

УДК 004.052

В. С. МИХАЙЛЕНКО, А. М. АСЛАНОВ, М. С. СОЛОДОВНИК*Одесская национальная академия пищевых технологий, Украина***ИСПОЛЬЗОВАНИЕ АЛГОРИТМА ТАКАГИ-СУГЕНО В АДАПТИВНЫХ МЕТОДАХ МАРШРУТИЗАЦИИ КОМПЬЮТЕРНЫХ СЕТЕЙ**

Предложен новый подход для решения задачи поиска оптимального маршрута в корпоративной сети, действующий на основе теории нечеткой логики. Данный подход позволяет значительно сократить время нахождения оптимального маршрута от узла к узлу при сетевой маршрутизации, так как учитывает структуру сети и её параметры. Рассмотрен классический метод поиска кратчайших путей и инновационный нечеткий адаптивный метод, проведено сравнение этих двух методов и выявлены преимущества предложенного метода перед классическим. Используя пакет Fuzzy logic Toolbox среды MatLab, разработана программа, моделирующая процесс нахождения кратчайшего пути от узла к узлу в корпоративной сети на основе нечеткого подхода, построены соответствующие графы.

Ключевые слова: оптимальный маршрут, адаптивная маршрутизация, метод Такаги-Сугено, интеллектуальный маршрутизатор, фаззификация.

Введение

Необходимость обеспечения качественного обслуживания современного трафика, передаваемого через IP-сети, обуславливает высокие требования к эффективности передачи пакетов данных от отправителя к получателю. Важнейшим условием повышения конкурентоспособности предприятий в условиях рынка является внедрение новых современных информационных и сетевых технологий, различных методов ускорения маршрутизации, поддержки требуемого качества обслуживания, передачи голосового и видео трафика, повышения уровня безопасности сети и т.д. Особую важность имеет эффективная маршрутизация сообщений в условиях отказов отдельных элементов сети, всплесков трафика и локальных перегрузок. Загрузка и пропускная способность линий связи корпоративной сети динамически меняются, что, в свою очередь, может приводить к относительно частой рассылке служебной информации об изменении маршрутов. Изменения характеристик каналов связи, модификация структуры сети, включение в нее новых узлов и линий связи приводят к полному пересчету таблиц маршрутизации [1].

В зависимости от того, отдельно ли выбирают маршруты для каждого пакета или же только один раз для соединения, необходимо, чтобы алгоритм маршрутизации обладал рядом важных свойств – корректностью, простотой, надежностью, устойчивостью и оптимальностью. Во время работы корпоративной сети часто происходят отказы сетевого оборудования и соответствующие изменения топологии. Алгоритм маршрутизации должен уметь справляться с изменениями топологии и трафика без

необходимости прекращения задач на хостах и перезагрузки сети при каждом выходе из строя маршрутизатора. Ряд данных задачи успешно выполняют адаптивные динамические алгоритмы.

Эти алгоритмы изменяют решение о выборе маршрутов при изменении топологии и зависимости от загруженности линий. Адаптивные алгоритмы отличаются источниками получения информации (соседние маршрутизаторы), моментами изменения маршрутов (через определенные интервалы времени при изменении нагрузки или других параметров) и данными, используемыми для оптимизации (расстояние, количество транзитных участков, ожидаемое время пересылки т.д.) [2].

Проблемами совершенствования методов и алгоритмов маршрутизации в вычислительных сетях занимались такие ученые, как Л. Беллман, Е. Дейкстра, С. Флойд, Л. Форд, Д. Фулкерсон, А. Камоун, и т.д. [2 - 5]. В современных вычислительных сетях имеет место прямая зависимость производительности сети от производительности обрабатывающих межсетевой трафик маршрутизаторов. Из-за высокой загруженности маршрутизаторов получить оптимальный маршрут передачи данных довольно сложно. Большинство вычислительных сетей использует при выборе маршрута передачи критерий максимальной пропускной способности канала передачи данных и не используют алгоритмов маршрутизации, которые дополнительно рассчитывают информацию о возможных частичных изменениях структуры корпоративной сети. В то же время интеллектуальный маршрутизатор, учитывающий динамику сети, может уменьшить задержку передачи пакетов данных. Вместе с тем, наряду с классиче-

скими методами и алгоритмами в проектировании сетей, получают распространение интеллектуальные подходы [6]. Использование методов теории искусственного интеллекта (нечеткой логики, нейронных сетей, генетических алгоритмов и т.д.) [7] в алгоритмах адаптации таблиц маршрутов позволяет привнести им дополнительные свойства самообучения, прогнозирования, учета знаний и опыта экспертов - сетевых администраторов. На основании этого можно сделать вывод об актуальности выбранной темы исследования.

Постановка задачи

Цель работы состоит в разработке интеллектуального метода маршрутизации, действующего на основе нечеткого алгоритма Тагаки-Сугено [8] для повышения эффективности функционирования компьютерных сетей с учетом изменений ее параметров и топологии.

Анализ классического подхода

В качестве примера рассмотрим матричный алгоритм поиска кратчайшего пути от узла к узлу (типичный подход) [3].

Допустим, есть компьютерная сеть, имеющая вид графа (рис. 1) и матрица L, отображающая текущее время задержки передачи данных от узла к узлу в компьютерной сети. Задачей маршрутизации является поиск такого маршрута, который позволит доставить информацию за минимальное время.

$$L = \begin{pmatrix} 0 & 50 & \infty & 80 & \infty \\ 50 & 0 & 40 & 90 & 130 \\ \infty & 40 & 0 & \infty & 60 \\ 80 & 90 & \infty & 0 & 25 \\ \infty & 130 & 60 & 25 & 0 \end{pmatrix} \quad (1)$$

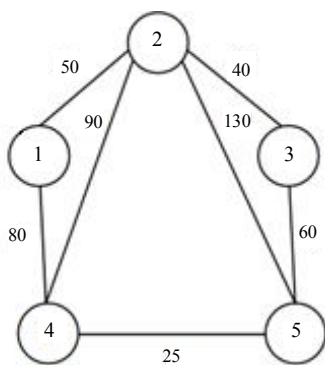


Рис. 1. Граф компьютерной сети

Возведение матрицы L в степень максимального ранга R_{max} даст нам матрицу самых коротких пу-

тей от узла к узлу между всеми парами узлов графа, матрицу оптимальных путей $L_{opt} = L^{(R_{max})}$.

Если при возведении матрицы L в некоторую степень q окажется, что:

$$L^q = L^{(q-1)}, \quad (2)$$

то вычислительный процесс следует прервать, так как тождество (2) влечет за собой тождество

$$L^q = L^{(q+1)}. \quad (3)$$

Элементы матрицы $L^{(2)}$ обозначим - $l_{ij}^{(2)}$ и будем рассчитывать по формуле (4).

$$l_{ij}^{(2)} = \min(l_{ik}^{(1)} + l_{kj}^{(1)}) = \min((l_{i1}^{(1)} + l_{1j}^{(1)}), k=1, \dots, n, (l_{i2}^{(1)} + l_{2j}^{(1)}), \dots, (l_{in}^{(1)} + l_{nj}^{(1)})). \quad (4)$$

Используя формулу (4), вычислим элементы матрицы $L^{(2)}$ (5) – (7).

$$l_{12}^{(2)} = \min(l_{11}^{(1)} + l_{12}^{(1)}, l_{12}^{(1)} + l_{22}^{(1)}, l_{13}^{(1)} + l_{32}^{(1)}, l_{14}^{(1)} + l_{42}^{(1)}, l_{15}^{(1)} + l_{52}^{(1)}) = \min(50, 50, \infty, 170, \infty) = 50, \quad (5)$$

$$l_{13}^{(2)} = \min(l_{11}^{(1)} + l_{13}^{(1)}, l_{12}^{(1)} + l_{23}^{(1)}, l_{13}^{(1)} + l_{33}^{(1)}, l_{14}^{(1)} + l_{43}^{(1)}, l_{15}^{(1)} + l_{53}^{(1)}) = \min(\infty, 90, \infty, \infty, \infty) = 90, \quad (6)$$

$$l_{14}^{(2)} = \min(l_{11}^{(1)} + l_{14}^{(1)}, l_{12}^{(1)} + l_{24}^{(1)}, l_{13}^{(1)} + l_{34}^{(1)}, l_{14}^{(1)} + l_{44}^{(1)}, l_{15}^{(1)} + l_{54}^{(1)}) = \min(80, 140, \infty, 80, \infty) = 80. \quad (7)$$

Аналогично рассчитав все остальные элементы матрицы $L^{(2)}$, получим следующую матрицу (8).

$$L^{(2)} = \begin{pmatrix} 0 & 50 & 90 & 80 & 105 \\ 50 & 0 & 40 & 90 & 130 \\ 90 & 40 & 0 & 85 & 60 \\ 80 & 90 & 85 & 0 & 25 \\ 105 & 130 & 60 & 25 & 0 \end{pmatrix} \quad (8)$$

Теперь сверяем полученную матрицу с исходной и получаем $L^{(2)} \neq L^{(1)}$. Это означает, что поиск оптимальных путей следует продолжить, рассчитав при этом элементы матрицы $L^{(3)} = L^{(2)} * L^{(1)}$ (9) – (11).

$$l_{12}^{(3)} = \min(l_{11}^{(2)} + l_{12}^{(1)}, l_{12}^{(2)} + l_{22}^{(1)}, l_{13}^{(2)} + l_{32}^{(1)}, l_{14}^{(2)} + l_{42}^{(1)}, l_{15}^{(2)} + l_{52}^{(1)}) = \min(50, 50, 130, 170, 235) = 50, \quad (9)$$

$$l_{13}^{(3)} = \min(l_{11}^{(2)} + l_{13}^{(1)}, l_{12}^{(2)} + l_{23}^{(1)}, l_{13}^{(2)} + l_{33}^{(1)}, l_{14}^{(2)} + l_{43}^{(1)}, l_{15}^{(2)} + l_{53}^{(1)}) = \min(\infty, 90, 130, 170, 165) = 90, \quad (10)$$

$$l_{14}^{(3)} = \min(l_{11}^{(2)} + l_{14}^{(1)}, l_{12}^{(2)} + l_{24}^{(1)}, l_{13}^{(2)} + l_{34}^{(1)}, l_{14}^{(2)} + l_{44}^{(1)}, l_{15}^{(2)} + l_{54}^{(1)}) = \min(80, 140, \infty, 80, 130) = 80. \quad (11)$$

По аналогии рассчитываем и остальные элементы матрицы $L^{(3)}$. Полученная матрица $L^{(3)}$ будет иметь вид (12):

$$L^{(3)} = \begin{pmatrix} 0 & 50 & 90 & 80 & 105 \\ 50 & 0 & 40 & 90 & 130 \\ 90 & 40 & 0 & 85 & 60 \\ 80 & 90 & 85 & 0 & 25 \\ 105 & 130 & 60 & 25 & 0 \end{pmatrix} \quad (12)$$

Теперь воспользуемся формулой (2) и получим $L^{(3)} = L^{(2)}$. Из этого делаем вывод, что $L_{\text{опт}} = L^{(3)} = L^{(2)}$, следовательно, оптимальные пути найдены, и расчёт матрицы 4 ранга можно не производить [6].

Следует отметить, что рассмотренный метод позволяет определить длины (ранги, стоимости и т. п.) оптимальных путей между всеми вершинами графа, но не указывает те ветки (или вершины), которые входят в оптимальный маршрут. Этот недостаток устраняется с помощью алгоритма Флойда [3]. Используя этот алгоритм, авторами была получена матрица оптимальных маршрутов – матрица Γ (13).

$$\Gamma = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{pmatrix} 1 & 2 & 2 & 4 & 4 \\ 1 & 2 & 3 & 4 & 3 \\ 2 & 2 & 3 & 5 & 5 \\ 1 & 2 & 5 & 4 & 5 \\ 4 & 2 & 3 & 4 & 5 \end{pmatrix} \end{matrix} \quad (13)$$

Данная матрица позволяет увидеть промежуточные узлы, составляющие оптимальный маршрут.

Классический метод является трудоёмким процессом, требующим проведения значительных расчетов, занимающий много времени и места в памяти маршрутизатора. При перепланировке сети, введении нового элемента или же при изменении других параметров каналов связи, придётся пересчитывать всё сначала, на что понадобится много времени, в результате чего новый маршрут может оказаться неоптимальным.

Предлагаемый альтернативный подход, основан на аппарате нечеткой логики [7] и устраняет недостатки классического подхода. Помимо этого, данный алгоритм основан на мнении экспертов – администратора сети или проектировщика. Эксперт

обладает знаниями о том, что, к примеру, данные узла 1 передаются узлу 3 через узел 2 и представляет эту информацию в виде правила **R** : **ЕСЛИ** задержка передачи данных на узле 1-2 Высокая **И** задержка передачи на узле 2-3 Высокая **ТО** задержка доставки данных от узла 1 к узлу 3 тоже будет Высокой и т.д. Использование опыта и знаний экспертов на этапе проектирования сети и расчете таблиц оптимальных маршрутов позволяет придать маршрутизатору свойство интеллектуальности (адаптивности) и повысит эффективность его работы. Также нечеткий алгоритм способен к учету нелинейных свойств системы (сети).

Разработка интеллектуального маршрутизатора

Предположим, что в процессе функционирования сети произошли изменения (значений задержки пакетов данных) в силу влияния повышенной нагрузки и других факторов. На основе тестовых сигналов маршрутизаторы обмениваются информацией и производят обновление своих таблиц. Так новая информация о задержках на узлах представлена в виде матрицы L^* . С учетом новой информации о задержке между узлами (14) перед устройством стоит задача поиска нового оптимального маршрута.

$$L^* = \begin{pmatrix} 0 & 150 & 250 & 80 & 105 \\ 50 & 0 & 100 & 90 & 100 \\ 90 & 40 & 0 & 85 & 60 \\ 80 & 90 & 85 & 0 & 25 \\ 105 & 130 & 60 & 25 & 0 \end{pmatrix} \quad (14)$$

Так к примеру, с учетом новых значений параметров узлов 1-2, и 1-3 необходим пересчет и поиск новых маршрутов, для передачи пакетов и узла 1 в 3. Воспользуемся интеллектуальным подходом. Так Такаги и Сугено в работе [8] предложили нечеткий алгоритм. Применяемые в нем правила нечетки только в части **IF**, тогда как в **THEN** имеются функциональные зависимости:

$$\begin{aligned} R^{(1)} : & \text{IF}(x_1 \text{ is } A_1^1 \text{ AND } x_2 \text{ is } A_2^1 \text{ AND } \dots x_n \text{ is } A_n^1) \\ & \text{THEN } y_1 = f^{(1)}(x_1, x_2, \dots, x_n), \\ & \dots \\ R^{(N)} : & \text{IF}(x_1 \text{ is } A_1^N \text{ AND } x_2 \text{ is } A_2^N \text{ AND } \dots x_n \text{ is } A_n^N) \\ & \text{THEN } y_N = f^{(N)}(x_1, x_2, \dots, x_n). \end{aligned} \quad (15)$$

Допустим, что на вход модуля нечеткой системы подается сигнал $\bar{x} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$, где x_n – текущее значение параметра задержки (мс), который

может интерпретироваться как вектор состояния объекта (линии связи между хостами), для исследуемого процесса (рис. 1) $n=7$. Определим выходной сигнал \bar{y} (задержку – суммарное значение всего участка). Для правила $R^{(1)}$ рассчитываем (16):

$$w^1 = \left\{ \begin{array}{l} \mu_{A_1^1}(\bar{x}_1), \mu_{A_2^1}(\bar{x}_2), \dots, \mu_{A_n^1}(\bar{x}_n), \\ \text{или} \\ \min \left\{ \mu_{A_1^1}(\bar{x}_1), \mu_{A_2^1}(\bar{x}_2), \dots, \mu_{A_n^1}(\bar{x}_n) \right\} \end{array} \right\}. \quad (16)$$

На следующем этапе рассчитываем:

$$\bar{y}_1 = f^{(1)}(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n).$$

Продолжаем расчеты для остальных правил $R^{(k)}$, $k=2, \dots, N$. Для правила $R^{(N)}$ получаем:

$$w^N = \left\{ \begin{array}{l} \mu_{A_1^N}(\bar{x}_1), \mu_{A_2^N}(\bar{x}_2), \dots, \mu_{A_n^N}(\bar{x}_n), \\ \text{или} \\ \min \left\{ \mu_{A_1^N}(\bar{x}_1), \mu_{A_2^N}(\bar{x}_2), \dots, \mu_{A_n^N}(\bar{x}_n) \right\} \end{array} \right\}, \quad (17)$$

$$\bar{y}_N = f^{(N)}(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n).$$

Выходной сигнал модуля нечеткого вывода Такаги-Сугено представляет собой нормализованную взвешенную сумму отдельных выходов $\bar{y}_1, \dots, \bar{y}_N$, т.е.

$$\bar{y} = \frac{\sum_{k=1}^N w^k \bar{y}_k}{\sum_{k=1}^N w^k}. \quad (18)$$

В сокращенном случае базу правил модуля можно записать в виде:

$$R^{(N)} : \text{IF}(x_1 \text{ is } A_1^k \text{ AND } x_2 \text{ is } A_2^k \text{ AND } \dots x_n \text{ is } A_n^k) \text{ THEN } y_k = c_0^{(k)} + c_1^{(k)}x_1 + \dots + c_n^{(k)}x_n \quad (19)$$

для $k=1, \dots, N$, N – количество нечетких правил.

В формулах (15) – (19): A_n^N – нечеткие множества (лингвистической переменной задержка «низкая», «средняя», «высокая»), $\mu_{A_n^N}(\bar{x})$ – степени истинности функций принадлежности нечетких множеств $\in [0,1]$, w^N – итоговая степень истинности, c_n – коэффициенты уравнения (19), определяемые экспертом [8].

Для апробации предложенного подхода используется пакет MatLab (Fuzzy Logic Toolbox). На начальном этапе выбираются входные и выходные параметры нечеткой модели (рис. 2).

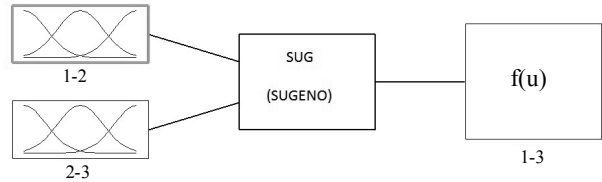


Рис. 2. Интерфейс программы

Следующим этапом является процесс фаззификации параметров (рис. 3-4), в котором используются функции принадлежности треугольного типа. Данный этап проводится с учетом мнения эксперта (системного администратора), который знает возможный диапазон изменения параметров состояния линии связи (задержки) или других данных.

На рисунках 3 и 4 обозначено: N – низкое значение термина, S – среднее, V – высокое.

Следующим, в этапе построения нечеткой модели, является создание базы знаний (реализующей продукционные правила вида «ЕСЛИ ... И ... ТО...» на основе инструкций эксперта (рис. 5). К примеру, ЕСЛИ значение параметра узла «1-2» низкое (N) И значения параметров узла «2-3» низкое (N) ТО значение параметра «1-3» низкое (N) и т.д. Следует отметить, что кроме задержки эксперт также может указать и другие важные параметры сети (длина линий, пропускная способность, надежность).

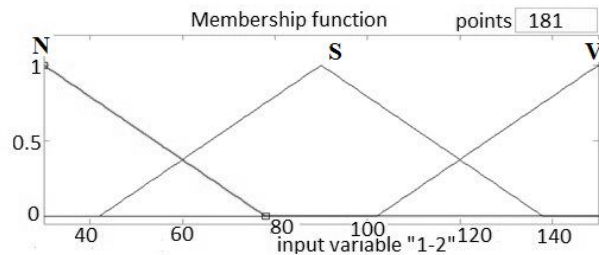


Рис. 3. Фаззификация параметра «задержка данных на узле 1-2»

После формирования базы знаний следует этап проверки адекватности работы интеллектуальной системы на способность расчета задержки (или другого важного параметра) по обновленным данным, полученным от соседних маршрутизаторов (рис. 6). С учетом новой информации задержка на узле 1-3

равняется 250, т.е. нужен новый оптимальный маршрут.

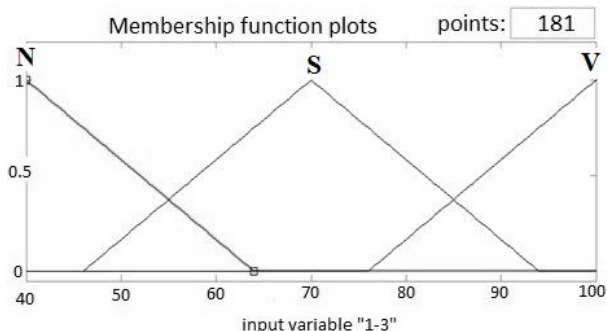


Рис. 4. Фаззификация параметра «задержка данных на узле 2-3»

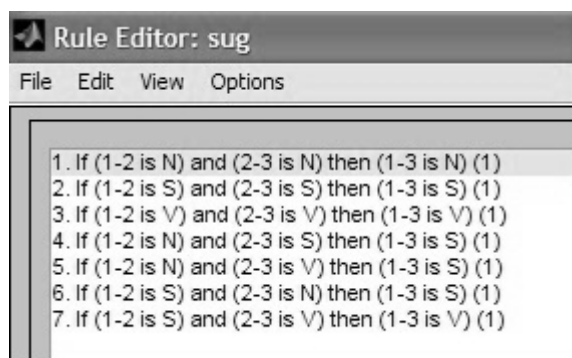


Рис. 5. Фрагмент базы знаний

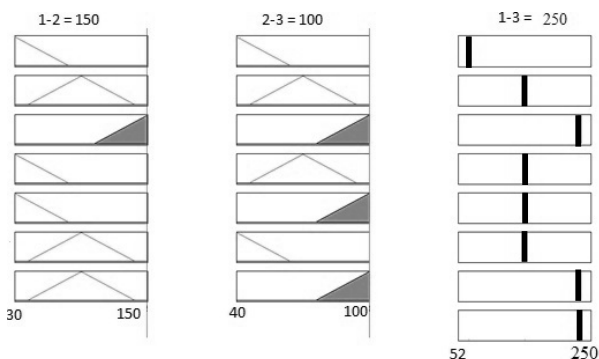


Рис. 6. Интерфейс просмотра результатов системы нечеткого вывода

Аналогично решается задача расчета времени задержки по другим каналам связи, например для 1-3, через 1-4, 4-5, 5-3.

Проверка системы на точность расчета при произвольных входных значениях на других узлах также показала ее адекватность. Представленная система расчета новых значений параметров сети является универсальным аппроксиматором и может быть рекомендована системным администраторам как экспертная или программно реализована в виде алгоритма расчета и выбора оптимального маршрута передачи данных (рис. 7).

Таким образом, интеллектуальный маршрутизатор, реализующий нечеткий подход, при выборе маршрута (1-3) вместо промежуточного узла 2 рекомендует выбор пути через узлы 4 и 5. Следует отметить, что количество входных и выходных параметров в нечеткой системе может быть неограниченно, но при этом пропорционально увеличивается и размер базы знаний, поэтому количество рекомендуемых выходов (информация от узлов), как показало моделирование, не должно превышать трех.

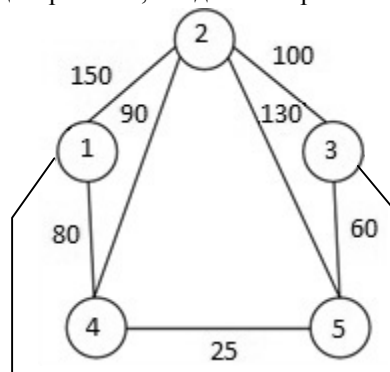


Рис. 7. Демонстрация оптимального маршрута «1-3»

Заключение

Рассмотрен классический метод определения кратчайшего пути от узла к узлу в компьютерной сети. Выявлены недостатки данного метода, а именно трудоемкость пересчета таблицы маршрутизации, а также временные затраты и невозможность самообучения.

Предложен новый научный подход решения задачи.

Рассмотрен интеллектуальный подход, реализующий новый метод адаптивной нечеткой маршрутизации который позволяет уменьшить размерность задачи нахождения оптимальных маршрутов и сократить трудоемкость и время нахождения задачи поиска кратчайших путей с учетом изменений структуры сети и ее параметров. Также подход учитывает рекомендации экспертов, является открытым и понятным обслуживающему персоналу, и легко подлежит коррекции.

Разработана программа (в среде MatLab), позволяющая находить оптимальные пути от узла к узлу в компьютерной сети, основываясь на правилах нечеткой логики.

Литература

1. Перепёлкин, Д. А. Методы и алгоритмы адаптивной маршрутизации в корпоративных вычислительных сетях [Электронный ресурс] : дис. ... канд. техн. наук: 05.13.13 / Перепёлкин Дмитрий Александрович ; Рязанский гос. радиотехн. ун-т.

– Рязань, 2009. – 148 с. – Режим доступа: <http://www.dissercat.com/content/metody-i-algoritmy-adaptivnoi-marshrutizatsii-v-korporativnykh-yuchislitelnykh-setyakh>. – 5.12.2013.

2. Гольдштейн, Б. С. Протоколы сети доступа. Том 2 [Текст] / Б. С. Гольдштейн. – М. : Радио и связь, 1999. – 313 с.

3. Князева, Н. А. Теория проектирования компьютерных систем и сетей [Текст] / Н. А. Князева. – О. : СПД Бровкин О.В., 2012. – 240 с.

4. Столингс, В. Современные компьютерные сети [Текст] / В. Столингс. – СПб. : Питер, 2003. – 783 с.

5. Тимофеев, А. В. Проблемы и методы адаптивного управления потоками данных в телекомму-

никационных системах [Текст] / А. В. Тимофеев. – СПб. : Информатизация и связь, 2003. – 73 с.

6. Тимофеев, А. В. Модели и методы маршрутизации потоков данных в телекоммуникационных системах с изменяющейся динамикой [Текст] / А. В. Тимофеев, А. В. Сырцев. – М. : Издательство «Новые технологии», 2005. – 85 с.

7. Леоненко, А. В. Нечеткое моделирование в среде MATLAB и FuzzyTECH [Текст] / А. В. Леоненко – СПб. : «БХВ-Петербург», 2005. – 736 с.

8. Рутковская, Д. Нейронные сети, генетические алгоритмы и нечеткие системы [Текст]: научное издание / Д. Рутковская, М. Пилиньский, Л. Рутковский ; под ред. А. С. Попова. – М. : Горячая линия-Телеком, 2006. – 383 с.

Поступила в редакцию 7.12.2013, рассмотрена на редколлегии 12.02.2014

Рецензент: д-р техн. наук, проф., зав. кафедрой информационных технологий и кибербезопасности В. М. Плотников, Одесская национальная академия пищевых технологий, Одесса.

ВИКОРИСТАННЯ АЛГОРИТМУ ТАКАГІ-СУГЕНО У АДАПТИВНИХ МЕТОДАХ МАРШРУТИЗАЦІЇ КОМП'ЮТЕРНИХ МЕРЕЖ

В. С. Михайленко, А. М. Асланов, М. С. Солодовник

Запропоновано новий підхід для вирішення завдання пошуку оптимального маршруту в корпоративній мережі, діючий на основі теорії нечіткої логіки. Даний підхід дозволяє значно скоротити час знаходження оптимального маршруту від вузла до вузла при мережевої маршрутизації, тому що враховує структуру мережі та її параметри. Розглянуто класичний метод пошуку найкоротших шляхів та інноваційний нечіткий адаптивний метод, проведено порівняння цих двох методів та виявлено переваги запропонованого методу перед класичним. Використовуючи пакет Fuzzy logic Toolbox середовища MatLab, розроблено програму, що моделює процес знаходження найкоротшого шляху від вузла до вузла в корпоративній мережі на основі нечіткого підходу, побудовано відповідні графи.

Ключові слова: оптимальний маршрут, адаптивна маршрутизація, метод Такагі-Сугено, інтелектуальний маршрутизатор, фазифікація.

USING THE ALGORITHM OF TAKAGI-SUGENO IN ADAPTIVE ROUTING METHODS OF THE COMPUTER NETWORKS

V. S. Mikhailenko, A. M. Aslanov, M. S. Solodovnik

A new approach for solving the problem of finding the optimal route in the corporate network, acting on the basis of the theory of fuzzy logic was proposed. This approach significantly reduces the time to find the optimal route from node to node in the network routing, as takes into account the structure of the network and its parameters. The classical method for finding shortest paths and innovative fuzzy adaptive method, a comparison of these two methods and revealed the advantages of the proposed method over the classical. Using the package Fuzzy logic Toolbox environment MatLab, developed a program that simulates the process of finding the shortest path from node to narrow in corporate network based on fuzzy approach, the corresponding graphs are constructed.

Keywords: optimal route, adaptive routing, Takagi-Sugeno approach, intellectual router, fuzzification.

Михайленко Владислав Сергеевич – канд. техн. наук, доцент кафедри інформаційних систем і мереж, Інститут холода, криотехнологій і екоенергетики ім. проф. В. С. Мартыновського, Одесская национальная академия пищевых технологий, Одесса, Украина, e-mail: vlad_mihailenko@mail.ru.

Асланов Алексей Михайлович – аспирант кафедри інформаційних систем і мереж, Інститут холода, криотехнологій і екоенергетики ім. проф. В. С. Мартыновського, Одесская национальная академия пищевых технологий, Одесса, Украина.

Солодовник Михаил Сергеевич – аспирант кафедри Інформаційних систем і мереж, Інститут холода, криотехнологій і екоенергетики ім. проф. В. С. Мартыновського, Одесская национальная академия пищевых технологий, Одесса, Украина, e-mail: respect_all@ukr.net.