**Исследование эффективности поиска глобального экстремума методом имитации отжига, распараллеленного различными способами**

**А. В. Высоцкий, А. С. Тараканов, К. И. Шоломов,**

**Н. Е. Тимофеева, А. А. Ерофтиев**

*Саратовский государственный университет имени Н.Г. Чернышевского, Саратов, Россия*

**Введение**

Современные задачи науки и техники, такие как моделирование климата, генная инженерия, проектирование интегральных схем, анализ загрязнения окружающей среды, создание лекарственных препаратов, наноматериалов и др. требуют для своего решения ЭВМ с огромной вычислительной производительностью. Единичные рабочие станции уже не способны в короткие сроки решать вышеперечисленные задачи. Здесь и возникает необходимость использования параллельных вычислений, когда над общей задачей работают десятки, сотни и даже тысячи рабочих станций.

Цель данной работы: реализация и проведение сравнительного анализа методов распараллеливания алгоритма нахождения глобального минимума многоэкстремальной целевой функции многих переменных с явными ограничениями (ограничениями типа равенств), основанного на стохастическом методе имитации отжига.

При этом решались следующие задачи:

* поиск эффективных способов распараллеливания алгоритма, с точки зрения минимизации времени и повышения надежности вычислений;
* программная реализация поиска глобального экстремума методом имитации отжига, распараллеленного различными способами;
* проведение численного эксперимента для оценки эффективности способов распараллеливания;
* анализ результатов численного эксперимента.

**Параллельные варианты метода имитации отжига**

Появление и развитие параллельных вычислительных систем привело к развитию нескольких подходов к распараллеливанию метода имитации отжига.

В настоящее время существуют различные подходы к реализации алгоритма имитации отжига на параллельных вычислительных системах, рассмотрим 3 из них:

1. параллельный запуск алгоритма имитации отжига (мультистарт);
2. параллельный запуск алгоритма имитации отжига с обменом результатами (кластерный алгоритм);
3. разбиение пространства решений на области [1].

Рассмотрим их подробнее.

**Параллельный запуск алгоритма имитации отжига**

Данная адаптация метода имитации отжига предполагает вычисление глобального минимума ЦФ одновременно на нескольких вычислительных узлах с последующим выбором лучшего решения на одном выделенном узле - управляющим.

В выполняемой на нескольких вычислительных узлах при одинаковых начальных условиях осуществляется поиск минимума ЦФ методом имитации отжига по схеме Больцмановского тушения [2].

Затем результаты, полученные на рабочих узлах, передаются в управляющий узел, в котором осуществляется выбор решения с наименьшим значением ЦФ, что должно обеспечивать глобальность найденного минимума.

Реализованный таким образом на параллельной вычислительной системе алгоритм оптимизации методом имитации отжига должен обеспечивать надёжный поиск глобального минимума целевой функции за счёт многократного повторения поиска в заданной явными ограничениями области.

При этом, так как многократный процесс поиска минимума осуществляется одновременно на соответствующем числе вычислительных узлов, то затраченное на выполнения всего параллельного алгоритма время приблизительно равно времени выполнения одного процесса и практически не зависит от числа процессов. Соответственно, при использовании рабочих узлов можно уменьшить временные затраты при поиске глобального минимума примерно в раз [3].

**Кластерный алгоритм**

Кластерный алгоритм является дальнейшим развитием алгоритма, который представлен выше. Основным отличием его является введение в выполняемую на каждом вычислительном узле параллельной системы последовательную часть алгоритма этапа обмена результатами между рабочими узлами.

Изначально на каждом из узлов параллельной вычислительной системы запускается алгоритм имитации отжига с разными начальными точками ,. По выполнении заданного количества итераций рабочие узлы обмениваются текущими результатами, после чего продолжают работу, исходя из лучшего значения результата. После *t* обменов рабочие узлы продолжают работу по получению индивидуального результата. По достижении условия завершения поиска главный узел вычислительной системы выбирает лучшее решение [3].

**Разбиение пространства решений на области**

В данном методе пространство решений алгоритма имитации отжига разбивается на несколько областей, в каждой из которых запускается последовательный алгоритм имитации отжига. В этом случае работа главного вычислительного узла, так же как и в упомянутых ранее адаптациях, сводится к выбору наилучшего решения среди полученных на рабочих узлах параллельной вычислительной системы.

Данный подход позволяет строить параллельные алгоритмы с низким трафиком обмена между параллельными процессами [3].

**Описание численного эксперимента и анализ полученных результатов**

Оценка надёжности и эффективности распараллеленного алгоритма имитации отжига по схеме Больцмановского тушения осуществлялась на многоэкстремальных ЦФ, с количеством параметров оптимизации , равным 2:

* гладкая многоэкстремальная функция:

(1)

где , ;

* многоэкстремальная функция Растригина:

, (2)

где , ;

* негладкая 4-х экстремальная функция:

;

;

; (3)

;

,

где , ;

* негладкая 4-х экстремальная функция, имеющая три равных по значению глобальных минимума:

;

;

; (4)

;

,

где , ;

Тестирование проводилось при следующих условиях: начальная температура , конечная , коэффициент изменения температуры . При таких значениях параметров обеспечивалось оптимальное соотношение точности результата и времени вычисления алгоритма.

Результаты поиска минимума ЦФ (1), (2), (3) и (4) тремя распараллеленными методами имитации отжига в зависимости от числа одновременно работающих процессов , усредненные по итогам 100 запусков вычислительной системы, приведены на рисунках 1, 2, 3 и 4 для каждой из функций.

Рис. 1.

Рис. 2.

Рис. 3.

Рис. 4.

Как видно из рисунков 1– 4, при малом количестве работающих процессов результат выдается с большим разбросом, но при увеличении результаты стабилизируются и приближаются к глобальному минимуму, так как поиск в этом случае ведётся в большей части заданной явными ограничениями области факторного пространства.

При решении реальных оптимизационных задач большое время занимает вычисление ЦФ. Соответственно, эффективность метода оптимизации можно оценивать по затраченному для поиска глобального минимума количеству временных ресурсов. Как говорилось ранее, в случае параллельного запуска алгоритма на вычислительных узлах среднее время вычисления алгоритма не будет зависеть от числа одновременно работающих процессов. Зависимость времени вычисления от числа одновременно работающих процессов оставшихся способов распараллеливания алгоритма приведена на рисунке 5.

Рис. 5. Время поиска минимума ЦФ 1 – 4 методом разбиения пространства решений(РПР) на области и кластерным алгоритмом(КА).

Значительное влияние на количество вычислений ЦФ и, соответственно, на эффективность тестируемого метода оптимизации оказывает значение температурного коэффициента. В ходе проведения численного эксперимента был сделан вывод, что изменение температурного коэффициента с 0.95 до 0.99 приводит к увеличению времени вычислений ЦФ практически в 6 раз, однако при этом точность нахождения решения повышается.

**Заключение**

В ходе проведенной работы были рассмотрены и реализованы 3 варианта распараллеливания алгоритма оптимизации многоэкстремальной целевой функции двух переменных с явными ограничениями, основанного на стохастическом методе имитации отжига, и проведен анализ эффективности работы каждого из способов распараллеливания.

СПИСОК ЛИТЕРАТУРЫ

1. *Kirkpatrick S., Gelatt C.D., Vecchi M.P.* Optimization by simulated annealing. // Science. v220. 1983. P. 671–680.
2. *Лопатин А.С*. Метод отжига // Межвуз. сб. «Стохастическая оптимизация в информатике». Выпуск 1, 2005. Санкт-Петербург: изд-воСПбГУ. С. 133–149.
3. *Janaki Ram D., Sreenivas T.H., Ganapathy Subramaniam K*. Parallel Simulated Annealing Algorithms// J. of parallel and distributed computing. 1996. №37, P.207-212.
4. *Калашников А.В., Костенко В.А.* Параллельный алгоритм имитации отжига для построения многопроцессорных расписаний// Известия РАН. Теория и системы управления, 2008., N.3, С.101–110.