

Адаптивная нейро-нечеткая система маршрутизации данных в сетях индустриального интернета вещей

К. Мукешимана

Санкт-Петербургский государственный электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)
e-mail: mucam2@mail.ru

Аннотация. В целях повышения продуктивности предприниматель активно внедряют решения интернета вещей на предприятиях – Индустриальный Интернет вещей (IIoT). IIoT-устройства могут генерировать большие объемы трафика - вплоть до одного петабайта в сутки и обработка и передача информации должны осуществляться в режиме реального времени с минимальной задержкой. К 2025 году должно быть подключено более 500 миллионов разнообразных предметов производства и логистики, находящихся в одной сети. Для управления таким сложным техническим объектом применяются сложные многоуровневые алгоритмы, которые базируются на серьезном математическом фундаменте. В этой работе предлагается система управления, основанная на новой и перспективную вычислительную технологию – нейронная сеть, базирующаяся на применении нечеткой логики для оптимального взаимодействия в IIoT-сетях и для интеллектуального управления маршрутизацией данных. Предложенный алгоритм выбирает оптимальный маршрут передачи пакетов с учетом качества обслуживания (QoS) и других необходимых параметров

Ключевые слова: *нейрокомпьютерная система; нечеткая логика; маршрутизация; качество обслуживания; Интернет вещей*

I. ВВЕДЕНИЕ

В настоящее время промышленный Интернет вещей (IIoT) превратился в многообещающую парадигму будущей сети, которая намеревается соединить различные виды интеллектуальных объектов, включая домашнее оборудование, приводные устройства, встроенные и внешние датчики, программное обеспечение, исполнительные механизмы и т.д., между которых устанавливается сетевое подключение. Несмотря на то, что проблема организации взаимодействия устройств IIoT уже привлекла большое внимание многих исследователей, текущие исследования маршрутизации в IIoT не могут эффективно решить проблему обмена данными самоадаптирующимся и самоорганизованным образом, потому что количество подключенных устройств совершенно большое [1]. В этой части диссертации предлагается новый механизм маршрутизации,

основанный на использовании теории нечеткой логики и нейронной сети [2].

Маршрутизация пакетов в сетях интернета вещей от источника к получателю стала критической, и для определения наилучших усилий по обслуживанию в сети используются дополнительные механизмы управления трафиком. Эти подходы позволяют уменьшить задержку канала передачи и скорость потери пакетов, а также выбирать оптимального маршрута для передачи данных [3].

Метод, основанный на нейро-нечеткой логике, является новым подходом для интеллектуальной маршрутизации. Предложенный алгоритм выбирает оптимальный маршрут пакета на основе качества обслуживания (quality of service QoS) и скорости передачи пакетов

II. МЕТОДОЛОГИЯ ИССЛЕДОВАНИЯ

Нейро-нечеткая модель, используемая в этой работе для разработки интеллектуального алгоритма маршрутизации, основана на адаптивной системе нейро-нечеткого вывода (Adaptive Neuro-fuzzy Inference System ANFIS) с использованием системы нечеткого вывода Такаги-Сугено-Канга (TSK), которая была предложена в 1900-х годах [4].

Этот метод во многом похож на метод Мамдани. Первые две части процесса нечеткого вывода, Фаззификация входных переменных и применения нечеткого оператора, одинаковы. Основное различие между ними состоит в том, что выходные функции принадлежности TSK являются либо линейными, либо постоянными [5]. Типичное правило в нечеткой модели Сугено имеет вид:

Правило 1: Если $x = A_1$ и $y = B_1$, то $f_1 = p_1x + q_1y + r_1$,

Правило 2: Если $x = A_2$ и $y = B_2$, то $f_2 = p_2x + q_2y + r_2$.

На этапе оптимизации нечетких правил, параметры исходного нечеткого множества уточняются с помощью нейросетевых методов. Здесь правила, полученные на этапе генерации правил, используются для построения нейронной сети с пятью слоями [2, 6]. Каждый слой построенной системы определяет определенный шаг в

механизме нейро-нечеткого вывода. Этот механизм вывода указывается на рис. 1.

Слой 1 (Входной слой) представляет входные переменные, которые представляют текущее состояние системы. Нейроны в этом слое передают эти входные переменные непосредственно своим соответствующим модулям членства в следующем слое. Выходы данного слоя представляют собой значения функций принадлежности (ФП) при определенных значениях входов. Каждый узел является адаптивным с ФП $\mu_{A_i}(x)$, где x – вход i -го узла, $i = 1, \dots, n$; A_i – ассоциированная с данным узлом лингвистическая нечёткая переменная [7].

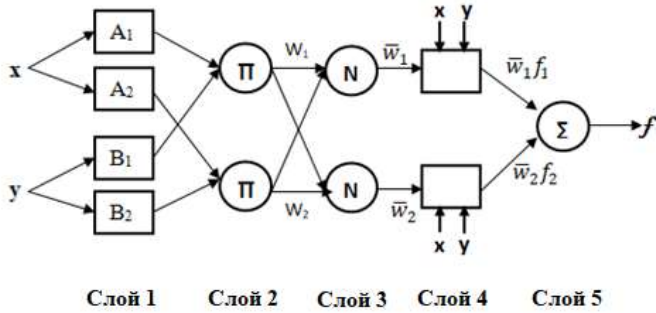


Рис. 1. Структура нейро-нечеткой сети ANFIS

Обычно применяются гауссовы функции принадлежности вида [3]:

$$A_{ij} = \exp \left[\frac{1}{2} \left(\frac{x - a_{ij}}{b_{ij}} \right)^2 \right] \quad (1)$$

$$B_{ij} = \exp \left[\frac{1}{2} \left(\frac{y - a_{ij}}{b_{ij}} \right)^2 \right] \quad (2)$$

где a_{ij} и b_{ij} – множество параметров, которые необходимо настраивать в процессе обучения и, как следствие, проявляет различные формы функций принадлежности.

Нейроны слоя 2 (слой фазификации) преобразуют скалярные значения, полученные из первого слоя, в соответствующие им нечеткие значения. Короче говоря, нейроны этого слоя содержат определения функций принадлежности и передают эти входные функции принадлежности слою нечетких правил.

$$w_i = A_i(x) \times B_i(y), \text{ для } i = 1, 2. \quad (3)$$

Слоя 3 называется ещё слоем нечетких правил и является фиксированным. Он вычисляет нормированные уровни активации правил [4].

$$\bar{w}_i = \frac{w_i}{\sum_{i=1}^n w_i}, \text{ для } i = 1, 2. \quad (4)$$

В слое 4 находятся адаптивные узлы, и каждый из них представляет собой произведение нормализованного веса из предыдущего слоя и полинома первого порядка (для

модели Сугено первого порядка), как показано в уравнении 5, где $\{p_i, q_i \text{ и } r_i\}$ являются последовательными или расчетными параметрами, используемые для обработки второй части нечеткого правила.

$$\bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i), \quad i = 1, 2. \quad (5)$$

Суммирующая операция выполняется в слое 5. Вычисляется результат нечёткого вывода по уравнению 6 в виде суммы всех аргументов.

$$\begin{aligned} f &= \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2 \\ &= \bar{w}_1 f_1 + \bar{w}_2 f_2 \\ &= (\bar{w}_1 x) p_1 + (\bar{w}_1 y) q_1 + (\bar{w}_1) r_1 + (\bar{w}_2 x) p_2 + (\bar{w}_2 y) q_2 + (\bar{w}_2) r_2. \end{aligned} \quad (6)$$

А. Нечеткое логическое моделирование

Фазификация входных данных осуществляется с использованием функций принадлежности, определенных выражениями 7 и 8.

$$Route = \begin{cases} 0, & \text{if } BS < MS \\ 1, & \text{if } BS \geq MS \end{cases} \quad (7)$$

где **Route** – достоверность маршрутизации сообщения (MS) по определенному пути в зависимости от доступного размера буфера (BS) соседних узлов.

$$Route = \begin{cases} 1, & \text{if } Dly_{in} < Dly_{jn} \\ 0, & \text{if } Dly_{in} \geq Dly_{jn} \end{cases} \quad (8)$$

где **Dly_{in}** – это задержка на определенном пути между двумя соседними узлами.

Функция принадлежности входного параметра задержки может быть {маленькая, средняя, большая}, а функция размера буфера {маленькая, адекватная}. Нечеткие правила вывода Базовые правила вывода типа «если – то», в английской нотации «if – then», также называемые нечеткой импликацией, принимают форму: если x это A , то y это B , где A, B – лингвистические значения, определенные нечетким способом через соответствующие функции принадлежности для переменных x и y [3, 4].

Пример одного из правил ЕСЛИ ... ТО нашей работы приведен ниже:

Если “ Dly ” это маленькая и BS это адекватная, то “ $Route$ ” где Dly – задержка, BS – доступный размер буфера; $Route$ – маршрут.

Система нечеткого вывода была адаптирована к комбинации трех входных переменных (задержка, размер сообщения и размер буфера) в соответствии с моделированием, проводимым в этом исследовании.

В. Архитектура нейро-нечеткой модели

Архитектура нейро-нечеткой модели показана на рис. 2. Основными частями архитектуры являются входные данные, база знаний, нейро-нечеткая система

вывода и система поддержки принятия решений. База знаний используется для хранения структурированных и неструктурированных данных, которые система использует при принятии решений. Он состоит из нейронной сети, нечеткой системы и базы данных. Нейро-нечеткая система вывода формирует отображение из заданных входных данных с использованием алгоритма для обеспечения шаблона / основы для принятия решений, в то время как система поддержки принятия решений принимает решения на основе шаблонов, генерируемых из вывода системы [5].

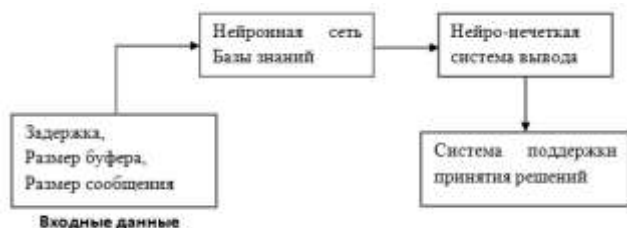


Рис. 2. Архитектура нейро-нечёткой системы

В этой модели для управления системой нечеткого вывода использовалась многослойная нейронная сеть, как показано на рис. 3. Входными параметрами в нейронную сеть являются задержка и размер буфера.



Рис. 3. Нейро-нечеткая модельная диаграмма

Разработанная пятиуровневая ANFIS модель, имеет слой 1, который вычисляет степень, в которой каждая из лингвистических входных переменных, подаваемых в сеть, соответствует функциям принадлежности, содержащимся в нечетких множествах. Входные переменные описываются следующим образом:

- задержка $[D] = \{\text{МАЛЕНЬКАЯ, СРЕДНЯЯ, БОЛЬШАЯ}\};$
- размер сообщения $[M] = \{\text{МАЛЕНЬКИЙ, СРЕДНИЙ, БОЛЬШОЙ}\};$
- размер буфера $[BS] = \{\text{МАЛЕНЬКИЙ, СРЕДНИЙ, БОЛЬШОЙ}\}.$

Ряд нечетких правил вывода, которые генерируются, для системы в слое 2 имеют общий вид:

IF $(D_1 \text{ is } d_{11}) \text{ AND } \dots \text{AND } (D_n \text{ is } d_{1n}) \text{ AND } (BS_1 \text{ is } bs_{11}) \text{ AND } \dots \text{AND } (BS_n \text{ is } bs_{1n}) \text{ AND } (M \text{ is } m_1) \text{ THEN } Z_1 = (P_{11}D_1) + \dots + (P_{1n}D_n) + \dots + (Q_{11}BS_1) + \dots + (Q_{1n}BS_n) + (R_1M) + S_1$

Общее количество правил рассчитывается по (9).

$$N = Mf^{Inputs} \quad (9)$$

где N – количество нечетких правил, Mf – количество функций принадлежности на вход и $Inputs$ – количество входов. В данной работе разработана модель с тремя функциями принадлежности и пятью выходными переменными. Используя выражение (9) мы получаем количество нечетких правил $N = 3^5 = 243$ правила.

В слое 3 нормализуется вес каждого ввода путем деления индивидуального значения веса на сумму всех других значений. Значения на выходе слоя 4 задается уравнением (10), и на выходе слоя 5 получается результат моделирования путём суммирования всех выходов из слоя 4 [9]

$$\overline{w_i} f_i = \overline{w_i} (P_{i1}D_1) + \dots + (P_{in}D_n) + \dots + (Q_{i1}BS_1) + \dots + (Q_{in}BS_n) + (R_iM) + S_i. \quad (10)$$

где P, Q, R и S являются последовательными параметрами сети.

В работе для обучения нейронной сети используется алгоритм обратного распространения ошибки (backpropagation) с применением рекуррентного метода наименьших квадратов (recursive least squares – RLS).

Реализация данного проекта осуществлялась в среде MATLAB R2019a с использованием нейро-нечеткого конструктора и нечеткого инструмента Fuzzy toolbox. Данные обучения и тестирования загружаются в конструктор из файла Excel. Блок-схема работы системы показан на рис. 4. Он содержит процессы, участвующие в интеллектуальной системе маршрутизации данных в IoT сети. Нейро-нечеткой конструктор вызывается командой *anfisedit* в окне командной строки [8].

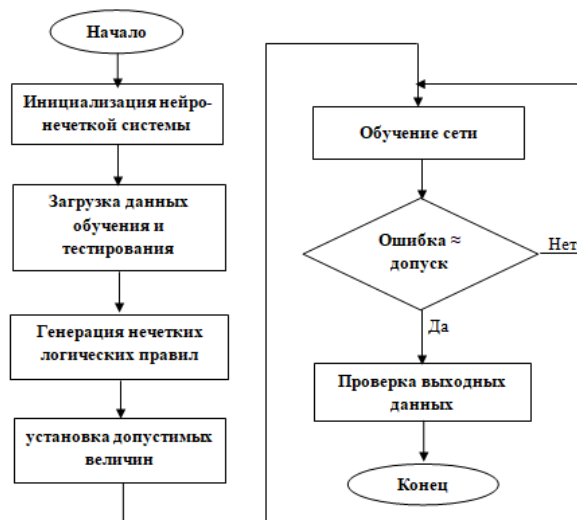


Рис. 4. Блок-схема интеллектуальной системы маршрутизации

На рис. 5 показан нейро-нечеткий конструктор после загрузки данных обучения и тестирования.

Нечеткие правила устанавливаются из интерфейса конструктора. Количество функций принадлежности, а

также их тип и форма задаются с помощью нейро-нечеткого конструктора [10].

Соответствующая нейро-нечеткая структура системы генерируется конструктором, показывающим входные слои и скрытые слои, а также правила. На рис. 6 показана нейро-нечеткая структура системы, приводящая к нечетким правилам системы.

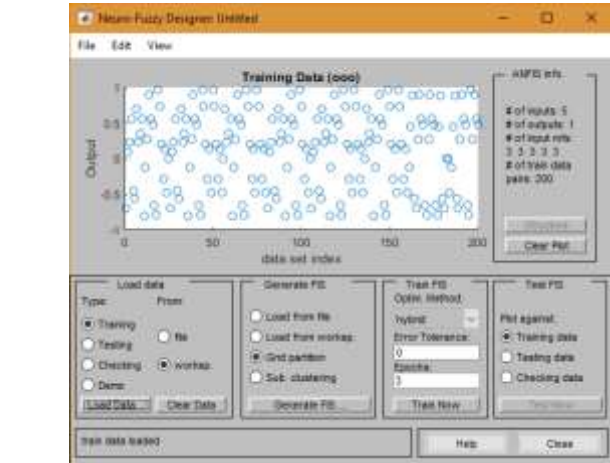


Рис. 5. Редактор нейро-нечеткой сети

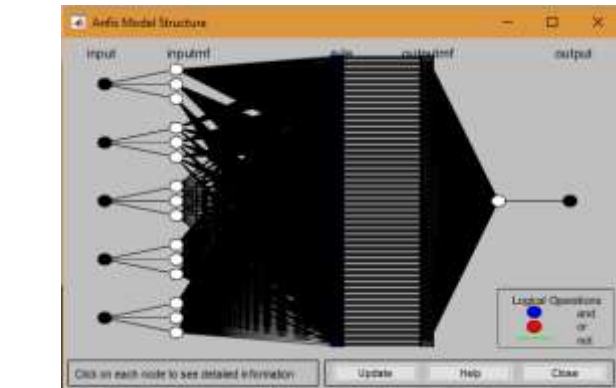


Рис. 6. Генерируемая нейро-нечеткая структура системы

III. АНАЛИЗ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЯ

Для проведения аналитического анализа использовался симулятор сети передачи данных, выпускаемый фирмой Cisco Systems Cisco Packet Tracer для запуска моделирования OSPF протокола на предлагаемую архитектуру сети. После настройки сети в packet tracer, необходимые конфигурации были выполнены на каждом из сетевых устройств.

Значения полученных результатов приведены в таблице, а их графическое сравнение на рис. 7 и 8. Значения задержки и пропускной способности.

ТАБЛИЦА 1 значения задержки и попускной способности

Размер сообщения (байт)	Задержка (с)		Скорость передачи (Кбит /с)	
	ANFIS	OSPF	ANFIS	OSPF
1000	0.051	0.078	23.555	14.438
2850	0.061	0.079	52.423	37.58
3212	0.078	0.080	57.344	55.5
3784	0.061	0.074	73.523	65.638
6156	0.194	0.181	87.449	79.257
7000	0.774	0.883	91.341	12.083
9962	1.045	1.422	11.024	9.932
12530	1.17	2.101	12.185	4.915
13969	1.25	2.47	12.8	6.144
15200	1.281	2.451	11.878	6.656
16035	1.299	2.5	11.47	6.348
18205	1.4	2.58	10.035	4.096

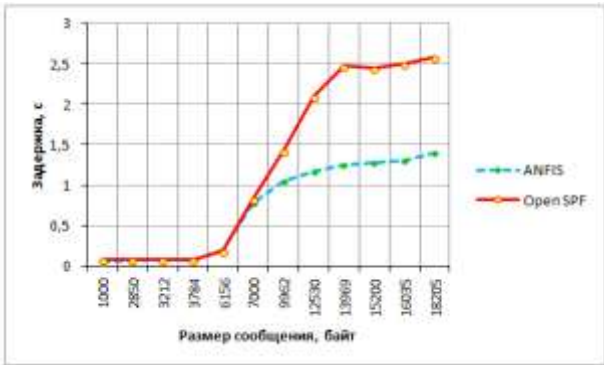


Рис. 7. Графически сравнения значений задержки

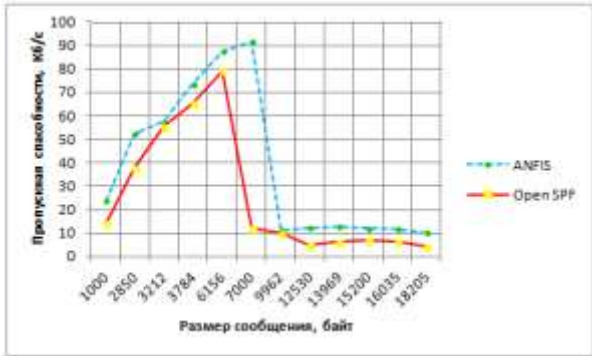


Рис. 8. Графически сравнения значений пропускной способности

IV. ЗАКЛЮЧЕНИЕ

Результаты показывают, что разработанная нейро-нечеткая модель предоставляет лучшую производительность. С увеличением размера передаваемого сообщения динамическая маршрутизация по протоколу OSPF, использующему алгоритм Дейкстры испытывает большую задержку. Так же нейро-нечеткий алгоритм имеет более высокую пропускную способность.

При проектировании сети в качестве входных параметров были выбраны задержка и размер буфера в качестве входных параметров, поскольку наличие этих факторов в любой компьютерной сети непосредственно влияет на качество обслуживания. Используемый, в работе, подход заключается в использовании нейросетевых структур и нечеткой логики для управления сложными техническими и динамическими объектами, способных функционировать в условиях неопределенности математического описания объекта управления.

Для дальнейшей исследовательской работы нейро-нечеткий алгоритм, разработанный в этом проекте, будет усовершенствован путем объединения его с другими существующими методами искусственного интеллекта, такими как генетические алгоритмы и глубокое обучение, с целью повышения эффективности взаимодействия устройств в сетях индустриального интернета вещей.

СПИСОК ЛИТЕРАТУРЫ

[1] Емельянов С.В. Информационные технологии и вычислительные системы: Интернет-технологии. Математическое моделирование. Системы управления. Компьютерная графика. М.: Ленанд, 2012. 96 с.

[2] Комарцова Л.Г., Максимов А.В. Нейрокомпьютеры: Учеб. пособие для вузов / 2ое изд, перераб. и доп. М.: изд-во МГТУ им. Н.Э. Баумана 2004. 400.с.

[3] Синюк В.Г., Бакшеева Е.Л. Гибкие нейро-нечеткие системы вывода и программная реализация для решения задач аппроксимации // Молодой ученый. 2014. №8. С. 108-112.

[4] Смолин Д.В. Введение в искусственный интеллект: клнспект лекций. М.: ФИЗМАТЛИТ, 2004. 208с.

[5] Shihg J. ANFIS: Adaptive-network based fuzzy inference System / J. Shihg, R. Jang // IEE Transactions on SMC. 1993. Vol. 23. P. 665–685.

[6] A. Adebare, O. Folorunso and O. Usman, "Developing a Neuro-fuzzy Model for Determining Shortest Routing Path in a Computer Network," THE JOURNAL OF COMPUTER SCIENCE AND ITS APPLICATIONS, vol. 23, no. 2, p. 13, 2016.

[7] Пегат А. Нечеткое моделирование и управление: пер. с англ. 2-е изд. М.:БИНОМ.Лаборатория знаний, 2017. 798с.

[8] MathWorks — MATLAB and Simulink for Technical Computing, Fuzzy Logic Toolbox, Documentation Center. – Режим доступа: <http://www.mathworks.com/help/fuzzy/index.html> (дата обращения 22.08.2019).

[9] Коваленко Т.А. Интегрированная система маршрутизации в компьютерных сетях. LAP LAMBER Academic Publishing GmbH & Co.KG. 2012. 153 с

[10] W. Guo and W. Zhang, "A survey on intelligent routing protocols in wireless sensor networks," Journal of Network and Computer Applications, vol. 38, pp. 185-201, 2//2014.