

ПАРАЛЛЕЛЬНЫЙ МЕТОД ДЛЯ РАСЧЕТА НАДЕЖНОСТИ СЕТЕЙ С ОГРАНИЧЕНИЕМ НА ДИАМЕТР

С. Н. Нестеров, Д. А. Мигов*

Новосибирский государственный университет,
630090, Новосибирск, Россия

*Институт вычислительной математики и математической геофизики СО РАН,
630090, Новосибирск, Россия

УДК 519.17+519.24

В статье рассматривается задача точного расчета структурной надежности сети с ненадежными ребрами и абсолютно надежными вершинами. В качестве показателя надежности используется вероятность связности соответствующего случайного графа с ограничением на диаметр. Точный расчет данного показателя — NP-трудная задача, что делает его затруднительным для сетей реальной размерности. Предлагается параллельная реализация для известного метода факторизации и для его модификации, основанной на формировании всех путей между каждой парой полюсов. Разработаны два параллельных алгоритма для расчета на суперЭВМ с распределенной памятью. Результаты численных экспериментов позволили настроить параметры параллельных алгоритмов оптимальным образом и значительно повысить масштабируемость.

Ключевые слова: надежность сети, случайный граф, диаметр графа, параллельные алгоритмы.

We consider the problem of exact calculation of structural reliability for networks with unreliable communication links and perfectly reliable nodes. As reliability index we use the diameter constrained reliability. The problem of computing this characteristic is known to be NP-hard, just like the problem of computing the probability of network connectivity. For solving this problem we propose the parallel methods for supercomputers with distributed memory. These methods are based on the well-known factoring method and modification of factoring method. The analysis of the numerical experiments allowed us to set some important parameters of the algorithms for increasing their scalability.

Key words: network reliability, random graph, graph diameter, parallel algorithm.

Введение. В процессе эксплуатации элементы сетей связи могут подвергаться отказам. Причины отказов могут быть различны, например, поломка, износ, природные явления, результат преднамеренных действий злоумышленника, и другие. Подобные сети обычно моделируются случайным графом $G = (V, E)$ [1], вершины которого соответствуют узлам сети, а ребра — каналам связи. Для каждого элемента графа задано некоторое вещественное число $p \in [0, 1]$ — вероятность исправной работы элемента, что соответствует надежности соответствующего элемента сети.

Наиболее часто рассматривается случай, когда все узлы в сети абсолютно надежны, в то время как для каждого ребра $e \in E$ задана вероятность отказа $q_e = 1 - p_e$. Предполагается, что отказы ребер происходят независимо. Сеть считается работоспособной, если она

связна, т. е. если ее узлы связаны друг с другом посредством исправных ребер. Надежностью сети называют вероятность соответствующего события, т. е. вероятность связности графа. Часто рассматривается выделенное множество узлов сети — терминалов (плюсов). В таком случае сеть полагается работоспособной, если каждая пара терминалов связана посредством исправных ребер. Задача расчета вероятности связности заданного подмножества узлов сети является NP-трудной [1]. Данный показатель надежности достаточно хорошо изучен, разработано большое количество различных точных и приближенных методов расчета [1–4].

Однако на практике во многих случаях требуется обеспечить не просто существование пути между каждой парой узлов, а существование пути, проходящего по ограниченному числу каналов связи [5–6]. Например, если есть ограничение на время передачи данных между двумя узлами — T , то количество транзитных узлов, участвующих в передаче данных, не должно превышать T/t , где t — требуемое время для обработки данных на каждом узле сети. Для таких случаев в [7] был введен другой показатель надежности — вероятность связности сети с ограничением на диаметр, $R_K(G, D)$. Этот показатель надежности есть вероятность того, что любые два узла из заданного множества K соединены в графе G путем длины не более d .

Как и задача расчета вероятности связности, задача расчета вероятности связности с ограничением на диаметр NP-трудна [7]. Вопрос о сложности данной задачи в зависимости от значений диаметра и количества полюсов является к настоящему моменту полностью изученным [8]. В наших прошлых работах совместно с А. С. Родионовым [9–11] были предложены различные методы структурной редукции и декомпозиции для ускорения расчета надежности сети с ограничением на диаметр.

В настоящей работе представлены и исследованы два параллельных алгоритма для вычисления точного значения надежности заданной сети с ограничением на диаметр. Один алгоритм основывается на методе факторизации [1], а второй на модификации данного метода, предложенной в [7]. Для распараллеливания использована парадигма параллельного программирования „ведущий-ведомый“, подобно тому, как это было ранее сделано для разработки параллельного алгоритма расчета надежности сети без ограничения на диаметр [4]. Проведенные численные эксперименты позволили значительно улучшить масштабируемость алгоритмов.

1. Основные определения и обозначения. Будем представлять сеть с ненадежными ребрами с помощью неориентированного случайного графа $G = (V, E)$, для каждого ребра e которого задана вероятность его присутствия в графе r_e . Задано выделенное множество вершин $K \subseteq V$ — полюсов, соответствующих узлам сети, для которых необходимо обеспечить возможность устанавливать соединение друг с другом.

Элементарным событием будем называть частную реализацию графа, определяемую присутствием или отсутствием каждого ребра. Таким образом, общее количество элементарных событий составляет $2^{|E|}$. Вероятность элементарного события равна произведению вероятностей присутствия исправных ребер, умноженному на произведение вероятностей отсутствия отказавших ребер.

Вероятность связности графа G с ограничением на диаметр D определяется как сумма вероятностей всех частных реализаций графа, в которых любая пара вершин из множества K связана не более чем D ребрами. Далее будем называть эту характеристику надежностью графа или соответствующей сети и обозначать как $R_K(G, D)$. Под путем понимается путь без самопересечений, под длиной пути в графе — количество ребер в этом пути.

2. Метод факторизации. Расчет $R_K(G, D)$ непосредственно по определению приведет к перебору всех реализаций графа, что делает расчет невозможным даже при небольшой размерности. Поэтому для расчета различных показателей надежности используются другие методы, самый распространенный из которых — метод факторизации [1]. Метод заключается в рекурсивном применении формулы полной вероятности при рассмотрении в качестве альтернативных гипотез наличия либо отсутствия очередного разрешающего ребра. В общем случае формула для выбранного ребра e имеет вид

$$R_K(G, D) = r_e \times R_K(G \setminus e, D) + (1 - r_e) \times R_K(G/e, D), \quad (1)$$

где G/e — граф G без ребра e , $G \setminus e$ — граф G , в котором ребро e абсолютно надежно. Множества вершин V и терминалов K остаются теми же.

В [7] предложен модифицированный метод ветвления для расчета надежности сети, который работает существенно быстрее, чем вышеописанный классический метод ветвления (1) в случае с ограничением на диаметр. Далее будем обозначать этот метод как C&P. Основная его идея состоит в переходе от графов к спискам путей P . На начальном этапе формируется список всех путей длины, не превосходящей D для каждой пары полюсов. Это автоматически исключает из рассмотрения все ребра, через которые не проходит ни один такой путь, например, так называемые „прикрепленные деревья“ без полюсов. Для каждого оставшегося ребра e формируется список $P(e)$ всех содержащих его путей. При рекурсивных вызовах процедуры в качестве параметров не передаются подграфы, вместо них передается шесть параметров, описывающих состояние соответствующего подграфа с точки зрения путей из P . Ниже приведены псевдокод этого алгоритма и описание используемых в нем вспомогательных параметров.

- 1) np_{st} — количество путей длины не более D между полюсами s и t ;
- 2) $links_p$ — число ребер в пути $p \in P$, не являющихся абсолютно надежными;
- 3) $feasible_p$ — принимает значение *false*, если путь p включает абсолютно ненадежное ребро, *true* — иначе;
- 4) $connected_{st}$ — принимает значение *true*, если вершины s и t соединены абсолютно надежным путем длины не более D , *false* — иначе;
- 5) $connectedPairs$ — число пар полюсов, связанных абсолютно надежными путями ограниченной длины.

Псевдокод алгоритма Cancela и Petingi [8].

```

1: function ФАКТО( $P, np_{st}, links_p, feasible_p, connected_{st}, connectedPairs$ )
2:    $RContract \leftarrow 0$                                 ▷ случай, отвечающий стягиванию ребра
3:    $RDelete \leftarrow 0$                                 ▷ случай, отвечающий удалению ребра
4:    $e \leftarrow$  случайное ребро :  $0 < r_e < 1$ 
5:   for all  $p = (s, \dots, t) \in P$ , где  $feasible_p = true$  do                                ▷ "Стягивание ребра"
6:      $links_p \leftarrow links_p - 1$ 
7:     if  $connected_{st} = false$  и  $links_p = 0$  then
8:        $connected_{st} \leftarrow true$ 
9:        $connectedPairs \leftarrow connectedPairs + 1$ 
10:      if  $connectedPairs = \frac{k \times (k-1)}{2}$  then
11:         $RContract \leftarrow 1$ 
12:        GO TO ("Удаление ребра")
13:      end if

```

```

14:     end if
15: end for
16: RContract ← FACTO( $P, np_{st}, links_p, feasible_p, connected_{st}, connectedPairs$ )
17: for all  $p = (s, \dots, t) \in P$ , где  $feasible_p = true$  do           ▷ "Удаление ребра"
18:      $feasible_p \leftarrow false$ 
19:      $np_{st} \leftarrow np_{st} - 1$ 
20:     if  $connected_{st} = false$  и  $links_p = 0$  then
21:          $feasible_p \leftarrow false$ 
22:          $np_{st} \leftarrow np_{st} - 1$ 
23:         if  $np_{st} = 0$  then
24:              $RDelete \leftarrow 0$ 
25:             GOТO("Финальные вычисления")
26:         end if
27:     end if
28: end for
29: RDelete ← FACTO( $P, np_{st}, links_p, feasible_p, connected_{st}, connectedPairs$ )
30:  $R_K(G, D) = r_e \times RContract + (1 - r_e) \times RDelete$            ▷ "Финальные вычисления"
31: Return  $R_K(G, D)$ 
32: end function

```

3. Параллельный алгоритм расчета надежности сети с ограничением на диаметр. Основная идея предлагаемых параллельных алгоритмов заключается в следующем. В процессе факторизации при разбиении задачи на две новые подзадачи одна из них может передаваться для решения на свободный процесс, а вторая оставаться на исходном. Если свободных процессов нет, то обе подзадачи остаются на исходном процессе. Например, подзадача „удаления“ выбранного ребра остается на текущем процессе, а подзадача „стягивания“ этого ребра или отправляется на любой из простаивающих процессов, если такой есть, или остается на этом же процессе. Разница для двух модификаций метода факторизации заключается только в формате передаваемых данных.

В таком виде данный подход имеет существенные недостатки. Поскольку алгоритм основан на формуле факторизации, вычисленное в процедуре значение необходимо отправить на родительский процесс для умножения этого значения на соответствующую вероятность и суммирования. Это может приводить к дополнительным пересылкам и простоям процессов. Например, первый процесс выполняет „удаление“ выбранного ребра e , вычисляя $R_K(G/e, D)$, а на другом процессе выполняется „стягивание“ для этого ребра, вычисляя $R_K(G \setminus e, D)$. И первый процесс должен будет ждать, пока второй процесс вычислит второе значение и перешлет его, и только после этого вычислит $R_K(G, D)$.

По аналогии с параллельным методом для расчета всетерминальной надежности без ограничения на диаметр [4] предлагается отправлять на процесс-помощник подзадачу вместе с вероятностью получения этой подзадачи. Вероятность получения реализации подсети — переменная, значение которой меняется в течение процесса. Для исходной сети это 1. Например, в выражении (1) вероятность подзадачи стягивания — это r_e , а вероятность подзадачи удаления равна $1 - r_e$. При этом нужно учесть, что исходная задача тоже имеет определенную вероятность, на которую нужно умножить приведенные значения. Данный подход помогает избежать обратных посылок значений надежности, накапливая на каждом процессе свое собственное значение. В конце все эти значения суммируются на отдельном процессе и получается точное значение надежности.

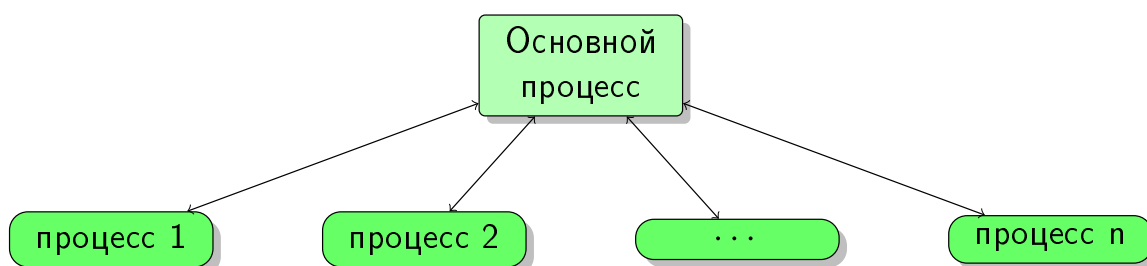


Рис. 1. Схема „ведущий — ведомый“

Для организации взаимодействия между процессами использована схема „ведущий — ведомый“ (рис. 1), в которой выделяется один „контролирующий“ процесс, называемый ведущим. Этот процесс ответственен за загрузку остальных и суммирование полученных значений. Ведомый процесс получает задачи от других ведомых процессов по указанию ведущего, выполняет их, возвращает полученные результаты, и снова получает новую задачу. При необходимости ведомый процесс может попросить помощи в виде одного из простаивающих процессов для распределения работы.

Ведущий процесс:

- 1) контролирует загруженность остальных процессов;
- 2) отправляет исходный граф для вычисления на один из ведомых процессов;
- 3) при получении запроса о помощи либо отказывает в ней, либо посылает номер свободного процесса;
- 4) суммирует вычисленные на каждом процессе значения, получая итоговое значение надежности.

Ведомый процесс:

- 1) в начале инициализируется как простаивающий;
- 2) при получении задачи процесс выполняет процедуру (1). Одна из подзадач, например, $R_K(G/e, D)$, остается для вычислений на этом процессе, после чего происходит запрос к ведущему процессу, от которого приходит ответ либо в виде номера свободного процесса, либо с сообщением отказа (т. е. если все процессы уже заняты). Далее, в зависимости от полученного ответа, вторая подзадача вместе с соответствующей вероятностью либо отправляется на вспомогательный процесс, либо остается на исходном для дальнейших вычислений;
- 3) после того, как все вычисления закончены, отправляет накопленные значения на ведущий процесс и ждет следующей задачи.

Как уже упоминалось, разница для двух модификаций метода факторизации заключается только в формате передаваемых данных. В модификации метода факторизации [7] работа ведется со списком структур, а не с конкретным графом. Необходимо оперировать с гораздо большим по размеру количеством информации. Пересылка подзадач от одного процессора к другому осуществляется за намного большее время, чем, в базовом методе факторизации. Но в последовательной реализации базовой метод факторизации значительно медленней, чем модифицированный [7] и дополнительно усиленный в соответствии с [11]. В следующем пункте выясняется, какой метод предпочтительней в параллельной реализации.

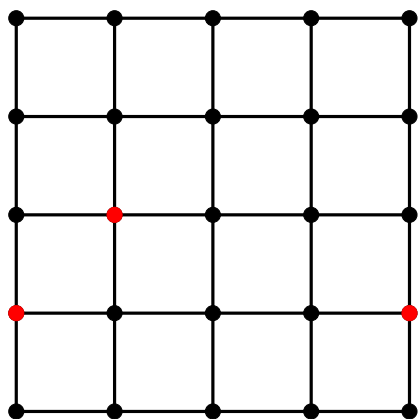


Рис. 2. Решетка для метода факторизации

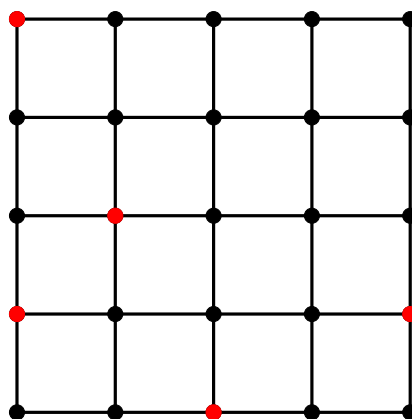


Рис. 3. Решетка для метода С&Р

4. Результаты численных экспериментов. Для сравнения масштабируемости двух параллельных алгоритмов (метод факторизации и метод С&Р) была выбрана решетка 5×5 с тремя терминалами (рис. 2–3), терминалы выделены цветом). Диаметр полагался равным 4. Значение надежности последовательным методом факторизации было вычислено за 16 мин 9,507 с, количество рекурсий было равным 249 602 055. В то же время второй алгоритм закончил работу всего за 0 мин 0,061 с с 74 рекурсиями. Поэтому для второго алгоритма были усложнены входные данные с целью сделать время исполнения обоих алгоритмов примерно одинаковым. Были выбраны следующие параметры: количество терминалов равно 5, значение диаметра равно 9. С новыми данными алгоритм С&Р закончил вычисления за 16 мин 35,169 с, количество рекурсий оказалось равным 1 154 905 688.

На рис. 4 изображен график масштабируемости данных алгоритмов. Можно видеть, что алгоритмы масштабируются примерно одинаково, то есть в обоих случаях ускорение идет лишь до 16 ядер, после чего время работы замедляется. В то же время следует помнить, что вычисления были сделаны для разных входных данных, то есть метод факторизации в действительности работает гораздо медленнее, чем С&Р.

В работе [4] для ускорения параллельного расчета всетерминальной надежности без ограничения на диаметр был введен параметр N_{edges} — минимальное число ребер в графе, необходимое для отправления этого графа на вспомогательный процесс. Например, для графа малой размерности быстрее посчитать надежность на одном процессе, без пересылок. Результаты численных экспериментов показали, что этот параметр существенно влияет на скорость работы алгоритма. После введения параметра N_{edges} результаты масштабируемости изменились в лучшую сторону. На рис. 5 и 6 можно видеть, что ускорение алгоритмов заметно улучшилось для обоих алгоритмов. Поиск оптимального значения параметра N_{edges} велся методом бинарного поиска. Видно, что итоговое значение лежит между значениями 15 и 20 (дальнейший поиск параметра бессмысленен, так как разница времени исполнения для параметров 15 и 20 отличалась лишь на одну секунду).

Таким образом, эксперименты показали, что алгоритмы масштабируются одинаково, и от использования алгоритма факторизации можно отказаться в пользу улучшенного алгоритма С&Р, так как метод факторизации работает гораздо медленнее для одинаковых входных данных.

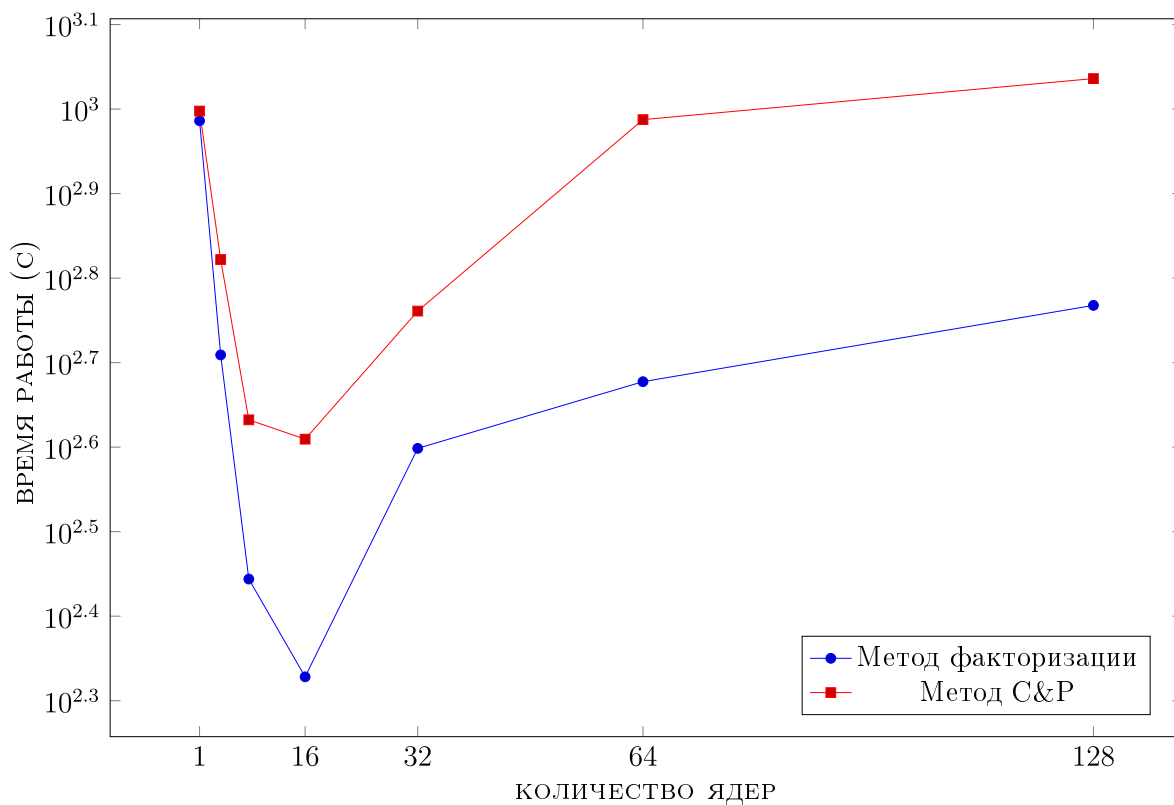


Рис. 4. Зависимость времени исполнения от количества ядер

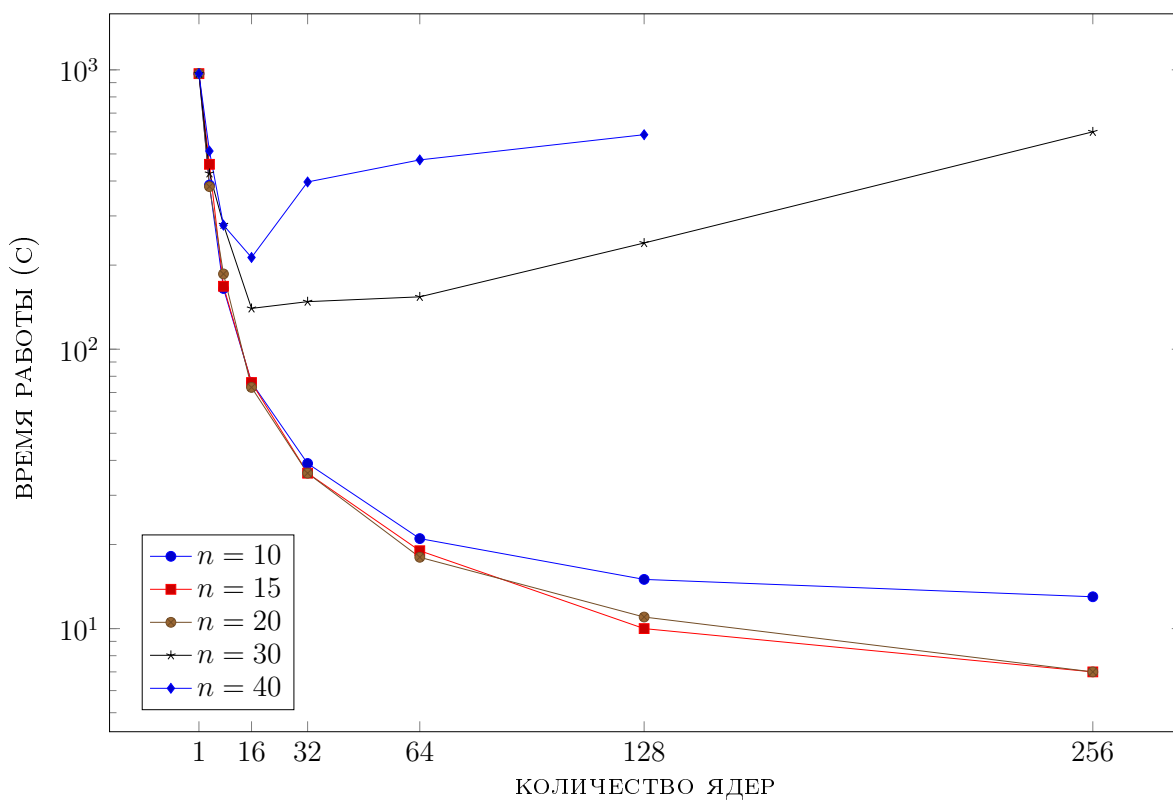


Рис. 5. Зависимость масштабируемости от параметра N_{edges} для метода факторизации

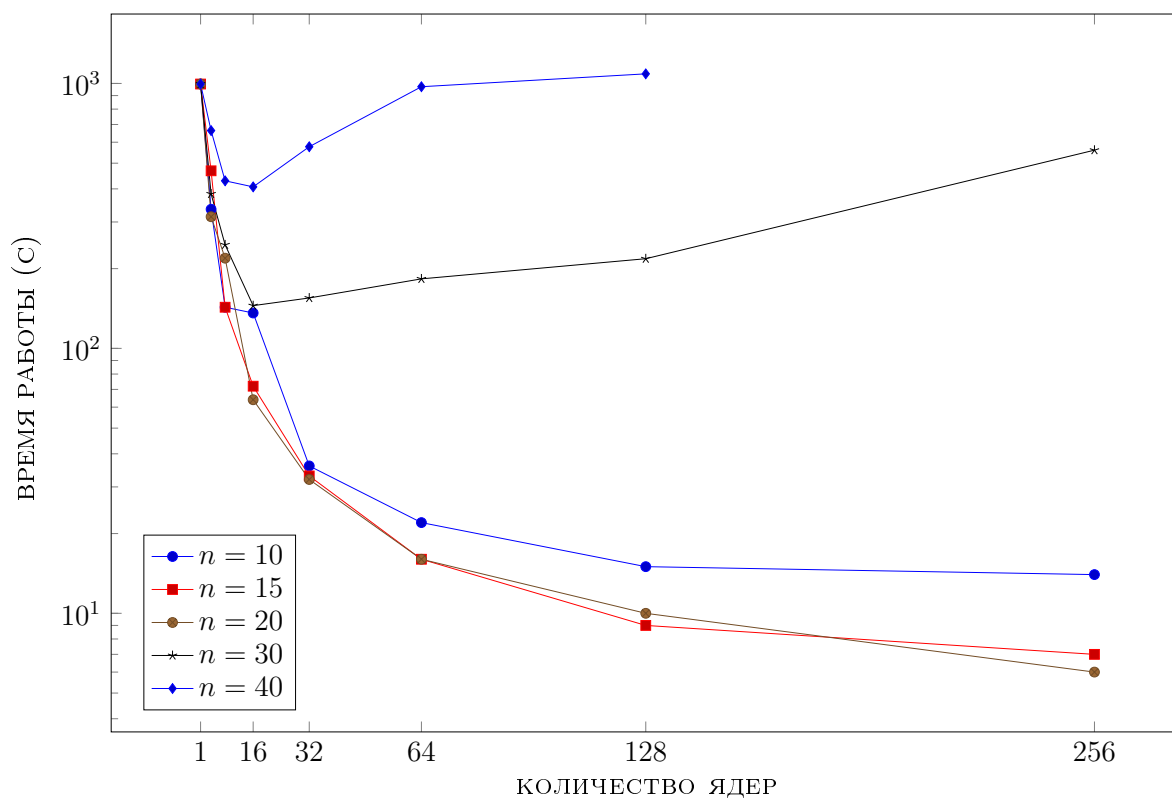


Рис. 6. Зависимость масштабируемости от параметра N_{edges} для метода С&Р

Заключение. Для NP-трудной задачи расчета надежности сети с ограничением на диаметр предложены параллельный алгоритм, основанный на методе факторизации, и параллельный алгоритм, основанный на модификации метода факторизации С&Р. Для распараллеливания использована парадигма параллельного программирования „ведущий — ведомый“. Численные эксперименты позволили оптимально настроить параметры алгоритмов и значительно улучшить их масштабируемость. Как и в последовательном случае, метод С&Р работает существенно быстрее в параллельной реализации, чем базовый метод факторизации.

Список литературы

1. COLBOURN CH. J. The combinatorics of network reliability. N.Y.: Oxford University Press, 1987.
2. Цициашвили Г. Ш., Осипова М. А., Лосев А. С. Асимптотика вероятности связности графа с низконадежными ребрами // Прикладная дискретная математика. 2013. № 1(19). С. 93–98.
3. МИГОВ Д. А. Формулы для быстрого расчета вероятности связности подмножества вершин в графах небольшой размерности // Проблемы информатики. 2010. № 2(6). С. 10–17.
4. MIGOV D., RODIONOV A. Parallel Implementation of the Factoring Method for Network Reliability Calculation // Springer Lecture Notes in Computer Science (in [4] 2014, Part 6). 2014. V. 8584. P. 654–664.
5. PANDURANGAN G., RAGHAVAN P., UPFAL E. Building Low-Diameter Peer-to-Peer Networks // IEEE Journal on Selected Areas in Communications (JSAC). 2003. V. 21(6). P. 995–1002.

6. МЕЛЕНТЬЕВ В. А. Функция структурной отказоустойчивости и d -ограниченная компонента связности графа вычислительной системы // Прикладная дискретная математика. 2008. № 2(2). С. 102–106.

7. CANCELA H., PETINGI L. Reliability of communication networks with delay constraints: computational complexity and complete topologies // Int. J. of Mathematics and Mathematical Sciences. 2004. V. 29. P. 1551–1562.

8. CANALE E., CANCELA H., ROBLEDI F., RUBINO G., SARTOR P. On Computing the 2-Diameter-Constrained K -Reliability of Networks // International Transactions in Operational Research. 2013. V. 20(1). P. 49–58.

9. МИГОВ Д. А. Расчет надежности сети с ограничением на диаметр с применением точек сочленения // Автоматика и телемеханика. 2011. № 7. С. 69–74.

10. МИГОВ Д. А. Расчет надежности двухполюсной сети с ограничением на диаметр с применением сечений // Проблемы информатики. 2011. № 11. С. 4–9.

11. МИГОВ Д. А., НЕСТЕРОВ С. Н., РОДИОНОВ А. С. Методы ускорения расчета надежности сетей с ограничением на диаметр // Вестник СибГУТИ. 2014. № 1. С. 49–56.

*Нестеров Сергей Николаевич —
студент Новосибирского государственного университета,
e-mail: sergea@inbox.ru.*

*Мигов Денис Александрович —
канд. физ.-мат. наук, науч. сотр. лаборатории моделирования
динамических процессов в информационных сетях Института
вычислительной математики и математической геофизики СО РАН,
e-mail: mdinka@rav.sccc.ru.*

Дата поступления — 10.07.2015