



Санкт-Петербургский государственный университет
Кафедра системного программирования

Разработка инфраструктуры для создания специализированных ускорителей на основе Interaction Nets

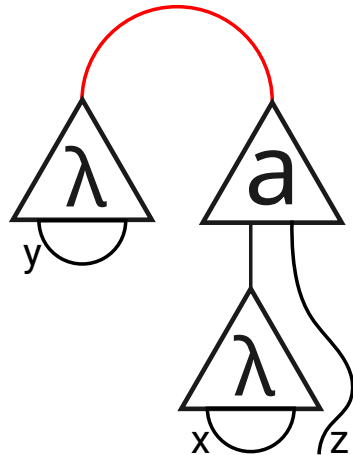
Ефим Кубышкин Николай Пономарев

Санкт-Петербург
2025

- Современные вычислительные системы сталкиваются с растущей потребностью в параллельной обработке данных
- Традиционные архитектуры процессоров неэффективны для нерегулярного параллелизма
- 👍 Interaction Nets — формализм, предложенный Ивом Лафоном в 1989 году, для которого нерегулярный параллелизм естественен
- 👎 На данный момент существуют только программные реализации

Interaction Nets

- Математическая модель, основанная на графовых грамматиках
- Переписывания (редукции) независимы
- Доказана полнота по Тьюрингу и свойство независимости от порядка вычислений
- Метки на вершинах и правила грамматики можно менять \Rightarrow высокая степень параметризуемости



Цель и задачи проекта

Цель: Разработка параметризуемого многоядерного вычислителя на основе Interaction Nets

Этапы достижения цели:

1. Создание минимальной инфраструктуры для создания специализированных вычислителей на основе Interaction Nets
2. Разработка eDSL для спецификации Interaction Nets
3. Добавление поддержки многоядерности
4. Проведение экспериментов

Цель и задачи проекта

Цель: Разработка параметризуемого многоядерного вычислителя на основе Interaction Nets

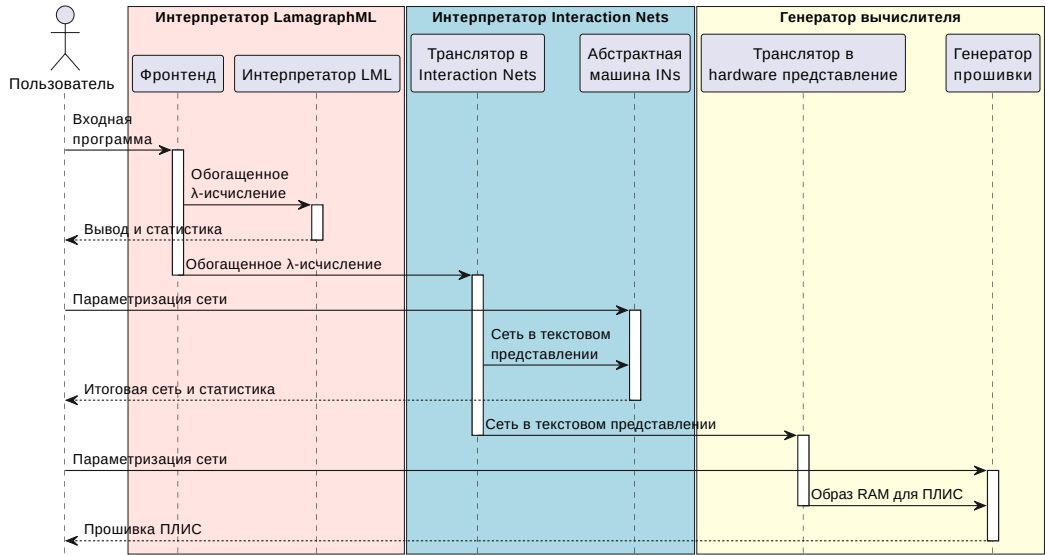
Этапы достижения цели:

- 1. Создание минимальной инфраструктуры для создания специализированных вычислителей на основе Interaction Nets
 - Реализация высокоуровневого языка программирования
 - Разработка транслятора из высокоуровневого языка в Interaction Nets
 - Реализация интерпретатора Interaction Nets
 - Разработка генератора прошивки вычислителя для ПЛИС
2. Разработка eDSL для спецификации Interaction Nets
3. Добавление поддержки многоядерности
4. Проведение экспериментов

Требования к первому этапу

- Использование единого стека технологий — гомогенность
 - Упрощение цепочки обработки данных
- Получение полнофункционального прототипа, содержащего все компоненты, важнее, чем детальная проработка какого-то отдельного компонента
 - Фокус на инфраструктуре
 - Сбор информации о трудностях на каждом этапе
- Возможность сбора статистики
 - Исследовательский проект
- Итеративность: на каждом этапе есть результат исполнения программы
 - Тестирование каждой из компонент
 - Сравнение результатов работы компонент

Lamagraph



- Существующие языки для программирования на Interaction Nets — низкоуровневые и с непривычным синтаксисом
- Пользователю должен быть доступен высокоуровневый и привычный язык
- Синтаксис LamagraphML основан на ML с системой типов Хиндли-Милнера
 - Функциональный в силу близости Interaction Nets и λ -исчисления
 - Чистые функции лучше для параллельности
- 💡 Для интерпретации и дальнейшей обработки используется обогащенное λ -исчисление

Интерпретатор Interaction Nets

- Стандартное представление Interaction Nets — графовое
- С графами трудно работать в функциональных языках программирования
⇒ используем текстовое представление Interaction Nets
- Для текстового представления существует абстрактная машина
- ☹ State-of-the-art наука умеет транслировать только чистое λ -исчисление в Interaction Nets

- Clash — язык описания программного обеспечения, основанный на Haskell
 - Unit тесты для комбинационной логики
 - Симуляция последовательностной логики
 - Test bench файлы на Verilog
- Сбор статистики
 - Количество тактов
 - Количество редукций
 - Максимальный размер сети
- Параметризация правилами переписывания и метками на узлах через типы

Программа на LamagraphML

- ✓ Интерпретация
- ⚙ Трансляция в Interaction Nets (только λ -исчисление)

Абстрактная машина

- ✓ Интерпретация
- ✓ Сбор статистики

Генерация вычислителя

- ✓ Прошивка вычислителя для ПЛИС
- ✓ Сбор статистики