

Технологии информационного общества

ОБЛАЧНЫЕ, ТУМАННЫЕ И РОСИСТЫЕ ВЫЧИСЛЕНИЯ В НЕЧЕТКИХ СЕНСОРНЫХ СИСТЕМАХ

Статья рекомендована к публикации членом редакционного совета А.Н. Райковым 01.04.2021.

Горячев Виктор Андреевич

Аспирант
Университет «Дубна»
Дубна, Россия
gva.asp@uni-dubna.ru

Аверкин Алексей Николаевич

Кандидат физико-математических наук, доцент
Федеральный исследовательский центр «Информатика и управление» Российской академии наук
Российский экономический университет им. Г.В. Плеханова
Москва, Россия
averkin2003@inbox.ru

Аннотация

Исследовательская деятельность авторов заключается в интеллектуализации систем мониторинга на основе беспроводных сенсоров с применением теории нечеткой логики. Конкретно в данной статье приводится пример программирования мягкого нечеткого сенсора на основе нечеткой меры (см. раздел 1). Кроме того, в этой работе рассмотрены модели распределенных вычислений в общем случае и в контексте систем мониторинга на основе беспроводных сенсорных сетей (см. разделы 2-4). По результатам изучения трёх наиболее перспективных моделей распределенных вычислений (cloud computing, fog computing, dew computing) описаны их преимущества и недостатки относительно маломощных систем мониторинга, а также сформулированы принципы оптимальной модели распределенных вычислений для обеспечения эффективной работы нечетких сенсорных систем мониторинга.

Ключевые слова

Облачные вычисления; туманные вычисления; росистые вычисления; мягкие сенсоры; нечеткие сенсоры; нечеткая мера; нечеткая логика

Введение

Системы мониторинга технических объектов на основе беспроводных сенсорных сетей предъявляют высокие требования к энергоэффективности, стабильности и защищенности потоков данных. Согласно представлениям авторов, интеллектуализация сенсорных узлов на основе средств нечеткой логики позволит существенно снизить энергопотребление внутри таких систем мониторинга, а также обеспечит более стабильную и защищенную передачу информации при условии применения соответствующей модели передачи, обработки и хранения данных.

Облачные, туманные и росистые вычисления – это три вида моделей передачи, обработки и хранения данных. В настоящее время большинство систем мониторинга работают на модели облачных вычислений. Данная модель обладает рядом недостатков (увеличенное энергопотребление, неконтролируемые сбои в передаче данных, возможность перехвата данных, регулярные финансовые затраты). В данной работе рассматриваются туманные и росистые модели распределенных вычислений, применение которых способствует исправлению или сглаживанию данных недостатков внутри маломощных, распределенных систем мониторинга.

© Горячев В.А, Аверкин А.Н, 2021. Производство и хостинг журнала «Информационное общество» осуществляется Институтом развития информационного общества.

Данная статья распространяется на условиях международной лицензии Creative Commons «Атрибуция — Некоммерческое использование — На тех же условиях» Всемирная 4.0 (Creative Commons Attribution – NonCommercial - ShareAlike 4.0 International; CC BY-NC-SA 4.0). См. <https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode.ru>

По авторскому мнению, комплексное применение средств нечеткой логики и росистых вычислений способно вывести системы мониторинга на основе беспроводных сенсоров на новый уровень энергоэффективности, адаптивности, стабильности и взломоустойчивости.

Статья организована следующим образом. Первый раздел содержит пример программирования нечеткого сенсора при помощи средств нечеткой логики, в частности, на основе нечетких мер. В последующих трех разделах описана работа модели облачных, туманных и росистых вычислений в общем случае и в контексте систем мониторинга на основе сенсорных сетей.

1 Нечеткие сенсоры

Нечеткие сенсоры (fuzzy sensors) – сенсоры, вычислительная работа которых основана на использовании методов нечеткой логики, что дает возможности для интеллектуализации беспроводных сенсорных сетей [Аверкин, 2017, с.1].

Мягкие сенсоры (soft sensors) – это датчики, способные проводить косвенные измерения; неизвестная величина определяется в них посредством измерения других величин. Например, определение степени помола целлюлозы, влагосодержания пищевых продуктов и т.п. [Херролд, 2001, с. 36].

В этой статье мы не будем рассматривать пример работы нечеткого сенсора на основе нечеткой искусственной нейронной сети, это будет темой для следующей статьи. В данной статье мы подробно рассмотрим пример функционирования нечеткого сенсора на основе нечеткой меры и принципа мягкого сенсора.

Предположим, что перед нашим нечетким сенсором стоит задача определения уровня комфорта в студенческой аудитории. Добиться решения задачи позволит загруженный в данный сенсор программный код на основе нечетко-логической системы вывода с использованием нечеткой меры [Емельянов, 2014, с. 19]. Так, для измеряемой датчиками температуры (например, от -10С до +35С) и влажности (например, от 20% до 80%) назначаются веса (от 0 - плохо до 1 - отлично). При этом в базе данных (БД) каждого датчика создается MySQL таблица соответствия параметра и меры (например, 0С = 0.1, 10С = 0.5). Для датчика температуры своя таблица (см. табл. 1), для датчика влажности своя таблица (см. табл. 2).

Таблица 1. Данные для датчика температуры.

	Т (Температура)	WT (Вес)
1.	-10	0
2.	0	0.1
3.	10 (ячейка T3 в примере)	0.5 (ячейка WT3 в примере)
4.	20 (ячейка T4 в примере)	1 (ячейка WT4 в примере)
5.	25	0.7
6.	35	0

Таблица 2. Данные для датчика влажности.

	Н (Влажность)	WH (Вес)
1.	10	0
2.	20	0.4
3.	30	0.8
4.	40	1
5.	50	0.8
6.	70	0.3
7.	90	0

Датчики должны уметь работать с данными из этих таблиц, поэтому в программном коде для каждого датчика прописываются свои правила поведения. Например, при фиксации датчиком температуры показателя +13С запустится функция комфорта, см. рис. 1.

```

1  <?php
2  // Исходные данные
3  $temp = 13; // температура, поступившая с датчика
4  $row = 6; // количество строк в таблице датчика температуры
5  // Запускается функция определения уровня комфорта
6  function comfort ($temp) {
7      for ($i=1; $i<=$row; $i++) {
8          // цикл будет продолжаться, пока не найдется такой диапазон температур (от $T_min до
           $T_max), в который входит +13С; в данном случае цикл остановится через 2 итерации на
           значениях ячеек T3 и T4 (диапазон от +10С до +20С).
9          if ( $temp>=$T_min && $temp<=$T_max ) { break;
10         // определяются $WT_min и $WT_max, равные ячейкам WT3 и WT4.
11         }}
12 // Определяем вес для температуры +13С, чтобы затем сопоставить с пороговым значением.
13 $WT = $WT_min + (($WT_max - $WT_min) * (($temp - $T_min) / ($T_max - $T_min)));

```

Рис. 1. Пример функции комфорта в обобщенном виде на PHP

В процессе исполнения кода получаем:

$$WT = 0.5 + ((1 - 0.5) * ((13 - 10) / (20 - 10))),$$

$$WT = 0.65.$$

Аналогичным образом стоит получить вес для показателя влажности. Например, при показаниях влажности 26% вес (\$WH) будет равен 0.64.

После определения весов необходимо проверить, пройдут ли они порог (\$Step), установленный экспертом. Например, мы считаем, что порог свыше 0.75 будет говорить об оптимальном уровне комфорта в аудитории. Если в результате решения функции мы получим значение 0.76 – в аудитории комфортно, если значение 0.65, в аудитории некомфортно и требуется включить осушитель/парогенератор и кондиционер, см. рис. 2.

```

27 $Step = 0.75;
28 if ($WT > $Step && $WH > $Step) { /* запуск проверки комфорта через некоторое время */ }
29 else { /* запуск исполнительных устройств */ }

```

Рис. 2. Пример порога в функции комфорта

В нашем примере, когда температура в аудитории +13С, а влажность 26% нечеткий сенсор отправит указание исполнительным устройствам: запустится кондиционер и парогенератор.

Итак, в описанном примере при проектировании базы данных использовалась нечеткая мера, а в программном коде нечеткая система вывода. Это, в свою очередь, позволило сократить объем памяти необходимой для хранения информации, что необходимо при работе с маломощными сенсорными устройствами.

2 Облачные вычисления

Прообразами современной концепции облачных вычислений (cloud computing) были коммунальные вычисления (utility computing) – 60е годы, а также сеточные вычисления (grid computing) – 90е годы. Развитию современных облачных было положено в 2006 году компанией Amazon. Именно в этом году компания запустила Elastic Compute Cloud: коммерческий веб-сервер, позволяющий пользователям арендовать ИТ-инфраструктуру для запуска своих приложений [Кирсанова, 2020, с. 37].

Хорошее определение облачных вычислений дает С.В. Прокопчина, говоря о том, что в соответствии с облачной концепцией пользователи в обмен на сформированные запросы и поставленную информацию, получают готовые решения. При этом вся измерительно-аналитическая деятельность производится облачными сервисами [Прокопчина, 2019, с. 6].

Отличительной особенностью облачных вычислений являются высокопроизводительные удаленно сосредоточенные вычислительные службы (сервера, базы данных, программное

обеспечение и т.п.), к которым пользователи могут подключиться через интернет. При наличии соединения между облаком и пользовательскими устройствами возникает двунаправленная связь: устройство отправляет телеметрические и пользовательские данные в облако, облако отправляет устройству команды/уведомления и результаты обработки пользовательских данных. Таким образом, облачные вычисления расширяют функциональность и производительность пользовательских устройств, а также позволяют множеству пользователей вне зависимости от их местоположения работать с данными одновременно.

Примером облачных вычислительных служб является Yandex Compute Cloud - вычислительные мощности для разработки IT-проектов, Azure Digital Twins - вычислительные мощности для создания цифровых моделей сложных физических систем, Google Drive - хранилище для пользовательских файлов.

В настоящее время сенсорные системы мониторинга, работающие на облачных технологиях, широко распространены, благодаря своей простоте и доступной цене. Такие платформы, как Apple HomeKit, Google Home и «Умный дом Яндекса» не требуют наличия шлюза (блок управления, центр управления). Поэтому данные, собираемые сенсорами таких систем, передаются с сенсоров через маршрутизатор (роутер) напрямую в облако, где обрабатываются при помощи библиотеки функций. Если пользователь желает оперировать данными, то он должен подключиться к облаку через интернет. У такой сенсорной сети есть существенный недостаток, при потере интернет соединения она теряет почти всю полезную функциональность.

3 Туманные вычисления

Концепция туманных вычислений была описана еще в 2012 году в статье Джонатана Нумхаузера [Numhauser, 2012]. «Информационный» туман, подобно природному собрату, состоит из частиц. Каждая частица «информационного» тумана - устройство с достаточными вычислительными возможностями, подключенное к сети Интернет. Примерами таких устройств могут служить (блоки управления, компьютеры с необходимым ПО, роутеры). Подобные устройства, разбросанные по миру, как туман окутывают нашу планету.

Отличительная особенность туманной концепции заключается в том, что вычислительные операции выполняются не только в удаленном вычислительном центре - облаке, но и на устройствах туманного уровня, расположенных между облаком и конечными устройствами. Другими словами, главное отличие между облачной и туманной концепциями можно выразить в том, что облачные вычисления производятся централизованно, а туманные в некоторой степени децентрализованно: на туманных устройствах производятся потоковые вычисления (в режиме реального времени, в режиме без доступа в интернет), а облако выступает средством для решения ресурсоёмких задач, например, глобального анализа и долгосрочного хранения данных [Wongpi, 2012].

На данный момент примеры реальной работы туманных вычислений в коммерческих проектах компания Foghorn. Кроме того, существует несколько публичных платформ, например, Microsoft предлагает платформу Azure IoT, Amazon вывела на рынок AWS IoT Greengrass.

Кроме коммерческих, существует ряд платформ с открытым исходным кодом, например, FogFlow, FogFrame2.0 или FogBus, которые позволяют реализовывать функции туманной инфраструктуры, реализовывать эвристические алгоритмы, разрабатывать приложения туманных вычислений [Кирсанова, 2020, с. 52].

Сенсорные системы мониторинга, построенные по туманной концепции, подразумевают наличие в своей архитектуре трёх уровней: пользовательский уровень и два вычислительных уровня - туман и облако [Halwai, 2019]. Пользовательский уровень отвечает за сбор данных и включает в себя различные сенсоры. Вычислительные уровни принимают сгенерированные на пользовательском уровне данные, которые прежде обрабатываются на туманном уровне (роутер, центр управления, компьютер), затем на облачном уровне (вычислительные службы, хранилище). Важно отметить, что при такой архитектуре с одной стороны устройства туманного уровня общаются с облаком по интернет соединению, а с противоположной стороны общаются с сенсорами по протоколам беспроводной передачи данных (Wi-Fi, BLE, ZigBee, Z-Wave и т.п.). В штатном режиме (стабильное подключение к облаку через интернет) устройства туманного уровня полезны тем, что передают в облако не «сырые», а прошедшие обработку (структурированные, отфильтрованные, адаптированные) телеметрические данные, что повышает эффективность

работы облачных служб [Rahman, 2018]. При нештатной ситуации (отсутствие связи с облаком) устройства туманного уровня полезны тем, что организуют текущую работу конечных устройств в рамках своей компетенции. Примером устройства туманного уровня может быть центр управления Ps-Link ZGW01, который общается с облаком через Интернет, но при обрыве сети продолжает поддерживать работу сенсорной сети по протоколу ZigBee.

4 Росистые вычисления

Первое упоминание о росистых вычислениях можно обнаружить в научной статье 2015 года [Wang, 2015]. Потенциал вычислений такого рода огромен, учитывая то, что в 2017 году к Интернету во всем было подключено 20,35 млрд. устройств [Hong, 2017].

Наиболее описательное определение росистых вычислений, на наш взгляд, приводит Yingwei Wang [Wang, 2016, с. 2]. Исходя из определения, можно выделить две основные идеи, которые отличают росистые вычисления. Первая отличительная особенность росистых вычислений заключается в использовании вычислительных ресурсов, расположенных на самом краю распределенной сети (Wang говорит про локальные компьютеры, но мы считаем, что стоит расширить это понятие до устройств, обладающих хотя бы минимальными вычислительными возможностями: сенсоры, розетки, часы и камеры с процессором). Вторая особенность концепции росистых вычислений заключается в максимально возможной обработке данных непосредственно на конечных узлах сети перед передачей на облачный уровень. Например, кооперируясь, «умные» сенсоры температуры, влажности и освещения, могут мгновенно определить текущее состояние окружающей среды и отдать управляющий сигнал на исполнительное устройство (парогенератор, водяную помпу и т.п.). Стоит отметить, что при этом полноценный анализ, а также хранение данных, собранных за время работы системы, осуществляется в облаке. Именно в облаке выстраиваются модели сбоев и вырабатываются меры по улучшению системы [Фаталиев, 2019, с. 20].

К наиболее известному из существующих приложений с возможностями росистых вычислений можно отнести Dropbox, который позволяет работать с файлами и папками даже при отсутствии связи с облачным хранилищем.

В настоящее время сенсорные системы мониторинга, работающие по принципам росистой концепции, нам неизвестны. Теоретически такие системы должны состоять из облака, роутера и сенсорных узлов. При этом сенсорные узлы даже при отсутствии подключения к облаку или при выходе шлюза из строя должны быть способны решать стоящие перед ними задачи без существенной потери функциональности. Кроме того, сенсорные узлы должны быть способны передавать обработанную информацию пользователю даже при отсутствии интернет соединения. В-третьих, сенсорные должны иметь возможность самостоятельно проводить интеллектуальные вычислительные операции без привлечения вычислительных возможностей IoT шлюза. Первым шагом для выполнения поставленных условий является внедрение в корпус сенсорных узлов приемо-передатчиков, работающих по стандартам связи IEEE 802.15.4 или IEEE 802.15.1. Вторым важным шагом, сопутствующим выполнению заявленных условий, будет создание нечетких сенсоров на основе мягких нейро-нечетких сетей, которые будут иметь возможности искусственных нейронных сетей (адаптация, обучение, предсказание) и возможности мягких сенсоров (определение неизвестного параметра через измеряемые величины).

Заключение

Для создания энергоэффективных и адаптивных сенсорных систем мониторинга требуется уменьшение объема вычислительных операций, уменьшение объема затрачиваемой памяти и построение гибкой вычислительной среды на стороне сенсорных узлов. С этими задачами позволяет справиться интеллектуализация сенсорных устройств на основе принципов нейронных сетей и нечеткой логики.

Для обеспечения автономной, стабильной и взломоустойчивой работы сенсорных систем мониторинга требуется широкое распределение вычислительных возможностей внутри сети с возможностью распределенного хранения важной информации. Для достижения таких результатов необходимо разрабатывать системы мониторинга сообразно модели росистых вычислений.

Именно комплексное применение средств нечеткой логики и речистых вычислений способно вывести системы мониторинга на основе беспроводных сенсоров на новый уровень энергоэффективности, адаптивности, стабильности и взломоустойчивости.

Литература

1. Аверкин А.Н., Лавров Г.К. Использование технологии fuzzy smart sensors wsn для интеллектуализации систем поддержки принятия решений в беспроводных сенсорных сетях // Системный анализ в науке и образовании. 2017. №1. С. 1-14.
2. Емельянов С.Г., Титов В.С., Бобырь М.В. Интеллектуальные системы на основе нечеткой логики и мягких арифметических операций: учеб. пособие. М.: АРГАМАК-МЕДИА, 2014. 341 с.
3. Кирсанова А.А., Радченко Г.И., Черных А.Н. Обзор технологий организации туманных вычислений // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Том 9. №3. С. 35-63.
4. Прокопчина С.В. Байесовские интеллектуальные технологии: методология и применение в задачах цифровизации // Международная конференция по мягким вычислениям и измерениям. 2019. Том 1. С. 3-6.
5. Фаталиев Т.Х., Мехтиев Ш.А. Децентрализация обработки данных в среде электронной науки // Информационные технологии. Проблемы и решения. 2019. №4(9). С. 17-22.
6. Херролд Д. "Мягкие" датчики - новейший инструмент промышленной автоматизации // Датчики и системы. 2001. №12. С. 36-41.
7. Bonomi F., Milito R., Zhu J., Addepalli S. Fog computing and its role in the internet of things // Proceedings of the first edition of the MCC workshop on Mobile cloud computing. 2012. P. 13-16.
8. Hong H. From Cloud Computing to Fog Computing: Unleash the Power of Edge and End Devices // IEEE International Conference on Cloud Computing Technology and Science. 2017. P. 331-334.
9. Halwai S., Pritika P., Seema S. 360 Deg. Overview of Fog Computing // International Journal for Research in Applied Science and Engineering Technology. 2019. Vol. 7. P. 376-378.
10. Numhauser J. Fog Computing - introduction to a new cloud evolution // Proceedings from the CIES III Congress. 2012. P. 111-126.
11. Rahman G., Chuah C. Fog computing, applications, security and challenges, review // International journal of engineering and technology. 2018. Vol. 7. P. 1615-1621.
12. Wang Y. Cloud-dew architecture // International Journal of Cloud Computing. 2015. Vol. 4. P. 199-210.
13. Wang Y. Definition and categorization of dew computing // Open Journal of Cloud Computing. 2016. Vol. 3. P. 1-7.

CLOUD, FOG, DEW, SOFT COMPUTING IN FUZZY SENSOR SYSTEMS

Goryachev, Victor A.

*Graduate student
University "Dubna"
Dubna, Russia
gva.asp@uni-dubna.ru*

Averkin, Alexei N.

*Candidate of physical and mathematical sciences, associate professor
Federal Research Center "Informatics and Management" of the Russian Academy of Sciences
Plekhanov Russian University of Economics
Moscow, Russia
averkin2003@inbox.ru*

Abstract

The research activity of the authors consists in the intellectualization of monitoring systems based on wireless sensors using the theory of fuzzy logic. Specifically, this article provides an example of programming a soft fuzzy sensor based on a fuzzy measure (see Section 1). In addition, this work considers distributed computing models in the general case and in the context of monitoring systems based on wireless sensor networks (see Sections 2-4). Based on the results of studying the three most promising models of distributed computing (cloud computing, fog computing, dew computing), their advantages and disadvantages related to low-power monitoring systems are described, and the principles of an optimal distributed computing model are formulated to ensure the effective operation of fuzzy sensor monitoring systems.

Keywords

Cloud computing; fog computing; dew computing; soft sensors; fuzzy sensors; fuzzy measure; fuzzy logic

References

1. Averkin, A.N, Lavrov, G.K. Ispol'zovanie tekhnologii fuzzy smart sensors wsn dlya intellektualizatsii sistem podderzhki prinyatiya reshenii v besprovodnykh sensornykh setyakh [The use of fuzzy smart sensors wsn technology for the intellectualization of decision support systems in wireless sensor networks]. Sistemnyi analiz v nauke i obrazovanii [Systems analysis in science and education]. 2017. №1. pp. 1-14.
2. Emel'yanov, S.G., Titov, V.S., Bobyr' M.V. Intellektual'nye sistemy na osnove nechetkoi logiki i myagkikh arifmeticheskikh operatsii: ucheb. posobie [Intelligent systems based on fuzzy logic and soft arithmetic operations]. Moscow: Publ. ARGAMAK-MEDIA, 2014. 341 p.
3. Kirsanova, A.A., Radchenko, G.I., Chernykh, A.N. Obzor tekhnologii organizatsii tumannykh vychislenii [Overview of Fog Computing Technologies]. Vestnik YuUrGU. Seriya: Vychislitel'naya matematika i informatika [Bulletin of SUSU. Series: Computational Mathematics and Informatics]. 2020. Tom 9. №3. pp. 35-63.
4. Prokopchina, S.V. Baiesovskie intellektual'nye tekhnologii: metodologiya i primeneniye v zadachakh tsifrovizatsii [Bayesian Intelligent Technologies: Methodology and Application in Digitalization Problems]. Mezhdunarodnaya konferentsiya po myagkim vychisleniyam i izmereniyam [International Conference on Soft Computing and Measurements]. 2019. Tom 1. pp. 3-6.
5. Fataliev, T.Kh., Mekhtiev, Sh.A. Detsentralizatsiya obrabotki dannykh v srede elektronnoi nauki [Decentralization of data processing in the e-science environment]. Informatsionnye tekhnologii. Problemy i resheniya [Information Technology. Problems and solutions]. 2019. №4(9). pp. 17-22.
6. Kherrold, D. "Myagkie" datchiki - noveishii instrument promyshlennoi avtomatizatsii ["Soft" sensors - the latest tool in industrial automation]. Datchiki i sistemy [Sensors and systems]. 2001. №12. pp. 36-41.

7. Bonomi, F., Milito, R., Zhu, J., Addepalli, S. Fog computing and its role in the internet of things. Proceedings of the first edition of the MCC workshop on Mobile cloud computing. 2012. pp. 13-16.
8. Hong, H. From Cloud Computing to Fog Computing: Unleash the Power of Edge and End Devices. IEEE International Conference on Cloud Computing Technology and Science. 2017. pp. 331-334.
9. Halwai, S., Pritika, P., Seema, S. 360 Deg. Overview of Fog Computing. International Journal for Research in Applied Science and Engineering Technology. 2019. Vol. 7. pp. 376-378.
10. Numhauser, J. Fog Computing - introduction to a new cloud evolution. Proceedings from the CIES III Congress. 2012. pp. 111-126.
11. Rahman, G., Chuah, C. Fog computing, applications, security and challenges, review. International journal of engineering and technology. 2018. Vol. 7. pp. 1615-1621.
12. Wang, Y. Cloud-dew architecture. International Journal of Cloud Computing. 2015. Vol. 4. pp. 199-210.
13. Wang, Y. Definition and categorization of dew computing. Open Journal of Cloud Computing. 2016. Vol. 3. pp. 1-7.