Алгоритм решения задачи связности в условиях наличия неисправных каналов связи для детерминированной маршрутизации, основанной на правиле порядка направлений, в сети с топологией «многомерный тор»

Пожилов И. А.

ОАО «НИЦЭВТ»

В статье рассматриваются высокоскоростные коммуникационные сети с топологией «многомерный тор» и детерминированная маршрутизация в них, основанная на правиле порядка направлений. Предполагается наличие некоторого множества каналов связи, вышедших из строя. Ставится задача связности для данной сети: проверка наличия детерминированного маршрута для каждой пары узлов, отвечающего правилам маршрутизации и не содержащего отказавших каналов связи. Предлагается алгоритм решения данной задачи для произвольно заданной маршрутизации. При помощи построенного алгоритма проводится исследование двух подходов к построению различных путей между парой вершин: неминимальная детерминированная маршрутизация с порядком направлений и маршрутизация, основанная на механизме First Step/Last Step. Оценивается вероятность потери связности при заданном количестве отказавших каналов связи и на основе этой зависимости производится оценка эффективности механизма First Step/Last Step. Производится сравнение исследуемых методов маршрутизации с полной маршрутизацией, которая использует все пути, удовлетворяющие правилу порядка направлений, и является теоретическим барьером для маршрутизаций, использующих это правило. Показано, что при использовании метода First Step/Last Step выбранная вероятность потери связности достигается на существенно большем количестве отказавших каналов (до трех раз).

1. Введение

В ОАО «НИЦЭВТ» ведется разработка высокоскоростной коммуникационной сети «Ангара» с топологией «многомерный тор» [1, 2]. В 2013 году был выпущен кристалл маршрутизатора этой сети, к 2015 году ожидается построение суперкомпьютера на его основе. Данная статья открывает цикл работ по исследованию отказоустойчивости сети «Ангара».

Современные суперкомпьютерные системы включают в себя большую совокупность элементов: процессоров, модулей памяти, сетевых устройств, устройств ввода-вывода, связанных в единый программно-аппаратный комплекс. И хотя каждый из этих элементов независимо обладает большой степенью надежности, с ростом количества подобных элементов система в целом обнаруживает тенденцию к возникновению непредвиденных неисправностей. Возникает необходимость контроля за развитием ситуации и поддержания работоспособности системы даже в условиях ограниченной функциональности отдельных ее элементов.

Одной из задач обеспечения отказоустойчивости является сохранение возможности доставки сообщений по коммуникационной сети в обход отказавших каналов связи. В данной статье рассматривается проблема обеспечения отказоустойчивости для детерминированной маршрутизации по порядку направлений (Direction ordered routing, DOR, [3, 4]).

В разделе 2 приводятся необходимые формальные определения. В разделе 3 приводится постановка задачи связности в общем виде. В разделе 4 рассматриваются два алгоритма решения задачи связности, приводятся оценки их эффективности и области применимости.

В разделе 5 дается формальное описание алгоритмов маршрутизации, которые исследованы в разделе 6.

2. Определения

В данном разделе введем некоторые формальные определения, с которыми будет удобно работать далее.

Рассмотрим вычислительную сеть с топологией n-мерный тор. Пусть размерности тора имеют значения $(d_1, d_2, \ldots, d_n)$. Общее количество узлов сети обозначим N_{nodes}.

В этой сети узел u имеет координаты $u = (u_1, u_2, \ldots, u_n)$, где $0 \leq u_j < d_j$. Множество всех узлов сети будем обозначать $\mathcal{N}$.

Соседними в рамках тороидальной топологии будут узлы $u = (u_1, u_2, \ldots, u_n)$ и $\tilde{u} = (u_1, \ldots, (u_j \pm 1) \mod d_j, \ldots, u_n)$ для любого индекса $1 \leq j \leq n$.

Видно, что каждый узел имеет $2n$ соседей (в случае $d_j > 2$). Будем говорить, что каждый из этих соседей находится в одном из *направлений* от узла u. Множество направлений $\overline{1, 2n}$ обозначим $\mathcal{D}$. Направления $1, \ldots, n$ будем называть *положительными*:

$$\tilde{u} = (u_1, \ldots, (u_j + 1) \mod d_j, \ldots, u_n),$$

если $\tilde{u}$ находится в направлении j от u, где $1 \leq j \leq n$.

Направления $n+1, \ldots, 2n$ будем называть отрицательными:

$$\tilde{u} = (u_1, \ldots, (u_j - 1) \mod d_j, \ldots, u_n)$$

в случае $n + 1 \leq j \leq 2n$.

Множество положительных и отрицательных направлений обозначим, соответственно, $\mathcal{D}_+$ и $\mathcal{D}_-$. Выражение «v находится в направлении $D \in \mathcal{D}$ от u» будем обозначать просто как $v = u + D$. Направление противоположное D будем обозначать $-D \in \mathcal{D}$. Индекс вектора координат, который изменяется в данном направлении обозначим $|D| \in \overline{1, n}$.[1)]

Определение 1 Путем P *в сети назовем последовательность вида* $u^0, D_1, u^1, D_2, \ldots, D_N, u^N$, *где* u^j *— узел вычислительной сети.* D_j *— направление, связывающее вершину* u^{j-1} *с вершиной* u^j.

Множество всех путей обозначим $\mathcal{P}$.

Иногда будем опускать указание промежуточных вершин u^j, т. к. они могут быть получены из направлений соответствующих переходов: $u^j = u^{j-1} + D_j$. Т. е.

$$u^0, D_1, u^1, D_2, \ldots, D_N = u^0, D_1, D_2, \ldots, D_N, u^N.$$

Заметим, что правило порядка направлений для детерминированной маршрутизации в данном случае формулируется как $D_{j-1} \leq D_j, j = \overline{2, N}$. Обозначим множество путей, удовлетворяющих правилу порядка направлений, $\mathcal{P}_D(u^0, u^N)$.

3. Постановка задачи связности

Среди всех возможных каналов связи определим *множество отказавших каналов*:

$$F = \left\{(u^j, D_j)\right\}_{j=1 \ldots N_F}.$$

Так как канал связи между двумя узлами функционирует согласно двустороннему протоколу, то разумно предположить, что каналы связи между соседними узлами выходят из строя одновременно, т. е. множество F будет включать в себя отказавшие каналы связи попарно.

[1)] Математически $-D \equiv D + n \mod 2n, |D| \equiv D \mod n$.

Определение 2 *Путь* $P = u^0, D_1, \ldots, D_N, u^N$ *назовем* допустимым*, если ни один из переходов* u^{j-1}, D_j, u^j *не осуществляется по одному из отказавших каналов, т. е.*

$$\forall j \in \overline{1, N} \Rightarrow (u^{j-1}, D_j) \notin F.$$

Множество допустимых путей обозначим $\mathcal{G}(u^0, u^N)$.

Пусть для каждой (упорядоченной) пары узлов (u, v) известно множество путей, которые соединяют их согласно правилам маршрутизации: $R : (u, v) \mapsto 2^{\mathcal{P}_{\mathcal{D}}(u,v)}$.

Определение 3 *Пару узлов* (u, v) *назовем* связанными*, если существует допустимый путь из узла u в узел v согласно правилам маршрутизации:*

$$R(u, v) \cap \mathcal{G} \neq \varnothing.$$

Определение 4 Задача связности *заключается в нахождении множеств*

$$R(u, v) \cap \mathcal{G},$$

т. е. в определении связанности каждой пары узлов и нахождении допустимых путей между ними.

4. Алгоритмы решения задачи связности

В этом разделе рассмотрим два алгоритма решения задачи связности. Они будут отличаться областью применимости. Первый, переборный алгоритм, решает задачу связности в общем виде для произвольной маршрутизации $R(u, v)$. Второй алгоритм решает задачу связности для *полной маршрутизации*, т. е. маршрутизации, включающей все пути, удовлетворяющие правилу порядка направлений.

4.1. Переборный алгоритм

Рассмотрим алгоритм, решающий задачу связности по определению, т. е.

1) Строится полное множество $R(u, v)$ допустимых путей между всеми парами узлов сети.

2) Из множества $R(u, v)$ удаляются пути, содержащие в себе отказавшие каналы связи.

3) Полученное множество является решением задачи связности.

Оценим вычислительную сложность этого алгоритма и покажем, что он применим для решения задачи связности для маршрутизаций $R(u, v)$ с ограничением на количество путей между парой вершин.

Путь $P \in R(u, v)$ представлен в памяти как последовательность отрезков, соединяющих узлы сети, в которых совершаются повороты. Для проверки принадлежности отказавшего канала связи этому пути необходимо проверить его принадлежность каждому из этих отрезков. Эта проверка осуществляется за $O(n^2)$ операций сравнения, т. к. проверяется принадлежность каждому из n отрезков и каждая проверка подразумевает $O(n)$ сравнений, где n — число размерностей тора.

Пусть множество различных путей между двумя вершинами ограничено константой N_{paths}:

$$\forall u, v \Rightarrow |R(u, v)| \leq N_{paths}.$$

Тогда сложность построения всех путей можно оценить как

$$T_{build} = O(N_{nodes}^2 \cdot N_{paths}),$$

здесь предполагается, что $n = O(1) \ll N_{nodes}$, т. е. что число измерений тора ограничено некоторой небольшой константой и мало по сравнению с количеством узлов.

Решение задачи связности для заданного множества отказавших каналов F оценивается как

$$T_{conn} = O(N_{nodes}^2 \cdot N_{paths} \cdot |F|).$$

Для некоторых алгоритмов маршрутизации (см. 5) количество путей между парой вершин $N_{paths} = O(n) = O(1)$, т. е. $N_{paths} \ll N_{nodes}$, откуда можно получить

$$\begin{aligned} T_{build} &= O(N_{nodes}^2), \\ T_{conn} &= O(N_{nodes}^2 \cdot |F|). \end{aligned} \tag{1}$$

Еще раз заметим, что оценки (1) опираются на ограниченность количества путей между парой вершин. Не все алгоритмы маршрутизации обладают этим свойством. Если количество путей увеличивается с ростом количества узлов, то необходим другой алгоритм решения задачи связности.

4.2. Алгоритм для полной маршрутизации

В случае *полной маршрутизации* количество путей между двумя вершинами будет увеличиваться с ростом количества узлов, что приведет к ухудшению оценки (1). В этом разделе рассмотрим алгоритм, решающий эту проблему в случае, если маршрутизация $R(u, v)$ включает в себя *все* пути, удовлетворяющие правилу порядка направлений:

$$R(u, v) = \mathcal{P}_{\mathcal{D}}(u, v).$$

Решение задачи связности в этом случае сведем к решению задачи о существовании пути для некоторого ориентированного графа $G(V, E)$. Построим граф G.

Множество вершин графа G имеет вид

$$V := \mathcal{N} \times \mathcal{D}.$$

Здесь вершины (u, d_0) соответствуют узлам u, путь до которых включает движение только по направлениям $d : 1 \le d \le d_0$.

Множество дуг включает дуги двух видов:

$$E := A \cup B,$$

где A — множество дуг, отвечающих каналам связи вдоль определенного направления:

$$A := \{((u, d), (u + d, d)) \mid u \in \mathcal{N}, d \in \mathcal{D}, (u, u + d) \notin F\},$$

где F — множество отказавших каналов связи; B — множество *тривиальных дуг*, отвечающих переходу к следующему направлению без изменения текущей вершины:

$$B := \{((u, d), (u, d + 1)) \mid u \in \mathcal{N}, d \in \overline{1, 2n - 1}\}.$$

Сведение задачи связности сети к задаче поиска пути в графе дает следующая теорема.

Теорема 1 *Путь из узла $u^0 \in \mathcal{N}$ в узел $v^0 \in \mathcal{N}$, удовлетворяющий правилу порядка направлений и не содержащий отказавших узлов существует в том и только в том случае, если в графе G существует путь из вершины $(u^0, 1)$ в вершину $(v^0, 2n)$.*

Доказательство. Для доказательства приведем взаимнооднозначное соответствие между множеством путей в сети согласно определению 1 и множеством путей в графе G. Рассмотрим путь

$$P = u^0, D_1, u^1, D_2, \ldots, D_N, u^N = v^0,$$

где $D_{j-1} \le D_j, i = \overline{2, N}$.

Построим соответствующий путь в графе G:

$$(u^0, 1), \ldots, (u^0, D_1), (u^1, D_1), \ldots, (u^1, D_2), (u^2, D_2), \ldots, (u^{N-1}, D_N), (u^N, D_N), \ldots, (u^N, 2n).$$

$$\underbrace{(u^0, 1), \ldots, \mathrlap{\overbrace{}^{\text{Ребро из } A}}(u^0, D_1)}_{\text{Ребра из } B}, \underbrace{(u^1, D_1), \ldots, \mathrlap{\overbrace{}^{\text{Ребро из } A}}(u^1, D_2)}_{\text{Ребра из } B}, (u^2, D_2), \ldots, \overbrace{(u^{N-1}, D_N), \mathrlap{\underbrace{}_{\text{Ребра из } B}}(u^N, D_N)}^{\text{Ребро из } A}, \ldots, (u^N, 2n).$$

Аналогичным образом можно по данному пути в графе G построить путь в сети согласно определению 1. Это соответствие показывает, что существование пути в сети из u^0 в v^0 равносильно существованию пути в графе G из $(u^0, 1)$ в $(v^0, 2n)$.

►

Заметим, что в указанном построении одному каналу связи сети соответствует единственное ребро графа G, это сводит задачу отказоустойчивости для полной маршрутизации полностью на язык теории графов и делает исследования удообными.

Для решения задачи связности при помощи графа G применим метод поиска вширь: для каждой из вершин $(u_0, 1)$ определим множество достижимых из нее вершин вида $(v, 2n)$. При этом соответствующие узлы v будут достижимы из u_0 согласно доказанной теореме.

Оценим трудоемкость указанного алгоритма. Количество вершин графа G равно $|V| = |\mathcal{N}| \cdot 2n = N_{nodes} \cdot 2n$. Количество ребер $|E| \le 2|V|$, т. к. $|A| = |V|$, $|B| \le |V|$. Алгоритм поиска вширь имеет сложность $O(V + E) = O(n \cdot N_{nodes})$. Но т. к. требуется N_{nodes} запусков поиска вширь, то общая сложность решения задачи связности

$$T_{conn} = O(n \cdot N^2_{nodes}).$$

Если применить предположение, которое было сделано для первого алгоритма, а именно $n \ll N_{nodes}$, то можно окончательно записать

$$T_{conn} = O(N^2_{nodes}).$$

Видим, что вычислительная сложность для полной маршрутизации практически аналогична оценке (1). Остановимся на этом немного подробнее. Напомним, что при построении оценки (1) было сделано предположение о малом количестве возможных путей между парой вершин. В случае полной машрутизации это предположение не выполняется и поэтому решение задачи связности в данном случае первым алгоритмом имело бы сложность порядка $O(N^3_{nodes})$. Но путем сведения задачи к задаче связности на графе удалось применить эффективный алгоритм поиска вширь, что и позволило получить более выгодную оценку сложности.

5. Исследуемые алгоритмы маршрутизации

В задаче связности важную роль играет отображение $R(u, v)$, которое показывает множество путей, которые данный алгоритм маршрутизации позволяет построить. Видно, что чем шире данное множество, тем большим потенциалом к отказоустойчивости обладает выбранная сеть. В этом разделе мы рассмотрим два алгоритма маршрутизации $R(u, v)$, а именно dirbit-маршрутизацию и метод First Step/Last Step. При помощи построенного алгоритма будет проведено исследование показателя, связанного с отказоустойчивостью рассматриваемой сети: вероятность потери связности при случайном выборе множества отказавших каналов связи.

5.1. dirbit-маршрутизация

Маршрутизация с использованием битов направлений предложена в [3]. Приведем здесь формальное описание этого алгоритма.

Пусть узел-отправитель имеет координаты $s = (s_1, \ldots, s_n)$, узел-получатель $t = (t_1, \ldots, t_n)$. По сути dirbit-маршрутизация является *покоординатной*: маршрут следования выравнивает координаты в определенном порядке. Пусть задано подмножество направлений $D \subseteq \mathcal{D}$, причем выполнено $d \in D \Rightarrow -d \notin D$, т. е. D не включает в себя противоположных направлений. Тогда путь P из s в t будет выглядеть следующим образом (здесь использована сокращенная запись пути, указаны не все промежуточные вершины):

$$
\begin{aligned}
P(D) = \qquad\qquad\qquad & u^0 = s, \\
D_1, D_1, \ldots, D_1, \quad & u^1 = (s_1, s_2, \ldots, t_{|D_1|}, \ldots, s_n), \\
D_2, D_2, \ldots, D_2, \quad & u^2 = (s_1, \ldots, t_{|D_1|}, \ldots, t_{|D_2|}, \ldots, t_n), \\
\ldots \quad & \ldots \\
D_{N-1}, \ldots, D_{N-1}, \quad & u^{N-1} = (t_1, t_2, \ldots, s_{|D_N|}, \ldots, t_n), \\
D_N, D_N, \ldots, D_N, \quad & u^N = t.
\end{aligned}
$$

где $D_i \in D$, $D_{i-1} < D_i$. Видно, что критерий существования этого пути

$$\{|d| \mid d \in D\} \supseteq \{i \mid s_i \neq t_i\},$$

т. е. множество D должно включать в себя направления, соответствующие отличающимся координатам узлов отправки и назначения.

Количество возможных dirbit-путей из s в t дается формулой

$$N_{paths}(s,t) = 2^{|\{i | s_i \neq t_i\}|} \leq 2^n,$$

т. к. для каждого индекса i различающихся координат в s и t возможно составление двух маршрутов: один будет включать движение в направлении i, другой — в противоположном направлении.

Отсюда видно, что количество путей $N_{paths} \ll N_{nodes}$, а значит построенный алгоритм будет эффективно применим к этой маршрутизации.

5.2. Метод First Step/Last Step

Метод First Step/Last Step был предложен в [4] как механизм обхода отказавших узлов, а так же для маршрутизации в необычных конфигурациях сети.

Добавление нестандартного первого шага и нестандартного последнего шага позволяет расширить множество возможных путей маршрутизации. Если $R_0(u, v)$ — множество путей, полученных при помощи некоторого другого алгоритма (например, dirbit-маршрутизации), то добавляя нестандартный первый и последний шаги мы можем получить маршрутизацию $R_{FL}(u, v)$:

$$\tilde{R}_{FL}(u,v) = R_0(u,v) \cup \{u, FS, p, LS, v \mid FS \in \mathcal{D}, LS \in \mathcal{D}, p \in R_0(u + FS, v - LS)\}.$$

Написанное означает: один шаг из u в направлении FS, затем произвольный путь из $u+FS$ в $v - LS$, затем один шаг из $v - LS$ в v.

Из указанной маршрутизации необходимо исключить пути, не удовлетворяющие правилу порядка направлений:

$$R_{FL}(u,v) = \tilde{R}_{FL}(u,v) \cap \mathcal{P}_D,$$

Далее будет рассмотрена задача связности для маршрутизаций R_0 и R_{FL}, где R_0 — построенная в 5.1 dirbit-маршрутизация.

6. Оценка отказоустойчивости маршрутизации

Пусть известно, что в сети имеется N_F отказавших каналов связи. Определим вероятность потери связности для заданной маршрутизации R:

$$P_F(N_F) = \mathbb{P}(\exists u, v \Rightarrow R(u, v) \cap \mathcal{G} = \varnothing \mid |F| = N_F),$$

где множество отказавших каналов F выбирается равномерно из всех множеств $|F| = N_F$. Т. е. P_F — вероятность появления хотя бы одной пары узлов, между которыми не существует допустимого пути.

Для нахождения зависимости $P_F(N_F)$ применим метод Монте-Карло: будем генерировать множества F заданной мощности N_F и определять наличие путей между всеми парами вершин:

$$P_F(N_F) \approx \frac{|\{\exists u, v \Rightarrow R(u, v) \cap \mathcal{G} = \varnothing \mid F \in \mathcal{F}\}|}{N_{iter}},$$

где $\mathcal{F}$ — набор случайно сгенерированных множеств отказавших каналов связи,

$$|\mathcal{F}| = N_{iter},$$

$$|F| = N_F, F \in \mathcal{F}.$$

Оценивать отказоустойчивость алгоритмов маршрутизации будем на основе вероятности потери связности.

На рис. 1 и 2 представлены графики зависимости $P_F(N_F)$, построенные методом Монте-Карло для $N_{iter} = 10\,000$. Обратим внимание на несколько фактов:

1) Полная маршрутизация задает теоретический барьер для задачи связности: она предполагает все возможные пути между парой узлов, удовлетворяющие правилу порядка направлений. Таким образом, увеличение отказоустойчивости за границы полной маршрутизации возможно только применением принципиально отличных методов маршрутизации, например [5].

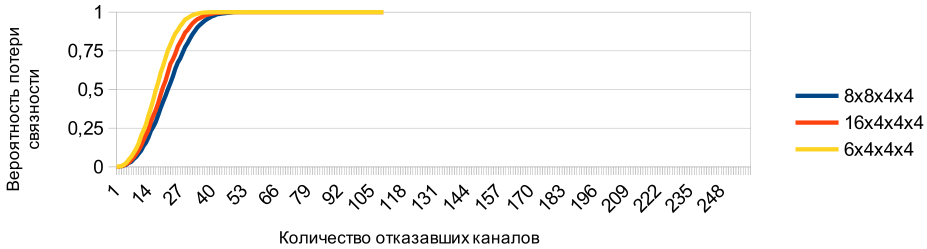
2) Даже при малом количестве отказов наблюдается появление ненулевой вероятности потери связности. Для dirbit-маршрутизации это объясняется тем, что два отказавших канала в одном кольце тора приводят к отсутствию связности между некоторыми узлами в этом кольце. Но даже полная маршрутизация обладает этим недостатком.

3) Маршрутизация с применением нестандартных шагов позволяет заметно снизить вероятность потери связности, т. к. вводит дополнительные пути. Из рис. 1 видно, что для тора размером $8 \times 8 \times 4 \times 4$ вероятность $1/2$ для dirbit-маршрутизации достигается при числе отказов каналов связи 22, а для маршрутизации с использованием First Step/Last Step — при числе отказов 81, и примерно то же соотношение верно для всех остальных значений вероятности. Для торов меньших размеров видно, что эффективность First Step/Last Step не столь высока: одинаковый уровень вероятности достигается на значительно более близких количествах отказавших каналов.

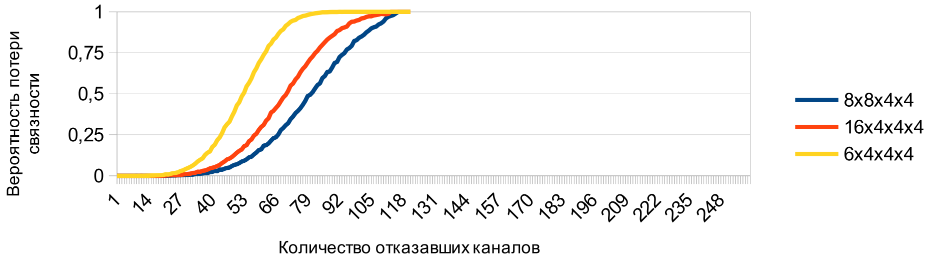
4) Из рис. 2 и равномерности изображенных на нем кривых P_F можно выдвинуть гипотезу о существовании простой математической оценки для функции P_F, но это исследование лежит за рамками данной статьи.

7. Заключение

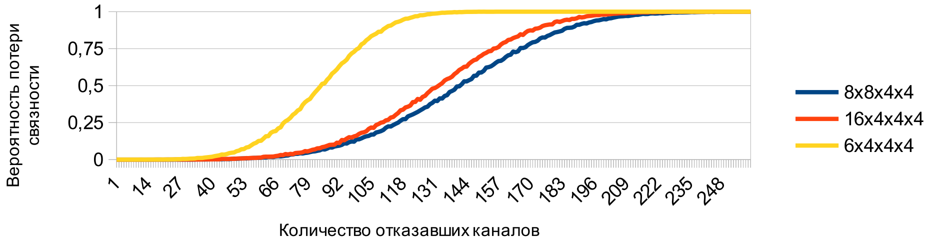
Построен алгоритм решения задачи связности для детерминированной маршрутизации в сети с топологией «многомерный тор». Этот алгоритм может использоваться как в составе



а) dirbit-маршрутизация



б) Применение First Step/Last Step



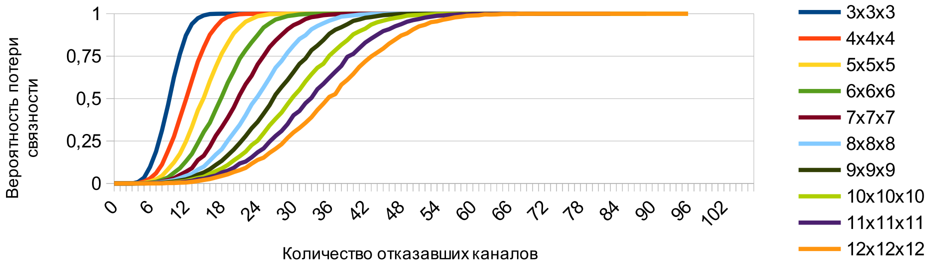
в) Полная маршрутизация

Рис. 1. Вероятность потери связности на примере 4D-торов.

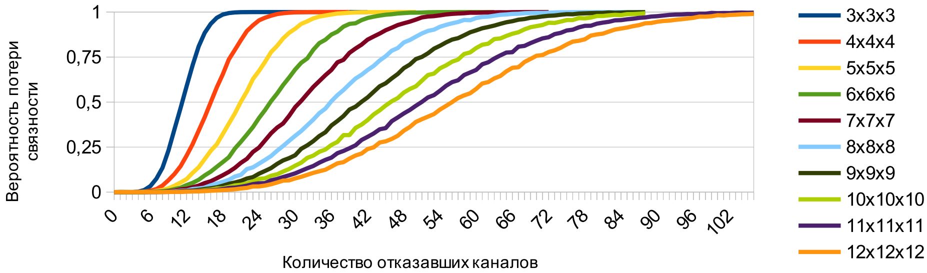
системного ПО суперкомпьютера для обеспечения связности сети в условиях наличия неисправных каналов связи, так и в качестве вспомогательного средства для теоретических исследований характеристик отказоустойчивости детерминированной маршрутизации. В ходе исследований вероятности потери связности для некоторых конфигураций сети показана эффективность метода First Step/Last Step, который позволяет увеличить количество отказов каналов связи с сохранением прежней вероятности потери связности. При помощи алгоритма для решения задачи связности для полной маршрутизации удается сравнить отказоустойчивость dirbit-маршрутизации и метода First Step/Last Step с теоретически максимальным результатом, который возможен для маршрутизации, удовлетворяющей правилу порядка направлений. Из построенных графиков видно, что потенциал отказоустойчивости такой маршрутизации выше, чем достигаемый при помощи метода First Step/Last Step, но в то же время эта разница не является огромной, поэтому с точки зрения отказоустойчивости реализация в сети полной маршрутизации не является задачей первостепенной важности.

Планируется продолжать исследование проблемы отказоустойчивости сетей, в частности построить математическую модель отказоустойчивости при помощи предложенного алгоритма, исследовать отказоустойчивость адаптивной маршрутизации, маршрутизации в подсети коллективных операций, оценить параметры производительности сети при наличии отказов, в частности, бисекционную пропускную способность.

а) dirbit-маршрутизация



б) Применение First Step/Last Step



в) Полная маршрутизация

Рис. 2. Вероятность потери связности на примере равносторонних 3D-торов.

Список литературы

1. А. А. Корж, Д. В. Макагон, И. А. Жабин, Е. Л. Сыромятников. Отечественная коммуникационная сеть 3D-тор с поддержкой глобально адресуемой памяти для суперкомпьютеров транспетафлопсного уровня производительности // Параллельные вычислительные технологии (ПаВТ'2010): Труды международной конференции (Уфа, 29 марта — 2 апреля 2010 г.). — Челябинск : Издательский центр ЮУрГУ, 2010. — С. 227–237.

2. И. Жабин, Д. Макагон, А. Симонов и др. Кристалл для Ангары // Суперкомпьютеры. — 2013. — Т. зима-2013. — С. 46–49.

3. Adiga N. R., Blumrich M., Chen D. et al. Blue Gene/L torus interconnection network // IBM Journal of Research and Development. — 2005. — Vol. 49, no. 2.3. — P. 265–276.

4. Scott S. L., et al. The Cray T3E Network: Adaptive Routing in a High Performance 3D Torus. — 1996.

5. Glass C., Ni L. The turn model for adaptive routing // Computer Architecture, 1992. Proceedings., The 19th Annual International Symposium on. — 1992. — P. 278–287.