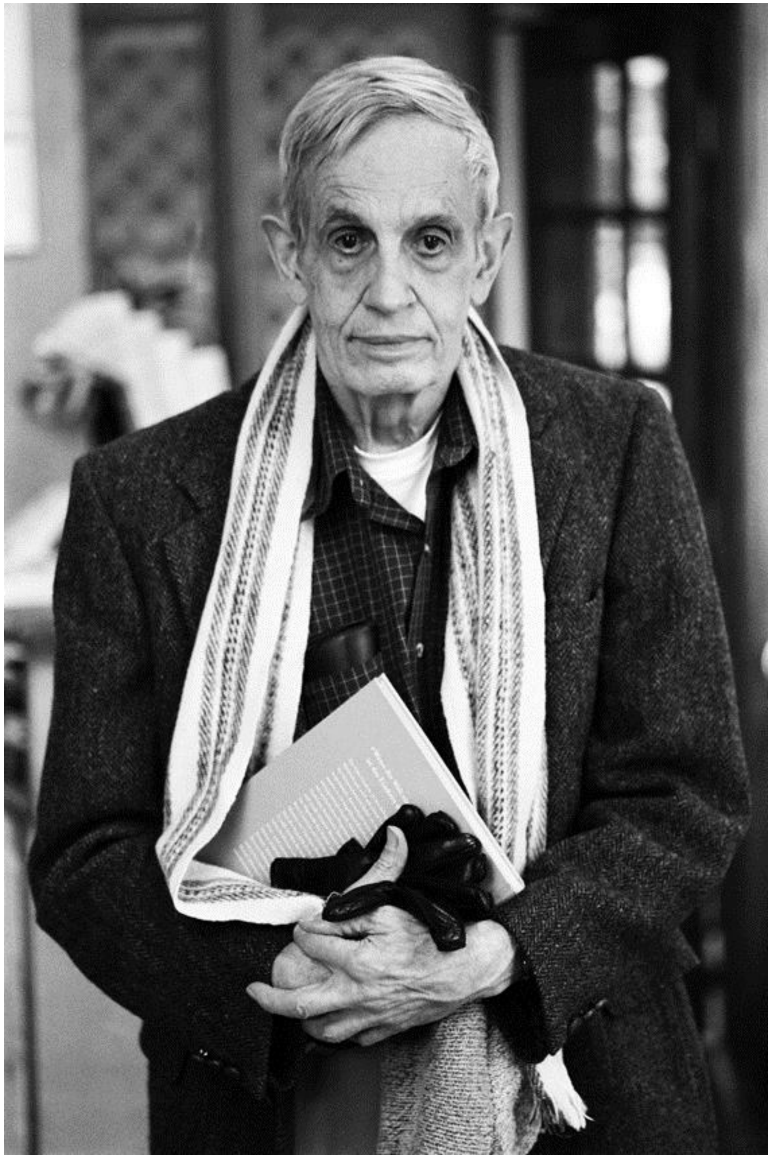
Лекция 5. Режим обслуживания потока задач

Перышкова Евгения Николаевна
к.т.н., доцент
Кафедра вычислительных систем
Сибирский государственный университет
телекоммуникаций и информатики

Теоретико-игровой подход к проблеме организации функционирования ВС

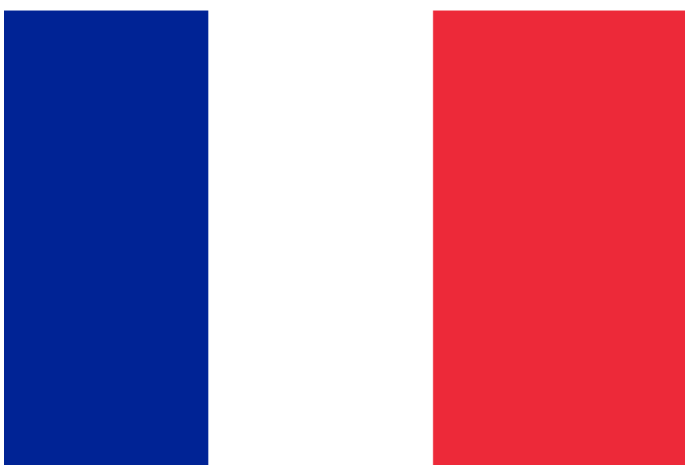


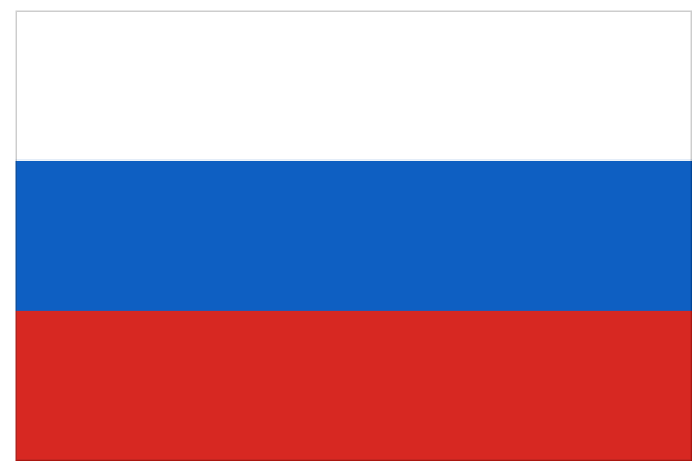
Джон Нэш (John Nash; род.1928)

- Американский математик, работающий в области теории игр и дифференциальной геометрии.
- Лауреат Нобелевской премии по экономике 1994 года «За анализ равновесия в теории некооперативных игр».
- Известен широкой публике большей частью по биографической драме Рона Ховарда «Игры разума» (англ. A Beautiful Mind) о его математическом гении и борьбе с шизофренией.

Теоретико-игровой подход к проблеме организации функционирования ВС

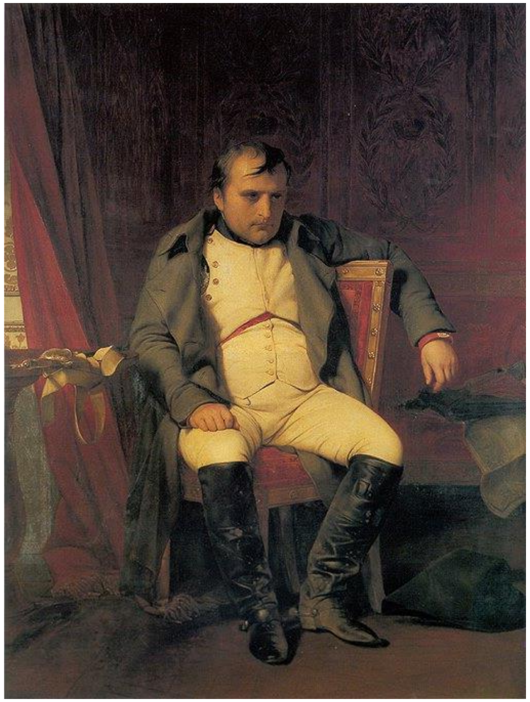
- В теории игр рассматриваются ситуации, связанные с принятием решений, в которых **два противника имеют конфликтующие цели**.

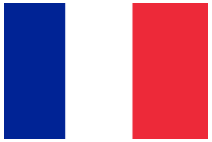




Теоретико-игровой подход к проблеме организации функционирования ВС

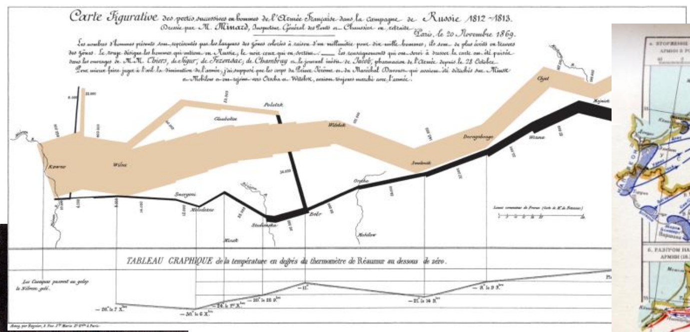
- В игровом конфликте участвуют два противника, именуемые **игроками**, каждый из которых имеет множество возможных выборов, которые называются стратегиями

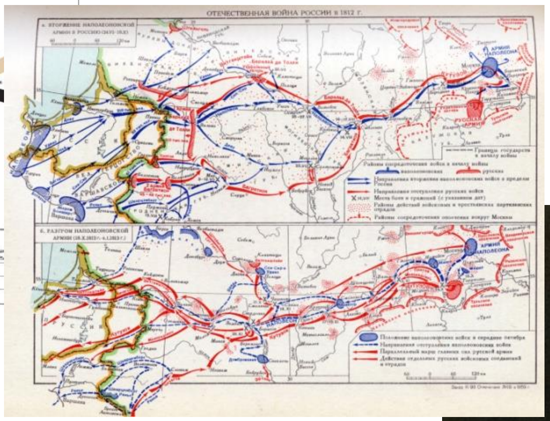


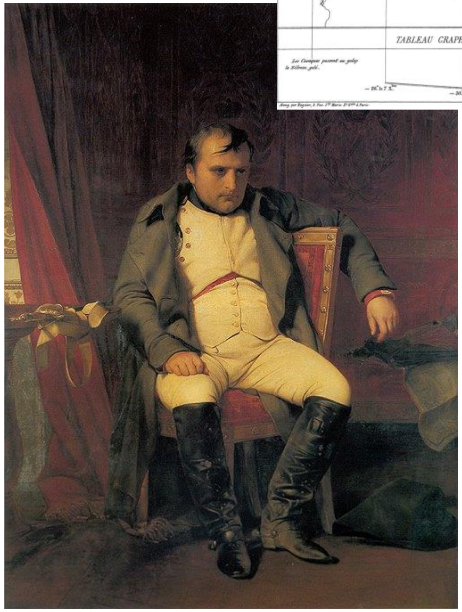


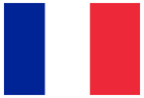
Теоретико-игровой подход к проблеме организации функционирования ВС

- Каждый из игроков имеет некоторое множество возможных выборов, которые называются **стратегиями**.

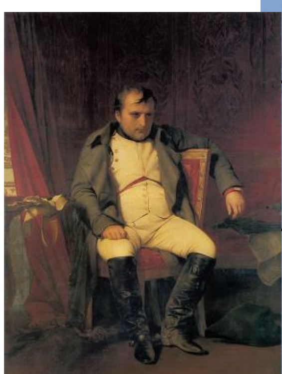








Теоретико-игровой подход к проблеме организации функционирования ВС



	B_1	B_2	…	B_n
A_1	a_{11}	a_{12}	…	a_{1n}
A_2	a_{21}	a_{22}	…	a_{2n}
…	…	…	…	…
A_m	a_{m1}	a_{m2}	…	a_{mn}

- С каждой парой стратегий связан **платёж**, который один из игроков выплачивает другому.
- Такие игры известны как **игры двух лиц с нулевой суммой**.

	B_1	B_2	…	B_n
A_1	a_{11}	a_{12}	…	a_{1n}
A_2	a_{21}	a_{22}	…	a_{2n}
…	…	…	…	…
A_m	a_{m1}	a_{m2}	…	a_{mn}

Если игрок *A* использует стратегию i, а игрок *B* – стратегию j, то:

- платёж игроку *A* составляет a_{ij},
- платёж игроку *B* составляет $-a_{ij}$

Теоретико-игровой подход к проблеме организации функционирования ВС

Что является оптимальным решением игры?

- **Оптимальным решением игры** является одна или несколько стратегий для каждого из игроков, при котором любое отклонение от данных стратегий не улучшает плату тому или другому игроку.
- Эти решения могут быть в виде единственной **чистой** стратегии
- или нескольких стратегий, которые являются **смешанными** в соответствии с заданными вероятностями.

Решение в чистых стратегиях

	B_1	B_2	B_3	B_4	Минимумы строк	
A_1	8	-2	9	-3	-3	
A_2	6	**5**	6	8	**5**	*максимин*
A_3	-2	4	-9	5	-9	
Максимумы столбцов	8	**5**	9	8		
		минимакс				

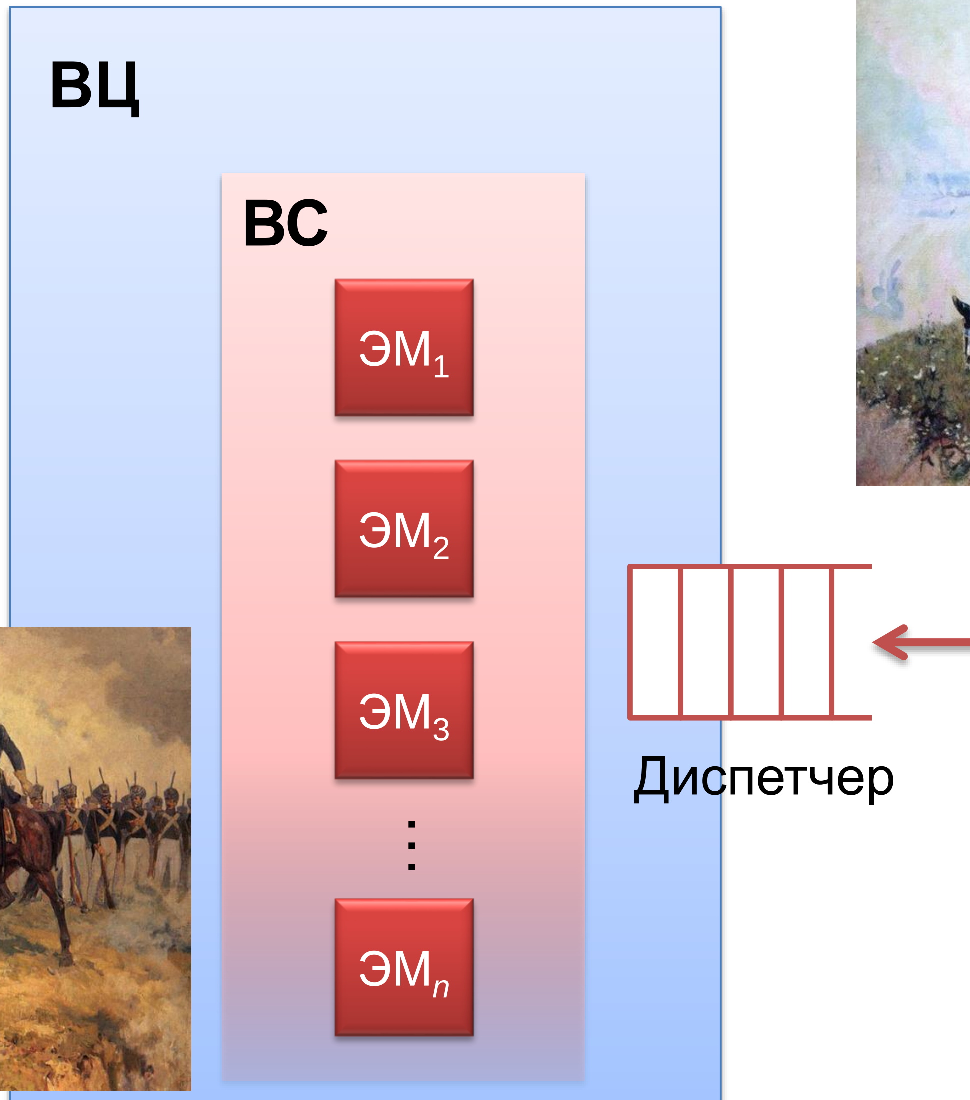
Теоретико-игровой подход к проблеме организации функционирования ВС

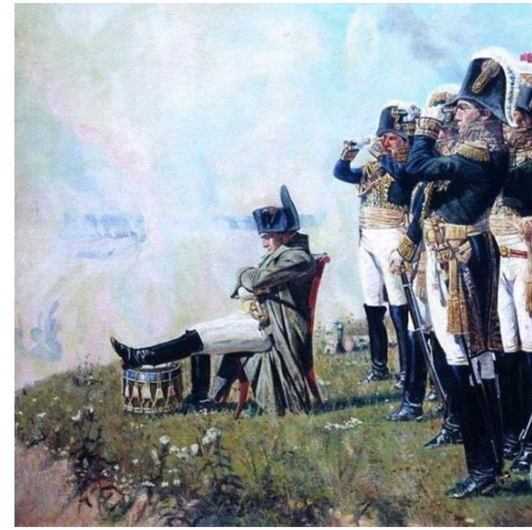
- Решение может быть основано на обеспечении **наилучшего результата из наихудших** для каждого игрока.
- Критерий наилучшего результата из наихудших соответствует выбору **минимаксному значению**.
- Оптимальным решением в игре является выбор стратегий A_2 и B_2.
- При этом цена игры составляет 5 и игроки A и B используют стратегии, соответствующие **седловой точке**.

Решение в смешанных стратегиях

	B_1	B_2	Минимумы строк
A_1	1	-1	-1
A_2	-1	1	-1
Максимумы столбцов	1	1	

Теоретико-игровой подход к проблеме организации функционирования ВС





Обслуживание потока задач

Дано: ВЦ, на котором эксплуатируется ВС из n ЭМ, и диспетчер.

Требуется выделять вычислительные ресурсы для задач.

Задача 1. [Евреинов, Хорошейвский, с. 185]

Имеется ВЦ, эксплуатирующий ВС из n ЭМ

ВЦ для решения задач может выставлять любое число ЭМ

$i \in E = \{0, 1 , \ldots, n\}$

На ВЦ поступает поток задач различных рангов

Считается, что интенсивность такая, что имеются задачи всех рангов.

Считается, что интенсивность такая, что имеются задачи всех рангов.

Считаем, что построены пакеты задач r_j'

Имеется диспетчер, который в дискретные моменты времени $t = 0, 1, 2, \ldots$ назначает на ВС задачи с различным рангом

Имеется игра с участием двух игроков: ВЦ и диспетчера

ВЦ использует *чистую стратегию* $i \in E$, если для решения задач отводит i ЭМ

Диспетчер использует *чистую стратегию* j, если для решения он назначает задачу ранга j.

Теоретико-игровой подход к проблеме организации функционирования ВС

Если ВЦ выбирает стратегию i, а диспетчер стратегию j, то диспетчер “платит” ВЦ сумму c_{ij}, $c = \| c_{ij} \|$ - матрица платежей.

Если ВЦ применяет смешанную стратегию $P = \{p_0, p_1,\ldots, p_n\}$, а диспетчер – смешанную стратегию $\Pi = \{\pi_0, \pi_1,\ldots, \pi_n\}$, то средний платёж вычислительному центру составит

$$\sum_{i=0}^{n}\sum_{j=0}^{n} c_{ij}\, p_i \pi_j$$

p_i и π_i – вероятности выбора соответственно вычислительным центром стратегии с номером i и диспетчером стратегии с номером j.

Теоретико-игровой подход к проблеме организации функционирования ВС

ВЦ имеет оптимальную смешанную стратегию $P^* = \{p_0{}^*, p_1{}^*,\ldots, p_n{}^*\}$, а диспетчер – оптимальную смешанную стратегию $\Pi^* = \{\pi_0{}^*, \pi_1{}^*,\ldots, \pi_n{}^*\}$ такие что

$$\sum_{i=0}^{n}\sum_{j=0}^{n} c_{ij}\, p_i^* \pi_j \geq V \geq \sum_{i=0}^{n}\sum_{j=0}^{n} c_{ij}\, p_i \pi_j^*$$

$$V = \sum_{i=0}^{n}\sum_{j=0}^{n} c_{ij}\, p_i^* \pi_j^*$$

Требуется для заданной матрицы платежей найти решение $\{P^*, \Pi^*\}$ и *цену* V игры

Подбор элементов платёжной матрицы

Будем считать, что если i, j – чистые стратегии соответственно ВЦ и диспетчера, то элементы матрицы платежей

$$c_{ij} = \begin{cases} jc_1 + (i-j)c_2 & при \quad i \geq j, \\ jc_1 + (j-i)c_3 & при \quad i < j, \end{cases}$$

c_1 – платёж за использование одной машины в течение единицы времени

c_2 – штрафы в единицу времени за простой одной машины $(i-j)$ машин

c_3 – штраф за недостающие машины $(j-i)$ машин

Подбор элементов платёжной матрицы

Наиболее вероятно, что в на ВЦ будут поступать задачи всех рангов. Поэтому алгоритм функционирования ВС, состоящий в назначении задач одного ранга, представляется неэффективным. Следовательно, игра не должна иметь решения в чистых стратегиях, т.е. матрица $\|c_{ij}\|$ **не должна иметь седловых точек**.

Теорема

Теорема. Матрица c_{ij} не имеет седловых точек тогда и только тогда, когда $c_1 < \min\{c_2, c_3\}$

Теорема

Доказательство.

<u>*Необходимость*</u> Пусть матрица $||c_{ij}||$ не имеет седловых точек. Из определения c_{ij} следует, что максимальным элементом 0-столбца является c_{n0}. Так как по условию $||c_{ij}||$ не имеет седловых точек, то c_{n0} не должен быть минимальным в своей строке. Минимум в последней строке должен достигаться в c_{nn}. Значит, $c_{nn} < c_{n0}$, но так как $c_{n0} = nc_2$, а $c_{nn} = nc_1$, то $c_1 < c_2$

Аналогично рассуждая для 0-строки и n-столбца, получим $c_1 < c_3$. Необходимость доказана.

Теорема

Достаточность

Если $c_1 < \min\{c_2, c_3\}$, то легко увидеть, что диагональные элементы являются минимальными в своей строке и в своём столбце, причём минимум строгий. Следовательно, матрица не имеет седловых точек.

■

Стохастический алгоритм формирования пакетов задач

Имеется ВС из n ЭМ и множество $J = \{1, 2, \ldots, m\}$ задач.

Задачи характеризуются рангом r_j и временем t_j решения.

Требуется множество всех задач ранга r, $i \le r \le n$, разбить на подмножества так, чтобы в каждое из них входили задачи, суммарное время решение которых было близко к заданному значению Θ.

Пусть подмножества $J^r \subseteq J$, $r = 1,\ldots,n$ – такие, что

в J^r входят все задачи $I_i^r \in J$, $i = 1, 2, \ldots, a$, которые

имеют ранг r. Время решения этих задач

$$T_r = \sum_{i=1}^{a} t_i^r$$

Стохастический алгоритм формирования пакетов задач

Пусть также $J' = \{I_i^r, I_2^r, \ldots, I_i^r, \ldots, I_a^r\}$ – некоторая последовательность, членами которой являются элементы J_i^r. Подмножество $J_j \subseteq J'$ включает в себя k_j задач, причём

$$J_j = \bigcup_{i=K_j+1}^{K_j+k_j} j_i^r,$$

где

$$K_j = \sum_{s=0}^{j-1} k_s, \quad k_0 = 0, \quad j = 1,2,\ldots,L_r, \quad L_r =]T_r / \Theta[$$

(L_r – ближайшее к T_r / Θ целое число, $L_r \geq T_r / \Theta$)

Стохастический алгоритм формирования пакетов задач

Каждое подмножество $J_j \subset J'$, $j = 1, 2, \ldots, L_r$, назовём укрупнённой задачей b_j ранга r. Время решения таких задач

$$T_j = \sum_{i=K_j+1}^{K_j+k_j} t_i^r$$

Будем считать, что для пакет укрупнённых задач ранга r сформирован, если справедливо равенство

$$|T' - \Theta| = o(T'),$$

где $T' = \max_{J_j \subset J} \{Tj\}$

Стохастический алгоритм формирования пакетов задач

Подмножества $J_j \subset J'$ выбираются следующим образом.

Пусть построены подмножества $J_s \subset J'$, $s = 1, 2, \ldots, j - 1$, тогда подмножества

$$J_j = \begin{cases} J''_j, k_j = k'_j, \text{если } (\Theta_u)_j > [T_j - (\Theta_b)_j](L_r - j)^{-1} \\ J''_j \cup I_{K_j + k'_j + 1}, k_j = k'_j + 1, \text{в противном случае} \end{cases}$$

где J''_j – часть последовательности J' такая, что

$$J'_j = \{I_{K_j+1}, I_{K_j+2}, \ldots, I_{Kj+k'_j}\}, \quad T^j = \sum_{i=K_j+1}^{L_r} t_i^r$$

а для величин $(\Theta_u)_j$ и $(\Theta_b)_j$ справедливы соотношения

$$\sum_{i=K_j+1}^{K_j+k'_j+1} t_i^r = (\Theta_u)_j > \Theta, \quad \sum_{i=K_j+1}^{K_j+k'_j} t_i^r = (\Theta_b)_j \le \Theta$$

Требуется найти такую последовательность J'^* задач, которая обеспечит выполнение условия

$$|T' - \Theta| = o(T') \qquad (*)$$

Последовательность J'^* отыскивается с помощью **метода цепей Монте-Карло**

Стохастический алгоритм формирования пакетов задач

1. Последовательность J' принимается за базовую.
2. Рассматриваем перестановки, находящиеся на расстоянии не больше k, $k \leq a$, от базовой. В качестве *расстояния* между двумя последовательностями J' и J'' принимают число индексов в J'', которые не следуют за теми же индексами, что и в базовой J'.

Стохастический алгоритм формирования пакетов задач

3. Сначала получаем $k - 1$ переменных x_i в непрерывном интервале $(0, a)$: $x_i = a\xi_I$, $i = 1, 2, \ldots, k-1$, где $\xi_i \in U(0, 1)$.

4. Если $0 = x_0 \leq x_1 \leq \ldots \leq x_i \leq x_k = a$, то x_i делят последовательность J' на k частей $J^s \subseteq J$, $s = 1, 2, \ldots, k$, содержащих задачи, номера которых являются целыми числами в $(x_{i-1}, x_i]$. Некоторые части могут оказаться пустыми. Случайная перестановка этих частей даёт новую последовательность J'' с расстоянием не больше k от базовой.

5. Если T'' для новой последовательности J'' меньше T, т.е. наилучшим образом удовлетворяет (*), то J'' берётся в качестве базовой.

Стохастический алгоритм формирования пакетов задач

6. Если сделано d попыток моделирования последовательностей с расстоянием k от базовой без изменения базовой, то рассматриваются последовательности с расстоянием $k / 2$ и так далее до тех пор, пока не будут смоделированы последовательности расстоянием 2.