Содержание

1 Постановка задачи **3**

2 Исследование последовательного алгоритма **4**

3 Исследование параллельного алгоритма **5**

1 Постановка задачи

Дано

Набор независимых работ $P = \{p_i\}_{i=1}^N$, где N - число работ. Работы не прерываемы, директивные сроки не установлены. Для каждой работы установлено время ее выполнения: работа p_i выполняется время $time_i$.

Набор процессоров $PU = \{PU_i\}_{i=1}^M$, где M - число процессоров. Каждый процессор может выполнять одновременно одну работу в сооветствии временем ее выполнения.

Требуется

Сформировать корректное расписание HP выполнения работ такое, что оно минимизирует заданный критерий $F(HP)$.

$HP = (HP_B, HP_L)$ называется расписанием, если заданы HP_B и HP_L, где:

$HP_B : P \to PU$ - привязка работ;

HP_L - порядок выполнения работ на каждом процессоре.

Расписание HP корректно, если выполнены следующие ограничения:

1. Каждая работа должна быть назначена на процессор.

2. Любую работу обслуживает лишь один процессор.

3. Процессор в каждый момент времени выполняет не более одной работы.

Минимизируемый критерий

Значение суммарного времени ожидания, то есть суммы времён завершения работ. Времена завершения работ определяются по временной диаграмме расписания.

Временная диаграмма строится следующим образом. Для каждого процессора последовательно выставляются задачи с соответствующей привязкой на выполнение в соответствии с порядком G_{HP} на этом процессоре. Время начала выполнения первой по порядку работы p на рассматриваемом процессоре определяется как $t_p^{start} = 0$. Для последующих работ $t_{p_i}^{start} = t_{p_j}^{finish}$, где p_j - работа, предшествующая работе p_i в соответствии с порядком G_{HP}. Время окончания выполнения работы p_i вычисляется как $t_{p_i}^{finish} = t_{p_i}^{start} + time_i$.

Тогда минимизируемый критерий вычисляется следующим образом:

$F(HP) = \sum_{i=1}^N t_{p_i}^{finish}$

2 Исследование последовательного алгоритма

Законы температуры

Рассматриваемые законы уменьшения температуры (за i обозначен номер итерации).

1. Cauchy: $T = \frac{T_0}{1+i}$
2. LogDiv: $T = T_0\frac{log(1+i)}{1+i}$
3. Boltzmann: $T = \frac{T_0}{log(1+i)}$

Экспериментально было определено, что при N=700 и M=50 последовательный алгоритм ИО работает больше 1 с законом LogDiv.

Было проведено 60 запусков алгоритма со всеми законами на этих данных:

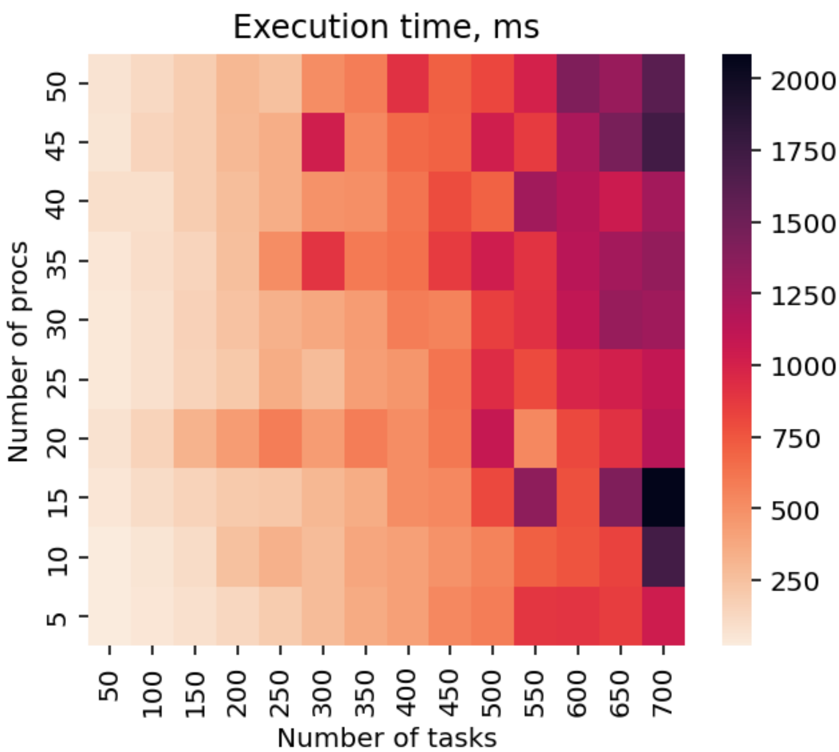
Закон	Среднее время выполнения, мс	Среднее значение целевой функции
Cauchy	872.205	42079
LogDiv	1128.09	39885
Boltzmann	1385.75	39259

Дольше всего алгоритм работает с законом Boltzmann, причем находит решение лучше, чем при других законах уменьшения температуры.

Далее будет использоваться Boltzmann.

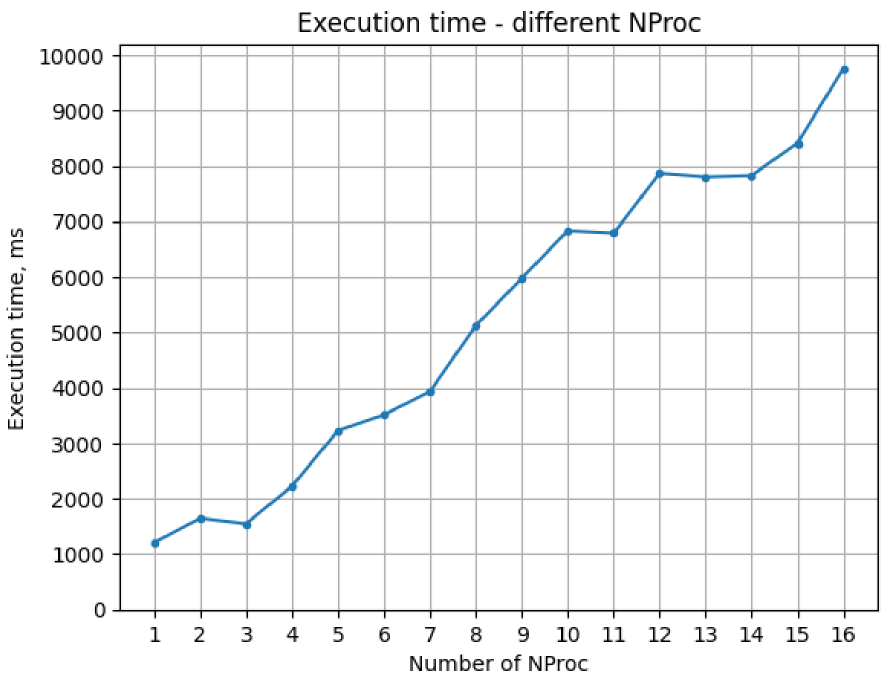
Время работы алгоритма в зависимости от размера входных данных

Была построена температурная карта зависимости среднего (по 5 прогонам) времени работы алгоритма от значений M и N.

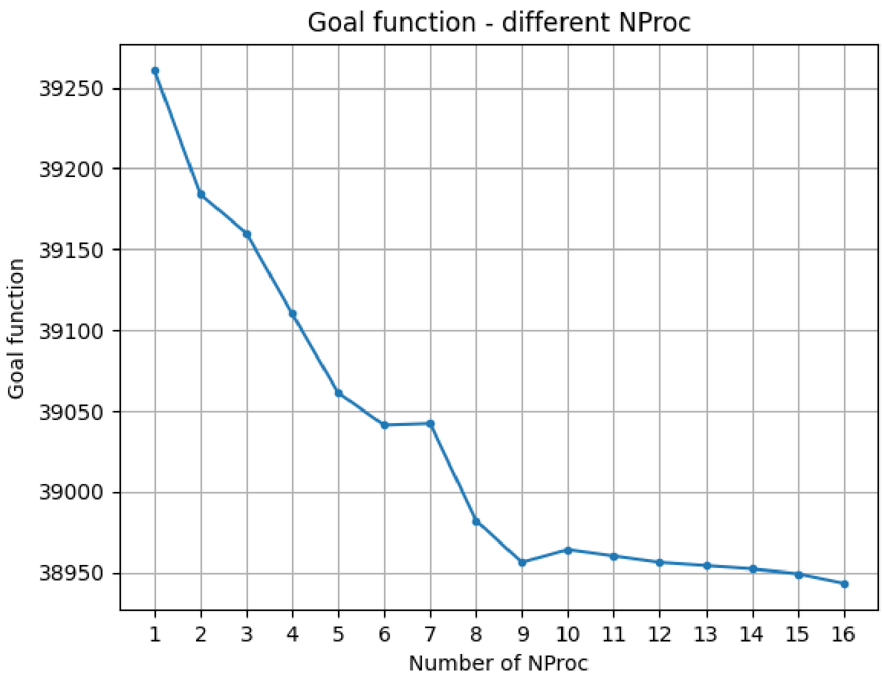


3 Исследование параллельного алгоритма

Был построен график зависимости времени работы параллельного алгоритма от числа параллельных процессов (NProc).



И график зависимости среднего значения целевой функции от NProc.



При увеличении NProc время работы алгоритма неуменьшается, так как приходится ждать завершения всех параллельно работающих отжигов. А так как разброс по времени довольно высокий, то при каждой итерации работы отжигов время ожидания нескольких отжигов в среднем дольше, чем одного.