

УДК 004.7
DOI: 10.15827/0236-235X.141.071-082

Дата подачи статьи: 01.06.22, после доработки: 13.09.22
2023. Т. 36. № 1. С. 071–082

Разработка и моделирование гибридного протокола динамической маршрутизации

Л.И. Абросимов¹, д.т.н., профессор, AbrosimovLI@mpei.ru

Х. Хаю¹, аспирант, hussain.khayou@gmail.com

М.А. Орлова¹, к.т.н., ассистент, OrlovaMA@mpei.ru

¹ Национальный исследовательский университет «МЭИ», г. Москва, 111250, Россия

Расширение сетевых услуг обуславливает существенное повышение требований к качеству и скорости решения задач сетевого управления постоянно растущими сетями в центрах обработки данных. Рост нагрузки приводит к необходимости структурного масштабирования, заключающегося в увеличении количества серверов и маршрутизаторов. Существует потребность в простых масштабируемых протоколах маршрутизации для облегчения автоматизации и управления постоянно растущими сетями, особенно в центрах обработки данных.

Целью работы является представление разработанного гибридного протокола динамической маршрутизации, включающего модернизированный алгоритм дистанционно-векторной маршрутизации и алгоритм состояния канала.

В статье показано решение проблем разработки гибридного протокола динамической маршрутизации, который гарантирует отсутствие циклов, обеспечивает требования масштабирования посредством разработки и реализации простых алгоритмов, позволяющих обеспечить надежную передачу рабочего и служебного трафиков, содержащих информацию о маршруте, и обнаружить подключенные к текущему маршрутизатору каналы, сети и непосредственно подключенные соседние маршрутизаторы. Требования масштабируемости нового гибридного протокола динамической маршрутизации выполняются за счет того, что алгоритм дистанционно-векторной маршрутизации вычисляет расстояния до узлов инфраструктуры, а не сами сетевые префиксы. Объявление сетевых префиксов производится алгоритмом состояния канала только один раз, что приводит к уменьшению БД о состоянии канала и к сокращению вычислений после изменения топологии. Исключение петель достигается за счет внедрения нового разработанного алгоритма распределенных порядковых номеров.

Для моделирования гибридного протокола динамической маршрутизации разработана имитационная модель. Моделирование позволило оценить количество и объем служебного трафика, что подтвердило эффективность функционирования разработанного протокола в условиях масштабирования сети центров обработки данных.

Ключевые слова: масштабируемость протокола маршрутизации, бесцикловая маршрутизация, модель, алгоритм, моделирование протоколов маршрутизации.

В последнее десятилетие интенсивное использование облачных сервисов, предоставляемых центрами обработки данных (ЦОД), привело к значительному увеличению трафика передаваемых и обрабатываемых данных.

Современный ЦОД включает серверный комплекс, систему хранения данных, систему эксплуатации и систему информационной безопасности, которые интегрированы между собой и объединены высокопроизводительной локальной вычислительной сетью. Солидные российские интернет-компании имеют уже по несколько ЦОД. Например, Яндекс открыл новый (уже четвертый по счету) дата-центр на 3 тысячи серверов.

Обычно рост нагрузки на ЦОД приводит к необходимости структурного масштабирования, заключающегося в увеличении количества

серверов и маршрутизаторов. При этом серьезные проблемы возникают при настройке протоколов маршрутизации, которые не всегда эффективно выполняют требования масштабируемости ЦОД.

Требование масштабируемости выполняется протоколом маршрутизации, если нагрузка служебного трафика, генерируемого протоколом маршрутизации, растет пропорционально количеству маршрутизаторов и/или каналов связи.

Использование доменной организации топологии сети ЦОД еще больше ужесточает требования к протоколам маршрутизации. Это, в свою очередь, привело к тому, что набор сетевых требований ЦОД с точки зрения топологии, масштаба и протокола маршрутизации в последнее время значительно изменился.

Протокол маршрутизации должен обладать простотой реализации по параметрам сложности программного кода и простоты операционной поддержки. Наборы функций и протоколов маршрутизации должны поддерживаться несколькими вендорами. Кроме того, указываются периоды повторных опросов и радиусы сетей, определяющие отказы обслуживания протоколом маршрутизации или оборудования.

Вычисление маршрутов рабочего и служебного трафика становится сложной задачей для протоколов маршрутизации [1–3]. Вот почему крупные предприятия прибегают к использованию *протокола динамической маршрутизации* (ПДМ) BGP (Border Gateway Protocol) в сверхбольших ЦОД [1, 3, 4]. Однако BGP подвержен постоянным колебаниям маршрута [5] и отличается медленной скоростью конвергенции таблицы маршрутизации из-за проблемы поиска путей [6]. К тому же BGP основан на префиксах, идентифицирующих маршрутизаторы, поэтому сбой одного канала связи может привести к повторному вычислению и передаче обновлений для сотен префиксов [7].

Проприетарный бесцикловый протокол маршрутизации EIGRP, разработанный Cisco, был опубликован в 2013 г. как имеющий информационный характер [8]. Ранее EIGRP продвигался как гибридный протокол маршрутизации, однако теперь Cisco классифицирует его как расширенный дистанционно-векторный протокол [8, 9]. В EIGRP используется алгоритм диффузного обновления DUAL, в котором узел может изменить свой следующий переход только при удовлетворении условия выполнимости [10]. Условие выполнимости гарантирует, что при выборе выполнимого маршрутизатора для определенного пункта назначения узел не образует петлю в маршрутизации к этому пункту назначения. Преимущество EIGRP заключается в быстрой сходимости и эффективности в малых и средних сетях, к тому же он поддерживает многопутевую маршрутизацию с неравной стоимостью, используя концепцию выполнимых маршрутизаторов. Однако EIGRP не был принят другими поставщиками. Кроме того, он не поддерживает механизм разделения домена маршрутизации на области или поддомены. Вычисления EIGRP также основаны на префиксах.

Протокол маршрутизации Babel – это протокол дистанционно-векторной маршрутизации [11], в котором к метрике добавляется убывающий порядковый номер. Babel обладает

широкими возможностями для ограничения циклов маршрутизации. В отличие от DUAL для избежания циклов маршрутизации Babel не требует распределенных вычислений, при которых узел становится активным и синхронизирует свои вычисления с другими узлами в сети. Преимущества протокола Babel в простоте и небольшом размере. Он подходит для маршрутизации в высокودинамичных беспроводных ячеистых сетях, однако недостаточно хорош для больших и относительно стабильных сетей, так как зависит от периодических обновлений [11].

В протоколе Babel также используются условия выполнимости для обнаружения маршрутов с петлями. Истощение (голодание) возникает, когда у маршрутизатора заканчиваются выполнимые маршрутизаторы. Тогда этот маршрутизатор выдает явный одноадресный запрос владельцу маршрута для увеличения порядкового номера. Запрос пересылается поэтапно владельцу маршрута независимо от условия выполнимости, чтобы добраться до владельца. При получении запроса владелец маршрута увеличивает свой порядковый номер и в широковещательном режиме передает обновление, которое пересылается запросившему узлу.

Основным недостатком протокола Babel является то, что некоторые вышестоящие узлы, не получившие информацию о недействительности их текущих маршрутов, по-прежнему будут перенаправлять трафик в соответствии с их недействительной информацией о маршруте и трафик будет отбрасываться до тех пор, пока не поступит новая информация от владельца маршрута. В этом протоколе также используются периодические обновления, и он зависит от надежности обмена информацией о маршруте. Другим недостатком является то, что необходимо распределять несколько явных запросов для каждого голодающего узла (это может повлиять на все узлы в его восходящем дереве), что, в свою очередь, приводит к повторению передачи. Кроме того, запросы могут быть потеряны (или образовать маршрутную петлю) и, поскольку они отправляются ненадежно, владелец может не получить никакого запроса на обновление маршрута. Babel основан на префиксе, а это может образовать маршрутную петлю [11].

В данной статье предлагается *гибридный протокол динамической маршрутизации* (ГПДМ). В протоколе используется модернизированный алгоритм распределенных после-

довательных номеров DSN (Distributed Sequence Number) на основе Babel и EIGRP.

Предложенный протокол также включает разработанный авторами алгоритм состояния канала, распределяющий информацию о состоянии каналов между сетями и маршрутизаторами инфраструктуры.

Корректность ГПДМ и его алгоритмов подтверждается результатами анализа алгебраической модели режима бесциклового маршрутизации, представленной в более ранних работах авторов [12–14], и результатами моделирования, приведенными в настоящей статье.

Постановка задачи

Дана составная сеть G , содержащая наборы взаимосвязанных маршрутизаторов ($R_1, R_2, \dots, R_i, R_j, \dots, R_M$) и сетей ($S_1, S_2, \dots, S_i, \dots$), соединенных посредством каналов K_{ij} связи (рис. 1). Требуется разработать ГПДМ, который в условиях динамического изменения топологии определяет для каждого маршрутизатора кратчайший путь к любой сети S_i в G и выполняет требования масштабируемости.

Любой ГПДМ создается, чтобы усовершенствовать характеристики ПДМ, сгруппированные функции которого выполняются поэтапно.

- *Этап 1: обнаружение соседей.* Посредством обмена служебными сообщениями маршрутизаторы обнаруживают друг друга, сравнивают параметры (например, адреса сетей, индикаторы маршрутизаторов и маршрутов) и определяют, должны ли они образовывать соседство.

- *Этап 2: обмен топологиями.* Если соседние маршрутизаторы решают сформировать соседство, они обмениваются друг с другом служебными сообщениями, включающими сведения своих таблиц топологии. Затем между маршрутизаторами передаются только

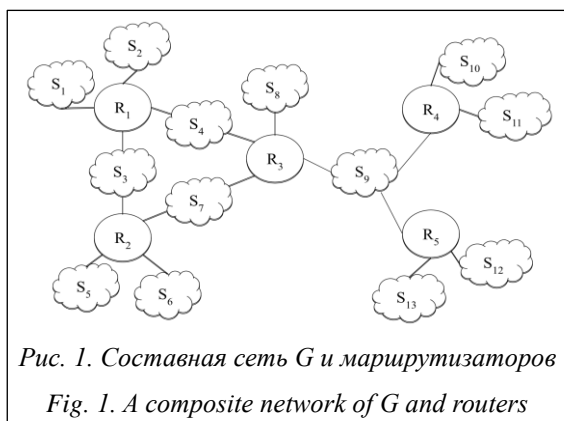


Рис. 1. Составная сеть G и маршрутизаторов
Fig. 1. A composite network of G and routers

служебные сообщения при изменении существующей топологии и/или через определенные интервалы времени (в зависимости от протокола).

- *Этап 3: выбор маршрутов.* Как только таблица топологии маршрутизатора заполнена, маршрутизаторы (в соответствии с протоколом) участвуют в вычислении лучшего маршрута сети.

Соединения в составной сети G являются двунаправленными. Каждый маршрутизатор R_i имеет уникальный идентификатор i . Маршрутизатор считается владельцем маршрута сети, если между этим маршрутизатором и сетью имеется непосредственный канал связи.

Маршрутизатор R_j – владелец маршрута (он же маршрутизатор назначения), он инициирует вычисление стоимостей (расстояний) маршрутов (путей) ко всем маршрутизаторам через коммутируемые каналы связи.

Каждый маршрутизатор R_i поддерживает таблицу стоимостей маршрутов для каждого другого маршрутизатора R_j . Вектор стоимости маршрута между текущим маршрутизатором и владельцем указан в виде $(s; d)$, где s – порядковый номер маршрутизатора, который является положительным целым числом, присвоенным владельцем маршрута маршрутизатором R_j ; d – расстояние, которое может быть составной метрикой, однако в этой статье принимается, что d является положительным числом. Стоимость $(s_1; d_1)$ считается меньше (дешевле или предпочтительнее), чем стоимость $(s_2; d_2)$, если $s_1 > s_2$ или $s_1 = s_2$ и $d_1 \leq d_2$. Соединения между напрямую подключенными маршрутизаторами R_i и R_k характеризуются расстоянием $d_{(i,k)}$. Каждый маршрутизатор R_i назначает вектор выполнимой стоимости \overline{FD}_j^i другим маршрутизаторам R_j . Вектор выполнимый стоимости \overline{FD}_j^i к маршрутизатору R_j , известный в маршрутизаторе R_i , является наименьшей стоимостью для маршрутизатора R_j , вычисленной в маршрутизаторе R_i . Каждый маршрутизатор R_i также хранит векторы объявленных стоимостей $\overline{RD}_{k,j}^i: R_k \in N_i, R_j \in G$ (где N_i – множество соседей маршрутизатора R_i) для своих непосредственно подключенных соседей ко всем маршрутизаторам в сети. Сосед $R_k \in N_i$ является одним из выполнимых маршрутизаторов R_i к маршрутизатору R_j назначения (то есть $R_k \in FS_j^i$), если вектор объявленной стоимости $\overline{RD}_{k,j}^i$ к маршрутизатору R_j строго дешевле,

чем вектор выполнимой стоимости \overline{FD}_j^i маршрутизатора R_i к маршрутизатору R_j (то есть $\overline{RD}_{k,j}^i < \overline{FD}_j^i$). Следующий маршрутизатор R_s ($R_s \in N_i$), подключенный к маршрутизатору-владельцу R_j , является одним из соседних маршрутизаторов R_i , если стоимость (\overline{D}_j^i) от маршрутизатора R_i до маршрутизатора R_j является самой низкой при вычислениях с использованием объявленной стоимости ($\overline{RD}_{s,j}^i$). Обновления, содержащие d и s , являются служебными сообщениями, которые доставляются по надежным соединениям.

Разрабатываемый ГПДМ должен устранить недостатки известных ПДМ.

В ПДМ Babel:

- маршрутизаторы, не получившие обновлений, продолжают пересылать служебный трафик, который не доставляется получателям;
- при использовании префиксов в служебных сообщениях возможны циклы служебного и рабочего трафиков;
- не выполняются требования масштабирования, поэтому протокол используется для небольших сетей, самый длинный путь в которых составляет 15 переходов.

В ПДМ EIGRP:

- не поддерживается механизм разделения домена маршрутизации на области;
- вычисления маршрутов основаны на префиксах и используют медленные алгоритмы расчета;
- условие выполнимости вычисляется для всех маршрутов сети;
- отсутствуют доказательства полноты условий выполнимости переходов, а следовательно, и оптимальности выбранного маршрута.

Разработанный ГПДМ на каждом технологическом этапе решает следующие задачи.

1. Обеспечение надежности передачи рабочего и служебного трафиков, содержащих информацию о маршруте.

2. Обнаружение подключенных к текущему маршрутизатору R_i каналов (K_{ij}), сетей S_i и непосредственно подключенных соседних маршрутизаторов (R_k), присвоение веса подключенным каналам связи, построение таблиц соседей для поддержания отношений смежности с непосредственно подключенными соседними маршрутизаторами.

3. Расчет таблицы маршрутизации, включающий две подзадачи:

3а) – вычисление наилучшего пути для текущего маршрутизатора R_j к соседним и определение следующих маршрутизаторов, обеспечивающих создание кратчайшего пути к текущему маршрутизатору (решается с использованием модифицированного алгоритма распределенных последовательных номеров DSN (Distributed Sequence Number) протокола RIP);

3б) – пересылка маршрутизатором сети, выполнившим вычисления, параметров вектора стоимости расстояний между своими соседями для определения своего участка наилучшего пути (решается с использованием предложенного алгоритма состояния канала, распределяющего информацию о состоянии каналов между сетями и маршрутизаторами инфраструктуры, используя результаты решения подзадачи 3а).

В статье предлагается решение задачи 3, используя результаты решения задач 1 и 2 как заданные.

Построение алгебраической модели для расчета кратчайших маршрутов

При маршрутизации требуется определить стоимость пути до узла назначения, а при пересылке – стоимость пути до некоторой сети на основании ее описания в удаленных маршрутизаторах. В работах [12, 13] приведены доказательства теорем, которые обоснуют использование полуколец для решения задачи о кратчайшем пути без циклов.

Алгебраическая модель для расчета кратчайших маршрутов формулируется следующим образом.

Пусть V – множество узлов инфраструктуры $n = |V|$, а R – множество узлов пересылки (сетей) $d = |R|$, где $V \cap R = \emptyset$. Матрица смежности A размерностью $n \times n$ использует параметры полукольца $(S, \oplus, \otimes, \bar{0}, \bar{1})$, S – непустое и нетривиальное множество, а аксиомы операторов, указанных в таблице, справедливы и представляют собой веса связей на графе $G = (V, E)$, в то время как матрица M размерностью $n \times d$, называемая матрицей отображения, моделирует, как узлы в R прикреплены (или подключены) к узлам инфраструктуры в V .

Для простейшего случая предполагаем, что матрица M также принимает свои значения в S . Если $\bar{1}$ – аннигилятор для \oplus (это подразумевает, что \oplus идемпотентна $a \oplus a = (a \otimes \bar{1}) \oplus \oplus (a \otimes \bar{1}) = a \otimes (\bar{1} \oplus \bar{1}) = a \otimes \bar{1} = a$), то A^* существует и решает уравнение $L = (A \otimes L) \oplus I$.

Тогда $A^* \otimes M$ решает уравнение $F = (A \otimes F) \oplus \oplus M$.

Аксиомы полукольца

Semiring axioms

Аксиома	Пояснение
\oplus Ассоциативность	$a \oplus (b \oplus c) = (a \oplus b) \oplus c$
\oplus Коммутативность	$a \oplus b = b \oplus a$
\oplus Нейтральный элемент	$a \oplus \bar{0} = \bar{0} \oplus a = a$
\otimes Ассоциативность	$a \otimes (b \otimes c) = (a \otimes b) \otimes c$
\otimes Нейтральный элемент	$a \otimes \bar{1} = \bar{1} \otimes a = a$
\otimes Поглощающий элемент	$a \otimes \bar{0} = \bar{0} \otimes a = \bar{0}$
Дистрибутивность слева	$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$
Дистрибутивность справа	$(a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c)$

Рассматриваемая задача нахождения матрицы пересылки F делится на две подзадачи: поиск кратчайшего маршрута к узлам инфраструктуры V и распределение матрицы отображения M по узлам инфраструктуры:

$$M = \begin{matrix} & \begin{matrix} d_1 & d_2 \end{matrix} \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} \infty & \infty \\ 3 & \infty \\ \infty & \infty \\ \infty & 1 \\ 2 & 3 \end{bmatrix} \end{matrix}, \quad F = \begin{matrix} & \begin{matrix} d_1 & d_2 \end{matrix} \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 5 & 6 \\ 3 & 7 \\ 5 & 5 \\ 9 & 1 \\ 2 & 3 \end{bmatrix} \end{matrix}.$$

Первая задача может быть решена с помощью алгоритма дистанционно-векторной маршрутизации с использованием порядковых номеров, а вторая с помощью алгоритма состояния каналов. Пример топологии, заданной для решения задачи 3а, представлен на рисунке 2.

Другая алгебраическая конструкция, называемая полумодулями, используется для моделирования порядкового номера. В более общем случае M принимает свои значения в другой алгебраической структуре $(N, \boxplus, \triangleright)$, где $\triangleright \in S \times N \rightarrow N$. Операция \boxplus между матрицами определяется как $A \boxplus B = C$, где A, B и C одинаковой размерности и $C(i, j) = A(i, j) \boxplus B(i, j)$. Более того, определяется \triangleright между матрицами как функция $A \triangleright B = C$. Если A имеет размерность $n \times m$ и B размерность $m \times d$, то C имеет размерность $n \times d$:

$$C(i, j) = \sum_{1 \leq k \leq m} A(i, k) \boxplus B(k, j).$$

Пусть $(S, \oplus, \otimes, \bar{0}, \bar{1})$ является полукольцом, структура $(N, \boxplus, \triangleright, \bar{0}_N)$ называется полумодулем над S , если выполняются следующие условия:

$(N, \boxplus, \bar{0}_N)$ – коммутативная полугруппа,

$\triangleright \in (S \times N) \rightarrow N$,

$(a \otimes b) \triangleright c = a \triangleright (b \triangleright c)$,

$\bar{0} \triangleright m = \bar{0}_N$,

$s \triangleright \bar{0} = \bar{0}_N$,

$\bar{1} \triangleright m = m$,

LD: $s \triangleright (m \boxplus n) = (s \triangleright m) \boxplus (s \triangleright n)$,

RD: $(s \oplus t) \triangleright m = (s \triangleright m) \boxplus (t \triangleright m)$.

Можно проверить, верны ли следующие утверждения о матрицах (размерности I, A, B равны $n \times n$, а C, M равны $n \times d$):

- 1) $I \triangleright C = C$,
- 2) $(A \otimes B) \triangleright C = A \triangleright (B \triangleright C)$,
- 3) $A \triangleright (C \boxplus M) = (A \triangleright C) \boxplus (A \triangleright M)$,
- 4) $(A \oplus B) \triangleright C = (A \triangleright C) \boxplus (B \triangleright C)$.

Пусть $(N, \boxplus, \triangleright, \bar{0}_N)$ является полумодулем полукольца $(S, \oplus, \otimes, \bar{0}, \bar{1})$, а $\bar{1}$ аннигилятором для \otimes . Если A – матрица смежности, принимающая свои значения в S , а F – матрица отображения, принимающая свои значения в N , то $A^* \triangleright M$ является решением уравнения $F = (A \triangleright F) \boxplus M$.

Для решения задачи пересылки необходимо решить уравнение $F = (A \triangleright F) \boxplus M$, которое

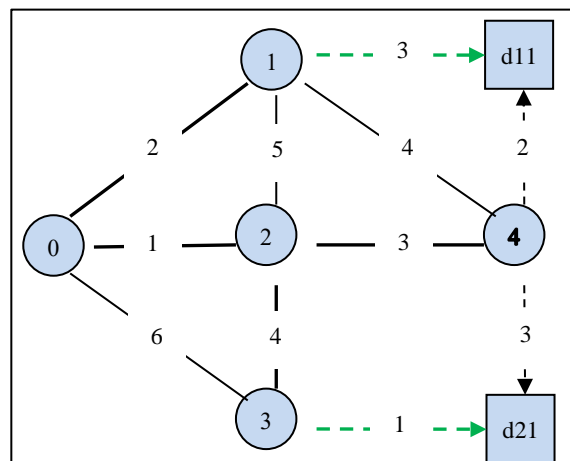


Рис. 2. Пример простой маршрутизации и пересылки

Fig. 2. A simple routing and forwarding example

также будет использоваться для решения задачи маршрутизации.

В предложенном алгоритме каждый узел выбирает порядковый номер. Тогда стоимость пути от узла i к самому себе имеет вид (s_i, \bar{l}_s) , например, если $(S, \oplus, \otimes) = (R_+, \min, +)$, то $\bar{0}_s = \infty$ и $\bar{l}_s = 0$. Определим коммутативные полугруппы $(N, \max, 0)$. Пусть \boxplus определяется следующим образом (лексикографическое произведение $\max \times \min$):

$$(s_1, d_1) \boxplus (s_2, d_2) = \begin{cases} (s_1, d_1) & s_1 > s_2, \\ (s_2, d_2) & s_1 < s_2, \\ (s, \min(d_1, d_2)) & s = s_1 = s_2. \end{cases}$$

Функция $\triangleright_{fst} \in (S \times (N \times S)) \rightarrow (N \times S)$ определяется как $d_1 \triangleright_{fst} (s, d_2) \triangleq (s, d_1 \otimes d_2)$, а отключенные узлы не сообщают никакой информации: $\bar{0}_s \triangleright_{fst} (s, d) \triangleq (0, \bar{0}_s)$.

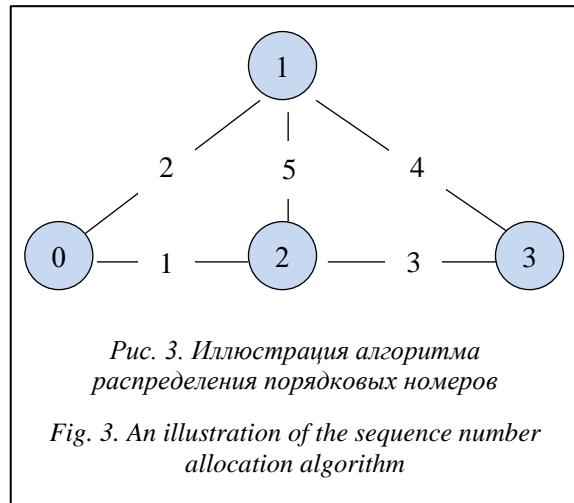
Отношение \triangleright определяется следующим образом: $L \triangleright M(i, j) = \sum_{q \in V} L(i, q) \triangleright_{fst} M(q, j)$.

Затем определяется матрица M , которая имеет размерность $n \times n$: $M(i, i) = (s_i, \bar{l}_s)$ и $M(i, j) = (0, \bar{0}_s)$, где $i \neq j$. Далее алгоритм решает уравнение $F = (A \triangleright F) \boxplus M$, где M и F имеют одинаковую размерность, а расчет стоимости производится для узлов инфраструктуры. Идея состоит в том, что при изменении матрицы смежности A у некоторых узлов инфраструктуры не будут найдены маршрутизаторы для некоторых пунктов назначения. Если увеличить порядковый номер для таких узлов назначения, это будет равносильно сбросу для маршрутизации до данных пунктов назначения. Рассмотрим пример на рисунке 3 и сформируем матрицы A, A^*, M, F :

$$A = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} \infty & 2 & 1 & \infty \\ 2 & \infty & 5 & 4 \\ 1 & 5 & \infty & 3 \\ \infty & 4 & 3 & \infty \end{bmatrix} & , & A^* = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} 0 & 2 & 1 & 4 \\ 2 & 0 & 3 & 4 \\ 1 & 3 & 0 & 3 \\ 4 & 4 & 3 & 0 \end{bmatrix} \end{matrix}$$

$$M = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (s_0, 0) & (0, \infty) & (0, \infty) & (0, \infty) \\ (0, \infty) & (s_1, 0) & (0, \infty) & (0, \infty) \\ (0, \infty) & (0, \infty) & (s_2, 0) & (0, \infty) \\ (0, \infty) & (0, \infty) & (0, \infty) & (s_3, 0) \end{bmatrix} \end{matrix}$$

$$F = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (s_0, 0) & (s_1, 2) & (s_2, 1) & (s_3, 4) \\ (s_0, 2) & (s_1, 0) & (s_2, 3) & (s_3, 4) \\ (s_0, 1) & (s_1, 3) & (s_2, 0) & (s_3, 3) \\ (s_0, 4) & (s_1, 4) & (s_2, 3) & (s_3, 0) \end{bmatrix} \end{matrix}$$



Рассмотрим ситуацию, в которой связь между 0 и 2 нарушена. В соответствии с алгоритмом увеличиваем порядковый номер в матрице M . Новая матрица B смежности имеет следующий вид:

$$B = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} \infty & 2 & \infty & \infty \\ 2 & \infty & 5 & 4 \\ \infty & 5 & \infty & 3 \\ \infty & 4 & 3 & \infty \end{bmatrix} \end{matrix}$$

$$M' = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (s'_0, 0) & (0, \infty) & (0, \infty) & (0, \infty) \\ (0, \infty) & (s'_1, 0) & (0, \infty) & (0, \infty) \\ (0, \infty) & (0, \infty) & (s'_2, 0) & (0, \infty) \\ (0, \infty) & (0, \infty) & (0, \infty) & (s'_3, 0) \end{bmatrix} \end{matrix}$$

где $s'_i = s_i + 1$,

$$F_0 = F \boxplus M' = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (s'_0, 0) & (s_1, 2) & (s_2, 1) & (s_3, 4) \\ (s_0, 2) & (s'_1, 0) & (s_2, 3) & (s_3, 4) \\ (s_0, 1) & (s_1, 3) & (s'_2, 0) & (s_3, 3) \\ (s_0, 4) & (s_1, 4) & (s_2, 3) & (s'_3, 0) \end{bmatrix} \end{matrix}$$

$$F_1 = (B \triangleright F_0) \boxplus M' = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (s'_0, 0) & (s'_1, 2) & (s_2, 5) & (s_3, 6) \\ (s'_0, 2) & (s'_1, 0) & (s'_2, 5) & (s'_3, 4) \\ (s_0, 7) & (s'_1, 5) & (s'_2, 0) & (s'_3, 3) \\ (s_0, 4) & (s'_1, 4) & (s'_2, 3) & (s'_3, 0) \end{bmatrix} \end{matrix}$$

$$F_2 = (B \triangleright F_1) \boxplus M' = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (s'_0, 0) & (s'_1, 2) & (s'_2, 7) & (s'_3, 6) \\ (s'_0, 2) & (s'_1, 0) & (s'_2, 5) & (s'_3, 4) \\ (s'_0, 7) & (s'_1, 5) & (s'_2, 0) & (s'_3, 3) \\ (s'_0, 6) & (s'_1, 4) & (s'_2, 3) & (s'_3, 0) \end{bmatrix} \end{matrix}$$

$$F_2 = B^* \triangleright M'$$

Переходные циклы могут образовываться, когда узлы используют маршруты со старыми порядковыми номерами. Чтобы разрешить данную ситуацию, добавлена проверка условия выполнимости и введен дополнительный бит запроса. Пусть $B = \{true, false\}$, а операция ∇ над полугруппой $B \times \mathbb{N}$ определяется следующим образом:

$$(b_1, s_1) \nabla (b_2, s_2) = (b_1 \vee b_2, \max(s_1, s_2)).$$

Ясно, что ∇ ассоциативно, коммутативно и $(false, 0)$ является нейтральным элементом. Далее операция \boxplus определяется соотношением

$$(b_1, s_1, d_1) \boxplus (b_2, s_2, d_2) = \begin{cases} (b_1, s_1, d_1) & s_1 > s_2, \\ (b_2, s_2, d_2) & s_1 < s_2, \\ (b_1 \vee b_2, s, \min(d_1, d_2)) & s = s_1 = s_2, \end{cases}$$

$$d_1 \triangleright (b, s, d_2) \triangleq (b, s, d_1 \otimes d_2),$$

$$\bar{0}_s \triangleright (b, s, d) \triangleq (false, 0, \bar{0}_s),$$

$$L \triangleright M(i, j) = \boxplus_{q \in V} L(i, q) \triangleright_{\text{fst}} M(q, j).$$

Теперь необходимо вычислить $F_{k+1} = (A_{k+1} \triangleright F_k) \boxplus M_k$. Введем матрицу D , которая содержит состояние выполнимости и вычисляется с использованием уравнения $D_{k+1} = F_{k+1} \boxplus D_k$.

Используем условие для проверки выполнимости построения кратчайшего бесциклового маршрута [13]. Узел s является возможным маршрутизатором для узла i при маршрутизации в направлении j , если $F_k(s, j) <_{\boxplus} D(i, j)$, где $F_k(s, j)$ – объявленное расстояние узла s до его непосредственно подключенного соседа i , а j – узел назначения. Если нет выполнимого маршрутизатора, запрос на увеличение порядкового номера устанавливается в $F_{k+1}(i, j)$. Если запрос на увеличение порядкового номера будет установлен для элемента в диагонали $F_{k+1} = (A_{k+1} \triangleright F_k) \boxplus M_k$, порядковый номер элемента в матрице M_k увеличивается, поэтому порядковый номер также будет увеличен в F_{k+1} , а бит останется неустановленным для диагонального элемента. Проиллюстрируем работу алгоритма, используя представленные на рисунке 3 значения:

$$D_0 = F_0 = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (f, s_0, 0) & (f, s_1, 2) & (f, s_2, 1) & (f, s_3, 4) \\ (f, s_0, 2) & (f, s_1, 0) & (f, s_2, 3) & (f, s_3, 4) \\ (f, s_0, 1) & (f, s_1, 3) & (f, s_2, 0) & (f, s_3, 3) \\ (f, s_0, 4) & (f, s_1, 4) & (f, s_2, 3) & (f, s_3, 0) \end{bmatrix} \end{matrix}$$

$$B = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} \infty & 2 & \infty & \infty \\ 2 & \infty & 5 & 4 \\ \infty & 5 & \infty & 3 \\ \infty & 4 & 3 & \infty \end{bmatrix} \end{matrix}$$

$$M_0 = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (f, s_0, 0) & (f, 0, \infty) & (f, 0, \infty) & (f, 0, \infty) \\ (f, 0, \infty) & (f, s_1, 0) & (f, 0, \infty) & (f, 0, \infty) \\ (f, 0, \infty) & (f, 0, \infty) & (f, s_2, 0) & (f, 0, \infty) \\ (f, 0, \infty) & (f, 0, \infty) & (f, 0, \infty) & (f, s_3, 0) \end{bmatrix} \end{matrix}$$

$$F_1 = (B \triangleright F_0) \boxplus M_0 =$$

$$\begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (f, s_0, 0) & (f, s_1, 2) & (t, s_2, \infty) & (t, s_3, \infty) \\ (f, s_0, 2) & (f, s_1, 0) & (f, s_2, 3) & (f, s_3, 4) \\ (t, s_0, \infty) & (f, s_1, 5) & (f, s_2, 0) & (f, s_3, 3) \\ (f, s_0, 4) & (f, s_1, 4) & (f, s_2, 3) & (f, s_3, 0) \end{bmatrix} \end{matrix}$$

$$M_1 = M_0, D_1 = F_1 \boxplus D_0 =$$

$$\begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (f, s_0, 0) & (f, s_1, 2) & (t, s_2, 1) & (t, s_3, 4) \\ (f, s_0, 2) & (f, s_1, 0) & (f, s_2, 3) & (f, s_3, 4) \\ (t, s_0, 1) & (f, s_1, 3) & (f, s_2, 0) & (f, s_3, 3) \\ (f, s_0, 4) & (f, s_1, 4) & (f, s_2, 3) & (f, s_3, 0) \end{bmatrix} \end{matrix}$$

$$F_2 = (B \triangleright F_1) \boxplus M_1 =$$

$$\begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (f, s_0, 0) & (f, s_1, 2) & (t, s_2, \infty) & (t, s_3, \infty) \\ (t, s_0, 2) & (f, s_1, 0) & (t, s_2, 5) & (t, s_3, 4) \\ (t, s_0, \infty) & (f, s_1, 5) & (f, s_2, 0) & (f, s_3, 3) \\ (t, s_0, 6) & (f, s_1, 4) & (f, s_2, 3) & (f, s_3, 0) \end{bmatrix} \end{matrix}$$

$$D_2 = F_2 \boxplus D_1 =$$

$$\begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (f, s_0, 0) & (f, s_1, 2) & (t, s_2, 1) & (t, s_3, 4) \\ (t, s_0, 2) & (f, s_1, 0) & (t, s_2, 3) & (t, s_3, 4) \\ (t, s_0, 1) & (f, s_1, 3) & (f, s_2, 0) & (f, s_3, 3) \\ (t, s_0, 4) & (f, s_1, 4) & (f, s_2, 3) & (f, s_3, 0) \end{bmatrix} \end{matrix}$$

$$M_2 = \begin{matrix} & 0 & 1 & 2 & 3 \\ \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} (f, s'_0, 0) & (f, 0, \infty) & (f, 0, \infty) & (f, 0, \infty) \\ (f, 0, \infty) & (f, s_1, 0) & (f, 0, \infty) & (f, 0, \infty) \\ (f, 0, \infty) & (f, 0, \infty) & (f, s'_2, 0) & (f, 0, \infty) \\ (f, 0, \infty) & (f, 0, \infty) & (f, 0, \infty) & (f, s'_3, 0) \end{bmatrix} \end{matrix}$$

$$F_3 = (B \triangleright F_2) \boxplus M_2 =$$

	0	1	2	3
0	$(f, s'_0, 0)$	$(f, s_1, 2)$	(t, s_2, ∞)	(t, s_3, ∞)
1	$(t, s_0, 2)$	$(f, s_1, 0)$	$(t, s_2, 5)$	$(t, s_3, 4)$
2	(t, s_0, ∞)	$(f, s_1, 5)$	$(f, s'_2, 0)$	$(t, s_3, 3)$
3	$(t, s_0, 6)$	$(f, s_1, 4)$	$(t, s_2, 3)$	$(f, s'_3, 0)$

$$M_3 = M_2, D_3 = F_3 \boxplus D_2 =$$

	0	1	2	3
0	$(f, s'_0, 0)$	$(f, s_1, 2)$	$(t, s_2, 1)$	$(t, s_3, 4)$
1	$(t, s_0, 2)$	$(f, s_1, 0)$	$(t, s_2, 3)$	$(t, s_3, 4)$
2	$(t, s_0, 1)$	$(f, s_1, 3)$	$(f, s'_2, 0)$	$(t, s_3, 3)$
3	$(t, s_0, 4)$	$(f, s_1, 4)$	$(t, s_2, 3)$	$(f, s'_3, 0)$

$$F_4 = (B \triangleright F_3) \boxplus M_3 =$$

	0	1	2	3
0	$(f, s'_0, 0)$	$(f, s_1, 2)$	(t, s_2, ∞)	(t, s_3, ∞)
1	$(f, s'_0, 2)$	$(f, s_1, 0)$	$(f, s'_2, 5)$	$(f, s'_3, 4)$
2	(t, s_0, ∞)	$(f, s_1, 5)$	$(f, s'_2, 0)$	$(f, s'_3, 3)$
3	$(t, s_0, 6)$	$(f, s_1, 4)$	$(f, s'_2, 3)$	$(f, s'_3, 0)$

$$M_4 = M_3, D_4 = F_4 \boxplus D_3 =$$

	0	1	2	3
0	$(f, s'_0, 0)$	$(f, s_1, 2)$	$(t, s_2, 1)$	$(t, s_3, 4)$
1	$(f, s'_0, 2)$	$(f, s_1, 0)$	$(f, s'_2, 3)$	$(f, s'_3, 4)$
2	$(t, s_0, 1)$	$(f, s_1, 3)$	$(f, s'_2, 0)$	$(f, s'_3, 3)$
3	$(t, s_0, 4)$	$(f, s_1, 4)$	$(f, s'_2, 3)$	$(f, s'_3, 0)$

$$F_5 = (B \triangleright F_4) \boxplus M_4 =$$

	0	1	2	3
0	$(f, s'_0, 0)$	$(f, s_1, 2)$	$(f, s'_2, 7)$	$(f, s'_3, 6)$
1	$(f, s'_0, 2)$	$(f, s_1, 0)$	$(f, s'_2, 5)$	$(f, s'_3, 4)$
2	$(f, s'_0, 7)$	$(f, s_1, 5)$	$(f, s'_2, 0)$	$(f, s'_3, 3)$
3	$(f, s'_0, 6)$	$(f, s_1, 4)$	$(f, s'_2, 3)$	$(f, s'_3, 0)$

$$F_5 = B^* \triangleright M_4.$$

Приведенные вычисления выполняются распределенно с помощью дистанционно-векторного алгоритма. Затем решение в форме таблицы маршрутизации (B^*) извлекается (удаляются порядковый номер и бит запроса) из приведенного выше расчета и используется для решения задач пересылки.

Концепция моделирования ГПДМ

Проверка работоспособности и оценка масштабируемости предложенного ГПДМ проведена методами математического моделирования.

В модели узлы графа – это маршрутизаторы, а ребра – связи между ними. Создан также генератор случайных графов для формирования топологий сети. Узлы обмениваются между собой сообщениями с обновлениями маршрутизации по ребрам графа. На рисунке 4 показана диаграмма классов различных служебных сообщений, которыми обмениваются узлы в системе. На рисунке 5 представлена диаграмма классов логики маршрутизации.

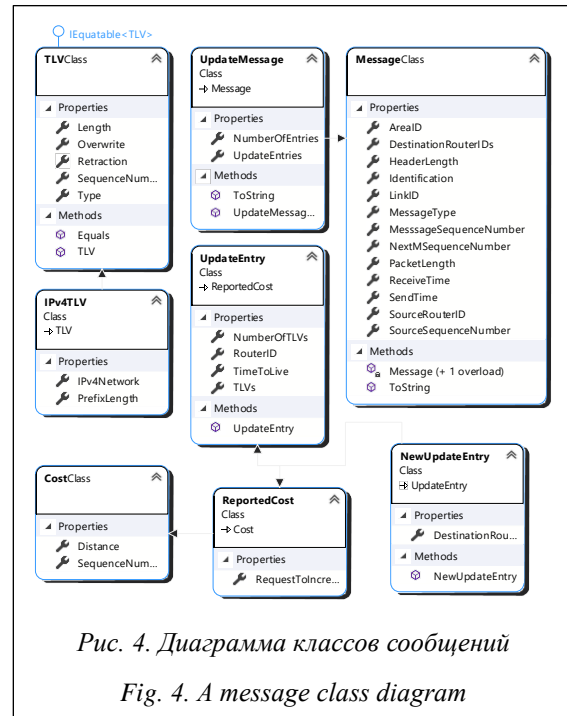


Рис. 4. Диаграмма классов сообщений

Fig. 4. A message class diagram

Каждое сообщение состоит из общего заголовка и основного текста. Общий заголовок включает следующие поля: длина пакета (общая длина сообщения), идентификатор сообщения, порядковый номер сообщения, идентификатор исходного маршрутизатора, порядковый номер исходного узла, идентификатор области, тип сообщения, длина заголовка, список идентификаторов узлов назначения. Следующий порядковый номер сообщения (NextMSequenceNumber) – это общий счетчик, используемый планировщиком управляемого событиями симулятора для присвоения порядкового номера сообщениям.

Сообщения об обновлениях содержат список записей обновлений. Каждая запись обновления состоит из идентификатора и порядкового номера владельца записи, а также включает расстояние от узла, который был источником сообщения об обновлении, до владельца. Флаг RISN необходимо установить при



Рис. 5. Диаграмма классов логики маршрутизации

Fig. 5. A routing logic class diagram

для поддержания информации об изученных маршрутах, таблицу новых записей обновлений, которые необходимо отправить соседним узлам.

Для каждого соседа в таблице соседей имеются таблицы каналов, по которым происходит обмен сообщениями, и таблицы сообщенных расстояний от них. В каждой записи узла сохраняются стоимость и выполняемая стоимость маршрута. Кроме того, в модели настроено поддержание таблицы для каждого маршрута с целью хранения выполнимых узлов, следующих переходов и сетей (записи информации о состоянии канала).

наличия запроса к владельцу на увеличение порядкового номера.

Каждая запись обновления также содержит список TLV (тип–длина–значение), а список – узлы, подключенные к владельцу маршрута и распределяемые с помощью алгоритма состояния канала. Например, адреса и префиксы IPv4 распространяются с использованием IPv4TLV. Флаг отзыва (Retracted-flag) должен быть установлен, когда узел-владелец потерял соединение с сетью и хочет удалить запись соединения из БД. Каждый TLV сопровождается отдельным порядковым номером, который создается и поддерживается узлом-владельцем. Когда в TLV происходит изменение, порядковый номер увеличивается. Отзыванные TLV не следует немедленно удалять из БД. Они будут отмечены флагом отзыва и могут быть удалены после истечения настраиваемого значения времени ожидания. Это делается во избежание путаницы, возникающей при получении узлом устаревшего объявления. В этом случае объявление будет проигнорировано.

Сообщения, которыми обмениваются узлы, хранятся в БД и в файле журнала для сбора статистики о производительности протокола.

Каждый узел поддерживает следующие таблицы состояний: таблицу подключенных каналов для хранения информации о непосредственно подключенных каналах, таблицу соседей для хранения информации о ближайших соседях узла, таблицу изученных записей узла

следующих переходов и сетей (записи информации о состоянии канала).

Результаты моделирования ГПДМ

Представлены результаты моделирования ГПДМ при обмене сообщениями обновления в случайно сгенерированных топологиях с различным количеством маршрутизаторов и каналов. В сценарии моделирования один из каналов «разъединяется», когда время моделирования равно $T/3$ (где T – время продолжительности моделирования), и восстанавливает соединение, когда время моделирования будет равно $2T/3$. На приведенных графиках (рис. 6–8) пред-

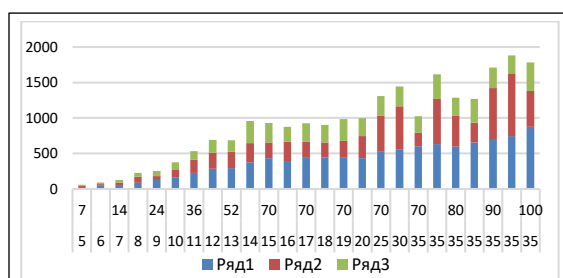


Рис. 6. Зависимость количества служебных сообщений об обновлении от количества каналов и маршрутизаторов

Fig. 6. The dependence of the number of update service messages on the number of channels and routers

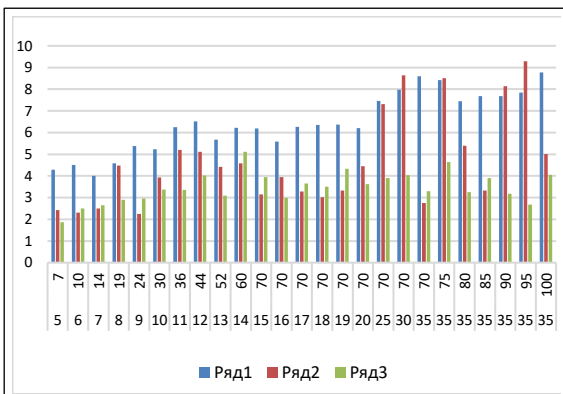


Рис. 7. Зависимость среднего количества служебных сообщений об обновлении, приходящихся на один канал, от количества каналов и маршрутизаторов

Fig. 7. The dependence of the average number of update service messages per channel on the number of channels and routers

ставлена статистика, собранная в результате тестовых запусков. По горизонтальным осям указывается количество каналов и маршрутизаторов в каждом эксперименте. Различные результаты моделирования представлены одно-типно в трех рядах: ряд 1 – это измеренное значение в первой трети до отключения канала, ряд 2 – во второй трети после отключения канала и ряд 3 – в последней трети после восстановления соединения.

В результате анализа выявленных зависимостей установлено следующее:

- модифицированный алгоритм распределенных последовательных номеров DSN функционирует устойчиво;

- разработанный алгоритм расчета кратчайшего маршрута в условиях динамического изменения топологий выполняет вычисления в полном объеме;

- полученные оценки масштабирования по параметру роста служебного трафика подтвердили, что разработанный ГПДМ обладает высокими показателями масштабируемости и позволяет после доработки пакета программ, реализующих ГПДМ, использовать его для крупных ЦОД.

Оценки зависимостей параметров служебного трафика от количества каналов и количества маршрутизаторов подтверждают критерий Q масштабируемости протокола, поскольку максимальное количество $\max Q$ сообщений об обновлении, необходимых для достижения конвергенции, составляет $\max Q = cE \log_2 M$, где M – количество маршрутизаторов, E – количество каналов, c – класс сети [3].

Заключение

Особенностью разработанного ГПДМ является отделение этапа пересылки от этапа маршрутизации. При расчете маршрутизации в ГПДМ вычисляется стоимость пути к узлу назначения, а при пересылке – стоимость пути до некоторой сети, подключенной к удаленному маршрутизатору. Поэтому в ГПДМ последовательно решаются две задачи: вычисление пути к узлам инфраструктуры, реализуемое модифицированным алгоритмом распределенных последовательных номеров DSN, и распределение информации о состоянии каналов между сетями и маршрутизаторами инфраструктуры, достигаемое с помощью алгоритма состояния канала.

ГПДМ имеет гораздо меньшую БД, чем известные протоколы состояния канала. Прежде всего это связано с тем, что объявляются только префиксы подключенных сетей, а не сетевые каналы, к тому же сетевые префиксы могут быть дополнительно обобщены с помощью суммирования маршрутов.

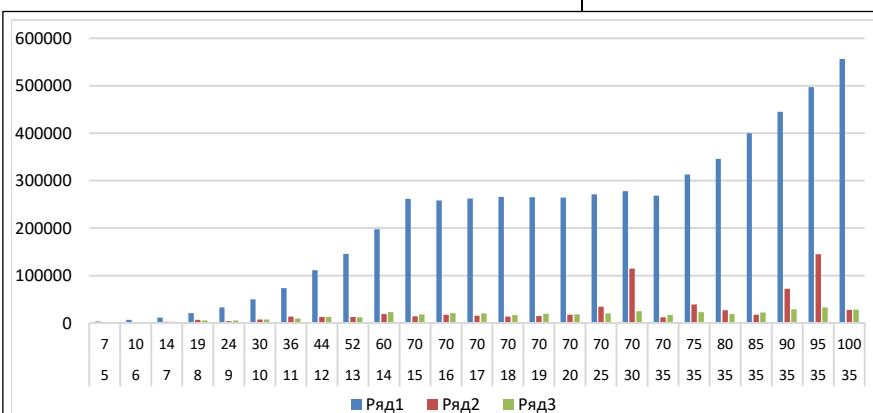


Рис. 8. Зависимость суммарного размера служебных сообщений (в байтах) от количества каналов и маршрутизаторов

Fig. 8. The dependence of the total size of service messages (in bytes) on the number of channels and routers

Кроме того, ГПДМ использует вектор расстояния только для узлов инфраструктуры, что ограничивает количество обновлений после изменения топологии. Следовательно, в ГПДМ параметры сети объявляются только один раз, а не при каждом изменении расстояния.

Реализация алгоритма имитационной модели ГПДМ показала, что размер обмениваемой между узлами информации для конвергенции маршрута после изменения топологии является предельным по сравнению с тем, который был на начальной фазе расчета маршрута.

Литература

1. Dutt D.G. BGP in the Data Center. O'Reilly Media Publ., 2017, 88 p.
2. Medhi D., Ramasamy K. Network Routing: Algorithms, Protocols, and Architectures. San Francisco, CA, Morgan Kaufmann Publ., 2007, 957 p.
3. Lapukhov P., Premji A. Use of BGP for routing in large-scale data centers. RFC, 2016, no. 7938, 35 p. DOI: 10.17487/RFC7938.
4. Abhashkumar A., Subramanian K., Andreyev A., et al. Running BGP in data centers at scale. Proc. XVIII USENIX Symposium on NSDI, 2021, pp. 65–81.
5. Garcia-Luna-Aceves J.J. Eliminating routing loops and oscillations in BGP using total ordering. Proc. XLVII IEEE Conf. LCN, 2022, pp. 9–17.
6. Fontugne R., Bautista E., Petrie C., Nomura Y., Abry P. et al. BGP zombies: An analysis of beacons stuck routes. Proc. Int. Conf. PAM, 2019, pp. 197–209. DOI: 10.1007/978-3-030-15986-3_13.
7. Patel K., Lindem A., Zandi S., Henderickx W. BGP link-state shortest path first (SPF) routing. Data-tracker. URL: <https://datatracker.ietf.org/doc/draft-ietf-lsvr-bgp-spf/> (дата обращения: 02.07.2022).
8. Savage D., Ng J., Moore S., Slice D., Paluch P., White R. Cisco's enhanced interior gateway routing protocol (EIGRP). RFC, 2016, no. 7868, 80 p. DOI: 10.17487/RFC7868.
9. Fuzi M.F.M., Abdullah K., Halim I.H.A., Ruslan R. Network automation using ansible for EIGRP network. JCRINN, 2021, vol. 6, no. 4, pp. 59–69. DOI: 10.24191/jcrinn.v6i4.237.
10. Garcia-Luna-Aceves J.J. Safe, fast, and cycle-free multi-path routing using vouchers. Proc. XLVII IEEE Conf. LCN, 2022, pp. 367–370. DOI: 10.1109/LCN53696.2022.9843624.
11. Chroboczek J., Schinazi D. The babel routing protocol. RFC, 2021, no. 8966, 54 p. DOI: 10.17487/rfc8966.
12. Khayou H., Rudenkova M.A., Abrosimov L.I. On the algebraic theory of loop free routing. Proc. Int. Conf. DCCN, 2020, pp. 161–175. DOI: 10.1007/978-3-030-66471-8_14.
13. Хаю Х., Орлова М.А., Абросимов Л.И. Алгебраическая методология моделирования бесцикло-вой маршрутизации // Вестн. ВГТУ. 2022. Т. 18. № 1. С. 47–61. DOI: 10.36622/VSTU.2022.18.1.006.
14. Khayou H., Orlova M.A., Abrosimov L.I. A Hybrid distance vector link state algorithm: distributed sequence number. IJCN, 2021, vol. 8, no. 3, pp. 203–213. DOI: 10.22247/ijcna/2021/209188.

Software & Systems
DOI: 10.15827/0236-235X.141.071-082

Received 01.06.22, Revised 13.09.22
2023, vol. 36, no. 1, pp. 071–082

Developing and modeling a hybrid dynamic routing protocol

L.I. Abrosimov¹, Dr.Sc. (Engineering), Professor, AbrosimovLI@mpei.ru

Kh. Khayou¹, Postgraduate Student, hussein.khayou@gmail.com

M.A. Orlova¹, Ph.D. (Engineering), Assistant, OrlovaMA@mpei.ru

¹National Research University "Moscow Power Engineering Institute",
Moscow, 111250, Russian Federation

Abstract. Nowadays, the growth of network services significantly increases the requirements for the quality and speed of solving network management problems in ever-growing data centers. The load increase in data centers leads to the need for structural scaling that implies increasing the number of servers and routers. There is a need for simple scalable routing protocols to facilitate automation and management of ever-growing networks, especially in data centers.

The work aims to present and simulate a new hybrid dynamic routing protocol including an upgraded distance vector routing algorithm and a link state algorithm.

The article discusses the solution to the problems of developing a hybrid dynamic routing protocol, which guarantees loop freedom and provides scaling requirements through the development and implementation of simple algorithms that ensure reliable transmission of data and service traffic containing route information and detects channels, networks, and directly connected neighboring routers connected to the current router. The scalability requirements of the new hybrid dynamic routing protocol are met since the distance vector routing algorithm calculates the distances to infrastructure nodes, and not to the network prefixes themselves. The link state algorithm advertises network prefixes only once, which leads to a reduction in the link state database and a reduction in calculations after topology changes. Loop freedom is achieved by introducing a newly developed distributed sequence number algorithm.

A simulation model has been developed to simulate the hybrid dynamic routing protocol. The simulation allowed estimating the amount and volume of service traffic, which confirmed the effectiveness of the developed protocol in the conditions of scaling the data center network.

Keywords: routing protocol scalability, loopless routing, algebraic routing protocol model, link state algorithm, distributed sequence number algorithm, routing protocol modeling, dynamic routing hybrid protocol.

References

1. Dutt D.G. *BGP in the Data Center*. O'Reilly Media Publ., 2017, 88 p.
2. Medhi D., Ramasamy K. *Network Routing: Algorithms, Protocols, and Architectures*. San Francisco, CA, Morgan Kaufmann Publ., 2007, 957 p.
3. Lapukhov P., Premji A. Use of BGP for routing in large-scale data centers. *RFC*, 2016, no. 7938, 35 p. DOI: 10.17487/RFC7938.
4. Abhashkumar A., Subramanian K., Andreyev A., et al. Running BGP in data centers at scale. *Proc. XVIII USENIX Symposium on NSDI*, 2021, pp. 65–81.
5. Garcia-Luna-Aceves J.J. Eliminating routing loops and oscillations in BGP using total ordering. *Proc. XLVII IEEE Conf. LCN*, 2022, pp. 9–17.
6. Fontugne R., Bautista E., Petrie C., Nomura Y., Abry P. et al. BGP zombies: An analysis of beacons stuck routes. *Proc. Int. Conf. PAM*, 2019, pp. 197–209. DOI: 10.1007/978-3-030-15986-3_13.
7. Patel K., Lindem A., Zandi S., Henderickx W. BGP link-state shortest path first (SPF) routing. *Data-tracker*. Available at: <https://datatracker.ietf.org/doc/draft-ietf-lsvr-bgp-spf/> (accessed July 02, 2022).
8. Savage D., Ng J., Moore S., Slice D., Paluch P., White R. Cisco's enhanced interior gateway routing protocol (EIGRP). *RFC*, 2016, no. 7868, 80 p. DOI: 10.17487/RFC7868.
9. Fuzi M.F.M., Abdullah K., Halim I.H.A., Ruslan R. Network automation using ansible for EIGRP network. *JCRINN*, 2021, vol. 6, no. 4, pp. 59–69. DOI: 10.24191/jcrinn.v6i4.237.
10. Garcia-Luna-Aceves J.J. Safe, fast, and cycle-free multi-path routing using vouchers. *Proc. XLVII IEEE Conf. LCN*, 2022, pp. 367–370. DOI: 10.1109/LCN53696.2022.9843624.
11. Chroboczek J., Schinazi D. The babel routing protocol. *RFC*, 2021, no. 8966, 54 p. DOI: 10.17487/rfc8966.
12. Khayou H., Rudenkova M.A., Abrosimov L.I. On the algebraic theory of loop free routing. *Proc. Int. Conf. DCCN*, 2020, pp. 161–175. DOI: 10.1007/978-3-030-66471-8_14.
13. Khayu Kh., Orlova M.A., Abrosimov L.I. Algebraic methodology for modeling loop-free routing. *Vestn. VSTU*, 2022, vol. 18, no. 1, pp. 47–61. DOI: 10.36622/VSTU.2022.18.1.006 (in Russ.).
14. Khayou H., Orlova M.A., Abrosimov L.I. A Hybrid distance vector link state algorithm: distributed sequence number. *IJCNA*, 2021, vol. 8, no. 3, pp. 203–213. DOI: 10.22247/ijcna/2021/209188.

Для цитирования

Абросимов Л.И., Хаю Х., Орлова М.А. Разработка и моделирование гибридного протокола динамической маршрутизации // Программные продукты и системы. 2023. Т. 36. № 1. С. 071–082. DOI: 10.15827/0236-235X.141.071-082.

For citation

Abrosimov L.I., Khayu Kh., Orlova M.A. Developing and modeling a hybrid dynamic routing protocol. *Software & Systems*, 2023, vol. 36, no. 1, pp. 071–082 (in Russ.). DOI: 10.15827/0236-235X.141.071-082.