

1. Титульник

Добрый день. Меня зовут Николай Пономарев и тема моей работы --- "Разработка транслятора модельного функционального языка в Interaction Nets".

2. Введение

Искусственный интеллект и анализ графов --- одни из наиболее перспективных областей науки в данный момент. Объемы обрабатываемых данных в этих областях стремительно увеличиваются, поэтому разумно пытаться распараллеливать вычисления. Однако при распараллеливании некоторых алгоритмов, используемых в данных областях, начинает проявляться такой эффект, как нерегулярный параллелизм, при котором обращения к памяти обычно нелокальны, а количество независимых подзадач постоянно меняется. К сожалению, традиционные архитектуры процессоров и видеокарт плохо справляются с нерегулярным параллелизмом, что заставляет сообщество разрабатывать ускорители на альтернативных архитектурах. Одной из таковых может стать архитектура на основе модели вычислений Interaction Nets, которой свойственен нерегулярный параллелизм. Была создана не одна программная реализация Interaction Nets, а вот попыток реализовать ускоритель на её основе пока не предпринималось.

3. Lamagraph

Таким образом, нами был начат проект Lamagraph для создания программно-аппаратного стека на основе Interaction Nets. Проект групповой и состоит из трех больших частей: части связанной с транслятором функционального языка; части, описывающей прошивку ускорителя; и связующей части для генерации программы для исполнения на ускорителе из высокоуровневого кода. Данная работа сосредоточена на блоке про транслятор и частично на связующей части.

4. Состояние проекта

Сейчас проект находится на самом первом этапе --- разработка минимальной инфраструктуры для создания ускорителей. На данном этапе от проекта ожидается следующее:

1. Приоритизация получения полнофункционального прототипа, содержащего все компоненты; сейчас нам нужно создать всю цепочку и выявить проблемы, возникающие на разных этапах.
2. Использование единого набора технологий; такое решение позволяет упростить интеграцию компонент и их тестирование, поэтому язык реализации --- Haskell.

3. Возможность анализа результатов каждого этапа работы нам необходима для тестирования, как изолированный компонент, так и их цепочки.

5. Постановка задачи

С учётом вышесказанного, целью данной работы является разработка транслятора модельного функционального языка в Interaction Nets. Для достижения этой цели были поставлены следующие задачи:

1. Реализовать интерпретатор модельного ML-подобного языка, в частности фронтенд языка и интерпретатор обогащенного λ -исчисления, получаемого после работы фронтенда.
2. Реализовать транслятор λ -исчисления в Interaction Nets.
3. Реализовать интерпретатор Interaction Nets со сбором метрик исполнения.

6. Реализация

На слайде представлена диаграмма взаимодействия пользователя с реализованным программно-аппаратным стеком, данная работа покрывает красный и синий блоки. Передав на вход программу на пользователь может получить вывод интерпретатора, а также вывод абстрактной машины и соответствующую статистику. Обсудим каждую компоненту отдельно.

7. Интерпретатор LamagraphML

Начнём с высокоуровневого языка и его интерпретатора. В языке используется ML-подобный синтаксис --- это связано в первую очередь с наличием схем трансляции λ -исчисления в Interaction Nets. Для представления дерева абстрактного синтаксиса используется функциональный паттерн Trees That Grow, позволяющий безопасно параметризовать узлы дерева различными аннотациями. Поскольку задача разработки транслятора ML-подобного языка не нова, используются стандартные подходы и инструменты, принятые в области. Для синтаксического анализа используется связка лексер- и парсер-генераторов Alex и Happy. Для вывода типов используется алгоритм Хиндли-Милнера. После вывода типов для упрощения AST происходит рассахаривание в λ -исчисление, обогащенное литералами, алгебраическими типами данных и сопоставлением с образом. При интерпретации используется подход с использованием замыканий. При этом интерпретатор наследует стратегию вычислений Haskell --- call-by-need, но в случае необходимости стратегия может быть изменена.

8. Транслятор λ -исчисления в Interaction Nets

Следующим компонентом является транслятор λ -исчисления, полученного после работы фронтенда, в Interaction Nets. Interaction Nets как модель вычислений перекладывает описание правил вычисления на пользователя, что делает систему параметризуемой. Потому существует не одна схема трансляции λ -исчисления в Interaction Nets. Поскольку наш язык ML-подобный, то для реализации была выбрана схема, реализующая стратегию вычислений call-by-value. Однако в процессе анализа выяснилось, что схемы трансляции существуют только для чистого λ -исчисления. Существуют работы по расширению схем трансляции, однако они требуют использования параметризации, специализированной под программу, что противоречит фиксированной параметризации ускорителя. Поэтому на данном этапе была реализована схема только для чистого λ -исчисления, а расширения схемы --- предмет дальнейшей работы.

9. Интерпретатор Interaction Nets

После трансляции программу на Interaction Nets нужно как-то исполнить. Стандартным представлением Interaction Nets является графовое. Однако представления графов в функциональных языках не позволяют использовать алгебраические типы данных и сопоставление с образцом, поэтому потребовалось найти альтернативное представление. Таковым стало текстовое представление, для которого существует абстрактная машина, которая и была реализована.

10. Метрики исполнения

От интерпретатора требовалась поддержка сбора метрик. Она была реализована. Получаемые метрики позволяют отвечать на вопросы о параллельности, важные для разработки ускорителя. Справа на слайде приведен пример анализа полученных метрик. На графике представлена оценка ускорения при параллельном исполнении итеративного факториала на входах от 0 до 6. На этом примере видно, что при малом аргументе параллельность только вредит --- ускорение меньше единицы, а при бóльших аргументах ускорение увеличивается вплоть до 1.7 раз.

11. Результаты

В рамках ВКР удалось выполнить все поставленные задачи и получить программный стек от входного ML-подобного языка до интерпретатора Interaction Nets, а также выявить важные проблемы, такие как необходимость расширения существующих схем трансляции λ -исчисления для полноценной поддержки нашего промежуточного представления. У меня всё. Готов ответить на вопросы.