УДК 002.53:002.55:533.6:681.3:681.6

# МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ: ОСНОВНЫЕ СЕГМЕНТЫ, ИХ ОСОБЕННОСТИ И ПРОБЛЕМЫ

**Г. А. Тарнавский[1], А. В. Алиев[1]**

Рассмотрена общая схема математического моделирования в области естественных наук как единая детерминированная последовательность этапов со строго определенным функциональным назначением. Проанализированы проблемы, возникающие на каждом этапе при создании теоретических методов, вычислительных алгоритмов и компьютерных программ, а также при их применении для фундаментальных научных исследований и прикладных разработок.

**Ключевые слова:** параллельные вычисления, математическое моделирование, вычислительный эксперимент, численные методы, автоматизация научных исследований.

**1. Введение.** Исключительно быстрое развитие компьютерной техники и распространение информационно-вычислительных технологий привело к их широкому использованию не только, как раньше, в области фундаментальных научных исследований и научно-технических разработок, но и для многочисленных прикладных проектов. Высокая востребованность "готового" компьютерного инструментария вызвала быструю разработку и появление на рынке потребительских пакетов и комплексов прикладных компьютерных программ для решения задач в широком спектре научных знаний с декларированием весьма значительных возможностей, в том числе параллельного счета. Как справедливо отмечено в [1], в связи с наличием таких пакетов создается иллюзия фактической завершенности создания теоретических методов, вычислительных алгоритмов и компьютерных программ в некоторых областях знаний. Так, например, в области аэрогидродинамики и физической газовой динамики программные комплексы Ansys CFX, Fluent, FlowVision, Comsol и некоторые другие, конкурирующие между собой, декларируют возможность качественного расчета обтекания объектов от автомобилей и кораблей до ракет и самолетов сложной формы, в том числе при высокоскоростном полете, с необходимостью учета изменения физических свойств газовой среды, газодинамики течения в тоннелях, шахтах и т.п., газодинамики различных сооружений и турбин и т.д. [2, 3]. Однако нельзя объять необъятное, и в целом к подобных декларациям следует подходить с осторожностью и разумным скептицизмом, как к элементу маркетинговой политики.

В связи с этим представляется необходимым и, на наш взгляд, весьма полезным, в особенности для прикладных пользователей вычислительного инструментария, не связанных напрямую с разработкой, и для не имеющих соответствующего опыта специалистов, дать некоторый общий обзор концепции математического моделирования научных задач с едиными для любой области знания позициями и чертами и строго детерминированной последовательностью ключевых этапов и провести анализ возникающих главных проблем.

**2. Основные сегменты математического моделирования: общий обзор.** Общая гносеологическая схема математического моделирования любой физической (химической, биологической и т.п.) задачи заключается в декомпозиции всего процесса исследования на ряд сегментов, последовательных этапов продвижения исследования от стартового состояния к финальному результату. Можно выделить десять основных этапов (рис. 1). Подчеркнем, что развитие высокопроизводительных мультипроцессорных систем и применение технологий параллельного счета вносит существенные изменения [4, 5] не только в сегменты математического моделирования, но и порождает ряд новых, специфических проблем и в самой методологии, и в информационно-телекоммуникационных аспектах получения, обработки, хранения и передачи данных сверхбольшого объема. Эти проблемы будут также рассмотрены в настоящей работе.

**3. Реальное физическое явление.** Уровень понимания реального физического явления фактически детерминирует весь дальнейший процесс математического моделирования этого явления. Знание ключевых особенностей физического явления (предыстории, динамики развития, возможности прогнозирования реакций на различные возмущения и т.п.) позволяет использовать адекватные реальности физико-математические модели, конструировать вычислительные алгоритмы, создавать компьютерные программы и проводить анализ полученных решений. Это совершенно не означает, что не следует начинать математическое моделирование какого-либо явления при отсутствии "достаточных знаний" о нем.

[1] Институт вычислительной математики и математической геофизики СО РАН, просп. Лаврентьева, 6, 630090, Новосибирск; e-mail: gennady@tarnavsky.ru, aliev@ssd.sscc.ru

© Научно-исследовательский вычислительный центр МГУ им. М. В. Ломоносова


Рис. 1. Основные этапы математического моделирования: реальный физический процесс — физическая модель — математическая модель — численный метод — вычислительный алгоритм — компьютерная программа — анализ полученной информации — принятие решения — завершение исследований

Приближение к “точному” знанию, как правило, является итерационным процессом всей гносеологической цепочки этапов математического моделирования (рис. 1). Например, моделирование астрофизических явлений практически за всю историю науки проводится в условиях недостаточного знания о ре-

Рис. 2. Спиральная галактика M51 "Whirlpool" (NGC 5194) и ее галактика-спутник (справа). Размер — 90 тыс. световых лет, расстояние до Земли — 31 млн. световых лет, масса — 160 млрд. солнечных масс. Фотография с сайта Hubblesite.org

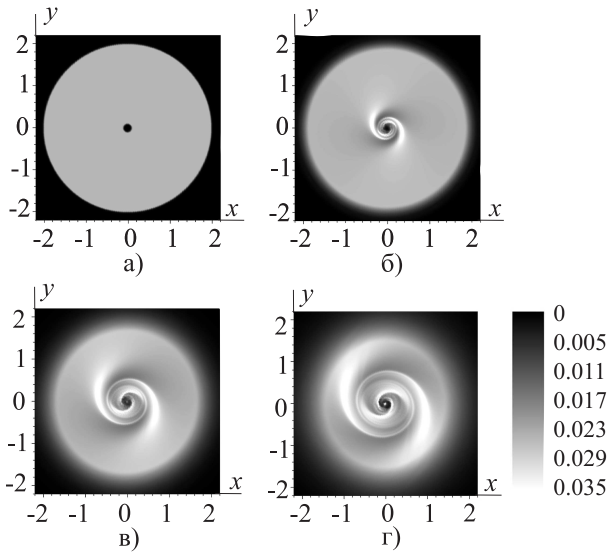


Рис. 3. Эволюция газового диска с формированием спиралей. Распределение плотности в различные моменты времени а) $t = 0$; б) 0.3; в) 0.6; г) 0.9. Масштаб времени $1:80$ млн. лет, геометрический масштаб $1:20$ тыс. световых лет

альном физическом явлении, но само стимулирует развитие этого знания и создает все более адекватные (в определенном смысле) физические модели явления от Геоцентрической системы до Большого взрыва. Так, на рис. 2 показано реальное физическое явление: спиральная галактика M51 Whirlpool (NGC5194), изображение которой получено с помощью орбитального телескопа "Хаббл". Естественно, что невозможно достоверно узнать историю происхождения этой звездной системы, однако современные знания позволяют проводить в некотором приближении моделирование возможной динамики ее развития (рис. 3, подробный анализ проведен в [6]).

**4. Второй этап математического моделирования: физическая модель.** Построение физической модели реального явления — важный сегмент математического моделирования, принципиально определяющий все дальнейшие этапы. Создание физической модели заключается в разделении, на основе знаний о реальном физическом явлении, общего процесса на главные, второстепенные и несущественные субпроцессы, а также в определении доминирующих факторов при вариации параметров в диапазонах их возможного изменения. Подчеркнем, что, после проведения всей цепочки математического моделирования, возможно, потребуется пересмотр, иногда даже кардинальный, физической модели. Считавшиеся второстепенными процессы окажутся равнозначными главным, выявятся новые процессы, доминирующие лишь в ограниченном диапазоне определяющих параметров. Весьма оптимальным оказывается создание многоуровневой физической модели.

Поясним сказанное на примере моделирования полета какого-либо объекта в атмосфере. Если достаточно лишь приближенного знания о траектории движения, то возможно использование физических моделей, которые зачастую даже не требуют компьютерных расчетов, а сводятся к аналитическим формулам [7]. Модели более высокого уровня требуют учета все новых и новых физических факторов с последовательным наращиванием сложности: сжимаемости, вязкости, теплопроводности среды, подключение модулей турбулентности [8], и далее, при необходимости точного расчета обтекания аппарата при высокоскоростном полете, учета изменения физических свойств газа, таких как возбуждение колебаний в молекулах, диссоциация, химические реакции, возбуждение электронных оболочек атомов, ионизация и т.д. [9].

На рис. 4 показаны картины обтекания (поля температур) объекта каплеобразной формы, движущегося в атмосфере Земли на высоте 40 км со скоростью 5 км/с. Для компьютерных расчетов использовалась модель сжимаемого вязкого теплопроводного газа для двух подмоделей: без учета изменения физических свойств среды (рис. 4 а) и с учетом их изменения [10] вследствие наличия области высоких температур в потоке (рис. 4 б). В этих двух подмоделях общие структуры обтекания достаточно адекватны, однако необходимо обратить внимание на следующее. Силовые характеристики (коэффициент сопротивления, давление на поверхности) отличаются незначительно, до 5 %. Гораздо более существенно отличаются тепловые характеристики (температурные нагрузки на поверхности) и в особенности треки ближнего следа, образующиеся сходящими с тела температурными пограничными слоями (вязкостные пограничные слои отличаются менее значительно). Таким образом, оптимально выбираемый уровень моделирования существенно зависит от его целей: не обязательно использовать модель высокого уровня, иногда существенно более затратную по компьютерным ресурсам, если в этом нет практической необходимости. Однако гораздо менее приемлемой является ситуация, зачастую встречающаяся в маркетинговой политике распространения пакетов компьютерных программ, ориентированных на массового пользователя, когда декларируются широкие возможности в той или иной прикладной области, а их физические модели не обладают соответствующей полнотой.

Развитие параллельных технологий высокопроизводительных вычислений на суперкомпьютерах привело к существенному изменению практической реализации этого этапа математического моделирования. Одним из типов декомпозиции задачи на ряд одновременно выполняемых подзадач является сегментация по физическим процессам, опирающаяся на философию рассматриваемой позиции “физическая модель” гносеологической последовательности (рис. 1) математического моделирования. На основе этой сегментации может быть проведено распараллеливание вычислительных операций — распределение блоков программного кода, соответствующих различным физическим процессам для индивидуального исполнения на отдельных машинах или процессорных элементах.

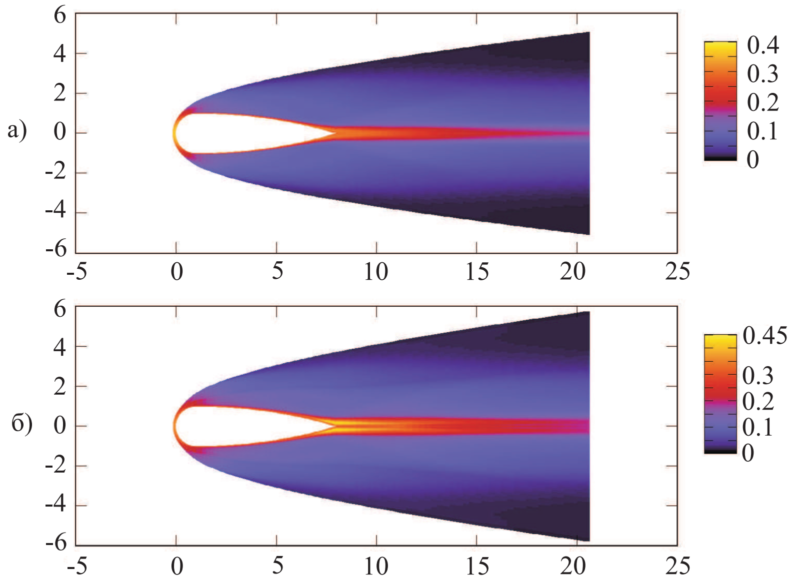


Рис. 4. Картина гиперзвукового обтекания капсулы каплевидной формы при полете на высоте 40 км со скоростью 5 км/с: а) без учета и б) с учетом изменения свойств среды

На рис. 5 показаны последовательная и параллельные схемы вычислений при решении задач гравитационной газовой динамики [5]. На схемах приведены главные моделируемые физические процессы, составляющие пять основные блоков программного кода: газовая динамика (1ГД), термодинамика (2ТД), излучение и другие тепловые процессы (3ТП), гравитационное взаимодействие (4ГР) и метрика пространства (5МП). На однопроцессорных системах эти блоки, естественно, выполняются последовательно (показано стрелками) в итерационном цикле моделирования эволюции астрофизической системы. На мультипроцессорных комплексах возможно существенное сокращение времени решения всей задачи, особенно нестационарной, вследствие одновременного решения некоторых подзадач с обменом необходимой информацией между программными модулями на каждом шаге итерационного цикла (направления потоков информации помечены стрелками с нумерацией двузначным числом: первая цифра соответствует модулю-отправителю, вторая — получателю). При этом в связи с различным временем исполнения модулей возникает проблема оптимального использования мультипроцессорной системы с минимизацией простоев отдельных машин. Эта проблема решается путем дальнейшей, более глубинной параллелизации вычислений в сегментах физической модели с динамической реконфигурацией общего процессорного пространства при решении задач. Алгоритмика распараллеливания в коммерческих пакетах в настоящее время основана на гораздо более простых принципах.

Уровень физической модели реального физического явления будет определять всю дальнейшую цепочку моделирования и влиять на ее результаты. Так, при изучении грозовых явлений принятие физической модели атмосферного электричества даст одни результаты, а модели “Ильи-пророка” — совершенно другие. Модель идеального газа не позволяет получить в расчетах и исследовать пограничные слои вблизи поверхности летательного аппарата. Модель негравитирующего газопылевого облака не позволит изучить коллапс межгалактических туманностей и образование звезд. Подводя итог, можно сказать, что физическая модель — это абстракция, содержащая в себе только те сущности, которые оказывают значимое

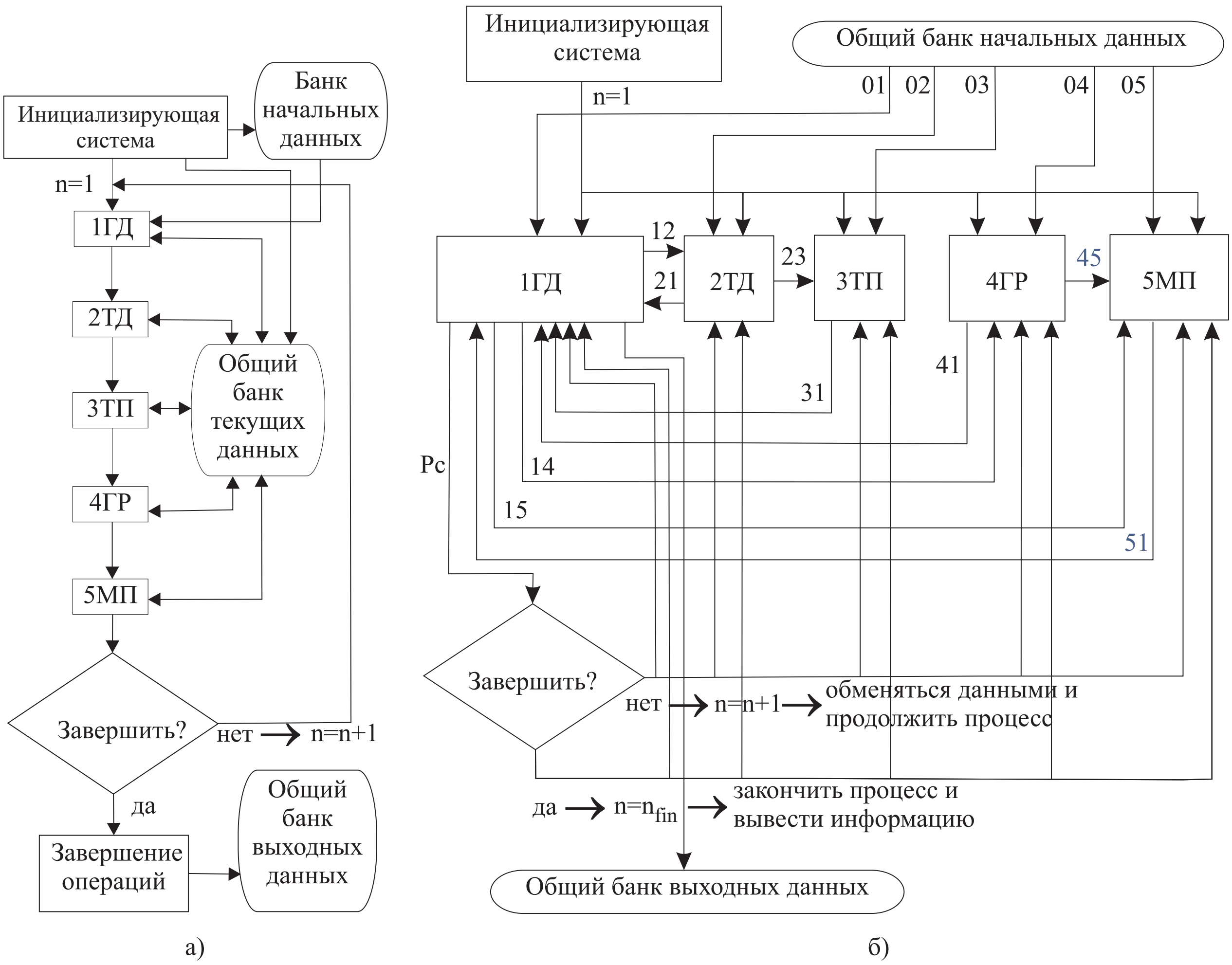


Рис. 5. Схема последовательного (а) и параллельного (б) счета

воздействие на наблюдаемую действительность. Критерий значимости, как правило, заранее известен исследователю и детерминирует класс исследуемых явлений. Физическая модель является адекватной тогда, когда результаты, полученные с ее помощью, удовлетворяют нуждам и задачам исследователя.

**5. Третий сегмент математического моделирования: математическая модель.** В соответствии с выбранной (построенной, принятой как гипотеза, декларированной и т.д.) физической моделью формируется математическая модель реального физического явления. Этот этап иногда определяют как главный в гносеологической последовательности исследования (рис. 1). Даже название данного сегмента почти совпадает с общим названием, хотя этимология этих двух терминов и их значения совершенно различны: они соотносятся (в философском смысле) как “частное” и “общее”. Вообще говоря, общий термин “математическое моделирование” является несколько устаревшим. Более точным в настоящее время следует признать термин “компьютерное моделирование” как некоторую методологию генерации знаний, входящую в область “Computer Science”. Разумеется, термин “математическое” несколько шире “компьютерного”, поскольку включает в себя и аналитические, и инженерные (например, на логарифмической линейке) методы решения конкретных задач.

В данном сегменте моделирования проводится математическая формализация описания каждого физического процесса, входящего в общую физическую модель. Осуществляется математическая постановка задачи: выписывается система уравнений (алгебраическая, дифференциальная, в том числе в частных производных, интегро-дифференциальная) с размерностью, соответствующей уровню модели физического процесса. Проводится постановка граничных условий и, для нестационарных задач, начальных данных. До перехода к следующему этапу моделирования — конструированию вычислительного алгоритма — необходим теоретический анализ математической модели: исследование корректности постановки, анализ возможной неединственности решения, точек бифуркаций [11] и т.п. Такой теоретический анализ позволит разрабатывать качественные высокоразрешающие численные методы и прогнозировать особенности решения в некоторых диапазонах определяющих параметров [12]. Однако в большом числе случаев не удается сформулировать корректную математическую постановку задачи вследствие недостатка информации о реальном физическом явлении. Компьютерное моделирование некорректных задач с неполной информацией является весьма трудным, но и наиболее интересным делом, находящимся в авангарде генерации научных знаний.

**6. Четвертый этап математического моделирования: создание вычислительного алгоритма.** На основании принятых физико-математических моделей реального физического явления выполняется следующий, четвертый сегмент математического моделирования — создание вычислительного алгоритма, включающее в себя его проектирование, конструирование, разработку и анализ свойств. В настоящее время этот этап существенным образом должен быть ориентирован на тип компьютерных систем, на которых предполагается базировать вычислительный комплекс.

Современные функционирующие и, в особенности, перспективные суперкомпьютеры предоставляют колоссальные возможности конструирования высокоразрешающих алгоритмов на основе проведенного уже на стадии проектирования распараллеливания вычислительных операций в алгоритме. Это распараллеливание может интегрированно включать в себя различные типы параллелизации алгоритма: по физическим процессам (см. раздел 4), геометрическую декомпозицию (наиболее распространенную) расчетной области на ряд подобластей (domain decomposition), технологическую декомпозицию [4, 13–15]. Комплексное распараллеливание алгоритмов создает большие возможности для разработки следующего этапа математического моделирования — компьютерных программ параллельного счета высокой эффективности. Эта эффективность может быть обеспечена оперированием с процессорными элементами, в частности, с их группированием и созданием специальных топологий связей между ними [16]. Разумеется, при использовании значительного числа процессоров возникают дополнительные проблемы, связанные с большими потоками информации. Кроме того, следует предусматривать безопасность выполнения всего задания в целом, поскольку нарушение работы (выход из строя) хотя бы одного процессора способно аннулировать все результаты, полученные на других. Обеспечение безопасности (безотказности) в системе становится важнейшей проблемой [17]. Например, вычислительные эксперименты по использованию распараллеливания, декларированного некоторыми универсальными пакетами математического моделирования, привели к неудовлетворительным результатам, вплоть до замедления выполнения расчета на нескольких процессорах по сравнению с последовательным счетом на одном процессоре. Такой эффект свидетельствует о неоптимальности унифицированного распараллеливания счета. Заметим, что разработка параллельного алгоритма на стадии его общего проектирования является лишь одним из аспектов данного этапа математического моделирования, хотя и весьма актуальным в настоящее время.

Однако это во многом лишь технологические проблемы. Главной же идейной трудностью этого этапа по-прежнему остается классическая проблема перехода от непрерывного представления математической задачи (аналитическое, дифференциальное, интегро-дифференциальное) к ее дискретному аналогу, поскольку компьютер оперирует только с дискретным представлением информации. Это весьма важный фундаментальный (и даже философский) скачок от "непрерывного" к "дискретному" и его нельзя недооценивать. Строгие доказательства аппроксимируемости непрерывной задачи ее дискретным представлением существует для ограниченного числа задач, линейных или квазилинейных. Можно согласиться с [18], что для "нелинейных задач таких теорем нет, и нет даже надежды, что они будут получены". Вследствие этого необходим очень тщательный и строгий контроль свойств разрабатываемого алгоритма. В сущности все разнообразие методов, на которых основаны вычислительные алгоритмы, — метод конечных разностей, конечных объемов, конечных элементов, методы частиц и т.д. — сталкивается с этой проблемой. Многие из этих методов имеют общие черты, свойства и проблемы. Многостороннее исследование различных аспектов (с их классификацией) создания вычислительных алгоритмов применительно к задачам механики сплошной среды выполнено в работе [1], в которой, на наш взгляд, весьма полно проведен анализ этого вопроса.

Во всех методах необходим предварительный анализ и последовательный контроль за выполнением законов сохранения, влияния числа узлов (ячеек, частиц) на точность решения, сходимость по этому параметру. В частности, сильная зависимость получаемого решения от числа узлов $N$ при их небольшом количестве является естественной и не вызывает опасения и неприятия, если эта зависимость исчезает при увеличении $N$. Если же при большом числе узлов эта зависимость не исчезает и решение продолжает меняться, то это — тревожный симптом низкого качества созданного алгоритма. Однако в математическом моделировании, при разнообразии задач, могут иметь место нелинейные проблемы, характеризующиеся высокими бифуркационными свойствами решения. И в этом случае сильная зависимость решения от $N$, наоборот, является показателем "качественности" дискретного алгоритма, отражающего фундаментальные черты решения непрерывной задачи — неединственность. Такие проблемы весьма трудны для решения, но часто встречаются в целом спектре важных научных задач астрофизики, высокоскоростной газовой динамики, механики разрушений, нанотехнологии в микроэлектронике и конструировании новых материалов, где имеют место неустойчивости и существенно нестационарные процессы [18]. Решение таких проблем индивидуально и, по-видимому, не имеет общих рецептов. Так, в [19] развитие схемной бифуркационной неустойчивости нанотрубок с пилообразным характером было устранено простым повышением точности представления чисел в формате с плавающей запятой (от одинарной до двойной) в компьютерной программе, а не в самом алгоритме, поскольку было замечено, что генерация неустойчивостей вызывается округлением последней цифры мантиссы.

Весьма важной проблемой данного сегмента математического моделирования нестационарных задач является проблема постановки начальных условий дискретного вычислительного алгоритма, соответствующих начальным условиям непрерывной дифференциальной задачи. При решении стационарных задач методом установления удачный выбор стартовых условий может существенно снизить затраты компьютерного времени, неудачный — значительно их увеличить или даже вообще не обеспечить получение стационарного решения вследствие выхода процесса установления на осцилляционный режим (периодический, квазипериодический или апериодический) [11].

Еще одним важным аспектом сегмента “4. Вычислительный алгоритм” является методика реализации граничных условий в случае существенной пространственной неоднородности отдельных физических процессов при декомпозиции общего процесса в сегменте “2. Физическая модель”. Так, при моделировании задач гравитационной газовой динамики постановка граничных условий для процесса конвективного переноса, описываемого в сегменте “3. Математическая модель” уравнениями Эйлера, может быть проведена на границе области пространства, занимаемой газовой средой, а граничные условия для процесса самогравитации, описываемые уравнением Пуассона, должны быть поставлены на бесконечности. Такое различие требует специальной организации в алгоритме двойной расчетной области (в приведенном примере область моделирования конвективного переноса вложена в область моделирования гравитационного взаимодействия). Далее в таком алгоритме вводится понятие “координаты бесконечности” вследствие невозможности использования математического понятия бесконечности в дискретном представлении чисел. Как правило, формируется специальная расчетная сетка с укрупнением шагов на периферии, от границы подобласти расчета конвекции до финальной границы на бесконечности. Другие алгоритмы решения этой задачи не имеют подобных проблем, например, комбинированный с tree-кодом иерархический SPH-метод [6], построенный на иных принципах, чем прямое решение уравнения Пуассона.

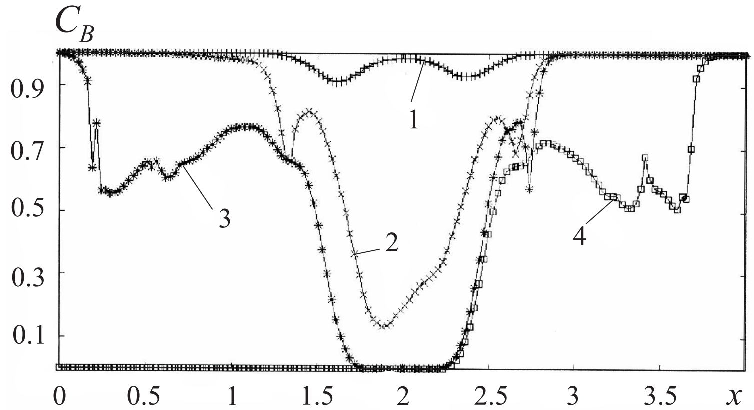


Рис. 6. Распределение концентраций бора $C_B(x, y)$ в кремнии и диоксиде кремния вдоль оси $x$ при различных значениях $y = 0.20$ (кривая 1), 0.61 (2), 1.0 (3) и 1.47 (4) для трехмасочной технологии отжига

Вообще говоря, оригинальность разработанных алгоритмов позволяет существенно продвинуться в области моделирования процессов, которые вызывали затруднения при использовании ранее разработанных алгоритмов, и “как-то” приспособиться путем различных модификаций (иногда весьма искусственных) к текущим задачам. Так, при моделировании физико-химических процессов, лежащих в основе производственного цикла для микроэлекроники, многочисленные вариации алгоритмов, основанных на модельном представлении пленки оксида кремния (сначала идеальной вязкой жидкостью, затем вязкопластичной или вязкоэластичной средой) приводили к большим трудностям при переходе от решения одномерных задач к двумерным. Конструирование алгоритмов на абсолютно иных принципах (модель “виртуальных направлений” [20] движения окислителя) позволили избежать традиционных трудностей и создать успешно функционирующий комплекс программ для расчета процессов формирования наноструктур донорных и акцепторных примесей в базовых подложках [21]. Эти структуры (рис. 6) вследствие образования зон проводимости электронного и дырочного типа обеспечивают требуемые электрофизические свойства полупроводникового материала. В качестве еще одного примера успешного конструирования оригинального вычислительного алгоритма можно привести ныне широко применяемый, полностью бессеточный гидродинамический метод сглаженных частиц (SPH-метод) [22]. Метод позволяет эффективно решать задачи с сильными деформациями и градиентами давлений: от космической газовой динамики до высокоскоростных соударений твердых тел с пробиванием и "разлетом" вещества, избегая при этом многих трудностей, которыми сопровождаются попытки решения подобных задач традиционными эйлеровыми (сеточными) методами.

Подводя итоги данного раздела, следует отметить, что сегмент "4. Вычислительный алгоритм" в цепочке этапов математического моделирования сложных физических процессов является не менее важным, чем рассмотренные ранее сегменты. Этот сегмент существенно определяет эффективность и интегральную точность создаваемого компьютерного инструментария, а также требует высоких интеллектуальных усилий и больших знаний в широком диапазоне научных областей.

**7. Пятый этап математического моделирования: разработка комплекса компьютерных программ.** Сразу подчеркнем, что этот, пятый, сегмент последовательности этапов математического моделирования не является завершающим. В какой-то мере в философском смысле его можно назвать "овеществлением" проведенных ранее стадий интеллектуальных разработок: эти стадии включают в себя всестороннее изучение физического явления, создание его физической модели и соответствующей математической модели, на основе которой проводится проектирование и отработка вычислительного алгоритма и анализ всех его свойств. Реализация пятого сегмента (программирование) также является деятельностью многостороннего плана, содержащей подсегменты: эскизное проектирование, конструирование, построение и отладку комплекса компьютерных программ. Этот комплекс включает в себя (здесь имеется в виду решение больших научных проблем, для которых разрабатываются соответствующие комплексы): генератор заданий (препроцессор), вычислительный модуль ("решатель"), который часто состоит из целого ряда подпрограмм для решения отдельных задач и таблично-графического интерпретатора вычислений (постпроцессор [23]).

В настоящем разделе рассматривается проблематика, связанная с разработкой генератора заданий и вычислительного модуля; проблеме интерпретаторов полученных результатов посвящены следующие пункты. Конструирование и разработка генератора заданий ориентированы на два принципиальных момента. Первый — это создание комфортных условий для потребителя вычислительных услуг, научного исследователя, цель которого — изучение явления, а не работа с программными кодами. Потребителю должны быть созданы условия для формирования своей задачи: ввода определяющих параметров и, если это необходимо, создания своей конфигурации расчета в вычислительном модуле. Как правило, именно эти позиции в существующих коммерческих пакетах, на наш взгляд, весьма недоработаны. Зачастую пользователя принуждают читать "толстые" инструкции, лежащие вне круга его интереса и с малодоступной терминологией. Чаще всего пользователю приходится проходить дополнительные курсы обучения (тренинги) перед началом использования пакета, проводимые фирмой-изготовителем. При этом стоимость такого курса на нескольких человек бывает сопоставимой с ценой на программную лицензию приобретаемого пакета. Фактически здесь имеет место перекладывания функций программиста на самого исследователя, который в принципе и не обязан владеть соответствующими знаниями, как не обязан водитель знать устройство двигателя внутреннего сгорания и его термодинамику. Принцип "каждому свое" здесь должен соблюдаться неукоснительно, и его несоблюдение зачастую приводит к недостаточно полному использованию возможностей даже хорошо задуманных и алгоритмически безупречно сконструированных программных комплексов.

Второй задачей генератора является построение по параметрам, введенным пользователем, исполняемого задания и запуск расчета. Этот чисто технический аспект здесь рассматриваться не будет. Следует лишь подчеркнуть, что его недооценка может привести к дискомфортным для пользователя условиям работы, а это, в свою очередь, может существенно ограничить возможности программного комплекса буквально "на ровном месте".

Главной задачей этого этапа математического моделирования является проектирование и разработка вычислительного модуля. В свойства модуля должны быть заложены принципиальные требования: расширяемость, адаптивность и переносимость. Свойство расширяемости предполагает гибкость создаваемых конструкций, с тем чтобы дополнение программного комплекса новыми сегментами и/или совершенствование уже разработанных не приводило к кардинальному пересмотру и переработке всего модуля в целом. Принцип максимальной автономии всех конструкций должен быть заложен в первоначальном проекте и в дальнейшем соблюдаться неукоснительно. Адаптивность программного комплекса (также называется масштабируемостью) подразумевает его способность оптимально настраивается на существующие аппаратные ресурсы и максимально полно их использовать. При этом, с одной стороны, программа обязана поддерживать аппаратные конфигурации существенно мощнее типичной, которая функционирует на момент создания комплекса программ. С другой стороны, должна также обеспечиваться работа комплекса программ и на относительно слабых вычислительных ресурсах. В этом случае, наоборот, функциональная деградация комплекса должна быть по возможности минимальной. В особенности свойство адаптивности важно при создании компьютерного инструментария, ориентированного на параллельные вычисления на суперкомпьютерах. Существующее сегодня множество суперкомпьютеров весьма неоднородно как по своей архитектуре, так и по числу процессорных элементов. В идеале свойство адаптивности должно гарантировать производительность программного комплекса, близкую к пиковой производительности на любой вычислительной архитектуре, что, конечно же, недостижимо на практике, но должно являться направлением развития программного обеспечения.

Переносимость — способность модуля к функционированию в программной и аппаратной среде с вариацией ее свойств. Крайне желательно, чтобы модуль (возможно, после перекомпиляции) функционировал в новой программно-аппаратной среде не только устойчиво, но и абсолютно одинаково, вплоть до совпадения знаков в последних порядках мантисс получаемых цифровых данных. При этом опасны не глобальные отказы, причины которых практически всегда легко обнаруживаются и устраняются, а малозаметные отклонения данных от контрольных значений. Зачастую такие отклонения вычислители с недостаточным опытом считают несущественными, но на самом деле это грозный признак, который может привести к неверным результатам математического моделирования.

**8. Шестой этап математического моделирования: организация полученных данных.** Этот сегмент последовательности этапов моделирования (рис. 1) выполняет две главных функции: предметную (научную) и техническую, которые соответствуют философским категориям "содержание и форма". Первая из них заключается в научном значении полученных данных, что и является целью всего процесса математического моделирования. Эта сторона рассматриваемого этапа ясна и не нуждается в особых комментариях. Следует кратко остановиться на второй стороне вопроса, поскольку ей при разработке вычислительных комплексов не всегда уделяется должное внимание. Если результатом моделирования какого-либо процесса являются всего несколько чисел, то проблемы не существует. Если же результат представляется тысячами и десятками тысяч (иногда даже гораздо больше) чисел, то возникает проблема структурирования, представления, хранения и передачи информации. Структурирование и представление полученных данных заключаются в организации цифровых таблиц в специальной форме, максимально облегчающей анализ информации большого объема. Эти таблицы могут быть размещены в базах данных долгосрочного хранения, системы управления которыми могут существенно оптимизировать сам процесс анализа результатов. Вообще говоря, эта проблема давно и успешно решается, однако в связи с ростом популярности коммерческих научных разработок часто появляются программные продукты, результатами которых пользоваться затруднительно вследствие плохо организованного вывода информации.

Распространение технологий параллельного счета на суперкомпьютерах вызывает появление новых проблем обработки информации сверхбольшого объема (в частности, проблем организации выходных данных, полученных в результате решения одной задачи на мультипроцессорной системе). Такие технологии существенно связаны со способом распараллеливания. Например, при геометрической декомпозиции (наиболее распространенной методики распараллеливания реальных задач) всей области решения на ряд подобластей расчет в каждой из них обеспечивается индивидуальным процессором с выводом информации в отдельный файл. Следует собрать все эту информацию в интегрированную базу данных для последующей обработки и анализа. Эту проблему можно решить по-разному.

Так, вывод и интегрирование полученной информации в вычислительном комплексе "Поток-3" [4], предназначенном для моделирования обтекания объекта высокоскоростным потоком сжимаемого вязкого теплопроводного газа, организован следующим образом. Имена файлов локальных данных (результатов, полученных на отдельном процессоре) составлялись из семи позиций, что представлялось в виде шестизначного числа с буквой в первой позиции. Эта буква маркировала глобальную подобласть проведения расчета. Использовалось только четыре литеры: H (head body) — подобласть обтекания головной части тела, L (lateral surface) — подобласть течения над боковой поверхностью (фюзеляжем), N (near wake) — подобласть обтекания кормы и течения в ближнем следе, F (far wake) — подобласть течения в дальнем следе. Первые три цифры имени соответствовали номеру процессора, на котором проводился расчет течения в локальной подобласти с тем же номером. Последние три цифры соответствовали номеру варианта задачи в каталоге исследований (вариации по скорости, высоте, конфигурации летательного аппарата и т.п.). Например, имя файла H571212 означало, что в нем содержатся результаты расчета обтекания головной части в подобласти 571 задачи 212. Все файлы с номером 212 в дальнейшем сводятся в общую базу данных H212, которая вместе с базами данных L212 и F212 составляет интегрированную БД 212. Разумеется, могут быть предложены и реализованы интегрированные банки данных и с другой организацией информационных потоков решений. Однако следует подчеркнуть, что в этой проблеме есть подводные камни, часть из которых рассматривается ниже.

**9. Проблемы визуализации решений, полученных при параллельном компьютерном моделировании на мультипроцессорных системах.** Большой объем данных, получаемых при компьютерном моделировании даже на однопроцессорных ЭВМ, приводит к затруднениям при анализе результатов. Этот анализ не облегчают даже хорошо организованные цифровые таблицы. В связи с этим используются графические системы, обеспечивающие визуализацию полученной цифровой информации, что в особенности важно на первых этапах анализа, при общем исследовании решения задачи. Все новые эффекты, динамика процессов, и т.п. выявляются исключительно визуальным способом. Табличная информация только поддерживает графическую и позволяет провести уточнение анализа решения, переводя его из глобального качественного и общего количественного в детально-количественный. При этом, естественно, графические системы (Origin, TecPlot и т.п.) должны быть надежными, не искажающими полученную информацию при построении 1D-, 2D- и 3D-изображений.

Еще раз подчеркнем, что одной из главных проблем, затрудняющих использование параллельных технологий, является проблема информации сверхбольшого объема, полученной при решении задачи. Вопросы ее интегрирования очень сложны. Собирать ли все данные со всех процессоров или только частично? Тогда как именно, ибо можно потерять важные данные? Как их складировать и где их хранить? Как анализировать полученный результат? По числовым таблицам гигантского размера это сделать затруднительно или даже невозможно. Здесь применение графических систем обработки информации ставит свои проблемы. Возможна неправильная работа этих систем (интерполяция, сглаживание), что может привести к потере интересных и важных результатов, особенно при резонансных явлениях.

Проанализируем некоторые количественные характеристики компьютерных ресурсов при организации параллельных вычислений. Типичная задача (гравитационная газовая динамика, высокоскоростная аэродинамика), решение которой имеет смысл проводить в настоящее время на мощном суперкомпьютере, является трехмерной и требует порядка $10^3$ узлов сетки по каждому направлению, т.е. всего порядка $10^9$ узлов. В каждом узле сетки требуется хранить значения не менее пяти основных (в соответствии с законами сохранения массы, энергии и трех компонент импульса) и около пяти дополняющих ($x$, $y$, $z$ координаты и др.) значений параметров. Итого — более $10^{10}$ единиц информации. Каждая из этих единиц требует 8 байт. В сумме, вместе с дополнительными затратами, требуется около 100 Гб памяти для хранения результатов решения только одной крупной современной задачи. Такой сверхбольшой объем создает значительные проблемы и для архивирования, и для пересылки по Интернету при использовании суперкомпьютера в режиме удаленного доступа, и для анализа решения как в цифровой форме, так и при ее визуализации. Ни одна современная графическая система не способна беспроблемно обработать такое количество информации. Это приводит к необходимости применения специальных приемов визуализации решения.

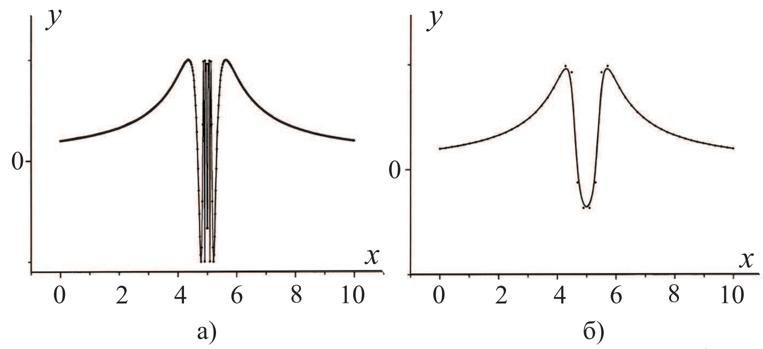


Рис. 7. График одной и той же функции $F(x) = \sin \dfrac{1}{|x-5|}$, отрисованный графической системой Origin с использованием дискретной сетки в 500 узлов (а) и 50 узлов (б)

Можно предложить два основных направления для решения этой проблемы. Первое из них заключается в обработке данных графическими системами по частям по принципу "Один процессор — один монитор — одна подобласть визуализации" без интегрирования данных, полученных на каждом процессоре, в единую базу. На наш взгляд, это крайне неудобный способ анализа, в особенности при необходимости проведения многовариантных расчетов. Второй путь — интегрирование данных в единый банк, с последующим созданием специальных "укороченных" баз данных. В каждой из них может содержаться часть общей информации с равномерной (или неравномерной) выборкой из общего банка: например, каждое второе, или пятое, или десятое, или сотое число, в зависимости от того, какой общий объем информации может обработать графическая система визуализации при построении 1D-, 2D- или 3D-изображений. Однако этот путь требует осторожности, поскольку приводит к проблемам, одна из которых иллюстрируется ниже.

Рассмотрим функцию $F(x) = \sin \dfrac{1}{|x-5|}$. График этой функции, отрисованный на интервале $x$ от 0 до 10, приведен на рис. 7. Графическая система Origin оперировала с дискретными значениями этой функции $F_i$, заданной в различных равномерно распределенных точках $x = x_i$ при $i$, изменяющимся от 1 до $I_{\max}$ (значение особой точки $x = 5$ исключалось из списка $x_i$). Приведены два графика: при $I_{\max} = 500$ (рис. 7 а) и $I_{\max} = 50$ (рис. 7 б). Первый из них моделирует отрисовку графика по значениям из полного интегрированного банка данных, второй — по значениям из укороченной базы данных с коэффициентом прореживания 10, т.е. взята каждая десятая точка. "Точное" решение, показанное на рис. 7 а, содержит цикл тонких структур, дислоцированных в окрестности точки $x = 5$ и моделирующих, например, резонансные явления, которые имеют место в различных физических процессах. Эти резонансные явления полностью потеряны (рис. 7 б) при визуализации данных из "укороченной" БД, хотя структуры решения в общем являются одинаковыми. Оба решения вне подобласти совпадают с высокой точностью, причем эта точность увеличивается к периферии. Таким образом, если не знать о существовании особой подобласти решения, то ее при визуализации можно и не обнаружить, хотя это решение получено вычислительным алгоритмом и содержится в интегрированном банке данных. Подчеркнем, что кажущийся близким вопрос о точности сплайн-аппроксимации непрерывных функций дискретными представлениями лежит совсем в другой плоскости, относится к характеристикам графических систем визуализации цифровых данных и к рассматриваемой проблеме отношения не имеет.

В настоящее время нет универсальных рецептов для решения этих проблем, однако есть определенный оптимизм в прогнозировании появления специальных методов и алгоритмов для их решения.

**10. Седьмой этап математического моделирования: анализ результатов.** Этот сегмент завершает первую итерацию последовательности этапов математического моделирования от реального физического явления к созданию физической, а затем математической модели, на базе которых проводится конструирование вычислительного алгоритма, разработка комплекса компьютерных программ и организация полученных данных в оптимальном для анализа таблично-графическом представлении результатов. Существуют два типа тестирования точности созданного вычислительного инструментария: внутренний и внешний контроль.

Внутренний контроль — это тип верификации, включающий в себя цикл экспериментов, достаточно хорошо разработанный и стандартизированный. Прежде всего должен существовать контроль за выполнением фундаментальных законов сохранения (массы, импульса, энергии), формулировки которых для различных отраслей знаний достаточно разнообразны. Результаты эксперимента не должны противоречить этим формулировкам. Контроль за точностью выполнения законов сохранения определяет степень пригодности выбранных физико-математических моделей описания явления. Абсолютно бессмысленно совершенствовать численный алгоритм и компьютерную программу, когда законы сохранения выполняются с недостаточной точностью (если, конечно, сам процесс совершенствования не направлен на улучшение выполнения законов сохранения). В этом случае результаты моделирования будут далеки от реальности, несмотря на некоторую правдоподобность. В то же время следует особо подчеркнуть, что этот вид контроля является необходимым, но не достаточным. Широко распространенной, на наш взгляд, ошибкой является "переоценка" законов сохранения, когда их контроль становится единственным способом верификации программы. Зачастую алгоритмы (и программы) выдают неверные результаты, несмотря на хорошую точность выполнения законов сохранения. В частности, многие консервативные неявные вычислительные схемы способны автоматически выполнять законы сохранения практически для любого отношения пространственного и временнóго шага, что, конечно же, еще не означает их высокую точность.

К внутреннему контролю также относится верификация инструментария по главным алгоритмическим параметрам. Для конечно-разностных схем ими являются шаги по времени $\tau$ и пространству $h$. Для всех явных схем существует жесткое ограничение на соотношение $\dfrac{\tau}{h}$ (условие Куранта). Для большинства неявных схем это ограничение формально отсутствует, но фактически имеет место, т.е. шаг $\tau$ нельзя выбирать произвольно большим. Существует оптимальное соотношение, при котором время решения, например стационарных задач методом установления, минимально, с резким его увеличением на порядки при отклонении $\dfrac{\tau}{h}$ от оптимального. При разработке компьютерного инструментария оптимальное соотношение алгоритмических параметров также должно быть найдено и заложено в автоматизированную систему управления комплексом [24].

Следует остановиться еще на одном аспекте внутреннего контроля — анализе сходимости алгоритма. Сравниваются решения $F$ при различных уменьшающихся $h$, т.е. $F_1(h)$, $F_2(h)$, …, $F_n(h)$. Предполагаются априори два свойства алгоритма, которые верифицируются в тестах — сходимость и аппроксимация.

Первое: решение перестает меняться или меняется незначительно (в какой-либо норме), начиная с некоторого $h < h^*$, где $h^*$ — некоторый порог “насыщения” для данного алгоритма, если этот порог существует. Второе: при уменьшении $h$ решение становится более “точным”, т.е. предполагается, что решение дискретной задачи приближается к решению непрерывной (исходной). Для существенно нелинейных задач это не всегда так. Вблизи точек бифуркаций, соответствующих в реальности физическим неустойчивостям, численные решения с меньшими $h$ (и $\tau$) могут оказаться менее качественными, поскольку выводят расчет на другую ветвь решения. Этот вопрос уже обсуждался (см. цитату из [18] в разделе 6); здесь он рассматривается еще раз вследствие своей важности. Подчеркнем, что это все относится только к моделированию сложных нелинейных задач, таких, как гравитационная газовая динамика, высокоскоростная газовая динамика и др.

Такие вопросы, связанные с глубинными свойствами вычислительных алгоритмов (переход от непрерывности к дискретности), которые проявляются при моделировании существенно нелинейных и нестационарных проблем с многочисленными неустойчивостями разной природы, будут все более и более заостряться с расширением применения суперкомпьютеров, вследствие возможности использования высокоразрешающих сеток (элементов, частиц).

Внешний контроль также хорошо отработан к настоящему времени и включает в себя сравнение результатов, полученных компьютерным моделированием, с результатами, получаемыми аналитически (в тех областях, где они существуют) и экспериментально, и с численными результатами других алгоритмов. Например, в газовой динамике существуют пять специальных тестов [25], верификация на которых является непременным этапом представления новых вычислительных алгоритмов. Следует отметить несколько проблем верификации алгоритмов на сложных задачах. Допустим, проводится сравнение решений, полученных разными вычислительными комплексами. Возможны два варианта: решения коррелируют между собой с достаточной точностью или решения различаются. В первом случае можно говорить о положительном результате тестирования. Второй случай далеко не прост и примыкает к обсуждавшейся выше проблеме моделирования сложных нестационарных процессов с физическими неустойчивостями. Какой алгоритм обеспечивает получение более качественного решения? Не всегда новый алгоритм следует признавать менее точным только лишь на том основании, что старый алгоритм уже был достаточно апробирован на широком классе задач в течение длительного времени. И наоборот, новый алгоритм не следует автоматически признавать более качественным вследствие использования в нем более современных методик расчета. В любом случае нужна дополнительная диагностика нового вычислительного инструментария и даже его длительная рабочая эксплуатация.

**11. Восьмой этап математического моделирования: принятие решения.** Этот сегмент в цепочке этапов математического моделирования (рис. 1) является финальным в работе вычислительного инструментария — своего рода квинтэссенцией всей интеллектуальной деятельности на предыдущих этапах. На основе анализа результатов (см. предыдущий пункт) в этом сегменте принимается решение о завершении или продолжении разработки. В последнем случае необходимо провести дополнительные исследования, какой из сегментов моделирования следует доработать, пересмотреть или кардинально изменить. Возникает итерационный цикл работ по сегментам, и глубина этого цикла, т.е. степень возврата на позиции предшествующих сегментов диктуется требованиями, которые были определены в процессе анализа:

1) провести дополнительный анализ;
2) предоставить новый пакет данных;
3) оптимизировать компьютерную программу;
4) улучшить вычислительный алгоритм;
5) уточнить математическую модель;
6) расширить физическую модель.

Список приведен в порядке возрастания степени доработки вычислительного инструментария. Первое требование означает необходимость возврата на шаг назад в последовательности этапов моделирования, к его сегменту “7. Анализ результатов”, к дополнительному рассмотрению уже полученных данных для принятия решения. Второе требование означает неполноту полученной информации для принятия решения с необходимостью предоставления дополнительных данных. Это приводит к возврату на этап “6. Организация полученных данных” для доработки, иногда значительной, таблично-графического интерпретатора данных, полученных комплексом компьютерных программ. Третье требование означает необходимость совершенствования компьютерного вычислительного модуля, поиск и устранение ошибок программирования, оптимизацию его функционирования, создание новых сервисов для пользователя. Заметим, что в любом крупном вычислительном комплексе всегда имеются ошибки программирования (“глюки” и “баги” на жаргоне программистов), полагать иначе было бы неоправданным оптимизмом. Еще раз подчеркнем, что при этом обычно опасны не глобальные отказы, легко обнаруживаемые и устраняемые, а малозаметные неточности. Четвертое требование означает возврат к сегменту “4. Вычислительный алгоритм” с необходимостью его доработки: от устранения возможных неточностей при первом возврате к этому сегменту, до пересмотра некоторых, иногда даже ключевых позиций (например, повышения порядка аппроксимации и т.п.) и/или дополнение некоторыми новыми функциями. Пятое требование о необходимости уточнения математической модели означает, что в ходе проведения исследований обнаружилась некоторая ограниченность возможности применения разрабатываемого инструментария, сужающая класс решаемых задач. Например, при разработке программного комплекса “Поток-1” [24], обеспечивающего расчет аэродинамики высокоскоростного полета, его функциональные возможности для моделирования полета на больших высотах были дополнены математическими моделями проскальзывания потока, а для малых высот была добавлена модель турбулентности. Шестое требование приводит к возврату в сегмент “2. Физическая модель” и означает, что во время создания вычислительного инструментария появилась необходимость существенного расширения учитываемых физических процессов. В частности, при моделировании проблем гравитационной газовой динамики [6] в одних случаях можно применять классические законы тяготения Ньютона, а в других — необходимо учитывать изменение метрики пространства вблизи сильных источников гравитации (черные дыры) согласно общей теории относительности. Аналогично и в других областях математического моделирования: увеличение функциональных возможностей связано с расширением модели. Так, при разработке вычислительного комплекса “Поток-5” физическая модель сжимаемого теплопроводного газа была дополнена моделью, учитывающей возбуждение колебаний в молекулах кислорода и азота и их последующую диссоциацию, и далее — ионизацию при высоких давлениях и температурах, что позволяет прогнозировать изменение свойств воздуха вокруг летательного аппарата на гиперзвуковых скоростях полета. В свою очередь, расширение физической модели потребовало дополнить и соответствующую ей математическую постановку (третий сегмент) математической моделью эффективного показателя адиабаты [10], количественно отражающей и учитывающей диссоциацию молекул газа в равновесном и квазиравновесном приближении в замкнутой системе уравнений Навье–Стокса. Естественно, что это, в свою очередь, требует проведения соответствующих модификаций в нижестоящих сегментах — вычислительном алгоритме и комплексе программ.

Такой подход к математическому моделированию, с неоднократным итерированием последовательности всех его основных этапов, связан с большими затратами времени и интеллектуальных усилий, но является неизбежным при разработке хорошо функционирующего вычислительного инструментария. Очевидно, что он также далеко не всегда используется при создании некоторых универсальных пакетов компьютерных программ в условиях жесткого цейтнота и постоянной необходимости “форсированного” выпуска их новых версий без надлежащего тестирования. Подобное скептическое отношение у авторов сформировалось на основе опыта эксплуатации некоторых таких пакетов в суперкомпьютерном центре.

**12. Девятый этап математического моделирования: завершение исследования.** Данный сегмент завершает процесс создания вычислительного инструментария для математического моделирования какого-либо реального физического явления. Расширяется область знаний с возможностью генерации новых представлений о природе изучаемого явления. Создаются новые научные теории и на их основе реализуются прикладные разработки действующих механизмов, приборов, аппаратов. Нередко на этом этапе наблюдается коммерциализация произведенного научного продукта: как собственно вычислительного комплекса в целом или его элементов, так и полученных научных результатов. Финальным сегментом последовательности этапов математического моделирования может быть назван его десятый (рис. 1) сегмент “Новая физическая проблема” — переход исследований на новый, более высокий уровень понимания данного явления природы или изучение нового явления.

## СПИСОК ЛИТЕРАТУРЫ

1. *Волков К.Н.* Разработка и реализация алгоритмов численного решения задач механики жидкости и газа // Вычислительные методы и программирование. 2007. **8**, № 1. 197–213.
2. *Аксенов А.А., Коньшин В.Н.* Применение программного комплекса FlowVision для проектирования авиакосмических конструкций // САПР и графика. 2004. **11**. 51–68.
3. *Грахов Ю.В.* Практика использования пакета ANSYS CFX для численных исследований на кластерных системах аэрогазодинамических характеристик многокомпонентных ракет // Параллельные вычислительные технологии. Труды международной конференции ПАВТ’2007. Изд-во ЮУрГУ, 2007. **2**. 44–50.
4. *Тарнавский Г.А., Корнеев В.Д., Вайнер Д.А., Покрышкина Н.М., Слюняев А.Ю., Танасейчук А.В., Тарнавский А.Г.* Вычислительная система “Поток-3”: опыт параллелизации программного комплекса. Часть 1. Идеология распараллеливания // Вычислительные методы и программирование. 2003. **4**, № 1. 37–48.

5. *Тарнавский Г.А., Тарнавский А.Г.* Мультипроцессорное компьютерное моделирование в гравитационной газовой динамике // Вычислительные методы и программирование. 2005. **6**. 71–87 (http://num-meth.srcc.msu.ru).
6. *Алиев А.В., Тарнавский Г.А.* Иерархические SPH-методы для математического моделирования в гравитационной газовой динамике // Сибирские электронные матем. изв. 2007. **6**. 376–434 (http://semr.math.nsc.ru).
7. *Тимбай И.А.* Движение капсулы Fotino на атмосферном участке траектории (http://volgaspace.ru/samara_coe/expertises/fotino_timb_r.doc).
8. *Тарнавский Г.А., Шпак С.И.* Проблемы численного моделирования сверхзвукового ламинарно-турбулентного обтекания тел конечного размера // Математическое моделирование. 1998. **10**, № 6. 53–74.
9. *Кларк Дж., Макчесни М.* Динамика реальных газов. М.: Мир, 1967.
10. *Тарнавский Г.А., Шпак С.И.* Эффективный показатель адиабаты в задачах гиперзвукового обтекания тел реальным газом // Теплофизика и аэромеханика. 2001. **1**. 41–57.
11. *Тарнавский Г.А., Хакимзянов Г.С., Тарнавский А.Г.* Моделирование гиперзвуковых течений: влияние стартовых условий на финальное решение в окрестности точек бифуркации // Инженерно-физический журн. 2003. **76**, № 5. 54–60.
12. *Тарнавский Г.А.* Неединственность ударно-волновых структур в реальных газах: маховское и/или регулярное отражение // Вычислительные методы и программирование. 2003. **4**, № 2. 258–276.
13. *Волков К.Н.* Применение средств параллельного программирования для решения задач механики жидкости и газа на многопроцессорных вычислительных системах // Вычислительные методы и программирование. 2006. **7**, № 1. 73–88.
14. *Тарнавский Г.А., Алиев А.В., Тарнавский А.Г.* Пространственное распараллеливание с переключением координатных направлений для решения уравнений математической физики на суперЭВМ // Параллельные вычислительные технологии. Труды международной конференции ПАВТ'2007. Изд-во ЮУрГУ, 2007. **2**. 109–112.
15. *Воеводин В.В., Воеводин Вл.В.* Параллельные вычисления. СПб: БХВ-Петербург, 2002.
16. *Корнеев В.Д.* Параллельное программирование в MPI. Москва; Ижевск: Институт компьютерных исследований, 2003.
17. *Selikhov A., Germai C.* A channel memory based on fault tolerance for MPI applications // Future Generation Comp. Syst. 2005. **4**, № 21. 709–715.
18. *Белоцерковский О.М.* Математическое моделирование на суперкомпьютерах (опыт и тенденции) // Ж. вычисл. матем. и мат. физ. 2000. **40**, № 8. 1221–1236.
19. *Korobeynikov S.N., Babichev A.V.* Numerical simulation of dynamic deformation and buckling of nanostructures // ICF Interquadrennial conference, CD full papers (Ed. by R.V. Goldstein), Inst. for Problems in Mechanics RAS, Moscow. 2007.
20. *Тарнавский Г.А., Шпак С.И., Обрехт М.С.* Численное моделирование и компьютерный алгоритм процесса сегрегации легирующих примесей на границе волны окисления в полупроводниковых подложках // Вычислительные методы и программирование. 2001. **2**, № 1. 16–30.
21. *Тарнавский Г.А., Алиев А.В., Тарнавский А.Г.* Создание специальных наноструктур донорных и акцепторных примесей в базовой подложке кремния для конструирования новых полупроводниковых материалов // Нано- и микросистемная техника. 2007. № 9. 51–62.
22. *Gingold R.A., Monaghan J.J.* Smoothed particle hydrodynamics — Theory and application to non-spherical stars // Royal Astronomical Society, Monthly Notices. Nov. 1977. **181**. 375–389.
23. *Тарнавский Г.А., Тарнавский А.Г., Гилев К.В.* Информационно-вычислительный Интернет-центр "Аэромеханика". Первая линия: программный комплекс "Удар" // Вычислительные методы и программирование. 2005. **6**. 27–48 (http://www.srcc.msu.su/num-meth).
24. *Ковеня В.М., Тарнавский Г.А., Черный С.Г.* Применение метода расщепления в задачах аэродинамики. Новосибирск: Наука, 1990.
25. *Toro E.F.* Riemann solvers and numerical methods for fluid dynamics. A practical Introduction. Berlin: Springer-Verlag, 1999.

Поступила в редакцию
27.09.2007