

Рассчитаем номиналы RC-элементов для фильтра нижних частот таким образом, чтобы частота среза равнялась $f_{\text{среза}} = 3.3 \text{ Гц}$. Пусть сопротивление R3 будет равно 680кОм, тогда емкость конденсатора C2:

$$C2 = \frac{1}{2\pi R3 f_{\text{среза}}} = \frac{1}{2\pi \cdot 430 \cdot 10^3 \cdot 3.3} = 100 \text{ нФ}$$

Правая часть рисунка 5 иллюстрирует результат моделирования рассчитанного ФНЧ. Как можно заметить, частота среза в этом случае составляет 3.477 Гц, что также удовлетворяет поставленным условиям.

После фильтрации и усиления аналоговый сигнал поступает на 10-битный АЦП, входящий в состав используемого микроконтроллера. В качестве периода получения данных в микроконтроллере выбрано значение 2 мс. Задачей программы, выполняемой на микроконтроллере, является поиск пиков и определение временного интервала между ними, с последующим пересчетом в количество ударов в минуту. Вычисленное значение передается по шине USB принимающей программе, запущенной на ПК, которая в свою очередь осуществляет усреднение, необходимое для исключения ошибок измерений, а также визуализацию данных.

Список цитированных источников

1. Martini F., Bartholomew E. *Essentials of Anatomy & Physiology*. / San Francisco: Benjamin Cummings, 2003. - P. 267

2. Костюк Д.А., Латий О.О. Оценка состояния пользователя с помощью платформы Arduino // Информационные технологии и системы 2014 (ИТС 2014): материалы международной научной конференции. Минск, БГУИР, 29 октября 2014 г. – С. 57–58.

УДК 681.3

Медведский К.И.

Научный руководитель: проф. Муравьев Г.Л.

РАЗРАБОТКА АЛГОРИТМОВ, МОДЕЛЕЙ, СРЕДСТВ ПОСТРОЕНИЯ ОПИСАНИЙ СИСТЕМ

Введение. Постановка задачи. В практике инженерной деятельности широко используется моделирование (имитационное, аналитическое), реализуемое на базе описания систем в терминах q-схем, стохастических сетевых моделей (ССМ), сетей массового обслуживания (СМО) [1, 2]. Вместе с тем существует широкий спектр универсальных систем, средств имитационного моделирования, избыточных для решения задач конкретного класса.

Здесь рассматриваются проблемы, связанные с автоматизацией получения модельных спецификаций по формальному описанию систем, то есть кодов имитационных моделей (ИМ) на входных языках систем моделирования. Также проблема автоматической генерации сетевых спецификаций (параметров архитектуры), согласованных с требованиями к их сложности и режиму функционирования, и тестовых спецификаций, включающих эталонные значения характеристик функционирования сети.

Возможные области применения: - имитационное моделирование стохастических сетей, специализация возможностей существующих систем моделирования за счет "обертывания" дружественным интерфейсом. Поддержка тести-

рования разработанных моделей (ИМ). Поддержка контролируемого обучения имитационному моделированию за счет обеспечения наиболее трудоемких технологических этапов, таких как "ручное" получение необходимого числа корректных, несовпадающих вариантов сетевых спецификаций, а также проведение аттестации разработанных моделей - оценка адекватности моделей, требующая наличия соответствующего количества тестовых спецификаций и др.

В качестве математических моделей систем используется класс сетевых моделей – СеМО и расширение СеМО до ССМ, упрощенных в части использования управляющих узлов и включающих ресурсы ограниченной емкости (обслуживающие узлы типа "память").

В качестве системы моделирования выбрана универсальная система GPSS World [2], имеющая бесплатные версии со входным языком GPSS, обеспечивающим полное отображение указанных сетей в процессном стиле.

Общая схема получения спецификаций приведена ниже и включает следующие процедуры:

Получение спецификаций каркасов (структур) сетей.

Генерация сетевых спецификаций в терминах СеМО.

Генерация сетевых спецификаций в терминах ССМ (на базе спецификаций СеМО).

Построение html, xml описаний сетевых спецификаций, включая "человеко-читаемые".

Генерация модельных спецификаций сетей (GPSS-кодов).

Построение тестовых спецификаций.

Хранение спецификаций.

Генерация сетевых спецификаций. Используемый аппарат: – теория графов, комбинаторные методы для порождения каркасов сетей; – модели массового обслуживания для расчета параметров сетей, вероятностные методы их доопределения; – объектно-ориентированный подход для реализации средств, xml-, html-форматы для организации хранения и отображения спецификаций.

Требования к сетевым спецификациям и средствам их получения: – управляемость сложностью и режимом функционирования генерируемых сетей (архитектур); уникальность, не повторяемость спецификаций; полнота описаний как достаточность данных для моделирования, тестирования сетей; документированность.

Укрупненный алгоритм получения сетевых спецификаций приведен ниже.

Определить ограничения на сложность сети (А)

Задать параметры режима функционирования сети (В)

ГЕНЕРИРОВАТЬ-КАРКАСЫ-СТРУКТУР сетей (С), удовлетворяющие ограничениям А

ПРОВЕРИТЬ-КОРРЕКТНОСТЬ структур сетей С

Сохранить описания корректных структур (D)

Разыграть количество потоков (Q)

Разыграть количество обслуживающих узлов (типа "устройство") и их каналности

ЦИКЛ-ПОКА не сгенерирована сетевая спецификация для каждой структуры из D

Распределить вероятностные узлы в соответствии с А, Q

Сформировать потоковые матрицы вероятностей переходов (Р)

Рассчитать интенсивности сетевых и узловых потоков

Рассчитать трудоемкости обслуживания в узлах, удовлетворяющие ограничениям В

Декомпонировать узловые параметры обслуживания (трудоемкости) по потокам Q

Разыграть законы распределения поступления и обслуживания заявок

Доопределить параметры законов распределения

КОНЕЦ-ЦИКЛА

Расставить сетевые и узловые узлы-памяти в соответствии с ограничениями А

ИСКАТЬ-СЕТЕВЫЕ-ФРАГМЕНТЫ (Е) для расстановки узлов-памятей

ПРОВЕРИТЬ-КОРРЕКТНОСТЬ сетевых фрагментов Е и сохранить корректные (F)

Разместить узлы-памяти в соответствии с ограничениями В и множеством фрагментов F

Корректировать сетевые спецификации (параметры сетей, матрицы P)

Сохранить спецификации в html, xml-форматах

Здесь в основу генерации СеМО [2, 3] положен аналитико-рандомизированный подход. Он обеспечивает: – получение каркасов сетей (КС) заданной сложности на базе рекурсивного алгоритма перебора; – “оснащение” КС вероятностными узлами; - отсеивание некорректных КС; – аналитическое до определение параметров КС, декомпозицию параметров с учетом неоднородности сети с последующим вероятностным до определением недостающих параметров.

В основу генерации ССМ положены алгоритмы поиска фрагментов сети с последующей расстановкой емкостных узлов и вероятностным до определением недостающих параметров ССМ. Для этого рассмотрены свойства фрагментов, позволяющие проводить их валидацию, а также при генерации заменить полный перебор направленным, выполняемым с учетом заданных ограничений на базе анализа матриц переходов $\{P^{(q)}\}$.

Соответственно введены определения, используемые при поиске фрагментов – кандидатов на размещение емкостных ресурсов. Пусть сгенерирована СеМО, структура которой описывается множеством узлов $V = \{b_i, i = \overline{1, N}\}$ и потоков Q. Вероятностные матрицы потоков $\{P^{(q)}\}$ определяют для каждого из них множество дуг-переходов $\{D^{(q)} = \{d^{(q)}_{i,j}\} | q = \overline{1, Q}\}$. Поточковый фрагмент это часть сети, составленная подмножеством узлов $V_1^{(q)}$ и подмножеством дуг q-го потока $D_r^{(q)} = D_{11}^{(q)} \cup D_{12}^{(q)} \cup D_{13}^{(q)}$. Здесь соответственно представлены множества дуг $d^{(q)}_{i,j}$, ведущих во фрагмент ($b_j \in V_r^{(q)}; b_i \in V_1^{(q)}$), внутренние дуги ($b_i \in V_1^{(q)}; b_j \in V_r^{(q)}$), выходные, ведущие за пределы фрагмента. Тогда для корректного произвольного потокового фрагмента в сети не должно быть других дуг $d_{i,j} \notin D_r^{(q)}$, не принадлежащих фрагменту и ведущих в него или из него, т.е. входных-выходных по определению. Корректный произвольный сетевой фрагмент это потоковый, одинаковый для всех потоков сети. Простой корректный потоковый фрагмент – такой произвольный, где $\|D_{11}^{(q)}\| = \|D_{13}^{(q)}\| = 1$, т.е. с единственной дугой-входом и дугой-выходом. Особенность фрагмента – внутренние дуги обеспечивают достижимость любого узла фрагмента (то есть существуют равновероятностные пути, позволяющие заявке, попавшей во фрагмент, посетить любой узел до выхода из фрагмента).

Для поддержки функциональности указанных процедур разработана иерархия классов C++ [4], упрощенная UML-схема которой приведена на рисунке 1 ниже.

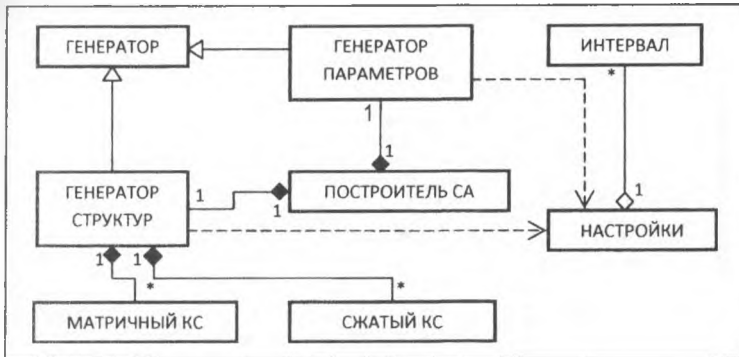


Рисунок 1 - Схема UML-иерархии генерации сетевых спецификаций

Генерация модельных и тестовых спецификаций Используемый аппарат: методы имитационного моделирования дискретных систем, теории массового обслуживания; объектно-ориентированный подход, методы каркасного программирования, принципы динамического полиморфизма для поддержки функциональности средств.

Базовые требования к спецификациям и средствам их получения: – согласованность по форматам с источниками формальных сетевых спецификаций систем; – полнота, структурированность, читаемость GPSS-кодов; – обеспечение управления сбором узловой, потоковой и системной статистик; – поддержка базового класса типов узлов, законов распределений параметров сетей.

Укрупненный алгоритм получения модельной спецификации приведен ниже.

Читать сетевую спецификацию из БД спецификаций

Конвертировать спецификацию в объектную модель (ОМ)

Инициализировать служебные структуры (описания имен, точек сбора статистики и др.)

ГЕНЕРИРОВАТЬ-КОД управления сетевой статистикой

ЦИКЛ-ПОКА не обработаны все потоковые спецификации Q

ГЕНЕРИРОВАТЬ-КОД управления потоковой статистикой

ЦИКЛ-ПОКА не просмотрены все потоковые узлы

ГЕНЕРИРОВАТЬ-КОД обслуживающего узла

ГЕНЕРИРОВАТЬ-КОД управления узловой статистики

ГЕНЕРИРОВАТЬ-КОД маршрутных узлов

КОНЕЦ-ЦИКЛА

КОНЕЦ-ЦИКЛА

Сохранить модельные спецификации

Работа алгоритма базируется на специально разработанной объектной модели (ОМ), обеспечивающей как генерацию кодов так и функционирование дружественного интерфейса, позволяющего пользователю при работе в диалоговом режиме вносить коррективы в параметры сети, оперативно управлять сбором статистики. Генерация выполняется с использованием разработанных правил именования и модельного отображения элементов сети. При расстановке маршрутных узлов минимизируется число переходов.

Для поддержки функциональности указанных процедур разработана иерархия классов C++ [4], упрощенная UML-схема которой приведена на рисунке 2 ниже.

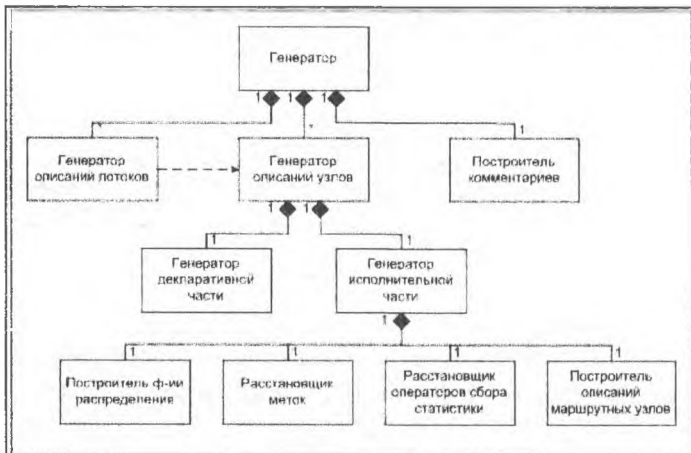


Рисунок 2 - Схема UML-иерархии генерации модельных спецификаций

Заключение. В работе обоснован подход к автоматизации построения сетевых спецификаций систем, включая модельные, удовлетворяющих пользовательским требованиям, заданным ограничениям. Разработаны макеты систем. Генератор GPSS-моделей реализован в двух вариантах: консольном для использования совместно с модулем генерации сетевых спецификаций для получения тестовых спецификаций, поддержки тестирования моделей, процессов обучения моделированию и в диалоговом варианте для автономного применения в качестве инструмента моделирования сетей.

Макетирование системы проводилось в объектно-ориентированной парадигме [4] средствами языка C++. Для поддержки вычислений использована библиотека линейной алгебры uBLAS (из собрания библиотек Boost), STL. При реализации генераторов отчетов, связанной с производством html-документов, экспортом, загрузкой xml-документов (в базе данных), а также для поддержки графического интерфейса был использован кросс-платформенный инструментариий QT.

Полученные результаты составляют основу для создания полнофункциональных систем авто генерации сетевых спецификаций и результативных имитационных моделей по их сетевым спецификациям.

Список цитированных источников

1. Ивницкий, В.А. Теория сетей массового обслуживания / В.А. Ивницкий. – М., Физико-математическая литература, 2004. – 772 с.
2. Рыжиков, Ю.И. Имитационное моделирование. Теория и технологии / Ю.И. Рыжиков. – СПб.: КОРОНА, 2004. – 320 с.
3. Муравьев, Г.Л. Автоматизация получения тестовых описаний систем для обучения моделированию / Г.Л. Муравьев, А.Н. Никонюк // Инновационные технологии обучения физико-математическим дисциплинам: материалы III междунар. научно-практ. конф., Мозырь, 2011. – С. 85-86.
4. Труб, И.И. Объектно-ориентированное моделирование на C++ / И.И. Труб. – СПб.: Питер, 2006. – 411 с.