АНАЛИЗ МОДЕЛЕЙ УПРАВЛЕНИЯ СЕТЕВЫМИ РЕСУРСАМИ В СЕТЯХ 5G

DOI: 10.36724/2072-8735-2023-17-5-32-41

Елагин Василий Сергеевич,
Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, г. Санкт-Петербург, Россия, ***elagin.vas@gmail.com***

Васин Антон Сергеевич,
Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, г. Санкт-Петербург, Россия, ***antoshca-vasin@yandex.ru***

Manuscript received 17 April 2023;
Accepted 11 May 2023

Ключевые слова: *NFV, SDN, сети пятого поколения, 5G, сетевая сегментация, виртуальное сетевое встраивание, сеть-субстрат, глубокое обучение, масштабирование ресурсов, MANO, VNFM, NFVO, LSTM*

В данной статье проводится анализ существующих моделей виртуального сетевого встраивания (VNE) и динамического масштабирования ресурсов виртуальных сетевых функций для сетевых сегментов (Network Slices). Данные модели обеспечивают предоставление услуг с требуемыми параметрами качества обслуживания (QoS) при эффективном использовании сетевых ресурсов. Динамические модели виртуального сетевого встраивания позволяют эффективно размещать виртуальные сети (VN) в сети-субстрате (SN) и реконфигурировать их по требованию. Для более гибкого динамического масштабирования дополнительно используются модели Holt-Winters, Bi-LSTM и т.д., использующие алгоритмы предсказания будущей утилизации сетевых ресурсов с целью уменьшения времени инициализации компонент виртуальных сетевых функций (VNF), задействованных для обслуживания определенных сетевых слоев. Произведено сравнение моделей динамического масштабирования и даны выводы о возможности их использования. В заключении сделан вывод о возможности использования данных моделей и необходимых доработках для более гибкого применения в сетях 5G.

Информация об авторах:
Елагин Василий Сергеевич, *к.т.н., доцент, кафедра Инфокоммуникационных систем Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича, г. Санкт-Петербург, Россия*

Васин Антон Сергеевич, *аспирант, кафедра Инфокоммуникационных систем Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича, г. Санкт-Петербург, Россия*

Для цитирования:
Елагин В.С., Васин А.С. Анализ моделей управления сетевыми ресурсами в сетях 5G // T-Comm: Телекоммуникации и транспорт. 2023. Том 17. №5. С. 32-41.

For citation:
Elagin V.S., Vasin A.S. (2023) Analysis of network resource scaling models in 5G network. *T-Comm*, vol. 17, no.5, pp. 32-41. (*in Russian*)

Введение

5-е поколение мобильных сетей является новым поколением сетей мобильной связи, которое имеет возможность для предоставления множества новых сервисов. Традиционные сети мобильной связи в основном используются только для предоставления услуг мобильной широкополосной связи и не могут быть адаптированы для различных вариантов использования 5G в будущем.

Строительство выделенных сетей для каждого из сценариев использования приведет к увеличению проблем, связанных с эксплуатацией сети, сложностью масштабирования и высоким затратам при развертывании и обслуживании. Поэтому одной из ключевых технологий, наряду с SDN и NFV, является технология сетевой сегментации – Network Slicing. Согласно определению, данным 3GPP, сетевой сегмент представляет логическую сеть, обеспечивающую определенные сетевые возможности и сетевые характеристики [1].

Таким образом возможно развернуть несколько независимых логических сетей, использующих ресурсы одной общей физической инфраструктуры. Каждый сетевой сегмент представляет собой логически независимую сквозную (E2E) сеть, которая состоит из набора сетевых функций (Network Functions) и соответствующих ресурсов для предоставления E2E услуг по запросу для конкретных сервисов [2]. Любая виртуальная сетевая функция (VNF) развертывается на виртуальных машинах (VM) или виртуальных контейнерах (например, Docker) с целью снижения затрат и энергопотребления.

Любой сервис представляет определенный набор сетевых функций, называемый цепочкой сервисных функций (Service Function Chain). Каждая сервисная функция в SFC может быть представлена только некоторыми сетевыми узлами. Чтобы добиться сетевой сегментации, необходимо иметь возможность выбора функциональных сетевых узлов в соответствии с SFC и определелить стратегию маршрутизации трафика функциональных узлов в необходимом порядке [3].

Хотя технология сетевой сегментации может обеспечить большую эффективность использования инфраструктуры оператора, она сталкивается со значительными проблемами при выборе требуемых сетевых функций, управлении ресурсами и масштабировании [4]. Операторы должны гарантировать, что требования к уровню обслуживания (SLA), задержке, полосе пропускания и ресурсам выполняются для каждого сетевого сегмента, несмотря на изменчивое поведение конечных пользователей в каждом сегменте. Поэтому должна быть возможность масштабирования сетевых сегментов в соответствии с потребностями пользователей, т.е. увеличения или уменьшения количества задействованных виртуальных ресурсов для каждой виртуальной сетевой функции при изменении колличества запросов услуг пользователями.

Исходя из всех этих требований определяются две основные проблемы, которые необходимо решить. Первая проблема заключается в правильном выборе физических узлов для размещения на них требуемого количества виртуальных сетевых функций с учетом всех требований сервиса и обозначается как виртуальное сетевое встраивание (Virtual Network Embedding (VNE)) [5]. Вторая проблема заключается в обеспечении динамического управления ресурсами виртуальных сетевых функций в зависимости от загруженности компонентов этих функций. Таким образом, в данной статье будут проанализированы существующие методы и модели, которые помогают решить данные проблемы, и даны возможные дополнения и рекомендации для этих моделей.

Формулирование задачи виртуального сетевого встраивания (VNE)

В концепции виртуализации основным компонентом является виртуальная сеть (VN). Виртуальная сеть представляет собой набор виртуальных сетевых узлов и виртуальных сетевых соединений поверх базовой физической сети, которая определяется как сеть-субстрат (Substrate Network) [5]. Виртуальные узлы связаны между собой виртуальными соединениями, образуя виртуальную топологию. Путем виртуализации вычислительных и сетевых ресурсов сети-субстрата можно создавать и размещать совместно несколько виртуальных сетей с различными характеристиками на одном и том же физическом оборудовании.

Проблема встраивания виртуальных сетей в сеть-субстрат является основной проблемой распределения ресурсов при сетевой виртуализации [6]. Благодаря динамическому встраиванию виртуальных ресурсов на физическое оборудование можно достигнуть многих преимуществ при организации будущих сетей с целью предоставления услуг конечным пользователям с гарантированым QoS.

На рисунке 1 показано, как при сетевой виртуализации используются встроенные алгоритмы сетевого встраивания для оптимального распределения виртуальных ресурсов на физической инфраструктуре. Оператор сетевых ресурсов использует алгоритмы встраивания для определения количества виртуальных ресурсов, которые необходимо запросить у поставщика сетевых ресурсов, который в свою очередь использует ресурсы сети-субстрата [5].

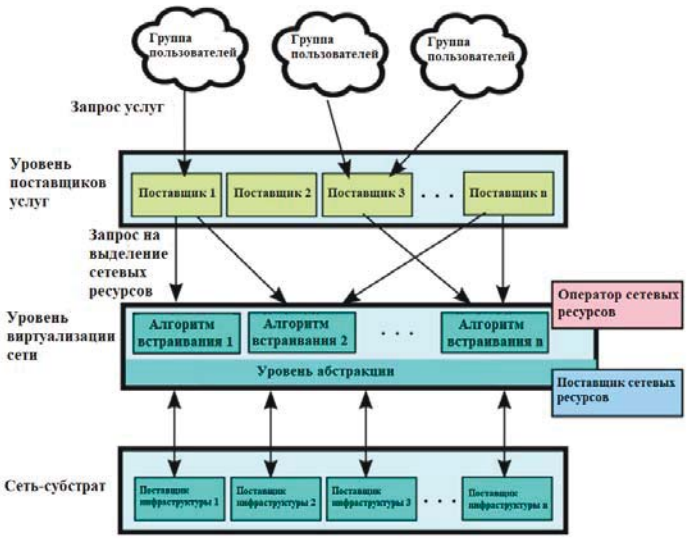

Рис. 1. Модель встраивания виртуальных сетевых функций в сеть-субстрат

Виртуальное сетевое встраивание связано с распределением виртуальных ресурсов как в вычислительных узлах, так и в сетевых соединениях.

Поэтому задачу виртуального сетевого встраивания можно декомпозировать на две подзадачи. Первая подзадача заключается в сопоставлении виртуальных узлов (Virtual Node Mapping (VNoM)), т.е. размещении виртуальных узлов на физических узлах сети-субстрата. Вторая подзадача заключается в сопоставлении виртуальных соединений (Virtual Link Mapping (VLiM)), т.е. сопоставлении виртуальных каналов, соединяющих виртуальные узлы, с физическими каналами, соединяющими узлы сети-субстрата.

Для решения этих задач в [5], [7] определены следующие подходы к разработке моделей и алгоритмов: топология виртуальной сети может быть произвольной, при распределении ресурсов для виртуальных сетей необходимо учитывать балансировку нагрузки как для физических каналов, так и для физических узлов. Запрос на выделение ресурсов определяется в виде графа, а не простого соединения между двумя точками в сети (конечные точки для соединений заранее неизвестны).

Эти задачи являются NP-трудными [8], поэтому получаемые в результате методы распределения и выделения ресурсов громоздки и неточны для сетей с большим количеством сетевых узлов. Таким образом, при решении данных задач используются различные алгоритмы машинного обучения (Machine Learning), федеративного обучения (Federated Learning), обучения с подкреплением (Reinforcement learning), глубокого обучения с подкреплением (Deep Reinforcement learning) и Q-обучения.

Решая проблему выделения ресурсов при виртуальном сетевом встраивании необходимо определить структуру математической модели, необходимые параметры модели, на основе которых происходит встраивание виртуальных сетей в сеть-субстрат, и найти точку соприкосновения VNoM и VLiM. Также необходимо класифицировать уже существующие модели виртуального сетевого встраивания и определить характеристики, на основе которых производится данная классификация.

Классификация моделей и определение параметров виртуального сетевого встраивания

Как было отмечено в предыдущем разделе, необходимо определить параметры, на основе которых осуществляется выбор физических узлов и соединений для размещения виртуальных узлов и соединений соответственно. В некоторых ранних трудах, описанных в [5], [7], [9], [10-13], модели виртуального сетевого встраивания состоят из различных параметров, в зависимости от используемого математического аппарата. Но основное сходство этих моделей заключается в том, что ресурсы сети-субстрата определяются набором узлов и соединений (N,L), а запрашиваемые ресурсы виртуальной сети определяются на основе набора виртуальных узлов и соединений (N^i, L^i), где $i = 1,...n$ обозначает количество запросов на выделение ресурсов для виртуальной сети.

Кроме того, каждый узел и соединение сети-субстрата имеет определенную емкость N_{cap} и L_{cap}, а любой виртуальный узел или соединение предъявляет требования к необходимой емкости для размещения N^i_{req} и L^i_{req}. В результате, для всех моделей, использующих различные алгоритмы, можно определить обобщенные функции для VNoM и VLiM: $fi : Ni \rightarrow N$, $gi : Li \rightarrow L$, где $\sum Nreq \leq Ncap$ и $\sum Lreq \leq Lcap$. В зависимости от варианта использования виртуальная сеть определяется в виде шаблона встраивания в сеть-субстрат (Virtual Network Template), на основе которого формируется запрос на выделение ресурсов (Virtual Network Request) [5]. Графически пример виртуального сетевого встраивания показан на рисунке 2. Две виртуальные сети, содержащие по три виртуальных узла встроены в одну сеть-субстрат с четырьмя узлами. Видно, что узлы сети-субстрата могут содержать несколько виртуальных узлов. Аналогично, сетевые соединения сети-субстрата могут предоставлять ресурсы для более чем одного виртуального соединения.

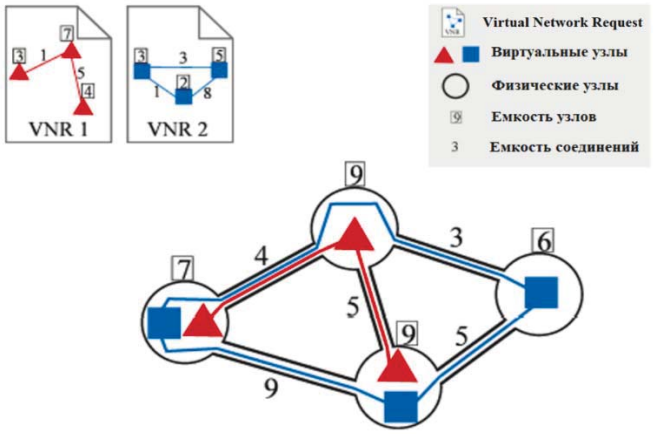

Рис. 2. Виртуальное сетевое встраивание

Все математические модели, на основе которых осуществляется виртуальное сетевое встраивание, сводятся к выбору оптимальных ресурсов сети-субстрата для виртуальных сетей. В завимости от поведения пользователей и сценария использования может возникнуть необходимость в перераспределении, изменении и высвобождении ресурсов виртуальной сети. Более того, некоторые сервисы требуют резервирования ресурсов для обеспечения высокой отказоустойчивости. Также не нужно забывать о необходимости минимизации использования ресурсов сети-субстрата и при этом гарантировать предоставление услуг с требуемым QoS. Поэтому из-за этих ограничений определяется несколько категорий алгоритмов виртуального сетевого встраивания.

В [5] предложено выделять 6 типов моделей виртуального сетевого встраивания: статическую, динамическую, централизированную, распределенную, с резервированием и без резервирования. Статическая или динамическая модель виртуального сетевого встраивания определяется на основе возможности реконфигурации уже размещенных виртуальных сетей при поступлении новых запросов на встраивание с целью повышения производительности сети-субстрата.

Статическое виртуальное сетевое встраивание не предусматривает возможности перераспределения ресурсов. Поэтому для данного варианта необходимо чтобы все запросы на встраивание были известны заранее, что не всегда соответствует потребностям пользователей. При динамическом варианте встраивания не требуется определять все запросы на встраивание заранее, что позволяет оптимизирвать использование ресурсов сети субстрата при добавлении или удалении виртуальной сети.

Примеры динамического виртуального сетевого встраивания описаны в [14-16]. Также хотелось бы выделить модель глубокого обучения с подкреплением [12], где агент глубинного распределения (Deep Allocation Agent (DAA)) использует комбинации глубинной нейронной сети и алгоритма распределения ресурсов для максимизации количества сетевых сегментов, которые должны быть успешно встроены в сеть-субстрат. Для этой цели в первую очередь встраивают сегмент с самыми максимальными требованиями к сети-субстрату (рис. 3).

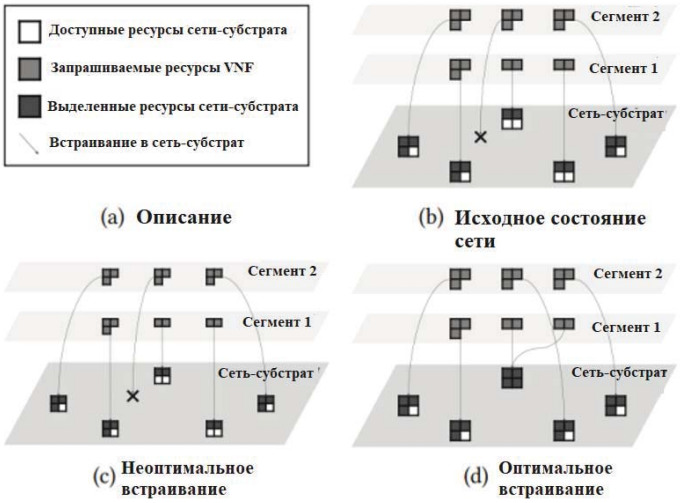

Рис. 3. Оптимальное встраивание сегментов в сеть-субстрат

Централизированное или распределенное сетевое встраивание определяется на основе количества узлов, отвечающих за встраивание. При централизированном подходе только один узел отвечает за встраивание. В этом случае вся информация о сети обрабатывается одним узлом, что положительно сказывается на встраивании во все элементы сети-субстрата. Но в этом случае централизированный узел является единственной точкой отказа, что приводит к полной остановке встраивания в случае выхода узла из строя. При распределенном подходе используется несколько узлов, отвечающих за встраивание. В этом случае обеспечивается высокая отказоустойчивость, но при этом необходимо обеспечить высокую синхронизацию между узлами для оптимального встраивания виртуальных сетей в сеть-субстрат.

Наличие резервирования при сетевом встраивании определяет потребность во встраивании дополнительных виртуальных узлов в сеть-субстрат с целью обеспечения высокой доступности сервисов при выходе из строя какого-либо узла сети-субстрата.

Данные типы сетевого встраивания являются взаимно независимыми. Поэтому любой выбранный алгоритм виртуального сетевого встраивания может быть статическим, централизированным и без резервирования одновременно. На основе этого определяются классы алгоритмов и описываются следующим синтаксисом: S|D, C|D, C|R [17]. Первый символ обозначает, является ли алгоритм статическим или динамическим (Static или Dynamic), второй символ обозначает, является ли алгоритм централизированным или распределенным (Centralized или Distributed) третий символ определяет наличие резервирования (Concise или Redundant).

Для использования в сетях 5G наиболее подходят динамические модели виртуального сетевого встраивания. Одна из первых моделей виртуального сетевого встраивания с возможностью динамической реконфигурации сетевых ресурсов описана в [18]. Уже тогда авторы определили ряд проблем, с которыми необходимо бороться. В первую очередь это частая реконфигурация сетевых ресурсов. В этом случае увеличиватются требования к производительности узла, отвечающего за реконфигурацию ресурсов, пропорционально частоте реконфигурации. Во-вторых, при таком подходе увеличивается вероятность отказа в предоставлении сервиса из-за частой реконфигурации ресурсов. Для решения этих проблем предлагается масштабировать только те виртуальные сегменты, для которых требуется перераспределение ресурсов в данный момент времени, а не для всех сегментов. Основным ограничением данной модели является то, что она основана только на анализе топологии виртуальной сети.

Каждый алгоритм содержит определенные параметры, на основе которых происходит встраивание. Этими параметрами могут быть емкость узла сети-субстрата, выраженная в CPU, расположение узлов, полоса пропускания соединений сети-субстрата, задержка и т.д. Поэтому для упрощения сравнения наиболее известных алгоритмов их основные характеристики были вынесены в в табл. 1:

Таблица 1

Модели виртуального сетевого встраивания

Модель	Алгоритм	Сущность	Параметры	Преимущество алгоритма
D/C/C	Cai et al. (2010) [19]	Сеть-субстрат Виртуальная сеть	Топология CPU Задержка	Масштабирование на основе изменений сети-субстрата
	Zhu and Ammar (2006) [18]	Сеть-субстрат Виртуальная сеть	Топология	Уменьшение количества операций масштабирования
	Fan and Ammar (2006) [20]	Сеть-субстрат	Топология	Уменьшение количества операций масштабирования
	Fajjari et al. (2011) [14]	Сеть-субстрат Виртуальная сеть	Топология CPU Пропускная способность Объем памяти	Миграция виртуальных узлов из зоны «узкого горла» сети
	Bienkowski et al. (2010) [21]	Пользователи Постащик инфраструктуры Поставщик услуг	Задержка Пропускная способность	Миграция виртуальных узлов при изменении местоположения пользователей
	Shun-li and Xue-song (2011) [22]	Сеть-субстрат Виртуальная сеть	Топология CPU Пропускная способность	Перераспределение ресурсов виртуальных ресурсов при неоптимальном встраивании

	Sun et al. (2012) [23]	Сеть-субстрат Виртуальная сеть	Топология CPU Пропускная способность	Определение проблем виртуального сетевого встраивания
	Sajjad Gholamipour et al. [24]	Сеть-субстрат Виртуальная сеть	Топология CPU Пропускная способность Задержка Объем памяти Утилизация пула IP	Снижение вероятности отказа в предоставлении ресурсов сети-субстрата, снижение энергопотребления
	DAA [12]	Сеть-субстрат Виртуальная сеть	Топология CPU Пропускная способность	Максимизация количества встраиваемых виртуальных ресурсов в сеть-субстрат
D/D/C	Marquezan et al. (2010) [25]	VNFM	Время прерывания сервиса Объем памяти	Первый распределенный алгоритм виртуального сетевого встраивания
D/C/R	Yu et al. (2010) [26]	Физическая сеть Виртуальная сеть	Топология	Виртуальное сетевое встраивание на основе параметров сетевых соединений сети-субстрата
	Butt et al. (2010) [27]	Сеть-субстрат Виртуальная сеть	CPU Пропускная способность	Перераспределение ресурсов при увеличении вероятности отказа в предоставлении сервиса
	Schaffrath et al. (2010) [28]	Сеть-субстрат Виртуальная сеть	Топология CPU Пропускная способность	Динамическое перераспределение виртуальных ресурсов
	Chen et al. (2011) [29]	Сеть-субстрат Виртуальная сеть	Топология CPU Пропускная способность	Периодическое перераспределение ресурсов высоконагруженных элементов сети-субстрата
D/D/R	Houidi et al. (2010) [30]	Постащик инфраструктуры Поставщик виртуализированной инфраструктуры	Приоритет узла Приоритет соединения	Высокая отказоустойчивость при выходе из строя узлов или соединений сети-субстрата

Почти все динамические модели сетевого встраивания, описанные выше, являются реактивными. Реактивные модели приводит к непредсказуемым и часто значительным задержкам при управлении ресурсами сетевых сегментов, поскольку необходимо иметь возможность быстрого динамического масштабирования виртуализированых ресурсов.

Формулирование задачи динамического масштабирования сетевых сегментов

Основной вопрос при развертывании сетевых сегментов заключается в снижении затрат операторов связи на эксплуатацию инфрастукруры, обеспечивая при этом предоставление услуг с требуемым QoS. Как было отмечено ранее, большинство исследований были сосредоточены на статической модели развертывании сетевых сегментов. Однако, процесс запроса сетевых сегментов и выделения для них ресурсов является динамическим, распределение пользователей по сегментам неравномерно, а статическое сетевое сегментирование приводит к напрасной трате ресурсов [31].

Во-вторых, определение сетевых узлов для сетевых сегментов влияет на маршрутизацию пользовательского трафика, что также сказывается на задержках, возникающих при обработке трафика элементами сети. Поэтому необходимо определить модель динамического развертывания сетевых сегментов и архитектуру управления и оркестровки сетевых сегментов. В дополнение к этому необходимо найти подходящий алгоритм для адаптивного масштабирования сетевых сегментов на основе анализа текущей утилизации элементов виртуальной сети.

Сети 5G используют принципы виртуализации сетевых функций (NFV) и программно-определяемых сетей (SDN) для достижения гибкости в управлении сетевыми ресурсами [32]. Поэтому структуру сети 5G на основе данных концепций можно представить в следующем виде (рис. 4). За управление ресурсами виртуализированной инфраструктуры отвечает модуль менеджмента и оркестрации (Management and Orchestration). В свою очередь MANO включает оркестратор NFV (NFVO), отвечающий за управление и мониторинг за виртуальными ресурсами, и блок управления VNF (VNFM), который отвечает за жизненный цикл VNF.

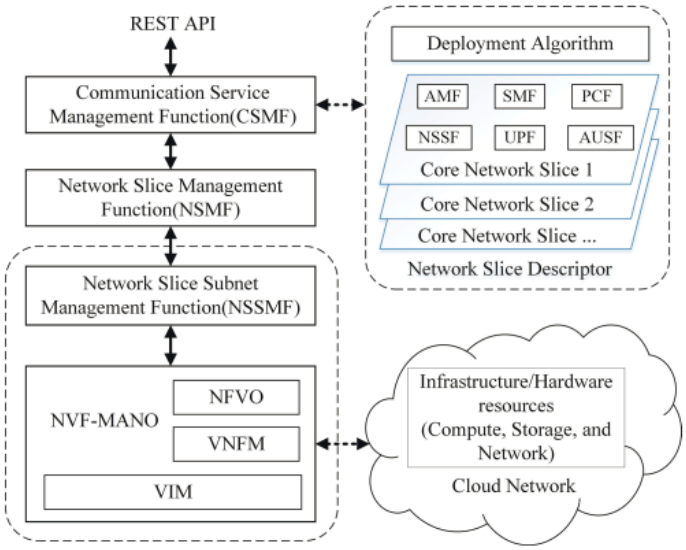

Рис. 4. Структура сети 5G на основе NFV

С технической точки зрения решения, реализующие NFV на различных сетевых уровнях, хорошо зарекомендовали себя. Примерами являются современные архитектуры платформ MANO, такие как ETSI NFV [33], OSM [34] или Cisco Elastic Services Controller 5.3 ETSI NFV MANO [35], которые

поддерживают динамическую реконфигурацию ресурсов VNF. Использование искусственного интеллекта и машинного обучения при эксплуатации мобильных сети находится на ранней стадии. В настоящее время выделение ресурсов для VNF представляет собой реактивный процесс т.е. основанный на установлении предопределенного гистерезисного порога [36]. Однако масштабирование ресурсов занимает некоторое время, так как время развертывания виртуальной машины (VM) может колебаться от нескольких секунд до нескольких минут, а современным программным решениям, таким как Kubernetes, требуется несколько секунд для инициализации новых модулей (PODs).

Эти неизбежные временные задержки влияют на качество услуг, предоставляемых сетевыми сегментами, использующих устаревший реактивный подход к управлению ресурсами. Поэтому возникает потребность в упреждающих, автоматизированных решениях, которые обеспечивают экономичное использование ресурсов сетевой инфраструктуры за счет прогнозирования будущих потребностей в сетевых ресурсах и их своевременного перераспределения именно там и тогда, где и когда они потребуются.

Модели динамического масштабирования сетевых сегментов

Модели динамического масштабирования можно разделить на две группы: основанные на пороговых значениях и основанные на прогнозируемом значении будущей утилизации сетевых ресурсов (анализ временных рядов, использование машинного обучения и т.д.). Пороговые модели используют предопределенные верхний и нижний пороги для масштабирования. Предопределенные верхний порог и нижний порог используются для любого параметра, характеризующего производительность.

Если показатель производительности выше верхнего порога за определенный период, будет осуществлено масштабирование в сторону увеличения количества требуемых ресурсов. И наоборот, масштабирование в сторону уменьшения количества задействованых ресурсов в обратной ситуации. Модели на основе предопределенного порога просты и удобны в реализации. Однако все эти модели являются реактивным, что может привести к нарушениям в SLA. Кроме того, пороги трудно выбрать, и, возможно, их придется часто менять.

Поэтому были предложены улучшенные модели с использованием динамического определения порогового значения [37-38]. В [38] представляют новый метод реализации динамического определения порога при изучении динамической консолидации параметров VM. Динамически изменяемые пороги получаются на основе оценки распределения и статистики использования CPU для физических хостов.

В других исследованиях, основанных на прогнозировании, использовались метод авторегрессии (AR) [39], метод скользящего среднего (MA) [40] и авторегрессионного скользящего среднего (ARMA) [41]. Данные методы обеспечивают ресурсоэффективный подход к автоматическому масштабированию на основе использования обучения с подкреплением. Обучение с подкреплением выполняется без каких-либо знаний о предшествующей модели трафика или информации о предыдущем масштабировании.

Прогнозирование использования ресурсов VM в центрах обработки данных является одной из основных областей исследований. Эти исследование фокусируются на прогнозировании временных рядов утилизации CPU VM виртуальных сетевых функций. Примером данных исследований являются модели авторегрессионного интегрированного скользящего среднего (ARIMA) [42], Хольт-Винтерса [43], различные типы и разновидности долгой краткосрочной памяти (LSTM)[44]: CNN-LSTM [45], XLSTM [46], multi-LSTM [47] и Bi-LSTM [48].

Все эти модели основаны на прогнозировании временных рядов (time series forecasting). Временной ряд относится к последовательности наблюдений, в который записываются параметры конкретных действий в течение определенного периода времени [49]. Так как наблюдения являются временными выборками, то они коррелируют между собой. Прогнозирование деятельности или наблюдения обычно включает в себя сопоставление исторических временных рядов и поиск в них закономерностей. Определение трендов и сезонных закономерностей данных временных рядов может быть определено на графике временных рядов и составляет первый шаг в любой задаче прогнозирования временных рядов.

Как правило, преобразование временного ряда в сглаженную версию и последующее его построение позволяет выявить неизвестные ранее закономерности. Одним из таких широко используемых методов сглаживания является метод скользящего среднего. Метод скользящего среднего используется для измерения сезонных вариаций во временном ряду путем вычисления среднего арифметического значений для временных интервалов во временном ряду и снижения любой волатильности данных.

Для определения случайности данных, вводится понятие стационарности. Временной ряд называется строго стационарным, если распределение вероятностей набора значений в этом временном ряду остается неизменным даже при смещении набора во времени. Многие из реальных временных рядов данных являются нестационарными по своей природе и, следовательно, требуют корректировки данных, чтобы сделать их стационарными путем удаления трендов, которые по своей природе увеличиваются или уменьшаются. Одним из таких широко используемых методов является дифференцирование временных рядов первого порядка. Операция дифференцирования выполняется путем вычитания предыдущих значений из текущих значений. Также дифференцирование может применяться последовательно для дальнейшего полного удаления трендов за счет потери все большего количества информации. В данном случае АКФ временного ряда — это степень корреляции текущих значений с прошлыми значениями.

Белый шум представляет временной ряд, в котором каждое наблюдение случайным образом берется из совокупности наблюдений с дисперсией и средним значением, равным нулю. Как правило, временные ряды должны следовать такому шаблону для более лучшего прогнозирования, и любое отклонение в этом случае критично. Модели авторегрессии (AR) и скользящего среднего (MA) помогают исправить эти отклонения. Модели ARIMA используются в качестве фильтра, который разделяет сигнал и шум, после чего сигнал используется для предсказания будущих значений.

Прогнозирование с использованием ARIMA осуществляется с помощью линейного уравнения, где предикторы представляют собой лаги зависимых переменных или ошибок прогнозирования. Модель ARIMA определяется параметрами ARIMA (p, d, q), где: p – общее количество членов авторегрессии, т. е. условий авторегрессии, d – общая разность, необходимая для стационарности, а q – общие ошибки прогнозирования с запаздыванием. В [42] на основе модели ARIMA определяется прогнозирование загрузки системы и оценивается точность предсказания на основе реальных запросов к VM.

В [43] описывается алгоритм прогнозирования Хольт-Винтерса (HW) для предсказания параметров трафиковой модели сетевых сегментов. В основном этот метод предназначен для предотвращения частого изменения сетевой топологии. Также исследуется динамическое развертывание сетевых сегментов, определяется архитектура управления и оркестровки ими. Этот алгоритм сводит к минимуму ошибки прогнозирования что позволяет снизить задержку создания VNF для сетевых сегментов. Немаловажной является и описанная стратегия адаптивного масштабирования VNF для определения требуемого количества задействованных ресурсов.

Другим немаловажным алгоритмом прогнозирования является LSTM и его разновидности, описанные в [44-48]. Сети LSTM – это класс рукуррентных нейронных сетей (RNN), которые используются для интерпретации и изучения долгосрочных зависимостей [50]. Особенностью алгоритма LSTM является способность сохранять информацию в течение длительного периода времени. LSTM состоит из трех уровней: слой фильтра забывания, слой входного фильтра и слой выходного фильтра (рис. 5) [51].

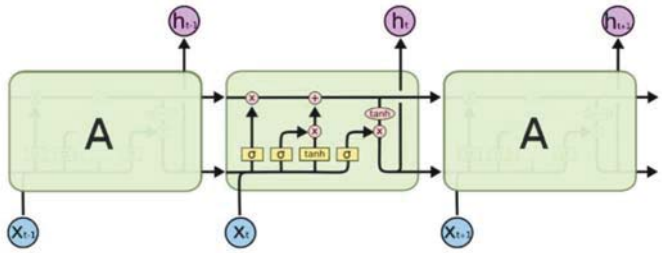

Рис. 5. Модель LSTM алгоритма

Слой фильтра забывания определяет, какая часть предыдущих данных будет забыта и какая часть предыдущих данных будет использоваться на следующием этапе. Значение этого вентиля находится в диапазоне 0-1. «0» определяет забытие предыдущих данных, «1» определяет использование предыдущих данных.

Слой входного фильтра включает слой *tanh* и отвечает за получение новых данных. Ненужная часть входных данных отфильтровывается сигмовидной функцией, затем с помощью функции *tanh* определяются новые возможные данные. Умножение результата на выходе сигмовидной функции и результата на выходе функции tanh определяет обновление и получение нового состояния ячейки.

Слой выходного фильтра определяет состояние ячейки с помощью функции tanh. Входные данные фильтруются сигмовидной функцией. Умножение результата фильтрации сигмовидной функцией и результата на выходе функции *tanh* определяет выходные данные.

Более усовершенствованным алгоритмом прогнозирования является двунаправленная LSTM (Bi-LSTM) (рис. 6) [52]. Алгоритм Bi-LSTM представляет процесс, при котором любая нейронная сеть получает информацию о последовательности в обоих направлениях: в прямом (из прошлого в будущее) и обратном (из будущего в прошлое). В этом случае происходит обработка входной последовательности в обоих направлениях, по сравнению с обычной LSTM.

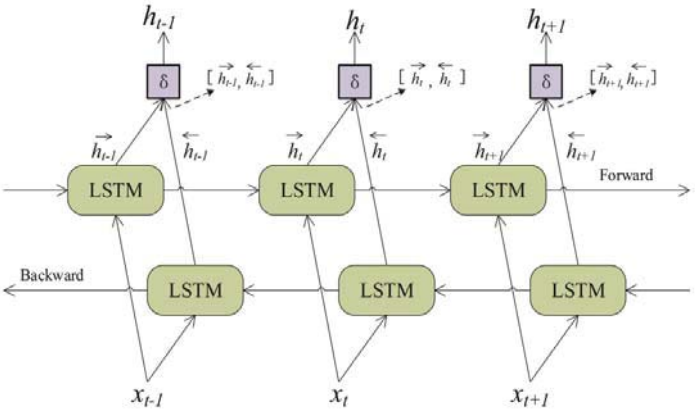

Рис. 6. Модель Bi-LSTM алгоритма

В [48] рассматривается новый, эффективный механизм упреждающего прогнозирования требуемых ресурсов с использованием кодера-декодера на основе Bi-LSTM. Модель на основе двунаправленной LSTM (Bi-LSTM) с механизмом внимания, предназначена для многомерного прогнозирования временных рядов. Она может достигать высокой точности как при краткосрочном, так и при долгосрочном прогнозировани, благодаря чему система отслеживает утилизацию ресурсов VNF, прогнозирует утилизацию ресурсов в будущем и автоматически добавляет или удаляет компоненты VNF для каждого сетевого сегмента.

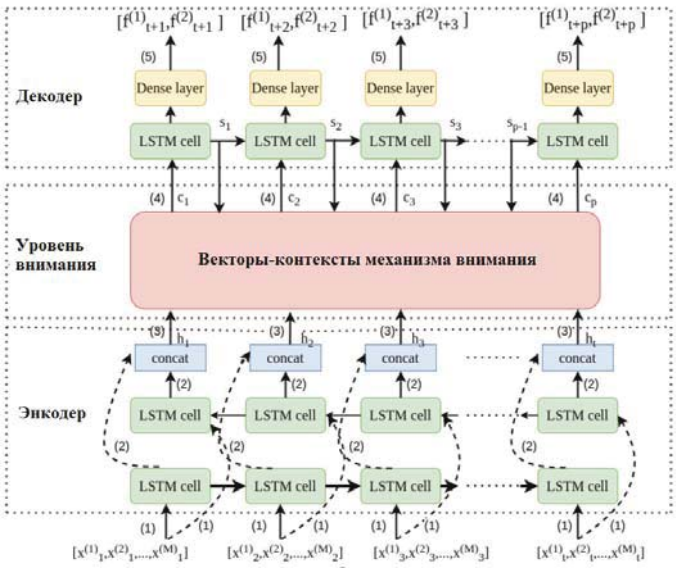

Рис. 7. Архитектура Bi-LSTM с механизмом внимания

Архитектура алгоритма Bi-LSTM с механизмом внимания представлена на рисунке 7 [38] и состоит из следующих компонент: Bi-LSTM в качестве компонента энкодера, LSTM в

качестве компонента декодера и механизма внимания. Bi-LSTM изучает скрытое представление последовательности входных данных (1) и извлекает признаки глубокой временной зависимости и корреляции из многомерного временного ряда (2, 3).

Затем последовательность на выходе кодера помещается во временной слой механизма внимания с выходными данными слоя декодера для построения векторов контекста внимания. Декодер LSTM обрабатывает векторы контекста внимания, чтобы определить будущую утилизацию ресурсов.

Для оценки алгоритмов используются следующие показатели: средняя квадратичная ошибка (MSE), корень из средней квадратичной ошибки (RMSE), средняя абсолютная ошибка (MAE) и коэффициент детерминации (R^2).

Средняя квадратичная ошибка определяет среднеквадратическую разницу между прогнозируемым и фактическим значениями:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}((x(i) - y(i))^2 \quad (1)$$

Корень из средней квадратичной ошибки определяет квадратный корень разницы между прогнозируемым и фактическим значениями:

$$RMSE = \sqrt{MSE} \quad (2)$$

Средняя абсолютная ошибка определяет среднее абсолютное различие между прогнозируемым и фактическим значениями:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x(i) - y(i)| \quad (3)$$

Коэффициент детерминации определяет квадрат множественного коэффициента корреляции между фактическим и прогнозируемым значением:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}((x(i)-y(i))^2}{\sum_{i=1}^{n}((x(i)-\overline{y(i)})^2} \quad (4)$$

где $x(i)$ – фактическое значение, $y(i)$ – прогнозируемое значение, $\overline{y(i)}$ – среднее значение $x(i)$ в момент времени i.

Также в [38] проведено сравнение некоторых алгоритмов динамического масштабирования, описаных выше. Результаты оценки представлены в табл. 2.

Таблица 2

Сравнение параметров моделей динамического масштабирования

Модель	MSE	RMSE	MAE	R^2
Хольт-Уинтерс	12,50	3,53	3,21	0,87
CNN-LSTM	11,10	3,33	3,13	0,89
XLSTM	11,30	3,36	3,15	0,90
multi-LSTM	10,50	3,24	2,80	0,95
Bi-LSTM с механизмом внимания	9,30	3,05	1,82	0,97

Модель Bi-LSTM с механизмом внимания показывает некоторое превосходство, по сравнению с остальными моделями, что позволяет определять количество требуемых для VNF ресурсов с более высокой точностью.

Все описаные выше модели динамического масштабирования основаны на использовании различных параметров для предсказания утилизации сетевых ресурсов в будущем. Основные характеристики моделей вынесены в таблице 3.

Таблица 3

Сравнение параметров моделей динамического масштабирования

Модель	Возможность предсказания утилизации ресурсов	Возможность предсказания параметров трафиковой модели	Возможность VNE
ARIMA	Нет	Да	Нет
Хольт-Уинтерс	Да	Да	Нет
CNN-LSTM	Да	Нет	Нет
XLSTM	Нет	Да	Нет
multi-LSTM	Да	Нет	Нет
Bi-LSTM с механизмом внимания	Да	Нет	Нет

На основе представленных характеристик можно сделать вывод, что при определении проблемы динамического масштабирования рассматривается только возможность перераспределения ресурсов на основе предсказания утилизации ресурсов VM. При этом возможность предсказания утилизации ресурсов на основе параметров трафиковой модели определяется только в некоторых моделях.

Это может быть связано с тем, что изначально модели динамического масштабирования виртуализированных ресурсов рассамтривались только в разрезе эксплуатации ЦОД, где более важными параметрами являются утилизация CPU и памяти VM. В этом случае использование параметров трафиковой модели только усложняет работу данных алгоритмов. Но в разрезе эксплуатации сетей 5G и использования сетевого сегментирования необходимо также учитывать параметры трафиковой модели, размер буфера обслуживающих VM и маршрутизаторов, утилизацию IP-пулов и задержку в виртуальных сетевых соединениях для гарантирования требуемых параметров QoS в каждом из вариантов использования.

Заключение

В данной статье были рассмотрены подходы к решению проблем виртуального сетевого встраивания (VNE) и динамического масштабирования ресурсов VNF сетей 5G. Почти во всех исследованиях оба этих подхода рассматриваются отдельно, так как это две совершенно разные задачи, которые решаются с помощью различных по функциональности алгоритмов. Большинство моделей динамического масштабирования ресурсов основаны только на использовании параметров утилизации виртуализированной инфраструктуры. Исключением является [43], где масштабирование виртуальных ресурсов для сетевых сегментов основано на предсказании будущей утилизации ресурсов виртуализированной инфраструктуры и анализе параметров трафиковой модели.

Также в статье представлено описание и сравнение различных моделей динамического масштабирования сетевых сегментов. Использование модели Bi-LSTM с механизмом внимания демонстрирует некоторое превосходство показателей, на основе которых производилось сравнение моделей. В описаных выше решениях не приводится анализ параметров

быстродействия моделей динамического масштабирования виртуальных сетевых ресурсов, что в свою очередь является важным критерием при оценке вероятности возникновения отказа в предоставлении услуг из-за несвоевременного уведомления NFVO о необходимости масштабирования ресурсов VNF.

Таким образом целью будущих исследований является разработка методологии построения модели управления ресурсами сетевых сегментов для обеспечения возможности виртуального сетевого встраивания и масштабирования ресурсов на основе анализа топологии сети, предсказания параметров трафиковой модели и утилизации ресурсов виртуализированной инфраструктуры.

Литература

1. 3GPP TS 23.501: “System Architecture for the 5G System; Stage 2”.
2. *P. Rost, C. Mannweiler, D. S. Michalopoulos, C. Sartori, V. Sciancalepore, N. Sastry, O. Holland, S. Tayade, B. Han, D. Bega, D. Aziz, and H. Bakker*, “Network slicing to enable scalability and flexibility in 5G mobile networks”, IEEE Commun. Mag., vol. 55, no. 5, pp. 72-79, May 2017.
3. *V.G. Nguyen, A. Brunstrom, K.-J. Grinnemo, and J. Taheri*, “SDN/NFVbased mobile packet core network architectures: A survey’’, IEEE Commun. Surveys Tuts., vol. 19, no. 3, pp. 1567-1602, 3rd Quart., 2017.
4. *A. Rizwan, M. Jaber, F. Filali, A. Imran, and A. Abu-Dayya*, “A zero-touch network service management approach using AI-enabled CDR analysis”, IEEE Access, vol. 9, pp. 157699-157714, 2021.
5. *A. Fischer, J. F. Botero, M. T. Beck, H. De Meer, and X. Hesselbach*, “Virtual network embedding: A survey”, IEEE Commun. Surv. Tuts., vol. 15, no. 4, pp. 1888-1906, Jan 2013.
6. *A. Haider, R. Potter, and A. Nakao*, “Challenges in resource allocation in network virtualization”, in 20th ITC Specialist Seminar, vol. 18, 2009, p. 20.
7. *A. Belbekkouche, M. Hasan, and A. Karmouch*, “Resource discovery and allocation in network virtualization”, IEEE Commun. Surveys Tutorials, vol. PP, no. 99, pp. 1-15, 2012.
8. *D.G. Andersen*, “Theoretical approaches to node assignment,” Dec. 2002, unpublished Manuscript.
9. *M. Leconte, G. S. Paschos, P. Mertikopoulos, and U. C. Kozat*, “A resource allocation framework for network slicing”, in Proc. IEEE Conf. Comput. Commun. (INFOCOM), Apr. 2018, pp. 2177-2185.
10. *S. Vassilaras* et al., “The Algorithmic Aspects of Network Slicing”, IEEE Commun. Mag., vol. 55, no. 8, pp. 112-19, 2017.
11. *Anuar Othman, Nazrul A. Nayan, Siti N. H. S. Abdullah*, “Automated Deployment of Virtual Network Function in 5G Network Slicing Using Deep Reinforcement Learning”, IEEE Access, vol.10, pp. 61065-61079, 2022.
12. *Linh Le, Tu N. Nguyen, Kun Suo, Jing (Selena) He*, “Efficient Embedding VNFs in 5G Network Slicing: A Deep Reinforcement Learning Approach”. URL: https://arxiv.org/abs/2207.11822. (Дата обращения: 02.12.2022).
13. *Han, Bin & Schotten, Hans*, “Machine Learning for Network Slicing Resource Management: A Comprehensive Survey”, ZTE Communications, 68(4), pp. 27-32, 2019
14. *I. Fajjari, N. Aitsaadi, G. Pujolle, and H. Zimmermann*, “Vnr algorithm: A greedy approach for virtual networks reconfigurations”, in Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE, pp. 1-5, DEC. 2011.
15. *R N. Tran, L. Casucci, and A. Timm-Giel*, “Optimal mapping of virtual networks considering reactive reconfiguration” in accepted for IEEE International Conference on Cloud Networking (Cloudnet'12), Paris, France. Nov 2012.
16. *Gao and G. N. Rouskas*, “Virtual network reconfiguration with load balancing and migration cost considerations”, IEEE INFOCOM, 2018.
17. *H. Cao, H. Hu, Z. Qu and L. Yang*, “Heuristic solutions of virtual network embedding: A survey”, China Communications, vol. 15, no. 3, pp. 186-219, 2018.
18. *Zhu and M. Ammar*, “Algorithms for assigning substrate network resources to virtual network components”, in INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proc., pp. 1-12, 2006.
19. *Z. Cai, F. Liu, N. Xiao, Q. Liu, and Z. Wang*, “Virtual network embedding for evolving networks”, in Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE, pp. 1 –5, Dec. 2010.
20. *J. Fan and M. H. Ammar*, “Dynamic topology configuration in service overlay networks: A study of reconfiguration policies”, in INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings, april 2006, pp. 1-12.
21. *M. Bienkowski, A. Feldmann, D. Jurca, W. Kellerer, G. Schaffrath, S. Schmid, and J. Widmer*, “Competitive analysis for service migration in vnets”, in Proc. second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures, ser. VISA ’10. New York, NY, USA: ACM, 2010, pp. 17-24.
22. *Z. Shun-li and Q. Xue-song*, “A novel virtual network mapping algorithm for cost minimizing”, Cyber Journals: J. Sel. Areas Telecommunications (JSAT), vol. 02, no. 01, pp. 1-9, January 2011.
23. *G. Sun, H. Yu, V. Anand, and L. Li*, “A cost efficient framework and algorithm for embedding dynamic virtual network requests”, Future Generation Computer Systems, no. 0, 2012, available online 25 August 2012.
24. *Sajjad Gholamipour* et al., “Online Admission Control and Resource Allocation in Network Slicing under Demand Uncertainties”. URL: https://arxiv.org/abs/2108.03710. (Дата обращения: 05.12.2022).
25. *C. Marquezan, L. Granville, G. Nunzi, and M. Brunner*, “Distributed autonomic resource management for network virtualization,” in Network Operations and Management Symposium (NOMS), 2010 IEEE, april 2010, pp. 463 –470.
26. *M. Yu, Y. Yi, J. Rexford, and M. Chiang*, “Rethinking virtual network embedding: Substrate support for path splitting and migration”, ACM SIGCOMM CCR, vol. 38, no. 2, pp. 17–29, Apr. 2008.
27. *N.F. Butt, N. M.M.K. Chowdhury, and R. Boutaba*, “Topology-awareness and reoptimization mechanism for virtual network embedding”, in Networking, 2010, pp. 27-39.
28. *G. Schaffrath, S. Schmid, and A. Feldmann*, “Optimizing long-lived cloudnets with migrations”, in Proc. 5th IEEE/ACM International Conference on Utility and Cloud Computing (UCC), 2012.
29. *D. Chen, X. Qiu, Z. Qu, S. Zhang, and W. Li*, “Algorithm for virtual nodes reconfiguration on network virtualization”, in Advanced Intelligence and Awareness Internet (AIAI 2011), 2011 International Conference on, oct. 2011, pp. 333-337.
30. *I. Houidi, W. Louati, D. Zeghlache, P.* Papadimitriou, and L. Mathy, “Adaptive virtual network provisioning”, in Proc. second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures, ser. VISA ’10. New York, NY, USA: ACM, 2010, pp. 41-48.
31. *H. Halabian*, “Distributed resource allocation optimization in 5G virtualized networks”, IEEE J. Sel. Areas Commun., vol. 37, no. 3, pp. 627-642, Mar. 2019.
32. ETSI GS NFV 002 ETSI, “Network Functions Virtualization (NFV); Architectural Framework” v1.1.1, 2013.
33. ETSI GS NFV “Network Function Virtualization (NFV) Management and Orchestration,” NFV-MAN, vol. 1, Dec. 2014.
34. ETSI, “Open Source MANO (OSM) Project.” URL: https://osm.etsi.org/. (Дата обращения: 20.12.2022).
35. Cisco Elastic Services Controller 5.3 ETSI NFV MANO. URL: https://www.cisco.com/c/en/us/td/docs/net_mgmt/elastic_services_controller/5-3/etsi/ guide/Cisco-Elastic-Services-Controller-ETSI-User-Guide-5-3.pdf. (Дата обращения: 20.12.2022).

36. *D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Pérez*, "DeepCog: Optimizing resource provisioning in network slicing with AI-based capacity forecasting", IEEE J. Sel. Areas Commun., vol. 38, no. 2, pp. 361376, Feb. 2020.

37. *H. C. Lim, S. Babu, J. S. Chase, and S. S. Parekh*, "Automated control in cloud computing: challenges and opportunities", Proc., 1st workshop on Automated Control for Datacenters and Clouds, pp. 13-18, 2009.

38. *A. Beloglazov and R. Buyya*, "Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers", Proc., 8th Intl. Wksp on Middleware for Grids, Clouds & e-Science, 2010.

39. *A. Chandra, W. Gong, and P. Shenoy*, "Dynamic resource allocation for shared data centers using online measurements", Proc., 11th Intl. Conf. on Quality of Service, pp. 381398, 2003.

40. *H. Mi, H. Wang, G. Yin, Y. Zhou, D. Shi, and L. Yuan*, "Online selfreconfiguration with performance guarantee for energy-efficient largescale cloud computing data centers", Proc., IEEE Intl. Conf. on Services Computing, 2010.

41. *W. Fang, Z. Lu, J. Wu, and Z. Cao*, "RPPS: a novel resource prediction and provisioning scheme in cloud data center", IEEE 9th Intl. Conf. on Services Computing, pp. 609616, 2012.

42. *R. Calheiros, E. Masoumi, R. Ranjan, and R. Buyya*, "Workload prediction using ARIMA model and its impact on cloud applications' QoS", IEEE Trans. Cloud Computing, vol. 3, no. 4, pp. 449- 458, Aug. 2014.

43. *J. Zhou, W. Zhao, and S. Chen*, "Dynamic network slice scaling assisted by prediction in 5G network", IEEE Access, vol. 8, pp. 133700-133712, 2020.

44. *D. Janardhanan and E. Barrett*, "CPU workload forecasting of machines in data centers using LSTM recurrent neural networks and ARIMA models", in Proc. 12th Int. Conf. Internet Technol Secured Trans. (ICITST), Dec. 2017, pp. 55-60.

45. *S. Ouhame, Y. Hadi, and A. Ullah*, "An efficient forecasting approach for resource utilization in cloud data center using CNN-LSTM model," Neural Comput. Appl., vol. 33, no. 16, pp. 10043-10055, Aug. 2021.

46. *C. Gutterman, E. Grinshpun, S. Sharma, and G. Zussman*, "RAN resource usage prediction for a 5G slice broker", in Proc. 20th ACM Int. Symp. Mobile Ad Hoc Netw. Comput., Jul. 2019, pp. 231-240.

47. *C. N. Nhu and M. Park*, "Optimizing resource scaling in network slicing", in Proc. Int. Conf. Inf. Netw. (ICOIN), Jan. 2022, pp. 413-416.

48. *Chien-Nguyen Nhu, Minho Park*, "Dynamic Network Slice Scaling Assisted by Attention-Based Prediction in 5G Core Network", IEEE Access, vol.10, pp. 72955-72972, 2022.

49. *Rainer Schlittgen*, "Robert H. Shumway and David S. Stoffer: Time series analysis and its applications with R examples, 2nd edn.", In: AStA Advances in Statistical Analysis 92.2 (2008), pp. 233–234.

50. Understanding LSTM Networks – colah's blog. URL: http://colah.github.io/posts/2015-08-Understanding-LSTMs. (Дата обращения: 15.12.2022).

51. *Akın, Cihan, Kacar, Umit, Kirci, Murvet*, "Twins Recognition Using Hierarchical Score Level Fusion", 2019.

52. *Zheng, Xiao, Chen, Wanzhong*, "An Attention-based Bi-LSTM Method for Visual Object Classification via EEG", Biomedical Signal Processing and Control, vol. 63, 2021.

ANALYSIS OF NETWORK RESOURCE SCALING MODELS IN 5G NETWORK

Vasily S. Elagin, *The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, St. Petersburg, Russia,* ***elagin.vas@gmail.com***

Anton S. Vasin, *The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, St. Petersburg, Russia,* ***antoshca-vasin@yandex.ru***

Abstract

This article analyzes the existing models of virtual network embedding and dynamic scaling of virtual network resources for Network Slices. These models make it possible to provide services with the required QoS while respecting the concept of efficient network resource usage. Dynamic virtual network embedding models allow virtual networks to be efficiently mapped on a substrate network and reconfigured on demand. But for more flexible dynamic scaling, Holt-Winters, Bi-LSTM and etc. models are additionally used, which are built according algorithms for predicting future resource utilization in order to reduce the initialization time of virtual network function instances serving certain Network Slices. Dynamic scaling models are compared and conclusions about the possibility of their use are given. As a conclusion we were made a summary about the possibility of using these models and the necessity of improvements for more flexible use in 5G networks.

Keywords: *, SDN, 5G Networks, Network Slicing, VNE, SN, virtual node mapping, virtual link mapping, deep learning, resource scaling, MANO, VNFM, NFVO, LSTM.*