DOI: 10.15514/ISPRAS–2020–33(2)–10

Решение проблемы обеспечения качества дерева многоадресной рассылки услуг

К. Риссо, ORCID: 0000-0003-0580-3083 <crisso@fing.edu.uy>
Ф. Робледо, ORCID: 0000-0003-4235-4221<frobledo@fing.edu.uy>
С. Несмачнов, ORCID: 0000-0002-8146-4012 <sergion@fing.edu.uy>

Республиканский университет,
Уругвай, 11200, Монтевидео, ул. 18 июля 1824-1850

Аннотация. В данной статье представлена основанная на потоках формулировка проблемы обеспечения качества дерева многоадресной рассылки услуг в терминах смешанного целочисленного программирования. Это актуальная проблема, связанная с современными телекоммуникационными сетями, обеспечивающие распространение мультимедийного контента через облачные Internet-системы. Насколько нам известно, для проблемы обеспечения качества дерева многоадресной рассылки услуг формулировка в терминах смешанного целочисленного программирования ранее не предлагалась. Экспериментальная оценка выполняется на наборе реалистичных примеров из SteinLib, чтобы показать применимость стандартных точных решателей для нахождения решений реальных задач. Точный метод применяется для бенчмаркинга предлагаемых формулировок, а также для поиска оптимальных или близких к оптимальным решений за приемлемое время исполнения.

Ключевые слова: многоадресная рассылка; качество; дерево многоадресной рассылки услуг; целочисленное программирование

Для цитирования: Риссо К., Робледо Ф., Несмачнов С. Решение проблемы обеспечения качества дерева многоадресной рассылки услуг. Труды ИСП РАН, том 33, вып. 2, 2021 г., стр. 163-172. DOI: 10.15514/ISPRAS–2021–33(2)–10

Solving the Quality of Service Multicast Tree Problem

C. Risso, ORCID: 0000-0003-0580-3083 <crisso@fing.edu.uy>
F. Robledo, ORCID: 0000-0003-4235-4221<frobledo@fing.edu.uy>
S. Nesmachnow, ORCID: 0000-0002-8146-4012 <sergion@fing.edu.uy>

Universidad de la Republica Uruguay,
Av. 18 de Julio 1824-1850, Montevideo, 11200, Uruguay

Abstract. This article presents a flow-based mixed integer programming formulation for the Quality of Service Multicast Tree problem. This is a relevant problem related to nowadays telecommunication networksto distribute multimedia over cloud-based Internet systems. To the best of our knowledge, no previous mixed integer programming formulation was proposed for Quality of Service Multicast Tree Problem. Experimental evaluation is performed over a set of realistic problem instances from SteinLib, to prove that standard exact solvers can find solutions to real-world size instances. Exact method is applied for benchmarking the proposed formulations, finding optimal solutions and low feasible-to-optimal gaps in reasonable execution times.

Keywords: multicasting; Quality; Service Multicast Tree; Integer Programming

For citation: Risso C., Robledo F., Nesmachnow S. Solving the Quality of Service Multicast Tree Problem. Trudy ISP RAN/Proc. ISP RAS, vol. 33, issue 2, 2021, pp. 163-172 (in Russian). DOI: 10.15514/ISPRAS–2021–33(2)–10.

1. Введение

В этой статье исследуется актуальная проблема проектирования сети: проблема обеспечения качества дерева многоадресной рассылки услуг (Quality of Service Multicast Tree Problem, QoSMTP). QoSMTP относится к построению оптимального дерева для многоадресной передачи и доставки мультимедийного контента с использованием минимального ожидаемого количества ресурсов полосы пропускания, например, программно-определяемых сетей (Software Defined Network, SDN).

Основным вкладом исследования, изложенного в данной статье, является новая формулировка QoSMTP в терминах смешанного целочисленного программирования (Mixed-Integer Programming, MIP), основанная на потоковой формулировке Гёманса (Michel Goemans) и Мюна (Young-Soo Myung) [1] проблемы дерева Штейнера (Steiner Tree Problem – STP). При экспериментальной оценке применяется точный метод для решения экземпляров из широко известной библиотеки SteinLib, модифицированных соответствующим образом, с целью доказать, что стандартные точные решатели могут находить решения реальных задач.

Результаты экспериментов указывают на то, что оптимальные и близкие к оптимальным решения вычисляются за приемлемое время. Наиболее эффективные существующие решения для задачи QoSMTP вычисляются с использованием приближенных алгоритмов, коэффициенты которых выше 3/2 [2]; таким образом, предлагаемый подход и результаты являются непосредственным вкладом в данное направление исследований.

Статья имеет следующую структуру. В разд. 2 приводится описание QoSMTP. Описание предлагаемой модели для задачи QoSMTP представлено в разд. 3. Результаты экспериментальной оценки на основе стандартных экземпляров из репозитория SteinLib изложены в разд. 4. В заключение, в разд. 5 представлены выводы и дальнейшие перспективы исследования.

2. Проблема обеспечения качества дерева многоадресной рассылки услуг

QoSMTP формулируется следующим образом. Пусть $G = (V, E, l, r)$ – неориентированный взвешенный граф, где V – множество узлов, а E – множество ребер. Также рассматривается подмножество $T \subseteq V$ выделенных терминальных вершин, одна из которых является корнем ($R \in T$). Цель состоит в том, чтобы найти подграф остовного дерева с минимальной стоимостью, который должен содержать все терминальные узлы. Основное отличие от STP в данном случае заключается в функции стоимости, так как оцениваются и узлы, и ребра. Каждый терминальный узел имеет вес от 0 до r_N, определяющий значимость данного терминального узла в сети. Функция $l: E \to R^+$ представляет длину каждого ребра, а функция $r: V \to R_0^+$ выдает вес каждого ребра. В этой проблеме корневой узел R представляет собой центральный узел проектируемой сети (например, узел, в котором находится весь доступный контент, который планируется совместно использовать / рассылать по многим адресам).

QoSMTP предполагает нахождение поддерева с минимальной стоимостью $F = (V_F, E_F)$ графа G, растущего от заданного корня R и покрывающее множество терминальных узлов. Узлы, не принадлежащие T, как и STP, называются узлами Штейнера; они представляют собой узлы с нулевым весом, для которых не требуется соединение с корневым узлом, но которые можно использовать для уменьшения стоимости решения.

Что касается функции стоимости, то основная идея заключается в том, что чем выше стоимость, тем больше число различных копий информационных единиц, которые должны быть переданы в данный узел от корня. Таким образом, стоимость непосредственного соединения на пути к корню многоадресного дерева должна быть умножена на некоторое число, в котором учитываются эти веса (r_i).

Известным свойством деревьев является то, если имеются два дерева $T_1 = (V_1, E_1)$, $T_2 = (V_2, E_2)$ и две произвольные вершины $n_1 \in V_1$, $n_2 \in V_2$, то результатом соединения этих графов ребром (n_1, n_2) также является дерево. Обратное также верно, так как в результате удаления любого ребра из любого нетривиального дерева (т.е. дерева, содержащего не менее одного ребра), образуются два дерева, которые, тем самым, являются двумя независимыми и связанными подкомпонентами.

Стоимость решения F частного случая (экземпляра) G QoSMTP определяется путем сложения стоимости $c(e)$ каждого ребра e в дереве F, где $c(e) = l(e) \times \bar{r}_e$. Выражения $l(e)$ являются частью самого примера, в то время как значения $\bar{r}_e$ зависят от конкретного решения.

Вычисления $\bar{r}_e$ производятся следующим образом: для ребра $e = (v_1, v_2) \in F$ без потери общности предположим, что v_2 находится в пути от v_1 к корню R или v_2 является самим корнем. После удаления e из F получаются два поддерева F_1 и F_2, при этом v_1 находится в F_1, а v_2 и корень (если они разные) – в F_2. Тогда значение $\bar{r}_e$ является максимальным среди всех r_i, $i \in F_1$.

Компоненты целевой функции QoSMTP зависят от самого решения, что значительно усложняет эту проблему по сравнению с STP. Насколько нам известно, до сих пор QoSMTP не формулировалась в терминах MIP, а существующие решения основаны на аппроксимационных алгоритмах, коэффициенты которых выше 3/2 [3]. Формально, как STP, так и QoSMTP являются NP-трудными задачами. STP является классической NP-трудной задачей [4], сводимой к QoSMTP за полиноминальное время (STP является частным случаем QoSMTP, когда веса всех узлов равны 1). Таким образом, в вычислительном отношении задача QoSMTP является не менее трудной, чем STP.

3. Предлагаемая MIP-модуль для QoSMTP

Предлагаемая модель основана на потоковой формулировке Гёманса и Мюна [1] для решения STP. Основная идея предлагаемой модели заключается в следующем. Для удовлетворения требования балансировки потока для каждого узла в графе $G = (V, E)$, за исключением одного (т.е. корня R), при вводе единицы трафика в каждую терминальную вершину T, за исключением корня, должен существовать путь от каждого терминального узла к R для слива этой единицы потока; таким образом, результат должен быть связным. Предположим, что найдено оптимальное решение G_T. Реализуемость при соблюдении ограничения потока гарантируют наличие связности. Кроме того, поскольку ребра имеют положительные веса, а множество ребер, используемое для маршрутизации этих потоков, должно иметь минимальную стоимость, решение будет минимально связным; следовательно, получаемый граф является деревом.

Известным фактом теории потоков в сетях является то, что если параметры являются целыми числами, то крайние точки многогранника, определяемого уравнениями баланса и ограничений потока, также являются целыми числами [5]. Этот результат позволяет представлять целочисленные потоковые проблемы как задачи линейного программирования, что ограничивает оптимум замкнутой выпуклой оболочкой линейной целевой функции. Существование алгоритмов с полиномиальным временем для решения задач линейного программирования [6] уменьшает сложность чисто целочисленных потоковых проблем до сложности класса P.

Для правильного учета целевой функции STP необходимо также принимать во внимание некоторые целочисленные переменные и ограничения. Наличие этих переменных подрывает предыдущие рассуждения о сложности, перемещая проблему в категорию NP-трудных проблем. Однако целостность потоковых переменных можно, по меньшей мере, ослабить. Это не меняет теоретическую сложность проблемы, но облегчает ее решение за счет уменьшения количества целочисленных переменных и полиномиальной сложности

проблемы после присвоения целочисленным переменным возможных значений (как это делается, например, в методах ветвей и границ, используемых точными решателями).

Любой экземпляр STP определяется неориентированным взвешенным графом $G = (V, E, c)$, где V – множество вершин, E – множество потенциальных ребер (т.е. тех, которые допустимы в решении), а функция $c: E \to R^+$ определяет стоимость каждого ребра. Кроме того, для обеспечения замкнутости экземпляра проблемы требуется подмножество $T \subseteq V$ терминальных вершин. Без потери общности в модели предполагается, что одной из терминальных вершин $R \in T$ должен быть корень создаваемого дерева. Этот выбор является произвольным, но он упрощает формулировку STP и позволяет легко построить формулировку QoSMTP, так как для полного определения экземпляра второй проблемы требуется наличие фиксированного корня.

Для $G = (V, E, c)$ возьмем ориентированный граф $G' = (V, E', c')$, который получается в результате дублирования каждого ребра E в G с использованием обоих направлений, за исключением только тех ребер, вторая узлом которых является R. Это означает, что если $\{ij\} \in E$ и $i, j \neq R$, то $(ij) \in E'$ и $(ji) \in E'$. Кроме того, если $\{iR\} \in E$, то только (iR) находится в E'. Перечисленные ребра – это единственные ребра G'. Наконец, $c': E' \to R^+$ определяется как $c'(\{ij\}) = c(\{ij\})$. Соотношения (3.1)-(3.3) задают формулировку потоковой подпроблемы:

$$\begin{cases} \sum\limits_{(ij)\in E'} x_{ij} - \sum\limits_{(ki)\in E'} x_{ki} = 1 & \forall i \in T \backslash \{R\}, \qquad (3.1) \\ \sum\limits_{(ij)\in E'} x_{ij} - \sum\limits_{(ki)\in E'} x_{ki} = 0 & \forall i \in S, \qquad (3.2) \\ 0 \leq x_{ij} & \forall (ij) \in E'. \qquad (3.3) \end{cases}$$

В соотношениях (3.1)-(3.3) переменные $x_{ij} (ij \in E')$ учитывают поток от i по направлению к j. Таким образом, соотношение (3.1) просто выражают тот факт, что баланс сетевого потока равен 1 для каждого терминального узла i, кроме корня, что эквивалентно вводу в эти вершины единицы трафика. Множество S содержит все узлы Штейнера: $S = V \backslash T$. Соотношение (3.2) устанавливают баланс потока, равным 0, для узлов Штейнера, так как разница между суммарными исходящими и входящими потоками должна быть равна 0. Наконец, соотношение (3.3) устанавливает нижние пределы значений потока. Кроме того, потоки не могут быть больше или равны $|T|$, поскольку суммарный поток, введенный в граф, совпадает с $|T| - 1$ (количество терминальных вершин, за исключением корня), а значит, это является объемом суммарного потока, протекающего через R. Это ограничение задается отдельно в соотношениях (3.4) и (3.5):

$$\begin{cases} (|T| - 1) y_{ij} \geq x_{ij} & \forall (ij) \in E' \qquad (3.4) \\ y_{ij} \in \{0,1\} & \forall \{ij\} \in E \qquad (3.5) \end{cases}$$

В соотношениях (3.4) и (3.5) вводятся булевские переменные y_{ij}, чтобы показать, входит ли в решение ребро $(ij) \in E'$. Соотношения (3.1)–(3.3) и (3.4) задают классический набор ограничений для потоковой проблемы, область допустимых решений которой – это выпуклая оболочка, что упрощает нахождение значений x_{ij}. Число переменных x_{ij} совпадает с числом переменных y_{ij}, поэтому объем вычислений уменьшается. Полная MIP-формулировка для STP объединяет уравнения (3.1)–(3.5) с целевой функцией $min \sum_{(ij)\in E} c_{ij} y_{ij}$, которая обеспечивает суммарную стоимость решения в соответствии со значениями y_{ij}.

Значения y_{ij} должны обеспечить достижимость решения потоковой подпроблемы (3.1)–(3.4). И, наоборот, для любого потока с $x_{ij} > 0$ должно выполняться равенство $y_{ij} = 1$, чтобы удовлетворить ограничения (3.4)–(3.5). Поскольку оптимизация будет искать минимальное значение $\sum_{(ij)\in E} c_{ij} y_{ij}$, она не будет выбирать какое-либо излишнее значение $y_{ij} = 1$, поэтому оптимум минимален, а результат должен быть древовидным графом. Следовательно, это и есть оптимальное решение STP, как и задумывалось.

Экземпляр QoSMTP определяется неориентированным взвешенным графом $G = (V, E, l, r)$, множеством терминальных вершин $T \subseteq V$ и корневым узлом R, который для этой проблемы не является произвольным. Функция $l: E \to R^+$ определяет длину ребер, а функция $r: V \to R_0^+$ выдает значение весов узлов. Узлы Штейнера имеют нулевой вес, т.е. $r(s) = 0$ для всех $s \in S$. Поскольку $r(R)$ в QoSMTP является бессмысленным, в модели предполагается, что $r(R) = 0$. Наконец, $r(t) > 0$ для каждого терминального узла за исключением корня ($t \in T \backslash \{R\}$). Если $r(t) = 1$ для каждого t в $T \backslash \{R\}$, то экземпляр QoSMTP эквивалентен экземпляру STP с тем же корнем, так как максимальный вес r_t в любом поддереве, содержащем некоторые терминальные узлы, кроме корня, равен 1. В отличие от STP, узлы Штейнера в экземпляре QoSMTP могут входить в оптимальное решение, так как они имеют нулевой вес, и поэтому длины исходящих из них ребер также равны нулю.

Аналогичная процедура применяется к формулировке QoSMTP. Создается ориентированная версия G' для экземпляра G QoSMTP. Потоковые переменные x_{ij} и соответствующие им y_{ij} такие же, как в STP, то же относится и к соотношениям (3.1)–(3.5). Такая формулировка исключает возможность потоковых циклов, т.е. позволяет иметь либо $y_{ij} = 1$, либо $y_{ji} = 1$, но не одновременно. Этот факт важен для согласования блоков соотношений, используемых для определения весов поддеревьев.

Чтобы обеспечить веса $\overline{r}_e$ (максимум среди весов узлов поддеревьев), используются дополнительные переменные z_u, $u \in V \backslash \{R\}$. Для заданного экземпляра QoSMTP введем константу $C = max\{r_v, v \in T \backslash \{R\}\}$ и рассмотрим соотношения (3.7)-(3.8). Основная идея введения дополнительных переменных заключается в том, что если значения переменных y_{uv} определяют поддерево в G, то соотношения (3.6)–(3.7) заставляют переменные z_v быть большими или равными $\overline{r}_e$, где e является звеном в единственном пути, который соединяет v с корнем внутри этого дерева.

$$\begin{cases} z_v \geq r_v & \forall v \in V \backslash \{R\}, \quad (3.6) \\ z_v \geq z_u + C(y_{uv} - 1) & \forall v \in V \backslash \{R\}, (uv) \in E'. \quad (3.7) \end{cases}$$

Таким образом, если переменные y_{uv} определяют дерево, ориентированное к корню, и процесс оптимизации ищет минимальное значение функции, определенное в уравнении (3.9), то процесс оптимизации будет опускать значения как можно ниже значения z_v, достигая при этом значений $\overline{r}_e$ для этого дерева, определенного переменными y.

$$\sum_{(uv) \in E'} l(uv) \times z_u \times y_{uv}. \quad (3.8)$$

Соотношение (3.6) определяют нижние границы для переменной z_v. В тех случаях, когда $y_{uv} = 0$, соотношение (3.7) принимает вид $z_v \geq z_u - C$ и деактивируется, так как C является максимумом среди разрядов, а $z_v \geq 0$ является не менее ограничивающим. И наоборот, для тех ребер, где $y_{uv} = 1$, соотношение (3.7) преобразуются в $z_v \geq z_u$, указывая на то, что значение z_v должно быть не больше максимума, определяемого активными соотношениями. Соотношение (3.7) действует в поддеревьях, содержащих v, в число которых не входит корень. Последние замечания верны при отсутствии ребер в E', выходящих из R, что изолирует значения между поддеревьями, и в направлении путей дерева от терминальных узлов к корню, что добавляет ограничения нижних границ в пути от каждого узла к R. Результатом оптимизации должно быть дерево, так как при любом возможном решении добавление ребра увеличило бы число активных нижних границ для z_v, тем самым увеличивая целевое значение, за исключением узлов-заглушек Штейнера.

Целевая функция в (3.8) квадратичная, что значительно затрудняет решение задачи. Для устранения этого недостатка в предлагаемой модели вводится вспомогательное множество

переменных $\eta_{ij}, (ij) \in E'$ и переформулируется целевая функция, определяемая теперь соотношениями (3.9)–(3.11):

$$\begin{cases} min \sum_{(ij) \in E'} l(ij) \times \eta_{ij} & (3.9) \\ \eta_{ij} \geq z_i + C(y_{ij} - 1) \quad \forall (ij) \in E' & (3.10) \\ \eta_{ij} \geq 0 \quad \forall (ij) \in E' & (3.11) \end{cases}$$

После объединения уравнений (3.1)–(3.7), (3.9)–(3.11) переменные η_{ij} захватывают значения $z_i \times y_{ij}$ в оптимальных решениях, так что новая целевая функция соответствует (3.9) для целей оптимизации. Соотношение (3.10) деактивируется всякий раз, когда $y_{ij} = 0$, оставляя соотношение (3.11) единственной нижней границей для η_{ij}. Оптимизация приводит к $\eta_{ij} = 0$, таким образом, $\eta_{ij} = z_i \times y_{ij}$. Если же $y_{ij} = 1$, то соотношение (3.10) преобразуется в $\eta_{ij} \geq z_i$, а предыдущее ограничение должно быть активным при оптимуме.

4. Экспериментальный анализ

В данном разделе представлена экспериментальная оценка предложенной модели для QoSMTP на конкретном наборе новых экземпляров. Топологическая структура новых экземпляров была выбрана из стандартных экземпляров классов B и I080 библиотеки SteinLib, общеизвестном репозитории экземпляров STP [7]. Предложенная модель для QoSMTP была реализована в оптимизационном решателе IBM ILOG CPLEX Interactive Optimizer 12.6.3. Экспериментальная оценка проводилась на сервере HP ProLiant DL385 G7 с 24 процессорами AMD Opteron 6172 и 64 ГБ оперативной памяти.

Табл. 1. Экспериментальные результаты предложенной модели для QoSMTP на модифицированных экземплярах b01-b18
Table 1. Experimental results of the proposed model for the QoSMTP over modified instances b01 to b18

Экземпляр	|V|	|T|	|E|	Число переменных	Число ограничений
mdfb01	50	9	63	424	472
mdfb02	50	13	63	418	464
mdfb03	50	25	63	421	468
mdfb04	50	9	100	640	686
mdfb05	50	13	100	634	678
mdfb06	50	25	100	634	678
mdfb07	75	13	94	635	708
mdfb08	75	19	94	632	704
mdfb09	75	38	94	623	692
mdfb10	75	13	150	959	1028
mdfb11	75	19	150	965	1036
mdfb12	75	38	150	962	1032
mdfb13	100	17	125	840	936
mdfb14	100	25	125	843	940
mdfb15	100	50	125	837	932
mdfb16	100	17	200	1287	1382
mdfb17	100	25	200	1287	1382

mdfb18	100	50	200	1296	1394

Пример	**Оптимум**	**Разрыв**	***TC*(c)**	***TV*(c)**
mdfb01	6080*	0.01%	0,59	0,59
mdfb02	5770*	0.01%	0,5	0,5
mdfb03	7925*	0.01%	0,71	0,71
mdfb04	3883*	0.01%	13,39	145,86
mdfb05	3146*	0.01%	10,06	41,84
mdfb06	7820	5.96%	9476,51	timeout
mdfb07	7616*	0.01%	10,11	10,11
mdfb08	7364*	0.01%	11,12	29,35
mdfb09	15877	0.01%	18,08	18,08
mdfb10	5096*	0.01%	15,67	3667,31
mdfb11	6718	11.78%	4600,61	timeout
mdfb12	10716	0.01%	1988,57	3276,68
mdfb13	12076	0.01%	32,19	797,12
mdfb14	15159	3.61%	18,54	timeout
mdfb15	20599	0.01%	33,56	118,26
mdfb16	5288	11.12%	56,34	timeout
mdfb17	6807	15.67%	19499,5	timeout
mdfb18	9884*	0.01%	19513,19	21588,79

В табл. 1 приведены результаты предложенной модели для QoSMTP, с указанием названия каждого примера, количества узлов, терминальных узлов и ребер в графе (|V|, |T|, и |E|). Оптимальные решения для экземпляров QoSMTP неизвестны, метод ветвей и границ и целочисленных релаксаций, применяемый в CPLEX, позволяет вычислить пороговые значения ошибок для вычисляемых решений (значение разрыва указано).

С учетом процедуры определения веса узлов (целочисленные значения, выбранные между 1 и 10 в соответствии с равномерным распределением), если абсолютная погрешность меньше одной единицы, то вычисленная оптимальная стоимость соответствует оптимальному значению (которое обозначается звездочкой *). Также сообщается количество переменных и ограничений, связанных с решением, время, требующееся решателю для расчета решения (TC, в секундах), время, прошедшее до того момента, как решатель подтвердил/проверил, что решение найдено (TV, в секундах). Все результаты соответствуют выполнению предложенной модели, реализованной в CPLEX, с указанным выше временным ограничением в шесть часов.

Результаты, приведенные в табл. 1, показывают, что предлагаемая модель способна вычислить точные решения для всех экземпляров проблемы класса B, включая нахождение оптимального решения для девяти примеров. В 15 из 18 случаев разрыв составил менее 6%, а худший показатель был на уровне 15,67%. Тринадцать решений были рассчитаны эффективно (т.е. время исполнения составило менее минуты), но предложенной модели не удалось проверить пять решений в установленный срок – за шесть часов. Остальные результаты представлены в [8] для экземпляров i080* в библиотеке SteinLib.

5. Заключение и перспективы исследований

В данной статье представлена основанная на потоках формулировка QoSMTP в терминах смешанного целочисленного программирования, расширяющая применение существующих потоковых моделей для STP, чтобы охватить дополнительную сложность QoSMTP.

Поскольку для QoSMTP не существует стандартных экземпляров проблемы, был создан новый набор путем модификации экземпляров STP класса B из библиотеки SteinLib, следуя рандомизированной процедуре для включения особенностей QoSMTP. В результате, 13 примеров задачи были решены оптимально. В 15 из 18 случаев максимальный разрыв составил менее 6%, а худший показатель был на уровне 15,67%. Большинство решений были рассчитаны менее, чем за одну минуту. Предлагаемая модель является эффективным и гибким вариантом решения QoSMTP, учитывая, что прежде для нее не предлагалось никаких формулировок смешанного целочисленного программирования или исчерпывающих экспериментальных оценок.

Основное направление будущих исследований связано с расширением экспериментальной оценки предлагаемой модели на большее количество экземпляров, тем самым совершенствуя вычислительные возможности предлагаемого подхода, например, путем применения методов предварительной обработки для повышения производительности решателей.

Список литературы / References

[1] Michel Goemans and Young-Soo Myung. A catalog of Steiner tree formulations. Networks, vol. 23, no.1, 1993, pp. 19-28.

[2] Mathias Hauptmann and Marek Karpinski. A Compendium on Steiner Tree Problems. Technical report. Department of Computer Science and Hausdorff Center for Mathematics, University of Bonn, 2014.

[3] Marek Karpinski, Ion Mandoiu, Alexander Olshevsky, and Alexander Zelikovsky. Improved Approximation Algorithms for the Quality of Service Multicast Tree Problem. Algorithmica, vol. 42, no. 2, 2005, pp. 109-120.

[4] Richard Karp. Reducibility among Combinatorial Problems. In Complexity of Computer Computations. The IBM Research Symposia Series. Springer, 1972, pp. 85-103.

[5] Lester Ford and Delbert Fulkerson. Flows in Networks. Princeton University Press, 2010. 212 p.

[6] Narendra Karmarkar. A new polynomial-time algorithm for linear programming. Combinatorica, vol. 4, no. 4, 1984, pp. 373-395.

[7] Thorsten Koch, Alexander Martin, and Stefan Voß. SteinLib: An Updated Library on Steiner Tree Problems in Graphs. Combinatorial Optimization, vol. 11, 2001, pp. 285-325.

[8] Claudio Risso, Sergio Nesmachnow, and Franco Robledo. Mixed Integer Programming Formulations for Steiner Tree and Quality of Service Multicast Tree problems. Programming and Computer Software, vol. 46, no. 8, 2020, pp. 661–678.

Информация об авторах / Information about authors

Клаудио Энрике РИССО-МОНТАЛЬДО, кандидат наук, научный сотрудник. Его исследовательские интересы включают оптимизацию сетей, многоуровневые сети, проектирование устойчивых сетей, комбинаторную оптимизацию, метаэвристику, теорию графов, оптические транспортные сети.

Claudio Enrique RISSO-MONTALDO, Ph.D., Researcher. His research interests include Network Optimization, Multilayer Networks, Design of Resilient Networks, Combinatorial Optimization, Metaheuristics, Graph Theory, Optical Transport Networks.

Франко Рафаэль РОБЛЕДО-АМОЗА, кандидат наук, профессор. Его исследовательские интересы включают исследования операций, модели надежности сетей, комбинаторную оптимизацию, моделирование методом Монте-Карло, дискретные события.

Franco Rafael ROBLEDO-AMOZA, Ph.D., Full Professor. His research interest include Operational Research, Network Reliability Models, Combinatorial Optimization, Monte Carlo Simulation, Discrete Events.

Серджио Энрике НЕСМАЧНОВ-КАНОВАС, кандидат наук, профессор, научный сотрудник. Область научных интересов: оптимизация, метаэвристика, высокопроизводительные вычисления, умные города.

Sergio Enrique NESMACHNOW-CÁNOVAS, Ph.D. in Computer Sciences, Full Professor and Researcher. Research interests: optimization, metaheuristics, high performance computing, smart cities.