

Реализация динамической балансировки нагрузки во фрагментированных программах методом JobStealing

Выполнили: Тренин С.А., Чмиль А.В.
Руководитель: Перепёлкин В.А.

Новосибирск 2019
XXXII Зимняя школа по параллельному программированию.

Введение

Проблема — дисбаланс вычислительной нагрузки при выполнении фрагментированных программ в силу неоднородности конфигурации, динамики явления и т. д.

Решение — применение динамической балансировки нагрузки.

Метод JobStealing

Суть метода — недогруженные (или простаивающие) узлы запрашивают работу у остальных узлов.

Работа — фрагмент вычислений без побочных эффектов и с заданными параметрами.

Метод JobStealing

Проблемы метода:

- 1) Избыточные коммуникации могут значительно замедлить исполнение.
- 2) Недостаточные коммуникации также замедляют исполнение.
- 3) При бесконтрольной передаче запросов на получение работы, узлы будут передавать данные по кругу.

Метод JobStealing

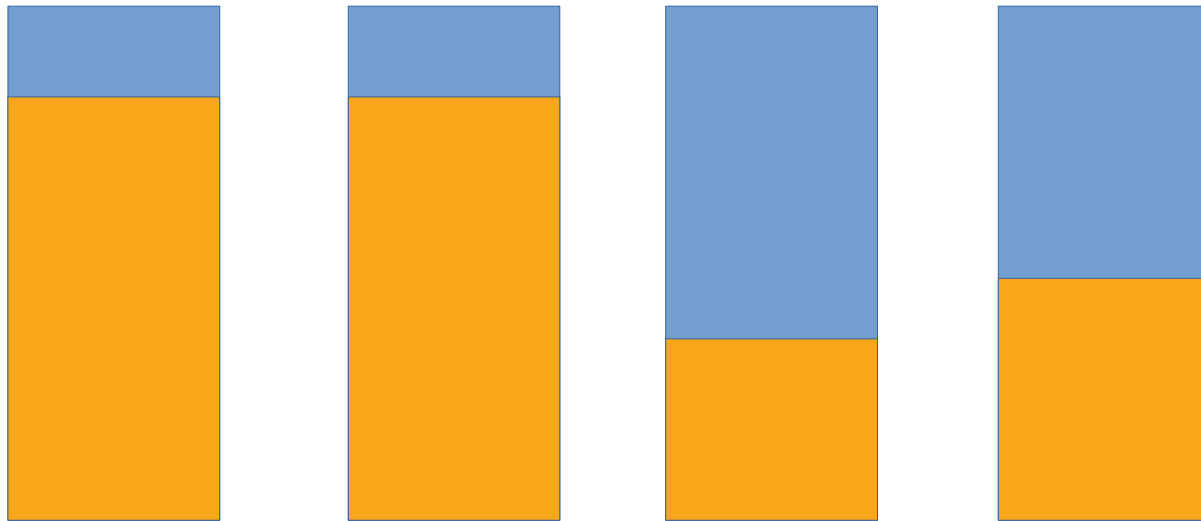
Детали нашей реализации:

- 1) Узел отправляет запрос на получение работы, только если он простаивает.
- 2) Запрос отправляется только на соседние узлы.
- 3) При получении запроса узел передаёт работу, только если число задач на нём больше, чем количество потоков исполнения.

Метод JobStealing



Метод JobStealing



Тестирование

Для тестирования была взята программа на языке LuNA, решающая трёхмерную задачу Пуассона с декомпозицией на «линейке».

Запуск производился на кластере МВС-10П ОП Межведомственного Суперкомпьютерного Центра.

Задача запускалась на 4 вычислительных узлах с 2 потоками исполнения на каждом. При этом, дабы наглядно продемонстрировать эффект динамической балансировки, один из узлов изначально не был загружен работой.

Тестирование

Использование JobStealing	Циклов исполнения задачи	Размер одного фрагмента вычислений	Общее число фрагментов	Время работы / Идеальное время работы
Нет	1	8x64x960	32	333.081413 sec./ 249.75 sec.
Да	1	8x64x960	32	558.757576 sec./ 249.75 sec.
Нет	250	8x64x960	32	104.544873 sec./ 78.00 sec.
Да	250	8x64x960	32	85.281108 sec./ 78.00 sec.
Нет	1000	16x64x960	32	310.256728 sec./ 232.50
Да	1000	16x64x960	32	228.624841 sec./ 232.50

Заключение

Нам удалось добавить к run-time системе seaLuNA возможность включать при выполнении задачи динамическую балансировку jobStealing. На данный момент ускорение на кластере происходит только при большом числе циклов выполнения задачи. В силу больших задержек коммуникации.

В дальнейшем планируется улучшить метод балансировки, опираясь на полученные результаты профилирования. В первую очередь, следует продолжить избавляться от излишних запросов и прочих коммуникаций.

Q&A?!