

УДК 004.942

К ВОПРОСУ О ТУМАННЫХ ВЫЧИСЛЕНИЯХ И ИНТЕРНЕТЕ ВЕЩЕЙ

Проферансов Дмитрий Юрьевич,

канд. тех. наук, заместитель генерального директора по международным связям,

ОАО «Т-Платформы»,

<https://www.t-platforms.ru>,

E-mail: dmitry.proferansov@t-platforms.ru,

тел.: +7 (499) 650-01-50,

Сафонова Ирина Евгеньевна,

д-р техн. наук, проф. каф. «Математика и информатика»,

Московский университет имени С.Ю. Витте,

<http://www.muiiv.ru>,

проф. каф. «Вычислительные системы и сети»

Российский университет транспорта (МИИТ),

<http://www.miit.ru>,

E-mail: iesafonova@yandex.ru,

тел.: +7 (963)925-32-27

Разработка и эффективное применение технологий «туманных вычислений» требует решения сложнейших задач, связанных с управлением и обработкой огромных массивов данных, в том числе задач рационального построения сетей «нижнего» уровня, обеспечивающих функционирование оконечных устройств в рамках концепции IoT. В статье описывается использование методов теории графов для решения таких задач.

Ключевые слова: многоуровневая графовая модель, Интернет вещей, туманные вычисления, эталонная архитектура OpenFog

DOI 10.21777/2500-2112-2017-4-30-39

Введение

В настоящее время, по мере того как стремительное развитие промышленной и бытовой электроники приводит к тому, что традиционное оборудование, используемое в производственных процессах и домашних условиях все более и более наделяется «умными» функциями и объединяется в сети, требующие практически непрерывной обработки данных, растут и требования к вычислительным возможностям этого оборудования. Необходимость контроля и управления как всевозможными отдельными механизмами и машинами, так и окружающей человека средой, транспортными потоками, производством, бизнесом, здравоохранением и образованием, безопасностью, социальными процессами и т.д. привело к созданию большого числа устройств, взаимодействующих с людьми, центрами обработки данных и между собой. Для их эффективного функционирования потребовалось создание глобальной системы связи, в качестве которой естественным образом выступил Интернет, а необходимость решения задачи управления и контроля породило концепцию сети, ориентированной на связь устройств между собой и получившей название «Интернет вещей» (Internet of Things, IoT).

Термин «Интернет вещей» впервые использовал в 1999 году руководитель исследовательского центра «Auto-ID» Массачусетского технологического института Кевин Эштон, предложивший компании Procter&Gamble использовать радиочастотные метки (RFID) на продукции для совершенствования управления цепочками поставок. Со временем это название стало общеупотребительным, хотя его смысл эволюционировал значительно расширившись.

Следует отметить, что в рамках концепции IoT основой эффективного функционирования как оконечных устройств, обладающих «умными» функциями, так и образуемых ими сетей, становятся работа в реальном времени, надежность и безопасность, что далеко не всегда может быть обеспечено при использовании схемы взаимодействия «клиент-сервер», характерной для «классического» Интернета и базирующейся на облачных технологиях/вычислениях (CloudComputing). Для решения этих задач была

разработана концепция «туманных вычислений», призванная, в первую очередь, приблизить обработку и хранение данных к устройствам их генерирующим и использующим.

Модифицированная концепция облачной технологии, применимой для IoT, и получившей название «туманные вычисления» (FogComputing, FC), была предложена исследователями компании Cisco. Название «туманные вычисления» призвано подчеркнуть идею приближения «облаков» к «земле». Согласно концепции, архитектура FC является трехуровневой. На нижнем уровне – «земле» – располагаются миллиарды вещей, на верхнем – множество облачных центров обработки данных (ЦОД), предоставляющих ресурсы для приложений, требующих значительных вычислительных мощностей и/или значительных объемов сохраняемых данных. И, соответственно, между ними находится «туман» – десятки тысяч географически распределенных более мелких управляющих ЦОД, достаточных для решения локальных задач.

1 Схема и типы систем обработки данных в рамках концепции туманных вычислений

Как было отмечено выше, концепция «туманных вычислений» в отличие от «облачных вычислений» предполагает приближение обработки данных к оконечным устройствам сетей – компьютерам, мобильным устройствам, датчикам и т.д. Следует отметить, что в настоящее время широкое распространение получил термин «граничные вычисления» (EdgeComputing), который по своей сути довольно близок к «туманным», поскольку так же предполагает осуществление ключевых операций по сбору и обработке данных вне «облака». Однако, ключевым различием между «туманными» и «граничными» вычислениями принято считать степень приближения точек обработки данных к оконечным устройствам сетей. Если концепция «тумана» предполагает отправку данных, сгенерированных оконечными устройствами, для обработки и/или хранения на локальные центры обработки (FogNodes), то в концепции «граничные» основные задачи обработки данных решаются непосредственно на оконечных устройствах. Типичным примером оконечных устройств, функционирование которых строится в рамках концепции «граничных вычислений» являются беспилотные автомобили. Таким образом, несмотря на концептуальную схожесть моделей «туманных» и «граничных» вычислений, в первую очередь, с точки зрения их отличия от «облачных», между ними имеются довольно серьезные различия и в рамках данной статьи речь идет именно о «туманных вычислениях».

Рассматривая вопрос о различиях между «облаками» и «туманом» необходимо отметить, что с точки зрения авторов, несмотря на имеющиеся различия, противопоставление моделей Fog/EdgeComputing модели CloudComputing является ошибочным и эти модели следует рассматривать не как альтернативные, а как взаимно дополняемые.

Итак, основная идея концепции FogComputing заключается в обеспечении эффективного, надежного и безопасного взаимодействия огромного множества устройств между собой, с локальными центрами обработки данных (ЦОД) и с облачными ЦОД. Принципиальная схема реализации концепции «туманных вычислений» представлена на рисунке 1.

На рисунке 2 приводятся различные типы систем обработки данных для «туманных вычислений» [1].

Как видно из приведенных схем, для «туманных вычислений» характерным является потребление пользователями сервисных функций ресурсов, размещенных на периферийных устройствах и в распределенной сети [1]. Данные располагаются на клиентских узлах, где предстоит их обработка или «поблизости». Основным способом сбора и передачи данных является беспроводная связь.

Преимущество сетей, построенных в соответствии с концепцией «туманных вычислений» перед «облачными» системами состоит в снижении латентности реакции на собираемые данные за счет обработки на месте их сбора, а для систем реального времени это является одним из ключевых факторов. Кроме того, в большинстве случаев защищенность систем «туманных вычислений» выше, чем для систем «облачных вычислений». Еще одним преимуществом «туманных вычислений» является снижение объема данных, передаваемых в «облако», что уменьшает требования к пропускной способности сети, увеличивает скорость обработки данных и снижает задержки в принятии решений. Таким образом, применение «туманных вычислений» позволяет полностью или частично решить ряд распространенных проблем, среди которых:

- высокие задержки в сети;
- трудности, связанные с подвижностью конечных точек;
- надежность связи;
- высокая стоимость полосы пропускания;
- непредвиденные сетевые «заторы»;
- большая географическая распределенность систем и клиентов.

2 Эталонная архитектура OpenFog

Консорциум OpenFog (Open Fog Consortium), учрежденный в 2015 году, предложил спецификацию эталонной архитектуры OpenFog (рисунок 3) – универсальной технологической модели для проектов, реализуемых в первую очередь в области Интернета вещей (IoT), сетей мобильной связи и приложений, реализующих функции искусственного интеллекта [1]. Ключевыми участниками консорциума являются компании Cisco, Intel, ARM, Dell, Microsoft.

Согласно предложенной модели, OpenFog-инфраструктура является набором узлов (FogNodes) на основе сетевых смарт-устройств, выполняющих обработку данных, также в спецификации содержатся описания вариантов иерархического выстраивания узлов, модели развертывания систем и примеры возможных реализаций.

Эталонная архитектура OpenFog основывается на следующих восьми технологических принципах (критериях):

- безопасность;
- масштабируемость;
- открытость;
- автономность;
- RAS (reliability, availability, serviceability – надежность, доступность, пригодность для сервисного обслуживания);
- адаптируемость;
- иерархический принцип выстраивания входящих элементов;
- программируемость.

Согласно мнению разработчиков компании Cisco, концепция туманных вычислений в наибольшей степени подходит для работы с системами межмашинного взаимодействия (machine-to-machine, M2M) и устройств, использующих человеко-машинный интерфейс (human-machineinterface, HMI). Они выделяют три основные группы таких устройств.

1 Устройства сбора данных, которые генерируются серийно различными сенсорами с частотой от нескольких миллисекунд до долей секунд. Примером являются устройства систем безопасности и комплексов управления промышленными объектами. Для них характерны требования низкой латентности при получении данных и высокой производительности для расчета требуемых характеристик в реальном времени.

2 Системы, которые отвечают за обработку данных, в том числе операционных. Здесь входные данные поступают с частотой от нескольких секунд до нескольких минут. Примерами таких систем являются устройства визуализации физических процессов, технологические промышленные системы.

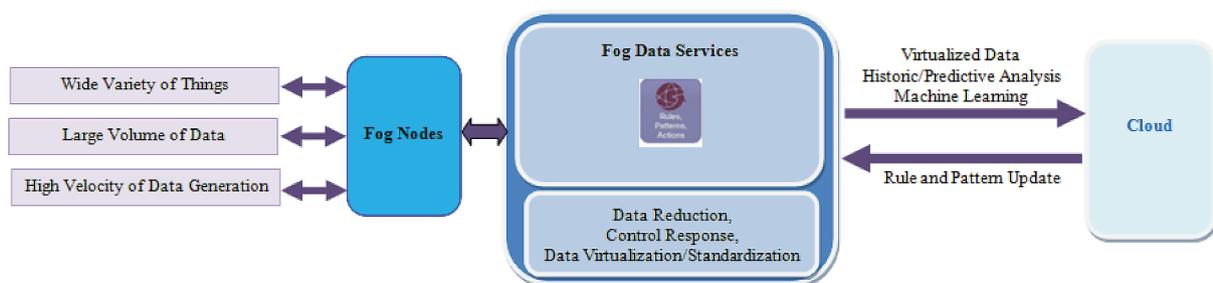


Рисунок 1 – Принципиальная схема реализации «туманных вычислений»

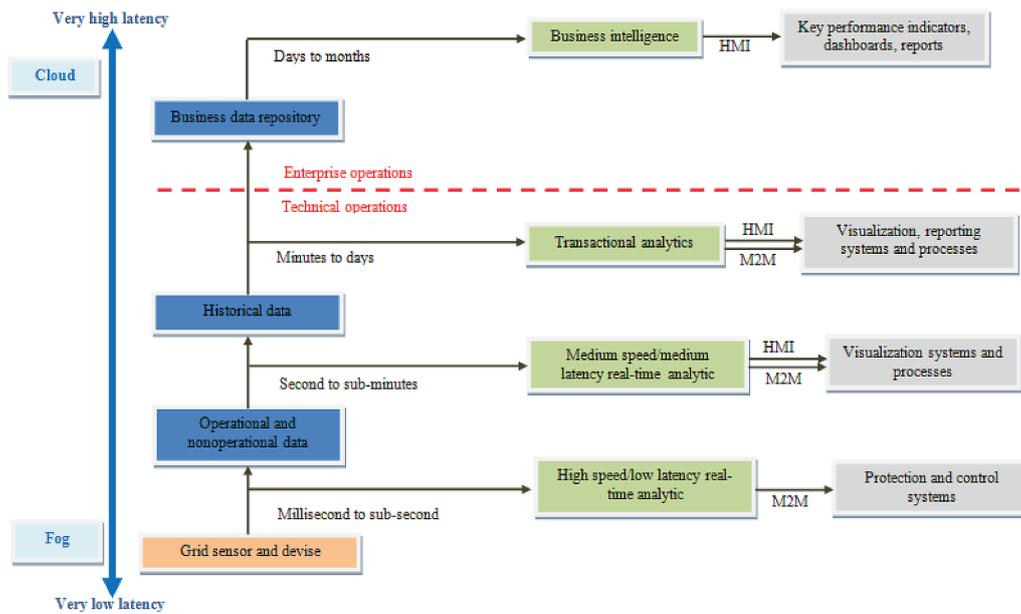


Рисунок 2 – Типы систем обработки данных для «туманных вычислений»

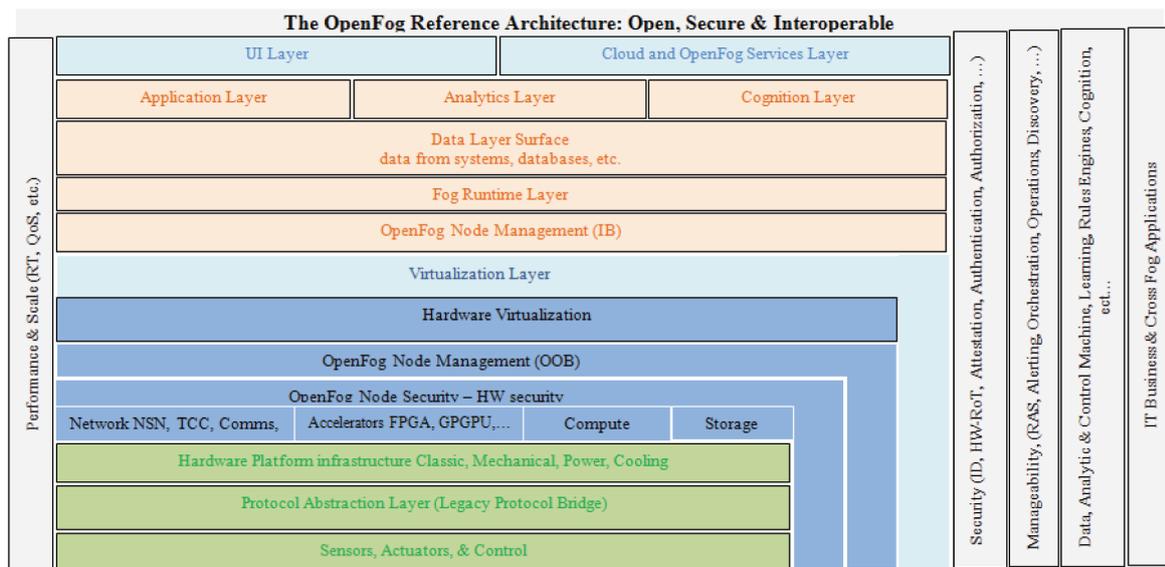


Рисунок 3 – Эталонная архитектура OpenFog

Требования к латентности собираемых данных и к их обработке не так велики, как для устройств первой группы, но все данные обрабатываются в режиме реального времени.

3 Устройства накопления и обработки исторических данных, собираемых с частотой от нескольких минут до нескольких дней. Пример – системы визуализации и подготовки отчетов.

3 Графовая модель FogComputing

При моделировании работы как отдельных элементов, так и системы, проектируемой и/или построенной на основе концепции FogComputing, в целом, например для оценки эффективности, работоспособности, оценки полосы пропускания, производительности, надежности оборудования, времени задержки для определенных видов сервиса, данных, программного обеспечения (ПО) и т.д., моделируемую систему целесообразно представить в виде нескольких взаимосвязанных уровней:

- 1) оборудование;
- 2) интерфейсы;

- 3) транспортная система (сеть);
- 4) операционных систем (ОС);
- 5) данные;
- 6) сервисы;
- 7) приложения.

Подобное представление моделируемого объекта позволяет задействовать аппарат теории графов.

На рисунке 4 в виде ориентированного графа $G=(Z,L)$ представлена многоуровневая графовая модель системы, выстроенной в соответствии с концепцией FogComputing. Множество вершин графа $Z_{\text{ОБОРУД}}, Z_{\text{ПУ}}, \dots, Z_{\text{ВУ}}, \dots; Z_{\text{ИНТЕРФ}}, Z_{\text{НМИ}}, \dots, Z_{\text{М2М}}, \dots; Z_{\text{ТС}}, Z_{\text{СУ}}, \dots, Z_{\text{КС}}, \dots; Z_{\text{ОС}}, Z_{\text{ОС1}}, \dots, Z_{\text{ОСп}}, Z_{\text{ДААННЫЕ}}, Z_{\text{О}}, \dots, Z_{\text{И}}, \dots, Z_{\text{Д}}, \dots; Z_{\text{СЕРВИС}}, Z_{\text{С1}}, \dots, Z_{\text{См}}; Z_{\text{ПРИЛОЖ}}, Z_{\text{П1}}, \dots, Z_{\text{Пк}}$ – это задачи, решаемые на каждом конкретном уровне. Множество дуг включает: $X_{\text{пр}}$ – множество параметров, необходимых для решения этих задач и множество информационных связей – H .

Множество параметров $X_{\text{пр}}$ составляют $X_{\text{пр}}^{\text{вх}}$ – входные и $X_{\text{пр}}^{\text{вых}}$ – выходные параметры:

$$X_{\text{пр}} = \{X_{\text{пр}}^{\text{вх}}, X_{\text{пр}}^{\text{вых}}\} \quad (1)$$

Причем:

$$L = X_{\text{пр}} \cup H \text{ и } X_{\text{пр}} \cap H = \emptyset, \quad (2)$$

здесь следует учитывать, что выходные параметры одних задач (вершин) могут являться входными для других [2].

Информационные связи преобразуются в булеву матрицу $n \times n$:

$$H = \left\| h_{gl} \right\|_{n \times n}, \quad (3)$$

элемент матрицы h_{gl} характеризует наличие информационных связей задач g и l :

$$h_{gl} = \begin{cases} 1, & \text{если задачи связаны по параметрам} \\ 0, & \text{в противном случае} \end{cases} \quad (4)$$

Решение какой-либо задачи можно представить следующим образом:

$$Z_i : \{X_{\text{пр}}^{\text{вх}}\} \Rightarrow Z_{\text{пр}}^{\text{вых}}. \quad (5)$$

Необходимо отметить, что для решения задач, которым соответствуют вершины графа $G=(Z,L)$ существует множество математических моделей, в последующем они используются для оценки параметров и характеристик элементов FogComputing. Каждая вершина графа соответствует одной или нескольким моделям, и выходные параметры одних моделей могут являться входными для других. Примеры моделей представлены в работах [2–5].

Каждый параметр, для решения задач характеризуется вектором признаков параметров, в который входят: единицы измерения параметра, вершина, где этот параметр является выходным; вершина, где этот параметр является входным, номер уровня, и другие индивидуальные признаки [2, 7]. Единицы измерения параметров определяются в соответствии с Международной системой единиц СИ, и в соответствии с семантикой конкретных задач моделирования [2]. При этом, все параметры целесообразно разделить на две группы:

- 1) параметры, соответствующие системе СИ;
- 2) все остальные параметры.

Для одних и тех же параметров единицы измерения могут быть различными и, следовательно, необходимо согласование параметров.

Как отмечалось выше, эталонная архитектура OpenFog основывается на следующих восьми критериях:

$$K = \{K_1^u, \dots, K_8^u\}, \quad (6)$$

где K – составной критерий;

u – номер уровня ($u=1, \dots, 7$);

K_i^u – критерий конкретного уровня, например – безопасность на уровне сервисов и т.д.

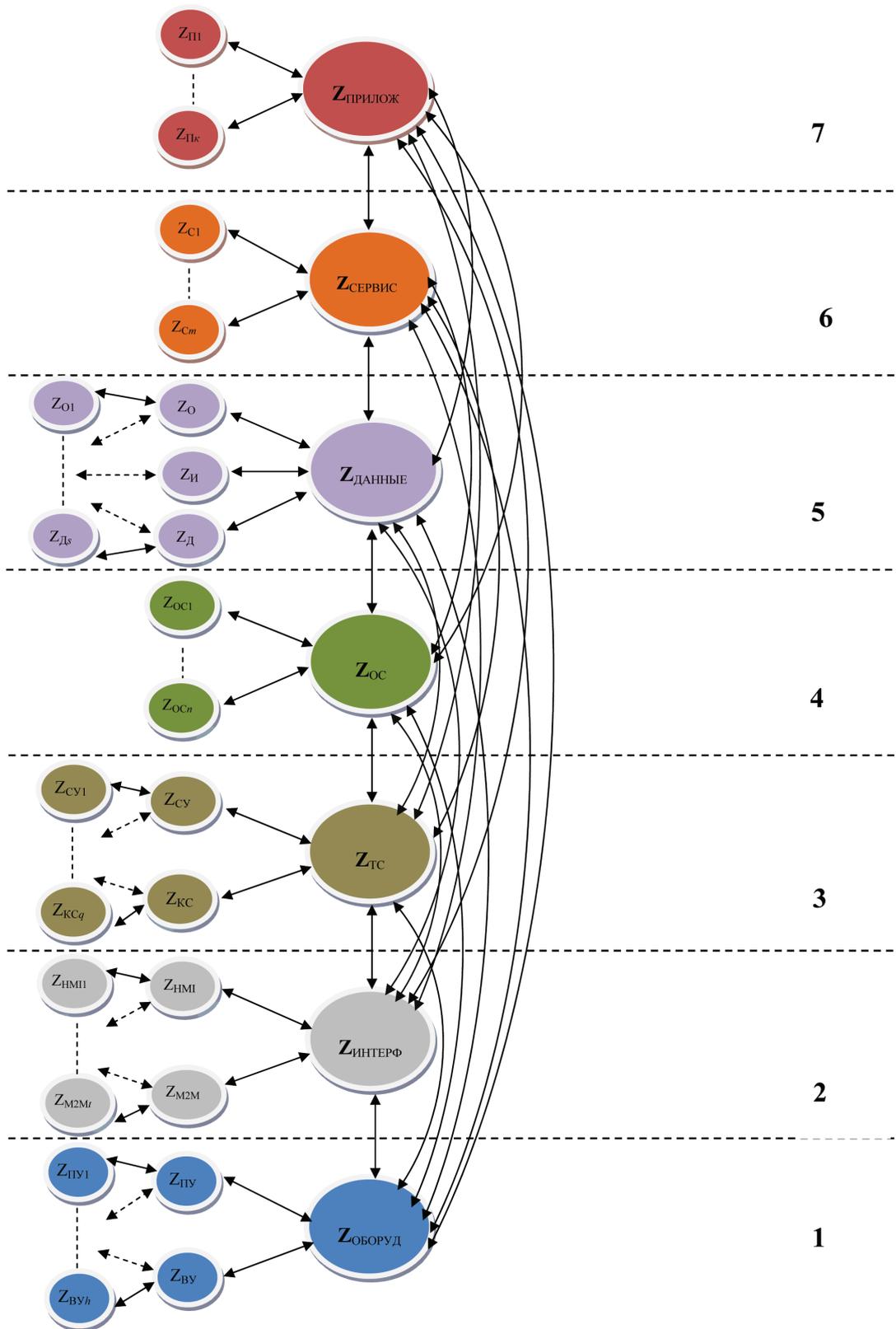


Рисунок 4 – Графовая модель системы, реализуемой в соответствии с концепцией FogComputing

Здесь следует остановиться на подробном описании каждого уровня.

1-й уровень, оборудование. Этому уровню соответствуют вершины графа $Z_{\text{ОБОРУД}}, Z_{\text{ПУ1}}, \dots, Z_{\text{ВУ1}}, \dots$. Вершина $Z_{\text{ОБОРУД}}$ определяет все задачи, решаемые на этом уровне; вершины $Z_{\text{ПУ1}}, \dots$ – (подключаемые устройства) задачи, решаемые на уровне широкого спектра подключаемых устройств; вершины $Z_{\text{ВУ1}}, \dots$ – задачи для устройств, которые позволяют проводить необходимые вычисления.

В качестве примера задач, решаемых на этом уровне можно привести расчет надежности и производительности работы устройств.

2-й уровень, интерфейсы. Как было отмечено выше, туманные вычисления в наибольшей степени подходят для работы с системами межмашинного взаимодействия – М2М, и устройств, использующих человеко-машинный интерфейс – НМІ. Уровню интерфейсов соответствуют вершины графа $Z_{\text{ИНТЕРФ}}, Z_{\text{НМІ1}}, \dots, Z_{\text{М2М1}}, \dots$. Вершина $Z_{\text{ИНТЕРФ}}$ описывает задачи, характерные для этого уровня; вершины $Z_{\text{НМІ1}}, \dots, Z_{\text{М2М1}}, \dots$ – определяют задачи для систем с межмашинным и человеко-машинным интерфейсами соответственно.

М2М – машино-машинное взаимодействие (Machine-to-Machine, Mobile-to-Machine, Machine-to-Mobile) – название технологии (сумма технологий), позволяющей обеспечить передачу данных между различными устройствами, причем, это могут быть группы устройств, например городской транспорт. Работы в области М2М координируют следующие организации: Eclipse Foundation, Focus Groupon Machine-to-Machine, входящая в Международный союз электросвязи, и инженерный комитет TR-50 М2М по интеллектуальным устройствам. Виды М2М – стационарный М2М, например управление технологическими процессами, платежные терминалы, счетчики и т.д.; и мобильный М2М, например, для управления автопарком, где М2М используется как бортовое устройство для мониторинга, диагностики, навигации, охраны и мобильной связи. Применение М2М – это системы доступа, системы охраны помещений, системы безопасности, дистанционный контроль и управление оборудованием, транспорт и мониторинг подвижных объектов, автоматы по продаже, платежные терминалы, здравоохранение и т.д.

НМІ – человеко-машинный интерфейс (Human – machine interface) – понятие, охватывающее инженерные решения, обеспечивающие взаимодействие человека с управляемыми им объектами (машинами, системами, устройствами). В качестве НМІ может выступать компьютер, типовое программное обеспечение, простой пульт из набора светодиодных индикаторов и т.д. Современные ЭВМ ориентированы на потоковую архитектуру, на реализацию интеллектуального человеко-машинного интерфейса, обеспечивающего не только системное решение задач, но и способность машины к логическому мышлению и самообучению, к ассоциативной обработке информации и получению логических выводов. Требования, которые различные пользователи предъявляют к НМІ, варьируются очень широко. Реализация интеллектуального человеко-машинного интерфейса сопряжена с возможностью решения задач распознавания и понимания естественного языка, для этого имеются системы распознавания (речи, рукописных текстов, изображений и т.д.). Создание удобного и эффективного человеко-машинного интерфейса является актуальной задачей.

Также среди задач уровня интерфейсов можно выделить, например – выбор стандартных шин интерфейса; оценка надежности на уровне интерфейсов и другие.

3-й уровень, транспортная система (сеть). Транспортная система служит для передачи информации и включает узлы туманной инфраструктуры – коммутаторы, маршрутизаторы и др.

Вершина $Z_{\text{ТС}}$ графа $G=(Z,L)$ определяет общие задачи, характерные для 3-го уровня, причем вершины $Z_{\text{СУ1}}, Z_{\text{СУ2}}, \dots$ – описывают задачи, решаемые на уровне сетевых устройств; вершины $Z_{\text{КС1}}, \dots, Z_{\text{КС4}}$ – задачи, решаемые на уровне каналов связи.

В качестве задач, решаемых на этом уровне можно привести – выбор канала связи; оценку пропускных способностей каналов; расчет коэффициента задержки сетевого оборудования; оценку задержки передачи сообщений и многие другие.

4-й уровень операционных систем (ОС). Здесь следует учитывать наличие различных типов ОС (UNIX-подобные ОС, ОС Windows, OS X и т. д.). Вершина $Z_{\text{ОС}}$ графа описывает общие задачи, характерные для уровня ОС. Вершины $Z_{\text{ОС1}}, \dots, Z_{\text{ОСn}}$ – это задачи, решаемые для каждой конкретной операционной системы, например:

- расчет коэффициента относительных потерь производительности ОС для многопроцессорной системы;

- определение среднего времени обработки запроса ОС;
- оценка среднего времени, затраченного на обращение к внешней памяти и анализ интенсивности поступления запросов ОС к устройствам внешней памяти;
- оценка времени реакции ОС при решении конкретных задач;
- оценка среднего времени, необходимого для передачи запроса ОС;
- оценка времени обращения к оперативной памяти;
- оптимизация структуры ядра открытых ОС по критерию информационной безопасности;
- оценка времени обнаружения ошибок в процессах;
- расчет вероятности пропуска контролируемого сигнала (количественной характеристики для задач контроля целостности файлов ОС) и многие другие.

5-й уровень, данные. Этому уровню соответствуют вершины $Z_{\text{данные}}, Z_{\text{о}}, \dots, Z_{\text{и}}, \dots, Z_{\text{д}}, \dots$ графа $G=(Z,L)$. Вершина $Z_{\text{данные}}$ описывает общие задачи, вершины $Z_{\text{о}}, Z_{\text{о1}}, \dots, Z_{\text{и}}, \dots, Z_{\text{и1}}, \dots, Z_{\text{д}}, \dots$ – соответственно задачи, характерные для оперативной информации (анализ в реальном времени), для исторических данных (анализ транзакций) и для данных длительного хранения (анализ BigData).

Примеры задач, решаемых на этом уровне:

- прогностический расчет скорости генерации новых данных;
- оптимизация размещения файлов и обработка запросов к СУБД;
- оценка объема данных;
- сжатие данных;
- распределенные вычисления при планировании запросов к СУБД;
- оценка целостности информации на уровне ссылок и другие задачи.

6-й уровень, сервисы. Это могут быть различные сервисы, например онлайн-сервисы (сервисы подобные Uber), сервисы, поставляющие потоковые услуги (например, сервисы Netflix, Amazon Prime, Hulu и Crunchyroll) и т.д.

Уровню сервисов соответствуют вершины $Z_{\text{сервис}}, Z_{\text{с1}}, \dots, Z_{\text{см}}$ графовой модели. Вершина $Z_{\text{сервис}}$ определяет общие задачи для уровня сервисов, вершины $Z_{\text{с1}}, \dots, Z_{\text{см}}$ – задачи для различных видов сервиса.

Примеры задач:

- расчет производительности для 6-го уровня;
- оценка качества обслуживания в виртуальных каналах VPN;
- оптимизация системных сервисов в соответствии с сетевыми ресурсами;
- оценка безопасности передачи конфиденциальной информации в широкополосных каналах связи;
- максимальная поддержка разных видов трафика 6-го уровня и много других задач.

7-й уровень, приложения. Приложениями являются ПО для научных исследований, системы автоматизированного проектирования, игры, приложения для работы художников, географически распределенные приложения для мониторинг трубопроводов, смарт-устройства в автомобиле, SmartGrid, системы управления работой светофоров и т.д.

Этому уровню соответствуют вершины $Z_{\text{прилож}}, Z_{\text{п1}}, \dots, Z_{\text{пк}}$ графа $G=(Z,L)$. Вершина $Z_{\text{прилож}}$ описывает общие задачи уровня приложений, вершины $Z_{\text{с1}}, \dots, Z_{\text{см}}$ – задачи для различных типов приложений. Это могут быть прикладные программы, установленные на компьютерах, планшетах, смартфонах пользователей, ПО ЦОДов и т.д.

Некоторые примеры задач:

- расчет максимальной производительности для 7-го уровня;
- распределение прикладных задач между пользователями по критерию средневзвешенной длины маршрута;
- прогностическая оценка условного среднего времени обслуживания заявки, необходимого для решения задачи продолжительностью в nt временной период;
- оценка среднего времени решения прикладных задач;

- расчет времени обмена с внешней памятью при решении прикладных корпоративных задач;
- прогностический расчет времени, необходимого для обработки данных в прикладной системе;
- расчет среднего времени обслуживания заявки для алгоритмов беспriorитетных дисциплин обслуживания;
- масштабируемость для 7-го уровня;
- многие другие задачи.

Основными свойствами многоуровневой графовой модели FogComputing являются [2, 7]:

- 1) свойство интеграции – должна быть возможность решения отдельных (частных) задач в зависимости от конкретной ситуации;
- 2) универсальность;
- 3) адекватность;
- 4) точность;
- 5) экономичность;
- 6) свойство развития – модель создается и функционирует с учетом пополнения, совершенствования и обновления.

Требования высокой степени универсальности, точности, адекватности с одной стороны, и высокой ее экономичности, с другой стороны, противоречивы. Поэтому компромиссные решения определяются характерными задачами [2, 3, 5, 6, 7].

В процессе анализа графовой модели системы, реализуемой в соответствии с концепцией FogComputing, проводится декомпозиция графа $G=(Z,L)$ на типовые подграфы (выделяются необходимые подграфы с целью экономии временных и вычислительных ресурсов), выясняется принадлежность параметров соответствующим вершинам графа, определяется множество связей между вершинами (задачами). Затем происходит подбор соответствующих моделей (формул) для расчета необходимых параметров при решении выбранных задач. Далее необходимо проанализировать всю графовую модель.

С помощью графа $G=(Z,L)$ можно интегрировать модели для расчета и оценки различных параметров в зависимости от конкретных задач, проводить многовариантные расчеты, эффективно оценивать практически все параметры и характеристики элементов FogComputing.

Графовая модель наглядно показывает взаимовлияние и взаимосвязь задач и параметров, а моделирование можно начать с любой задачи, которой соответствует вершина графа (одна или несколько).

Заключение

По мнению авторов, разработанная графовая модель может стать достаточно универсальным инструментом, как для оптимизации существующих сетей, так и для реализации новых систем и сетей, создаваемых в соответствии с концепцией FogComputing, и, в первую очередь, ориентированных на работу «Интернета Вещей». Интернет вещей и туманные вычисления еще только входят в повседневную жизнь, а модель предоставляет возможность рассчитывать, выполнять прогностическую оценку, оптимизировать необходимые характеристики всей системы, и отдельных элементов, определить «узкие места» и избыточность на этапе разработки с учетом эталонной архитектуры OpenFog, критериев (технологических принципов) и типов систем обработки данных.

Предложенная графовая модель может обеспечивать возможности определения основных свойств систем, сетей и сетевых устройств в рамках концепции FogComputing, оптимальных характеристик и способов их поддержания в рабочем состоянии.

Список литературы

1. Новиков И. Консорциум OpenFog опубликовал эталонную архитектуру туманных вычислений [электронный ресурс] / И.Новиков // портал PCWEEK АО «СК ПРЕСС»: [сайт]. [2017]. URL: <https://www.itweek.ru/infrastructure/article/detail.php?ID=192666> (дата обращения: 03.09.2017).
2. Профферансов Д.Ю., Сафонова И.Е. Многоуровневая графовая модель корпоративной телекоммуникационной сети // Телекоммуникации. – 2011. – №11. – С. 2–7.
3. Сафонова И.Е. Организация взаимодействия моделей элементов корпоративных функционально-ориентированных сетей // Качество. Инновации. Образование. – 2009. – № 7. – С. 49–57.

4. Вишнеvский В.М. Теоретические основы проектирования компьютерных сетей. – М.: Техносфера, 2003. – 396 с.
5. Сафонова И.Е. Методы прогностической оценки основных характеристик корпоративных функционально-ориентированных сетей // Качество. Инновации. Образование. – 2009. – № 3. – С. 68–73.
6. Сафонова И.Е., Макаров А.С., Морозов А.М. Алгоритм определения оптимального числа каналов в сети передачи данных с ненадежными каналами // Телекоммуникации. – 2013. – № 12. – С. 02–03.
7. Safonova I.E. On one approach to modeling enterprise-wide function-oriented computer networks // Telecommunications and Radio Engineering. – 2012. – Т. 71. – № 12. – С. 1087–1101.

TO THE QUESTION OF FOG COMPUTING AND THE INTERNET OF THINGS

Proferansov D.Y.,

*Candidate of Technical Sciences, Deputy CEO on International Communications, «T-Platforms»,
E-mail: dmitry.proferansov@t-platforms.ru,
tel.: +7 (499) 650-01-50,*

Safonova I.E.,

*Doctor of Technical Sciences, professor of Moscow Witte University,
professor of Russian University of Transport (RUT – MIIT)
E-mail: iesafonova@yandex.ru,
tel.: +7 (963)925-32-27*

The development and effective usage of Fog Computing technologies requires the solutions of the most complicated problems first of all, processing and management of big data, including the problems of rational construction of «lower level networks» that support the operation of terminal devices within the framework of the IoT concept. The article describes the use of methods of graph theory for solving such problems.

Keywords: multi-level graph model, Internet of Things, Fog Computing, reference architecture OpenFog