УДК 004.942
https://doi.org/10.31854/2307-1303-2025-13-2-43-51
EDN: FMMVHK

Синтез аналитических моделей и методов машинного обучения для балансировки нагрузки при строгих ограничениях на время отклика

Редругина Н. М.✉, **Тарабанов И. Ф.**

Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича, Санкт-Петербург, 193232, Российская Федерация

Предмет и цель работы. *Статья посвящена решению проблемы балансировки нагрузки в условиях нетерпеливых пользователей и разнородного трафика в телекоммуникационных системах. Целью работы является разработка концептуального фреймворка, сочетающего аналитические методы теории массового обслуживания и современные подходы машинного обучения для минимизации совокупного штрафа, учитывающего задержки обслуживания и стоимость отказов.* ***Используемые методы.*** *В основе исследования лежит аналитическая модель M/G/1/K с нетерпеливыми пользователями, позволяющая оценить ключевые показатели системы. Для случаев, когда аналитическое решение невозможно или неэффективно, предложено применение: (1) прогнозирования временных рядов для предсказания нагрузки, (2) бинарной классификации для оценки вероятности оттока, обучения с подкреплением для оптимизации целевой функции.* ***Новизна*** *работы заключается в системном подходе к сочетанию аналитических методов и методов машинного обучения для задач балансировки, а также в том, что учитывается разнородная стоимость отказов для различных классов трафика. Предложена новая формализация задачи через призму обучения с подкреплением.* ***Основные результаты.*** *Разработана концепция интеллектуальной системы балансировки, демонстрирующая потенциальные преимущества перед традиционными методами.* ***Практическая значимость.*** *Результаты исследования могут быть использованы при проектировании систем управления нагрузкой в автономных сетях.*

Ключевые слова: *машинное обучение, теория массового обслуживания, нагрузка, математическое моделирование, миграция, балансировка*

Введение

Стремительная цифровизация экономики, появление сервисов, критичных к задержкам, и экспоненциальный рост трафика требуют от телекоммуникационных сетей нового качества – способности к автономной работе. Ответом на этот вызов становится концепция сетей Post-NGN – архитектурная парадигма, ориентированная на создание самоуправляемых, самооптимизирующихся и интел-

Библиографическая ссылка на статью:
Редругина Н. М., Тарабанов И. Ф. Синтез аналитических моделей и методов машинного обучения для балансировки нагрузки при строгих ограничениях на время отклика // Информационные технологии и телекоммуникации. 2025. Т. 13. № 2. С. 43–51. DOI: 10.31854/2307-1303-2025-13-2-43-51. EDN: FMMVHK

Reference for citation:
Redrugina N. M., Tarabanov I. F. Synthesis of Analytical Models and Machine Learning Methods for Load Balancing under Strict Response Time Constraints // Telecom IT. 2025. Vol. 13. Iss. 2. PP. 43–51. (in Russian). DOI: 10.31854/2307-1303-2025-13-2-43-51. EDN: FMMVHK

лектуальных инфраструктур[1] [1]. Эволюция сетей управления движется от ручного администрирования к проактивной автономии. Современные сети проектируются с учетом самостоятельной интерпретации задач высокого уровня (Intent-Based Networking), динамического перераспределения ресурсов, оптимизации энергопотребления и противодействия угрозам в реальном времени без постоянного вмешательства человека[2, 3] [1]. Для реализации этой концепции отраслевые организации (например, TM Forum) разработали: эталонную бизнес-архитектуру[4]; стандарт и методологию оценки уровня автономности сети[5] (рисунок 1); техническую архитектуру[6].

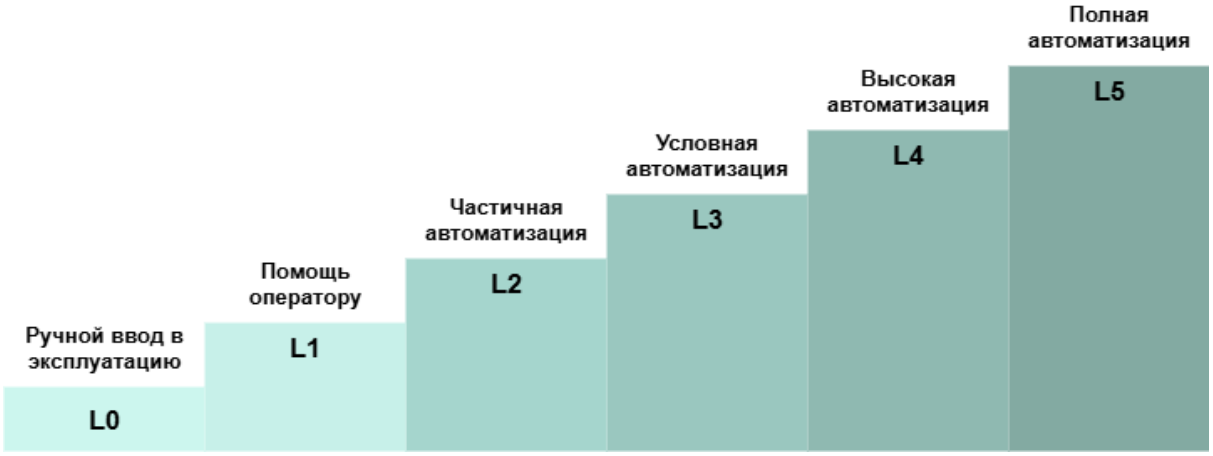

Рис. 1. Уровни зрелости автономных сетей (от реактивного до полностью автономного)

Рабочая группа 3GPP SA WG5 инициировала серию проектов по разработке автономных сетей, охватывающих весь жизненный цикл управления сетью. В настоящее время достигнуты соглашения по многоуровневой архитектуре [2–4], ключевым концепциям [5], методам оценки автономности [6] и реализации замкнутых контуров управления для целенаправленного управления [7, 8]. При этом исследование[7] показывает, что, хотя многие поставщики коммуникационных услуг (CSP, *аббр. от англ.* Communications Service Providers) уровня L3–L4 активно преобразуют концепции в возможности, деятельность CSP

[1] Autonomous Networks: Empowering digital transformation – evolving from Level 2/3 towards Level 4 (IG1326) // TM Forum. 2023. URL: https://www.tmforum.org/resources/standard/ig1258-autonomous-networks-empowering-digital-transformation-r19-0-0 (дата обращения 05.09.2025)

[2] The path towards autonomous network operation: can you let go? // Nokia. 2024. URL: https://www.nokia.com/blog/the-path-towards-autonomous-network-operation-can-you-let-go (дата обращения 05.09.2025)

[3] Harrod J. The Path to an Autonomous Network // ISG. URL: https://isg-one.com/articles/the-path-to-an-autonomous-network (дата обращения 05.09.2025)

[4] IG1218 Autonomous Network Business Requirements and Framework // TM Forum. 2022. URL: https://www.tmforum.org/resources/how-to-guide/ig1218-autonomous-networks-business-requirements-and-framework-v2-1-0 (дата обращения 11.09.2025)

[5] IG1252 Autonomous Network Levels Evaluation Methodology // TM Forum. 2021. URL: https://www.tmforum.org/resources/introductory-guide/ig1252-autonomous-network-levels-evaluation-methodology-v1-2-0 (дата обращения 09.09.2025)

[6] IG1251 Autonomous Network Reference Architecture // TM Forum. 2021. URL: https://www.tmforum.org/resources/how-to-guide/ig1251-autonomous-networks-reference-architecture-v1-0-0 (дата обращения 05.09.2025)

[7] Navigating autonomous networks // IBM Institute for Business Value. 2025. URL: https://www.ibm.com/thought-leadership/institute-business-value/en-us/report/autonomous-networks (дата обращения 05.09.2025)

уровня L0–L2 в этой области ограничивается пилотными проектами. Это подчеркивает необходимость дальнейшего развития средств автоматизации и интеллектуализации для решения широкого круга задач.

Постановка задачи

Однако за сложными архитектурами и красивыми концепциями не всегда очевидна практическая необходимость и реальная ценность интеллектуализации. Возникает закономерный вопрос: насколько оправдан столь сложный переход и можно ли в ряде случаев ограничиться традиционными, проверенными методами?

Ответ прост: классические подходы перестают справляться, а принятие решений человеком становится «узким местом». Проиллюстрируем это на конкретном и жизненно важном примере из ядра любой сети – задачи балансировки нагрузки при ограниченном времени отклика. На данный момент «терпение» – это крайне ограниченный ресурс как для конечных абонентов, ожидающих начала сеанса связи или ответа от системы, так и для устройств Интернета вещей, выполняющих транзакционные операции (далее – пользователей). Промоделируем ситуацию и для математической простоты предположим, что время ожидания у каждого потребителя услуги будет ограничено. Пользователи, покинувшие систему до начала обслуживания, будут классифицированы как потерянные.

Ниже приведена приближенная формула (1) для оценки доли потерянных пользователей (π) в долгосрочной перспективе:

$$\pi(\tau) = (1 - \zeta^2)\pi^{det}(\tau) + \zeta^2\pi^{exp}(\tau), \quad (1)$$

где коэффициент вариации $\zeta^2 = (H[S^2] - H^2[S])/H^2[S]$; $\pi^{det}(\tau)$ – вероятность потери, если время обслуживания детерминировано; $\pi^{exp}(\tau)$ – вероятность потери, если время обслуживания экспоненциально.

Задача автоматизации системы заключается в поиске компромисса между качеством обслуживания запроса и его постановки в очередь. Мы стремимся либо максимизировать полезность системы (v_{use}), либо минимизировать общий штраф (штраф за потерянные запросы и за задержку – z_{lose}, z_{wait}, соответственно).

Таким образом, целевая функция будет складываться из следующих параметров:

$$V = v_{use} - z_{lose} - z_{wait} \quad (2)$$

или

$$V = \beta\big(1 - \pi(\tau)\big) - \sum_{i=1} C_i\pi_i(\tau) - \alpha H\big[W_{qp}\big], \quad (3)$$

DOI: 10.31854/2307-1303-2025-13-2-43-51
URL: https://www.sut.ru/doci/nauka/1AEA/ITT/2025_2/43-51.pdf

где β – средняя выгода от обслуженного запроса; $\sum_{i=1} C_i \pi_i(\tau)$ – суммарный взвешенный штраф, учитывающий вероятность потерь для класса i с ценой отказа C_i; α – штрафной коэффициент за высокую задержку; $H[W_{qp}]$ – средняя задержка в очереди (4) для обслуженных пользователей.

Нетрудно заметить, что каждый потерянный пользователь проводит в очереди некоторое время τ, что приводит к оценке общего времени, проведенного в очереди всеми пользователями:

$$H\big[W_q\big] = \tau\pi(\tau) + H\big[W_{qp}\big]\big(1 - \pi(\tau)\big). \quad (4)$$

Ключевая задача связана с определением времени ожидания запросов, принятых к обслуживанию. Для этого применяются аппроксимационные формулы, указанные в [9]. Однако по мере роста неопределенности распределений времени обслуживания и увеличения интенсивности поступления запросов усложняются и модели, используемые для вычисления выходных параметров системы.

Методика внедрения машинного обучения в автоматизацию сетевых процессов

Аналитическая модель, представленная в предыдущем разделе, дает фундаментальное понимание системы и позволяет вычислить оптимальные параметры статически, в предположении, что все они (интенсивность входа λ, распределение времени обслуживания и приоритетность источников) известны и неизменны. Однако в реальной динамической среде телекоммуникационных сетей эти параметры являются стохастическими и нестационарными: нагрузка носит пульсирующий характер, а параметры трафика постоянно меняются.

Здесь на первый план выходят методы машинного обучения (ML, *аббр. от англ.* Machine Learning), которые позволяют перейти от статической регуляризации к динамическому и адаптивному управлению системой в реальном времени. Данную идею поддержали авторы [10], указав что последние модели глубокого обучения (DL, *аббр. от англ.* Deep Learning) обладают рядом преимуществ по сравнению с традиционными инструментами сетевого моделирования (симуляторами, теорией массового обслуживания и др.). Например, модели, основанные на DL, продемонстрировали высочайшую производительность при моделировании фиксированных сетей [11], превзойдя известные аналитические модели, основанные на теории массового обслуживания. Рассмотрим три возможных сценария применения ML-методов.

Сценарий 1: прогнозирование необходимости балансировки

Аналитическая модель определяет критический порог нагрузки λ_{crit}, при достижении которого общий штраф целевой функции V становится недопустимым и требуется миграция части трафика. Однако в реальном времени будущие значения λ не известны. Так обозначается задача прогнозирования временных рядов, определяемая необходимостью предсказать вероятность нарушения соглашения об уровне сервиса (SLA, *аббр. от англ.* Service Level Agreement) (т. е., что

$\pi(\tau)C_i$ > критического порога) на некотором горизонте прогнозирования T_{pred}. Данные для обучения (таблица 1) необходимо получить из исторических данных с контроллера или серверного оборудования, в зависимости от цели исследования.

Результатом работы станет возможность получения информации о необходимости превентивного запуска алгоритма балансировки до наступления перегрузки и потери критически важных запросов.

Таблица 1 – Признаки для обучения к Сценарию 1

Признак	Описание
$\lambda_{in}(t)$	интенсивность входящего трафика (пакетов / запросов)* во временны́х окнах
$H[W_q]$	текущая средняя задержка
$\pi(\tau)$	текущая доля отказов
utilization(t)	загрузка CPU / канала*

* – в зависимости от цели и объекта исследования.

Сценарий 2: прогнозирование оттока и определение кандидатов на миграцию

Когда балансировщик принимает решение перераспределить нагрузку, встает вопрос: какие именно запросы или сессии необходимо мигрировать. Очевидно, что это задача бинарной классификации. Тут могут подойти как классические модели («логистическая регрессия», «решающие деревья»), так и ансамбли (XGBoost или LightGBM), которые лучше всего обрабатывают табличные данные с категориальными признаками (таблица 2) [12]. Для каждого запроса x_i, находящегося в очереди, требуется рассчитать вероятность того, что время его ожидания $P_{lose}(x_i)$ превысит его «терпение» τ_i.

Таблица 2 – Признаки для обучения к Сценарию 2

Признак	Описание
class	класс обслуживания [13]
Заявленные требования SLA	
τ_i	время, возможное для ожидания
C_i	«цена» отказа*
Динамические параметры состояния системы	
W_q	текущая средняя задержка
L_q	текущая длина очереди
$\pi(\tau)$	текущая доля отказов
A_i	время, которое запрос x_i уже находится в очереди

* – в зависимости от цели и объекта исследования.

В результате исполнения алгоритма с обученной на исторических данных моделью ML балансировщик получает ранжированный список запросов в очереди по убыванию $P_{lose}(x_i)C_i$ (т. е. по величине ожидаемого штрафа). Первыми для миграции выбираются запросы с наибольшим ожидаемым штрафом (проактивное управление рисками).

Сценарий 3: динамическая оптимизация политик через обучение с подкреплением

Сценарии 1 и 2 рассматривают систему фрагментарно. Задача выбора, куда именно мигрировать трафик для минимизации глобального штрафа во всей гетерогенной системе, чрезвычайно сложна. Аналитическое решение невозможно из-за огромной размерности пространства состояний. К тому же аналитические модели и даже предиктивные ML-методы часто работают в рамках заранее заданных, жестких политик (если задержка больше x, то необходима миграция y запросов).

Обучение с подкреплением позволяет отказаться от этого подхода и научить Агента самостоятельно находить наиболее эффективную стратегию в процессе взаимодействия со средой. Агент будет учиться на собственном опыте, понимая, какие действия в различных состояниях приводят к максимальной совокупной «награде» (в нашем случае – к минимизации итогового штрафа).

В итоге должен быть реализован Агент – интеллектуальный алгоритм, встроенный в контроллер балансировки нагрузки (например, в SDN-контроллер или оркестратор). Среда, в которой должен работать Агент – это распределенная управляемая система: пул серверов (или сетевых срезов), их очереди, а также генераторы входящего трафика. Состояния системы $S(t)$ – это векторное представление системы в момент времени t. Иными словами, это совокупность признаков (таблица 3), которые получает Агент для принятия решений.

Таблица 3 – Признаки для обучения к Сценарию 3

Признак	Описание
Общая информация	
λ_{in}	общая интенсивность входящего трафика
По каждому серверу $j \in N$ (где N – пул серверов)	
CPU_j^{util}	нормализованная загрузка CPU $(1 - TP_{average} / TP_{total}) \times 100$ %
RAM_j^{util}	доступная оперативная память
v_j^{net}	текущая используемая пропускная способность
L_j^q	текущее количество запросов в очереди
W_j^q	среднее время ожидания в очереди на этом сервере ($H[W_q]$ для этого узла)
π_j	текущая доля отказов $\pi(\tau)$ на этом сервере
По входящему запросу i (если он есть в момент времени t)	
L_i^{req}	категория запроса [13]
$t_i^{\text{обсл.}}$	прогнозируемое время обслуживания
τ_i	время, возможное для ожидания
C_i	«цена» отказа*

$TP_{average}$ – среднее количество процессорного времени в режиме ожидания в секунду на все ядра;
TP_{total} – общее количество процессорного времени в секунду на все ядра;
* – в зависимости от цели и объекта исследования.

В данной задаче действие (решение, которое принимает Агент) может быть дискретным (выбор из конечного набора вариантов) или непрерывным (задание приоритета). Для нового запроса происходит выбор сервера j (от 1 до N), на который будет направлен запрос. Для миграции запроса происходит выбор пары

DOI: 10.31854/2307-1303-2025-13-2-43-51
URL: https://www.sut.ru/doci/nauka/1AEA/ITT/2025_2/43-51.pdf

(запрос i – целевой сервер j). Это огромное пространство действий, поэтому его можно упростить, например, периодически мигрируя запрос с наибольшим $P_{lose}(x_i)C_i$ на сервер с наименьшей загрузкой.

Нельзя забывать об обратной связи от среды о «качестве действия». Наша глобальная цель – максимизировать целевую функцию V за счет снижения штрафов за потери и задержку.

Награду на каждом шаге можно определить как отрицательное значении штрафа (максимизация награды должна равняться минимизации штрафа):

$$r(t) = -(\Delta z_{lose} + \alpha \times \Delta z_{wait}), \quad (5)$$

где Δz_{lose} – изменение суммарного взвешенного штрафа за отказы с момента предыдущего действия (например, если за последний шаг из-за отказа потеряно 2 запроса с C_i = 10, то Δz_{lose} = 20); Δz_{wait} – изменение суммарной задержки всех обслуженных запросов; α – коэффициент важности задержки относительно отказов.

Заключение

В рамках данной работы была обоснована необходимость перехода от классических аналитических методов балансировки нагрузки к интеллектуальным системам, способным работать в условиях динамически меняющейся нагрузки и нетерпеливых пользователей. Сформулировано концептуальное решение, синтезирующее аппарат теории массового обслуживания и современные ML-методы.

Перспективы дальнейших исследований связаны с переходом от концептуального обоснования к практической реализации и тестированию предложенного решения. Последующие работы будут посвящены разработке и формализации принципов обучения интегрированной системы, а именно:

– разработке итеративного рабочего процесса для совместного обучения и взаимодействия моделей разных уровней (прогнозирования, классификации и RL-агента);

– формализации и решению проблемы совмещения временны́х горизонтов принятия решений: от долгосрочного стратегического планирования на основе прогнозов до тактических действий RL-агента в реальном времени;

– исследованию и созданию эффективных механизмов для сбора и подготовки данных, а также применению методов обучения с частичным привлечением учителя и самоконтроля для минимизации зависимости от размеченных наборов данных;

– всестороннему тестированию предложенных решений на симуляторах сетевой среды и в условиях, приближенных к реальным, для валидации теоретических выводов и точной оценки производительности.

Реализация указанных шагов позволит создать полноценный прототип автономной системы управления нагрузкой, соответствующей уровню зрелости L3–L4, и сделать значительный шаг в направлении внедрения принципов автономных сетей (Post-NGN) в практику эксплуатации телеком-инфраструктур.

Литература

1. ETSI GS ZSM 002 V1.1.1 (2019-08). Zero-touch network and Service Management (ZSM). Reference Architecture.

2. ETSI TS 128 533 V15.0.0 (2018-10). 5G. Management and orchestration. Architecture framework (3GPP TS 28.533 version 15.0.0 Release 15).

3. ETSI TS 128 535 V16.1.0 (2020-11). 5G. LTE. Management and orchestration. Management services for communication service assurance. Requirements (3GPP TS 28.535 version 16.1.0 Release 16).

4. ETSI TS 128 536 V16.0.0 (2020-07). LTE. 5G. Management and orchestration. Management services for communication service assurance. Stage 2 and stage 3 (3GPP TS 28.536 version 16.0.0 Release 16).

5. ETSI TR 28 810 V17.0.0 (2020-09). Study on concept, requirements and solutions for levels of autonomous network. (3GPP TR 28.810 version 17.0.0 Release 17).

6. ETSI TS 128 100 V17.0.0 (2022-05). 5G. Management and orchestration. Levels of autonomous network (3GPP TS 28.100 version 17.0.0 Release 17).

7. ETSI TR 28 812 V17.1.0 (2020-12). Telecommunication management; Study on scenarios for Intent driven management services for mobile networks. (3GPP TR 28.812 version 17.1.0 Release 17).

8. ETSI TS 28.312 V17.1.1 (2022-09). Management and orchestration; Intent driven management services for mobile networks (3GPP TR 28.312 version 17.1.1 Release 17).

9. De Kok A. G., Tijms H. C. A queueing system with impatient customers // Journal of Applied Probability. 1985. Vol. 22. Iss. 3. PP. 688–696.

10. Almasan P., Ferriol-Galmes M., Paillisse J., Suarez-Varela J., Perino D., et al. Network Digital Twin: Context, Enabling Technologies and Opportunities // IEEE Communications Magazine. 2022. Vol. 60. Iss. 11. PP. 22–27. DOI: 10.1109/MCOM.001.2200012. EDN: JBSBYX

11. Eisen M., Ribeiro A. Optimal Wireless Resource Allocation with Random Edge Graph Neural Networks // IEEE Transactions on Signal Processing. 2020. Vol. 68. PP. 2977–2991. DOI: 10.1109/TSP.2020.2988255

12. Rashka S., Liu Y. H., Mirjalili V. Machine Learning using PyTorch and Scikit-Learn. Developing machine learning and deep learning models in Python. Packt Publishing Ltd, 2022.

13. ETSI TS 23 501 V19.4.0 (2025-06). System architecture for the 5G System (5GS). Stage 2. (3GPP TS 23.501 version 19.4.0 Release 19).

Статья поступила 05 сентября 2025 г.
Одобрена после рецензирования 23 сентября 2025 г.
Принята к публикации 26 сентября 2025 г.

Информация об авторах

Редругина Наталия Михайловна – кандидат технических наук, доцент кафедры инфокоммуникационных систем Санкт-Петербургского государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича.
E-mail: redrugina.nm@sut.ru

Тарабанов Илья Федорович – аспирант, старший преподаватель кафедры инфокоммуникационных систем Санкт-Петербургского государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича. E-mail: tarabanov.if@sut.ru

https://doi.org/10.31854/2307-1303-2025-13-2-43-51
EDN: FMMVHK

Synthesis of Analytical Models and Machine Learning Methods for Load Balancing under Strict Response Time Constraints

Redrugina N.✉, **Tarabanov I.**

The Bonch-Bruevich Saint Petersburg State University of Telecommunications,
St. Petersburg, 193232, Russian Federation

The subject and purpose of the work. *The article is devoted to solving the problem of load balancing in conditions of impatient users and heterogeneous traffic in telecommunication systems. The aim of the work is to develop a conceptual framework that combines analytical methods of queuing theory and modern machine learning approaches to minimize the cumulative penalty, taking into account maintenance delays and the cost of failures.* ***The methods used.*** *The research is based on the M/G/1/K analytical model with impatient users, which makes it possible to evaluate the key indicators of the system. For cases where an analytical solution is impossible or ineffective, the use of (1) time series forecasting to predict load, (2) binary classification to estimate the probability of outflow, and reinforcement learning to optimize the objective function is proposed.* ***The novelty.*** *The difference lies in a systematic approach to combining analytical and ML methods for balancing tasks, as well as taking into account the heterogeneous cost of failures for different traffic classes. A new formalization of the task is proposed through the prism of reinforcement learning.* ***The main results.*** *The concept of an intelligent balancing system has been developed, demonstrating potential advantages over traditional methods.* ***Practical significance.*** *The results can be used in the design of load management systems in autonomous networks.*

Key words: *machine learning, queuing theory, load, mathematical modeling, migration, balancing*

Information about Authors

Redrugina Natalia – Candidate of Technical Sciences. Associate Professor at the Department of Information and Communication Systems (The Bonch-Bruevich Saint Petersburg State University of Telecommunications). E-mail: redrugina.nm@sut.ru

Tarabanov Ilya – the Postgraduate Student, Senior Lecturer at the Department of Information and Communication Systems (The Bonch-Bruevich Saint Petersburg State University of Telecommunications). E-mail: tarabanov.if@sut.ru