



**ПОЛИТЕХ**

Санкт-Петербургский  
политехнический университет  
Петра Великого

COMPUTO ERGO SUM

**XXXV МЕЖДУНАРОДНАЯ НАУЧНАЯ КОНФЕРЕНЦИЯ ММТТ-35»**

**УПРАВЛЕНИЕ ПРОЦЕССАМИ ПЛАНИРОВАНИЯ ЗАДАНИЙ  
В ГЕТЕРОГЕННОМ СУПЕРКОМПЬЮТЕРНОМ ЦЕНТРЕ  
С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ**

Работа выполнена в рамках  
госзадания ФГАОУ ВО  
СПБПУ FSEG-2022-0001

Минск,  
25 октября 2022 г.

Заборовский В.С., Уткин Л. В., Лукашин А. А.,  
Мулюха В.А.  
Политехнический университет Петра Великого,  
Санкт-Петербург

- Введение в проблему. Статистика ТОП 500
- «Мировая линия» эволюции КТ: «Less Moor more Brain»
- Чему надо «учить» современный суперкомпьютер
- «Умный планировщик», использующий «суррогатную» модель пользователя
- Выводы

Ключевые слова: *управление высокопроизводительными вычислениями, машинное обучение, «умное» планирование*



# ВВЕДЕНИЕ.

политех

## Классическая парадигма научных знаний

- **Cogito** ergo sum («**Мыслю**, следовательно, существую» - ).

была 17 в сформулирована Р. Декартом.

Однако, противоречия этой парадигмы с практикой экспериментальных исследований **стали ясным указанием на необходимость объединить методы «дедукции» и «индукции».**

в 18 веке это отметил И. Кант в книге «Критика чистого разума», в 20 веке в теореме Левингейма-Скулемато уже было показано, что у каждой модели счётной сигнатуры имеется элементарная подмодель произвольной мощности, а доказанные утверждения К. Геделя о «неполноте» формальных систем» стали основой новой (вычислительной) парадигмы процессов мышления

### - Computo ergo sum

практическое воплощение которой в настоящее время реализовано в технологиях «машинного обучения», которое рассматривается как обобщение **формализмов теории вероятности, статистики и алгоритмической теории информации**, предложенных А. Н. Колмогоровым, на случай «больших данных».

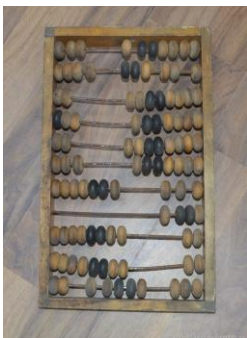




# «МИРОВАЯ ЛИНИЯ» ЭВОЛЮЦИИ: ОТ ПРОГРАММНЫХ АВТОМАТОВ К «УМНЫМ» ВЫЧИСЛИТЕЛЬНЫМ ПЛАТФОРМАМ

**Физика достигла таких высот, что мы можем вычислить даже то, что невозможно себе представить**  
*Л. Ландау*

Эра  
**механических автоматов, исполняющих** алгоритм, представленный в структуре самого «вычислителя»



эра  
**конечных автоматов, которые реализуют** заранее составленные программы вычислений

**Программа – мыслимая последовательность команд вычислений**

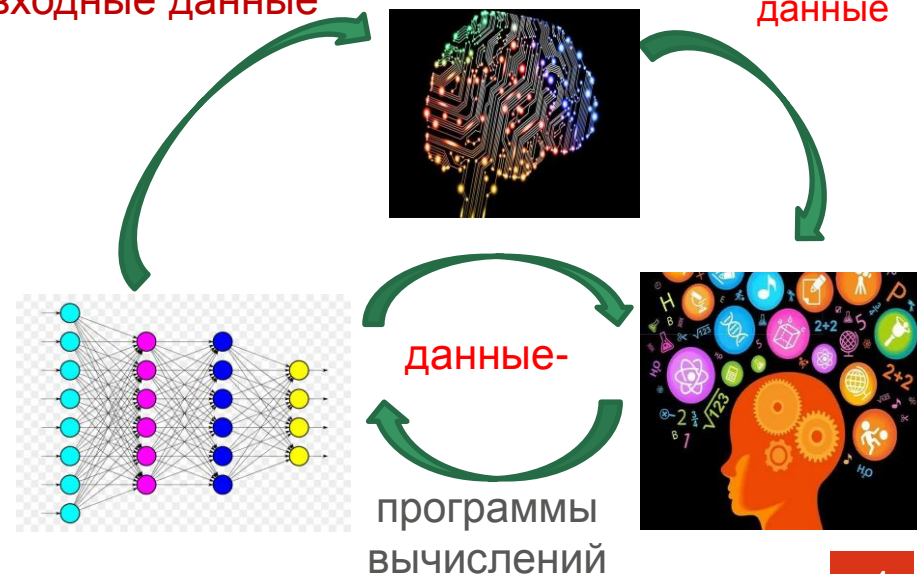


**Суть «умного» подхода:** «бесконечную ленту» из счетного множество команд «машины Тьюринга» заменить на «функцию распределения» множества входных  $X$  и целевых переменных уактуальных в момент времени  $t$  - оператор  $P(X, y, t)$

эра  
**«умных» компьютерных платформ,** которые реализуют алгоритмы вычислений, формируемые «в процессе» обработки входных и целевых данных

**X-входные данные**

**y –целевые данные**



год	число ядер	R <sub>реак</sub> , ПФлопс	R <sub>max</sub> , ПФлопс	эл. мощность, МВт
2022 Frontier	<b>8,700,XXX</b>	1680.XX (1.7 ЭФлопс)	1100.XX (1.1 ЭФлопс)	<b>21</b>
2020 Fugaku	7,300,XXX	513.XX	415.XX	28
2010 Tianhe-1	186,XXX	4.7X	2.6X	4
2000 ASCI Intel	9,6XX	0.03	0.02	-

- 1 кг угля -> 3 кВтч = 0.003 МВтч
- 1 тонна угля -> 3 МВтч

21 МВт -> 21/3 =

- 7 тонн в час
- 168 тонн в день
- 60480 тонн в год

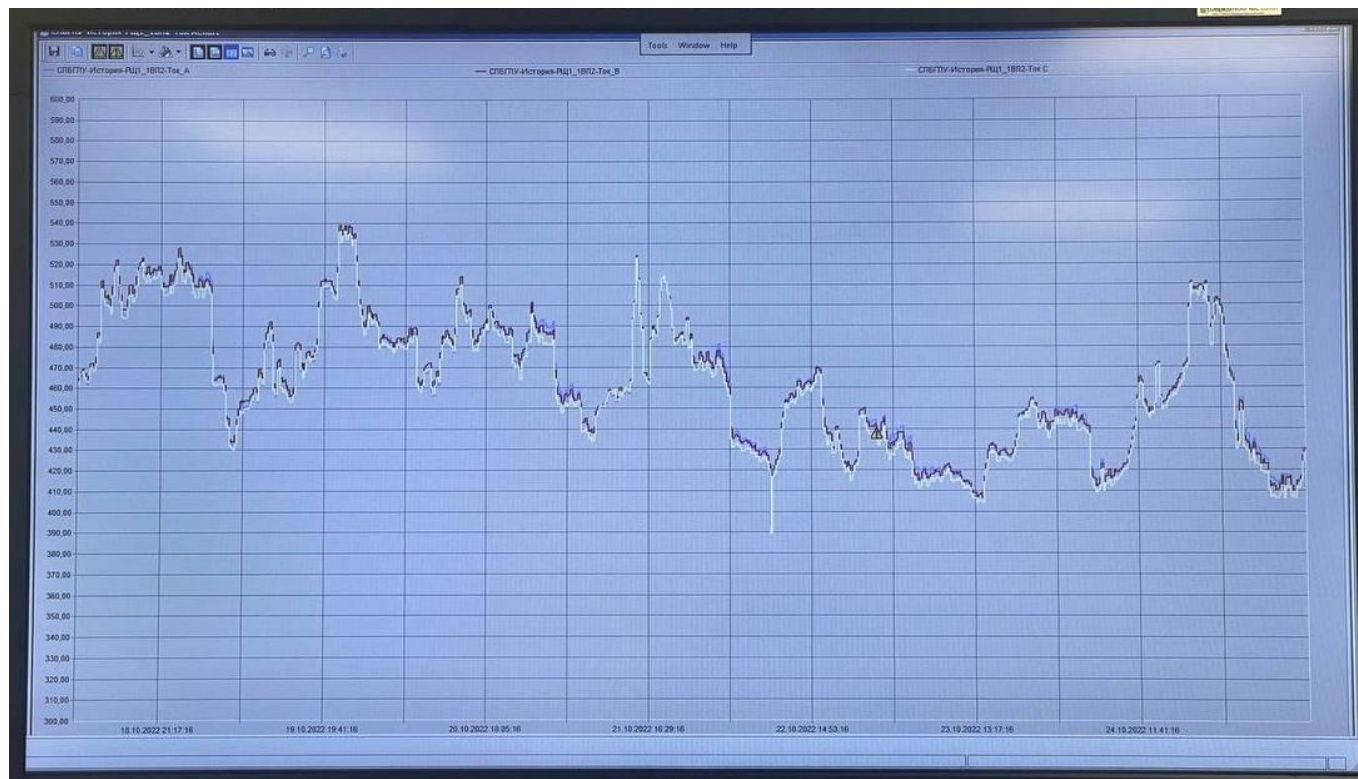
Вся Европа **потребляет**  
400 000 **МВт.**

По линии электропередачи (500 кВ) можно передать  
500 **МВт**

## Проблемы:

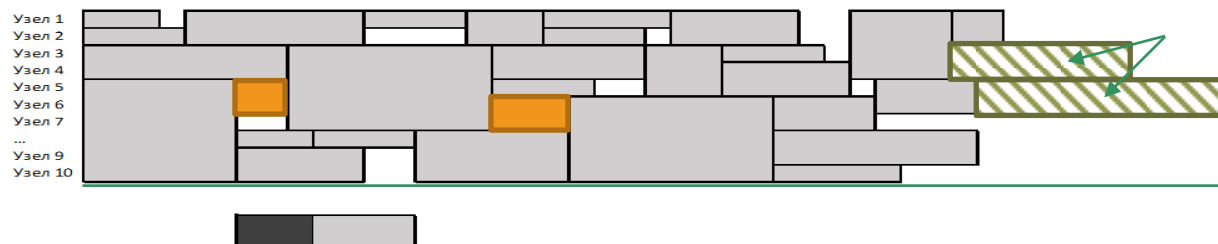
- **УПРАВЛЕНИЕ ПРОЦЕССАМИ ПЛАНИРОВАНИЯ ЗАДАНИЙ** так, чтобы **снизить ПОТРЕБЛЯЕМУЮ МОЩНОСТЬ** (уменьшить «углеродный след цифровизации»)

# ГРАФИК ПОТРЕБЛЕНИЯ ЭЛ. МОЩНОСТИ ПРИ ВЫПОЛНЕНИИ ЗАДАНИЙ



Ошибки ПЛАНИРОВАНИЯ ЗАДАНИЙ  
ПРИВОДЯТ К НЕРАВНОМЕРНОСТИ  
ЗАГРУЗКИ ВЫЧИСЛИТЕЛЬНЫХ  
РЕСУРСОВ

«визуализация»  
очереди исполняемых  
заданий в узлах  
кластера :





# А «ПОЧЕМУ ЗАКОН МУРА» УЖЕ НЕ В ПОМОЩЬ: КАТАСТРОФИЧЕСКАЯ ДЕГРАДАЦИЯ ПРОИЗВОДИТЕЛЬНОСТИ СК

HPCG Benchmark June 2020

Rank	Site	Computer	Cores	HPL Rmax (Pflop/s)	TOP500 Rank	HPCG (Pflop/s)	Fraction of Peak
1	RIKEN Center for Computational Science Japan	<b>Fugaku</b> , Fujitsu A64FX, Tofu	7,299,072	415.53	1	13.4	2.5%
2	DOE/SC/ORNL USA	<b>Summit</b> , AC922, IBM POWER9 22C 3.7GHz, Dual-rail Mellanox FDR, NVIDIA Volta V100, IBM	2,414,592	143.50	2	2.926	1.5%
3	DOE/NNSA/LLNL USA	<b>Sierra</b> , S922LC, IBM POWER9 20C 3.1 GHz, Mellanox EDR, NVIDIA Volta V100, IBM	1,572,480	94.64	3	1.796	1.4%
4	Eni S.p.A. Italy	<b>HPC5</b> , PowerEdge, C4140, Xeon Gold 6252 24C 2.1 GHz, Mellanox HDR, NVIDIA Volta V100	669,760	35.45	6	0.860	2.4%
5	DOE/NNSA/LANL/SNL USA	<b>Trinity</b> , Cray XC40, Intel Xeon E5-2698 v3 16C 2.3GHz, Aries, Cray	979,072	20.16	11	0.546	1.3%
6	NVIDIA USA	<b>Selene</b> , DGX SuperPOD, AMD EPYC 7742 64C 2.25 GHz, Mellanox HDR, NVIDIA Ampere A100	277,760	27.58	7	0.5093	1.8%
7	Natl. Inst. Adv. Industrial Sci. and Tech. (AIST) Japan	<b>ABCI</b> , PRIMERGY CX2570M4, Intel Xeon Gold 6148 20C 2.4GHz, Infiniband EDR, NVIDIA Tesla V100, Fujitsu	391,680	16.86	12	0.5089	1.7%
8	Swiss National Supercomputing Centre (CSCS) Switzerland	<b>Piz Daint</b> , Cray XC50, Intel Xeon E5-2690v3 12C 2.6GHz, Cray Aries, NVIDIA Tesla P100 16GB, Cray	387,872	19.88	10	0.497	1.8%
9	National Supercomputing Center in Wuxi China	<b>Sunway TaihuLight</b> , Sunway MPP, SW26010 260C 1.45GHz, Sunway, NRCP	10,649,600	93.01	4	0.481	0.4%
10	Korea Institute of Science and Technology Information Republic of Korea	<b>Nurion</b> , CS500, Intel Xeon Phi 7250 68C 563584C 1.4GHz, Intel Omni-Path, Intel Xeon Phi 7250, Cray	570,020	13.93	18	0.391	1.5%

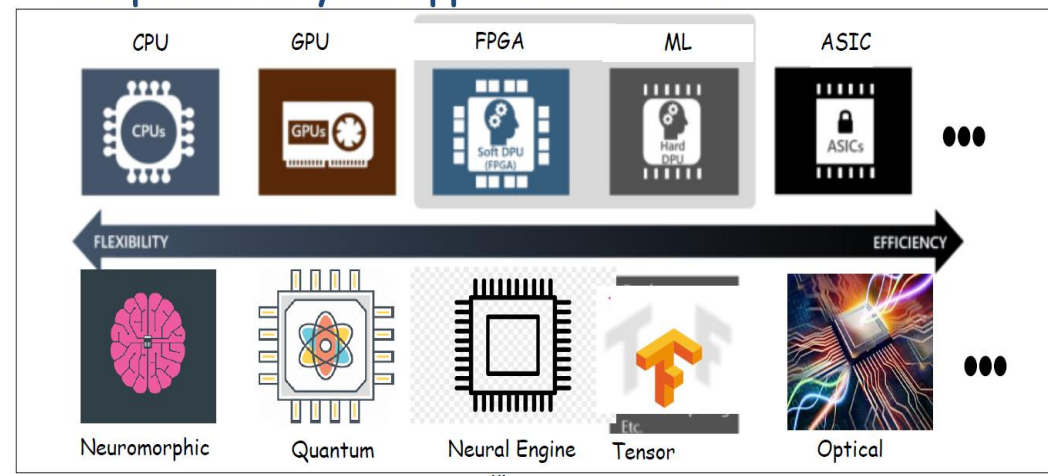
Смена алгоритма решения  
СЛАУ с LU разложения на  
метод сопряженных  
градиентов  
«катастрофически»  
деградирует  
производительность СК из  
рейтинга TOP 500

# «КТО ВИНОВАТ И ЧТО ДЕЛАТЬ»: ЧЕМ ЗАМЕНИТЬ «ЗАКОН МУРА» ???

Новая формула стратегии развития ИТ:

Future HPC Systems Will be Customized...

- ♦ You will be able to dial up what you need in your computer for your application mix ...



## Less Moore, More Brain

Меньше Мура, Больше Мозга

- ♦ HPC will have extreme heterogeneity and build custom systems for each important application.



«More Brain» – это как и ...куда?

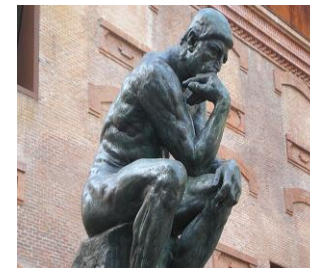
Jack Dongarra (основатель рейтинга суперкомпьютеров TOP 500)



Обучающая выборка **10<sup>4</sup> КТ снимков**

$p=10^{10}$  - число настраиваемых  
весов нейронной сети

## КТ легких



Для обучения н/с с точностью классификации **90%** надо:

- Число операций  $\approx 10^{17}$

Время обучения на СК  $\approx 10^2$  сек

- t обучения на ПК  
(200 ГФлопс)  $\approx$   **$2 \times 10^6$  сек**

**Выход у:**

**Вход х:**

сверточная  
нейронная се  
ция:

KHO:

$$\begin{array}{r} 100 \\ \times 100 \\ \hline \end{array}$$

## Вектор

признаков  $\approx 10$

Число слоев н/с  $10^2$ 

## Скользящее окно алгоритма свертки

**n** – число слоев  
КТ ( n=100)

«Вес» данных КТ  
одного слоя  $2 \times 10^6$  Байт

Входной слой нейронной сети  
размером  **$10^3$  нейронов**

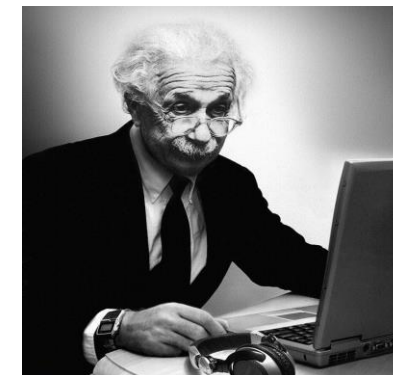
## Градиентный алгоритм обучения глубокой сверточной нейронной сети:

- функция ошибки  $F = \|y^* - \hat{y}\|^2$ ,  $y^*$  - эталонный вектор признаков
- **Алгоритм использует  $10^{10}$  частных производных**  $F$  по всем настраиваемым параметрам;
- Число операций численного **дифференцирования** на одну итерацию  **$Q = 10^{15}$**

## А нужен ли ученым и инженерам «черный ящик» ?

# «MORE BRAIN» НАЧИНАЕТСЯ С ... УМНОГО ПЛАНИРОВЩИКА ЗАДАНИЙ

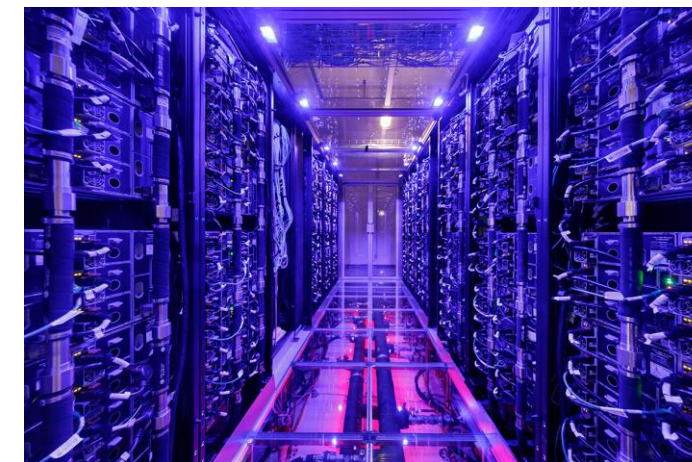
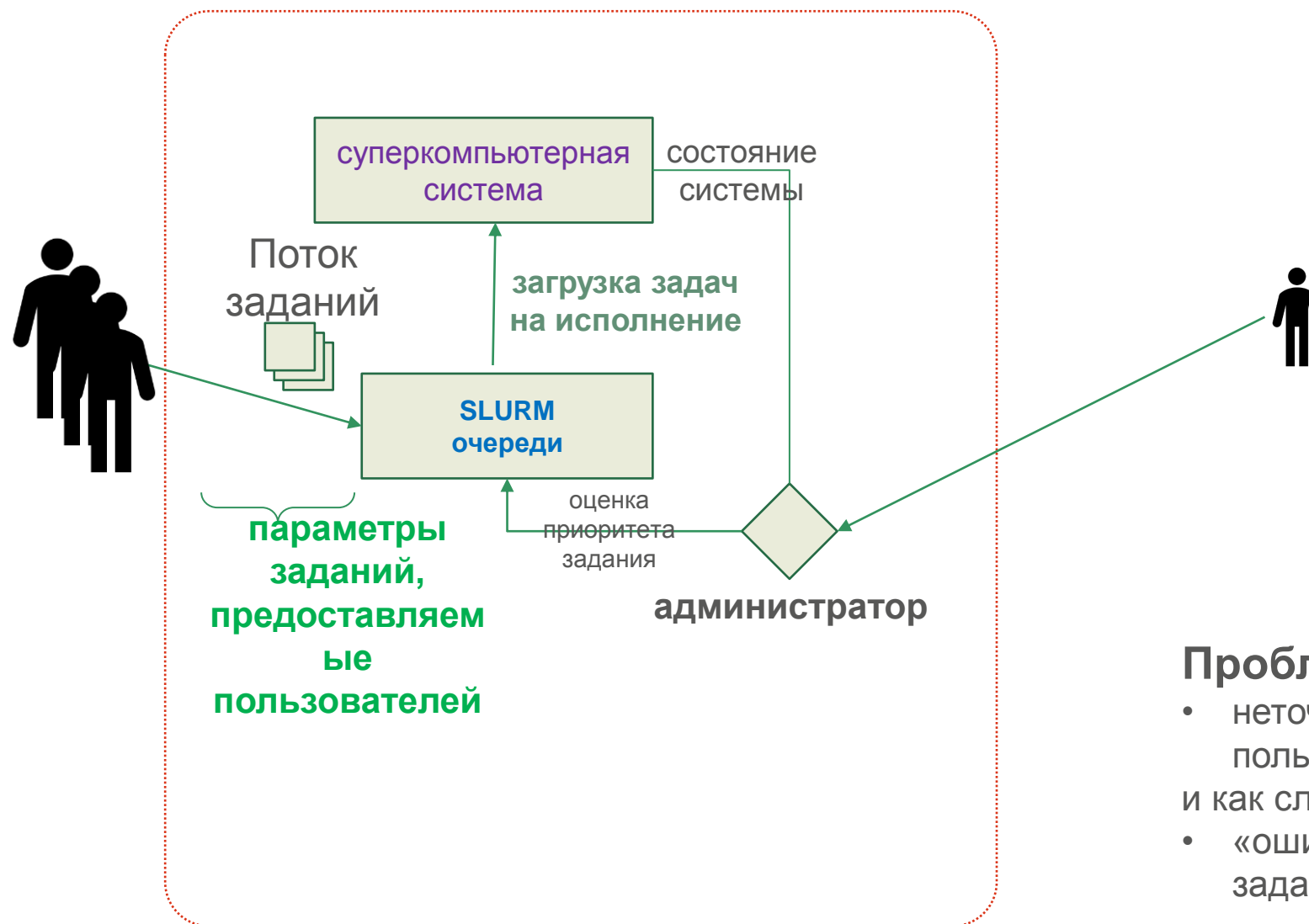
- Масштаб и сложность вычислений продолжают увеличиваться, а планирование процессов выполнения заданий **становится все более сложным процессом**, поэтому :
  - особую актуальность приобретают подходы к созданию систем управления, обеспечивающих **точное прогнозирование времени выполнения заданий** с целью формирования равномерной загрузки имеющихся вычислительных ресурсов: &...снижения потребляемой электрической мощности.
- Исследование «лучших практик» показали, пользователь — неотъемлемая часть СК системы в целом, а в **функции планировщика заданий** должны входить не только анализ данных
  - о состоянии ресурсов **непосредственно самого СК** (загрузка узлов, наличие свободных ресурсов памяти и др.), **но и ....**
  - предоставляемых **пользователем** при формировании заданий



«умный»  
планировщик  
заданий



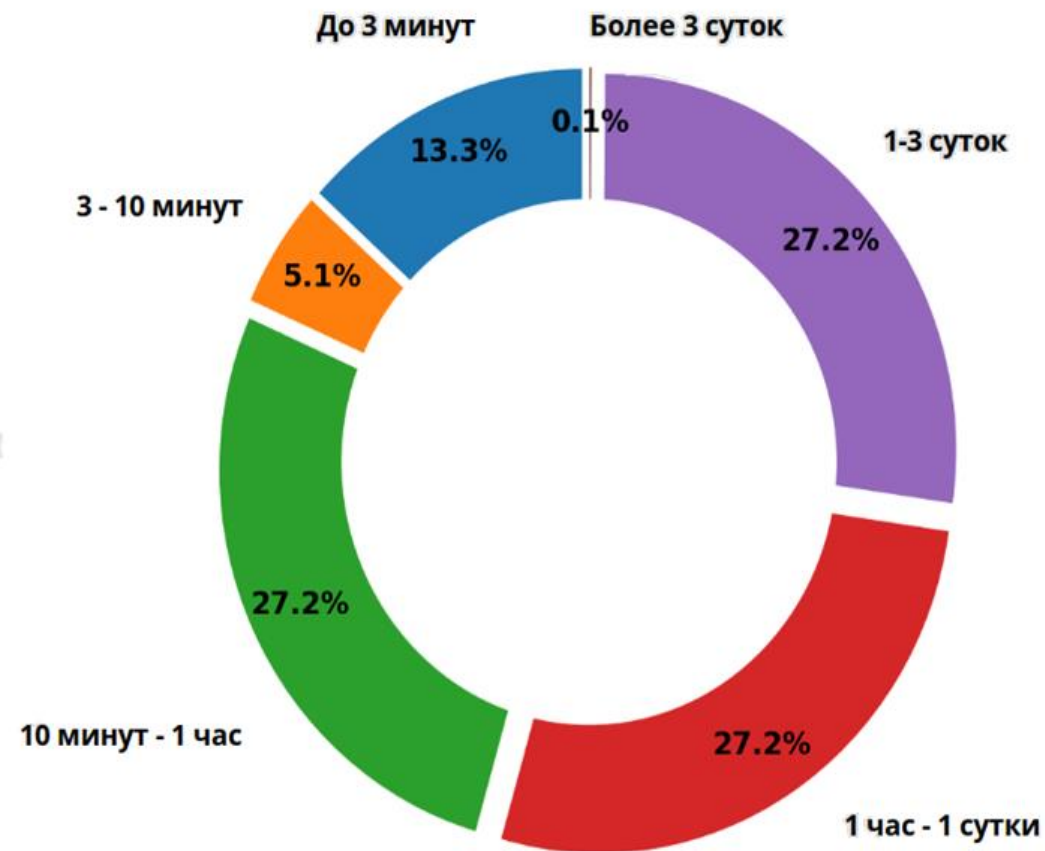
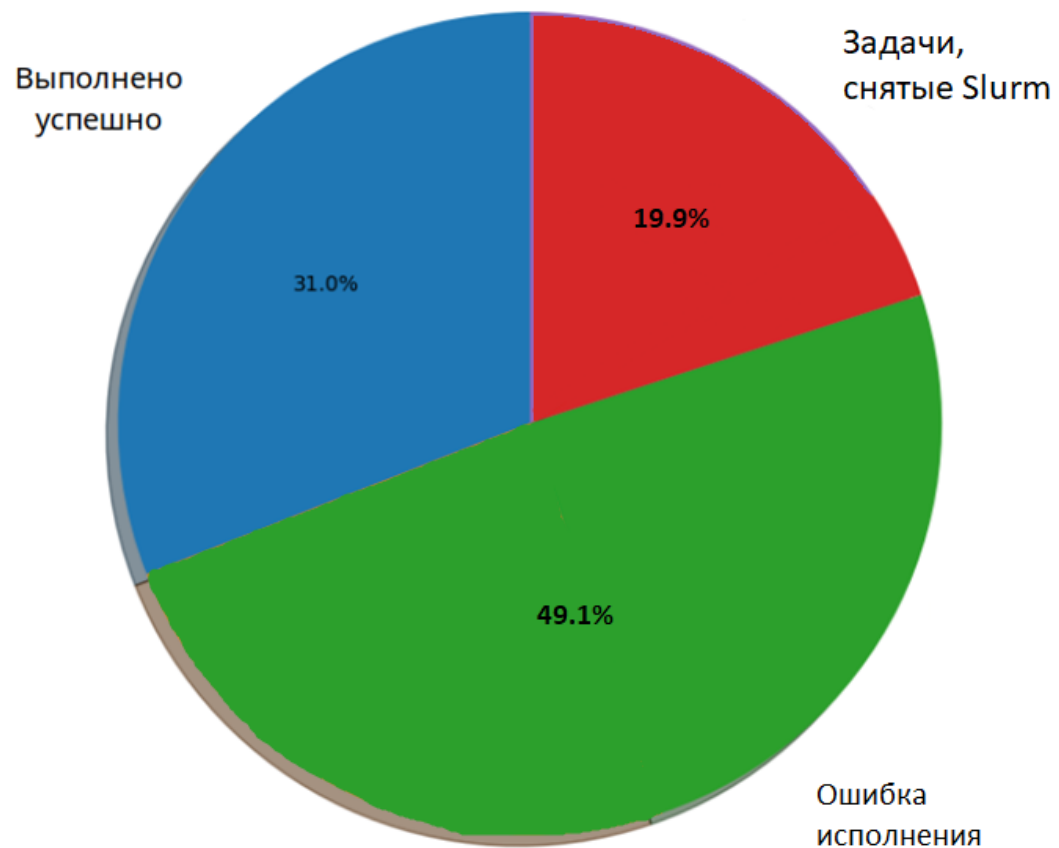
# КАК ПРОЦЕСС ОБРАБОТКИ ЗАДАНИЙ ОРГАНИЗОВАН СЕЙЧАС



## Проблемы:

- неточность в параметрах заданий пользователей
- и как следствие
- «ошибки» slurm в оценке времени исполнения заданий

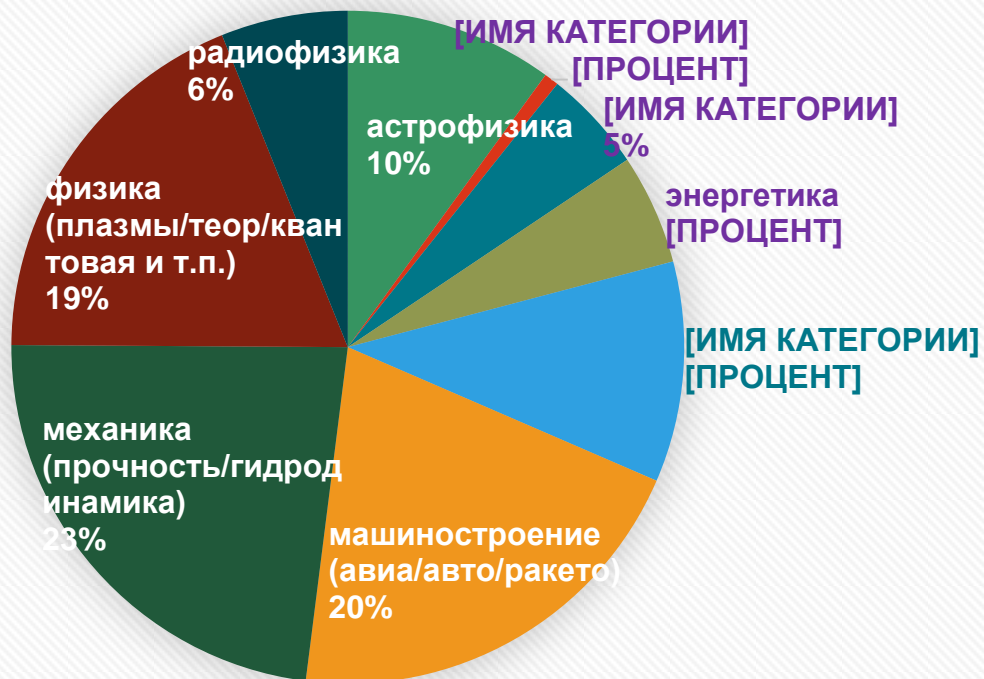
## МОТИВАЦИЯ ПЕРЕХОДА К «УМНЫМ» РЕШЕНИЯМ. СТАТИСТИКА «ВЫЖИВАНИЯ» И ПРОДОЛЖИТЕЛЬНОСТИ



«Успешно выполнено» - составляет **меньше 1/3** от общего числа обработанных заявок пользователей



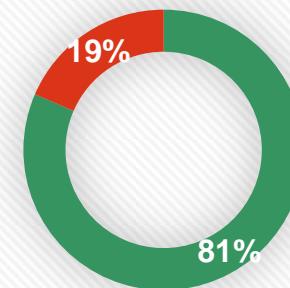
# Модель «КОМПЕТЕНЦИИ» ПОЛЬЗОВАТЕЛЕЙ. СТАТИСТИКА ПО КЛАССАМ ЗАДАЧ ( ПЕРИОД 2021-2022 гг.)



- астрофизика
- биофизика
- геофизика (сейсмика/геофизика)
- механика (прочность/гидродинамика)
- радиофизика
- биоинформатика
- энергетика (энергомаш)
- машиностроение (авиа/авто/ракето)
- физика (плазмы/теор/квантовая и т.п.)



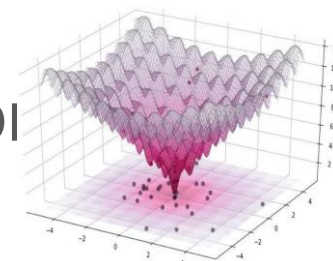
■ Прикладной ■ Фундаментальный



■ Внутренний ■ Внешний

Так как запуск задачи пользователя на исполнение не гарантирует ее успешное завершение, нужны «умные» метри

- «тонкая» настройка заданий пользователей с использованием механизма внимания, когда каждому параметру задания ставится в соответствие «эмбединг» вектор внимания
- «объяснимость полученных результатов» (успешное завершение, снятие диспетчером, ошибка в процессе выполнения)



Необходимо использовать также метрики производительности СК, которые позволяет определять какие:

- параметры заданий следует принять, чтобы повысить вероятность «выживания» задания в процессе его исполнения
- стратегии исполнения заданий могут обеспечить равномерную загрузку имеющихся аппаратных ресурсов СК

“Одно дело различать вещи и совсем другое –  
познавать различие между вещами”  
И. Кант

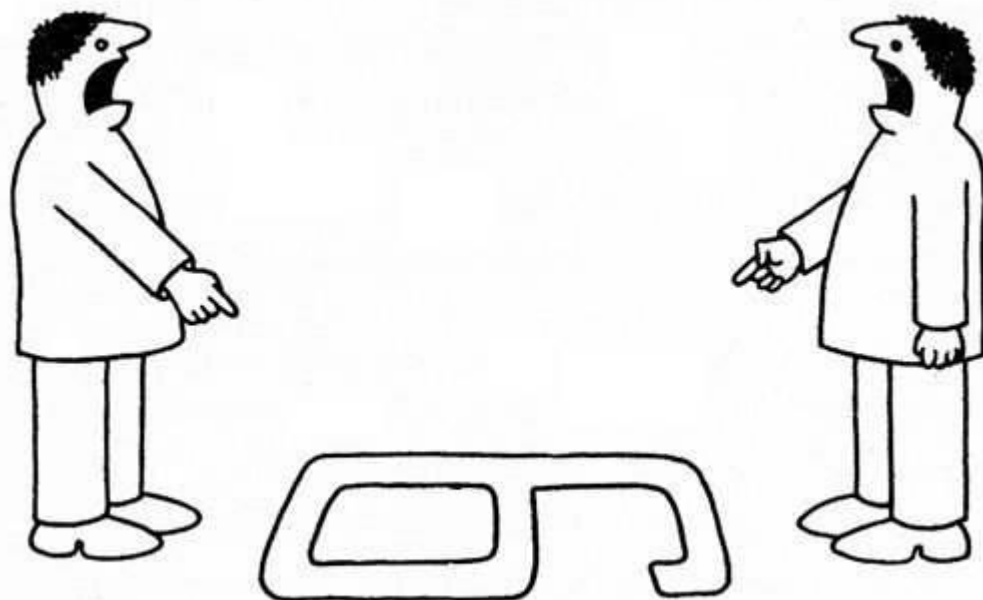
В основе интеллектуализации лежит:

- 1) абстракция отождествления и сравнения индивидуально различных объектов ( А.А. Марков, 1954 г.) и
- 2) 2) принцип тождества неразличимых (Г. Лейбниц, 1646-1716).

Для реализации концепции интеллектуализации планировщика необходимо:

- Отказ от использования параметров исполняемых заданий несущественных для выбранного критерия эффективности...
- Замена многих «эквивалентах» параметров на один абстрактный с помощью оператора «обобщения»
- Учитывать, что для различных заданий пользователей фактор внимания ( ВЕКТОР СОПОСТАВЛЯЕМЫЙ КАЖДОМУ ПАРАМЕТРУ В ЗАДАНИИ ПОЛЬЗОВАТЕЛЯ) может быть многозначным и контекстно зависимым, т.е. может динамически меняться в процессе выполнения задания.....

# СУТЬ ВОЗНИКАЮЩИХ СЛОЖНОСТЕЙ - В «ОТНОСИТЕЛЬНОСТИ» ФАКТОРА ВНИМАНИЯ



Суть в том, что при рассмотрении потока входных данных **принимать во внимание** лишь те их них, которые в данной ситуации по тем или иным причинам **оказываются**

**существенными** для выбранных целей использования полученных результатов, **игнорируя** другие — несущественные (Марков А.А., Теория алгоритмов., 1954)

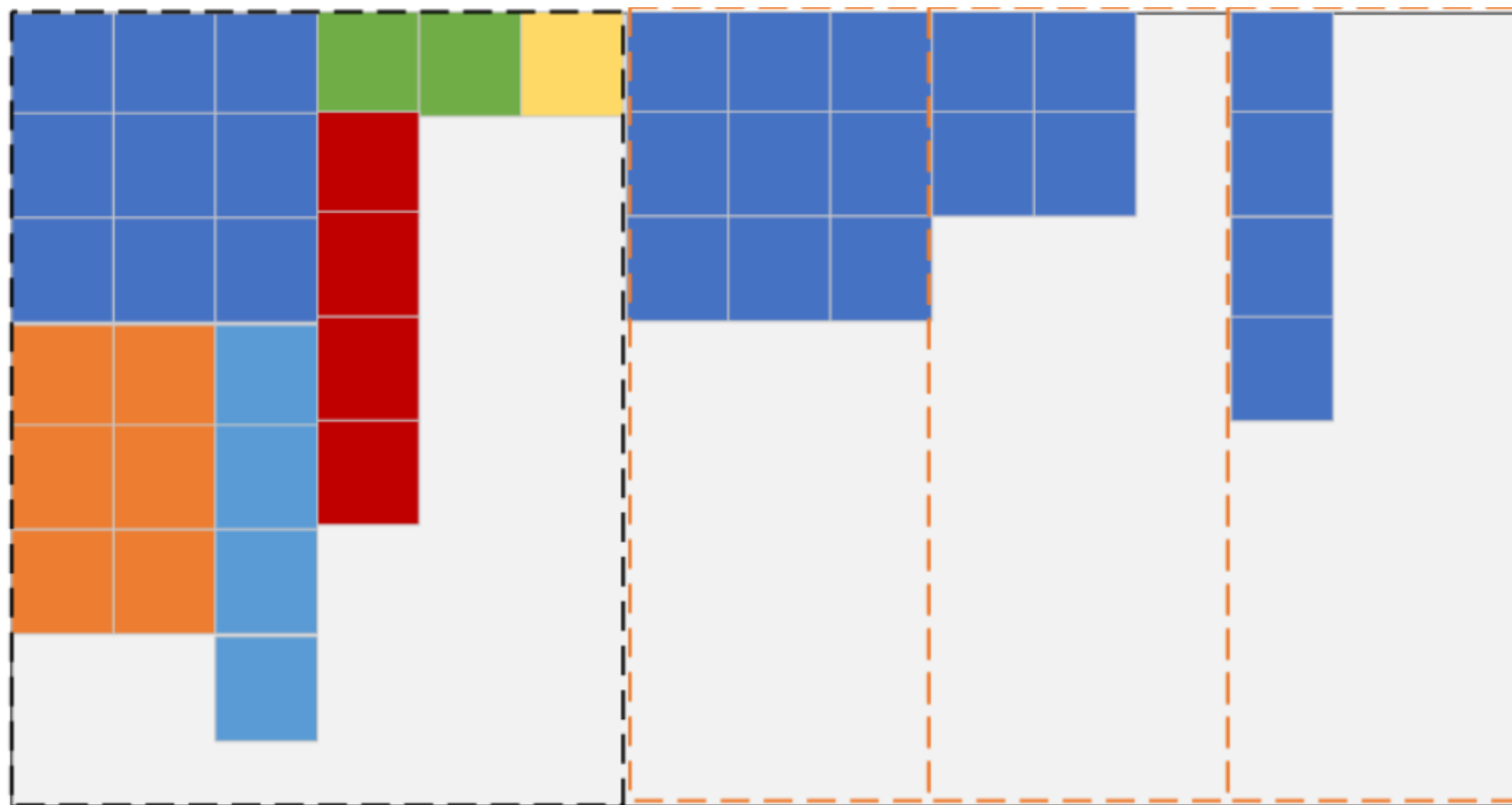
В рассматриваемой задаче «планирования» заданий с точки зрения эффективности использования ресурсов **«существенный» параметр — точная оценка**





ПОЛИТЕХ

# ВИЗУАЛИЗАЦИЯ ЗАГРУЗКИ СК И ОЧЕРЕДИ ЗАДАНИЙ



**Статус кластера**

(цветом отмечены уровни загрузки узлов кластера)

**Статус очереди на исполнение**

Наличие «разрывов» в потоке исполнения заданий из-за ошибок в **оценке времени решений**

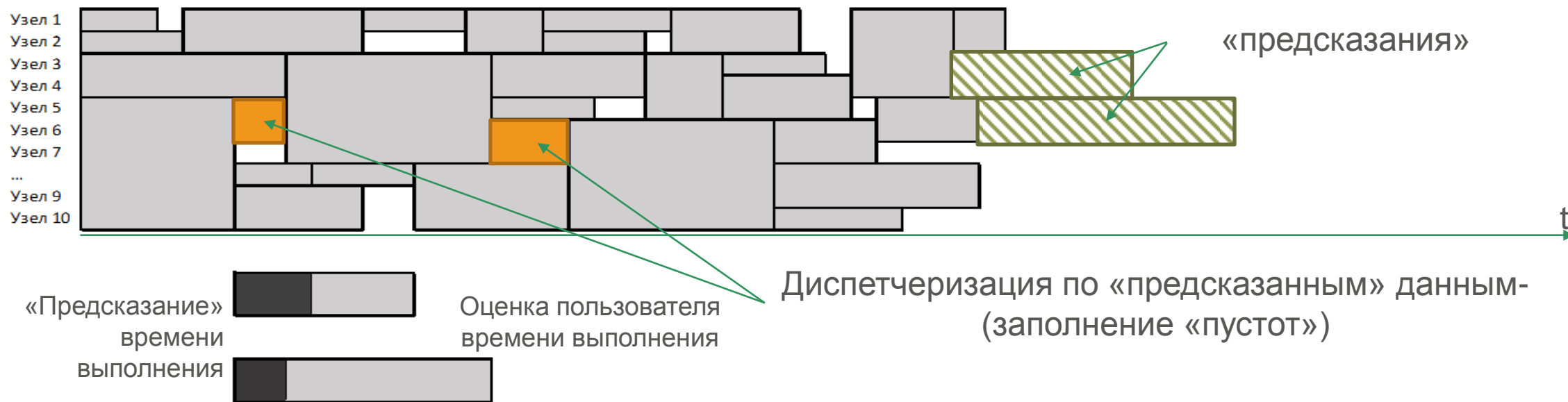
## Описание проблемы:

Существующие методы планирования загрузки имеют **ограниченные возможности предсказания** времени исполнения заданий, поэтому статус «очереди заданий» необходимо постоянно корректировать.

## Предлагаемое решение:

Выделение «классов эквивалентности» исполняемых заданий и построение **модели прогнозирования времени решений** на заданном наборе аппаратных ресурсов с помощью алгоритма, параметры которого **уточняются с помощью методов машинного обучения....**

# ГДЕ РЕЗЕРВЫ УВЕЛИЧЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ СК



Задача – предсказывать время выполнения задач пользователей и составлять расписание на основании этого предсказания

Что для предсказания требуется :

- сбор статистических данных, характеризующих процесс выполнения заданий
- анализа факторов риска «выживания» заданий
- реализация механизма «внимания» к параметрам заявки пользователей – суррогатная модель доверия к данным заявки

**Одновременное использование этих** двух подходов может реально повысить эффективность применения суперкомпьютерных систем, так как «обучения» реализовывать наиболее эффективные классы алгоритмов планирования вычислительной нагрузки, так, чтобы:

**последовательно** помещая в очередь «исполняемых задач» наибольшее количество задний пользователей, контролируя характеристики загрузки аппаратных ресурсов

**и**

**одновременно** корректируя параметры заявок пользователей в очереди «ожидания» исполнения, в частности, формируя оценку времени выполнения заданий, обеспечивая их «выживание» в процессе обработки

Гипотеза : «**Больше обучающих данных – выше точность модели**». Это так, если *обучающим, тестовым и реальным выборкам соответствует одна функция распределения  $P(X,y)$ , где  $X$  – входные переменные,  $y$  – целевые переменные*. Если это не так, то возникают:

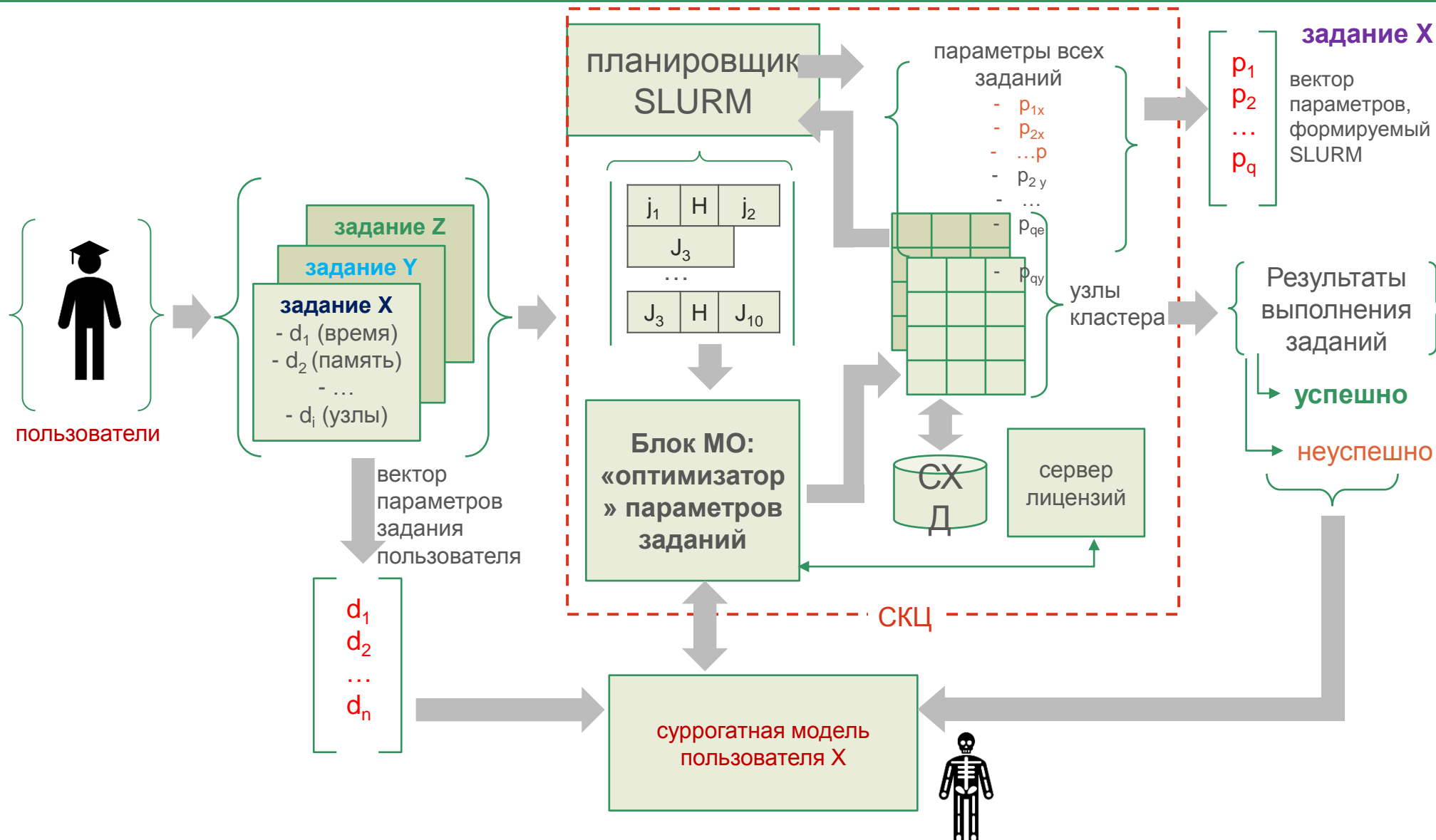
- **Проблема** «Утечка данных»: ситуация, когда существует признак, который содержит больше информации о целевой переменной, чем  $X$  используемые на практике
- **Проблема** Shortcut learning: ситуация, когда обученные модели получают верный ответ с помощью неверных в общем случае рассуждений ("right for the wrong reasons"), которые работают только для «обучающего» распределения  $P(X,y)$  данных, а для real-world scenarios дают ошибочные результаты.
- **Проблема** Overfitting features: Обученная модель использует признаки, которые позволяют эффективно предсказывать ответ на обучающей выборке, но не на всех реализациях, из которого получена выборка  $X$ .

**Решения** – ведение в процесс обучения функции «**внимания**» и «**объяснения**», исходя из того, что

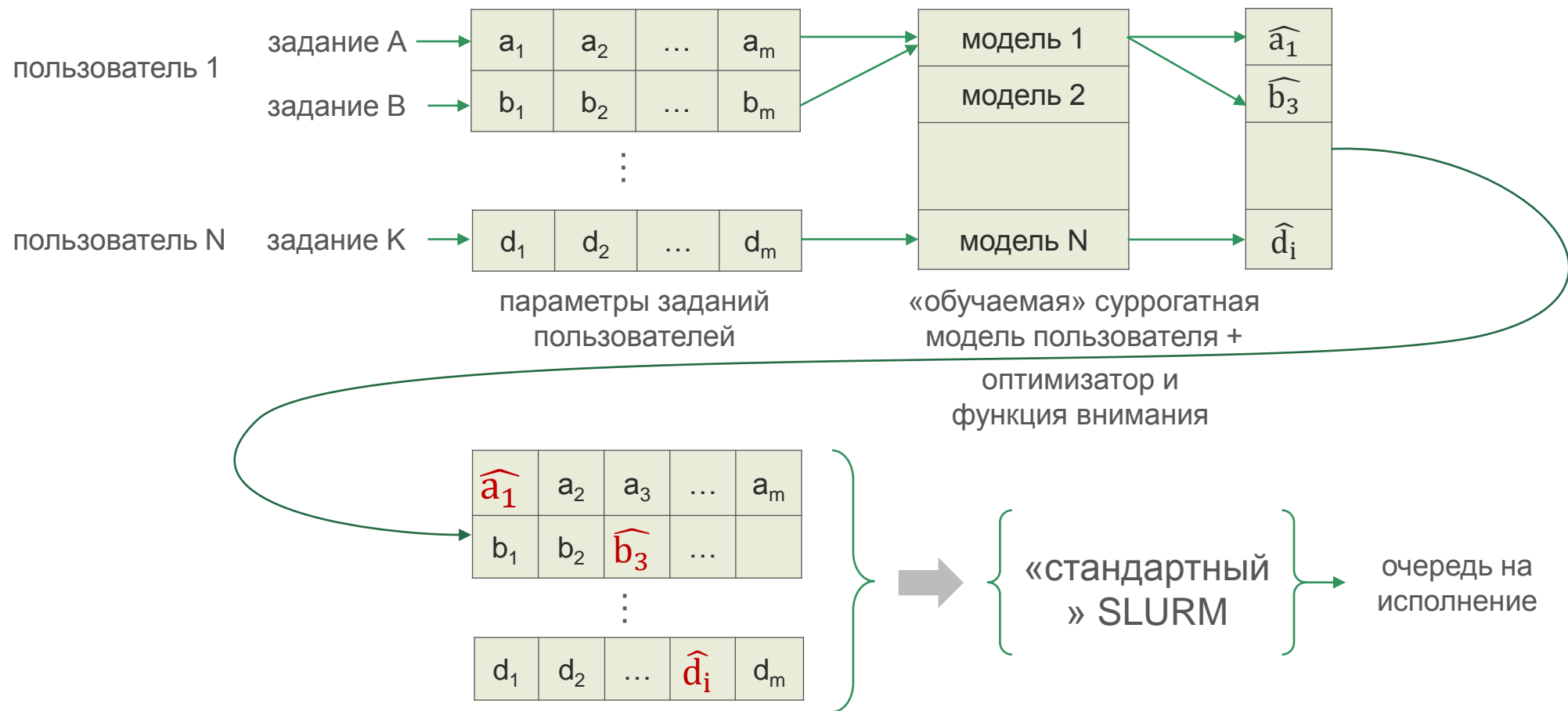
- конечный набор данных имеет **ограниченное разнообразие**, не «покрывая» всех возможных ситуаций,
- в данных  $X$  могут существовать **«паразитные корреляции» не имеющие причинно-следственных оснований**, которые характерны для моделируемых процессов



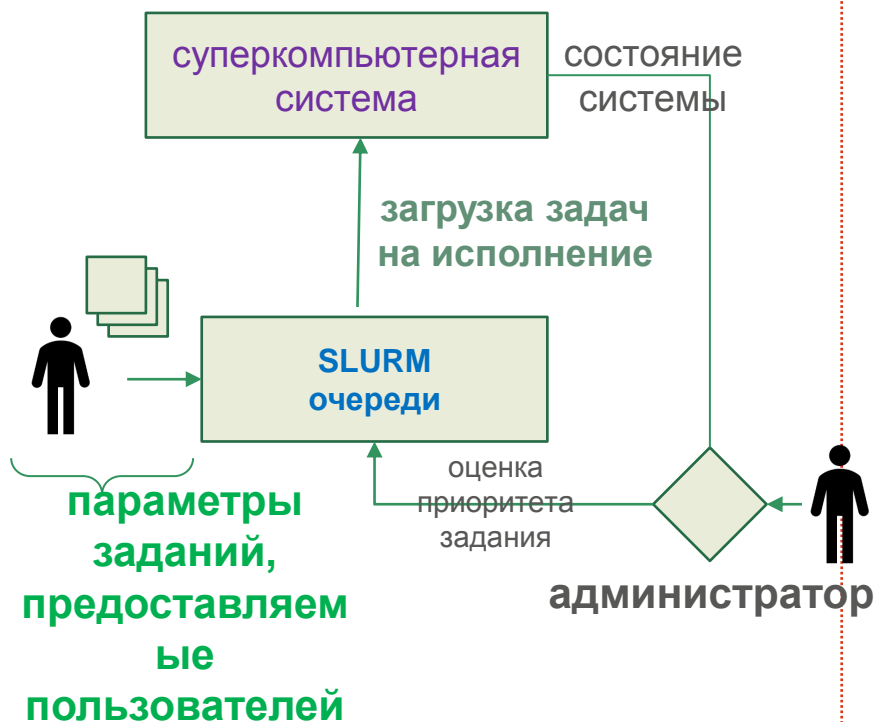
# ВАРИАНТ РЕШЕНИЯ “MORE BRAIN” С ИСПОЛЬЗОВАНИЕМ БЛОКА «ОПТИМАЛЬНОГО» ПРЕДСКАЗАНИЯ ПАРАМЕТРОВ ЗАДАНИЙ ПОЛЬЗОВАТЕЛЕЙ



# ПРОЦЕССНОЕ ОПИСАНИЕ ОЧЕРЕДИ ЗАДАНИЙ В БЛОКЕ «ОПТИМИЗАЦИИ»



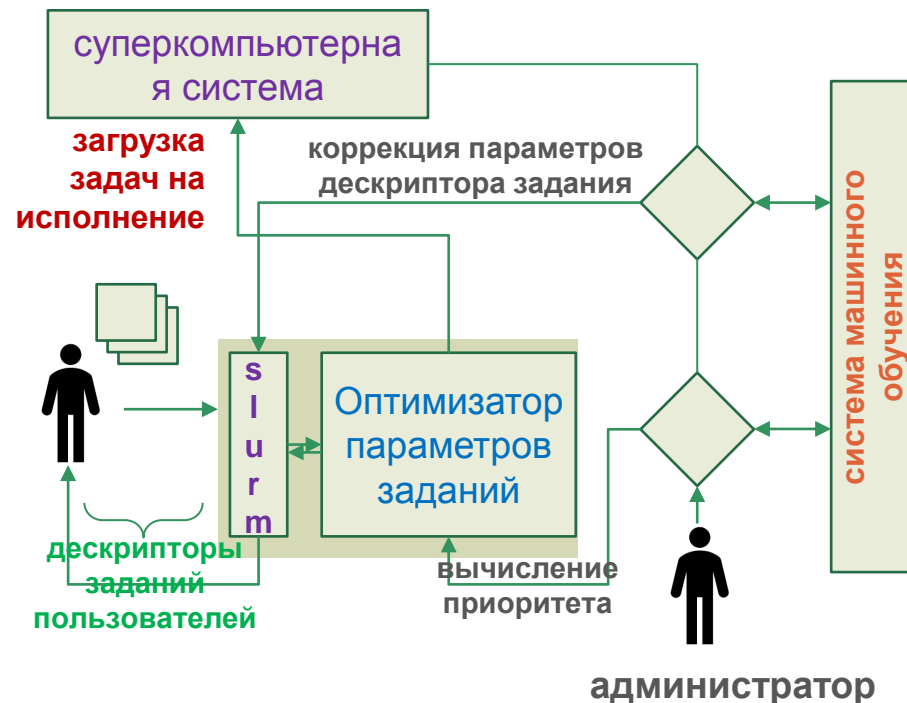
## Что есть сейчас



Проблемы:

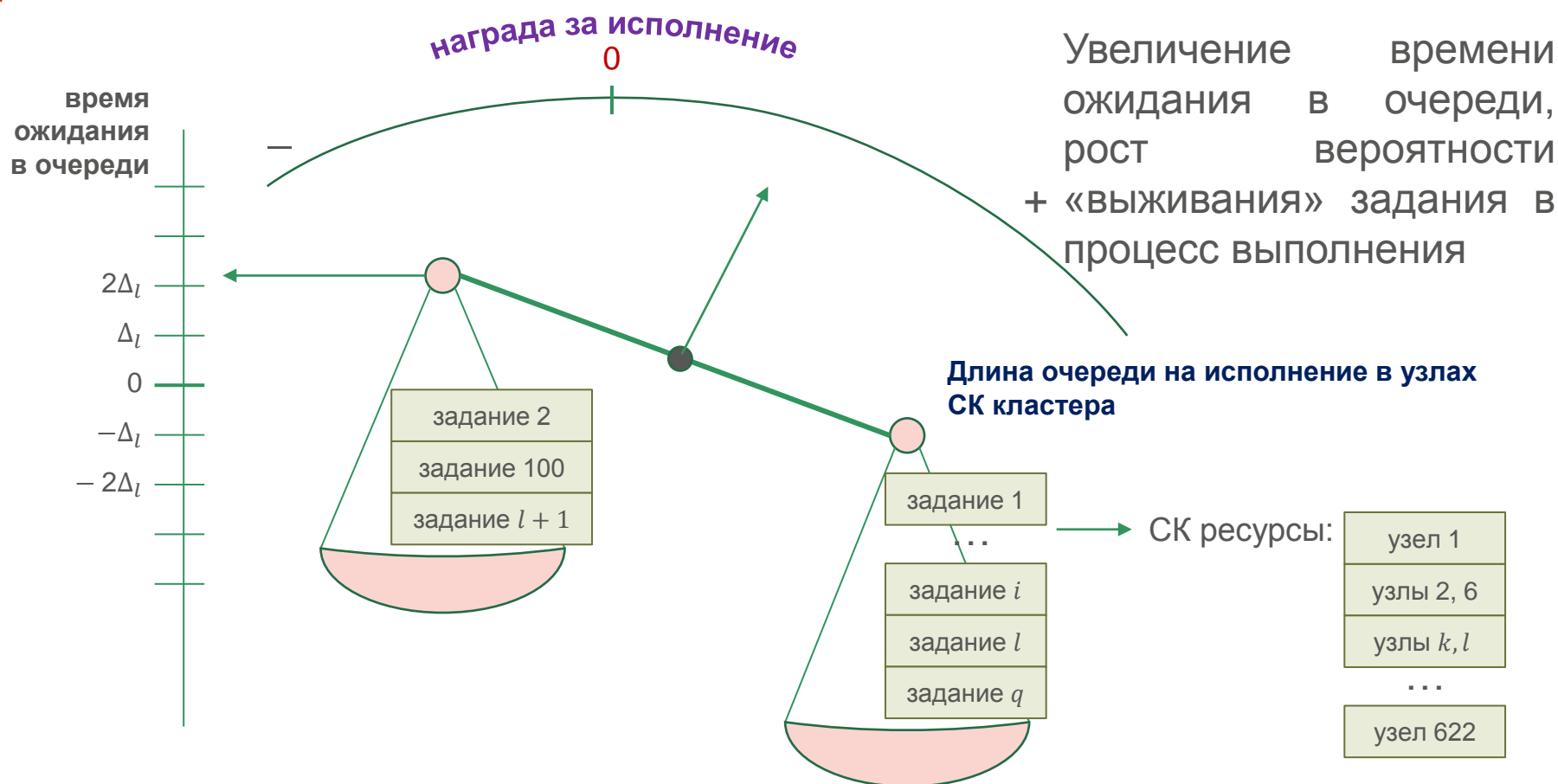
- неточность в параметрах заданий
- «ошибки» slurm в оценке времени исполнения заданий

## Что предлагается сделать



Создание блока МО для «оптимизации» параметров заданий, которые использует система slurm

# «БАЛАНС» АЛГОРИТМ МАШИННОГО ОБУЧЕНИЯ: ВЫБОР КРИТЕРИЕВ



Алгоритмы балансирования очереди могут меняться в зависимости критериев

- 1) min среднего времени «простоя» (ожидания)
- 2) min «время выполнения» задания
- 3) min время ожидания + время выполнения



# ИСХОДНЫЕ ДАННЫЕ ДЛЯ СИСТЕМЫ ОБУЧЕНИЯ

Пользователь x

Пользователь y

.....

Пользователь z

Пользователь q

параметры номер задания	1	2	...	m
1	$a_1^1$	$a_2^1$		$a_m^1$
2	$a_1^2$	$a_2^2$		$a_m^2$
...				
k - 1	$a_1^{k-1}$	$a_2^{k-1}$		$a_m^{k-1}$
k	$a_1^k$	$a_2^k$		$a_m^k$



диспетчер задач  
SLURM

критерий:  
точность прогноза  
времени решения

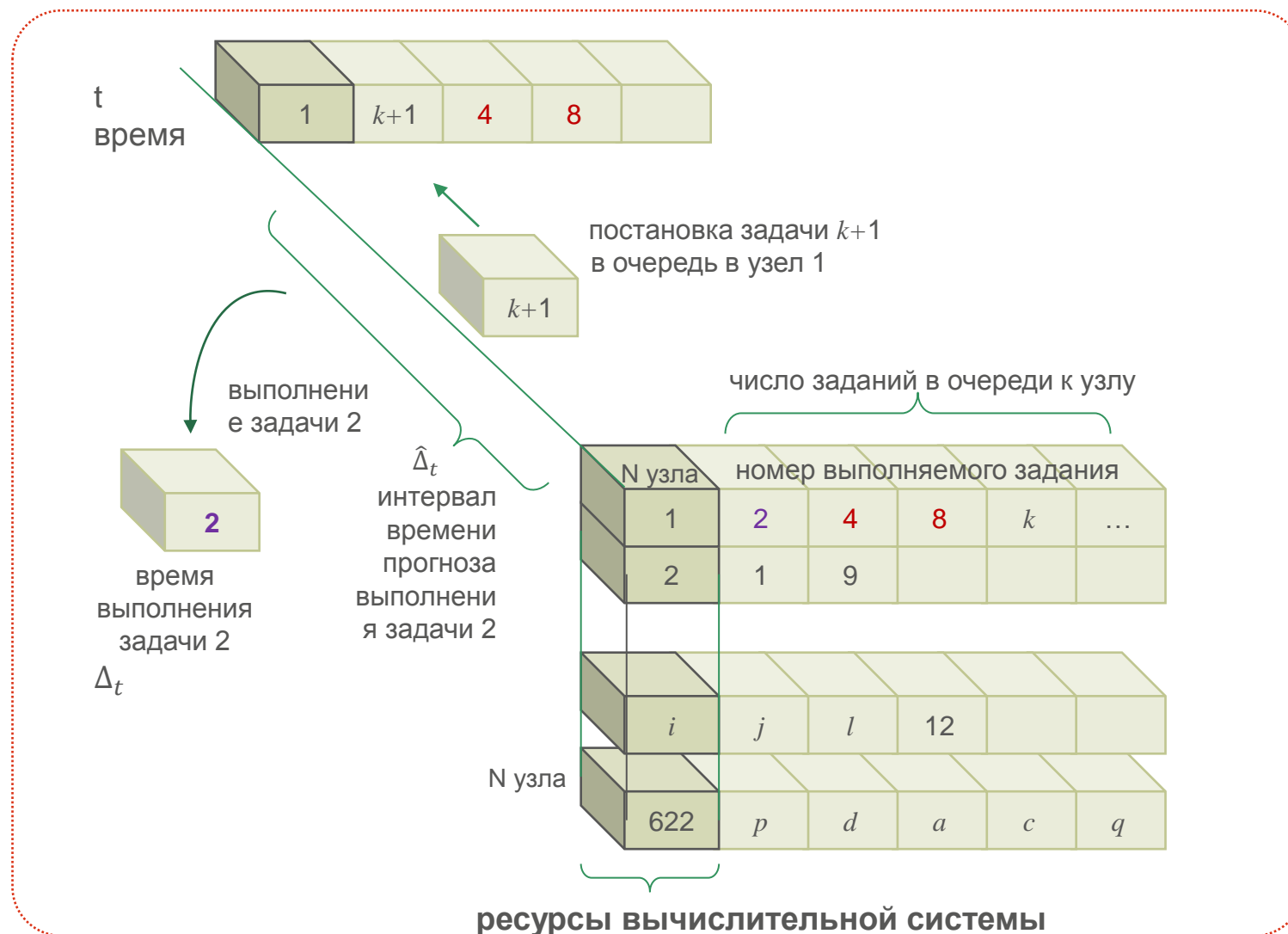
Значение  
параметра,  
выделенного  
функцией  
внимания, после  
завершения  
задания

$$J = \sum_{i=1}^N \|\Delta_t - \hat{\Delta}_t\| \rightarrow \min_{a_i}$$



параметры заданий пользователей, учитываемые в планировщике slurm

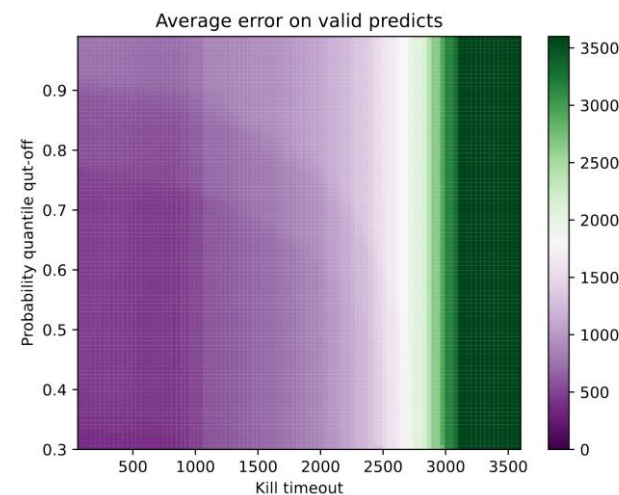
# ДИНАМИКА ПРОЦЕССА УПРАВЛЕНИЯ ОЧЕРЕДЬЮ ЗАДАНИЙ В ПРОСТРАНСТВЕ СОСТОЯНИЙ «ВРЕМЯ ИСПОЛНЕНИЯ – АППАРАТНЫЕ РЕСУРСЫ» СК



И . . .

[illegible]

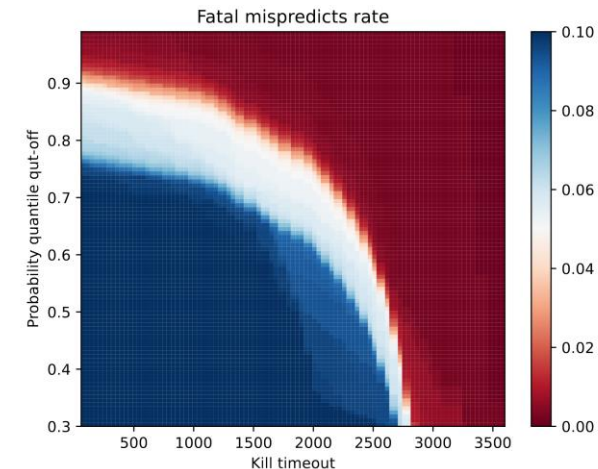
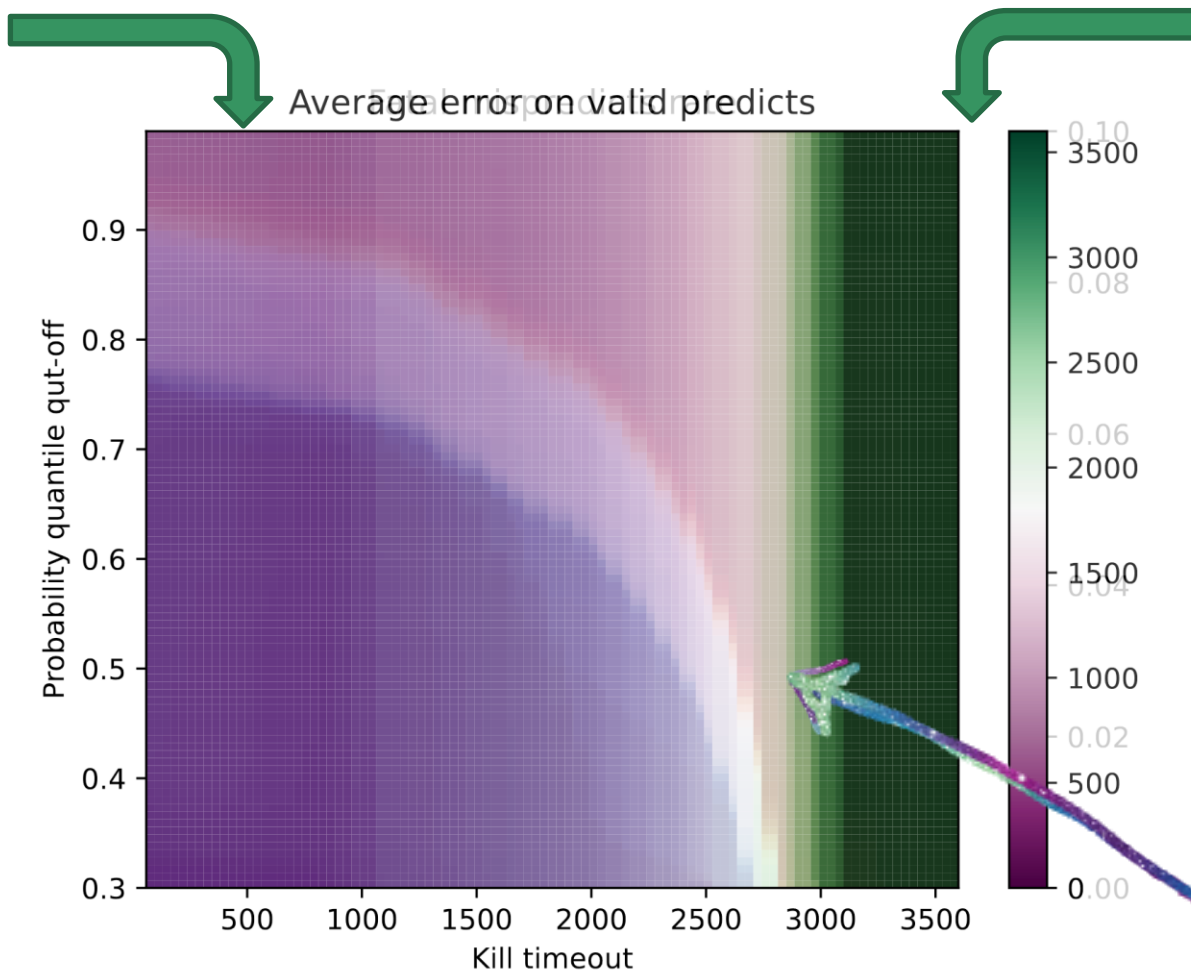
# «УСТОЙЧИВОЕ НЕРАВНОВЕСИЕ» ОШИБОК ПРЕДСКАЗАНИЙ



Комментарий:

Слева – средняя ошибка  
«**правильных**»  
предсказаний (время  
равно или больше  
фактического)

Справа – доля  
«**неправильных**»  
предсказаний (с  
занижением времени  
выполнения)



Комментарий:

В качестве критерия  
«выживаемости» заданий  
нужна функция «засорения»  
множества «правильных»  
множеством «неправильных»  
ошибок

Такая область на среднем  
рисунке имеет «розовый  
цвет» (выделил красным  
контуром)

Требуется динамическая корректировка параметров приоритета заданий

1. Существующие методы машинного обучения СК **не стоит переоценивать**, однако их «корректное» использование позволяет реально повысить эффективность работы СК платформы, если в систему управления исполнения заданий включить «обучаемую» суррогатную модель пользователя.
2. Задача **«обучения» СК точному планированию времени** поддается эффективному решению, но требует встраивания в контур исполнения заданий «умного» блока (с точки зрения выбранной политики выполнения заданий) оценки корректности данных, представленных пользователем.
3. Так как пользователь непрерывно взаимодействует СК платформой и постоянно совершенствует свой собственный опыт, основываясь на результатах вычислений с учетом значений выбранных параметров заданий, то суррогатная модель должна носить объяснительный характер и учитывать фактор «внимания» (мульти-внимания) по отношению к вектору параметров исполняемого задания
4. Эффективная работы «умного» блока требует постоянного уточнения «суррогатной модели пользователей» СК платформы, поэтому для использования «умного» блока оценки времени исполнения для других СК платформ необходимо проводить процесс «дообучения» моделей с учетом конкретного разнообразия классов решаемых прикладных задач