

The 8th International Conference
"Distributed Computing and Grid-technologies in Science and
Education",
GRID 2018

MODELING OF TASK SCHEDULING IN DESKTOP GRID SYSTEMS AT THE INITIAL STAGE OF DEVELOPMENT

Evgeny Gerk, Ilya Kurochkin

МОДЕЛИРОВАНИЕ ПЛАНИРОВАНИЯ ВЫЧИСЛИТЕЛЬНЫХ ЗАДАНИЙ ГРИД-СИСТЕМЫ ИЗ ПЕРСОНАЛЬНЫХ КОМПЬЮТЕРОВ

Герк Е.А., Курочкин И.И.

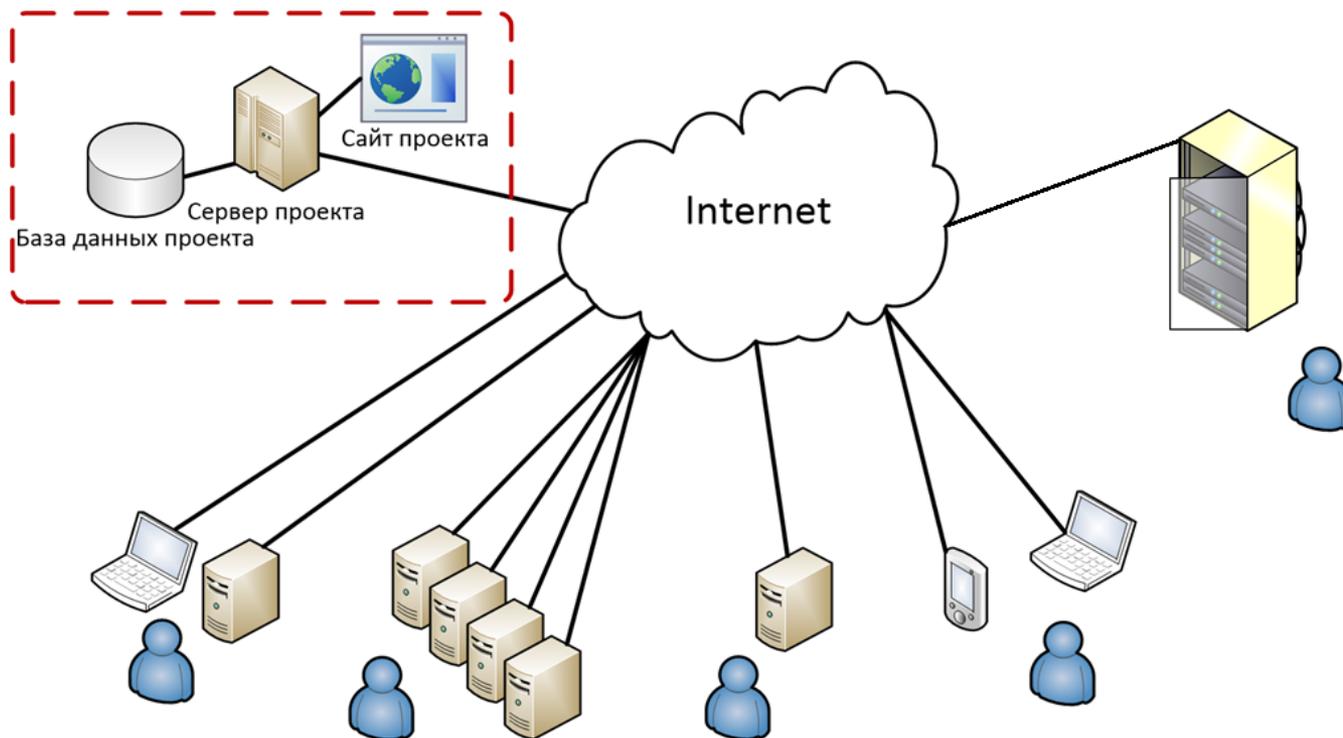
Dubna, 10-14 September 2018

Грид-системы из персональных компьютеров (ГСПК)



- ***ГСПК (Desktop Grid)*** – это система, объединяющая через Интернет неспециализированные вычислители и использующая их простаивающие вычислительные ресурсы для выполнения вычислительноемких расчетов;
- ***утилизация простаивающих ресурсов;***
- ***автономные подзадачи;***
- ***автономные вычисления на отдельных узлах.***

Особенности современных ГСПК



- простота развертывания и поддержки;
- низкая стоимость;
- высокая масштабируемость (сотни тысяч узлов);
- высокая аппаратная и программная разнородность;
- отсутствие информации о доступности узлов;
- низкая надежность узлов.

Планирование заданий в ГСПК

Планирование заданий в ГСПК – это процесс назначения задач к наиболее подходящим ресурсам, нацеленный на повышение эффективности использования грид-инфраструктуры.

- равномерное распределение нагрузки;
- производительность вычислений;
- надежность вычислений.

Содержательная постановка задачи

Цель работы - разработать адаптивный подход к составлению расписаний выполнения вычислительных заданий в грид-системе из персональных компьютеров на ранних этапах ее развития для обеспечения высокой производительности и надежности системы.

Содержательная постановка задачи

Необходимо решить ряд задач:

- провести анализ современных подходов и методов составления расписаний выполнения заданий в грид-системах;
- проанализировать и реализовать методы планирования заданий с учетом специфики ограничений ГСПК;
- промоделировать начальный этап развития ГСПК на базе инструментария GridSim;
- провести ряд экспериментов;
- осуществить сравнительный анализ предложенных методов планирования заданий.

Формализация исходных данных

Пусть $T = \{t_i : i = 1, 2, \dots, n\}$ – множество вычислительных заданий в системе, при этом каждое задание t_i обладает следующим перечнем базовых количественных характеристик:

- $length_{t_i}$ – длина (FLOP);
- $priority_{t_i}$ – приоритет;
- $deadline_{t_i}$ – дедлайн выполнения (сек);

Пусть $H = \{h_j : j = 1, 2, \dots, m\}$ – множество вычислительных узлов в системе, при этом каждый вычислительный узел h_j обладает следующим перечнем базовых количественных характеристик:

- $power_{h_j}$ – мощность (FLOP/s);
- $preference_{h_j}$ – уровень предпочтения;

Математическая постановка задачи

Расписание выполнения множества заданий T на множестве вычислительных узлов H системы определяется следующим выражением:

$$S = \{s_{ij} : i \in N, j \in M\}. \quad (1)$$

$$f(S_k^*) = \min_{k \in K} f(S_k), \quad (2)$$

где S_k^* - оптимальное расписание из множества расписаний $\hat{S} = \{S_k : k \in K\}$, K - множество номеров всех возможных расписаний для заданий из множества T , $f : \hat{S} \rightarrow R_+$ и R_+ - множество действительных положительных чисел.

Математическая постановка задачи

Пусть длина слота s_{ij} расписания S задается следующим выражением:

$$l_{s_{ij}} = e_{ij} + r_j, \quad (3)$$

где e_{ij} – ожидаемое время выполнения задания t_i на узле h_j ;

r_j – ожидаемое время готовности (освобождения) ресурса h_j .

$$e_{ij} = \frac{\text{length}_{t_i}}{\text{power}_{h_j}}. \quad (4)$$

$$r_j = r_j + e_{lj}. \quad (5)$$

Математическая постановка задачи

Тогда длина всего расписания S задается функцией:

$$L(S) = \max_{s_{ij} \in S} l_{s_{ij}}. \quad (6)$$

Уровень утилизации ресурсов грид-системы:

$$U(S) = \frac{\sum_{j=1}^m r_j}{m \times L(S)}. \quad (7)$$

И целевая функция задачи оптимизации (2) принимает следующий

вид:

$$f^* = \min_{S_k \in \hat{S}} (L(S_k), \max_{S_k \in \hat{S}} U(S_k) - U(S_k)). \quad (8)$$

Рассмотренные эвристики планирования

- Minimum Execution Time (MET)
- Minimum Completion Time (MCT)
- Min-min
- Max-min
- Resource Aware Scheduling Algorithm (RASA)
- K-percent Best (KPB)
- Load Balanced Min-Min (LBMM)
- ***Sort-Median***
- ***Penalty Sort-Median***

Алгоритм Sort-Median

$$c_{ij} = e_{ij} + r_j, \quad (9)$$

где e_{ij} – ожидаемое время выполнения задания t_i на узле h_j ,
 r_j – время готовности (освобождения) узла h_j .

Сортировка
строк матрицы c
и поиск медиан
по строкам

Назначение задания
 t_k с максимальным
значением медианы
на вычислительный
узел h_p с
минимальным
временем
завершения.

Обновление r_p
и c_{ip}

Удаление
задания t_k из
множества T

Планирование вычислительных экспериментов

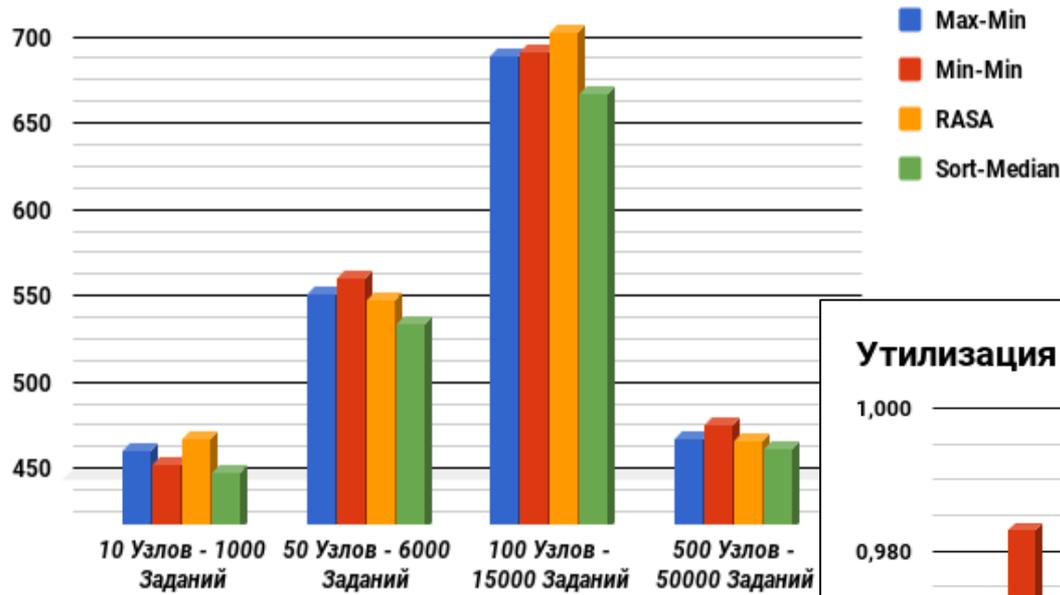
Сценарии 1-4 – без отказов, 10-500 ВУ, 1к-50к вычислительных заданий

Сценарии 5-6 – высокий/низкий разброс производительности узлов

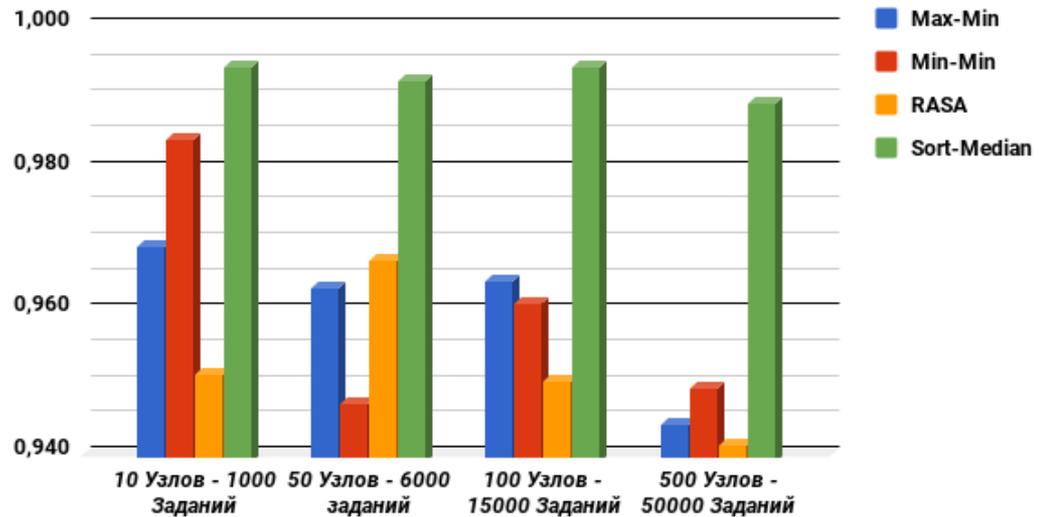
Характеристика	Значение
Количество вычислительных узлов	10
Количество вычислительных задач	1000
Производительность узлов (<i>GFLOPS</i>)	$N(50, 10)$
Длина вычислительного задания (<i>TFLOP</i>)	$N(700, 50)$

Сравнительный анализ производительности и уровня утилизации ресурсов грид-системы

Общее время выполнения заданий (час)

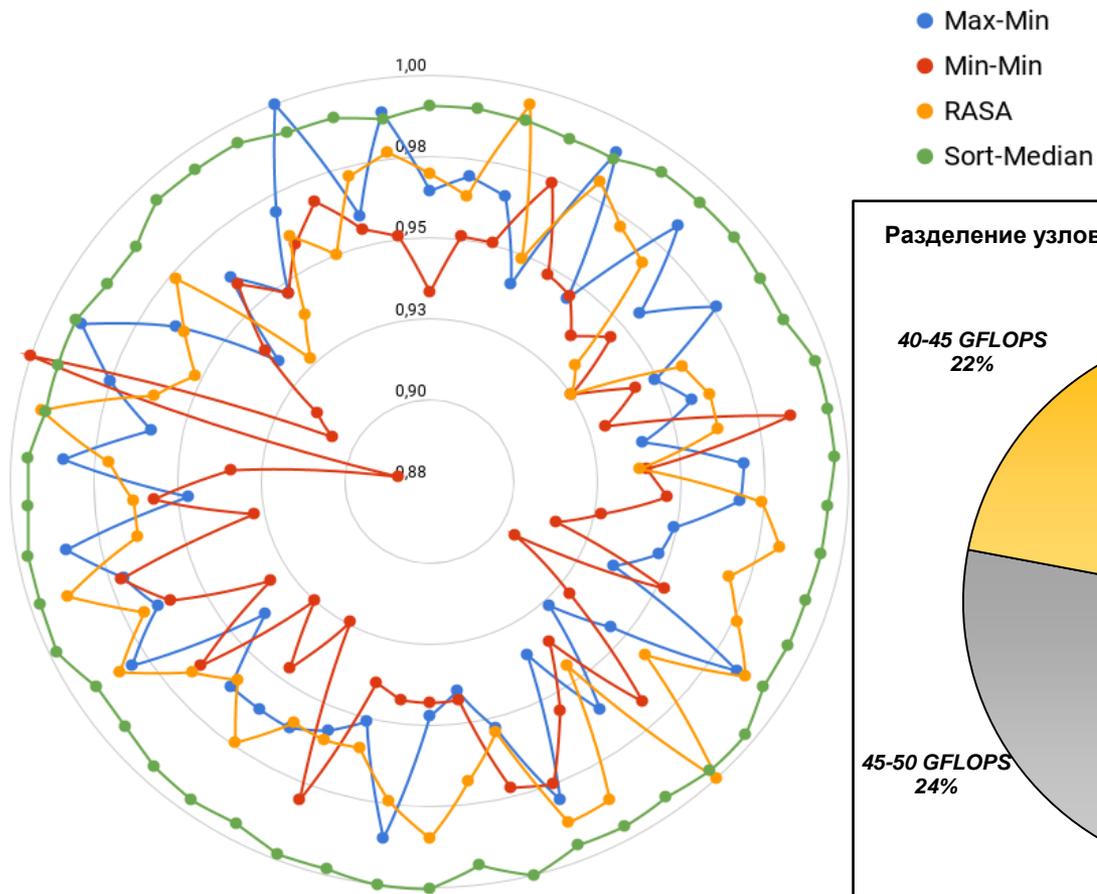


Утилизация ресурсов

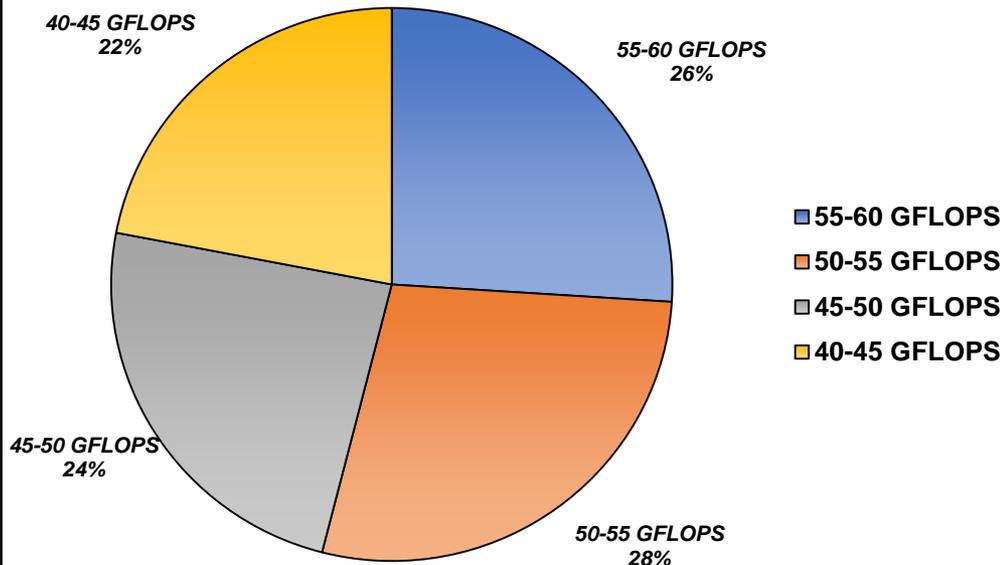


Сравнительный анализ распределения нагрузки на вычислительные узлы грид-системы

Распределение нагрузки / 50 узлов - 6000 заданий



Разделение узлов на группы по вычислительной мощности



Поиск оптимального алгоритма планирования

Допустимое множество альтернатив

		<i>Max-Min</i>	<i>Max-Min</i>	<i>RASA</i>	<i>Sort-Median</i>
x_1	$L(S)$	546,02	548,85	550,30	531,88
x_2	$U(S)$	0,96	0,95	0,95	0,98

Нормализованное множество альтернатив

		<i>Max-Min</i>	<i>Max-Min</i>	<i>RASA</i>	<i>Sort-Median</i>
x_1	$L(S)$	0,77	0,92	1,00	0,00
x_2	$U(S)$	0,83	1,00	0,99	0,00

Алгоритм Penalty Sort-Median

- краткосрочное планирование
- два типа штрафов

$$d_j = \frac{\sum_{i=1}^q (e_{ij} - a_{ij})}{q}, \quad (10)$$

где a_{ij} – актуальное время выполнения задания t_i на узле h_j , а q – количество верно посчитанных заданий узлом h_j .

$$p_j = \sum_{i=1}^g e_{ij}, \quad (11)$$

где g – количество сбоев и неверных ответов для узла h_j .

$$f_{ij} = c_{ij} + w(d_j + p_j), \quad (12)$$

где w – доля верных посчитанных результатов (валидных заданий) от всех заданий проекта

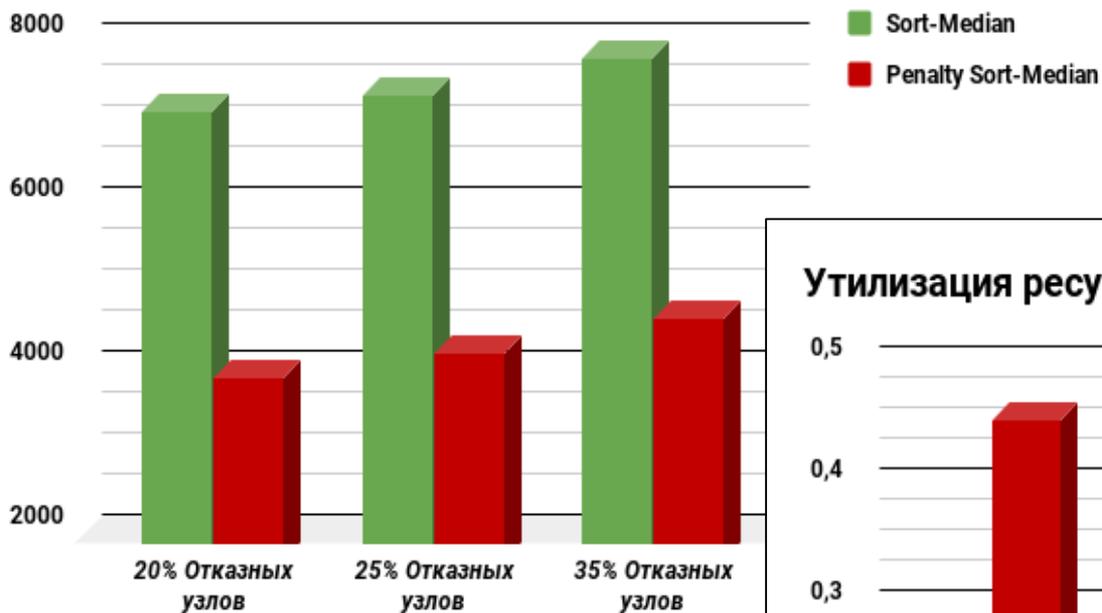
Планирование вычислительных экспериментов с отказами

Сценарии 7-9 – процент узлов, подверженных отказам, 20-35%

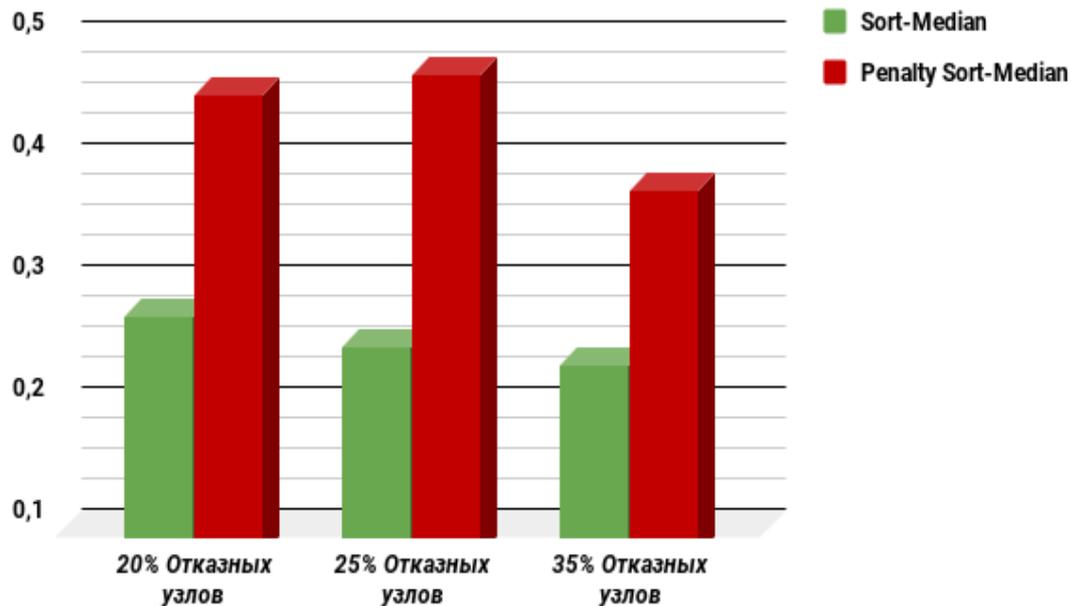
Характеристика	Значение
Количество вычислительных узлов	10
Количество вычислительных задач	1000
Производительность узлов (<i>GFLOPS</i>)	$N(50, 10)$
Длина вычислительного задания (<i>TFLOP</i>)	$N(700, 50)$
Процент узлов, подверженных отказам (%)	20-35
Время наступления отказов (час)	$U(0, 700)$

Сравнительный анализ производительности и уровня утилизации ресурсов грид-системы при наличии отказов

Общее время выполнения заданий (час)



Утилизация ресурсов



Применение результатов

Полученные результаты исследований:

- разработанные эвристические алгоритмы планирования заданий в ГСПК;
- результаты имитационного моделирования;
- сравнительный анализ алгоритмов с уже существующими.

Данные результаты будут использованы в центре распределенных вычислений ИППИ РАН при разворачивании проектов распределенных и добровольных распределенных вычислений на базе платформы VOINC.

Выводы

- Проведен анализ проблем современных грид-систем из персональных компьютеров и методов планирования заданий в таких системах.
- Разработано приложение на базе библиотек инструментария GridSim для моделирования и тестирования алгоритмов.
- Промоделированы различные сценарии раннего этапа развития ГСПК и получены оценки производительности, а также уровня утилизации ресурсов системы.
- Проведен анализ полученных результатов и сравнительный анализ исследованных алгоритмов планирования.
- Благодаря эвристики Sort-Median удалось повысить производительность грид-системы на 3% в сравнении с уже существующими алгоритмами.
- Эвристика Penalty Sort-Median снизила количество неправильных ответов на 30%.

Спасибо за внимание

<http://distributed-computing.ru>

E-mail: kurochkin@iitp.ru