

УДК 539.3

*И.А. ПОТАПЧУК, А.В. БАГРЯНЦЕВА***МЕТОД МАРШРУТИЗАЦИИ В БЕСПРОВОДНОЙ САМООРГАНИЗУЮЩЕЙСЯ СЕТИ НА ОСНОВЕ АППАРАТА НЕЧЕТКОЙ ЛОГИКИ**

Разработаны и исследованы модель и метод применения нечеткой логики для решения задач маршрутизации в БСС–сетях. Предложен модифицированный метод маршрутизации для БСС–сети, базирующийся на аппарате нечеткой логики. Разработана общая структура контроллера нечеткой логики, определены входные и выходные параметры. Проведен сравнительный анализ предложенного метода с классическим AODV.

**Ключевые слова:** беспроводные сети, маршрутизация, нечеткая логика, нечеткий контроллер, нечеткие данные о состоянии радиосети, QoS, FAODV.

**Актуальность темы.** В последнее время развитие беспроводных сетевых технологий идет в направлениях: увеличения скорости передачи данных, повышения степени мобильности пользователей, расширения количества предоставляемых услуг, улучшения степени использования радиочастотного спектра и степени интеллектуальности сетевого и абонентского оборудования. В этих условиях начал проявляться ряд противоречий в области беспроводных сетевых технологий, наиболее острыми из которых являются:

– противоречие между увеличивающимся спросом на услуги беспроводных сетей связи (на рабочие частоты) и природными ограничениями частотных ресурсов,

– противоречие между расширением спектра предоставляемых беспроводными сетями услуг, увеличением требований к их качеству и использованием традиционных (устаревших) технологий управления.

Одним из эффективных методов разрешения перечисленных противоречий является применение в беспроводных телекоммуникационных системах элементов искусственного интеллекта, а, в свою очередь, одним из современных подходов к реализации данных элементов является применение нечеткой логики [1] и нечетких процессоров.

Отличительная черта беспроводных самоорганизующихся сетей [2] (БСС) – отсутствие фиксированной структуры и централизованного управления. Поэтому каждый узел в такой сети должен самостоятельно определять наилучший маршрут для передачи данных другим узлам. Алгоритмы, разработанные специально для решения этой задачи, получили название алгоритмов маршрутизации [3].

В отличие от классических проводных сетей [4], где топология изменяется очень редко, маршрутизация в мобильных сетях сопряжена со значительными трудностями. С одной стороны, маршрутизация является основой функционирования всей сети и должна работать максимально надежно. С другой, неустойчивая природа БСС–сетей не дает возможности применять «проверенные» способы поддержки маршрутной информации, сводя их эффективность к нулю.

Единого метода маршрутизации, удовлетворяющего требования QoS и обеспечивающего оптимизацию всех показателей эффективности функционирования сети при различных условиях ее работы, не существует. В связи с этим возникает задача разработки

и исследования новых подходов к решению задачи маршрутизации в БСС–сетях.

Таким образом, **целью данной работы** является разработка эффективного метода маршрутизации в БСС–сетях с применением элементов нечеткой логики.

**Для решения задачи маршрутизации** рассматривается реактивный протокол FAODV (FuzzyAd-hocOn-demandDistanceVector– нечеткая одноранговая дистанционно–векторная маршрутизация по требованию). Реактивный протокол маршрутизации был выбран по причине его лучшей масштабируемости в больших самоорганизующихся сетях. FAODV строит маршруты, используя цикл «запрос–ответ». Реактивные протоколы не требуют периодических обновлений таблиц маршрутизации, сохраняя пропускную способность беспроводной среды и экономя запас энергии батарей мобильных терминалов. Такие протоколы не требуют никаких излишних затрат при возникновении изменений в топологии мобильной сети, а особенно, если возникает незначительное движение узлов. Именно протоколы по запросу способны динамически определять пути передачи с достаточной полосой пропускания и требуемой задержкой, несмотря на частые изменения топологии сети и выполнять необходимые сервисные требования [5].

Основным недостатком работы такого протокола является внесение задержки [6,7] между моментом прихода запроса установить соединение от приложения и моментом установления соединения из–за реактивной природы протокола. Этот недостаток в предложенном протоколе частично компенсируется механизмом кэширования, который является настраиваемым в зависимости от условий работы сети.

Алгоритм работы протокола состоит в следующем (события и соответствующие им процедуры обработки):

- Запрос установления соединения. Если маршрут до требуемого узла назначения не был найден в кэше, то узел формирует пакет RouteRequest (запрос маршрута), где указывает требуемый адрес назначения и свой адрес, рассылает этот пакет широкоэмитивно и устанавливает таймер ожидания ответов. Сессия передачи данных переводится в режим ожидания.
- Получение пакета RouteRequest. Узлы, получая пакет RouteRequest, проверяют, не являются ли

они тем узлом, с которым запрашивающий узел хочет установить соединение. Если нет, то этот узел добавляет себя в список промежуточных узлов, через которые прошел пакет, и рассылает пакет дальше широко-вещательно. Кроме того, узел записывает в пакет некоторые показатели своего состояния, которые потребуются в конечной стадии для выбора оптимального маршрута (эти показатели подробно рассмотрены ниже в данной главе). В противном случае пакет достиг узла назначения, и данный узел отправляет ответ RouteReply по маршруту, записанному в пришедшем пакете RouteRequest.

- Получение пакета RouteReply. Как было сказано выше, после отправки запроса поиска маршрута узел устанавливает таймер ожидания ответов, во время работы которого он собирает пакеты RouteReply с найденными маршрутами. Если же пакет RouteReply приходит после срабатывания этого таймера, то он отбрасывается.

- Срабатывание таймера ожидания ответов RouteReply. Когда приходит запрос установления соединения, узел отправляет RouteRequest и устанавливает таймер ожидания ответов. В течение этого периода узел собирает полученные пакеты RouteReply с различными маршрутами к узлу назначения во временную таблицу. Пакеты с описанием потенциальных маршрутов, пришедшие после истечения таймера ожидания, не учитываются. При срабатывании таймера собранные маршруты передаются в блок принятия решения, который и определяет, какой из маршрутов оптимален для передачи пакета с требуемым QoS.

После выбора оптимального маршрута узел записывает этот маршрут в таблицу маршрутизации на определенный период времени (кэширует маршрут). Если приходит новый запрос соединения, и адрес назначения содержится в таблице, новый запрос RouteRequest не формируется, а данные отправляются по ранее сохраненному маршруту, как было сказано выше. Кроме того, для обеспечения устойчивости к сбоям узлов на выбранном маршруте создается резервная таблица маршрутизации. Так как протокол маршрутизации обнаруживает множество маршрутов до узла назначения, то из данного множества выбирается один оптимальный маршрут, как было показано выше, плюс два запасных. Запасные маршруты выбираются по следующему алгоритму:

- Результаты оценки маршрута проходят через блок принятия решения, где вычисляется их рейтинг. Все маршруты записываются с соответствующими им рейтингами во временную таблицу.

- Затем выполняется проход по всем строкам таблицы и отмечаются маршруты, множество промежуточных узлов которых не пересекается с множеством промежуточных узлов выбранного оптимального маршрута. Таким образом, мы находим множество непересекающихся маршрутов. Это важно, поскольку снижается вероятность того, что давший сбой на оптимальном маршруте узел повлияет на работу двух запасных маршрутов.

- Из множества отмеченных маршрутов выбирается два маршрута с наибольшим рейтингом, кото-

рые и становятся запасными.

- Если множество отмеченных маршрутов пусто, повторяем процедуру выбора запасных маршрутов, но отмечаем те маршруты, которые пересекаются с выбранным в одном узле.

- Получение пакета данных. При получении пакета данных узел просто извлекает из него адрес следующего узла по маршруту передачи этого пакета и передает его данному узлу.

- Срабатывание таймера сброса кэша. При срабатывании этого таймера соответствующая запись удаляется из таблицы маршрутизации. Таким образом, данный протокол позволяет обнаруживать подмножество всех возможных маршрутов до узла назначения, передавать данные о состоянии узлов и каналов связи в контрольных сообщениях, а также кэшировать выбранные маршруты для повторного использования позднее. Кроме того, он обеспечивает устойчивость к сбоям на маршруте передачи данных за счет выбора двух резервных маршрутов, не имеющих пересечения с основным.

**Модель сети с нечеткими данными о состоянии радиолиний.** Учет большего количества параметров состояния узлов и каналов связи позволяет лучше оценить картину распределения трафика в сети и выбирать более подходящий маршрут. Для предлагаемого метода маршрутизации были выбраны следующие параметры: пропускная способность канала, джиттер, задержка передачи, загруженность (размер свободной очереди) и число «скачков» (количество узлов).

Сеть представляется ненаправленным графом  $G = (V, E)$ , где  $V$  – множество узлов;  $E$  – множество каналов. Каждый канал  $(i, i+1) \in E$  и узел  $V_i \in V$  могут характеризоваться положительной метрикой  $c_\eta(i, i+1), c_\eta(v_i) \in Z^+$ .

Например, значениями  $c_\eta$  могут быть:

- для канала: пропускная способность  $B(e) \in Z^+$ , задержка передачи  $d(e) \in Z^+$  и джиттер  $\Delta d(e) \in Z^+$ ;

- для узла: количество узлов  $q \in Z^+$ , загруженность пакетного буфера  $l(v) \in Z^+$ .

Обозначим циклический маршрут  $p$  в  $G$  как последовательность узлов  $(v_1, \dots, v_i, \dots, v_n)$  таких, что  $i, 1 < i < n, (v_i, v_{i+1}) \in E$ . Определим параметры состояния узлов и каналов связи для каждого  $p$  (маршрута): пропускная способность  $B(p)$ , задержка распространения  $d(p)$ , джиттер задержки  $\Delta d(p)$ , количество узлов  $N(p)$  и загруженность пакетного буфера для всего маршрута  $\bar{l}(p)$ . Далее приведено краткое описание каждого из них:

- Пропускная способность канала связи является минимумом пропускной способности среди всех каналов на маршруте:

$$B(p) = \min_{i \in p} \{B(v_i, v_{i+1})\}. \quad (1)$$

- Задержка передачи пакета. Задержка в данном случае означает период времени, проходящий от момента отправки кадра уровня MAC до момента прихо-

да подтверждения о его получении от другого узла. Задержка передачи пакета зависит от множества факторов, но ее значение мы будем определять как значение задержки для последней произведенной передачи. Значения задержки для всех промежуточных каналов связи суммируются:

$$d(p) = \sum_{i=1}^{n-1} d(v_i, v_{i+1}). \quad (2)$$

• Джиттером всего маршрута считается максимум джиттера среди всех каналов на маршруте:

$$\Delta d(p) = \max_{i \in p} \{B(v_i, v_{i+1})\}. \quad (3)$$

• Загруженность пакетного буфера узла составляет:

$$l_i = N_i / B_i. \quad (4)$$

Суммарная загруженность буферов на маршруте выражается как:

$$L = \sum_{i=1}^n l_i. \quad (5)$$

Весовой коэффициент для загруженности каждого узла в отдельности в составе суммарной загруженности:

$$\lambda_i = l_i / L, \text{ при условии } L \neq 0. \quad (6)$$

Относительная загруженность пакетных буферов узлов на маршруте вычисляется как:

$$\bar{l} = \sum_{i=1}^n l_i \cdot \lambda_i. \quad (7)$$

Предлагаемая оценочная функция позволяет учитывать относительную загруженность пакетного буфера каждого из узлов на маршруте. Например, пусть маршрут состоит из четырех узлов, и оценка загруженности маршрута даст следующее (табл. 1):

Таблица 1 – Оценка загруженности маршрута

	У1	У2	У3	У4	
$l_i$	0,9	0,2	0,1	0,1	$L = 1,3$
$\lambda_i$	0,69	0,15	0,08	0,08	
$l_i \cdot \lambda_i$	0,621	0,03	0,008	0,008	$\bar{l} = 0,668$

Для сравнения, среднее значение загруженности пакетных буферов дало бы значение  $\frac{0,9+0,2+0,1+0,1}{4} = 0,325$ . Предлагаемая же оценочная

функция отражает тот факт, что пакетный буфер одного из узлов практически полный. Таким образом, каждый из промежуточных узлов при получении пакета запрос маршрутизации RouteRequest вычисляет значение  $l_i$ , и записывает его вместе со своим адресом в пакете. В результате в пакете RouteRequest передается два вектора: вектор адресов промежуточных узлов и вектор значений  $l_i$  промежуточных узлов. Когда узел-источник запроса RouteRequest получает ответ RouteReply, он извлекает все значения  $l_i$ , для всех промежуточных узлов и вычисляет значение  $\bar{l}$  для оценки загруженности маршрута.

• Количество промежуточных узлов. При прочих равных условиях более короткие маршруты могут оказаться предпочтительнее, поскольку передача через меньшее количество узлов оказывает меньшее влияние на текущую картину передачи радиосигналов между узлами сети. Список всех промежуточных узлов до узла назначения содержится в ответе маршрутизации.

В результате, после получения узлом-источником

ответа маршрутизации, значения перечисленных параметров извлекаются из пакета и подаются на вход контроллера нечеткой логики, принимающего решения о выборе маршрута.

**Применение аппарата нечеткой логики для решения задач маршрутизации.** В качестве блока принятия решения о выборе оптимального маршрута и двух запасных маршрутов предлагается использовать контроллер нечеткой логики. Одним из основных преимуществ контроллера нечеткой логики является то, что он позволяет учитывать множество параметров для принятия решения, не требуя сложных математических вычислений.

В [8] нечеткая логика используется для много-маршрутной маршрутизации пакетов различного приоритета в БСС-сетях. Для обнаружения возможных маршрутов использовался реактивный протокол маршрутизации DSR, затем контроллер на основе нечеткой логики принимал решение, посылать ли пакет по одному маршруту, по нескольким маршрутам одновременно или вовсе отбросить пакет. Контроллер строил свое решение на основе лишь двух параметров: загруженности буферов промежуточных узлов обнаруженных маршрутов и приоритета пакета.

Правила применения нечеткой логики для решения задачи маршрутизации выглядят следующим образом: «Если буферы узлов на маршруте загружены СИЛЬНО и приоритет пакета НИЗКИЙ, то ОТБРОСИТЬ пакет». Нечеткая логика позволяет в простой форме без построения точной математической модели учитывать множество параметров при управлении процессом маршрутизации и превосходит по некоторым параметрам традиционные протоколы маршрутизации.

Предлагаемый метод маршрутизации в БСС-сетях включает три этапа:

1) Запуск процедуры обнаружения возможных маршрутов до узла назначения, при этом контрольные пакеты маршрутизации также передают параметры состояния узлов и каналов связи.

2) Вычисление рейтингов обнаруженных маршрутов. Значения параметров состояния каждого маршрута подаются на вход контроллера нечеткой логики, который проводит фазификацию всех значений и затем в соответствии с установленной базой правил выполняет операцию нечеткого вывода. На выходе контроллера получается четкий (численный) рейтинг каждого маршрута. Маршрут с наибольшим рейтингом считается оптимальным.

3) Из множества остальных обнаруженных маршрутов происходит выбор еще двух маршрутов с наибольшими рейтингами, которые и становятся запасными.

В результате работы предлагаемого алгоритма, на этапе установления соединения по запросу формируется несколько (в рассматриваемом случае три: один основной и два резервных) маршрутов. В процессе ведения информационного обмена осуществляется динамическая оценка рабочего маршрута. В случае снижения характеристик основного маршрута ниже порогового значения (определяемого приложением) из-за повышенной мобильности узлов или из-за

плохих погодних умов, или если узел или часть узлов покинет маршрут, уточняются характеристики двух оставшихся (запасных) маршрутов и выбирается лучший из них.

Если среди оставшихся (запасных) маршрутов не находится подходящего по качеству, производится повторное восстановление соединения.

**Разработка блока принятия решения (контроллера) и результаты моделирования.** С каждым входным параметром нечеткого контроллера сопоставляется лингвистическая переменная, имеющая пять термов:

- 1) пропускная способность – «очень низкая», «низкая», «средняя», «большая», «очень большая»;
- 2) задержка передачи пакета – «очень низкая», «низкая», «средняя», «большая», «очень большая»;
- 3) джиттер – «очень малый», «малый», «средний», «высокий», «очень высокий»;
- 4) загруженность пакетных буферов – «очень малая», «малая», «средняя», «высокая», «очень высокая»;
- 5) количество промежуточных узлов – «очень малое», «малое», «среднее», «большое», «очень большое».

В качестве функций принадлежности для каждого терма всех лингвистических переменных выбраны треугольные функции принадлежности, поскольку, как показывает практика и исследования [9], они хорошо подходят для использования при решении подобных проблем.

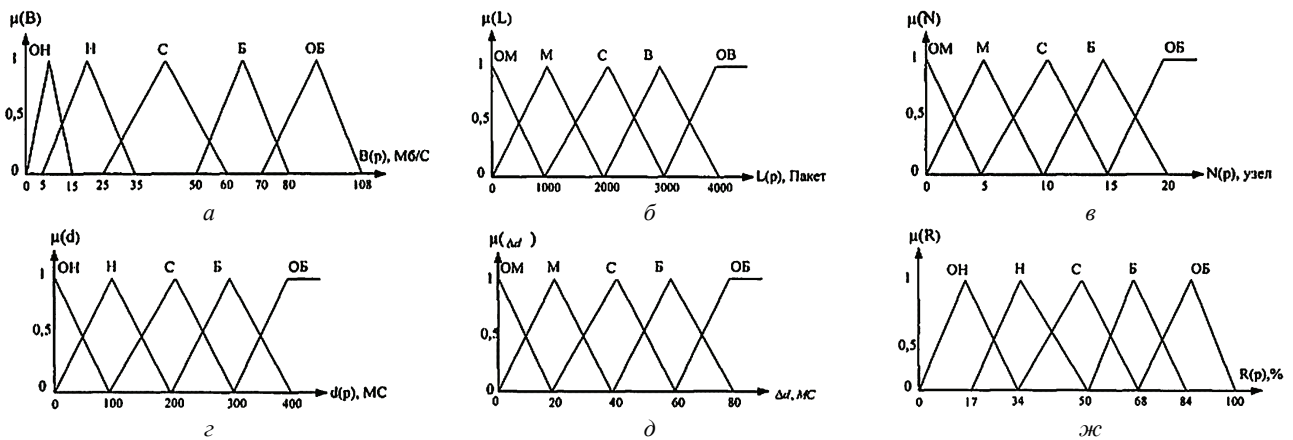


Рисунок 2 – Функции принадлежности лингвистических переменных: а – пропускная способность; б – загруженность пакетных буферов; в – количество скачков; г – задержка пакета; д – джиттер; ж – рейтинг маршрута

Выводом каждого правила импликации [10] является лингвистическая переменная «рейтинг маршрута»  $R(p)$ , множество значений которой также состоит из пяти термов от «очень низкий» до «очень большой».

Логика работы нечеткого контроллера состоит в следующем: для каждого обнаруженного маршрута вычисляется его точный рейтинг в соответствии с базой правил.

База правил нечеткого вывода состоит из правил следующего вида: «если загруженность пакетных буферов ВЫСОКАЯ и доступная пропускная способность НИЗКАЯ и задержка передачи пакета ВЫСОКАЯ и джиттер СРЕДНИЙ и количество промежуточных узлов БОЛЬШОЕ, то рейтинг маршрута

Функция принадлежности треугольного вида определяется как:

$$f(x, a, w) = \begin{cases} 0, & \text{если } x \leq a - w; \\ \frac{x - (a - w)}{w}, & \text{если } (a - w) \leq x \leq a; \\ \frac{(a + w) - x}{w}, & \text{если } a \leq x \leq (a + w); \\ 0, & \text{если } x \geq a + w. \end{cases} \quad (8)$$

где  $a$  является точкой, где значение функции принадлежности равно 1,  $w$  – половина ширины основания треугольника функции принадлежности,  $\mu(x)$  – собственно значение самой функции (рис. 1).

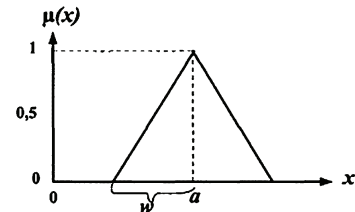


Рисунок 1 – Функция принадлежности треугольного вида

Функции принадлежности показаны на рис. 2 (условные обозначения лингвистических переменных на рис. 2: ОН – «очень низкая», Н – «низкая», С – «средняя», Б – «большая», ОБ – «очень большая», ОМ – «очень малая», ОВ – «очень высокая»).

ОЧЕНЬ НИЗКИЙ».

Количество правил нечеткого вывода в базе правил  $5^5 = 3125$ , поскольку каждая из пяти лингвистических переменных может принимать пять различных значений (полная база правил не приведена здесь из-за ее большого размера). На основе разработанных правил вычисляются значения соответствующей четкой переменной (в нашем случае рейтинг маршрута), которые и управляют выбором оптимального маршрута.

На выходе нечеткого контроллера находим точный рейтинг маршрута, который получается в результате дефазификации выходного нечеткого решения. В качестве Т-нормы будем использовать операцию минимума (min), в качестве S-нормы – операцию максимума (max):

$$\mu_A(x)^{-T} \mu_B(x) = \min(\mu_A(x), \mu_B(x)), \quad (9)$$

$$\mu_A(x)^{-S} \mu_B(x) = \max(\mu_A(x), \mu_B(x)) \quad (10)$$

Правило нечеткой импликации задается правилом Мамдани:

$$\mu_{A \rightarrow B}(x, y) = \mu_R(x, y) = \mu_A(x) \cap \mu_B(y) = \min(\mu_A(x), \mu_B(y)), \quad (11)$$

где  $A$  и  $B$  – нечеткие множества  $A \subseteq X$ ,  $B \subseteq Y$ , отношение  $R$  определено на  $X \times Y$ .

На выходе блока выработки решения получаем один нечеткий вывод  $\bar{B}^k \subseteq Y$  (рейтинг маршрута). Нечеткий вывод определяется следующим.

Условие:

$$X = (x_1, x_2, x_3, x_4, x_5) - \text{это } A', \quad (12)$$

где  $A' = (A'_1 * A'_2 * A'_3 * A'_4 * A'_5)$ .

Импликация:

$$U_{k=1}^N R^{(k)}, R^{(k)} : A^{(k)} \rightarrow B^{(k)}. \quad (13)$$

Вывод:

$$y - \text{это } B',$$

где  $\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4, \bar{x}_5$  – соответственно входные переменные,  $R^{(k)}, k = 1, \dots, N$  – правила нечеткого вывода,  $B'$  – выходное нечеткое множество, определяющее рейтинг маршрута.

Таким образом, выходной рейтинг маршрута вычисляется как:

$$B' = A' \circ U_{k=1}^N R^{(k)}. \quad (14)$$

Так как правило нечеткой импликации мы определили T-нормой типа min, то выполняется:

$$B' = A' \circ U_{k=1}^N R^{(k)} = U_{k=1}^N R^{(k)} \circ A'. \quad (15)$$

Следовательно, функция принадлежности нечеткого множества  $B'$  равна:

$$\mu_{\bar{B}^k}(y) = \max_{k=1, \dots, N} \mu_{\bar{B}^k}(y), \quad (16)$$

где функция принадлежности  $\mu_{\bar{B}^k}(y)$  задается выражением:

$$\mu_{\bar{B}^k}(y) = \mu_{A^k \rightarrow \bar{B}^k}(\bar{x}, y), \quad (17)$$

Таким образом, на вход нечеткого контроллера подается  $\bar{x} = (\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4, \bar{x}_5)$ . Исходя из (17), получаем:

$$\mu_{\bar{B}^k}(y) = \mu_{A^k \rightarrow \bar{B}^k}(\bar{x}, y) =$$

$$\mu_{(A_1^k * A_2^k * A_3^k * A_4^k * A_5^k) \rightarrow \bar{B}^k}(\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4, \bar{x}_5, y). \quad (18)$$

Так как в качестве нечеткой импликации используется правило минимума, а декартово произведение нечетких множеств задается также операцией минимум, получаем:

$$\begin{aligned} \mu_{\bar{B}^k}(y) &= \min \left[ \mu_{(A_1^k * A_2^k * A_3^k * A_4^k * A_5^k) \rightarrow \bar{B}^k}(\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4, \bar{x}_5), \mu_{\bar{B}^k}^k(y) \right] = \\ &= \min \left[ \mu_{A_1^k}(\bar{x}_1), \mu_{A_2^k}(\bar{x}_2), \mu_{A_3^k}(\bar{x}_3), \mu_{A_4^k}(\bar{x}_4), \mu_{A_5^k}(\bar{x}_5), \mu_{\bar{B}^k}^k(y) \right]. \quad (19) \end{aligned}$$

В результате, используя выражение (16), получаем:

$$\begin{aligned} \mu_{\bar{B}'}(y) &= \max_{k=1, \dots, N} \left\{ \min \left[ \mu_{A_1^k}(\bar{x}_1), \mu_{A_2^k}(\bar{x}_2), \mu_{A_3^k}(\bar{x}_3), \right. \right. \\ &\quad \left. \left. \mu_{A_4^k}(\bar{x}_4), \mu_{A_5^k}(\bar{x}_5), \mu_{\bar{B}^k}^k(y) \right] \right\}. \quad (20) \end{aligned}$$

Дефазификацию выходного значения контроллера (точный рейтинг маршрута) будем производить

по методу центра тяжести. Поскольку используются функции принадлежности треугольного вида, то:

$$R' = \frac{\int_R r u_R(r) d(r)}{\int_R u_R(r) d(r)} \quad (21)$$

где  $\mu_R(r)$  – функция принадлежности правила выходного нечеткого множества. Ранее было установлено, что функции принадлежности входных и выходных переменных являются симметричными треугольными функциями принадлежности, описываемыми правилами нечеткого вывода (8).

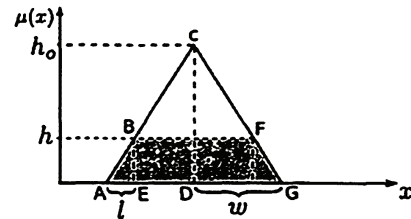


Рисунок 3 – Вычисление площади фигуры, образованной выходным нечетким множеством

Площадь закрашенной трапеции на рис. 3 вычисляется по формуле:

$$S = \frac{1}{2} (AG + BF)h = \frac{1}{2} (2w + (2w - 2l))h. \quad (22)$$

Для вычисления  $l$  используем тот факт, что треугольники  $\triangle ABE$  и  $\triangle ACD$  подобны (по двум углам).

Следовательно,  $\frac{CD}{BE} = \frac{AD}{AE}$  то есть  $\frac{h_0}{h} = \frac{w}{l}$ , откуда

$$l = \frac{w * h}{h_0}. \quad (23)$$

Таким образом, площадь фигуры внутри треугольника с вершущкой, «отрезанной» на высоте  $h$  (как на рис. 3), вычисляется по формуле:

$$S = \frac{1}{2} (2w + (2w - 2l))h = 2wh \left( 1 - \frac{h}{2h_0} \right), \quad (24)$$

где  $h_0 = 1$  (максимальное значение функции принадлежности). Поэтому вычисление точного рейтинга маршрута производится достаточно просто.

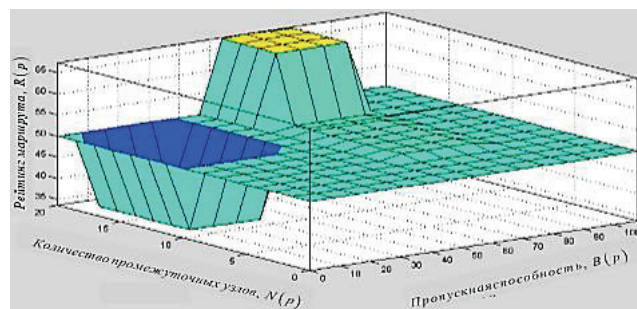


Рисунок 4 – Зависимость рейтинга маршрута  $R(p)$  от входных переменных нечеткой системы – пропускной способности  $B(p)$  и числа узлов  $N(p)$

Для более наглядного отображения работы нечеткого контроллера в соответствии с этим методом был построен график (рис. 4), отображающий зависимости выходной переменной от входных. По осям откладываются входные переменные  $B(p)$ ,  $d(p)$ ,  $\Delta d(p)$ ,  $N(p)$  и  $l(p)$ , а по вертикальной оси – соответствующие

значение  $R(p)$ . Полученный график представляет собой дефазифицированную поверхность нечеткого рейтинга маршрута.

Для получения графического отображения этой функции, была построена визуальная модель (рис. 5) в пакете MatLab. Из-за большого размера базы правил (3125) были приведены только 25 правил.

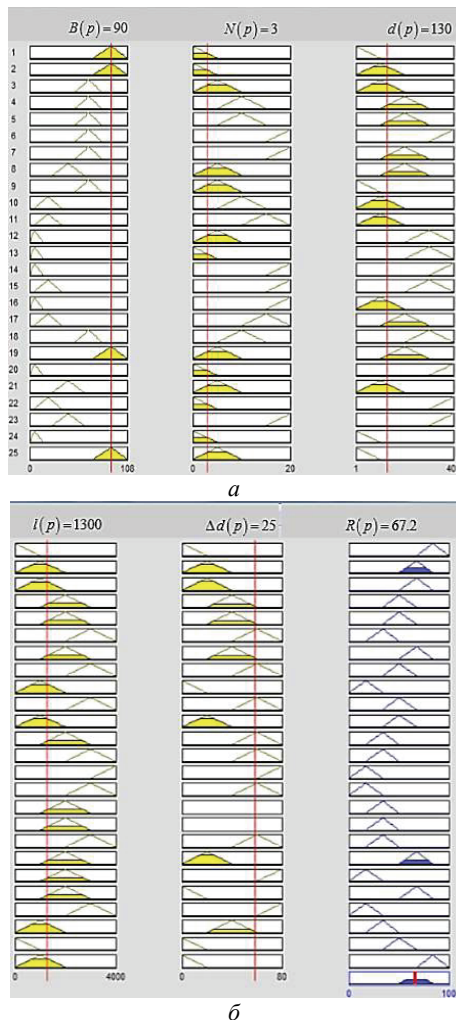


Рисунок 5 – Визуализация правил нечеткого вывода:  $B(p)$  – пропускная способность,  $N(p)$  – число узлов,  $d(p)$  – задержка распространения,  $l(p)$  – пакетная загруженность буфера,  $\Delta d(p)$  – джиттер задержки,  $R(p)$  – рейтинга маршрута

Таблица 2 – Параметры маршрутов в сети

Маршрут	$B(p)$	$d(p)$	$\Delta d(p)$	$N(p)$	$\bar{l}(p)$
A–D–G–J	90	130	25	3	1300
A–B–E–K–J	68	200	40	5	2200
A–C–F–I–J	70	100	35	4	3300

Рассмотрим пример, когда сеть состоит из 10 узлов, расположенных на участке площадью 700x700 кв.м. Предложим что станция А ищет оптимальный маршрут для передачи пакетов данных станции J. Таким образом, протокол маршрутизации по требованию запускает процедуру обнаружения возможных маршрутов до узла J, при этом контрольные пакеты маршрутизации также передают параметры состояния

узлов и каналов связи. В результате найдены три возможных маршрута «A–D–G–J», «A–B–E–K–J», «A–C–F–I–J», и их параметры как показаны в табл. 2.

Получая ответы от станции J, станция А производит оценки пяти параметров на каждом обнаруженном маршруте.

Собрав необходимые данные, станция А осуществляет перевод параметров состояния каждого из обнаруженных маршрутов в нечеткий формат, используя функции принадлежности (рис. 2) для каждого из параметров (табл. 3).

Таблица 3 – Оценка параметров маршрутов

М-г	Л.П/Ф.П	$\mu(B)$	$\mu(d)$	$\mu(\Delta d)$	$\mu(N)$	$\mu(l)$
A–D–G–J	ОБ	0,95	0	0	0,45	0
	Б	0	0	0,85	0,55	0,65
	С	0	0,3	0,2	0	0,25
	Н	0	0,6	0	0	0
A–B–E–K–J	ОБ	0	0	0	0	0
	Б	0,85	0	0	0	0,15
	С	0	0,95	1	0	0,9
	Н	0	0	0	1	0
A–C–F–I–J	ОБ	0	0	0	0	0,4
	Б	0,6	0	0,4	0	0,6
	С	0	0	0,55	0	0
	Н	0	1	0	0,65	0
	ОН	0	0	0	0,2	0

С помощью построенной модели нечеткой логики в пакет MatLab, рейтинг каждого маршрута получается так:

$$R(p)_{A-D-G-J} = 67,2 \%, \\ R(p)_{A-B-E-K-J} = R(p)_{A-C-F-I-J} = 50 \%$$

Следовательно, при данных значениях параметров состояния канала и узлов связи, оптимальным маршрутом считается маршрут «A–D–G–J», по которому и будет осуществляться передача. В случае снижения характеристик основного маршрута ниже порогового значения, или если узел или часть узлов покинет маршрут, осуществляется уточнение характеристик двух оставшихся (запасных) маршрутов и выбирается лучший из них.

Для сравнения эффективности работы предложенного метода маршрутизации с существующими методами выбрано два критерия качества обслуживания: доля доставленных пакетов и средняя задержка передачи пакета для всей сети в целом. В качестве протокола для сравнения выбран AODV [3, 11].

Топология смоделированной сети включает 10/20 узлов, расположенных на участке площадью 700x700 кв.м. Скорость движения узлов от 0 до 10 м/с. Размер пакета данных 512 бит. Каждый источник передает как минимум 2 пакета, как максимум 4 пакета в секунду.

Результаты моделирования на рис. 6. показывают, что при количестве узлов в сети 10 доля доставленных пакетов увеличивается на 5,07 %, средняя задержка пакета на маршруте снижается на 49,517 %. При 20 узлах, доля доставленных пакетов увеличивается на 4,96 %, задержка пакета на маршруте снижается на 45,32 %.

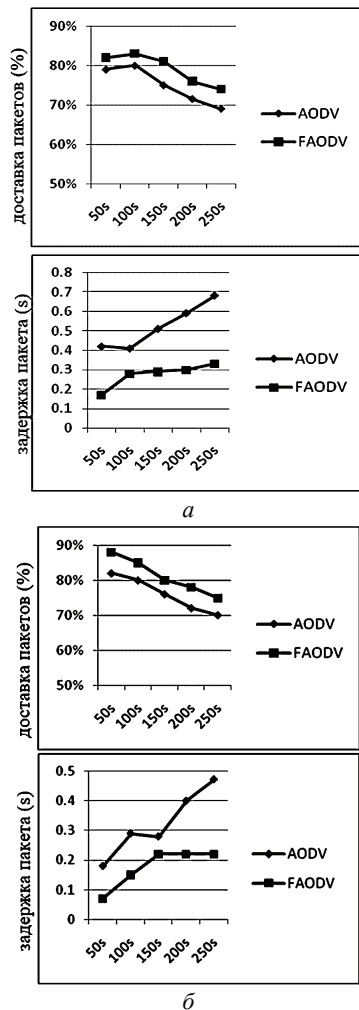


Рисунок 6 – Сравнения эффективности предлагаемого алгоритма маршрутизации с классическим AODV: а – количество узлов – 10; б – количество узлов – 20

**Выводы.** Предложен модифицированный метод маршрутизации для БСС-сети, базирующийся на аппарате нечеткой логики. Основными достоинствами метода являются:

- Возможность поиска множества маршрутов при поступлении запроса приложения установить сессию.
- Возможность учета одновременно 5 параметров состояния узлов и каналов связи при выборе оптимального маршрута из множества найденных.
- Возможность оптимизации разработанного протокола маршрутизации через настраиваемые параметры для заданного сценария работы сети, например, с целью повышения показателей качества обслуживания и непрерывности обслуживания абонентов.

- Повышение отказоустойчивости при сбое на маршруте передачи данных за счет двух резервных маршрутов.

Была разработана общая структура контроллера нечеткой логики, определены входные и выходные параметры. В процессе анализа различных параметров оценки маршрутов был определен набор параметров, которые могут быть использованы на практике: доступная пропускная способность, задержка передачи пакета, загруженность пакетных буферов, джиттер, количество промежуточных узлов.

Проведен сравнительный анализ предложенного метода с классическим AODV.

**Список литературы:** 1. Баер П. Введение в нечеткую логику и системы нечеткого управления / П. Баер, С. Новак, Р. Винклер. Режим доступа: <http://softlab.od.ua/algo/neuro/fazzv-intro/>. 2. Thomas R.W. Cognitive Networks / R.W. Thomas, L.A. DaSilva, A.B. MacKenzie // Proc. IEEE DySPAN 2005, Nov. 2005. – P. 352–360. 3. Siva Ram Murthy C. Ad Hoc Wireless Networks: Architectures and Protocols / C. Siva Ram Murthy, B. S. Manoj // Prentice Hall, 2004. – 880 p. 4. Романюк В. Мобильные радиосети – перспективы беспроводных технологий // Сети телекоммуникации, декабрь 2003. – С. 62–68. 5. Жуков И. А. Обеспечение заданного уровня качества обслуживания в объединенных сетях / И.А. Жуков, И.А. Клименко // Проблемы информатизации та управління: Зб. наук.пр. – К.: НАУ, 2005. – Вип. 13. – С. 5–14. 6. Миночкин А.И. Протоколы маршрутизации в мобильных радиосетях / А.И. Миночкин, В.А. Романюк // Зв'язок. – 2001. – № 1. – С. 31–36. 7. Громовский А.В. Алгоритм маршрутизации EXZRP в интегрированных сетях нового поколения / А.В. Громовский // Комп'ютерні засоби, мережі та системи. – 2007. – № 6. – С. 111–118. 8. Alandjani G. Fuzzy routing in ad hoc networks / G. Alandjani, E. Johnson // Performance, Computing, and Communications Conference: Conference Proceedings of the 2003 IEEE. 9. J. Faedah, K. Sabira Adaptive Modulation for OFDM Systems. – Vol. 1, No 2. – August 2009. 10. Тэрано Т. Прикладные нечеткие системы / Т. Тэрано, К. Асаи, М. Сугэно. – М.: Мир, 1993. – 368 с. 11. Perkins C. Ad hoc On-Demand Distance Vector (AODV) Routing / C. Perkins, E. Belding-Royer // Request for Comments: 3561. – 2003. – Режим доступа: <http://tools.ietf.org/html/rfc3561>.

**Bibliography (transliterated):** 1. P. Baer, S. Novak, R. Vinkler. Baer P. Vvedenie v nechetkuju logiku i sistemy nechetkogo upravlenija. n.d.<<http://softlab.od.ua/algo/neuro/fazzv-intro/>>. 2. R.W. Thomas, L.A. DaSilva, A.B. MacKenzie. Cognitive Networks. Proc. IEEE DySPAN, 2005. 352–360. 3. C. Siva Ram Murthy, B. S. Manoj. Ad Hoc Wireless Networks: Architectures and Protocols. Prentice Hall, 2004. 880. 4. Romanjuk, V. Mobil'nye radioseti – perspektivy besprovodnyh tehnologij. Seti telekommunikacii (2003): 62–68. 5. I.A. Zhukov, I.A. Klimenko. Obespechenie zadannogo urovnja kachestva obsluzhivaniya v ob#edinennyh setjah. Problemy informatizacii ta upravlinnja 13 (2005): 5–14. 6. A.I. Minochkin, V.A. Romanjuk. Protokoly marshrutizacii v mobil'nyh radiosetjah. Zv'jazok 1 (2001): 31–36. 7. A.V., Gromovskij. Algoritm marshrutizacii EXZRP v integrirovannyh setjah novogo pokolenija. Komp'juterni zasobi, merezhi ta sistemi 6 (2007): 111–118. 8. G. Alandjani, E. Johnson. Fuzzy routing in ad hoc networks. Performance, Computing, and Communications Conference. 2003. 9. J.Faegah, K.Sabira. Adaptive Modulation for OFDM Systems. 2. Vol. 1. 2009. 10. T. Tjerano, K. Asai, M. Sugeno. Prikladnye nechetkie sistemy. Moscow: Mir, 1993. 11. C. Perkins, E. Belding-Royer. Ad hoc On-Demand Distance Vector (AODV) Routing. 2003. <<http://tools.ietf.org/html/rfc3561>>.

Поступила (received) 26.02.2016

Відомості про авторів / Сведения об авторах / About the Authors

**Потапчук Ілья Александрович** – студент кафедри «Комп'ютерних та інформаційних технологій» НТУ «ХПІ», тел. (063) 4349001, e-mail: potapchuk.ilya@gmail.com

**Potapchuk Illia Aleksandrovych** – student of "Computer and informational technologies" department of NTU "KhPI", cell. (063) 434 90 01, e-mail: potapchuk.ilya@gmail.com

**Багрянцева Александра Вячеславовна** – студент кафедри «Комп'ютерних та інформаційних технологій» НТУ «ХПІ», тел. (093) 973 25 73, e-mail: bagryantseva.sasha@gmail.com

**Bagryantseva Aleksandra Vyacheslavovna** – student of "Computer and informational technologies" department of NTU "KhPI", cell. (093) 9732573, e-mail: bagryantseva.sasha@gmail.com