

**САМОПОДСТРАИВАЮЩАЯСЯ КОНКУРЕНТНАЯ СТРУКТУРА,  
ОПТИМИЗИРУЮЩАЯ ЗАПРОСЫ МЕЖДУ ВЕРШИНАМИ НА ОСНОВЕ SPLAYLIST**

**Надуткин Ф.М. (Университет ИТМО)**

**Научный руководитель – доцент Аксенов В.Е.**

**(Университет ИТМО)**

**Введение.** В большинстве имеющихся датацентрах вычисления связанные с той или иной задачей проводятся на распределённых вычислительных устройствах, связанных между собою физически. Зачастую, в виду большого количества машин, обеспечить общение каждой машины с каждой представляется физически и экономически невозможным, поэтому датацентры представляют из себя некоторую «древообразную» структуру. Как следствие, общение между двумя машинами может быть осложнено тем, что запрос должен проходить по другим машинам тем самым увеличивая время взаимодействия. В качестве решения предлагается возможное пересоединение машин между собой с целью уменьшить путь, который будет проходить между запросами машин друг от друга. Для того, чтобы этот процесс происходил эффективно существуют различные самоподстраивающиеся структуры [1], [3], в нашем случае мы возьмём за основу SplayList [2].

**Основная часть.** С помощью созданной структуры решаются следующие задачи:

- 1) Общение различных вычислительных устройств между собою, с последующим сбором и обработкой данных. В случае если в ходе работы алгоритма будет потеряна связь между какой-то парой машин, то это негативно скажется на работе всего датацентра. Как следствие, на любом этапе граф сети машин должен быть связан.
- 2) Поддержание распределённой структуры, корректно обрабатывающей запросы от одной машины к другой. Так как в один момент времени может происходить общение множества пар машин, то структура должна работать таким образом, чтобы быть защищённой от возможных dead lock, обеспечивать отсутствие голодания (когда какой-то запрос не может выполняться, потому что его прерывают другие) и корректно исполнять запросы (конечный результат должен быть одинаковым как для последовательного исполнения, так и для параллельного).
- 3) Задача реконфигурации имеющейся сети. Для эффективного исполнения запросов мы должны поддерживать возможность перестройки сети и переподсоединения машин. Основная сложность заключается в том, что данная структура конкурентна и как следствие в ходе переустройства могут возникать конфликты.
- 4) Ускорение запросов. Реконфигурация сети не бесплатная, поэтому алгоритм должен выдерживать баланс между её перестройкой и доставкой сообщения, чтобы получить максимальный выигрыш в скорости.

**Выводы.** Проведено изучение статей с похожей тематикой. Были придуманы, разработаны и протестированы оптимизации алгоритма SplayList.

**Список использованных источников:**

1. Bruna Peres, Otavio A. de O. Souza, Olga Goussevskaia, Chen Avin, Stefan Schmid, Distributed Self-Adjusting Tree Networks.
2. Aksenov V., Alistarh D., Drozdova A., Mohtashami A., The Splay-List; A Distribution-Adaptive Concurrent Skip-List.
3. Otavio A. de O. Souza, Olga Goussevskaia, Stefan Schmid, CBNet: Demand-Aware Tree Topologies for Reconfigurable Datacenter Networks.

Надуткин Ф.М.. (автор)

Подпись

Аксенов В.Е (научный руководитель)

Подпись