# ГИПОТЕЗЫ ВОЕВОДИНА И ПРОБЛЕМЫ СЛОЖНОСТИ СТРУКТУР РЕАЛЬНЫХ АЛГОРИТМОВ ИЗ ЛИНЕЙНОГО КЛАССА

**А.В. Фролов**

Настоящая статья касается ряда тех проблем, что при изучении параллельных структур программ возникали перед исследователями алгоритмов, работавших под руководством академика В.В.Воеводина. Главные результаты, касающиеся работы с параллельными структурами, опубликованы в [1-7], а также в других работах. Однако ряд проблем, касающихся математических основ распараллеливания программ и поднятых ещё в 90-х гг. XX века В.В.Воеводиным, не получил должного освещения в литературе. Отчасти это произошло потому, что часть результатов по данным проблемам имела вид «отрицательных результатов», и уж во всяком случае не приносила какого-то облегчения на пути алгоритмизации процессов анализа параллельных структур. При этом, однако, некоторая их часть уже была озвучена, но не на научных конференциях, а на занятиях со студентами. Более того, учебные вопросы по данной проблематике даже вошли в представленную в [8] систему Сигма – банк вопросов по параллельным вычислениям, используемый в учебном процессе в МГУ, МФТИ и ряде других вузов. Поэтому автор, чувствуя некий долг перед другими исследователями параллельной структуры алгоритмов, считает необходимым изложить эти проблемы и полученные по ним результаты в данной статье.

## Введение: линейный класс и его преимущества

Как известно тем, кто интересуется методикой распараллеливания, разработанной В.В.Воеводиным в его работах, она ориентирована на работу с так называемым *линейным классом программ* (см. [5], стр. 349 или [6], стр. 79). Напомню, что фортран-программа (мы здесь предполагаем, что она записана с помощью Фортрана 77, см. [9]) относится к линейному классу, если выполнены следующие ограничения:
(1) единственным типом исполнительного оператора может быть оператор присваивания, правая часть которого есть арифметическое выражение;
(2) все повторяющиеся операции описываются только с помощью циклов DO; структура их вложенности может быть произвольной; шаги изменения параметров циклов равны +1;
(3) условные операторы перехода идут только «вниз» по тексту, запрещены «побочные» выходы из циклов;
(4) все индексы массивов, границы изменения параметров циклов и условия передач команд управления заданы явно; они являются *линейными функциями* как по параметрам циклов, так и по внешним переменным программы; коэффициенты всех линейных функций являются целыми числами;
(5) внешние переменные программы (размеры массивов и т.п.) всегда целочисленные; вектор их значений принадлежит некоторому целочисленному многограннику.

Несмотря на кажущуюся жёсткость требований, видно, что этот класс может быть легко расширен. Так, требование (1), запрещающее подпрограммы и подпрограммы-функции, обходится контекстной вставкой в месте вызовов этих подпрограмм. Требование (2), запрещающее даже нисходящий перебор значений параметра цикла, обходится заменой параметра, причём *линейной*. Требования на условия в операторах обходятся с помощью замен последовательностей операторов на более сложные, и т.п. Да, не все реальные алгоритмы можно свести к набору фрагментов из линейного класса (наиболее яркими примерами несводимых являются БПФ и другие «быстрые» алгоритмы, техника сдваивания и т.п.), однако даже для ряда нелинейных программ (см. [10]) оказываются применимы некоторые методики, разработанные для линейного класса. Введение линейного класса позволяет довольно далеко продвинуться в анализе графа алгоритма. Причина этого – в **теореме об информационном покрытии** (см. [5], стр. 348-349, доказательство в [13]):

«Для любой линейной программы любой минимальный снизу или сверху граф зависимостей покрывается конечной системой функций, линейных как по пространству итераций, так и по пространству внешних переменных программы. Число этих функций не зависит от значений внешних переменных. В линейном пространстве итераций функции определены на линейных многогранниках. Грани многогранников описываются уравнениями, также линейными как по пространству итераций, так и по пространству внешних переменных».

Говоря попросту, у линейной программы линейный граф, что позволяет его достаточно эффективно находить.

## Параллельная сложность линейных программ

После получения инструментария для нахождения графа алгоритма (V-Ray, см. например, [5], стр. 434) начались поиски как способов его распараллеливания, так и попытки математически оценить параллельную сложность. В частности, в [7] представлен инструмент так называемых линейных развёрток графа алгоритма и даны попытки полиномиально оценить такую важную характеристику графа, как её критический путь. Однако эти конструктивные методы дают только лишь оценки сверху. Причиной этого является тот факт, что *до сих пор не определён тот класс функций, в котором следует искать формулы для критического пути графа* (и связанных с ним высот ярусно-параллельных форм графа). Эта проблема породила и группу гипотез, которую мы здесь, спустя годы, назовём **гипотезами Воеводина**.

В.В.Воеводин высказал первую из них относительно всего линейного класса, предположив, что (как и количество операций в линейном алгоритме) критический путь графа должен являться многочленом, ну или хотя бы дробно-рациональной функцией, относительно внешних переменных алгоритма (таких, как размеры массивов). Затем он уточнил, что, возможно, это справедливо только для тех программ из линейного класса, где коэффициенты при параметрах циклов и внешних переменных могут быть только 0, 1 или -1.

Рассмотрим опровержение первой из гипотез – фрагмент программы №1:

```
DO I=1,N
X (2*I) = X(I) * A(I)
END DO
```

После построения графа мы видим, что его критический путь равен $\lceil \log_2 N \rceil$, что выходит за рамки не только многочленов, но и дробно-рациональных функций. Однако, надо сказать, что этот простой пример был найден гораздо позже, чем опровержение второй из гипотез (с коэффициентами 0, 1 и -1) – фрагмент программы №2:

```
DO J=1, N
DO I=1, M
X(I) = X(I-J)*X(M)
END DO
END DO
```

Здесь все внутренние циклы (по I) должны быть выполнены последовательно друг за другом, а каждый из них критический путь порядка M/J. Суммируя, вроде бы получаем сумму дробно-рациональных выражений, но в асимптотике-то получаем M ln N.

Учитывая то, что параметры циклов могут быть «внешними параметрами» для внутренних циклов, имеем то, что логарифмы (как двоичные, так и натуральные) в формулах критического пути вполне могут иметь различные сочетания с многочленами и дробно-рациональными функциями внешних параметров задач. Не исключено, что читатели найдут и другие элементарные функции, которые могут встретиться в формулах критических путей графов.

### Параллельная сложность реальных программ из линейного класса

Читатель, возможно, уже недоумевает: зачем ему тогда вообще рассказали про гипотезы, которые для программ из линейного класса не выполняются? Закавыка, однако, в том, что приведённые выше программы, являющиеся их опровержением, не являются программами реально существующих алгоритмов, то есть таких алгоритмов, которые решают реальную математическую задачу. Эти контрпримеры являются чистой «игрой ума» - они выдуманы именно для опровержения гипотез (замечу также, что в обоих примерах наличествуют излишние вычисления). Между тем, Валентина Васильевича интересовали как раз алгоритмы реальные, а не абстрактные, которые фактически не решают никаких математических задач. Его предположения относительно коэффициентов линейных функций, которые ограничивают линейный класс, были вызваны вовсе не желанием получить такой подкласс линейных алгоритмов, для которых выполнялась бы его гипотеза. Нет. Дело в том что из подклассов линейного класса программ наибольшую важность представляет его пересечение с классом алгоритмов реальных программ, то есть тех программ, которые решают вполне содержательные математические задачи. Именно для него были совершены попытки формализовать ограничения на коэффициенты.

И, что особенно интересно, вот на этом-то подклассе гипотеза Воеводина до сих пор не опровергнута. Именно этот факт в значительной мере обуславливает то, что в методике анализа параллельных структур программ, заложенной в такие системы, как V-Ray, МАКРОГРАФ и т.п., используются такие инструменты, как нахождение развёрток из линейных классов.

**Реальные программы: какие они?**

Обнаруженное выше свойство пересечения классов линейных программ с классом реальных программ ставит вопрос: если всё же гипотеза Воеводина справедлива для него, то какое именно свойство реальных линейных программ обусловливает это? Высказанные выше предположения относительно коэффициентов оказались в качестве такого условия негодными. Формулировка такого свойства является интересной задачей для алгоритмистов. Пока что очередным кандидатом на него является такое: коэффициенты при параметрах циклов и внешних переменных могут быть только 0, 1 или -1, и при этом в одном индексе может появляться не более одного ненулевого коэффициента при параметре цикла и не более одного при внешней переменной.

Автору данной статьи пока что не удалось найти опровержения гипотезе Воеводина для этого подкласса линейных алгоритмов, и это внушает определённые надежды на то, что искомое свойство найдено. Однако меньший оптимизм у автора по отношению к другому — что именно это свойство и ограничивает пересечение линейного класса программ с классом программ, реализующих реальные математические алгоритмы. Не исключено, что за время, оставшееся до конференции, автору уже удастся найти опровержение этому (скорее всего, среди алгоритмов для ленточных матриц или т.п.).

Поэтому прошу воспринимать данную публикацию в качестве призыва к специалистам по алгоритмам попытаться решить данные проблемы. То есть, во-первых — найти свойство, которому удовлетворяют реальные программы из линейного класса (вполне возможно, что оно вообще описывается не через величины коэффициентов линейных функций). Во-вторых — найти свойство программ/алгоритмов линейного класса, для которых справедлива гипотеза Воеводина. В-третьих — сопоставить эти два свойства между собой.

ЛИТЕРАТУРА:


1. Воеводин В.В. Математические модели и методы в параллельных вычислениях. – М.: Наука, 1986. 296с.
2. Воеводин В.В. Параллельные структуры алгоритмов и программ. – М.: ОВМ АН СССР, 1987. 148 с.
3. Воеводин В.В. Математические основы параллельных вычислений. – М.: Изд-во МГУ, 1991. 345 с.
4. Воеводин В.В. Информационная структура алгоритмов. – М.: Изд-во МГУ, 1997. 139 с.
5. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. – СПб.: БХВ-Петербург, 2002. 608 с.
6. Воеводин В.В. Вычислительная математика и структура алгоритмов. – М.: Изд-во МГУ, 2006. 112 с.
7. Воеводин В.В. Полиномиальное оценивание сложности алгоритмов. – ЖВМиМФ, 1999, т.39, №6, с. 1032-1040.
8. Воеводин Вад.В., Соболев С.И., Фролов А.В. Архитектура и принципы реализации коллективного банка тестов в сети Интернет. – в сб.: Численные методы, параллельные вычисления и информационные технологии. – М. Изд-во МГУ, 2008. с. 91-97.
9. Катцан Г. Язык Фортран 77: Пер. с англ. – М.: Мир, 1982. 208 с.
10. Фролов А.В. Косвенная и другие виды нелинейной адресации в фортран-программах, подвергаемых распараллеливанию. – Матричные методы и алгоритмы. М.: ИВМ РАН, 1993, с. 96-101.
11. Фролов А.В. Нахождение обменоэкономных распределений массивов в фортран-программах. – Программирование, 1997, №4, с. 71-80.
12. Фролов А.В. Оптимизация размещения массивов в фортран-программах на многопроцессорных вычислительных системах. – Программирование, 1998, №3, с. 70-80.
13. Voevodin V.V. Information structure of sequential programs. – Russ. J. of Num. An. and Math. Modelling, 1995, V.10, №3, p.279-286.