Многоцелевые алгоритмы: структура, программная реализация и приложения

Исламов Г.Г., Исламов А.Г.

Аннотация

Рассматриваются основы математической теории специализированного процессора высокопроизводительных вычислений для многоцелевых алгоритмов.

Abstract

We consider the foundations of the mathematical theory of special high-performance computing processor for multi-purpose algorithms.

Введение

Высокопроизводительные вычисления, как правило, ассоциируются с применением высокопроизводительной вычислительной техники. Однако это не единственно возможный путь ускорения расчётов. Мы считаем, что огромные резервы увеличения производительности вычислений можно обнаружить при правильном выборе алгоритма для конкретных типов задач, когда в процессе вычислений одновременно удаётся решить весь класс проблем, относящихся к данной задаче. Такой алгоритм мы называем многоцелевым. Ярким примером многоцелевого алгоритма является алгоритм Горнера деления многочлена от одной переменной на линейный многочлен. В ходе его исполнения для многочлена, заданного координатами в степенном базисе, можно одновременно получить переразложение по базисам Тейлора, Ньютона и Эрмита. Менее тривиальные примеры многоцелевых алгоритмов будут приведены ниже.

Быстродействие современных компьютеров и развитые средства межпроцессорной обработки данных предъявляют повышенные требования к форме представления арифметических данных и точности арифметических вычислений. Некоторая опасность таится в попытке организации высокопроизводительных вычислений на базе машинной арифметики, реализованной в стандартных системах программирования C++ и CUDA [1]. Дело в том, что принятая в них форма представления отрицательных целых чисел сужает диапазон целых положительных чисел, представимых фиксированным числом двоичных разрядов. Кроме того, лежащий в основе представления вещественных чисел стандарт IEEE 754 по точности вычислений значительно уступает машинной арифметике с 40 двоичными разрядами под мантиссу при одинарной точности и 80 двоичными разрядами при двойной точности вычислений, реализованной в отечественной ЭВМ БЭСМ-6 ([2], с. 24). Большой объём вычислений, как правило, сопровождается значительной абсолютной и относительной погрешностью конечного результата. Поэтому, эффективные высокопроизводительные вычисления можно организовать лишь на базе специализированных вычислительных процессоров [3]. Эмуляция таких процессоров программными и микропрограммными средствами возможна и представляет собой типичное решение, однако более перспективной, нам представляется, разработка соответствующей интегральной схемы на СБИС.

В настоящей работе рассматриваются основы математической теории спецпроцессора высокопроизводительных вычислений, реализующего многоцелевые алгоритмы для определённого класса практических задач.

1 Фундаментальная структура числовой информации

Теория представления алгебраических структур, обладающих важными для приложений свойствами, приводит нас к пониманию фундаментальной роли кольца матриц над заданным числовым полем, в общем случае телом (конечномерной алгебры с делением). Как известно ([4], с. 464), n- мерная ассоциативная алгебра A над полем P изоморфна некоторой подалгебре в $M_k(P)$, где $k \le n+1$. Здесь $M_k(P)$ есть кольцо матриц порядка k с элементами из поля P. Теорема Веддербарна-Артина ([5], с. 122) доставляет примеры колец с единицей, изоморфных прямой сумме конечного числа полных колец матриц над некоторыми телами. Знаменитая теорема Фробениуса утверждает, что над полем вещественных чисел R существует лишь три конечномерных тела: R, C и H ([5], с. 202). Здесь C - поле комплексных чисел, изоморфное алгебре квадратных матриц вида $\begin{pmatrix} a & b \\ -b & a \end{pmatrix} \approx a+bi$, где $a,b \in R$; H - алгебра кватернионов Гамильтона, изоморфная алгебре квадратных матриц вида $\begin{pmatrix} u & v \\ -\bar{v} & \bar{u} \end{pmatrix} \approx \alpha_0 + \alpha_1 i + \alpha_2 j + \alpha_3 k$, где $u = \alpha_0 + \alpha_1 i, v = \alpha_2 + \alpha_3 i \in C$ ([6], с. 137). Теория линейных представлений конечных групп ([4], с. 347) указывает на важную роль кольца матриц над полем вещественных чисел в изучении свойств симметрии окружающего нас мира. В теории формальных языков ([7], с. 30) показывается эквивалентность для языка L следующих четырёх условий: L представим; L матрично представим; L представим конечным детерминированным автоматом; L является языком типа 3 в иерархии Хомского. Здесь используется морфизм, отображающий совокупность всех слов над заданным алфавитом Σ в мультипликативный моноид квадратных матриц. Другой пример применения матричной техники к изучению конечных распознавателей и регулярных алгебр приводится в курсе лекций ([8], с. 339) и монографии ([9], с. 127).

Сказанное выше позволяет нам при описании числовой информации моделей реального мира ограничиться агрегатами из матричных структур над полем вещественных чисел.

2 Универсальная операция над матричными структурами

Стандартные бинарные операции в алгебре матриц $M_n(R)$ порядка n над полем R вещественных чисел дополним унарной операцией, которая в работе [10] названа *универсальной операцией над матричными структурами*. Рассмотрим две группы переменных $(x_1,...,x_n)$ и $(y_1,...,y_n)$. Линейное отображение $y_i = \sum_{i=1}^{n} a_{ij} x_j, i=1,...,n$ и ненулевой элемент a_{kl} порождают новое линейное отображение группы переменных $(x_1,...,x_{l-1},y_k,x_{l+1},...,x_n)$ в группу переменных $(y_1,...,y_{k-1},x_l,y_{k+1},...,y_n)$ по формулам

$$\begin{cases} x_l = \dfrac{1}{a_{kl}} y_k + \sum\limits_{\substack{j=1 \\ j \neq l}}^{n} \dfrac{-a_{kj}}{a_{kl}} x_j, \\ y_i = \dfrac{a_{il}}{a_{kl}} y_k + \sum\limits_{\substack{j=1 \\ j \neq l}}^{n} \left(a_{ij} - \dfrac{a_{il} \cdot a_{kj}}{a_{kl}} \right) x_j,\ i \neq k. \end{cases}$$

Обозначая через b_{ij} элементы матрицы нового линейного отображения, получим формулы универсальной операции над матричными структурами

$$b_{ij} = \frac{1}{a_{kl}} \begin{cases} 1, & \text{если } i=k \text{ и } j=l, \\ a_{il}, & \text{если } j=l, \\ -a_{kj}, & \text{если } i=k, \\ a_{ij} \cdot a_{kl} - a_{il} \cdot a_{kj}, & \text{если } i \neq k \text{ и } j \neq l. \end{cases}$$

Ряд свойств этой операции рассмотрен в работе [11]. Ниже будут представлены многоцелевые алгоритмы, в основе которых лежит эта универсальная операция.

3 Тетрада как форма представления вещественного числа

Вид формул универсальной операции над матричными структурами указывает на целесообразный путь её программной реализации. Как известно, операция деления является наиболее затратной при выполнении машинной арифметики [12]. Значит, необходима такая форма представления числа, которая исключала бы деление при получении промежуточных результатов. Этого можно достичь за счёт введения упорядоченной пары (α,β) ([13], с. 119), которая представляет отношение вещественных чисел α/β. С другой стороны, операция вычитания близких чисел приводит к значительному повышению относительной погрешности вычислений. Поэтому вычитание также следует исключить при получении промежуточных результатов за счёт введения дополнительной упорядоченной пары $<\alpha,\beta>$ ([13], с. 120), обозначающей разность $\alpha-\beta$. В любой реализации множество компьютерных чисел конечно, арифметические операции коммутативны, но не ассоциативны и не дистрибутивны. Этот недостаток устраняется за счёт введения класса *приближённых чисел* как совокупности пар вида $\{a,\delta_a\}$, где a есть компьютерное число, а δ_a - переменная с известной границей изменения. Каждая такая пара представляет вещественное число вида $A=a(1+\delta_a)$. Следует помнить, что результатом вычислений на компьютере является не компьютерное число, а приближённое число с указанной границей изменения переменной δ_a относительной погрешности. Для положительных чисел $A=a(1+\delta_a)$ и $B=b(1+\delta_b)$, их суммы $C=A+B$ и произведения $D=A\cdot B$ имеем $C=a\oplus b(1+\delta_c)$ и $D=a\otimes b(1+\delta_d)$, где $a\oplus b$ и $a\otimes b$ есть компьютерные числа и можно указать вполне определённые границы для переменных $\delta_c=\delta_a\dfrac{a}{a+b}+\delta_b\dfrac{b}{a+b}+\delta_+$ и $\delta_d=\delta_a+\delta_b+\delta_*$. Всё это указывает на то, что при программной реализации универсальной операции следует ограничиться машинным представлением только положительных чисел. Структура $(<\alpha_1,\beta_1>,<\alpha_2,\beta_2>)$, обозначающая число вида $\dfrac{\alpha_1-\beta_1}{\alpha_2-\beta_2}$ позволяет при работе с компьютерными числами применять только две операции: сложение и умножение положительных чисел.

Воспользуемся встроенным векторным типом float4 CUDA-расширения языка C [1] для программной реализации введённой тетрады представления чисел и арифметических операций над этими тетрадами. Ниже приводится функция операции вычитания. Аналогичные функции могут быть выписаны для операций сложения, умножения и деления тетрад.

__device__ float4 operator – (const float4 & a**, const float4 &** b**);**

{ $float4\ c;$

$$c.x = a.x \cdot b.z + a.y \cdot b.w + a.z \cdot b.y + a.w \cdot b.x;$$

$$c.y = a.y \cdot b.z + a.x \cdot b.w + a.z \cdot b.x + a.w \cdot b.y;$$

$$c.z = a.z \cdot b.z + a.w \cdot b.w;$$

$$c.w = a.w \cdot b.z + a.z \cdot b.w;$$

$return\ c;$

}

4 Примеры многоцелевых алгоритмов

Здесь будет представлен класс алгоритмов, которые в ходе своего исполнения позволяют достичь нескольких целей, относящихся к определённой проблематике. В основе всех алгоритмов класса лежит универсальная операция над матричными структурами. Она применяется к агрегату из матриц, отвечающих решаемой проблеме, согласно правилу выбора *разрешающего элемента*. Представленные в докладе алгоритмы завершаются через конечное число шагов применения универсальной операции. В качестве примера рассмотрены симметричная задача линейного программирования, балансовая модель экономики, управление спектром дискретных процессов методом минимальной обратной связи, одномерное и двумерное сглаживание экспериментальных данных. Эти результаты служат прямым подтверждением тезиса проф. В.А. Вышинского, что основой вычислительных машин будущего поколения должна стать *универсальная алгоритмическая матрично-алгебраическая система* [14]. В новой машинной технологии обработка сложных структур данных сведётся к обработке матриц, при этом будет использоваться математический аппарат регулярного матричного представления для реализации отвечающей решаемой проблеме числовой алгебры. Аппаратурная реализация принципиально нового языка высокого уровня позволит существенно упростить и удешевить программное обеспечение создаваемой машины будущего поколения. При этом, удастся заменить маломощную машинную арифметику машинной алгеброй и, тем самым, добиться устранения противоречия, возникшего между интеграцией вычислительных схем в аппаратуре вычислительных систем и отсутствием адекватной интеграции в вычислительном технологическом процессе. Кроме того, откроется прямой путь повышения производительности вычислений не за счёт повышения тактовой частоты генератора и последующего преодоления «теплового барьера», а за счёт извлечения преимущества из сверхбольшой интеграции.

Программная реализация многоцелевых алгоритмов указанного класса представлена для гибридных вычислительных систем, поддерживающих технологии многопоточного параллельного программирования OpenCL и CUDA. Один из этих алгоритмов одновременно находит решение прямой и двойственной задачи линейного программирования, заданной в симметричной форме, а также области устойчивости решений этих задач. Другой алгоритм для балансовой модели материальных и финансовых ресурсов находит одновременно объёмы и цены производимой продукции, матрицу полных затрат и отвечает на вопрос о продуктивности балансовой модели. Третий алгоритм для матриц простой структуры одновременно находит подобную матрицу Фробениуса, преобразующую матрицу и обратную к ней, что позволяет строить одноранговые возмущения матриц простой структуры с заданными спектральными свойствами.

4.1 Симметричная задача линейного программирования

Рассмотрим однородную цепь Маркова с переходной матрицей L порядка n, в которой строка $x(t)=(x_1(t),\ldots,x_n(t))$ представляет безусловные вероятности системы в фиксированный момент времени $t=0,1,\ldots,N$. Эволюция системы описывается равенством $x(t+1)=x(t)\,L$, где $t=0,1,\ldots,N-1$ и начальное состояние $x(0)$ выбирается из условия выполнения конечной системы неравенств

$$x(t_k)\,c^k \geq \beta_k\,, k=1,\ldots,m. \tag{1}$$

Здесь c^k представляет собой столбец «доходов», которые обеспечивает система в момент времени $t_k \in \{1,\ldots,N\}$ $(t_{k-1}<t_k, t_0=0)$ и все «пороговые» значения $\beta_k > 0$. Нас интересует случай, когда величины n и m достаточно велики $(n \approx 2^{16}, m \approx 2^{10})$. Условия разрешимости относительно начальных безусловных вероятностей $x(0)=(x_1(0),\ldots,x_n(0))$ системы формулируются в терминах следующих двух расчётных задач большой размерности.

Первая задача, требующая высокой точности вычислений, состоит в нахождении m столбцов $y^k(t)=(y_1^k(t),\ldots,y_n^k(t))^T$ - решений сопряжённой разностной системы $y(t+1)=Ly(t), t=1,\ldots,N$ с начальным условием $y(0)=c^k\,(k=1,\ldots,m)$.

Вторая расчётная задача состоит в отыскании максимального значения следующей экстремальной задачи

$$\sum_{k=1}^{m}\beta_k\,\lambda_k \to \max,\ \sum_{k=1}^{m} y^k(t_k)\,\lambda_k \le e, \lambda_k \ge 0, =1,\ldots,m, \qquad (2)$$

где e - столбец, состоящий из n единиц (сравнение векторов-столбцов покоординатное). Аналогично работе [15] показывается, что задача (1) разрешима тогда и только тогда, когда максимальное значение целевой функции задачи (2) больше нуля и не превосходит единицы и после нормировки решение $q=(q_1,\ldots,q_n)$ двойственной к (2) экстремальной задачи

$$q\,e \to \min, q_i \ge 0, i=1,\ldots,n, q\,y^k(t_k) \ge \beta_k, k=1,\ldots,m \qquad (3)$$

даёт одно из начальных состояний $x(0)=q$, обеспечивающее выполнение системы неравенств (1).

Для нахождения решений этой пары симметричных двойственных задач и их областей устойчивости мы применяем параллельную реализацию алгоритма симплекс-метода, описанного в [16]. Первая симплекс-таблица предложенного в [16] алгоритма одновременного решения задач (2) и (3) имеет вид

$$\begin{pmatrix}
 & & & \lambda_1 & \cdots & \lambda_m & & & \\
 & & & q_{n+1} & \cdots & q_{n+m} & & & \\
 & & 0 & -\beta_1 & \cdots & -\beta_m & & & \\
q_1 & \lambda_{m+1} & 1 & y_1^1(t_1) & \cdots & y_1^m(t_m) & 1 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
q_n & \lambda_{m+n} & 1 & y_n^1(t_1) & \cdots & y_n^m(t_m) & 0 & \cdots & 1 \\
 & & & 1 & \cdots & 0 & & & \\
 & & & \vdots & \ddots & \vdots & & & \\
 & & & 0 & \cdots & 1 & & &
\end{pmatrix},$$

где справа и снизу приписаны единичные матрицы размеров соответственно n и m. В пособии [16] указано правило преобразования симплекс-таблиц такой структуры, условие окончания алгоритма, правило нахождения экстремальных значений, оптимальных планов и их областей устойчивости, а также рассмотрены иллюстративные примеры.

Обозначим через P матрицу порядка n, стоящую справа в последней симплекс-таблице, а через Q матрицу порядка m, стоящую снизу в этой таблице. Пусть, далее, $\Delta=(\Delta_1,\ldots,\Delta_m)$ - строка, стоящая в последней симплекс-таблице в положении строки $(-\beta_1,\ldots,-\beta_m)$ первой симплекс-таблицы, а нулевые координаты этой строки имеют номера $j_1,\dots,j_t\,(1\le t\le m)$; $U=(u_1,\dots,u_n)^T$ - столбец, стоящий в последней симплекс-таблице в положении столбца из единиц e первой симплекс-таблицы, а нулевые координаты этого столбца имеют номера $i_1,\dots,i_s\,(1\le s\le n)$. Заметим, что величины Δ и U определяют соответственно решения двойственной задачи (3) и прямой задачи (2), а экстремальное значение целевых функций этих задач находится в последней симплекс-таблице на месте «жирного» нуля в первой симплекс-таблице.

Имеет место следующее утверждение, частная формулировка которого дана в учебном пособии [16] для невырожденных задач линейного программирования.

Теорема 1. *Область устойчивости прямой задачи (2) даётся системой неравенств*

$$\delta\beta\cdot Q\le\Delta+\sum_{k=1}^{s}\nu_k\,\tilde{y}_{i_k},$$

где $\delta\beta=(\delta\beta_1,\dots,\delta\beta_m)$, *а* $\nu_k\ge 0, k=1,\dots,s$ *произвольно заданы и* $\tilde{y}_{i_k}$ *есть* i_k *–я строка основной матрицы последней симплекс-таблицы. Область устойчивости двойственной задачи (3) описывается системой неравенств*

$$P\cdot\delta e+U\ge\sum_{k=1}^{t}\mu_k\,\tilde{y}^{j_k},$$

где $\delta e=(\delta e_1,\dots,\delta e_n)^T$, *а* $\mu_k\ge 0, k=1,\dots,t$ *произвольно заданы и* $\tilde{y}^{j_k}$ *есть* j_k *–ый столбец основной матрицы последней симплекс-таблицы.*

4.2 Балансовая модель производства

Метод балансов материальных и финансовых ресурсов [17] служит для учёта, контроля и планирования их воспроизводства и расходования в ходе любой экономический деятельности. Рассматриваемая нами математическая модель, лежащая в основе этого метода, состоит из двух балансовых уравнений

$$x-Ax=y,\quad p-pA=q.$$

Первое (основное) уравнение описывает баланс в натуральном выражении, а второе (двойственное) уравнение - баланс в стоимостном выражении.

Показателями модели служат $n \times n$ - матрица A неотрицательных коэффициентов норм затрат, где величиной n задаётся число видов продукции, учитываемой в модели. Компоненты вектора-столбца x длины n содержат объёмы рассматриваемой продукции и тем самым могут быть лишь неотрицательными. Вектор-столбец y называется вектором экспорта-импорта. Положительные компоненты этого вектора отвечают конечной продукции и соответствуют экспорту, тогда как отрицательные компоненты характеризуют объёмы импорта. Величина Ax описывает затраты в сфере производства. Компоненты вектора-строки p длины n содержат цены на производимую продукцию и могут быть лишь неотрицательными. Вектор-строка q называется стоимостным вектором единичной продукции. Положительные компоненты этого вектора отвечают прибыли и характеризуют доходную продукцию, тогда как отрицательные компоненты свидетельствуют об убыточности производства соответствующих видов продукции и определяют убытки от реализации единичной продукции. Величина pA описывает себестоимость производимой в экономике продукции.

Задача описания экономики на основе метода балансов включает в себя ряд проблем. Первые две проблемы состоят в численном решении основного и двойственного балансового уравнения. Третья проблема, которую мы относим к категории наиболее важных, заключается в оценке относительной погрешности результата x, p при вариации исходных величин модели: матрицы A и векторов y и q. Как известно, в этой проблеме ведущая роль принадлежит числу обусловленности [18] $cond_\infty B = \|B\|_\infty \|B^{-1}\|_\infty$ матрицы $B = I - A$. Здесь I - единичная матрица порядка n, $\|B\|_\infty = \max\limits_{1 \le i \le n} \sum\limits_{j=1}^{n} |\delta_{ij} - a_{ij}|$ - норма матрицы, согласованная с чебышевской нормой вектора $\|x\|_\infty = \max\limits_{1 \le j \le n} |x_j|$, δ_{ij} - символ Кронекера. Отметим ещё одну проблему, имеющую непосредственное отношение к нашей задаче. Это вычисление определителя $\det B$ и обратной матрицы B^{-1} полных затрат. Наконец, немаловажное значение имеет проблема проверки продуктивности балансовой модели [17].

Мы показываем, что *универсальная операция над матричными структурами* [10] позволяет одновременно решить все указанные выше проблемы нашей задачи в ходе излагаемого ниже алгоритма для гибридных вычислительных систем, которые состоят из многоядерного центрального процессора и многопроцессорных графических устройств компании NVIDIA, образующих параллельную вычислительную архитектуру CUDA [1]. Эта операция естественно реализуется на архитектуре CUDA [19], а сам алгоритм требует интенсивного взаимодействия с центральным процессором лишь на этапе передачи матрицы A, векторов y, q и выдачи конечного результата x, p.

Начнём с описания информационной структуры нашего алгоритма. Начальная таблица имеет вид

		p_{n+1}	p_{n+2}	…	p_{2n}	
	1	$-q_1$	$-q_2$	…	$-q_n$	
x_{n+1}	y_1	b_{11}	b_{12}	…	b_{1m}	p_1
x_{n+2}	y_2	b_{21}	b_{22}	…	b_{2n}	p_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
x_{2n}	y_n	b_{n1}	b_{n2}	…	b_{nn}	p_n
		x_1	x_2	…	x_n	

Выделенная цветом часть информационной таблицы является числовой и в дальнейшем будет изменяться при выполнении универсальной операции над матричной структурой. Текущее её значение будет обозначаться буквой D. Матрица D окружена одномерными столбцовыми и строчными массивами x-переменных и p-переменных, положения которых будут меняться при пересчёте этой матрицы согласно правилу, учитывающему положение *разрешающего элемента* числовой таблицы. Разрешающие элементы не равны нулю и выбираются последовательно на основной диагонали позиции матрицы B, которая учитывается в начальной информационной таблице.

Опишем теперь правило пересчёта текущей информационной таблицы по выбранному разрешающему элементу $\gamma_k = d_{kk}, k = 1, \dots, n$. Переменные x_k и x_{n+k}, соответственно p_k и p_{n+k}, меняются местами, а остальные переменные остаются на своих местах. Элементы числовой матрицы D пересчитываются по правилу (универсальной матричной операции): $d_{00} := d_{00} * \gamma_k$,

$$d_{ij} := \frac{1}{\gamma_k} \begin{cases} 1, \text{ если } i = k, j = k; \\ d_{ij}, \text{ если } i = k; \\ -d_{ij}, \text{ если } j = k; \\ d_{ij} * \gamma_k - d_{ik} * d_{kj}, \text{ если } i \neq k, j \neq k. \end{cases}$$

Согласно критерию Хокинса-Саймона [17] матрица A продуктивна тогда и только тогда, когда главные последовательные миноры $b_1, \dots, b_n$ матрицы B больше нуля. Нетрудно заметить, что дополненная единицей с начала эта последовательность $1, b_1, \dots, b_n$ при делении текущего элемента на предыдущий элемент даёт последовательность $\gamma_1, \dots, \gamma_n$ разрешающих элементов нашего алгоритма. Это значит, что положительность всех разрешающих элементов алгоритма есть необходимое и достаточное условие продуктивности матрицы норм затрат.

Далее, очевидно, что произведение всех разрешающих элементов равно определителю матрицы B, который будет расположен в нулевой строке и нулевом столбце заключительной таблицы D.

Заметим также, что в последней таблице на месте матрицы B мы будем иметь матрицу полных затрат B^{-1}, которая позволит нам вычислить число обусловленности матрицы B.

Наконец, на позиции компонент векторов экспорта-импорта y и стоимостного вектора единичной продукции q мы найдем соответственно объёмы x и цены p производимой продукции.

Если в процессе алгоритма текущий диагональный элемент матрицы D равен нулю, то следует придерживаться стратегии выбора разрешающего элемента, следуя работе [11].

4.3 Управление финансовым рынком

Ниже приводится очередное применение универсальной операции над матричными структурами при расчёте вектора-столбца гонораров и налоговых отчислений, а также вектора-строки приоритетов в задаче государственного управления финансовым рынком. Финансовый рынок служит важной, но не определяющей, составляющей экономики государства. Заложенные исторически правила его функционирования не являются совершенными, что показывает затянувшийся мировой финансовый кризис. Обладая ограниченным в объёме универсальным ресурсом, выражающим всеобщий эквивалент труда, финансовый рынок искусственно создаёт предпосылки для безграничного печатания денег как необходимого условия своего существования. Это приводит к хронической инфляции, последствия которой оказываются катастрофическими как для населения государства, так и для отдельных отраслей экономики. Известно, что финансовый рынок обязан стимулировать инновации в сфере производства материальных благ и воспроизводства интеллектуальных ресурсов, финансировать планомерное развитие потребностей людей и бесперебойное потребление материальных и информационных ресурсов. Это значит, что требуется эффективное управление финансовым рынком со стороны государства. Предлагаемая нами математическая модель государственного управления финансовым рынком показывает, что финансовый рынок может находиться в одном из трёх состояний: *деградация, кризис и стабильность. Первое* состояние свидетельствует о крайне неэффективном управлении и отвечает процессу разорения финансового рынка. *Второе* состояние связано с необходимостью включения печатного станка для обеспечения экспоненциального роста денежной массы в зависимости от номера временного периода. *Третье* состояние соответствует степенному росту денежной массы. Всё это есть следствие иерархической структуры финансового рынка и проявление закона его эволюции, которые задаются матрицей A процентов взаимных платежей участников финансового рынка и правилом внешнего управления со стороны государства.

Структура матрицы A описана в работе [20]: имеется m участников финансового рынка, образующих иерархическую структуру с выделенным корнем. Единственный путь, ведущий из этого корня к любой другой вершине, определяет полное имя участника этого рынка. Каждый из участников финансового рынка обладает уникальной способностью увеличивать за указанный период времени (час, неделя, месяц) имеющийся у него капитал на определенное число процентов, не зависящее от объема капитала и порядкового номера периода, но своё для каждого участника. В конце любого периода каждый родитель перечисляет на счет своего дочернего участника определенный процент (для каждого дочернего участника свой) от той суммы, которой родитель располагал в начале временного периода. В это же время все участники финансового рынка (включая и корневого) перечисляют на счёт корневого участника сумму, которая была у них на начало периода, и весь доход, который был получен в течение текущего периода от использования этой суммы. Принятая в модели иерархия финансового рынка обеспечивает его мобильность и управляемость. Однако эти положительные качества могут быть использованы в разрушительных целях, придавая финансовому рынку уязвимость при неблагоприятных внешних воздействиях: интервенция и вывоз капитала, и т. д.

Рассматриваемая модель эволюции финансового рынка описывается дискретным управляемым процессом $x[k+1]=Ax[k]+u[k],\ k=0,1,2,\ldots,$где управление со стороны государства формируется по правилу $u[k]=p\cdot qx[k]$. Здесь $p=(p_1,\ldots,p_m)^T$ есть столбец гонораров и штрафов, установленных государством. Положительная компонента p_i выражает величину гонорара i – го участника финансового рынка, тогда как абсолютное значение отрицательной компоненты p_i определяет величину налога, которым облагается участник i в конце планового периода k. Строка $q=(q_1,\ldots,q_m)$

содержит государственные приоритеты, а столбец $x[k]=(x_1[k],...,x_m[k])^T$ - капиталы участников рынка.

В силу иерархической структуры возможна такая нумерация участников финансового рынка, при которой номер дочернего участника больше номера родителя. Мы предполагаем, что такая нумерация уже имеет место. В таком случае матрица A, содержащая проценты перевода денежного капитала среди участников рынка представима в виде суммы нижнетреугольной матрицы с нулями на основной диагонали и одноранговой матрицы. Нетрудно построить подпространства Крылова для этой матрицы. Как известно, каждому базису такого подпространства отвечает фробениусова форма матрицы и отвечающая ей преобразующая матрица, которые могут быть эффективно построены. Можно показать, что каждое ненулевое собственное значение матрицы A имеет геометрическую кратность, равную единице. В силу теоремы двойственности работы [21] за счёт выбора одноранговой матрицы $K=p\cdot q$ можно получить наперёд заданный спектр возмущённой матрицы $B=A+K$ без учёта нуля, и, тем самым, обеспечить заданный темп роста (убывания) капитала каждого из участников финансового рынка. Мы видим, что рассматриваемая модель финансового рынка приводит к задаче модального управления, в которой требуется построить одноранговую матрицу K, гарантирующую заданный темп роста (убывания) капитала. Анализ доказательства теоремы двойственности работы [21] показывает, что можно указать эффективный алгоритм построения такой матрицы. Таким образом, за счёт выбора управляющих векторов p, q можно обеспечить любое из указанных выше трёх состояний финансового рынка.

Ниже описывается начальная информационная структура высокопроизводительного алгоритма построения управляющих векторов. Все вычисления организованы в виде двумерной таблицы, первая из которых имеет вид

$$\begin{array}{cccccc} & & -s_1 & +s_2 & \dots & (-1)^m s_m \\ t_1 & b_1 & d_{11} & d_{12} & \dots & d_{1m} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ t_m & b_m & d_{m1} & d_{m2} & \dots & d_{mm} \end{array}$$

В первой строке находятся s-переменные, которые обозначают коэффициенты характеристического многочлена матрицы A. В первом столбце находятся вспомогательные t-переменные, которые будут обмениваться с s-переменными на каждом шаге применения универсальной операции над матричными структурами. Следующие столбцы исходной таблицы составлены из координат соответственно векторов $A^m e, A^{m-1}e,...,Ae,e$, где e есть порождающий вектор базиса Крылова. Они, без учёта первого вектора, образуют *матрицу перехода* от единичного базиса к базису Крылова. После m-кратного применения универсальной операции получим последнюю таблицу, в первом столбце которой окажутся все s-переменные, во втором столбце будут находиться коэффициенты характеристического многочлена, а остальные столбцы образуют обратную матрицу к матрице перехода.

Построение управляющих векторов p и q мы производим следующим образом. Пусть $G=(A^{m-1}e, A^{m-2}e, \ldots, Ae, e)$ - матрица базиса Крылова с циклическим вектором e длины m, $g=A^m e$. Тогда столбец $h=-G^{-1}g$ даёт нам коэффициенты характеристического многочлена матрицы A. Добавим, справа, к противоположному столбцу (-h) единичные столбцы так, чтобы получилась матрица Фробениуса F характеристического многочлена матрицы A. Непосредственно проверяется справедливость факторизации $A=GFG^{-1}$ матрицы динамической системы через эту матрицу Фробениуса. По заданному многочлену $\prod_{j=1}^{n}(\lambda-\mu_j)$ возмущённой матрицы B построим матрицу Фробениуса $\tilde{F}$ такой же структуры, что и матрица F. Из очевидного разложения $A=G\tilde{F}G^{-1}+G(F-\tilde{F})G^{-1}$ находим, что спектр первого слагаемого состоит из набора $\{\mu_1, \ldots, \mu_m\}$, а разность $F-\tilde{F}$ имеет структуру произведения вектора-столбца на вектор-строку $r\cdot(1,0,\ldots,0)$. Здесь r обозначает вектор-столбец длины m, составленный из разностей коэффициентов характеристических многочленов матриц A и B. Теперь понятно как строить управляющие вектора: в качестве p можно взять $-Gr$, а q положить равным первой строке обратной матрицы G^{-1}.

Ввиду большого объёма вычислений все расчёты производятся на вычислительном устройстве с быстродействующим многоядерным центральным процессором, усиленным системой графических процессоров, поддерживающих технологию многопоточного программирования CUDA.

Поиск циклического вектора e и построение хорошо обусловленной матрицы базиса Крылова G выполняется на центральном процессоре. Однако вычисление столбца h коэффициентов характеристического многочлена матрицы A, обратной матрицы G^{-1} к матрице базиса Крылова, детерминанта $\det G$ и управляющих векторов p и q осуществляется на системе графических процессоров одновременно в ходе многократного применения универсальной операции. Обмен информацией между центральным процессором и графическим устройством происходит лишь на этапе передачи матрицы G с центрального устройства на графическое устройство и выдачи управляющих векторов p и q на центральный процессор.

4.4 Метод наименьших квадратов

Рассмотрим классическую задачу обработки экспериментальных данных. В классе линейных комбинаций $y=\sum_{j=1}^{m}\lambda_j u_j(t)$ заданных вещественнозначных функций $u_1(t),\ldots,u_m(t)$ требуется найти такую функцию, которая по методу наименьших квадратов даёт наименьшую погрешность для наблюдаемой последовательности пар числовых значений (t_i, y_i), $i=1,\ldots,n$. Предполагается, что число результатов наблюдения n не меньше количества пробных функций $u_1(t),\ldots,u_m(t)$. Коэффициенты $\lambda_1,\ldots,\lambda_m$ функции с наименьшей погрешностью доставляют минимум функционалу $\sum_{i=1}^{n}\left(y_i-\sum_{j=1}^{m}\lambda_j u_j(t_i)\right)^2$. Такие коэффициенты существуют всегда. Если же пробные функции образуют систему Чебышева на замкнутом интервале $[a,b]$, содержащем все наблюдаемые значения $t_i, i=1,\ldots,n$, то оптимальные коэффициенты определяются однозначно. В работе [22] рассмотрена задача о нахождении для заданной системы функций $u_1(t),\ldots,u_m(t)$ такого отрезка $[a,b]$, на котором она образует систему Чебышева, т. е. на этом отрезке любая нетривиальная линейная комбинация $\sum_{j=1}^{m}\lambda_j u_j(t)$ имеет не более $m-1$ нуля. После выбора необходимой системы пробных функций оптимальные коэффициенты находятся однозначно из алгебраической системы порядка m. Обозначим через A матрицу порядка $n\times m$ с элементами $u_j(t_i)$, где i - номер строки, а j - номер столбца, а через Y вектор-столбец $(y_1,\ldots,y_n)^T$.

Теорема 2. *Пусть система пробных функций* $u_1(t),\ldots,u_m(t)$ *образует систему Чебышева на отрезке* $[a,b]$, *содержащем все наблюдаемые значения* $t_i, i=1,\ldots,n$. *Тогда матрица* $B=A^T A$ *порядка* m *положительно определена и компоненты вектора-столбца* $\Lambda=B^{-1}A^T Y=(\lambda_1,\ldots,\lambda_m)^T$ *определяют набор коэффициентов, который по методу наименьших квадратов даёт* *наименьшую погрешность для экспериментальной последовательности пар числовых значений* $(t_i, y_i),\ i=1,\ldots,n$.

Для нахождения вектора-столбца Λ мы применяем параллельный вариант алгоритма поиска базисного минора [11] для расширенной матрицы $(B, A^T Y)$ со стратегией последовательного выбора разрешающих элементов на основной диагонали матрицы B.

Список литературы

1. NVIDIA CUDA Programming Guide Version 4.0 // www.nvidia.com.
2. Годунов С.К. Решение систем линейных уравнений. – Новосибирск: Наука, 1980. – 177 с.
3. Байков В.Д., Смолов В.Б. Специализированные процессоры: Итерационные алгоритмы и структуры. – М.: Радио и связь, 1985. – 288 с.
4. Кострикин А.И. Введение в алгебру. – М.: Наука, 1977. – 496 с.
5. Скорняков Л.А. Элементы общей алгебры. – М.: Наука, 1983. – 272 с.
6. Демидов И.Т. Основания арифметики. – М. Учпедгиз, 1963. – 160 с.
7. Саломаа А. Жемчужины теории формальных языков. – М.: Мир, 1986. – 159 с.
8. Кук Д., Бейз Г. Компьютерная математика. – М.: Наука, 1990. – 384 с.
9. Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции. Т. 1. Синтаксический анализ. – М.: Мир, 1978. – 616 с.
10. Исламов Г.Г. Универсальная операция над матричными структурами // Современные проблемы вычислительной математики и математической физики: Международная конференция, Москва, МГУ имени М.В. Ломоносова, 16-18 июня 2009 г. Тезисы докладов. – М.: Издательский отдел факультета ВМК МГУ имени М.В. Ломоносова; Макс Пресс, 2009. – 396 с.
11. Исламов Г.Г., Коган Ю.В. Об одном алгоритме поиска базисного минора матрицы // Вестник Удмуртского университета. – 2006. - № 1: Математика. – С. 63 – 70.
12. Шауман А. М. Основы машинной арифметики. – Л.: Изд-во ЛГУ, 1979. – 312 с.
13. Арнольд И В. Теоретическая арифметика. – М.: Учпедгиз, 1939. – 400 с.
14. Вышинский В.А. Решение фундаментальной проблемы современного развития вычислительной техники // Труды Первой международной конференции «Трёхмерная визуализация научной, технической и

социальной реальности. Кластерные технологии моделирования, Т. 1. Ижевск, УдГУ (4-6 февраля 2009 г.), С. 27-36.

15. Исламов Г.Г., Коган Ю.В. Алгоритм нахождения начального состояния марковского процесса с полиэдральными ограничениями // Вестник Удмуртского университета. – 2007. – № 1: Математика. – С. 67–74.
16. Исламов Г.Г. Принципы оптимальности. Учебное пособие. Ижевск, Изд-во УдГУ, 1998. – 124 с.
17. Ашманов С.А. Введение в математическую экономику. – М.: Наука, 1984. – 296 с.
18. Бабенко К.И. Основы численного анализа. – Москва-Ижевск: РХД, 2002. – 848 с.
19. Исламов Г.Г. Реализация на процессорах с технологией CUDA универсальной операции над матричными структурами // Труды Второй международной конференции «Трехмерная визуализация научной, технической и социальной реальности. Технологии высокополигонального моделирования», Т. 1, Ижевск – УдГУ, 2010. - С. 97-100.
20. Исламов А.Г., Исламов Г.Г., Лукин О Л. Об одной модели финансового рынка. Режим доступа: http://www.iton.mfua.ru/tesis/islamov2.doc (дата обращения: 9.11.2011)
21. Исламов Г.Г. Об управлении спектром динамической системы // Дифференциальные уравнения. – 1987. – Т.23, № 8. – С. 1299-1302.
22. Исламов Г.Г., Исламов А.Г., Лукин О.Л. Об одном применении чебышевской системы функций // Труды XV Всероссийской научно-методической конференции «Телематика’2008» (23-26 июня 2008 г., Санкт-Петербург). – С. 53–54.