Модернизация транспортных сетей связи методом наложения с использованием минимальных связывающих деревьев Штейнера

Transport communication networks modernization by overlay method using minimal Steiner connecting trees

Зюзин / Zyuzin A.
Александр Николаевич
(alexz01@bk.ru)
кандидат технических наук.
ФГКВОУ ВО «Военная академия связи имени Маршала Советского Союза С. М. Буденного» (ВАС им. С. М. Буденного) МО РФ, преподаватель.
г. Санкт-Петербург

Репин / Repin B.
Борис Григорьвич
(rbg@inbox.ru)
кандидат военных наук,
ВАС им. С. М. Буденного, доцент.
г. Санкт-Петербург

Ясинский / Yasinskii S.
Сергей Александрович
(yasinsky777@mail.ru)
доктор технических наук, доцент.
Филиал ФГУП «Ленинградское отделение центрального научно-исследовательского института связи», старший научный сотрудник.
г. Санкт-Петербург

Ключевые слова: телекоммуникационная система – telecommunications system; транспортная сеть связи – communication transport network; метод наложения – overlay method; взвешенное покрытие – weighted coverage; дерево Штейнера – Steiner tree; библиотека NetworkX – NetworkX library.

В статье описывается подход к применению метода наложения для модернизации фрагмента транспортной сети связи. При определении участков сети для установки нового оборудования требуется решить задачу поиска множества линий связи, которое будет иметь минимальную общую протяженность и будет связывать корреспондирующие узлы между собой. Математическое решение этой задачи сводится к поиску минимального дерева Штейнера, связывающего заданный набор узлов. Для автоматизации процесса расчёта предлагается использовать возможности библиотеки NetworkX для языка программирования Python. Приводится порядок расчета, фрагменты программного кода и пример расчета.

The article describes an approach to the application of the overlay method for the modernization of a fragment of a transport communication network. When determining the network sections for installing new equipment, it is necessary to solve the problem of finding a set of communication lines that will have a minimum total length and will connect the corresponding nodes with each other. The mathematical solution of this problem is reduced to finding a minimal Steiner tree connecting a given set of nodes. To automate the calculation process, it is proposed to use the capabilities of the NetworkX library for the Python programming language. The calculation procedure, fragments of the program code and an example of the calculation are given.

Введение

Цифровизация повседневной жизни современного человека постоянно увеличивает объемы обрабатываемой информации и передаваемые информационные потоки. Активно развиваются различные инфо-телекоммуникационные системы. Новые потребности интеллектуального сбора и обмена больших объемов данных требуют построения высокопроизводительных и надежных телекоммуникационных систем (ТКС). Проблемы построения новых, а также модернизации и развития существующих ТКС в части их транспортных составляющих (транспортных или первичных сетей связи) в настоящий момент приобретают ключевой характер. Это объясняется тем, что транспортная компонента ТКС определяет потенциальные и реально достижимые показатели функционирования и эксплуатации ТКС в целом [1].

Транспортные сети связи (ТрСС) относятся к консервативной и самой дорогостоящей составляющей ТКС. Это подтверждается высокой стоимостью строительства (развертывания) и большим периодом функционирования и эксплуатации (десятки лет) ТрСС, в течение которого сменяется несколько поколений оконечных устройств пользователей, видов информационных сервисов и подходов к их реализации с учетом новых, более перспективных информационных технологий [2].

Известно, что развитие ТрСС может осуществляться в двух основных направлениях:

– модернизация и развитие участков существующих сетей;

– построение новых участков сети.

Для построения новых участков ТрСС используются известные методы, методики и алгоритмы системного проектирования (синтеза), которые в основном использовались для построения аналоговых и цифровых плезиохронных сетей связи с последующим совершенствованием и модификацией применительно к построению более перспективных ТКС и их транспортных компонент.

Как показывает мировой опыт, для модернизации существующих участков первичных сетей связи и ТрСС может быть использовано сочетание методов замещения и наложения [1, 2, 3].

Метод замещения заключается в постепенной замене устаревшего оборудования ТрСС на существующих линиях передачи. Это длительный (многолетний) эволюционный процесс, который зависит от наличия инвестиционных возможностей владельцев ТКС и от текущих потребностей пользователей ТКС.

Метод наложения (МН) подразумевает строительство современных ТрСС на базе существующей инфраструктуры устаревших сетей связи без вывода их из эксплуатации.

Рассмотрим более подробно возможности использования МН при модернизации ТрСС [1].

Постановка задачи модернизации ТрСС методом наложения

Метод наложения обеспечивает модернизацию устаревших ТрСС путем построения сетей связи нового поколения, которые способны функционировать на базе ранее созданных направляющих систем без дополнительных затрат на их замену методом замещения. Этот процесс может проходить поэтапно на выбранных участках сети связи и не влияет на работу устаревшего оборудования. Для реализации МН могут использоваться:

– свободные оптические волокна в проложенных ранее волоконно-оптических кабелях;

– резервные и свободные стволы радиорелейных систем передачи;

– технология спектрального уплотнения каналов для существующих волоконно-оптических линий связи (ВОЛС);

– «пакетные» каналы передачи в групповых каналах и трактах цифровых систем передачи.

Важно отметить, что технология спектрального уплотнения каналов уже давно используется на ТрСС как наиболее простое средство модернизации участков существующих ВОЛС ТрСС. Однако использовать технологию спектрального уплотнения каналов для модернизации всех участков ВОЛС не целесообразно. Сформулируем оптимизационную задачу модернизации ТрСС методом наложения с учетом минимизации капитальных затрат [4].

Для формализации задачи модернизации ТрСС методом наложения будем использовать известные элементы теории графов. Структуру фрагмента ТрСС ТКС смоделируем при помощи ненаправленного графа $G = \{A, B, L\}$ В этом графе множество вершин $A = \{a_i; i = \overline{1, N}\}$ соответствует множеству узлов связи фрагмента ТрСС с общим числом N, а множество ребер связи $B = \{b_{ij}; i, j = \overline{1, N}; i \neq j\}$ описывает линии передачи между этими узлами, на котором определены значения длин линий передачи $\{l_{ij}; i, j = \overline{1, N}; i \neq j\}$. Кроме того, обозначим множество корреспондирующих пар узлов (КПУ) $Z = \{z_m; m = \overline{1, M}\}$, где M – общее количество КПУ [4].

Для решения задачи требуется из множества ребер связи B найти подмножество ребер $B' = \{b'_{ij}\}$, которое будет иметь минимальную общую протяженность, снижая расходы на модернизацию и связывая КПУ между собой. Известен подход, который формулируется как задача линейного булева программирования – о минимальном взвешенном покрытии множества возможных сечений путей различных направлений связи (КПУ) минимальным количеством ребер сети. Задача сводится к поиску множества допустимых сечений между КПУ и определению взвешенного покрытия множества сечений [4].

Решение задачи модернизации ТрСС методом наложения как задачи поиска минимального дерева Штейнера

Задача модернизации ТрСС МН может быть решена как задача поиска минимального дерева Штейнера, связывающего заданный набор узлов:

$$\sum\nolimits_{b'_{ij} \in B'} b'_{ij} l_{ij} \Rightarrow \min ; \quad (1)$$

$$b'_{ij} = \begin{cases} 1, \text{ если } b'_{ij} \in B' \\ 0 \text{ в противном случае} \end{cases} ; \quad (2)$$

$$T = \{Z, B'\}, Z \in A, \ B' \in B . \quad (3)$$

Решение этой оптимизационной задачи приводит к определению топологии модернизируемых линий связи и узлов связи ТрСС средствами технологии спектрального уплотнения каналов, которая соответствует минимальному связывающему дереву Штейнера в виде графа $T = \{Z, B'\}$.

Известно, что задача поиска минимального дерева Штейнера является NP-трудной в решении, поэтому предлагается возможность ее решения приближенным алгоритмом в два этапа: сначала решать задачу поиска кратчайших путей для всех сочетаний КПУ, а затем решать задачу поиска минимального связывающего дерева на полученном дереве кратчайших путей [5, 6].

Для снижения трудоемкости расчетов предлагается использовать возможности современных языков программирования. Например, для популярного языка

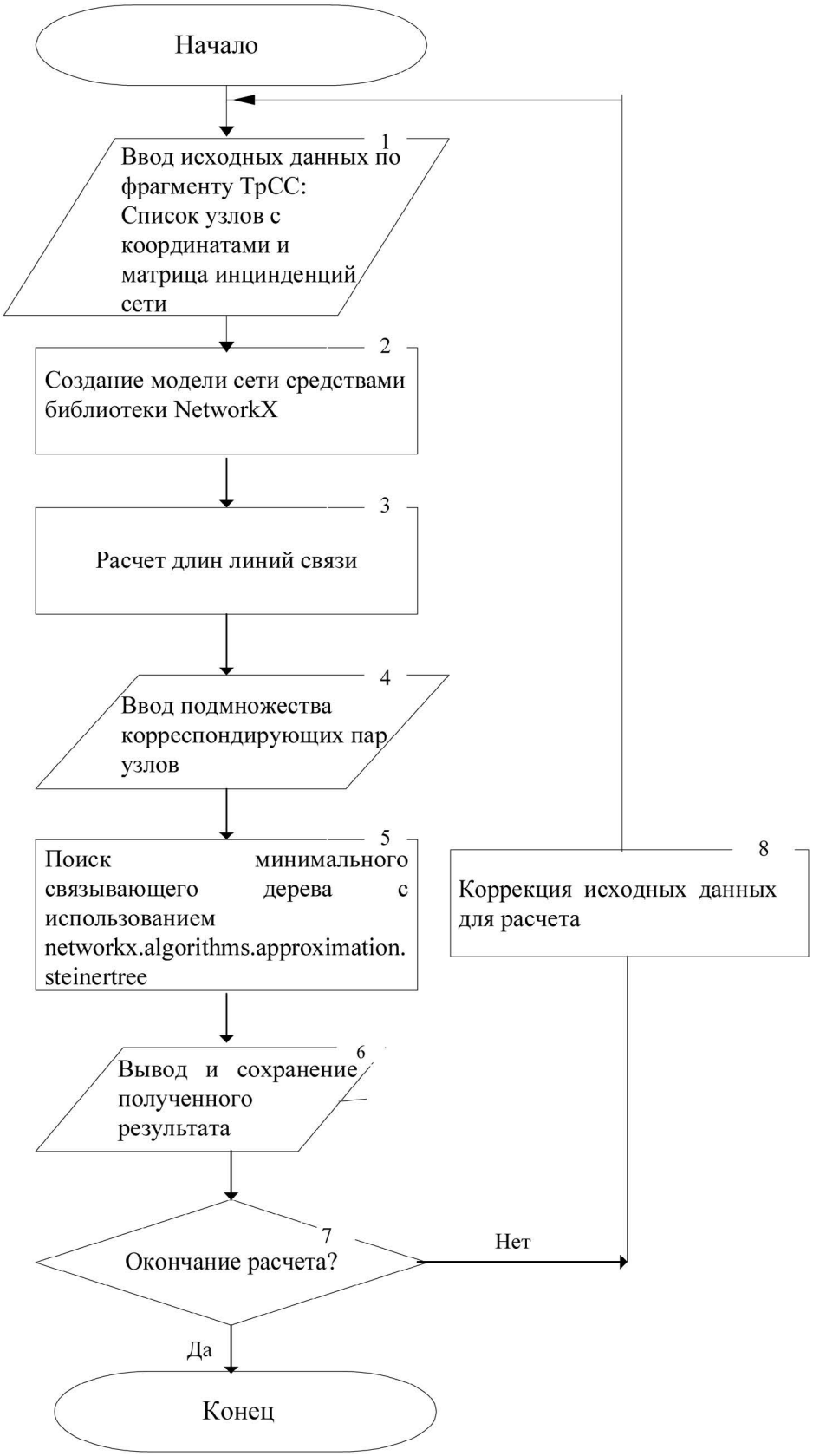


Рис. 1. Упрощенная блок-схема решения задачи по определению топологии модернизируемых линий связи ТрСС средствами пакета библиотек NetworkX для Python

```
73  # Создаю граф
74  G = nx.Graph()
75  # Добавляю в граф сформированный список узлов
76  G.add_nodes_from(Nodes)
77  # Добавляю в граф сформированный список линий
78  G.add_edges_from(Edges)
```

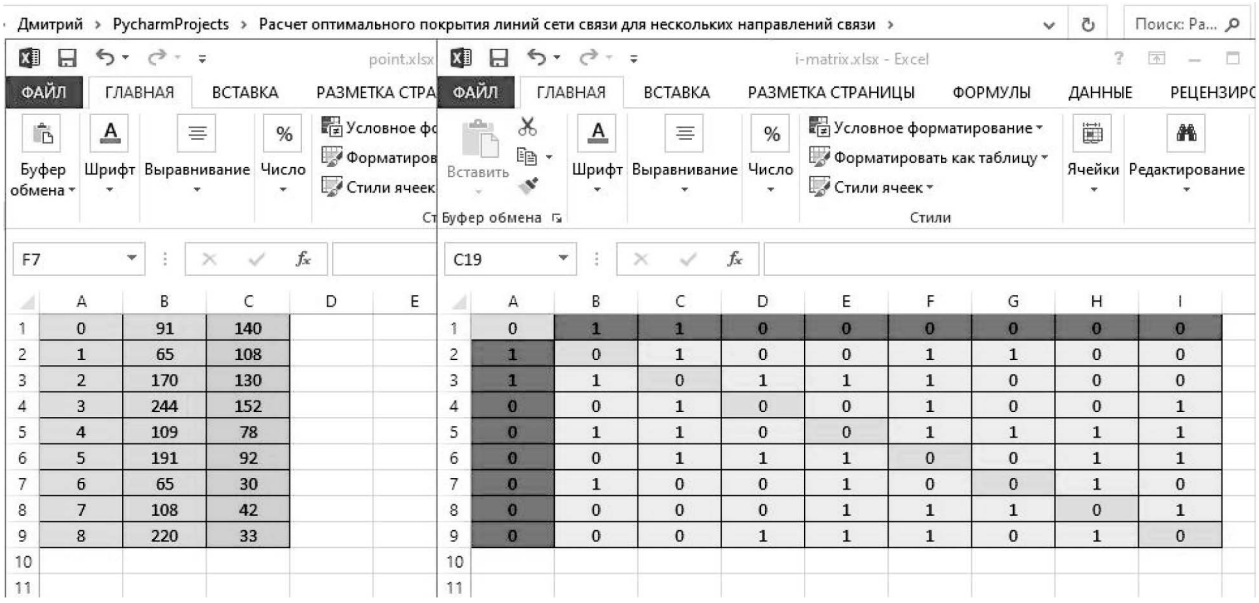
Рис. 2. Фрагмент программного кода для создания модели сети

```
80  # расчет длин линий
81  Lines = []
82  for edge in G.edges:
83      distance = sqrt((Nodes[edge[0]][0] - Nodes[edge[1]][0]) ** 2 + (Nodes[edge[0]][1] - Nodes[edge[1]][1]) ** 2)
84      # расчет длин ребер между узлами
85      line = (edge[0], edge[1], distance)
86      Lines.append(line)
87  G.add_weighted_edges_from(Lines)
88  # добавляем длины линий в граф
```

Рис. 3. Фрагмент программного кода для расчета протяженности линий передачи

```
100  # находим список линий составляющих минальное дерево Штейнера для корреспондирующих пар узлов
101  edge_list = list(nx.algorithms.approximation.steinertree.steiner_tree(G, direct_node).edges)
```

Рис. 4. Фрагмент программного кода для расчета дерева Штейнера



	A	B	C
1	0	91	140
2	1	65	108
3	2	170	130
4	3	244	152
5	4	109	78
6	5	191	92
7	6	65	30
8	7	108	42
9	8	220	33

	A	B	C	D	E	F	G	H	I
1	0	1	1	0	0	0	0	0	0
2	1	0	1	0	0	1	1	0	0
3	1	1	0	1	1	1	0	0	0
4	0	0	1	0	0	1	0	0	1
5	0	1	1	0	0	1	1	1	1
6	0	0	1	1	1	0	0	1	1
7	0	1	0	0	1	0	0	1	0
8	0	0	0	0	1	1	1	0	1
9	0	0	0	1	1	1	0	1	0

Рис. 5. Exel таблицы с исходными данными для расчета минимального дерева Штейнера

```
Укажите количество направлений связи для расчета
Введите число: 2
Укажите первый узел корреспондирующей пары №0
Перечень возможных Узлов ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10']
Укажите узел: 0
Укажите втрой узел корреспондирующей пары №0
Перечень возможных Узлов ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10']
Укажите узел: 8
Укажите первый узел корреспондирующей пары №1
Перечень возможных Узлов ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10']
Укажите узел: 3
Укажите втрой узел корреспондирующей пары №1
Перечень возможных Узлов ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10']
Укажите узел: 6
```

Рис. 6. Фрагмент диалогового окна с запросом КПУ

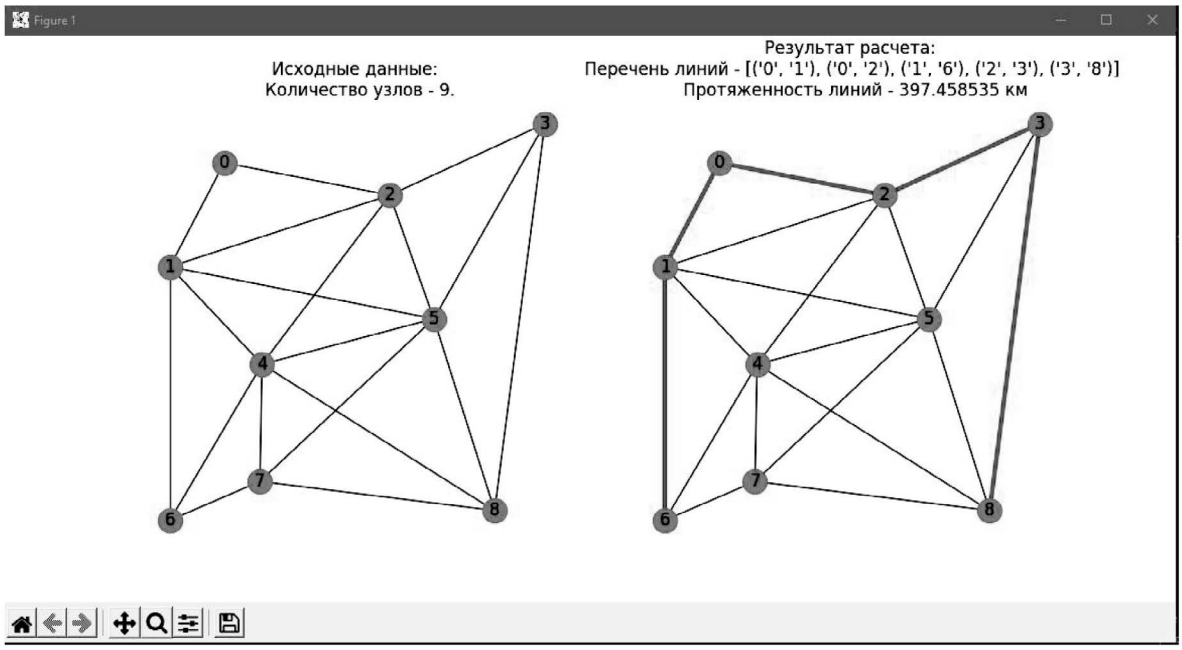


Рис. 7. Результат расчета минимального дерева Штейнера средствами пакета библиотек NetworkX для Python

программирования Python существует пакет библиотек NetworkX, который возможно использовать для изучения структуры, динамики и функционирования сложных сетей. Этот пакет может работать со структурами, представленными в виде графов.

При использовании NetworkX имеется возможность загружать и хранить модели сложных сетей, анализировать сетевую структуру, строить модели сетей, разрабатывать новые сетевые алгоритмы и графически отображать полученные результаты [7].

В составе пакета библиотек NetworkX для языка программирования Python имеется модуль networkx.algorithms.approximation.steinertree, который, используя приближенный алгоритм, позволяет найти минимальное дерево Штейнера для подмножества КПУ на топологической структуре в виде графа. На входе модуль получает граф сети в формате NetworkX и список КПУ, а на выходе выдает список ребер графа сети, составляющий минимальное дерево Штейнера [8].

Упрощенная блок-схема решения задачи по определению топологии модернизируемых линий передачи и ТрСС средствами пакета библиотек NetworkX для Python представлена на рис. 1.

На первом шаге нужно сформировать список узлов связи с их координатами в двухмерной плоскости и список линий передачи с атрибутами номера узла начала и узла окончания линии. Далее на основе сформированных списков необходимо создать модель сети в формате NetworkX. Фрагмент программного кода для создания модели сети связи представлен на рис. 2 [9].

Расчет протяженности линий связи на двумерной плоскости реализуется с использованием теоремы Пифагора, а фрагмент программного кода для расчета протяженности линий передачи приведен на рис. 3 [9].

Перечень КПУ оформляется в виде списка узлов связи, а затем, используя программный модуль networkx.algorithms.approximation.steinertree, подготовленную модель графа сети и список КПУ, получаем список линий передачи, составляющий минимальное дерево Штейнера. Фрагмент программного кода для расчета дерева Штейнера приведен на рис. 4 [9].

На последнем этапе расчета для графического вывода полученного результата воспользуемся средствами библиотеки matplotlib с сохранением полученного списка линий передачи.

Пример расчета минимального дерева Штейнера для модернизации ТрСС методом наложения

Рассмотрим пример расчета минимального дерева Штейнера средствами пакета библиотек NetworkX для Python для модернизации ТрСС методом наложения.

Пусть координаты 9 узлов связи и топология фрагмента ТрСС заданы в виде Exel таблиц – рис. 5. Так же известны 2 пары КПУ: узел «0» - узел «8», узел «3» - узел «6».

Исходные данные считываются средствами библиотеки openpyxl и преобразуются в список узлов связи и список линий передачи. Далее в диалоговом режиме определяются КПУ – рис. 6.

После задания КПУ получаем результат расчета минимального дерева Штейнера, который графически отображается средствами библиотеки matplotlib – рис. 7.

Таким образом, кроме получения общей протяженности линий передачи получено минимальное дерево Штейнера для пары КПУ («0»-«8», «3»-«6»), которое состоит из линий передачи: (0,1), (0,2), (1,6), (2,3), (3,8).

Заключение

Для решения задачи модернизации ТрСС предложен подход к ее решению методом наложения как задачи поиска минимального дерева Штейнера. В основу этого подхода положено применение метода наложения для принятия решения на модернизацию и развитие ТрСС, а также применения средств моделирования и расчета NetworkX для Python, что позволяет с минимальными затратами на расчет и оптимально с точки зрения образования необходимых направлений связи (КПУ) производить наращивание пропускных способностей систем передачи транспортной составляющей ТКС.

Рассмотренный пример расчета минимального дерева Штейнера для модернизации ТрСС методом наложения наглядно показывает возможность практического применения пакета библиотек NetworkX для Python с целью решения задач сетевой оптимизации в терминах теории графов.

Литература

1. Грязев, А. Н. Методы развития транспортных сетей связи для цифровой экономики РФ / А.Н. Грязев, С.А. Ясинский, А.Н. Зюзин // Вестник связи. – 2018. – № 2. – С. 19 – 22.

2. Грязев, А. Н. Многослойная графовая модель для модернизации и развития транспортных сетей связи / А.Н. Грязев, А.Н. Зюзин, С.А. Ясинский // Вестник связи. – 2018. – № 5. – С. 13 – 16.

3. Рунеев, А. Ю. Концепция ТСС. Первичные сети связи. Концепция создания и развития первичной сети территориальной системы связи / А.Ю. Рунеев, Н.А. Зюзин // Телекоммуникационные технологии. – 2001. – Вып. 1. – С. 3–13.

4. Сторожук, Н. Л. Формализованный подход к применению метода наложения для принятия решения на модернизацию и развитие транспортных сетей связи / Н.Л. Сторожук, А.Н. Зюзин, С.А. Ясинский // Информация и Космос. – 2018. – № 4. – С. 15–19.

5. Иванов, А. О. Задача Штейнера на плоскости или плоские минимальные сети / А.О. Иванов, А.А. Тужилин //

Математический сборник. – 1991. – Т. 182, № 12. – С. 1813–1844.

6. Минимальное остовное дерево – Minimum spanning tree [Электронный ресурс] : Википедия. Свободная энциклопедия. – Режим доступа: https://ru.abcdef.wiki/wiki/Minimum_spanning_tree [дата обращения 30.05.22], свободный. – Загл. с экрана.

7. NetworkX is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks [Электронный ресурс] : библиотека Python. – Режим доступа: https://networkx.org/ [дата обращения 30.05.22], свободный. – Загл. с экрана.

8. Source code for networkx.algorithms.approximation.steinertree [Электронный ресурс] : библиотека Python. – Режим доступа: https://networkx.org/documentation/stable/_modules/networkx/algorithms/approximation/steinertree.html#steiner_tree [дата обращения 30.05.22], свободный. – Загл. с экрана.

9. Свидетельство об официальной регистрации программы для ЭВМ № 21021 Российская Федерация. Расчет оптимального покрытия линий сети связи для нескольких направлений связи / Зюзин А.Н., Зюзин Д.Н., Малюков К.А.; Военная академия связи. – 06.04.2021.