

УДК 004.94

## ИМИТАЦИОННАЯ МОДЕЛЬ КОМПЬЮТЕРНОГО КЛАСТЕРА ДЛЯ ОПТИМИЗАЦИИ ПРОЦЕССА РАСПРЕДЕЛЕННО-ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

О.М. Замятина, В.В. Соколова, Е.С. Малахова, Е.В. Ушакова

Томский политехнический университет  
E-mail: zamyatina@tpu.ru

Проведены исследования параметров суперкомпьютерного кластера, на основе которых разработана его имитационная модель. Модель позволила оптимизировать загрузку кластера путем ускорения обработки поступающих заданий с уменьшением времени их нахождения в очереди на обслуживание при максимальной загрузке компьютеров кластера.

### Ключевые слова:

Имитационная модель, компьютерный кластер, сервер, хранилище данных.

### Key words:

Simulation model, computer cluster, server, data warehouse.

### Введение

Большие объемы накопленной информации требуют огромных мощностей вычислительных ресурсов не только для ее хранения, но и для обработки хранимых данных. Одним из возможных вариантов решения данной задачи является использование кластеров для распределенно-параллельных вычислений.

Под кластером будем понимать группу компьютеров, объединенных высокоскоростными каналами связи и представляющую, с точки зрения конечного пользователя, единый аппаратный ресурс [1, 2].

Вычислительные суперкомпьютерные кластеры (СКК) высоко востребованы для разных отраслей, таких как машиностроение, судостроение, энергетическая отрасль, наука и др.

Использование суперкомпьютерного кластера позволяет [3]:

- моделировать аэрогазодинамические процессы, химические реакции, физические процессы;
- проверять статическую и динамическую прочность элементов и механических систем;
- испытывать гидравлику, термодинамику, теплопроводность;
- решать сложные математические уравнения;
- проводить расчеты геологических пластов;
- и решать многие другие задачи.

Результатом использования СКК является снижение количества требуемых натурных и дорогостоящих испытаний, предсказание поведения системы во внештатных условиях, что позволяет существенно снизить затраты на производство тестовых образцов [1, 3].

В свою очередь, часто возникает сложная задача моделирования работы самого компьютерного кластера для распределенно-параллельных вычислений. Решение данной задачи позволит следующее:

- упростить и оптимизировать процесс распределения поступающих данных между членами кластера – компьютерами;

- выделить параметры кластера, наиболее важные для успешного функционирования, и проанализировать имеющиеся между ними связи;
- рассмотреть процессы в различных масштабах времени;
- разработать процессы, позволяющие бороться с тупиками и неопределенностью, вызванными случайностью и непостоянством систем;
- обнаружить скрытые резервы и ресурсы моделируемой системы и оптимально их использовать.

Цель данной работы – смоделировать процесс функционирования компьютерного кластера для оптимального распределения заданий между компьютерами, входящими в его состав. Для достижения поставленной цели необходимо решить следующие задачи:

1. Определить основные параметры СКК, подлежащие моделированию.
  2. Построить модель СКК.
  3. Определить оптимальный порядок обработки поступающих заданий.
- Рассмотрим каждую из задач подробнее.

### Основные параметры суперкомпьютерного кластера

Суперкомпьютерный кластер можно охарактеризовать следующими параметрами:

- общее количество обработанных заданий;
- время их обработки;
- количество заданий, обработанных каждым компьютером;
- количество обработанных заданий в единицу времени в целом и на каждом компьютере кластера;
- время ожидания задания в очереди на обработку;
- среднее, минимальное и максимальное число заданий в очереди;
- среднее число занятых компьютеров и т. п.

В рамках поставленной цели наибольший интерес представляют такие параметры, как время обработки заданий, загрузка компьютеров и среднее число заданий в очереди.

### Модель компьютерного кластера для распределенно-параллельных вычислений

В качестве среды для разработки модели был выбран программный продукт дискретно-событийного моделирования Arena (разработчик Rockwell Software, США), позволяющий создавать имитационные модели разнородных процессов и систем в терминах предметной области для дальнейшей работы с ними. В Arena предусмотрена возможность сбора статистики по результатам моделирования, что облегчает дальнейший анализ полученных моделей. Изначально, продукт Arena был разработан для создания моделей бизнес-процессов [4, 5]. В рамках данного исследования это средство дискретно-событийного моделирования было применено для моделирования компьютерного кластера.

Рассмотрим подробнее логику разработанной модели (рис. 1) компьютерного кластера для распределенно-параллельных вычислений, которую условно можно разбить на 2 части: *поступление заданий* и *их обработка*. Задание, в терминах математического аппарата систем массового обслуживания и программного пакета Arena, будем называть *сущностью*.

#### Поступление заданий

Модуль *Create 1* генерирует поступление заданий (сущностей) в систему генератором случайных чисел. Задания поступают в модуль *Decide 3*, откуда случайным образом с равной вероятностью направляются в модули *Assign 1*, *Assign 2*, *Assign 3*, *Assign 5* или *Assign 6*. После прохождения одного из этих модулей каждой сущности присваивается время обработки, которое равно значению атрибута *Attribute 2*. Время обработки, *Attribute 2*, будет влиять на время обработки поступившего задания и моделирует алгоритм обработки задания компьютерами кластера, т. е. влияет на загрузку узлов кластера – компьютеров.

Затем задания проходят через модуль *Assign 4*, где задается значение атрибута *Attribute 1*. *Attribute 1* – размер задания изменяется от 10 до 100 Мб. В результате на компьютерный кластер поступают задания, имеющие определенный конечный размер (*Attribute 1*) и требующие определенного времени обработки (*Attribute 2*).

В модуле *Decide 2* в зависимости от длины задания происходит распределение заданий по пяти очередям. В очередь к модулю *Hold 1* поступают задания, длина которых от 10 до 20 Мб, к модулю *Hold 2* – с длиной 21...40 Мб, к *Hold 3* – с длиной 41...60 Мб, к *Hold 4* – с длиной 61...80 Мб, и к модулю *Hold 5* – с длиной 81...100 Мб. Каждый модуль выпускает сущности в зависимости от назначаемого условия (количество свободных единиц ресурса и назначаемый приоритет модуля).

#### Обработка заданий

Вычислительные машины, на которых происходит обработка заданий, представлены в виде одного ресурса (*Resource 1*) с заданной емкостью 5.

Данный ресурс смоделирован в пяти различных модулях *Process*, различной емкостью: 1, 2, 3, 4, 5. В зависимости от размера, задания поступают на обработку в различные процессы и захватывают различное количество ячеек ресурса. Сущности, выходящие из очереди *Hold 1.Queue*, поступают в модуль *Process 1* и захватывают одну единицу ресурса *Resource 1*. Соответственно, сущности из очереди *Hold 2.Queue* захватывают две единицы ресурса, из *Hold 3.Queue* – три единицы, из *Hold 4.Queue* – четыре, и из *Hold 5.Queue* – все пять единиц. В общем случае длительность обработки задания равна значению атрибута *Attribute 2* сущности (время обработки задания).

После обработки на вычислительных машинах, задания поступают в модуль *Dispose 1* и удаляются из системы.

Первоначально была разработана модель, в которой наивысшим приоритетом обладали большие по размеру задания, требующие для своей обработки максимальное число ресурсов.

### Анализ вариантов распределения приоритетов обработки заданий

Проанализируем эффективность работы кластера, изменяя последовательность выборки заданий из очередей. Для этого необходимо рассмотреть работу суперкомпьютера с определенным количеством поступивших заданий.

Смоделируем обработку 100 заданий и посмотрим, как изменяются такие параметры, как время обработки заданий (рис. 2), загрузка машин (рис. 3) и среднее число заданий в очереди (рис. 4–6) при изменении последовательности обработки заданий. При наличии 5 узлов в кластере количество вариантов последовательностей обработки равно  $5!$ , т. е. 120.

Рассмотрим примеры возможных вариантов последовательностей обработки заданий:

- сначала будут обрабатываться задания с наименьшим *Attribute 1*, т. е. задания размером от 0 до 20 Мб, затем по возрастанию *Attribute 1* – от 20 до 40 Мб и т. д., таким образом, эту последовательность обозначим как 1\_2\_3\_4\_5;
- возможна обратная ситуация, когда будем обрабатывать задания по уменьшению значения *Attribute 1*, сначала обработав самые большие по размеру задания 80...100 Мб, затем 60...80 Мб и т. д.; эту последовательность обозначим 5\_4\_3\_2\_1;
- далее моделируются остальные возможные варианты обработки заданий.

На рис. 2 приведены последовательности из серий обработки заданий, представляющие наибольший интерес.

Наименьшее (лучшее для данного исследования) время обработки заданий, равное 6,28 с, получено в следующих последовательностях: 1\_4\_3\_2\_5, 1\_4\_3\_5\_2, 1\_5\_4\_3\_2, 4\_1\_3\_2\_5, 4\_3\_2\_1\_5, 4\_3\_2\_5\_1, 4\_5\_3\_2\_1. Худший результат, со временем равным 15:03:27, был получен

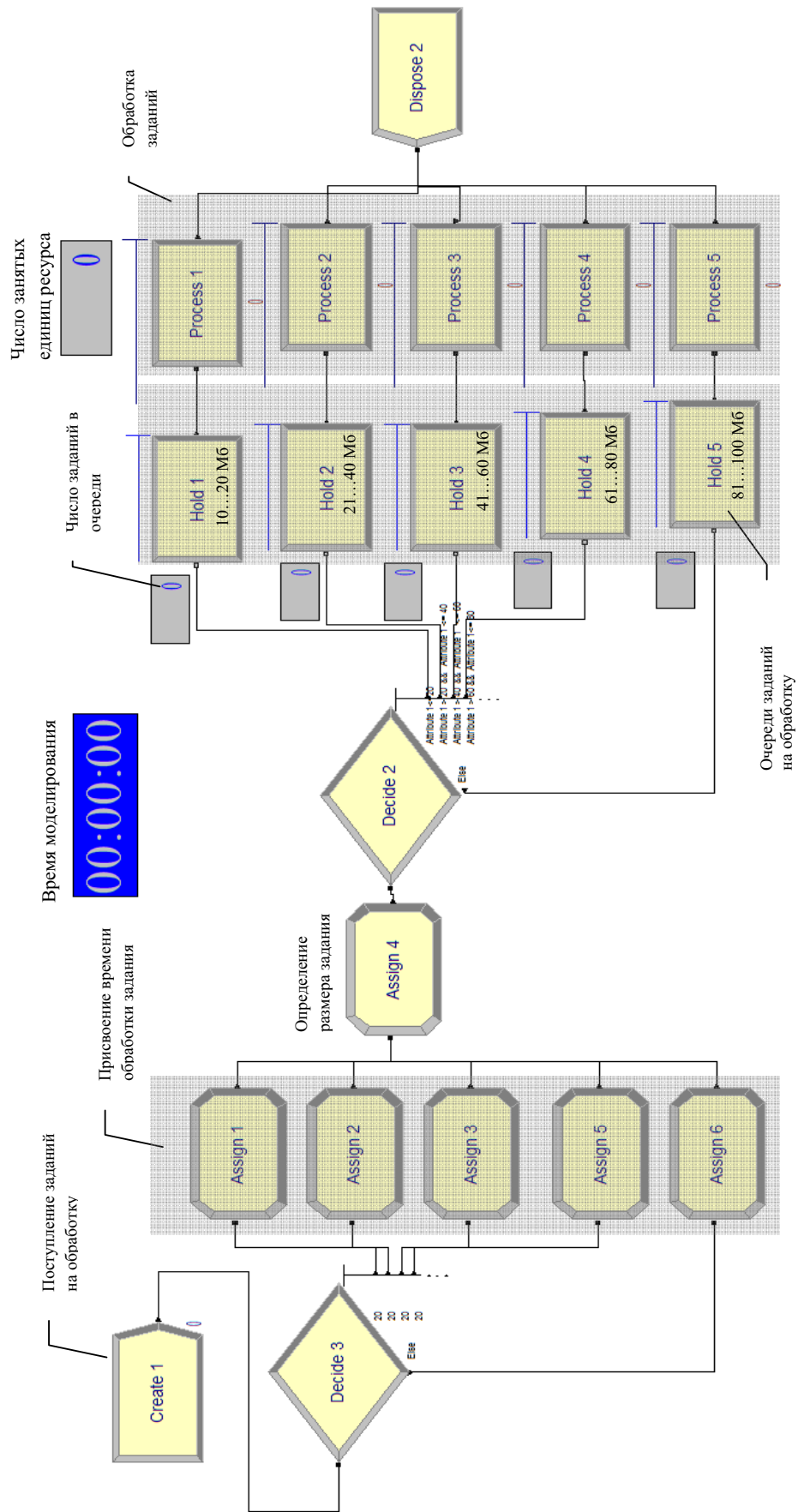


Рис. 1. Имитационная модель СКК для распределено-параллельных вычислений

в последовательностях: 2\_4\_5\_3\_1, 2\_4\_5\_1\_3, 2\_3\_4\_1\_5, 2\_3\_5\_1\_4.

На рис. 3 приведен график загруженности компьютеров кластера. Для данного исследования наилучшим результатом является максимальная загрузка компьютеров. Т. е. распределение поступающих заданий организовано таким образом, что работа кластера проходит без простоя компьютеров. Наилучшими экспериментальными данными (максимальная загрузка) обладают следующие последовательности: 1\_4\_3\_2\_5, 1\_4\_3\_5\_2, 1\_5\_4\_3\_2, 4\_1\_3\_2\_5, 4\_3\_2\_1\_5, 4\_3\_2\_5\_1, 4\_5\_3\_2\_1. Худший результат был получен в последовательностях: 2\_4\_5\_3\_1, 2\_4\_5\_1\_3, 2\_3\_4\_1\_5, 2\_3\_5\_1\_4.

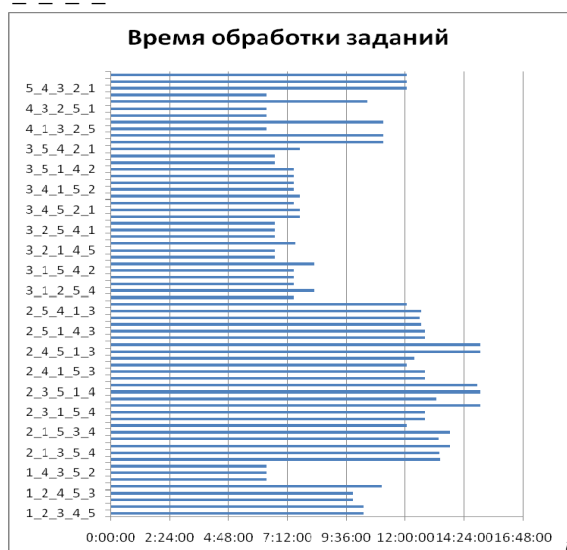


Рис. 2. Результаты эксперимента по обработке заданий

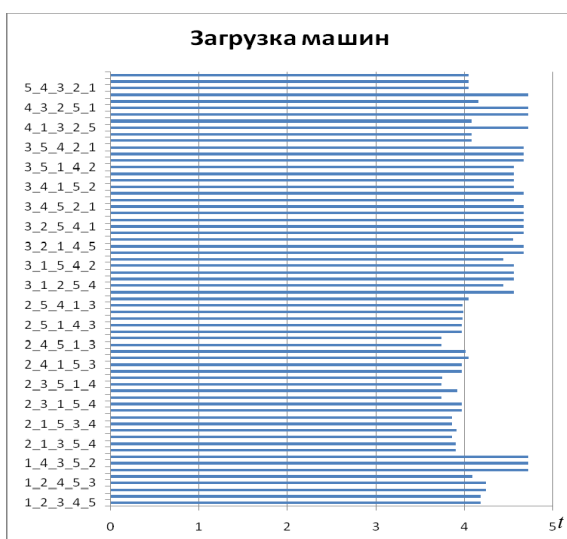
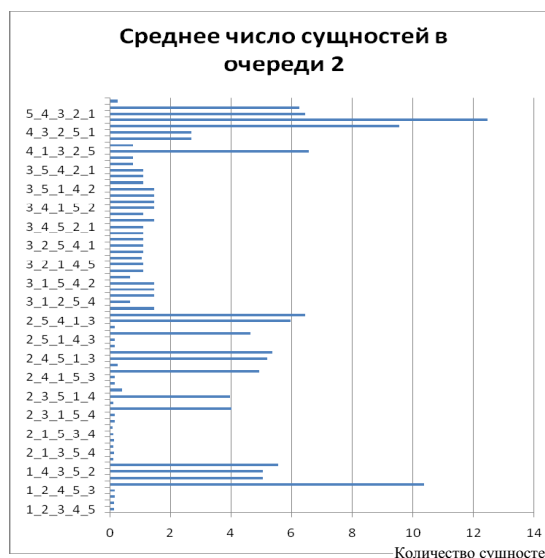
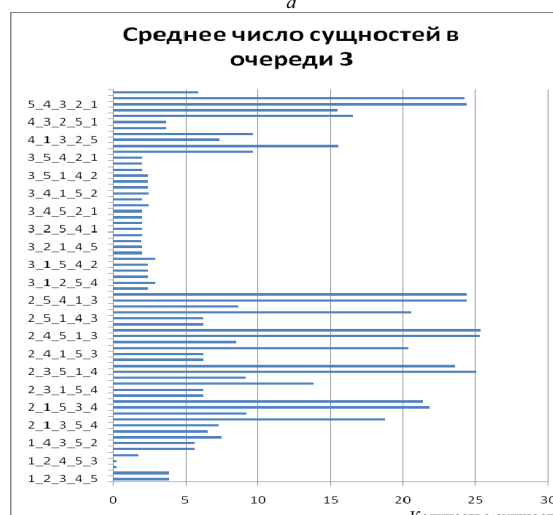


Рис. 3. Результаты эксперимента по загрузке компьютеров кластера

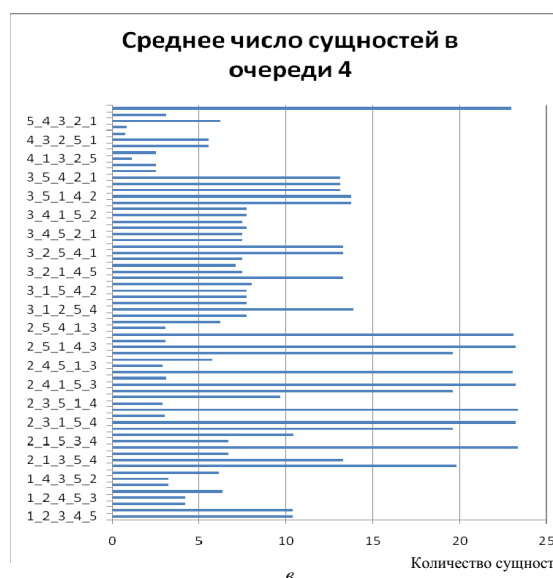
На рис. 4 приведены графики, отображающие среднее количество заданий в очереди к компьютерам моделируемого кластера. В качестве примеров выбраны графики очереди 2 ко второму компьюте-



а



б



в

Рис. 4. Результаты эксперимента по среднему числу заданий в очередях 2-4

**Таблица.** Сравнительные данные по результатам моделирования

Последовательность	Время, с	Среднее число сущностей. Очередь					Загрузка машин, с	Количество совпадений
		1	2	3	4	5		
1_2_4_3_5	9:50:43	0,0418	0,1566	<b>0,2138</b>	4,2137	17,3516	4,2454	1
1_2_4_5_3	9:50:43	0,0418	0,1566	<b>0,2138</b>	4,2137	17,3516	4,2454	1
1_4_3_2_5	<b>6:22:29</b>	0,0359	5,0602	5,6218	3,2509	16,592	<b>4,7193</b>	2
1_4_3_5_2	<b>6:22:29</b>	0,0359	5,0602	5,6218	3,2509	16,592	<b>4,7193</b>	2
1_5_4_3_2	<b>6:22:29</b>	0,0207	5,5708	7,4727	6,1213	11,6352	<b>4,7193</b>	2
2_1_5_4_3	12:04:45	0,0138	<b>0,0838</b>	21,3764	10,4509	1,1969	3,8569	1
2_3_4_1_5	15:03:27	<b>0,0000</b>	4,0222	13,8152	3,0450	7,7039	3,7348	1
2_3_5_1_4	15:03:27	<b>0,0000</b>	3,9644	25,0443	2,9005	3,9653	3,7348	1
2_4_3_1_5	12:04:45	<b>0,0000</b>	4,9231	20,3758	3,1399	7,5969	4,0431	1
2_4_5_1_3	15:03:27	<b>0,0000</b>	5,2127	25,3230	2,9005	3,0675	3,7348	1
2_4_5_3_1	15:03:27	0,0429	5,3694	25,3566	5,7620	<b>0,2370</b>	3,7348	1
2_5_3_1_4	12:39:34	<b>0,0000</b>	4,6468	20,5508	3,0902	7,6191	3,9791	1
2_5_4_1_3	12:39:34	<b>0,0000</b>	5,9767	24,3638	3,0902	3,2681	3,9791	1
4_1_3_2_5	<b>6:22:29</b>	0,0487	6,5768	7,3073	1,1409	16,592	<b>4,7193</b>	2
4_3_2_1_5	<b>6:22:29</b>	1,6565	2,6841	3,6266	5,5577	16,592	<b>4,7193</b>	2
4_3_2_5_1	<b>6:22:29</b>	1,6565	2,6841	3,6266	5,5577	16,592	<b>4,7193</b>	2
4_5_3_2_1	<b>6:22:29</b>	0,0584	12,4622	15,4654	<b>0,8454</b>	7,1423	<b>4,7193</b>	<b>3</b>

ру (рис. 4, а), очереди 3 (рис. 4, б), очереди 4 (рис. 4, в). Наилучшим результатом моделирования является минимизация количества заданий, ожидающих обработки. Таким образом, наилучшими результатами являются для компьютера:

1. 1\_3\_4\_5\_2, 1\_4\_3\_2\_5, 1\_4\_3\_5\_2, 1\_5\_4\_3\_2, 2\_3\_4\_1\_5, 2\_3\_5\_1\_4, 2\_4\_3\_1\_5, 2\_4\_5\_1\_3, 4\_5\_3\_2\_1;
2. 2\_1\_5\_4\_3;
3. 1\_2\_4\_3\_5, 1\_2\_4\_5\_3;
4. 4\_5\_2\_3\_1, 4\_5\_3\_2\_1;
5. 2\_4\_5\_3\_1.

Создадим сводную таблицу экспериментальных данных, которая позволит определить оптимальные результаты распределения заданий на обработку (табл.).

В таблице приведена выборка данных, из проведенных 120 экспериментов имитационной модели СКК, в которой были получены лучшие комбинации хотя бы по 1 параметру (выделено полужирным шрифтом): время обработки заданий, среднее число заданий в очередях, загрузка компьютеров.

**СПИСОК ЛИТЕРАТУРЫ**

1. Высокопроизводительные вычисления // Официальный сайт компании Trinity. 2009. URL: <http://www.trinitygroup.ru/> (дата обращения: 09.06.2010).
2. Гладыш В.А., Яновский В.В. Средства моделирования вычислительных сетей. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2001. – 128 с.
3. Giambene G. Queuing theory and telecommunications: networks and applications. – N.Y.: Springer Science, 2005. – 588 p.

Последний столбец таблицы отображает количество наилучших параметров по комбинациям. Проанализировав полученную таблицу, можно сделать вывод о том, что наибольшая эффективность работы СКК достигается при комбинации 4\_5\_3\_2\_1.

**Выводы**

1. При разработке модели суперкомпьютерного кластера в качестве основных параметров, характеризующих его работу, выбраны время обработки заданий, загрузка компьютеров и среднее число заданий в очереди.
2. Разработана модель суперкомпьютерного кластера с использованием программного пакета дискретно-событийного моделирования Arena.
3. Показано, что оптимальный вариант работы кластера обеспечивается при комбинации, когда первыми обрабатываются задания размером 60...80 Мб, затем 80...100 Мб, 40...60 Мб, 20...40 Мб и последними 0...10 Мб (4\_5\_3\_2\_1).

4. Law A.M., Kelton W.D. Simulation Modeling and Analysis. – N.Y.: Mc.Graw-Hill, 2000. – 544 p.
5. Замятина О.М. Моделирование систем. – Томск: Изд-во ТПУ, 2009. – 186 с.

Поступила 08.06.2010 г.