

## Перспективы применения оптимизаторов при проектировании корпоративной компьютерной сети

*П.О. Сиротин, В.Э. Шмаков*

*Санкт-Петербургский политехнический университет Петра Великого*

**Аннотация:** В статье рассматривается проблема оптимизации глобальных сетей, а также существующие на данный момент программно-аппаратные решения. Цель исследования – определение технологической основы для разработки прототипа отечественного оптимизатора глобальной компьютерной сети. При исследовании предметной области выяснилось, что отечественных решений в данной сфере, находящихся в свободном доступе, не существует. Полученное решение можно будет приспособить под конкретные требования компании-заказчика с помощью добавления необходимых модификаций в прототип.

**Ключевые слова:** глобальная сеть, дедупликация данных, оптимизатор, пропускная способность.

### Введение

Организация корпоративной сети передачи данных – необходимый этап, который должна пройти каждая компания, заинтересованная в создании комфортного и безопасного рабочего процесса. Организация корпоративной сети заключается не только в выборе сетевого оборудования, но и в разделении групп устройств на филиалы, офисы и подразделения, а также в реализации возможности дистанционного подключения сотрудников к рабочим устройствам.

Как правило, специалисты выделяют два основных вида компьютерных сетей – локальные (Local Area Network – LAN) и глобальные (Wide Area Network – WAN). Технология LAN позволяет объединять устройства, находящиеся в пределах одного предприятия (офиса, филиала), тогда как технология WAN объединяет устройства, находящиеся на большом отдалении друг от друга. Вне зависимости от того, насколько большая компания, сетевым администраторам приходится работать как с локальными (объединение сотрудников в подразделения и офисы), так и с глобальными сетями (простейший пример – доступ сотрудников в Интернет, который

является самой популярной глобальной сетью). Если же говорить о крупных компаниях, имеющих несколько филиалов, то для объединения данных узлов тоже используется глобальная сеть.

Технология WAN при этом имеет ряд существенных недостатков в сравнение с технологией LAN. Среди них стоит выделить увеличенную задержку передачи данных и повышенный процент потерь пакетов, причинами возникновения которых являются: расстояние между устройствами, низкая полоса пропускания канала связи и некоторые физические факторы. При этом скорость передачи данных по каналу связи WAN (от нескольких Кбит/с до десятков Мбит/с) кратно меньше скорости передачи данных в LAN (сотни и тысячи Мбит/с). Наличие большого количества потерь пакетов данных, а также низкая пропускная способность каналов делают довольно затруднительным, например, проведение видеоконференций между сотрудниками из разных филиалов. Кроме того, стоит учитывать, что, в частности, транспортный протокол передачи данных (Transmission Control Protocol – TCP) не отличается эффективностью и быстродействием при использовании в глобальных сетях. Причиной этому являются присущие любой WAN потери пакетов, поскольку встроенный в протокол TCP механизм подтверждения передачи пакетов крайне чувствителен к любым потерям пакетов данных.

Задача оптимизации глобальных сетей заключается как в увеличении пропускной способности канала, так и в снижении процента потерь пакетов, а также в уменьшении длительности задержки передачи данных. Объединение некоторых решений данных подзадач было реализовано в качестве устройства, которое впоследствии было названо оптимизатором WAN.

В современных оптимизаторах WAN реализованы такие механизмы, как сжатие данных, TCP-оптимизация, дедупликация данных и оптимизация

---

используемых приложений и др. (работа с качеством услуг (quality of service – QoS) – куда более редкий подход, встречающийся, например, в устройствах компании Citrix).

Оптимизаторы WAN, как правило, являются программно-аппаратными решениями, но могут быть реализованы в программном виде и на основе виртуальной машины. Существующие решения в данной сфере практически все являются проприетарными и при этом принадлежат иностранным компаниям (Cisco, Blue Coat, Riverbed, Citrix), тогда как отечественных аналогов, находящихся в свободном доступе, попросту не существует.

### **Постановка задачи**

Задачей работы является анализ имеющихся решений в области оптимизации WAN, а также определение технологической базы для разработки прототипа отечественного оптимизатора WAN.

Несмотря на то, что подавляющее большинство решений, представленных на рынке, являются проприетарными, обратиться к ним следует как минимум для изучения реализованных в таких оптимизаторах алгоритмах. Что же касается решений, находящихся в свободном доступе, то сперва следует установить, представляет ли возможных их модификация для улучшения производительности, а уже затем определиться с основой для разработки прототипа.

### **Анализ имеющихся решений**

Наиболее крупными вендорами на рынке оптимизаторов WAN на данный момент являются такие компании, как Riverbed (SteelHead), Citrix (CloudBridge), HUAWEI (механизмы оптимизации встроены, например, в маршрутизаторы NetEngine AR600 [1], CloudWAN), FORTINET и Cisco Systems [2].

В первую очередь обратимся к документации от компании Riverbed (крупнейший вендор на рынке оптимизаторов WAN). Создатели RiOS (программное обеспечение, встроенное в оптимизаторы SteelHead) используют разные подходы к решению проблемы оптимизации WAN – среди реализованных механизмов есть как дедупликация данных, так и оптимизация некоторых протоколов (TCP, HTTP, FTP и др.), благодаря которой удается ускорить работу приложений, которые используют оптимизированные протоколы (например, Microsoft SQL Server становится более производительным из-за оптимизации протокола TDS) [3]. Разработчики заявляют, что с помощью RiOS можно устранить проблемы, связанные с полосой пропускания, задержками и «болтливостью» протоколов прикладного уровня. Более детально некоторые из методов оптимизации будут рассмотрены далее.

WANOS (WAN Optimization Software) – условно бесплатный оптимизатор WAN со встроенным веб-интерфейсом [4]. В базовую (бесплатную) версию встроены такие механизмы, как: компрессия, дедупликация и восстановление потерянных пакетов. Платные версии оптимизатора WANOS (SD-WAN и SD-WAN PLUS) кроме вышеперечисленных составляющих содержат в себе алгоритмы ускорения протокола TCP и некоторые улучшения для 7 (прикладного) уровня межкоммуникационной модели открытых систем (The Open Systems Interconnection model – OSI) (QoS, Firewall).

В свободном доступе имеется проект SoloWAN [5], разработанный испанскими учеными из Мадридского политехнического университета. SoloWAN является открытым решением, выполняющим функции оптимизатора WAN. В рамках данного проекта были реализованы такие методы, как дедупликация и сжатие данных, но при этом отсутствует какая-либо модификация TCP-протокола.

---

Проект WANOS не является проектом с открытым исходным кодом, поэтому использовать его в качестве основы для прототипа оптимизатора не представляется возможным, если заказчику требуются какие-то модификации (например, оптимизация отдельных протоколов прикладного уровня).

Команда исследователей-разработчиков проекта SoloWAN сравнили временные затраты при передаче одного и того же файла в трех различных сценариях: без какой-либо оптимизации, с использованием WANOS, с использованием SoloWAN (рис.1). Эксперимент показал, что передача файла с любым из оптимизаторов происходит быстрее, нежели чем без оптимизации. Примечательно, что первая передача файла с использованием SoloWAN требует больше времени, чем при использовании WANOS, при второй же передаче ситуация с точностью до наоборот – SoloWAN лучше справляется с передачей файла, что говорит об эффективности реализованного алгоритма дедупликации.

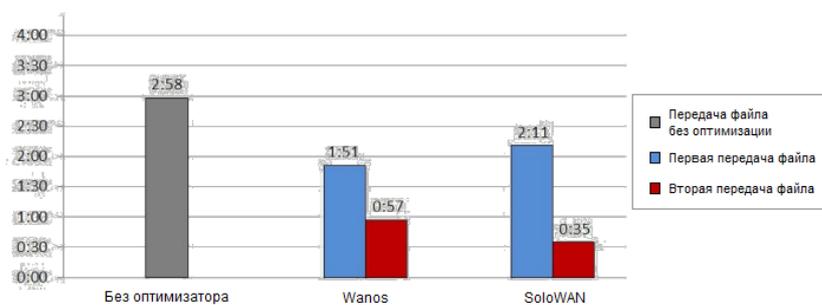


Рис. 1. – Сравнение временных затрат: без использования оптимизатора WAN, с использованием WANOS, с использованием SoloWAN [5]

Но действительно ли SoloWAN требует каких-либо модификаций? Обратимся к результатам эксперимента, представленным на рис.1 – при первой передаче файла WANOS оказался эффективнее SoloWAN именно из-

за наличия механизмов оптимизации протоколов (в частности, TCP, который наиболее важен при передаче файлов).

Ещё один серьёзный недостаток SoloWAN, из-за которого данный оптимизатор не подойдет в оригинальной комплектации – отсутствие модуля коррекции ошибок, из-за чего производительность неизбежно снижается, поскольку в любой глобальной сети имеются потери пакетов (в некоторых случаях значительные).

Проведем эксперимент для трёх разных конфигураций: без оптимизации, с применением SoloWAN, с применением SoloWAN и предварительной корректировкой ошибок (Forward Error Correction – FEC) скачаем тестовый набор файлов суммарным весом 46 Мбайт и замерим время, затраченное на данный процесс. Тестовый набор файлов для каждой из конфигураций будем скачивать 5 раз, увеличивая процент потерь пакетов с каждым повторением.

Результаты эксперимента представлены на рис.2.

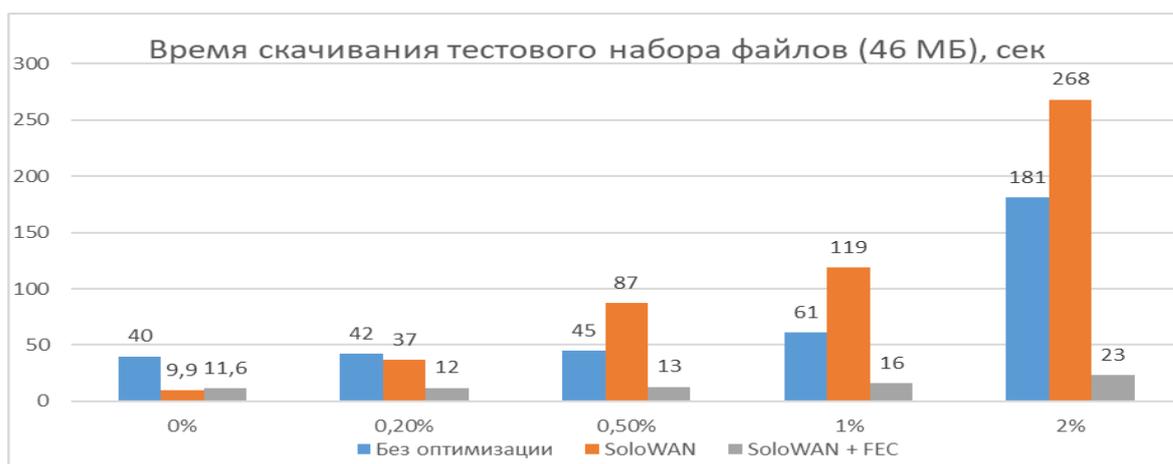


Рис. 2. – Передача файла в трёх различных конфигурациях

Как показывают замеры, SoloWAN без FEC довольно быстро теряет свою пользу – уже при 0.5% потерь пакетов производительность сети хуже

базовой конфигурации практически в два раза, когда как вариант SoloWAN + FEC показал себя куда более надежным вариантом во всех случаях, кроме ситуации, когда какие-либо потери пакетов отсутствуют (что для глобальных сетей является довольно редким событием).

В качестве модуля предварительной корректировки ошибок использовалась утилита `tinyfecVPN`, находящаяся в свободном доступе [6]. Разработка FEC с последующим встраиванием такого модуля в состав оптимизатора WAN – модификация, которую следует провести компаниям, которые особо обеспокоены надежностью решений, размещенных в свободном доступе, при этом разработка собственного модуля FEC будет куда менее трудоемкой задачей, нежели чем создание собственного полноценного оптимизатора WAN.

### **Механизмы оптимизации**

Далее будут рассмотрены реализации различных методов оптимизации WAN: дедупликация, TCP-оптимизация, оптимизация приложений, и тем самым определим, насколько необходима та или иная модификация для прототипа оптимизатора WAN.

#### *Дедупликация данных*

Рассмотрим дедупликацию данных на примере RiOS [3]. Оптимизатор WAN в топологии сети находится на границе локальной и глобальной сети, через него проходит весь трафик (что входящий, что исходящий). Оптимизатор WAN хранит внутри себя «словарь» с парами типа «ссылка-данные». На другом конце канала глобальной сети должен быть установлен такой же оптимизатор WAN.

Сперва TCP-трафик перехватывается и анализируется, а данные сегментируются (средний вес сегмента – 100 байт) и индексируются. Индексированные данные сравниваются с данными, которые уже находятся

---

на диске или в памяти, тем самым определяются сегменты данных, ранее занесенные в «словарь» оптимизатора. Дублирующиеся сегменты данных заменяются на меньшую по весу ссылку, чтобы не нагружать сеть WAN передачей однажды уже переданных сегментов. Если же сегменты данных никогда ранее не передавались, то происходит сжатие (в RiOS сжатие данных реализовано на основе алгоритма Лемпеля-Зива (Lempel-Ziv – LZ) [7], после чего происходит передача сегментов вместе с ссылками по WAN-каналу.

Принимающий оптимизатор WAN после восстановления данных выполняет соответствия «ключ – сегмент» с удалением ключей из данных (при этом в «словаре» строятся новые соответствия), а уже затем файл передается клиенту.

Если, например, пользователь отправляет некоторому сотруднику из другого филиала отредактированный файл (ранее переданный по глобальной сети), то по сети WAN будут переданы лишь изменения.

Компания Riverbed отчитывается, что с помощью оптимизации передачи данных (сжатия и дедупликации) использование полосы пропускания WAN сокращается на 60-95%, а объем передаваемых данных может быть сокращен в 100 и более раз. Стоит учесть, что эффективность оптимизации передачи данных зависит как от используемых алгоритмов сжатия (можно выбрать иной алгоритм сжатия, даже из того же семейства LZ), так и от коэффициента сжатия.

### ***ТСР-оптимизация***

Как было сказано ранее, протокол ТСР не приспособлен для использования в глобальных сетях, поскольку подтверждение передачи данных работает только в том случае, когда в сети задержки достаточно малы и отсутствуют потери пакетов данных. При потерях пакетов данных скорость передачи значительно снижается, поскольку существует необходимость перепроверки целостности переданной информации.

Кроме того, следует выделить такие недостатки протокола, как неэффективное использование пропускной полосы, механизмы медленного старта (Slow Start) и избегания перегрузок (Congestion Avoidance), так как они ограничивают скорость передачи данных, что в контексте применения TCP в WAN лишь усугубляет положение.

Таким образом, оптимизировать протокол TCP можно несколькими способами (отметим, что методы можно совмещать):

1. Сокращение времени старта;
2. Увеличение полосы пропускания;
3. Сокращение снижения скорости передачи данных при возникновении потерь;
4. Изменение характера набора скорости после потери данных (нелинейное увеличение вместо линейного).

Перечисленные выше шаги реализованы в оптимизаторе Cisco WAAS [9].

Вновь обратимся к документации RiOS [3]. Вендор Riverbed использует в своих решениях две модификации протокола TCP: MX-TCP (Max Speed TCP) и HS-TCP (High-Speed TCP).

На рис.3 представлено сравнение протоколов TCP, MX-TCP и HS-TCP.

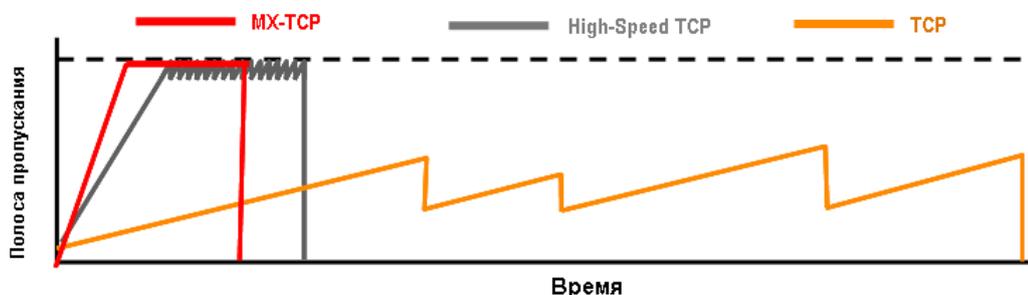


Рис. 3. Производительность различных реализаций TCP

МХ-ТСР игнорирует congestion avoidance (что может привести к перегрузке канала), а размер ТСР-окна (объем данных, отправленных в одном пакете) администратор должен задавать самостоятельно, но при пропускная полоса канала используется на максимум, при потерях пакетов пропускная способность не уменьшается, slow start сокращен по длительности.

HS-ТСР использует пропускную полосу практически на максимум (при этом размер окна определяется автоматически), скорость передачи данных может уменьшаться при потерях пакетов, но в меньшей мере, чем у стандартного ТСР, однако медленный старт занимает больше времени, чем у МХ-ТСР.

Майк Фримон, сотрудник компании Cloudflare (предоставляет услуги по защите и ускорению работы веб-ресурсов), в своей статье описывает одну из модификаций протокола ТСР, обеспечивающую высокую пропускную способность глобальной сети с сохранением низких задержек [9].

На рис.4 представлены результаты измерений пропускной способности ТСР при передаче данных по глобальной сети: из США во Францию и из Австралии во Францию. Как можно заметить из сравнения, пропускная способность улучшенного варианта ТСР намного больше пропускной способности классической реализации ТСР-протокола.

Подобных результатов Cloudflare получилось добиться посредством увеличения начального окна перегрузки и буферов, замены алгоритма управления перегрузками (был взят алгоритм BBR, разработанный компанией Google [10]), оптимизации повторных передач, динамического масштабирования приемного окна и некоторых других настроек (оптимизация таймеров, управление памятью).

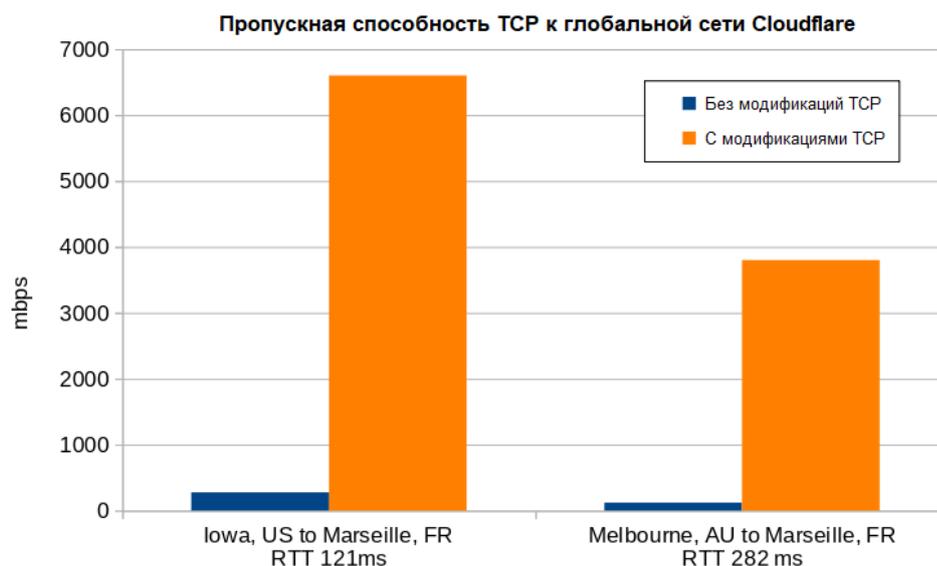


Рис.4. – Пропускная способность TCP в глобальной сети Cloudflare [7]

### *Оптимизация приложений и другие методы*

Иные способы оптимизации и необходимость их использования целиком зависит от требований и пожеланий компании – от того, какими приложениями пользуются сотрудники во время рабочего процесса, зависит то, какие протоколы (используемые в приложениях) необходимо оптимизировать. Как правило, разработчики стараются сделать используемые приложения менее «болтливыми» (из-за наличия механизма подтверждения отправки-получения, например), чтобы снизить нагрузку на канал связи.

### **Архитектура оптимизатора**

Прототип оптимизатора разрабатывается с применением языков программирования C/C++ на основе проекта SoloWAN. Прототип является модификацией OpenNOP (Open Network Optimization Platform), внутри которой реализованы компрессия данных, перехват трафика и оптимизация TCP, а также поддерживается многопоточное выполнение задач.

Оптимизатор использует механизмы, реализованные в OpenNOP, а также содержит в себе алгоритм дедупликации данных. Прототип способен представлять информацию по пропускной способности, задержкам и иным показателям в качестве графиков в веб-интерфейсе.

Прототип разработан как программа-демон (daemon), которая запускается только на дистрибутивах Linux.

Для исследования эффективности оптимизатора и определения его дальнейших модификаций используется стенд виртуальных машин, созданный в среде Oracle VirtualBox. Используемые дистрибутивы – РЕД ОС, Debian, Arch Linux.

В качестве одного из сценариев для тестирования функциональности и эффективности оптимизатора был выбран случай со скачиванием клиентом файлов (pdf, txt, xls и другие) с общедоступного сервера (сервис SAMBA), при этом клиентский ПК и сервер находятся в разных «филиалах» и разделены виртуальной машиной, эмулирующей сегмент глобальной сети. Эмуляция WAN достигается с помощью модуля netem (команда tc), который позволяет создать задержки и потери пакетов, подобные тем, что встречаются в глобальных сетях. Оптимизатор успешно справляется с поставленными задачами, уменьшая время, затраченное на передачу файлов через канал глобальной сети. Но при этом стоит отметить, что оптимизатор без модуля FEC теряет эффективность даже при низком проценте потерь пакетов.

### **Заключение**

В данной работе были проанализированы различные решения в области оптимизации глобальных сетей и была выбрана основа для создания прототипа отечественного оптимизатора WAN, а именно проект SoloWAN, размещенный в свободном доступе в сети Интернет.

Кроме того, в работе были выявлены недостатки такого решения и были предложены способы улучшения прототипа.

Особенностью выбранного решения является то, что такой прототип может быть приспособлен под нужды конкретной компании путем внесения необходимых модификаций (оптимизация протоколов, замена алгоритмов сжатия данных и др.).

### Литература

1. NetEngine AR600, AR6100, AR6200, and AR6300 V300R019 CLI-based Configuration Guide - WAN Optimization. URL: [support.huawei.com/enterprise/en/doc/EDOC1100189918/c51882e4/wan-optimization-configuration](https://support.huawei.com/enterprise/en/doc/EDOC1100189918/c51882e4/wan-optimization-configuration).

2. Zhang Y. Ansari N., Wu M., Yu H. On wide area network optimization //IEEE Communications surveys & tutorials. 2011. V.14. №4. pp. 1090-1113.

3. Ali M.M.N. Optimizing Local Area Network efficiency using Riverbed Academic Edition 17.5 // GUB Journal of Science and Engineering. 2022. V. 9. №1. pp. 71-81.

4. WAN optimization (WAN acceleration). URL: [techtarget.com/searchnetworking/definition/WAN-optimization-WAN-acceleration](https://techtarget.com/searchnetworking/definition/WAN-optimization-WAN-acceleration)

5. SoloWAN Project. URL: [github.com/solowan/solowan?tab=readme-ov-file](https://github.com/solowan/solowan?tab=readme-ov-file).

6. tinyfecVPN. URL: [github.com/wangyu-tinyfecVPN](https://github.com/wangyu-tinyfecVPN).

7. Кудинов В. В., Сапрыкин О. К. Сравнение эффективности алгоритмов сжатия LZ77 и LZ78 // Вопросы электромеханики. Труды ВНИИЭМ Т. 2024. Т. 201. №4. С. 37-41.

8. Volpe F. MTRs Based on Windows and Surface Hub // Microsoft 365 Certified: Collaboration Communications Systems Engineer Associate Certification Companion: A Guide to Prepare for Exam MS-721. Berkeley: Apress, 2024. pp. 257-276.

9. Schwarz M., Tierney B., Vasu K., Dart E. Recent Linux Improvements that Impact TCP Throughput: Insights from R&E Networks // SC24-W: Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis. IEEE, 2024. pp. 775-784.

10. Han Y., Zuo M., Yuan H., Zhong Y., Yuan Zh., Bi T. A QoS-Based Fairness-Aware BBR Congestion Control Algorithm Using QUIC // Wireless Communications and Mobile Computing. 2022. V.2022(1). URL: [onlinelibrary.wiley.com/doi/10.1155/2022/7222030](https://onlinelibrary.wiley.com/doi/10.1155/2022/7222030).

### References

1. NetEngine AR600, AR6100, AR6200, and AR6300 V300R019 CLI-based Configuration Guide - WAN Optimization. URL: [support.huawei.com/enterprise/en/doc/EDOC1100189918/c51882e4/wan-optimization-configuration](https://support.huawei.com/enterprise/en/doc/EDOC1100189918/c51882e4/wan-optimization-configuration).

2. Zhang Y. Ansari N., Wu M., Yu H. IEEE Communications surveys & tutorials. 2011. V.14. №4. pp. 1090-1113.

3. Ali M.M.N. GUB Journal of Science and Engineering. 2022. V.9. №1. pp. 71-81.

4. WAN optimization (WAN acceleration). URL: [techtarget.com/searchnetworking/definition/WAN-optimization-WAN-acceleration](https://techtarget.com/searchnetworking/definition/WAN-optimization-WAN-acceleration)

5. SoloWAN Project. URL: [github.com/solowan/solowan?tab=readme-ov-file](https://github.com/solowan/solowan?tab=readme-ov-file).

6. tinyfecVPN. URL: [github.com/wangyu/tinyfecVPN](https://github.com/wangyu/tinyfecVPN).

7. Kudinov V. V., Saprykin O. K. Voprosy elektromekhaniki. Trudy VNIEMT. 2024. V. 201. №4. С. 37-41.

8. Volpe F. Microsoft 365 Certified: Collaboration Communications Systems Engineer Associate Certification Companion: A Guide to Prepare for Exam MS-721. Berkeley: Apress, 2024. pp. 257-276.



9. Schwarz M., Tierney B., Vasu K., Dart E. SC24-W: Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis. IEEE, 2024. pp. 775-784.

10. Han Y., Zuo M., Yuan H., Zhong Y., Yuan Zh., Bi T. A Wireless Communications and Mobile Computing. 2022. V.2022(1). URL: [onlinelibrary.wiley.com/doi/10.1155/2022/7222030](https://onlinelibrary.wiley.com/doi/10.1155/2022/7222030).

**Дата поступления: 11.04.25**

**Дата публикации: 25.05.25**