



ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ ВЫЧИСЛЕНИЙ

И.В. Бобылева, С.В. Востокин

АЛГОРИТМИЧЕСКИЙ КАРКАС ДЛЯ БЛОЧНОЙ ОБРАБОТКИ ДАННЫХ ПО ПРИНЦИПУ КРУГОВОГО ТУРНИРА

(Самарский университет)

В докладе представлена методика построения алгоритмического каркаса для автоматизации попарной обработки блочных данных по принципу кругового турнира (каждый с каждым) в гридах настольных компьютеров предприятия. Приведено описание семантики вычислений и декомпозиции каркаса на последовательные подпрограммы с использованием модели акторов. Введена графическая нотация, поясняющая связь между частями каркаса. Применимость предлагаемого метода программирования грид-приложений исследована экспериментально при решении задачи блочной сортировки большого массива данных.

Вычислительные гриды настольных компьютеров давно и успешно применяются в области научных вычислений как в масштабе сети Интернет, так и в масштабе отдельных предприятий. Основным мотивом такого использования является возможность радикально уменьшить затраты на вычислительное оборудование. Технология грид-систем настольных компьютеров (или шире – любых персональных вычислительных систем: смартфонов, планшетов, ноутбуков и т.п.) позволяет воспользоваться для вычислений большим количеством устройств в сети Интернет с согласия их пользователей. Это во многих случаях удобнее и дешевле, чем проводить вычисления на кластере или суперкомпьютере. Другой альтернативой уменьшения аппаратных затрат является использование для вычислений временно простаивающего (например, в ночные часы) оборудования предприятия – десктоп гридов предприятия [1,2].

В то время как уменьшаются затраты на оборудование, важную роль в себестоимости вычислений начинают играть затраты на программирование. Затраты остаются невысокими, пока десктоп гриды применяются для реализации простых стратегий поиска или перебора с массовым параллелизмом. Это обусловлено тем, что такие задачи относительно легко программируются на базе API десктоп гридов, или имеются разработанные алгоритмические скелетоны типа map (применить ко всем), в которых общее управление вычислениями уже реализовано. Исследователю требуется лишь определить алгоритмы решения задач, передаваемых на компьютеры десктоп грида.



Однако по мере улучшения характеристик коммуникационного и вычислительного оборудования становится актуальным применение десктоп-грид-систем для решения вычислительных задач с более сложным (по сравнению с массивно-параллельного типа map) управлением. Например, это могут быть задачи анализа данных, которые накапливаются на предприятии в течение рабочего дня и обрабатываются на настольных компьютерах предприятия в ночные часы.

Целью работы является формулировка методики и построение алгоритмического каркаса для организации вычислений на базе десктоп-грид-систем. Каркас предназначен для упрощения программирования задач, в которых используется попарная обработка блочных данных по принципу кругового турнира (каждый с каждым). Такой тип обработки может применяться для сортировки, построения частотных распределений и подобных задач.

Доклад имеет следующую структуру. В соответствии с принципом нисходящей декомпозиции, вначале на основе модели акторов [3] и специальной графической нотации рассматривается обобщенное описание семантики параллельных вычислений в десктоп-гриде. Предлагаемая графическая нотация позволяет выделить типы данных и последовательные процедуры их обработки. Далее мы конкретизируем описание типов данных и процедур, определяя алгоритмический каркас «портфель задач» [4]. В свою очередь, на основе каркаса «портфель задач» строится каркас «асинхронный круговой турнир». Практическая применимость каркаса «асинхронный круговой турнир» для обработки данных в десктоп-гриде предприятия исследуется экспериментально на примере задачи блочной сортировки с использованием кластерной системы «Сергей Королев». В заключении рассматривается применение описанного подхода для повышения эффективности вычислений в гриде настольных компьютеров предприятия под управлением платформы Everest, описанных в работе [5].

Литература

1. Афанасьев, А.П. Увеличение вычислительной мощности распределенных систем с помощью грид-систем из персональных компьютеров [Текст] / А.П. Афанасьев, R. Lovas, В.В. Волошинов [и др.] // Параллельные вычислительные технологии (ПаВТ'2011): труды международной научной конференции (Москва, 28 марта – 1 апреля 2011 г.). – Челябинск: Издательский центр ЮУрГУ, 2011. – С. 6-14.
2. Ivashko E. Enterprise Desktop Grids. CEUR Workshop Proceedings, Volume 1502, 2015, PP. 16–21.
3. De Koster, J., Van Cutsem, T., De Meuter, W., 43 years of actors: A taxonomy of actor models and their key properties. AGERE 2016 - Proceedings of the 6th International Workshop on Programming Based on Actors, Agents, and Decentralized Control, co-located with SPLASH, 2016. PP. 31-40.
4. H.Senger, F.A.B. da Silva. Bounds on the scalability of bag-of-tasks applications running on master-slave platforms. Parallel Processing Letters, Volume 22, Issue 2, 2012.



5. Vostokin S.V., Sukhoroslov O.V., Bobyleva I.V., Popov S.N. Implementing computations with dynamic task dependencies in the desktop grid environment using Everest and Templet Web. CEUR Workshop Proceedings, Volume 2267, 2018. PP. 271-275.

С.Р.Валеев, А.О.Новиков

СОЗДАНИЕ РАСПРЕДЕЛЕННЫХ ДИСПЕТЧЕРСКИХ СИСТЕМ НА БАЗЕ SCADA-ПАКЕТА WINCC OA

(Самарский университет)

Невозможно представить современное АСУ ТП без использования SCADA-систем. Их основными задачами являются: сбор и обработка данных в режиме реального времени, обмен данными с ПЛК, архивирование данных, аварийная сигнализация и т.д. Такими функциями должна обладать каждая SCADA-система. Для одних станций диспетчерские системы нужны только для отображения информации в реальном времени, архивирование и алармирование, для других, таких как крупные ГЭС, входящих в большой каскад ГЭС, одним из важнейших условий является возможность создания распределённых систем. Такие системы позволяют создавать единый диспетчерский пункт в одном географически распределенном месте. Такие SCADA для обмена данными между друг другом должны использовать свой внутренний протокол.

Рассмотрим пример распределенной диспетчерской системы, построенной на базе SCADA-пакета WinCC OA. На рисунке 1 показана структурная схема такой системы.

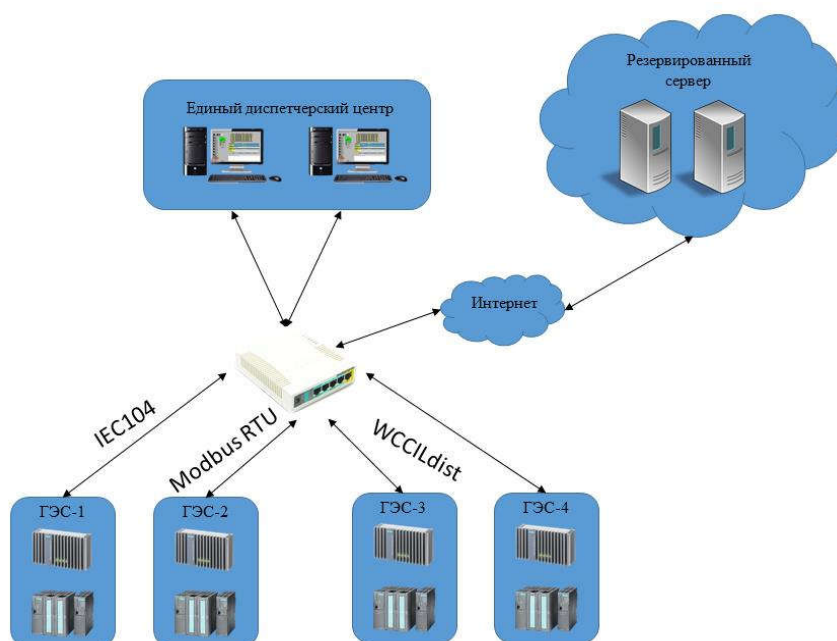


Рис. 1. Структурная схема распределенной диспетчерской системы