Современные проблемы моделирования в свете идей Н.Н.Яненко [*]

В.П.Ильин

Институт вычислительной математики
и математической геофизики СО РАН,
Новосибирский государственный университет
Новосибирск

1 Введение

Современное бурное развитие концепции, методов и технологий математического моделирования подтверждает ту известную истину, что новое — это хорошо забытое старое. В значительной степени решаемые в начале 21-го века актуальные проблемы вычислительных наук (computer science) представляют собой переосмысливание понятий, заложенных еще в 70-е годы пионерскими работами ученых, среди которых выдающуюся роль сыграл Н.Н.Яненко. Здесь следует отметить не только его личные результаты, но и фактически коллективные решения, сформировавшиеся в многочисленных бурных дискуссиях на организованных им регулярных семинарах и конференциях, сыгравших огромную роль в создании научных школ в различных городах и республиках СССР. Рождение новых определений и методик в этих мозговых штурмах, как правило, было обязано инициативе Николая Николаевича, [1] – [2].

Несколько парадоксальным, на первый взгляд, оказался такой термин, как “инвариант Н.Н.Яненко” — “большая задача”. Смысл его был в том, что на ЭВМ любого поколения большая (для своего текущего момента) задача решается примерно одинаковое астрономическое время — часы, десятки часов или более. Аналогично определялись малые и средние задачи — соответственно с длительностью счета в минуты и десятки минут. За прошедшие десятилетия продолжает неуклонно выполняться закон Мура, в согласии с которым за каждые 11 лет характерное быстродействие компьютеров увеличивается в 1000 раз: именно такой период проходил между появлениями первых гигафлопников (1986 г.), терафлопников (1997 г.) и петафлопников (2008 г.). И по этому же закону в 2019 году появится первый экзафлопный компьютер. Но тем не менее тезис Н.Н.Яненко о большой задаче остается в силе. Понятно, что это уже качественно новые проблемы: междисциплинарные, обратные, со сверхвысоким разрешением или недоопреденными данными. Размерности, или числа степеней свободы, теперь измеряются миллионами или сотнями миллионов. “Дорожная карта” (Road Map) Международного комитета по экзафлопным вычислениям (IESP – International Exascale Software Program) содержит список впечатляющих задач из биологии, материаловедения, геофизики и т.п., реально демонстрирующий, что математическое моделирование стало третьим путем познания, осуществляя роль посредника между теоретиками и экспериментаторами, см. [3] и цитируемые там работы.

Большое значение, как методологическое, так и практическое, сыграли проводимые Николаем Николаевичем острые обсуждения понятия математического и программного модуля. По-

[*]Работа поддержана грантами РФФИ № 008-01-00526 и ОМН РАН № 1.3.4

явившиеся впоследствии названия сборочного програмирования, фрагментарного и объектно-ориентированного программирования фактически так или иначе следуют модульному принципу. Описанная Н.Н.Яненко знаменитая “технологическая цепочка” крупномасштабного вычислительного эксперимента, реализуемая на этапах жизненного цикла пакетов прикладных программ (ППП): проектирования, разработки, эксплуатации, сопровождения и развития,— углубляется и специализируется, чему будет посвящен следующий пункт данной работы. Здесь ключевыми моментами, как и в те годы, остаются вопросы автоматизации построения алгоритмов, а также их распараллеливания и отображения на архитектуру ЭВМ (сейчас точнее говорить МВС – многопроцессорных и многоядерных вычислительных систем).

Творческая судьба Николая Николаевича, который еще до своих выдающихся результатов в вычислительной математике и решении важнейших прикладных задач, включая оборонные, успел защитить докторскую диссертацию по дифференциальной геометрии, подтверждает истину о единстве теоретической физики, математики и вычислительных наук (computer science) [4]. Для текущего момента знаменательно не только пришествие суперкомпьютеров и сверхзадач, но и внедрение в расчетную практику качественно новых алгоритмов на основе теории групп, многообразий, высшей алгебры и топологии, гамильтоновых формализмов и т.д.

Мировой рынок прикладного программного обеспечения перенасыщен библиотеками и пакетами программ для всевозможных областей (NETLIB, NAG, IMSL, NASTRAN, ANSYS, MATLAB,...), как свободно распространяемыми, так и коммерческими, и измеряются в денежном выражении миллиардами. Однако, как авторитетно заявляет IESP, вычислительное сообщество оказалось в драматической ситуации, обусловленной неизбежной сменой всего программного обеспечения, в связи с уже начавшимся переходом от компьютеров с тысячами процессоров к суперцентрам с миллионами процессоров и миллиардами ядер. Предстоит кардинальная смена технологий и парадигм программирования, формирующая уже такие очевидные принципы, как создание общедоступного кода (Open Source) и организация всемирной кооперации его разработчиков. При такой глобализации проектов безусловно заманчиво попытаться осмыслить возможные пути развития алгоритмов и технологий программного обеспечения наукоемких вычислений.

2 Алгоритмические структуры и технологии на стадиях вычислительного эксперимента

Мы рассмотрим на содержательном уровне достаточно общую задачу математического моделирования, а также основные технологические этапы крупномасштабного вычислительного эксперимента, закладываемых в концепцию базовой системы моделирования (БСМ, [3]), которая должна служить инструментальной, по принципу популярного детского конструктора LEGO, естественным образом расширяемого и общедоступного для создания конкретных приложений. Компоненты БСМ должны поддерживать следующие алгоритмические структуры и технологии, с обязательным акцентом на эффективность распараллеливания [5].

Геометрическое функциональное моделирование. Задание пользователем исходных данных задачи и указаний по методам ее решения (формирование физической, математической и вычислительной моделей) — это первая стадия вычислительного эксперимента, которую назовем этапом геометрического и функционального моделирования. Его содержанием является описание и возможное редактирование геометрических объектов расчетной области (вершины, ребра, грани, фигуры), вместе с указанием топологических связей между ними, а также функциональных объектов, характеризующих типы решаемых уравнений, краевые и начальные условия, а также представляемых в них коэффициентах (с привязкой к соответствующим подобластям и граничным поверхностным сегментам). Для обратных задач дополнительно указываются параметризованные исходные данные, совместно с описанием ограничений на возможные вариации параметров и формулировкой целевого функционала. Для обеспечения дружественного интерфейса пользователю должны на данном этапе предоставляться интеллектуальные графические и текстовые средства формирования математической модели и вычислительного задания. А результатом работы этой стадии является геометрическая и функциональная структуры данных (ГСД и ФСД), однозначно определяющие исходную информацию, необходимую для выполнения последующих этапов моделирования. В целях придания гибкости БСМ предусматривается возможность множественного представления форматов, с их взаимной конвертизацией. В частности, это необходимо для поддержки взаимодействия с распространенными внешними приложениями, например, графическими или САПРовскими продуктами, базирующимися на общепринятых геометрических форматах, см. подробнее [6], [7].

Дискретизация математической модели. Построение сетки представляет особо сложную проблему в многомерных случаях с кусочно-гладкими криволинейными и, возможно, движущимися границами (для последних специальный класс составляют т.н. свободные границы, положение которых заранее неизвестно). Сама сетка определяется совокупностью своих объектов различной размерности: узлы, ребра, грани, конечные объемы,– а также топологическими связями между ними. Между геометрической структурой расчетной области с подобластями и сеточной структурой существует большая аналогия, и они содержат по сути однотипные наборы объектов макро- и микроуровня.

Методы построения сеток отличаются большим разнообразием и имеют обширную профессиональную литературу. Здесь используются и квазиконформные отображения, и вариационные принципы, и специальные метрические пространства, и многочисленные эмпирические приемы. Соответствующее программное обеспечение, или генераторы сеток, существует в широком ассортименте, или в свободном доступе через Интернет, или в качестве коммерческих продуктов. Зачастую процедуры построения сеток являются неотъемлемой частью проблемно-ориентированных пакетов прикладных программ (ППП), но они также существуют и в автономной форме, позволяющей их встраивать в различные приложения.

Согласно идеологии БСМ, стадия дискретизации задачи реализуется независимо от остальных, используя на входе только один из форматов ГСД, ГФСД и формируя на выходе сеточную структуру данных (ССД), также допускающую множественные форматы данных с конверторами. Подчеркнем, что методология квазиструктурированных сеток допускает использование внешних существующих сеточных генераторов, осуществляя таким образом актуальные возможности интеграции различных приложений и переиспользования программного кода.

Алгоритмы аппроксимации. Конечной целью дискретизации является переход от исходных функциональных соотношений к конечно-мерным уравнениям, неравенствам или рекурсиям. При этом фактически задача алгебраизируется, что достигается путем аппроксимации функций, производных и интегралов. Основные подходы к построению сеточных соотношений – это методы конечных разностей, конечных объемов, конечных элементов (МКР, МКО, МКЭ, см. [8], [9]), коллокаций и спектральные методы, связанные с разложениями в ряды Фурье. Не пытаясь делать обзора имеющегося огромного материала по данным вопросам, мы коротко остановимся главным образом на технологических аспектах наиболее универсальных МКО и МКЭ, позволяющих конструировать сеточные аппроксимации высоких порядков точности на различных типах конечных объемов для широкого класса задач математического моделирования.

Важным моментом этих двух подходов является поэлементная технология вычисления локальных матриц и векторов правых частей с последующей сборкой (ассемблированием) глобальных матриц и систем алгебраических уравнений (линейных или нелинейных – СЛАУ или СНАУ). Этот прием значительно упрощает программную реализацию и естественным образом обеспечивает эффективное распараллеливание данного вычислительного этапа, поскольку такие процедуры аппроксимации носят локальный характер, реализация которых использует свойства только топологически ближайших сеточных объектов, но никак не использует их дальние связи..

В нестационарных проблемах пространственная аппроксимация осуществляется на каждом временном шаге, а при наличии нелинейностей такая процедура повторяется итерационно для всех шагов. Наиболее трудоемкими оказываются задачи с динамическими сетками, если их приходится перестраивать на каждой итерации для каждого шага по времени.

Результатом выполнения этапа аппроксимации является алгебраическая структура данных (АСД), аккумулирующая на дискретном уровне всю исходную информацию о решаемой проблеме, и для которой имеются общепринятые в мире сжатые матричные форматы, вместе с соответствующими переходниками – конверторами.

Задачи вычислительной линейной алгебры. Если решаемая проблема в целом является нестационарной и нелинейной, все равно после этапов временной аппроксимации и квазилинеаризации мы приходим к задачам линейной алгебры, из которых наиболее типичны – это решение СЛАУ или проблемы собственных значений.

Характерные матрицы, возникающие в сеточных методах решения дифференциальных задач, являются разреженными, ленточными и большими. Это означает, во-первых, что порядки N достигают десятков и сотен миллионов, а ненулевые элементы сосредоточены в некоторой полосе ширины m около главной диагонали, причем величина m и число ненулевых элементов в каждой строке не зависят от N. Дискретизированные алгебраические системы, возникающие из аппроксимации интегральных уравнений (определенных на границе или в объеме расчетной области) являются, наоборот, плотными, но зачастую имеют специальные структурные свойства (например, являются теплицевыми, квази- или блочно-теплицевыми).

Вычислительная линейная алгебра – хорошо продвинутая математическая дисциплина и содержит большое количество алгоритмов для решения задач с самыми разными типами матриц: вещественными и комплексными, квадратными и прямоугольными, эрмитовыми и неэрмитовыми, положительно определенными и знаконеопределенными. Имеется также соответствующее обширное программное обеспечение в виде библиотек или прикладных пакетов, как свободно распространяемых в Интернете, так и коммерческих.

Алгебраические задачи – "узкое горлышко" математического моделирования, поскольку потребляемые вычислительные ресурсы нелинейно растут с увеличением порядка N. Поэтому здесь особенно актуально распараллеливание алгоритмов на МВС с общей, разделенной и гибридной памятью.

Существенным моментом программного обеспечения для разреженных матриц является оптимизация кода, поскольку применение сжатых форматов данных позволяет кардинально сокращать объем хранимой информации, но существенно усложняет реализацию доступа к матричным элементам, что особенно критично при многоуровневой неоднородности кэша и оперативной памяти.

Методы оптимизации и решения нелинейных уравнений. Алгоритмы оптимизации решения обратных задач, сводящихся к проблеме условной минимизации функционала, являются бурно развивающейся в последние десятилетия областью вычислительной математики. Здесь развиты модификации методов множителей Лагранжа и внутренних точек, являющихся развитием подходов со штрафными функциями, использование доверительных интервалов для регулировки последовательности шагов, а также варианты алгоритмов Ньютона и последовательного квадратичного программирования для решения возникающих на промежуточных этапах СНАУ.

В данной тематике имеется много актуальных и далеко не решенных вопросов, связанных с поиском глобального минимума и оптимального управления, применения методов теории возмущений и сопряженных уравнений, нахождения градиентов функционалов или функций чувствительности к вариациям исходных данных. Зачастую постановки требуют штучного исследования устойчивости и корректности задачи для поиска соответствующего регуляризационного подхода. К этой же области можно отнести изучение нелинейных динамических систем, связанных с эффектами бифуркаций, самоорганизации и хаоса, странных аттракторов и т.д., где зачастую еще остаются открытыми методологические принципы математического моделирования.

Постобработка и визуализация результатов. Непосредственная реализация наукоемких алгоритмов в многомерных задачах может составлять отнюдь не главную долю общего вычислительного процесса, поскольку анализ получаемых результатов требует их презентабельной визуализации с предварительной обработкой сеточных функций, что требует выполнения ресурсоемких технологических операций по формированию сечений, изолиний и изоповерхностей, различных графиков с возможной анимацией многоцветных изображений. Такие технологические проблемы особенно актуальны в системах принятия решений по результатам математического моделирования.

3 Концепция и архитектура базовой системы моделирования

Рассмотренные технологические этапы решения больших задач естественным образом отображаются на архитектуру и состав основных функциональных и информационных компонент БСМ, изображенных на рис. 1. Приведенная схема фактически представляет последовательность действий, осуществляемых при изучении какого-то технического процесса или природного явления.

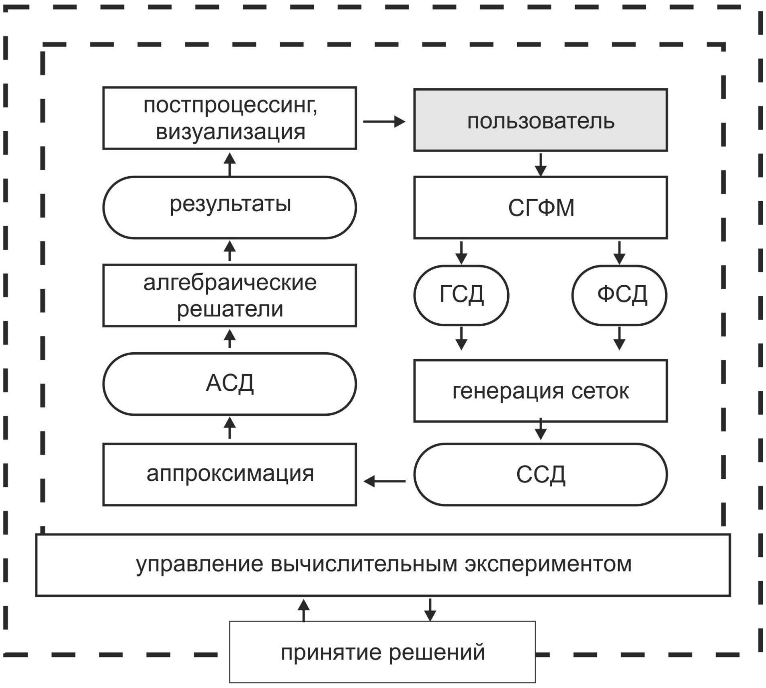

Рис. 1: Структура функциональных и информационных компонент БСМ

Для конечного пользователя (“модельера”) операционное окружение представляется, с одной стороны, входным интерфейсом, т.е. системой геометрического и функционального моделирования (СГФМ), включающей средства управления вычислительным процессом и принятия решений. Выходной же интерфейс обеспечивается обработкой и визуализацией результатов расчета (постпроцессинг), на основе анализа которых могут меняться исходные данные, математическая модель или алгоритмы и формироваться новые вычислительные сеансы. Получаемые в итоге выполнения этого этапа информационные массивы ГСД и ФСД содержат, как правило, до нескольких сот объектов, в силу чего при его параллельной реализации на МВС формируемые данные целесообразно копировать в памяти всех процессоров, во избежание излишних коммуникационных потерь.

Библиотека сеточных генераторов (условное название – DELAUNAY), говоря формальным языком, является преобразователем данных ГСД+ФСД→ССД и, как уже отмечалось, служит интегратором программных процедур от различных разработчиков, на основе гибкой системы внутренних интерфейсов. Актуальную роль играет библиотека конечно-объемных и конечно-элементных "аппроксиматоров", которая на основе ССД, ГСД и ФСД формирует алгебраическую структуру данных.

На основе АСД функционирует библиотека алгебраических решателей KRYLOV, концепция и общая структура которой описана в [10].

Решение сложной математической проблемы организуется в общем случае по многократно вложенным циклам: численное интегрирование нестационарных задач по временным шагам, проведение на каждом шаге итераций по нелинейности, если таковая присутствует, расщепление по физическим процессам или/и по пространственным переменным, варьирование исходных данных в многовариантных расчетах или в обратных задачах. На всех этих уровнях могут строиться свои тактики и стратегии распараллеливания, но в любом случае одним из наиболее критичных моментов является параллельная реализация СЛАУ. Здесь главный подход заключается в алгебраической декомпозиции, которая на дискретном представлении отражает геометрическое разделение сточной расчетной области со сбалансированным разделением подобластей по соответствующим процессорам. Данная проблема представляет собой специальную задачу на графах с высокой вычислительной сложностью, и здравый компромисс заключается в поиске ее приближенного решения.

Следует сказать, что структура АСД с обычными сжатыми матричными форматами ориентирована на экономию памяти при решении больших разреженных СЛАУ и слабо "помнит" топологические свойства исходной расчетной области, которые могут иметь значительное влияние на эффективность метода декомпозиции. Поэтому разделение областей целесообразно начинать еще на этапе построения сетки, что должно отражаться в формировании и сеточных, и алгебраических структур данных.

Оптимизация массивного распараллеливания в конкретных случаях требует анализа коэффициентов ускорения и использования вычислительной системы

$$S_p = T_1/T_p, \quad E_p = S_p/p,$$

где T_p – время решения задачи (или реализации алгоритма) на p процессорах. К сожалению, сделать это непросто в силу отсутствия адекватных моделей машинных вычислений на МВС с иерархической памятью. При оценке эффективности производительности многопользовательского вычислительного центра ситуация значительно усложняется и требует учета распределения глобальных ресурсов между потоками задач.

В соответствии с концепцией IESP, БСМ разрабатывается как свободно распространяемое программное обеспечение (Open Source), развиваемое, поддерживаемое и адаптируемое к новым компьютерным платформам в течение своего длительного жизненного цикла. На основе базовой системы моделирования могут разрабатываться ППП, в том числе коммерческие, для конкретных предметных областей. Функционирование БСМ естественно вписывается в парадигмы интеллектуального программного сервиса (SaaS – Software as Service) на суперцентрах коллетивного пользования (Data Center) в режиме сетевого доступа с организацией "облачных" вычислений (Cloud

Computing). Данная естественным образом наметившаяся стратегия математического моделирования является осуществлением перехода прикладного программного обеспечения от кустарного и ремесленнического способа производства к индустриальному, который дебатировался еще на семинарах Н.Н.Яненко и давно стал реальностью для профессиональных продуктов (операционные системы, компиляторы и т.д.).

Список литературы

[1] Яненко Н.Н., Коновалов А.Н. Современные проблемы математической физики и вычислительной математики.-Новосибирск: Наука, 1982.

[2] Яненко Н.Н., Рычков А.Д. Актуальные проблемы прикладной математики и математического моделирования.-Новосибирск: Наука, 1982.

[3] Ильин В.П. Экзапроблемы математического моделирования.//Вестник ЮУрГУ, сер. "Математическое моделирование и программирование", вып. 6, N 35(211), 2010, 28-39.

[4] Ильин В.П. Что такое вычислительная наука?//Вестник РАН, т. 80, N 4, 2010, 329-336.

[5] Ильин В.П. Параллельные алгоритмы для больших прикладных задач: проблемы и технологии.//Автометрия, N 2, 2007, 3-21.

[6] Ильин В.П. Геометрическое и функциональное моделирование в задачах математической физики.–Новосибирск, Вычислительные технологии, т. 6, ч. 2, 2001, 315-321.

[7] Ушаков Д.М. Введение в математические основы САПР.–Новосибирск, изд. комп. Ледас, 2006.

[8] Ильин В.П. Методы конечных разностей и конечных объемов для эллиптических уравнений.–Новосибирск, изд. ИВМиМГ СО РАН, 2001.

[9] Ильин В.П. Методы и технологии конечных элементов.–Новосибирск, изд. ИВМиМГ СО РАН, 2007, 370с.

[10] Бутюгин Д.С., Ильин В.П., Ицкович Е.А., Петухов А.В., Кныш Д.В. Krylov: библиотека высокопроизводительных алгоритмов для решения разреженных СЛАУ.–Труды 13-й Всероссийской конференции "Современные проблемы математического моделирования". – Ростов, изд. ЮФУ, 2009, 110-128.