**Gergel V. and Kozinov E. A Highly Parallel Approach for Solving Computationally Expensive Multicriteria Optimization Problems**

Ответы на замечания рецензентов:

|  |  |
| --- | --- |
| Замечание | Ответ |
| REVIEW 1 замечаний нет | |
|  |  |
| REVIEW 2 | |
| Представлен параллельный алгоритм и оценки параллельной эффективности.  Результаты по тестированию непонятные. Сказано, тестируется на Сэндиках 2х8. В таблице какие-то непонятные P и Q - число процессоров и ядер. Откуда взялось число 40? | Эксперименты проводились на нескольких кластерах, в частности СК Intel Endeavor. На СК Intel Endeavor есть процессоры с числом ядер >=40.  При проведении экспериментов мы ориентировались на процессоры Intel Xeon E5-2698V4.  P – соответствует числу запускаемых MPI процессов.  Q – соответствует числу потоков используемых в каждом MPI процессе. |
| Почему измеряется число каких-то итераций, а не фактическое время выполнения? | При решении реальных задач, как правило, вычисление значений функций занимает значительное время. Время работы алгоритма многокритериальной оптимизации при этом становиться менее существенным. Тем самым, на первый план выходит число итераций, за которое алгоритм смог достичь решения задачи. При уменьшении числа итераций уменьшается фактическое время счета. Число итераций позволяет оценить максимально ускорение которое можно достичь. |
| Приходим к выводу, что алгоритм в реальности на параллельной системе не тестировался (возможно, даже и не реализован?). | Алгоритм на реальных вычислительных системах тестировался – таки, например, как отмечено выше, на суперкомпьютере Intel Endeavor . В статье оценивалась масштабируемость алгоритма при использовании вычислительных систем с большим числом вычислительных ядер. |
| Заявленные в аннотации ускорения 10-100 (даже непонятно относительно чего) вообще никак не обоснованы. | При решении сложных задач многокритериальной оптимизации с большим числом критериев необходимо перебирать большое число наборов коэффициентов свертки . Предложенный алгоритм позволяет накопить и использовать информацию о решаемой задаче. Как следствие, при каждом новом наборе коэффициентов свертки число итераций требуемых на поиск решения задачи глобальной оптимизации будет уменьшаться вплоть до полного их отсутствия. В предельном случае решение можно найти по накопленной информации. В базовом алгоритме необходимо испытания проводить повторно.  Следовательно, при большом числе наборов коэффициентов свертки можно получить большое ускорение благодаря эффективному процессу поиска. Ускорение определось относительно базового последовательного алгоритма, не использующего повторную получаемую поисковой информации. |