Распределенные информационно-аналитические системы

Лекция № 9.

Маршрутизация с компактными таблицами и иерархическая маршрутизация

Профессор кафедры КБ-2: д.т.н. Шатовкин Р.Р.

Учебные цели:

Изучить основы научных знаний по схеме разметки деревьев; интервальной маршрутизации; префиксной маршрутизации; иерархической маршрутизации.

Учебные вопросы:

1. ***Схема разметки деревьев.***
2. ***Интервальная маршрутизация.***
3. ***Префиксная маршрутизация.***
4. ***Иерархическая маршрутизация.***

Рассмотренные ранее алгоритмы маршрутизации требуют, чтобы каждый узел содержал таблицу маршрутизации с отдельной ссылкой для каждого возможного пункта назначения. Когда пакет передается через сеть, эти таблицы используются в каждом узле пути (исключая пункт назначения).

В лекции рассматриваются вопросы организации таблиц маршрутизации, которые описывают хранение и поиск механизмов маршрутизации. Как эти таблицы могут быть вычислены распределенными алгоритмами в лекции не рассматривается. Для простоты положим, что сеть связная.

Стратегию уменьшения таблицы в каждом из рассматриваемых механизмов маршрутизации можно объяснить следующим образом. Если таблицы маршрутизации узла хранят выходящий канал для каждого пункта назначения отдельно, то таблица маршрутизации имеет длину N; следовательно таблицы требуют W(N) бит, и не важно, как выходящий канал закодирован для каждого пункта назначения. Теперь рассмотрим перестройку таблицы: таблица содержит для каждого канала узла ссылку, говорящую, какие пункты назначения должны быть маршрутизированы через этот канал. Таблица теперь имеет «длину» deg для узла с deg каналами; теперь компактность зависит от того, насколько компактно множество пунктов назначения для каждого канала может быть представлено (рисунок 1).

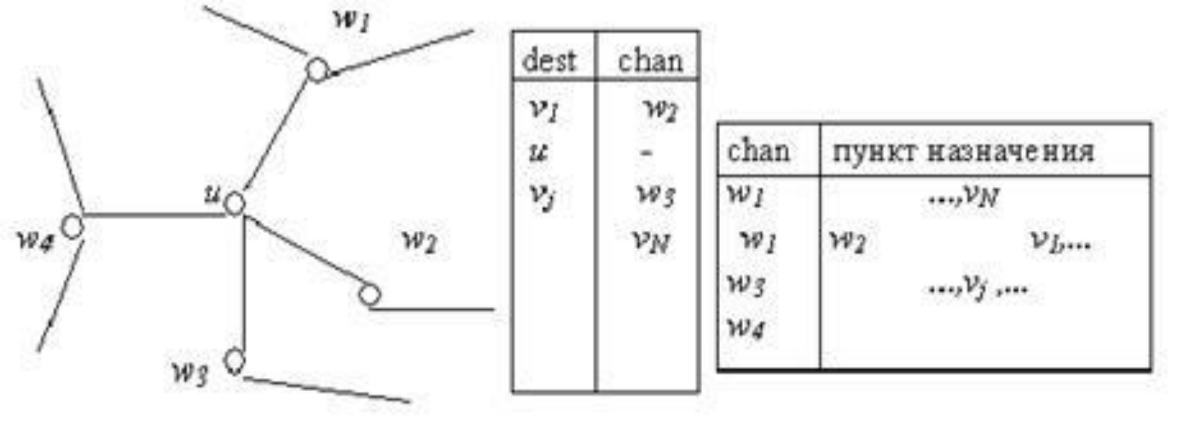

Рисунок 1 – Уменьшение размера таблицы маршрутизации

1. Схема разметки деревьев

Первый метод компактной маршрутизации был предложен Санторо и Кхатибом. Метод основан на пометке узлов целыми от 0 до N – 1, таким образом чтобы множество пунктов назначения для каждого канала было интервалом. Пусть Z_N обозначает множество {0, 1, ..., N – 1}. В этой части все арифметические операции по модулю N, то есть $(N – 1) + 1 \equiv 0$.

Определение 1. Циклический интервал [a, b) в Z_N – множестве целых, определенном как

[a,b) = {a, a + 1, ..., b – 1} если a < b;

[a,b) = {0,. . ., b – 1, a,..., N – 1} если a ≥ b.

Заметим, что [a, b) = Z_N, и для a ≠ b дополнение [a, b) – [b, a). Циклический интервал [a, b) называется линейным, если a < b.

Теорема 1. Узлы дерева T могут быть пронумерованы таким образом, что для каждого выходящего канала каждого узла множество пунктов назначения, которые маршрутизируются через данный канал, есть циклический интервал.

Доказательство. Возьмем произвольный узел v_0 за корень дерева и для каждого w обозначим как T[w] поддерево T с корнем в w. Возможно перенумеровать узлы таким образом, чтобы для каждого w числа для обозначения узлов в T[w] составляли линейный интервал, например с помощью префиксного обхода дерева (рисунок 2).

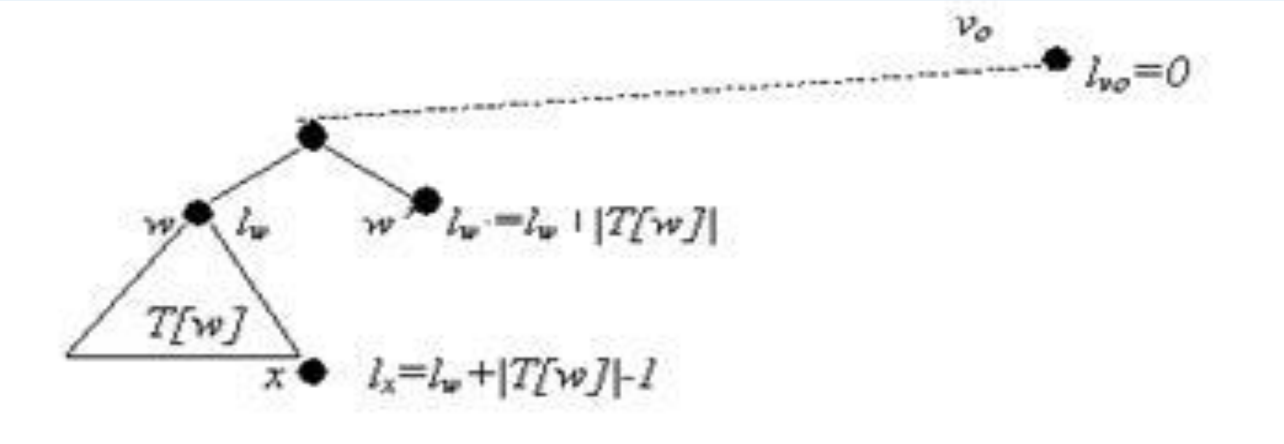

Рисунок 2 – Префиксный обход дерева

В таком порядке w – первый узел дерева T[w], который посетится, причем после w все узлы T[w] посетятся перед узлами, не входящими в T[w]; следовательно узлы в T[w] перенумерованы линейным интервалом $[l_w, l_w + |T[w]|)$, где 1_w – метка w.

Через $[a_w, b_w)$ обозначим интервал чисел означивающие узлы в T[w]. Сосед узла w – один из двух его сыновей или его отец w. Узел w передает к сыну u пакеты с пунктом назначения в T[u], то есть, узлы с числами в $[a_u, b_u)$. Узел w передает своему отцу пакеты с пунктами назначения не в T[w], то есть, узлы с номерами в $Z_N \setminus [a_w, b_w) = [b_w, a_w)$.

Циклический интервал может быть представлен использованием только 2logN бит, дающих начальный и конечный пункт. Так как объединение разъединенных интервалов должно храниться в множестве Z_N, то достаточно logN бит для каждого интервала. Хранится только начальная точка интервала для каждого канала; конечная точка эквивалентна начальной точке следующего интервала в том же узле. Начальная точка интервала, приписанного каналу uw в узле u определяется как

$a_{uw} = l_w$, если w сын u;

$a_{uw} = l_u + |T[u]|$, если w отец u.

Положим что каналы узла u со степенью deg_u помечены $a_1, ..., a_{degu}$, где $a_1 < ... < a_{degu}$, и для узла u реализуется интервальная передача. Метки каналов разбивают Z_N на сегменты deg_u, причем каждый сегмент соответствует одному каналу. Заметим, что существует (не более чем) один интервал, который не линейный. Если метки отсортированы в узле, то соответствующая метка находится за $O(\log deg_u)$ шагов при бинарном поиске. Индекс i вычисляется по модулю deg_u, то есть, $a_{deg\ u+1} = a_1$.

Интервальная передача (для узла u):

(* пакет с адресом d был получен в узле u *)

if d = l_u

then доставить пакет локально

else begin выбрать a_i, т.ч. $d \in [a_i, a_{i+1})$;

послать пакет через канал с меткой a^i

end

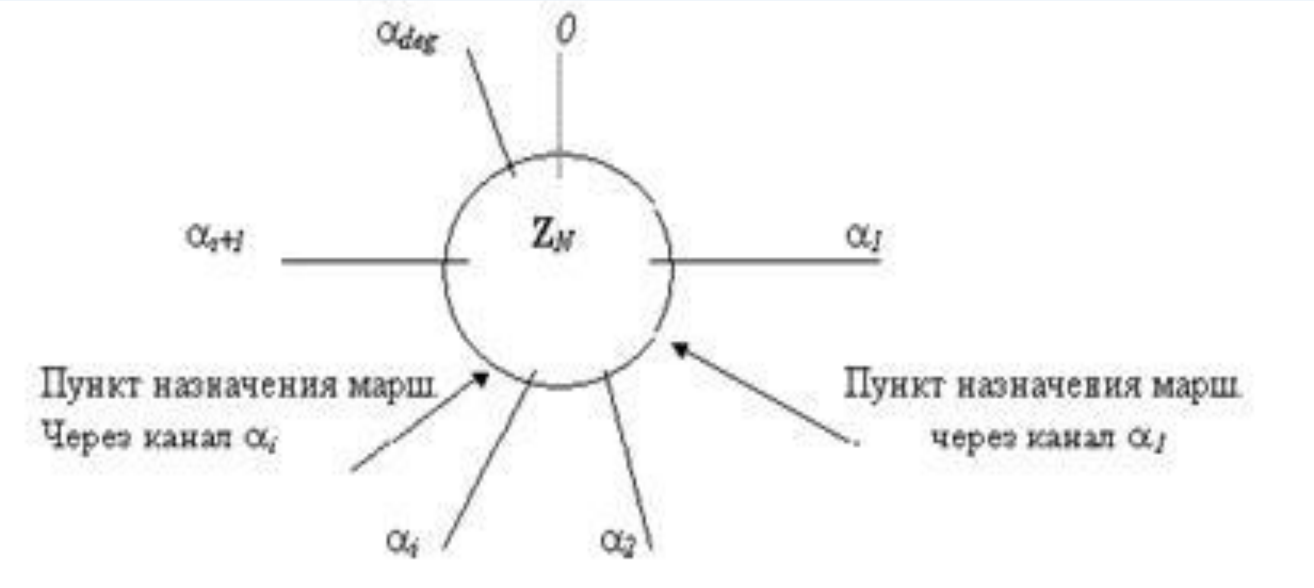

Рисунок 3 – Множество Z_N в узле

Схема разметки деревьев представляет оптимальную маршрутизацию потому, что в дереве существует только один простой путь между любыми двумя узлами. Схема может также использоваться, если сеть не является деревом. Выбирается фиксированное дерево охвата T в сети, и схема применяется на этом дереве. Каналы, не принадлежащие этому дереву, никогда не используются; каждый помечен специальной меткой в таблице маршрутизации чтобы показать, что нет пакетов, проходящих через этот канал.

Сравним длины путей, выбранные этой схемой, с оптимальными путями. Пусть $d_T(u, v)$ обозначает расстояние от u до v в T, и $d_G(u, v)$ – расстояние от u до v в G. Пусть D_G обозначает диаметр G, определенный как максимум для любых u и v из $d_G(u, v)$.

Лемма 1. Нет общего ограничения пропорции между $d_T(u, v)$ и $d_G(u, v)$.Это имеет силу только в специальном случае измерения переходами путей.

Доказательство. Определим G как кольцо из N узлов, и заметим, что дерево охвата G получается удалением одного канала, например, xy из G. Теперь $d_G(x, y) = 1$ и $d_T(x, y) = N - 1$, таким образом получаем пропорцию N – 1. Пропорция может быть гораздо больше при выборе большего кольца.

Следующая Лемма определяет симметричность стоимостей каналов, то есть, это значит, что $w_{uw} = w_{wu}$. Это подразумевает, что $d_G(u, v) = d_G(v, u)$ для всех u и v.

Лемма 2. T может быть выбрано таким образом, чтобы для всех u и v $d_T(u, v) \leq 2G$.

Доказательство. Выберем T как оптимальное дерево стока для узла w_0 (как в Теореме 1). Тогда

$d_T(u, v) \leq d_T(u, w_0) + d_T(w_0, v)$

$= d_T(u, w_0) + d_T(v, w_0)$ – в случае симметричности w;

$= d_G(u, w_0) + d_G(v, w_0)$ – в случае симметричности T;

$= D_G + D_G$ – по определению D_G.

Схема разметки деревьев имеет следующие недостатки:

- каналы, не принадлежащие T, не используются;
- трафик сосредоточен в дереве (может произойти перегрузка);
- каждый отказ канала делит сеть.

2. Интервальная маршрутизация

Ван Ливен и Тэн расширили схему разметки деревьев до сетей, не являющихся деревьями, таким образом, что каждый канал используется для передачи пакета.

Определение 2. Схема разметки деревьев (ILS) для сети это:

- обозначение различными метками из Z_N узлов сети;
- для каждого узла используется обозначение различными метками из Z_N каналов данного узла.

Алгоритм интервальной маршрутизации предполагает, что ILS дана, и пакеты передаются как в алгоритме интервальной маршрутизации.

Определение 3. Схема интервальной разметки применима, если все пакеты передаются тем путем, которым достигнут своего пункта назначения.

Можно показать, что схема интервальной разметки применима для каждой связной сети G, но для произвольной связной сети схема, как правило, не очень эффективна.

Теорема 2. Для каждой связной сети G применимая схема интервальной разметки существует.

Доказательство. Схему применимой интервальной разметки построим расширением схемы разметки деревьев Санторо и Кхатиба, применив к дереву охвата сети T. В данном дереве охвата ребро ветви – ребро, которое не принадлежит дереву охвата. Более того, v потомок u если только $u \in T[v]$. Основная проблема, как означить метки ребер ветвей (ребра дерева будут помечены схемой разметки деревьев), дерево охвата выбирается таким образом, чтобы все ребра ветвей приняли ограниченную форму.

Лемма 3. Существует дерево охвата между узлом и потомком этого узла.

Доказательство. Каждое дерево охвата, полученное обходом в глубину сети, имеет это свойство.

Пусть T будет зафиксированное дерево охвата поиска в глубину в G.

Определение 4. Поиск в глубину ILS для G (в отношении к T) – схема разметки, для которой выполняются следующие правила:

(1) Метки узлов означены префиксным обходом в T, то есть, узлы в поддереве T[w] помечены числами из $[l_w, \ldots, l_w + |T[w]|)$. Обозначим $k_w = l_w + |T[w]|$.

(2) Метку ребра uw в узле u обозначим a_{uw}:

(a) Если uw ребро ветви, то $a_{uw} = l_w$;

(b) Если w сын u (в T), то $a_{uw} = l_w$;

(c) Если w отец u, то $a_{uw} = k_u$; если $k_u \neq N$ и u имеет ветвь к корню. (В последней ситуации, ребро ветви помечаем 0 в u по правилу (a), таким образом означивание метки k_u нарушило бы требование, что все метки различны. Метки считаются по модулю N, то есть $N \equiv 0$.)

(d) Если w отец u, u имеет ветвь к корню, и $k_u \equiv N$, тогда $a_{uw} = l_w$.

Все примеры поиска в глубину ILS проиллюстрированы на рисунке 4. Заметим, что все ребра ветвей помечены по правилу (2a), ребра к отцам узлов 4, 8, и 10 помечены по правилу (2c), и ребро к отцу узла 9 помечено по правилу (2d).

Теперь покажем верность схемы обхода в глубину ILS. Заметим, что $v \in T[u] \Leftrightarrow l_v \in [l_u, k_u)$. Следующие три леммы относятся к ситуации, когда узел u передает пакет с пунктом назначения v к узлу w (соседу u), используя алгоритм интервальной маршрутизации. Это подразумевает что $l_v \in [a_{uw}, a)$ для некоторой метки a в u, и что нет метки $a' \neq a_{uw}$ в узле u такой, что $a' \in [a_{uw}, l_v)$.

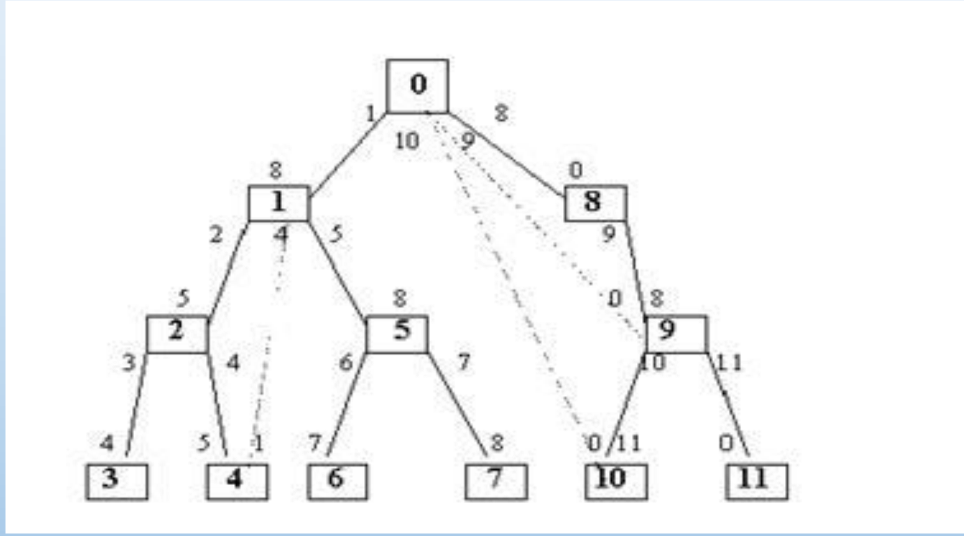

Рисунок 4 – Интервальная разметка поиском в глубину

Лемма 4. Если $l_u > l_v$, то $l_w < l_u$.

Доказательство. Во-первых, рассмотрим случай $a_{uw} \le l_v$. Узел w не сын u потому, что в этом случае $a_{uw} = l_w > l_u > l_v$. Если uw ветвь, то также $l_w = a_{uw} < l_v < l_u$. Если w отец u, то $l_w < l_u$ выполняется в любом случае. Во-вторых, рассмотрим случай, когда a_{uw} – наибольшая метка ребра в u, и нет метки $a' \le l_v$ (то есть, l_v – нижняя граница нелинейного интервала). В этом случае ребро к отцу u не помечено 0, но имеет метку k_u (потому что $0 \le l_v$, и нет метки $a' \le l_v$). Метка k_u – наибольшая метка в данном случае; ребро к сыну или ветви вниз w' имеет метку $a_{uw'} = l_{w'} < k_u$, и ветвь с предком w' имеет метку $a_{uw'} = l_{w'} < l_u$. Таким образом w – отец u в данном случае, что подразумевает $1_w < l_u$.

Следующие две леммы относятся к случаю когда $l_u < l_v$. В первом случае мы докажем, что каждый $v \in T[u]$ или $l_v > k_u$, а в во втором случае, что $k_u < N$ тогда, когда ребро к отцу u помечено k_u.

Лемма 5. Если $l_u < l_v$, то $l_w \le l_v$.

Доказательство. Во-первых, рассмотрим случай, когда $v \in T[u]$; пусть w' сын u такой что $v \in T[w']$. Мы имеем $a_{uw'} = l_{w'} \le l_v$, и это подразумевает, что $a_{uw'} \le a_{uw} \le l_v < k_{w'}$. Мы вывели, что w не является отцом u, следовательно $1_w = a_{uw}$, что подразумевает $l_w < l_v$.

Во-вторых, рассмотрим случай, когда $l_v \ge k_u$. В этом случае w отец u, это можно показать следующим образом. Ребро к отцу помечено k_u, и $k_u \le l_v$. Ребро к сыну w' узла u помечено $1_{w'} < k_u$, ребро к ветви вниз w' помечено $1_{w'} < k_u$, и ребро ветви вверх w' помечено $1_{w'} < l_u$. Поэтому w отец u, $l_w < l_u < l_v$.

Нормфункция, относящаяся к передаче в v, может быть определена следующим образом. Наименьший общий предок двух узлов u и v – это наименьший узел в дереве, который отец u и v. Пусть lca(u, v) означает метку наименьшего общего предка u и v; также определим, что

$f_v(u) = (-\,lca(u, v), 1_u)$.

Лемма 6. Если $l_u < l_v$, то $f_v(w) < f_v(u)$.

Доказательство. Во-первых, рассмотрим случай когда $v \in T[u]$, что подразумевает $lca(u, v) = l_u$. Если w' сын u такой, что v î T[w'], мы имеем (как в предыдущей Лемме), что $1_{w'} \le l_w < k_{w'}$, следовательно $w \in T[w']$, что подразумевает $lca(u, v) \ge l_{w'} > l_u$. Таким образом, $f_v(w) < f_v(u)$.

Во-вторых, рассмотрим случай, когда $1_v > k_u$. Как в предыдущей Лемме, w отец u, и поэтому $v \notin T[u]$, $lca(w, v) = lca(u, v)$. Но теперь $1_w < l_u$, таким образом $f_v (w) < f_v (u)$.

Может быть показано что каждый пакет достигает пункта назначения. Поток пакетов для v показан на рисунке 5.

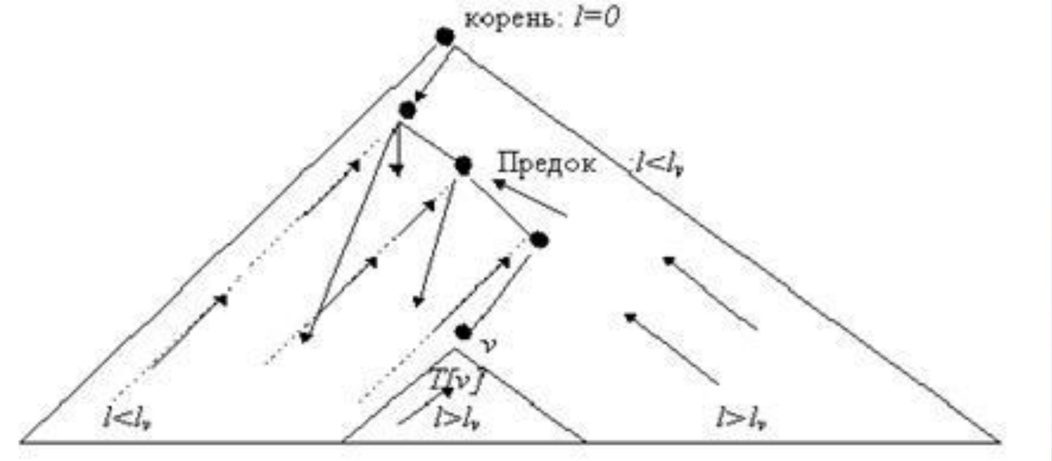

Рисунок 5 – Маршрутизация пакетов для v в схеме разметки ILS

Пусть пакет для v сгенерирован в узле u. По Лемме 4, метка узла уменьшается с каждым переходом, до тех пор пока, за конечное число шагов пакет не будет получен узлом w с $1_w \le l_v$. Каждый узел к которому пакет пересылается после w также имеет метку $\le l_v$ по Лемме 5. За конечное число шагов пакет получает v, потому что с каждым шагом f_v уменьшается или пакет прибывает в v, по Лемме 6. Это завершает доказательство Теоремы 2.

Эффективность интервальной маршрутизации: общий случай. Теорема 2 говорит о том, что корректная ILS существует для каждой сети, но не предполагает ничего об эффективности путей, выбранных схемой. Ясно что ILS с поиском в глубину используется для демонстрации существования схемы для каждой сети, но это не обязательно лучшие из возможных схем. Например, если схема с обходом в глубину применена к кольцу из N узлов, существуют узлы u и v с d(u, v) = = 2, и схема использует N – 2 переходов для передачи пакета от u к v. Также существует ILS для кольца, когда каждый пакет пересылается через путь с минимальным количеством шагов.

Определение 5. ILS оптимальна, если она передает все пакеты через оптимальные пути.

ILS общительна, если она передает пакет от одного узла к соседу данного узла за один шаг.

ILS линейна, если интервал для передачи в каждом ребре линейный.

Мы называли ILS с минимальным количеством шагов (или кратчайший путь), если она оптимальна относительно оценки пути мерой минимальным количеством шагов (или кратчайший путь, соответственно).

Просто показать, что если схема удовлетворяет мере минимального количества шагов, то схема общительна.

Также легко проверить, что ILS линейна тогда и только тогда, когда в каждом узле u с $l_u \neq 0$ существует ребро с пометкой 0, и в узле с пометкой 0 существует ребро с меткой 0 или 1. Это показывает, что для сетей в общем виде качество методов маршрутизации плохое, но для некоторых классов специальной сетевой топологии качество схемы очень неплохое. Это делает метод процессорных сетей с регулярной структурой, которые используются для реализации параллельных вычислений с виртуальной общей разделяемой памяти.

Теорема 3. Существует сеть G такая, что для каждой верной ILS в G существуют узлы u и v такие, что пакет от u к v доставлен только за не более, чем $3/2D_G$ переходов.

Доказательство. Известно как лучшие схемы ILC с поиском в глубину сравниваются с общими лучшими схемами ILS для этих же сетей.

В ситуациях, когда большинство соединений происходят между соседями, будучи общительными, достаточны требования для ILS. Так, можно показать из рисунка 4, что схема ILS с поиском в глубину не обязательно общительна: узел 4 передает пакеты для узла 2 через узел 1.

Это существенно для применимости метода интервальной маршрутизации, с помощью которого рассматриваются циклические интервалы. Хотя некоторые сети допустимы, и даже оптимальны, однако невозможно маркировать каждую сеть линейными интервалами в соответствии со схемами линейно-интервальной маршрутизации.

Применимость схем линейно-интервальной разметки была исследована Бэккером, Ван Лиуином, и Таном.

Теорема 4. Существует сеть, для которой нет применимой схемы линейно-интервальной разметки.

Доказательство. Рассмотрим граф-паук с тремя ногами длины 2, как представлено на рисунке 6.

Рисунок 6 – Граф-паук с тремя ногами

Наименьшей меткой (0) и наибольшей меткой (6) означены два узла, и так как всего три ноги, то существует (по крайней мере) одна нога, которая не содержит ни меньшую, ни большую метку. Пусть x будет первым узлом от центра в этой ноге. Узел x передает пакеты, адресованные к 0 и 6 в центр, и единственный линейный интервал, который содержит и 0, и 6 – это полное множество Z_N. Следовательно, x также пересылает пакеты для своих соседей через центр, и эти пакеты никогда не достигнут своих пунктов назначения.

Бэккер, Ван Лиуин и Тан полностью описали класс сетей, топологии которых допускают линейные схемы ILS кратчайших путей, и представили результаты, содержащие классы графических топологий, которые допускают адаптацию, и что линейные схемы ILS с минимальным количеством шагов линейны.

Оптимальность интервальной маршрутизации: специальные топологии. Было показано, что существуют оптимальные схемы интервальной разметки для некоторых классов сетей, имеющих регулярную структуру. Сети таких структур используются, например, в реализации параллельных вычислений.

Теорема 5. Существует схема ILS с минимальным количеством шагов для кольца из N узлов.

Доказательство. Метки узлов означены от 0 до N – 1 по часовой стрелке. Для узла i канал по часовой стрелке означен меткой i + 1, а канал против часовой стрелки означен (i + [N/2]) mod N (рисунок 7).

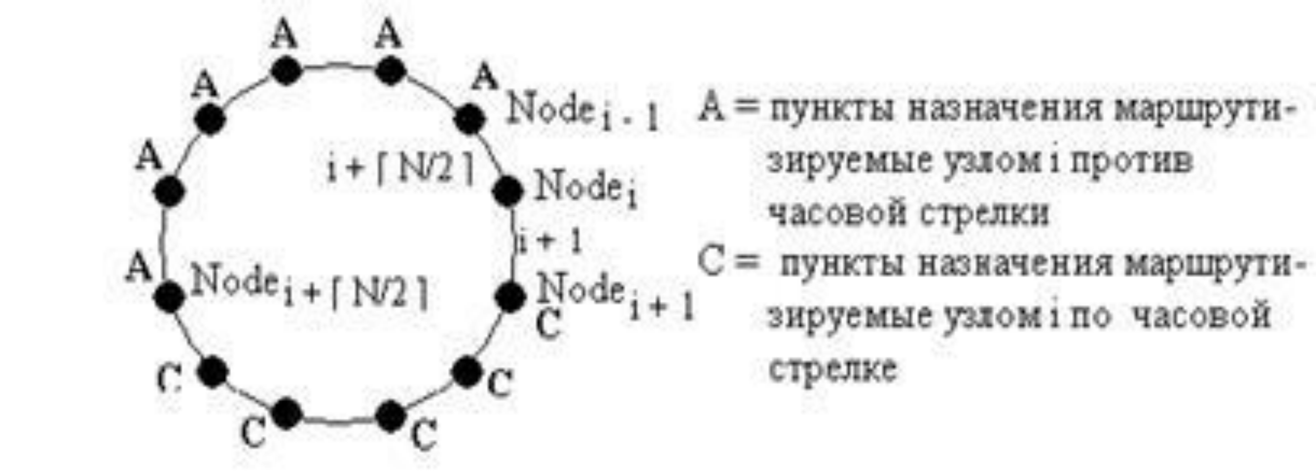

Рисунок 7 – Оптимальная ILS для кольца

В соответствии с такой схемой разметки узел с меткой i посылает пакеты для узлов i + 1, ..., (i + [N/2]) – 1 через канал по часовой стрелке и пакеты для узлов (i + [N/2]), . . . , i – 1 через канал против часовой стрелки, что является оптимальным.

Так как ILS в доказательстве Теоремы 5 оптимальна, то она общительна и линейна.

Теорема 6. Существует схема ILS с минимальным количеством шагов для сетки n×n.

Доказательство. Метки узлов означены по рядам в возрастающем порядке, то есть, i-й узел в j-м ряду помечен (j – l)n + (i – 1) . Канал вверх этого узла помечен 0, канал налево этого узла помечен (j – l)n, канал направо помечен (J – l)n + i, и канал вниз помечен jn (рисунок 8).

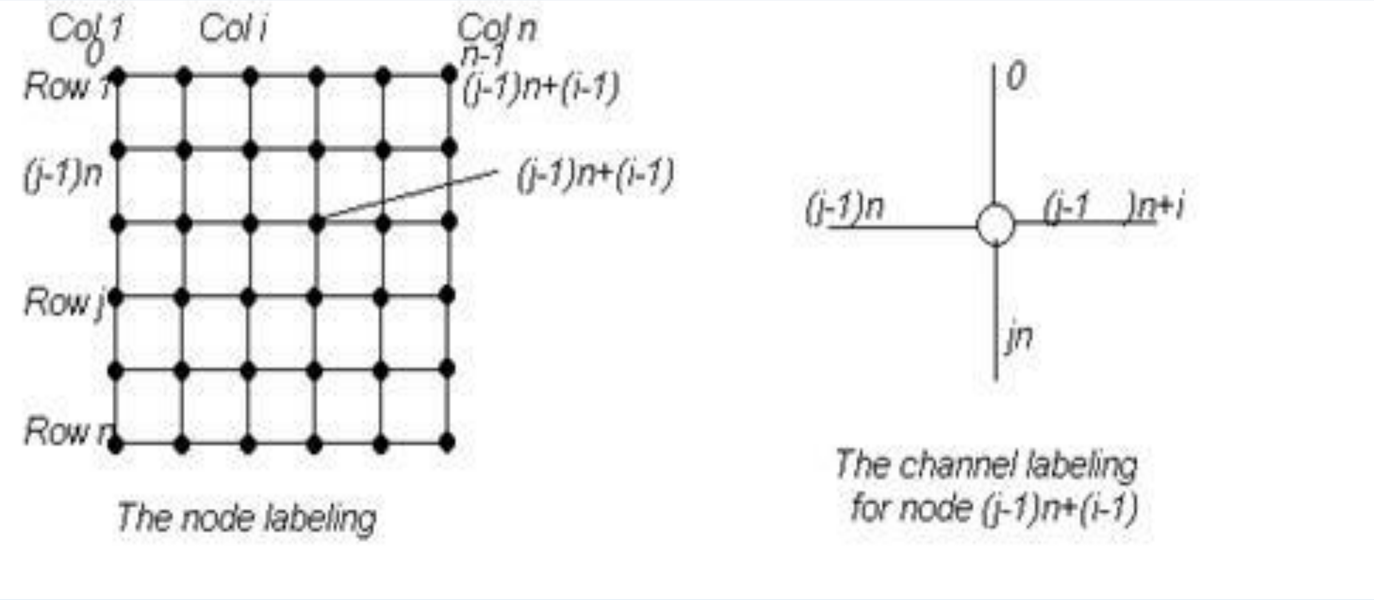

Рисунок 8 – Оптимальная ILS для сетки n×n

Теперь легко проверить, что когда узел u передает пакет к узлу v:

Случай 1: если v в ряду большем чем u, тогда u посылает пакет через свой канал наверх;

Случай 2: если v в ряду меньшем чем u, тогда u посылает пакет через свой канал вниз;

Случай 3: если v в том же ряду что и u, но левее, u посылает пакет через свой левый канал; и

Случай 4: если v в том же ряду что и u, но правее, то u посылает пакет через свой канал направо.

Во всех случаях u посылает пакет к узлу ближайшему к v, что и подразумевает что выбранный путь оптимальный.

Так как ILS в Доказательстве Теоремы 6 оптимальна и общительна, то схема также линейна.

Теорема 7. Существует линейная схема ILS с минимальным количеством шагов для гиперкуба.

Теорема 8. Существует схема ILS кратчайших путей для непланарных сетей с произвольными весами каналов.

Интервальная маршрутизация имеет некоторые привлекательные ***преимущества*** перед механизмами классической маршрутизации, основанными на хранении привилегированных каналов отдельно для каждого пункта назначения:

(1) Малая пространственная сложность. Таблицы маршрутизации могут хранится в 0(deg logN) бит для узла степенью deg.

(2) Эффективность вычислений таблиц маршрутизации. Таблицы маршрутизации для схем ILC с поиском в глубину могут быть вычислены с применением распределенного обхода сети в глубину, который может использовать O(E) сообщений за время 0(N).

(3) Оптимальность. Метод маршрутизации способен выбирать оптимальный путь в некоторых классах сетей.

Эти преимущества делают метод применимым для процессорных сетей с регулярной топологией.

К сожалению, для сетей с произвольной топологией, когда методы используют схемы ILS с поиском в глубину, присутствует несколько ***минусов***:

(1) Плохая живучесть. Невозможна легкая адаптация схемы ILS с поиском в глубину при добавлении или удалении узла в сети. Дерево ILS не может долго удовлетворять требованию, по которому ветвь существует только между узлом и его предком. В результате минимальное изменение топологии сети может потребовать полного пересчета таблиц маршрутизации, включая вычисление новых адресов (меток) для каждого узла.

(2) Неоптимальность. Схема ILS поиска в глубину может направлять пакет через пути длиной W(N), даже в случаях сетей с малым диаметром.

3. Префиксная маршрутизация

Рассмотрев недостатки интервальной маршрутизации, Бэккер, Ван Лиуин, и Тан разработали метод маршрутизации, в котором таблицы могут быть вычислены, используя произвольное дерево охвата. Использование неограниченного дерева охвата может увеличить как живучесть, так и эффективность. Если а канал добавлен между двумя существующими узлами, то дерево охвата остается деревом охвата, а новый канал будет ветвью. Если новый узел добавляется вместе с некоторым количеством каналов, соединяющих его с существующими узлами, то дерево охвата расширяется, используя один из каналов и новый узел. Остальные каналы становятся ветвями. Оптимальность может быть улучшена выбором дерева охвата малой глубины.

Как метки узлов и каналов в префиксной маршрутизации предпочтительнее использовать строки чем целые числа, используемые в интервальной маршрутизации. Пусть S – алфавит; в последствии метка будет строкой над S, s обозначает пустую строку, и S* множество строк над S. Для отбора канала для передачи пакета, алгоритм рассматривает все каналы, которые являются префиксами адреса пункта назначения. Выбирается самая длинная из этих меток, и выбранный канал используется для передачи пакета. Например, предположим, что узел имеет каналы с метками aabb, abba, aab, aabc и aa, и должен переслать пакет с адресом aabbc. Метки каналов aabb, aab и aa являются префиксами aabbc, и самая длинная из этих трех меток aabb, следовательно, узел передаст пакет через канал помеченный aabb. Считаем a < b для обозначения, что a – префикс b.

***Алгоритм префиксной маршрутизации (передачи данных) (для узла u)*:**

(* A пакет с адресом d был получен или создан узлом u *)

i f d = l_u

then обработать пакет локально

else begin a_i : = самая длинная метка канала так, что $a_i < d$;

послать пакет через канал помеченый a_i

end

Определение 6. Схема префиксной разметки (над S) для сети G это:

(1) обозначение различными строками из S* узлов G;

(2) Для каждого узла означивание производится различными строками каналов данного узла.

Алгоритм префиксной маршрутизации предполагает, что схема префиксной разметки (PLS) дана.

Определение 7. Схема префиксной разметки приемлема, если все пакеты в конечном счете достигнут своих пунктов назначения.

Теорема 9. Для каждой связной сети G существует приемлемая схема PLS.

Доказательство. Мы определим класс схем префиксной разметки и докажем, что схемы в этом классе приемлемы. Пусть T обозначает произвольное дерево охвата в G.

Определение 8. Дерево T схемы PLS для G – это схема префиксной разметки, при которой выполняются следующие правила:

(1) Метка корня – s.

(2) Если w сын u, то 1w расширяет l_u одним символом; то есть, если u_1, ..., u_k – сын u в T, то $1_{ui} = l_u a_i$, где a_1, ... , a_k – k различных символов из S.

(3) Если uw ветвь, то $a_{uw} = l_w$.

(4) Если w сын u, то $a_{uw} = l_w$.

(5) Если w отец u, то a_{uw} = s, и если u не имеет ветви к корню, то в этом случае $a_{uw} = l_w$.

В дереве PLS каждый узел исключая корень имеет канал, помеченный s, и этот канал соединяет узел с предком (отцом узла или корнем дерева). Заметим, что для каждого канала uw, $a_{uw} = l_w$ или a_{uw} = s. Для всех u и v, v предок u тогда и только тогда, когда $l_v < l_u$.

Нужно показать, что пакет никогда не «вклинивается» в узел отличный от его пункта назначения, который, каждый узел отличный от пункта назначения может перенаправить пакет используя алгоритм префиксной маршрутизации.

Лемма 7. Для всех узлов u и v таких, что $u \neq v$, существует канал в u, помеченный префиксом l_v.

Доказательство. Если u не корень T, то u имеет канал помеченный s, который является префиксом l_v. Если u корень, тогда v не является корнем, и $v \in T[u]$. Если w сын u такой, что $v \in T[w]$, то по построению $a_{uw} < l_v$.

Следующие три леммы имеют отношение к ситуации, когда узел u передает пакет для узла v к узлу w (соседу u), используя алгоритм префиксной маршрутизации.

Лемма 8. Если $u \in T[v]$, то w предок u.

Доказательство. Если a_{uv} = = s, то w предок u (как упоминалось выше). Если $a_{uw} = l_w$, то, так как $a_{uw} < l_v$, также $l_w < l_v$. Это подразумевает, что w является предком v и u.

Лемма 9. Если u предок v, то w предок v, ближе к v чем u.

Доказательство. Пусть w' будет сыном u таким что $v \in T[w']$, тогда $a_{uw'} = l_w$ не пустой префикс l_v. Так как a_{uw} самый длинный префикс (в u) l_v, то $a_{uw'} < a_{uw} < l_v$, таким образом w предок v ниже u.

Лемма 10. Если $u \notin T[v]$, то w предок v или $d_T(w, v) < d_T(u, v)$.

Доказательство. Если a_{uw} = s, то w отец u или корень; отец u ближе к v чем u, потому что $u \notin T[v]$, и корень – предок v. Если $a_{uw} = l_w$, то, так как $a_{uw} < l_v$, w – предок v.

Пусть depth будет обозначать глубину T, то есть, число переходов в самом длинном простом пути от корня к листьям. Может быть показано, каждый пакет с пунктом назначения v прибудет в свой пункт назначения за не более, чем 2*depth переходов.

Если пакет создан в предке v, то v достигнется не более, чем за depth переходов по Лемме 9.

Если пакет создан в поддереве T[v], тогда предок v достигнется за не более, чем за depth переходов по Лемме 8, после которых v достигнется за другие depth переходов по Лемме 9 (по причине того, что путь содержит только предков источника в этом случае, его длина ограничена также depth).

Во всех других случаях предок v достигнется в пределах depth переходов по лемме 10, после этого v достигнется в пределах других depth переходов (таким образом, в этом случае длина пути ограничена 2*depth.) Это завершает доказательство Теоремы 9.

Следствие 1. Для каждой сети G с диаметром D_G (измеренным в переходах) существует схема префиксной разметки, которая доставляет все пакеты за не более чем $2D_G$ переходов.

Доказательство. Воспользуемся деревом PLS, выбранным в Лемме 2.

Рассмотрим схему разметки деревьев с грубым анализом его пространственных требований. Как и раньше, depth – глубина T, и пусть k будет максимальным количеством сыновей любого узла T. Тогда, самая длинная метка состоит из depth символов, и S должен содержать (по крайней мере) k символов, метка может храниться в depth*logA бит. Таблица маршрутизации а узла с deg каналами хранится в 0(deg*depth*logk) бит.

Несколько другая схема префиксной разметки бала предложена Бэккером. Его подход также характеризует класс топологий, который допускает оптимальные схемы префиксной разметки, когда веса связей могут меняться динамически.

4. Иерархическая маршрутизация

Путь сокращения различных параметров стоимости метода маршрутизации – использование иерархического разделения сети и метода ассоциативной иерархической маршрутизации. Цель в большинстве случаев состоит в том, чтобы использовать факт, что многие связи в сетях компьютеров являются локальными, то есть, с узлами на относительно малых расстояниях друг от друга. Некоторые из параметров стоимости метода маршрутизации зависят от размера полной сети больше, чем длина выбранного пути, по следующим причинам:

(1) Длина адресов. Так как каждый из N узлов имеет отличный от других адрес, каждый адрес состоит из по крайней мере logN бит; может потребоваться даже больше бит, если существует информация, включенная в адреса, такая как в префиксной маршрутизации.

(2) Размер таблицы маршрутизации. При маршрутизации по алгоритмам кратчайших путей, Флойда-Уоршелла и Netchange, таблица содержит ссылку на каждый узел, и таким образом имеет линейный размер.

(3) Цена табличного поиска. Цена простого табличного поиска больше для большей таблицы маршрутизации или для больших адресов. Полное время табличного поиска для обработки простого сообщения также зависит от количества обращений к таблице.

В методе иерархической маршрутизации, сеть разделена на кластеры, каждый кластер есть связное подмножество узлов. Если источник и пункт назначения пакета находятся в одном кластере, то цена передачи сообщения низка, потому что пакет маршрутизируется внутри кластера, кластер трактуется как небольшая изолированная сеть. Если в каждом кластере зафиксирован простой узел (центр кластера), то можно реализовывать сложные маршрутизационные решения, необходимые для пересылки пакетов в другие кластера. Таким образом, большие таблицы маршрутизации и манипуляция длинными адресами необходимы только в центрах. Каждый кластер сам может разделиться на подкластеры для многоуровневого деления узлов.

Не необходимо, но желательно, чтобы каждая коммутация между кластерами велась через центр; этот тип конструкции имеет такой недостаток – весь кластер становится уязвимым при отказе центра.

Лентферт предложил метод иерархической маршрутизации, в котором все узлы в равной степени могут посылать сообщения другим кластерам. Также метод использует только маленькие таблицы, потому что ссылки на кластеры, к которым узел не принадлежит, трактуется как простой узел.

Овербух использует парадигму иерархической маршрутизации для конструирования класса схем маршрутизации, которые всегда балансируют между эффективностью и пространственными требованиями.

Все обсужденные методы маршрутизации требуют, чтобы решения маршрутизации принимались в каждом промежуточном узле, что подразумевает что для маршрута длиной l происходит l обращений к таблицам маршрутизации. Для алгоритмов с минимальным количеством шагов l ограничено диаметром сети, но в целом, для алгоритмов маршрутизации без циклов (такие как интервальная маршрутизация) N – 1 – лучшая граница, которая может быть достигнута.

Рассмотрим, как можно уменьшить таблицы поиска.

Используем следующую лемму, которая подразумевает существование подходящей для разбиения на связные кластеры сети.

Лемма 11. Для каждого $s \le N$ существует разбиение сети на кластера $C_1, \ldots, C_m$ такие, что:

(1) каждый кластер – связный подграф,

(2) каждый кластер содержит по крайней мере s узлов,

(3) каждый кластер имеет радиус не более, чем 2s.

Доказательство. Пусть $D_1, \ldots, D_m$ будет максимальное подмножество разделенных связных подграфов таких, что каждый D_i имеет радиус $\le s$ и содержит по крайней мере s узлов. Каждый узел, не принадлежащий $\bigcup^{m} D_i$, соединен с одним из подмножеств путем длиной не больше, чем s, иначе путь может быть добавлен как отдельный кластер. Сформируем кластеры C_i включением каждого узла, не входящего в $\bigcup^{m} D_i$, в кластер, ближайший к нему. Расширенные кластеры содержат по крайней мере s узлов каждый, они остаются связными и разделенными, и они имеют радиус не более чем 2s.

Метод маршрутизации означивает цветом каждый пакет. Предполагается, что используется только несколько цветов. Узлы теперь действуют следующим образом. В зависимости от цвета, пакет или отправляется немедленно по установленному каналу (соответствующему цвету) или в соответствии с более сложным решением. Это подразумевает, что узлы имеют различные протоколы для обработки пакетов.

Теорема 10. Для каждой сети из N узлов существует метод маршрутизации, который требует не более, чем $0(\sqrt{N})$ решений маршрутизации для каждого пакета, и использует три цвета.

Доказательство. Предположим что решения (по Лемме 11) даны и заметим что каждый C_i содержит узел c_i такой что $d(v, c_i) \le 2s$ для каждого $v \in C_i$, потому что C_i имеет радиус не более, чем 2s. Пусть T будет поддеревом минимального размера из G соединяющее все c_i. Так как T минимально, то оно содержит не более, чем m листьев, следовательно, оно содержит не более, чем m – 2 узлов разветвлений (узлы степенью большей чем 2).

Рассмотрим узлы T как центры (c_i), узлы разветвлений и узлы пути.

Метод маршрутизации сначала посылает пакет к центру c_i кластера источника (зеленая фаза), затем через T к центру c_j кластера пункта назначения (синяя фаза), и наконец внутри C_j к пункту назначения (красная фаза).

Зеленая фаза использует фиксированному дерево стока для центра каждого кластера, а не решений маршрутизации. Узлы пути в T имеют два инцидентных канала и передают каждый синий пакет через каналы в дереве, которые не принимают пакет. Узлы ветвлений и центры в T должны принимать решения маршрутизации.

Для красной фазы используется алгоритм кратчайшего пути внутри кластера, который ограничивает число решений в этой фазе до 2s. Это в целом ограничивает число решений маршрутизации до 2m – 2 + 2s, что не более, чем 2N/s – 2 + 2s . Выбор $s \approx \sqrt{N}$ дает ограничение $0(\sqrt{N})$.

Теорема 10 устанавливает границу общего числа решений маршрутизации, необходимого для обработки каждого пакета, но не полагается на любой практический алгоритм, с помощью которого эти решения принимаются. Метод маршрутизации использованный в T может быть схемой маршрутизации деревьев Санторо и Кхатиба, но также возможно применить принцип кластеризации к T, тем самым уменьшив число решений маршрутизации еще больше.

Теорема 11. Для каждой сети из N узел s и каждого положительного целого числа $f \le \log N$ существует метод маршрутизации, который требует не более чем $O(f N^{1/f})$ решений маршрутизации для каждого пакета, и использует $2f + 1$ цветов.

Доказательство. Доказательство подобно доказательству теоремы 10, но вместо выбора $s \approx \sqrt{N}$ конструирование применяется рекурсивно к дереву T (оно является кластером размера s). Дерево – связная сеть, по существу < 2m узлов, потому что узлы пути в T только перенаправляют пакеты из одного фиксированного канала в другой, и он может быть игнорирован.

Кластеризация повторяется f раз. Сеть G имеет N узлов. Дерево содержит после одного уровня кластеризации не более чем N/s центров и N/s узлов ветвления, то есть, N (2/s) необходимых узлов. Если дерево полученное после i уровней кластеризации имеет m_i необходимых узлов, тогда дерево полученное после i + 1 уровней кластеризации имеет не более чем m_i/s центров и m_i/s узлов ветвлений, то есть, $m_i(2/s)$ необходимых узлов. Дерево полученное после f уровней кластеризации имеет не более, чем $m_f = N (2/s)^f$ необходимых узлов.

Каждый уровень кластеризации увеличивает количество цветов на два, следовательно с f уровнями кластеризации будут использоваться $2f + 1$ цветов. Не более чем $2m_f$ решений необходимо на самом высоком уровне, и s решений необходимо на каждом уровне кластеризации в кластере пункта назначения, отсюда количество решений маршрутизации $2m_f + fs$. Выбирая $s \approx 2N^{1/f}$ получим $m_f = O(1)$, следовательно число решений маршрутизации ограничено $f_s = O(f N^{1/f})$.

Использование приблизительно logN цветов приводит к методу маршрутизации, которые требуют O(logN) решений маршрутизации.

Выводы

В ходе лекции рассмотрены следующие вопросы:

- схема разметки деревьев;

- интервальная маршрутизация;

- префиксная маршрутизация;

- иерархическая маршрутизация.

Задание на самостоятельную работу

1. Конспект лекций.

Вид и тема следующего занятия

Практическое занятие №9. Маршрутизация (ч. 1)