Генетический алгоритм для задачи балансировки нагрузки на серверы

Дорожко Антон Владимирович

Научный руководитель: д.ф.-м.н., профессор Кочетов Юрий Андреевич

30 апреля 2015г.

Постановка задачи балансировки нагрузки

Данные

S - множество серверов

D - множество дисков

T - плановый период

R - множество характеристик (ЦП, ОЗУ, ...)

c_{drt} - нагрузка диска d в момент t по характеристике r

$\bar{c}_{sr}$ - пороговая нагрузка сервера s по характеристике r

x^0_{ds} - начальное распределение дисков по серверам

$b^w_{sdr}(b^e_{sdr})$ - накладные расходы на вставку и изъятие

B_{sr} - предельно допустимые накладные расходы

Задача

Минимизировать суммарную перегрузку серверов за счет перераспределения дисков по серверам на протяжении планового периода при установленных ограничениях на накладные расходы по каждому серверу

Математическая постановка задачи

Переменные задачи

$$x_{ds} = \begin{cases} 1 \text{ , если } d \text{ ставится на сервер } s \\ 0 \text{ , в противном случае} \end{cases}$$

ЦЛП формулировка

$$\min \sum_{s \in S} \sum_{t \in T} \sum_{r \in R} \max(0, \sum_{d \in D} c_{drt} x_{ds} - \bar{c}_{sr}) \tag{1}$$

при ограничениях:

$$\sum_{s \in S} x_{ds} = 1 \tag{2}$$

$$\sum_{d \in D} b^{w}_{sdr} x_{ds} (1 - x^{0}_{ds}) + \sum_{d \in D} b^{e}_{sdr} (1 - x_{ds}) x^{0}_{ds} \leq B_{sr} \tag{3}$$

Известные результаты

- Сформулированная задача (1)-(3) является NP-трудной *Кочетов Ю. А., Кочетова Н. А.* Задача балансировки нагрузки на серверы. // Вестник НГУ. 2013
- *Бежецков Д. Е.* Поиск с запретами для задачи балансировки нагрузки на серверы. // Труды X Междунароной Азиатской школы-семинара "Проблемы оптимизации сложных систем". 2014
- *Давыдов И. А., Кононова П. А., Кочетов Ю. А.* Локальный поиск с экспоненциальной окрестностью для задачи балансировки нагрузки на серверы. // Дискретный анализ и исследование операций. 2014
- *Davydov I., Kochetov Y.* VNS-based heuristic with an exponential neighborhood for the server load balancing problem. // Electronic Notes in Discrete Mathematics. 2014

Более сложная задача

Классы дисков

D_0 - множество больших дисков (проблемы с перемещением)
D_1 - множетсво потенциально больших дисков (могут стать большими)
S_0 - сервера, содержащие большие диски

На большие диски:

$$x_{ds} = x^0_{ds}, \quad \forall d \in D_0, s \in S_0 \tag{4}$$

$$x_{ds} = 0, \quad \forall d \notin D_0, s \in S_0 \tag{5}$$

На потенциально большие диски:

$$\sum_{d \in D_1} x_{ds} \leq 1, \quad \forall s \in S \backslash S_0 \tag{6}$$

Сведение задачи с классами к задаче без классов

Разгрузка серверов с большими дисками

Получаем новое начальное решение x'_{ds} в котором выполняется

$$x_{ds} = 0, \quad \forall d \notin D_0, s \in S_0$$

Исключаем из рассмотрения сервера S_0 и учитываем транспортные затраты на получение этого решения

Целевая функция со штрафом

$$y = \sum_{t \in T} \sum_{r \in R} \max(0, \sum_{d \in D} c_{drt} x_{ds} - \bar{c}_{sr})$$

$$\min \sum_{s \in S \setminus S_0} (y + W \max(0, \sum_{d \in D_1} x_{ds} - 1))$$

Общая схема генетического алгоритма

1. Сгенерировать начальную популяцию
2. Применить алгоритм локального поиска к решениям популяции
3. Пока не выполнен критерий остановки
   - 3.1 Выбрать решения x_1, x_2
   - 3.2 применить оператор скрещивания, получить x'_{12}
   - 3.3 Выбрать решение x_3
   - 3.4 Применить оператор мутации, получить x'_3
   - 3.5 Улучшить новые решения x'_{12} и x'_3 алгоритмом локального поиска
   - 3.6 Применить стратегию замещения
4. В качестве ответа предоставить лучшее решение из популяции

Заполняем схему 1

1. **Генерация начальной популяции**

   Метод: Последовательная диверсификация

   1.1 Выбрать минимальное расстояние между решениями
   1.2 Каждое решение должно быть на расстоянии от других решений в популяции не меньше выбранного

   Дополнить популяцию до нужного размера **жадной рандомизированной генерацией**

2. **Стратегия замещения**

   Оставлять решения с наилучшим значением целевой функции

3. **Критейрий остановки**

   Заданное количество итераций или остановка по времени

Заполняем схему 2

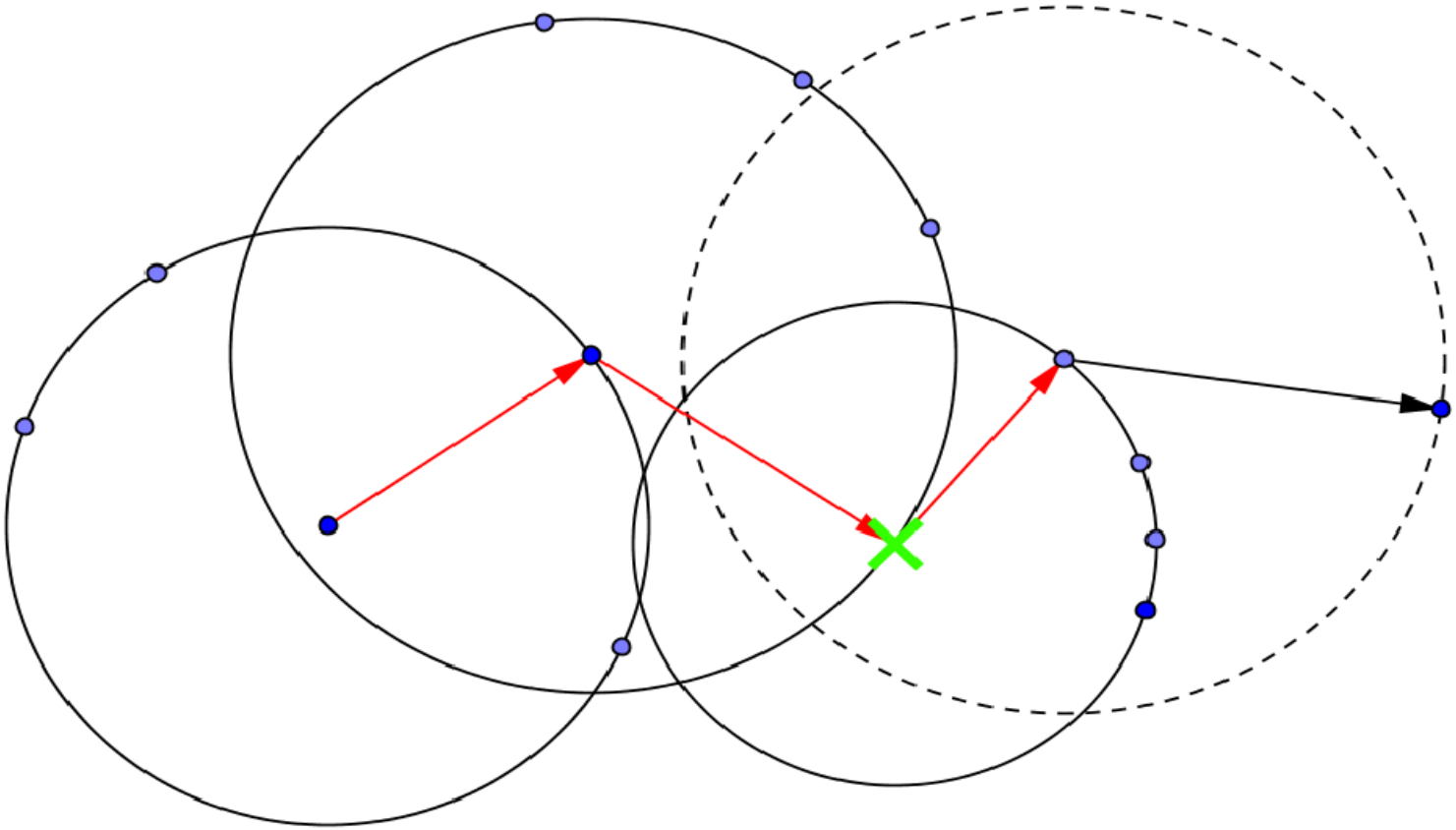
Локальный поиск : Окрестности

1. Move() - двигаем диск на другой сервер, мощность окрестности $\mathcal{O}(|D||S|)$
2. Swap() - меняем один диск на другой, мощность окрестности $\mathcal{O}(|D|^2)$
3. Lin-Kernighan() - некоторое обобщение Move()

Поиск по окрестностям рандомизирован

Сервера сортируются по перегрузке
Для Move() и Swap() выбирается один перегруженный и один недогруженный сервер

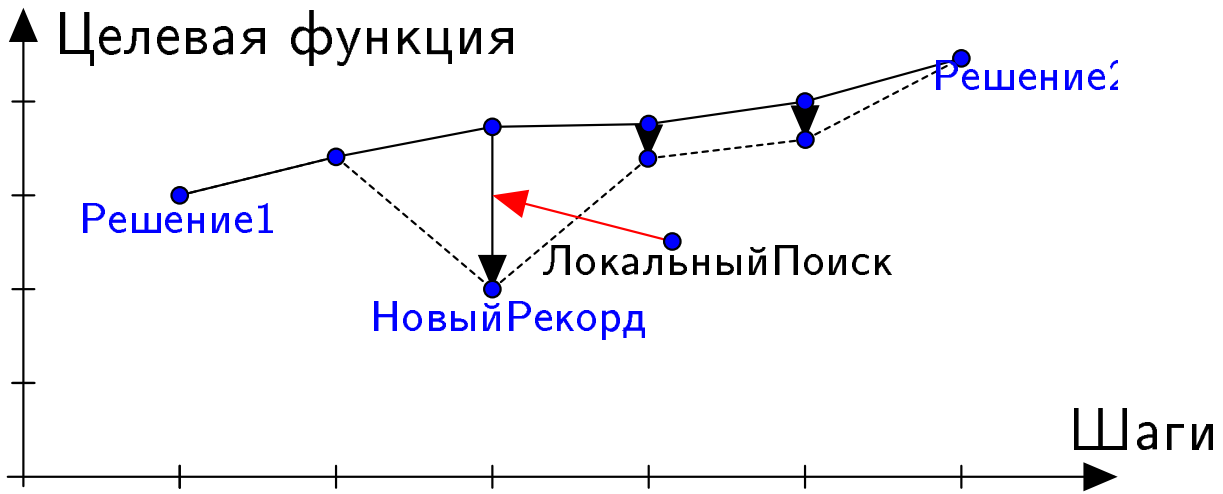
Рис.: Lin-Kernighan



Заполняем схему 3

Скрещивание - Построение путей (Path relinking)

Выбираем 2 решения. Строим путь от решения с меньшей целевой функций к решению с большей целевой функцией. Выбираем несколько лучших новых решений



Мутация - не используется

Причины:

1. поиск по окрестностям рандомизирован
2. диверсификация достаточна

Результаты

Тестовый пример

$|D| = 200, |S| = 20, |R| = 6, T = 1008$

	Без классов дисков			
Итерации	Окрестность	t_{GA}	F_{GA}	
1078	$Move()$	$600''$	$1.61933 * 10^6$	
1552	$Swap()$	$600''$	$6.72452 * 10^6$	
1178	$Move() + Swap()$	$600''$	$1.53738 * 10^6$	
675	$LK()$	$600''$	$1.57442 * 10^6$	
1832	$Move()$	$1200''$	$1.55846 * 10^6$	
3024	$Swap()$	$1200''$	$6.09213 * 10^6$	
2337	$Move() + Swap()$	$1200''$	$1.53709 * 10^6$	
1249	$LK()$	$1200''$	$1.54124 * 10^6$	
	С классами дисков			
	Красные	Желтые	t_{GA}	F_{GA}
500	3	0	$386''$	$5.64 * 10^6$
		7	$360''$	$5.64 * 10^6$
		15	$394''$	$5.64 * 10^6$

Тестовый пример

$|D| = ?, |S| = 20, |R| = 6, T = 1008$

Окрестность: объединение Move() и Swap()

Таблица: Результаты работы алгоритма

$\|D\|$	t_{GA}	RAM, MB	F^0_{GA}	F_{GA}
2680	$10'$ $20'$	< 300	72038	55 0
10720	$10'$ $20'$	< 700	$2.57277 * 10^7$	$2.56154 * 10^7$ $2.56350 * 10^7$
26800	$10'$ $20'$	< 1500	$1.30869 * 10^8$	$1.30842 * 10^8$ $1.30834 * 10^8$

Публикации

- *Дорожко А. В.* Генетический алгоритм для задачи балансировки нагрузки на северы. // Труды X международной Азиатской школы-семинара «Проблемы оптимизации сложных систем». 2014
- *Дорожко А. В.* Генетический алгоритм для задачи балансировки нагрузки на северы. // Материалы 53-й международной научной студенческой конференции МНСК-2015: Математика. 2015

Спасибо за внимание!