

УДК 004.72

МЕТОДЫ И ТЕХНОЛОГИИ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ СИСТЕМ И КОМПЬЮТЕРНЫХ СЕТЕЙ

В. П. Корячко, д.т.н., профессор, зав. кафедрой «Систем автоматизированного проектирования вычислительных средств» ФГБОУ ВО «РГРТУ»; koryachko.v.p@rsreu.ru

С. В. Скворцов, д.т.н., профессор кафедры «Систем автоматизированного проектирования вычислительных средств» ФГБОУ ВО «РГРТУ»; s.v.skvor@gmail.com

А. П. Шибанов, д.т.н., профессор кафедры «Систем автоматизированного проектирования вычислительных средств» ФГБОУ ВО «РГРТУ»; apshibanov@yandex.ru

Д. А. Перепелкин, к.т.н., доцент кафедры «Систем автоматизированного проектирования вычислительных средств» ФГБОУ ВО «РГРТУ»; dmitryperpelkin@mail.ru

Рассмотрены основные результаты, полученные сотрудниками кафедры САПР ВС РГРТУ, в области автоматизации проектирования высокопроизводительных систем и компьютерных сетей.

Ключевые слова: GERT-сети, эквивалентные преобразования, аналитические функции, программно-конфигурируемые сети, адаптивная маршрутизация, многопутевая маршрутизация, балансировка нагрузки, многопоточные программы, многоядерные процессоры, графический ускоритель.

DOI: 10.21667/1995-4565-2017-60-2-94-104

Применение теории GERT-сетей для моделирования телекоммуникаций

Рассматриваются результаты использования теории GERT-сетей [1] (GERT – *graphical evaluation and review technique*) для расчета телекоммуникаций: 1) на основе метода эквивалентных упрощающих преобразований структуры сети [2-5], 2) на основе теории аналитических функций комплексного переменного [6-8]. Преимущества метода эквивалентных упрощающих преобразований по сравнению с применяемыми в классической теории топологическими методами с использованием формулы Мейсона в наибольшей степени проявляются тогда, когда модель содержит большое число циклов. Возможности метода раскрываются на примерах моделей программ микроконтроллеров с контрольными точками [9] и построения систем с выполнением операций восстановления в случае отказов оборудования. Кроме того, рассматривается пример использования теории вычетов для планирования процессов передачи трафика при проведении испытаний летательных аппаратов [8].

Метод эквивалентных упрощающих преобразований структуры GERT-сети. На каждом шаге из сети исключается один узел. Фрагменты «последовательные дуги», «параллельные дуги», «дуга-петля» заменяются эквивалентными дугами. Вычисляются их веса в виде характеристических

функций. В конечном итоге структура GERT-сети приводится к единственной эквивалентной дуге. Плотность распределения вероятностей $f(t)$ выходной величины GERT-сети находится из выражения: $f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-ix\zeta} \chi_E(\zeta) d\zeta$. В этом выражении $\chi_E(\zeta)$ – эквивалентная характеристическая функция GERT-сети. Функция $\chi_E(\zeta)$ интерполируется многочленом Лагранжа $L_2(\zeta)$ второй степени с равноотстоящими узлами. Вычисляется плотность распределения вероятностей $f(t)$ в заданном диапазоне. Временная и асимптотическая емкость сложности алгоритма есть $O(n^4)$, где n – число узлов в сети.

Разработана программа GERT Explorer. Она реализует:

- русскоязычный и англоязычный многооконные интерфейсы;
- масштабирование изображения моделей;
- возможность копирования в буфер и считывания из него фрагментов изображения GERT-сети с нумерацией узлов и дуг;
- цветовую раскраску узлов и дуг; представление дуг в виде прямых линий, кривых Безье, ступенчатых и округлых линий;
- совмещение на экране изображения нескольких плотностей и функций распределения вероятностей выходных величин GERT-сетей;

- привязку элементов графического изображения к масштабной сетке;
- функцию «отката»;
- сохранение и считывание моделей;
- выдачу таблиц вычисленных распределений в файловую систему для интерфейса с другими системами моделирования и т.д.

Моделирование систем с контрольными точками. Работоспособность микроконтроллеров, работающих в условиях помех, может нарушаться. Это происходит из-за заикливания и непредусмотренных остановов программ. Для выхода из этих состояний микроконтроллеры оснащаются сторожевыми таймерами (СТ). Это счетчик, в который записывается двоичный код, определяющий интервал выдержки T . В процессе функционирования микроконтроллера из значения T последовательно вычитается единица. При достижении нулевого значения СТ срабатывает. При срабатывании СТ может выполнять две функции: 1) при неработоспособности микроконтроллера СТ запускает его с предыдущей контрольной точки (КТ); 2) пусть ξ – выходная случайная величина GERT-сети, связывающая две соседние КТ. Если микроконтроллер работоспособен, но $\xi \geq T$, то также выполняется переход к предыдущей КТ.

Находится компромисс между увеличением времени работы протокола, затрачиваемого на организацию КТ, и уменьшением среднего значения интервалов выдержки сторожевого таймера, обра-

батывающего «зависания» микроконтроллера.

Уменьшение значения T необходимо для сокращения времени неработоспособности микроконтроллера при «катастрофическом» отказе. Он ликвидируется при срабатывании сторожевого таймера. С другой стороны, уменьшение значения T увеличивает число КТ в программе и время ее выполнения. Местами установки КТ в модели являются узлы-доминаторы. Узел d графа доминирует над узлом n , если любой путь от входного узла графа к n проходит через d .

Пример GERT-сети с КТ приведен на рисунке 1. Три одинаковых фрагмента отмечены прямоугольниками пунктирной линией. Они соединены КТ, которым соответствуют узлы 7 и 14. На операции чтения (W_{10} , W_{12} , W_{22}) и записи (W_{11}), (W_{21}), (W_{31}) КТ затрачивается случайное время, распределенное равномерно от 8 до 12 ед. машинного времени. Остальные операции в модели характеризуются распределением Эрланга 2 порядка с интенсивностью $\mu = 4$.

Особенностью данной модели является то, что она является ячеистой сетью, содержащей большое число петель первого и более высоких порядков. Классический подход с использованием формулы Мейсона характеризуется экспоненциальной сложностью. Поэтому использован метод эквивалентных упрощающих преобразований, для которого большое число петель любых порядков не имеет значения, а вычислительная сложность является полиномиальной.

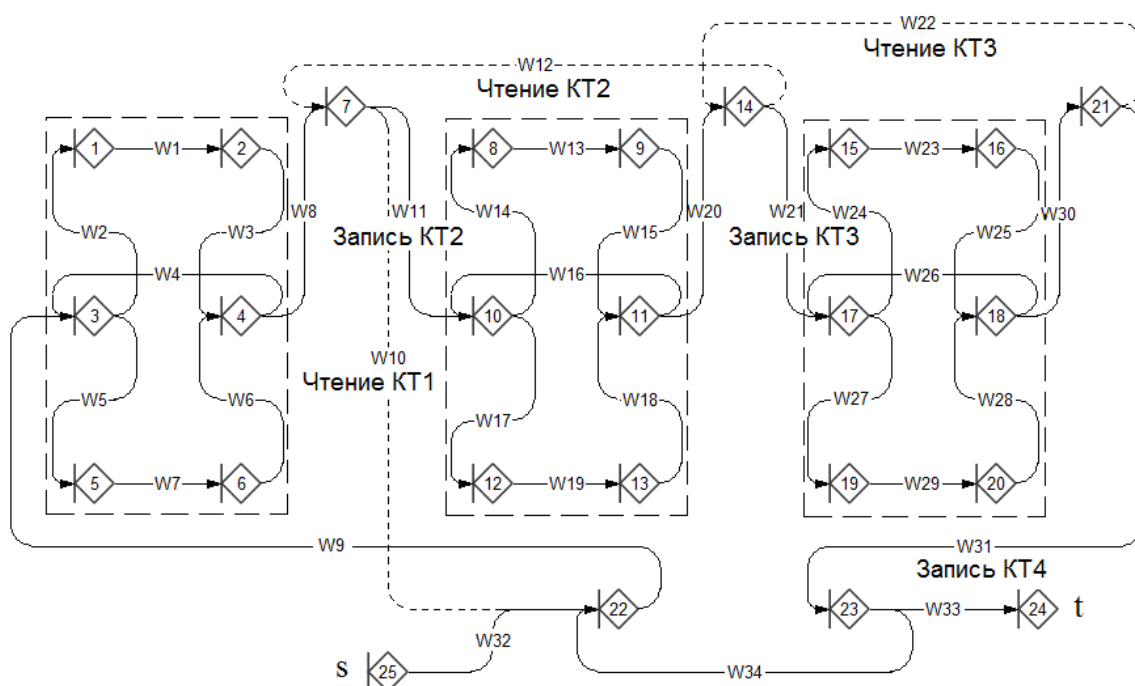


Рисунок 1 – Модель GERT-сети с КТ

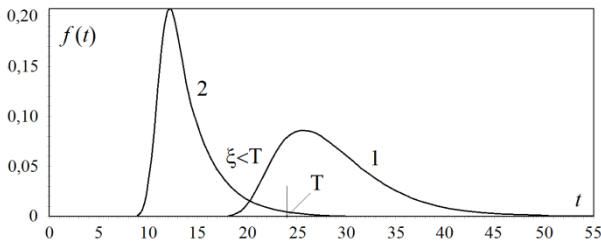


Рисунок 2 – Плотности распределения вероятностей: 1) полной сети; 2) фрагмента между КТ

На рисунке 2 приведена плотность распределения вероятностей выходной величины GERT-сети (кривая 1). Она получена при вероятности успешного прохождения КТ $p = 0,999$. Среднее время выполнения модели $\bar{t} = 28,82$, а его дисперсия $\sigma^2 = 37,17$.

Плотность распределения 2 получена при условии возможного срабатывания сторожевого таймера. Среднее время прохождения фрагмента $\bar{t}^* = 14,01$, а его дисперсия $\sigma^{2*} = 9,88$. Когда исходы случайной величины ξ не превышают величины T_i , сторожевой таймер не успеваешь обнулиться. Тогда в него записывается новое значение T_{i+1} . Значение T_{i+1} выбирается по интервалу «трех сигм». При этом вводится некоторый запас на разброс параметров Δ в технической системе, то есть $T_{i+1} = \bar{t}^* + 3\sigma + \Delta$.

При использовании нескольких КТ сокращается интервал выдержки сторожевого таймера. При этом уменьшается время неработоспособности технической системы в случае «катастрофического» отказа.

В модели проходятся различные фрагменты между i -й и $i+1$ КТ. Каждому такому фрагменту может назначаться свой интервал выдержки T_i сторожевого таймера. В любой технической системе принято определять основные показатели качества. К таким параметрам относится и значение интервала выдержки сторожевого таймера T микроконтроллера.

Модель канала передачи данных с обнаружением отказов и восстановлением работоспособности. Рассматривается задача нахождения вероятностно-временных характеристик специализированного канала передачи информации. В канале имеются средства проверки его готовности к выполнению текущей операции. Реальные каналы могут быть не готовыми к функционированию по различным причинам. Это может происходить:

– из-за неисправности технических средств;

– из-за изменения режимов работы канала, в частности включения резервного оборудования и т.п.

Задаются вероятности неготовности узлов канала и его восстановления. Известен закон распределения времени восстановления канала.

В высокопроизводительных базовых каналах крупных сетей и специализированных сетях реального времени выделяются относительно независимые функциональные группы, имеющие средства восстановления функционирования (например, за счет резервирования аппаратуры). В моделях таких систем имеется большое число операций циклического характера.

Модель, в которую включены операции проверки и восстановления работоспособности блоков, изображена на рисунке 3. Операции W4, W23, W26 выполняют восстановление узлов 5, 13, 10. Они составляют петли 1 порядка простейшего вида: (4, 4), (15, 15), (17, 17). Совместно с петлями первого порядка, содержащими более двух узлов, (6, 7, 6), (8, 9, 16, 10, 8) и (8, 9, 16, 17, 10, 8) они в различных сочетаниях друг с другом образуют петли более высоких порядков. Из-за наличия большого числа циклических операций расчеты с использованием формулы Мейсона, как это принято в классической теории GERT-сетей, характеризуются экспоненциальной сложностью. Даже для этой простой модели приходится оперировать петлями до 5 порядка включительно, что увеличивает число слагаемых в формуле Мейсона и усложняет расчеты. При применении метода эквивалентных упрощающих преобразований на первом же шаге выполняется замена фрагментов «дуга-петля» графа на эквивалентные дуги, что уменьшает число петель первого и более высоких порядков. При исключении узлов с эквивалентным пересчетом характеристической функции выходной величины GERT-сети число петель разных порядков последовательно уменьшается. Это значительно упрощает расчеты.

На рисунке 4 изображены результаты компьютерного моделирования с использованием программы GERT Explorer. Плотность распределения вероятностей времени передачи пакета при вероятностях отказов в узлах 0,01 и вероятности восстановления работоспособности узлов с первой попытки 0,99 отмечена цифрой 1.

Плотности при вероятностях отказов 0,05; 0,1 и восстановлений 0,95; 0,9 показаны цифрами 2 и 3.

Все распределения имеют две моды. Для уменьшения колебаний времени выполнения процесса следует так подбирать параметры операций, чтобы итоговое распределение стало одномодовым.

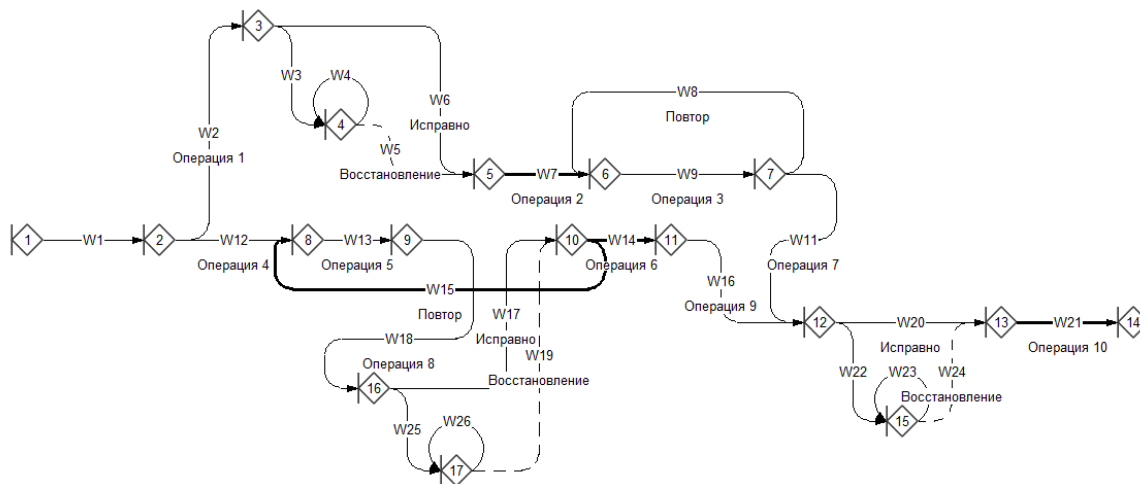


Рисунок 3 – Пример GERT-сети передачи кадра с отказами и восстановлениями каналов

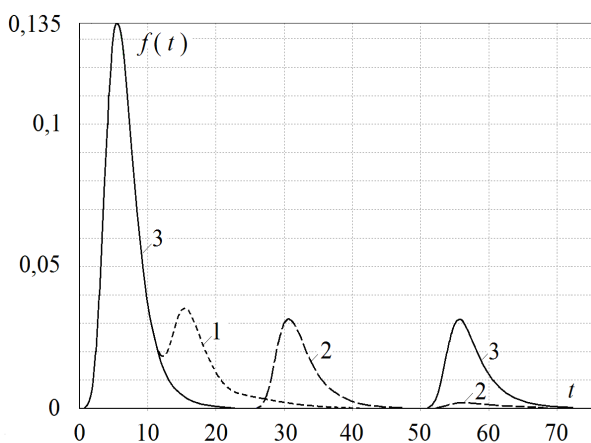


Рисунок 4 – Плотность распределения времени передачи кадра с отказами и восстановлениями каналов

Использование метода эквивалентных упрощающих преобразований структуры GERT-сети позволяет получить точное распределение вероятностей ее выходной случайной величины и учесть степень влияния длительности операций восстановления на вероятностно-временные характеристики моделируемого процесса.

Планирование полосы пропускания сетевых каналов при проведении испытаний летательных аппаратов

При проведении летных испытаний изделий авиационной и ракетно-космической техники летательные аппараты (ЛА) переходят из зоны слежения одних измерительных пунктов (ИП) в зоны слежения других ИП. Поэтому трафик реального времени, передаваемый по каналам в центры управления (ЦУ) ходом полета, динамически меняется. Актуальной является задача обеспечения достаточной величины полосы пропускания каналов связи для обеспечения своевременной передачи измерительной информации (ИИ).

Для одного виртуального пути «из конца в конец» могут решаться следующие задачи:

- полный расчет распределения времени передачи кадра ИИ от маршрутизатора зоны до ЦС;
- частичная корректировка расчетов при удалении объекта в направлении от ЦС с сохранением данных предварительных вычислений;
- частичная корректировка расчетов при перемещении объекта в направлении к ЦС с сохранением данных предварительных вычислений.

Время передачи кадра через канал связи отражается нормированным распределением Эрланга 2 порядка. Изначально для текущих значений географических координат ЛА выполняются расчеты плотности распределения вероятностей времени передачи кадра ИИ для множества N виртуальных каналов соединения «из конца в конец» по выражению

$$f(t) = \sum_{m=1}^N \text{Res}_{z=-2\mu_m} \left[4^N e^{zt} \prod_{\forall m} \frac{e^{-zT_m} \mu_m^2}{(2\mu_m + z)^2} \right]$$

через сумму вычетов относительно всех особых точек $z = -2\mu_m$.

Если данный путь удлиняется на одно или несколько звеньев до величины N^+ , то производится корректировка расчетов по формуле:

$$\text{Res}_{z=-2\mu_m} = 4 \prod_{\varepsilon=1}^{N^+} \mu_\varepsilon^2 e^{-2\mu_m \left(t - \sum_{\varepsilon=1}^{N^+} T_\varepsilon \right)} \times \left\{ \left[\left(t - \sum_{\varepsilon=1}^{N^+} T_\varepsilon \right) - A \right] / B \right\}, m = \overline{1, N^+}.$$

В выражении $A = \sum_{\forall n, n \neq m}^{N^+} 1 / (\mu_n - \mu_m)$ содержится $N^+ - 1$ слагаемых. В разностях $\mu_n - \mu_m$ индекс n пробегает все значения от 1 до N^+

кроме $n = m$; при этом значение m постоянно и равно номеру особой точки. В формуле $B = \prod_{\forall n, n \neq m}^{N^+} (\mu_n - \mu_m)$ разности $\mu_n - \mu_m$ образуются подобно тому, как это выполняется для определения A . Последовательным добавлением сомножителей μ_ε^2 и T_ε формируется выражение

$$\prod_{\varepsilon=1}^{N^+} \mu_\varepsilon^2 \exp \left[-2\mu_m \left(t - \sum_{\varepsilon=1}^{N^+} T_\varepsilon \right) \right].$$

Аналогично выполняется и корректировка вычетов, если число звеньев в пути «из конца в конец» уменьшается.

Предлагаемый метод определения полосы пропускания каналов полигонной сети передачи данных может быть использован для:

- нахождения резерва полосы пропускания наиболее загруженного канала сети;
- определения среднего значения резерва полосы пропускания либо всего множества каналов сети, либо наиболее загруженных из них;
- модернизации сети в тех случаях, когда величина резерва полосы пропускания какого-либо канала недостаточна.

Корректировка расчетов при удалении объекта в направлении от ЦС или при приближении к нему (при сохранении данных предварительных вычислений) может выполняться в режиме реального времени.

Исследования по данному направлению выполнены при поддержке Российского фонда фундаментальных исследований, грант 14-07-00106-а и в рамках выполнения госзадания, НИР № 115011560084.

МЕТОДЫ И АЛГОРИТМЫ АДАПТИВНОЙ МАРШРУТИЗАЦИИ В КОМПЬЮТЕРНЫХ СЕТЯХ

Одной из сложных научно-технических задач, возникающих при проектировании телекоммуникационных систем, является задача адаптивной маршрутизации в компьютерных сетях. В работах [10, 11] разработаны эффективные методы и алгоритмы адаптивной ускоренной маршрутизации, позволяющие снизить трудоемкость построения таблиц маршрутизации и не производить их полный перерасчет. Также авторами были получены новые результаты в области динамического

управления потоками данных в компьютерных сетях нового поколения – программно-конфигурируемые сети (ПКС). Отметим наиболее важные из них.

1. Разработан метод и алгоритм сегментации структур компьютерных сетей по критерию минимальной стоимости [12, 13].

2. Предложены эффективные алгоритмы и программные средства динамического управления потоками данных на основе метода маршрутизации по подсетям, обеспечивающие параллельную обработку информации в компьютерных сетях [14, 15].

3. Разработан концептуальный подход повышения быстродействия программно-конфигурируемых сетей за счет применения стратегии многопутевой маршрутизации и параллельной передачи потоков данных [16-19].

4. На основе данных о парных перестановках маршрутов предложен метод и алгоритм динамической балансировки потоков данных в программно-конфигурируемых сетях с обеспечением качества сетевых сервисов [20, 21].

5. Предложены эффективные методы и алгоритмы адаптивной ускоренной маршрутизации в программно-конфигурируемых сетях распределенных центров обработки данных нескольких провайдеров связи с различными зонами покрытия абонентов [22-26].

Получение данных результатов позволило повысить эффективность функционирования компьютерных сетей и обеспечить гибкость в управлении потоками данных. В рамках проведенных исследований создана визуальная программная среда автоматизации проектирования процессов динамического управления потоками данных в программно-конфигурируемых сетях с обеспечением качества сетевых сервисов.

Исследования по данному научному направлению были поддержаны грантами Президента РФ для молодых ученых – кандидатов наук МК-819.2014.9, МК-6016.2016.9, стипендией Президента РФ молодым ученым и аспирантам, осуществляющим перспективные научные исследования и разработки по приоритетным направлениям модернизации российской экономики (СП-505.2016.5), грантом Российского фонда фундаментальных исследований (РФФИ) и правительства Рязанской области № 16-47-620300 p_a.

ТЕХНОЛОГИИ РАЗРАБОТКИ МНОГОПОТОЧНЫХ ПРОГРАММ ДЛЯ СОВРЕМЕННЫХ МНОГОЯДЕРНЫХ СИСТЕМ

В настоящее время широкое распространение получили многоядерные процессоры и вычислительные системы на их основе, которые предоставляют большие потенциальные возможности для параллельных вычислений, но требуют значительных усилий при разработке прикладных программ. На сегодняшний день широко применяются процессоры с четырьмя и более ядрами, а для графических ускорителей количество ядер уже измеряется сотнями. Подобная технологическая революция в архитектуре компьютера предоставляет существенный потенциал для роста производительности программного обеспечения (ПО), однако реализация этого потенциала требует столь же серьезных революционных изменений современных технологий программирования.

Это обусловлено концепцией многопоточности вычислений, согласно которой в рамках одного вычислительного процесса могут порождаться параллельные потоки (threads), выполняемые в едином адресном пространстве процесса. Основные сложности организации многопоточных процессов связаны с использованием общих ресурсов, включая разделяемую память. При этом необходимо учитывать и другие архитектурные особенности, например, наличие кэш-памяти процессорных ядер, которая может быть как общей, так и распределенной для разных ядер.

Эффективность многопоточных программ во многом зависит от организации взаимодействия потоков, что требует их синхронизации при обмене данными и доступе к общим ресурсам. В этих условиях многопоточность распространяется, в первую очередь, на параллелизм уровня приложений, реализуемых последовательно в каждом потоке, что при возрастании числа ядер становится все менее эффективным.

Поэтому при создании масштабируемых высокопроизводительных приложений, работающих на многоядерном оборудовании, важной задачей является разработка такого подхода к параллелизму [27], при котором в программе целесообразно заранее предусмотреть последовательности команд (потоки), которые могли бы быть выполнены параллельно и независимо друг от друга. Очевидно, что полную независимость потоков в рамках одного приложения получить сложно [28-30]. Отсюда актуальной становится задача организации многопоточных вычислительных процессов на уровне операторов прог-

раммы, с целью уменьшения потенциальных взаимодействий между потоками.

В работе [31] предлагается алгоритм планирования параллельных вычислений в однородной вычислительной среде (на множестве процессорных ядер с общей памятью), направленный на минимизацию числа обменов данными между процессорными ядрами (потоками), что может повысить эффективность многопоточных приложений за счет уменьшения суммарного количества обращений к общей оперативной памяти, через которую производится обмен данными. Алгоритм основан на построении двумерной укладки графа потока данных по критерию времени при дополнительном условии сохранения числа соседней связности вершин исходного графа [32]. Такой подход в конечном итоге обеспечивает уменьшение числа обменов данными между процессорными ядрами.

Параллельные программы должны обладать необычными динамическими свойствами (масштабируемость на все доступные ресурсы, балансировка загрузки ядер, анализ зависимостей по данным и управлению и др.), без которых обходятся последовательные программы. В результате нынешнее общесистемное ПО и системы программирования оказались плохо приспособленными для создания параллельных программ, трудоёмкость их написания оказалась чрезвычайно велика.

Существуют задачи, которые сравнительно просто запрограммировать для исполнения на многоядерных процессорах. Однако, как показывает практика, не так много в мире разработано алгоритмов, пригодных для хорошей параллельной реализации. Все это требует разработки новых подходов к параллельному программированию, которые направлены на упрощение и повышение эффективности работы программиста.

В работе [28] предлагается оригинальное решение задачи создания новых инструментальных средств для автоматического распараллеливания существующих последовательных программ на основе метода спекулятивной многопоточности. В основе этого метода лежит идея параллельного многопоточного выполнения таких фрагментов исходной программы (регионов), информационные зависимости не удается однозначно определить в процессе трансляции. Такой подход использует позицию крайнего оптимизма в расчёте на то, что информационные зависимости не проявятся также и на стадии исполнения программы. Если подобные надежды оправдаются, то будет получен заметный выигрыш в производительности, в противном случае, результаты вычислений некоторого ре-

гиона должны быть аннулированы, и он будет выполнен повторно.

Массовое внедрение многоядерных технологий возродило интерес к функциональной парадигме программирования, а также к разработке новых языков и техник для обработки данных. В работах [28, 33] показано, что использование новых свойств современных функциональных языков программирования (например, языка Clojure) для многопоточной реализации последовательных алгоритмов обеспечивает такие преимущества перед императивной реализацией, как единый поток управления программой и неявная синхронизация обрабатываемых данных.

Примером практической реализации функционального подхода может служить разработка алгоритма и многопоточного приложения для решения задачи сортировки методом параллельного слияния, который демонстрирует идею декомпозиции данных в рамках известного подхода «разделяй и властвуй». Разработанный алгоритм включает два этапа: деление списка на подписки; параллельное слияние полученных подписков.

Первый этап не сильно отличается от классического варианта, где деление списка на подписки выполняется рекурсивно. В предлагаемой параллельной реализации первого этапа при рекурсивном спуске каждая итерация слияния выделяется в отдельную нить и ожидает выполнения. Как только в подписках останутся единственные элементы, начинается их слияние, т.е. второй этап алгоритма. На втором этапе происходит возврат из рекурсии, и каждая пара подписков сливается в отдельном потоке (thread). Тем самым загружаются все свободные ядра процессора и минимизируется время их простоя.

В программе на языке Clojure сортируемые данные представляются в виде связанного однонаправленного списка. Для обработки списка используется функция `map`, которая представляет собой функцию высшего порядка, т.е. она применяет некоторую функцию к каждому элементу списка и возвращает список результатов. В функциональных языках функция `map` часто называется «применить ко всем», что достаточно точно определяет ее сущность.

Практическая реализация алгоритмов параллельного и последовательного слияния подтверждает рассмотренные преимущества функционального подхода. В частности, анализ экспериментальных данных, полученных при сортировке больших объемов данных для процессора с 4 ядрами, показывает, что для размера списка от 10^3 до 10^4 среднее время многопоточной сорти-

ровки резко уменьшается (примерно в 6 раз), а для больших размеров списка (10^5 , 10^6 и более) ускорение составляет от 20 до 25 %.

В настоящее время также интенсивно развивается еще одно направление параллельного программирования – разработка многопоточных приложений для графических ускорителей, вычислительная мощность которых в данный момент нередко превышает возможности центральных процессоров. Поэтому возник интерес к использованию графических ускорителей для решения задач, не связанных напрямую с обработкой графики.

Использование графических ускорителей в качестве средств высокопроизводительных вычислений поддерживается усилиями ведущих разработчиков аппаратуры. Так, компания nVidia предоставляет платформу CUDA (Compute Unified Device Architecture) для вычислений на графическом ускорителе. Аналогично компания AMD выступила с инициативой Stream.

Такие платформы облегчают разработку программ для решения различных задач на графических ускорителях, поскольку поддерживают модель программирования, которая более приспособлена к реализации общецелевых вычислений по сравнению со средствами программирования графики. Тем не менее, разработка приложений для графических ускорителей остается достаточно сложной задачей. Разработчик должен быть знаком с устройством графического ускорителя, он должен понимать особенности работы его компонент и принципы организации вычислений, включая взаимодействие с центральным процессором.

В работах [29, 30, 34] развивается направление, связанное с разработкой и многопоточной реализацией алгоритмов решения задач обработки данных средствами графических ускорителей на платформе CUDA. Полученные результаты показывают, что использование графического ускорителя помогает значительно увеличить скорость работы многопоточных программ, выполняющих массовую обработку слабосвязанных данных. При необходимости распараллеливания более сложных с точки зрения логической организации алгоритмов, целесообразно использовать технологии многопоточного программирования для многоядерных центральных процессоров. В качестве прикладных задач, допускающих параллельные вычисления, для дальнейших исследований могут быть выбраны задачи организации самодиагностики специализированных многопроцессорных систем, разработки и тестирования их программного обеспечения [35-37].

Библиографический список

1. **Pritsker A. A.** GERT Graphical evaluation and review technique. Memorandum RM-4973-NASA. 1966. 138 p.
2. **Shibanov A. P.** Finding the distribution density of the time taken to fulfill the GERT network on the basis of equivalent simplifying transformation. Automation and Remote Control. Plenum Press New York, NY, USA. February 2003. Vol. 64. Issue 2. P. 279-287.
3. **Shibanov A. P.** A Software Implementation Technique for Simulation of Ethernet Local Area Networks. Programming and Computing Software. Plenum Press New York, NY, USA. Volume 28 Issue 6. November-December 2002. P. 349-355.
4. **Шибанов А. П.** Метод эквивалентных упрощающих преобразований GERT-сетей и его приложения // Вестник Рязанского государственного радиотехнического университета. 2012. № 39-2. С. 76-83.
5. **Шибанов А. П., Шибанов В. А.** Вычисление распределения дискретной выходной случайной величины GERT-сети // Системы управления и информационные технологии. 2016. Т. 63. № 1. С. 86-91.
6. **Шибанов А. П., Шибанов В. А.** Вычисление распределения выходной величины GERT-сети с экспоненциальными и равномерно распределенными случайными величинами // Приборы и системы. Управление, контроль, диагностика. 2015. № 7. С. 1-9.
7. **Корячко В. П., Шибанов А. П., Сапрыкин А. Н.** Расчет вариации времени передачи пакетов в сети с двухфазной маршрутизацией // XII Всероссийское совещание по проблемам управления ВСПУ-2014. Москва, 16-19 июня 2014 г.
8. **Корячко В. П., Шибанов А. П., Сапрыкин А. Н., Фам Х. Л.** Планирование полосы пропускания сетевых каналов полигонного измерительного комплекса // Радиотехника. 2016. № 11. С. 103-109.
9. **Корячко В. П., Шибанов А. П., Кравчук Н. В., Шибанов В. А.** Метод расстановки контрольных точек в программах микроконтроллеров со сторожевыми таймерами // Приборы и системы. Управление, контроль, диагностика. 2008. № 11. С. 44-49.
10. **Корячко В. П., Перепелкин Д. А.** Анализ и проектирование маршрутов передачи данных в корпоративных сетях. М.: Горячая линия – Телеком, 2012. 236 с.
11. **Перепелкин Д. А.** Динамическое формирование структуры и параметров линий связи корпоративной сети на основе данных о парных перестановках маршрутов // Информационные технологии. 2014. № 4. С. 52-60.
12. **Перепелкин Д. А.** Алгоритм формирования оптимальных структур сегментов корпоративных сетей с учетом данных о структуре базовой сети и связности каналов связи // Вестник Рязанского государственного радиотехнического университета. 2014. № 50-1. С. 59-64.
13. **Перепелкин Д. А., Цыганов И. Ю.** Усовершенствованный алгоритм сегментации структур корпоративных сетей по критерию минимальной стоимости // Вестник Рязанского государственного радиотехнического университета. 2015. № 53. С. 48-58.
14. **Перепелкин Д. А.** Динамическое формирование трафика корпоративных сетей на основе метода маршрутизации по подсетям // Вестник Рязанского государственного радиотехнического университета. 2015. № 51. С. 35-41.
15. **Перепелкин Д. А., Цыганов И. Ю.** Алгоритм парных переходов в компьютерных сетях на основе метода маршрутизации по подсетям // Вестник Рязанского государственного радиотехнического университета. 2016. № 57. С. 56-62. DOI: 10.21667/1995-4565-2016-57-3-56-62.
16. **Корячко В. П., Перепелкин, Д. А.** Разработка и исследование математической модели многопутевой адаптивной маршрутизации в сетях связи с балансировкой нагрузки // Электросвязь. 2014. № 12. С. 27-31.
17. **Перепелкин Д. А.** Концептуальный подход динамического формирования трафика программно-конфигурируемых телекоммуникационных сетей с балансировкой нагрузки // Информационные технологии. 2015. Том 21. № 8. С. 602-610.
18. **Перепелкин Д. А.** Математическая модель многопутевой адаптивной маршрутизации с балансировкой неоднородной нагрузки в условиях динамических подключений узлов и линий связи в телекоммуникационных сетях // Радиотехника. 2015. № 5. С. 46-54.
19. **Перепелкин Д. А.** Модель отказоустойчивой многопутевой адаптивной маршрутизации с балансировкой неоднородной нагрузки в сетях связи // Радиотехника. 2015. № 11. С. 40-47.
20. **Перепелкин Д. А., Бышов В. С.** Балансировка потоков данных в программно-конфигурируемых сетях с обеспечением качества обслуживания сетевых сервисов // Радиотехника. 2016. № 11. С. 111-119.
21. **Koryachko V. P., Perepelkin D. A., Byshov V. S.**, «Development and Research of Improved Model of Multipath Adaptive Routing in Computer Networks with Load Balancing». Automatic Control and Computer Sciences, 2017, vol. 51, no. 1, pp. 63-73. DOI: 10.3103/S0146411617010047.
22. **Корячко В. П., Перепелкин Д. А., Иванчикова М. А.** Алгоритм адаптивной маршрутизации в корпоративных сетях нескольких провайдеров связи // Вестник Рязанского государственного радиотехнического университета. 2013. № 2 (44). С. 52-56.
23. **Корячко В. П., Перепелкин Д. А., Иванчикова М. А.** Алгоритм парных переходов каналов связи при динамическом изменении нагрузки в корпоративных сетях нескольких провайдеров связи с различными зонами покрытия // Вестник Рязанского государственного радиотехнического университета. 2014. № 48. С. 68-76.
24. **Перепелкин Д. А., Иванчикова М. А.** Разработка программного обеспечения моделирования процессов адаптивной маршрутизации в корпоративных сетях нескольких провайдеров связи с различными зонами покрытия абонентов // Программная инженерия. 2015. № 11. С. 34-40.
25. **Perepelkin D. A., Ivanchikova M. A.**, «Fast ReRouting Traffic Algorithm between Data Centers». Proceedings ELEKTRO 2016 – IEEE 11th International

Conference, 2016, pp. 68-71. DOI: 10.1109/ELEKTRO.2016.7512037.

26. **Корячко В. П., Перепелкин Д. А., Иванчикова М. А.** Разработка и исследование алгоритма быстрой перемаршрутизации трафика между центрами обработки данных // Радиотехника. 2016. № 8. С. 133-139.

27. **Гергель В. П.** Высокопроизводительные вычисления для многопроцессорных многоядерных систем. М.: Изд-во МГУ, 2010. 544 с.

28. **Бакулев А. В., Бакулева М. А., Козлов М. А., Скворцов С. В.** Технологии разработки параллельных программ для современных многоядерных процессоров // Экономика, статистика и информатика. Вестник УМО. 2014. № 6. С. 211-215.

29. **Лунин Д. В., Скворцов С. В.** Организация параллельных вычислений на платформе CUDA // Вестник Рязанского государственного радиотехнического университета. 2014. № 49. С. 77-82.

30. **Лунин Д. В., Скворцов С. В.** Разработка параллельного генетического алгоритма для решения задачи коммивояжера на платформе CUDA // Системы управления и информационные технологии. 2015. Т. 60. № 2. С. 50-55.

31. **Скворцов С. В.** Алгоритм планирования параллельных вычислений в многоядерных процессорах // Радиотехника. 2016. № 8. С. 153-159.

32. **Корячко В. П.** Конструирование микропроцессорных систем контроля радиоэлектронной аппаратуры. М.: Радио и связь, 1987. 160 с.

33. **Козлов М. А., Скворцов С. В.** Алгоритмы параллельной сортировки данных и их реализация на языке Clojure // Вестник Рязанского государственного радиотехнического университета. 2013. № 4-1 (46). С. 92-96.

34. **Скворцов С. В., Пюрова Т. А.** Параллельные алгоритмы сортировки данных и их реализация на платформе CUDA // Вестник Рязанского государственного радиотехнического университета. 2016. № 58. С. 42-48.

35. **Скворцов Н. В., Скворцов С. В., Хрюкин В. И.** Синтез диагностических графов для многопроцессорных систем с активной отказоустойчивостью // Вестник Рязанского государственного радиотехнического университета. 2012. № 39-2. С. 83-89.

36. **Рудаков В. Е., Скворцов С. В.** Построение базового множества независимых путей потокового графа для тестирования программных модулей // Системы управления и информационные технологии. 2012. Т. 50. № 4. С. 67-70.

37. **Скворцов Н. В., Скворцов С. В.** Автоматизация проектирования процессов самодиагностики для многопроцессорных систем с активной отказоустойчивостью // Вестник Рязанского государственного радиотехнического университета. 2013. № 4-2 (46). С. 71-77.

UDC 004.72

METHODS AND TECHNOLOGIES OF DESIGN AUTOMATION OF HIGH PERFORMANCE SYSTEMS AND COMPUTER NETWORKS

V. P. Koryachko, PhD (technical sciences), full professor, head of CAD department, RSREU, Ryazan;

S. V. Skvortsov, PhD (technical sciences), full professor, CAD department, RSREU, Ryazan; s.v.skvor@gmail.com

A. P. Shibanov, PhD (technical sciences), full professor, CAD department, RSREU, Ryazan; apshibanov@yandex.ru

D. A. Perepelkin, PhD (technical sciences), associate professor, CAD department, RSREU, Ryazan; dmitryperpelkin@mail.ru

The main results obtained by the staff of the CAD department of RSREU in the field of design automation high performance systems and computer networks are considered.

Key words: GERT-networks, equivalent transformations, analytical functions, software defined networks, adaptive routing, multipath routing, load balancing, multi-threaded programs, multi-core processors, graphics accelerator.

DOI: 10.21667/1995-4565-2017-60-2-94-104

References

1. **Pritsker A. A.** GERT Graphical evaluation and review technique. Memorandum RM-4973-NASA. 1966. 138 p.

2. **Shibanov A. P.** Finding the distribution density of the time taken to fulfill the GERT network on the basis of equivalent simplifying transformation. Automation and Remote Control. Plenum Press New York, NY, USA. February 2003. Vol. 64. Issue 2. P. 279-287.

3. **Shibanov A. P.** A Software Implementation Technique for Simulation of Ethernet Local Area Networks. Programming and Computing Software. Plenum Press New York, NY, USA. Volume 28 Issue 6. November-December 2002. P. 349-355.

4. **Shibanov A. P.** Metod jekvivalentnyh upro-shhajushih preobrazovanij GERT-setej i ego prilozhenija. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2012, No. 39-2, pp. 76-83 (in Russian).

5. **Shibanov A. P., Shibanov V. A.** Vychislenie raspredelenija diskretnoj vyhodnoj sluchajnoj velichiny GERT-seti. Sistemy upravlenija i informacionnye tehnologii. 2016, Vol. 63, No. 1, pp. 86-91 (in Russian).

6. **Shibanov A. P., Shibanov V. A.** Vychislenie raspredelenija vyhodnoj velichiny GERT-seti s jeks-ponecial'nymi i ravnomerno raspredelennymi sluchajnymi velichinami. Pribory i sistemy. Upravlenie, kontrol', diagnostika. 2015, No. 7, pp. 1-9 (in Russian).

7. **Koryachko V. P., Shibanov A. P., Saprykin A. N.** Raschet variacii vremeni peredachi paketov v seti s dvuhfaznoj marshrutizaciej. XII Vserossijskoe soveshanie po problemam upravlenija (VSPU-2014). Moskva, 16-19 ijunja 2014 g. (in Russian).

8. **Koryachko V. P., Shibanov A. P., Saprykin A. N., Fam H. L.** Planirovanie polosy propuskanija setevyh kanalov poligonogo izmeritel'nogo kompleksa. Radiotekhnika. 2016, No. 11, pp. 103-109 (in Russian).

9. **Koryachko V. P., Shibanov A. P., Kravchuk N. V., Shibanov V. A.** Metod rasstanovki kontrol'nyh to-check v programmah mikrokontrollerov so storozhevymi tajmerami. Pribory i sistemy. Upravlenie, kontrol', diagnostika. 2008, No. 11, pp. 44-49 (in Russian).

10. **Koryachko V. P., Perepelkin D. A.** Analiz i proektirovanie marshrutov peredachi dannyh v korporativnyh setjah (Analysis and design of data transmission routes in corporate networks), Moscow, Hot-line Telecom, 2012, 236 p. (in Russian).

11. **Perepelkin D. A.** Dinamicheskoe formirovanie struktury i parametrov linij svjazi korporativnoj seti na osnove dannyh o parnyh perestanovkah marshrutov. Informacionnye tehnologii. 2014, No. 4, pp. 52-60 (in Russian).

12. **Perepelkin D. A.** Algoritm formirovanija optimal'nyh struktur segmentov korporativnyh setej s uchetom dannyh o strukture bazovoj seti i svjaznosti kanalov svjazi. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2014, No. 50-1, pp. 59-64 (in Russian).

13. **Perepelkin D. A., Tsyganov I. Yu.** Uovershenstvovannyj algoritm segmentacii struktur korporativnyh setej po kriteriju minimal'noj stoimosti. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2015, No. 53, pp. 48-57 (in Russian).

14. **Perepelkin D. A.** Dinamicheskoe formirovanie trafika korporativnyh setej na osnove metoda marshrutizacii po podsetjam. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2015, No. 51, pp. 35-41 (in Russian).

15. **Perepelkin D. A., Tsyganov I. Yu.** Algoritm parnyh perehodov v komp'juternyh setjah na osnove metoda marshrutizacii po podsetjam. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2016, No. 57, pp. 56-62. DOI: 10.21667/1995-4565-2016-57-3-56-62.

16. **Koryachko V. P., Perepelkin D. A.** Razrabotka i issledovanie matematicheskoj modeli mnogoputevoj adaptivnoj marshrutizacii v setjah svjazi s balansirovkoj nagruzki. Jelektrosvjaz'. 2014, No. 12, pp. 27-31 (in Russian).

17. **Perepelkin D. A.** Konceptual'nyj podhod dinamicheskogo formirovanija trafika programmno-konfi-

guriruemyh telekommunikacionnyh setej s balansirovkoj nagruzki. Informacionnye tehnologii. 2015, Vol. 21, No. 8, pp. 602-610 (in Russian).

18. **Perepelkin D. A.** Matematicheskaja model' mnogoputevoj adaptivnoj marshrutizacii s balansirovkoj neodnorodnoj nagruzki v uslovijah dinamicheskikh podkljuchenij uzlov i linij svjazi v telekommunikacionnyh setjah. Radiotekhnika. 2015, No. 5, pp. 46-54 (in Russian).

19. **Perepelkin D. A.** Model' otkazoustojchivoj mnogoputevoj adaptivnoj marshrutizacii s balansirovkoj neodnorodnoj nagruzki v setjah svjazi. Radiotekhnika. 2015, No. 11, pp. 40-47 (in Russian).

20. **Perepelkin D. A., Byshov V. S.** Balansirovka potokov dannyh v programmno-konfiguriruemyh setjah s obespecheniem kachestva obsluzhivanija setevyh servisov. Radiotekhnika. 2016, No. 11, pp. 111-119 (in Russian).

21. **Koryachko V. P., Perepelkin D. A., Byshov V. S.** «Development and Research of Improved Model of Multipath Adaptive Routing in Computer Networks with Load Balancing». Automatic Control and Computer Sciences, 2017, Vol. 51, No. 1, pp. 63-73. DOI: 10.3103/S0146411617010047.

22. **Koryachko V. P., Perepelkin D. A., Ivanchikova M. A.** Algoritm adaptivnoj marshrutizacii v korporativnyh setjah neskol'kih provajderov svjazi. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2013, No. 2 (44), pp. 52-56 (in Russian).

23. **Koryachko V. P., Perepelkin D. A., Ivanchikova M. A.** Algoritm parnyh perehodov kanalov svjazi pri dinamicheskom izmenenii nagruzki v korporativnyh setjah neskol'kih provajderov svjazi s razlichnymi zonami pokrytija. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2014, No. 48, pp. 68-76 (in Russian).

24. **Perepelkin D. A., Ivanchikova M. A.** Razrabotka programmnogo obespechenija modelirovanija processov adaptivnoj marshrutizacii v korporativnyh setjah neskol'kih provajderov svjazi s razlichnymi zonami pokrytija abonentov. Programmaja inzhenerija. 2015, No. 11, pp. 34-40 (in Russian).

25. **Perepelkin D. A., Ivanchikova M. A.** «Fast ReRouting Traffic Algorithm between Data Centers». Proceedings ELEKTRO 2016 – IEEE 11th International Conference, 2016, pp. 68-71. DOI: 10.1109/ELEKTRO.2016.7512037.

26. **Koryachko V. P., Perepelkin D. A., Ivanchikova M. A.** Razrabotka i issledovanie algoritma bystroj peremarsrutizacii trafika mezhdru centrami obrabotki dannyh. Radiotekhnika. 2016, No. 8, pp. 133-139 (in Russian).

27. **Gergel' V. P.** Vysokoproizvoditel'nye vychislenija dlja mnogoprocessornyh mnogojadernyh sistem (High-performance computing for multi-core systems). Moscow, Izdatel'stvo Moskovskogo universiteta, 2010. 544 p. (in Russian).

28. **Bakulev A. V., Bakuleva M. A., Kozlov M. A., Skvorcov S. V.** Tehnologii razrabotki paralel'nyh program dlja sovremennyh mnogojadernyh processorov. Jekonomika, statistika i informatika. Vestnik UMO. 2014, No. 6, pp. 211-215 (in Russian).

29. **Lunin D. V., Skvortsov S. V.** Organizacija paralel'nyh vychislenij na platforme CUDA. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2014, No. 49, pp. 77-82 (in Russian).

30. **Lunin D. V., Skvortsov S. V.** Razrabotka paralelnogo geneticheskogo algoritma dlja reshenija zadachi kommivojazhera na platforme CUDA. Sistemy upravlenija i informacionnye tehnologii. 2015, Vol. 60, No. 2, pp. 50-55 (in Russian).

31. **Skvortsov S. V.** Algoritm planirovanija paralel'nyh vychislenij v mnogojadernyh processorah. Radio-tehnika. 2016, No. 8, pp. 153-159 (in Russian).

32. **Koryachko V. P.** Konstruirovanie mikroprocessornyh sistem kontrolja radioelektronnoj apparatury (Design of microprocessor control systems for radio electronic equipment). Moscow, Radio i svjaz', 1987, 160 p. (in Russian).

33. **Kozlov M. A., Skvortsov S. V.** Algoritmy paralel'noj sortirovki dannyh i ih realizacija na jazyke Clojure. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2013, No. 4-1 (46), pp. 92-96 (in Russian).

34. **Skvortsov S. V., Pyurova T. A.** Parallel'nye algoritmy sortirovki dannyh i ih realizacija na platforme CUDA. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2016, No. 58, pp. 42-48 (in Russian).

35. **Skvortsov N. V., Skvortsov S. V., Khryukin V. I.** Sintez diagnosticheskikh grafov dlja mnogoprocessornyh sistem s aktivnoj otkazoustojchivost'ju. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2012, No. 39-2, pp. 83-89 (in Russian).

36. **Rudakov V. E., Skvortsov S. V.** Postroenie bazovogo mnozhestva nezavisimyh putej potokovogo grafa dlja testirovanija programmnyh module. Sistemy upravlenija i informacionnye tehnologii. 2012, Vol. 50, No. 4, pp. 67-70 (in Russian).

37. **Skvortsov N. V., Skvortsov S. V.** Avtomatizacija proektirovanija processov samodiagnostiki dlja mnogoprocessornyh sistem s aktivnoj otkazo-ustojchivost'ju. Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2013, No. 4-2 (46), pp. 71-77 (in Russian).