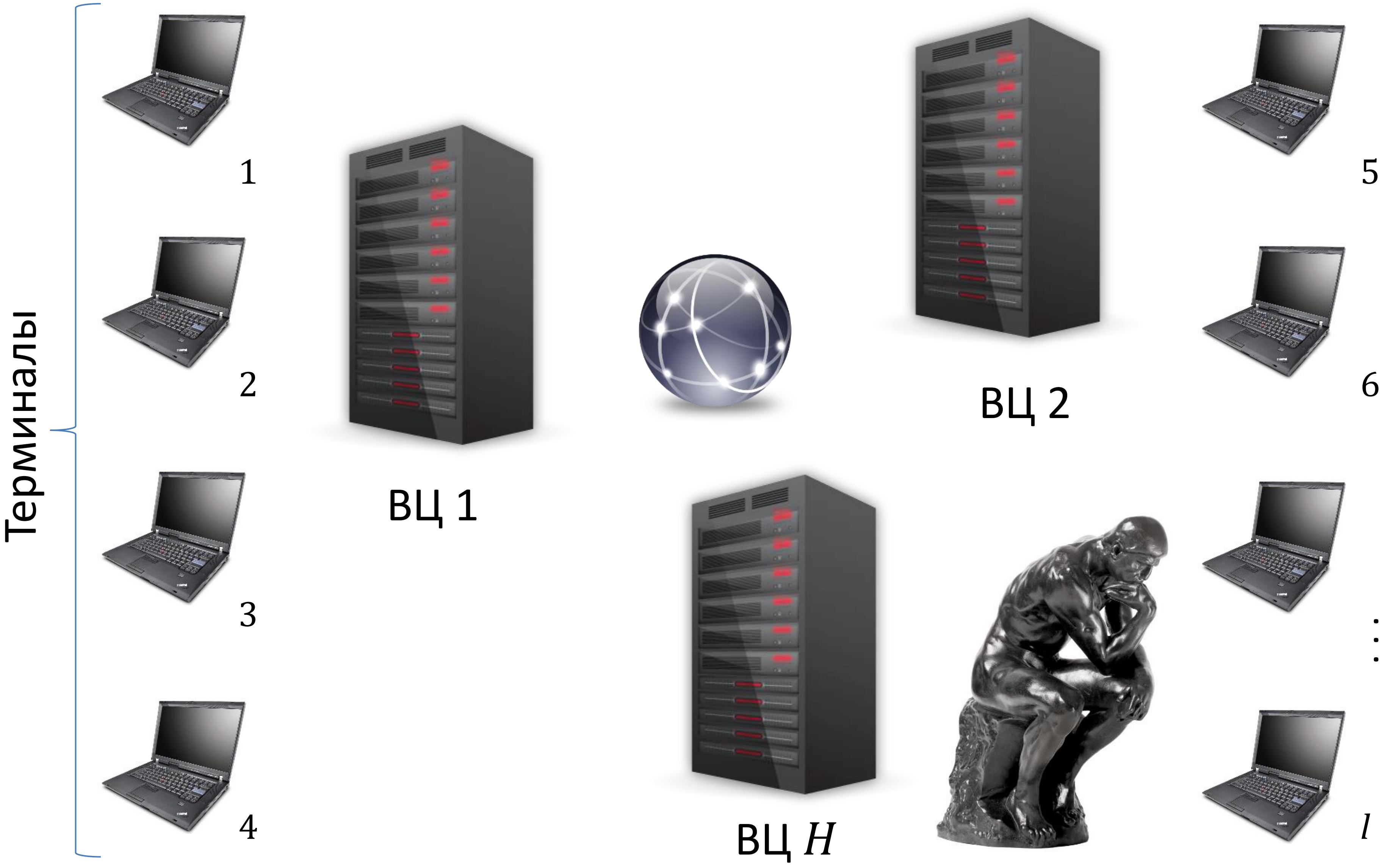
Лекция 9. Организация функционирования распределённых ВС с привлечением аппарата стохастического программирования

Пазников Алексей Александрович

Ассистент Кафедры вычислительных систем
Сибирский государственный университет
телекоммуникаций и информатики

http://cpct.sibsutis.ru/~apaznikov

Задача V



Задача V

Имеется распределённая ВС:

- n – количество элементарных машин (ЭМ)
- H – число вычислительных центров (ВЦ)
- n_i – число ЭМ на ВЦ i, $i \in \{1,2, \ldots, H\}$
- l – количество терминалов,
- a_j – ранг подсистемы, требуемой на терминале j (число требуемых машин),
- $p_j(a_j)$ – распределение вероятностей a_j
- x_j – число используемых машин на терминале j (число используемых машин)

Задача V

- e_{ij} – стоимость использования с терминала j одной ЭМ i-го ВЦ
- d_j – средняя цена эксплуатации,
- c_j – средняя стоимость эксплуатации одной ЭМ для терминала j;

Для терминала j м.о. потерь при использование подсистем с рангом менее требуемого:

$$(dj - cj) \int_{x_j}^{\infty} (a_j - xj) p_j(a_j) da_j$$

При использование подсистем с рангом более требуемого:

$$c_j(x_j - \rho_j) + c_j \int_{x_j}^{\infty} (a_j - xj) p_j(a_j) da_j$$

где ρ_j – м.о. требуемого ранга подсистемы для терминала j

А что минимизируем?

Обозначим χ_{ij} – число машин ВЦ i, которое используется на терминале j. Сформулируем задачу

$$\sum_{j=1}^{l}\left(\sum_{i=1}^{H} e_{ij}\chi_{ij} - cj\rho_j + cjxj + d\int_{x_j}^{\infty}(a_j - xj)\, p_j(a_j)da_j\right) \to \min_{x_{ij}},$$

при ограничениях

$$\sum_{j=1}^{l}\chi_{ij} \le ni, i = 1, 2, \dots, H, \qquad \sum_{i=1}^{l} n_i = n$$

$$\sum_{j=1}^{l}\chi_{ij} = xj,\ \chi \ge 0, xj > 0, j = 1,2, \dots, l$$

Задача V

И это ещё не всё…

Задача IV

- На терминалы поступают потоки задач различных рангов.
- Спрос с каждого терминала $j \in \{1,\ 2,\ \ldots,\ l\}$ подчинён пуассоновскому распределению.
- ρ_j и ρ_j^* – среднее значение спроса при минимальном и максимальном требуемом ранге.
- c_j – стоимость использования одной ЭМ с терминала j.
- $||c_{ik}||$ – стоимости каналов между i-м и k-м ВЦ; $i, j \in 1,2, \ldots, H$

Задача IV

Пусть

- x_j – ранг подсистемы, назначенной терминалу j, $j \in \{1, 2, \dots, l\}$
- a_j и a_j^* – минимальный и максимальный ранги подсистем, которая потребуется на терминале j.
- χ_{ij} – число ЭМ i-го ВЦ, используемых терминалом j
- Введём функцию $\delta(\chi_{ij})$

$$\delta = \begin{cases} 1, \text{если } \chi_{ij} > 0 \\ 0, \text{если } \chi_{ij} = 0 \end{cases}$$

Задача IV

Задача, которую необходимо решить:

$$\min_{x_j} z = \sum_{j=1}^{l} \left\{ \sum_{i=1}^{H} \delta(\chi_{ij}) + \delta_j(x_j)\left(c_j x_j + k\rho_j^*\right) \right.$$

$$\left. +cj\left(x_j - \rho_j^*\right) + c_j[\rho_j^* p_j^*(x_j - 1) - x_j p_j^*(x_j)] \right\}$$

при условиях

x_j, χ_{ij} - неотрицательные целые, $j = 1, \ldots, l, i = 1, \ldots, H$

$$\sum_{j=1}^{l} \chi_{ij} = ni\,, \sum_{j=1}^{H} \chi_{ij} - xj = 0$$

Задача IV

Здесь

$$\delta j(xj) = \begin{cases} 1, & \text{если } \rho_j p_j(x_j - 1) - xjp_j(xj) \geq 0{,}5, \\ 0, & \text{если } \rho_j p_j(x_j - 1) - xjp_j(x_j) < 0{,}5, \end{cases}$$

$$p_j(x_j) = \sum_{a_j = x_j}^{\infty} \frac{\rho_j^{a_j}}{a_j!} e^{-\rho_j}, \qquad p_j^*(x_j) = \sum_{a_j^* = xj}^{\infty} \frac{(\rho_j^*)^{a_j^*}}{a_j^*!} e^{-\rho_j^*}$$

$k\rho_j$ – штраф за нерешение задачи на j-м терминале,

k – коэффициент штрафа

Метод цепей Монте-Карло

- Существуют задачи стохастического программирования (например, **IV**), которые не могут быть точно решены известными методами.
- Целесообразно использовать метод цепей Монте-Карло

Метод цепей Монте-Карло

1. Поиск лучшего распределения, находящегося *на некотором расстоянии от базового*.
2. Найденное распределение становится *новым базовым*, относительно которого снова отыскивается лучшее распределение.
3. Продолжать до тех пор, пока будут наблюдаться улучшения.

Метод цепей Монте-Карло

Примем за базовое следующее распределение:

$$J_1 = \{q_{11}, \dots, q_{1\mathrm{k}1}, 0, q_{21}, \dots, q_{2k2}, 0, \dots, 0, q_{j1}, \dots, q_{jkj}, 0, \dots, 0\}$$

где q_{jk_j} – номер ВЦ, которым пользуется j-й терминал
k_j – число ЭМ, выделенных терминалу j, $j \in \{1, 2, \dots, l\}$

$$k_j = \begin{cases}]p_j[& \text{если } i > j > 1 \\ n - \sum_{r=1}^{i-1} k_r & \text{если } j = i \\ 0, & \text{если } j > i \end{cases}$$

i – номер терминала, начиная с которого

$$\sum_{r=1}^{i} k_r = n$$

Метод цепей Монте-Карло

Примем за базовое следующее распределение:

$J_1 = \{q_{11}, \ldots, q_{1\mathrm{k}_1}, 0, q_{21}, \ldots, q_{2k_2}, 0, \ldots, 0, q_{j1}, \ldots, q_{jk_j}, 0, \ldots, 0\}$

- Последовательность J_1 состоит из числовых интервалов, разделённых $l - 1$ нулём.
- Интервал между соседними нулями соответствует определённому терминалу – сколько ЭМ с каких ВЦ выделено данному терминалу.

Метод цепей Монте-Карло

Последовательность J_1 получена при условии, что спрос на терминалах превышает возможности систем, т.е.

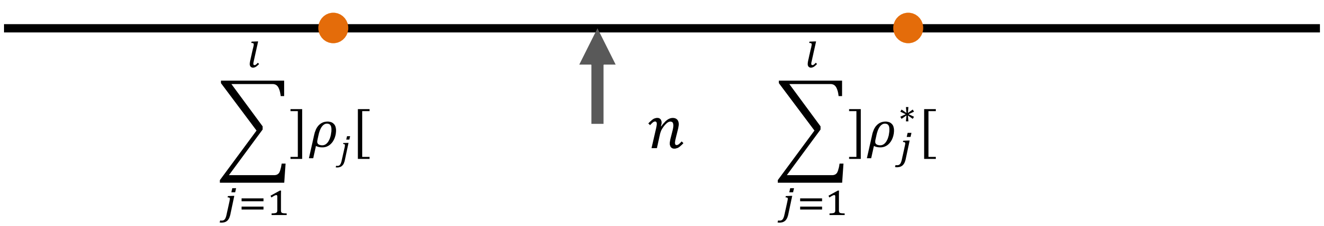
$$\sum_{j=1}^{l}]\rho_j^*[\geq n$$

Это наиболее вероятный случай.

Метод цепей Монте-Карло

Если же

$$\sum_{j=1}^{l}]\rho_j[\, < n \qquad \sum_{j=1}^{l}]\rho_j^*[\, \geq n$$



то

$$k_j = \begin{cases}]\rho_j^*[& \text{если } i > j \geq 1 \\ n - \sum_{r=1}^{i-1} k_r & \text{если } j = i \\ 0, & \text{если } j > i \end{cases}$$

Если

$$\sum_{j=1}^{l}]\rho_j^*[< n$$

то

$$k_j = \begin{cases}]\rho_j^*[& \text{если } j = 1,2,\dots,l-1 \\ n - \sum_{r=1}^{l-1} k_r & \text{если } j = l \end{cases}$$

Метод цепей Монте-Карло

1. Для текущего базового распределения J_1 вычисляется целевая функция f.
2. Получаем новое распределение на расстоянии h, $1 < h < n + l - 1$, от базового.
3. Если полученное решение характеризуется меньшим значением f, полученное распределение берётся в качестве *нового базового*.
4. Если сделано g попыток без уменьшения f, то берутся распределения с расстоянием h / 2 от базового.
5. Условие завершения алгоритм – расстояние $h = 2$.

Как получать новое распределение?
Что есть расстояние?

Число порядковых номеров ВЦ новой последовательности, которые не следуют за тем же номером, что и в базовой последовательности.

$J_1 = (1, 1, 2, 2, 0, 2, 3, 3, 4, 4, 4, 0, 3, 3, 4)$ – базовое

$J_1 = (1, 1, 2, 2, 3, 0, 2, 3, 4, 4, 4, 0, 3, 3, 4)$ – $h = 1$

$J_1 = (1, 1, 2, 3, 0, 2, 2, 3, 4, 4, 4, 0, 3, 3, 4)$ – $h = 2$

$J_1 = (1, 1, 2, 3, 0, 2, 2, 3, 4, 4, 4, 4, 0, 3, 3)$ – $h = 3$

$J_1 = (1, 2, 3, 0, 2, 2, 3, 4, 4, 4, 4, 0, 1, 3, 3)$ – $h = 4$

Метод цепей Монте-Карло

ALGORITHM ***MonteCarloChain***

```
J ← GetBaseSol();                       // e.g.  Greedy
h ← InitDistance();                     // e.g.  h ← n + l – 2;
do
        do
                J′ ← GetNewSol(J, h)
                nattempts ← 0
                if f (J′) < f (J) then
                        J ← J′
                        nattempts ← nattempts + 1
                else
                        nattempts ← 0
                end if
        while nattempts < g
        h ← h / 2
while h ≠ 2
```

Чем характеризуется эффективность метода цепей Монте-Карло?

- Время распределения машин по терминалам.
- Разность $\Delta z = z_0 - z^*$, где z_0 – значение f для исходного базового распределения

Резюме

1. Для организации стохастически оптимального функционирования ВС при обслуживании потока задач целесообразно использовать методы теории игр и стохастического программирования.
2. Теоретико-игровые методы целесообразно использовать тогда, когда поток имеет высокую интенсивность и допустимо образование пакетов задач из их очереди.
3. Методы стохастического программирования основываются на знании распределений вероятностей различных спросов (например, спросов на сервисные программы, на подсистемы с заданным числом машин и т.д.).

Модильяни. “Портрет Жанны Эбютерн”