

СПОСОБ ФОРМИРОВАНИЯ ВИРТУАЛЬНОЙ СТРУКТУРЫ GRID, ОРИЕНТИРОВАННОЙ НА МНОГОПУТЕВУЮ МАРШРУТИЗАЦИЮ

В статье рассмотрены вопросы формирования виртуальных сетей для Grid. Приведена временная сложность формирования виртуальных каналов. Разработан алгоритм формирования с минимальной временной сложностью. Приведен пример формирования множества путей от одного узла к нескольким узлам Grid.

The article discusses the formation of virtual networks for Grid. Shows the time complexity of the formation of virtual channels. An algorithm for the formation of the minimum time complexity. An example of forming a plurality of paths from one node to multiple nodes Grid.

1. Введение

В настоящее время при организации Grid систем широко используются виртуальные частные сети (VPN) [1,2,3]. Узлы VPN объединяются между собой с помощью виртуальных каналов или туннелей, представляющих собой последовательность каналов передачи данных сети передачи. Как правило, топология виртуальной сети формируется с учетом информационных потоков между ее узлами. Например, при использовании технологии « клиент-сервер» VPN характеризуется звездообразной топологией. Узлы одноранговых VPN связываются по технологии P2P. При организации многоуровневых VPN виртуальные каналы формируют древовидную структуру сети.

Минимальным количеством виртуальных каналов $k = N - 1$, где N – количество вершин, характеризуется VPN с топологией минимального покрывающего дерева. Наиболее универсальной является полносвязная топология VPN с количеством каналов $k = N(N - 1)$.

В настоящее время при организации передачи информации в распределенных компьютерных системах широко используется многопутевая маршрутизация [4] при которой несколько физических каналов объединяются в один многоканальный виртуальный путь. Это позволяет повысить эффективность процедуры конструирования трафика (TE), обеспечить заданные параметры QoS при изменении параметров системы передачи информации, в частности при изменении метрики каналов, путем замены одного физического канала другим в

рамках одного многоканального виртуального пути.

Временная сложность формирования многоканального виртуального пути состоящего из q физических каналов с помощью комбинаторных алгоритмов, например, алгоритма Дейкстры равна $O(qN^2)$. В связи с этим актуальным является разработка и использования потоковых алгоритмов многопутевой маршрутизации, характеризующихся минимальной временной сложностью [5,6]. Уменьшить временную сложность можно за счет исключения операции перебора вариантов на основе метода « ветвей и границ».

2. Способ формирования множества MVP в Grid системах

Как правило, в Grid системах используется многоточечное соединение, при котором одна вершина соединяется с несколькими вершинами, образуя звездообразную, древовидную или полносвязную топологию. С учетом этого предлагается использовать модифицированный алгоритм « ветвей и границ» [7] формирования множества путей между двумя вершинами. С целью дальнейшего уменьшения временной сложности формирования виртуальных каналов в данной работе предлагается формировать одновременно виртуальные каналы от одного узла v_i виртуальной сети к множеству смежных узлов $V_i = \{v_j | \forall e_{i,j}\}$ виртуальной сети, представленной в виде графа $G_z(V, E)$. В рамках каждого виртуального канала e_i между вершинами v_i и v_j формируется множество $W_{i,j} = \{L_r | r = 1, 2, \dots, m\}$ не пересекающихся физических каналов.

Вершины множества $B^i = \{v_i | i=1,2,\dots,n\}$ подграфа $G_1(B_1, E_1)$ не смежные с граничными вершинами множества B_s будем относить к внутренним вершинам. Для множества внутренних вершин $B^i \subset B_1$ 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 $B^o = \{v^o_i | i=1,2,\dots,n\}$, смежные с вершинами множества B_s , будем называть граничными вершинами подграфа $G_1(B_1, E_1)$. Удаление граничных вершин приводит к тому, что граф становится несвязным. Соответственно ребро $e^o_{k,j}$ будем называть граничным ребром. Для множества граничных вершин $B^o \subset B_1$ выполняется условие $E^o = \{e^o_{k,j} | v_k \in B_1, v_j \in B_2\}$.

Будем различать внутреннюю S^i_k и внешнюю S^o_k степени вершины.

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

Для подграфа $G_1(B_1, E_1)$, представленного на рис.1, множество вершин которого $B_1 = \{v_0, v_1, v_2, v_3\}$ граничными являются вершины $B_s = \{v_1, v_2, v_3\}$.

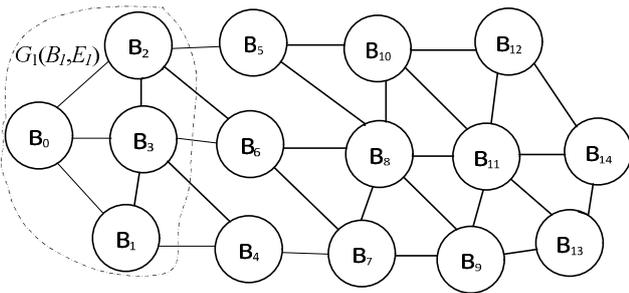


Рис.1. Граф сети

Минимальное множество граничных вершин $B_{s\ min}$ графа $G_0(B_0, E_0)$ составляют вершины v_6 и v_7 , то есть $B_{s\ min} = \{v_6, v_7\}$. Множество внутренних вершин подграфа $G_1(B_1, E_1)$ равно $B^i = \{v_0, v_1, v_2, v_3\}$, а множество граничных вершин данного подграфа равно $B^o = \{v_3, v_4, v_5\}$.

Процесс формирования множества осуществляется следующим образом. На начальном этапе предлагается формировать пути от исходной вершины до смежных с ней вершин $v_i \in B_s$, которые в данном случае являются гра-

ничными для множества, состоящего из одной исходной вершины. Затем граничные вершины включаются в множество внутренних вершин подграфа $G_1(B_1, E_1)$ и формируется новое множество B_s . Таким образом строится дерево решений с корнем в вершине v_0 до тех пор пока не сформируются все непересекающиеся пути ко всем заданным вершинам.

Временная сложность данного алгоритма равна $O(S_m D)$, где S_m – средняя степень вершин и D – диаметр графа.

С учетом введенных обозначений алгоритм формирования множества путей от одной вершины к заданному множеству вершин заключается в следующем.

1. Формируем подграф $G_1(B_1, E_1)$ в котором $B_1 = \{v_0, B_c\}$, где: v_0 – исходная вершина; $B_s = \{v_i | i=1,2,\dots,p\}$ – в данном случае представляет множество вершин, смежных с вершиной v_0 .

2. Для подграфа $G_1(B_1, E_1)$ определяем множество путей $W = \{L_{o,r} | r=1,2,\dots,p\}$, где $L_{o,j} = \{v_0, v_j\}$.

3. Формируем новое множество $B_1 = B_1 \cup B_s$

4. Формируем подграф $G_1(B_1, E_1)$ в котором $B_1 = B_1 \cup B_s$

5. Формируем подграф $G_2(B_2, E_2)$ в котором $B_2 = B_2 \setminus B_s$

6. Для подграфа $G_1(B_1, E_1)$ формируем множество граничных вершин B_s .

7. Среди множества вершин $B_s = \{v_i | i=1,2,\dots,p\}$ выбираем вершину v_i с минимальной внешней степенью $S^o_k = \min \{S^o_i | i=1,2,\dots,p\}$.

8. Для вершины v_i находим внутреннюю вершину $v_j \in B^i_2$ подграфа $G_2(B_2, E_2)$ с минимальной степенью.

9. Продлеваем путь $L_{o,i}$ до вершины $v_j \in B^i_2$

10. Если $v_j \notin B^V$ то ($v_j \subset B_c, v_i \subset B^i_1$) иначе (путь $L_{o,j}$ сформирован)

11. Если множество $B_1 \neq \emptyset$ то (переход к п. 7)

12. Конец.

Рассмотрим пример формирования множества путей от вершины v_0 к вершине v_{14} .

На рис. 2 представлен пример формирования первого пути от вершины v_0 к вершине v_{14} .

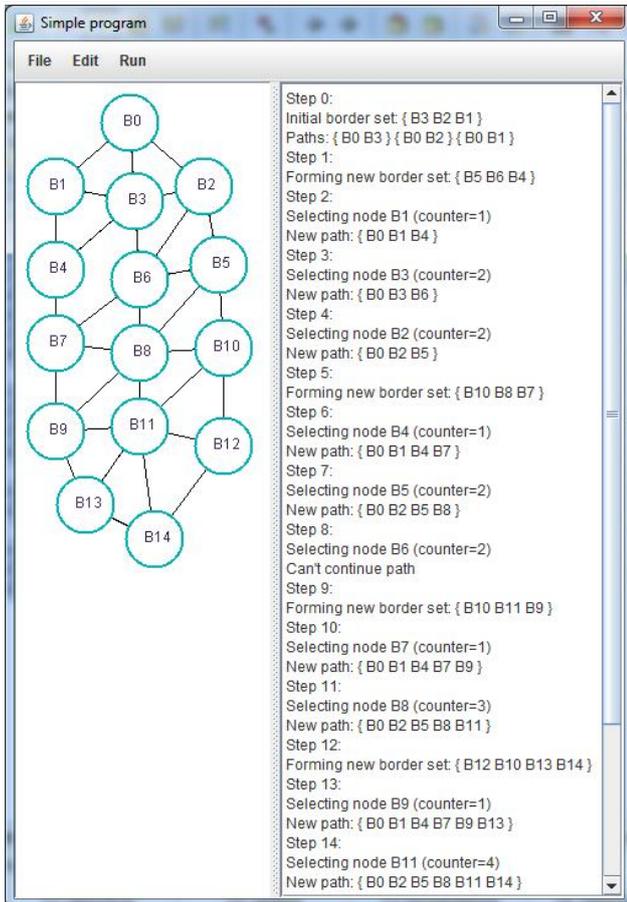


Рис. 2. Формирование первого пути от вершины v_0 к вершине v_{14} .

Формирование всего множества путей между вершинами v_0 и v_{14} заключается в следующем:

Step 0:

Initial border set: { B1 B3 B2 } /* формирование множества граничных узлов для вершины v_0 */

Paths: { B0 B1 } { B0 B3 } { B0 B2 } /* формирование путей от вершины v_0 к вершинам v_3, v_2, v_1 . */

Step 1:

Forming new border set: { B4 B6 B5 } /* новое множество граничных узлов */

Step 2:

Selecting node B1 (counter=1) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B1 B4 } /* продолжение пути от вершины v_0 к вершине v_4 */

Step 3:

Selecting node B3 (counter=2) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B3 B6 } /* продолжение пути от вершины v_0 к вершине v_6 */

Step 4:

Selecting node B2 (counter=2) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B2 B5 } /* продолжение пути от вершины v_0 к вершине v_5 */

Step 5:

Forming new border set: { B8 B7 B10 } /* новое множество граничных узлов */

Step 6:

Selecting node B4 (counter=1) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B1 B4 B7 } /* продолжение пути от вершины v_0 к вершине v_7 */

Step 7:

Selecting node B6 (counter=2) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B3 B6 B8 } /* продолжение пути от вершины v_0 к вершине v_8 */

Step 8:

Selecting node B5 (counter=2) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B2 B5 B10 } /* продолжение пути от вершины v_0 к вершине v_{10} */

Step 9:

Forming new border set: { B11 B9 B12 } /* новое множество граничных узлов */

Step 10:

Selecting node B7 (counter=1) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B1 B4 B7 B9 } /* продолжение пути от вершины v_0 к вершине v_9 */

Step 11:

Selecting node B8 (counter=2) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B3 B6 B8 B11 } /* продолжение пути от вершины v_0 к вершине v_{11} */

Step 12:

Selecting node B10 (counter=2) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B2 B5 B10 B12 } /* продолжение пути от вершины v_0 к вершине v_{12} */

Step 13:

Forming new border set: { B13 B14 } /* новое множество граничных узлов */

Step 14:

Selecting node B9 (counter=1) /* выбор граничной вершины с минимальной значением внешней степени */

New path: { B0 B1 B4 B7 B9 B13 } /* продолжение пути от вершины v_0 к вершине v_{13} */

Step 15:

Selecting node B12 (counter=1) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { B0 B2 B5 B10 B12 B14 } /* формирование первого пути от вершины v₀ к вершине v₁₄ */

Step 16:

Selecting node B11 (counter=2) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { B0 B3 B6 B8 B11 B14 } /* формирование второго пути от вершины v₀ к вершине v₁₄ */

Step 17:

Forming new border set: { B14 } /* новое множество граничных узлов */

Step 18:

Selecting node B13 (counter=1) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { B0 B1 B4 B7 B9 B13 B14 } /* формирование третьего пути к вершине v₁₄ */

Done

В результате формируются следующая структура графа путей (рис.3).

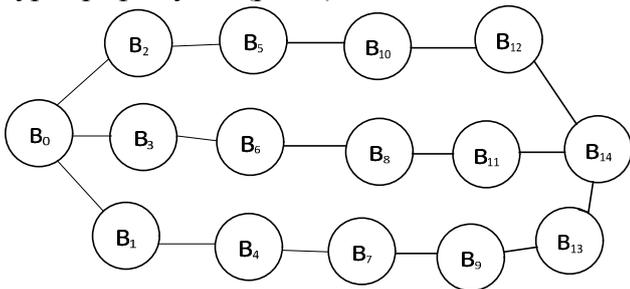


Рис.3. Множество путей между вершинами v₂ и v₁₄

Формирование первого пути { B0 B2 B5 B10 B12 B14 } к вершине v₁₄ происходит на 15 шаге из 18 шагов, второго пути { B0 B3 B6 B8 B11 B14 } происходит на 16 и третьего пути происходит на последнем 18. Таким образом, все множество путей формируется за один проход алгоритма. На формирование первого пути затрачивается 30 операций, на второго пути еще 2 операции и на третьего пути еще 3 операции, всего 35 операций. В среднем количество шагов равно числу пройденных вершин, умноженному на их среднюю степень.

Данный алгоритм позволяет сформировать множества путей от одной вершины к нескольким вершинам. Рассмотрим случай формирования путей от вершины v₀ множества вершин { B7, B10, B14 }.

На рис. 4 представлен пример формирования первого пути от вершины v₀ к вершине v₇

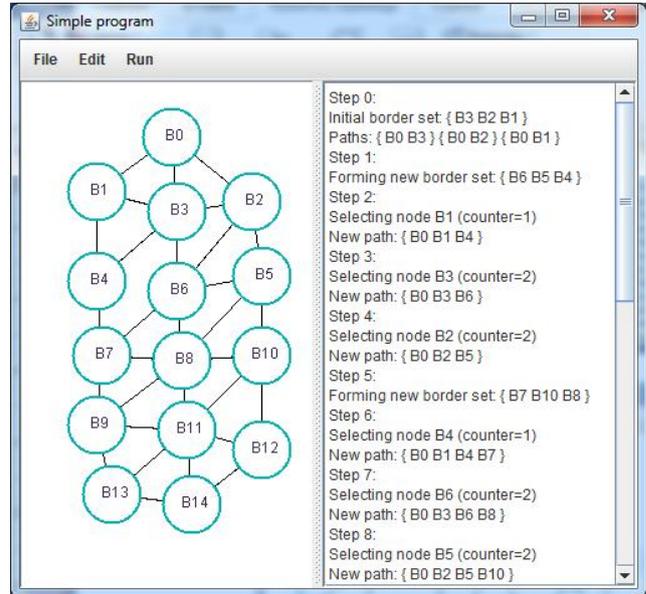


Рис.4. Формирование первого пути от вершины v₀ к вершине v₇.

В результате формируется дерево путей, представленное на рис. 5.

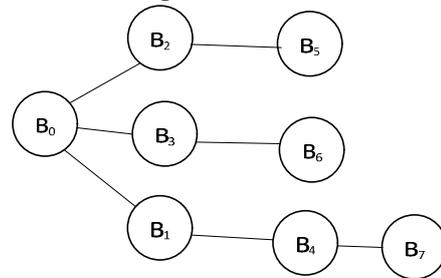


Рис. 5. Дерево путей

Следует заметить, что формирование первого пути к вершине v₇ (рис.4) совпадает с началом формирования пути к вершине v₁₄ (рис.2). Формирование всего множества путей между вершинами v₀ и v₇ заключается в следующем:

Step 0:

Initial border set: { B3 B2 B1 } /* формирование множества граничных узлов для вершины v₀ */

Paths: { B0 B3 } { B0 B2 } { B0 B1 } /* формирование путей от вершины v₀ к вершинам B₃, B₂, B₁. */

Step 1:

Forming new border set: { B4 B6 B5 } /* новое множество граничных узлов */

Step 2:

Selecting node B1 (counter=1) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { B0 B1 B4 } /* продолжение пути от вершины v₀ к вершине v₄ */

Step 3:

Selecting node B3 (counter=2) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { V₀ V₃ V₆ } /* продолжение пути от вершины v₀ к вершине v₆ */

Step 4:

Selecting node B₂ (counter=2) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { V₀ B₂ B₅ } /* продолжение пути от вершины v₀ к вершине v₅ */

Step 5:

Forming new border set: { B₁₀ B₇ B₈ } /* новое множество граничных узлов */

Step 6:

Selecting node B₄ (counter=1) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { V₀ B₁ B₄ B₇ } /* сформирован первый путь к вершине v₇ */

Step 7:

Selecting node B₆ (counter=2) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { V₀ V₃ B₆ B₇ } /* сформирован второй путь к вершине v₇ */

Step 8:

Selecting node B₅ (counter=2) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { V₀ B₂ B₅ B₈ } /* продолжение пути от вершины v₀ к вершине v₈ */

Step 9:

Forming new border set: { B₁₀ B₇ B₁₁ B₉ } /* новое множество граничных узлов */

Step 10:

Selecting node B₈ (counter=4) /* выбор граничной вершины с минимальным значением внешней степени */

New path: { V₀ B₂ B₅ B₈ B₇ } /* сформирован третий путь к вершине v₇ */

Finish riched

Done

В результате работы алгоритма формируются виртуальный канал c_{0,7}, состоящий из трех физических каналов (рис. 6):

$c_{0,7} = \{ L_1 = \{ v_0, v_1, v_4, v_7 \}; L_2 = \{ v_0, v_3, v_6, v_7 \}; L_3 = \{ v_0, v_2, v_5, v_8, v_7 \} \}$.

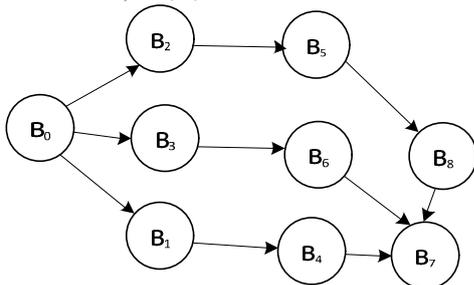


Рис.6. Множество путей от вершины v₀ до вершины v₇

После формирования пути до вершины v₇ (рис.4) продолжается формирование множества путей до следующей ближайшей вершины, в данном случае это вершина v₁₀.

Step 8:

Selecting node B₅ (counter=2) /* граничная вершины с минимальной внешней степенью */

New path: { V₀ B₂ B₅ B₁₀ } /* сформирован первый путь к вершине v₁₀ */

Step 9

New path: { V₀ B₂ B₅ B₈ } /* продолжение пути от вершины v₀ к вершине v₈ */

Step 10:

Forming new border set: { B₁₀ B₇ B₁₁ B₉ } /* новое множество граничных узлов */

Step 11:

Selecting node B₈ (counter=4) /* граничная вершины с минимальной внешней степенью */

New path: { V₀ B₂ B₅ B₈ B₇ } /* сформирован третий путь к вершине v₇ */

Step 12:

Forming new border set: { B₁₁ B₉ B₁₀ } /* новое множество граничных узлов */

Step 13:

Selecting node B₇ (counter=1) /* граничная вершина с минимальной внешней степенью */

New path: { V₀ B₁ B₄ B₇ B₉ } /* сформирован второй путь к вершине v₉ */

Step 14:

Selecting node B₈ (counter=3) /* граничная вершина с минимальной внешней степенью */

New path: { V₀ V₃ B₆ B₈ B₁₀ } /* сформирован второй путь к вершине v₁₀ */

Step 15:

Forming new border set: { B₁₁ B₁₃ } /* новое множество граничных узлов */

Step 16:

Selecting node B₉ (counter=2) /* граничная вершина с минимальной внешней степенью */

New path: { V₀ B₁ B₄ B₇ B₉ B₁₁ } /* новое множество граничных узлов */

Step 17:

Selecting node B₁₁ (counter=2) /* новое множество граничных узлов */

New path: { V₀ B₁ B₄ B₇ B₉ B₁₁ B₁₀ } /* сформирован третий путь к вершине v₁₀ */

Done

В результате работы алгоритма в дополнение к виртуальному каналу c_{0,7} формируются виртуальный канал c_{0,10}, состоящий из трех физических каналов (рис7):

$c_{0,10} = \{ L_4 = \{ v_0, v_1, v_4, v_7, v_9, v_{11}, v_{10} \}; L_5 = \{ v_0, v_3, v_6, v_8, v_{10} \}; L_6 = \{ v_0, v_2, v_5, v_{10} \} \}$.

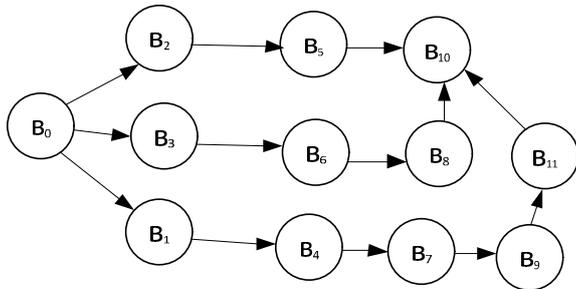


Рис. 7. Формирование путей к вершине v10

После формирования пути до вершины v10 (рис.5) на следующем 18 шаге продолжается формирование множества путей до последней вершины v14 .

Step 18:

Forming new border set: { B11 B9 B12 } /* новое множество граничных узлов */

Step 19:

Selecting node B8 (counter=2) /* граничная вершина с минимальной внешней степенью */

New path: { B0 B3 B6 B8 B11 } /* сформирован новый путь к вершине v11 */

Step 20:

Selecting node B10 (counter=2) /* граничная вершина с минимальной внешней степенью */

New path: { B0 B2 B5 B10 B12 } /* сформирован путь к вершине v12 */

Step 21:

Forming new border set: { B13 B14 } /* новое множество граничных узлов */

Step 22:

Selecting node B9 (counter=1) /* граничная вершина с минимальной внешней степенью */

New path: { B0 B1 B4 B7 B9 B13 } /* сформирован путь к вершине v13 */

Step 23:

Selecting node B12 (counter=1) /* граничная вершина с минимальной внешней степенью */

New path: { B0 B2 B5 B10 B12 B14 } /* сформирован первый путь к вершине v14 */

Step 24:

Selecting node B11 (counter=2) /* граничная вершина с минимальной внешней степенью */

New path: { B0 B3 B6 B8 B11 B14 } /* сформирован второй путь к вершине v14 */

Step 25:

Selecting node B13 (counter=1) /* граничная вершина с минимальной внешней степенью */

New path: { B0 B1 B4 B7 B9 B13 B14 } /* сформирован третий путь к вершине v14 */

В результате работы алгоритма в дополнение к виртуальным каналам c0,7 и c0,10 формируются виртуальный канал c0,14, состоящий из трех физических каналов (рис. 8):

$$c_{0,14} = \{ L_7 = \{ B_0, B_1, B_4, B_7, B_9, B_{13}, B_{14} \}; L_8 = \{ B_0, B_3, B_6, B_8, B_{11}, B_{14} \}; L_9 = \{ B_0, B_2, B_5, B_{10}, B_{12}, B_{14} \} \}.$$

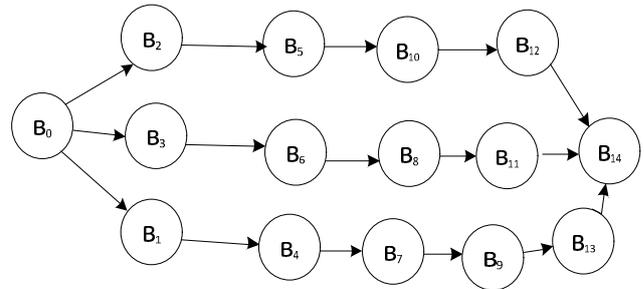


Рис. 8. Формирование путей к вершине v14

Таким образом, формируется виртуальная звездообразная топологи Grid поверх физической структуры сети (рис.9).

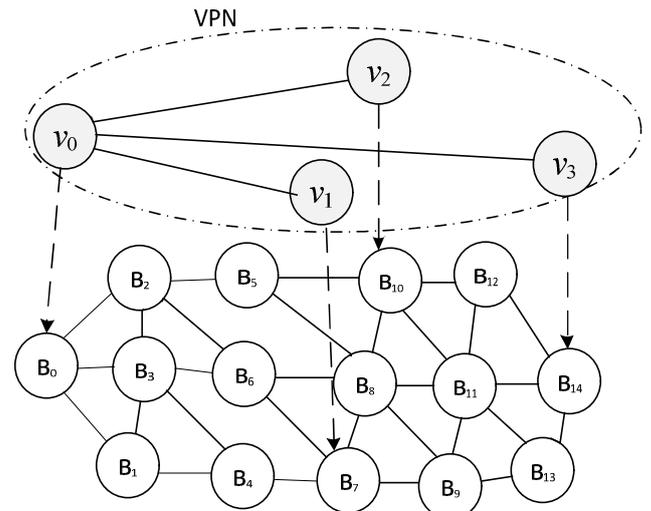


Рис. 9. Виртуальная топология Grid системы

Для формирования полносвязной топологии необходимо дополнительно сформировать виртуальные пути от вершины v7 к вершинам v8, v12 и от вершины v8 к вершине v12 .

3. Заключение

Организация виртуальной структуры с многоканальными связями позволяет повысить эффективность функционирования Grid систем и обеспечить заданные параметры QoS. Предложенный в работе модифицированный алгоритм «ветвей и границ» за счет одновременного формирования множества путей от одного узла ко всем остальным узлам существенным образом уменьшает временную сложность формирования виртуальной структуры Grid систем. В качестве одного из дальнейших направлений исследования является организации параллельной передачи частей сообщения по многоканальным виртуальным соединениям.

Список литературы

1. N. Mahesh Kumar Proposed Architecture For Implementing Privacy In Cloud Computing Using Grids And Virtual Private Network / N. Mahesh Kumar, K. Senthilkumar // International Journal Of Technology Enhancements And Emerging Engineering Research, Vol 1, ISSUE 3 ISSN 2347-4289 2013. P. 12-15.
2. Francesco Palmieri Introducing Virtual Private Overlay Network services in large scale Grid infrastructures/ Journal Of Computers, Academy Publisher VOL. 2, NO. 2, APRIL 2007. P. 61-72.
3. A. Di Costanzo "Harnessing cloud technologies for a virtualized distributed computing infrastructure,"/ A. Di Costanzo, M. D. De Assuncao, and R. Buyya // internet Computing, IEEE, vol. 13, pp. 24-33, 2009.
4. Eun-Sung Jung Distributed Multipath Routing Algorithm for Data Center Networks/ Eun-Sung Jung, Venkatram Vishwanath, Rajkumar Kettimuthu // 2014 International Workshop on Data Intensive Scalable Computing Systems
5. Lemeshko A.V. Research on Tensor Model of Multipath Routing in Telecommunication Network with Support of Service Quality by Greate Number of Indices / Lemeshko A.V., Evseeva O.Yu., Garkusha S.V // Telecommunications and RadioEngineering, Vol.73, No 15. - P. 1339-1360
6. Mahalakshmi.C Multipath Data Transfer Scheme for Virtual Private Networks/ Mahalakshmi.C, Ramaswamy. M// International Journal of Computer Applications (0975 – 8887) Volume 44– No.8, April 2012 P.27-31
7. Kulakov Y. The method of plurality generation of disjoint paths using horizontal exclusive scheduling / Y. Kulakov, A. Kogan // The science advanced. – 2014. – issue 9. – pp.16-18.