**Моделирование информационных процессов**

**1. Общее описание, список некоторых команд NS-2. Файл трассировки.**

**NS-2 (Network Simulator 2)** – объектно-ориентированный симулятор сетей с дискретными событиями для моделирования и анализа сетевых протоколов и алгоритмов. Популярен в научных исследованиях благодаря гибкости и открытому исходному коду.

**Архитектура NS-2** основана на двух языках: C++ (для основных функций) и OTcl (для настройки симуляций). Включает ядро симулятора, интерпретатор OTcl, библиотеки сетевых компонентов и средства визуализации (NAM).

**Основные команды NS-2:** - *set ns [new Simulator]* – создание нового объекта симулятора - *$ns node* – создание узла сети - *$ns duplex-link $node1 $node2 bandwidth delay queue\_type* – создание двустороннего канала - *set tcp [new Agent/TCP]* – создание TCP-агента - *$ns attach-agent $node $tcp* – присоединение агента к узлу - *$ns connect $tcp $sink* – соединение агентов (создание связи) - *$ns at time "$node cmd"* – планирование события - *$ns run* – запуск симуляции

**Файл трассировки** содержит запись всех событий в сети. Каждая строка представляет событие и включает: - время события - источник и получатель пакета - тип и размер пакета - флаги и другие параметры - тип события (отправка ‘+’, извлечение ‘-’, получение ‘r’, отбрасывание ‘d’)

Анализ файла трассировки позволяет получить статистику: пропускную способность, задержку, джиттер, потери пакетов. NS-2 применяется для исследования сетевых протоколов, механизмов QoS, беспроводных сетей и алгоритмов маршрутизации. Сценарии моделирования описываются в виде OTcl-скриптов, что позволяет быстро настраивать и тестировать различные конфигурации сети.

**2. Реализация компонентного моделирования в подсистеме xcos математического пакета Scilab.**

**Xcos** – графический редактор блочных диаграмм в составе Scilab для моделирования динамических систем, аналогичный Simulink в MATLAB.

**Особенности Xcos:** - Графический интерфейс с drag-and-drop функциональностью - Обширные библиотеки блоков для различных областей - Моделирование непрерывных, дискретных и гибридных систем - Интеграция с основной средой Scilab

**Основные библиотечные блоки:** - Источники сигналов (генераторы, константы) - Математические операции (сложение, умножение, интеграторы) - Линейные системы (передаточные функции) - Нелинейные элементы (ограничители, переключатели) - Визуализация (осциллографы, дисплеи)

**Последовательность построения и отладки моделей:** 1. Создание новой диаграммы 2. Добавление блоков из библиотеки 3. Настройка параметров блоков 4. Соединение блоков линиями связи 5. Настройка параметров моделирования 6. Запуск симуляции 7. Анализ результатов

**Средства анализа результатов:** - Графики временных зависимостей - Спектральный анализ - Экспорт данных в рабочую область Scilab

**3. Основные понятия теории сетей Петри. Задачи анализа сетей Петри.**

**Сети Петри** – математический аппарат для моделирования динамических дискретных систем, особенно распределенных, параллельных и асинхронных процессов.

**Основные понятия:** - **Позиция** – пассивный элемент (кружок), представляющий условия, ресурсы, состояния системы - **Переход** – активный элемент (прямоугольник), представляющий события, действия - **Дуга** – соединяет позиции с переходами (входные и выходные) - **Маркировка** – распределение маркеров (фишек) по позициям, определяющее текущее состояние - **Входная функция** – отображает переход на мультимножество входных позиций - **Выходная функция** – отображает переход на мультимножество выходных позиций - **Разрешенный переход** – переход, для которого все входные позиции содержат достаточное количество маркеров

**Задачи анализа сетей Петри:** 1. **Безопасность** – свойство, при котором число маркеров в любой позиции не превышает 1 2. **Ограниченность** – существование числа k, ограничивающего количество маркеров в позициях 3. **Сохранение** – постоянство суммарного числа маркеров в сети 4. **Достижимость** – возможность перехода из начальной маркировки в заданную 5. **Покрываемость** – возможность достижения маркировки, не меньшей заданной 6. **Живость** – отсутствие тупиков, возможность срабатывания любого перехода

**4. Анализ сетей Петри путём построения и анализа дерева достижимости.**

**Дерево достижимости** – метод анализа сетей Петри, представляющий все возможные маркировки, достижимые из начальной, в виде направленного дерева.

**Принцип построения:** 1. Корень дерева – начальная маркировка M₀ 2. Для каждой маркировки определяются разрешенные переходы 3. Для каждого перехода вычисляется новая маркировка 4. Если маркировка новая – добавляется вершина, иначе – только дуга к существующей вершине 5. Процесс продолжается рекурсивно

**Проблема бесконечности** решается введением символа “ω”, обозначающего произвольно большое число маркеров. Если обнаруживается маркировка M’, которая больше некоторой маркировки M на пути от корня, в позициях с увеличением числа маркеров значение заменяется на ω.

**Анализ дерева позволяет определить:** - Ограниченность (отсутствие ω в дереве) - Безопасность (только 0 и 1 в маркировках) - Достижимость (наличие вершины с искомой маркировкой) - Наличие тупиков (листья дерева)

**5. Применение метода построения дерева достижимости к решению задач определения безопасности и ограниченности сети Петри.**

**Определение безопасности:** 1. Построить дерево достижимости для начальной маркировки 2. Проверить все вершины дерева: - Если в любой позиции встречается число маркеров > 1 или символ ω, сеть не безопасна - Если все маркировки содержат только 0 и 1, сеть безопасна

**Определение ограниченности:** 1. Построить дерево достижимости для начальной маркировки 2. Проверить наличие символа ω: - Если символ ω присутствует, сеть неограничена - Если символ ω отсутствует, сеть ограничена, с максимальным числом маркеров k

**Определение свойства сохранения:** 1. Построить дерево достижимости 2. Если в дереве есть символ ω, сеть не является строго сохраняющей 3. Иначе вычислить сумму маркеров для каждой маркировки: - Если сумма одинакова для всех маркировок, сеть сохраняющая - Если суммы различаются, сеть не сохраняющая

**6. Функциональность, назначение и параметры блоков GPSS.**

**GPSS (General Purpose Simulation System)** – язык и система имитационного моделирования дискретных событий для моделирования систем массового обслуживания.

**Основные концепции GPSS:** - **Транзакты** – динамические объекты, моделирующие заявки, клиентов, документы, **Блоки** – функциональные элементы, обрабатывающие транзакты, **Устройства** – ресурсы, которые могут быть заняты транзактами, **Очереди** – структуры для ожидающих транзактов

**Основные блоки GPSS:**

**GENERATE** – создает транзакты с заданным интервалом

Параметры: интервал времени, смещение, лимит, приоритет

**TERMINATE** – уничтожает транзакты, уменьшает счетчик завершения

Параметры: значение уменьшения счетчика

**ADVANCE** – задерживает транзакт на указанное время

Параметры: среднее время задержки, разброс времени

**SEIZE** – захват устройства транзактом

Параметры: имя устройства

**RELEASE** – освобождение устройства транзактом

Параметры: имя устройства

**QUEUE/DEPART** – вход/выход из очереди

Параметры: имя очереди, количество мест

**ENTER/LEAVE** – занятие/освобождение многоканального устройства

Параметры: имя устройства, количество каналов

**TRANSFER** – передача транзакта по одному из направлений

Параметры: режим передачи, основной и альтернативный адреса

**7. Дополнительные блоки GPSS и их параметры.**

**Дополнительные блоки GPSS:**

**PRIORITY** – изменение приоритета транзакта

Параметры: новое значение приоритета, режим прерывания

**SPLIT** – создание копий транзакта

Параметры: количество копий, адрес направления копий

**ASSIGN** – присвоение значения параметру транзакта

Параметры: номер параметра, значение, модификатор операции

**TEST** – сравнение операндов для определения пути транзакта

Параметры: операция сравнения, операнды, адрес при невыполнении

**GATE** – проверка состояния устройства или логического ключа

Параметры: тип условия, проверяемый объект, адрес перехода

**LOGIC S/R** – установка/сброс логического ключа

Параметры: имя логического ключа

**SAVEVALUE** – сохранение значения в ячейке памяти

Параметры: имя величины, значение, модификатор операции

**LOOP** – организация цикла в модели

Параметры: параметр-счетчик, адрес для повторения

**PREEMPT/RETURN** – прерывание устройства с последующим возвратом

Параметры: имя устройства, режим прерывания, адрес перехода

GPSS автоматически собирает статистику по очередям, устройствам и другим элементам модели, делая его эффективным инструментом для анализа систем обслуживания.