.

All modern computational software should be oriented on systems with sets of CPUs (CPU cores), or some other kind of parallel hardware including vector coprocessors and GPUs; implementation of old sequential algorithms in the new parallel environment is an important field of activity.

The materials of the report are dedicated to the question of SWC computation using parallel-ready hardware.

Введение

Равномерные асимптотические методы, применяемые в рамках волновой теории катастроф, позволяют находить решения ряда задач о распространении, дифракции и рассеянии электромагнитных волн вблизи каустических поверхностей (огibaющих лучевых семейств) и границ «свет-тьнь». Применение данных асимптотических методов сопряжено с вычислением коэффициентов асимптотического разложения, спецфункций и ряда их первых производных [1-6].

Вопросы вычисления коэффициентов и специальных функций в таких задачах довольно сложны и требуют отдельного изучения. В рамках данной работы рассматривается комплекс программ, созданный специально для расчета специальных функций волновых катастроф (СВК).

Традиционно СВК определяются в виде многократных осциллирующих интегралов, которые невозможно взять аналитически и существует набор численных методов, позволяющих рассчитывать эти спецфункции.

Среди численных методов – метод рядов Тейлора, контурный метод и метод неравномерных асимптотик. Перечисленные методы известны рядом проблем: их сложность значительно вырастает с ростом сложности особенности (увеличением количества параметров) [7].

В качестве ответа на эти проблемы появился метод обыкновенных дифференциальных уравнений. Метод базировался на ряде подходов, использующих систему канонических дифференциальных уравнений в частных производных. В отличие от многих из них, метод ОДУ концентрируется на выводе системы ОДУ и использование именно обыкновенных дифференциальных уравнений считается его основным и неоспоримым достоинством [7].

Принято считать, что, если некоторая задача, приводящая к необходимости решения дифференциальных уравнений, может быть сведена к системе ОДУ – она является фактически решенной. Отчасти это так, поскольку существует множество известных методов для интегрирования ОДУ и систем ОДУ, начиная с простейших, вроде метода Эйлера и заканчивая сложными методами высоких порядков точности с автоматической корректировкой шага и оценкой локальных погрешностей.

Было замечено, что метод ОДУ отлично подходит для решения задачи, поскольку, в силу ее характера, можно производить интегрирование системы ОДУ в разных точках независимо (в отдельных процессах, потоках, и т.д.). Этот факт был использован при создании программного комплекса для вычисления СВК – естественного развития подходов к вычислениям необходимых спецфункций.

Описание комплекса программ

Идея программного средства для расчета СВК не нова [8-11], но, тем не менее, до сих пор не было создано сколько-нибудь готовой системы, позволяющей интегрировать СВК в необходимых точках или на заданных интервалах.

Таким образом, было решено создать комплекс программ, который позволил бы производить численный расчет СВК путем интегрирования соответствующих им систем ОДУ, делать это наиболее эффективно и легко адаптироваться для нужд исследователей [12-15].

Полученная система представляет собой небольшой комплекс программ, состоящий из двух основных компонент: вычислительного ядра, написанного на языке Си, сочетающего в себе вычислительную инфраструктуру и модули для каждой из СВК, и пользовательского интерфейса, исполняемого в браузере и созданного при помощи набора WEB-языков (html5, css, javascript). Связующим звеном между

вычислительным ядром программы и интерфейсом является простой и быстрый открытый WEB-сервер mongoose.

Инфраструктура вычислительного ядра построена таким образом, чтобы упростить расширение программы новыми модулями: список СВК, доступных для вычисления, может быть расширен путем добавления новых модулей. При этом предусмотрено две возможности: разработка модулей на языке Си и их создание при помощи специального предметно-ориентированного языка, напоминающего Pascal или Modula [16]. В обоих случаях, на заключительном этапе, результат представляется на языке Си и равноценен с точки зрения производительности.

Результаты вычисления представляются интерфейсной частью в виде графика линий равного уровня, готового к публикации – графики могут быть свободно сохранены в файлы изображений PNG и строятся только в градациях серого, подходя, соответственно, для черно-белой печати.

Параллельное интегрирование в вычислительном ядре

Вычислительное ядро поддерживает параллельное интегрирование системы ОДУ: в том случае, если при сборке программы была включена опция CONFIG_PARALLEL_COMP, ядро будет производить декомпозицию задачи на ряд подзадач и выполнять их в отдельных потоках.

Поскольку при параллелизме по данным не требуется выполнять промежуточных синхронизаций, вычисления происходят с максимально возможной скоростью.

Данные, представляющие частные результаты вычислений, объединяются так, чтобы результат не отличался от того, что может быть получен при последовательном решении задачи.

Для оценки производительности параллельного решения удобно пользоваться следующей формулой:

$$T' = \sum_{i=1}^N \sum_{k=1}^3 c_{k,i} + \frac{T}{N} \quad (1)$$

Константы $c_{k,j}$ – постоянные временные задержки на создание потоков, включая передачу начальных данных, сбор результатов вычисления и их композицию, соответственно. T – время работы последовательного алгоритма, N – число независимых вычислительных устройств.

Это приближение времени выполнения характерно как для систем с общей памятью (многопроцессорные и многоядерные ЭВМ, традиционные рабочие станции), так и для систем, взаимодействующих при помощи механизма сообщений (кластеры, MPP системы).

Такая простая оценка может применяться лишь для случая с равноценными по своим характеристикам вычислительными устройствами, для НМР и других видов несимметричных гетерогенных систем оценка усложняется.

Рядом констант можно пренебречь в случае систем с общей памятью, в то же время для систем с отдельной памятью эти временные задержки становятся значительными. Таким образом, в некоторых случаях выполнение параллельного интегрирование системы может быть нецелесообразным в случае использования кластера.

Рассмотрим время расчета СВК особенности $K_{4,2}$ последовательным и параллельным методом.

Таблица 1. Сравнение времени параллельного и последовательного интегрирования СВК К_{4,2}

№	Время (сек.)	Время пар. (сек.)
1	6,830	2,392
2	6,664	2,376
3	6,929	2,407
4	6,608	2,400
5	6,711	2,371
Среднее	6,748	2,389

Приложение может использоваться как в качестве «настольного», так и в виде WEB-сервиса. При особых подходах к конфигурации (требуется разработка ряда дополнительных модулей) возможно развертывание облачного сервиса с несколькими вычислительными узлами и балансировщиком нагрузки.

Вычисления производились на компьютере с процессором Intel i3-3120 с тактовой частотой 2.50 GHz. В процессоре доступно четыре виртуальных ядра, то есть присутствуют два физических ядра с раздвоенными наборами регистров и локальными контроллерами прерываний (LAPIC). Таким образом, можно предсказывать как минимум удвоенную производительность за счет двух физических вычислительных устройств, и примерно 30% улучшение производительности за счет технологии hyper threading.

Реальное улучшение в среднем достигло 2.82 раз, что полностью соответствует оценке при помощи формулы (1).

Формирование подзадач при параллельном интегрировании

Для вычисления СВК вычислительное ядро решает задачу численного интегрирования системы ОДУ, при этом интегрирование выполняется множество раз в различных точках. Каждой точке соответствует свое сочетание параметров из некоторого множества $\{\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{n-1}, \lambda_n, a\}$.

Для интегрирования используются традиционные численные методы, которые по большей части являются последовательными и плохо поддаются распараллеливанию.

Но, в то же время, интегрирование в различных точках может выполняться независимо. Так, теоретически, наименьшая единица, которую можно выполнять в отдельном вычислительном контексте – интегрирование системы для какого-то одного сочетания параметров (в одной точке). Естественно, такой подход не эффективен – требуется декомпозиция всей задачи на несколько более мелких, но при этом количество подзадач должно оставаться небольшим.

Рассмотрим вопрос декомпозиции для какого-то одного параметра λ_m .

Пусть интегрирование требуется выполнить на отрезке $[b, a]$, s – количество шагов, k – величина шага, то есть эти величины связаны тождеством:

$$k = \frac{b - a}{s} \quad (2)$$

Задача декомпозиции формулируется следующим образом: требуется получить границы изменения параметра ($[a', b']$) для n подзадач, то есть для количества шагов s' , определяемого как:

$$s' = \frac{s}{n} \quad (3)$$

Для декомпозиции задачи по какому-либо из параметров применяется формула 4.

$$b' = ks' + a' \quad (4)$$

Формула может быть применена в следующем итеративном алгоритме:

Исходные параметры: a – левая граница, b – правая граница, s – количество шагов.

Результат: $T[n]$ – множество верхних границ

$$a' = a;$$

для каждого $i \in [0, n)$ выполнять

$$b' = ks' + a';$$

$$T[i] = b';$$

$$a' = b';$$

конец цикла

Алг. 1. Вычисление верхних границ изменения параметра при формировании подзадач

Пользовательский интерфейс и взаимосвязь компонент

Построение амплитудных и фазовых структур СВК происходит при помощи алгоритма, основанного на методе «марширующие квадраты». Алгоритм строит графики линий равного уровня с закрашиванием сечений. Для изображения аппликаты на графике применяется цветовое кодирование. Для цветового кодирования используется палитра в оттенках серого, генерируемая в зависимости от числа секущих плоскостей. Пользователь управляет числом секущих плоскостей, что позволяет просматривать более или менее точные изображения. Возможно включение и отключение контуров, управление количеством шагов и экспорт графиков (рис. 1).

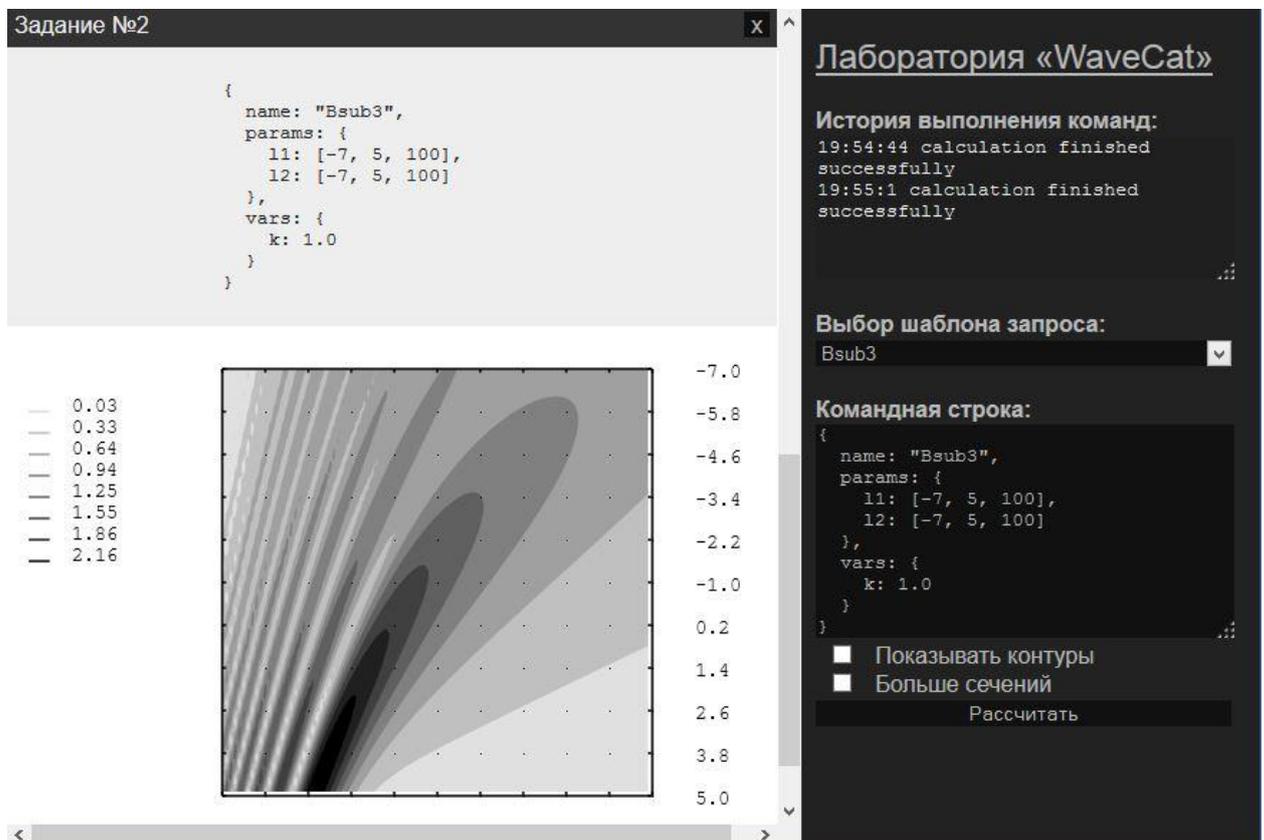


Рис.1. Вид графического интерфейса пользователя

Вся история вычислений сохраняется: каждый раз, выполнив задание, система выводит информацию об этом задании в рабочей области и строит соответствующий график. Поля заданий визуально разделены. Информацию о ненужных более результатах легко скрыть.

Для общения между компонентами системы используются текстовые объекты в формате JSON. Так представляются, например, задания, получаемые вычислительным ядром, а также результаты вычислений, отправляемые графическому интерфейсу для построения графика. Важно заметить, что модуль анализа языка запросов в вычислительном ядре позволяет принимать даже те задания, в описании которых были допущены некоторые синтаксические ошибки. Это допущение сделано с целью упрощения взаимодействия с программой.

Пользователю предоставляется практически полный контроль над процедурой формирования задания: возможно управлять границами изменения параметров, шагом изменения каждого из них, значениями переменных, режимом вычисления. Программа должна прощать незначительные ошибки при описании задания, тем самым поощряя активные эксперименты при вычислениях.

```
{
  name: "Asub1sup4"
  params {
    I1 : [-10, 10, 100]
    I2 : [-15, 5, 100]
    a : 0
  }
  vars {
    k1: -1.0
    k2 1.0
  }
}
```

Прог. 1. Пример объекта задания с явными синтаксическими ошибками

В примере (Прог. 1) приведен код JSON объекта, содержащего описание задания на интегрирование СВК особенности A_1^4 : отсутствуют символы «;» между описаниями полей, имена полей не отделены от своих значений символом «:», форматирование также может быть нарушено. Не смотря на это, вычислительное ядро верно распознает все элементы задания и произведет вычисления.

Благодаря формату взаимодействия компонентов системы удается добиться высокой степени гибкости. Например, путем небольших изменений возможно преобразовать программный комплекс в облачный сервис (Рис. 2). Возможны два подхода для достижения этой цели.

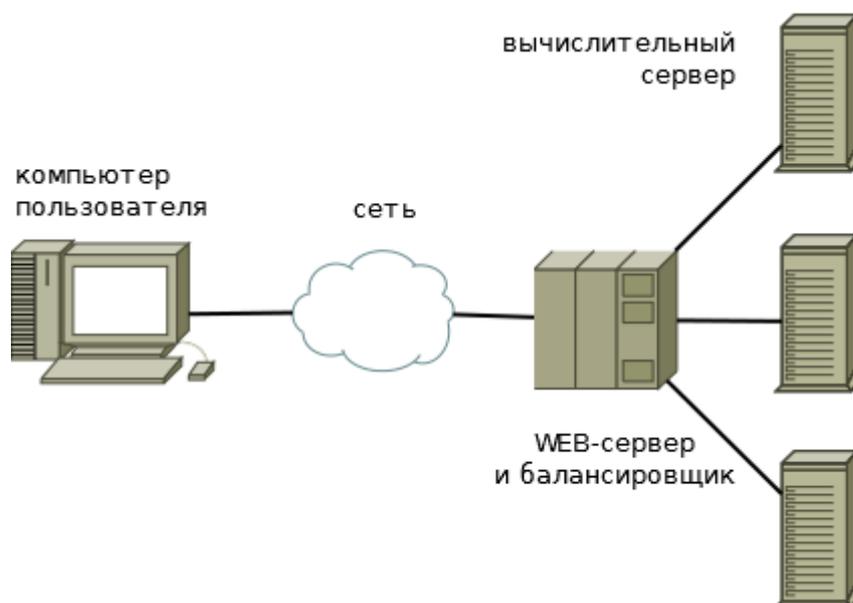


Рис. 2. Архитектура вычислительного облака

Первый подразумевает модификацию части, отвечающий за интерфейс пользователя, и является наиболее простым. Вместо отправки запроса на один и тот же вычислительный сервер программа может производить начальную декомпозицию задачи и отправлять подзадачи в виде JSON объектов на несколько доступных серверов. Их список можно динамически обновлять, как и информацию о загруженности каждого. Результаты вычислений требуется объединить и построить график решения, что не требует серьезных вычислений и может быть выполнено на компьютере пользователя.

Второй путь несколько более сложный, но позволяет добиться большей инкапсуляции и изоляции системы. Он подразумевает введение специально сконфигурированного сервера, вычислительного ядра, которое, получив задачу, производит ее декомпозицию и рассылает подзадачи, упаковав их в JSON объекты другим серверам. Должны учитываться их загруженность и производительность.

Примеры расчета СВК при помощи вычислительного комплекса

Рассмотрим ряд построений для СВК особенности A_1^4 , пропорциональной главному члену асимптотики решения задачи о дифракции волны на плоском экране с угловым вырезом.

$$V_1^4(\lambda_1, \lambda_2, a) = \int_0^{+\infty} \int_0^{+\infty} \exp(i(k_1 z^2 + azy + k_2 y^2 + \lambda_1 y + \lambda_2 z)) dy dz \quad (5)$$

Вид системы ОДУ для этой особенности оставим за рамками статьи.

На рисунке 3 изображен модуль СВК, характерный для случая прямоугольного выреза в экране.

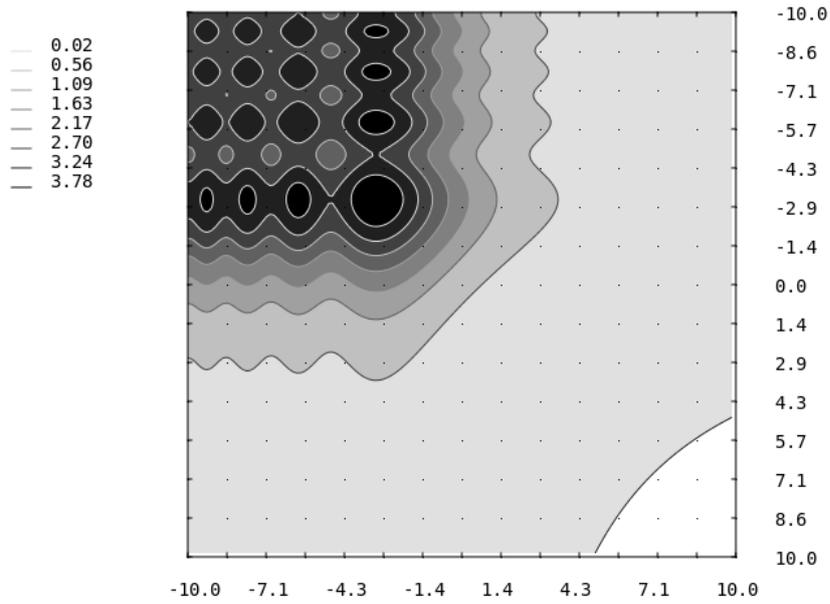


Рис. 3. Модуль СВК особенности $A_1^4(a=0, k_1=k_2=1)$

Рисунок 4 содержит построение фазовой структуры упомянутой выше СВК.

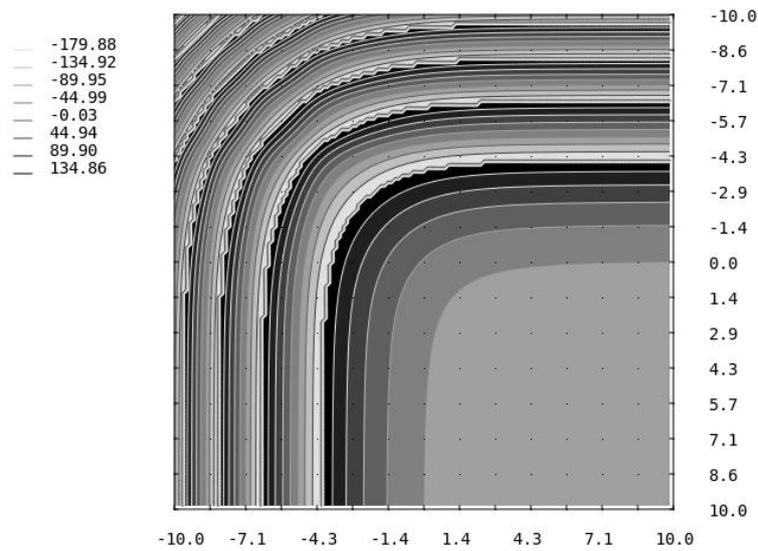


Рис. 4. Фаза СВК особенности $A_1^4(a=0, k_1=k_2=1)$

Построим модуль СВК для случая острого угла выреза (45 градусов). Наблюдается удаление главного максимума от угловой точки (рис. 5).

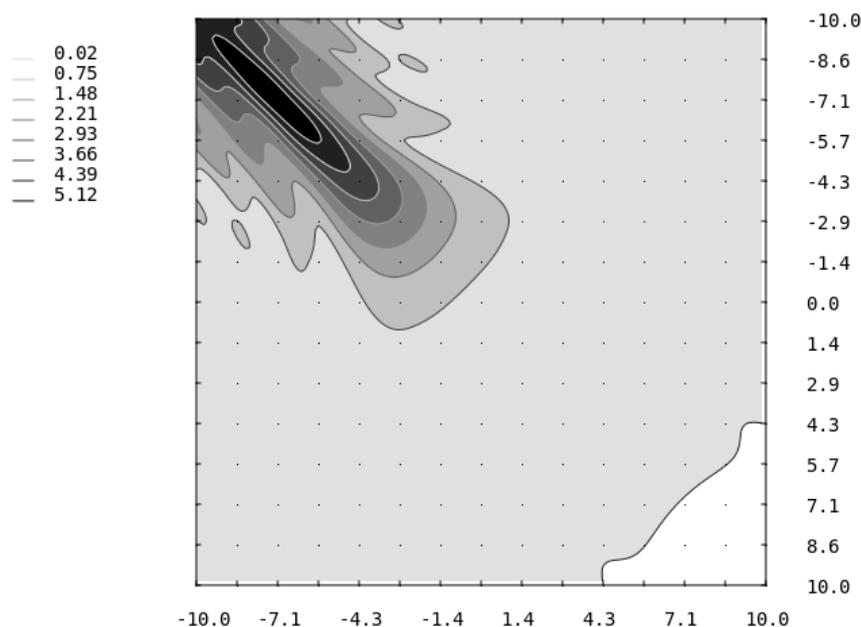


Рис. 5. Модуль СВК особенности $A_1^4 (a = \sqrt{2}, k_1 = k_2 = 1)$

Заключение

Рассмотренный комплекс программ еще далек от совершенства, хотя и продолжает инкрементально развиваться. Производительность системы в целом, как и отдельных вычислительных модулей, постоянно улучшается за счет макро и микро-оптимизаций. Список поддерживаемых спецфункций постепенно расширяется. На данный момент поддерживаются СВК следующих особенностей: A_3 , B_3 , C_4 , E_6 , P_8 , F_4 , $K_{4,2}$, A_1^4 . Разрабатываются модули для особенностей D_4^\pm , $A_1A_2A_1A_1$ и других.

Можно заключить, что сформировалась стабильная программная инфраструктура, которая может наращиваться и улучшаться в дальнейшем без ощутимых архитектурных изменений. Каждое новое изменение хорошо ложится в общую концепцию системы.

Направление облачных вычислений пока не покрыто и ему обязательно будет уделено внимание в дальнейшем, особенно благодаря тому факту, что преобразование системы в облачный сервис на данном этапе кажется достаточно легким шагом.

Литература

1. Лукин Д.С., Палкин Е.А. Численный канонический метод в задачах дифракции и распространения электромагнитных волн в неоднородных средах. М.: МФТИ, 1982. 159 с.
2. Крюковский А. С., Лукин Д. С., Палкин Е. А. Применение теории краевых катастроф для построения равномерных асимптотик быстроосциллирующих интегралов // Дифракция и распространение волн. Междувед. сборник / МФТИ. М., 1985. С. 4 - 21.
3. Kryukovsky A. S., Lukin D. S., Palkin E. A. Uniform asymptotics for evaluating oscillatory edge integrals by methods of catastrophe theory // Soviet journal of Numerical Analysis and Mathematical Modelling. 1987. V. 2. № 4. P. 219 - 312.
4. Крюковский А.С. Равномерная асимптотическая теория краевых и угловых волновых катастроф М.: РосНОУ, 2013. 368 с.

5. Крюковский А.С., Лукин Д.С., Растягаев Д.В. Классификация и равномерное асимптотическое описание пространственно-временных трехмерных краевых фокусировок волновых полей. // Радиотехника и электроника, 2005. Т.50. №10. С. 1221-1230.
6. Крюковский А.С., Лукин Д.С. Теория расчета эталонных фокальных и дифракционных электромагнитных полей на основе специальных функций волновых катастроф. // Радиотехника и электроника, 2003. Т.48. №8. С. 912-921.
7. Крюковский А. С. Метод обыкновенных дифференциальных уравнений для расчета специальных функций волновых катастроф (СВК) // Дифракция и распространение электромагнитных волн: Межведомственный сборник научных трудов – М.: Московский физико-технический институт, 1992 – С. 29 — 48.
8. Дорохина Т. В., Крюковский А. С., Малышенко А. Б. Разработка численных алгоритмов расчета и визуализации волновых катастроф // Вестник РосНОУ. Управление вычислительная техника и информатика. – М.: Российский новый университет, 2008. – Вып. 3. – С. 25 – 47.
9. Дорохина Т.В., Крюковский А.С., Лукин Д.С. Информационная система «Волновые катастрофы в радиофизике, акустике и квантовой механике». // Электромагнитные волны и электронные системы. 2007. Т.12. № 8. С.71-75.
10. Дорохина Т.В., Крюковский А.С., Лукин Д.С., Волкова Е.В., Костьо А.О., Павлова М.В. Создание информационной системы волновой теории катастроф и её применение при математическом моделировании. // Вестник Российского нового университета. 2007. Выпуск 2. / М.: РосНОУ, 2007. С. 91-107.
11. Дорохина Т.В., Ипатов Е.Б., Крюковский А.С., Лукин Д.С., Палкин Е.А., Растягаев Д.В. Математическое компьютерное моделирование волновых полей типа катастроф. // Распространение радиоволн: сборник докладов XXI Всероссийской научной конференции. Йошкар-Ола, 25-27 мая 2005 г. / Йошкар-Ола: МарГТУ, 2005. Т.2. С. 336-339.
12. Крюковский А.С., Рогачев С.В. Система расчета и визуализации специальных функций волновых катастроф // Электромагнитные волны и электронные системы. 2013. Т. 18. № 8. С. 10-17.
13. Крюковский А.С., Рогачев С.В. Разработка специализированной системы расчета специальных функций волновых катастроф. // Всероссийские Армандовские чтения [Электронный ресурс]: Сверхширокополосные сигналы в радиолокации, связи и акустике / Материалы Всероссийской научной конференции (Муром, 25-27 июня 2013 г.) — Муром: Изд.- полиграфический центр МИ ВлГУ, 2013. — 271с. ISSN 2304-0297 (CD-ROM). С. 129-144.
14. Крюковский А.С., Рогачев С.В. Специализированная система расчета и визуализации специальных функций волновых катастроф. // Труды международной научной конференции “Цивилизация знаний: проблемы и смыслы образования”, Москва, 26-27 апреля 2013 г. / М.: РосНОУ, 2013. Часть 2. — С. 117-174.
15. Andrew S. Kryukovsky, Sergey V. Rogachev, Dmitry S. Lukin. Special Software for Computing the Special Functions of Wave Catastrophes. // Revista de Matemática: Teoría y Aplicaciones / San Pedro Montes de Oca, San José, Costa Rica: Universidad de Costa Rica, 2015. — С. 21-30.
16. Рогачев С.В. Проектирование предметно-ориентированного языка для расчета специальных функций волновых катастроф. // Вестник Российского нового университета. Серия «Управление, вычислительная техника и информатика» / М.: РосНОУ, 2013. Выпуск 4. – С. 47–52.