

Экспериментальное исследование разработанного алгоритма имитации отжига в рамках второго задания по курсу «Практикум на ЭВМ.»

Никифоров Никита Игоревич, 421 группа

Введение

Необходимо было разработать алгоритм имитации отжига в применении к задаче построения распределённого расписания. В моём варианте необходимо было реализовать критерий K3, а именно разбалансированность расписания (т.е. значение разности $T_{\max} - T_{\min}$, где T_{\max} — наибольшая, по всем процессорам, длительность расписания на процессоре; T_{\min} — аналогично, наименьшая длительность). Также необходимо было реализовать многопоточный алгоритм имитации отжига с синхронизацией через разделяемую память.

Методика экспериментального исследования

Все тестовые данные используемые в данном экспериментальном исследовании доступны в папке `tests` проекта. Так как алгоритм является недетерминированным бралась средняя оценка по критерию среди 10 прогонов. Время выполнения фиксировалось утилитой `std::chrono`. Исследование проводилось без запущенных посторонних программ.

Тестовый стенд

- Процессор — AMD Ryzen 2700X 8/16 (ядер/потоков) 4.2 Ghz на одно ядро, 4.0 Ghz на все ядра.
- Память — 32Gb DDR4 3200Mhz (Пропускная способность 44GB/s).

Экспериментальное исследование

Поиск опорной точки

Необходимо найти опорную точку экспериментального исследования, а именно вычислить количество задач и процессоров, такое, что время выполнения на одном потоке будет больше 1 минуты при использовании какого либо закона понижения температуры.

Послесловие

В папке `tests/root_point` находится файл `50000_100.xml`, если запустить программу вместе с сидом равным 1604022559, то при использовании оптимизированного начального решения, алгоритм не сходится. А именно после завершения Параллельного $N_{proc} = 1$ алгоритма имитации отжига, оценка лучшего найденного решения равна оценке исходного 3196.39. При этом алгоритм работает меньше секунды. Если же задать прямоё начальное решение, а именно, когда все задачи находятся на одном процессоре алгоритм имитации отжига работает более 2-х минут, и находит оптимальное решение оценка которого равна 51.69.

В данном случае мы наблюдаем стандартный вариант нахождения локального минимума.

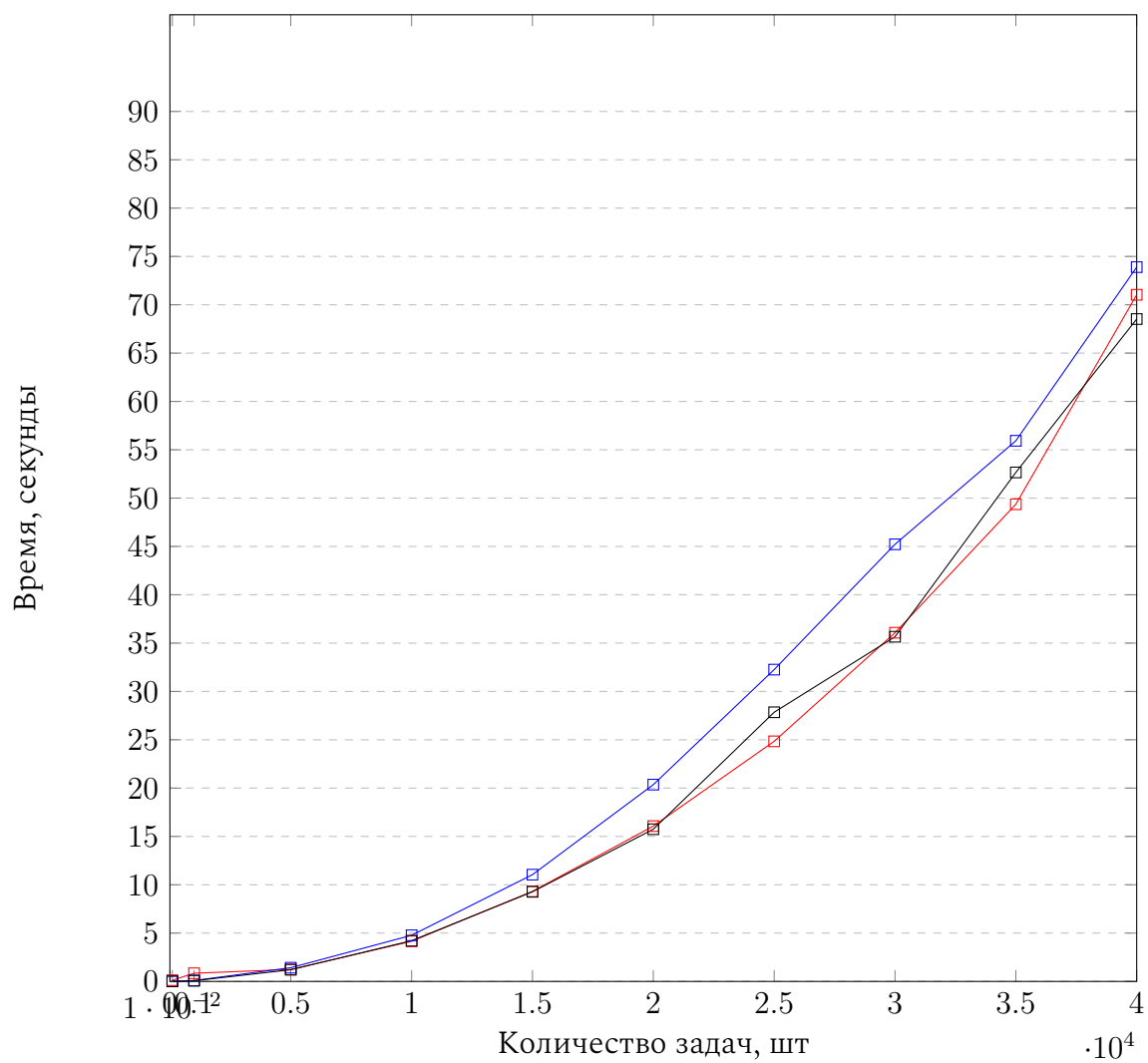


Рис. 1: Dependence of memory from number of rules.

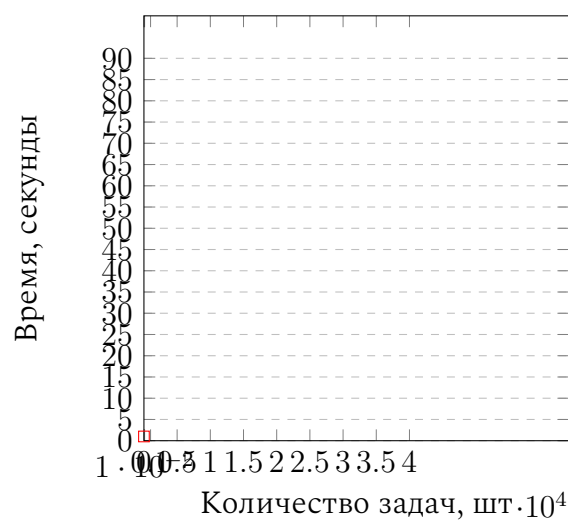


Рис. 2: Dependence of memory from number of rules.