1. Последовательная и параллельная сложность алгоритмов, информационный граф и ресурс параллелизма алгоритмов.

Последовательная и параллельная сложность, граф алгоритма, ярусно-параллельная форма, высота и ширина ярусно-параллельной формы, каноническая ярусно-параллельная форма.

Конечный, массовый, координатный, скошенный параллелизм.

2. Архитектурные особенности графических процессоров, направленные на массивно-параллельные вычисления. Методы эффективной организации параллельных вычислений на графических процессорах.

Полный курс по CUDA: <https://drive.google.com/drive/folders/0B3XUBeyj27tALThMUE9tSVhzWFk>

А также см. прилагающиеся лекции.

GPGPU – general purpose GPU

Tesla -> ***Fermi*** -> Kepler -> Maxwell

Hyper-Threading

SM – мультипроцессор, текстурный, глобальный, константный кеш

SIMT – Single instruction multiple threads

Нить, варп, блок, грид,

Поток выполнения, асинхронность, параллельность

Паддинг, банк памяти

Обработка ошибок, компиляция

UVA – unified virtual addressing

3. Основные принципы организации оптических и беспроводных систем передачи данных.

Спектр затухания сигнала в оптоволокне – минимум на 1600 нм

Оптика: 180-370 THz, 40 km,

Манчестерский код NRZ код и другие

PDH – 64 kbps – 10 гигабит (E0, E1, E2)

T1, T2, T3 (4 – 6 – 7) – 1,544 Mbps

SONET – STS 1/3/12/48 – 51,84 Mbps

SDH – STM 1/3/4/6/8 – SPE-конверт (810 байт – 125 мкс) – 155,52 Mbps

OTN (G.709) – ODU 0/1/2/3/4 (1,25 – 2,5 – 10 – 40 – 100 Gbps)

OTU: FAS – первые 6 байт для определения границ кадра, payload, FEC – 4x4080 bytes

ODU: source, destination, path monitoring, … … OPU payload (4x3808 bytes)

OPU: multiplex structure identifier

DWDM (x18, x40, x80, x126)

частотное мультиплексирование (WDM – wave division multiplexing x18)

*Модуляции*: Частотно-фазовая или частотно-амплитудная модуляция, поляризация

SONET/SDH (810 байт – 125 мкс), SPE-конверт

Секция – линия - путь

Достоинства OTN: FER - forward error correction, несколько уровней сквозного мониторинга пути - Tandem Connection Monitoring (TCM) (смена пути при бите ошибке в обратную сторону), прозрачная передача данных пользователей, Switching Scalability

Дефрагментация загрузки линий

OTN – достоинства

1. OTN обеспечивает предсказуемое и простое предоставление услуг.

2. Эффективная интеграция коммутации OTN с существующей транспортной сетью OTN.

3. Упрощение операций.

4. Измерение задержки в режиме реального времени.

5. Виртуализация сети.

6. Базовая сеть без потерь.

7. Несколько классов обслуживания.

8. Усовершенствованные возможности сквозного мониторинга услуг.

9. Четкий путь развития до 100G и далее.

10. Динамическая инфраструктура.

ALOHA

802.11b – 5

802.11g – 2,4

4. Сети хранения данных – архитектура и основные сервисы.

Storage Area Network

Server Centric Architecture, Storage Centric Architecture

SCSI – 25 метров (16 разных портов) (макс. 320 Mbps)

Архитектура дисковой подсистемы: сервер – канал – controller с кешем – диски

Много дисков для – ускорения и RAID

Диски подключены к контроллеру одним каналом или несколькими (нагрузка между каналами распределяется или нет)

Контроллер может виртуализовывать диски

RAID – Redundant Array of Independent Disks

RAID0 – поочерёдная запись в диски по Round-Robin

RAID1 – полное дублирование

RAID0+1 или RAID1+0

RAID4 – контрольная сумма на отдельном диске

RAID5 – контрольная сумма раскидана на разные диски по Round-Robin

RAID6 – RAID5+1

Fault tolerance vs Read performance vs write performance vs space requirement

Кеши (где установлен, устойчивость к отключению питания при кеше на запись)

ИАД – интеллектуальная дисковая подсистема

Синхронное и асинхонное зеркалирование «на ходу»

LUN-маскировка – позволяет разделять доступ серверов к различным дискам системы

FiberChannel – стек протоколов

FC-0 определяет физические характеристики интерфейса и среды, включая кабели, разъемы, драйверы (ECL, LED, лазеры), передатчики и приемники. Вместе с FC-1 этот уровень образует физический слой.

FC-1 определяет метод кодирования/декодирования (8B/10B) и протокол передачи, где объединяется пересылка данных и синхронизирующей информации.

FC-2 определяет правила сигнального протокола, классы услуг, топологию, методику сегментации, задает формат кадра и описывает передачу информационных кадров.

FС-3 определяет работу нескольких портов на одном узле и обеспечивает общие виды сервиса.

FC-4 обеспечивает реализацию набора прикладных команд и протоколов вышележащего уровня (например, для SCSI, IPI, IEEE 802, SBCCS, HIPPI, IP, ATM и т.д.)

FiberChannel – типы портов

5. Принципы организации и основные достоинства MPLS технологии.

6. Программно-конфигурируемые сети (SDN). Основные принципы, архитектура и преимущества. Протокол OpenFlow. Структура OpenFlow контроллера и коммутатора. Примеры применения.

SDN – разделение плоскости передачи и управления данными.

Даёт гибкость, снижение CAPEX и OPEX, позволяет вводить новые сервисы за счёт управления сетью целиком, удешевление разработки, программируемость, открытость.

Автоматизация управлением сети.

*Основные принципы SDN*:

* Физическое разделение уровня передачи данных от уровня управления сетевых устройств.
* Логически централизованное управление.
* Программируемость.
* Открытый единый интерфейс управления

Контроллер как сервер

Проблема: производительность, задержка и пропускная способность, надёжность, безопасность

Проблема взаимосвязей и конкуренции между applications

OpenFlow

1.0: Правила, 1.1: последовательные таблицы правил matching-a, групповые таблицы правил-действий, счётчики, 1.2: множество контроллеров с ролями

TCAM – троичная, ассоциативная память

Активный режим, проактивный режим,

Обнаружение связей.

Применение – частные компании (ограничения на части предприятия (бухгалтерия), wifi), ЦОД (повышение утилизации, виртуализация сети, качество доступа, мониторинг и оптимизация потоков, балансировка нагрузки), телеком-операторы (реакция на отказ канала, проблема интеграции с существующей сетью)

7. Виртуализация сетевых сервисов (NFV). Основные принципы, этапы развития, архитектура, преимущества. Примеры применения.

NFV - перенос сетевых функций на виртуальные машины:

* Упрощение развертывания и обновления как софта, так и железа
* Уменьшение стоимости за счет использования стандартных серверов
* Объединение сервисов в группы

Масштабируемость и устойчивость!

NF – Network function, virtualized network function, VNF forwarding graph, NFV infrastructure, Virtualized Infrastructure manager, User Service, NFV Service, Deployment behavior, NFV Orchestrator, VNF Forwarding Graph

TOSCA – описание входных параметров, логических компонент, политик оркестрации, выходных параметров

Варианты применения: Cloud (NFV infrastructure as a service, VNF as a service), mobile (пакетная кора), Datacenter (virtualization CDN), virtualization of home environment (домохозяйство)

8. Качество сервиса в компьютерных сетях: модели распределения ресурсов сети и методы борьбы с перегрузками.

Качество сервиса – качество обслуживания потоков на коммутационных устройствах: пропускная способность, задержка при передаче пакетов, уровень потерь, etc.

Управление качеством при помощи: очередизации (политика выбора пакетов) и шейпинга трафика (задержка или сброс).

Профиль потока определяется интенсивностью пересылки его данных.

Token Bucket – токены постепенно наполняют ведро.

Дисциплины очередизации – FIFO, drop-tail; Priority queuing; fair queuing; weighted fair queuing; weighted round robin; shared round robin; deficit weighted round robin

Reno, Tahoe, Cubic, …

9. Основные подходы математического моделирования компьютерных сетей. Прототипирование компьютерных сетей: преимущества, недостатки, ограничения применимости.

Модель – сущность/объект, который отображает процессы, протекающие в реальных системах с помощью математических или натурных средств.

Основные условия выбора метода:

• Постановка задачи

• Составом, характером и объемом исходных данных

• Временем на решение исследовательской задачи

Методы моделирования:

- ***натурный*** (адекватность и точность результатов, но необходимость доработки заглушек и отсутствие воспроизводимости результата)

- ***аналитический*** (на основе аналитических выражений) (быстро и для простых систем, много допущений, малая точность, сложность аналитического описания)

- ***иммитационный*** (изучаемая система заменяется моделью с достаточной точностью)

- ***комбинированные*** методы

Достоинства имитационного моделирования

– Высокая адекватность между физической сущностью описываемого процесса и его моделью

– Возможность описать сложную систему на достаточно высоком уровне детализации

– Значительно большие охват исследования, чем аналитическое моделирование

– Отсутствие ограничений на зависимости между параметрами модели

– Возможность оценки функционирования системы не только в стационарных состояниях, но и в переходных процессах (режимах)

– Получение большого числа данных об исследуемом объекте (закон распределения случайных величин, числовые значения абсолютные и относительные, и многое другое)

– Наиболее рациональное отношение «результат – затраты» по отношению к аналитическому и физическому моделированию

Недостатки: сложность создания модели, необходимость квалификации, необходимость верификации, индивидуальность реализации

Виды иммитационного моделирования:

- ***системная динамика*** - парадигма моделирования, где для исследуемой системы строятся графические диаграммы причинных связей и глобальных влияний одних параметров на другие во времени

- ***агентное моделирование*** – относительно новое (1990е-2000е гг.) направление в имитационном моделировании, которое используется для исследования децентрализованных систем, динамика функционирования которых определяется не глобальными правилами и законами (как в других парадигмах моделирования), а наоборот. Когда эти глобальные правила и законы являются результатом индивидуальной активности членов группы.

- ***дискретно-событийное моделирование*** – подход к моделированию, предлагающий абстрагироваться от непрерывной природы событий и рассматривать только основные события моделируемой системы, такие как: «ожидание», «обработка заказа», «движение с грузом», «разгрузка» и другие.

LXC-containers, mininet, ns3

10. Динамическое планирование задач в ИУС РВ. Схемы планирования Rate Monotonic (фиксированные приоритеты) и Earliest Deadline First (динамические приоритеты). Оценка времени отклика задач для схемы Rate Monotonic.

Для задач определены следующие характеристики:

* С – время выполнения задачи в наихудшем случае (WCET);
* D – дедлайн задачи;
* T – время между активациями задачи (период) (может быть просто минимальным временем между активациями, если период не фиксированный).

T == D



11. Понятие наихудшего времени выполнения программы (WCET). Факторы, влияющие на WCET. Фазы анализа WCET. Использование абстрактной интерпретации для выявления недопустимых путей. Анализ влияния конвейера на время выполнения программы.

12. Архитектура интегрированной модульной авионики (ИМА), её основные преимущества, примеры типов модулей (шина VME). Статико-динамическая схема планирования вычислений в системах ИМА.

Интегрированная модульная авионика (5 поколение)

• Унификация: вычислительные модули, сетевое оборудование и протоколы

• Интеграция: программное обеспечение, потоки данных

• Виртуализация: процессоры, память, сеть

Преимущества: надёжно, переносимо, возможность повторного использования, модульность, упрощение верификации и сертификации.

Примеры модулей: вычислительный, графический, ввода-вывода

VME – параллельная, адресная шина с арбитражём и прерываниями, прямой доступ к памяти модулей VME32 – 40 Mbyte, VME64 – 80 Mbyte

VME – медленная, блокирующая,

VME -> VPX

Статико-динамическое планирование: модули, ядра, окна выполнения, задачи и их характеристики, создать разделы задач, и привязать их к ядрам различных модулей, определить подходящие окна.

В одном окне – один раздел. В рамках окна решаются задачи вытеснения, приоритетов, ожидания входных данных.

13. V-образный жизненный цикл (ЖЦ) программного обеспечения. Основные виды инструментальных средств поддержки ЖЦ, их отнесение к фазам ЖЦ. Структура комплекса стендов для поэтапной интеграции ПО и аппаратуры ИУС РВ на восходящей фазе ЖЦ.

Информационно-управляющая система (ИУС) –вычислительная система верхнего уровня, обеспечивающая:

–функциональную и информационную интеграцию составных частей управляемого объекта

–взаимодействие между объектом и оператором

Средства тестирования

Средства поддержки разработки ПО

14. Средние и эмпирические операционные характеристики стратегий распознавания (классификаторов, регрессий). Проблема переобучения. Проблема устойчивости решений. Роль обучающей, валидационной и контрольной выборок при построении распознающей системы. Скользящий контроль (кросс-валидация). Регуляризация на примере линейной регрессии.

Семейство алгоритмов прогнозирования, генеральная совокупность, обучающая выборка, обобщающая способность.

***Обобщающую способность*** алгоритма прогнозирования можно определить, как точность на всей генеральной совокупности. Мерой обобщающей способности служит математическое ожидание потерь на генеральной совокупности E Ω {λ[Y, A(x)]}.

Под **устойчивыми** обучающими алгоритмами понимаются такие, которые дают решение, незначительно изменяющееся при малом изменении обучающей выборки.

многомерная регрессия, матрица плана.

Обучающая, валидационная, контрольная выборка.

Скользящий контроль.

Если выборка независима, то средняя ошибка скользящего контроля дает несмещенную оценку вероятности ошибки. Это выгодно отличает её от средней ошибки на обучающей выборке, которая может оказаться смещенной (оптимистически заниженной) оценкой вероятности ошибки, что связано с явлением переобучения.

Недостатками скользящего контроля являются: вычислительная неэффективность, высокая дисперсия, неполное использование имеющихся данных для обучения из-за сокращения длины обучающей выборки с L до ℓ.

Регуляризация - борьба с неустойчивостью.

регуляризация по Тихонову – включение штрафного слагаемого в оптимизируемый функцонал, несколько видов: ***гребневая регрессия***, метод Лассо, эластичная сеть.

15. Ансамбли классификаторов. Основные этапы работы типичного базового классификатора, возможность коррекции на разных этапах. Бэггинг и случайные подпространства. Бустинг. Случайный лес как композиция основных подходов к построению ансамбля.

Ансамбль, Бэггинг (коммитетный метод),

Бэггинг хорош, если вариационная составляющая ошибки базового метода высока (например, нейронная сеть или двоичное решающее дерево)

Решающий лес – это ансамбль решающих деревьев, которые строятся по искусственно сгенерированным обучающим выборкам, статистически сходных с исходной обучающей выборкой.

Бустинг – итерационное наращивание ансамбля алгоритмов.

16. Задача кластеризации как фундаментальная задача интеллектуального анализа данных, сопоставление с операцией группирования и задачей классификации. Различные постановки: разбиение, стохастическая, нечёткая, иерархическая, упорядочивание, однокластерная (последовательная). Примеры методов кластеризации для разных постановок.

Кластеризация: вектор характеристик объектов, нормализация, расстояние (евклидова метрика, квадрат евклидова расстояния, максимум модуля разности)

Иерархические строят вложенное дерево кластерного разбиения (восходящие и нисходящие подходы) (не всегда разбиение нужно доводить до конца). Плоские строят одноуровневое разбиение.

Чёткие выбирают класс, нечёткие – выдают вероятность принадлежности к классу.

Разбиение на классы на основе минимизации среднеквадратичной ошибки. (плоская класт.). Метод k-средних – последовательный пересчёт центра масс. Критерий останова: отсутствие изменений или малое изменение среднеквадратичной ошибки.

Нечёткий алгоритм с-средних – породия на k-means.

17. Дискреционное управление доступом. Модели HRU и Take-Grant. Задача проверки безопасности системы защиты от НСД.

А также см. прилагающиеся лекции.

Модели доступа:

* мандатная (бывает в обе стороны)
* ролевая
* дискретная (самое распространённое)

идентификация, аутентификация, авторизация

***модель Хариссона-Руззо-Ульмана (HRU)*** - суть в том, что здесь рассматриваются субъекты как объекты у которых можно менять права, и которые можно удалять или создавать

объекты, субъекты, права доступа, матрица прав доступа

enter delete create destroy

Можно проверить безопасность монооперационной системы перебором

Нельзя сделать универсальный алгоритм для этого из-за сводимости по Тьюрингу

Проблема троянских программ

Модель Белла-Лападулы (с уровнями секретности как в армии) - частный случай HRU

***Модель Take-Grant***

Теоретико-графовая модель

Take grant create remove

Предикат «возможен доступ»

tg-путь, остров – это макс. tg-связный подграф, начальный пролёт, конечный пролёт

Предикат «возможно похищение»

Расширенная модель take-grant: read write

Монитор безопасности – полнота и непрерывность, изолированность, верифицируемость.

18. Методы аутентификации в сети. Протокол аутентификации Kerberos.

Идентификация, аутентификация

Аутентификация – имущественные, биометрические, владение информацией

PAP, CHAP, MS-CHAP, цифровая подпись

1, 2, 3-х факторная аутентификация

Kerberous - AS - Authentication server, TGS - Ticket granting server

19. Пассивные и активные сетевые атаки (снифинг, спуфинг, MITM, имперсонация).

Пассивные атаки – сбор информации, сниффинг, немного активной атаки для дальнейшего пассивного прослушивания.

Host-lookup, port-lookup

AntiSniff

Активные атаки: arp-spoofing, dns-spoofing, tcp-spoofing, MITM, backdoors, DoS, DDoS, tiny fragment attack.

20. Коммуникационные протоколы. Ошибки, возникающие при передаче сообщений. Задача надежного обмена сообщениями. Симметричные протокол скользящего (раздвижного) окна: устройство протокола и обоснование его корректности. Протокол альтернирующего бита.[1, стр. 83-94]

21. Задача маршрутизации. Алгоритм Флойда-Уоршалла построения кратчайших путей в графе. Алгоритм маршрутизации Туэга: описание алгоритма, обоснование его корректностии оценка сложности по числу обменов сообщениями.[1, стр. 113-128]

Маршрутизация: вычисление таблиц маршрутизации, продвижение пакета.

Критерии оценки качества маршрутизации: корректность, эффективность, сложность, устойчивость, адаптивность, справедливость.

Теорема о дереве оптимальных путей

Ациклические таблицы – если они не содержат цикла ни для одной вершины.

Алгоритм Флойда-Уоршелла, Алгоритм Туэга, сложность по количество передач сообщений = N3

22. Общие принципы дедуктивной верификации программ. Операционная семантика императивных программ. Формальная постановка задачи верификации программ. Логика Хоара: правила вывода и свойства. Автоматизация проверки правильности программ. [4, c. 47- 70]

Интерпретация, сигнатура = <const, func, pred>

Отношение между входом и выходом

Операционный способ формального описания семантик (императивное программирование)

Состояние вычисления = состояние управления + состояние даннх

Пример – вычисление НОД

Отношения переходов для императивных программ: ASS, Comp, if, while

Частичное вычисление, если конечное, то просто «вычисление программы», результат вычисления.

Предусловие, постусловие, частично корректный, тотально корректный

Триплет Хоара

Правила вывода Хоара: ASS, CONS, COMP, IF, WHILE

Вывод в логике Хоара

Успешный вывод в интерпретации

Полнота правил вывода Хоара – нет

Слабейшее предусловие

23. Темпоральная логика деревьев вычислений CTL. Синтаксис и семантика CTL. Примеры спецификаций моделей в терминах формул CTL. Темпоральная логика линейного времени PLTL. Синтаксис и семантика PLTL. Свойства живости и безопасности. Ограничения справедливости. Задача верификации моделей (model-checking).[2, c. 55-63]

Атомарные высказывания

Модель Крипке – состояние, отношение переходов, функция разметки

событие, трасса, свойство

Свойства безопасности, живости, справедливости.

Темпоральные логики – A, **E**, **X**, F, G, **U**, R, и, **или**, **не**

CTL – логика ветвящегося времени.

LTL – логика линейного времени.

A, E - называются ***кванторами пути***, X А G U R - ***темпоральные операторы***, соответственно CTL формулы состоят из ***формул состояния*** и ***формул пути***

CTL\*, CTL, LTL – имеют разные выразительные мощности.

Не существует CTL, эквивалентной LTL формуле A(FG stable)

Не существует LTL, эквивалентной CTL формуле AG (EF restart)

Задача Model-Checking:

Смотри табличный алгоритм и его сложность как O(|глубина формулы| \* (|S| + |R|))

Сильно-связная нетривиальная компонента

Ограничения справедливости, справедливый путь

Делать вид, что нам не рассказывали символьный алгоритм верификации CTL.

24. Временные автоматы как формальные модели распределенных систем реального времени. Вычисления временных автоматов. Примеры использования временных автоматов для моделирования встроенных систем. Зеноновские вычисления. Синтаксис и семантика Timed CTL. Задача верификации моделей программ реального времени. Программно- инструментальное средство верификации моделей программ реального времени UPPAAL. [2, 344-353]

25. Дискретные цепи Маркова. Метод вложенных цепей Маркова при исследовании систем массового обслуживания.

Марковский процесс

Однородная марковская цепь, стационарное распределение, эргодичнось, неразложимость, периодичность.

Метод вложенных цепей Маркова – рассмотрение процесса в специально подобранные моменты времени, которые связаны в цепь Маркова.

26. Процессы гибели и рождения. Исследование марковских систем обслуживания с помощью теории процессов гибели и рождения.

Непрерывный марковский процесс, инфинити-земальные параметры, матрица инфинити-земальных параметров, консервативная цепь Маркова

Обратные и прямые уравнения Колмогорова

Разложимость, сжимаемость

Теорема Фостера (консервативная цепь маркова с непрерывным временем и счётным множеством состояний и неразложимая и сист ур. … имеет нетрив решение и сумма решений не бесконечная, тогда цепь Маркова эргодична)

Уравнения равновесия.

27. Понятие антагонистической игры. Верхнее и нижнее значения конечных и бесконечных антагонистических игр. Седловая точка. Необходимые и достаточные условия существования седловой точки. Теорема Фон Неймана о существовании седловой точки у вогнуто-выпуклых функций.

Антагонистическая игра, матричная игра.

Седловая точка

Нижнее и верхнее значение игры.

Необходимое и достаточно условие существования седл. точки – нижнее и верхнее значения игры должны быть равны.

28. Понятие потока в сети. Задача о максимальном потоке. Алгоритмы Форда-Фалкерсона и Карзанова. Теорема о максимальном потоке и минимальном разрезе. Сведение задачи составления допустимого расписания с прерываниями для многопроцессорной системы при заданных директивных интервалах к задаче о максимальном потоке в сети.

Граф, ориентированный граф, сеть, поток, увеличивающий путь, алгоритм форда-фалкерсона O(m \* nU), разрез, поиск увеличивающего пути методом пометок

алгоритм Корзанова, остаточная сеть, слоистая сеть, тупиковый поток

Алгоритм Упаковки – сведение задачи расписания к задаче построения максимального потока. Теорема Танаева.

29. Псевдополиномиальные алгоритмы решения задач: разбиение, рюкзак, расписание для многопроцессорной системы (число процессоров фиксировано).

Алгоритм решения задачи P называется псевдополиномиальным, если его вычислительная сложность ограничена сверху полиномом r(l(P), M(P)) (от функции длины входа и функции от максимального числа).

30. Метод ветвей и границ на примере минимаксной задачи теории расписаний Приближенные алгоритмы решения NP-трудных задач: упаковка в контейнеры, рюкзак, коммивояжер, расписание для многопроцессорной системы, вершинное покрытие. Оценки их сложности и погрешности.

УК – число контейнеров заполненных на половину или менее – не более 1-го

ЗК – строим остовное дерево, потом пользуемся неравенством треугольника

ВП – выбираем ребра, которые не связаны

ЗР – заполняем дикую таблицу всех комбинаций, для приближённого результата – удаляем младшие разряды из всех чисел.