Институт прикладной математики им. М.В. Келдыша РАН

Суперкомпьютерные алгоритмы: отказоустойчивость, генерация сеток

С.К.Григорьев, **М.В. Якобовский**
lira@imamod.ru

04 февраля 2020
Семинар НИУ ВШЭ по высокопроизводительным вычислениям

Текущее состояние

- Относительно малое число примеров использования вычислительных мощностей превышающих 100 TFLOPs
  причины: острый дефицит математических моделей, численных алгоритмов и программных средств для высокопроизводительных вычислительных систем
- Необходимы логически простые и эффективные алгоритмы для современных и для будущих архитектур высокопроизводительных вычислительных систем
- Основные проблемы инвариантны относительно типа используемых вычислительных систем (CPU, GPU)
- Решение на основе фундаментальной науки

Ближайшие перспективы

- Реальная необходимость высокопроизводительных вычислительных систем следующего поколения для решения задач:
  - нефтегазовые проблемы разведки и оптимизации добычи
  - экологические двигатели
  - ядерная энергетика и термоядерный синтез
  - фундаментальные проблемы астрофизики

- 2015 - Достаточно широкое использование PetaFLOPs вычислительных систем

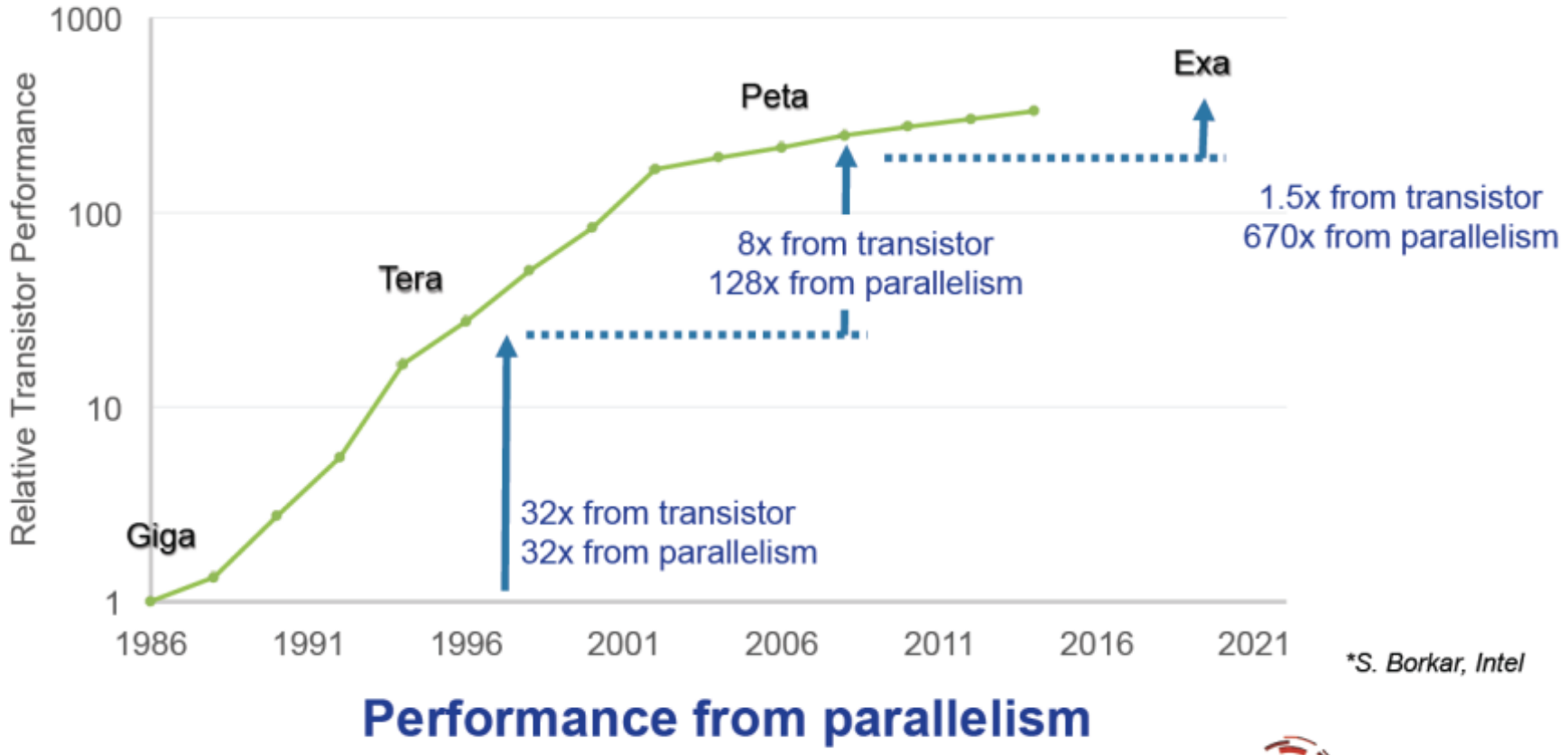
- 2018…2021… - Производительность суперкомпьютеров 1 ExaFLOPs

July 31, 2016

Paul Messina, Argonne National Laboratory, ECP Director

A Path to Capable Exascale Computing

From Giga to Exa, via Tera & Peta*

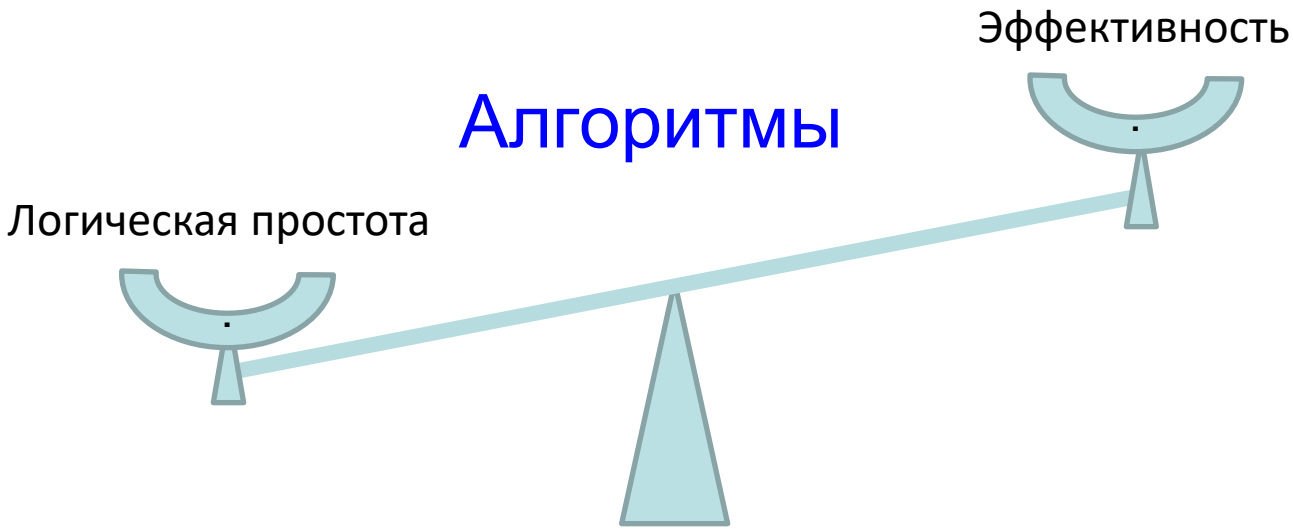


July 31, 2016

Paul Messina, Argonne National Laboratory, ECP Director

A Path to Capable Exascale Computing

- ExaScale computer
  - Позволяет решить научную задачу
    - в 50 раз быстрее (или сложнее) чем на системах 20 Pflops (Titan, Sequoia)
  - Мощность порядка 20-30 МВт
  - Отказоустойчивость позволяет пользователю вмешиваться на чаще одного раза в неделю
  - Имеет стек программного обеспечения, который отвечает потребностям широкого спектра приложений и рабочих нагрузок



- Явные схемы позволяют создавать логически простые алгоритмы, но имеют строгие ограничения на дискретизацию по времени из условий устойчивости:
  - для уравнений гиперболического типа условие устойчивости

$$\Delta t \le h,$$

где Δt - шаг дискретизации по времени, h - шаг дискретизации по пространству

  - для параболического типа уравнений условие устойчивости

$$\Delta t \le h^2$$

Это условие практически исключает возможность использования высокого разрешения по пространству

$$\frac{\delta^2 \varphi}{\delta x_i^2} = 4\pi G \rho(r)$$

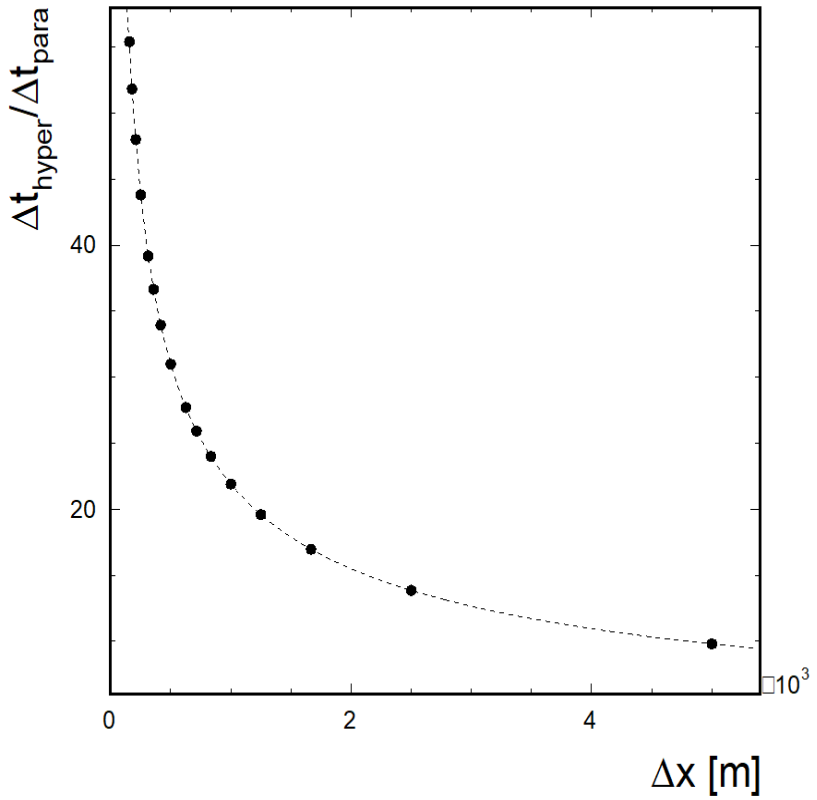
$$\rho(r) = \begin{cases} \rho & r \leq R \\ \delta & r > R \end{cases}$$

$$\frac{\varphi_i^{j+1} - \varphi_i^j}{\Delta t} = (\varphi_{\bar{x}})_x^j + F$$

$$\frac{\varphi_i^{j+1} - \varphi_i^{j-1}}{2\Delta t} + \tau^* \frac{\varphi_i^{j+1} - 2\varphi_i^j + \varphi_i^{j-1}}{\Delta t^2} = (\varphi_{\bar{x}})_x^j + F$$

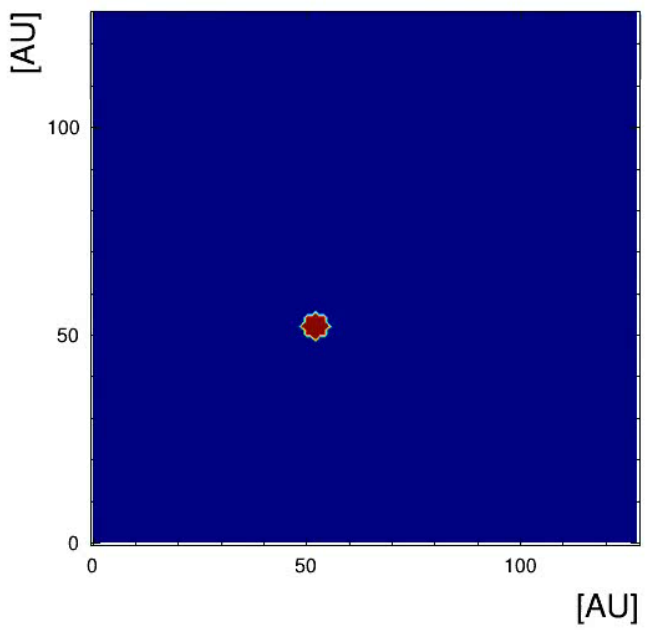
Уравнение Пуассона (гравитационный потенциал)

Отношение временной дискретизации параболического и гиперболического метода как функция дискретизации по пространству



Аккреция облака межзвездного газа на компактном астрономическом объекте

низкое разрешение



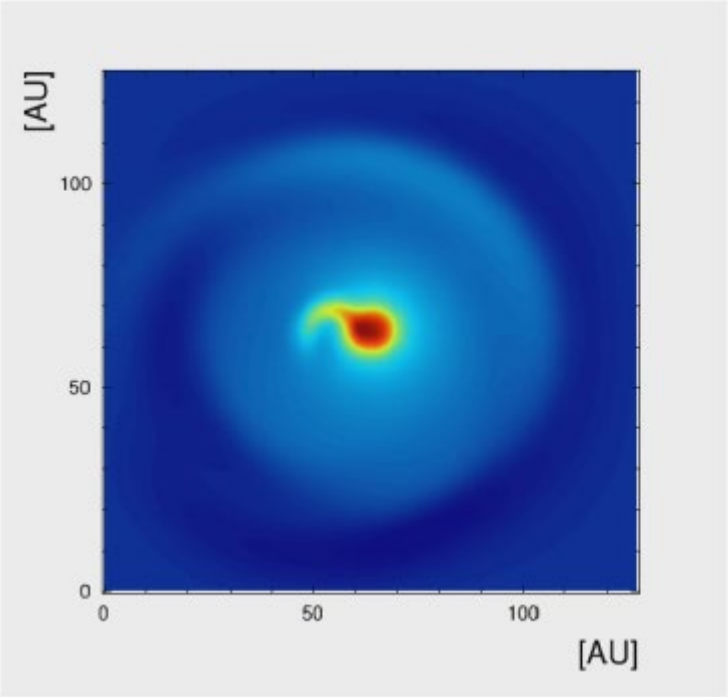
высокое разрешение

- Межзвездное облако 5 AU
- Плотность 0.8 x 10^{-11} кг/m^3
- Скорость облака 300 m/s
- Импакт параметер 4-10 AU

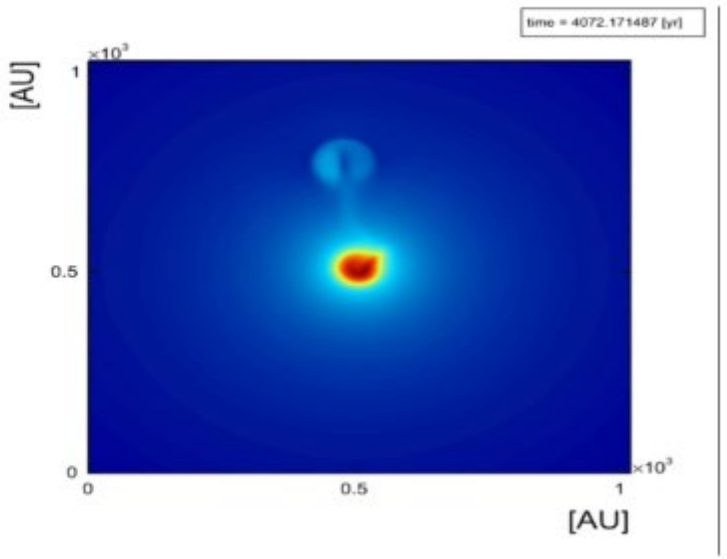
- Компактный обьект:
  - масса 10^{30} Kg
  - радиус 0.5 AU
- Температура пространства T = 20 K

Accretion of a cloud of interstellar gas on a compact astronomical object

Low resolution



High resolution



- Interstellar cloud 5 AU
- Density 0.8 x 10^{-11} кг/m^3
- Cloud speed 300 m/s
- Impact parameter 4-10 AU

- compact astronomical object:
  - Weight 1030 Kg
  - Radius 0.5 AU
- Temperature of space T = 20 K

Blue Waters system

Cray HLRS – Germany, Stuttgart

- Каждые 4.2 часа фиксируется отказ, требующий восстановления части системы
- Полный отказ системы каждые 160 часов

Di Martino, Catello, Zbigniew Kalbarczyk, Ravishankar K. Iyer, Fabio Baccanico, Joshi Fullop, and William Kramer. "Lessons learned from the analysis of system failures at petascale: The case of blue waters." In*Dependable Systems and Networks (DSN), 2014 44th Annual IEEE/IFIP International Conference on*, pp. 610-621. IEEE, 2014.

Статистика отказов

Системы	Количество ядер	Надежность
ASCI Q	8192	**MTBF: 6.5 часов** Источники аппаратные сбоев: устройство хранения, CPU, память
ASCI White	8192	**MTBF: 5 часов (2001 год) и 40 часов (2003 год)** Источники аппаратные сбоев: устройство хранения, CPU, внешние устройства
PSC Lemieux	3016	**MTBI: 9.7 часов**
Google	15000	20 перезагрузок/день 2-3% компьютеров заменяется ежегодно Источники аппаратных сбоев: устройства хранения, память

D. Reed. High-end computing: The challenge of scale. Director's Colloquium, Los Alamos National Laboratory, May **2004**

Время между отказами на экзафлопсных системах ~ 30 минут

Marc Snir, et al. Addressing failures in exascale computing. International Journal of High Performance Computing Applications, 28(2):129–173, May 2014

Частота аппаратных отказов будет возрастать

- Уменьшение размера транзистора делает его менее устойчивым к космической радиации
- Ёмкости меньшего размера содержат меньший заряд, - его проще изменить

Программное обеспечение становится сложнее и содержит больше ошибок

- Оборудование становится сложнее (неоднородные ядра, многоуровневая иерархия памяти, сложная топология объединения узлов) существенно усложняет программное обеспечение
- *Мультифизичность и многомасштабность* решаемых задач приводит к объединению большого числа программных модулей.
- Сокращение обменов, использование асинхронных взаимодействий, обеспечение защищённости от отказов оборудования приводит к созданию сложных прикладных кодов

Cappello, F., Geist, A., Gropp, W., Kale, S., Kramer, B., & Snir, M. (2014). Toward exascale resilience: 2014 update. *Supercomputing frontiers and innovations*, *1*(1).

Время создания контрольной точки ~ 30 минут

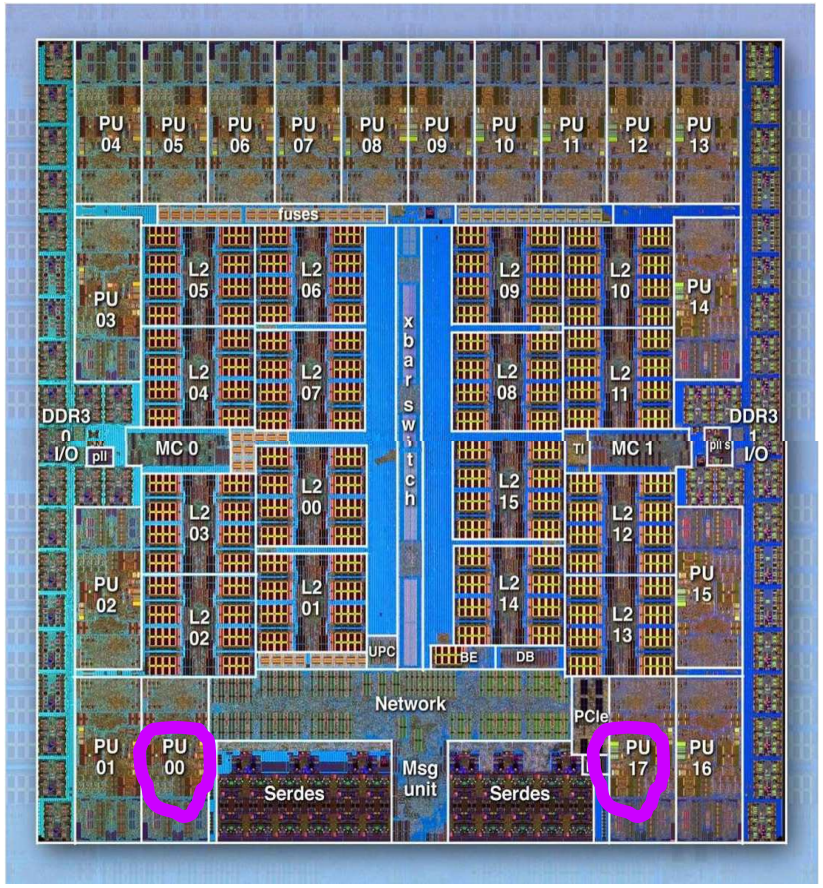
System from TOP 500	Max performance	Checkpoint time (minutes)
LLNL Zeus Lawrence Livermore National Laboratory	11 TeraFLOPS	26
LLNL BlueGene/L	500 TeraFLOPS	20
Argonne BlueGene/P	500 TeraFLOPS	30
LANL RoadRunner **Los Alamos Natlional Labs**	1 PetaFLOPS	~ 20

Cappello F. 2009. Fault Tolerance in Petascale/ Exascale Systems: Current Knowledge, Challenges and Research Opportunities. International Journal of High Performance Computing Applications 23, 3, 212–226.

Откакзоустойчивость вычислений

- Уже в обозримом будущем возможность использования всей вычислительной системы для решения большой задачи окажется под вопросом

IBM PowerPC® A2 1.6 GHz, 16 ядер на процессор



Robert W. Wisniewski.

BlueGene/Q: Architecture,

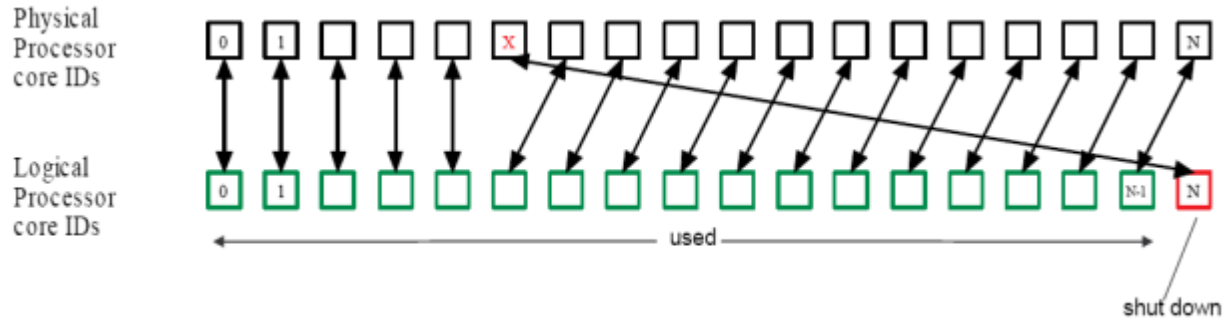
CoDesign; Path to Exascale / Blue

Gene Supercomputer Research,

January 25, 2012

Однако ядер 18, а не 16
Одно – сервисное
Одно – запасное

В последних процессорах поддерживается горячая замена ядра



Уровни управления контрольными точками

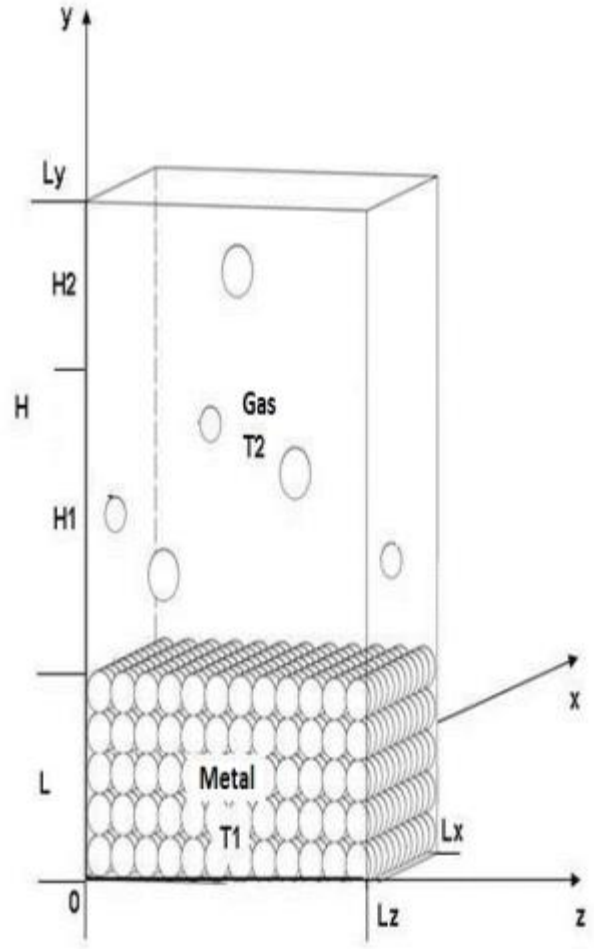
- Системный уровень
  - Простота использования
- Уровень пользователя
  - Радикальное сокращение объёмов контрольных точек
  - Вместо рестарта всей системы - замена вычислительного узла
  - Хранение данных не только на локальных дисках HDDs но и в оперативной памяти

Численное моделирование

- Масштабное МД моделировании:
  - – - 5 вариантов расчетов
  - – - 3 разных вычислительных ресурса:
    - MVS10-P (MSC RAS)
    - K1 (NICEVT)
    - IMM6 (KIAM RAS)

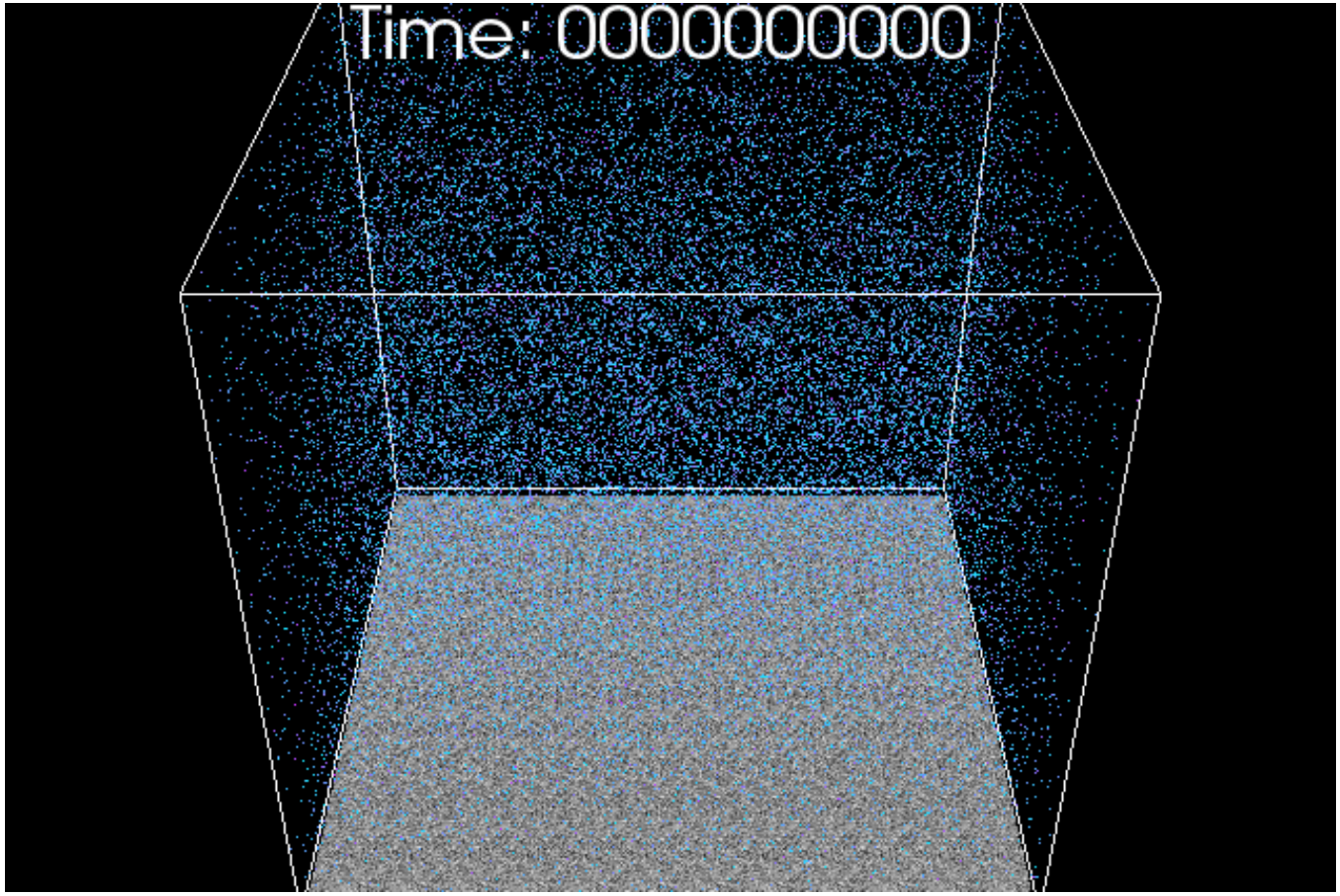
Проблемы относительно управления задачами:

- Ручной запуск и мониторинг задач
- Ручная переброска данных с одного ресурса на другой
- Квота на дисковое пространство



Моделирования взаимодействия Ni-N2

Size: 8 128 512 + 423 840 = 8 552 352 particles,
Temperature T_{Ni} = 273.15 K, T_{N2} = 273.15 K



The problem is split into gas dynamics and molecular dynamics:
Flow and Particles

ULFM – User-Level Failure Mitigation

Версия MPI 3.1 не имеет механизмов управления и штатной работы при отказах

Стандарт ULFM предложен в качестве минимального, но достаточного интерфейса, обеспечивающего восстановление MPI приложений в случае отказа отказах

- **MPI_COMM_REVOKE**
- **MPI_COMM_SHRINK**
- **MPI_COMM_FAILURE_GET_ACKED**
- **MPI_COMM_FAILURE_ACK**
- **MPI_COMM_AGRE**

http://fault-tolerance.org/

ULMF часть версии MPI 4.1

HPC вызов

- Разработка принципов управления контрольными точками, при которых время накладных расходов меньше чем MTBF
- Разработка алгоритмов, дающих возможность продолжать расчет даже при регулярных отказах части процессов

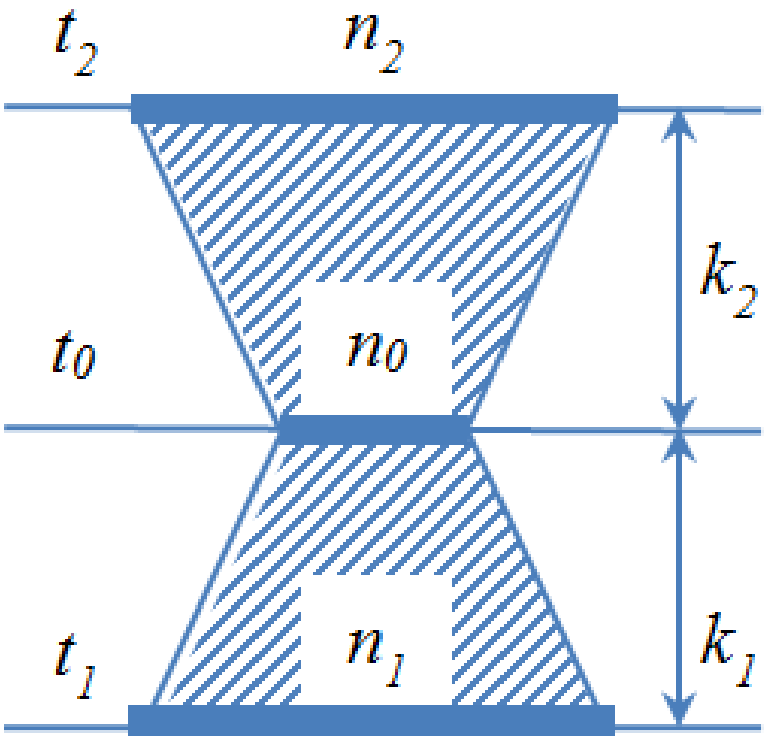
Одномерное гиперболическое уравнение

$$\frac{\partial^2 \Phi}{\partial x^2} - \frac{1}{C^2}\frac{\partial^2 \Phi}{\partial t^2} = F(x,t)$$

Две характеристики $x - ct$ и $x + ct$, определяющими область, влияющую на решение $\Phi(x,t)$ в точке (x,t)

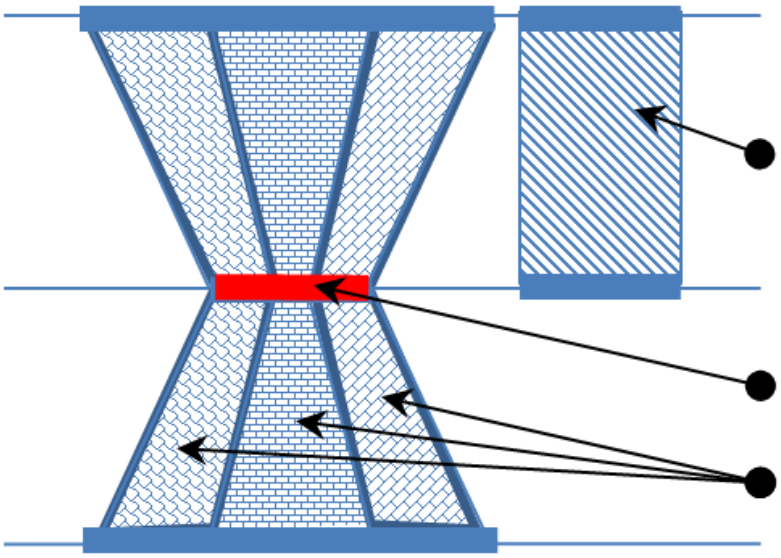
Геометрические размеры области на

Область ускоренного расчета при возмещении потери данных, вызванной выходом из строя одного процессора



Подход применим для гиперболических систем и для любых явных разностных схем

Замена испорченного процессора тремя запасными



Полное восстановление расчета

Продолжение работы исправных процессоров

Момент поломки процессора

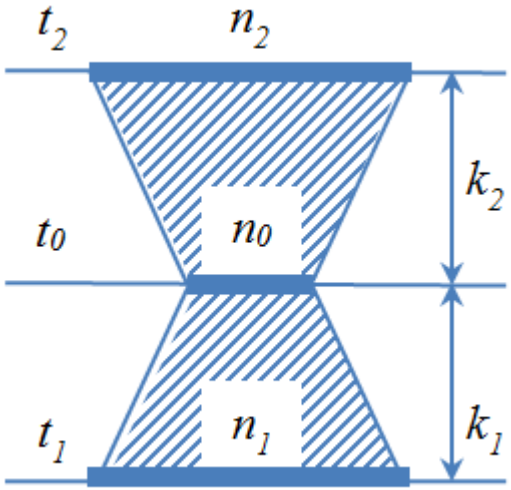
Утерянная часть расчетной сетки

Перерасчет части данных

Последняя локальная контрольная точка

Оценка числа дополнительных процессоров

$$p_d > \frac{1}{d+1}\frac{1}{k_2}\sum_{j=1}^{2} k_j \sum_{i=0}^{d} {\alpha_j}^i$$



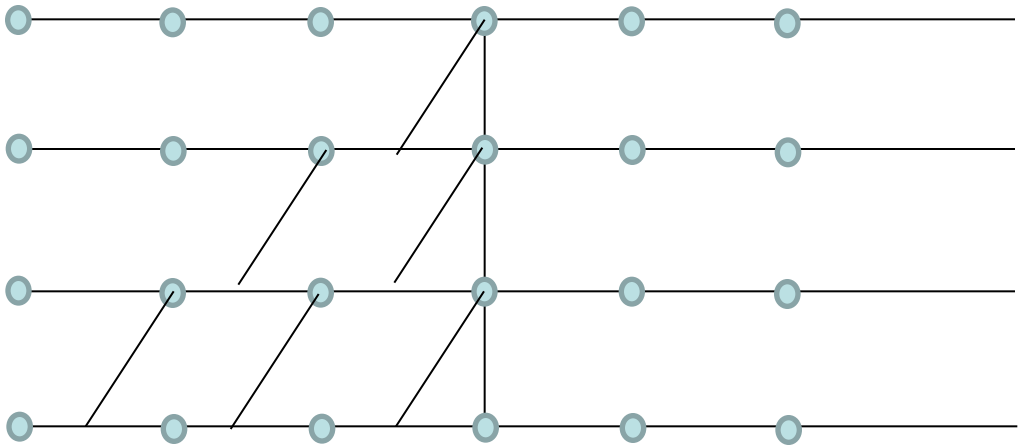
$$\alpha_j = 1 + 2\gamma\frac{k_j}{n_0}$$

$\gamma = \frac{C\Delta t}{h}$ - число Куранта

d – *размерность пространства*

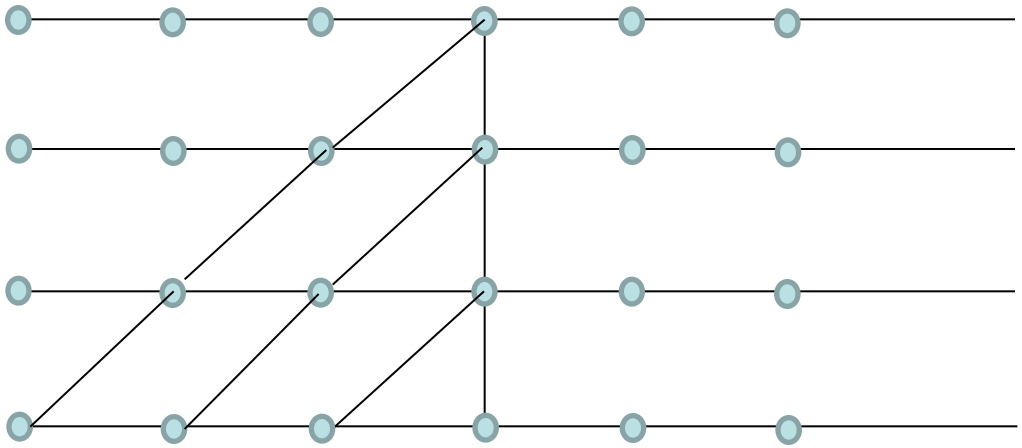
n_0^d - *число точек обрабатываемых одним процессором*

$\gamma < 1 - approximate\ result$



$\gamma = 1$ Точный результат

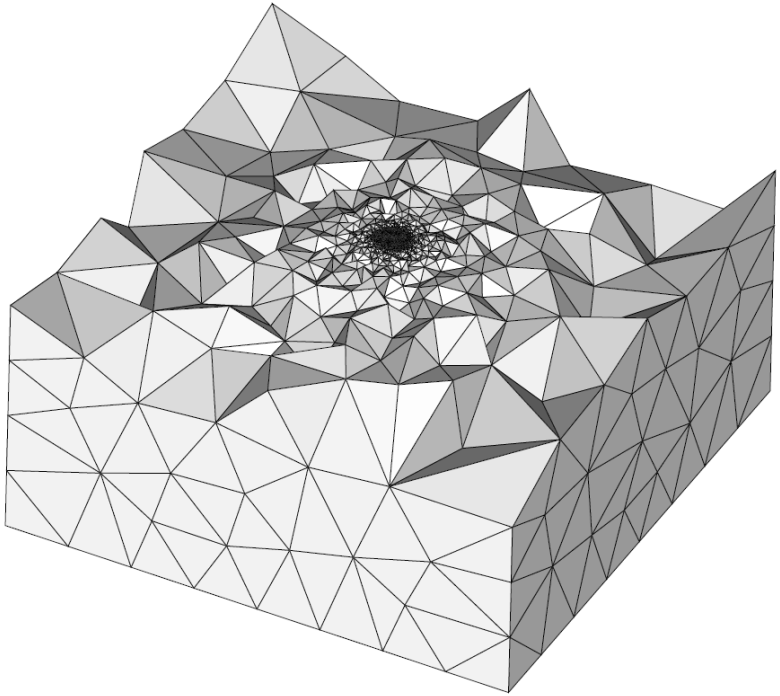
- Только гиперболические?
  - Нет, любые явные схемы

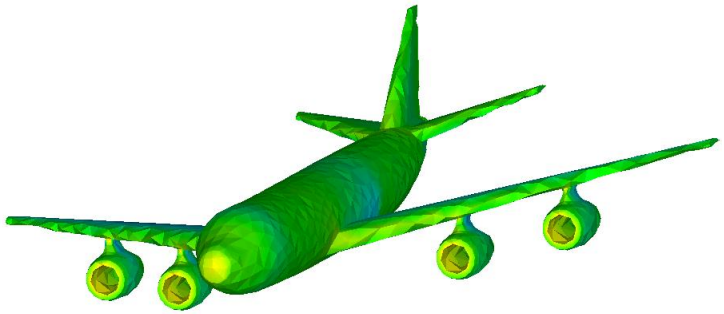


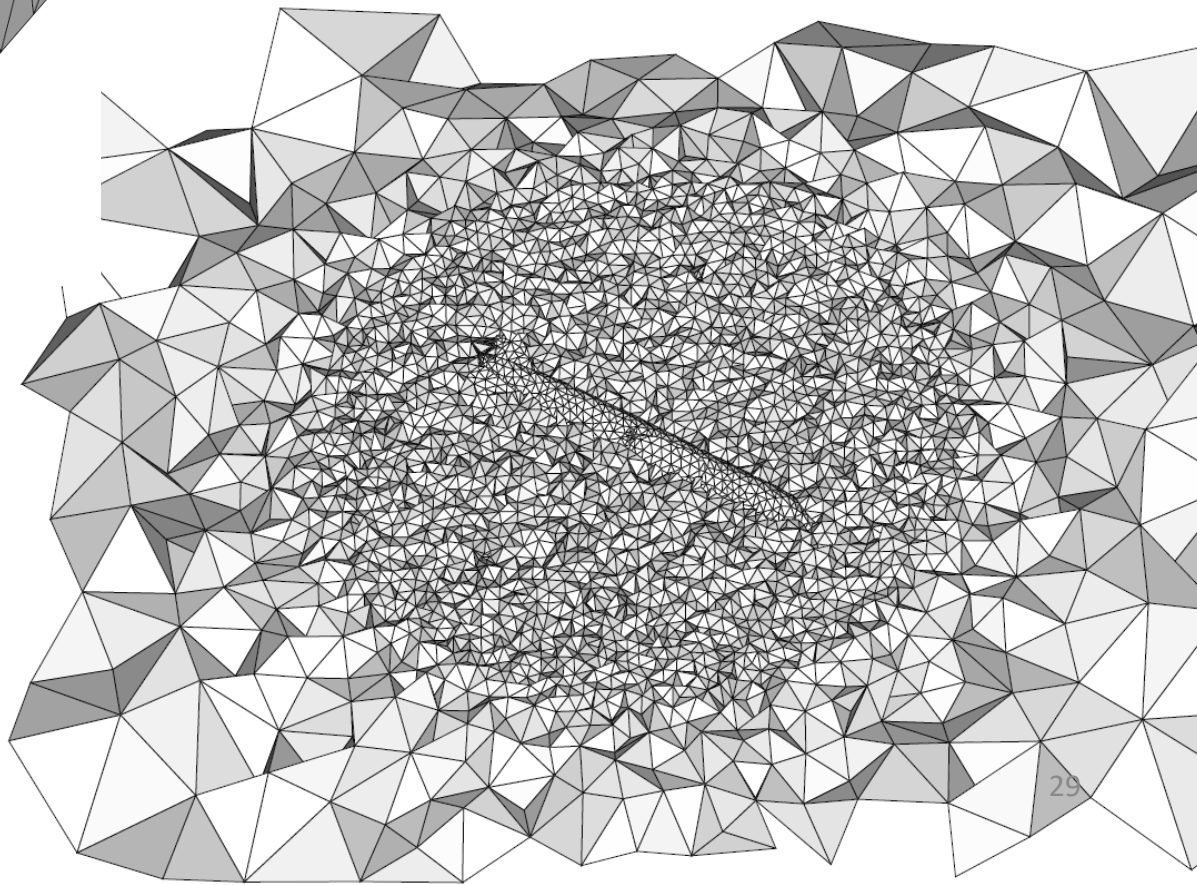
- Каждый шаг увеличивает зону влияния на 1 ячейку

Предложен метод, обеспечивающий независимость времени расчета от факта возникновения отказов, в том числе множественных

Тетраэдральные сетки 10^8 узлов



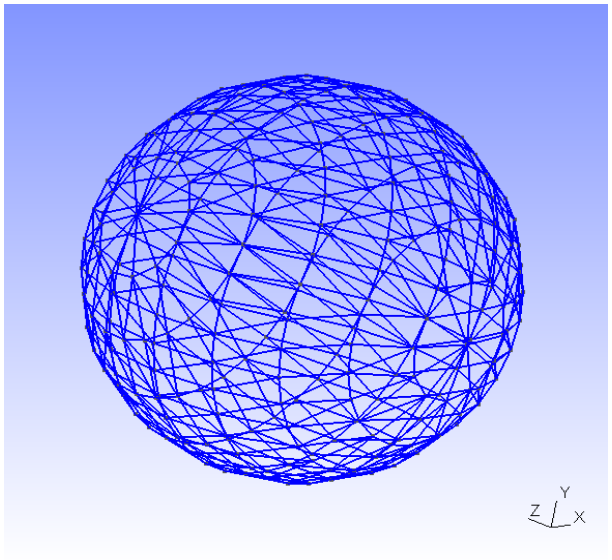




Постановка задачи

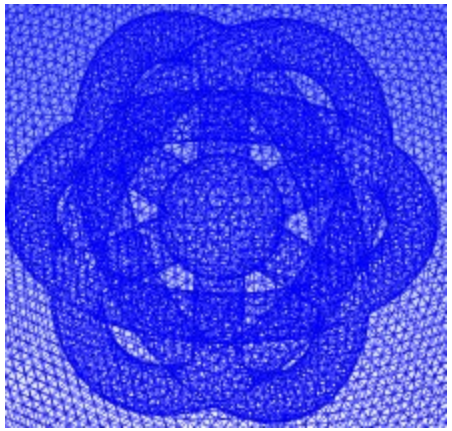
- Дано:

  Замкнутая поверхность, заданная конечным множеством согласованных плоских треугольников

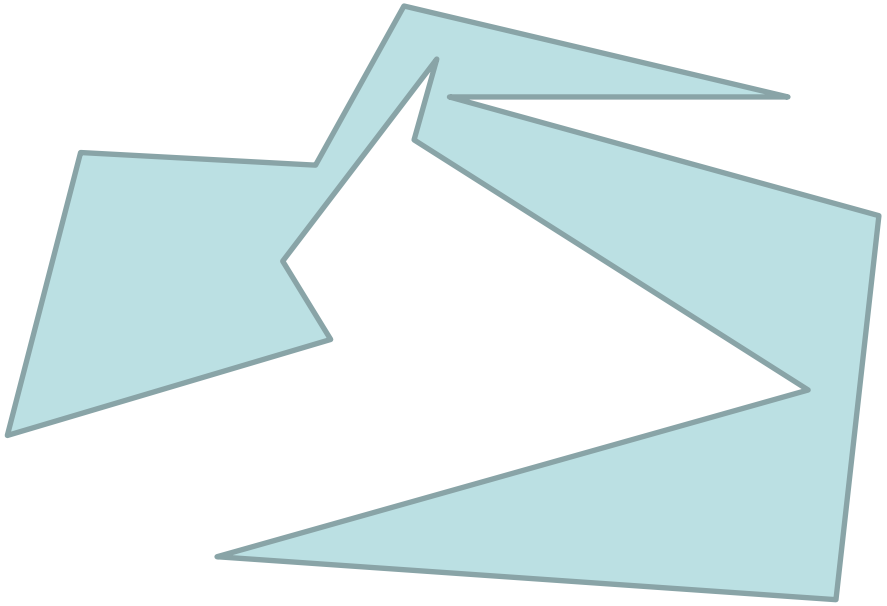


- Требуется:

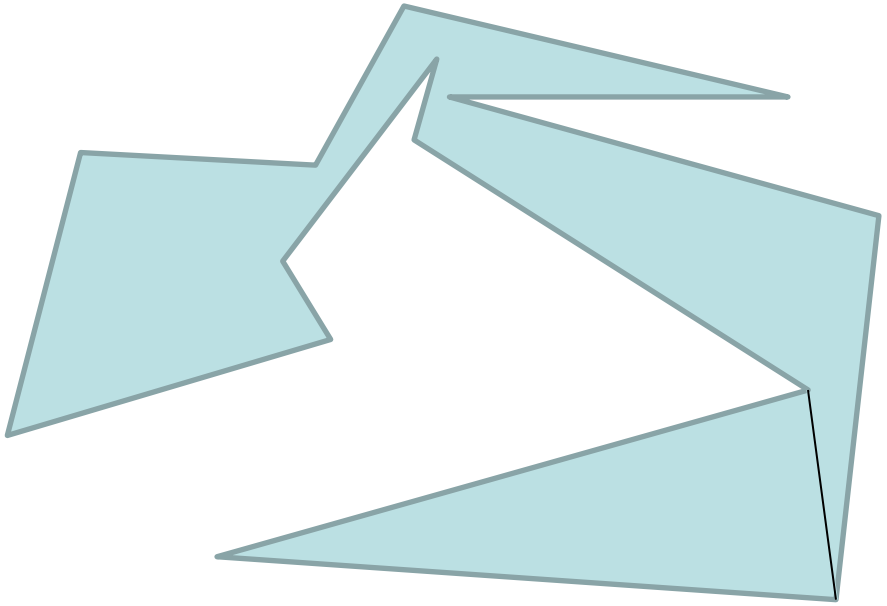
  Построить множество согласованных тетраэдров, совокупность которых полностью заполняет объём трёхмерной конечной фигуры, ограниченной заданной поверхностью



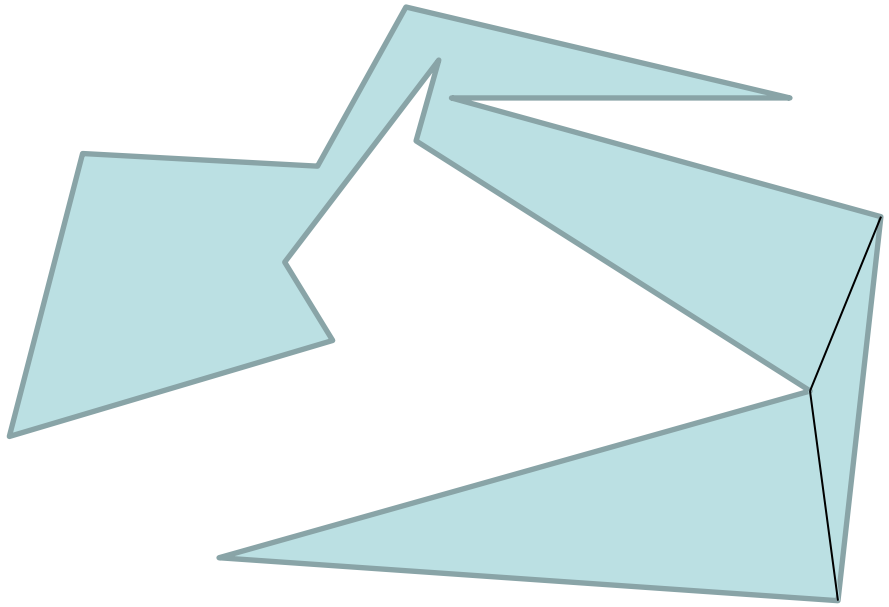
Триангуляция многоугольника



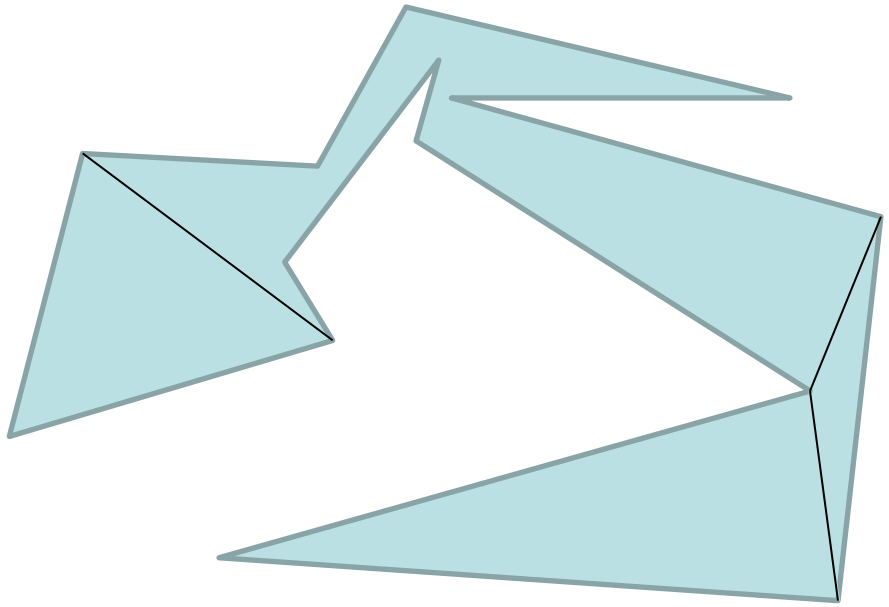
Триангуляция многоугольника



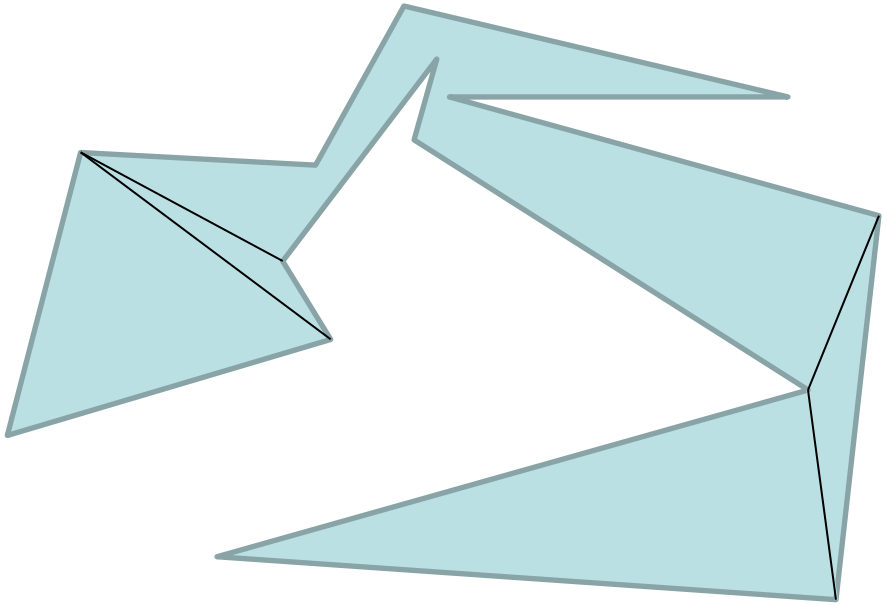
Триангуляция многоугольника



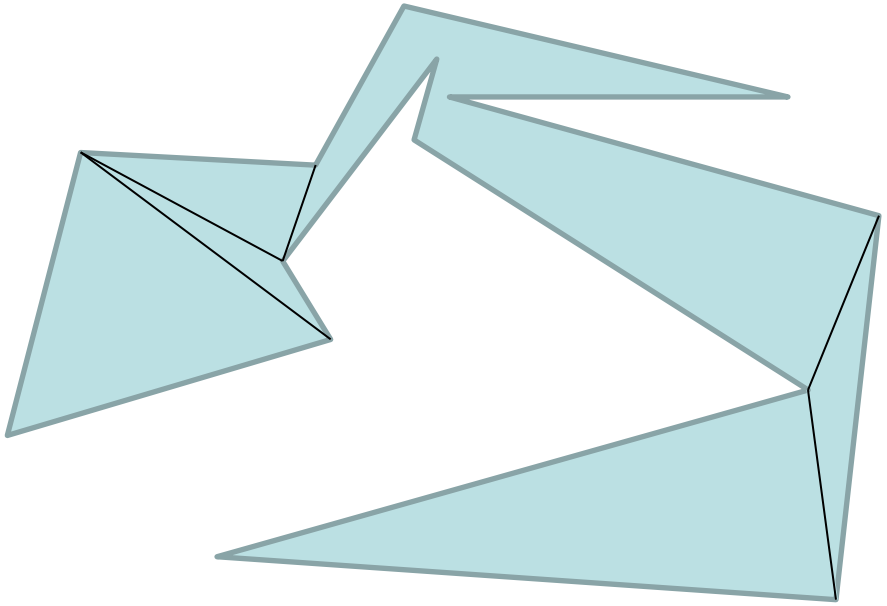
Триангуляция многоугольника



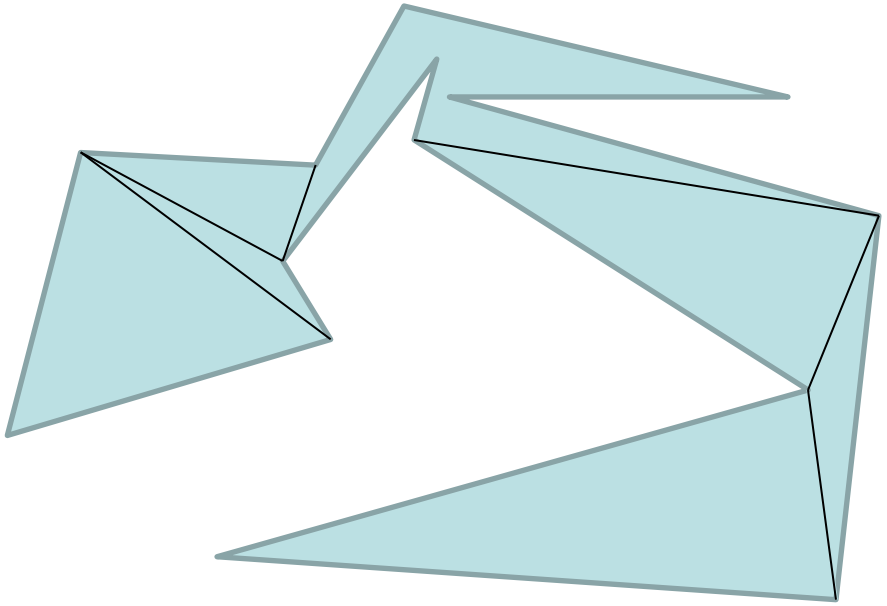
Триангуляция многоугольника



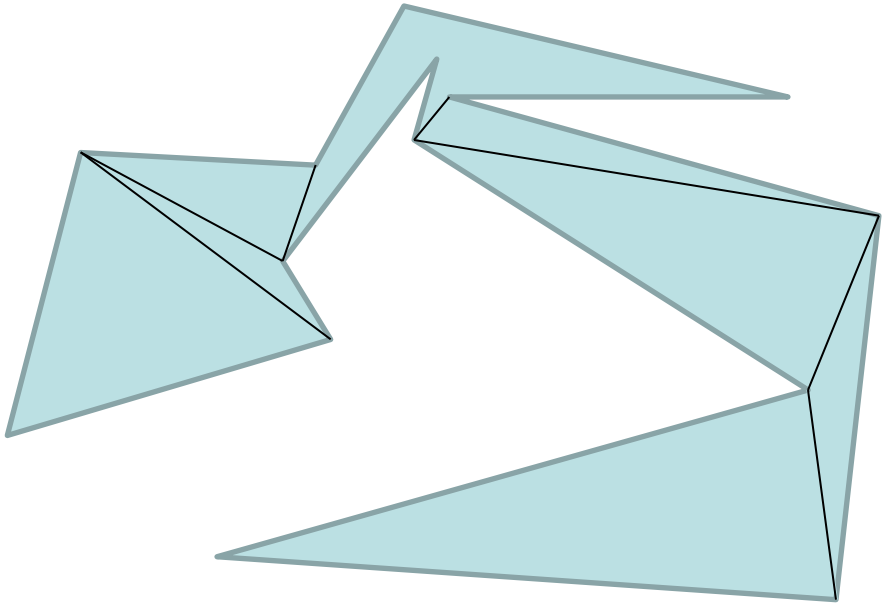
Триангуляция многоугольника



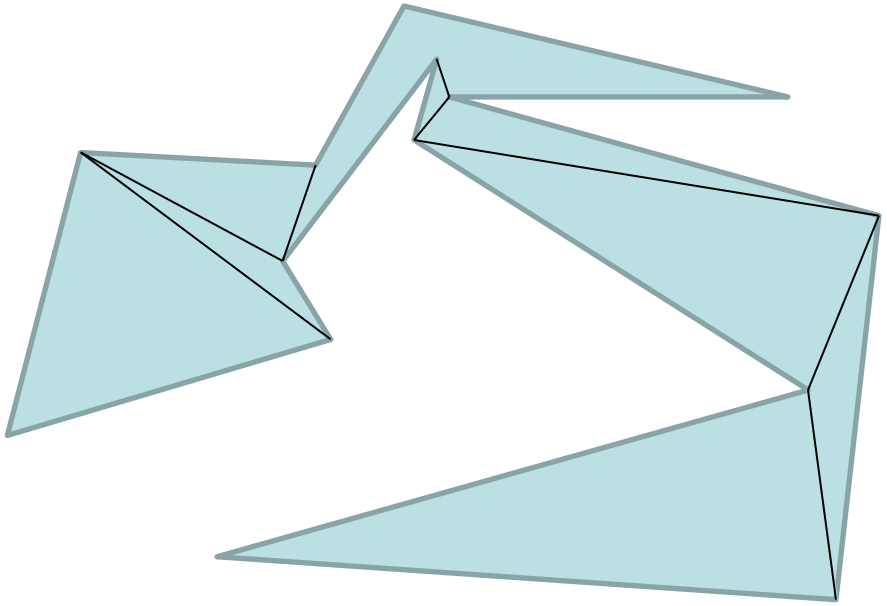
Триангуляция многоугольника



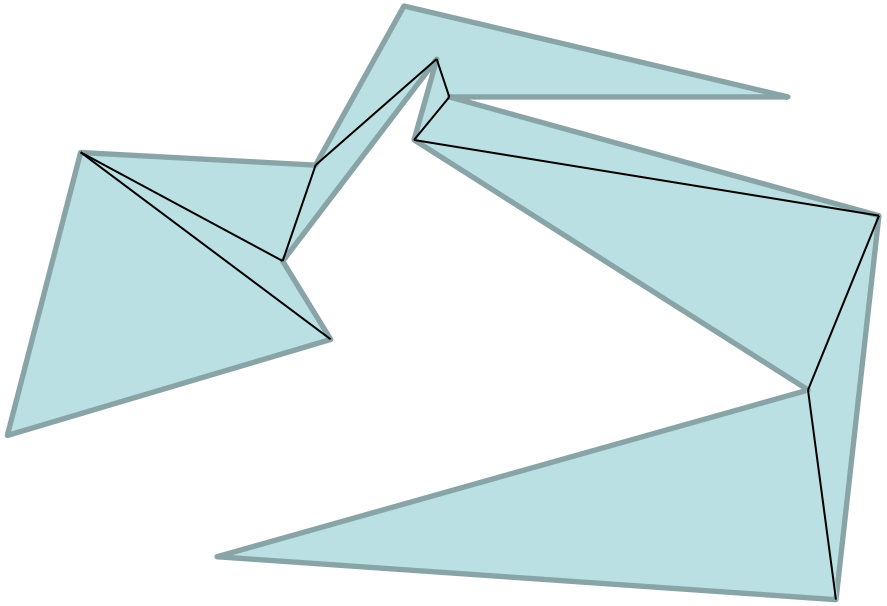
Триангуляция многоугольника



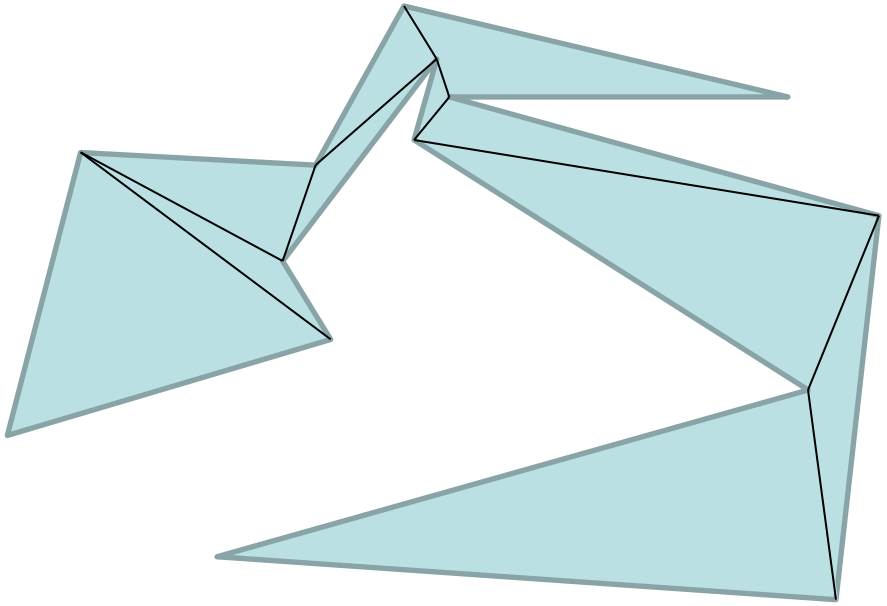
Триангуляция многоугольника



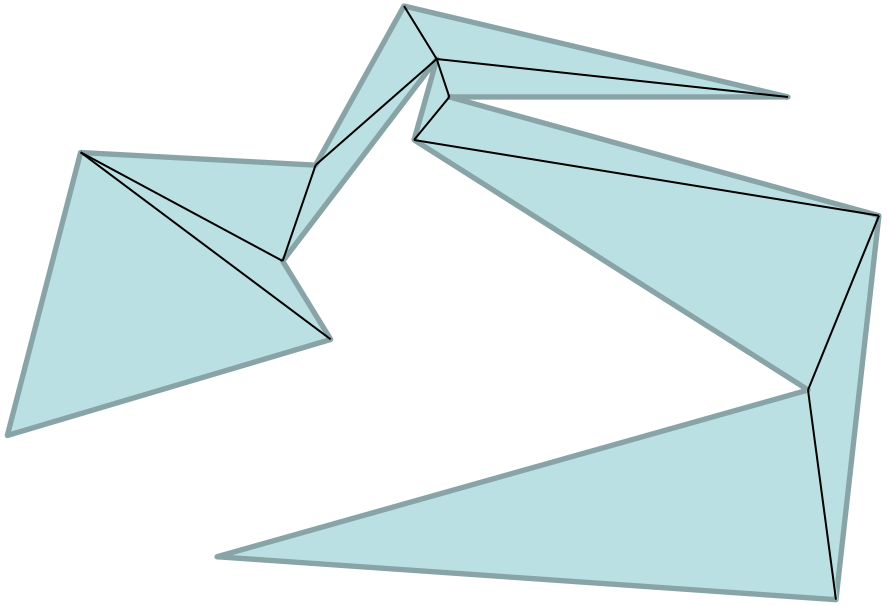
Триангуляция многоугольника



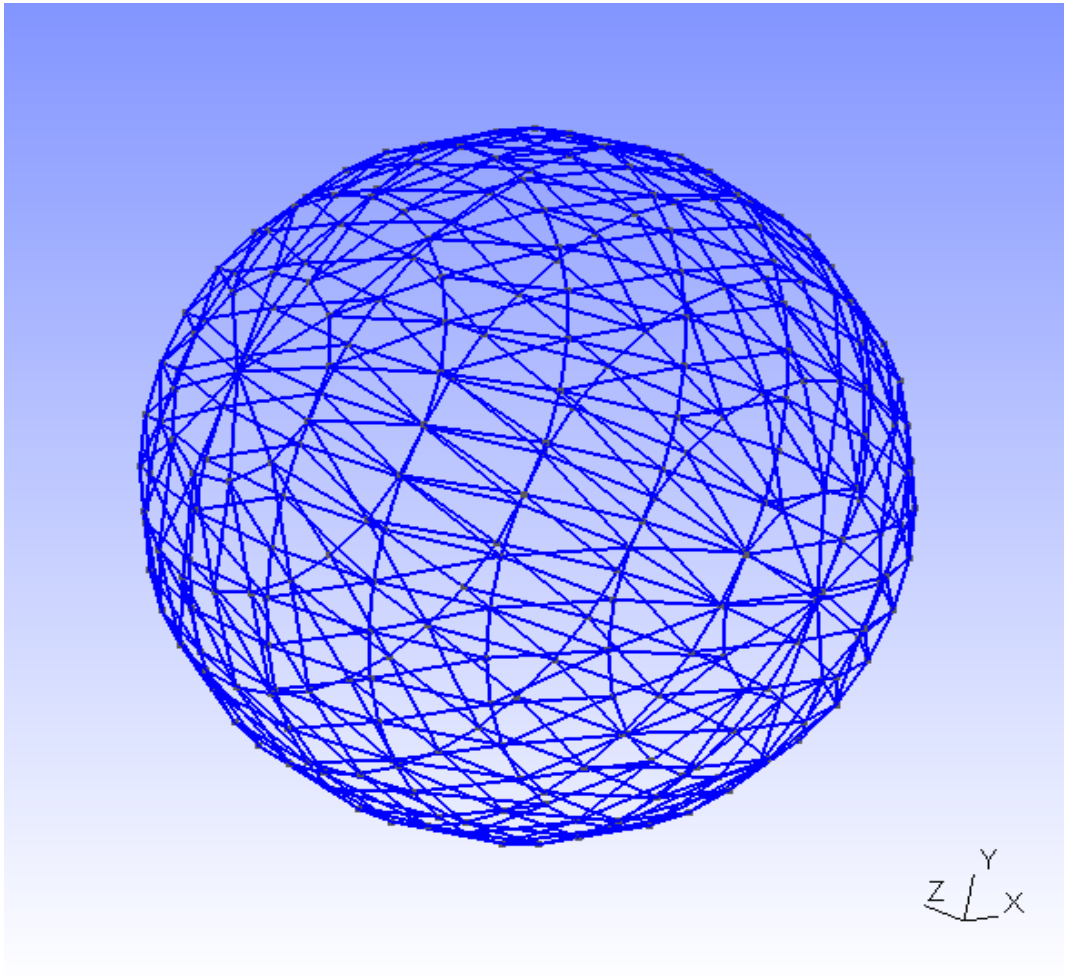
Триангуляция многоугольника



Триангуляция многоугольника

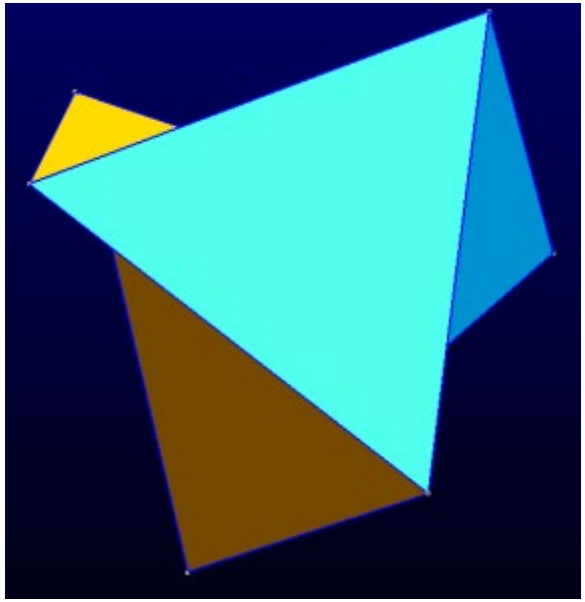


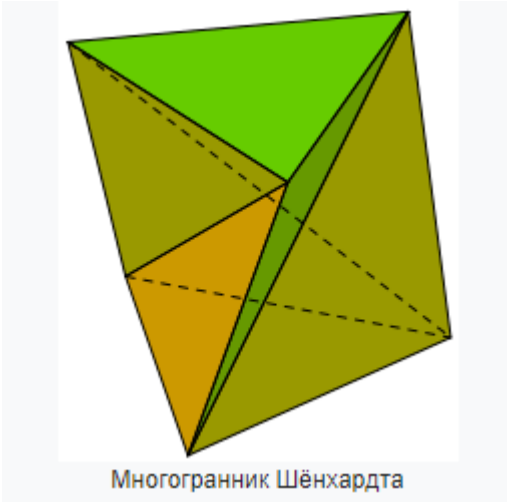
Аналогичный метод не работает в 3D



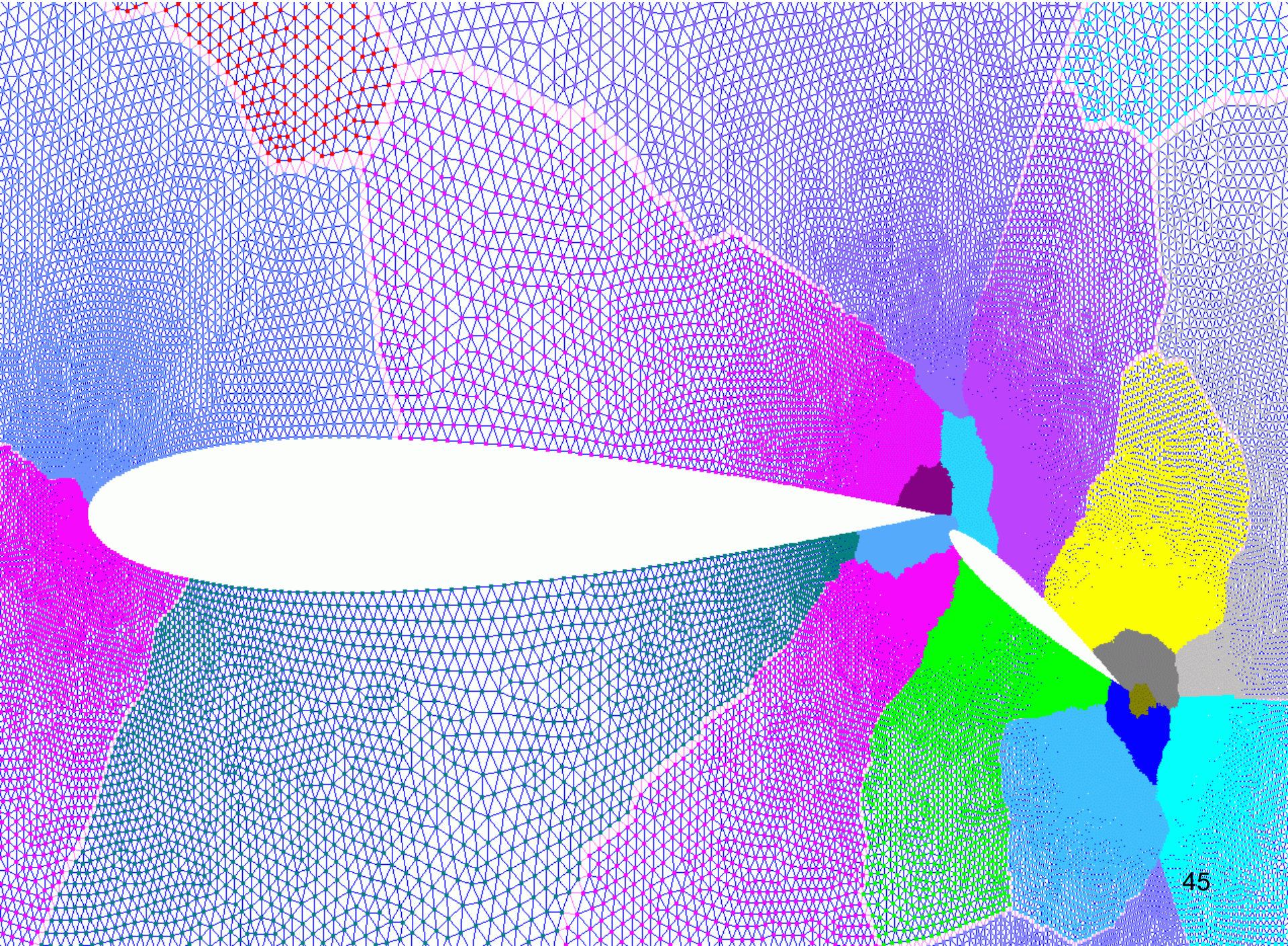
Многогранник Шёнхардта

- # points=6 triangles=8
- v -0.866 -0.5 0
- v 0 1 0
- v 0.866 -0.5 0
- v -0.342 -0.94 1
- v -0.643 0.766 1
- v 0.985 0.174 1
- f 3 2 1
- f 6 4 5
- f 4 1 2
- f 5 4 2
- f 5 2 3
- f 6 5 3
- f 6 1 4
- f 6 3 1

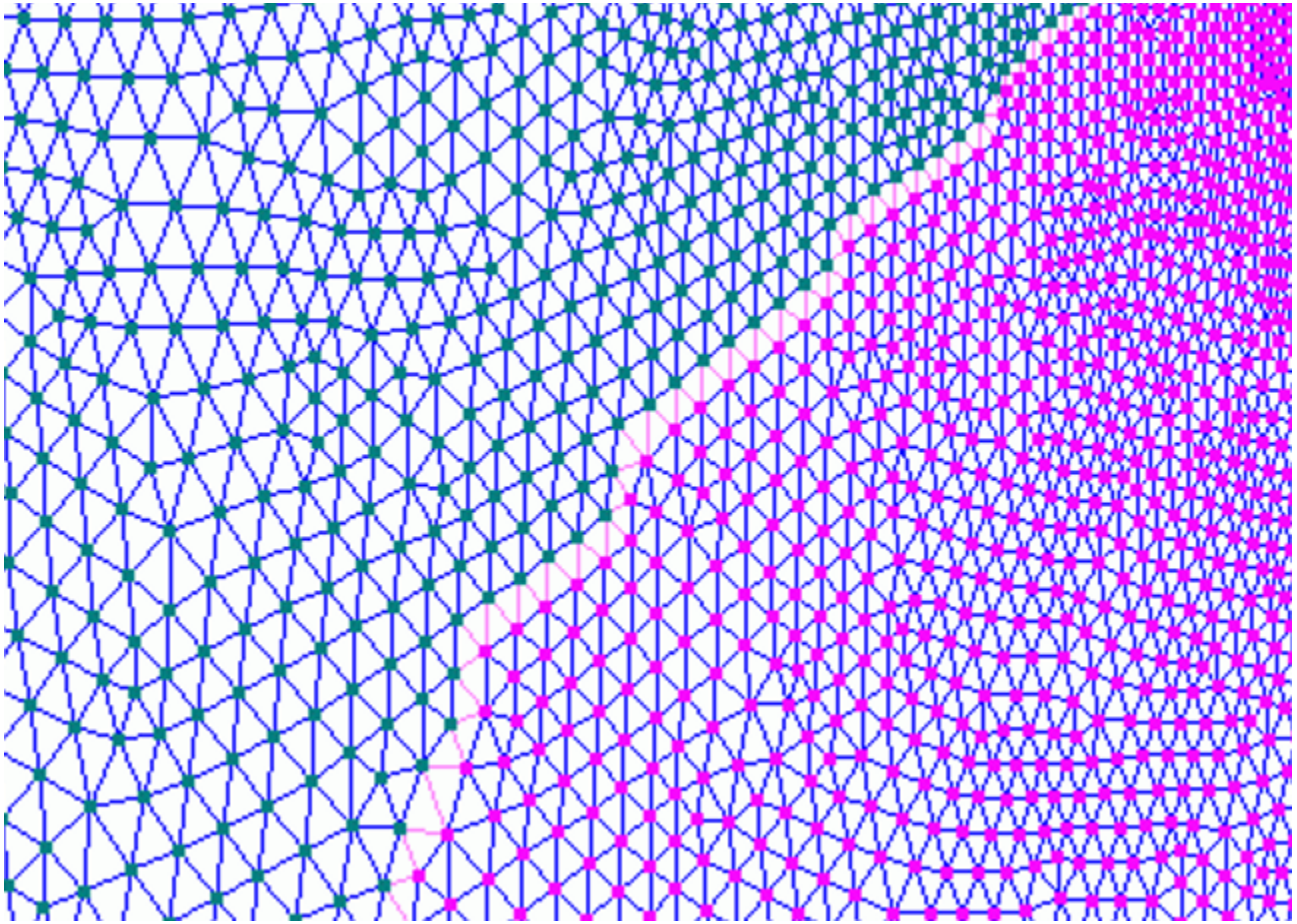




Невозможно разбить многогранник Шёнхардта на тетраэдры, все вершины которых являются вершинами многогранника

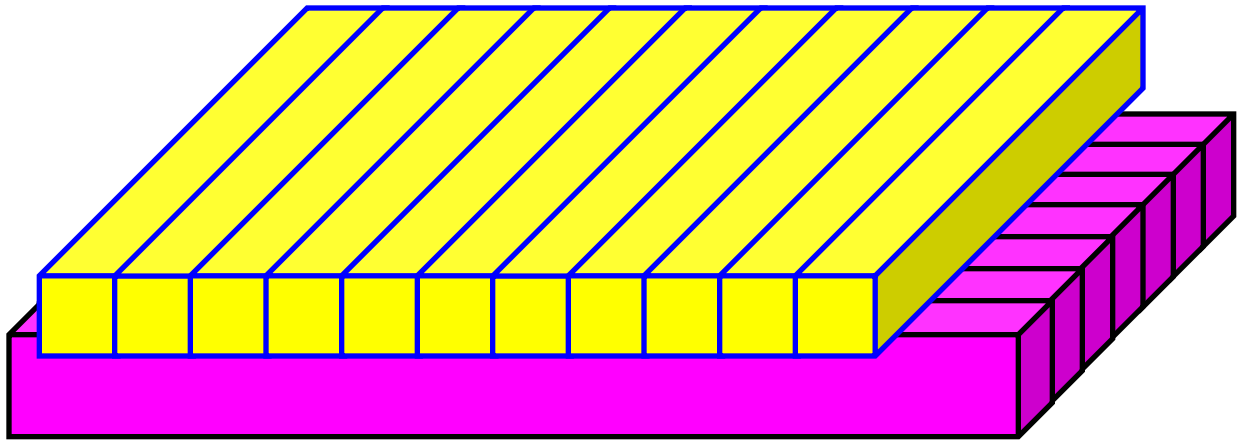


В 2D в среднем меньше 6 соседей



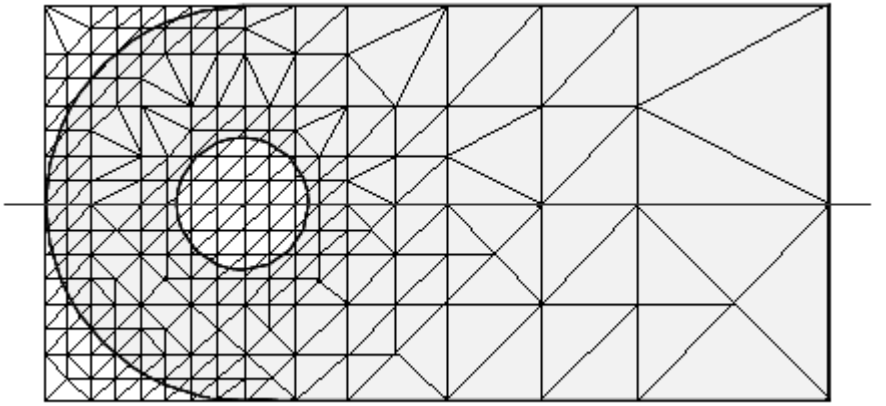
В 3D в среднем _??_ соседей

В 3D в среднем больше $\frac{n}{2}$ соседей



Доступные методы генерации расчетных сеток

- Граничная коррекция

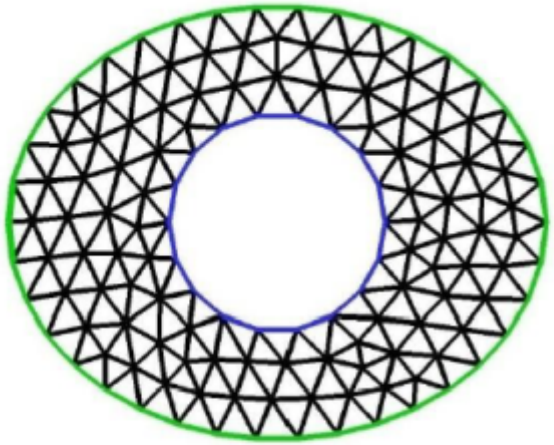


- Делоне с ограничениями

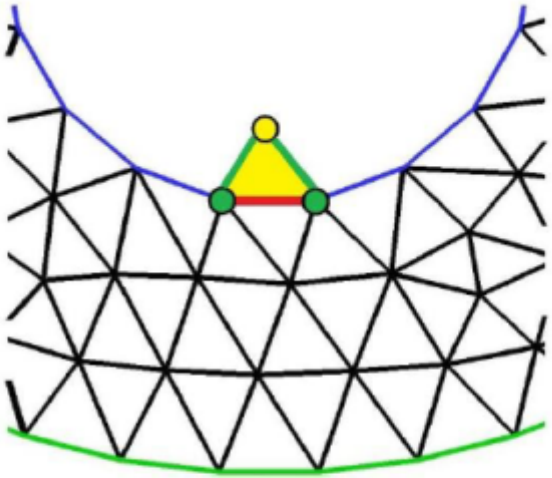
- Движущийся фронт

Галанин М.П., Щеглов И.А. Препринты ИПМ 2006
Разработка и реализация алгоритмов трехмерной триангуляции сложных пространственных областей

Метод движущегося фронта



а) Общий вид триангулируемой области: зеленая линия – граница области, синяя – текущий фронт, черные линии – ребра элементов уже построенной сетки



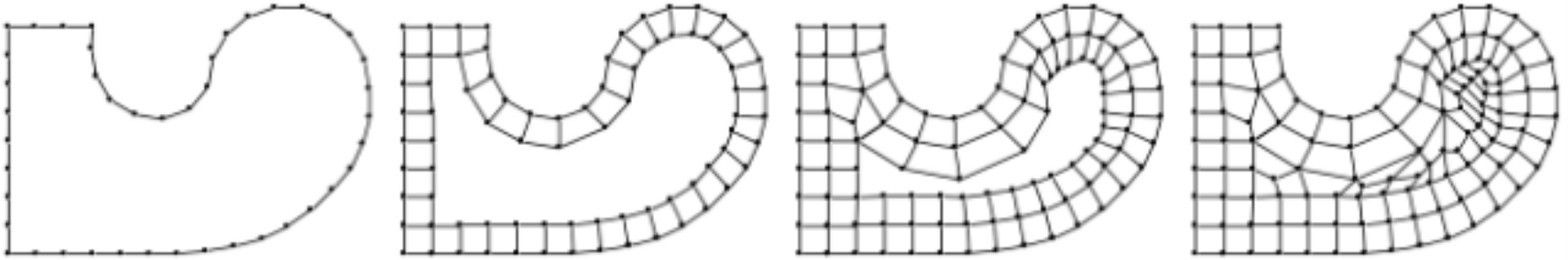
б) Добавление узла с продвижением фронта

Суков С.А. Методы генерации тетраэдральных сеток и их программные реализации // Препринты ИПМ им. М.В.Келдыша.
2015. № 23. 22 с. URL: **http://library.keldysh.ru/preprint.asp?id=2015-23**

Фронтальный метод

Проблема замыкания движущегося фронта

Отсутствие доказательства сходимости алгоритмов движущегося фронта

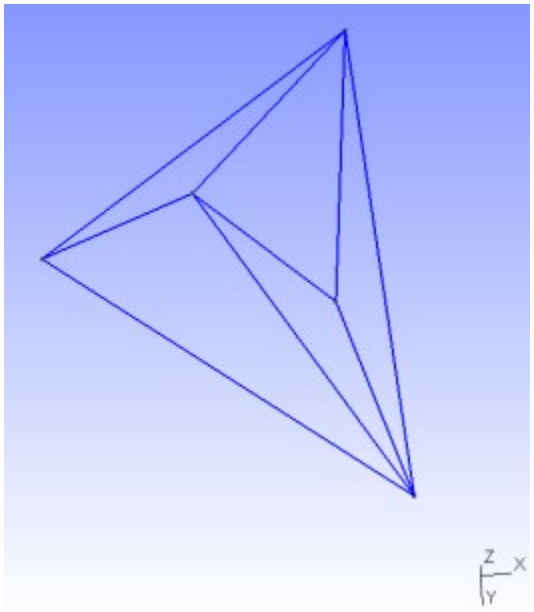


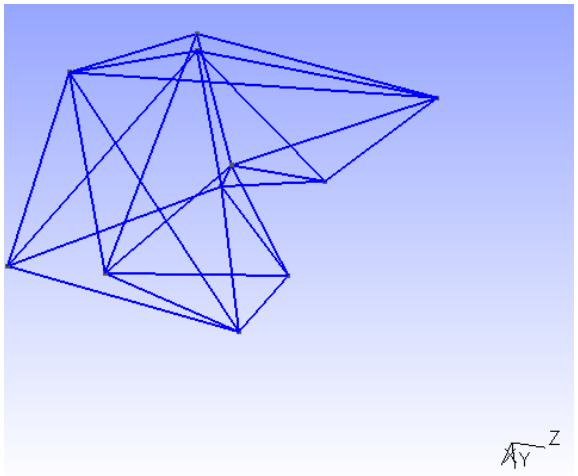
МЕТОДЫ И ПОДХОДЫ К МОДЕЛИРОВАНИЮ ГЕОМЕТРИЧЕСКИХ ОБЪЕКТОВ В КОНТАКТНЫХ ЗАДАЧАХ

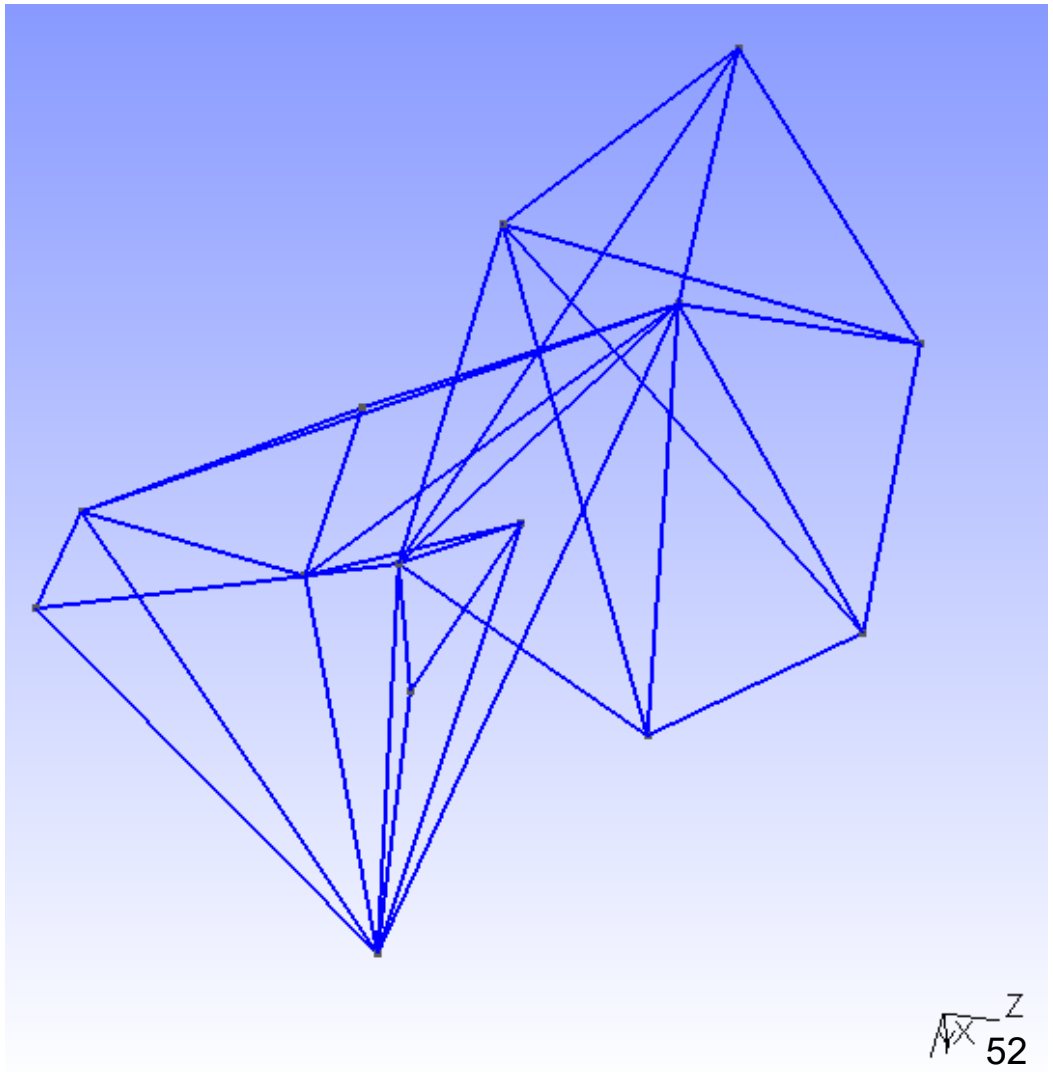
Снежкова Л. С., аспирант, Чопоров С. В., к. т. н.

Запорожский национальный университет,

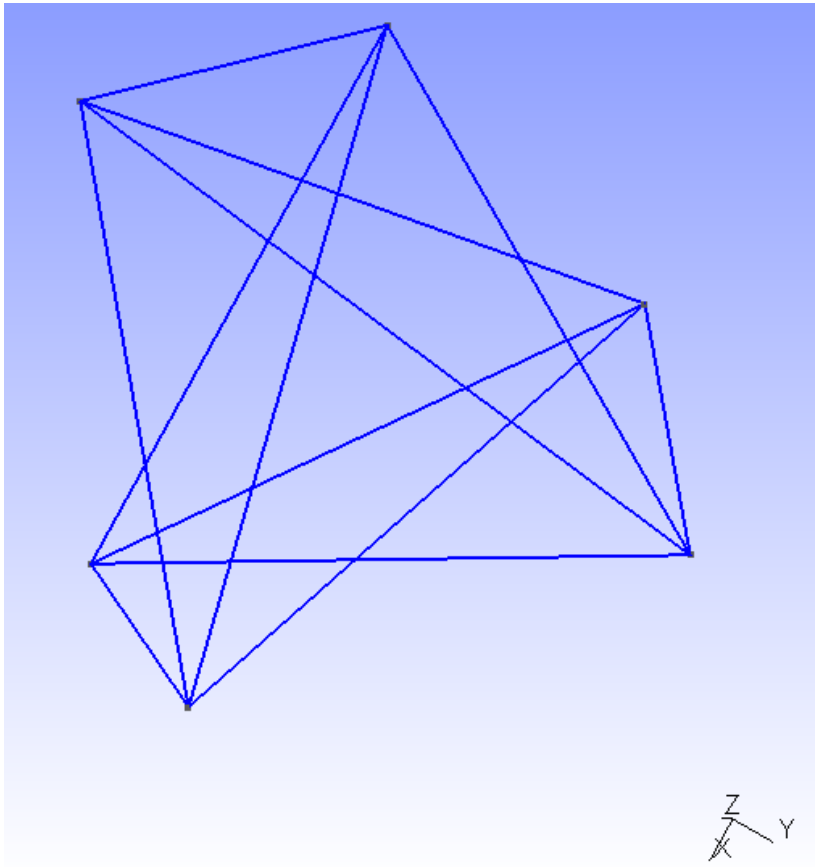
3D лакуны – фрагменты, не заполненные фронтальным методом

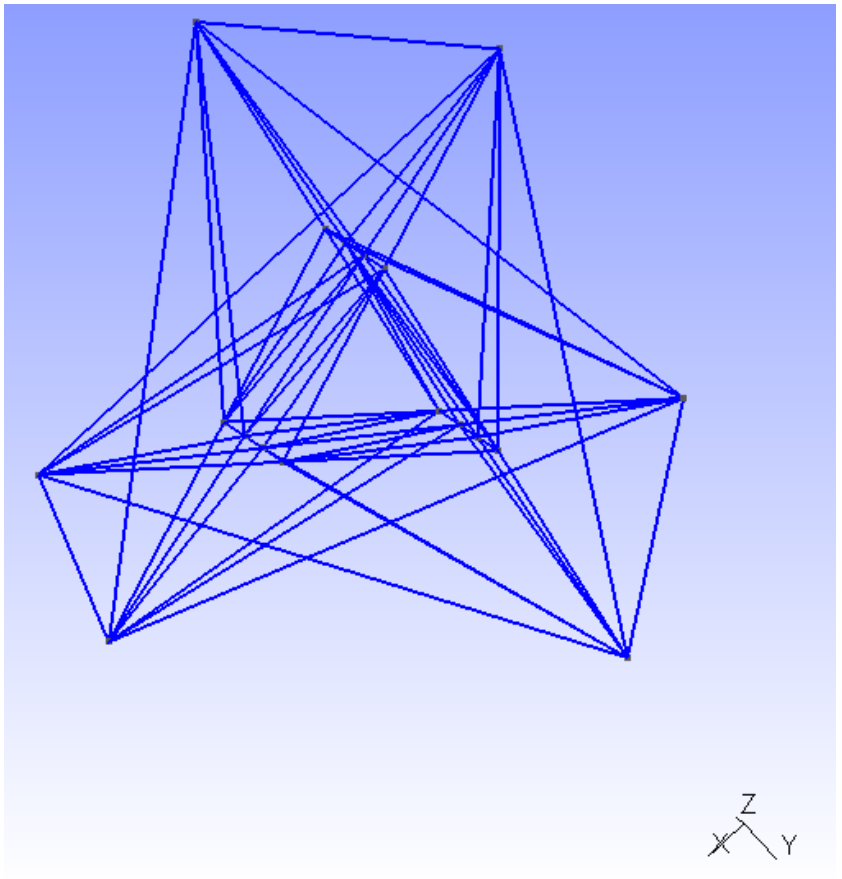




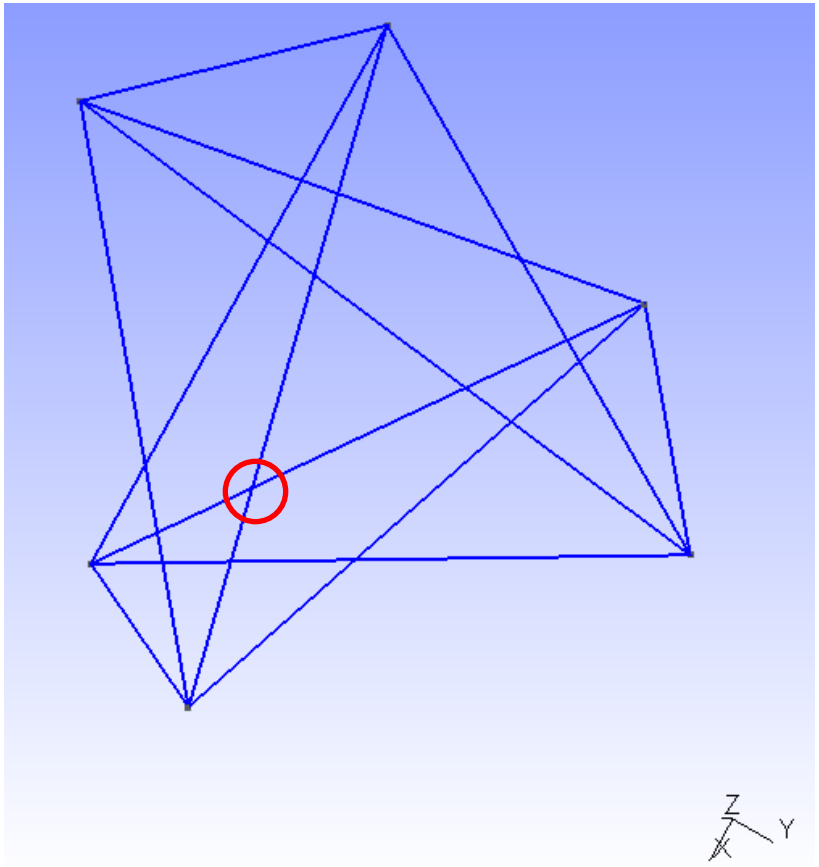


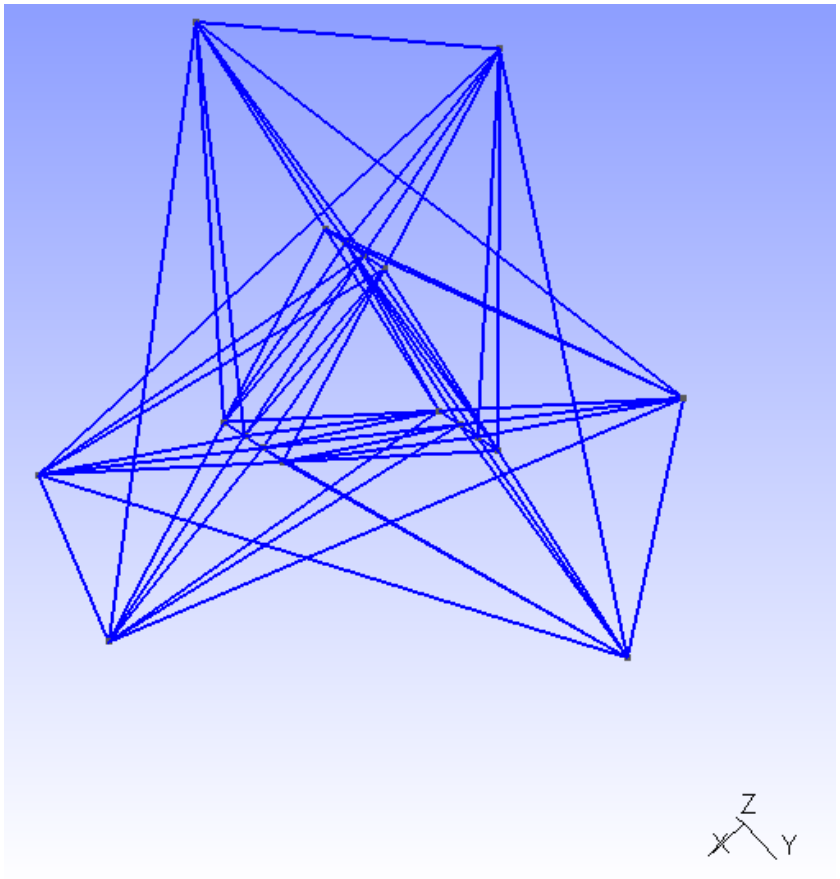
Прямой метод заполнения выпуклыми полиэдрами



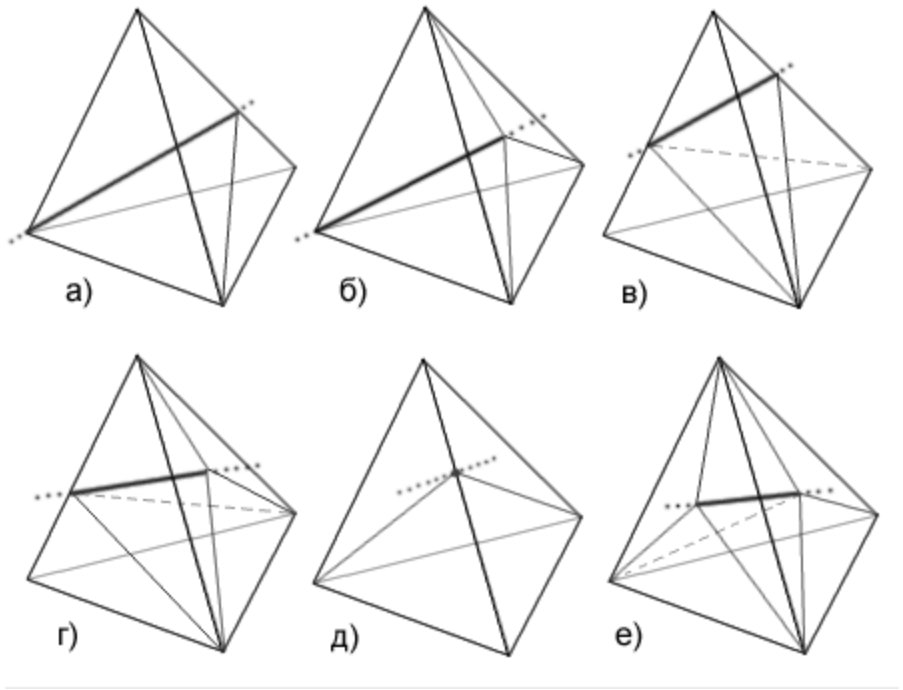


1) определение пересечения проекций ребер
2) заполнение тетраэдрами призм, ограниченных проекциями рёбер



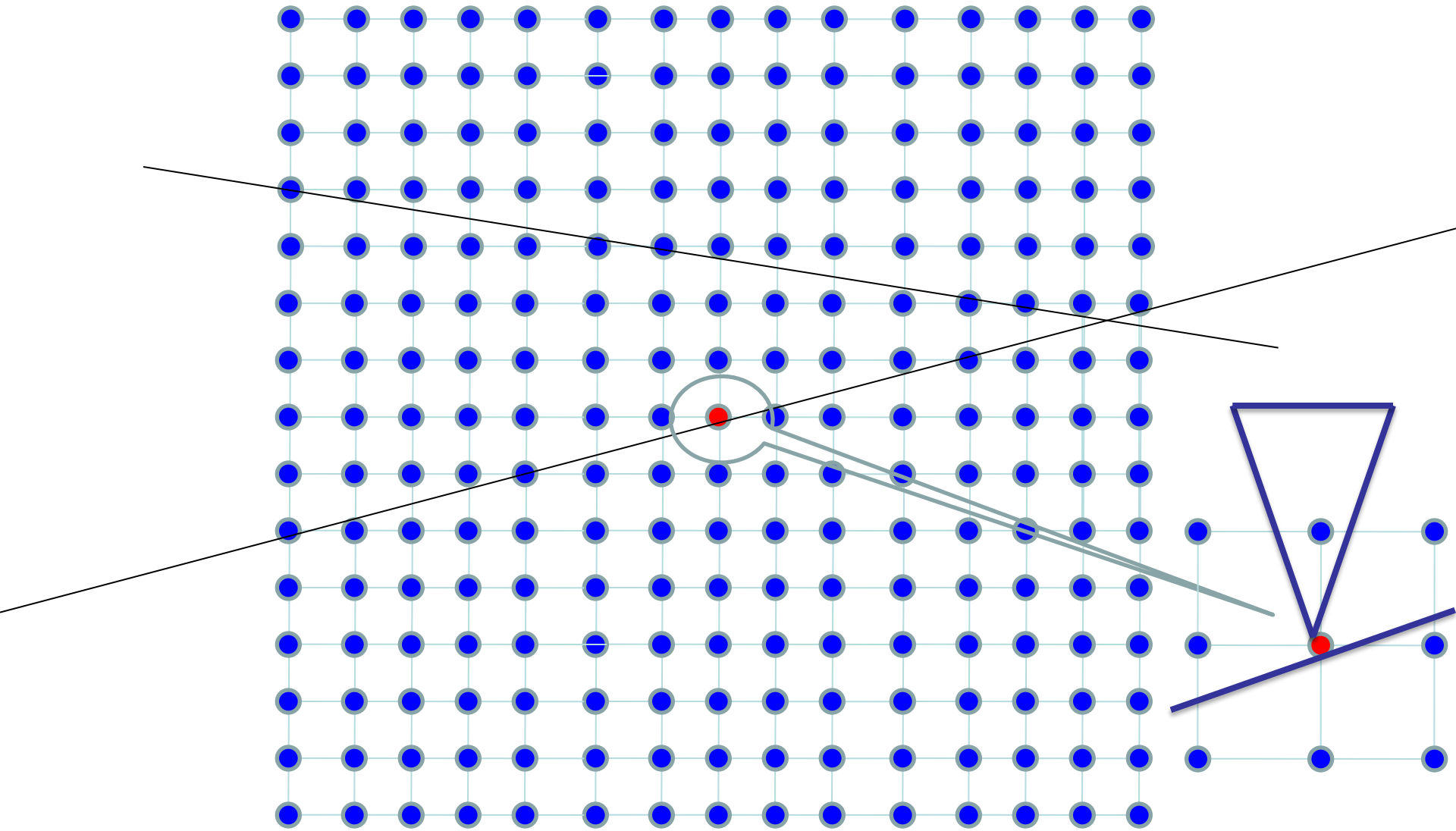


Пересечение тетраэдра и прямой

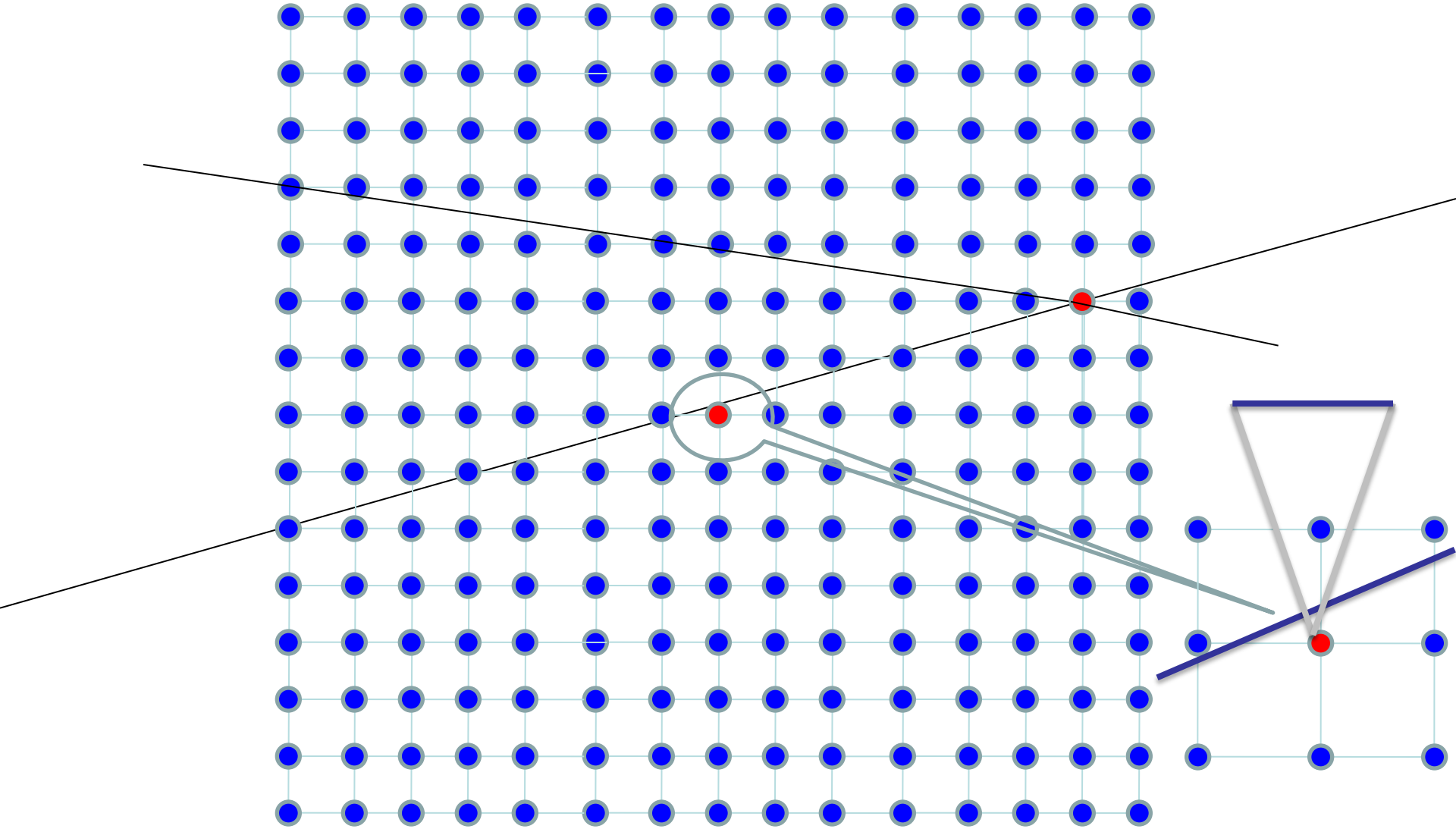


Галанин М.П., Щеглов И.А. Препринты ИПМ 2006 Разработка и реализация алгоритмов трехмерной триангуляции сложных пространственных областей

Пересечения на сетке вещественных чисел



Пересечения на сетке вещественных чисел



Использование рациональных чисел $\frac{a}{b}$

- Обеспечивает
  - Гарантированное построение покрытия 3D области объёмной сеткой
  - Гарантированно правильное построение топологии пересечения триангулированных объектов (в том числе, лучей, поверхностей и 3D сеток)
  - Отсутствие проблемы выбора ε
- Приложения
  - Генерации сеток
  - Камера обскура
  - Генерация длинных характеристик для моделирования лучистого переноса энергии
  - …

Рациональные числа $\frac{a}{b}$

Преимущества:

- прямой перенос математических формул в код
- упрощение написания и отладки кода
- гарантия правильности кода

Недостаток:

- уменьшение скорости выполнения арифметических операций

Рациональные числа $\frac{a}{b}$

Преимущества:

- прямой перенос математических формул в код
- упрощение написания и отладки кода
- гарантия правильности кода

Недостаток?
-уменьшение скорости выполнения арифметических операций

Преимущество!

- Увеличение отношения вычислительной нагрузки к коммуникационной

Максимально требуемая разрядность НЕ зависит от топологии исходной сетки поверхности 3D объекта

n – начальная разрядность Для ***float*** формата чисел **n < 100**

- максимальная разрядность решения

  $36n + 6$

- максимальная промежуточная разрядность

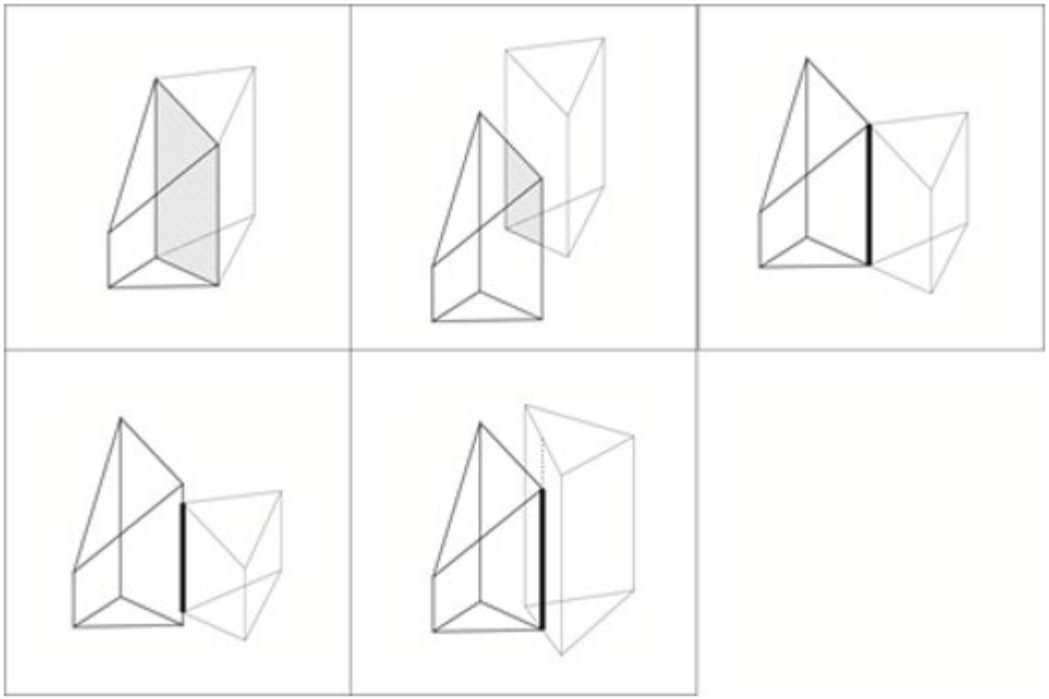
  $252n + 117$

Основные положения и этапы

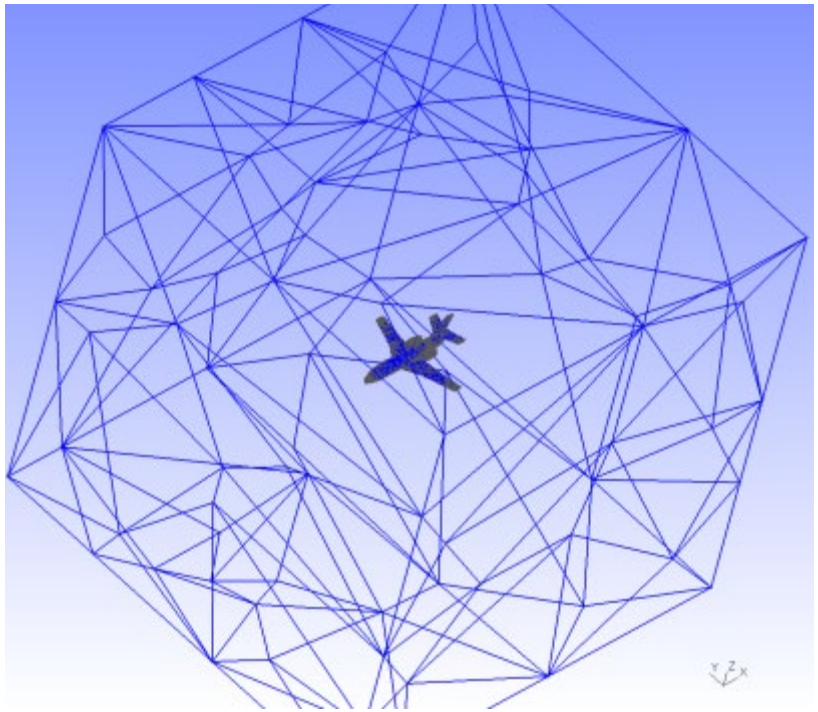
- Выполнение большинства операций в рациональных числах произвольной разрядности
- Построение треугольных призм, боковые грани которых параллельны оси Z
- Согласование боковых граней призм путем включения в каждую грань всех ребер принадлежащих соседним граням
- Построение согласованной триангуляции соприкасающихся граней

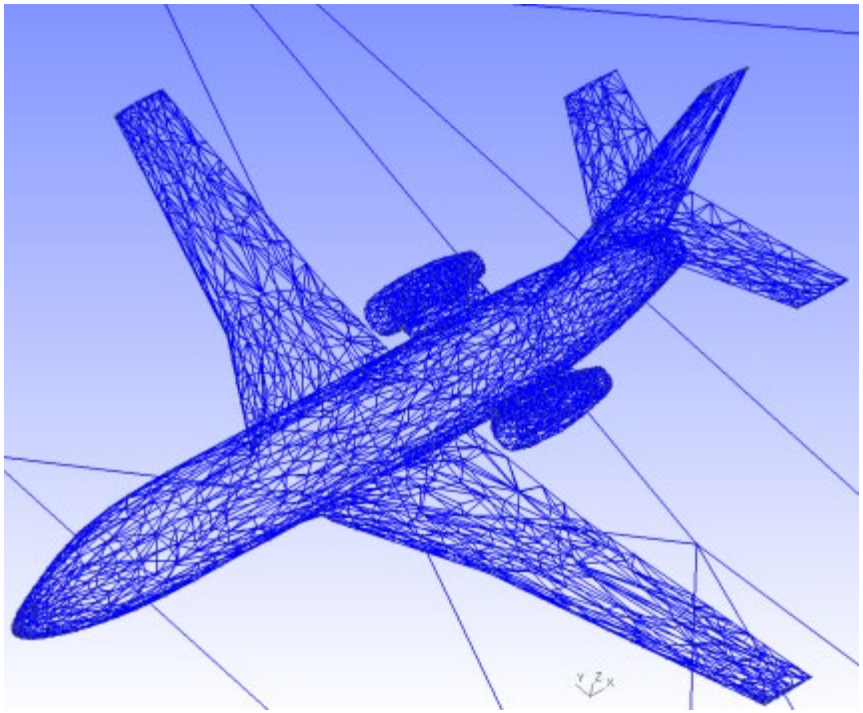
Основные проблемы метода

1) Наиболее трудоёмкий этап – согласование боковых поверхностей призм



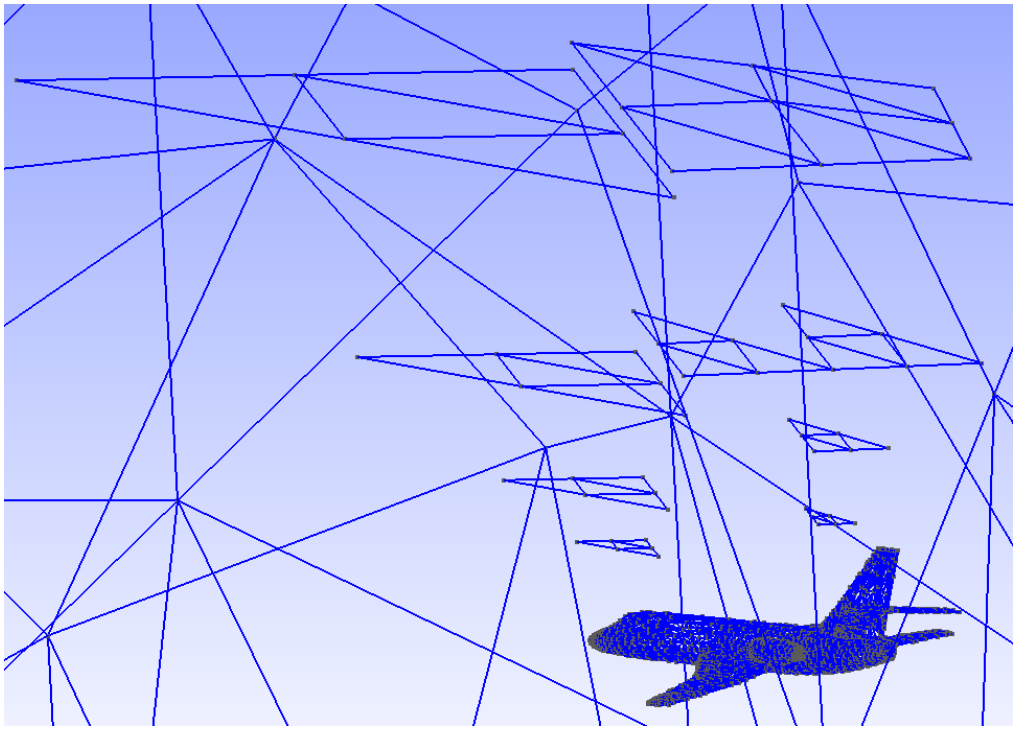
2) Высокая трудоёмкость построения триангуляции при большой разнице размеров пересекающихся треугольников



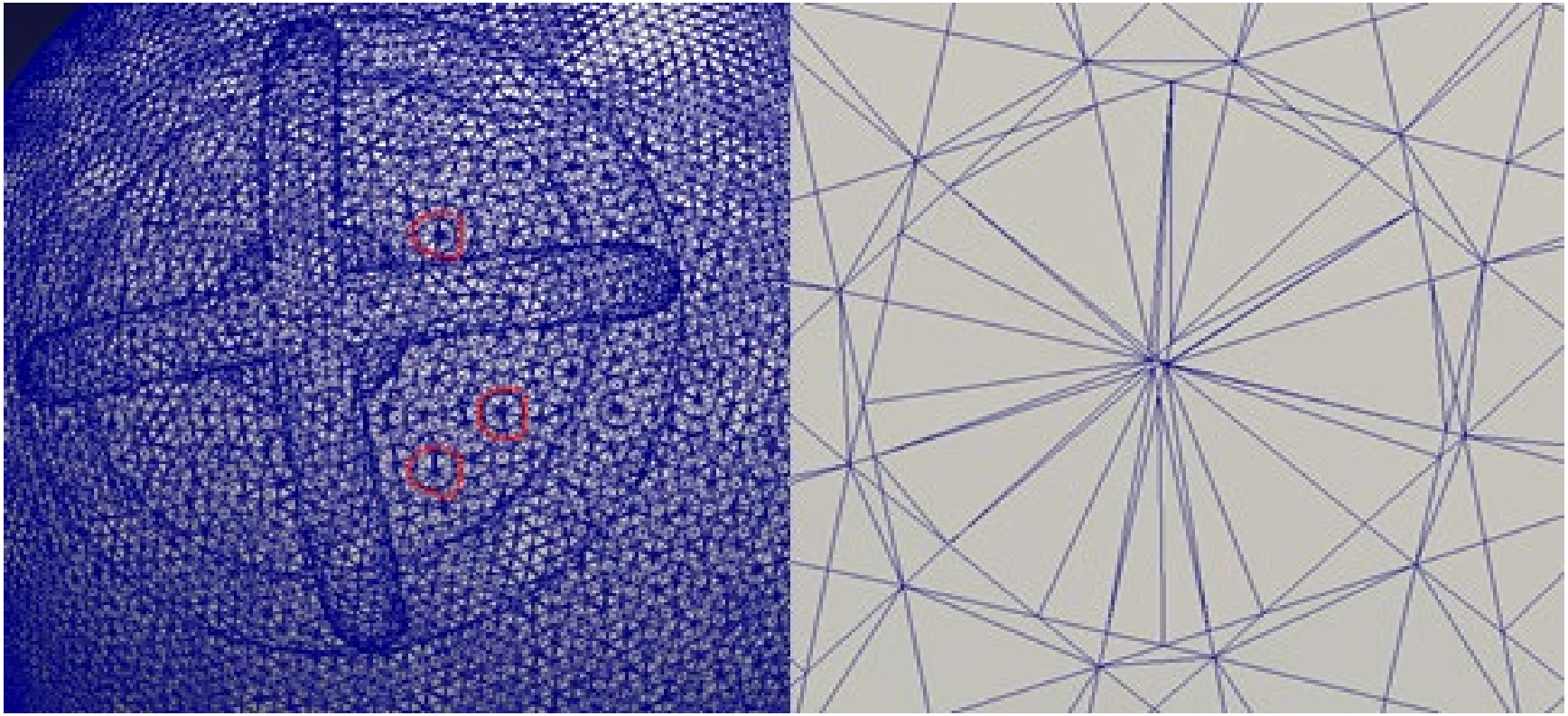


Сокращение трудоёмкости – добавление промежуточных слоёв

Рекурсивное дробление исходного треугольника поверхности и размещение дополнительных фрагментов

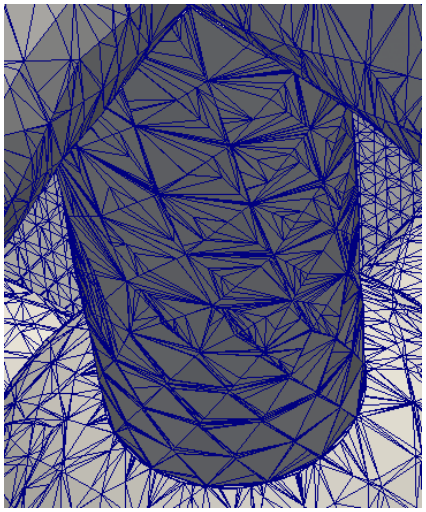


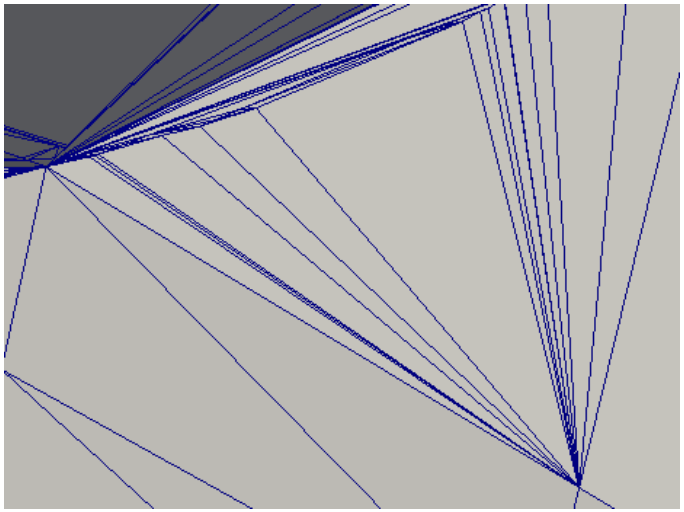
3) образование множества тонких призм

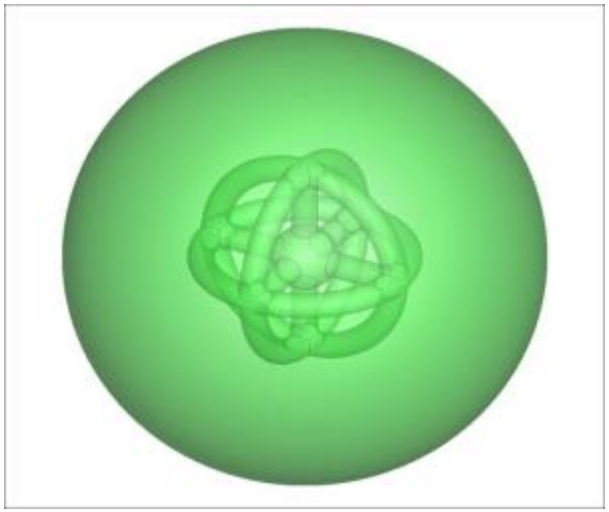


4) проблемное взаимное расположение призм

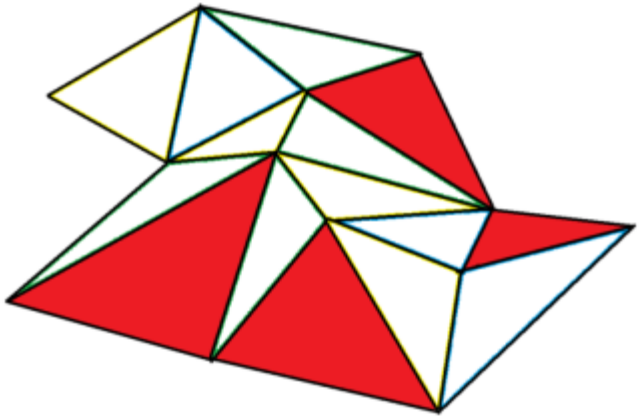
- Большое количество расположенных близко друг к другу призм приводит к квадратичной оценке вычислительной сложности
- Применение кэширования не позволяет уменьшить число рассматриваемых призм

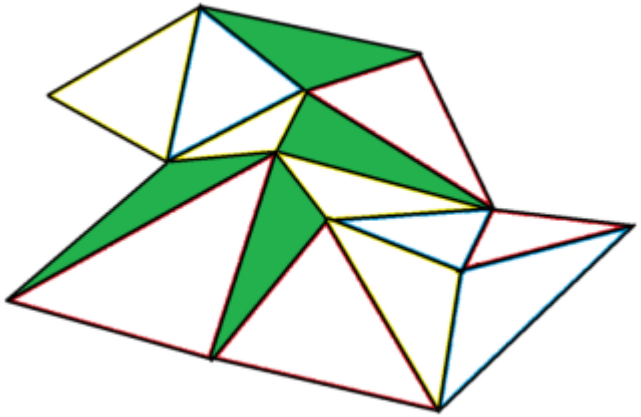


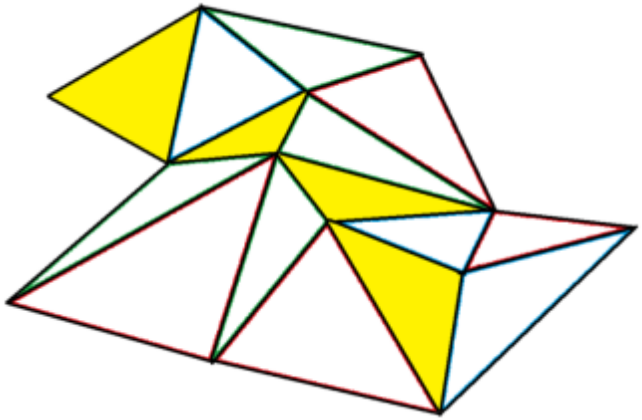


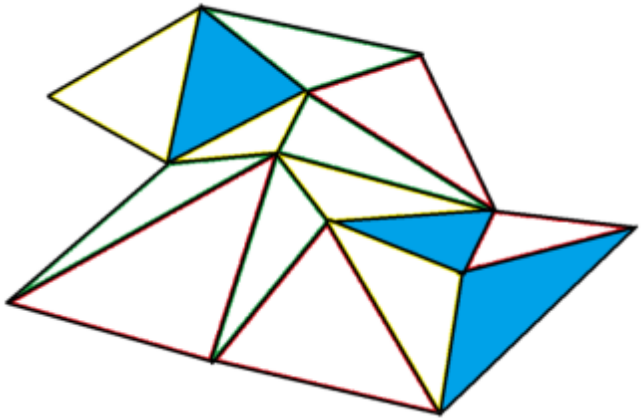


Области хорошего параллелизма (теорема о четырёх красках)

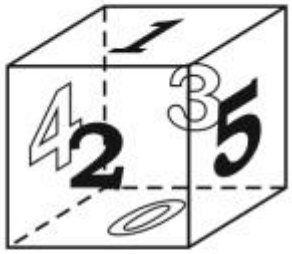


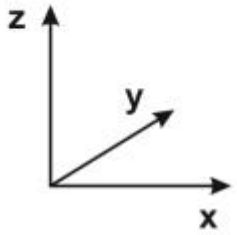


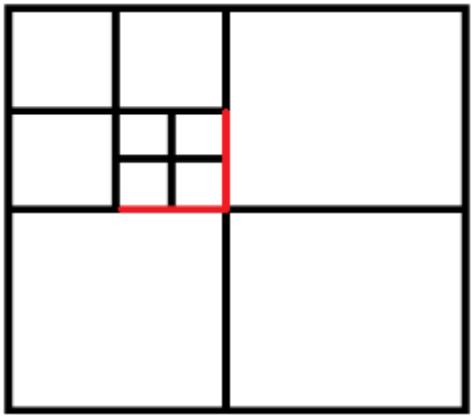


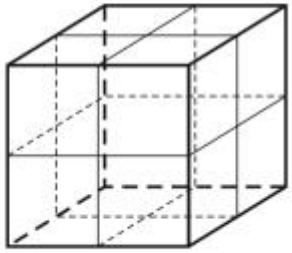


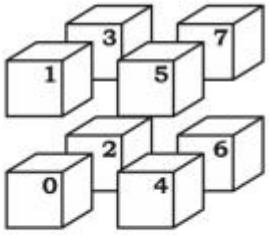
Прямоугольные адаптивные сетки на основе octree-технологии

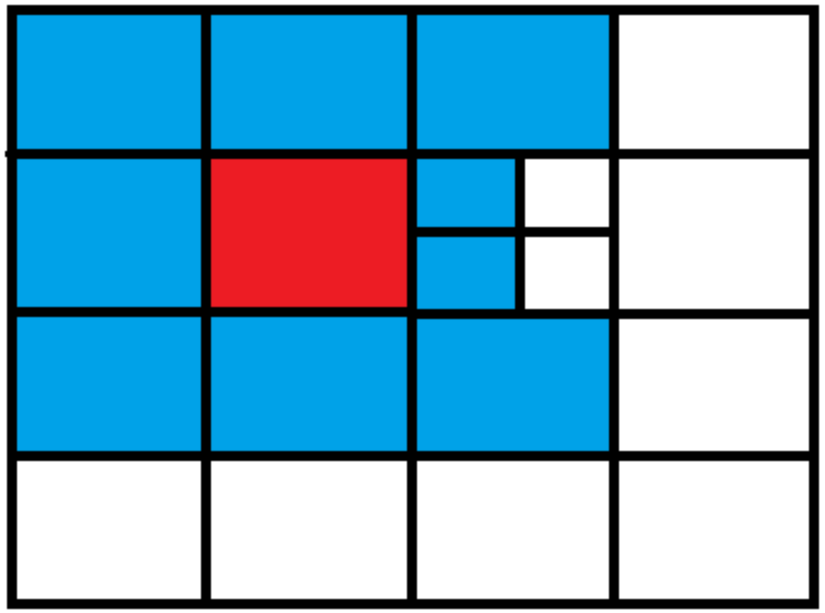












Рассмотрен метод, обеспечивающий выполнение гарантированной триангуляции трёхмерных тел

Заключение

- *Эффективное использование высокопроизводительных вычислительных систем требует создания качественно новых алгоритмов решения прикладных задач и средств описания и создания параллельных программ*
- *Дальнейшее развитие возможно на пути тесного взаимодействия специалистов по вычислительным методам, прикладному и системному программированию*

Поддержка работ

- Работы выполнены при поддержке ряда проектов РФФИ, РНФ и программ фундаментальных исследований РАН

- Расчеты выполнялись на суперкомпьютерных системах
  - К-100 (ИПМ РАН)
  - МВС-100К (МСЦ РАН)
  - Ломоносов (МГУ им. М.В.Ломоносова)
  - Барселона

Контакты

Якобовский М.В.

чл.-корр. РАН, д.ф.-м.н., проф.,

зам. директора по научной работе
Института прикладной математики
им. М.В.Келдыша Российской академии наук

mail: lira@imamod.ru

web: http://lira.imamod.ru

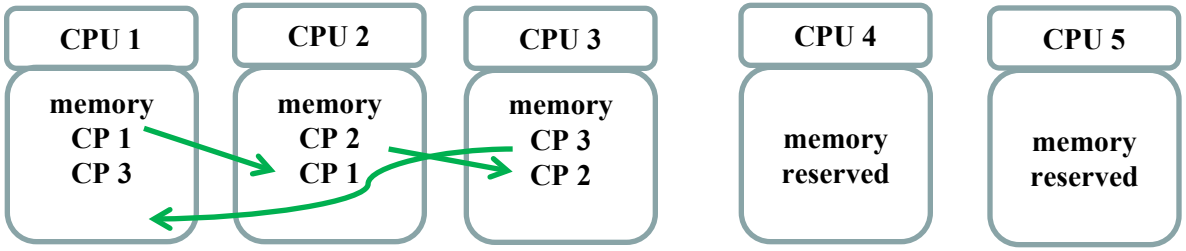
Результат: точки

n_points= 18
n_trg= 65
n_convex_bodies= 13

points			rpoints			
985	174	1000	0	985 / 1	174 / 1	1000 / 1
-342	-940	1000	1	-342 / 1	-940 / 1	1000 / 1
341.794	-60.209	1000	2	47857325 / 140018	-4215169 / 70009	1000 / 1
341.794	-60.209	652.509	3	47857325 / 140018	-4215169 / 70009	45681500 / 70009
-643	766	1000	4	-643 / 1	766 / 1	1000 / 1
-223.169	-265.927	1000	5	-15623814 / 70009	-55851940 / 210027	1000 / 1
-223.169	-265.927	652.539	6	-15623814 / 70009	-55851940 / 210027	137051000 / 210027
-118.783	326.2	1000	7	-8315901 / 70009	22836930 / 70009	1000 / 1
-118.783	326.2	652.606	8	-8315901 / 70009	22836930 / 70009	137065000 / 210027
-223.169	-265.927	347.288	9	-15623814 / 70009	-55851940 / 210027	72940000 / 210027
-118.783	326.2	347.32	10	-8315901 / 70009	22836930 / 70009	24315500 / 70009
341.794	-60.209	347.386	11	47857325 / 140018	-4215169 / 70009	72960500 / 210027
-118.783	326.2	0	12	-8315901 / 70009	22836930 / 70009	0 / 1
-223.169	-265.927	0	13	-15623814 / 70009	-55851940 / 210027	0 / 1
341.794	-60.209	0	14	47857325 / 140018	-4215169 / 70009	0 / 1
-866	-500	0	15	-866 / 1	-500 / 1	0 / 1
0	1000	0	16	0 / 1	1000 / 1	0 / 1
866	-500	0	17	866 / 1	-500 / 1	0 / 1

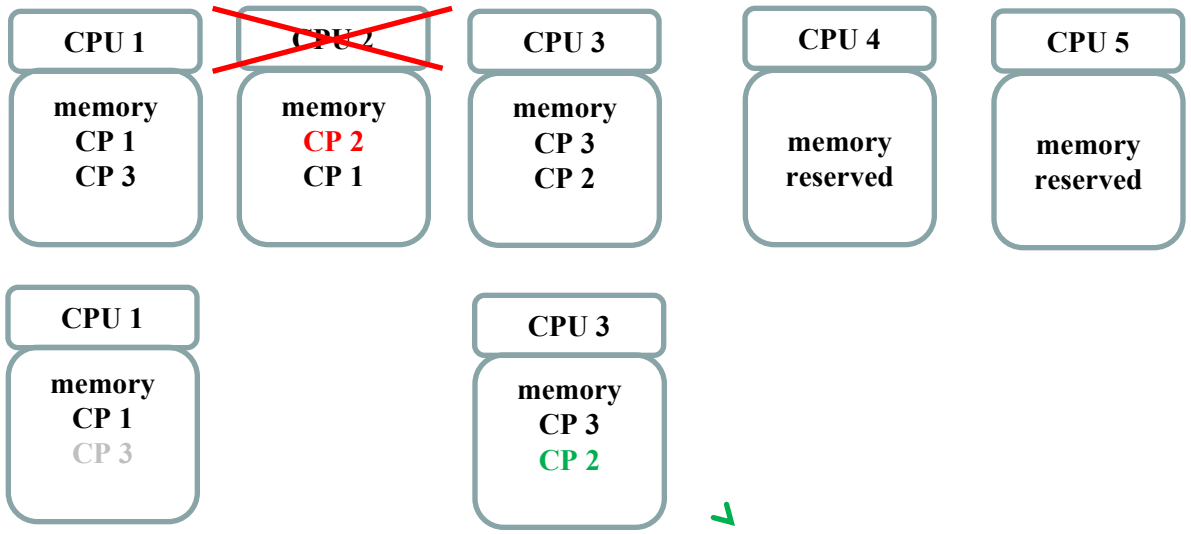
Fault tolerance approach

If we do not want to rollback we must have several copies.
We store them in the local memory of other working processors.



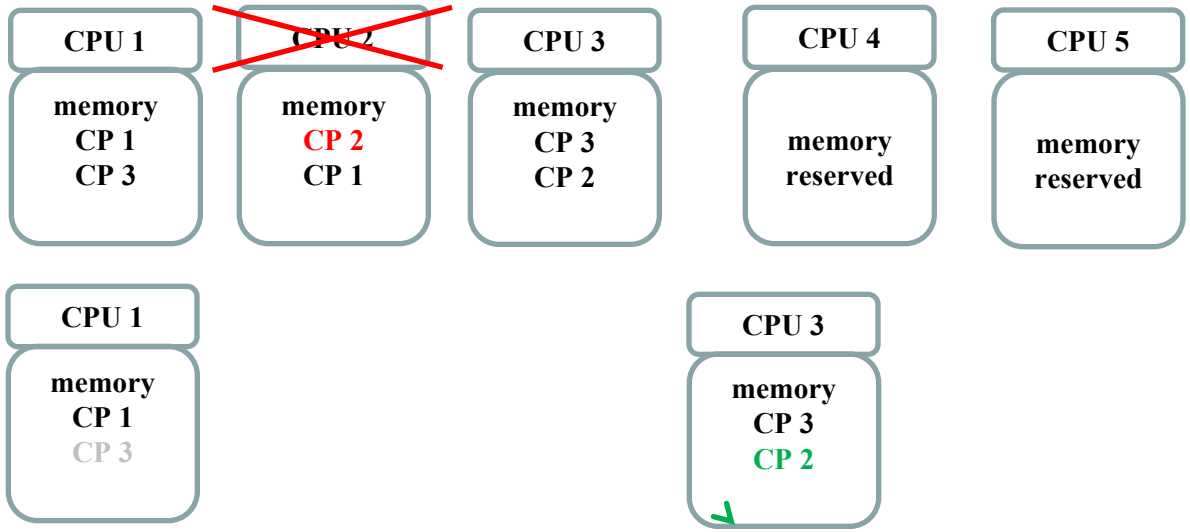
Fault tolerance approach

If we do not want to rollback we must have several copies.
We store them in the local memory of other working processors.



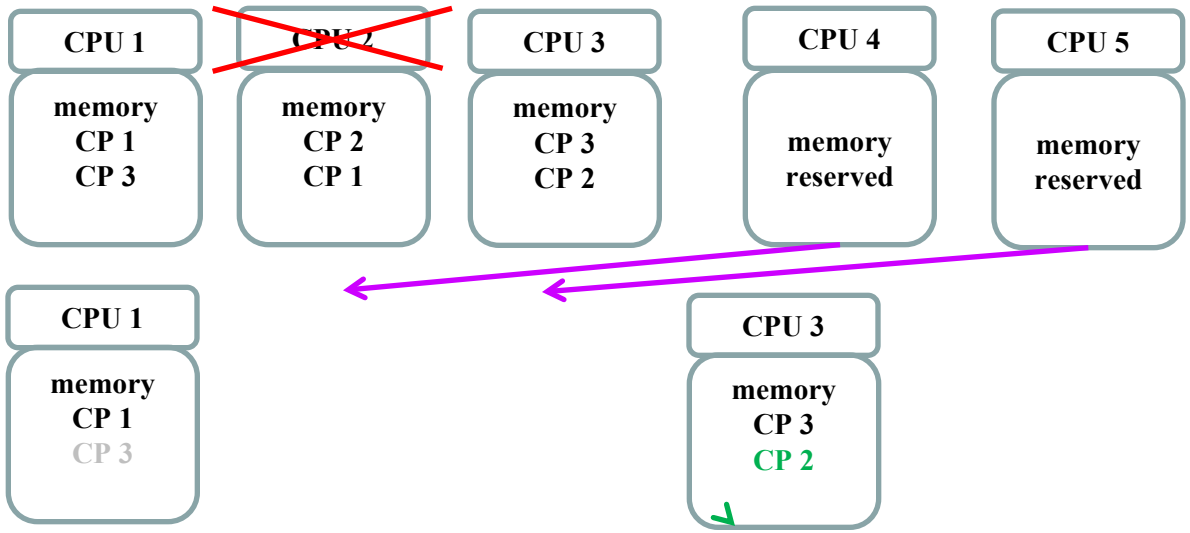
Fault tolerance approach

If we do not want to rollback we must have several copies.
We store them in the local memory of other working processors.



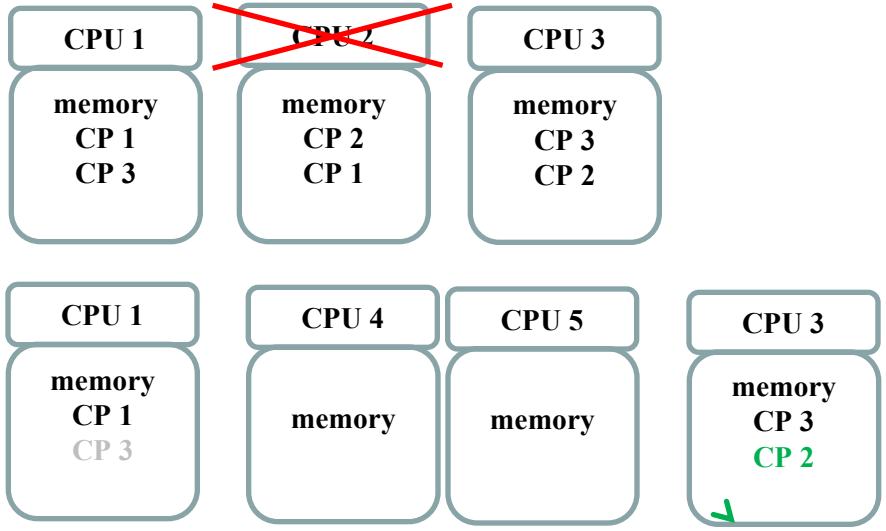
Fault tolerance approach

If we do not want to rollback we must have several copies.
We store them in the local memory of other working processors.



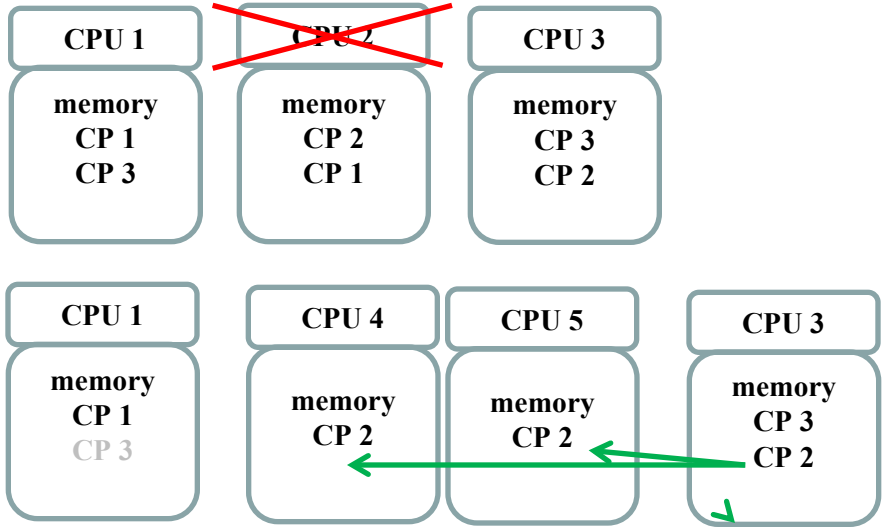
Fault tolerance approach

If we do not want to rollback we must have several copies.
We store them in the local memory of other working processors.



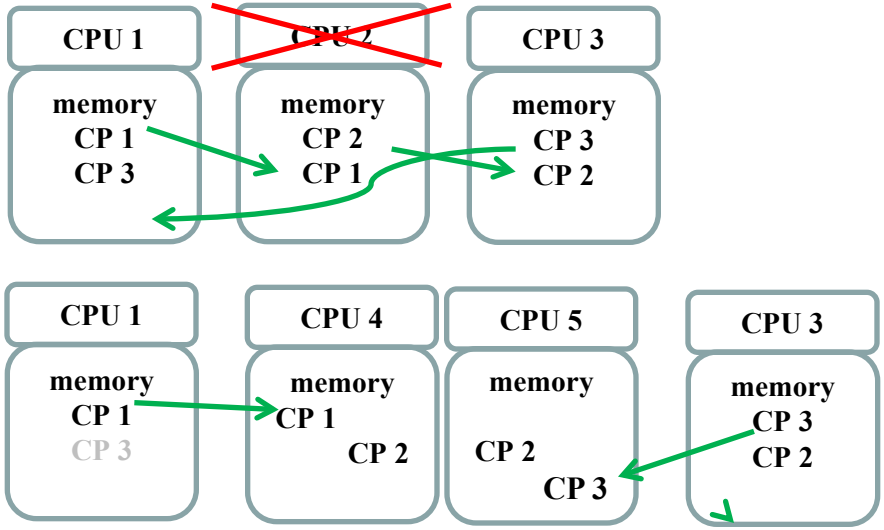
Fault tolerance approach

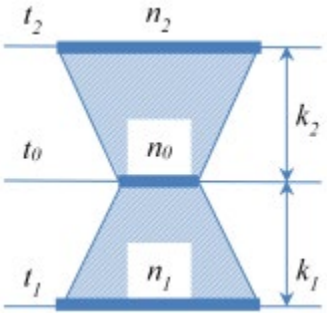
If we do not want to rollback we must have several copies.
We store them in the local memory of other working processors.



Fault tolerance approach

If we do not want to rollback we must have several copies.
We store them in the local memory of other working processors.





Сохранение контрольных точек локально – в память вычислительных узлов

Исключение необходимости перезапуска всех процессоров