Russian Supercomputing Days

МОСКВА, 25-26 сентября 2017 г.
Объединеная суперкомпьютерная конференция

Адаптация решателя динамического напряженно-деформированного состояния методом дискретных элементов для неоднородных вычислительных систем

Алексеев С.С. Гетманский В.В., Андреев А.Е.
Волгоградский государственный технический университет
кафедра ЭВМ и С

Вычислительный кластер ВолгГТУ – неоднородный вычислительный комплекс

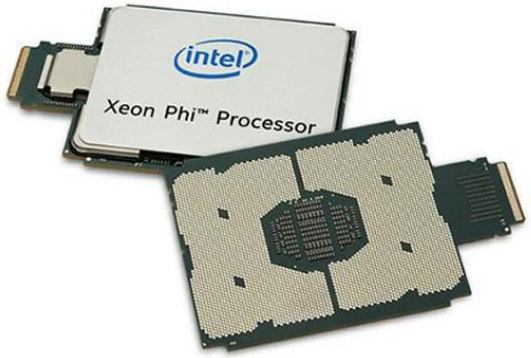
Вычислительный кластер ВолгГТУ на базе Intel Xeon E5 / Intel Xeon Phi / KNL – Топ-50 (40 - март, 47 – сентябрь)

(24 nodes – >39 TFlops on Intel HPL with offload, E5, KNC and **KNL** used)

На Linpack. А что на реальных задачах ?..

Вычислительный кластер ВолгГТУ – неоднородный вычислительный комплекс

Четыре и более типов вычислителей (+ FPGA)

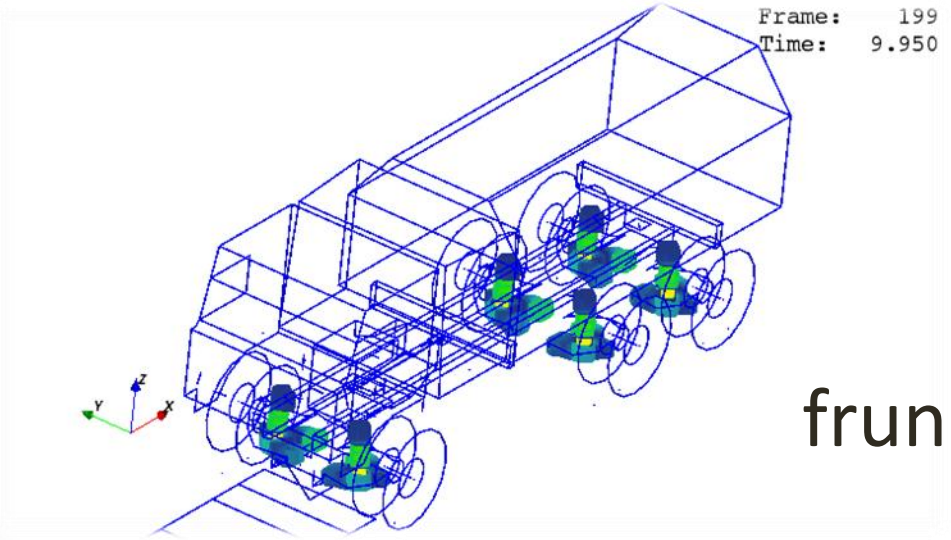


OpenCL + MPI

Программный комплекс ФРУНД

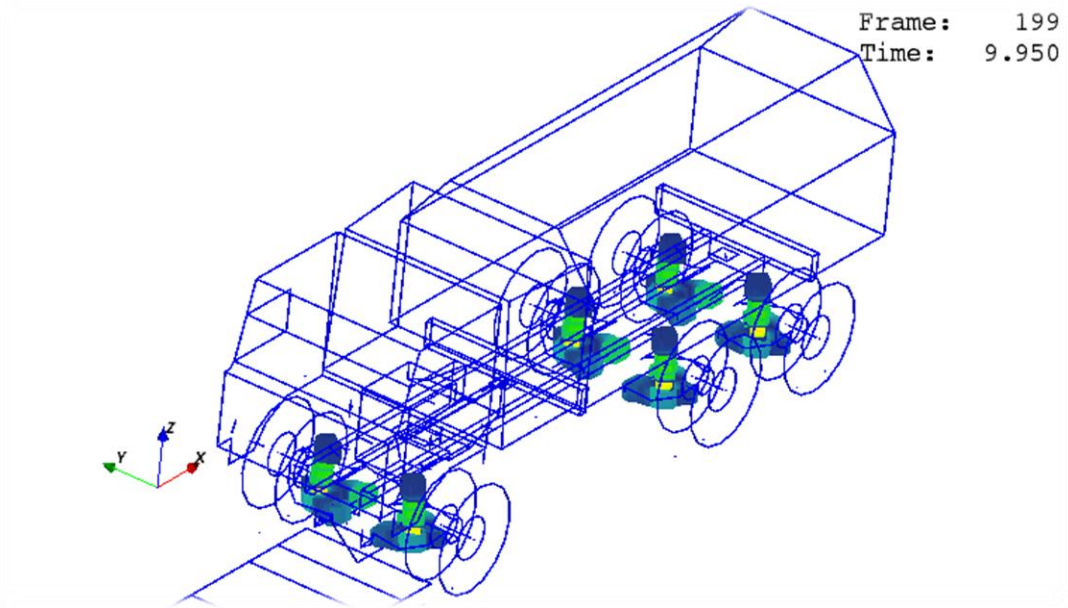
(*формирование и решение уравнений нелинейной динамики*)

Пакет относится к CAE системам, использующим т.н. многотельные модели (MBS) представления сложных технических систем, а также междисциплинарные расчеты для анализа их динамических характеристик. Применяется для решения задач из разных областей, включая создание новых транспортных систем, спецмашин, управление шагающими машинами, включая антропоморфных роботов.

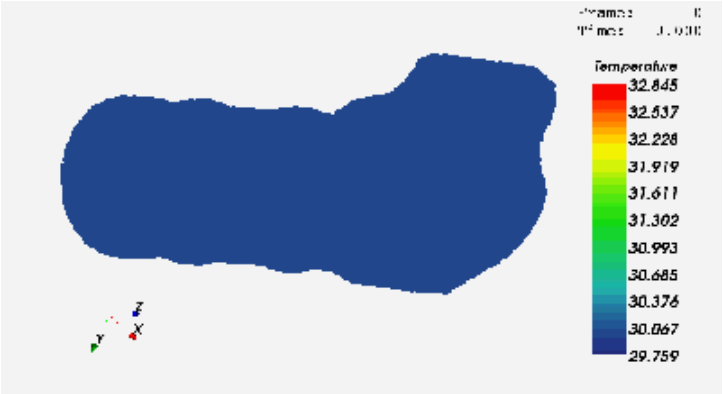


frund.vstu.ru

Междисциплинарное моделирование в пакете ФРУНД



Движение по неровной поверхности



Процесс теплопередачи (нагрев тела)

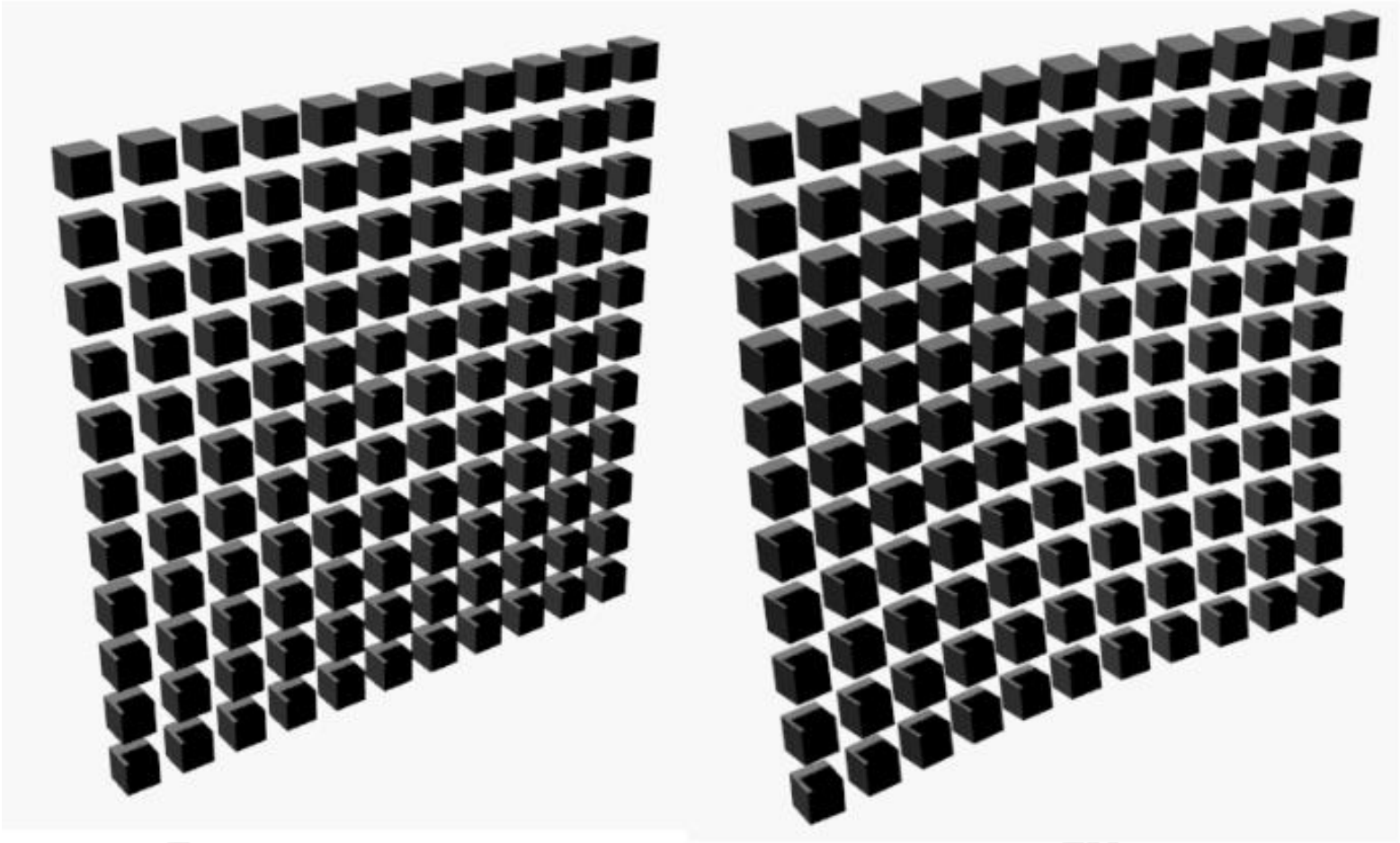
Напряжения при деформации тела (расчет НДС)

Общая постановка задачи – динамика систем тел

- Задача динамики (постановка в форме системы уравнений Лагранжа 1-го рода, программный комплекс ФРУНД)

$$\begin{cases} \mathbf{M}\ddot{\mathbf{x}} + \mathbf{D}_x^{\mathrm{T}}\boldsymbol{\alpha}(2\boldsymbol{\mu}\boldsymbol{\Omega}\dot{\mathbf{D}} + \boldsymbol{\Omega}^2\mathbf{D}) + \mathbf{D}_x^{\mathrm{T}}\boldsymbol{\lambda} = \mathbf{q}(\dot{\mathbf{x}}, \mathbf{x}, t), \\ \mathbf{D}_x\ddot{\mathbf{x}} = \mathbf{h}(\dot{\mathbf{x}}, \mathbf{x}, t). \end{cases}$$

Дискретное представление тел для расчета НДС



Дискретные элементы описываются ДУ вида :

$$M\ddot{y} = q(\dot{y}, y, t) - Ma(t) + s(\dot{y}, y)$$

Модернизация и развитие пакета ФРУНД

- Внедрение параллелизма на разном уровне
- Разработка нового пред-пост процессора
- Дополнительные модули (например, для управления актуаторами роботов)
- Переработка кода (исходные решатели написаны на Фортране достаточно давно – унаследованный код) – переход к ОО – версии (тепловой решатель, решатель НДС)

ОО-версия решателя НДС для FRUND

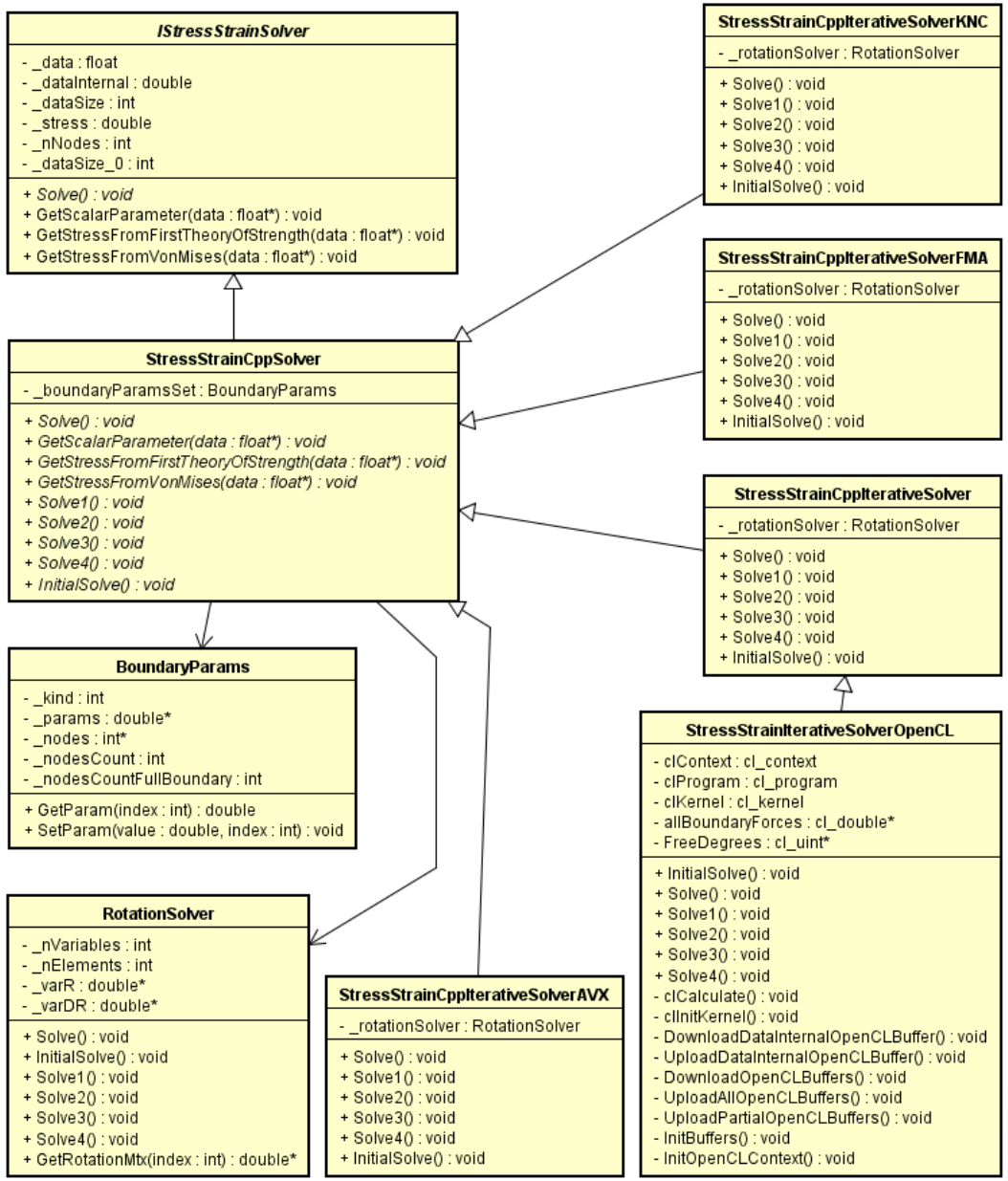
Новая версия FRUND, разрабатываемая на C++.

+ Модульность

+ Масштабируемость

+ Портируемость

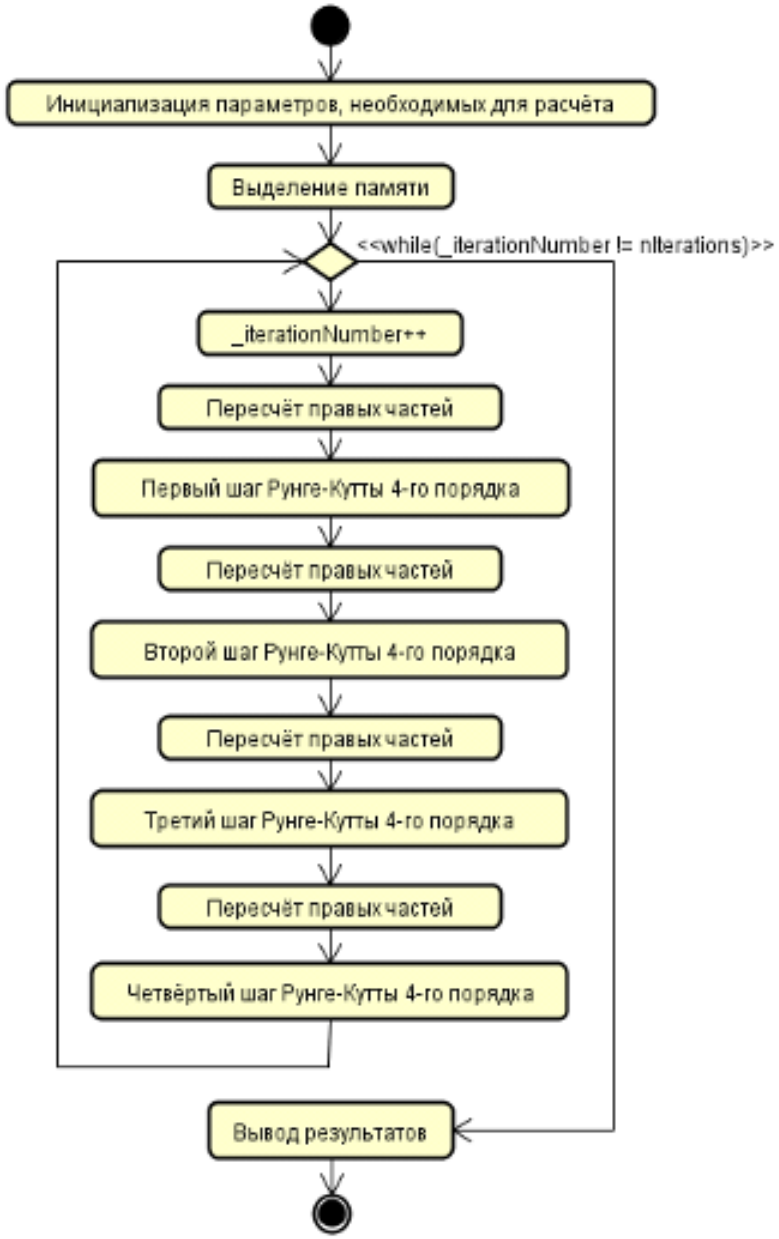
+ Гибкость применения

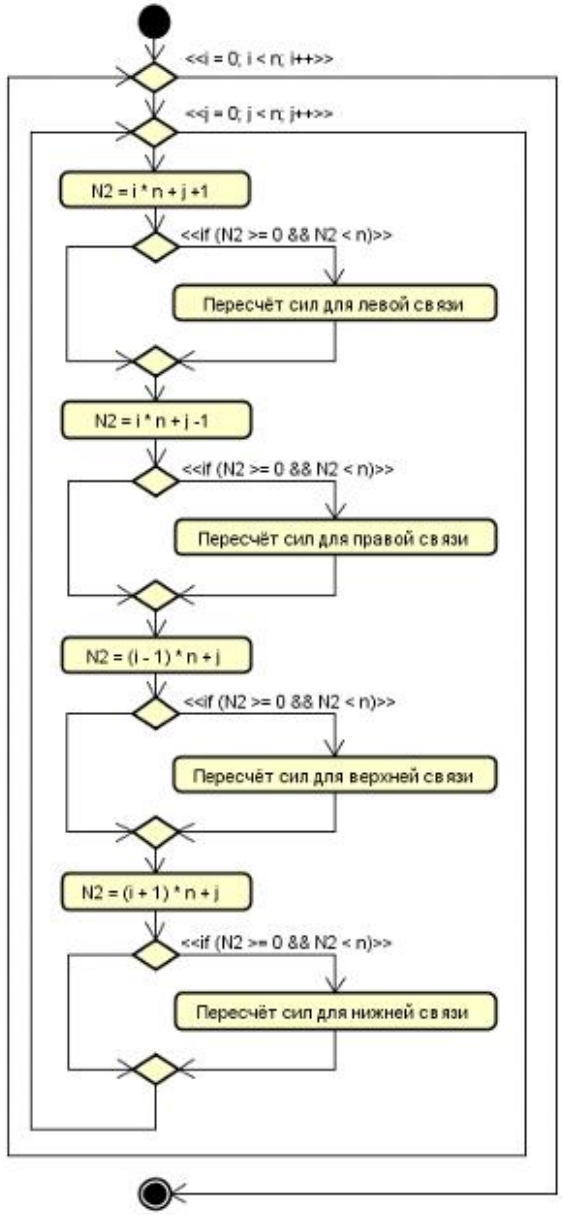


Параллелизм в пакете ФРУНД

- Декомпозиция и параллельный расчет сложной многотельной модели (MPI, OpenMP) – ограничение масштабируемости, сверхбольшие модели встречаются редко
- Параллельный междисциплинарный расчет (динамика, деформация, нагрев)
- **Расчет тепловых задач (нагрев узлов), напряженно-деформируемого состояния на CUDA / OpenCL**
- Векторизация решателей ФРУНД
  - Интегрирование СОДУ методом РК4 (простая арифметика над векторами)
  - Вычисление правых частей (матричная алгебра)

Расчет – метод РГ4 и перерасчет правых частей





Особенности OpenCL - реализации

- выполнение расчета максимально на устройстве (GPU), все циклы расчета и все основные шаги
- загрузка данных в начале расчета и минимизация коммуникации между хостом и устройством на каждой итерации
- устранение или минимизация ветвлений (используем массивы флагов)
- разворачивание циклов, выравнивание данных в памяти и другие приемы

Функции хостовой части приложения

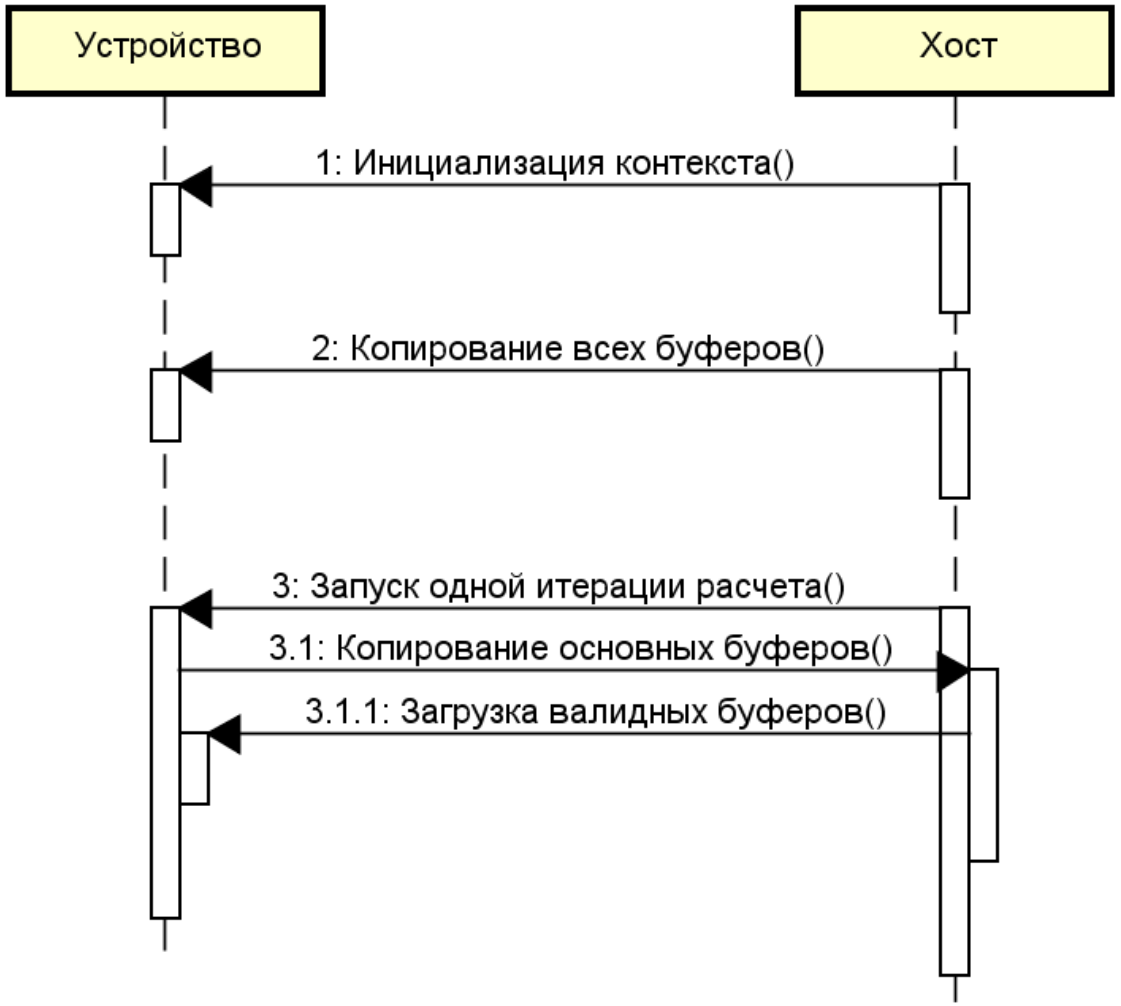
Важные аспекты

1. Взять на себя нагрузку по ветвлениям.
2. Организовать хранение, передачу и синхронизацию данных.
3. Валидация данных.
4. Поддержка всех возможных платформ для сборки и исполнения ядра

Синхронизируемые между хостом и устройством данные

Название буфера	Флаги памяти хоста	Содержимое
DataInternal	Read/Write	Линейные и угловые координаты, первые и вторые производные
Stress	Read/Write	Напряжения
BoudaryFixed	Write	Граничные условия для координат в виде флагов
BoudaryForces	Write	Граничные условия для вторых производных в виде слагаемых
LinkedElements	Write	Номера соединенных элементов по шести граням, нуль если сосед отсутствует

Взаимодействие хоста и устройства



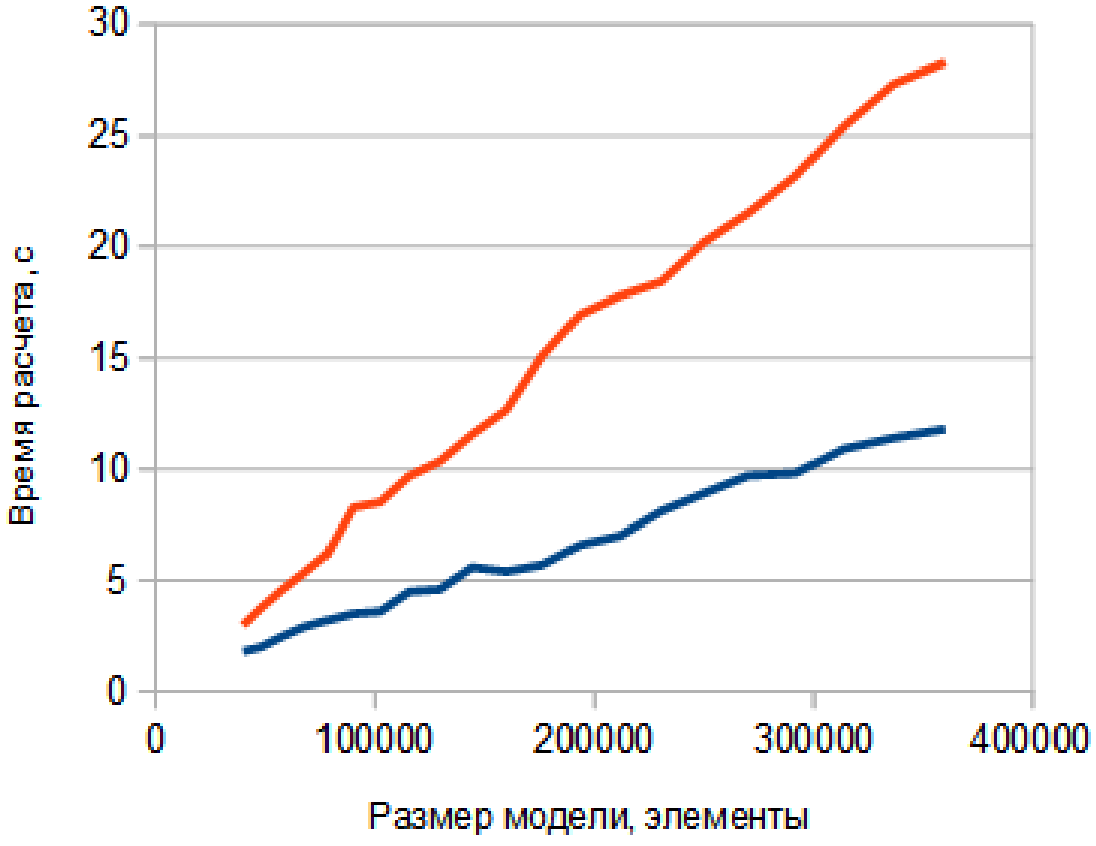
Платформы для тестирования

На хосте был реализован код, выбирающий произвольную платформу.

Ядро было успешно запущено на:

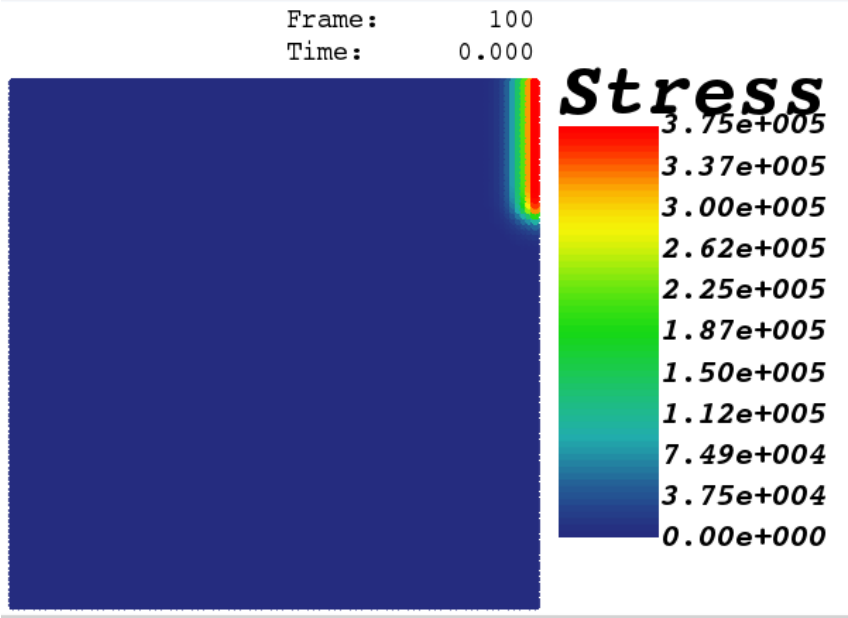
- GPU
    - AMD RX 460
    - AMD RX 470
    - Nvidia GTX 780
    - Nvidia Tesla K20
    - Nvidia GeForce 610m
- CPU
    - AMD FX 6300
    - Intel Core i3 3210m
    - Intel Xeon E5 2660

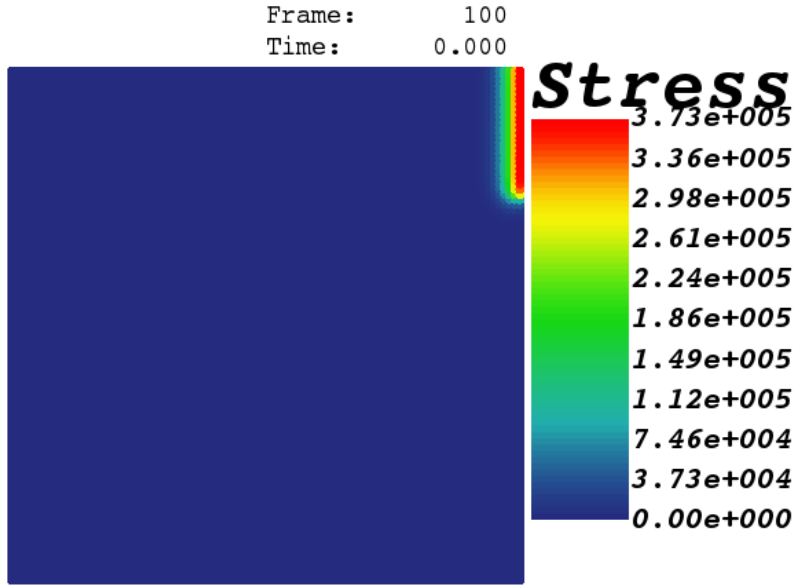
Время расчета на 2 CPU E5 vs TESLA K20



Ускорение до 2,5 раз

Визуализация напряжений – CPU (слева), GPU – OpenCL (справа)

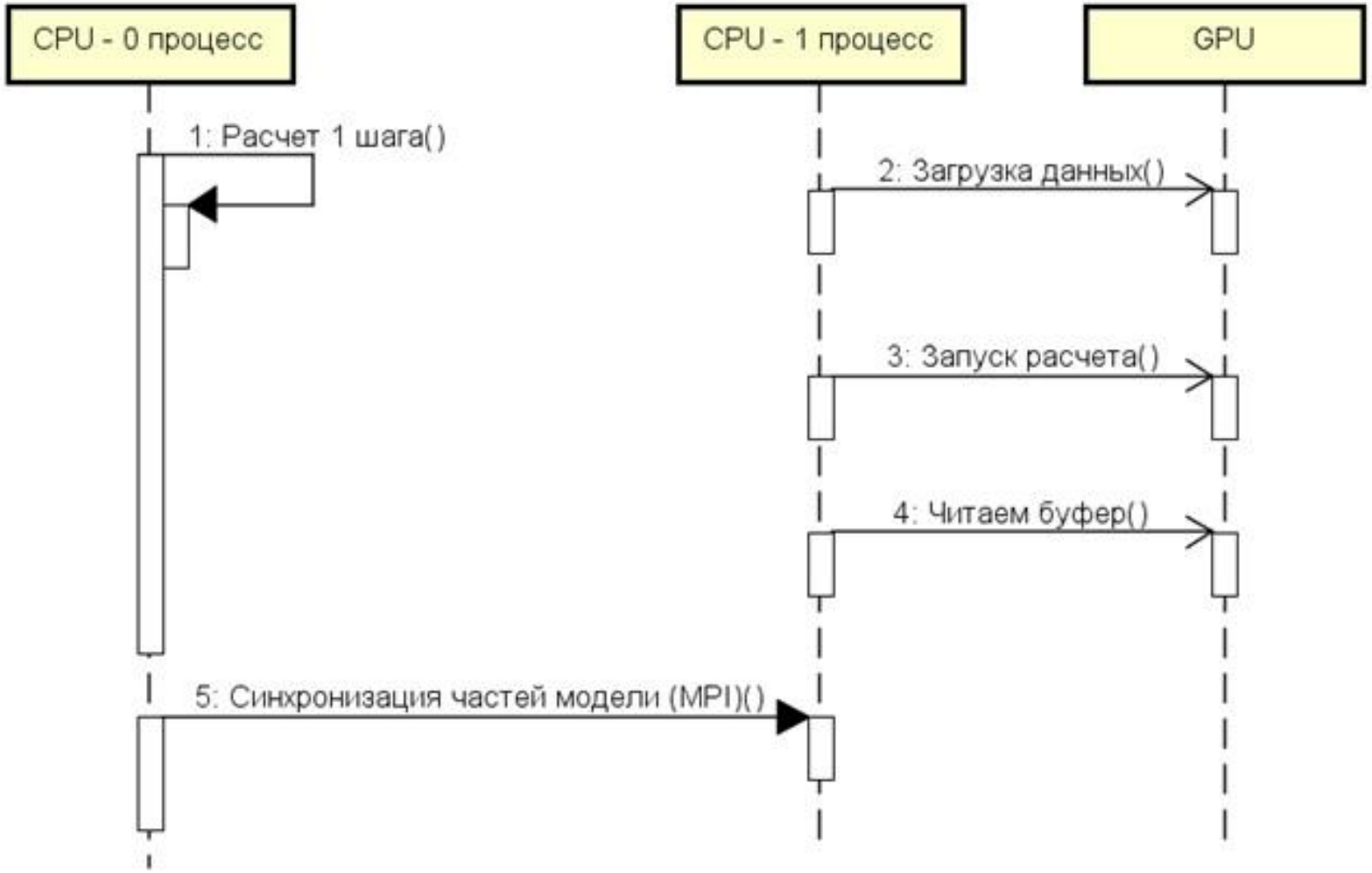




Перспективы развития

- реализация OpenCL расчета на float (требует исследования – частично нуждается в double) – позволит использовать «бытовые» GPU
- перенос OpenCL - решателя на FPGA (прототип был реализован для Altera DE5-Net, но только на float и только для расчета правых частей)
- перенос всех вычислений на GPU
- совместный расчет на разных вычислителях для больших моделей – пока реализован прототип для MPI-расчета на одном узле (CPU + GPU) – ускорение до 1,46 раз, для реализации в реальном решателе необходимо доработать систему декомпозиции и синхронизации моделей

Совместный расчет на CPU + GPU (OpenCL+MPI)



Спасибо за внимание !

Контакты:

зав. каф. ЭВМ и С ВолгГТУ
к.т.н. Андреев Андрей Евгеньевич
e-mail: andan2005@yandex.ru

к.т.н., инженер компании Intel
Гетманский Виктор Викторович
e-mail: victor.getmanski@gmail.com

магистрант каф. ЭВМ и С ВолгГТУ
Алексеев Сергей Сергеевич
e-mail: serales96@mail.ru

Работа выполнена при поддержке РФФИ и администрации Волгоградской области грант № № 16-47-340385.