

Параллельные гибридные алгоритмы на основе схемы “иди с победителями” для некоторых задач составления расписаний

П.А. Борисовский

Институт математики им. С.Л. Соболева СО РАН
(Омский филиал)

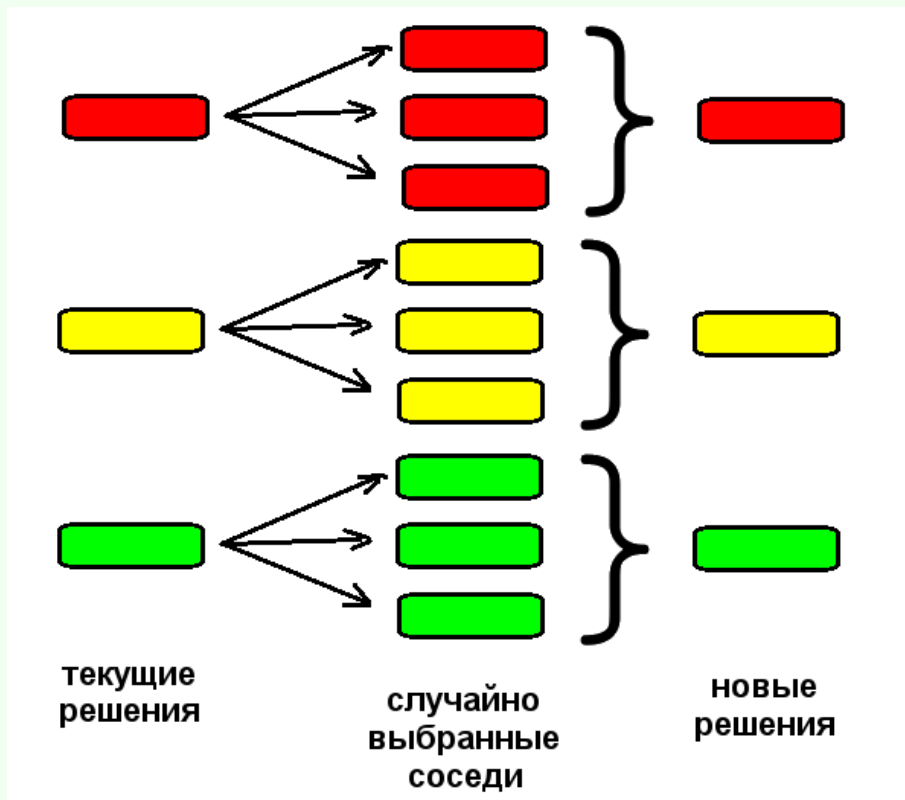
Исследование выполнено за счет гранта Российского научного фонда
№ 22-71-10015

Проблемы практического применения GPU

- Сложные модели (большое количество переменных и ограничений, постоянно меняющиеся требования).
- Сложные технологии (управление множеством потоков, оптимизация работы с памятью, сложность отладки и поиска ошибок).
- Сложные алгоритмы не всегда удобные для распараллеливания (методы ЦЛП, ветвей и границ и т.д.)

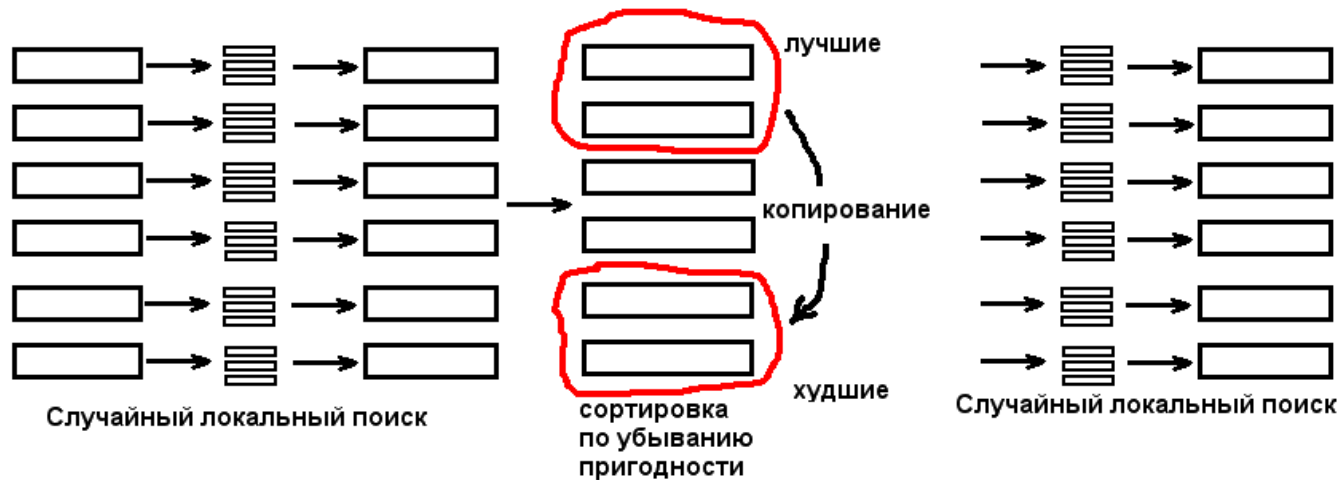
Цель состоит в разработке простой схемы алгоритма с высокой производительностью, подходящей для реализации на GPU.

Параллельный случайный локальный поиск



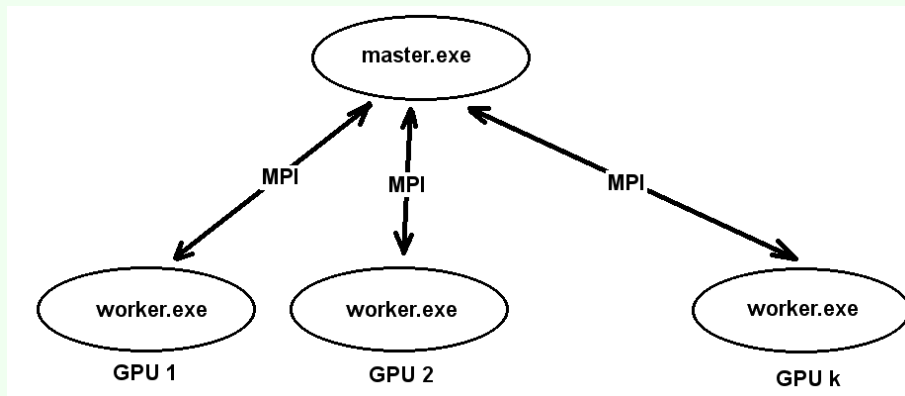
Алгоритм

«Иди с победителями»^a с локальным поиском



^a Aldous, D., Vazirani, U.: "Go with the winners" Algorithms. Proceedings of 35th IEEE Symposium on Foundations of Computer Science (FOCS), pages 492–501, 1994.

Реализация на нескольких GPU



- Удобнее всего использовать архитектуру «главный – подчиненный».
- Один процесс работает с одним GPU.
- Взаимодействие осуществляется с помощью сообщений MPI.
- GPU могут находиться на разных компьютерах.

Варианты реализации алгоритма

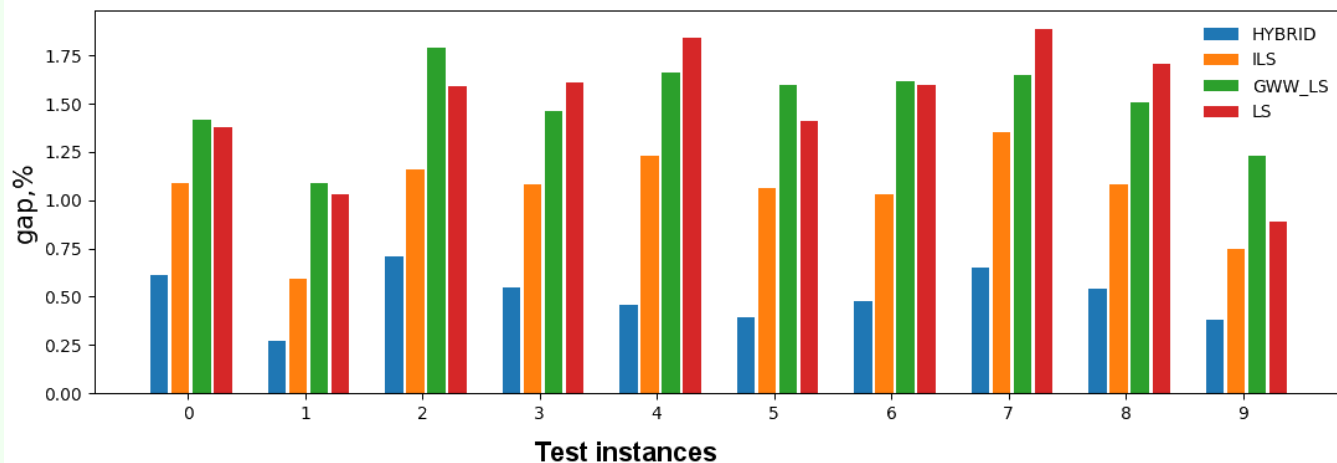
- Несколько независимых процессов (самый простой способ).
- Общая популяция и обмен лучшими решениями.

Гибридные алгоритмы

Использование метаэвристики вместо простого локального поиска.

- Поиск с запретами. Требует хранить большое количество списков запретов. Частое обращение к памяти сильно замедляет работу GPU приложения.
- Итеративный локальный поиск (ILS) и поиск с переменными окрестностями (VNS). Основаны на общей идее применять большие изменения к текущему решению при застревании в локальном оптимуме ("встряска" в ILS и переход к другой окрестности в VNS).

Эксперименты на задаче FlowShop



Hybrid – параллельный гибридный алгоритм с ILS.

ILS – параллельный алгоритм ILS без «иди с победителями».

GWW_LS – параллельный гибридный алгоритм с LS без встряски.

LS – параллельный алгоритм LS.

Задача составления производственного расписания

Постановка задачи^a.

J – множество работ, I – множество машин.

Каждая работа должна быть выполнена ровно один раз.

p_{ij} – длительность выполнения работы j на машине i .

$[e_j, d_j]$ – временное окно для окончания работы j .

s_j – максимально возможное запаздывание работы j .

z_i – «стоимость» запуска любой работы на машине i .

^a J. Berndorfer, S. N. Parragh. Modeling and solving a real world machine scheduling problem with due windows and processing set restrictions. Procedia Computer Science, V. 200, 2022, P. 1646–1653. <https://doi.org/10.1016/j.procs.2022.01.365>

Задача составления производственного расписания

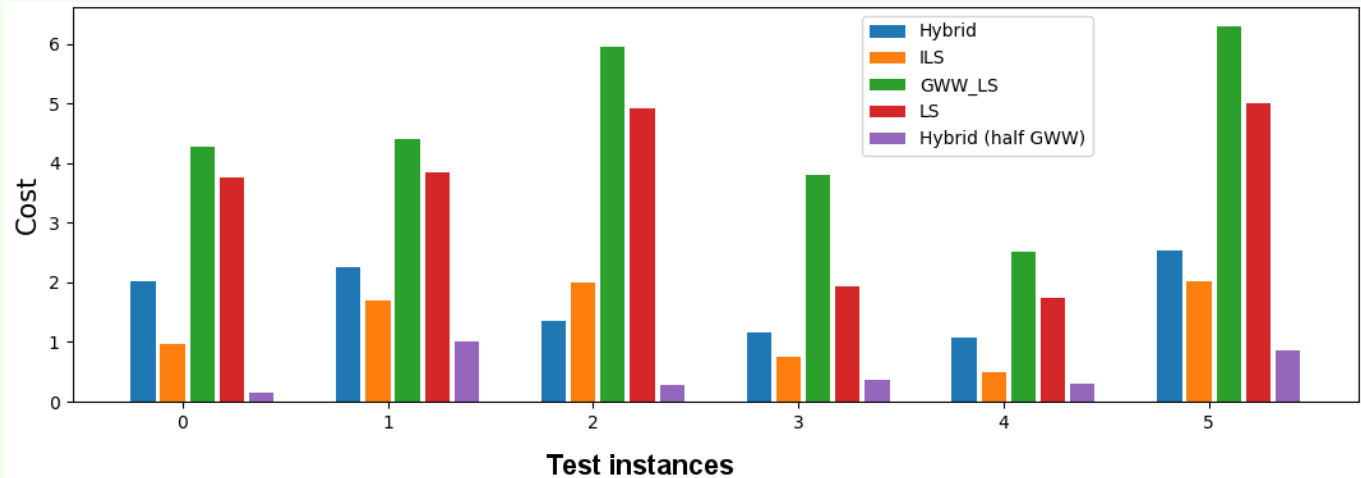
Критерий.

$$C = L + 0.001 \cdot Z \rightarrow \min,$$

где L – суммарное запаздывание

Z – суммарная стоимость запуска всех работ.

Результаты экспериментов



Hybrid – параллельный гибридный алгоритм с ILS.

ILS – параллельный алгоритм ILS без «иди с победителями».

GWW_LS – параллельный гибридный алгоритм с LS без встряски.

LS – параллельный алгоритм LS.

Hybrid (half GWW) – параллельный гибридный алгоритм с ILS, в котором «иди с победителями» применяется только во второй половине отведенного времени.

Задача составления расписаний выполнения заказов клиентов

m клиентов,

n продуктов,

$p_{ij} \geq 0$ длительность производства продукта j для клиента i ,

$s_{jj'} \geq 0$ длительность переналадки с продукта j на продукт j' ,

s_j начальная переналадка для продукта j ,

d_i срок выполнения заказов для клиента i ,

q_i "вес" клиента i .

Критерии: время завершения последней операции (makespan),
суммарное запаздывание и т.д.

$$\sum_{k=1}^{|N|} x_{ijk} = 1, \quad i = 1, \dots, m, \quad j = 1, \dots, n, \quad (1)$$

$$\sum_{i=1}^m \sum_{j=1}^n x_{ijk} = 1, \quad k = 1, \dots, |N|, \quad (2)$$

$$t_1^f \geq \sum_{i=1}^m \sum_{j=1}^n x_{ij1} (p_{ij} + s_j'), \quad (3)$$

$$t_k^f \geq t_{k-1}^f + p_{ij} + \sum_{i'=1}^m \sum_{j'=1}^n x_{i'j',k-1} s_{j'j} - H(1 - x_{ijk}), \quad (4)$$

$$k = 2, \dots, |N|, \quad i = 1, \dots, m, \quad j = 1, \dots, n,$$

$$T_i^f \geq t_k^f - H(1 - x_{ijk}), \quad (5)$$

$$k = 1, \dots, |N|, \quad i = 1, \dots, m, \quad j = 1, \dots, n,$$

$$H(1 - z_i) \geq T_i^f - d_i, \quad i = 1, \dots, m, \quad (6)$$

$$T_i^f \geq 0, \quad t_k^f \geq 0, \quad x_{ijk} \in \{0, 1\}, \quad y_i \in \{0, 1\}, \quad (7)$$

$$i = 1, \dots, m, \quad j = 1, \dots, n, \quad k = 1, \dots, |N|.$$

Если предположить, что есть несколько машин, и выполнение заказов может происходить по частям, то после фиксации бинарных переменных остается задача ЛП.

Существуют другие постановки с таким же свойством.

Представляет интерес разработка гибридного алгоритма на GPU с методами ЛП для решения таких задач.

Спасибо за внимание!