

Распределение поступающей нагрузки с применением SS-метода

Гадасин Денис Вадимович

к.т.н., доцент, доцент кафедры СИТиС, МТУСИ, Россия, dengadiplom@mail.ru

Тремасова Лилия Андреевна

ассистент кафедры СИТиС, МТУСИ, Россия, dengadiplom@mail.ru

Гадасин Даниил Денисович

студент гр. БСТ2103, МТУСИ, Россия, dengadiplom@mail.ru

АННОТАЦИЯ

Введение: Процесс передачи информации между двумя произвольными приложениями базируется на семиуровневой модели OSI. В процессе передачи можно выделить четыре основных участника, процесс представить в виде кортежа, а каждого участника описать множеством. **Методы:** Для осуществления процесса передачи возможно применить алгоритм, состоящий из четырех шагов: формирование БД участников, определение схемы передачи данных, определение модели распределения нагрузки, определение алгоритма минимизации целевой функции. Для минимизации целевой функции предлагается воспользоваться одним из способов решения транспортной задачи, которая относится к виду математических задач линейного программирования специального вида. Предлагается решить задачу на основе SS-метода, а начальный опорный план составить методом северо-западного угла. **Результаты исследования:** В работе рассматриваются задачи балансировки нагрузки для двух случаев. Параметром минимизации, «ценой перевозки», выступает время передачи информации между двумя узлами. В первом случае нагрузка распределяется между узлами сети, когда параметры качества заданы явными значениями. Для второго случая параметры качества задаются интервальными значениями, в пределах которого они изменяются, т.е. задача сводится к нечеткому виду. Приводится пример решения для заданных значений и определяется результирующий алгоритм решения сбалансированной задачи.

КЛЮЧЕВЫЕ СЛОВА: нечеткость, транспортная задача, балансировка, SS-метода, дефазификация.

Введение

Процесс передачи информации между источником и получателем моделируется семи-уровневой моделью OSI/ISO. На сетевом уровне определяется маршрут доставки информации, который может проходить как через разные подсети, так и сегменты, с разной топологией и различными технологиями передачи [1-4]. На каждом уровне модели определены протоколы, которые осуществляют процесс передачи. Основным принципом работы модели является инкапсуляция, позволяющая скрыть содержимое одного уровня от другого. Упрощенно процесс инкапсуляции заключается в том, что на входе уровня, протоколы добавляют к поступающим данным свой заголовок, в котором содержится информация, необходимая для корректного процесса передачи. На любое действие затрачивается определенное количество времени, сумма которого определяет итоговую задержку передачи информации между двумя точками. Параметр задержки, наряду с параметрами скорости передачи и надежности входит в основные, определяющие качество обслуживания, одним из способов повышения качества связи в информационных системах является их интеграция с системами интеллектуального тестирования [5,6].

Правильная и согласованная организация работы всех составных частей определенного уровня определяет конечную эффективность процесса передачи, но оптимальная организация процесса обработки информации на нескольких уровнях модели не всегда ведет к оптимальности процесса доставки информации от произвольной точки А в произвольную точку В. Возможность определения оптимального решения для одного этапа позволяет определить степень его влияния на эффективность работы всей модели в рамках заданных параметров качества исходя из более точной маршрутизации пакетов [7], или применяя алгоритмы и модели кластеризации [8-13].

В процессе передачи информации от одного участника к другому возможно выделить четыре основных составляющих вовлеченные в данный процесс:

1. Отправитель информации
2. Средства и алгоритмы обработки информации в виде необходимый для передачи
3. Средства и алгоритмы передачи информации
4. Получатель информации

Тогда модель процесса передачи (TP) возможно определить в виде кортежа:

$$TP = \langle S, H, T, R \rangle \quad (1)$$

где

TP (transfer process) – процесс передачи;

S (sender) – отправитель;

H (handler) – обработка;

T (transfer) – передача;

R (receiver) – получатель.

Каждый участник процесса передачи определяется своим множеством:

$$S = S(x) = \{x_1, x_2, \dots, x_n\} \text{ – множество отправителей;}$$

$$H = H(k) = \{k_1, k_2, \dots, k_i\} \text{ – множество систем обработки;}$$

$$Tr = Tr(l) = \{p_1, p_2, \dots, p_j\} \text{ – множество протоколов передачи;}$$

$$R = R(y) = \{y_1, y_2, \dots, y_m\} \text{ – множество получателей.}$$

Модель описывает процесс доставки информации от n отправителей к m получателям, причем обработкой информации занимаются k устройств, а сам процесс доставки осуществляется посредством p протоколов. Тогда целевой функцией Z данной модели является минимизация временных затрат процесса передачи T на каждом этапе:

$$T = T_S + T_H + T_{Tr} + T_R$$

$$Z(TP) \rightarrow \min(T) \quad (2)$$

Основным параметром, которые будет определять качество системы является время доставки сообщения, которое должны быть меньше реального времени системы, в условиях неравномерной поступающей нагрузки на узлы обработки информации, т.е. система должна обладать как гибкостью перенастройки узлов, так и скоростью адаптации к изменяющейся ситуации.

Обобщенный алгоритм решения.

Для решения поставленной задачи предполагается использовать следующий обобщенный алгоритм решения.

1. Формирование базы данных для отдельных узлов сети, где консолидируется и интерпретируется информация для последующей обработки [14], которая должна в себя включать сведения:

- о количестве входящих и исходящих узлах;
- об объеме трафика, который должен быть перераспределен между входящими и исходящими узлами;
- о типах каналов, которые организованы между узлами;
- о скорости передачи информации по каждому из каналов;
- о временных ограничениях и существующих задержках;
- о временной стоимости передачи одной единицы сообщения;

2. Определение схемы передачи данных, которая может быть организована одним из трех способов: «один к одному», «один ко многим», «многие ко многим». Исходя из того, что решением задачи должен быть план распределения поступающей нагрузки, необходимо выбрать схему «многие ко многим» (рис. 1).

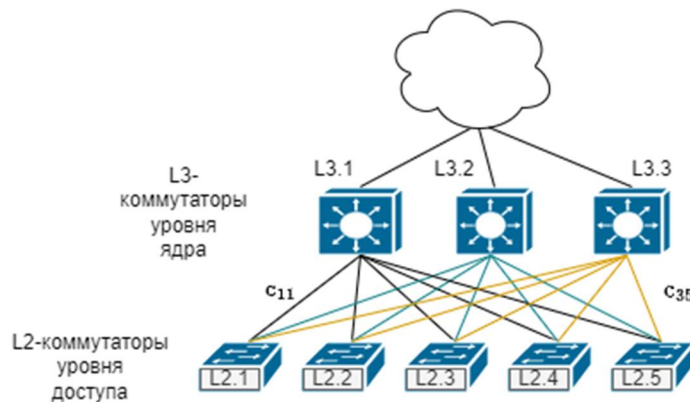


Рис. 1. Схема сети передачи данных

3. Определение модели распределения нагрузки исходя из обобщенной модели процесса передачи и целевой функции, например, модель Бэкмана или роя частиц [15, 16]. В данном случае, определяемая модель должна соответствовать модели для решения транспортной задачи. Модель для постановки классической транспортной задачи [17] определяется выражением:

$$\sum_{i=1}^n \sum_{j=1}^m (c_{ij} \cdot x_{ij}) \rightarrow \min_{x_{ij} \in \Psi}$$

$$\Psi : \sum_{j=1}^m x_{ij} = a_i, i = \overline{1, n},$$

$$\sum_{i=1}^n x_{ij} = b_j, j = \overline{1, m},$$

$$x_{ij} \geq 0, i = \overline{1, n}, j = \overline{1, m} \quad (3)$$

где n – количество исходящих узлов, m – количество входящих узлов, a_i – максимальное количество информации, которое может быть передано с исходящего узла, b_j – максимальное количество информации, которое может быть обработано входящим узлом, c_{ij} – весовой коэффициент, который определяет какое количество информации должно быть передано с определенного узла i на определенный узел j , для того что бы целевая функция была бы минимальной, x_{ij} – определяет параметр стоимости передачи данных по определенному каналу, стоимость может выражаться в количестве информации, надежности. Исходя из того, что параметром, по которому проводится минимизация целевой функции является время, то стоимость определяется во временных единицах кратных секунде. Зная размер пакета в битах и битовую скорость, которую поддерживает определенный канал возможно определить время передачи.

4. Решение транспортной задачи, состоит в том, чтобы определить такие весовые коэффициенты для математической модели, чтобы целевая функция была бы минимальной, что соответствует решению задачи оптимизации на сетях и графах [18]. Решение транспортной задачи возможно разными методами: потенциалов, симплекс метод Данцига, stepping-stone метод (ss-метод). Рассмотрим решение данной задачи ss-методом.

Решение задачи делится на два этапа. На первом этапе производится построение опорного плана и определяется значение целевой функции. На втором этапе производится пошаговое построение новых опорных планов по определенному правилу, для которых определяются значения целевой функции, из которых выбирается минимальное. После чего определяется возможно ли улучшение опорного плана и если оно невозможно – то считается что решение найдено.

Пусть есть сегмент сети, в который поступает информация из внешней сети. Сеть состоит из коммутаторов уровня ядра и коммутаторов уровня распределения, к которым подсоединяются коммутаторы уровня доступа, т.е. трехуровневая архитектура сети Ethernet. Топология организации сети смешанная, но третий и второй уровень в целях повышения надежности связности сети образуют полносвязную топологию, т.е. каждое коммутационное устройство уровня ядра связано с каждым коммутатором уровня доступа. Каждое устройство может обработать определенное количество информации в единицу времени, а канал, который связывает два устройства на разных уровнях может пропустить через себя определенное количество информации, что в конечном итоге определяет стоимость канала. Для

примера будем использовать целочисленные значения, при решении задачи в качестве исходных данных возьмём максимально близкие к реальной ситуации.

Для рассматриваемого случая определим, что коммутирующих устройств уровня ядра будет три, а уровня L2 пять. Устройства уровня ядра могут обработать за единицу времени соответственно 90, 130 и 130 единиц информации, а устройства уровня распределения 50, 60, 70, 80 и 90 единиц. Временные затраты, между уровнями определяются следующей матрицей T:

	L2.1	L2.2	L2.3	L2.4	L2.5	
L3.1	4	2	3	7	8	90
L3.2	4	9	4	6	9	130
L3.3	6	3	8	5	6	130
	50	60	70	80	90	

Рис. 2. Матрица временных затрат.

На первом этапе решения задачи необходимо определить под какой вид подходят начальные условия: сбалансированный или не сбалансированный. Сбалансированной называются такие условия если суммарная производительность узлов отправителей равна суммарной мощности узлов получателей. В решении будем рассматривать только сбалансированные задачи, т.к. в противном случае возможна либо недозагрузка линий передач или образование очередей на узлах получателей, что негативно отражается на величине итоговой задержки. Проверка сбалансированности для настоящих условий определяется путем решения равенства:

$$L3.1+L3.2+L3.3 = L2.1+L2.2+L2.3+L2.4+L2.5$$

$$90+130+130=50+60+70+80+90$$

$$350=350$$

Следующим этапом решения задачи является составление начального опорного плана. Начальным опорным планом будем называть матрицу, которая показывает какое количество нагрузки должно быть направлено от узла L3.x на узел L2.y. Исходя из того, что задача является сбалансированной, то в данной матрице будет занято m+n-1 клетка, т.е. 3+5-1=7. Теперь необходимо построить матрицу методом северо-западного угла. Прежде чем начать заполнение матрицы необходимо провести упорядочивание коммутирующих устройств исходя из их мощности по обработке трафика, упорядочивание проводится по возрастанию. По строкам устройства уровня L3, а по столбцам устройства уровня L2. Заполнение матрицы подчиняется единому правилу обхода «слева-направо, сверху-вниз». Для этого в ячейке a₁₁ определяется значение которое есть наименьшее между min(L3.1;L2.1). Если выбранное минимальное значение соответствует L2.1 то новое значение для L2.1 равно нулю, новое значение L3.1=L3.1-L2.1, осуществляется переход в ячейку a₂₁ и для данной ячейки проводится аналогичное сравнение только между значениями L3.1 и L2.2. Если выбранное минимальное значение соответствует L3.1, то новое значение L3.1 равно нулю, новое значение L2.1=L2.1-L3.1, после чего осуществляется переход в ячейку a₂₁ и проводится выбор наименьшего между L3.2 и L2.1. Алгоритм первоначального заполнения проводится до тех

пор, пока все значения L3.x и L2.y не станут равными нулю. В нашем случае начальным опорным планом является матрица:

$$A_0^0 = \begin{vmatrix} 50 & 40 & & & \\ & 20 & 70 & 40 & \\ & & & 40 & 90 \end{vmatrix} \quad (4)$$

После того, как построен начальный опорный план для него вычисляется значение целевой функции. Для определения значения целевой функции необходимо произвести последовательное сложение произведений соответствующих ячеек матрицы временных затрат с матрицей опорного плана.

$$Z(A_1^0) = \sum_{i=1}^n \sum_{j=1}^m t_{ij} \cdot a_{ij} = 4 \cdot 50 + 2 \cdot 40 + 9 \cdot 20 + 4 \cdot 70 + 6 \cdot 40 + 5 \cdot 40 + 6 \cdot 90 = 1720$$

Третьим и основным этапом является нахождение нового опорного плана и определение для него значения целевой функции. Для этого применяется циклическая процедура обмена единицей времени, что изменяет конечную временную стоимость. Исходя из того, что общее количество ненулевых ячеек в матрице m+n-1=7, а общее количество m*n=15, то необходимо произвести m*n-(m+n-1)=15-7=8 вариантов обмена. Для каждого варианта вычисляются изменения и выбирается тот вариант, где величина изменения отрицательна, в том случае если образуется несколько отрицательных изменений, то выбирается наименьшее, а если два или более отрицательных изменений равны между собой, то необходимо рассмотреть все эти варианты, т.к. в данном случае получается несколько оптимальных вариантов.

$$\begin{matrix} A_1^1 = \begin{vmatrix} 50 \cdot \cdot & 40 \cdot \cdot - & & + & \\ & 20 \cdot \cdot + & 70 \cdot \cdot - & 40 \cdot \cdot & \\ & & & 40 \cdot \cdot & 90 \cdot \cdot \end{vmatrix} & A_2^1 = \begin{vmatrix} 50 \cdot \cdot & 40 \cdot \cdot - & & + & \\ & 20 \cdot \cdot + & 70 \cdot \cdot & 40 \cdot \cdot - & \\ & & & 40 \cdot \cdot & 90 \cdot \cdot \end{vmatrix} \\ A_3^1 = \begin{vmatrix} 50 \cdot \cdot & 40 \cdot \cdot - & & & + \\ & 20 \cdot \cdot + & 70 \cdot \cdot & 40 \cdot \cdot - & \\ & & & 40 \cdot \cdot + & 90 \cdot \cdot - \end{vmatrix} & A_4^1 = \begin{vmatrix} 50 \cdot \cdot - & 40 \cdot \cdot + & & & \\ + & 20 \cdot \cdot - & 70 \cdot \cdot & 40 \cdot \cdot & \\ & & & 40 \cdot \cdot & 90 \cdot \cdot \end{vmatrix} \\ A_5^1 = \begin{vmatrix} 50 \cdot \cdot & 40 \cdot \cdot & & & \\ & 20 \cdot \cdot & 70 \cdot \cdot & 40 \cdot \cdot - & + \\ & & & 40 \cdot \cdot + & 90 \cdot \cdot - \end{vmatrix} & A_6^1 = \begin{vmatrix} 50 \cdot \cdot - & 40 \cdot \cdot + & & & \\ & 20 \cdot \cdot - & 70 \cdot \cdot & 40 \cdot \cdot + & \\ + & & & 40 \cdot \cdot - & 90 \cdot \cdot \end{vmatrix} \\ A_7^1 = \begin{vmatrix} 50 \cdot \cdot & 40 \cdot \cdot & & & \\ & 20 \cdot \cdot - & 70 \cdot \cdot & 40 \cdot \cdot + & \\ & + & & 40 \cdot \cdot - & 90 \cdot \cdot \end{vmatrix} & A_8^1 = \begin{vmatrix} 50 \cdot \cdot & 40 \cdot \cdot & & & \\ & 20 \cdot \cdot & 70 \cdot \cdot - & 40 \cdot \cdot + & \\ & & + & 40 \cdot \cdot - & 90 \cdot \cdot \end{vmatrix} \end{matrix} \quad (6)$$

Вычислим теперь величины изменений Δt, для чего будем складывать значения t из матрицы временных затрат исходя из матриц обмена единицей времени.

$$\Delta t_1^1 = 3 - 2 + 9 - 4 = 6$$

$$\Delta t_2^1 = 7 - 2 + 9 - 6 = 8$$

$$\Delta t_3^1 = 8 - 2 + 9 - 6 + 5 - 6 = 8$$

$$\Delta t_4^1 = 4 - 4 + 2 - 9 = -7$$

$$\Delta t_5^1 = 9 - 6 + 5 - 6 = 2$$

$$\Delta t_6^1 = 6 - 4 + 3 - 4 + 6 - 5 = 2$$

$$\Delta t_7^1 = 3 - 9 + 6 - 5 = -5$$

$$\Delta t_8^1 = 8 - 4 + 6 - 5 = 11$$

Исходя из полученных значений изменений времени необходимо выбрать вариант четыре, так как он имеет самое наименьшее отрицательное значение. Следующим шагом строится новый опорный план, для которого вычисляется новое значение целевой функции. Для построения плана, в ячейках, которые отмечены знаками плюса и минуса происходит уменьшение/увеличение их значений на минимальную величину ячейки, отмеченную знаком минус. В нашем случае такой ячейкой является ячейка $a_{22}=20$. Новый опорный план выглядит следующим образом.

$$A_0^1 = \begin{vmatrix} 30 \cdot & 60 \cdot & & & \\ 20 \cdot & & 70 \cdot & 40 \cdot & \\ & & & 40 \cdot & 90 \cdot \end{vmatrix} \quad (7)$$

Значение целевой функции для этого опорного плана $Z(A_0^1) = 1580$.

Следующим шагом решения задачи является поиск возможности улучшения данного опорного плана путем применения циклической процедуры обмена единицей времени. В результате реализации данного шага работы получается опорный план (8), для которого значение целевой функции $Z(A_0^2) = 1550$.

$$A_0^2 = \begin{vmatrix} & 60 \cdot & 30 \cdot & & \\ 50 \cdot & & 40 \cdot & 40 \cdot & \\ & & & 40 \cdot & 90 \cdot \end{vmatrix} \quad (8)$$

Дальнейшие попытки улучшения опорного плана, полученного на третьей итерации с использованием циклической процедуры обмена единицей информации приводят к тому, что величины изменения Δt для нулевых ячеек получаются больше нуля, т.е. опорный план для значения целевой функции $Z(A_0^3) = 1550$ является оптимальным, а временные затраты по распределению информации минимальны.

Практическое решение для интервальных значений.

Применение навыков по распределению нагрузки, полученных в ходе решения данной задачи в практической плоскости, не всегда удается использовать напрямую в связи с тем, что на практике очень редко удается задать параметры строго фиксированной величиной.

Параметры качества в большинстве случаев варьируются в пределах определенных границ, которые могут динамически изменяться, тем самым образуя нечеткие динамические системы [19]. В этих границах можно определить наиболее оптимальное значение при сложившихся условиях или определить отрезок, на котором достигается величина, определяющая качество передачи [20-23]. Тогда, для первого случая возможно поставить в соответствие произвольное $A=(a, b, c)$, а второй ситуации $A=(a, b, c, d)$, где величина b и отрезок $[b,c]$ определяют наиболее подходящие условия для передачи трафика, а величины a,c и a,d определяют границы за пределами которых качество связи ни при каких условиях не может быть достигнуто.

Интервальные отрезки, на которых достигается определенный максимум или включает в себя отрезок, на котором держится определенный максимум, можно интерпретировать как «около b » или «на интервале bc ». Если рассматривать вариант «около b », то можно предположить, что показатель качества возрастает от a к b , а затем убывает от b к a . Для варианта «на интервале bc » показатели качества возрастают от a к b , держатся на максимальном уровне на $[bc]$, после чего убывают от c к d . Данные события можно выразить в виде функции $\mu(x)$.

$$\mu(x) = \begin{cases} \frac{x-a}{b-a}, & \text{если } x \in [a, b] \\ \frac{x-c}{b-c}, & \text{если } x \in [b, c] \\ 0, & \text{во всех остальных случаях} \end{cases} \quad (9)$$

$$\mu(x) = \begin{cases} \frac{x-a}{b-a}, & \text{если } x \in [a, b[\\ 1, & \text{если } x \in [b, c] \\ \frac{x-d}{c-d}, & \text{если } x \in]c, d] \\ 0, & \text{во всех остальных случаях} \end{cases} \quad (10)$$

В следствии того, что на интервале возможно любое значение параметра, по которому высчитывается показатель качества, удовлетворяющий определенному диапазону, то возникает эффект нечеткости, а функции, указанные в (9,10) определяют этот параметр и рассматриваются как функция принадлежности нечеткого треугольного (9) и трапециевидного (10) числа. Данным числам соответствуют графики, где интервал достоверности определяет истинное относительное текущее значение показателя:

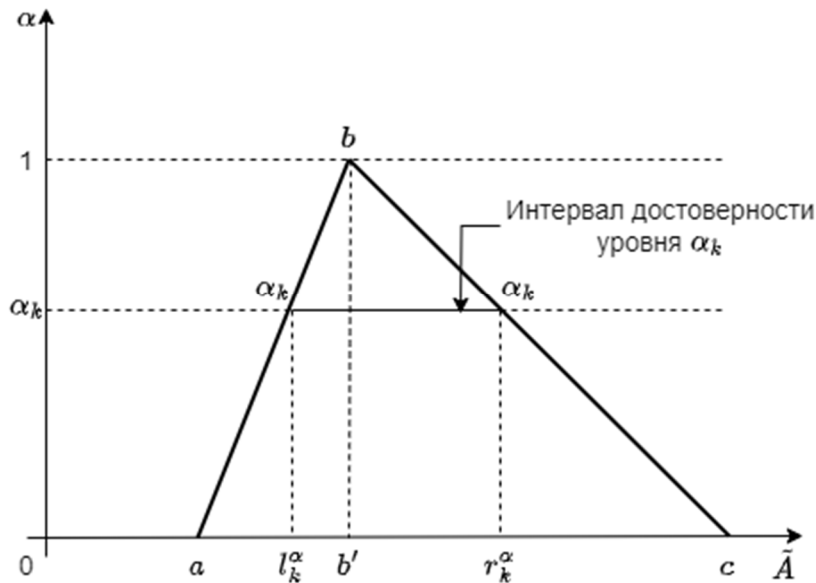


Рис. 3. Нечеткое треугольное число

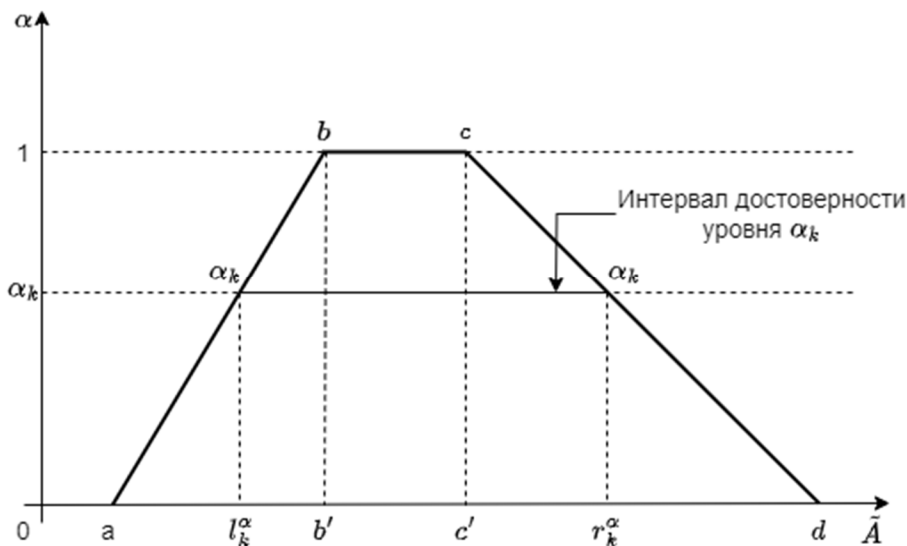


Рис. 4. Нечеткое трапециевидное число

Рассмотрим решение задачи распределения нагрузки для случая, когда временные показатели заданы интервальными значениями и на этом интервале присутствует значение или интервал значений, который наиболее предпочтителен. Данное условие рассматривается как решение задачи для мягкого реального времени. Основной сложностью при решении такого типа задач является переход от нечетких значений к четкому ответу, т.е. дефазификация нечеткого числа. В соответствии с тем, что нечеткие числа являются треугольными и для решения задачи применяются только основные арифметические операции, то действия сложения, вычитания, умножения, деления и умножения на константу проводятся аналогично действиям с многочленами.

$$\begin{aligned} \tilde{A}_1 &= (a_1, b_1, c_1) \\ \tilde{A}_2 &= (a_2, b_2, c_2) \\ \tilde{A}_1 \square \tilde{A}_2 &= (a_1 \square a_2, b_1 \square b_2, c_1 \square c_2) \end{aligned} \quad (11)$$

Для сравнения значений нечетких чисел будем проводить дефазификацию, исходя из метода средних:

$$\begin{aligned} \tilde{A}_{mp} = A_{mp} &= \frac{a + 2b + c}{2} \\ \tilde{A}_{mnp} = A_{mnp} &= \frac{a + b + c + d}{2} \end{aligned} \quad (12)$$

$A_{тр}$ – четкое значение нечеткого треугольного числа \tilde{A}_{mp} ;

$A_{трп}$ – четкое значение нечеткого трапециевидного числа \tilde{A}_{mnp} .

Дефазификация может быть произведена в двух случаях. В первом случае производится дефазификация матрицы временных затрат, после чего задача распределения нагрузки решается как для случая с четкими значениями, во втором случае дефазификация производится только для сравнения значений целевых функций. Алгоритм решения задачи претерпевает небольшое изменение для второго варианта. Исходя из того, что предельные колебания трафика, которые позволяют передавать данные в допустимом диапазоне качества небольшие, то для решения задачи предлагается четкие значения из первого варианта определить, как значения, при которых показатели качества достигают максимального значения, а допустимые отклонения вправо и влево не должны превышать трех единиц. Левая граница интервала должна всегда быть строго больше нуля и запрещаются равные отклонения для всех ячеек матрицы.

Технические параметры исходящих и входящих устройств остаются прежними. Основная цель работы, помимо нахождения оптимального плана распределения проверка совпадений на равное количество итераций при определении оптимального опорного плана и совпадающая структура плана распределения, т.е. одинаковость индексов i и j при a в итоговых целевых функциях.

Преобразуем исходные условия задачи для решения в условиях нечетких исходных данных.

	L2.1	L2.2	L2.3	L2.4	L2.5	
L3.1	(3,4,5)	(1,2,5)	(2,3,4)	(5,7,9)	(6,8,10)	90
L3.2	(1,4,5)	(7,9,11)	(1,4,7)	(3,6,7)	(6,9,10)	130
L3.3	(4,6,8)	(1,3,5)	(5,8,9)	(2,5,8)	(3,6,9)	130
	50	60	70	80	90	

Рис. 5. Исходные данные для задачи в нечетком виде.

В исходных данных все значения временных издержек заданы нечеткими треугольными числами, тогда, воспользовавшись правилом преобразования (12) исходной матрицей, исходя из которой будет строится начальный опорный план будет:

	L2.1	L2.2	L2.3	L2.4	L2.5	
L3.1	8	5	6	14	16	90
L3.2	7	18	8	11	17	130
L3.3	12	6	15	10	12	130
	50	60	70	80	90	

Рис. 6. Дефазифицированные данные.

После того, как получены четкие значения для временных ограничений приступаем к решению задачи исходя из алгоритма, определяющего ss-метод, а начальный опорный план строится методом северо-западного узла и полностью совпадает с описанным. Хотя значения в матрице являются четкими, но получены они из нечетких при помощи одного из способов дефазификации, поэтому матрицу будем обозначать как нечеткую. Для начального опорного плана значение целевой функции будет:

$$Z(\tilde{A}_0^0) = \sum_{i=1}^n \sum_{j=1}^m t_{ij} \cdot a_{ij} = 8 \cdot 50 + 5 \cdot 40 + 18 \cdot 20 + 8 \cdot 70 + 11 \cdot 40 + 10 \cdot 40 + 12 \cdot 90 = 3440$$

После проведения первого шага итерации для поиска оптимального решения получен опорный план из матрицы \tilde{A}_4^1 :

$$A_0^1 = \begin{vmatrix} 30 \cdot \cdot & 60 \cdot \cdot & & & & \\ 20 \cdot \cdot & & 70 \cdot \cdot & 40 \cdot \cdot & & \\ & & & 40 \cdot \cdot & 90 \cdot \cdot & \end{vmatrix} \quad (13)$$

для этого опорного плана целевая функция имеет значение

$$Z(\tilde{A}_0^1) = \sum_{i=1}^n \sum_{j=1}^m t_{ij} \cdot a_{ij} = 8 \cdot 30 + 5 \cdot 60 + 7 \cdot 20 + 8 \cdot 70 + 11 \cdot 40 + 10 \cdot 40 + 12 \cdot 90 = 3160$$

Проверим возможно ли улучшения данной целевой функции, приступим ко второй итерации. На данном шаге получается новый опорный план, дальнейшее улучшение которого невозможно:

$$A_0^2 = \begin{vmatrix} & 60 \cdot \cdot & 30 \cdot \cdot & & & \\ 50 \cdot \cdot & & 40 \cdot \cdot & 40 \cdot \cdot & & \\ & & & 40 \cdot \cdot & 90 \cdot \cdot & \end{vmatrix} \quad (14)$$

Целевая функция при этом имеет значение $Z(\tilde{A}_0^2) = 3070$.

После того, как задача распределения путем преобразования нечетких значений в четкие решена, необходимо решить задачу в нечетких числах. Для решения поставленной зада-

чи необходимо изменить алгоритм решения. Если в предыдущие шаги в качестве величины изменения на плюс/минус бралась единица времени, то теперь порядок обхода плюса/минуса определяет тот же алгоритм, а в качестве единицы отклонения берется минимальное число. После чего высчитывается целевая функция и за основу берется тот опорный план, где целевая функция будет минимальна. Исходя из того, что целевая функция будет представлять из себя нечеткое число, то дефазификация будет применяться только для сравнения целевых функций между собой.

Исходные данные для решения поставленной задачи остаются такими же как и в (Рис.5), а начальный опорный план будет как и (4). Вычислим для данного опорного плана значение целевой функции:

$$Z(\tilde{A}_0^0) = \sum_{i=1}^n \sum_{j=1}^m t_{ij} \cdot a_{ij} = (3, 4, 5) \cdot 50 + (1, 2, 5) \cdot 40 + (7, 9, 11) \cdot 20 + (1, 4, 7) \cdot 70 + (3, 6, 7) \cdot 40 + (2, 5, 8) \cdot 40 + (3, 6, 9) \cdot 90 = (870, 1720, 2570)$$

После того, как было вычислено начальное значение целевой функции необходимо перейти к следующему шагу алгоритма, определить матрицы изменений, для каждой матрицы рассчитать значение целевой функции и сравнить как между собой, так и со значением опорного плана. Если минимальным будет значение из опорного плана, то дальнейшее улучшение не требуется так как оно уже найдено, в противном случае строится новый опорный план, для которого и повторяется данная процедура. Преобразования на плюс/минус полностью соответствуют (6). В следствии того, что в данном варианте алгоритма необходимо каждый раз рассчитать целевую функции, то в ячейку с нулевым значением число, которое равно по модулю наименьшему числу, отмеченному знаком минус, а остальные значения высчитываются соответственно сложением или вычитанием данного числа, в зависимости от того какое знак стоит. В результате преобразований матрицы будут выглядеть так:

$$\begin{aligned} \tilde{A}_1^1 &= \begin{vmatrix} 50 \cdot & 0 \cdot & 40 & & \\ & 60 \cdot & 30 \cdot & 40 \cdot & \\ & & & 40 \cdot & 90 \cdot \end{vmatrix} & \tilde{A}_2^1 &= \begin{vmatrix} 50 \cdot & 0 \cdot & & 40 & \\ & 60 \cdot & 70 \cdot & 0 \cdot & \\ & & & & 40 \cdot & 90 \cdot \end{vmatrix} \\ \tilde{A}_3^1 &= \begin{vmatrix} 50 \cdot & 0 \cdot & & & 40 \\ & 60 \cdot & 70 \cdot & 0 \cdot & \\ & & & 80 \cdot + & 50 \cdot \end{vmatrix} & \tilde{A}_4^1 &= \begin{vmatrix} 30 \cdot & 60 \cdot & & & \\ 20 & 0 \cdot & 70 \cdot & 40 \cdot & \\ & & & & 40 \cdot & 90 \cdot \end{vmatrix} \\ \tilde{A}_5^1 &= \begin{vmatrix} 50 \cdot & 40 \cdot & & & \\ & 20 \cdot & 70 \cdot & 0 \cdot & 40 \\ & & & 80 \cdot & 50 \cdot \end{vmatrix} & \tilde{A}_6^1 &= \begin{vmatrix} 30 \cdot & 60 \cdot & & & \\ & 0 \cdot & 70 \cdot & 60 \cdot & \\ 20 & & & & 20 \cdot & 90 \cdot \end{vmatrix} \\ \tilde{A}_7^1 &= \begin{vmatrix} 50 \cdot & 40 \cdot & & & \\ & 0 \cdot & 70 \cdot & 60 \cdot & \\ & 20 & & & & 20 \cdot & 90 \cdot \end{vmatrix} & \tilde{A}_8^1 &= \begin{vmatrix} 50 \cdot & 40 \cdot & & & \\ & 20 \cdot & 30 \cdot & 80 \cdot & \\ & & & 40 & 0 \cdot & 90 \cdot \end{vmatrix} \end{aligned} \tag{15}$$

Для этих матриц целевые функции будут иметь нечеткие и дефазификационные значения:

$$\begin{aligned}
 Z(\tilde{A}_1^1) &= (1150, 1960, 2690) = 3880 \\
 Z(\tilde{A}_2^1) &= (1190, 2040, 2890) = 4080 \\
 Z(\tilde{A}_3^1) &= (1190, 2040, 2890) = 4080 \\
 Z(\tilde{A}_4^1) &= (710, 1580, 2450) = 3160 \\
 Z(\tilde{A}_5^1) &= (950, 1800, 2650) = 3600 \\
 Z(\tilde{A}_6^1) &= (790, 1640, 2490) = 3280 \\
 Z(\tilde{A}_7^1) &= (770, 1620, 2430) = 3220 \\
 Z(\tilde{A}_8^1) &= (1070, 1920, 2610) = 3760
 \end{aligned}$$

Как видно наименьшим значением является $Z(\tilde{A}_4^1) = (710, 1580, 2450) = 3160$ что так же меньше исходного опорного плана

$Z(\tilde{A}_0^0) = (870, 1720, 2570) = 3440$, таким образом новым опорным планом становится план, идентичный в (7).

На следующем шаге итерации, получается конечная целевая функция, для которой опорный план имеет вид

$$A_0^2 = \left| \begin{array}{ccc|c} & 60 \cdot \cdot & 30 \cdot \cdot & \\ 50 \cdot \cdot & & 40 \cdot \cdot & 40 \cdot \cdot \\ & & 40 \cdot \cdot & 90 \cdot \cdot \end{array} \right| \quad (16)$$

а целевая функция $Z(\tilde{A}_4^2) = (680, 1550, 2360) = 3070$, дальнейшее решение задачи не приводит к минимизации целевой функции.

После проведение расчетов, необходимо провести анализ полученных результатов. Для рассматриваемого примера при решении задачи в четком виде и в двух вариантах нечеткого вида, приведение матрицы временных затрат к четкому виду и сравнение целевых функций в нечетком виде, финальные опорные планы по своей структуре идентичны и отличаются лишь только значениями целевых функций. Для случая, когда временные параметры задаются в интервальном виде, как структура, так и значения целевых функций имеют одинаковые значения, что может говорить о том, что заданный интервал, определяющий предельные характеристики качества обслуживания для каждого канала передачи не является предельным, т.е. изменения характеристик трафика не носят критического значения.

Заключение

В результате применения алгоритма решения транспортной задачи в целях оптимального распределения нагрузки между узлами сети, был рассмотрен алгоритм решения задачи как для случая если показатели определены строго определенными параметрами или пара-

метры заданы интервалом, в пределах которого они могут изменяться. Результирующий алгоритм работы для сбалансированной задачи состоит в следующем:

1. Определить в каком виде, четком или нечетком, заданы временные ограничения.
2. Если ограничения заданы в нечетком виде, то необходимо определить тип нечетких чисел, треугольные или трапециевидные, которые наиболее подходят для отображения.
3. Для нечеткого решения задачи определить способ дефазификации нечетких чисел, по среднему отклонению или по центру тяжести.
4. Провести упорядочивание коммутирующих устройств по объему обрабатываемого трафика, упорядочивание производится по возрастанию.
5. Составить матрицу временных издержек по передаче сообщения от устройств уровня L3 к устройствам уровня L2.
6. Составить начальный опорный план методом северо-западного угла и для этого плана определить начальное значение целевой функции.
7. Начать итерационный процесс по улучшению опорного плана способом изменения единицы отклонения, метод плюс/минус. Если временные ограничения заданы в четком виде или была применена дефазификация для матрицы времени, то в качестве нового опорного плана выбирается вариант, в котором величина изменений Δt является минимально отрицательной. Данный вариант становится новым опорным планом для которого вычисляется значение целевой функции, если $\Delta t \geq 0$ значит рассматриваемый опорный план является оптимальным, а значение целевой функции минимально возможной. Если временные ограничения заданы в нечетком виде, то после применения метода плюс/минус, для каждого варианта необходимо рассчитать значение целевой функции и сравнить их как между собой, так и со значением целевой функции для опорного плана, если есть минимальное, то тогда такой вариант является новым опорным планом и итерационный процесс повторяется, в противном случае рассматриваемый опорный план является оптимальным, а значение целевой функции минимально возможной.
8. Провести анализ полученных результатов и составить отчет по выполненному заданию.

Литература

1. *Shvedov A.V., Gadasin D.V., Klygina O.G.* Determining Shortest Paths between Two Arbitrary Nodes in a Composite Transport Network Using Segment Routing // 2021 Systems of Signals Generating and Processing in the Field of on Board Communications, Conference Proceedings, Moscow, 2021. P. 9416072. DOI 10.1109/IEEECONF51389.2021.9416072. EDN QJRLWF.
2. *Kalmykov N.S., Dokuchaev V.A.* Segment routing as a basis for software defined network // T-Comm: Телекоммуникации и транспорт. 2021. Vol. 15, No. 7. P. 50-54. DOI 10.36724/2072-8735-2021-15-7-50-54. EDN LYVZCV.
3. *Gadasin D.V., Shvedov A.V., Klygina O.G., Shulginov I.V.* Routing Management system formation for machine-to-machine interaction in a decentralized environment // 2021 Wave Electronics and its Application in Information and Telecommunication Systems, WECONF 2021 - Conference Proceedings, Saint-Petersburg, 2021. P. 9470710. DOI 10.1109/WECONF51603.2021.9470710. EDN RUMWBY.
4. *Shvedov A.V., Gadasin D.V., Alyoshintsev A.V.* Segment routing in data transmission networks // T-Comm: Телекоммуникации и транспорт. 2022. Vol. 16, No. 5. P. 56-62. DOI 10.36724/2072-8735-2022-16-5-56-62. EDN VAYLJQ.
5. *Zolotukhin P.A., Melkova E.K., Gadasin D.V., Korovushkina V.M.* Using Intelligent Testing as a Tool to Improve the Quality of Transport Information Systems // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 - Conference Proceedings, Moscow, Moscow, 2022. DOI 10.1109/IEEECONF53456.2022.9744348. EDN NOMJLX.

6. Шведов А.В., Гадасин Д.В., Коропушкина В.М., Мелькова Е.К. Интеллектуальное тестирование как способ повышения качества информационной системы // REDS: Телекоммуникационные устройства и системы. 2022. Т. 12, № 2. С. 43-52. EDN GOLZGE.
7. Gadasin D.V., Shvedov A.V., Klygina O.G., Shulginov I.V. Routing Management system formation for machine-to-machine interaction in a decentralized environment // 2021 Wave Electronics and its Application in Information and Telecommunication Systems, WECONF 2021 - Conference Proceedings, Saint-Petersburg, Saint-Petersburg, 2021. P. 9470710. DOI 10.1109/WECONF51603.2021.9470710. EDN RUM-WBY.
8. Gadasin, D. V. Cluster model for edge computing / D. V. Gadasin, A. V. Shvedov, A. V. Koltsova // 2020 International Conference on Engineering Management of Communication and Technology, EMCTECH 2020 - Proceedings, Vienna, 20–22 октября 2020 года. – Vienna, 2020. – P. 9261538. – DOI 10.1109/EMCTECH49634.2020.9261538. – EDN FGDLA.
9. Gadasin D.V., Koltsova A.V., Gadasin D.D. Algorithm for Building a Cluster for Implementing the 'Memory as a Service' Service in the IoT Concept // 2021 Systems of Signals Generating and Processing in the Field of on Board Communications, Conference Proceedings, Moscow, 2021. P. 9416112. DOI 10.1109/IEEECONF51389.2021.9416112. EDN VRPCFG.
10. Гадасин Д.В., Кольцова А.В., Полякова А.Н. Модель построения кластера для пограничных вычислений // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2020. № 1. С. 86-92. EDN NOUODG.
11. Гадасин Д.В., Юдина А.А. Кластеризация в крупномасштабных сетях // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2020. № 1. С. 19-26. EDN OYSXON.
12. Vakurin I.S., Gadasin D.V. Determining cluster distance when designing high - Density WI-FI // 2020 International Conference on Engineering Management of Communication and Technology, EMCTECH 2020 - Proceedings, Vienna, 2020. P. 9261565. DOI 10.1109/EMCTECH49634.2020.9261565. EDN ZSXXCK.
13. Gadasin D.V., Shvedov A.V., Koltsova A.V. Cluster model for edge computing // 2020 International Conference on Engineering Management of Communication and Technology, EMCTECH 2020 - Proceedings, Vienna, 2020. P. 9261538. DOI 10.1109/EMCTECH49634.2020.9261538. EDN FGDLA.
14. Статьев В.Ю., Докучаев В.А., Маклачкова В.В. Информационная безопасность на пространстве "Больших данных" // Т-Comm: Телекоммуникации и транспорт. 2022. Т. 16, № 4. С. 21-28. DOI 10.36724/2072-8735-2022-16-4-21-28. EDN IXUYWS.
15. Shvedov A.V., Gadasin D.V., Pak E.V. Application of the Backman Model for the Distribution of Traffic Flows in Networks with Segment Routing // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 - Conference Proceedings, Moscow, 2022. DOI 10.1109/IEEECONF53456.2022.9744344. EDN RBMTBQ.
16. Гадасин Д.В., Смальков Н.А., Кузин И.А. Использование метода роя частиц для балансировки нагрузки в сетях Интернета вещей // Системы синхронизации, формирования и обработки сигналов. 2022. Т. 13, № 2. С. 17-23. EDN LIUWNT.
17. Юдин Д.Б., Гольштейн Е.Г., Юдин Д.Б., Гольштейн Е.Г. Задачи и методы линейного программирования. Задачи транспортного типа. Изд. 3-е. М.: URSS, 2010. ISBN 978-5-397-01334-5. EDN QJWOTJ.
18. Минизка Э. Алгоритмы оптимизации на сетях и графах. М.: Мир, 1981.
19. Pavlov S.V., Dokuchaev V.A., Mytenkov S.S. Model of a fuzzy dynamic decision support system // Т-Comm: Телекоммуникации и транспорт. 2020. Vol. 14, No. 9. P. 43-47. DOI 10.36724/2072-8735-2020-14-9-43-47. EDN VYFNLB.
20. Заде Л.А. Основы нового подхода к анализу сложных систем и процессов принятия решений // Математика сегодня. М.: Знание, 1974. С. 5-49.
21. Беллман Р.Э., Заде Л.А. Принятие решений в расплывчатых условиях // Вопросы анализа и процедуры принятия решений. М.: Мир, 1976. С. 172-215.
22. Борисов А.Н., А.В. Алексеев, Меркурьев Г.В. Обработка нечеткой информации в системах принятия решений. М.: Радио и связь, 1989.
23. Заде Л.А. Понятие лингвистической переменной и его применение к принятию приближенных решений. М.: Мир, 1976. 165 с.

DISTRIBUTION OF INCOMING LOAD USING THE SS-METHOD

DENIS V. GADASIN

Moscow Technical University of Communication and Informatics, Moscow, Russia, dengadiplom@mail.ru

LILY A. TREMASOVA

Moscow Technical University of Communication and Informatics, Moscow, Russia, dengadiplom@mail.ru

DANIIL D. GADASIN

Moscow Technical University of Communication and Informatics, Moscow, Russia, dengadiplom@mail.ru

ABSTRACT

Introduction: The process of transferring information between two arbitrary applications is based on a seven-level OSI model. In the process of transmission, four main participants can be distinguished, the process can be presented in the form of a tuple, and each participant can be described by a set. **Methods:** To implement the transfer process, it is possible to apply an algorithm consisting of four steps: the formation of a database of participants, the definition of a data transmission scheme, the definition of a load distribution model, the definition of an algorithm for minimizing the objective function. To minimize the objective function, it is proposed to use one of the ways to solve the transport problem, which belongs to the type of mathematical linear programming problems of a special kind. It is proposed to solve the problem on the basis of the SS-method, and to make an initial reference plan using the method of the northwest corner. **Results:** The paper considers load balancing tasks for two cases. The minimization parameter, the "cost of transportation", is the time of information transfer between two nodes. In the first case, the load is distributed between the network nodes when the quality parameters are set to explicit values. For the second case, the quality parameters are set by interval values, within which they change, i.e. the task is reduced to a fuzzy form. An example of a solution for given values is given and the resulting algorithm for solving a balanced problem is determined.

Keywords: fuzziness, transport problem, balancing, SS-method, defasification.

REFERENCES

1. A. V. Shvedov, D. V. Gadasin, O. G. Klygina. Determining Shortest Paths between Two Arbitrary Nodes in a Composite Transport Network Using Segment Routing // 2021 Systems of Signals Generating and Processing in the Field of on Board Communications, Conference Proceedings, Moscow, Moscow, 2021. P. 9416072. DOI 10.1109/IEEECONF51389.2021.9416072.
2. N. S. Kalmykov, V. A. Dokuchaev. Segment routing as a basis for software defined network // T-Comm. 2021. Vol. 15, No. 7. P. 50-54. DOI 10.36724/2072-8735-2021-15-7-50-54.
3. D. V. Gadasin, A. V. Shvedov, O. G. Klygina, I. V. Shulginov. Routing Management system formation for machine-to-machine interaction in a decentralized environment // 2021 Wave Electronics and its Application in Information and Telecommunication Systems, WECONF 2021 - Conference Proceedings, Saint-Petersburg, 2021. P. 9470710. DOI 10.1109/WECONF51603.2021.9470710.
4. A. V. Shvedov, D. V. Gadasin, A. V. Alyoshintsev. Segment routing in data transmission networks // T-Comm. 2022. Vol. 16, No. 5. P. 56-62. DOI 10.36724/2072-8735-2022-16-5-56-62.
5. P. A. Zolotukhin, E. K. Melkova, D. V. Gadasin, V. M. Korovushkina. Using Intelligent Testing as a Tool to Improve the Quality of Transport Information Systems // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 - Conference Proceedings, Moscow, 2022. DOI 10.1109/IEEECONF53456.2022.9744348.
6. A.V. Shvedov, D. V. Gadasin, V. M. Korovushkina, E. K. Melkova. Intelligent testing as a way to improve the quality of an information system // REDS: Telecommunication devices and systems. 2022. V. 12, No. 2. P. 43-52.
7. D. V. Gadasin, A. V. Shvedov, O. G. Klygina, I. V. Shulginov. Routing Management system formation for machine-to-machine interaction in a decentralized environment // 2021 Wave Electronics and its Application in Information and Telecommunication Systems, WECONF 2021 - Conference Proceedings, Saint-Petersburg, 2021. P. 9470710. DOI 10.1109/WECONF51603.2021.9470710.
8. D. V. Gadasin, A. V. Shvedov, A. V. Koltsova. Cluster model for edge computing // 2020 International Conference on Engineering Management of Communication and Technology, EMCTECH 2020 - Proceedings, Vienna, 2020. P. 9261538. DOI 10.1109/EMCTECH49634.2020.9261538.
9. D. V. Gadasin, A. V. Koltsova, D. D. Gadasin. Algorithm for Building a Cluster for Implementing the 'Memory as a Service' Service in the IoT Concept // 2021 Systems of Signals Generating and Processing in the Field of on Board Communications, Conference Proceedings, Moscow, 2021. P. 9416112. DOI 10.1109/IEEECONF51389.2021.9416112.
10. D. V. Gadasin, A.V. Koltsova, A. N. Polyakova. Cluster construction model for boundary computing // Proceedings of the North Caucasus Branch of the Moscow Technical University of Communications and Informatics. 2020. No. 1. P. 86-92.
11. D. V. Gadasin, A. A. Yudina. Clustering in large-scale networks // Proceedings of the North Caucasus Branch of the Moscow Technical University of Communications and Informatics. 2020. No. 1. P. 19-26.
12. I. S. Vakurin, D. V. Gadasin. Determining cluster distance when designing high - Density WI-FI // 2020 International Conference on Engineering Management of Communication and Technology, EMCTECH 2020 - Proceedings, Vienna, 2020. P. 9261565. DOI 10.1109/EMCTECH49634.2020.9261565.
13. D. V. Gadasin, A. V. Shvedov, A. V. Koltsova. Cluster model for edge computing // 2020 International Conference on Engineering Management of Communication and Technology, EMCTECH 2020 - Proceedings, Vienna, 2020. P. 9261538. DOI 10.1109/EMCTECH49634.2020.9261538.
14. Statev V. Yu., Dokuchaev V. A., Maklachkova V. V. Information security in the big data space. T-Comm, 2022, vol. 16, no.4, pp. 21-28.
15. A. V. Shvedov, D. V. Gadasin, E. V. Pak. Application of the Backman Model for the Distribution of Traffic Flows in Networks with Segment Routing // 2022 Systems of Signals Generating and Processing in the Field of on Board Communications, SOSG 2022 - Conference Proceedings, Moscow, 2022. DOI 10.1109/IEEECONF53456.2022.9744344.

16. D. V. Gadasin, N. A. Smalkov, I. A. Kuzin. Using the particle swarm method for load balancing in Internet of Things networks // Synchronization systems, signal generation and processing. 2022. Vol. 13, No. 2, pp. 17-23.
17. D. B. Yudin, E. G. Holstein. Tasks and methods of linear programming. Transport type tasks. Moscow: URSS, 2010. ISBN 978-5-397-01334-5.
18. E. Minięka. Optimization algorithms on networks and graphs. Moscow: Mir, 1981.
19. S. V. Pavlov, V. A. Dokuchaev, S. S. Mytenkov. Model of a fuzzy dynamic decision support system // T-Comm. 2020. Vol. 14, No. 9. P. 43-47. DOI 10.36724/2072-8735-2020-14-9-43-47.
20. L. A. Zadeh. Fundamentals of a new approach to the analysis of complex systems and decision-making processes // Mathematic Today. Moscow: Znanie, 1974, pp. 5-49.
21. R. E. Bellman, L. A. Zadeh. Decision-making in vague conditions // Issues of analysis and decision-making procedures. Moscow: Mir, 1976, pp. 172-215.
22. A. N. Borisov, A. V. Alekseev, G. V. Merkuriev. Processing of fuzzy information in decision-making systems / M.: Radio i svyaz, 1989.
23. L. A. Zadeh. The Concept of a Linguistic Variable and Its Application to Approximate Reasoning. Moscow: Mir, 1976. 165 p.