

ДИБРОВА М.А.,
 КОГАН А.В.,
 КУЦЕНКО В. А.

СПОСОБ КОНСТРУИРОВАНИЯ ТРАФИКА В GRID СИСТЕМАХ

В статье рассмотрены вопросы повышения эффективности конструирования трафика на основе многопутевой маршрутизации при использовании частично пересекающихся путей. Задача нахождения максимального числа частично пересекающихся путей сведена к задаче нахождения минимальной области сочленения графа с последующим формированием множества непересекающихся путей к узлам области сочленения.

In the article questions increasing the efficiency of traffic engineering based on multipath routing using partially overlapping paths. The problem of finding the maximum number of partially overlapping paths is reduced to the problem of finding the minimum of the junction area of the graph with the subsequent formation of a plurality of disjoint paths to the nodes of the junction area.

Введение

Одной из основных задач организации функционирования Grid является организация эффективного доступа к распределенным ресурсам за счет оптимизации процедуры конструирования многопутевого трафика (TE).

Конструирование многопутевого трафика включает следующие задачи:

1. Формирование множества путей с заданными параметрами качества обслуживания (QoS).
2. Организация многопутевых виртуальных каналов, ориентированных на передачу трафика разного типа.
3. Управление передачей информации

В статье [1] рассмотрены вопросы конструирования трафика при распределенной многопутевой маршрутизации. Предложен алгоритм распределенного управления трафиком при многопутевой маршрутизации, обеспечивающий оптимизацию нагрузки на сеть передачи и минимизирующий стоимость передачи. В статье [2] рассмотрены вопросы конструирования трафика на основе адаптивной многопутевой маршрутизации при использовании виртуальных каналов.

В общем случае максимальное количество непересекающихся путей между двумя вершинами v_i и v_j определяется минимальным значением степени вершин v_i , v_j и минимальным множеством сочленения графа $G=(B,E)$. В данном случае под минимальным множеством сочленения $B_S = B \setminus (B_1 \cup B_2)$ графа будем понимать множеством вершин, удаление которых делит

граф $G=(B,E)$ на два подграфа $G_1=(B_1,E_1)$ и $G_2=(B_2,E_2)$ при условии, $v_i \in B_1$, а $v_j \in B_2$.

Для графа, представленного на рис.1 степень начальной вершины v_0 равна 3, степень конечной вершины v_{17} равна 4, а степень минимального множества сочленения $B_S=\{v_7, v_8\}$ равна 2. Таким образом, максимальное число непересекающихся путей между вершинами v_0 и v_{17} будет равно 2.

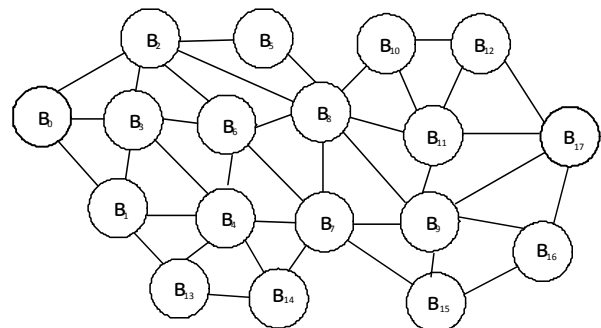


Рис.1. Структура графа

Выбор того или иного набора путей зависит от требований, предъявляемых к качеству обслуживания (QoS) передаваемой информации, и эффективности функционирования сети передачи информации.

При ограниченном количестве непересекающихся путей используются пересекающиеся пути, то есть пути, имеющие общие несмежные узлы. Под коэффициентом пересечения k_{ri} пути $P_i=(B_i,E_i)$ будем понимать отношение N_r общих вершин с другими путями к множеству N_{P_i} собственных вершин пути $P_i (B_i,E_i)$, то есть: $k_{ri}=N_r/N_{P_i}$. При $k_{ri} =0$ путь не пересекается с другими путями. При $N_r =1$ путь является частично пересекающимся.

1. Способ определения минимального множества сочленения графа.

В отличие от работы [3] предлагаемый алгоритм определения минимального множества сочленения основан на процедуре формирования множеств сочленения двух подграфов $G_1=(B_1, E_1)$ и $G_2=(B_2, E_2)$ графа $G=(B, E)$ при которой количество вершин N_1 подграфа $G_1=(B_1, E_1)$ изменяется от 1 до $(N-1)$. В результате этого последовательно формируется несколько различных множеств B_S , среди которых выбирается B_{Smin} с минимальной мощностью h_{Smin} .

При формировании множеств сочленения в подграфе $G_1=(B_1, E_1)$ будем различать внутренние и граничные вершины.

Вершины множества $B^i_l = \{v_i | i=1, 2, \dots, n\}$ подграфа $G_1(B_1, E_1)$ не смежные с вершинами разделяющего множества B_S будем относить к внутренним вершинам. Для множества внутренних вершин $B^i_l \subset B_1$ выполняется условие $E^i = \{e^i_{k,j} | v_k \in B_1, v_j \in B_1\}$. Соответственно ребро $e^i_{k,j}$ является внутренним ребром.

Вершины множества $B^b = \{v^b_i | i=1, 2, \dots, n\}$, смежные с вершинами множества B_S , будем называть граничными вершинами подграфа $G_1(B_1, E_1)$. Соответственно ребро $e^b_{k,j}$ будем называть граничным ребром. Для множества граничных вершин $B^b \subset B_1$ выполняется условие $E^b = \{e^b_{k,j} | v_k \in B_1, v_j \in B_S\}$.

Будем различать внутреннюю S^i_i и внешнюю S^b_i степени вершины v_i .

Количество внутренних ребер вершины v_k определяет ее внутреннюю степень S^i_k . В свою очередь ребро $e^b_{k,j} = \{v_k \in B_1, v_j \notin B_1\}$ является внешним для подграфа $G_1(B_1, E_1)$, при этом вершина $v_k \in B_S$. Количество внешних ребер вершины v_k определяет ее внешнюю степень S^b_k .

Процесс определения минимального множества сочленения B_{Smin} заключается в последовательном формировании множества вершин B_S и определении среди них B_{Smin} , а именно:

1. Из вершин, смежных с начальной вершиной v_i , формируется множество вершин B_S , которое в данном случае является B_{Smin} .

2. Множество смежных вершин включается в подграф $G_1=(B_1, E_1)$

3. Формируется новое множество вершин $B_l = B_1 \cup B_S$ подграфа $G_l=(B_l, E_l)$

4. Формируем новое множество граничных вершин B^o_l .

5. На основе вершин $v_i \notin B_l$ смежных с вершинами множества B^o_l формируется множество вершин B_S .

6. Вычисляется мощность h_S множества вершин B_S .

7. Сравниваем мощность h_S множества B_S с мощностью h_{Smin} множества B_{Smin} . Если $h_{Smin} > h_S$ то множество сочленения B_S становится B_{Smin} .

8. Формируется граф $G_2=(B_2, E_2)$ с новым множеством вершин $B_2 = B_l \setminus B_S$.

9. Если $B_2 \neq \{v_i\}$ то переход к пункту 2.

10. Конец.

Рассмотрим процесс определения минимального множества сочленения при формировании пути между вершинами v_0 и v_{17} (рис.1). На первом этапе формируется $B_{Smin} = B_S = \{v_1, v_2, v_3\}$, затем формируется $B_S = \{v_8, v_4, v_{13}\}$. На следующем этапе формируется $B_S = \{v_8, v_7\}$, которое является минимальным $B_{Smin} = B_S = \{v_8, v_7\}$. Следующее значение $B_S = \{v_{10}, v_{11}, v_9, v_{15}\} > B_{Smin}$. Таким образом, остается $B_{Smin} = \{v_8, v_7\}$.

Формирование множества непересекающихся путей

Следует отметить, что между начальной вершиной v_i и вершинами множества сочленения B_S может существовать несколько непересекающихся путей, количество которых больше или равно мощности множества B_{Smin} . Между вершинами множества сочленения B_S и конечной вершиной v_j также может существовать несколько непересекающихся путей.

С целью исключения операций направленного перебора, характерного для комбинаторных алгоритмов формирования путей, в работе [4] предлагается потоковый алгоритм формирования путей от одного узла к нескольким узлам на основе метода «ветвей и границ». На начальном этапе дерево решений состоит из начальной вершины, например, вершины v_0 (рис. 1) и смежных с ней вершин $B_b = \{v_1, v_2, v_3\}$, которые в данном случае являются граничными вершинами для подграфа $G_1=(B_1, E_1)$. Вершина v_0 относится к множеству B_0 внутренних вершин подграфа $G_1=(B_1, E_1)$. При формировании путей в множество B_0 каждый раз добавляется вершина $v_i \in B_b$, имеющая меньшее число внешних ветвей по сравнению с остальными граничными вершинами.

Соответственно, в множество B_b добавляется вершина v_j , смежная с вершиной v_i с минимальной внешней степенью S^b_j . Таким образом, строится дерево решений с корнем в вершине v_0 пока не сформируются все непересекающиеся пути к заданным вершинам.

С учетом введенных обозначений алгоритм формирования множества путей от вершины v_i к вершине заданного множества B_z вершин заключается в следующем.

1. Формируем начальное множество $B_0 = \{v_i\}$ внутренних вершин подграфа $G_1 = (B_1, E_1)$.

2. Формируем множество граничных вершин $B_b = \{v_j | j=1, 2, \dots, k\}$, которое в данном случае представляет множество вершин, смежных с вершиной v_i .

3. Для $j=1$ до k определяем путь $P_j = \{v_i, v_j \in B_b\}$.

4. Для подграфа $G_1(B_1, E_1)$ формируем множество путей $W_1 = \{P_j\}$.

5. Среди вершин $v_j \in B_b$ определяем вершину v_m с минимальной внешней степенью S_m^b .

6. Перемещаем вершину v_m в множество внутренних вершин, $v_m \in B_0$.

7. Формируем подграф $G_1(B_1, E_1)$ в котором $B_0 = B_0 \cup v_m$.

8. Если среди вершин v_i , смежных с вершиной v_m , нет вершины $v_k \in B_z$, то переход к пункту 9.

9. Если среди вершин v_i , смежных с вершиной v_m , имеется вершина $v_k \in B_z$, то формирование пути P_i до вершины v_k завершается.

10. Путь P_i заносится в множество путей.

11. Если множество внешних вершин $B_b \neq \emptyset$, то переход к пункту 4.

12. Конец.

Рассмотрим пример формирования множества путей для графа, представленного на рис.1. На первом этапе формируются пути от начальной вершины v_0 до смежных с ней вершин: $P_1 = \{v_0, v_1\}$; $P_2 = \{v_0, v_2\}$; $P_3 = \{v_0, v_3\}$. На следующем этапе данные пути продлеваются соответственно до вершин v_8 и v_4 , а именно: $P_1 = \{v_0, v_1, v_4\}$; $P_2 = \{v_0, v_2, v_8\}$; $P_3 = \{v_0, v_3, v_6\}$. На данном этапе путь $P_2 = \{v_0, v_2, v_8\}$ является сформированным. Затем формируются пути $P_1 = \{v_0, v_1, v_4, v_7\}$ и $P_3 = \{v_0, v_3, v_6, v_8\}$.

Формирование всего множества путей между граничными вершинами (v_7, v_8) и конечной вершиной v_{17} осуществляется аналогичным образом, начиная с конечной вершины v_{17} , при этом формируются следующие непересекающиеся пути: $P_4 = \{v_8, v_{10}, v_{12}, v_{17}\}$; $P_5 = \{v_8, v_{11}, v_{17}\}$; $P_6 = \{v_7, v_9, v_{17}\}$; $P_7 = \{v_7, v_{15}, v_{16}, v_{17}\}$ (рис.2).

В данном случае между вершинами v_0 и v_{17} формируются 6 путей: $P_{16} = \{P_1, P_6\}$; $P_{17} = \{P_1, P_7\}$; $P_{24} = \{P_2, P_4\}$; $P_{25} = \{P_2, P_5\}$; $P_{34} = \{P_3, P_4\}$; $P_{35} = \{P_3, P_5\}$.

В результате между вершинами v_0 и v_{17} можно сформировать 8 пар различных не пересекающихся путей: (P_{16}, P_{24}) ; (P_{16}, P_{25}) ; (P_{16}, P_{34}) ; (P_{16}, P_{35}) ; (P_{17}, P_{24}) ; (P_{17}, P_{25}) ; (P_{17}, P_{34}) ; (P_{17}, P_{35}) .

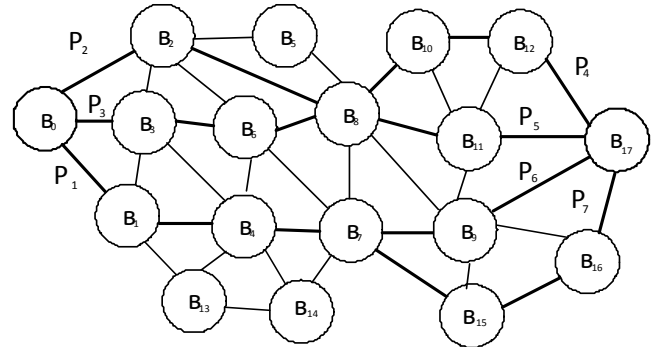


Рис. 2. Формирование множества непересекающихся путей

При этом, в зависимости от требуемых параметров QoS между вершинами v_0 и v_{17} могут быть сформированы пути различной длины, например путь $P_{25} = \{v_0, v_2, v_8, v_{11}, v_{17}\}$ является самым коротким с длиной $L_{25} = 4$. Самыми длинными является пути $P_{17} = \{v_0, v_4, v_7, v_{15}, v_{16}, v_{17}\}$ и $P_{34} = \{v_0, v_3, v_6, v_8, v_{10}, v_{12}, v_{17}\}$ длиной равной 6.

При организации параллельной передачи могут быть использованы пути P_{16} и P_{35} одинаковой длины равной 5. В этом случае параллельно передаваемые части сообщения будут собираться без дополнительной задержки в приемном узле.

На рис. 3. представлена сборка трех частей сообщения, переданных параллельно по маршрутам с одинаковой задержкой передачи $\tau_i = const$, где: T_i – время передачи i -й части сообщения, T_d – время добавления (записи) части сообщения.

В данном случае время сборки всего сообщения $T_{сб} = (t_2 - t_0)$ минимально и равно $2 T_d$. При задержке передачи каждой части сообщения на величину $\nabla t = (t_1 - t_0)$ относительно предыдущей части сообщения время сборки всего сообщения остается минимальным и равно $T_{сб} = (N-1) T_d$.

При задержке $\tau_i > T_d$ передачи i -ой части сообщения время сборки сообщения увеличивается на величину $t_z = \tau_i - T_d$.

Наличие достаточно большого набора всех возможных путей затрудняет процесс конструирования трафика при организации многопутевой передачи информации. В этом случае с целью обеспечения заданных параметров QoS целесообразно формировать многопутевые ви-

ртуальные каналы с учетом требований к характеру трафика.

Выводы

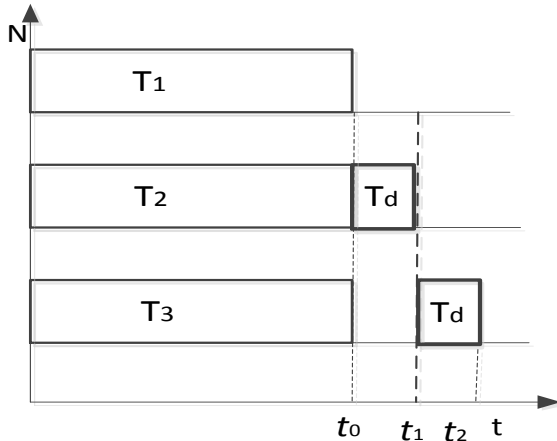


Рис. 3. Параллельная передача частей сообщения

Таким образом, возможность формирования различных многопутевых виртуальных каналов позволяет оптимизировать процесс передачи информации в Cloud Computing.

Предложенный в работе алгоритм определения минимальной области сочленения позволяет определить максимальное число непересекающихся путей и вершины частично пересекающихся путей. Предложенный способ формирования частично пересекающихся путей за счет формирования непересекающихся путей до вершин области сочленения позволяет существенно образом уменьшить временную сложность алгоритма формирования непересекающихся и частично пересекающихся путей. Формирование многопутевых виртуальных каналов с учетом требований *QoS* позволяет упростить процедуру конструирования трафика и повысить эффективность Cloud Computing. Возможность формирования многопутевых виртуальных каналов с одинаковой задержкой передачи по каждому пути обеспечивает минимальное время сборки сообщения из параллельно передаваемых его частей.

Список литературы

1. Ditiya Vyas Survey of Distributed Multipath Routing Protocols for Traffic Management/Ditiya Vyas, Ritesh Patel, Amit Ganatra // International Journal of Computer Applications (0975 – 8887) Volume 63. – No.17, February 2013 p. – P. 42-48.
2. Karthiga. S Traffic Engineering System Based on Adaptive Multipath Routing/ Karthiga. S, Balamurugan. M. S // International Journal of Emerging Technology and Advanced Engineering ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 2, February, 2013. – P.659-664. Website: www.ijetae.com
3. Кулаков Ю.А. Определение оптимального числа граничных маршрутизаторов при организации безопасной многопутевой маршрутизации в мобильной компьютерной сети большой размерности / Кулаков Ю.А., Коган А.В. // Вісник національного технічного університету України «КПІ», Інформатика, управління та обчислювальна техніка. – 2013. – № 31. – С. 128-134.
4. Кулаков Ю.А. Способ формирования виртуальной структуры grid, ориентированной на многопутевую маршрутизацию / Кулаков Ю.А. Диброва М.А.// Вісн. Національного техн. ун-ту України “КПІ”: Інформатика, управління та обчислювальна техніка. – К.: ТОВ “ВЕК+”, 2014. – Вип. 61. – С.65–69.