

А. С. Румянцев

Задача оптимизации времени выполнения проекта в вычислительной сети из персональных компьютеров

Аннотация. Рассмотрена модель процесса выполнения одного проекта в вычислительной сети из персональных компьютеров при условии равного времени выполнения подзаданий. Исследованы условия, при которых репликация подзаданий между узлами сети сокращает общее время выполнения проекта в двух частных случаях модели: при выполнении чачек подзаданий, а также при ненулевом штрафе за невыполнение подзадания.

Ключевые слова и фразы: вычислительные сети, стохастическое моделирование, оптимизация.

Введение

Наряду с суперкомпьютерами, высокопроизводительные вычислительные сети из персональных компьютеров (ВСПК) типа Desktop Grid являются важным средством ускорения вычислений. Гибкость и простота реализации проектов являются привлекательными для исследователей, что отражается в широте спектра прикладных задач, решаемых с помощью распределенных вычислений, см., напр. [1–4]. В то же время для ВСПК достаточно остро стоит задача оптимизации времени выполнения проекта [5, 6].

Существенным отличием ВСПК от вычислительных кластеров является территориальная распределенность первых, а также нестационарность во времени вычислительных узлов сети. Это связано с тем, что значительная часть ВСПК основана на принципах добровольных вычислений, когда участники ВСПК на безвозмездной основе предоставляют авторам проектов простаивающие вычислительные мощности своих персональных компьютеров. При этом приложения, связанные с проектом, имеют более низкий приоритет по сравнению

Работа поддержана грантом РФФИ, проект 13-07-00008.

с программами пользователя, работающего на этом компьютере. В случае превышения пользовательской нагрузкой некоторого порогового значения, связанные с проектом вычисления приостанавливаются, чтобы исключить воздействие на производительность компьютера в момент решения задач пользователя. При этом вычисление нарушивших так называемый дедлайн (предельное время выполнения) подзаданий останавливается. Кроме того, в отличие от ресурсов вычислительных кластеров, компьютеры пользователей доступны, как правило, не круглосуточно. Таким образом, наиболее частой причиной остановки выполнения подзаданий проекта является нарушение дедлайна по причине отсутствия свободных вычислительных ресурсов и/или недоступности компьютеров [7, 8].

Для борьбы с излишне частыми нарушениями дедлайна применяют репликацию подзаданий проекта, когда подзадание с одними и теми же исходными данными рассылается одновременно нескольким участникам [9]. При этом в качестве *канонического* может приниматься результат, полученный первым по времени, либо может использоваться схема кворума, когда необходимо совпадение полученных результатов с заданной точностью от нескольких вычислительных узлов. Репликация позволяет повысить надежность получаемых результатов, в то же время такой метод может увеличивать общее время вычисления проекта. Таким образом, актуальной является задача нахождения условий, при которых репликация не ухудшает ключевых показателей эффективности проекта, среди которых чаще всего рассматривается общее время выполнения проекта [10].

В данной работе рассматривается вероятностная модель выполнения одного проекта на одном сервере ВСПК. Целью работы является нахождение условий для модели ВСПК, при которых репликация приводит к снижению общего времени вычисления проекта. В модели предполагается, что количество участников ВСПК неизменно (новые участники не добавляются, а старые не покидают проект). Кроме того, для простоты предполагается, что все подзадания проекта вычисляются одно и то же детерминированное время (*такт*, равный дедлайну). Это упрощение справедливо для тех систем, в которых проект состоит из решения достаточно вычислительноемких подзаданий при условии, что для различных исходных данных время вычисления подзаданий различается незначительно. Предполагается

также, что причиной остановки подзадания на узле является нарушение дедлайна (по причине отсутствия свободного времени для вычислений или в связи с выключением узла). Таким образом, в рассматриваемой модели за один вычислительный такт каждое подзадание гарантированно либо получает верный результат, либо останавливается в связи с истечением времени, отведенного на расчет. В случае остановки подзадания сервер ВСПК вынужден отправлять подзадание с теми же исходными данными на повторный расчет.

1. Оценка необходимости репликации

Пусть $p \in (0, 0.5)$ есть вероятность остановки одного подзадания. Для снижения вероятности остановки подзадания в процессе вычислений сервер может посылать одно и то же подзадание ν узлам одновременно. При этом для получения канонического результата сервер ожидает кворум из $\nu_0 \geq 1$ ответов, полученных до дедлайна (в случае $\nu_0 = 1$ сервер ожидает любой из ответов, полученный первым по времени). Тогда вероятность успешного достижения кворума за один такт равна

$$q_* := \sum_{i=\nu_0}^{\nu} \binom{\nu}{i} p^{\nu-i} (1-p)^i \geq (1-p) =: q.$$

Отметим, что репликация замедляет вычисления в ν раз, однако при этом вероятность удачного вычисления подзадания за такт возрастает. Поскольку сервер отправляет остановленные подзадания повторно неограниченное число раз, то процесс вычисления подзадания представляет собой классическую схему Бернулли. При этом среднее число передач одного и того же подзадания до первого успеха составит $1/q$ тактов для схемы без репликации и $1/q_*$ тактов для схемы с репликацией. С учетом замедления вычислений при использовании репликации, разность между средним временем вычисления N подзаданий проекта в двух рассматриваемых схемах равна

$$\delta(q, \nu, \nu_0) := \frac{N}{q} - \frac{\nu N}{q_*}.$$

Тогда репликация была бы оправдана для тех значений p, ν, ν_0 , для которых $\delta(q, \nu, \nu_0) > 0$, т. е.

$$(1) \quad q_* - \nu q > 0.$$

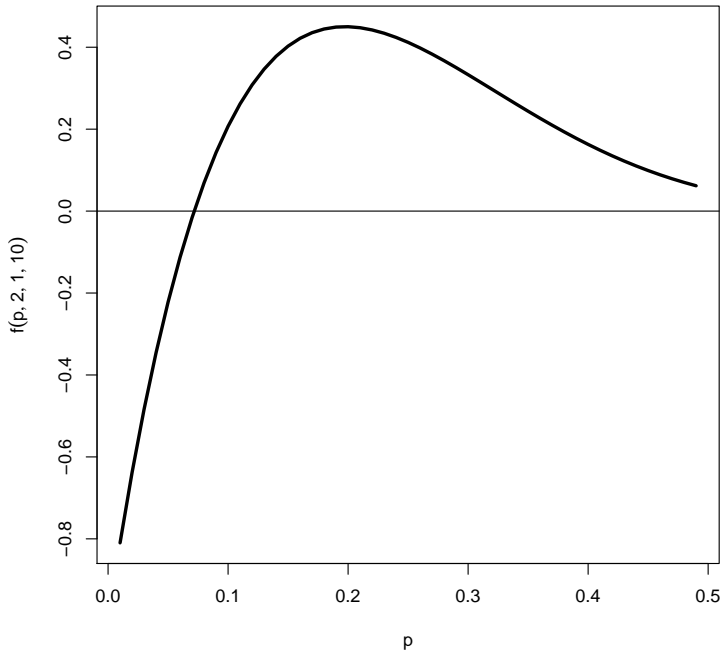


Рис. 1. Значения функции $f(p, \nu, \nu_0, n)$ при $\nu = 2$, $\nu_0 = 1$, $n = 10$ для $0 \leq p \leq 0.5$; положительность функции означает оправданность репликации

Однако, поскольку $q \geq 0.5$, $\nu \geq 2$, $q_* \leq 1$, то в области допустимых значений параметров нет значений, удовлетворяющих неравенству (1). Таким образом, репликация в данном случае невыгодна.

Рассмотрим модификацию модели, в которой в целях экономии трафика сервер при отправке подзаданий группирует подзадания в пачки по n штук с одними и теми же исходными данными, при этом ненулевое число остановок в пачке приводит к повторной отправке целой пачки (по аналогии с моделью, предложенной в [11], где ошибка одного из процессов в параллельно выполняемой задаче на вычислительном кластере приводила к остановке всей задачи). Вероятность успешного завершения вычисления пачки за один такт есть q^n ,

а среднее число тактов до удачного вычисления всей пачки равно $1/q^n$. Рассмотрим одновременно схему с репликацией, предполагая, что сервер ожидает кворум из не менее, чем ν_0 ответов, в каждом из n подзаданий пачки. Тогда среднее число повторных передач в схеме с репликациями равно $1/q_*^n$. С учетом замедления вычислений, по аналогии с неравенством (1), репликация при отправке пачек подзаданий оправдана для тех значений параметров p, ν, ν_0, n , для которых

$$(2) \quad f(p, \nu, \nu_0, n) := q_*^n - q^n \nu > 0.$$

Множество решений неравенства (2) для $n \geq 2$ не пусто. Для иллюстрации соотношения (2) на рис. 1 приведена зависимость $f(p, \nu, \nu_0, n)$ для $\nu = 2, \nu_0 = 1, n = 10$ при $0 \leq p \leq 0.5$.

В качестве развития модели можно предложить нахождение в явном виде асимптотических неравенств для вероятности остановки вычисления p при большом размере пачки n . Также отметим, что стратегия получения канонического результата в случае вычисления пачки может быть иной, а именно, сервер может ожидать окончания вычисления пачек целиком, и лишь затем сравнивать результаты внутри пачек [12].

2. Репликация при штрафе за ошибку

Вернемся к условиям модели без использования пачек. Пусть также $\nu_0 = 1$, т. е. сервер ожидает любой ответ в качестве канонического. Предположим, что в случае остановки расчета всеми ν репликациями сервер вынужден выполнять дополнительную работу (по пересылке подздания новому вычислительному узлу, обновлению базы данных и т. п. [7]) длительностью M тактов. Обозначим $p_\nu := p^\nu$. Вычислим среднее время выполнения одного подздания проекта с учетом за-

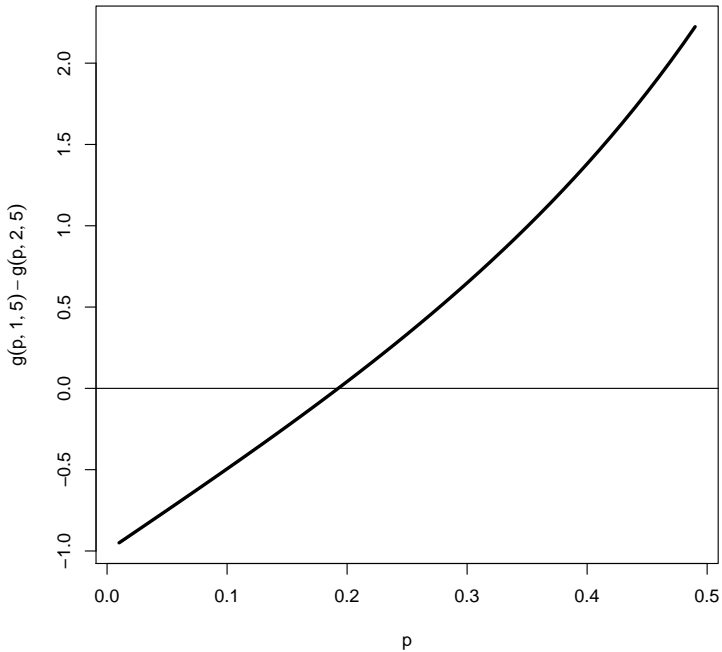


Рис. 2. Значения разности функций $g(p, 1, M) - g(p, \nu, M)$ при $\nu = 2$, $M = 10$ для $0 \leq p \leq 0.5$; положительность функции означает оправданность репликации

медления и штрафа:

$$\begin{aligned}
 g(p, \nu, M) &:= \sum_{j=0}^{\infty} \nu(jM + 1)p_{\nu}^j(1 - p_{\nu}) = \\
 &= \nu(1 - p_{\nu}) \left(M \sum_{j=1}^{\infty} j p_{\nu}^j + \sum_{j=0}^{\infty} p_{\nu}^j \right) = \\
 &= \nu(1 - p_{\nu}) \left(M p_{\nu} \frac{\partial}{\partial p_{\nu}} \sum_{j=0}^{\infty} p_{\nu}^j + \frac{1}{1 - p_{\nu}} \right) = M \nu \frac{p_{\nu}}{1 - p_{\nu}} + \nu.
 \end{aligned}$$

Тогда репликация целесообразна для тех p, ν, M , для которых верно неравенство $g(p, 1, M) - g(p, \nu, M) > 0$, т. е.

$$(3) \quad 1 - \nu + M \left(\frac{p}{1-p} - \frac{p^\nu}{1-p^\nu} \right) > 0.$$

Для иллюстрации соотношения (3) на рис. 2 приведена зависимость $g(p, 1, M) - g(p, \nu, M)$ для $\nu = 2, M = 5$ при $0 \leq p \leq 0.5$.

3. Заключение

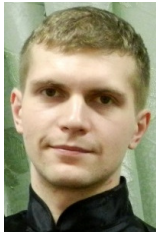
В данной работе рассмотрена вероятностная модель выполнения одного проекта на одном сервере ВСПК. В неявном виде найдены условия необходимости репликации в случае вычисления пачки подзаданий и в случае штрафа за остановку вычисления подзадания.

Список литературы

- [1] D. P. Anderson, C. Christensen, B. Allen *Designing a runtime system for volunteer computing* // Proceedings of the SC 2006 Conference: ACM/IEEE, 2006. (english) ↑175
- [2] Е. Е. Ивашко, Н. Н. Никитина *Организация квантовохимических расчетов с использованием пакета Firefly в гетерогенной Грид на базе VOINC* // Труды Международной суперкомпьютерной конференции Научный сервис в сети Интернет: эксафлопсное будущее, 2011, с. 178–181. (russian) ↑175
- [3] Н. Н. Никитина, Е. Е. Ивашко *Использование VOINC-грид в вычислительноемких научных исследованиях* // Вестник НГУ. Информационные технологии, 2013. Т. 11, № 1, с. 53–57 (russian). ↑175
- [4] Е. Е. Ивашко, А. С. Головин *Методы Data Mining для анализа больших массивов данных в гетерогенной грид на базе VOINC* // Труды Международной суперкомпьютерной конференции «Научный сервис в сети Интернет: поиск новых решений», 2012, с. 196–199. (russian) ↑175
- [5] Е. Е. Ивашко *Выполнение вычислительноемких научных исследований в использовании VOINC-ГРИД* // Материалы 2-й научно-практической школы-семинара молодых ученых: Тольятти: ТГУ, 2012, с. 55–62. (russian) ↑175
- [6] Е. Е. Ивашко, А. С. Головин *Вычислительная эффективность VOINC-GRID* // Proceedings of 2nd International Conference on High Performance Computing HPC-UA 2012, 2012, с. 183–187. (russian) ↑175
- [7] D. Kondo, M. Tauffer, C. L. Brooks, H. Casanova, A. A. Chien *Characterizing and evaluating desktop grids: An empirical study* // Proceedings of the 18th International Parallel and Distributed Processing Symposium, 2004. (english) ↑176, 179
- [8] D. Kondo, G. Fedak, F. Cappello, H. Casanova, A. A. Chien *Characterizing resource availability in enterprise desktop grids* // Future Generation Computer Systems, 2007. Vol. 23, p. 888–903 (english). ↑176

- [9] D. P. Anderson *BOINC: A system for public-resource computing and storage* // Proceedings of the Fifth IEEE/ACM International Workshop on Grid Computing, 2004, p. 4–10. (english) ↑176
- [10] G. D. Ghare, S. T. Leutenegger *Improving Speedup and Responce Times by Replicating Parallel Programs on a SNOW* // Job Scheduling Strategies for Parallel Processing. Lecture Notes in Computer Science: Springer Berlin Heidelberg, 2005. Vol. **3277**, p. 264–287. (english) ↑176
- [11] E. Heien, D. LaPine, D. Kondo, B. Kramer, A. Gainaru, F. Cappello *Modeling and tolerating heterogeneous failures in large parallel systems* // International Conference for High Performance Computing, Networking, Storage and Analysis (SC), 2011, p. 1–11. (english) ↑178
- [12] M. Silberstein, A. Sharov, D. Geiger, A. Schuster *GridBot: execution of bags of tasks in multiple grids* // Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis: ACM, 2009, p. 11. (english) ↑179

Об авторе:



Александр Сергеевич Румянцев

К.ф.-м.н., младший научный сотрудник лаборатории телекоммуникационных систем.

e-mail:

ar0@krc.karelia.ru

Образец ссылки на эту публикацию:

А. С. Румянцев. *Задача оптимизации времени выполнения проекта в вычислительной сети из персональных компьютеров* // Программные системы: теория и приложения: электрон. научн. журн. 2014. Т. 5, № 1(19), с. 175–182.

URL: http://psta.psiras.ru/read/psta2014_1_175-182.pdf

A. S. Rumyantsev. *Optimizing the Execution Time of a Desktop Grid Project.*

ABSTRACT. A model of computation time of a project in a Desktop Grid is viewed, under the following restrictions: the number of nodes is stationary, it takes each task the same time equal to deadline to complete, under the condition of deadline violation the task is to be calculated again. An inequality connecting probability of a single error in calculation of a task, number of replicas and number of tasks is concluded that shows when replication is necessary. An inequality connecting number of replicas, probability of a single error and penalty for error is concluded, which shows when replication is profitable. (in Russian).

Key Words and Phrases: Desktop Grid, stochastic modeling, execution time.