**Ministerul Educației, Culturii și Cercetării**

**Universitatea Liberă Internațională din Moldova**

**Facultatea Informatică, Inginerie și Design**

**Disciplina: Sisteme Operare**

**Thema: Network layer Routing algorithms and congestion control algorithms**

**A efectuat:**   
**Урсуленко Никита**

**Grupa: TIR-36**

**A verificat: Semnătura profesorului:**  
**Морарь Виктор**

**Data prezentării:\_\_\_\_\_\_\_ Nota:\_\_\_\_\_\_\_\_**

**Chișinău, 2023**

СОДЕРЖАНИЕ

Структура реферата.................................................................................. 3

1. Введение..............................................................................................5

2. Сетевой уровень .................................................................................6

3. Алгоритмы маршрутизации.....................................................................9

4. Алгоритмы управления перегрузкой......................................................12

5. Важность алгоритмов маршрутизации и управления перегрузками................................................................................................ 14

ЗАКЛЮЧЕНИЕ........................................................................................... 17

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .................................

Приложение A ............................................................................................. 36

**I. Введение**

* **Краткий обзор сетевого уровня**
* **Важность алгоритмов маршрутизации и алгоритмов управления перегрузкой**

**II. Сетевой уровень**

* **Определение и функции сетевого уровня**
* **Модель OSI и сетевой уровень**
* **IP-адресация и фрагментация пакетов**

**III. Алгоритмы маршрутизации**

* **Определение и функции алгоритмов маршрутизации**
* **Типы алгоритмов маршрутизации (например, вектор расстояния, состояние канала, вектор пути)**
* **Сравнение алгоритмов маршрутизации**

**IV. Алгоритмы управления перегрузкой**

* **Определение и функции алгоритмов управления перегрузкой**
* **Типы алгоритмов управления перегрузкой (например, TCP, UDP)**
* **Сравнение алгоритмов управления перегрузкой**

**V. Важность алгоритмов маршрутизации и управления перегрузками**

* **Необходимость в эффективных алгоритмах маршрутизации и алгоритмах управления перегрузкой**
* **Влияние неэффективных алгоритмов маршрутизации и алгоритмов управления перегрузкой**
* **Примеры успешных алгоритмов маршрутизации и управления перегрузками**

**VI. Заключение**

* **Повторение важности сетевого уровня, алгоритмов маршрутизации и алгоритмов управления перегрузкой.**
* **Будущие разработки и усовершенствования алгоритмов маршрутизации и управления перегрузками**

**I. Введение**

Сетевой уровень — это один из семи уровней модели OSI, представляющей собой концептуальную модель, описывающую коммуникационные функции телекоммуникационной или вычислительной системы. Сетевой уровень отвечает за логическую адресацию сетевых устройств и доставку пакетов данных по нескольким сетям.

Алгоритмы маршрутизации играют важнейшую роль на сетевом уровне, определяя оптимальный путь для пакетов данных, которые должны пройти от источника к месту назначения. Алгоритмы маршрутизации учитывают такие факторы, как топология сети, нагрузка трафика и перегрузка сети, чтобы определить наилучший путь для пакетов данных.

Алгоритмы управления перегрузкой также являются важной частью сетевого уровня, поскольку они помогают предотвратить перегрузку сети, регулируя скорость, с которой данные передаются по сети. Алгоритмы управления перегрузкой направлены на обеспечение эффективного использования ресурсов сети при минимизации риска потери данных и задержек.

Сетевой уровень, алгоритмы маршрутизации и алгоритмы управления перегрузкой — все это важные концепции компьютерных сетей. В этом эссе мы подробно рассмотрим эти концепции, изучив их роль в сетевой коммуникации, их различные типы и то, как они используются для оптимизации производительности сети. Кроме того, мы обсудим важность эффективных алгоритмов маршрутизации и алгоритмов управления перегрузкой для обеспечения бесперебойной работы компьютерных сетей, а также то, как усовершенствование этих алгоритмов может привести к повышению производительности и надежности сети.

**II. Сетевой уровень**

Сетевой уровень является важнейшим компонентом набора интернет-протоколов и отвечает за передачу пакетов данных между хостами в разных сетях. В модели OSI сетевой уровень является третьим уровнем и находится между уровнем канала передачи данных и транспортным уровнем.

Основной функцией сетевого уровня является предоставление услуг логической адресации и маршрутизации, чтобы гарантировать, что пакеты данных будут доставлены в правильное место назначения. Это достигается за счет использования IP-адресации, которая однозначно идентифицирует каждое устройство в сети.

Сетевой уровень также отвечает за фрагментацию пакетов, т. е. процесс разбиения больших пакетов на более мелкие фрагменты для обеспечения возможности их передачи по сетям с различными максимальными размерами передаваемых единиц (MTU). Затем хост-получатель снова собирает фрагменты в исходный пакет.

Одним из ключевых протоколов, используемых на сетевом уровне, является Интернет-протокол (IP), который является основным протоколом Интернета. IP — это протокол без установления соединения, что означает, что пакеты отправляются независимо друг от друга и могут идти разными путями, чтобы достичь места назначения.

Другим важным протоколом, используемым на сетевом уровне, является протокол управляющих сообщений Интернета (ICMP), который используется для отчетов об ошибках, устранения неполадок и проверки связи. Сообщения ICMP инкапсулируются в IP-пакеты и используются для указания на ошибки или для запроса информации о состоянии сети.

Одной из основных задач сетевого уровня является обеспечение эффективной маршрутизации пакетов между хостами в разных сетях. Это достигается за счет использования алгоритмов маршрутизации, которые определяют наилучший путь для прохождения пакетов по сети.

Алгоритмы маршрутизации можно разделить на три основных типа: алгоритмы вектора расстояния, алгоритмы состояния канала и вектора пути. Алгоритмы вектора расстояния используют простой итеративный подход для поиска кратчайшего пути к узлу назначения, в то время как алгоритмы состояния канала используют более сложный алгоритм, учитывающий всю топологию сети. Алгоритмы вектора пути представляют собой гибрид алгоритмов вектора расстояния и алгоритма состояния канала и используются в крупномасштабных сетях.

Другой серьезной проблемой, с которой сталкивается сетевой уровень, является контроль перегрузки, который относится к процессу регулирования потока пакетов данных для предотвращения перегрузки сети. Перегрузка может возникнуть, когда сеть становится перегруженной из-за высокого уровня трафика, что может привести к низкой скорости передачи, потере пакетов и снижению общей производительности сети.

Для решения этой проблемы были разработаны различные алгоритмы управления перегрузкой, включая протокол управления передачей (TCP) и протокол пользовательских дейтаграмм (UDP). TCP — это надежный протокол, использующий механизм скользящего окна для регулирования потока пакетов и обеспечения надежной и упорядоченной передачи пакетов. UDP, с другой стороны, является протоколом без установления соединения, который не обеспечивает надежность или управление потоком, что делает его подходящим для приложений, требующих передачи с малой задержкой, таких как онлайн-игры или потоковая передача.

В заключение, сетевой уровень играет критическую роль в передаче пакетов данных между хостами в разных сетях. Его основные функции включают логическую адресацию, маршрутизацию и фрагментацию пакетов. Чтобы обеспечить эффективную передачу данных, сетевой уровень использует алгоритмы маршрутизации для определения наилучшего пути для пакетов и алгоритмы управления перегрузкой для регулирования потока данных и предотвращения перегрузки сети.

**III. Алгоритмы маршрутизации**

Алгоритмы определяют путь, по которому пакеты данных проходят от исходного устройства к целевому устройству через сеть. Алгоритмы маршрутизации отвечают за поиск наиболее эффективного пути для передачи пакетов данных и обеспечение того, чтобы они быстро и надежно достигали места назначения.

Существует несколько типов алгоритмов маршрутизации, которые обычно используются в компьютерных сетях, включая маршрутизацию на основе вектора расстояния, маршрутизацию на основе состояния канала и маршрутизацию на основе вектора пути. Каждый из этих алгоритмов имеет свои преимущества и недостатки, и сетевые администраторы должны выбрать лучший алгоритм маршрутизации для своей конкретной сетевой архитектуры и требований.

Маршрутизация на векторе расстояния — это простой алгоритм маршрутизации, который вычисляет кратчайший путь между узлами на основе количества переходов или количества сетевых устройств, через которые должен пройти пакет, чтобы достичь пункта назначения. Этот алгоритм прост в реализации и требует мало памяти и вычислительной мощности. Однако он не всегда может найти наиболее эффективный путь и может быть уязвим для петель маршрутизации.

С другой стороны, маршрутизация на основе состояния канала представляет собой более сложный алгоритм, который вычисляет кратчайший путь между узлами на основе фактического расстояния или времени, необходимого пакету для перемещения между узлами. Этот алгоритм требует больше памяти и вычислительной мощности, чем дистанционно-векторная маршрутизация, но он более точен и эффективен. Маршрутизация на основе состояния канала также обеспечивает более подробное представление топологии сети и может быстрее адаптироваться к изменениям в сети.

Маршрутизация по вектору пути — это гибридный алгоритм, который сочетает в себе преимущества как маршрутизации по вектору расстояния, так и маршрутизации по состоянию канала. Этот алгоритм вычисляет кратчайший путь между узлами на основе как количества переходов, так и фактического расстояния. Это более сложный алгоритм, чем маршрутизация по дистанционному вектору, но менее сложный, чем маршрутизация по состоянию канала.

В дополнение к этим алгоритмам маршрутизации существует также несколько протоколов, которые обычно используются в компьютерных сетях для реализации маршрутизации, такие как протокол пограничного шлюза (BGP), протокол открытия кратчайшего пути (OSPF) и протокол информации о маршрутизации (RIP). Эти протоколы используют разные алгоритмы маршрутизации для определения наилучшего пути для пакетов данных и обеспечения эффективной и надежной доставки пакетов.

Алгоритмы маршрутизации играют решающую роль в производительности и надежности компьютерных сетей. Выбрав наиболее эффективный алгоритм маршрутизации и внедрив соответствующие протоколы маршрутизации, сетевые администраторы могут обеспечить быструю и надежную доставку пакетов данных между устройствами в сети. Поскольку компьютерные сети продолжают расти в размерах и сложности, важность алгоритмов маршрутизации для обеспечения оптимальной производительности сети будет только расти. Сетевой уровень является важнейшим компонентом набора интернет-протоколов и отвечает за передачу пакетов данных между хостами. в разных сетях. В модели OSI сетевой уровень является третьим уровнем и находится между уровнем канала передачи данных и транспортным уровнем.

Основной функцией сетевого уровня является предоставление услуг логической адресации и маршрутизации, чтобы гарантировать, что пакеты данных будут доставлены в правильное место назначения. Это достигается за счет использования IP-адресации, которая однозначно идентифицирует каждое устройство в сети.

Сетевой уровень также отвечает за фрагментацию пакетов, т. е. процесс разбиения больших пакетов на более мелкие фрагменты для обеспечения возможности их передачи по сетям с различными максимальными размерами передаваемых единиц (MTU). Затем хост-получатель снова собирает фрагменты в исходный пакет.

Одним из ключевых протоколов, используемых на сетевом уровне, является Интернет-протокол (IP), который является основным протоколом Интернета. IP — это протокол без установления соединения, что означает, что пакеты отправляются независимо друг от друга и могут идти разными путями, чтобы достичь места назначения.

Другим важным протоколом, используемым на сетевом уровне, является протокол управляющих сообщений Интернета (ICMP), который используется для отчетов об ошибках, устранения неполадок и проверки связи. Сообщения ICMP инкапсулируются в IP-пакеты и используются для указания на ошибки или для запроса информации о состоянии сети.

Одной из основных задач сетевого уровня является обеспечение эффективной маршрутизации пакетов между хостами в разных сетях. Это достигается за счет использования алгоритмов маршрутизации, которые определяют наилучший путь для прохождения пакетов по сети.

Алгоритмы маршрутизации можно разделить на три основных типа: алгоритмы вектора расстояния, алгоритмы состояния канала и вектора пути. Алгоритмы вектора расстояния используют простой итеративный подход для поиска кратчайшего пути к узлу назначения, в то время как алгоритмы состояния канала используют более сложный алгоритм, учитывающий всю топологию сети. Алгоритмы вектора пути представляют собой гибрид алгоритма вектора расстояния и алгоритма состояния канала.

**IV. Алгоритмы управления перегрузкой**

Алгоритмы управления перегрузкой являются важнейшим компонентом сетевого уровня, помогающим предотвратить перегрузку сети и обеспечить надежную передачу данных. Перегрузка сети происходит, когда в сети больше трафика, чем она может обработать, что приводит к замедлению или нарушению передачи данных. Алгоритмы управления перегрузкой регулируют объем данных, отправляемых по сети, помогая предотвратить перегрузку и обеспечить надежную передачу данных.

Существует несколько типов алгоритмов управления перегрузкой, включая протокол управления передачей (TCP) и протокол пользовательских дейтаграмм (UDP). TCP является наиболее часто используемым алгоритмом управления перегрузкой и используется для большинства интернет-протоколов связи. TCP работает с использованием алгоритма скользящего окна, при котором данные отправляются в виде потока пакетов, а получатель отправляет подтверждения обратно отправителю для каждого полученного пакета. Размер окна динамически регулируется в зависимости от состояния сети и количества полученных подтверждений, помогая регулировать поток данных по сети.

UDP, с другой стороны, не имеет встроенных механизмов управления перегрузкой и обычно используется для приложений реального времени, таких как потоковое видео или аудио. Хотя UDP не так надежен, как TCP, он часто предпочтительнее для приложений такого типа, поскольку обеспечивает более быструю передачу данных с меньшими затратами.

В дополнение к TCP и UDP существуют и другие алгоритмы управления перегрузкой, такие как явное уведомление о перегрузке (ECN) и случайное раннее обнаружение (RED). ECN — это механизм управления перегрузкой, который работает, помечая пакеты, испытывающие перегрузку, позволяя маршрутизаторам приоритизировать трафик и избегать перегрузки. RED, с другой стороны, работает путем случайного отбрасывания пакетов до того, как сеть станет перегруженной, помогая регулировать объем трафика в сети.

Выбор алгоритма управления перегрузкой зависит от конкретного приложения и сетевых требований. Например, приложениям реального времени может потребоваться более быстрая передача данных с меньшими затратами, в то время как для более традиционных приложений передачи данных может потребоваться более надежный и надежный механизм управления перегрузкой, такой как TCP.

В целом, алгоритмы управления перегрузкой являются важным компонентом сетевого уровня, помогающим обеспечить надежную и эффективную передачу данных по сетям. Выбор алгоритма управления перегрузкой зависит от конкретного приложения и сетевых требований, и существует несколько различных алгоритмов, каждый из которых имеет свои сильные и слабые стороны.

**V. Важность алгоритмов маршрутизации и управления перегрузками**

Сетевой уровень является важной частью сетевой инфраструктуры, которая обеспечивает связь между различными устройствами в сети. Алгоритмы маршрутизации и алгоритмы управления перегрузкой являются важными компонентами сетевого уровня, которые играют жизненно важную роль в обеспечении эффективной и надежной передачи данных. В этом разделе мы подробно обсудим важность алгоритмов маршрутизации и управления перегрузкой.

Одной из основных функций сетевого уровня является маршрутизация пакетов между различными устройствами в сети. Алгоритмы маршрутизации определяют наилучший путь для пакетов от исходного устройства к целевому. Эффективность и действенность алгоритма маршрутизации напрямую влияют на производительность сети. Плохо разработанный алгоритм маршрутизации может привести к задержкам, потере пакетов и перегрузке сети, что может существенно повлиять на пропускную способность сети.

Существуют различные типы алгоритмов маршрутизации, такие как вектор расстояния, состояние канала и вектор пути. Каждый из этих алгоритмов маршрутизации имеет свои сильные и слабые стороны, и выбор алгоритма маршрутизации зависит от конкретных потребностей сети. Например, алгоритмы дистанционно-векторной маршрутизации просты и легки в реализации, но могут не подходить для больших сетей. С другой стороны, алгоритмы маршрутизации на основе состояния канала более сложны, но могут более эффективно работать с большими сетями.

Алгоритмы управления перегрузкой — еще один важный компонент сетевого уровня. Перегрузка возникает, когда сеть не может справиться с объемом передаваемого трафика, что приводит к возникновению узкого места. Алгоритмы управления перегрузкой предназначены для предотвращения или уменьшения возникновения перегрузки путем управления скоростью, с которой данные передаются по сети.

Существуют различные типы алгоритмов управления перегрузкой, такие как TCP и UDP. TCP (протокол управления передачей) — это надежный транспортный протокол, использующий контроль перегрузки для предотвращения перегрузки сети. TCP использует алгоритм скользящего окна, который динамически регулирует скорость передачи в зависимости от состояния сети. UDP (протокол пользовательских дейтаграмм), с другой стороны, является протоколом без установления соединения, который не обеспечивает контроль перегрузки. UDP часто используется для приложений реального времени, где скорость важнее надежности.

Нельзя переоценить важность эффективных алгоритмов маршрутизации и управления перегрузкой. Неэффективные алгоритмы маршрутизации и управления перегрузкой могут привести к серьезным проблемам с производительностью, включая задержки, потерю пакетов и перегрузку сети. Эти проблемы могут негативно повлиять на работу пользователей, затрудняя доступ к ресурсам в сети.

Эффективные алгоритмы маршрутизации и управления перегрузкой могут повысить пропускную способность сети, уменьшить задержки и потери пакетов, а также предотвратить перегрузку сети. Эти алгоритмы имеют решающее значение для обеспечения того, чтобы сеть могла обрабатывать растущий объем передаваемого трафика. Поскольку количество устройств в сети увеличивается, а спрос на данные продолжает расти, потребность в эффективных алгоритмах маршрутизации и управления перегрузкой будет только расти.

В заключение следует отметить, что алгоритмы маршрутизации и алгоритмы управления перегрузкой являются важными компонентами сетевого уровня, которые играют жизненно важную роль в обеспечении эффективной и надежной передачи данных. Эффективные алгоритмы маршрутизации и управления перегрузкой могут повысить пропускную способность сети, уменьшить задержки и потери пакетов, а также предотвратить перегрузку сети. Таким образом, важно выбрать соответствующие алгоритмы маршрутизации и управления перегрузкой, основанные на конкретных потребностях сети, чтобы обеспечить оптимальную производительность.

**VI. Заключение**

В заключение, сетевой уровень является важнейшим компонентом компьютерных сетей, который обеспечивает связь между устройствами в сети. Алгоритмы маршрутизации и алгоритмы управления перегрузкой являются двумя важными механизмами, которые работают на сетевом уровне для обеспечения бесперебойной передачи пакетов данных.

Алгоритмы маршрутизации определяют оптимальный путь для доставки пакетов данных к месту назначения с учетом таких факторов, как расстояние, перегрузка и топология сети. Существуют различные типы алгоритмов маршрутизации, каждый из которых имеет свои преимущества и недостатки. Например, дистанционно-векторная маршрутизация проста и требует меньше вычислительных ресурсов, но подвержена петлям маршрутизации и медленной сходимости. С другой стороны, маршрутизация на основе состояния канала предоставляет более точную информацию о сети, но требует большей вычислительной мощности и памяти.

Алгоритмы управления перегрузкой, с другой стороны, предназначены для предотвращения перегрузки сети путем регулирования скорости, с которой передаются пакеты данных. Перегрузка может возникнуть, когда сеть перегружена большим объемом трафика, чем она может обработать, что приводит к отбрасыванию или задержке пакетов. TCP является наиболее широко используемым протоколом управления перегрузкой, который гарантирует надежную и упорядоченную доставку пакетов, адаптируясь к перегрузке сети.

Эффективные алгоритмы маршрутизации и управления перегрузкой необходимы для производительности и стабильности компьютерных сетей. Плохо разработанные алгоритмы могут привести к перегрузке сети, потере пакетов и задержкам, что приведет к ухудшению работы пользователей и снижению производительности. Поскольку сети продолжают расти в размерах и усложняться, становится все более важным разрабатывать и улучшать эти алгоритмы, чтобы не отставать от спроса на высокоскоростную передачу данных.

В будущем достижения в области сетевых технологий, таких как программно определяемые сети (SDN) и виртуализация сетевых функций (NFV), откроют новые возможности для оптимизации и автоматизации сети. Алгоритмы маршрутизации и управления перегрузкой будут продолжать играть решающую роль в производительности и надежности этих новых технологий. Поэтому важно продолжать исследования и разработку новых алгоритмов, чтобы не отставать от меняющихся требований компьютерных сетей.

<https://ru.wikipedia.org/wiki/Протоколы_сетевого_уровня>

<https://neerc.ifmo.ru/wiki/index.php?title=Алгоритмы_маршрутизации>

<https://opengl.org.ru/upravlenie-trafikom-i-kachestvo-obsluzhevaniya-v-seti/algoritm-upravleniya-peregruzkoi.html>

https://intuit.ru/studies/courses/2/2/lecture/48