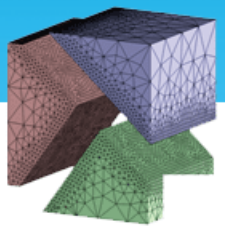
НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ им. Н.И. ЛОБАЧЕВСКОГО
Национальный исследовательский университет

Факультет вычислительной математики и кибернетики
Кафедра информатики и автоматизации научных исследований



Николай Старостин, к.т.н., доцент
Маргарита Панкратова, аспирант

Архитектурно-зависимая декомпозиция в методиках суперкомпьютерного моделирования

Научный семинар «Методы суперкомпьютерного моделирования», Таруса, 2014

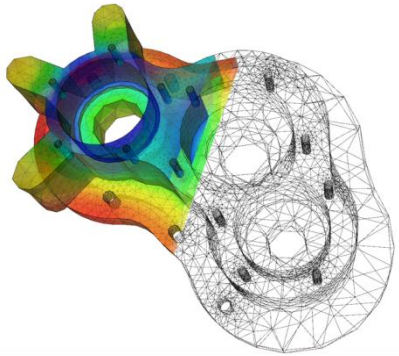
Предметная область

Инструменты инженерного анализа на супер-ЭВМ:
ЛОГОС, ЛЭГАК-ДК, ДАНКО+ГЕПАРД, НИМФА, АРМ WinMachine 2010, Autodesk Simulation CFD, Autodesk Simulation Mechanical, Autodesk Simulation MoldFlow

Развитие пакетов

Многие методики численного моделирования физических процессов основываются на аппроксимационных сетках, которые моделируются с помощью графов. На практике сеточные структуры имеют большие порядки (10^6 - 10^9 узлов), что приводит к необходимости их распределения по узлам параллельной вычислительной сети.

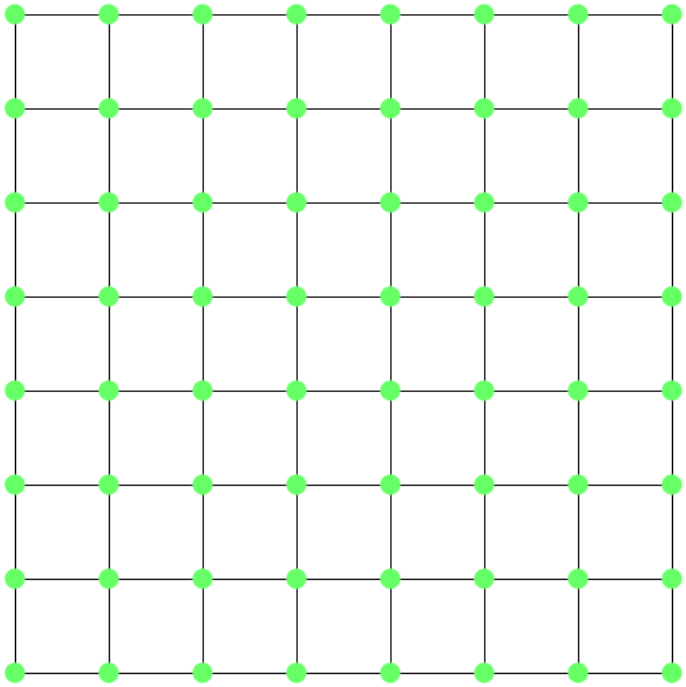
Эффективность параллельных вычислений
в значительной мере зависит от объёма межпроцессорных коммуникаций и сбалансированности загрузки процессоров.



Моделируемые процессы:
газодинамика, аэродинамика, гидродинамика, турбулентное перемешивание, прочность, разрушение, тепломассоперенос, многокомпонентная многофазная фильтрация и др.

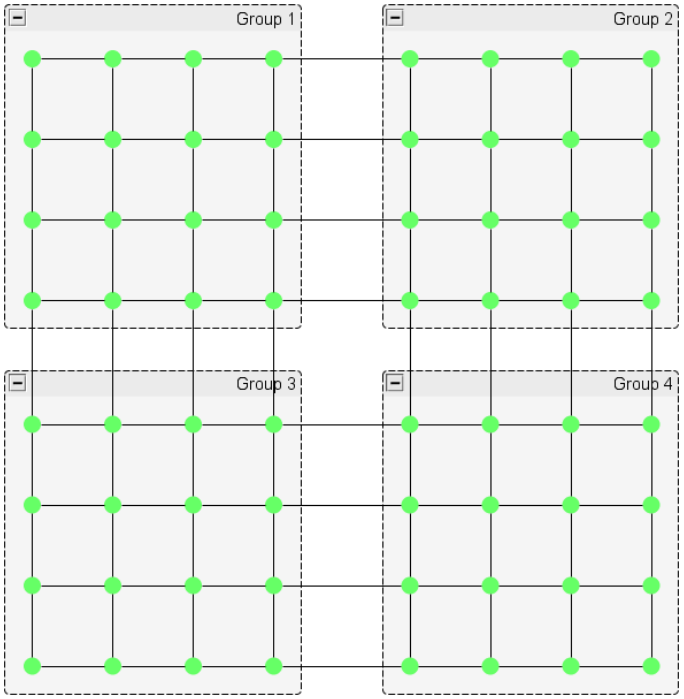
Общая постановка задачи

Расчетная сетка



Распределить

Вычислительная сеть, физическая топология



Цель: минимизация издержек на коммуникации.

? Как изначально разбить сетку и распределить ее по узлам вычислительной сети, чтобы минимизировать издержки на межпроцессорные коммуникации.

Общая постановка задачи
Исходные данные

$G(V, E, w, u)$ - расчетная сетка

$V = \{v_1, \dots, v_n\}$ – множество узлов сетки

$E \subseteq V^{(2)}$ – связи между узлами

Веса вершин $w(v_i) \in N, i = \overline{1, n}$ - затраты на расчет

Веса ребер $u(v_i, v_j) \in N, i, j = \overline{1, n}$ – интенсивность коммуникаций

Матрица $T = (t_{ij})_{k \times k}$ - вычислительная сеть

$t_{ij} \in N$ - оценка затрат на передачу данных между парой вычислителей

Общая постановка задачи
Варьируемые параметры

Решение задачи - вектор $x = \{x_1, \dots, x_n\}, x_i \in \{1, \dots, \mathrm{k}\}, i = \overline{1, n}$

i-ый элемент соответствует узлу расчетной сетки

Значение x_i - узел вычислительной сети, на который он распределен

Общая постановка задачи
Ограничения

$W = \sum_{i=1}^{n} w(v_i)$ – суммарный вес узлов сетки

$W_j = \sum_{\{i:x_i=j, i=\overline{1,n}\}} w(v_i), \quad j = \overline{1,k}$ – загрузка (оценка относительного времени работы) вычислительного узла j

$\widetilde{W} = \frac{W}{k}$ – идеальная загрузка вычислительных узлов

$\varepsilon > 0$ – коэффициент отклонения от идеальной загрузки

$$\left|\frac{W_j}{\widetilde{W}} - 1\right| \leq \varepsilon, j = \overline{1,k} \qquad (1)$$

- ограничения по балансировке нагрузки вычислительных узлов

Общая постановка задачи
Критерии

Проблема: очень сложно, а зачастую практически невозможно спрогнозировать: точное время каждой коммуникации; возникновение коллизий на уровне сетевых устройств; снижение пропускной способности каналов в связи с повышением интенсивности обменов; маршрутов, по которым происходит передача пакетов.

Вывод: нет возможности применения модели в терминах планирования коммуникационных обменов. С практической точки зрения удобнее оперировать оценками затрат на всю совокупность коммуникационных обменов.

$\beta(x, v_i, v_j) = u(v_i, v_j) \cdot t_{x_i x_j}$ - оценка затрат на коммуникационный обмен между парой узлов вычислительной сети

$$F_1(x) = \max_{(v_i, v_j) \in E} \beta(x, v_i, v_j) \to min \quad (2)$$

$$F_2(x) = \sum_{(v_i, v_j) \in E} \beta(x, v_i, v_j) \to min \quad (3)$$

Проблемы организации исполнения параллельной программы

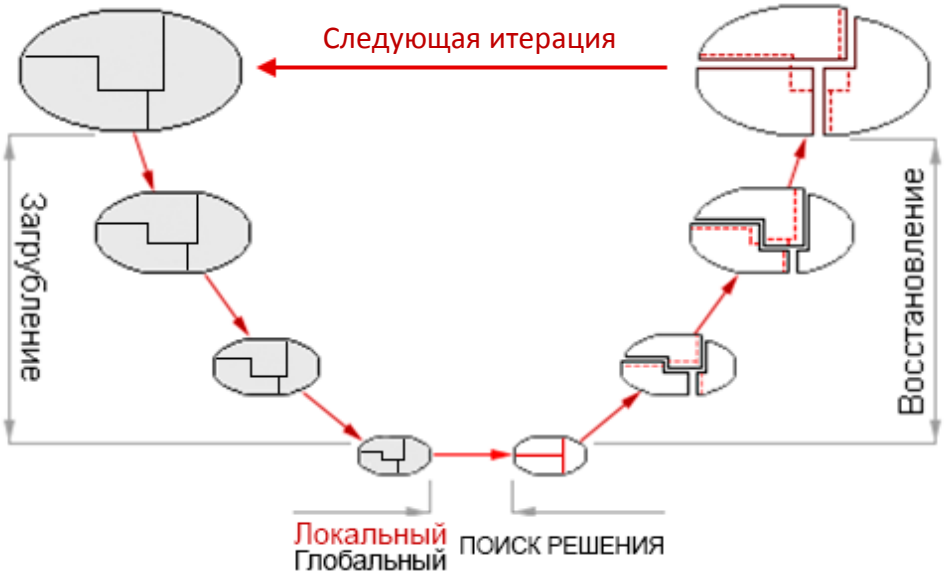
- Как правило, при организации исполнения параллельной программы заранее известна только структура расчетной сетки и общее число вычислительных узлов, которое требуется для расчета
- Структура выделенного участка вычислительной сети определяется только в момент начала расчета, когда планировщик выделяет очередной задаче требуемое число вычислителей
- Вывод: поэтому предварительная декомпозиция сеточной структуры (она необходима в силу слишком большого объема данных) осуществляется без учета информации о коммуникационной топологии
- Задача (1),(2) вырождается в задачу минимизации ширины ленты матрицы
- Задача (1),(3) – в задачу k-разбиения графа

Пакеты, которые решают обозначенный класс задач:

METIS, JOSTLE, SCOTCH, CHACO, **MATRUZ**

MATRUZ – инструментарий для суперЭВМ, разработанных ННГУ совместно с РФЯЦ ВНИИЭФ.

Многоуровневый итерационный метод



Идея использовать полученное решение на следующей итерации.

В традиционной схеме некоторое разбиение возникает только как результат работы фазы поиска. Возникают ВОПРОСЫ:

1. **Как изменится цель и принципы редукции?**
2. Какие алгоритмы применимы на фазе поиска решений?
   Локальный поиск исследует окрестность текущего решения.
   Алгоритмы глобального поиска пытаются найти новое решение.

Фаза редукции

1. Эксплуатирует информацию о структуре найденного решения.
2. Реализует однозначную проекцию решения на грубый граф.
3. Обеспечивает принцип сравнения проекций на всех уровнях.

Фаза поиска

1. Разные стратегии поиска решения.
2. Локальный поиск пытается улучшить проекцию решения.
3. Глобальный поиск пытается найти новое решение.

Фаза восстановления

1. Проекция рекорда на исходный граф.
2. Пытается улучшить рекорд на разных уровнях редукции

Тесты на бенчмарках

http://staffweb.cms.gre.ac.uk/~wc06/partition

graph	|V|	|E|	k=2			k=4			k=8			k=32			k=32			k=64		
			record	METIS	MATRUZ	record	METIS	MATRUZ	record	METIS	MATRUZ	record	METIS	MATRUZ	record	METIS	MATRUZ	record	METIS	MATRUZ
add20	2395	7462	536 [IPMNE2]	744	700	1134 [HYPAL]	1272	1239	1657 [HYPAL]	1863	1840	2028 [*+IPMNE2]	2396	2309	2342 [KaBaPET]	2897	2704	2923 [KaBaPET]	3444	3602*
data	2851	15093	181 [SDP]	200	195	363 [KaFFPa]	433	397	628 [NW]	787	686	1076 [FSMAGP]	1289	1171	1743 [FSMAGP]	2032	1894	2747 [KaBaPET]	3461	2938
3elt	4720	13722	87 [JE]	87	87	197 [NW]	214	203	329 [KaFFPaE]	394	346	557 [KaFFPaE]	597	595	930 [FSMAGP]	1021	1019	1498 [KaBaPET]	1658	1642
uk	4824	6837	18 [JE]	27	20	39 [*+KFFP]	57	42	75 [KaFFPaE]	124	87	137 [*+IPMNE2]	185	157	236 [KaBaPE]	303	269	394 [KaBaPE]	498	446
add32	4960	9462	10 [J2.2]	11	10	33 [JE]	51	33	66 [FF]	100	69	117 [JE]	197	117	212 [JE]	292	212	476 [*+IPMNE2]	610	556
bcsstk33	8738	291583	9914 [U]	10453	9914	20158 [FSMAGP]	23140	21679	33908 [FSMAGP]	35048	37514	54119 [FSMAGP]	59422	58911	76079 [*+IPMNE2]	83710	83250	105338 [*+IPMNE2]	114366	115538*
whitaker3	9800	28989	126 [JE]	135	126	378 [NW]	402	378	644 [*+IPMNE2]	693	679	1068 [KaBaPET]	1218	1177	1632 [KaBaPET]	1798	1781	2438 [KaBaPET]	2703	2687
crack	10240	30380	182 [NW]	211	182	360 [NW]	450	360	666 [KaFFPaE]	735	717	1063 [FSMAGP]	1246	1123	1655 [*+KFFP]	1846	1780	2490 [KaBaPET]	2720	2821
wing_nodal	10937	75488	1668 [SDP]	1852	1689	3534 [FSMAGP]	3820	3667	5339 [FSMAGP]	5772	5906	8160 [FSMAGP]	8848	8678	11536 [FSMAGP]	12558	12682	15515 [FSMAGP]	16977	16954
fe_4elt2	11143	32818	130 [MRSB]	132	130	335 [FSMAGP]	385	350	578 [KaFFPaE]	673	620	979 [FSMAGP]	1117	1057	1571 [*+IPMNE2]	1742	1695	2415 [KaBaPET]	2693	2653
vibrobox	12328	165250	10310 [JE]	11269	10310	18943 [FSMAGP]	21642	20346	23924 [KaFFPaE]	29021	31470	31232 [*+IPMNE2]	36418	41622	38848 [*+IPMNE2]	44588	46038	46094 [*+IPMNE2]	51874	51029
bcsstk29	13992	302748	2818 [GrPart]	3057	2818	7925 [KaFFPaE]	9317	8328	13540 [KaFFPaE]	18769	13736	20924 [NW]	30563	23691	33451 [KaBaPET]	41319	37153	53981 [*+IPMNE2]	62837	60330*
4elt	15606	45878	137 [NW]	148	137	319 [NW]	381	331	515 [FSMAGP]	600	547	887 [KaBaPE]	1084	1008	1493 [KaBaPET]	1757	1642	2485 [KaBaPET]	2772	2690
fe_sphere	16386	49152	384 [JE]	430	384	764 [FSMAGP]	820	806	1152 [JE]	1289	1230	1678 [FSMAGP]	1871	1954	2427 [FSMAGP]	2825	2761	3456 [FSMAGP]	3969	4027
cti	16840	48232	318 [JE]	357	318	889 [FSMAGP]	1213	976	1684 [*+KFFP]	2020	1813	2701 [KaBaPET]	3294	2948	3911 [KaBaPET]	4791	4308	5473 [KaBaPET]	6457	6471
memplus	17758	54196	5263 [HYPAL]	6367	6566	9292 [KaFFPaE]	10239	10305	11543 [*+KFFP]	12446	12531	12799 [*+IPMNE2]	15674	14583	13877 [*+IPMNE2]	16623	16171	15877 [*+IPMNE2]	18213	18081
cs4	22499	43858	353 [KaFFPa]	425	391	917 [*+IPMNE2]	1072	1019	1420 [*+IPMNE2]	1743	1624	2043 [KaBaPET]	2465	2331	2859 [*+IPMNE2]	3390	3237	3962 [KaBaPE]	4628	4377
bcsstk30	28924	1007284	6251 [JE]	7054	6251	16372 [KaBaPE]	17671	16552	34068 [FSMAGP]	35793	34541	68787 [*+IPMNE2]	81063	72924	109548 [*+IPMNE2]	125964	122824	167049 [*+IPMNE2]	186904	191159*
bcsstk31	35588	572914	2662 [HYPAL]	2983	2740	7082 [KaBaPE]	8372	7368	12843 [*+IPMNE2]	14685	13458	22859 [*+IPMNE2]	28824	24651	36383 [KaBaPET]	44237	39610	55396 [*+IPMNE2]	64750	67656
fe_pwt	36519	144794	340 [KasPar]	374	340	700 [KaFFPaE]	841	704	1405 [FSMAGP]	1634	1435	2737 [FSMAGP]	2945	2807	5306 [KaBaPE]	6063	5575	7959 [FSMAGP]	8792	8653
bcsstk32	44609	985046	4667 [JE]	5330	4953	9181 [KaBaPET]	13826	11176	18955 [*+IPMNE2]	25107	21254	34374 [KaBaPE]	47024	36522	58352 [*+IPMNE2]	69063	62047	88598 [*+IPMNE2]	108568	97405
fe_body	45087	163734	262 [MQI]	289	262	598 [*+IPMNE2]	674	643	1012 [*+IPMNE2]	1183	1093	1683 [KaBaPE]	1975	2057	2678 [*+IPMNE2]	3381	3135	4501 [*+IPMNE2]	6004	5096
t60k	60005	89440	65 [SDP]	84	82	203 [*+KFFP]	272	219	441 [*+IPMNE2]	547	475	787 [KaBaPE]	937	905	1291 [KaBaPE]	1579	1501	2015 [*+IPMNE2]	2445	2323
wing	62032	121544	773 [*+KFFP]	898	850	1593 [*+IPMNE2]	1990	1798	2440 [*+IPMNE2]	2952	2794	3775 [*+IPMNE2]	4535	4359	5513 [*+IPMNE2]	6539	6301	7534 [*+IPMNE2]	9033	8578
brack2	62631	366559	660 [SDP]	767	706	2731 [KaFFPa]	3126	2826	6592 [KaFFPaE	7762	7062	11055 [KaBaPET]	13399	11912	16765 [KaBaPET]	19951	17996	25108 [FSMAGP]	29096	27002
finan512	74752	261120	162 [Ch2.0]	162	162	324 [Ch2.0]	324	324	648 [Ch2.0]	648	648	1296 [Ch2.0]	1296	1296	2592 [Ch2.0]	2592	2592	10560 [NW]	11214	10560
fe_tooth	78136	452591	3773 [SDP]	4339	4044	6765 [KaBaPE]	7976	7915	11151 [*+IPMNE2]	13292	12220	16975 [*+IPMNE2]	20500	18483	24274 [*+IPMNE2]	28632	26518	33403 [*+IPMNE2]	39133	36685
fe_rotor	99617	662431	1940 [KaFFPa]	2206	2025	7050 [KaBaPE]	8104	7606	12309 [*+IPMNE2]	14587	13386	19680 [KaFFPaE]	22860	21608	30356 [KaFFPaE]	34860	33874	44381 [*+IPMNE2]	50818	48778
598a	110971	741934	2336 [MQI]	2505	2424	7816 [*+IPMNE2]	8447	8185	15413 [*+IPMNE2]	17282	16630	25198 [*+IPMNE2]	28328	28516	37644 [*+IPMNE2]	43549	42138	54684 [FSMAGP]	61961	60593
fe_ocean	143437	409593	311 [GrPart]	529	336	1693 [FSMAGP]	2099	1785	3886 [KaBaPE]	4927	4117	7338 [FSMAGP]	9347	8177	12034 [*+IPMNE2]	15818	13395	19394 [*+IPMNE2]	23834	20463
144	144649	1074393	6345 [*+IPMNE2]	6793	6648	15140 [*+IPMNE2]	17179	16779	24179 [*+IPMNE2]	27621	27193	36608 [*+IPMNE2]	41880	40057	54168 [*+IPMNE2]	61070	60702	75777 [*+IPMNE2]	86198	84564
wave	156317	1059331	8524 [KaFFPaE]	9044	8883	16638 [*+IPMNE2]	19596	18503	28489 [*+IPMNE2]	33586	31116	42025 [*+IPMNE2]	47321	46821	59611 [*+IPMNE2]	67068	65972	82006 [*+IPMNE2]	93122	90867
m14b	214765	1679018	3802 [MQI]	4003	3895	12948 [*+IPMNE2]	14023	13317	25126 [*+IPMNE2]	27869	26926	41098 [*+IPMNE2]	45718	46868	63411 [*+IPMNE2]	72841	70544	94140 [*+IPMNE2]	109045	104437
auto	448695	3314611	9450 [MQI]	10648	10290	25789 [KaFFPaE]	28783	27827	44732 [*+IPMNE2]	48953	48399	74272 [*+IPMNE2]	84400	80607	119004 [*+IPMNE2]	130074	128003	169290 [*+IPMNE2]	187554	183198

Обновление осень 2013 года

Chris Walshaw

The Graph Partitioning Archive

University of Greenwich

Продукт JOSTLE

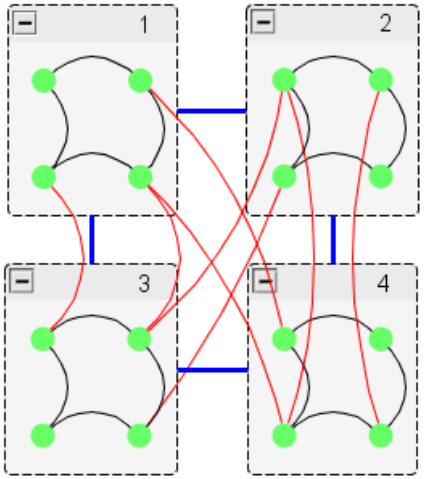
В среднем вес сечения разбиений программы MATRUZ на 6.7% меньше сечений METIS.

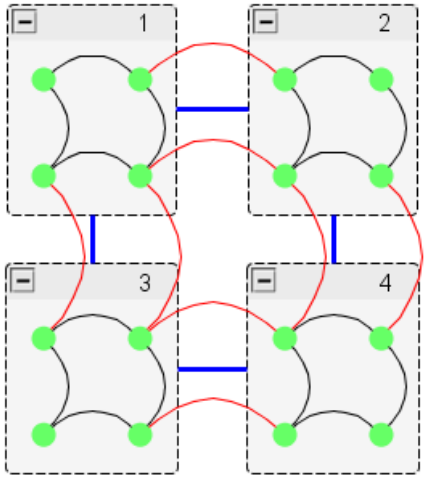
MATRUZ удалось воспроизвести 24 рекорда.

Процесс организации параллельного расчета

1. Подготовка данных
2. Декомпозиция расчетной сетки
3. Запуск параллельной программы численного расчета
   1. **Перенумерация вычислительных узлов с учетом внешних связей декомпозиции**
   2. Загрузка декомпозированной расчетной сетки
   3. Процесс численного расчета

В итоге даже для качественной декомпозиции в процессе работы параллельного расчета в выделенном сегменте физической топологии могут наблюдаться значительные потери при межпроцессорных обменах. **Подобные негативные эффекты можно минимизировать за счет удачного распределения расчетной сетки по узлам вычислительной сети.**





Постановка задач перенумерации вычислительных узлов

Расчетная сетка:

$G(V, E, w, u), V = \{v_1, \dots, v_k\}, E \subseteq V^{(2)}$

$w(v_i) \in N, i = \overline{1,k}$ - затраты на расчет, $u(v_i, v_j) \in N, i, j = \overline{1,k}$ интенсивность обменов

Вычислительная сеть:

$T = (t_{ij})_{k \times k}$, $t_{ij} \in N$ - затраты на передачу данных между парой вычислителей

Решение:

Перестановка $x = \{x_1, \dots, x_k\}$

Критерии:

$\beta(x, v_i, v_j) = u(v_i, v_j) \cdot t_{x_i x_j}$ - оценка затрат на коммуникационный обмен между парой узлов вычислительной сети

$$F_1(x) = \max_{(v_i, v_j) \in E} \beta(x, v_i, v_j) \to min \quad (4)$$

$$F_2(x) = \sum_{(v_i, v_j) \in E} \beta(x, v_i, v_j) \to min \quad (5)$$

Пути решения задачи

1
- **Известна топология ВС**

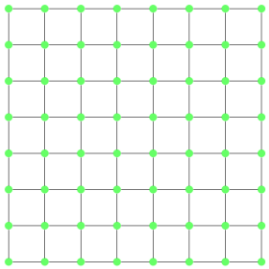
1.1
- Решение задачи архитектурно-зависимой декомпозиции методом рекурсивной бисекции

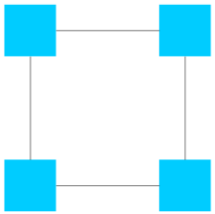
2
- **Неизвестна топология ВС**

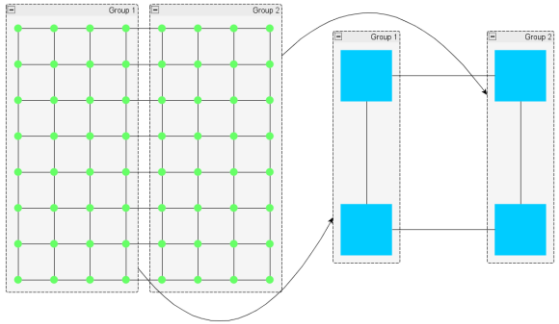
2.1
- Решение задачи декомпозиции расчетной сетки многоуровневым алгоритмом

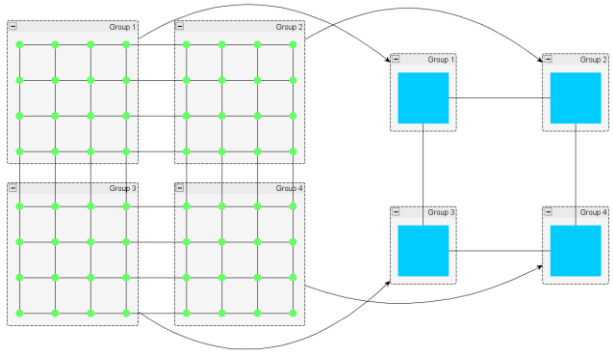
2.2
- Решение задачи перенумерации вычислительных узлов методом рекурсивной бисекции

Концепция рекурсивной бисекции



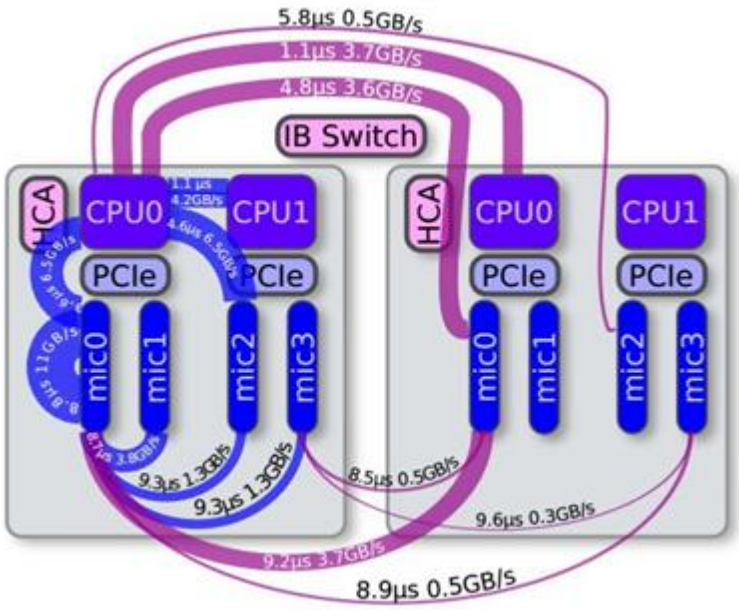




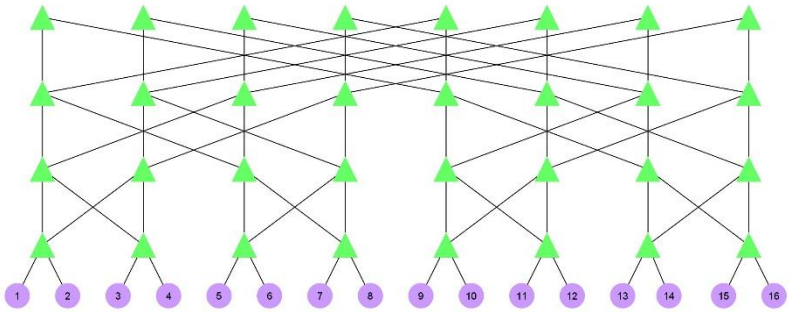


Топология тестовых вычислительных сетей

Symmetric



Fat tree



Вычислительный эксперимент

http://staffweb.cms.gre.ac.uk/~wc06/partition http://www.cise.ufl.edu/research/sparse/matrices/

Расчетная сетка	Размер	BC	Размер	1-этапный		2-этапный	
				F_1	F_2	F_1	F_2
gr_30_30	900	Fattree	64	8	6734	8	6467
gr_30_30	900	Symmetric	288	430	619860	430	479564
ef_body	45087	Fat tree	64	8	29986	8	32671
ef_body	45087	Symmetric	288	430	3248324	430	3224784
fe_tooth	78136	Fat tree	64	8	218430	8	210407
fe_tooth	78136	Symmetric	288	430	18033219	430	16097627
fe_rotor	99617	Fat tree	64	8	290339	8	261618
fe_rotor	99617	Symmetric	288	430	24332380	430	22415356

Вывод: 2-этапный алгоритм показывает лучшие результаты.

1. Рекурсивная бисекция – эвристический алгоритм
2. Качественная предварительная декомпозиция расчетной сетки минимизирует внешние связи, меньше связей проявляется в виде межпроцессорных коммуникаций