

СОДЕРЖАНИЕ

Корячко В.П., Перепелкин Д.А. Математическая модель управления трафиком в корпоративных сетях	6
Андреев В.Г., Коробков М.А. Метод спектрального сверхразрешения, основанный на модификации собственных значений корреляционной матрицы	12
Арефьева М.И., Скворцов С.В. Программные средства для оптимизации диагностических графовых моделей многопроцессорных систем	16
Бакулев А.В., Бакулева М.А. О некоторых свойствах метода спекулятивной многопоточности	20
Варнавский А.Н., Чекан Н.В. Динамическое определение коэффициента вариации участков биоэлектрических сигналов	25
Ветров А.А., Кочергин А.М. Организация параллельных вычислений в Web-геопортале спутниковых изображений	28
Виноградова М.Е. Итеративная процедура снижения размерности пространства признаков	36
Воронков А.А., Перепелкин Д.А. Поиск кратчайших путей в корпоративных сетях нескольких провайдеров	40
Генеральский И.И., Генеральский И.М. Вариантная методология формирования производственной программы в условиях постоянных изменений	46
Гильман Д.В., Таганов Р.А. Алгоритмы преобразования нечетких графовых моделей процессных рисков проекта	48
Егоров Е.А. Классификация подходов к семантическому поиску	53
Еремеев В.В., Макаренков А.А., Юдаков А.А. Новые технологии автоматического выделения границ объектов на гиперспектральных изображениях земной поверхности	56

Жильников А.А., Жильников Т.А. Способ неразрушающего объемного измерения векторной функции магнитной индукции неоднородно распределенного в пространстве и периодически изменяющегося во времени магнитного поля.....	61
Иванчикова М.А., Перепелкин Д.А. Оптимизация корпоративной сети по критерию минимальной стоимости	69
Кистенёв П.И. Проектирование устройств на базе микроконтроллеров с использованием информации с GPS датчиков и передачей данных по GSM каналу.....	71
Кожанова Т.В. Энергосберегающий аналог 100 Вт лампы накаливания.....	73
Косоруков С.А. Алгоритм определения качества расположения реперов на кожухе реперного узла	76
Курасбедиани Д.Т. Проблемы информационного обеспечения в банковской деятельности	81
Маторин А.В., Балюк С.А. Анализ взаимодействия излучателей в адаптивной антенной решётке.....	85
Мирохина И.Н. Нечеткое когнитивное моделирование для анализа рисков в проектно-производственной системы	92
Мирошин В.М., Перепелкин Д.А. Анализ алгоритмов сегментации структур компьютерных сетей	96
Овечкин Г.В., Овечкин П.В. Применение не двоичных многопороговых декодеров в каналах связи с группирующимися ошибками	106
Перепелкин Д.А. Применение алгоритма парных переходов в протоколе OSPF	109
Петухова В.О., Лобовикова Е.Л. Вариант энергосберегающей «умной» лампы – аналог люминесцентной лампы освещения.....	116
Подгорнова Н.А. Разработка комплексной системы краткосрочного прогнозирования денежных потоков предприятия с использованием методов нелинейной динамики	118

Ревуцкий В.А. Алгоритм обнаружения и оценки параметров сверточных помехоустойчивых кодов	128
Скворцов С.В., Козлов М.А. Программная реализация алгоритма сортировки для многоядерных процессоров	131
Скворцов Н.В., Скворцов С.В., Хрюкин В.И. Разработка программных средств теоретико-вероятностных расчетов в единой системе допусков и посадок	134
Скрипкина О.В., Гусейнова Ф.Н. Использование информационных технологий при сдаче бухгалтерской (финансовой) отчетности в контролирующие органы	137
Солдатов В.В., Солдатова О.В. Мониторинг критически важных и/или потенциально опасных объектов, смещений (деформаций) и частоты колебаний критических точек конструкций зданий, сооружений на основе космических снимков	141
Таганов А.И., Гильман Д.В. Аспекты нечеткой идентификации и анализа процессных рисков по стадиям жизненного цикла проекта ..	143
Таганов А.И., Гильман Д.В. Формализация задач мониторинга процессных рисков по стадиям жизненного цикла проекта	147
Таганов А.И., Гильман Д.В. Задачи и алгоритмы структурной оптимизации процессов проекта по критериям риска.....	153
Товпеко А.В. Особенности управления проектом системы анализа телеизмерений.....	158
Фролова Н.С., Скворцов С.В. Система анализа и отображения параметров технологического процесса защиты на электростанции.....	169
Цветков И.А. Исследование функций альтернативного одиночного выбора.....	172
Цырульникова Е.А. Неразрушающий контроль трубопровода	178

В.П. КОРЯЧКО, Д.А. ПЕРЕПЕЛКИН

Рязанский государственный радиотехнический университет

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ УПРАВЛЕНИЯ ТРАФИКОМ В КОРПОРАТИВНЫХ СЕТЯХ

Предложена математическая модель управления трафиком в корпоративных сетях.

В современных корпоративных сетях широкое внимание уделяют вопросам управления сетевого трафика. При этом под управлением понимают возможность контроля и изменения состояния основных элементов сети, выявления и разрешения проблем, возникающих при работе сети, выполнения анализа производительности и планирования развития сети, а также осуществления текущего мониторинга. Все эти вопросы и проблемы требуют быстрого решения. В связи с этим необходимы актуальные математические модели управления сетевым трафиком.

Представим корпоративную сеть в виде ориентированного взвешенного связного графа $G = (V, E, W)$, где V – множество вершин, $|V| = N$, E – множество ребер, $|E| = M$, W – множество весов ребер, показанного на рис. 1. Направление возможной передачи информации указано стрелками. Возле каждой дуги приведено число – значение метрики этой дуги, например, задержка передачи по линии связи. Разрешается произвольно двигаться в направлении стрелок, причем метрика пути определяется как сумма значений метрик всех пройденных дуг. Среди путей, выходящих из точки x и оканчивающихся в точке y , требуется выбрать путь с минимальным значением метрики, который назовем оптимальным.

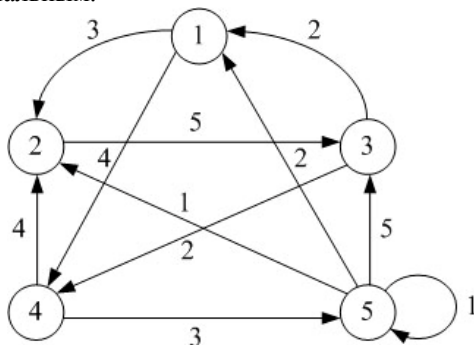


Рис. 1. Граф G корпоративной сети

При выборе оптимального пути на каждом шаге надо учитывать не только в какой точке мы находимся, но и сколько шагов еще остается сделать. Поэтому целесообразно заменить граф корпоративной сети, представленный на рис. 1 схемой, изображенной на рис. 2. На этой схеме 5 столбцов изображают 5 узлов корпоративной сети, представленной на рис. 1 в моменты времени $i=0,1, 2, 3, 4$. Дуги, ведущие из столбца X_{t-1} в столбец X_t , показывают переходы, возможные на t -ом шаге. Если выбрать любой путь из четырех звеньев на рис. 1, то соответствующая цепочка дуг на рис. 2 построит график движения.

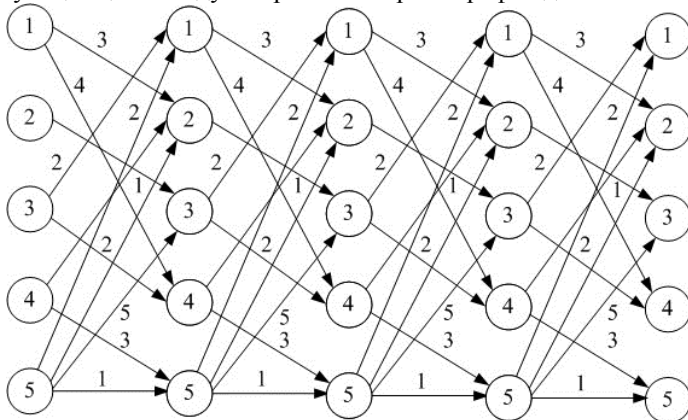


Рис. 2. Схема процесса управления

Схемы, подобные представленной на рис. 2, позволяют задавать системы, меняющиеся во времени. На рис. 3 приведен пример подобной схемы.

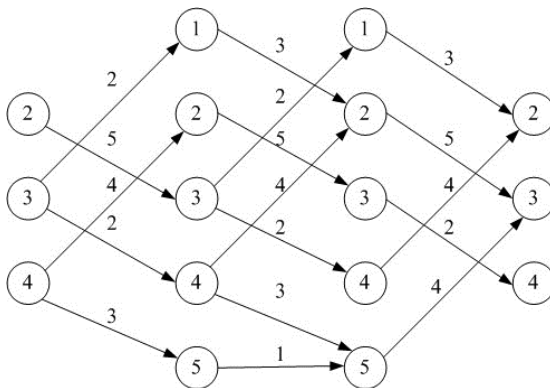


Рис. 3. Изменение процесса во времени

В общем случае, на рис. 4 имеем конечные множества точек X_m, X_{m+1}, \dots, X_n (столбцы) и конечные множества дуг A_{m+1}, \dots, A_n (дуги множества A_i ведут из X_{i-1} в X_i). Точки множества X_n называются финальными. Из каждой не финальной точки выходит, по крайней мере, одна дуга. Последовательность дуг образует путь, если начало каждой из них (кроме первой) совпадает с концом предыдущей дуги и последняя оканчивается в X_n . На множестве всех дуг задана функция q . Сумма значений этой функции на всех дугах пути называется метрикой этого пути. Среди путей, выходящих из заданной точки x , требуется выбрать путь с наименьшим значением метрики, т. е. оптимальный путь [1].

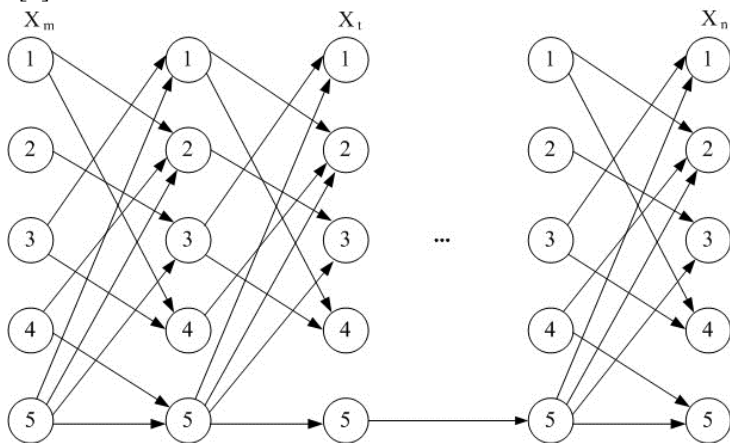


Рис. 4. Общая схема управления

Значение метрики оптимального пути, выходящего из точки x , называется оценкой x и обозначается $v(x)$. Для того, чтобы функция $v(x)$ была определена на множестве X всех точек, положим ее равной 0 на X_n .

Задача, будет решена, если перебрать все пути, выходящие из x , и сравнить их оценки. Но этот метод редко применим, ввиду огромного количества вариантов даже в не очень сложных схемах. Предположим, что функция $v(x)$ известна. Тогда задача легко решается с помощью следующего критерия: для оптимальности пути l необходимо и достаточно, чтобы для любой дуги a , принадлежащей l , выполнялось равенство:

$$q(a) = v(x) - v(y), \quad (1)$$

где x – начало дуги a , y – ее конец.

В самом деле, складывая равенства (1) по всем дугам пути l , получим, что оценка l равна значению v в начальной точке этого пути. В то же время для любой дуги a и ее начальной и конечной точек y и x выполняется неравенство:

$$v(x) \leq q(a) + v(y), \quad (2)$$

т. к. правая часть равна оценке пути, состоящего из дуги a и оптимального пути, выходящего из y . Если хотя бы для одной дуги пути l в формуле (2) имеет место строгое неравенство, то, складывая (2) по всем дугам пути l , получим, что оценка l больше оценки v в начальной точке, и, следовательно, путь l не оптимален. Заметим далее, что в не финальной точке x :

$$v(x) = \min_{a \in A(x)} [q(a) + v(y)], \quad (3)$$

где y – конец дуги a , $A(x)$ – пучок дуг, выходящих из x . Действительно, для любой дуги a из $A(x)$ выполняется неравенство (2), а для дуги a , являющейся началом оптимального пути из точки x , оно заменяется равенством (1). Формула (3) выражает значение оценки v на шаге X_{i-1} через ее значение на шаге X_i и позволяет вычислять v рекуррентно, двигаясь, справа налево.

Оценка пути складывается из значений метрик входящих в него дуг. Можно рассмотреть обобщенную задачу, когда в оценку пути вносят вклад не только дуги, но и точки пути. Пусть $I = x_m a_{m+1} x_{m+1} \dots a_n x_n$ – путь, состоящий из последовательно проходимых дуг a_{m+1}, \dots, a_n ($a_i \in A_i$) и точек x_m, x_{m+1}, \dots, x_n ($x_i \in X_i$), причем началом дуги a_i служит точка x_i , концом – точка x_i . Определим оценку I пути l формулой:

$$I(l) = r(x_m) + q(a_{m+1}) + r(x_{m+1}) + \dots + q(a_n) + r(x_n), \quad (4)$$

где r – функция на точках, q – функция на дугах. Если положить

$$\tilde{q}(a) = r(x) + q(a),$$

где x – начало дуги a , то оценка (4) запишется в виде:

$$\sum_{t=m+1}^n \tilde{q}(a_t) + r(x_n)$$

Поэтому мы ничего не потеряем в общности, считая, что r равно 0 на не финальных точках. Тогда оценкой пути l будет

$$I(l) = \sum_{t=m+1}^n q(a_t) + r(x_n)$$

Функции q и r называются текущей и финальной платами.

Критерий оптимальности пути и формула (3) сохраняют силу и при наличии финальной платы r . Единственное различие состоит в том, что теперь оценка v равна на X_n не 0, а r .

Предположим, что выбор дуги в точке x определяет не состояние y , а лишь распределение вероятностей для этого состояния [2, 3]. Как и прежде, необходимо найти путь с минимальной оценкой. Однако теперь проходимый путь зависит не только от нашего выбора, но и от случайной величины, и мы хотим минимизировать математическое ожидание оценки.

Пусть X_t ($t=m, m+1, \dots, n$) и A_t ($t=m+1, \dots, n$) – произвольные конечные множества. Каждому a из A_t сопоставлено распределение вероятностей $p(\bullet|x)$ на A_t . Задать распределение вероятностей p на конечном (или счетном) множестве E – значит отнести каждому $a \in A$ неотрицательное число $p(a)$ так, чтобы сумма этих чисел была равна 1. Для каждого $\Gamma \subseteq E$ $p(\Gamma)$ обозначает сумму $p(a)$ по всем a из Γ . Если для подмножества Γ множества E выполняется условие $p(\Gamma)=1$, то считается, что распределение p сосредоточено на Γ . Обозначение $p(\bullet|x)$ вместо $p(a|x)$ применяется, чтобы отличить закон, задающий распределение, от числа $p(a|x)$, отвечающего конкретному a .

Функцию p , определяющую закон перехода из X_t в A_t будем называть переходной функцией. Естественно считаем, что точка множества X_m , из которой начинается путь, также случайна, и дано ее распределение вероятностей μ (начальное распределение).

Переход из $x \in X_{t-1}$ в A_t определяется выбором. При этом мы выбираем a не из всего A_t , а из его подмножества $A(x)$, зависящего от состояния x . Элементы множества $A(x)$ назовем управлениями в точке x . Множества $A(x)$ определены (и не пусты) для всех не финальных состояний x . Предполагается, что $A(x)$ попарно не пересекаются и их сумма по всем x из X_{t-1} равна A_t . Другими словами, каждое управление a может быть использовано в одном и только одном состоянии. Это состояние обозначим $j(a)$, так что запись $x=j(a)$ равносильна записи $a \in A(x)$. Отображение j называется проекцией, а множества $A(x)=j^{-1}(x)$ – слоями.

На множестве всех управлений задана текущая плата $q(a)$, на множестве финальных состояний – финальная плата $r(x)$.

Таким образом, мы приходим к понятию управляемого марковского процесса на промежутке времени $[m, n]$. Этот процесс задают следующие элементы:

- а) множества X_m, X_{m+1}, \dots, X_n (пространства состояний);
- б) множества A_{m+1}, \dots, A_n (пространства управлений);
- в) отображение j множества управлений

$$A = \bigcup_{t=m+1}^n A_t \quad \text{во множество состояний} \quad X = \bigcup_{t=m}^n X_t$$

такое, что $j(A_t) = X_{t-1}$ (проекция);

г) распределения вероятностей $p(\bullet|x)$ на A_t , зависящие от $x \in X_t$ ($t=m, m+1, \dots, n$) (переходная функция);

д) функция q на множестве A управлений (текущая плата);

е) функция r на множестве X_n финальных состояний (финальная плата);

ж) распределение вероятностей μ на X_m (начальное распределение).

При изучении управляемых процессов полезно сокращать отрезок $[m, n]$ до меньшего отрезка $[m_1, n]$. Для сокращенного процесса элементы, перечисленные в пунктах а) .. е), однозначно определяются по первоначальному процессу. Напротив, начальное распределение (в момент m_1) зависит от способа управления на промежутке времени $[m, m_1]$. Объект, который определяется элементами а) .. ж) называется моделью. Если задана модель Z , то каждому распределению μ на X_m соответствует однозначно определенный управляемый процесс Z , для которого μ – начальное распределение (в случае, когда μ сосредоточено в точке x , вместо Z_μ обозначим Z_x).

В итоге получаем задачу определения способа управления, при котором минимально математическое ожидание оценки:

$$\min_l I(l) = \sum_{t=m+1}^n q(a_t) + r(x_n)$$

пути $l = x_m a_{m+1} x_{m+1} \dots a_n x_n$; $x_t \in X_t$, $m \leq t \leq n$; $a_t \in A(x_{t-1}) \subset A_t$.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Дынкин Е.Б., Юшкевич А.А. Управляемые марковские процессы и их приложения. – М.: Наука. 1975. – 338 с.
2. Кемени Дж. и др. Счетные цепи Маркова / Пер. с англ. Зубкова А.М., Севастьянова Б.А. – М.: Наука. 1987. – 413 с.
3. Миллер Б.М., Панков А.Р. Теория случайных процессов в примерах и задачах / Под ред. Кибзуна А.И. – М.: Физматлит. 2002. – 320 с.

В.Г. АНДРЕЕВ, М.А. КОРОБКОВ

Рязанский государственный радиотехнический университет

МЕТОД СПЕКТРАЛЬНОГО СВЕРХРАЗРЕШЕНИЯ ОСНОВАННЫЙ НА МОДИФИКАЦИИ СОБСТВЕННЫХ ЗНАЧЕНИЙ КОРРЕЛЯЦИОННОЙ МАТРИЦЫ

Рассматриваются методы спектрального сверхразрешения, основанные на модификациях собственных значений корреляционной матрицы принимаемого сигнала.

Применение параметрических моделей случайных процессов обусловлено возможностью получения на основе этих моделей более точных оценок спектральной плотности мощности (СПМ), чем это возможно с помощью классических методов спектрального оценивания [1]. Классические методы дают оценки СПМ по взвешенной последовательности данных или оценок автокорреляции. Отсутствующие данные или неоценённые значения автокорреляционной последовательности за пределами применяемого окна неявно полагаются равными нулю, что является нереалистичным допущением и, как следствие, приводит к искажениям спектральных оценок. На практике часто имеется некоторая априорная информация относительно процесса, из которого берутся отсчёты данных, что позволяет принимать более адекватные допущения об обрабатываемых данных.

Так, при короткой выборке x процесса параметрические методы оценки СПМ дают лучшее спектральное разрешение и оценивание частоты, однако при низких отношениях сигнал-шум параметрические методы (например, авторегрессионный метод и метод Прони) не в состоянии разрешить близкие по частоте моды [1].

Одним из методов улучшения характеристик спектрального оценивания является анализ собственных значений λ_n оценок \mathbf{R} автокорреляционной матрицы \mathbf{R}_x анализируемого процесса, полученных по зашумлённой выборке x . Ключевой операцией в этих методах является разделение информации, содержащейся в \mathbf{R} , на два векторных подпространства — подпространство сигнала и подпространство шума [1].

Сравним различные методы оценки СПМ на примере анализа процесса, многомодового по спектру процесса с нормированной корреляционной матрицей $\mathbf{R}_x=[R_{j,k}]$, соответствующей аддитивной сумме M односвязных марковских процессов:

$$R_{j,k} = \sum_{m=0}^{M-1} \left(P_m \cdot e^{-\pi \cdot |j-k| \cdot \Delta FT_m} \cdot e^{-i \cdot 2\pi \cdot \varphi_m \cdot (j-k)} \right) \bigg/ \sum_{m=0}^{M-1} P_m,$$

где ΔFT_m — относительная ширина m -ой спектральной моды, φ_m — относительная частота m -ой моды, P_m — относительная мощность моды, M — количество спектральных мод, i — мнимая единица.

В качестве контрольной модели используем авторегрессионную модель (АР) 200-го порядка, коэффициенты которой рассчитываются по нормальному уравнению Юла-Уолкера [1]:

$$\mathbf{a}_x = \mathbf{R}_x^{-1} \mathbf{i},$$

где \mathbf{a}_x — вектор коэффициентов авторегрессии контрольной АР-модели, \mathbf{i} — крайний левый столбец единичной матрицы. Она является идеализированной и не содержит мешающие компоненты, в качестве которых используется аддитивный комплексный белый гауссовский шум с нулевым средним и дисперсией σ_n^2 .

Для сравнения используются два метода, основанные на анализе собственных чисел λ_n $(p+1) \times (p+1)$ -мерной матрицы \mathbf{R} для её последующей модификации $\tilde{\mathbf{R}}$ с целью получения вектора $\mathbf{a} = \tilde{\mathbf{R}}^{-1} \mathbf{i}$ оценок коэффициентов авторегрессии оптимизированной АР-модели p -го порядка.

В качестве критерия эффективности спектрального анализа используется нормированный к числу L спектральных отсчётов квадрат среднего отклонения ε между нормированными СПМ $\mathbf{c} = [c_l]$ контрольной модели и $\mathbf{s} = [s_l]$ оптимизированной модели, полученной по зашумлённым данным:

$$\varepsilon = \sum_{l=0}^{L-1} (c_l - s_l)^2 \bigg/ L = (\mathbf{c} - \mathbf{s})^T (\mathbf{c} - \mathbf{s}) / L, \quad (1)$$

где T — знак транспонирования.

Моделирование зашумлённой выборки \mathbf{x} осуществляется путём суммирования полученной при помощи контрольной АР-модели выборки \mathbf{x} идеализированного процесса с отсчётами комплексного белого гауссовского шума.

Первый метод основан на получении матрицы $\tilde{\mathbf{R}}_1$ из оценки автокорреляционной матрицы \mathbf{R} путём вычитания из её главной диагонали оценки мощности шума, в качестве которой принимается минимальное собственное число $\lambda_{\min} \approx \sigma_n^2$ матрицы \mathbf{R} [2]:

$$\tilde{\mathbf{R}}_1 = \mathbf{R} - \lambda_{\min} \mathbf{I},$$

где $\lambda_{\min} = \min(\lambda)$ — минимальное собственное число матрицы \mathbf{R} , соответствующее оценке мощности шумовой компоненты аддитивной

смеси \mathbf{x} сигнала и шума, \mathbf{I} — единичная матрица, $\boldsymbol{\lambda}$ — вектор собственных чисел матрицы \mathbf{R} . Тогда вектор AP-коэффициентов \mathbf{a}_1 для первой модели: $\mathbf{a}_1 = (\hat{\mathbf{R}}_1)^{-1} \mathbf{i}$.

Второй метод основан на введении в сигнал априорной информации, содержащей значение максимального собственного числа λ_x матрицы \mathbf{R}_x . Зная λ_x и максимальное собственное значение $\lambda_{\max} = \max(\boldsymbol{\lambda})$ матрицы \mathbf{R} можно найти коэффициент k , характеризующий изменение собственных чисел корреляционной матрицы \mathbf{R}_x сигнала, вызванное его зашумлением при передаче по каналу связи, исходя из равенства: $(\lambda_{\max})^k = \lambda_x$. Тогда модифицированный вектор $\tilde{\boldsymbol{\lambda}} = [\tilde{\lambda}_0, \tilde{\lambda}_1, \dots, \tilde{\lambda}_p]$ собственных значений может быть рассчитан следующим образом: $\tilde{\boldsymbol{\lambda}} = [(\tilde{\lambda}_0)^k, (\tilde{\lambda}_1)^k, \dots, (\tilde{\lambda}_p)^k]$, где $k = \ln(\lambda_x) / \ln(\lambda_{\max})$.

Восстанавливая модифицированную корреляционную матрицу $\hat{\mathbf{R}}_2$ через преобразование Карунена-Лоева, получаем:

$$\hat{\mathbf{R}}_2 = \mathbf{G} [\text{diag}(\tilde{\boldsymbol{\lambda}})] \mathbf{G}^{-1},$$

где \mathbf{G} — собственный вектор матрицы \mathbf{R} . Вектор AP-коэффициентов для второй модели обозначим как $\mathbf{a}_2 = (\hat{\mathbf{R}}_2)^{-1} \mathbf{i}$.

С целью оценки эффективности первого и предлагаемого второго методов проведём статистическое моделирование, основанное на спектральном анализе зашумлённой выборки \mathbf{x} , описание которой приведено выше. При этом параметры модели следующие: $M=3$, $\Delta FT_0 = \Delta FT_1 = \Delta FT_2 = 0.1$, $\varphi_0 = 0.25$, $\varphi_1 = 0.1$, $\varphi_2 = 0.2$, $P_0 = 1$, $P_1 = P_2 = 0.2$. Модель имитирует сигнал $3U_0$ напряжения, возникающего в устройстве релейной защиты высоковольтных (6...10 кВ) кабельных сетях при однофазном замыкании на землю [3] и содержит три низкочастотных гармоники переменного тока частотой 50 Гц.

Проведённое моделирование выбранных методов показывает, что при малых (-10 ... -15 дБ) отношениях b сигнал-шум наилучшим, с точки зрения выбранного критерия (1), является предлагаемый второй метод, основанный на учёте априорной информации о максимальном собственном значении λ_x . Первый метод даёт выигрыши при значительном превышении уровня полезного сигнала над шумом ($b > 5$ дБ). Отсутствие модификации корреляционной матрицы при синтезе AP-моделей целесообразно в рассмотренной задаче при $b > 10$ дБ.

Таблица 1

b , дБ	-15	-10	-5	0
ε_1	0.05	$2.162 \cdot 10^{-3}$	$1.234 \cdot 10^{-3}$	$8.607 \cdot 10^{-4}$
ε_2	$5.013 \cdot 10^{-3}$	$1.361 \cdot 10^{-3}$	$8.931 \cdot 10^{-4}$	$5.744 \cdot 10^{-4}$
ε_3	0.45	0.138	0.032	$1.65 \cdot 10^{-3}$

Анализ таблицы показывает, что при $b=-5$ дБ отношение $\mu=\varepsilon_1/\varepsilon_2$ СКО отклонений спектров ε_1 и ε_2 для первого и второго методов соответственно составляет $\mu=1.38$, а при $b=-15$ дБ достигает величины $\mu=9.97$.

Таким образом, использование априорной информации о максимальном собственном числе λ_x нормированной корреляционной матрицы \mathbf{R}_x передаваемого сигнала позволяет сократить в 9.97 раз отклонения ε спектров контрольной и рабочей моделей, построенных по модифицированной по второму методу корреляционной матрице $\tilde{\mathbf{R}}_2$ по сравнению с известным первым методом, использующим величину $\tilde{\mathbf{R}}_1$. Выигрыш достигается за счёт учёта дополнительной информации о спектре λ собственных значений сигнала для синтеза модифицированной матрицы $\tilde{\mathbf{R}}_2$ путём использования преобразования Карунена-Лоева.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Марпл-мл. С.Л. Цифровой спектральный анализ и его приложения: Пер. с англ. – М.: Мир, 1990. – 584 с.
2. Кошелев В.И., Андреев В.Г. Оценка собственных значений в задаче обработки эхо-сигналов // 100-летие начала использования электромагнитных волн для передачи сообщений и зарождения радиотехники: тезисы докладов Международной конференции, г. Москва, 4-6 мая 1995 г. – М.: Изд-во журнала «Радиотехника», 1995. – С. 180-181.
3. Андреев В.А. Релейная защита и автоматика систем электроснабжения. – М.: Высшая школа, 1991. – 496 с.

М.И. АРЕФЬЕВА, С.В. СКВОРЦОВ

Рязанский государственный радиотехнический университет

**ПРОГРАММНЫЕ СРЕДСТВА ДЛЯ ОПТИМИЗАЦИИ
ДИАГНОСТИЧЕСКИХ ГРАФОВЫХ МОДЕЛЕЙ
МНОГОПРОЦЕССОРНЫХ СИСТЕМ**

Сформулирована задача синтеза диагностических графов. Представлен алгоритм ее решения методом ветвей и границ, обеспечивающий минимальное покрытие булевой матрицы.

Методами обеспечения отказоустойчивости многопроцессорных систем являются резервирование аппаратуры и сравнение результатов решения задачи основными и резервными микропроцессорами (МП), периодическое тестирование оборудования, проведение взаимных тестовых проверок МП посредством дублирования вычислений для одинаковых фрагментов решаемой задачи с последующим сравнением результатов.

Диагностический граф (ДГ) – одна из наиболее известных формальных моделей, используемых в задачах автоматического обнаружения неисправностей в отказоустойчивых многопроцессорных системах, в общем случае представляющим собой ориентированный граф $D = (U, T)$. ДГ описывает множество ЭП, выполненных за цикл контроля, и их результаты, позволяющие определить техническое состояние ВМ в реальном времени. В классическом определении ДГ представляется как ориентированный граф $G=(U, D)$, вершинам $U = \{u_1, u_2, \dots, u_n\}$ которого сопоставлено множество ВМ системы, а ребра задают пары модулей, выполняющих ЭП. Каждое ребро $(u_i, u_j) \in D$ взвешено булевым значением $s_{ij} \in \{0, 1\}$ результата ЭП, причем его направление определяет функции ВМ: u_i является контролирующим, а u_j - контролируемым. Совокупность всех значений s_{ij} , полученных за цикл контроля, образует синдром S системы.

Анализ синдрома S выполняется на основе некоторой диагностической модели (ДМ), согласно которой любой модуль системы тестируется с помощью нескольких других модулей, т.е. определяются возможные результаты $s_{ij} \in \{0, 1\}$ одной элементарной проверки (ЭП) для различных состояний пары МП u_i, u_j . Диагностическая модель задается четверкой логических переменных вида: $(z_{r\tau}, z_{r\tau f}, z_{f\tau}, z_{ff})$, где первый индекс указывает состояние контролирующего u_i , а второй - контролируемого МП u_j (r - исправен, f - отказал).

Диагностические графы, порождаемые различными дисциплинами активной отказоустойчивости при заданном значении z , могут отличаться числом и структурой проверочных связей. Отсюда актуальной является задача синтеза ДГ $G=(U,D)$ с минимальным числом ребер $|D| \rightarrow \min$, дешифрация которого с использованием выбранной ДМ обеспечивает необходимое значение t .

Многие задачи, связанные с диагностированием многопроцессорных систем, требуют для решения построения двоичных таблиц, описывающих их возможные состояния. Таблица функций неисправности это универсальная математическая модель для диагностирования систем, в которой столбцы соответствуют допустимым ЭП, а строки — техническим состояниям объекта диагностирования, т.е. значениям синдрома S для каждого допустимого состояния системы. Размер такой таблицы определяет количество взаимных проверочных связей процессоров в цикле контроля технического состояния. Корректная дешифрация текущего синдрома S , с помощью известных диагностических моделей, позволяет правильно идентифицировать все неисправные ВМ, и определить меру t -диагностируемости.

Для того чтобы ускорить время поиска неисправностей необходимо уменьшать размеры таких таблиц, что приводит к решению задачи, связанной с определением наименьшего покрытия.

Поставленную задачу предлагается решать в два этапа. На первом этапе производится выбор дисциплины активной отказоустойчивости, которая порождает ДГ $G^*=(U,D^*)$ [2], возможно избыточный, но обеспечивающий заданный уровень диагностируемости t . На втором этапе из графа G^* исключаются избыточные связи с целью получения диагностического графа G , ребра которого имеют минимальную суммарную стоимость.

В результате выполнения первого этапа для ДГ $G^*=(U,D^*)$ формируется таблица функций неисправностей системы. На втором этапе производится исключение избыточных ребер ДГ с учетом следующего требования: если все строки матрицы попарно различимы, то можно однозначно выполнить дешифрацию синдрома и установить текущее техническое состояние ВМ системы.

Задача построения минимального покрытия полученной матрицы относится к NP-полным и требует экспоненциальных затрат времени для ее решения.

Задача о покрытии является достаточно сложной комбинаторной задачей, и на ЭВМ она чаще всего решается в два этапа. На первом (предварительном) этапе выясняется, имеет ли вообще исходная задача решение, и если оно существует, то определяется приближенное ре-

шение. Здесь же могут быть найдены элементы покрывающего множества (столбцы), включаемые в оптимальное решение, что позволяет упростить исходную задачу. На втором этапе на основе приближенного определяется оптимальное решение.

Нахождение приближенного решения. Задача о покрытии имеет решение, если в исходной матрице нет нулевых строк. Приближенным решением задачи о покрытии является подматрица матрицы A без нулевых строк с числом столбцов, близким к минимально возможному. Для нахождения приближенного решения, как правило, используется градиентный метод, который состоит в пошаговом выделении столбцов (включении в приближенное решение) результирующей матрицы:

- на первом шаге выделяется столбец, содержащий наибольшее число единиц (если таких несколько, то берется любой из них), и в матрице вычеркиваются (считаются покрытыми) все строки, содержащие единицу в выделенном столбце;

- на k -м шаге выполняются те же действия над матрицей, полученной (в результате вычеркивания строк) на предыдущем шаге.

Этот процесс заканчивается, если на очередном шаге все строки рассматриваемой матрицы оказались вычеркнутыми. Подматрица, составленная из тех столбцов исходной матрицы A , которые выделялись в процессе выполнения алгоритма, и является искомым подматрицей.

После получения приближенного решения выясняется, можно ли некоторые из столбцов включить в оптимальное решение. Это осуществляется по следующему правилу: если в какой-либо строке имеется всего один единичный элемент, то содержащий его столбец должен быть включен в оптимальное решение. При этом из подматрицы, передаваемой на оптимизацию, вычеркиваются эти столбцы и покрываемые ими элементы.

Определение оптимального решения. Полученное на предыдущем этапе приближенное решение передается алгоритму минимизации. Целью этого этапа является дальнейшее уменьшение (если это возможно) числа столбцов подматрицы.

Для определения оптимального решения, как правило, используется метод ветвей и границ.

Исходными данными для применения алгоритма является матрица, представляющая собой приближенное решение задачи о наименьшем покрытии с вычеркнутыми столбцами, включенными в оптимальное решение, и строками, которые они покрывают.

Нахождение оптимального решения задачи о наименьшем покрытии в табличной форме состоит из двух основных повторяющихся этапов.

На первом этапе находят одно из допустимых решений, а на втором оно проверяется на оптимальность. Если текущее решение не оптимально, то возвращаются на первый этап, где формируют новое допустимое решение, иначе алгоритм заканчивает работу.

В ходе выполнения первого этапа над столбцами, включенными в решение, ставят индексы, а снизу эти столбцы помечают знаком *.

Строки, содержащие единицу в столбцах, имеющих индекс и метку, считаются покрытыми.

В процессе проверки текущего решения на оптимальность индексы и метки со столбцов снимаются, после чего соответствующие строки считаются непокрытыми. Текущее решение является оптимальным, если число столбцов, входящих в него, меньше числа столбцов, включенных в предыдущее решение.

Алгоритм ветвей и границ для решения задачи о наименьшем покрытии в табличной форме состоит из следующих шагов.

1. Среди столбцов, не имеющих индекса, находится столбец, обладающий максимальной мощностью (мощностью столбца называют число единиц в нем, расположенных в непокрытых строках). Над ним указывается индекс, значение которого равно, например, числу обращений к п.1, а снизу он помечается знаком *. Покрываемые им строки отмечаются справа знаком +.

2. Находится оценка нижней границы количества столбцов L_1 текущего решения как сумма числа столбцов, имеющих метку *, и числа столбцов, необходимых для покрытия непокрытых строк. Последнее определяется как минимальное число столбцов, не имеющих индекса, суммарная мощность которых больше или равна числу непокрытых строк (если мощность оказывается меньшей, то оценка нижней границы считается равной общему числу столбцов матрицы покрытий).

3. Проверяется, если число столбцов L_1 текущего решения больше или равно числу столбцов L_0 предыдущего решения, то переходят к п. 4, иначе, если не все строки покрыты, возвращаются к п.1. (Первоначально L_0 приравнивают к числу столбцов в матрице покрытий.) Если же $L_1 < L_0$ и все строки покрыты, то формирование очередного допустимого решения закончено. В этом случае запоминают номера и число помеченных столбцов и переходят к проверке решения на оптимальность.

4. Проверяется, помечен ли столбец, включенный в решение последним. Если помечен, то метка с него снимается (соответствующие строки считаются непокрытыми), и переходят к п. 2. Если же столбец,

включенный в решение последним, не помечен, то с него снимается индекс.

5. Проверяется наличие столбцов с индексами. Если таких нет, то исследуемое решение оптимально, иначе возвращаются на п.4.

Качество решения задачи синтеза ДГ с минимальным числом ребер существенно зависит от эффективности используемого алгоритма поиска минимального покрытия. Предложенный в статье алгоритм, основанный на методе ветвей и границ, гарантирует получение точного решения, но требует больших временных затрат.

В результате проделанной работы разработаны программные средства синтеза диагностических графовых моделей многопроцессорных систем, которые выполняют следующие функции: построение таблицы функций неисправностей, дешифрацию текущего синдрома S и визуальное отображение результатов, при этом можно использовать любую из известных ДМ, что позволяет применять разработанную программу в учебных и исследовательских целях.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Скворцов Н.В., Скворцов С.В., Хрюкин В.И.. Синтез диагностических графов для многопроцессорных систем с активной отказоустойчивостью // Вестник РГРТУ. 2012. № 1 (вып. 39). Ч. 2. С. 83-89.

2. Основы технической диагностики. В 2-х книгах. Кн. 1. Модели объектов, методы и алгоритмы диагноза / Под ред. П.П. Пархоменко. – М.: Энергия, 1976. – 464 с.

А.В. БАКУЛЕВ, М.А. БАКУЛЕВА

Рязанский государственный радиотехнический университет

О НЕКОТОРЫХ СВОЙСТВАХ МЕТОДА СПЕКУЛЯТИВНОЙ МНОГОПОТОЧНОСТИ

Рассматриваются свойства метода спекулятивной многопоточности, используемого для динамического распараллеливания программ. На основе исходных посылок формулируются и строго доказываются леммы, которые можно использовать в дальнейшем для исследования характеристик данного метода.

Наиболее подходящим методом динамического распараллеливания последовательных программ для вычислительных систем с многоядерными процессорами является метод *спекулятивной многопоточности* [1].

Суть метода состоит в следующем. Среди множества всех регионов G последовательной программы выявляются регионы, имеющие зависимости типа δ_{out-in}^∞ , δ_{in-out}^∞ , $\delta_{out-out}^\infty$, характер которых не может быть установлен на этапе трансляции программы из-за неоднозначности [2]. Статические методы при планировании параллельного исполнения подобных регионов вынуждены использовать более сильные виды отношений, прибегая к позиции крайнего пессимизма. Такой подход негативно отражается на результатах распараллеливания. Метод спекулятивной многопоточности предписывает параллельное многопоточное выполнение подобных регионов, в расчёте на то, что информационные зависимости на стадии исполнения не проявятся, прибегая к позиции крайнего оптимизма (спекулируя на удаче). В случае если подобные надежды оправдают себя, будет получен выигрыш в производительности, в противном случае, результаты вычислений региона должны быть аннулированы, и он будет выполнен повторно, что приведет к накладным расходам. Регионы программы, к которым применяется подобный метод, называют **спекулятивными** [1]

$$\forall g_i \in G^{sp} \mid G^{sp} \subseteq G, i = \overline{1, K}, \exists (v_i, v_m), v_i \delta_v^\infty v_m, (v_i \in \delta_v(g_i) \vee v_m \in \delta_v(g_i)).$$

Для того, чтобы организовать многопоточное выполнение спекулятивных регионов, формируется динамическая последовательность стадий их исполнения, называемых **эпохами** [1]

$\forall e_i \in EP \mid i = \overline{1, N}, N = |EP|$. Например, для циклического региона эпохами являются отдельные итерации цикла. Каждая эпоха снабжается локальным буфером памяти для сохранения критических к возможным нарушениям зависимости данных. Результаты выполнения эпохи приводятся в соответствие со следующими правилами.

Правило 1. После окончания спекулятивного исполнения каждой эпохи региона проверяется, не произошло ли при этом нарушения информационных зависимостей.

Правило 2. Если нарушения не было зафиксировано, результаты вычисления эпохи переносятся из защищённого буфера в общую память программы и эпоха освобождает свой процессорный элемент.

Правило 3. Если было зафиксировано нарушение информационной зависимости, результаты аннулируются и эпоха запускается на исполнение повторно.

Правило 4. Всем эпохам присваиваются уникальные номера. Нумерация эпох осуществляется строго в порядке их исполнения в последовательной программе. Поэтому исполнение эпох в порядке следования их номеров гарантирует точное соответствие получаемых результатов результатам последовательного выполнения программы.

Правило 5. Проверка результатов выполнения эпохи на корректность осуществляется только после того, как эпоха с предыдущим номером успешно завершила своё выполнение.

Правило 6. Новые, ожидающие запуска на исполнение, эпохи распределяются по освободившимся от выполнения предыдущих эпох процессорным элементам строго в порядке следования их номеров.

Учитывая правила 1–6, сформулируем следующее утверждения.

Лемма 1. При одновременном параллельном спекулятивном исполнении p эпох $e_j, e_{j+1}, \dots, e_{j+p-1}$, на вычислительной системе состоящей из p процессорных элементов, гарантируется корректность выполнения $\forall e_j$ и отсутствие её повторного перезапуска, где $j, j+1, \dots, j+p-1$ – номера эпох $e_j, e_{j+1}, \dots, e_{j+p-1} \in EP \mid j = \overline{1, (N-p+1)}$.

Доказательство. Если в данный момент времени выполняются эпохи с номерами $j, j+1, \dots, j+p-1$, значит либо эпохи с номерами $1, 2, \dots, j-1$ уже успешно завершились ранее, либо $j=1$ и e_j — это самая первая эпоха региона. В первом случае результаты выполнения эпохи e_j уже не смогут нарушить информационные зависимости между операторами внутри региона, так как в соответствии с правилом 4 нумерация эпох соответствует порядку их исполнения в последовательной программе, т.е. между эпохами e_{j-1} и e_j соблюдается самое строгое отношение типа «последовательность» [2] $e_{j-1} \delta_{seq} e_j \mid j > 1$ и e_{j-1} уже успешно завершилось. Во втором случае эпоха $e_j = e_1$ является самой первой в последовательности эпох региона и может быть выполнена первой без нарушения зависимостей. Таким образом, в соответствии с правилом 2 эпоха e_j успешно завершится без повторного перезапуска, что и требовалось доказать.

Лемма 2. При одновременном параллельном спекулятивном исполнении p эпох $e_j, e_{j+1}, \dots, e_{j+p-1} \in EP \mid j = \overline{1, (N-p+1)}, N = |EP|$ на вычислительной системе состоящей из p процессорных элементов, корректность выполнения $\forall e_{j+k} \mid k = \overline{1, (p-1)}$ гарантируется, если подтверждена корректность $\forall e_{j+k-1}$ и подтверждена корректность результатов между каждой парой эпох $\forall (e_{j+k}, e_{j+m}) \mid m = \overline{0, (k-1)}$.

Доказательство. Докажем лемму 2 сначала для эпохи $\forall e_{j+k} \mid k = 1$. Согласно лемме 1, корректность выполнения эпохи $\forall e_j$

можно считать гарантированной, а в соответствии с правилом 4 нумерация эпох соответствует порядку их исполнения в последовательной программе, т.е. между эпохами e_j и e_{j+1} соблюдается самое строгое отношение типа «последовательность» $e_j \delta_{seq} e_{j+1}$. Тогда, если e_j уже успешно завершилась, и подтверждается проверка корректности результатов выполнения e_{j+1} относительно e_j , то корректность выполнения эпохи e_{j+1} считается гарантированной относительно $\forall e_m \mid m = \overline{(j+2), (j+p-1)}$, что и требовалось доказать. Рассмотрим теперь эпоху $\forall e_{j+k} \mid k = 2$. Допустим что корректность $\forall e_{j+1}$ была подтверждена способом, доказанным только что. Так как по правилу 4 нумерация эпох соответствует порядку их исполнения в последовательной программе, то между эпохами e_j , e_{j+1} , e_{j+2} соблюдается самое строгое отношение типа «последовательность»: $e_j \delta_{seq} e_{j+1}$ и $e_{j+1} \delta_{seq} e_{j+2}$. В силу транзитивности этого отношения можно установить и отношение $e_j \delta_{seq} e_{j+2}$. Тогда, если подтверждается проверка корректности результатов выполнения e_{j+2} относительно e_j и e_{j+2} относительно e_{j+1} , то корректность выполнения эпохи e_{j+2} считается гарантированной относительно $\forall e_m \mid m = \overline{(j+3), (j+p-1)}$, что и требовалось доказать. Опираясь на доказательство леммы 2 для $k = 1$ и $k = 2$, далее по индукции можно доказать лемму 2 и для $\forall k = \overline{1, (p-1)}$.

Лемма 3. Если при одновременном параллельном спекулятивном исполнении p эпох $e_j, e_{j+1}, \dots, e_{j+p-1} \in EP \mid j = \overline{1, (N-p+1)}, N = |EP|$ на вычислительной системе состоящей из p процессорных элементов $\exists e_{j+k} \mid k = \overline{1, (p-1)}$, для которой было зафиксировано нарушение корректности выполнения, то после перезапуска и повторного исполнения e_{j+k} её корректность будет гарантирована.

Доказательство. Согласно правилу 5 проверка корректности $\forall e_{j+k} \mid k = \overline{1, (p-1)}$ производится только после того, как была подтверждена корректности выполнения эпохи e_{j+k-1} . Следовательно, учитывая лемму 2, можно заключить, что факт нарушения корректности выполнения эпохи e_{j+k} мог быть вызван только отрицательным результатом одной из проверок корректности, проводимых между каждой па-

рой эпох $\forall (e_{j+k}, e_{j+m}) \mid m = \overline{0, (k-1)}$. Чтобы устранить нарушение информационной зависимости, в соответствии с правилом 3 происходит перезапуск эпохи e_{j+k-1} . Рассмотрим момент времени, связанный с началом перезапуска эпохи e_{j+k} . Эпохи $\forall e_{j+m} \mid m = \overline{0, (k-1)}$ к этому моменту времени уже будут успешно завершены и на их место в параллельный конвейер будут запущены согласно правилу 6 новые эпохи $e_{j+p}, \dots, e_{j+p+k-1} \mid j+p+k-1 \leq N$. Таким образом, после повторного запуска e_{j+k} параллельно будут выполняться следующие эпохи: $e_{j+k}, e_{j+k+1}, \dots, e_{j+p-1}, e_{j+p}, \dots, e_{j+p+k-1} \mid j+p+k-1 \leq N$. Тогда согласно лемме 1 корректность выполнения e_{j+k} будет гарантирована, что и требовалось доказать.

Сформулированные и доказанные леммы 1–3 позволят проанализировать работу метода спекулятивной многопоточности и обосновать расчет важных характеристик, определяющих область его эффективной применимости.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Steffan J.G., Mowry T.C. The Potential for Using Thread-Level Data Speculation to Facilitate Automatic Parallelization. Carnegie Mellon University, HPCA-4, February 1-4, 1998.
2. Бакулев А.В. Выявление информационных зависимостей между операторами последовательной программы//Тез. докл. 15-ой Всероссийской научно-технической конференции студентов, молодых ученых и специалистов «Новые информационные технологии в научных исследованиях и в образовании». – Рязань: РГРТУ, 2010. С. 185-187.

А.Н. ВАРНАВСКИЙ, Н.В. ЧЕКАН

Рязанский государственный радиотехнический университет

ДИНАМИЧЕСКОЕ ОПРЕДЕЛЕНИЕ КОЭФФИЦИЕНТА ВАРИАЦИИ УЧАСТКОВ БИОЭЛЕКТРИЧЕСКИХ СИГНАЛОВ

Рассматривается способ нахождения временных параметров отсчетов биоэлектрических сигналов на основе интегрального определения коэффициентов вариации участков сигнала.

Одной из актуальных задач является определение состояния операторов человеко-машинных систем, в частности, транспортных. Данная задача может быть решена путем анализа биоэлектрических сигналов.

Система кровообращения отличается высокой реактивностью и играет первостепенную роль в адапционных перестройках функционального состояния организма. При быстром течении стрессовых реакций сердечная деятельность во многих случаях является наиболее информативным показателем изменений состояния организма. Поэтому информация о состоянии оператора может быть получена путем анализа электрокардиосигнала (ЭКС) [1].

При этом задача анализа ЭКС связана с определением амплитудно-временных параметров отсчетов, в частности, t_k^* - времени k -го отсчета относительно начала соответствующего зубца. Для определения этого времени можно сравнивать отсчеты сигнала с двумя пороговыми уровнями $\pm \Delta_t$, расположенными выше по модулю отсчетов шума сегментов. В случае превышения пороговых уровней отсчетами сигнала u_k необходимо осуществлять формирование значений времени t_k таких отсчетов:

$$t_k = \begin{cases} 0, & \text{если } |u_k| \leq \Delta_t, \\ t_{k-1} + \Delta t, & \text{если } |u_k| > \Delta_t. \end{cases} \quad (1)$$

Для формирования пороговых уровней определения t_k^* можно использовать интегральный анализ по нахождению коэффициента вариации совокупности отсчетов ЭКС.

Рассмотрим n отсчетов u_k флуктуационного шума в диапазоне $[-1; 1]$ (рисунок 1, а). Определим основные статистические характеристики такого сигнала: мат. ожидание M , дисперсию S^2 и коэффициент вариации v :

$$M = \frac{1}{n} \sum_{n=0}^{n-1} u_n, \quad S^2 = \frac{1}{n-1} \sum_{n=0}^{n-1} (u_n - M)^2, \quad v = \frac{S}{M}.$$

Очевидно, что значение коэффициента вариации приобретает достаточно большое значение.

В случае если сигнал шума u_k смещен на некоторое значение (рисунок 1, б), то величина коэффициента вариации близка к нулю.

Таким образом, рассматривая значение коэффициента вариации на ограниченной совокупности отсчетов ЭКС, можно сделать вывод о том, какой элемент представляет данная совокупность отсчетов: при большом значении v совокупность отсчетов представляет собой интервал, при малом – зубец.

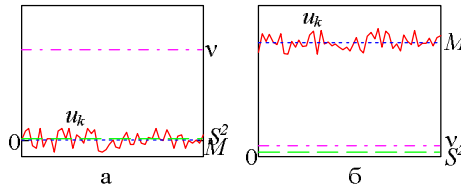


Рис. 1. Различия в значениях коэффициента вариации при разных уровнях сигналов

В кардиоцикле электрокардиосигнала человека-оператора, занимающегося нормальной деятельностью, TP-сегмент имеет наибольшую длительность среди других элементов. Значения амплитуд отсчетов на этом участке могут использоваться для формирования пороговых уровней определения t_k . В этом случае ширина окна может быть выбрана равной 0,2 с.

Рассмотрим ЭКС u_k (рис), частота дискретизации которого $\Delta t=250$ отс/с. Сформируем окно анализа длительностью $N1=50$ (0,2 с), в котором будем определять мат. ожидание u_k

$$M_k = \frac{1}{N1} \sum_{i=k-N1}^k |u_i|,$$

дисперсию относительно мат. ожидания в окне

$$S1_k^2 = \frac{1}{N1-1} \sum_{i=k-N1}^k (u_i - M_k)^2,$$

дисперсию относительно мат. ожидания каждого шага движения окна

$$S2_k^2 = \frac{1}{N1-1} \sum_{i=k-N1}^k (u_i - M_i)^2.$$

Определим коэффициенты вариации при расчете дисперсий разными способами:

$$v1_k = \frac{S1_k}{M_k} \text{ и } v2_k = \frac{S2_k}{M_k}$$

и найдем разность полученных значений:

$$v_k = v1_k - v2_k .$$

Полученный сигнал v_k представляет собой нулевой уровень, относительно которого в области ТР-сегмента образуются пики. Используя этот сигнал и пороговый уровень равный 2 можно выделить точки, соответствующие интервалу ТР.

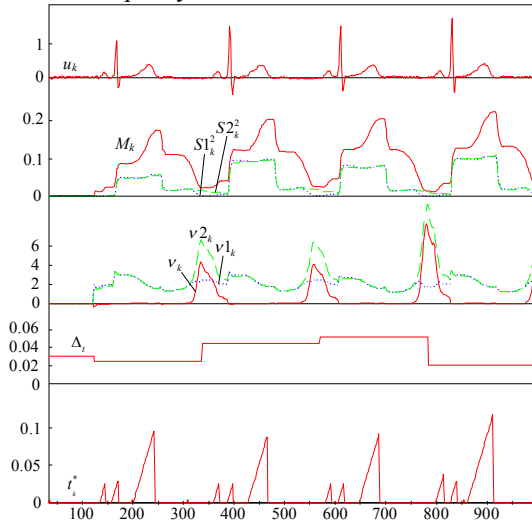


Рис. 2. Пример анализа ЭКС

Дисперсия сигнала $S1_k^2$ на участке ТР-сегмента $S1_{ТР}$ соответствует дисперсии шума на этом участке, поэтому значение порогового уровня для определения времени t_k^* отсчетов для каждого следующего кардиоцикла может быть выбрано равным [2]:

$$\Delta_t = 3S1_{ТР} .$$

После формирования Δ_t , по формуле (1) определяется t_k^* .

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Панченко Л.Л. Диагностика стресса: Учеб. пособие. – Владивосток: Мор. гос. ун-т, 2005. 35 с.
2. Варнавский А.Н., Мельник О.В, Михеев А.А. Метод выделения опорной точки в каждом кардиоцикле //Биомедицинские технологии и радиоэлектроника. 2005. №1-2. С. 36-39.

А.А. ВЕТРОВ, А.М. КОЧЕРГИН

Рязанский государственный радиотехнический университет

ОРГАНИЗАЦИЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ В WEB-ГЕОПОРТАЛЕ СПУТНИКОВЫХ ИЗОБРАЖЕНИЙ

В статье рассматривается организация параллельных вычислений в web-геопортале, разработанном для online-доступа к спутниковым изображениям.

Введение.

В настоящее время всё большее распространение получают Web-приложения, т.е. приложения, выполняемые в браузере пользователя. По сравнению с традиционными, они имеют ряд достоинств:

- не требуют установки на компьютер;
- являются кроссплатформенными;
- используемая версия приложения всегда актуальна;
- большая безопасность, достигаемая за счёт контроля выполняемого кода со стороны браузера.

В то же время, имеется и ряд недостатков:

- используемые ресурсы находятся на удалённом сервере, вследствие чего скорость доступа к ним на порядки ниже, чем к ресурсам традиционных приложений, находящимся на локальном жёстком диске;
- ограничения со стороны браузера. Так, пользователь может запретить Web-приложениям доступ жёсткому к диску, ограничить объём используемой оперативной памяти и т.д.;
- меньшее быстродействие по сравнению с традиционными приложениями, поскольку для обеспечения кроссплатформенности и безопасности используются интерпретируемые языки.

Одной из специфических особенностей геоинформационных систем (ГИС) является обработка больших объёмов данных, таких как растровые изображения, векторные карты, метаданные, цифровые модели рельефа и т.д. Вследствие этого Web-ГИС приложение, не использующее параллельную обработку является неудобным для пользователя – во время длительных процессов загрузки и обработки данных он не может с ним взаимодействовать и вынужден простаивать. Одним из путей решения этой проблемы является использование параллельных вычислений, позволяющее, кроме того, повысить эффективность использования вычислительных мощностей современных компьютеров. Этот путь и используется в описываемом в данной статье геопортале.

Общая архитектура геопортала.

Основной частью разработанного геопортала, написанного на языке Java, является встраиваемое в Web-страницу приложение – Java-апплет. Данный апплет формирует интерфейс пользователя, предназначенный для поиска снимков по различным параметрам и 3D модель Земли с возможностью их отображения на её поверхности. Визуализация 3D модели осуществляется средствами API OpenGL (Open Graphics Library) и библиотеки JOGL (Java OpenGL). Разработанный апплет является кроссплатформенным и корректно функционирует под Windows и Linux. При проектировании ставилась задача минимизировать вычислительную нагрузку на серверную часть, в результате в юрисдикции сервера остались только вопросы учёта пользователей и предоставления спутниковых изображений и предварительно снятых ресурсов. Укрупнённая схема геопортала представлена на рис. 1.

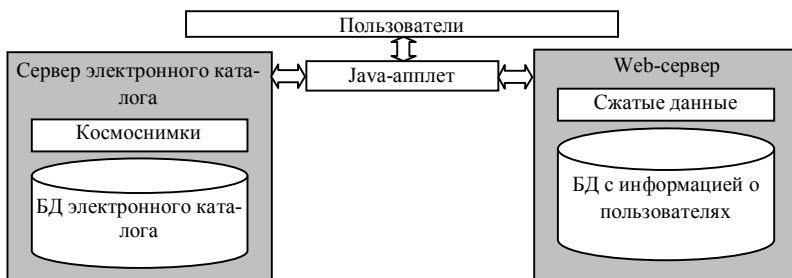


Рис. 1. Укрупнённая схема геопортала

Внутренняя структура Java-апплета показана на рис. 2. В процессе запуска апплета выполняется несколько предварительных операций. Вначале управление передаётся JNLP-загрузчику – специальному модулю, задачей которого является определение пользовательской платформы и загрузка специфических для неё файлов библиотеки JOGL. Затем начинает выполняться загрузчик ресурсов, выполняющий поиск ранее сохранённых на жёстком диске пользователя файлов, необходимых для работы апплета и загружающий их с Web-сервера при их отсутствии на диске.

После завершения загрузки всех требуемых файлов производится запуск основных потоков:

- поток визуализации. В рамках этого потока функционируют следующие модули:

- модуль отображения и управления 3D моделью;
- модуль фрагментации и выбора отображаемых фрагментов;
- модуль геометрических преобразований;

– модуль взаимодействия с базой данных электронного каталога (ЭК).

- потоки управления кэшированием:
 - модуль управления кэшем снимков;
 - модуль управления кэшем городов;
 - модуль управления кэшем векторных карт;
- потоки-загрузчики.

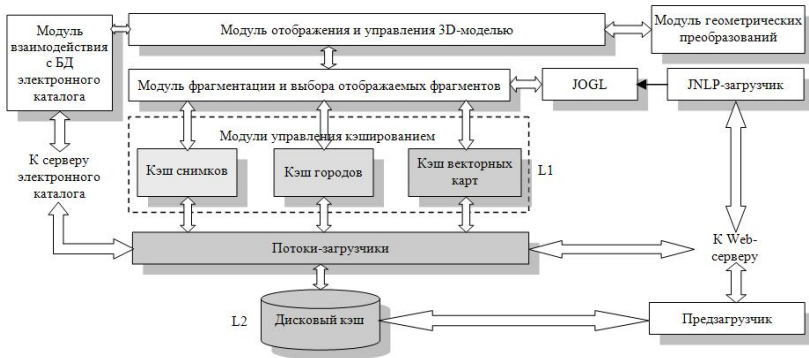


Рис. 2. Внутренняя структура Java-апплета (цветом показаны модули, выполняющиеся в различных потоках)

Кратко рассмотрим функции этих модулей. Модуль отображения и управления 3D моделью формирует трёхмерную модель Земли и отображает на ней различные объекты, такие как графическая подложка, векторные карты, растры и контуры найденных снимков и другие, в том числе и анимированные. Он обеспечивает возможность навигации по модели путём её поворота, масштабирования и перехода к заданным регионам. Основной задачей модуля фрагментации и выбора отображаемых фрагментов является повышение быстродействия геопортала, в том числе за счёт снижения передаваемых по сети данных. Для этого все отображаемые объекты разбиваются на множество небольших фрагментов с различными разрешениями, после чего загрузка, обработка и визуализация выполняется только для видимых фрагментов. Модуль геометрических преобразований обеспечивает трансформацию снимков от различных КА в проекцию, используемую при их отображении на поверхности модели. Модуль взаимодействия с базой данных электронного каталога (ЭК) формирует запросы к базе данных на основе критериев поиска, заданных пользователем, и возвращает ссылки на найденные снимки. Каждый из модулей управления кэшированием выполняется в отдельном потоке и ответственен за хранение объектов определённого типа в кэше в оперативной памяти

(L1), выдачу их другим потокам и формирование команд для потоков-загрузчиков. Потоки-загрузчики служат для поиска, загрузки и сохранения объектов в дисковом кэше (L2), загрузки и распаковки данных с Web-сервера и предварительной обработки данных.

Методы уменьшения передаваемого по сети объёма данных.

Для уменьшения передаваемого по сети объёма данных и, следовательно, ускорения работы приложения, используется несколько подходов.

Во-первых, все объекты, для которых возможно достижение высокого коэффициента сжатия, перед помещением на сервер архивируются с использованием алгоритма LZMA. Данный алгоритм обеспечивает высокий коэффициент сжатия [1], но распаковка заархивированных им объектов обладает высокой вычислительной сложностью. В связи с этим для каждого объекта распаковка выполняется только один раз – после первоначальной загрузки с Web-сервера. В кэшах (L1 и L2) хранятся уже распакованные данные.

Во-вторых, используется механизм фрагментации. В его основе лежит следующая идея. Очевидно, что при любом положении модели Земли и любом её масштабе вся её поверхность никогда не будет отображена на экране. В связи с этим можно значительно снизить объём передаваемых по сети данных и затраты на отображение объектов, разбив их на фрагменты и выполнять загрузку и отображение только видимых в текущий момент времени фрагментов. Кроме того, фрагментация позволяет снизить объём передаваемых по сети данных за счёт использования пирамидального механизма. Его суть в том, что помимо пространственного разбиения объектов на фрагменты выполняется разбиение по уровню детализации. В результате фрагменты больших размеров содержат данные с низкой детализацией и отображаются при меньших масштабах модели. При больших масштабах используются фрагменты малого размера, хранящие данные высокой детализации. Это позволяет загружать и отображать высокодетальные объекты только для отображаемой зоны и только в том случае, когда они действительно будут видны. Более того, при мелких масштабах часть векторных объектов (таких как карты дорожной сети и рельефа) не передаётся и не отображается вовсе. Описанный механизм поясняется рис. 3.

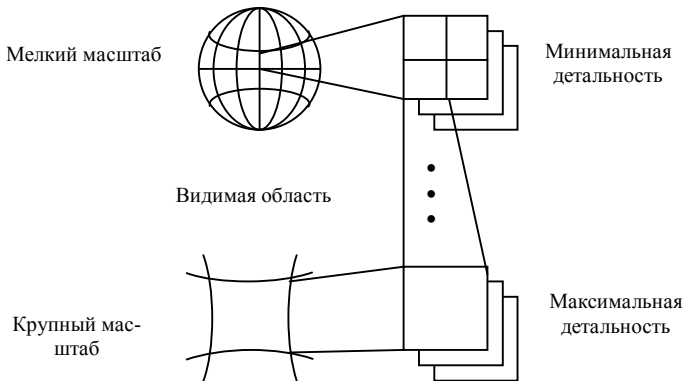


Рис. 3. Механизм фрагментации

Особенности реализации параллельных вычислений.

Рассмотрим особенности реализации параллельных вычислений подробнее. Обобщённая диаграмма межпоточных взаимодействий показана на рис. 4.

Для выполнения визуализации необходимо выполнить ряд ресурсоёмких операций. К ним относится определение набора отображаемых фрагментов в модуле фрагментации и формирование так называемых дисплейных списков OpenGL – групп команд графической подсистемы, которые сохраняются для последующего выполнения. При построении списка отображаемых фрагментов необходимо выполнять обратное проецирование точек с экранной плоскости на 3D модель. Помимо того, что данный процесс приводит к значительным вычислительным затратам, он может выполняться только во время визуализации кадра, что связано с особенностями архитектуры библиотеки JOGL. Из этого следует, что построение списка отображаемых фрагментов при визуализации каждого кадра приведёт к недопустимому снижению быстродействия. Для преодоления этой проблемы обновление списка отображаемых фрагментов выполняется во время простоя геопортала. Геопортал считается простаивающим, если в течение заданного времени не выполняется визуализация. Данное решение не оказывает значительного влияния на удобство использования геопортала, так как изменение списка фрагментов, отображаемых на экране, происходит только при повороте и масштабировании 3D модели. Аналогично во время простоя выполняется формирование дисплейных списков.

В процессе отображения 3D модели поток визуализации, основываясь на списке отображаемых фрагментов, запрашивает их у пото-

ков управления кэшированием, которые возвращают ему найденные объекты. Если для объекта уже существует дисплейный список, то он будет немедленно отображён, в противном случае отображение не выполняется, а дисплейный список будет сформирован во время простоя. В случае если запрошенный объект отсутствует, отображение пропускается, а поток управления кэшированием помещает запрос на загрузку объекта в специальную очередь. В то же время, каждый поток управления кэшированием выполняет периодический опрос потоков-загрузчиков. Все объекты, загрузка которых была завершена, переносятся в кэш в оперативной памяти (L1), после чего производится проверка очереди требующихся объектов и потоку визуализации передаётся запрос на перерисовку сцены. В случае её ненулевого размера и наличия простаивающих потоков-загрузчиков соответствующее число объектов извлекается из очереди и передаётся потокам-загрузчикам.

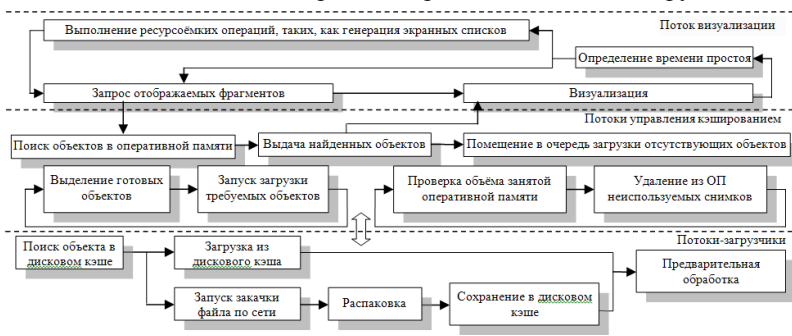


Рис. 4. Обобщённая диаграмма межпоточных взаимодействий

Также, периодически производится проверка объёма оперативной памяти, занимаемой L1 кэшем и удаление из него объектов, в случае, если он превышает заданный порог. Для этого используется следующий механизм. У каждого хранящегося в L1 кэше объекта присутствуют два поля: счётчик ссылок на объект и время последнего обращения к нему. При каждом запросе доступа к объекту счётчик ссылок увеличивается на 1 и обновляется время последнего доступа к объекту. По завершении использования объекта счётчик ссылок уменьшается на 1. При удалении объектов из кэша формируется список объектов, счётчик ссылок на которые равен 0, отсортированный по времени последнего доступа. Объекты, включённые в него, последовательно удаляются до тех пор, пока объём занимаемой оперативной памяти не станет меньше допустимого.

Для ускорения поиска объектов в L1 кэше используются бинарные деревья. В них в качестве ключа используется хэш, формируемый

по url-адресу загружаемого объекта. Использование хэшей вместо url-адресов обусловлено тем, что адреса всех объектов совпадают в своей левой части, что приводит к снижению эффективности поиска с использованием такого ключа. В то же время первые символы хэшей от url-адресов равномерно распределены по всему алфавиту, а их вычисление увеличивает вычислительные затраты незначительно.

При запуске апплета проверяется доступность для апплета жёсткого диска пользователя. В случае если он недоступен (настройки безопасности Java позволяют пользователю запретить апплету сохранять данные на жёстком диске), кэш L2 отключается. В противном случае выполняется проверка актуальности L2 кэша. Для этого с удалённого сервера скачивается файл версий, хранящий в себе таблицу версий различных категорий кэшируемых объектов (карт населённых пунктов, векторных карт, файлов и др.), после чего производится их сравнение с версиями объектов, хранящихся в L2 кэше. Объекты, принадлежащие к категориям с меньшей версией, чем полученная из файла удаляются из кэша.

При поступлении запроса на начало загрузки потоки-загрузчики действуют по следующему алгоритму:

- в случае если доступен L2 кэш, выполняется поиск объекта в нём. Если объект найден, производится переход к предобработке;
- если объект не найден в L2 кэше, начинается его загрузка с удалённого сервера;
- в случае, если объект упакован, после завершения загрузки выполняется его распаковка;
- производится сохранение объекта в L2 кэше;
- выполняется предварительная обработка. Она включает в себя выделение из файла данных, проверку их корректности и формирование различных объектов на основе прочитанных данных.

Заключение.

Разработанный геопортал был успешно апробирован в рамках проекта MEDEO. Экспериментально была проверена эффективность распараллеливания вычислений – даже при выполнении на 8-и ядерном компьютере вычислительная нагрузка равномерно распределяется по всем ядрам. На тестовом ПК с двухъядерным процессором с тактовой частотой 2ГГц, 3Гб оперативной памяти, видеокартой ATI Radeon HD3600 и подключением к Internet со скоростью 5Мбит/с полностью отсутствуют заметные для пользователя задержки в работе интерфейса при количестве одновременно отображаемых изображений меньше 200. При наложении 100 прореженных снимков (квиклуков) от КА «Ресурс-ДК» объём используемой геопорталом оперативной памяти не

превышает 400Мб. Скорость отображения изображений и векторных карт при этом ограничена только скоростью передачи данных по сети. При первом запуске геопортала с очищенным L2 кэшем к моменту загрузки векторных карт с наименьшей детальностью для всей поверхности Земли от сервера к апплету происходит передача ~ 22.3 Мб данных, при повторном запуске к тому же моменту времени происходит передача ~ 1.6 Мб данных, что указывает на высокую эффективность кэширования. При скорости сетевого подключения в 5Мбит/с время передачи данных равно ~ 45 с и ~ 3 с соответственно. При переходе от минимального уровня детализации к максимальному и отображении 10-и квилкуков объём переданных данных составляет ~ 11.3 Мб, из которых ~ 10 Мб приходится на данные изображений, а время загрузки ~ 23 с. При этом, благодаря многопоточной архитектуре, пользователь может продолжать взаимодействие с геопорталом во время загрузки данных. При использовании же традиционного однопоточного приложения в течение этого времени геопортал не отвечал бы на запросы пользователя.

На рис. 5 и 6 показаны скриншоты разработанного геопортала.

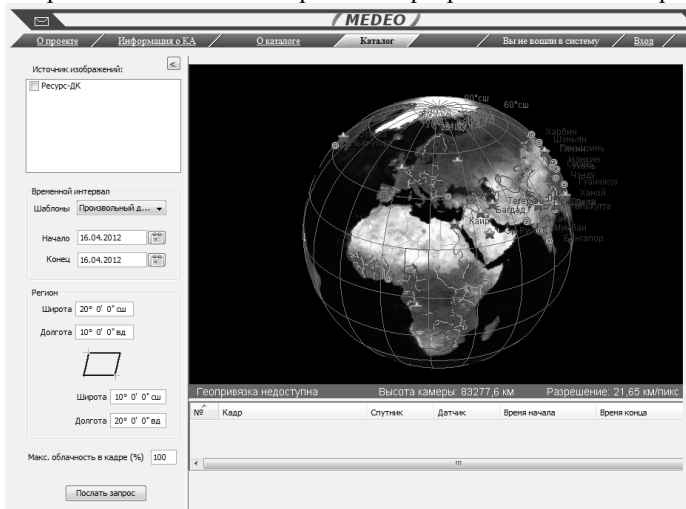


Рис. 5. Общий вид геопортала MEDEO

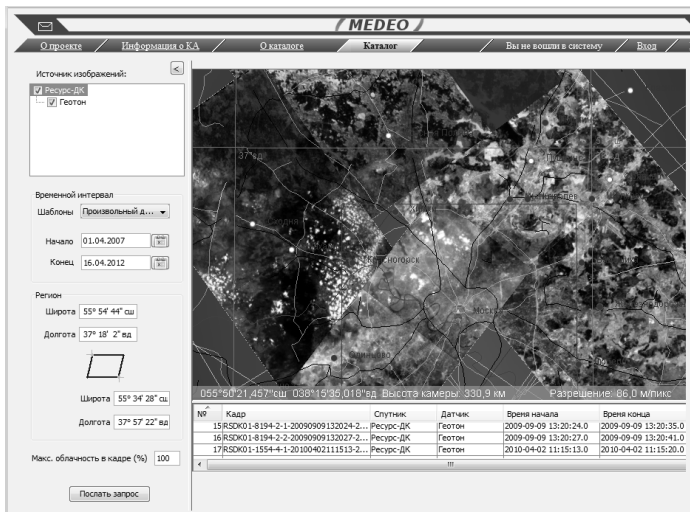


Рис. 6. Геопортал с наложенным множеством разновременных снимков

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Электронная публикация <http://blog.terzza.com/linux-compression-comparison-gzip-vs-bzip2-vs-lzma-vs-zip-vs-compress/>

М.Е. ВИНОГРАДОВА

Рязанский государственный радиотехнический университет

ИТЕРАТИВНАЯ ПРОЦЕДУРА СНИЖЕНИЯ РАЗМЕРНОСТИ ПРОСТРАНСТВА

Рассматриваются вопросы построения итеративной процедуры снижения размерности пространства признаков для решения задачи классификации акустических шумов и разбиения пространства признаков акустических шумов по результатам работы процедуры на примере акустических шумов автомобильных двигателей.

При решении задач классификации приходится сталкиваться с тем, что размерность исходного пространства признаков может доходить до десятков тысяч и более [1]. В таких условиях решение задачи построения решающего правила для отнесения объектов к тому или иному классу оказывается вычислительно сложной задачей для любого метода классификации. Кроме того, исходное множество описаний

класса может содержать признаки не эффективные с точки зрения возможности классификации, которые могут быть отброшены.

В связи с тем, что для акустических шумов автомобильных двигателей при работе в режиме холостого хода характерно малое изменение спектральной динамики с течением времени, в качестве исходного набора признаков предложено использовать математическое ожидание нормированного логарифма спектральной плотности мощности, (СПМ) полученной периодограммным методом [2] для обеспечения гладкой и статистически устойчивой оценки, усредненной в каждой точке по k отсчетам при помощи свертки с окном W . Анализ экспериментальных данных показал, что спектральные составляющие рассматриваемого типа сигналов находятся в полосе частот 20 ... 22000 Гц. В связи с чем, для получения периодограммной оценки СПМ целесообразно выбирать длительность окна анализа не менее 0,1 с [3]. Таким образом, при частоте дискретизации 44100 Гц была выбрана ширина окна анализа 8192 точки, что составляет 0,186 с во временной области. Соответственно, размерность исходного пространства признаков с учетом симметрии СПМ составила $8192/2 = 4096$.

Блок-схема алгоритма предложенной процедуры показана на рисунке 1.

1) На пространстве признаков A сигналов M классов рассматривается множество $A_J, J \subset M$ сигналов класса J и множество $R = A_K, K \subset M \cap \bar{J}$, где K - множество сигналов из M классов, включающее в себя все сигналы, кроме сигналов класса J .

2) Для i -го признака j -го класса производится оценка распределений и определяется динамический диапазон $(P_{\min}^{ij} \dots P_{\max}^{ij})$. Далее в i ю ячейку массива m заносится число элементов множества $A_K, K \subset (M \cap \bar{J}) \cap ((P_{\min}^{ij} < A_J^i) \cap (P_{\max}^{ij} > A_J^i))$ сигналов, не пересекающегося со множеством $A_J(i)$ распределения i -го признака j -го класса.

Вычисляется номер ячейки, содержащей максимальное значение массива m , записывается в массив I , и определяется множество C :

$$C = A_K, K \subset (M \cap \bar{J}) \cap ((P_{\min}^{Ij} < A_J^I) \cap (P_{\max}^{Ij} > A_J^I)).$$

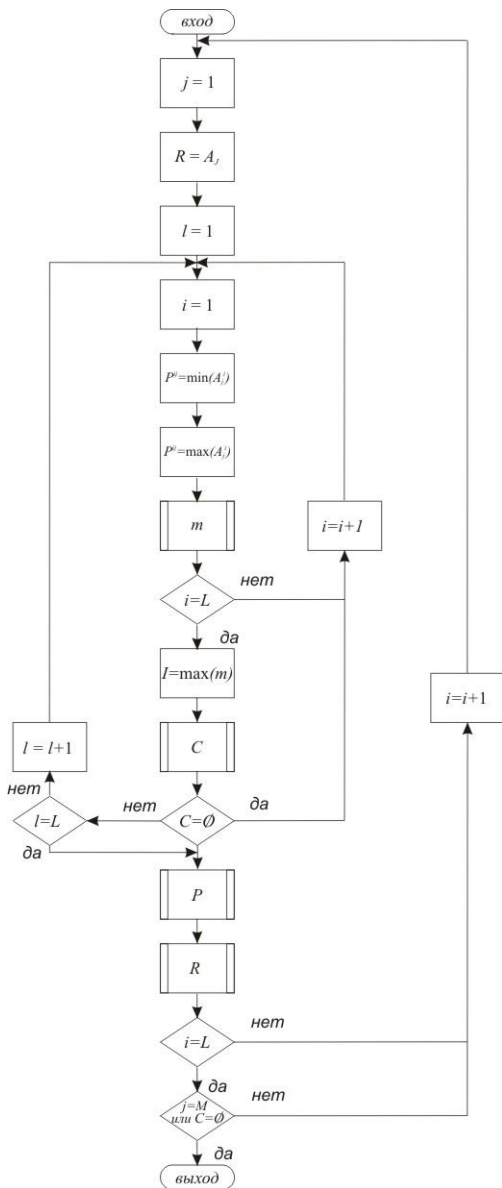


Рис. 1. Блок-схема алгоритма итеративной процедуры снижения размерности пространства

Если множество S оказывается не пустым, оно исключается из дальнейшего рассмотрения, т.е. на следующей итерации множество R будет определяться как:

$$R = A_K, K \subset (M \cap \bar{J}) \cap ((P_{\min}^{ij} < A_K^i) \cap (P_{\max}^{ij} > A_K^i)).$$

Значения $P_{\min}^{ij}, P_{\max}^{ij}$ записываются в массив P .

Этап 2 процедуры последовательно выполняется для всех i признаков j -го класса L раз и завершается, если множество R оказывается пустым.

3) Этапы 1...3 процедуры выполняются для каждого класса сигналов $j = 1, 2, \dots, M$.

Входными данными для процедуры являются пространство признаков (A) и множество всех классов (M). На выходе получаем массив значений динамических диапазонов P и вектор I по которым может быть произведена классификация.

В ходе работы был получен массив фонограмм 10 секундных записей работы двигателя в режиме холостого хода при трех различных положениях устройства звукозаписи для 34 автомобилей различных марок (легковых, грузовых, дизельных, карбюраторных, инжекторных). Полученный массив был представлен сегментами длительностью 0,74 с и параметризован в соответствии с предложенным способом.

Предложенная процедура позволила снизить размерность пространства признаков с 4096 до 122, разделить пространство признаков акустических шумов для 34 классов автомобилей для 97,2% сигналов обучающей выборки и классифицировать сигналы тестовой выборки с вероятностью правильной классификации 91,2%.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Аграновский А.В., Леднов Д.А. Теоретические аспекты алгоритмов обработки и классификации акустических сигналов, М.: Радио и связь, 2004. – 164 с.
2. Марпл-мл. С.Л. Цифровой спектральный анализ и его приложения. – М.: Мир, 1990. – 584 с.
3. Шелухин И.О., Лукьянцев Н.Ф. Цифровая обработка и передача речи. М.: Радио и связь 2000. – 456 с.

А.А. ВОРОНКОВ, Д.А. ПЕРЕПЕЛКИН

Рязанский государственный радиотехнический университет

ПОИСК КРАТЧАЙШИХ ПУТЕЙ В КОРПОРАТИВНЫХ СЕТЯХ НЕСКОЛЬКИХ ПРОВАЙДЕРОВ

Рассмотрена задача поиска кратчайших путей в корпоративных сетях нескольких провайдеров.

В настоящее время большой класс распределенных и телекоммуникационных сетевых структур оптимально строить на основе базовых (опорных) сетей. Такие задачи актуальны при формировании региональной телекоммуникационной инфраструктуры с учетом особенностей действующих на территории региона операторов связи. Это дает возможность сократить общие затраты на создание высокоскоростных каналов связи, построить устойчивую сетевую структуру, к которой легко добавлять новые сегменты с учетом специфики предприятия.

Для обеспечения этих условий необходимо сформировать и построить оптимальные маршруты передачи данных. В общем случае корпоративная сеть состоит из множества базовых (коммуникационных) узлов, к которым подключаются пользователи сети, соединенных скоростными каналами связи. Стоимость или цена маршрута складывается из стоимости используемых каналов связи и базовых узлов.

Задача формирования и построения оптимальных маршрутов передачи данных часто формулируется как задача поиска кратчайших путей на графе. При этом вершинами графа являются базовые узлы, а ребрами – каналы связи, соединяющие эти базовые узлы. В настоящее время известно большое число алгоритмов поиска кратчайших путей.

Наиболее эффективный алгоритм для решения задачи поиска кратчайших путей предложил Дейкстра. Данный алгоритм основан на присвоении вершинам временных пометок, определяющих верхнюю границу длины пути от начальной вершины до всех остальных, которые уменьшаются с помощью некоторой итерационной процедуры; после выполнения каждой итерации одна из пометок становится постоянной, что указывает на то, что данная пометка обозначает точную длину пути от начальной вершины до вершины с этой пометкой. Алгоритм заканчивает свою работу, когда просмотрены все вершины или когда найден путь от начальной вершины до искомой.

Однако в реальных случаях задача поиска кратчайших путей в корпоративной сети осложняется наличием нескольких альтернатив-

ных вариантов реализации, что приводит к необходимости стыковки каналов в узлах сети и возникновению зависимости стоимости узла от подключаемых к нему каналов связи. Примером может служить задача оптимального построения базовой сети региона при наличии нескольких операторов связи с различными зонами покрытия.

В общем случае для решения данной задачи применяется графовая модель корпоративной сети, в которой множество вершин графа соответствует множеству базовых узлов, а множество ребер соответствует возможным каналам связи между базовыми узлами. Каждое ребро, соответствующее каналу связи также имеет свой вес. На практике весу ребра могут соответствовать стоимость аренды канала связи, затраты на оплату единицы трафика, передаваемого по каналу связи, соответствующему данному ребру, либо более сложная функция, учитывающая большее число параметров корпоративной сети. При нескольких провайдерах связи между отдельными базовыми узлами возможно наличие нескольких каналов связи, что соответствует нескольким ребрам, связывающим соответствующие вершины графа.

Рассмотрим задачу поиска кратчайших путей в корпоративной сети нескольких провайдеров. Чтобы модифицировать алгоритм Дейкстры применительно к данной формулировке задачи, достаточно ввести временные метки по каждому провайдеру и уменьшать их в соответствии с классическим алгоритмом Дейкстры, отличие будет заключаться лишь в выборе постоянной пометки, равной

$$\min_{i=1..K} \{d_j^i\},$$

где d – метка вершины с номером j по провайдеру с номером i ($i=1..K$, $j=1..N$, где K – число провайдеров, N – число вершин или маршрутизаторов в корпоративной сети).

Рассмотрим модифицированный алгоритм Дейкстры по шагам.

Пусть d_j^i – пометка вершины v_j по провайдеру i , v_s – начальная вершина, v_p – текущая вершина.

Шаг 1. Присвоение начальных значений. Положить $d^{i*}(v_s)=0$ и считать эту пометку постоянной. Положить $d^i(v_j)=\infty$ (бесконечность) для всех $v_j \neq v_s$ и считать эти пометки временными. Положить $p=s$.

Шаг 2. Обновление пометок. Для всех вершин $v_j \in G(v_p)$, инцидентных вершине v_p и имеющих временные пометки, изменить пометки в соответствии с выражением:

$$d_j^i = \min \{d_j^i, \min_{i=1..K} \{d_p^i\} + w_{pj}^i\} \quad (1)$$

где d_p^i – метка текущей вершины, d_j^i – метка вершины, смежной с текущей вершиной, w_{pj}^i – вес ребра, соединяющего вершину v_p с вершиной v_j .

Шаг 3. Превращение пометки в постоянную. Среди всех вершин с временными пометками найти такую, для которой

$$d_j^* = \min_{i=1 \dots K} \{d_j^i\} \quad (2)$$

Шаг 4. Считать пометку вершины v_j^* постоянной и положить $v_p = v_j^*$.

Шаг 5. Если все вершины имеют постоянные пометки, то конец алгоритма. Иначе перейти к шагу 2.

Сам маршрут можно найти, применяя рекурсивно процедуру, в которой реализовано выражение:

$$d_k^* = \min_{i=1 \dots K} \{d_j^{i*}\} + w_{jk}^i \quad (3)$$

Данное выражение справедливо для последующей и предыдущей вершин, принадлежащих одному пути.

Рассмотрим пример корпоративной сети двух провайдеров ($K=2$). Представим корпоративную сеть в виде неориентированного взвешенного связного графа $G = (V, E, W)$, показанного на рис. 1, где V – множество вершин графа или базовых узлов в сети, E – множество ребер графа или каналов связи между базовыми узлами, W – множество весов ребер; причем $E = \{E_1, E_2\}$, где E_i – подмножество ребер i -го провайдера, $W = \{W_1, W_2\}$, где W_i – подмножество весов ребер i -го провайдера, представляющее собой матрицу смежности.

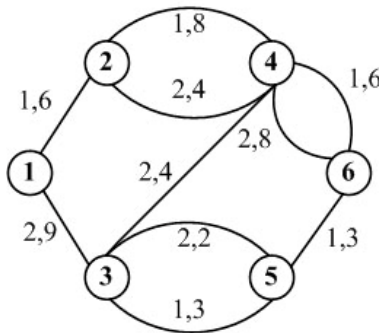


Рис. 1. Граф G корпоративной сети

Пусть d^1 и d^2 – векторы меток вершин по первому и второму провайдеру соответственно, L – вектор вершин с постоянными пометками: 1 – вершина имеет постоянную пометку, 0 – вершина имеет временную пометку.

Шаг 1. Инициализация: пусть начальной вершиной является v_1 , тогда положим $d_1^1 = 0$ и $d_1^2 = 0$, для вершин $d_j^i \neq d_j^i$, $d_j^i = \infty$. Так как первую вершину считаем уже просмотренной, то $l_1=1$, $l_i=0$ для $i=2..N$.

$$d^1 = [0, \infty, \infty, \infty, \infty, \infty]$$

$$d^2 = [0, \infty, \infty, \infty, \infty, \infty]$$

$$L = [1, 0, 0, 0, 0, 0]$$

Первая итерация.

Шаг 2. $G(v_p) = G(v_1) = \{v_2, v_3\}$ – все пометки временные. По формуле (1) получим оценки по каждому провайдеру для смежных вершин:

$$d_2^1 = \min\{\infty, 0 + 6\} = 6;$$

$$d_3^2 = \min\{\infty, 0 + 9\} = 9.$$

Получили следующие векторы меток:

$$d^1 = [0, 6, \infty, \infty, \infty, \infty];$$

$$d^2 = [0, \infty, 9, \infty, \infty, \infty].$$

Шаг 3. По формуле (2) определяем постоянную пометку:

$$d_j^* = \min_{i=1..2} \{d_2^i, d_3^i\} = \min_{i=1..2} \{6, 9\} = 6.$$

Шаг 4. Вершина v_2 получает постоянную пометку, $v_p = v_2$.

$$L = [1, 1, 0, 0, 0, 0]$$

Шаг 5. Так как не все вершины имеют постоянные пометки, то переход на Шаг 2.

Вторая итерация.

Шаг 2. $G(v_p) = G(v_2) = \{v_4\}$ – все пометки временные. По формуле (1) получим оценки по каждому провайдеру для смежных вершин:

$$d_4^1 = \min\{\infty, 6 + 8\} = 14;$$

$$d_4^2 = \min\{\infty, 9 + 4\} = 10.$$

Получили следующие векторы меток

$$d^1 = [0, 6, \infty, 14, \infty, \infty];$$

$$d^2 = [0, \infty, 9, 10, \infty, \infty].$$

Шаг 3. По формуле (2) определяем постоянную пометку:

$$d_j^* = \min_{i=1..2} \{d_3^i, d_4^i, d_4^i\} = \min_{i=1..2} \{9, 10, 14\} = 9$$

Шаг 4. Вершина v_3 получает постоянную пометку, $v_p = v_3$.

$$L = [1, 1, 1, 0, 0, 0].$$

Шаг 5. Так как не все вершины имеют постоянные пометки, то переход на Шаг 2.

Третья итерация.

Шаг 2. $G(v_p) = G(v_3) = \{v_4, v_5\}$ – все пометки временные. По формуле (1) получим оценки по каждому провайдеру для смежных вершин:

$$d_4^2 = \min\{10, 9 + 4\} = 10$$

$$d_5^1 = \min\{\infty, 9 + 3\} = 12$$

$$d_5^2 = \min\{\infty, 9 + 2\} = 11$$

Получили следующие векторы меток

$$d^1 = [0, 6, \infty, 14, 12, \infty];$$

$$d^2 = [0, \infty, 9, 10, 11, \infty].$$

Шаг 3. По формуле (2) определяем постоянную пометку:

$$d_j^* = \min_{i=1..2} \{d_4^1, d_4^2, d_5^1, d_5^2\} = \min_{i=1..2} \{10, 14, 11, 12\} = 10.$$

Шаг 4. Вершина v_3 получает постоянную пометку, $v_p = v_4$.

$$L = [1, 1, 1, 1, 0, 0].$$

Шаг 5. Так как не все вершины имеют постоянные пометки, то переход на Шаг 2.

Четвертая итерация

$$G(v_4) = \{v_6\}$$

$$d_6^1 = \min\{\infty, 10 + 6\} = 16$$

$$d_6^2 = \min\{\infty, 10 + 8\} = 18$$

$$d^1 = [0, 6, \infty, 14, 12, 16]$$

$$d^2 = [0, \infty, 9, 10, 11, 18]$$

$$v_p = v_5$$

$$L = [1, 1, 1, 1, 1, 0]$$

Пятая итерация

$$G(v_5) = \{v_6\}$$

$$d_6^1 = \min\{16, 11 + 3\} = 14$$

$$d^1 = [0, 6, \infty, 14, 12, 14]$$

$$d^2 = [0, \infty, 9, 10, 11, 18]$$

$$v_p = v_6$$

$$L = [1, 1, 1, 1, 1, 1].$$

Все вершины имеют постоянные метки, конец алгоритма. Минимальные метки вершин по провайдерам и являются оптимальными длинами путей от начальной вершины до всех остальных.

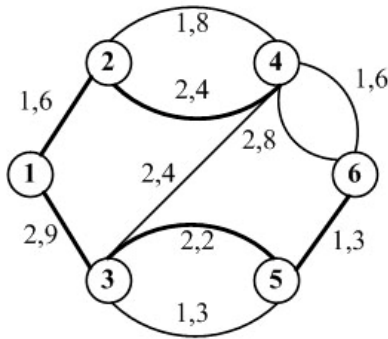


Рис. 2. Кратчайшие пути графа G корпоративной сети

Найдем кратчайший путь $\mu(v_1, v_4)$ от вершины v_1 до v_4 .

Итерация первая

По выражению (3) определим номер провайдера, относительно которого нужно вычислять предшествующую вершину

$$d_4^* = \min_{i=1..2} \{d_4^1, d_4^2\} = \min_{i=1..2} \{d_j^{i*}\} + w_{jk}^i = \min_{i=1..2} \{14, 10\} = \min_{i=1..2} \{d_2^{i*}\} + w_{24}^i = 6 + 4$$

Включаем вершину v_2 в путь. $\mu(v_1, v_4) = (v_2, v_4)$. Так как начальная вершина не достигнута, то переход к следующей итерации.

Итерация вторая

$$d_3^* = \min_{i=1..2} \{d_3^1, d_3^2\} = \min_{i=1..2} \{d_j^{i*}\} + w_{jk}^i = \min_{i=1..2} \{6, \infty\} = \min_{i=1..2} \{d_1^{i*}\} + w_{13}^i = 0 + 6$$

Включаем вершину v_1 в путь $\mu(v_1, v_4) = (v_1, v_3, v_4)$. Начальная вершина достигнута, конец алгоритма. Кратчайшие пути из начальной вершины v_1 в другие вершины показаны на рис. 2 жирными линиями.

По сравнению с классическим алгоритмом Дейкстры предложенный алгоритм имеет большую вычислительную сложность, порядка $O(N^3)$ за счет добавления циклов поиска минимальной пометки вершины по каждому провайдеру, затраты памяти на хранение информации о вершинах и ребрах растет пропорционально числу провайдеров в корпоративной сети. Тем не менее, данная модификация алгоритма Дейкстры позволяет эффективно выполнять поиск кратчайших путей в корпоративных сетях нескольких провайдеров.

И.И. ГЕНЕРАЛЬСКИЙ, И.М. ГЕНЕРАЛЬСКИЙ

Рязанский государственный радиотехнический университет

**ВАРИАНТНАЯ МЕТОДОЛОГИЯ ФОРМИРОВАНИЯ
ПРОИЗВОДСТВЕННОЙ ПРОГРАММЫ В УСЛОВИЯХ
ПОСТОЯННЫХ ИЗМЕНЕНИЙ***Анализ на основе вариантного подхода с применением ИТ технологий.*

Развитию рыночных отношений для отечественного производственного сектора способствуют практики внедрения на действующих предприятиях (ООО Рязанский станкозавод, ОАО Тяжпрессмаш, ОАО Приборный завод и т. д.) новых методов экономического анализа, инструментальных средств на базе ИТ технологий.

Индивидуализация спроса на выпускаемые изделия и последующая работа строго под конкретного заказчика накладывают ограничения на выпуск однотипных изделий. По существу, через производственные линии, настроенные на серийный выпуск, вынужденно запускаются поочередно не координируемые между собой единичные заказы. В условиях рынка это приводит к снижению серийности фактического запуска (на порядок) вплоть до единичного (рис. 1).

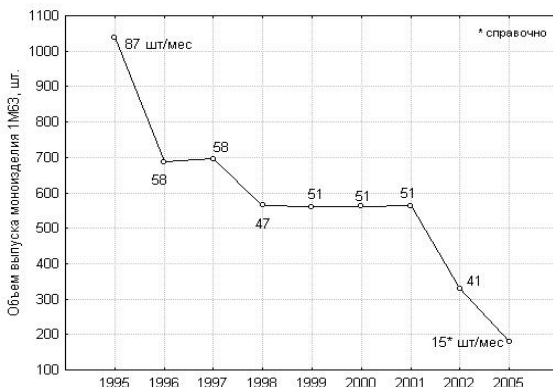


Рис. 1. Изменение серийности запуска, основного в условиях РСЗ, изделия «1М63»

Проблему перехода с серийного выпуска под конкретного заказчика лидеры мирового станкостроения в Германии и США решают планомерно за счет вариантного расчета производственной

программы, одновременно внедряя в технологический процесс новейшее оборудование. Для этого и российским предприятиям понадобится применить у себя передовые методы организации и управления производственными мощностями.

Предлагаемый вариантный подход к формированию плана производства влечет за собой изменение модели расчета ПМ. В целях обеспечения продуктовой гибкости, с учетом рациональной специализации предприятия, предлагается следующее выражение для расчета суммарного времени изготовления высокотехнологичного изделия:

$$T_{\Sigma} = \sum_{q=1}^Q \left\{ \sum_{j=1}^J \left[\sum_{i=1}^I (t_m + t_a)_i + t_j \right] + t_q \right\}$$

где Q – количество видов разнообразных изделий, t_q – время технической подготовки к запуску q -ого изделия. Введение третьей переменной (q) максимально охватывает многочисленные виды потенциальных к выпуску изделий, их изготовлению, обработке, сборке.

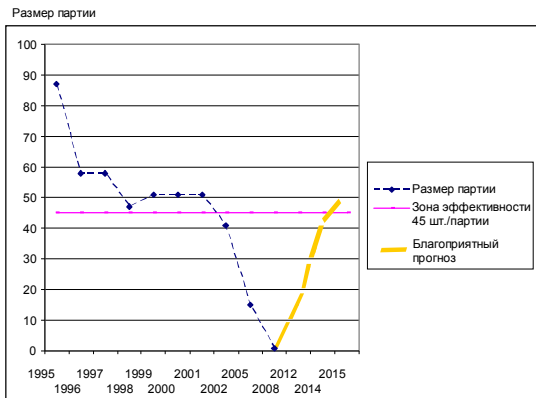


Рис. 2. Динамика падения и перспектива роста в условиях РСЗ

Реализация модели развития сделает реальным отечественное производство станков вместо импорта из развивающихся стран.

Вывод: проведенный анализ позволил выявить зависимость экономического эффекта от оптимального размера партии запуска.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. www.ito.ru
2. www.rsz.ru

Д.В. ГИЛЬМАН, Р.А. ТАГАНОВ

Рязанский государственный радиотехнический университет

АЛГОРИТМЫ ПРЕОБРАЗОВАНИЯ НЕЧЕТКИХ ГРАФОВЫХ МОДЕЛЕЙ ПРОЦЕССНЫХ РИСКОВ ПРОЕКТА

Рассматриваются правила и алгоритмы преобразования нечетких графовых моделей процессных рисков проекта для решения задачи общей оценки рисков процессов проекта.

Для решения задачи оценки рисков заданных процессов проекта необходимо предварительное построение соответствующих математических моделей процессных рисков, адекватно отражающих как структуру элементов рассматриваемых процессов, так и их взаимосвязи с соответствующими параметрами. Для этих целей предлагается использовать нечеткие вероятностные графы, которые широко используются во многих задачах нечеткой оценки надежности алгоритмических процессов [2].

1. Теоретический анализ

Анализ моделей рисков проекта показал [1, 3], что для построения формальных алгоритмов преобразования нечетких графовых моделей процессных рисков проекта, целесообразным является остановиться на представлении нечеткого графа рисков проекта виде так называемой ленточной (L -) матрицы [2].

Определение. L -матрицей назовем матрицу размером $4 \times N$, для которой каждая i -я строка отождествляется с дугой графа и имеет следующий вид:

$$l_i = \{x_i, y_i, \tilde{p}_i, \tilde{t}_i\},$$

где x_i – номер вершины, из которой выходит i -я дуга;

y_i – номер вершины, в которую входит i -я дуга;

$\tilde{p}_i(\tilde{t}_i)$ – нечеткая вероятность (время) перехода из вершины x_i ,

вершину y_i ;

N – количество дуг вероятностного графа ($i = \overline{1, N}$).

Основной замысел, лежащий в основе рассматриваемых алгоритмов эквивалентного преобразования процессных моделей рисков проекта, заключается в укрупнении нечеткого графа с целью получения итоговой модели процессного риска, отражающей итоговое значение параметра риска процесса. Такие преобразования состоят в последовательном применении правил объединения параллельных дуг гра-

фа, удаления дуги-петли и удаления вершины без петли. При этом правила распознавания укрупняемых фрагментов графа с нечеткими весами дуг, являются обобщением аналогичных правил для обычного вероятностного графа [1, 2]. Эти правила преобразования, в ходе укрупнения нечеткого вероятностного графа, рассматриваются здесь для представления нечеткого графа в форме ленточной (L -) матрицы.

2. Алгоритмы преобразования моделей процессных рисков

2.1. Удаление вершин без петель в нечеткой модели процессных рисков

Обозначим удаляемую вершину через j . Тогда дугам, входящим в эту вершину соответствуют строки L -матрицы, второй элемент которых равен j . Обозначим эти строки через a_1, a_2, \dots, a_m , где m – количество дуг, входящих в вершину j . Дугам, выходящим из вершины j соответствуют строки L -матрицы, первый элемент которых равен j .

Обозначим эти строки через b_1, b_2, \dots, b_d , где d – количество дуг, выходящих из вершины j .

Тогда учитывая правила и формулы [2], можно представить следующий алгоритм удаления вершины без петли.

Алгоритм 1. Удаление вершины без петли

Шаг 1. Выделить множество $A = \{a_1, a_2, \dots, a_m\}$ дуг, входящих в вершину j , где $a_r = \{x_r, y_r, \tilde{p}_r, \tilde{t}_r\}$ – строка L -матрицы, $y_r = j$; $r = \overline{1, m}$.

Шаг 2. Выделить множество $B = \{b_1, b_2, \dots, b_d\}$ дуг, выходящих из вершины j , где $b_z = \{x_z, y_z, \tilde{p}_z, \tilde{t}_z\}$ – строка L -матрицы, $x_z = j$; $z = \overline{1, d}$.

Шаг 3. Вычеркнуть из L -матрицы строки, входящие в множество A или B .

Шаг 4. Образовать множество $C = A \times B$, каждый элемент которого представляет собой строку L -матрицы:

$$c_{rz} = \{x_{rz}, y_{rz}, \tilde{p}_{rz}, \tilde{t}_{rz}\},$$

где $x_{rz} = x_r$; $y_{rz} = y_r$;

$\tilde{p}_{rz} = \tilde{p}_r \cdot \tilde{p}_z$ – определяется по формуле:

$$\tilde{p}_{ij} = \bigcup_{\alpha \in [0,1]} (\underline{p}_{ij\alpha}, \overline{p}_{ij\alpha});$$

$\tilde{t}_{rz} = \tilde{t}_r + \tilde{t}_z$ – определяется по формуле:

$$\tilde{t}_{ij} = \bigcup_{\alpha \in [0,1]} (\underline{t}_{ij\alpha}, \bar{t}_{ij\alpha}),$$

$$\text{где } \bar{p}_{ij\alpha} = \bar{p}_{ik\alpha} + \bar{p}_{kj\alpha}; \underline{p}_{ij\alpha} = \underline{p}_{ik\alpha} + \underline{p}_{kj\alpha};$$

$$\bar{t}_{ij\alpha} = \bar{t}_{ik\alpha} + \bar{t}_{kj\alpha}; \underline{t}_{ij\alpha} = \underline{t}_{ik\alpha} + \underline{t}_{kj\alpha}.$$

Шаг 5. Добавить в L -матрицу строки, входящие в множество S .

2.2. Объединение параллельных дуг в нечеткой модели процессных рисков

Признаком параллельных дуг является наличие на L -матрице строк, у которых первые два элемента попарно равны.

Учитывая правила и формулы [2] можно представить следующий алгоритм для объединения параллельных дуг.

Алгоритм 2. Объединение параллельных дуг

Шаг 1. Найти на L -матрице две строки

$$\{x_i, y_i, \tilde{p}_i, \tilde{t}_i\},$$

$$\{x_j, y_j, \tilde{p}_j, \tilde{t}_j\}$$

такие, что $x_i = x_j$ и $y_i = y_j$.

Шаг 2. Заменить две строки, найденные на шаге 1, одной эквивалентной строкой

$$\{x, y, \tilde{p}, \tilde{t}\},$$

где $x = x_i$; $y = y_i$;

$\tilde{p} = \tilde{p}_i + \tilde{p}_j$ – определяется по формуле:

$$\tilde{p}_{ij} = \bigcup_{\alpha \in [0,1]} (\underline{p}_{ij\alpha}, \bar{p}_{ij\alpha});$$

$\tilde{t} = \frac{\tilde{p}_i \tilde{t}_i + \tilde{p}_j \tilde{t}_j}{\tilde{p}_i + \tilde{p}_j}$ – определяется по формуле:

$$\tilde{t}_{ij} = \bigcup_{\alpha \in [0,1]} (\underline{t}_{ij\alpha}, \bar{t}_{ij\alpha}),$$

где $\underline{p}_{ij\alpha} = \underline{p}'_{ij\alpha} + \underline{p}''_{ij\alpha}$; $\bar{p}_{ij\alpha} = \min(1, \bar{p}'_{ij\alpha} + \bar{p}''_{ij\alpha})$;

$$\underline{t}_{ij\alpha} = \frac{p_1 \underline{t}'_{ij\alpha} + p_2 \underline{t}''_{ij\alpha}}{p_1 + p_2}; \bar{t}_{ij\alpha} = \frac{p_3 \bar{t}'_{ij\alpha} + p_4 \bar{t}''_{ij\alpha}}{p_3 + p_4};$$

$$p_1 = \begin{cases} \underline{p}'_{ij\alpha}, & \underline{t}'_{ij\alpha} \geq \underline{t}''_{ij\alpha} ; \\ \underline{p}''_{ij\alpha}, & \underline{t}'_{ij\alpha} < \underline{t}''_{ij\alpha} ; \end{cases}$$

$$p_2 = \begin{cases} \overline{p}''_{ij\alpha}, & \overline{t}''_{ij\alpha} \geq \overline{t}'_{ij\alpha} ; \\ \overline{p}'_{ij\alpha}, & \overline{t}''_{ij\alpha} < \overline{t}'_{ij\alpha} ; \end{cases}$$

$$p_3 = \begin{cases} \overline{p}'_{ij\alpha}, & \overline{t}'_{ij\alpha} \geq \overline{t}''_{ij\alpha} ; \\ \underline{p}'_{ij\alpha}, & \overline{t}'_{ij\alpha} < \overline{t}''_{ij\alpha} ; \end{cases}$$

$$p_4 = \begin{cases} \overline{p}''_{ij\alpha}, & \overline{t}''_{ij\alpha} \geq \overline{t}'_{ij\alpha} ; \\ \underline{p}''_{ij\alpha}, & \overline{t}''_{ij\alpha} < \overline{t}'_{ij\alpha} . \end{cases}$$

2.3. Удаление дуги-петли в нечеткой модели процессных рисков

Признаком дуги-петли при вершине j есть строка L -матрицы, у которой первый и второй элементы равны j . Дугам, выходящим из вершины j , соответствуют строки, у которых первый элемент равен j .

Учитывая правила и формулы [2] можно представить следующий алгоритм удаления дуги-петли при вершине j .

Алгоритм 3. Удаление дуги-петли

Шаг 1. Определить строку $s = \{x, y, \tilde{p}, \tilde{t}\}$, для которой $x=y=j$.

Шаг 2. Выделить множество $A = \{a_1, a_2, \dots, a_m\}$ дуг, выходящих из вершины j , где $a_r = \{x_r, y_r, \tilde{p}_r, \tilde{t}_r\}$ – строка L -матрицы, $x_r = j$; $r = \overline{1, m}$, причем $s \notin A$.

Шаг 3. Вычеркнуть из L -матрицы строку s и строки, входящие в множество A .

Шаг 4. Каждую строку a_r , найденную на шаге 2, заменить строкой

$$a_r = \{x'_r, y'_r, \tilde{p}'_r, \tilde{t}'_r\},$$

где $x'_r = x_r$; $y'_r = y_r$;

$$\tilde{p}'_r = \frac{\tilde{p}_r}{1 - \tilde{p}} \text{ – определяется по формуле:}$$

$$\tilde{p}_{ij} = \bigcup_{\alpha \in [0,1]} (p_{-ij\alpha}, \overline{p}_{ij\alpha}).$$

$$\tilde{t}'_r = t_r + \frac{\tilde{t} \cdot \tilde{p}_r}{1 - \tilde{p}} - \text{определяется по формуле:}$$

$$\tilde{t}'_{ij} = \bigcup_{\alpha \in [0,1]} (t_{ij\alpha}, \overline{t}_{ij\alpha}),$$

$$\text{где } \underline{p}'_{ij\alpha} = \frac{p_{ij\alpha}}{1 - p_{ii\alpha}}; \quad \overline{p}'_{ij\alpha} = \min(1, \frac{\overline{p}_{ij\alpha}}{1 - p_{ii\alpha}});$$

$$\underline{t}'_{ij\alpha} = t_{ij\alpha} + \frac{t_{ij\alpha} \cdot p_{ii\alpha}}{1 - p_{ii\alpha}}; \quad \overline{t}'_{ij\alpha} = \overline{t}_{ij\alpha} + \frac{\overline{t}_{ij\alpha} \cdot \overline{p}_{ii\alpha}}{1 - \overline{p}_{ii\alpha}};$$

$$r = \overline{1, m}.$$

Шаг 5. Добавить в L -матрицу строки, найденные на шаге 4.

Алгоритм укрупнения нечеткого вероятностного графа, использующий правила 2.1-2.3, состоит в следующем.

3. Обобщенный алгоритм укрупнения нечеткого графа процессных рисков

Шаг 1. Построить граф $\tilde{G}(X, \Gamma)$ с нечеткими весами дуг.

Шаг 2. Если есть параллельные дуги, то объединить их по алгоритму 2.

Шаг 3. Если $\pi = \emptyset$, то взять промежуточную вершину из множества π , иначе перейти к шагу 8.

Шаг 4. Если промежуточная вершина с петлей, то перейти к шагу 5, иначе – к шагу 6.

Шаг 5. Удалить петлю по алгоритму 3.

Шаг 6. Удалить промежуточную вершину по алгоритму 1.

Шаг 7. Перейти к шагу 2.

Шаг 8. Записать результат в виде нечетких весов дуг укрупненного графа $\tilde{G}^*(X, \Gamma)$.

Заключение

Представленные в работе алгоритмы преобразования нечетких графовых моделей процессных рисков проекта основываются на использовании положений метода нечеткого анализа надежности алгоритмических процессов [2] и по своей сути являются альтернативой

известному методу моделирования надежности с помощью аппарата полумарковских процессов, в котором неопределенность исходных данных описывается произвольными законами распределений времени переходов между вершинами. При этом приведенные в литературе результаты исследований в указанной области показывают, что вычислительная трудоемкость алгоритмов укрупнения нечетких графовых моделей значительно ниже, чем в случае применения теории полумарковских процессов и решетчатых функций. Это обстоятельство определило выбор нечетких графов в качестве моделей для описания моделей процессных рисков проекта с последующим решением задачи построения нечеткой системы поддержки процесса анализа и сокращения процессных рисков проекта [3].

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Гильман Д.В. Использование нечеткого вероятностного графа для оценки показателя надежности проекта // В сб. трудов Всероссийской научно-технической конференции «Новые информационные технологии в научных исследованиях и в образовании». – Рязань: РГРТУ, 2010.
2. Ротштейн А.П., Штовба С.Д. Нечеткая надежность алгоритмических процессов. – Винница: Континент, 1997. – 142 с.
3. Таганов А.И. Методика анализа и сокращения рисков проектов сложных программных систем по характеристикам качества // Вестник РГРТУ. – Рязань, 2010. – Вып. 30. – С. 77-82.

Е.А. ЕГОРОВ

Рязанский государственный радиотехнический университет

КЛАССИФИКАЦИЯ ПОДХОДОВ К СЕМАНТИЧЕСКОМУ ПОИСКУ

Предложена классификация существующих подходов к семантическому поиску.

Введение. В основе любой системы информационного поиска лежит использование логических представлений информационной потребности пользователя и объектов пространства поиска, между которыми производится сравнение. Логическое представление, с одной стороны, должно быть доступным для эффективной машинной обработки, что неизбежно приводит к потере некоторой информации об исходном объекте. С другой стороны, оно должно наиболее полно выражать семантику информационной потребности и документов. Широ-

ко применяемое в области информационного поиска логическое представление в форме неупорядоченного множества ключевых слов недостаточно выразительно, поэтому разрабатываются более абстрактные модели, основанные на концептуальном представлении доступной и искомой информации, которые позволят повысить полноту и точность поиска. Поиск с использованием таких технологий носит название семантического.

Критерии классификации. Существующие подходы предлагаются классифицировать в соответствии со следующим списком критериев (таблица 1).

- **Способ представления знаний.** Выделены три основных: на основе латентно-семантического анализа, на основе лингвистической концептуализации и подходы на основе онтологий. Первые два допускают представление лишь нескольких видов свойств и отношений между концептами. Онтологии позволяют описать предметную область более детально.

- **Область применения.** Семантический поиск используется в самых различных средах, таких как Всемирная паутина, управляемые хранилища и локальные компьютеры. Наиболее сложным является поиск во Всемирной паутине, поскольку это открытое пространство, где информация распределена по миллионам компьютеров, охватывает множество различных предметных областей; содержимое документов часто меняется и стремительно возрастает в объеме; миллионы пользователей ежедневно обращаются к ней для удовлетворения своих самых разнообразных информационных потребностей. Представление содержимого Всемирной паутины в форме концептов является крайне сложной и пока нерешенной задачей. Тем не менее, разработаны подходы для ее частичного решения в изолированных управляемых хранилищах, где доступная информация относится к одной или нескольким предметным областям. Поиск на локальных компьютерах менее трудоемок, поскольку семантическая информация может быть легко извлечена из полуструктурированных источников, таких как электронные письма, каталоги и т.д., и количество пользователей системы невелико.

- **Цель.** Подходы могут быть разделены по цели поиска: информация или данные. В большинстве в ответ на запрос пользователя возвращаются документы, и поэтому они могут быть охарактеризованы как модели информационного поиска. В подходах на основе онтологий возвращаются экземпляры онтологий и их частей, что позволяет классифицировать их как модели поиска данных.

• **Форма поискового запроса.** Еще одним аспектом, характеризующим модели семантического поиска, является способ, которым пользователь выражает свои информационные потребности. Можно выделить четыре основных подхода, отличающихся постепенным возрастанием уровня формальности и сложности использования. Наиболее традиционным, но также и наименее выразительным, является представление информационной потребности в форме неупорядоченного множества терминов без указания отношений между ними. Следующим по уровню формальности является выражение информационной потребности на естественном языке. Эта форма запроса более выразительна, чем предыдущая, поскольку к ней может быть применен лингвистический анализ, позволяющий извлечь синтаксическую информацию (субъект, объект, предикат и другие детали предложения). Развитием этой формы является управляемый запрос на естественном языке, который содержит теги, представляющие свойства, значения или объекты. Такой тип запроса проще машинно обработать и соотнести с классами, свойствами и отношениями онтологии. Наконец, наиболее формальными являются запросы, выраженные при помощи специализированных языков запросов к онтологиям (RDQL, SPARQL и т.д.). Их выразительная сила позволяет системе автоматически находить точную информацию, удовлетворяющую информационной потребности. [1]

• **Тип искомой информации.** Подходы, относящиеся к поиску информации, можно разделить на системы поиска текстовой информации и поиска мультимедиа. Среди подходов, нацеленных на поиск данных, можно выделить системы поиска XML-документов и те, которые производят поиск отдельных экземпляров и частей онтологий.

• **Ранжирование результатов.** Вопрос ранжирования в моделях на основе онтологий остается открытым. Во многих подходах ранжирование отсутствует, в других оно базируется на традиционных техниках (по ключевым словам), и лишь некоторые модели используют преимущества семантической информации, но в общем случае производится ранжирование не документов, а отдельных экземпляров онтологий или их составных частей.

Таблица 1. Критерии классификации

Критерии	Подходы
Представление знаний	Лингвистическая концептуализация Латентно-семантический анализ Онтологии
Область	Всемирная паутина

применения	Управляемые хранилища Локальные компьютеры
Цель	Поиск информации Поиск данных
Форма запроса	Множество ключевых слов Предложение на естественном языке Управляемое предложение на естественном языке Структурированный запрос
Тип информации	Текстовые документы Мультимедиа документы XML-документы Экземпляры онтологий и их частей
Ранжирование	Отсутствует На основе ключевых слов На основе семантической информации

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. M. Hildebrand, J. van Ossenbruggen, L. Hardman. An analysis of search-based user interaction on the semantic web. Technical report. Centrum voor Wiskunde en Informatica. NL, 2007.

В.В. ЕРЕМЕЕВ, А.А. МАКАРЕНКОВ, А.А. ЮДАКОВ

Рязанский государственный радиотехнический университет

НОВЫЕ ТЕХНОЛОГИИ АВТОМАТИЧЕСКОГО ВЫДЕЛЕНИЯ ГРАНИЦ ОБЪЕКТОВ НА ГИПЕРСПЕКТРАЛЬНЫХ ИЗОБРАЖЕНИЯХ ЗЕМНОЙ ПОВЕРХНОСТИ

Рассматриваются технологии автоматического выделения границ объектов на гиперспектральных изображениях земной поверхности на основе анализа спектральных характеристик. Проведен анализ влияния искажающего действия атмосферы на процесс сегментации объектов на изображении.

В последнее десятилетие в практику дистанционного зондирования Земли стала внедряться гиперспектральная съемочная аппаратура. Она является эволюционным развитием многозональных датчиков и позволяет вести съемку объектов в нескольких сотнях спектральных каналов непрерывно разбивающих диапазоны видимого (длины волн от ~400 нм до ~700 нм), ближнего инфракрасного (длины волн от ~700 нм до ~1000 нм) и коротковолнового инфракрасного (длины волн от

~1000 нм до ~2500 нм) диапазонов излучения. Гиперспектральная съемка производится путем расщепления отраженной от поверхности Земли энергии по длине волны и подачи излучения на фотоприемные элементы. В результате формируется гиперспектральное изображение (ГСИ) $B \in \{B_{kmn}, k = \overline{1, K}\}$, $m = \overline{1, M}$, $n = \overline{1, N}$, где k – номер спектрального диапазона, (m, n) – координаты элемента изображения (его номер по строке и по столбцу), B – энергетическая яркость. Точке изображения с фиксированными координатами (m, n) соответствует спектральная характеристика (СХ) в виде вектора из K значений энергетических яркостей для каждого частотного диапазона: B_{kmn} , $k = \overline{1, K}$, $m = const$, $n = const$. Анализ спектральной характеристики позволяет определить физико-химические свойства объектов, что предоставляет новые возможности по решению важной задачи – автоматизации процесса выделения объектов путем сравнения их СХ.

Выделение границ объектов на изображениях традиционно выполняется путем анализа пространственно-яркостных характеристик, которые в значительной степени подвержены влиянию шума и неравномерности освещения сцены. Знание спектральной характеристики для каждой точки земной поверхности позволяет на принципиально новой основе построить процесс выделения границ объектов на гиперспектральном изображении. С привлечением ГСИ от различных съемочных систем апробирован ряд мер сходства СХ от двух соседних элементов ГСИ: евклидова, корреляционная, спектрально-угловая [1, 2]:

$$\rho_{mn} = \frac{\overline{B_{mn}} \cdot \overline{B_{m+1,n}} - \overline{B_{mn}} \cdot \overline{B_{m+1,n}}}{\sigma_{mn} \cdot \sigma_{m+1,n}}, \quad (1)$$

$$\Delta_{mn} = \sqrt{(B_{mn} - B_{m+1,n})^2}, \quad (2)$$

$$\alpha_{mn} = \arccos \left(\frac{\overline{B_{mn}} \cdot \overline{B_{m+1,n}}}{\sqrt{B_{mn}^2} \cdot \sqrt{B_{m+1,n}^2}} \right), \quad (3)$$

где σ_{mn} и $\sigma_{m+1,n}$ – среднеквадратические отклонения отсчетов спектральных характеристик для точек (m, n) и $(m+1, n)$, а черта над переменными и выражениями означает операцию усреднения по K спектральным диапазонам: $\overline{B_{mn}} = \frac{1}{K} \sum_{k=1}^K B_{mnk}$.

Для применения данных мер при выделении границ объектов необходимо распространить их на случай оценки сходства элемента снимка с его ближайшим окружением. Для этого можно использовать традиционные операторы (градиента, Собела, Кирша и др.), которые являются определенными функционалами от разности яркостей анализируемого элемента и его окружающими. При анализе ГСИ можно использовать те же функционалы, но относительно не разности яркостей, а относительно мер сходства SX (1), (2), (3). Например, корреляционная мера сходства в форме оператора Кирша имеет вид:

$$X = \max\{1, \max_{i=0 \div 7} [5S_i - 3T_i]\}, \quad (4)$$

где $S_i = \rho_i + \rho_{i+1} + \rho_{i+2}$, $T_i = \rho_{i+3} + \rho_{i+4} + \rho_{i+5} + \rho_{i+6} + \rho_{i+7}$, ρ_i – корреляция SX элемента с координатами (m, n) со SX i -го ближайшего элемента ($i=0$ соответствует ближайшему элементу в левом верхнем углу, а отсчет ведется по часовой стрелке), индекс у ρ – вычисляется по модулю 8. Примеры выделения границ на гиперспектральном аэроснимке оператором (4) (справа) и пространственно-яркостным оператором Кирша (слева) представлены на рис. 1. Как видим, частотно-пространственной мера дает более качественный результат.

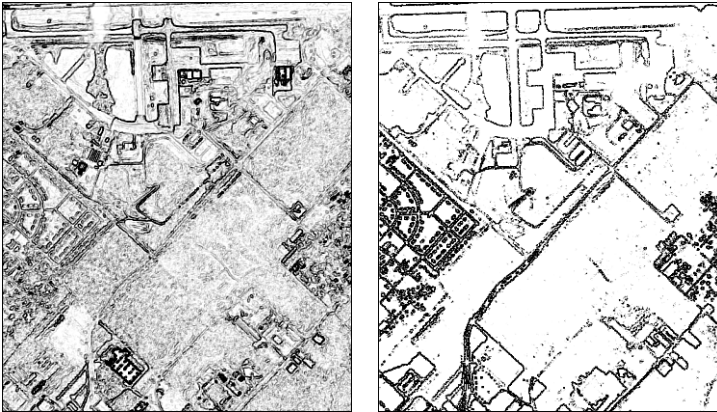


Рис. 1. Выделение границ объектов

Качество частотно-пространственных мер сходства снижается из-за искажающего действия атмосферы. При гиперспектральной съемке датчиком фиксируется излучение Солнца, дважды прошедшее через слой атмосферы. В результате датчик фиксирует SX , описывающую не только физико-химические свойства объекта съемки, но и

состояние атмосферы. В первом приближении искажающее действие атмосферы можно представить в виде мультипликативного воздействия на энергию излучения: $B_{kmn} = S_k B_{kmn}^*$, где B_{kmn} и B_{kmn}^* – энергия, зафиксированная в k -м канале для элемента (m, n) ГСИ при искажающем действии атмосферы и при его отсутствии, S_k – коэффициент пропускания атмосферы в диапазоне длин волн $(\lambda_k', \lambda_k'')$.

На рис. 2 приведен спектр растительности, зафиксированный датчиком Hyperion космического аппарата EarthObserver-1 (США) и спектр растительности без атмосферных искажений. Как видим, атмосферные искажения носят динамичный характер и существенно изменяют форму спектральных характеристик, что ухудшает качество сегментации объектов на основе анализа СХ. Поэтому важной задачей является исключение искажающего действия атмосферы.

Аналитические модели атмосферы требуют знания большого числа параметров атмосферы и метеопараметров, получение которых обычно затруднено. Учитывая, что задача выделения контуров не требует метрического описания скорректированного изображения, более предпочтительной является коррекция атмосферы на основе статистического анализа ГСИ.

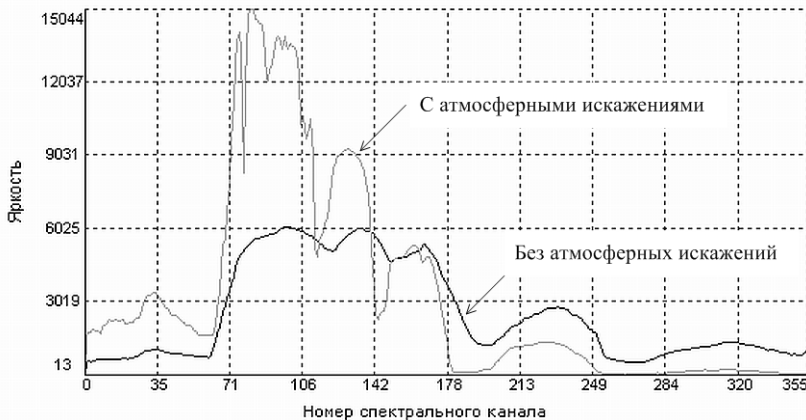


Рис. 2. Спектральная характеристика растительного покрова

Рассмотрим один из таких подходов. Модель формирования ГСИ может быть представлена как

$$B_{mnk} = \int_{\lambda_k'}^{\lambda_k''} G_{mn}(\lambda) [S_k \cdot K_{mk}] d\lambda, \quad k = \overline{1, K},$$

где B_{mnk} – сигнал на выходе k -го канала ГСА, $k = \overline{1, K}$, соответствующий точке (m, n) изображения; $G_{mn}(\lambda)$ – спектральная плотность излучения от точки Земли, соответствующей элементу (m, n) изображения; S_k – коэффициент пропускания атмосферы в направлении надира; K_{km} – функция, описывающая изменение S_k вдоль строки, т.е. по мере изменения m ; λ'_k и λ''_k – левая и правая граница k -го спектрального диапазона. Учитывая близость λ'_k и λ''_k ,

$B_{mnk} = G_{mnk} S_k K_{mk} \Delta_k$, где $\Delta_k = \lambda''_k - \lambda'_k$.



Рис. 3. Выделение границ объектов с учетом и без учета атмосферы

При $k = const$ и $n = const$: G_{mnk} – динамично изменяется при изменении m , K_{mk} – плавная функция от m , $S_k = const$, $\Delta_k = const$. Усредним значения B_{mnk} по столбцу n : $B_{mk} = G_{mk} S_k \cdot K_{mk} \Delta_k$ и образуем отношение $\gamma_{mnk} = \frac{B_{mnk}}{B_{mk}} = \frac{G_{mnk}}{G_{mk}}$, $k = \overline{1, K}$. Получено некое новое

гиперспектральное изображение γ_{mnk} , $k = \overline{1, K}$, свободное от искажающего действия атмосферы и описывающее только наблюдаемую сцену. На рис. 3 представлен результат выделения границ объектов на ГСИ оператором (4): слева – по исходному ГСИ, справа – по изображению γ_{mnk} . Как видим, устранение атмосферных искажений повышает качество выделения границ.

Экспериментальные исследования рассмотренных алгоритмов с привлечением ГСИ от авиационных съемочных систем AVIRIS, Pro-

spectir-V (США), НуМар (Австралия), спутникового датчика Hyperion (EarthObserver-1, США) показали существенное повышение качества выделения границ объектов по сравнению с традиционными подходами.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Антонушкина С.В., Еремеев В.В., Макаренков А.А., Москвитин А.Э. Особенности анализа и обработки информации от систем гиперспектральной съемки земной поверхности // Цифровая обработка сигналов. 2010. №4. С.38-43.

2. Yuhas, R.H., Goetz, A. F. H., and Boardman, J. W., 1992, Discrimination among semiarid landscape endmembers using the spectral angle mapper (SAM) algorithm. In Summaries of the Third Annual JPL Airborne Geoscience Workshop, JPL Publication 92-14, vol. 1, pp.147-149.

А.А. ЖИЛЬНИКОВ, Т.А. ЖИЛЬНИКОВ

Академия ФСИИ России

СПОСОБ НЕРАЗРУШАЮЩЕГО ОБЪЕМНОГО ИЗМЕРЕНИЯ ВЕКТОРНОЙ ФУНКЦИИ МАГНИТНОЙ ИНДУКЦИИ НЕОДНОРОДНО РАСПРЕДЕЛЕННОГО В ПРОСТРАНСТВЕ И ПЕРИОДИЧЕСКИ ИЗМЕНЯЮЩЕГОСЯ ВО ВРЕМЕНИ МАГНИТНОГО ПОЛЯ

Рассматривается способ неразрушающего объемного измерения векторной функции магнитной индукции неоднородно распределенного в пространстве и периодически изменяющегося во времени магнитного поля, основанный на последовательно-поступательных перемещениях и поворотах на углы рабочего магнитоизмерительного органа и регистрации индуцируемых в нем напряжений, посредством применения обратного преобразования Радона.

Существующие способы магнитоизмерений исследовательского характера в большинстве своем ориентированы на так называемые однокомпонентные «точечные» измерения в пространстве и имеют узкую специализацию, не позволяя выполнять неразрушающую регистрацию магнитного поля [1]. Тем самым, они предоставляют ограниченные возможности в плане измерения и регистрации магнитных полей, неоднородно распределенных в пространстве (рельефов полей), т.к. одновременно допускают измерения только в одной точке. Однако при детальном исследовании и контроле изменяющихся во времени полей часто возникает необходимость одновременных измерений во

множестве точек пространства в местах, недоступных для механического проникновения [2].

В связи с этим, для решения данной проблемы, предлагается оригинальный способ неразрушающей объемной регистрации значений векторной функции магнитной индукции в каждой точке неоднородно распределенного в пространстве и периодически изменяющегося во времени магнитного поля, основанный на использовании индукционного метода измерения, учитывающего необходимое условие периодичности поля во времени [3].

Суть способа заключается в регистрации в местах, недоступных для механического проникновения, мгновенных объемных состояний распределения магнитного поля. Благодаря такой регистрации в конкретных точках пространства и моменты времени определяются значения векторной функции индукции, получаемые реконструкцией посредством применения обратного преобразования Радона к измеренным проекциям магнитного потока. Эти проекции получают с помощью магниточувствительного рабочего органа в ходе его управляемого пространственного перемещения в объеме измерения способом параллельного формирования исходных проекционных данных.

Магниточувствительный рабочий орган представляет собой плоский контур, на основе катушки индуктивности. Причем его размеры задаются таким образом, чтобы независимо от его положения в объеме измерения исходная векторная функция поля достаточно быстро убывала на границах этого контура.

Скалярное значение магнитного потока Φ_D исходной векторной функции магнитной индукции $B(x, y, z, t)$ через поверхность D , образованную плоскостью контура, есть интеграл по плоскости, перпендикулярной вектору ее нормали \bar{n} , и проходящей на расстоянии s от начала системы координат объема измерения:

$$\Phi_D = \int_{D(s, \bar{n})} \bar{B} d\bar{\sigma}, \quad (1)$$

где $d\bar{\sigma}$ – вектор нормали, выставленный к элементарной площадке плоскости интегрирования и численно равный ее площади. Переменные \bar{n} и s задают положение контура в объеме измерения, причем в метрике сферической системы координат вектор нормали $\bar{n} = (n, \theta, \alpha)$.

Технически регистрация магнитного потока осуществляется благодаря закону Фарадея, согласно которому напряжение $u(t)$, наводимое в контуре, определяется выражением:

$$u(t) = -\frac{d\Phi_D(t)}{dt}, \quad (2)$$

где $d\Phi_D(t)$ – дифференциал потока векторной функции индукции; dt – дифференциал времени.

Благодаря введенному условию периодичности во времени, согласно которому поле точно повторяет свои мгновенные состояния через одинаковые промежутки времени равные периоду, продолжительность регистрации одного мгновенного объемного состояния распределения магнитного поля уже не ограничена по времени длительностью интервала дискретизации, потому что регистрировать состояние можно, осуществляя измерения в пространстве через интервалы времени, кратные периоду. И несмотря на то, что измерения значительно разнесены по времени, все они будут производиться для какого-то одного мгновенного состояния на периоде.

Интегрирование по времени t выраженного из равенства (2) дифференциала $d\Phi_D(t)$ приводит к записи значения магнитного потока:

$$\Phi_D(t) = -\int_{nT}^{nT+t} u(t) dt, \quad (3)$$

где T – период, в течение которого происходит одно полное изменение магнитной индукции; $n = 1, 2, \dots$ – номер периода.

Измерение начинается с процедуры управляемого пространственного перемещения рабочего органа в объеме измерения с получением проекций, которая соответствует прямому преобразованию Радона для трехмерного пространства (рис. 1). Управляемое пространственное перемещение реализует способ параллельного формирования исходных проекционных данных, для которого рабочий орган совершает поступательно-поворотное движение, предполагающее чередование дискретных параллельных перемещений в направлении вектора нормали органа \vec{n} по оси OS и поворотов направления этих перемещений, задаваемых зенитным θ и азимутальным α углами сферической системы координат. Таким образом, дискретные параллельные перемещения многократно повторяются под разными углами, причем для зенитного в интервале от 0 до $\pi/2$, а для азимутального в интервале от 0 до π . В ходе управляемого пространственного перемещения рабочего органа после каждого его дискретного передвижения вдоль оси OS для текущего значения переменной положения s под углами, определенными вектором нормали \vec{n} , регистрируется плоскостная

проекция p – значение интеграла исходной функции индукции B по плоскости D , численно равное с учетом (1) значению пронизывающего магнитного потока $\Phi_D(t)$:

$$p(s, \bar{n}, t) = \int_{D(s, \bar{n})} \bar{B} d\bar{\sigma} = \Phi_D(t). \quad (4)$$

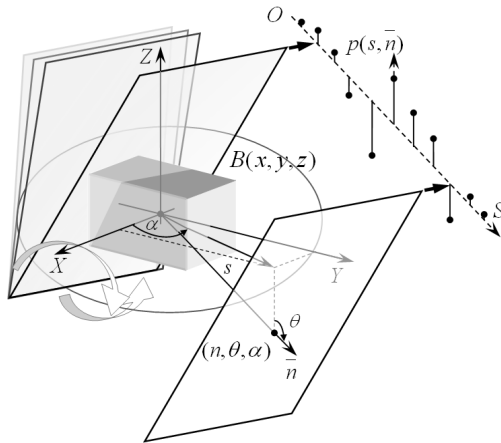


Рис.1. Преобразование Радона

Принимая во внимание выражение (3), на основании (4) проекция значений интеграла исходной функции индукции $p(s, \bar{n}, t)$ с учетом введенной зависимости от переменной времени записывается посредством интегралов по времени напряжений, индуцируемых изменением магнитного потока в соответствии с законом электромагнитной индукции Фарадея в контуре:

$$p(s, \bar{n}, t) = \int_{D(s, \bar{n})} \bar{B} d\bar{\sigma} = - \int_{nT}^{nT+t} u(s, \bar{n}, t) dt. \quad (5)$$

Уравнение (5) приведено для сферической системы координат, при этом в декартовой системе координат (X, Y, Z) выражение компонент данного равенства имеет вид:

$$\begin{cases} p_x(s, \bar{n}, t) = [\sin \theta \cdot p(s, \bar{n}, t) + \cos \theta \cdot p(s, \bar{e}_\theta, t)] \cos \alpha - \\ - [\sin \theta \cdot p(s, \bar{e}_\alpha, t) + \cos \theta \cdot p(s, \bar{e}_{\theta, \alpha}, t)] \sin \alpha; \\ p_y(s, \bar{n}, t) = [\sin \theta \cdot p(s, \bar{n}, t) + \cos \theta \cdot p(s, \bar{e}_\theta, t)] \sin \alpha + \\ + [\sin \theta \cdot p(s, \bar{e}_\alpha, t) + \cos \theta \cdot p(s, \bar{e}_{\theta, \alpha}, t)] \cos \alpha; \\ p_z(s, \bar{n}, t) = \cos \theta \cdot p(s, \bar{n}, t) - \sin \theta \cdot p(s, \bar{e}_\theta, t), \end{cases} \quad (6)$$

где \bar{e}_θ , \bar{e}_α , $\bar{e}_{\alpha, \theta}$ – единичные векторы соответственно коллинеарные перпендикулярно повернутому в направлении углов α и θ вектору нормали \bar{n} плоскости интегрирования, т.е. $\bar{e}_\theta = (n, \theta + \pi/2, \alpha)$, $\bar{e}_\alpha = (n, \theta, \alpha + \pi/2)$ и $\bar{e}_{\theta, \alpha} = (n, \theta + \pi/2, \alpha + \pi/2)$, соответственно; где $p(s, \bar{n}, t)$ – преобразование Радона для скалярного произведения исходной векторной функции магнитной индукции \bar{B} и вектора нормали \bar{n} плоскости, а $p(s, \bar{e}_\theta, t)$, $p(s, \bar{e}_\alpha, t)$, $p(s, \bar{e}_{\alpha, \theta}, t)$ – преобразование Радона для скалярного произведения исходной векторной функции магнитной индукции \bar{B} и единичного вектора $\bar{e}_\theta, \bar{e}_\alpha, \bar{e}_{\alpha, \theta}$ лежащего в плоскости, соответственно.

С учетом выражения (5), система уравнений (6) записывается в виде:

$$\begin{cases} p_x(s, \bar{n}, t) = - \left[\sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{n}, t) dt + \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\theta, t) dt \right] \cos \alpha + \\ + \left[\sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\alpha, t) dt + \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_{\theta, \alpha}, t) dt \right] \sin \alpha; \\ p_y(s, \bar{n}, t) = - \left[\sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{n}, t) dt + \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\theta, t) dt \right] \sin \alpha - \\ - \left[\sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\alpha, t) dt + \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_{\theta, \alpha}, t) dt \right] \cos \alpha; \\ p_z(s, \bar{n}, t) = - \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{n}, t) dt + \sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\theta, t) dt. \end{cases} \quad (7)$$

Декартовы компоненты распределения векторной функции индукции в объеме измерения получают посредством применения алго-

ритма реконструкции к исходным проекционным данным (7), полученным в ходе управляемого пространственного перемещения рабочего органа (рис. 2).

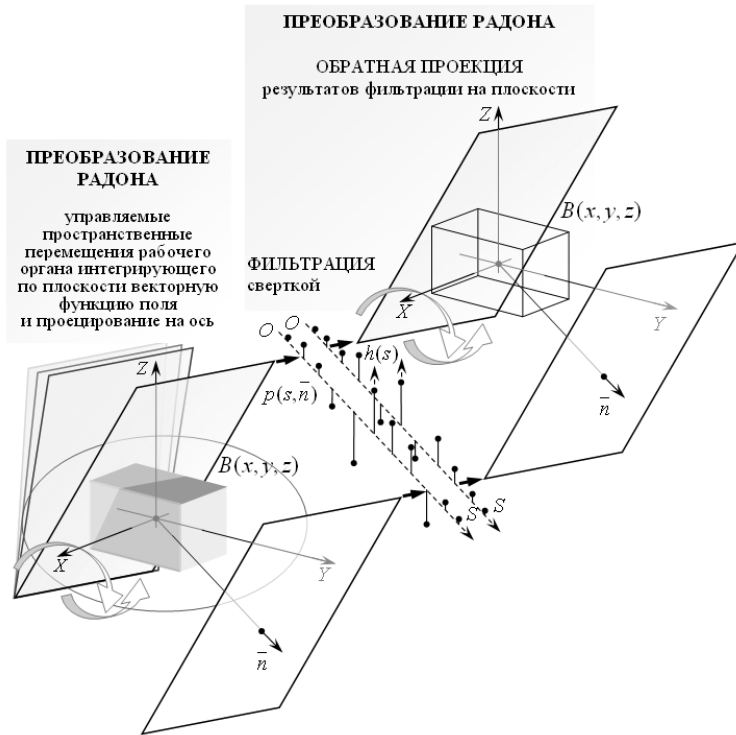


Рис. 2. Прямое и обратное преобразование Радона

Задача реконструкции декартовых компонент в способе неразрушающего объемного измерения сводится к решению основного интегрального уравнения (7) с нахождением распределения компонент B_x , B_y , B_z по измеренным значениям плоскостных проекций p_x , p_y , p_z . Решение основного интегрального уравнения (7) предполагает использование алгоритма реконструкции, основанного на обратном преобразовании Радона, посредством обратной проекции с Фурье-фильтрацией. Фурье-фильтрация осуществляется путем свертки проекций компонент непосредственно в пространстве оригинала Фурье-преобразования с соответствующей реализующей фильтрацию свертывающей функцией $h(s)$, являющейся обратным Фурье-

преобразованием квадрата частоты пространственного спектра. Так из (7) получают:

$$\begin{aligned}
 & \left\{ \begin{aligned}
 & B_x(x, y, z, t) = \int (p_x(s, \bar{n}, t) * h(s)) d\bar{n} = \\
 & = \int_0^{\pi/2} \left[\int_0^{\pi} (p_x(s, \bar{n}, t) * h(s)) d\alpha \right] \cos \theta d\theta = \\
 & = \int_0^{\pi/2} \left[\int_0^{\pi} \left(- \left[\sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{n}, t) dt + \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\theta, t) dt \right] \cos \alpha + \right. \right. \\
 & \left. \left[\sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\alpha, t) dt + \right. \right. \\
 & \left. \left. + \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_{\theta, \alpha}, t) dt \right] \sin \alpha \right) * h(s) d\alpha \right] \cos \theta d\theta; \\
 & B_y(x, y, z, t) = \int (p_y(s, \bar{n}, t) * h(s)) d\bar{n} = \\
 & = \int_0^{\pi/2} \left[\int_0^{\pi} (p_y(s, \bar{n}, t) * h(s)) d\alpha \right] \cos \theta d\theta = \\
 & = \int_0^{\pi/2} \left[\int_0^{\pi} \left(- \left[\sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{n}, t) dt + \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\theta, t) dt \right] \sin \alpha + \right. \right. \\
 & \left. \left[\sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\alpha, t) dt + \right. \right. \\
 & \left. \left. + \cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_{\theta, \alpha}, t) dt \right] \cos \alpha \right) * h(s) d\alpha \right] \cos \theta d\theta; \\
 & B_z(x, y, z, t) = \int (p_z(s, \bar{n}, t) * h(s)) d\bar{n} = \\
 & = \int_0^{\pi/2} \left[\int_0^{\pi} (p_z(s, \bar{n}, t) * h(s)) d\alpha \right] \cos \theta d\theta = \\
 & = \int_0^{\pi/2} \left[\int_0^{\pi} \left(- \left[\cos \theta \cdot \int_{nT}^{nT+t} u(s, \bar{n}, t) dt - \right. \right. \right. \\
 & \left. \left. - \sin \theta \cdot \int_{nT}^{nT+t} u(s, \bar{e}_\theta, t) dt \right] * h(s) d\alpha \right] \cos \theta d\theta,
 \end{aligned} \right.
 \end{aligned}$$

где символ «*» есть оператор свертки; $\int d\bar{n} = \int \int d\alpha \cos \theta d\theta$ – дифференциал вектора нормали; $d\theta$ и $d\alpha$ – дифференциал зенитного и азимутального углов, соответственно; $B_x(x, y, z, t)$, $B_y(x, y, z, t)$, $B_z(x, y, z, t)$ – x -, y -, z - компоненты векторной функции магнитной индукции \bar{B} , соответственно.

Таким образом, в данной работе предложенный способ позволяет получить в местах, недоступных для механического проникновения, в объеме измерения распределения компонент векторной функции магнитной индукции неоднородно распределенного в пространстве и периодически изменяющегося во времени магнитного поля. Эти компоненты реконструированы посредством применения обратного преобразования Радона к измеренным проекциям магнитного потока, полученным путем управляемого пространственного перемещения рабочего органа, интегрирующего по плоскости векторную функцию поля.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Жильников А.А., Жильников Т.А., Жулев В.И. Получение изображения распределения магнитного поля внутри биологических объектов. Биомедицинские технологии и радиоэлектроника. – М.: Издательское предприятие редакции журнала «Радиотехника», № 7, 2011. С. 41-46.
2. Системы комплексной электромагнитотерапии. Учебное пособие для вузов / Под редакцией Беркутова А.М., Жулёва В.И., Кураева Г.А., Прошина Е.М. – М: Лаборатория базовых знаний, 2000. 376 с.
3. Жильников А.А. Способ неразрушающей объемной регистрации магнитных полей в магнитотерапии // Материалы Всероссийской НТК «Биотехнические, медицинские и экологические системы и комплексы БИОМЕДСИСТЕМЫ - 2011». Рязань: РГРТУ, 2012. С. 150-152.
4. Клюев В.В. Рентгенотехника. Справочник в 2-х книгах. Кн. 2. М.: «Машиностроение», 1998, С. 319-326.
5. Наттерер Ф. Математические аспекты компьютерной томографии. Пер. с англ.: монография / Ф. Наттерер. – М.: Мир, 1990. - 288 с.
6. Хелгасон С. Преобразование Радона: Пер. с англ.: монография / С. Хелгасон. – М.: Мир, 1983. - 152 с.

М.А. ИВАНЧИКОВА, Д.А. ПЕРЕПЕЛКИН

Рязанский государственный радиотехнический университет

ОПТИМИЗАЦИЯ КОРПОРАТИВНОЙ СЕТИ ПО КРИТЕРИЮ МИНИМАЛЬНОЙ СТОИМОСТИ

Рассмотрена задача оптимизации корпоративной сети по критерию минимальной стоимости. Ее решение дает возможность сократить общие затраты на создание высокоскоростных каналов связи, построить устойчивую сетевую структуру, к которой легко добавлять новые сегменты с учетом специфики предприятия.

На сегодняшний день задача формирования структуры корпоративной сети актуальна при создании региональной телекоммуникационной инфраструктуры [1]. Корпоративная сеть состоит из множества базовых (коммуникационных) узлов, к которым подключаются пользователи сети, соединенные скоростными каналами связи. В идеальном случае задача формирования структуры базовой сети может быть сформулирована как задача формирования системы связи, соединяющей все базовые узлы и имеющей минимальную стоимость при заданных параметрах и ограничениях. Стоимость корпоративной сети складывается из стоимости используемых каналов связи и базовых узлов.

Задача формирования структуры базовой сети часто формулируется как задача построения минимального покрывающего дерева (spanning tree) на графе. При этом вершинами графа являются базовые узлы, а ребрами – каналы связи, соединяющие эти базовые узлы. Известно большое число алгоритмов построения минимального покрывающего дерева, однако при оптимизации и формировании структуры корпоративной сети наиболее часто используется алгоритм Прима.

Представим корпоративную сеть в виде неориентированного взвешенного связного графа $G = (V, E, W)$, в котором V – множество базовых узлов графа (маршрутизаторов), E – множество ребер (каналов связи), показанного на рис. 1. Для каждого ребра графа $e(i, j)$ задан вес $w(i, j)$, под которым понимается цена или стоимость канала связи корпоративной сети между базовыми узлами i и j . Общая стоимость графа G корпоративной сети составляет $w(G) = 66$.

Задача оптимизации корпоративной сети по критерию минимальной стоимости состоит в нахождении такого подграфа $G_1 \subset G$, для которого суммарный вес входящих в него ребер минимален.

$$w(G_1) = \sum_{(i,j) \in G_1} w(i, j)$$

Связный подграф графа G , являющийся деревом и содержащий все его вершины, называют покрывающим деревом этого графа или

остовным деревом. Следовательно, необходимо найти такое остовное дерево данного графа G , чтобы

$$\sum_{(i,j) \in G_1} w(i,j) = w(G_1) = \min_{(i,j) \in G} w(G)$$

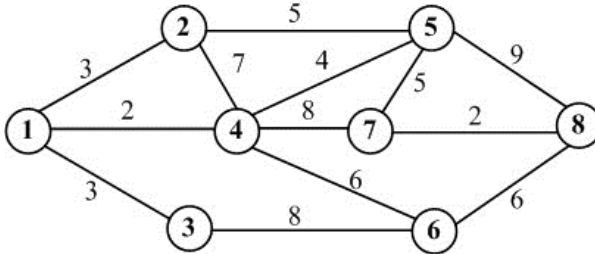


Рис. 1. Граф G корпоративной сети

Формирование дерева начинается с произвольной корневой вершины s . Искомый остов строится постепенно: к изначально пустому множеству G_1 на каждом шаге добавляется одно ребро наименьшего веса среди ребер соединяющих вершины этого дерева с вершинами не из дерева. Таким образом, множество G_1 всегда является подмножеством некоторого минимального остова. Следующее ребро $e(i, j)$ добавляется на очередном шаге, и выбирается так, чтобы не нарушить этого свойства. Процесс добавления ребер продолжается до тех пор, пока все вершины не будут связаны. В результате будет построено остовное дерево корпоративной сети с минимальной стоимостью, показанное на рис. 2. В приведенном примере минимальная стоимость базовой сети составит $w(G_1) = 25$.

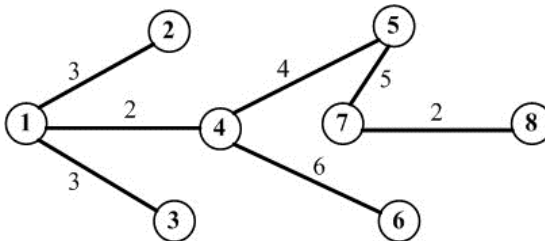


Рис. 2. Минимальное покрывающее дерево корпоративной сети

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Вишнеvский В.М. Теоретические основы проектирования компьютерных сетей. – М.: Техносфера. 2003. – 512 с.

П.И. КИСТЕНЁВ

Рязанский государственный радиотехнический университет

**ПРОЕКТИРОВАНИЕ УСТРОЙСТВ НА БАЗЕ
МИКРОКОНТРОЛЛЕРОВ С ИСПОЛЬЗОВАНИЕМ
ИНФОРМАЦИИ С GPS ДАТЧИКОВ И ПЕРЕДАЧЕЙ ДАННЫХ
ПО GSM КАНАЛУ**

Рассматриваются устройства определения местоположения и отправки данных с помощью GSM канала сотовой связи.

Идея использования микроконтроллеров для получения, обработки и отправки данных по GSM каналу владельцу не нова, но до недавнего времени была практически недоступна рядовому человеку ввиду дороговизны. Такая ситуация сохранялась до недавнего времени. Массовое производство сотовых телефонов, GPS приемников, GSM модулей, а также жесткая конкуренция операторов сотовой связи, сделали свое дело и привели к существенному удешевлению рынка данных устройств и услуг. В результате этого уже сейчас рядовому пользователю по карману, как само устройство, так и услуги оператора сотовой связи. Все это позволяет теперь по-новому взглянуть на данные системы.

Охрана частного имущества, например автомобиля, является одной из сфер применения данных устройств. Что же дает подобное охранное устройство? В первую очередь такая система дает невиданную ранее мобильность – пользователь может контролировать охраняемый объект как на расстоянии 100 метров, так и на расстоянии 100 километров и при этом быть всегда в курсе всех событий происходящих на объекте. Во-вторых, мобильный телефон дает пользователю привычный интерфейс, что намного упрощает передачу данных от охранного устройства владельцу и гарантирует адекватное их восприятие. В третьих, сложность построения такого прибора намного проще, нежели у стандартных охранных устройств, использующих для связи радиоканал. В четвертых, в устройстве подключается GPS-GSM устройство для определения координат и передачи их на сервер. Таким образом, хозяин автомобиля в любой момент может отслеживать местонахождение, скорость перемещения, направление движения автомобиля, и в случае угона владелец может установить местонахождение автомобиля, или же в случае обнаружения охранного устройства злоумышленниками, владелец автомобиля может определить, где в последний раз GPS-GSM устройство было на связи.

Система состоит из датчиков вскрытия автомобиля, блока GSM-сигнализации, блока GPS-GSM.



Рис. 1. Функциональная схема

Блок GSM анализирует показания датчиков и в случае срабатывания датчиков в режиме охраны, в зависимости от того как запрограммировано устройство либо отправляет SMS сообщение либо совершает вызов владельцу. Также совершается отсылка SMS на GPS-GSM блок, который включается и начинает передачу своих координат на сервер через определенное время или же через определенное расстояние, в зависимости от конфигурации. Что позволяет после обработки определить местоположение автомобиля. Время «старта» необходимое GPS-GSM блоку на определение позиции после включения, зависит от имеющейся в памяти начальной информации. Выделяются следующие режимы:

- «Холодный» старт («автопоиск») – время, позиция, альманах и эфемериды неизвестны.
- «Теплый» старт – позиция и эфемериды неизвестны, время и альманах известны.
- «Горячий» старт («перезахват») – альманах, эфемериды известны, время и позиция известны с некоторой ошибкой [1].

GPS-GSM блок содержит аккумуляторную батарею которая поддерживает устройство в режиме ожидания, что сокращается время старта при периодическом использовании автомобиля.

Данные устройства также можно использовать в целях мониторинга автотранспорта предприятия. Когда необходимо в любой момент времени знать, где находится то или иное транспортное средство организации, что в свою очередь позволяет оперативно реагировать на внештатные ситуации, например поломка автотранспорта или дорожно-транспортное происшествие.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Время старта GPS навигатора. Данные альманаха и эфемерид. [Электронный ресурс] URL: <http://www.gpsinfo.ru/articles.php?article=2292> (дата обращения: 20.04.2012).

Т.В. КОЖАНОВА

Рязанский государственный радиотехнический университет

**ЭНЕРГОСБЕРЕГАЮЩИЙ АНАЛОГ 100 ВТ ЛАМПЫ
НАКАЛИВАНИЯ***Рассматривается современный аналог традиционной лампы накаливания*

Будущее освещения – за светодиодными решениями. По оценкам специалистов Philips, к 2015 году уже 50 % всего освещения в мире будет основано на светодиодах, а к 2020 году – более чем 75 %. Светодиодная индустрия сегодня является одной из самых быстроразвивающихся отраслей. По прогнозам специалистов, гонка технологий продлится до середины 2020-х годов, когда светодиоды достигнут своих пиковых характеристик. В этом аспекте необходимо рассматривать светодиодные системы освещения как модернизируемые системы.

Рост цен на энергоносители продолжает оставаться одной из главных проблем экономики. Поэтому при конкурентной борьбе на рынке в конечном выигрыше окажутся компании, уделяющие особое внимание внедрению энергосберегающих технологий. Одним из путей решения является применение энергосберегающих технологий в освещении. Для этого необходимо внедрять полупроводниковые светотехнические изделия, показатели которых уже достигли уровня, позволяющего использовать их в освещении самых разных объектов: уличном освещении, освещении в ЖКХ и аварийном освещении.

Являясь сравнительно новой технологией, светодиоды в большинстве случаев превосходят традиционные источники света по энергоэффективности, качеству света, рентабельности и экологичности. Светодиодные осветительные приборы превосходят лампы накаливания практически во всех областях применения, а разрядные лампы высокого давления – в областях, требующих использования цветного света.

Светодиодные системы способны обеспечить ряд преимуществ перед традиционными системами освещения:

1. Энергоэффективность светодиодов может быть до пяти раз выше, чем у ламп накаливания и галогенных ламп, и сравнима с компактными люминесцентными лампами.

2. Светодиодные источники света являются направленными и излучают свет только в нужном направлении. Значительно меньшая, чем у люминесцентных, светящая поверхность позволяет использовать более эффективную оптику и лучше управлять светом.

3. Качество света белых светодиодов теперь сравнимо с качеством света КЛЛ, разрядных ламп высокого давления и люминесцентных ламп. Последние достижения в области производства светодиодов обеспечивают постоянство цвета и цветовую температуру, эквивалентные или превосходящие эти характеристики у традиционных источников света.

4. Существенно увеличенный полезный срок службы светодиодов по сравнению с традиционными источниками света. В результате этого сокращаются затраты на замену и обслуживание. Так, галогенные лампы следует менять в 12–20 раз чаще, чем светодиодные. В отличие от традиционных источников света, светодиоды могут использоваться даже после существенного снижения светового потока. При этом полный выход из строя происходит очень редко.

5. Общая эффективность светодиодов удваивается каждые 1,5–2 года. В связи с этим необходимо использовать модернизируемые светильники и менять в них только светодиодный модуль примерно раз в 5 лет, можно экономить 30 % материальные средства в долгосрочной перспективе.

6. Светодиоды не излучают в ИК- и УФ-диапазонах.

7. Управление работой светодиодных систем освещения может осуществляться при помощи цифровых контроллеров, обеспечивающих максимальную эффективность и высокую гибкость.

8. Светодиодные световые приборы является безынерционными: не требуется времени для прогрева или отключения, отсутствует вредное воздействие циклической подачи питания и диммирования.

9. Качественно разработанные светодиодные системы освещения обеспечивают простоту и гибкость установки, не требуя балластов и дополнительных источников питания.

10. В отличие от люминесцентных ламп, содержащих ртуть и требующих специального обращения и утилизации, светодиоды не содержат ртути и являются безопасными для окружающей среды [1].

Благодаря этим факторам, а также увеличившейся в последние годы световой отдаче, светодиоды стали очень перспективными источниками света.

Современная альтернатива традиционной 100 Вт лампы накаливания - лампа на основе мощного светодиода американской фирмы «Bridgelux» с питанием непосредственно от сети 220 В. Основным фактором, определяющим продолжительность жизни светодиода и стабильность его хроматических показателей, является температура кристалла. Повышенная температура кристалла приводит к ускоренной деградации параметров светодиода и снижает надежность изделия

в целом. Поэтому анодированный алюминиевый корпус нашего прибора, изготавливаемый литьевым методом, является одновременно и радиатором. Между светодиодом и радиатором находится термоинтерфейсный материал (продукт «Arctic Silver» фирмы «Bridgelux») с низким тепловым сопротивлением для заполнения воздушных промежутков, препятствующих хорошему отводу тепла. Долговечность без потери основных светотехнических параметров обеспечивается специфической микросхемой HVLED815 фирмы «STMicroelectronics» (драйвером светодиода), приспособленной под данный тип нагрузки [3].

Основные параметры:

- 1) световой поток порядка 1320 Лм, что эквивалентно ЛН мощностью 100 Вт;
- 2) световая отдача 98 Лм/Вт;
- 3) индекс цветопередачи 65;
- 4) стандартный цоколь E27.

Отличные потребительские свойства, такие как экономичность и долговечность, оправдывают высокие первоначальные затраты и делают светодиодные лампы новым поколением светотехники.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Джонатан Вейнерт Светодиодное освещение; пер. с англ. – Philips Solid-State Lighting Solutions, 2010. С. 13-17.
2. Давиденко Ю.Н. Современная схемотехника в освещении. – СПб.: Наука и техника, 2008.
3. Косенко С. «Альтаир» – новая звезда в созвездии LED-драйверов STMicroelectronics / С. Косенко// Вестник электроники. – 2011. – № 4 (32). – С. 34-39.

С.А. КОСОРУКОВ

ОАО «Государственный Рязанский приборный завод»

АЛГОРИТМ ОПРЕДЕЛЕНИЯ КАЧЕСТВА РАСПОЛОЖЕНИЯ РЕПЕРОВ НА КОЖУХЕ РЕПЕРНОГО УЗЛА

Рассматривается задача анализа расположения и ориентации реперов реперного узла оптико-электронной системы позиционирования. Определяется способность реперного узла обеспечивать видимость реперных треугольников во всем диапазоне углов поворота головы летчика.

Авиационная техника и вооружение требуют постоянного совершенствования путем оснащения их современными оптико-электронными приборами и приборными комплексами, в том числе такими, как нашлемные системы целеуказания и индикации (НСЦИ).

НСЦИ представляет собой сложный оптико – электронный комплекс, состоящий из системы индикации (СИ) и системы позиционирования (СП), размещаемых на защитном шлеме летчика (ЗШ).

СИ обеспечивает предъявление летчику визуальной информации в коллимированном виде на фоне внешней обстановки.

СП (рис.1) является составной частью нашлемной системы целеуказания и индикации НСЦИ и предназначена для определения угловых координат линии визирования летчика в системе координат летательного аппарата (ЛА), управления подвижным модулем оптико – электронной системы наблюдения и вооружением бортового оружейного комплекса посредством движения головы пилота.

Реперы (светодиоды), размещенные на кожухе реперного узла (УР) системы позиционирования, формируют световое излучение, которое регистрируется датчиками позиционирования (камерами).

Двухкамерный метод, реализуемый в данной СП, заключается в следующем: каждая из двух камер в конкретный момент времени должна «видеть» не менее трех одних и тех же реперов из общего их числа, чтобы на количестве «видимых» реперов можно было построить общий для двух камер реперный треугольник. Поэтому, например, если первая камера «видит» пять реперов из девяти, а вторая четыре из пяти, «видимых» первой камерой, то наличие хотя бы одного общего треугольника обеспечено.

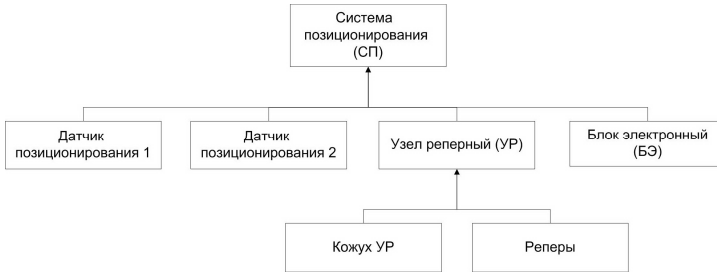


Рис. 1. Система позиционирования

При конструкторском проектировании УР возникает задача расположить и ориентировать реперы так, чтобы во всем заданном диапазоне углов поворота головы по азимуту, месту и крену, на каждой комбинации была обеспечена видимость хотя бы одного треугольника.

Область излучения светодиода (рис.2), с учетом установленного порога, можно рассматривать как область, ограниченную конической поверхностью, с вершиной B в центре излучения и углом при вершине γ , который соответствует удвоенному угловому отклонению на графике зависимости относительной интенсивности излучения от углового отклонения (рис.3). Ось конической поверхности определяется точками B и C , координаты которых известны.

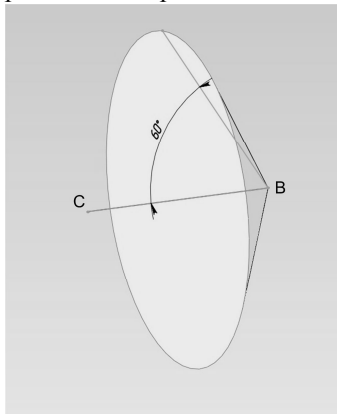


Рис. 2. Область излучения светодиода

В нашем случае (светодиод IR Edixeon) за порог интенсивности принято 50%. Этому порогу соответствует значение углового отклонения 60 градусов (рис.3). Значит, угол при вершине конуса будет равен 120 градусам. Основными исходными данными алгоритма являются

координаты точек A_1, A_2 (фокусы камер 1 и 2 соответственно), B_i и C_i (i – количество реперов) (рис.4). Координаты точек A_1, A_2 остаются неизменными, а координаты точек B_i и C_i будут изменяться в зависимости от поворота кожуха. В данном алгоритме рассматривается вращение кожуха УР относительно точки O , которая соответствует соединению седьмого шейного и первого грудного позвонков $C7/T1$ (рис.4), как последовательность поворотов:

- 1) вокруг оси Y (угол по азимуту)
- 2) вокруг оси X (угол по месту). При этом ось Z преобразуется в ось Z'
- 3) вокруг оси Z' (угол по крену).

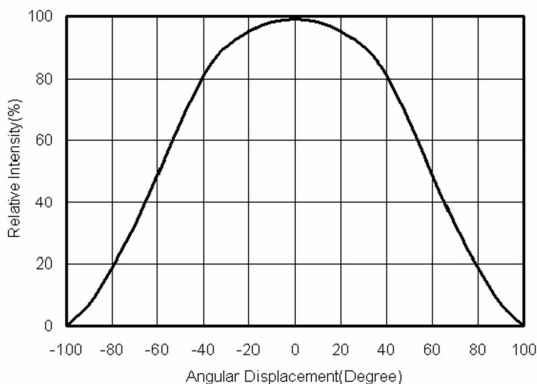


Рис. 3. Характеристика светодиода IR Edixeon

Матрица преобразованных однородных координат будет иметь вид:

$$[X'] = [X][M] = [X][R_y][R_x][R_z],$$

где: $[X]$ – матрица исходных однородных координат точек B_i и C_i , $[R_y][R_x][R_z]$ – матрицы поворота относительно осей X, Y и Z' соответственно.

В действительности кинематика движений головы и шеи более сложная, но данная модель охватывает крайние значения углов поворота и ее достаточно для решения поставленной задачи.

Чтобы определить «видимость» i -го репера одной камерой, нужно вычислить угол β_i между осью BC конуса и прямой AB , проведенной через фокус камеры A и вершину конуса B_i (рис.4). Затем нужно сравнить угол β_i с половиной угла при вершине конуса: если $\beta_i \leq \gamma/2$, то репер «видно», а если $\beta_i > \gamma/2$, то репер «не видно».

Параллельно определяются множества $N1$ и $N2$ номеров реперов, «видимых» соответственно камерами $A1$ и $A2$ для каждой комбинации углов поворота по азимуту, месту и крену. Затем определяется пересечение множеств:

$$N = N1 \cap N2 .$$

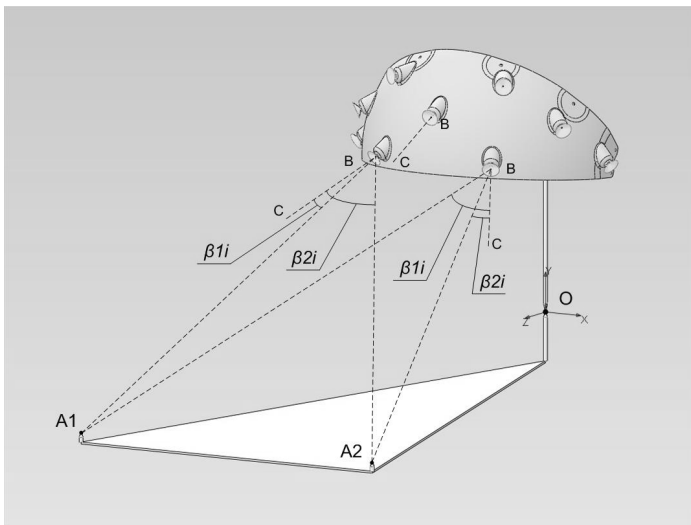


Рис. 4

Таким образом, получается множество N номеров реперов «видимых» обеими камерами на данной комбинации углов.

Затем следует формирование матрицы V_{lmn} , которую можно назвать матрицей видимости, где l, m, n – количества интервалов по азимуту, месту и крену соответственно. Количество интервалов зависит от шага разбиения углового диапазона. Чем меньше шаг, тем больше размерность матрицы V_{lmn} . Если число элементов множества N больше или равно трем:

$$|N| \geq 3,$$

то наличие общего реперного треугольника гарантировано, и в матрицу видимости записывается единица на позицию, соответствующую данной комбинации углов поворота.

Затем вычисляется качество $Q(V)$ в процентном отношении количества единичных элементов к общему количеству элементов матрицы V :

$$Q(V) = \frac{\text{Кол-во единиц}}{l * m * n} * 100\%.$$

В идеальном случае $Q = 100\%$. В случае, если $Q \neq 100\%$, то по матрице видимости легко определить участки диапазонов углов поворота, на которых нет общих реперных треугольников. В этом случае либо корректируется положение и (или) ориентация соответствующих реперов, либо вводятся дополнительные реперы. Введение дополнительных реперов негативно сказывается на инерционно-массовых характеристиках, а также на технологичности изготовления УР, поэтому важно обойтись минимально возможным их количеством.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Роджерс Д., Адамс Д. Математические основы машинной графики: Пер. с англ. – М.: Мир, 2001. – 604с., ил.
2. Голованов Н.Н. Геометрическое моделирование. – М.: Издательство физико-математической литературы, 2002. – 472с. – ISBN 5-94052-048-0
3. Корячко В.П. и др. Теоретические основы САПР: Учебник для вузов/ В.П. Корячко, В.М. Курейчик, И.П. Норенков. – М.: Энергоатомиздат, 1987. – 400с.: ил.

Д.Т. КУРАСБЕДИАНИ

Рязанский государственный радиотехнический университет

ПРОБЛЕМЫ ИНФОРМАЦИОННОГО ОБЕСПЕЧЕНИЯ В БАНКОВСКОЙ ДЕЯТЕЛЬНОСТИ

Рассматриваются проблемы, возникающие в процессе информационного обеспечения современной банковской деятельности

Осуществляя обработку основного объема информации интегрированная автоматизированная банковская система (АБС) является технологической базой современного банка. Интегрированной АБС присущи взаимосвязь всех информационных процессов, единая модель данных, единая технология их обработки, общее программное ядро и т.д. Необходимо, чтобы все подразделения банка функционировали в едином информационном пространстве. Это делает более эффективным управление банком, который имеет, как правило, территориально распределенную структуру, разнообразные взаимодействия со множеством клиентов, организаций, населением. Единое информационное пространство делает доступными, объединяет все виды информации, обеспечивает оперативный доступ к ней, позволяет добиться полной прозрачности информации и др.

Выделим следующие составные части информационного обеспечения: информационная модель, система показателей, система классификации и кодирования, база данных как способ организации информации [1].

База данных создается на основе определенного способа структурирования данных при разработке модели бизнес-процесса. Главной задачей моделирования является построение информационного обеспечения пользователей всех рангов со всей полнотой отражения свойств объектов. Возможности будущего развития специфики предметной области банковского бизнеса связаны с совершенствованием методов моделирования информационной базы системы. Решение таких задач опирается на использование многофункциональных системных программ, которые должны работать с актуальными, полными и достоверными данными. В базе данных информационная модель отражает взаимосвязь объектов предметной области, их состав и свойства на уровне файлов, документов, показателей, реквизитов. На рис. 1 показан пример модели базы данных (из программного комплекса ИНВОБАНК), отражающий состав и связи базовых банковских объектов.

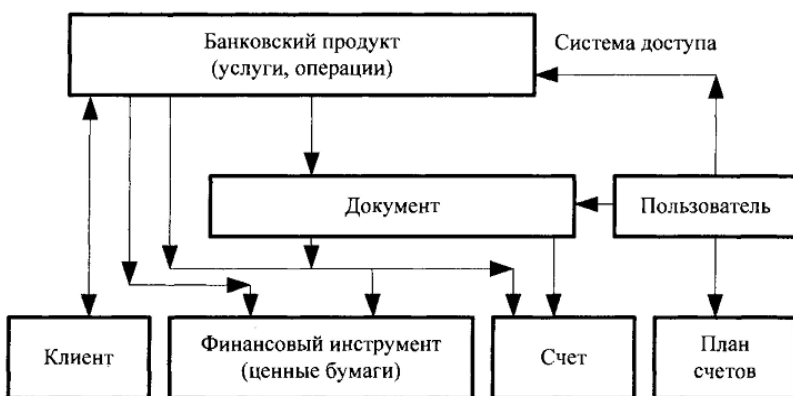


Рис. 1. Состав и связи базовых банковских объектов

Функционирование объектов в различных процессах и операции с ними представляются в виде алгоритмов решения задач. Поэтому модель данных в базе существенно зависит от типа операций, выполняемых над данными при решении функциональных задач. Особенностью банковской деятельности является необходимость обработки нескольких типов данных — оперативных, аналитических, прогнозных. В рамках АБС решаются разные задачи: оперативная обработка больших объемов данных в режиме реального времени (в темпе протекания процессов), аналитическая работа и работа с данными по прогнозированию будущего развития. В зависимости от целевого использования информация структурируется как в виде баз данных, так и в виде хранилищ данных.

Для оперативной обработки текущих данных используются базы данных системы OLTP (On-Line Transaction Processing — процессы транзакций в линии связи). Они основаны на постоянном обновлении информации в базе, данные регулярно добавляются, удаляются, корректируются. Важной задачей является надежное обеспечение транзакций. Под транзакцией понимается изменение состава данных в базе во время короткого по времени цикла взаимодействия с ней (запрос — выполнение — ответ) по линиям связи.

Хранилище — это ориентированные на предметную область, поддерживающие хронологию наборы данных, неизменяемые, интегрированные. Оно работает со значительными объемами данных по сравнению с базой данных и имеет более сложную систему организации. Технология хранилищ данных используется прежде всего с целью обобщения данных для всестороннего и глубокого анализа деятельно-

сти банка, для переосмысления его бизнеса, для получения консолидированной, объединенной по разным направлениям отчетности [2].

При решении аналитических задач пользователям верхнего уровня (администрация, менеджеры, специалисты) требуются данные, выбранные и обобщенные (агрегированные) по ряду признаков. Таким требованиям удовлетворяют системы аналитической обработки данных OLAP (On-Line Analytic Processing — процессы анализа в линии связи). OLAP-системы построены на следующих базовых принципах: данные, необходимые для принятия решений, предварительно агрегированы в нужных разрезах; организация информации обеспечивает максимально быстрый доступ к ней; язык манипулирования данными основан на использовании бизнес-понятий и близок пользователям. Например, информация собирается по бизнес-процессам и их конкретным показателям для специалистов по анализу отдельных направлений в деятельности банка.

Изменение условий ведения бизнеса и соответствующее перепроектирование бизнес-процессов приводят к изменениям в информационных технологиях и, прежде всего в информационном обеспечении. Границы эволюционного (постепенного) развития информационного обеспечения зависят от свойств адаптивности бизнес-процессов и информационной базы. Свойство базы развиваться и адаптироваться к новым условиям означает возможность встраивания новых объектов в модель базы, а также их свойств и связей.

Одной из главных проблем на этапе эксплуатации базы данных является проблема ее эффективного ведения в условиях изменения информационных требований пользователей, которые приводят к изменению концептуальной основы и, как следствие, к изменению модели данных. При этом возникает необходимость в перенастройке бизнес-логики, реструктурировании файлов базы, корректировке ретроспективных данных, обновлении алгоритмов, перенастройке интерфейса и т.д.

Автоматизируя новый или меняющийся бизнес-процесс, необходимо решать проблемы структуризации новых или реструктуризации имеющихся данных, реализации алгоритмов протекания процессов решения задач, обеспечивать комплекс операций по вводу, обработке, передаче, хранению и выводу данных. Анализ и проектирование структуры данных являются ключевыми этапами разработки информационного обеспечения не только на стадиях создания АБС, но и в процессе адаптации базы к новым условиям работы. Так, расширение возможностей предоставления банком новых услуг требует развития информационных систем, ИТ и зависит не только от наличия в базе

данных о составе объектов, операций, но и способов их изменения или расширения.

Характерная для банковской сферы высокая скорость изменения условий реализации бизнеса требует разработки специализированных методов ведения базы данных, соответствующих быстрым изменениям бизнес-среды, а также использования современного высокотехнологичного программно-технического инструментария.

При построении модели функций банка и его бизнес-процессов следует учитывать фактор повышенной заинтересованности банка в информации о ссудозаемщиках (клиенты, которые обращаются в банк за кредитом). Во многом существование кредитных организаций определяется их способностью извлекать такую информацию. Поэтому новые технологии обработки и передачи информации приводят к появлению новых видов банковских продуктов, возрастает необходимость для банков ее сбора и использования. Расходы на приобретение информации постоянно снижаются благодаря технологическим изменениям в ее обработке и распространении. В результате информационной специализации в банках появляются новые продукты и услуги. Что в свою очередь приводит к необходимости совершенствования, модернизации информационной системы банка. Переход к новой базе данных и технологии ее ведения происходит при возрастании объема и номенклатуры услуг, клиентов, контрагентов, числа сделок и операций, появлении новых и сложных задач, росте прямых и косвенных издержек, снижении производительности и эффективности труда.

Необходимость соблюдения баланса между качеством информационного обеспечения АБС и затраченными на разработку и функционирование ресурсами приводит к тому, что современные технологии в основном ориентированы на массовое, промышленное создание сложных систем большими коллективами специалистов. Разрабатываются и создаются корпоративные АБС крупными специализированными фирмами — разработчиками банковских информационных технологий, которые имеют соответствующую научную базу и высококвалифицированных профессионалов в области создания новых информационных технологий [1].

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Титоренко Г.А. Информационные системы в экономике. М.: ЮНИТИ. 2008. 463 с.
2. Веб-сайты www.bankmib.ru и programbank.ru.

А.В. МАТОРИН, С.А. БАЛЮК

Рязанский государственный радиотехнический университет

**АНАЛИЗ ВЗАИМОДЕЙСТВИЯ ИЗЛУЧАТЕЛЕЙ
В АДАПТИВНОЙ АНТЕННОЙ РЕШЁТКЕ**

Рассматривается задача анализа взаимодействия излучателей в адаптивной антенной решётке. Предложенный в работе алгоритм позволяет оценивать влияние взаимодействия излучателей на параметры стационарного режима адаптивной антенной решётки.

Постановка задачи. Постоянное возрастание потребностей в услугах сотовой связи и средствах беспроводного доступа к телекоммуникационным сетям, в частности беспроводного доступа в Интернет, приводит к необходимости непрерывного увеличения ёмкости и улучшения других качественных показателей существующих и перспективных сетей связи. В настоящее время одним из важнейших факторов, благодаря которому открываются дополнительные возможности улучшения качественных характеристик сетей подвижной связи, является использование так называемых интеллектуальных антенн.

Адаптивными антеннами, которые в первую очередь относят к классу интеллектуальных, называют антенны, автоматически изменяющие свои направленные, частотные и поляризационные характеристики в зависимости от изменяющейся во времени информации, в частности содержащейся в принятых сигналах [1]. Наибольший практический интерес представляют адаптивные антенные решётки (ААР), которые обеспечивают автоматическое подавление помех, приходящих с различных направлений, путём формирования глубоких провалов в характеристике направленности. Фактически такие антенны непрерывно осуществляют мониторинг зоны покрытия, пытаются адаптироваться к изменяющейся среде, включающей пользователей и источники помех.

Применение ААР в сотовых системах связи позволяет получить ряд преимуществ [2]:

Во-первых, при той же мощности базовая станция, оснащённая ААР, имеет большую дальность действия, и, следовательно, для покрытия заданной территории требуется меньшее число базовых станций.

Во-вторых, улучшаются характеристики качества связи, т.к. базовая станция может скорректировать свои «усилия» в каждой конкретной ситуации и передавать данные к каждому отдельному абоненту индивидуально. Таким образом, где бы абонент ни находился, в его

сторону будет передан сигнал с достаточной мощностью, чтобы обеспечить необходимый уровень качества.

В-третьих, использование интеллектуальных антенн позволяет использовать то же число кодов в каждом луче, что и во всей соте без использования ААР. Поэтому многократно увеличивается емкость соты и скорость передачи данных.

Однако, несмотря на то, что уже созданы и успешно испытаны экспериментальные системы, использующие интеллектуальнее антенны, остаётся ещё много вопросов, связанных с поиском и реализацией оптимальных алгоритмов обработки сигналов в таких антеннах и их конструктивной реализацией, в частности с глубоким учётом электродинамического взаимодействия [3,4].

Учёт взаимодействия элементов ААР особенно необходим в тех случаях, когда необходимо разработать антенну, имеющую ограниченные размеры (например, установленную на борту летательного аппарата). При этом для увеличения передаваемой мощности и чувствительности желательно использовать антенны с большим числом приёмо-передающих элементов. Однако, увеличение числа излучателей в антенне приводит к увеличению их взаимного влияния, которое выражается в искажении их индивидуальных характеристик.

Таким образом, возникает задача оценки влияния электродинамического взаимодействия излучателей на параметры стационарного режима ААР.

Способ построения ААР определяется алгоритмом решения оптимизационных задач, который зависит от выбранного критерия адаптации. В системах радиосвязи и телекоммуникаций, естественно, соотношение сигнал/(помеха + шум) остается важнейшей характеристикой качества приема, поэтому будем оценивать взаимодействие излучателя на основе данного критерия.

Максимизация отношения сигнал/помеха+шум на выходе адаптивной антенной решётки. Критерий максимизации отношения сигнал/помеха+шум (ОСШП) заключается в максимизации отношения [3,4]:

$$ОСШП = \frac{P_{\Sigma S}}{P_{\Sigma ИП}}, \quad (1)$$

где $P_{\Sigma S}$ - мощность сигнала на выходе ААР, $P_{\Sigma ИП}$ - мощность шумов и помех на выходе ААР.

Определим мощность шумов и помех на выходе ААР как среднее значение квадрата модуля результирующей комплексной огибаю-

щей шумов и помех $I_{\Sigma}(t)$ [4]:

$$P_{\Sigma \text{шп}} = \frac{1}{2} \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} |I_{\Sigma}(t)|^2 dt = \frac{1}{2} \overline{|I_{\Sigma}(t)|^2}. \quad (2)$$

Представим $I_{\Sigma}(t)$ в форме скалярного произведения двух векторов:

$$I_{\Sigma}(t) = \langle W I(t) \rangle, \quad (3)$$

где $I(t)$ - вектор-столбец с компонентами $I_n(t)$, $\langle W$ - вектор-строка весовых коэффициентов. Делая соответствующую подстановку выражения (3) в (2), мощность шумов и помех $P_{\Sigma \text{шп}}$ представим в виде квадратичной формы:

$$P_{\Sigma \text{шп}} = \langle W^* M_I W \rangle, \quad (4)$$

где $[M_I] = \overline{I^*(t) \langle I(t) \rangle}$ - корреляционная матрица вектора $I(t)$, компонентами которой являются средние значения различных комбинаций сопряжённых произведений результирующих помеховых сигналов на входах управляемых четырёхполюсников.

Корреляционная матрица $[M_I]$ характеризует взаимные статистические свойства различных компонент вектора $I(t)$. В рассматриваемом случае диагональные элементы матрицы $[M_I]$ характеризуют мощности помеховых сигналов на соответствующих входах. Недиагональные элементы определяют статистические связи помеховых сигналов на различных входах устройств. Элементы матриц $[M_j] = \overline{\mathfrak{I}_j^{a*}(\theta_j, \varphi_j) \langle \mathfrak{I}_j^a(\theta_j, \varphi_j) \rangle}$, $j = \overline{1, L}$, определяются формулой $M_{jmn} = (\overline{\mathfrak{I}_{jm}^{a*}(\theta_j, \varphi_j) \mathfrak{I}_{jn}^a(\theta_j, \varphi_j)})$, откуда следует, что $M_{jmn} = M_{jnm}^*$ и, таким образом, матрица $[M]$ является эрмитовой. Более того, поскольку по определению, $P_{\Sigma \text{шп}} > 0$ и $P_{\emptyset 0} > 0$, квадратичная форма принимает только положительные значения $\langle W^* M W \rangle > 0$ для любых W , то матрица $[M]$ является положительно определённой.

Аналогичное рассмотрение может быть проведено относительно энергетики сигнала на выходе ААР, который также можно представить в виде квадратичной формы:

$$P_{\Sigma S} = \langle W^* M_{0S} W \rangle, \quad (5)$$

где M_{0S} - корреляционная матрица сигнала.

С учётом выражений (4) и (5) перепишем выражение (1) в виде:

$$ОСШП = \frac{P_{\Sigma S}}{P_{\Sigma ШП}} = \frac{\langle W^* M_{0S} W \rangle}{\langle W^* M_I W \rangle}. \quad (6)$$

Экспериментальная часть. Количественные результаты, характеризующие влияние электродинамического взаимодействия на оптимальные параметры стационарного режима ААР, рассмотрим на примере антенной решётки, имеющей длину 3λ и состоящей из тонких параллельных полуволновых вибраторов, расположенных параллельно идеально проводящей плоскости на расстоянии $h = 0.3\lambda$ (рисунок 1).

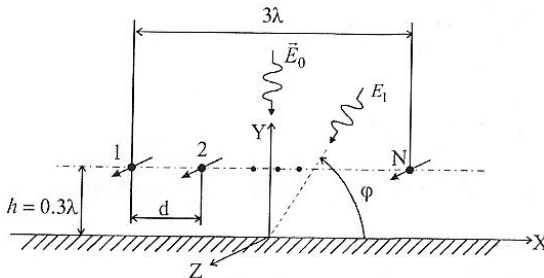


Рис. 1. Анализируемая антенная решётка

Система координат, в которой рассматривается геометрия решётки, выбрана таким образом, что ось Z совпадает с осями вибраторов, ось X – с осью решётки. Взаимодействие элементов в решётке учитывается в рамках метода интегральных уравнений Галена, с помощью которого определяются характеристики направленности $\tilde{\mathfrak{S}}_n^a(\theta, \varphi)$, необходимые для расчета параметров ААР.

Будем считать, что источники сигнала и помех расположены в плоскости XY ($\theta = \pi/2$). Пусть имеется одна помеха E_0 , направление которой определяется углом φ и варьируется в пределах от 0 до $\pi/2$. Направление прихода сигнала E_1 соответствует углу $\varphi = \pi/2$, отношение мощности помехи к мощности внутренних шумов $\nu_1 = 100$, отношение мощности сигнала к мощности внутренних шумов $\nu_0 = 1$.

Предложим следующий алгоритм расчёта оптимальных параметров стационарного режима ААР:

1. Определяются характеристики направленности элементов антенной решётки $\mathfrak{S}_m^a(\varphi)$, $m = \overline{1, N}$:

$$\mathfrak{S}_m^a(\varphi) = \frac{iZ_C}{\pi\sqrt{Z_B}} \sin(kh \sin \varphi) \sum_{n=1}^N I_{mn} e^{ikx_n \cos \varphi}, \quad (7)$$

где x_n - координаты излучателей, Z_B - волновое сопротивление линий, возбуждающих вибраторы, $Z_{\tilde{N}}$ -

Коэффициенты I_{mn} соответствуют нормированным токам на входах элементов решётки при возбуждении m -го элемента решётки ($m = \overline{1, N}$) и остальных элементах, нагруженных на неотражающие нагрузки. Они находятся из матричного уравнения [4]:

$$[Z_A + E]I = 2[E],$$

где $[Z_A]$ - нормированная матрица сопротивлений системы полуволновых вибраторов над идеально проводящей плоскостью, которая рассчитывается в рамках метода интегральных уравнений галленовского типа [5]; $[E]$ - единичная матрица.

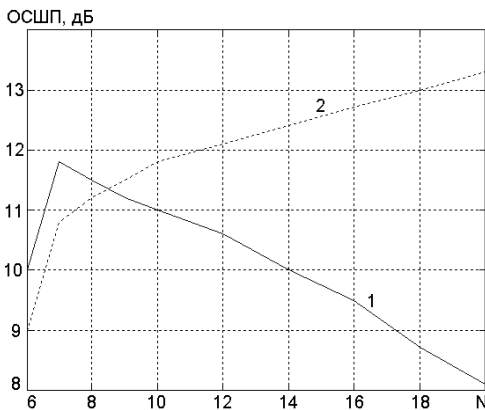


Рис. 2. Зависимость отношения сигнал/шум+помеха от числа излучателей

Структура формулы (7) обусловлена тем, что вибраторы являются полуволновыми и их характеристики направленности, вычисленные

в рамках синусоидального приближения токов, обладают достаточно высокой точностью.

2. Определяется нормированная корреляционная матрица шумов и помех ААР:

$$[M] = [E] + \frac{4\pi}{Z_C} \nu_1 \mathfrak{I}^{a*}(\varphi) \mathfrak{I}^a(\varphi).$$

3. Из уравнения [4] $[M]W_{opt} \rangle = \mathfrak{I}^{a*}(\pi/2) \rangle$ определяется оптимальный вектор весовых коэффициентов $W_{opt} \rangle$:

4. Оптимальное значение ОСШП рассматриваемой ААР как функция угла прихода помехи рассчитывается по формуле:

$$ОСШП = \frac{4\pi}{Z_C} \langle \mathfrak{I}^a(\pi/2) W_{opt} \rangle.$$

Результаты расчётов по приведённому алгоритму для различного числа излучателей и различных направлений прихода помехи показаны на рисунках 2 и 3.

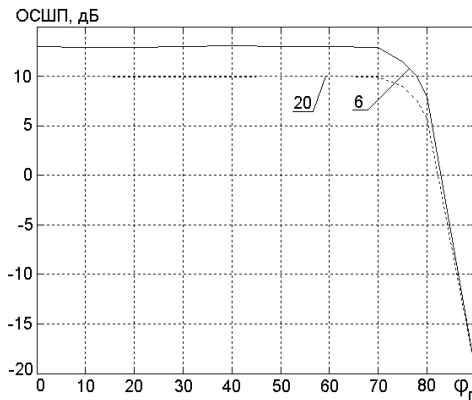


Рис. 3. Зависимость отношения сигнал/шум+помеха от направления прихода помехи для разного числа излучателей

На рисунке 2 показана зависимость ОСШП от числа элементов решётки, когда угол прихода помехи составляет 75° , 1 – измерение ОСШП с учётом взаимодействия излучателей, 2 – без учёта взаимодействия. Из рассмотрения этих графиков видим, что для $N = 7-8$ (расстояние между элементами решётки при этом 0.42λ) взаимодей-

стве элементов позволяет несколько улучшить величину ОСШП. При дальнейшем увеличении количества элементов решётки взаимодействия элементов решётки приводит к уменьшению ОСШП. Уменьшение ОСШП при уплотнённом расположении элементов решётки объясняется их рассогласованием и искажением их индивидуальных ДН. Такого рода эффекты могут существенно ограничивать возможности адаптации, несмотря на увеличение степеней свободы.

На рисунке 3 показана зависимость ОСШП для различных направлений прихода помехи и различного количества излучателей. Уменьшение ОСШП, для углов, близких к 90° , объясняется тем, что в этом случае помеха входит в область главного лепестка ДН антенной решётки. Когда направление прихода помехи совпадает с направлением прихода сигнала $\varphi = \pi / 2$ ОСШП становится отрицательным, т.к. по условию мощность помехи больше мощности сигнала.

Выводы. Из анализа полученных результатов видно, что взаимное влияние элементов оказывает достаточно сильное влияние на параметры стационарного режима ААР и может приводить к значительному снижению ОСШП. Данное обстоятельство особенно актуально в системах радиолокации и радиосвязи, когда требуется обеспечивать приём сигнала, приходящего с достаточно большого расстояния, и максимизировать отношение сигнал-шум на выходе системы для обеспечения последующей обработки сигналов слабой мощности в условиях наличия помех.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Монзинго Р.А., Миллер Т.У. Адаптивные антенные решётки. – М.: Радио и связь, 1986. – 446 с.
2. Пистолькорс А.А., Литвинов О.С. Введение в теорию адаптивных антенн. – М.: Наука, 1991. – 200 с.
3. Журавлёв А.К., Лукошкин А.П., Поддубный С.С. Обработка сигналов в адаптивных антенных решётках. – Л.: Изд-во ЛГУ, 1983. – 239 с.
4. Маторин А.В. Распространение радиоволн и антенно-фидерные устройства средств связи с подвижными объектами: учеб. пособие ч. 3. – Рязань: РГРТУ, 2009. – 148 с.
5. Айзенберг Г.З., Ямпольский В.Г., Терешин О.Н. Антенны УКВ, ч. 1. – М.: Связь, 1977. – 382 с.

И.Н. МИРОХИНА

Рязанский государственный радиотехнический университет

**НЕЧЕТКОЕ КОГНИТИВНОЕ МОДЕЛИРОВАНИЕ
ДЛЯ АНАЛИЗА РИСКОВ
ПРОЕКТНО-ПРОИЗВОДСТВЕННОЙ СИСТЕМЫ**

Рассматриваются вопросы применения аппарата нечеткого когнитивного моделирования для анализа рисков проектно-производственной системы на этапе мониторинга.

Введение

Управление рисками проектно-производственных систем представляет собой сложный итерационный процесс [1, 5]. Ключевым фактором мониторинга рисков является учет эффективности мероприятий по предотвращению рисков или смягчению их последствий в сравнении с ущербом при наступлении рисков.

Существенной особенностью мониторинга риска является то, что для его качественного осуществления предварительно должны быть решены основные задачи управления рисками: планирование управления рисками, выявление рисков, анализ и оценка приоритетности рисков, планирование мероприятий по предотвращению и снижению рисков [5].

Перспективным направлением решения задач мониторинга рисков является развитие методов формализации, анализа и моделирования риска на основе их обобщения для случаев информации нестатистической природы, в том числе информации, имеющей качественный характер, характеризующейся различного рода неопределенностью, оценкой и агрегированием с помощью различных шкал [1, 3].

Современные средства мониторинга риска, в том числе и определенная машинная обработка не всегда позволяют извлечь новые или желаемые знания из потока информации. Поэтому возникает необходимость в качественно ином уровне ее обработки, предусматривающем использование методов и средств когнитивного компьютерного моделирования [1 - 5].

Способ нечеткого когнитивного моделирования для анализа рисков проектно-производственной системы

Когнитивное моделирование рассматривается здесь как способ анализа, обеспечивающий определение силы и направления влияния факторов на перевод объекта управления (проекта, производственной

системы и т.п.) в целевое состояние с учетом сходства и различия во влиянии различных факторов на объект управления. Основу таких когнитивных моделей обычно представляет классическая когнитивная карта [1, 2].

Анализ возможности применения нечетких когнитивных карт для мониторинга рисков в проектно-производственных системах показывает, что характеристическими признаками концептов различных типов нечетких когнитивных карт является отсутствие учета ситуационных аспектов когнитивных структур. В объективных же условиях незнания внутренней структуры концептов именно эти ситуационные факторы позволяют сформировать механизмы принятия решений и оценки их предпочтений.

Обобщенной нечеткой когнитивной картой здесь будем называть нечеткую причинно-следственную сеть [2] (сеть взаимовлияния) $G = (E, W)$, где E - множество концептов $e_j \in E = \{e_1, e_2, \dots, e_p\}$. Каждый концепт e_i $i \in I = (1, 2, \dots, p)$ описывается соответствующей лингвистической переменной (e_i, T_i, D_i) - терм-множество лингвистической переменной e_i (набор лингвистических значений концепта, характеризующих его типовые состояния); m_i - число типовых состояний данного концепта; D_i - базовое множество концепта e_i ; W - множество обусловленных связей элементов E .

Использование обобщенных нечетких когнитивных карт для построения моделей и анализа нечетко-целевых производственных систем позволяет учесть ситуационные аспекты формируемых когнитивных структур, характеризующие пространство возможных состояний системы и оценку мотивационных действий в анализируемых системах, а также является основой развития методов когнитивного моделирования.

В работе предлагается способ построения обобщенной нечеткой когнитивной карты сложной проектно-производственной системы, который включает следующие этапы:

Этап 1. Задание списка концептов обобщенной нечеткой когнитивной карты, моделирующей нечетко-целевую проектно-производственную систему.

Этап 2. Определение терм-множеств лингвистических переменных, характеризующих типовые состояния концептов нечеткой когнитивной карты.

Этап 3. Формирование нечетких переменных для описания термов, определяющих типовые состояния концептов.

В зависимости от характера и способа представления информации, используемой при описании термов, определяющих типовые состояния концептов, могут быть рассмотрены прямые и косвенные методы построения функций принадлежности. Прямые методы основаны на непосредственном задании экспертом нечетких множеств S_k . Они используются для измеримых понятий или когда выделяются их полярные значения [1]. В случае опроса и согласования мнений группы экспертов, как правило, применяется прямой метод построения функций принадлежности, приближенно равных некоторому нечеткому числу, и приближенных интервальных оценок [2].

Этап 4. Определение отношений причинности между каждой парой концептов.

Этап 5. Определение терм-множеств лингвистических переменных, характеризующих связи между типовыми состояниями каждой пары концептов.

Этап 6. Формирование нечетких переменных для описания термов, характеризующих значения связей между типовыми состояниями каждой пары концептов.

Связи, характеризующие нечеткую степень влияния между типовыми состояниями каждой пары концептов, описываются нечеткими переменными, которые могут задаваться либо значениями из отрезка $[-1, 1]$, либо функциями принадлежности [2].

В первом случае степень позитивных и негативных влияний концептов друг на друга представляется нечеткими логическими связями в виде нормированной степени влияния концептов друг на друга (значения из отрезка $[-1, 1]$), обладающую следующими свойствами: равна 0, если концепт e_i не влияет на концепт e_j ; равна 1, если позитивное влияние концепта e_i на концепт e_j максимально; равна -1 , если негативное влияние концепта e_i на концепт e_j максимально; принимает положительное и одновременно отрицательное значение при одновременном позитивном и негативном влиянии концептов друг на друга [1, 2].

Задание же взаимосвязей между концептами с помощью функций принадлежности позволяет формировать продукционные модели принятия решений и предпочтений, двойственные к представлению когнитивной карты, т. е. относительно влияний, а не концептов.

Применительно к мониторингу рисков вследствие различных причин появления и типов неопределенности при мониторинге рисков

когнитивное моделирование систем позволяет выполнить все требования, предъявляемые к моделям мониторинга рисков.

Заключение

Проведенный анализ позволяет утверждать, что наиболее целесообразным является мониторинг рисков с помощью использования обобщенной нечеткой когнитивной модели. При этом система мониторинга проектно-производственных рисков может быть представлена в виде концептов обобщенной нечеткой когнитивной карты.

Нечеткое когнитивное моделирование может выполняться как в режиме реального времени при состояниях, полученных в результате работы регистрирующей части разрабатываемой проектно-производственной системы и отражающих реальную обстановку в данный момент времени, так и для ответа на вопрос: «что будет, если?» при произвольных значениях лингвистических переменных и значениях концептов.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Балабанов И.М. Риск-менеджмент. – М.: Финансы и статистика. 1996.
2. Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. – М.: Горячая линия – Телеком. 2007.
3. Корячко В.П., Таганов А.И. Программный метод управления рисками качества проекта информационной системы // Известия Белорусской инженерной академии. 2004. Вып. 1(17) / 4. С. 168 - 179.
4. Нечеткие множества в моделях управления и искусственного интеллекта / Под ред. Д. А. Поспелова. – М.: Наука. 1986.
5. Таганов А.И. Методика анализа и сокращения рисков проектов сложных программных систем по характеристикам качества // Научно-технический журнал «Вестник РГРТУ». № 1. Рязань. 2010.

В.М. МИРОШИН, Д.А. ПЕРЕПЕЛКИН

Рязанский государственный радиотехнический университет

АНАЛИЗ АЛГОРИТМОВ СЕГМЕНТАЦИИ СТРУКТУР КОМПЬЮТЕРНЫХ СЕТЕЙ

Рассматриваются основные комбинаторные алгоритмы сегментации структур компьютерных сетей. На основе трудоемкости и оптимальности итогового разбиения производится анализ каждого из алгоритмов.

Задача сегментации графов на непересекающиеся подмножества вершин актуальна во многих областях современной науки и техники. Разбиение графа используется при сегментации изображений для их последующего анализа и интеллектуальной обработки. Широкое применение задача сегментации нашла также в социальных, информационных и биохимических сетях, где часто требуется выявлять в структуре сети сильно связанные сообщества. Однако одной из самых пространственных областей науки, в которой проблема разбиения графа имеет важное значение, являются компьютерные сети. Для улучшения маршрутизации и разбиения на широковебательные домены сеть разделяется на отдельные непересекающиеся подмножества элементов (компьютеров или роутеров), причем количество связей внутри этих подмножеств должно превышать количество внешних связей с другими подмножествами. Оптимальность разбиения зависит от выбора алгоритма сегментации. В данной статье будут рассмотрены комбинаторные алгоритмы сегментации структур компьютерных сетей, так как они обеспечивают более сбалансированное разбиение и меньшее информационное взаимодействие полученных подсетей по сравнению с другими алгоритмами и используют информацию о связности вершин в графе [6].

Постановку задачи сегментации графов можно сформулировать следующим образом. Представим компьютерную сеть в виде неориентированного взвешенного связанного графа $G=(V, E)$, каждой вершине $v \in V$ и каждому ребру $e \in E$ которого приписан вес. Задача оптимального деления графа состоит в разбиении его вершин на непересекающиеся подмножества с максимально близкими суммарными весами вершин и минимальным суммарным весом ребер, проходящих между полученными подмножествами вершин. Следует отметить возможную противоречивость указанных критериев разбиения графа – равновесность подмножеств вершин может не соответствовать минимальности весов граничных ребер и наоборот. В большинстве случаев

необходимым является выбор того или иного компромиссного решения [7].

Задача разбиения графа относится к классу NP-полных, верхняя оценка числа разбиений определяется числом Белла:

$$B_{n+1} = \sum_{k=0}^n \binom{n}{k} \cdot B_k$$

Рассмотрим следующие комбинаторные алгоритмы сегментации структур компьютерных сетей:

- Алгоритм рекурсивного деления пополам;
- «Жадный» алгоритм разбиения графов;
- Спектральный алгоритм Ньюмана;
- Алгоритм Гирвана - Ньюмана.

Алгоритм рекурсивного деления пополам относится к числу самых простых и наименее трудоемких алгоритмов сегментации. В его основе лежит метод бинарного деления, при котором на первой итерации граф разделяется на две равные части, на втором шаге каждая из полученных частей также разбивается на две части, аналогичное разбиение производится и на всех последующих итерациях. Применительно к данному алгоритму сначала находится диаметр графа, определяемый как наибольшее расстояние между вершинами графа. Расстояние между двумя вершинами графа определяется как наименьшее число граней, которые нужно пройти от узла i к узлу j . Начиная с одной из вершин, половина вершин присваивается одной из подобластей, а половина - другой подобласти. Затем рекурсивная процедура применяется к каждой из подобластей. Время работы алгоритма оценивается как $O(N)$, где N – число вершин графа или маршрутизаторов в компьютерной сети [2].

Для нахождения диаметра графа, начиная с исходной вершины v_s , принятой за потенциальный корень, каждой соседней вершине присваивается метка 1, а соседям соседей присваивается метка 2. Последняя из помеченных вершин имеет метку m , которая имеет смысл расстояния от корневой вершины. Далее следующая вершина v_i обозначается как исходная, процесс повторяется. Вершина v_i с максимальным значением m считается корнем, при этом m является диаметром графа. Реализация алгоритма нахождения диаметра графа также требует $O(N)$ операций.

Пример работы алгоритма рекурсивного деления пополам показан на рис. 1. Разобьем исходный граф на $p = 3$ части. Диаметр графа будет равняться $m = 3$, вершина v_0 будет являться корнем графа.

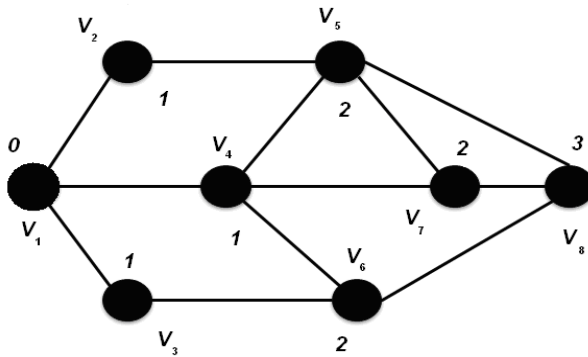


Рис. 1. Алгоритм рекурсивного деления пополам, определение диаметра графа

Вершина v_4 с меткой 1 будет являться границей раздела исходного графа на 2 части (см. рис. 2). Произведем разбиение графа. Далее перейдем к первому подграфу и применим к нему процедуру разбиения. Диаметр подграфа будет равен $m = 2$, вершина v_2 будет являться корнем подграфа. В соответствии с вычисленными метками вершин данный подграф разделится ещё на 2 области: в первой области будут расположены вершины v_1 и v_2 , во второй - вершины v_3 и v_4 . В результате работы алгоритма исходный граф разобьётся на 3 подграфа (см. рис. 3).

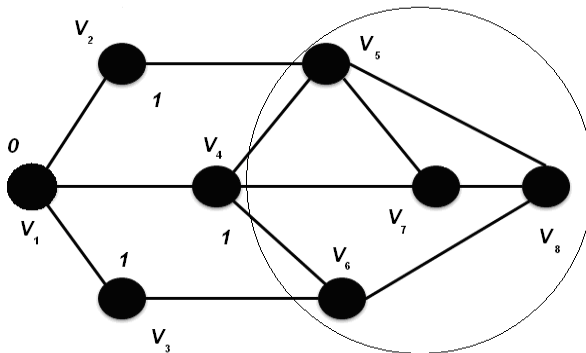


Рис. 2. Разбиение графа на 2 части

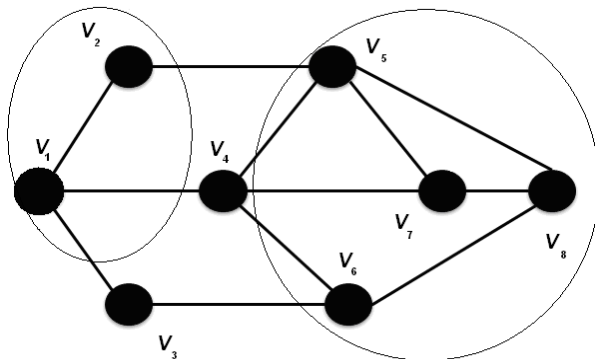


Рис. 3. Результат разбиения исходного графа на 3 части

Однако алгоритм рекурсивного деления пополам имеет существенный недостаток: в нем мало учитывается связность вершин, и получаемое в итоге разбиение оказывается неоптимальным.

«Жадный» алгоритм разбиения заключается в принятии локально оптимальных решений на каждом этапе, допуская, что конечное решение также окажется оптимальным [1].

Процедура начинается с вершины, имеющей наименьшую степень. Маркируются все соседние вершины, а затем вершины, соседние с соседями. Первые N/p маркированных вершин присваиваются одной подобласти, и процедура применяется к оставшейся части графа до тех пор, пока все вершины не являются маркированными. Данный алгоритм имеет схожие черты с алгоритмом рекурсивного деления пополам, хотя и не является таковым. Трудоемкость алгоритма оценивается как $O(N)$ [2].

Пример работы «жадного» алгоритма разбиения показан на рис. 1. Разобьем исходный граф на $p = 3$ части. Разбиение будем начинать с вершины v_3 , так как она имеет самую низкую степень, равную 2. Произведем маркировку всех вершин графа и выделим $8/3 = 2$ вершины с наименьшими пометками в отдельное подмножество (см. рис. 4).

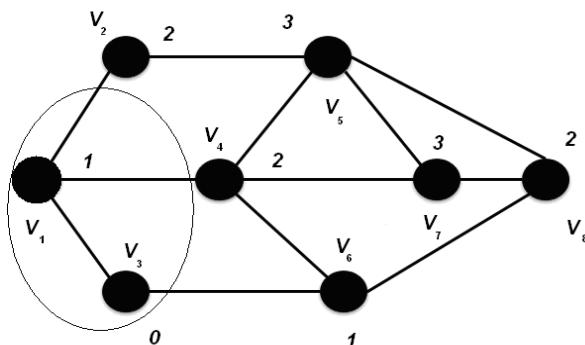


Рис. 4. Разбиение графа на 2 части с помощью «жадного» алгоритма

Повторим процедуру для оставшегося подграфа. Разбиение начнем с вершины v_2 с самой низкой степенью, равной 2. Произведем маркировку всех вершин графа и выделим $6/2 = 3$ вершины с наименьшими пометками в отдельное подмножество. Результат разбиения графа показан на рис. 5.

Преимуществом данного алгоритма является простота и высокая скорость работы, однако итоговое разбиение графа получается неоптимальным, что является весомым недостатком при сегментации компьютерных сетей.

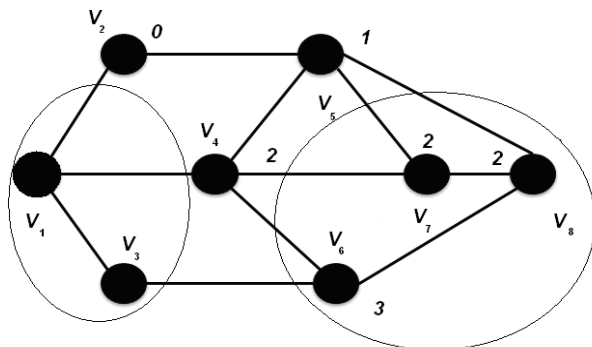


Рис. 5. Результат разбиения исходного графа на 3 части

Спектральный алгоритм Ньюмана, основанный на использовании собственных векторов и собственных значений, производит разбиение исходного графа, минимизировав при этом количество внешних ребер. Размер разреза определяется как:

$$P = \frac{1}{2} \sum_{i \in \gamma_1, j \in \gamma_2} A_{ij}$$

Если ввести индексный вектор $s_i \in \{-1; 1\}$, то:

$$R = \frac{1}{2} \sum_{i,j} (1 - s_i s_j) A_{ij}$$

Далее если k_i – мощность вершины i , $\delta_{ij} = 1$, если $i = j$, и 0 в противном случае, то:

$$R = \frac{1}{2} \sum_{i,j} s_i s_j (k_i \delta_{ij} - A_{ij})$$

В матричной форме:

$$R = \frac{1}{4} s^T \cdot L \cdot s$$

Матрица Лапласа представляется в виде $L = A - D$, где A – матрица смежности графа, а D – диагональная матрица, состоящая из степеней вершин графа.

Матрица L является положительно полуопределенной и имеет наименьшее собственное значение $\lambda_1 = 0$ и соответствующий ему собственный вектор $e_1 = \{1, 1, \dots, 1\}$. Вектор e_1 не является решением задачи, поскольку найденное решение означает, что все вершины при разбиении попадут в одну группу. Для связного графа следующее наименьшее собственное значение матрицы Лапласа λ_2 является положительным [2]. Собственный вектор e_2 , связанный с собственным значением λ_2 является искомым решением задачи минимизации. Этот собственный вектор (вектор Фидлера) состоит из весовых множителей, связанных с вершинами графа, что позволяет применить метод деления пополам, разделяя вершины по различным подобластям в соответствии с их весовыми множителями. Собственный вектор матрицы Лапласа, соответствующий ее наименьшему нетривиальному собственному значению, определяет связность графа и его протяженность, а также выступает в качестве метрики.

Для нахождения собственного вектора, связанного со вторым наименьшим собственным значением, используются алгоритм Ланцоша или метод сопряженных градиентов.

На рис. 6 изображен исходный граф, который необходимо разбить на 2 части.

Для данного графа матрицы A , D и L выглядят следующим образом:

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} \quad D = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 \end{pmatrix}$$

$$L = \begin{pmatrix} -2 & 0 & 1 & 0 & 1 & 0 \\ 0 & -3 & 0 & 1 & 1 & 1 \\ 1 & 0 & -3 & 0 & 1 & 1 \\ 0 & 1 & 0 & -2 & 0 & 1 \\ 1 & 1 & 1 & 0 & -3 & 0 \\ 0 & 1 & 1 & 1 & 0 & -3 \end{pmatrix}$$

Вектор Фидлера будет иметь вид : $a_2 = (2 \ -1 \ 1 \ -2 \ 1 \ -1)$.
Итоговое разбиение графа изображено на рис. 7. Величина разреза, т.е. количество внешних ребер примет минимальное значение и будет равно $R = 2$.

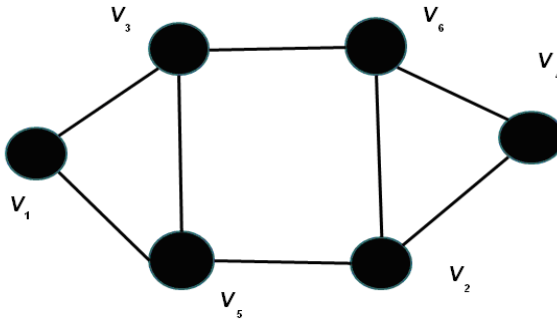


Рис. 6. Исходный граф для разбиения

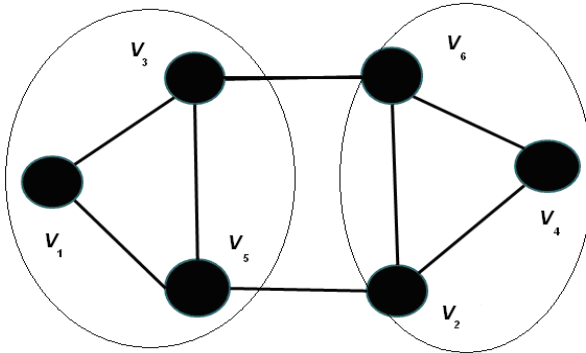


Рис. 7. Итоговое разбиение графа

В отличие от ранее рассмотренных алгоритмов спектральный алгоритм Ньюмана приводит к связным подобластям при условии, что исходный граф также является связным, при этом число ребер графа, разрезаемых при сегментации области, оказывается примерно в два раза меньшим по сравнению с остальными алгоритмами. Трудоемкость нахождения собственных векторов оценивается как $O(N \log N)$.

Алгоритм Гирвана - Ньюмана основывается на понятии центральности вершин и производит разбиение графа в соответствии с данной величиной. Центральность – мера важности вершины в графе (применительно к сетевым сообществам) [3]. Для определения данной величины рассчитывается смежность ребер графа – мера важности, пропорциональная числу кратчайших путей от всех вершин до всех, проходящих через ребро:

$$C_{B(V)} = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(V)}{\sigma_{st}}$$

Рассмотрим основные шаги алгоритма Гирвана – Ньюмана:

Шаг 1. Вычислить смежность для всех ребер графа. Вычисление можно произвести по алгоритму Ньюмана с трудоемкостью $O(|V||E|^2)$ или по алгоритму Джонсона с трудоемкостью $O(V^2 \log V + VE)$.

Шаг 2. Удалить ребро с наивысшим показателем.

Шаг 3. Пересчитать величину смежности с учетом изменений.

Шаг 4. Выполнять шаги 2 и 3 до тех пор, пока рёбра не закончатся.

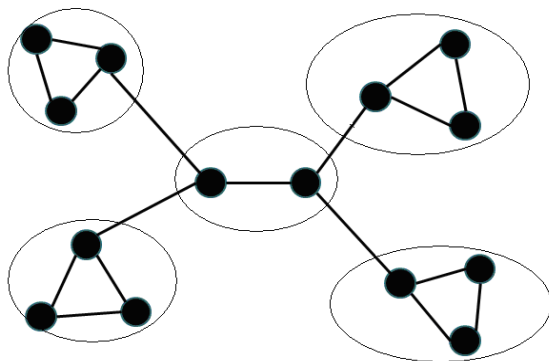


Рис. 8. Разбиение графа алгоритмом Гирвана - Ньюмана

Алгоритм Гирвана - Ньюмана выполняет иерархическую кластеризацию дивизионным методом. Результаты разбиения структуры сети при помощи данного алгоритма показаны на рис. 8. Однако существенным недостатком данного способа разбиения является высокая вычислительная сложность.

Существует ещё несколько комбинаторных алгоритмов сегментации графов, в том числе алгоритм Кернигана - Лина, алгоритм распространения меток, алгоритм k - кластеризации. Однако все эти алгоритмы редко применяются в компьютерных сетях и поэтому не рассматриваются в данной статье.

Для сравнения представленных алгоритмов сегментации по критерию оптимальности итогового разбиения будем использовать меру модулярности. Мера модулярности - показатель связности сообщества в виде отношения количества связей внутри сообщества к внешним связям:

$$Q = \frac{1}{4m} \cdot \sum_{ij} \left[A_{ij} - \frac{k_i \cdot k_j}{2m} \right] \cdot (s_i s_j + 1),$$

где $m = |E|$, A_{ij} - элемент матрицы смежности, s_j - показатель вхождения в сообщество (+1/ -1), k_i - мощность вершины i [4].

Смысл данной меры - разница между долей ребер, входящих в первое или во второе сообщество, и ожидаемой долей ребер в первом, либо втором сообществе для случайного графа с распределением мощностей вершин, соответствующим данному [5].

Подведем итоги. Алгоритм рекурсивного деления пополам и «жадный» алгоритм имеют низкую вычислительную сложность, однако итоговое разбиение оказывается неоптимальным, что доказывает

низкая мера модулярности. Алгоритм Гирвана - Ньюмана обладает слишком высокой трудоемкостью. Исходя из этого, оптимальным среди рассмотренных является спектральный алгоритм Ньюмана, так он осуществляет сегментацию исходного графа на сообщества с самой высокой мерой модулярности. Его трудоёмкость составляет $O((k-1) N \log N)$, где k - число сообществ, на которые разбивается исходный граф.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Беликов Д.А., Говязов И.В., Данилкин Е.А., Лаева В.И., Проханов С.А., Старченко А.В. Высокопроизводительные вычисления на кластерах. Томск: Изд-во ТГУ. 2008.
2. Волков К.Н. Применение средств параллельного программирования для решения задач механики жидкости и газа на многопроцессорных вычислительных системах. Вычислительные методы и программирование. 2006. № 1. С. 73–88.
3. Schloegel, K., Karypis, G., Kumar, V. (2000). Graph Partitioning for High Performance Scientific Simulations.
4. M. E. J. Newman and M. Girvan, Finding and evaluating community structure in networks. *Phys. Rev. E* 69, 026113. 2004.
5. M. E. J. Newman, Detecting community structure in networks. *Eur. Phys. J. B* 38, 321–330. 2004.
6. R. Guimera and L. A. N. Amaral, *Nature* 433, 895900. 2005.
7. J. P. Bagrow and E. M. Bollt, *A local method for detecting communities*, *Physical Review. E* 72, 046108. 2005.

Г.В. ОВЕЧКИН, П.В. ОВЕЧКИН

Рязанский государственный радиотехнический университет

**ПРИМЕНЕНИЕ НЕДВОИЧНЫХ МНОГОПороГОВЫХ
ДЕКОДЕРОВ В КАНАЛАХ СВЯЗИ С ГРУППИРУЮЩИМИСЯ
ОШИБКАМИ**

Рассматривается эффективность недвоичных многопороговых декодеров (qMPPD) символьных самоортогональных кодов в канале Гилберта-Эллиотта. Показано, что и при наличии группирующихся ошибок qMPPD оказывается значительно лучше декодеров кодов Рида-Соломона как по вероятности ошибки декодирования, так и по сложности реализации.

К современным системам передачи цифровых данных предъявляются очень жесткие требования по безошибочности передачи информации. Для обеспечения таких требований используют методы помехоустойчивого кодирования, применение которых позволяет улучшать многие важные характеристики систем передачи данных, например, экономить мощность передатчика, увеличивать дальность связи, уменьшать размеры антенн и др. Немаловажную роль помехоустойчивые коды играют и в системах хранения данных, в которых необходимо обеспечивать высокую надежность долговременного хранения информации на носителе.

В настоящее время специалисты в области помехоустойчивого кодирования проявляют большой интерес к недвоичным кодам, работающим с цифровыми данными на уровне символов, например, с байтами информации. Недвоичные коды применяются в каналах с группирующимися ошибками, в качестве составляющих элементов различных каскадных кодов, для защиты от ошибок информации на различного рода носителях (CD, DVD, Blu-ray и др.).

Анализ недвоичных корректирующих кодов и алгоритмов их декодирования показал, что широкое применение в реальных системах передачи и хранения информации из недвоичных кодов нашли только коды Рида-Соломона (РС) [1]. Однако декодеры коротких кодов РС, которые и применяются на практике, не могут обеспечить высокую эффективность декодирования, а для длинных кодов РС невозможно создать декодер из-за высокой сложности его реализации.

Среди других методов коррекции ошибок наиболее перспективным является метод недвоичного многопорогового декодирования [2, 3], обеспечивающий практически оптимальное декодирование даже очень длинных недвоичных самоортогональных кодов (СОК) при любом размере символа. Отличительными особенностями данного метода

являются линейная сложность реализации и высокая эффективность коррекции ошибок.

До настоящего времени были известны характеристики q МПД только для q -ичных симметричных каналов, где ошибки появлялись независимо друг от друга с равной вероятностью. Однако часто в реальных системах передачи и хранения информации возникают пакеты ошибок, которые обычно существенно ухудшают эффективность классических методов коррекции ошибок. Пакеты ошибок могут быть вызваны источником периодического шума, например, расположенным поблизости радиолокатором, многолучевым распространением сигнала, каким-то вращающимся механизмом или в случае возникновения царапин и отпечатков пальцев на оптическом носителе. Традиционно, для того чтобы разбить пакеты ошибок применяется перемежение. Однако использование перемежителя усложняет процесс обработки сигнала, особенно при большом размере блока. Кроме того, использование информации о состоянии канала без применения перемежения иногда позволяет достигать лучшей эффективности декодирования [4]. Каналы с пакетирующимися ошибками могут быть описаны моделью канала Гилберта-Эллиота [5, 6], который является наиболее простым представителем данного класса каналов.

Канал Гилберта-Эллиота может находиться в двух состояниях: “плохой” и “хороший”. С вероятностью P_{bad} канал изменяет свое состояние с хорошего на плохое, а с вероятностью $1-P_{\text{bad}}$ остается в состоянии “хороший”. С вероятностью P_{good} канал изменяет свое состояние с плохого на хорошее, а с вероятностью $1-P_{\text{good}}$ остается в состоянии “плохой”. В том случае, если канал находится в состоянии “хороший”, вероятность искажения символа при передаче по каналу достаточно мала и равна P_g , а вероятность правильной передачи символа равна $1-P_g$. Когда канал находится в состоянии “плохой”, вероятность искажения символа достаточно высока и равна P_b , а вероятность правильной передачи символа равна $1-P_b$.

На рис. 1 показана зависимость частоты символьных ошибок на выходе q МПД от средней вероятности ошибки P_e в канале Гилберта-Эллиота с параметрами $P_{\text{good}}=0.1$, $P_g=0.01$, $P_b=0.99$ и при различных P_{bad} , определяемой в соответствии с выражением

$$P_e = \frac{P_{\text{bad}} \cdot P_b + P_{\text{good}} \cdot P_g}{P_{\text{bad}} + P_{\text{good}}}$$

На данном рисунке кривой 3 показаны характеристики q МПД для кода с $R=1/2$, $d=17$ и длиной блока n около 100000 однобайтовых символов. В процессе декодирования данного кода выполнялось от 10

до 40 итераций. Видно, что такой q МПД эффективно исправляет около 25% ошибок в данном канале, в то время как декодер кодов РС длиной 255 однокбайтовых символов, характеристики которого представлены кривой 1, справляется только с 1% ошибок. Использование несколько более длинных кодов РС для символов большего размера позволяет получить лишь немного лучшие результаты (кривая 2 на рис. 1), а использование еще более длинных кодов РС ограничивается высокой сложностью реализации их декодеров. Следовательно, недвоичные МПД оказываются значительно эффективнее декодеров для кодов РС и проще для реализации и в каналах с группирующимися ошибками.

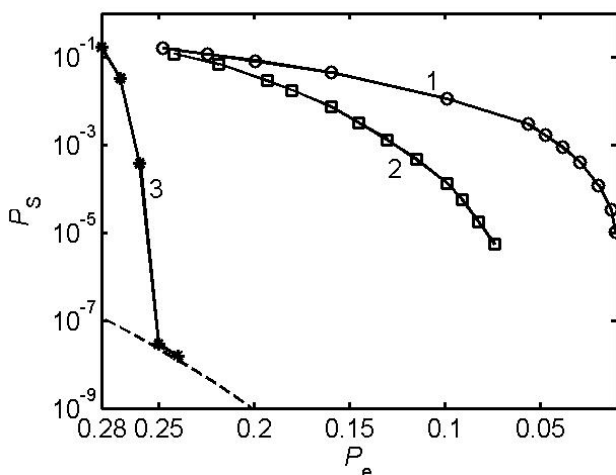


Рис. 1. Характеристики q МПД и декодера кодов РС в канале Гилберта-Эллиота

Таким образом, сравнение кодов РС и недвоичных самоортогональных кодов, декодируемых q МПД, показало, что коды РС уступают недвоичным СОК по эффективности и при этом имеют гораздо большую вычислительную сложность декодирования. Кроме того, q МПД эффективно работает не только в каналах с независимыми ошибками, но и в каналах с пакетирующимися ошибками, что делает превосходство q МПД над кодами РС еще более существенным.

Это позволяет считать, что в дальнейшем q МПД смогут заменить коды РС в разнообразных системах передачи и хранения данных, обеспечивая работу подобных систем в значительно более сложных условиях.

Работа выполнена при финансовой поддержке РФФИ (грант №12-07-00418а) и РГРТУ.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Reed I. S., Solomon G. Polynomial codes over certain finite fields // J. Soc. Industrial Appl. Math., 1960, vol. 8, P. 300–304.
2. Золотарёв В.В., Зубарев Ю.Б., Овечкин Г.В. Многопороговые декодеры и оптимизационная теория кодирования. – М.: Горячая линия – Телеком, 2012. – 239 с.
3. Веб-сайт www.mtdbest.ru.
4. Chen J., Tanner R. M. A hybrid coding scheme for the Gilbert-Elliott channel // IEEE Transactions on Communications, №54(10), p. 1787–1796, 2006.
5. Mushkin M., Bar-David I. Capacity and Coding for the Gilbert-Elliott channels // IEEE Trans. Inform. Theory, vol. IT-35 №.6, p. 1277–1290, 1989.
6. Richardson T. Modern coding theory – Cambridge University Press, 2008, 272 p.

Д.А. ПЕРЕПЕЛКИН

Рязанский государственный радиотехнический университет

ПРИМЕНЕНИЕ АЛГОРИТМА ПАРНЫХ ПЕРЕХОДОВ В ПРОТОКОЛЕ OSPF

Предложена практическая реализация алгоритма парных переходов в корпоративных сетях на базе протокола OSPF.

Для повышения качества функционирования корпоративных сетей наиболее важной задачей является выбор эффективного алгоритма маршрутизации, который будет обеспечивать поиск оптимальных маршрутов с учетом различных свойств корпоративной сети. В настоящее время широкое распространение получили алгоритмы состояния каналов (Link State Algorithm, LSA), которые нашли свое применение в протоколе OSPF (Open Shortest Path First).

В протоколе OSPF все маршрутизаторы работают на основании одинаковых графов, и каждый из них самостоятельно решает задачу оптимизации маршрутов, что делает процесс маршрутизации более устойчивым к изменениям конфигурации. «Широковещательная» рассылка (то есть передача пакета всем непосредственным соседям маршрутизатора) используется здесь только при изменениях состояния свя-

зей. Вершинами графа являются как маршрутизаторы, так и объединяемые ими сети. Распространяемая по сети информация состоит из описания связей различных типов: маршрутизатор – маршрутизатор, маршрутизатор – сеть. В процессе выбора оптимального маршрута анализируется неориентированный граф сети. Метрики выбранного маршрута могут характеризоваться следующими параметрами качества обслуживания (QoS):

- 1) пропускной способностью канала;
- 2) задержкой (время распространения пакета);
- 3) числом дейтограмм, стоящих в очереди для передачи;
- 4) загрузкой канала;
- 5) требованиями безопасности;
- 6) типом трафика;
- 7) числом шагов до цели;
- 8) возможностями промежуточных связей.

Выбор оптимального маршрута в протоколе OSPF определяется по алгоритму Дейкстры. Трудоемкость построения таблиц маршрутизации с использованием этого алгоритма составляет порядка $O(N^2)$, где N – число маршрутизаторов в корпоративной сети.

Применение данного алгоритма в условиях динамического изменения нагрузок на линиях связи в современных корпоративных сетях является неэффективным из-за высокой трудоемкости поиска оптимальных маршрутов и необходимости полного пересчета таблиц маршрутизации.

В последнее время разработаны новые алгоритмы маршрутизации, например алгоритм парных переходов [1], который позволяет уменьшить трудоемкость построения таблиц маршрутизации до величины $O(k \cdot N)$, где k – число парных переходов, при изменении нагрузки на линиях связи.

Алгоритм нахождения кратчайших путей на основе парных переходов характеризуется необходимостью расчета дополнительной информации. Поэтому объем данных, рассчитываемых на подготовительном этапе для обеспечения общности условий принимаемых решений, является избыточным. В данном алгоритме для того, чтобы обработать любое изменение веса некоторого ребра, необходимо присвоить этому весу граничные значения и провести поиск кратчайших путей для полученного графа.

Рассмотрим работу алгоритма парных переходов. Представим корпоративную сеть в виде неориентированного взвешенного связного графа $G = (V, E, W)$, где V – множество вершин, $|V| = N$, E – множество ребер, $|E| = M$, W – множество весов ребер, показанного на рис. 1.

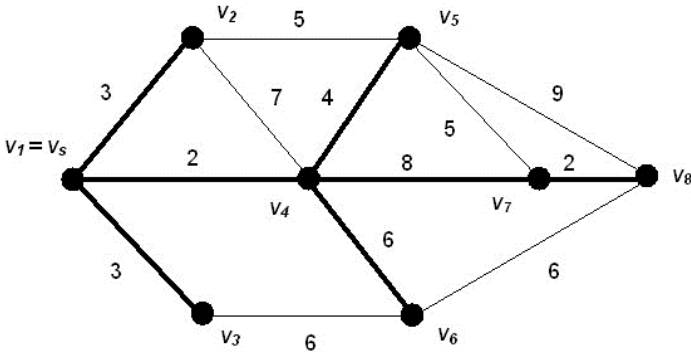


Рис. 1. Граф G корпоративной сети

Пусть на графе G в некоторый момент времени уже решена задача поиска кратчайших путей до всех вершин множества $V_s = V \setminus v_s$ из начальной вершины v_s , т.е. построено дерево кратчайших путей с корнем в вершине v_s . Обозначим это дерево как T_g . На рис. 1 жирными линиями обозначено построенное дерево кратчайших путей.

Для повышения качества функционирования корпоративных сетей предложена практическая реализация применения алгоритма парных переходов в протоколе OSPF, позволяющая уменьшить трудоемкость построения таблиц маршрутизации до величины $O(k \cdot N)$.

Рассмотрим работу алгоритма парных переходов на базе протокола OSPF в корпоративных сетях. Укрупненная схема алгоритма имеет вид.

Шаг 1. Первоначальная инициализация исходных данных. Используя пакет HELLO протокола OSPF, определить веса линий связи $w_{i,j}$.

Шаг 2. Построить дерево оптимальных маршрутов корпоративной сети.

Шаг 3. Исходное множество ребер графа разбить на подмножество ребер, которые входят в дерево кратчайших путей и подмножество ребер замены, которые не входят в дерево кратчайших путей.

Шаг 4. Для каждого ребра графа корпоративной сети определить точку вхождения в дерево оптимальных маршрутов и точку вхождения во множество замены.

Шаг 5. Для вершины, являющейся листом дерева, произвести поиск всех парных переходов без ограничений. Эти списки для удобства дальнейшей работы привязываются к вершине, инцидентной рассматриваемому ребру и расположенной ниже по иерархии.

Шаг 6. Если вершина не является листом дерева, то вычислить парные переходы для этой вершины и выбрать лучшие значения потенциалов парных переходов для потомков вершины и собственных парных переходов. Подобная процедура выполняется для формирования списков парных переходов в случае динамического изменения нагрузки на линиях связи.

Шаг 7. Для каждой вершины формируется полный список парных переходов. Число элементов в каждом из этих списков не превышает количества вершин графа. Такое решение позволяет отказаться от предварительной сортировки потенциалов или приращений для парных переходов без значительного усложнения алгоритма обработки изменения.

Шаг 8. Определить, есть ли пакеты на передачу:

- а) если да, то перейти к шагу 9;
- б) иначе – к шагу 17.

Шаг 9. Используя поля «Время жизни» и «Контрольная сумма заголовка» протокола IP определить, требуется ли уничтожить (отбросить) данный пакет:

- а) если да, то перейти к шагу 25;
- б) иначе - к шагу 10.

Шаг 10. Используя поле «Тип сервиса», определить, требуется ли создание виртуального соединения:

- а) если да, то перейти к шагу 11;
- б) иначе – к шагу 13.

Шаг 11. Организовать виртуальное соединение, послав первый пакет «запрос вызова» адресату:

а) если пакет согласия на соединение от адресата пришел, то перейти к шагу 14;

- б) если адресат отклонил вызов, то перейти к шагу 13.

Шаг 12.

а) установить флаг передачи;
б) послать адресату пакет ликвидации соединения;
в) получить пакет подтверждения рассоединения от адресата.

Шаг 13. Проверка флага передачи:

- а) если флаг установлен, то перейти к шагу 25;
- б) иначе – к шагу 8.

Шаг 14. Используя поле «Тип сервиса», определить необходимую таблицу маршрутизации с учетом желаемого уровня качества обслуживания:

- а) если таблица 1, то перейти к шагу 15;

б) иначе – к шагу 16.

Шаг 15.

а) передать пакет, используя первую таблицу маршрутизации;

б) перейти к шагу 12.

Шаг 16.

а) передать пакет, используя вторую таблицу маршрутизации;

б) перейти к шагу 12.

Шаг 17. Анализируя полученную протоколом OSPF информацию, определить, произошло ли динамическое изменение нагрузки на линиях связи корпоративной сети:

а) если да, то перейти к шагу 18;

б) иначе - к шагу 8.

Шаг 18. Используя список парных переходов, определить, требуется ли сделать парный переход:

а) если да, то перейти к шагу 19;

б) иначе – к шагу 22.

Шаг 19. Для вершины, у которой потенциал уменьшился и у которой в список парных переходов входит ребро с изменившейся метрикой, определить путь минимальной длины и поместить ребро, которое привело к уменьшению потенциала вершины в дерево кратчайших путей, а сменившееся ребро из дерева кратчайших путей – во множество ребер замены.

Шаг 20. Определить, уменьшился ли потенциал других вершин графа корпоративной сети, расположенных выше по иерархии, после выполнения парного перехода:

а) если да, то перейти к шагу 21;

б) иначе – к шагу 22.

Шаг 21. Для каждой вершины, у которой потенциал уменьшился определить путь минимальной длины и просмотреть список парных переходов. Если новый минимальный путь для каждой вершины содержит ребро из списка парных переходов, то поместить данное ребро в дерево кратчайших путей, а ребро из дерева кратчайших путей – во множество ребер замены.

Шаг 22. Построить новое дерево оптимальных маршрутов с учетом изменений.

Шаг 23. Пересчитать точки вхождения в дерево и во множество замены, переформировать список парных переходов для каждой изменившейся вершины.

Шаг 24. Сформировать таблицы маршрутизации.

Шаг 25. Проверка окончания работы маршрутизатора:

- а) если да, то перейти к шагу 26;
- б) иначе – сбросить флаг передачи и перейти к шагу 8.

Шаг 26. Завершение работы маршрутизатора.

Для подтверждения правильности предложенного алгоритма парных переходов на базе протокола OSPF разработано программное обеспечение моделирования процессов маршрутизации в корпоративных сетях.

В разработанном пакете программ реализована возможность определять оптимальные маршруты передачи данных от узла-источника до всех остальных узлов сети с помощью алгоритма парных переходов на базе протокола OSPF для графов с различной структурой, производить построение таблиц маршрутизации необходимого участка сети. Кроме того, реализована возможность пошагового отображения работы алгоритма.

При разработке основное внимание уделялось корректности предлагаемого алгоритма и размерности решаемой задачи.

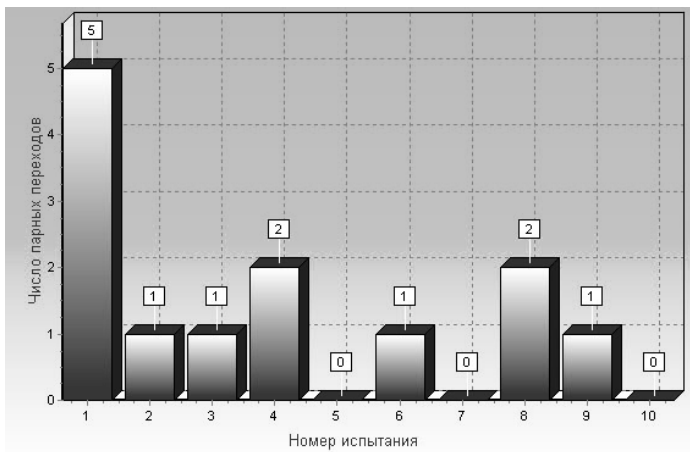


Рис. 2. Число изменений дерева в графе из 10 вершин

Для каждого испытания на множестве обработанных изменений выбиралось минимальное, максимальное и среднее значения размерности задачи, выраженное через количество вершин, для которых необходим поиск кратчайшего пути. Для каждого эксперимента были найдены значения оценок математического ожидания и среднего квадратичного отклонения числа изменений. Для предложенного алгоритма на базе протокола OSPF определялось число фактически выполнен-

ных парных переходов при динамических изменениях на линиях связи корпоративной сети.

На рис. 2 – 4 представлены результаты моделирования алгоритма парных переходов на базе протокола OSPF.

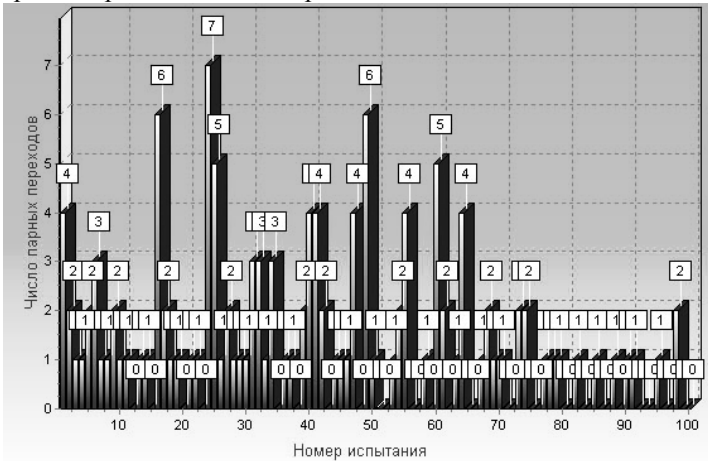


Рис. 3. Число изменений дерева в графе из 100 вершин

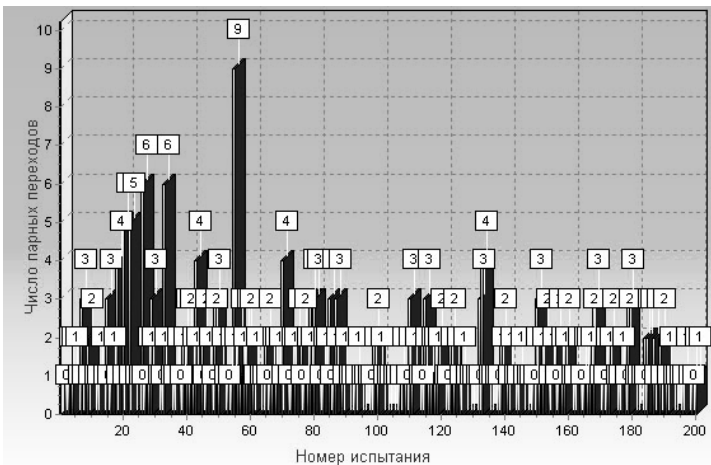


Рис. 4. Число изменений дерева в графе из 200 вершин

В таблице приведены обобщенные статистические характеристики для средней размерности задачи. В представленной таблице че-

рез СКО обозначено значение оценки среднего квадратичного отклонения.

Число вершин графа	Min значение	Max значение	Значение оценки МО	Значение оценки СКО
10	0	0,5	0,130	0,1226
100	0	0,07	0,013	0,0130
200	0	0,045	0,005	0,0057

Были проведены исследования графов, состоящих из 10, 100 и 200 вершин. Исследование разработанного алгоритма парных переходов показало, что максимальное значение числа изменений существенно ниже размерности для каждого из рассмотренных графов, а значение оценки математического ожидания числа изменений не превышает единицы. Более того, обнаружена тенденция уменьшения значения оценки математического ожидания числа изменений дерева оптимальных маршрутов с увеличением количества вершин графа.

На основе этого можно сделать вывод, что предложенный алгоритм парных переходов на базе протокола OSPF является эффективным при поиске оптимальных маршрутов в условиях динамического изменения на линиях связи корпоративной сети за счет использования дополнительной информации о возможных изменениях.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Уваров Д.В., Перепелкин А.И. Построение дерева кратчайших путей на основе данных о парных переходах // Системы управления и информационные технологии. Москва–Воронеж. 2004. №4 (16).

В.О. ПЕТУХОВА, Е.Л. ЛОБОВИКОВА

Рязанский государственный технический университет

ВАРИАНТ ЭНЕРГОСБЕРЕГАЮЩЕЙ «УМНОЙ» ЛАМПЫ - АНАЛОГ ЛЮМИНЕСЦЕНТНОЙ ЛАМПЫ ОСВЕЩЕНИЯ

Рассматриваются современные лампы на основе светодиодов.

Научно-технический прогресс не стоит на месте и к настоящему времени к светильникам предъявляются совершенно иные требования, чем были в XX-м веке. Основной акцент при разработке принципиально новых светильников ставится на их энергосберегающих свойствах и

экологической безопасности для окружающей среды. Весьма значителен резерв энергосбережения в системах освещения, на которые тратится до 20 % всей производимой в мире электроэнергии. Ведется целенаправленный поиск новых энерго- и ресурсосберегающих источников света с высокой световой отдачей до 100–150 Лм/Вт и сроком службы до 10 лет. В числе наиболее перспективных разработок — светодиоды.

Существующие светодиодные лампы предназначены для замены трубчатых люминесцентных ламп длиной 60 и 120 см. Лампочки светодиодные имеют стандартные цоколи для установки в светильники общего назначения. Бытовые энергосберегающие LED - лампы питаются непосредственно от сети переменного тока напряжением 220В без использования балластов, которые необходимы для запуска люминесцентных ламп.

Заменить светодиодной можно не только люминесцентную лампу, но и например фонари уличного освещения. Это осветительные приборы общего уличного освещения. Можно выделить два типа конструкции фонарей уличного освещения: торшерные фонари уличного освещения - осветительные приборы, которые крепятся на опору сверху и излучают свет вверх. Предусмотрены одно- и многорожковые модификации устройства, с разной высотой опоры. Вторую группу образуют консольные фонари уличного освещения, которые представляют собой светильники общего света, этот вид светильников направляет световой поток вниз.

Прогресс не стоит на месте и параллельно с разработкой новых конструкций энергосберегающих осветительных приборов, нужно вести разработку способов управления ими. Мы предлагаем сделать это управление умнее, используя искусственный интеллект. Основная идея состоит в автоматическом включении света в помещении при появлении там человека и автоматическом выключении, при его удалении. Принцип действия датчика движения несложен — при появлении движения в зоне обнаружения он замыкает цепь, тем самым включая подсоединенные к нему приборы. При отсутствии движения цепь автоматически размыкается, отключая все приборы. Применение данной технологии позволяет еще больше уменьшить энергопотребление осветительными приборами.

Расчетный срок службы светодиодной «умной» лампы составляет 100000 часов, т.е. более 10 лет непрерывной работы (у люминесцентных — 12000 часов — 1,5 года). Постоянный рост тарифов на электроэнергию, высокая энергоэффективность и столь длительный

срок службы делают оправданной покупку светодиодной продукции. В первую очередь это целесообразно для предприятий.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. М.Л.Бадгутдинов, Н.А.Гальчина, Л.М.Коган, И.Т.Рассохин Н.П.Сощин, А.Э.Юнович.: Мощные светодиоды белого свечения для освещения. Светотехника, 2006, N3, с. 36-40.
2. <http://ryazan.bestru.ru/help/show/328/>

Н.А. ПОДГОРНОВА

Рязанский государственный радиотехнический университет

РАЗРАБОТКА КОМПЛЕКСНОЙ СИСТЕМЫ КРАТКОСРОЧНОГО ПРОГНОЗИРОВАНИЯ ДЕНЕЖНЫХ ПОТОКОВ ПРЕДПРИЯТИЯ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ НЕЛИНЕЙНОЙ ДИНАМИКИ

Предлагается система краткосрочного прогнозирования нестационарных временных рядов потоков денежных средств предприятия как инструментарий качества модели управления остатками денежных средств, обеспечивающая высокую достоверность прогноза за счет использования предпрогнозного анализа на основе современного инструментария анализа временных рядов и обработки входных данных методами нелинейной динамики, фрактального анализа, вейвлет-преобразования и применения метода прогнозирования – инструментария нейросетевых технологий.

Существенной проблемой управления остатками денежных потоков является их неопределенность и значительная дисперсия, определяющиеся влиянием совокупности неуправляемых (слабоуправляемых) макро- и микрофакторов (состоянием рынка, финансовым состоянием потребителей, реализуемой ими политикой, низкой дисциплиной платежей и др.). Поэтому план-график поступления денежных средств практически всегда не выполняется. Но практически все известные модели основаны на ожиданиях платежей, предусмотренных планами поступлений, а отклонения от плановых поступлений имеют нормальный закон распределения, поэтому их применение в современных условиях снижает обоснованность принимаемых решений. Соответственно обоснование ожидаемых поступлений денежных средств рассматривается нами как задача каждодневного анализа и прогнозирования денежных потоков. Это превращает процесс обоснования метода и формирования инструментария прогнозирования де-

нежных потоков в один из ключевых вопросов совершенствования планирования ОДС предприятий.

Классические методы математической статистики, применяемые для исследования временных рядов (ВР) денежных потоков предприятия в условиях стабильной экономики, в этих условиях неадекватны. Инструментарий исследования и прогнозирования временных рядов методами статистики, ставшим классическим, дает эффективный прогноз только для систем, подчиняющихся нормальному закону распределения. Финансовые потоки предприятия обладают так называемой долговременной памятью, как правило, имеющие законы распределения, отличного от нормального и поэтому традиционные методы исследования и прогнозирования для решения задач этого класса не обеспечивают должной точности. Смена линейной парадигмы математическими методами нелинейной динамики и глобальный прогресс компьютерных технологий, сделавший возможным исследование сложных систем с помощью пакетов прикладных программ, позволил существенно повысить адекватность исследования и прогнозирования нестабильных денежных потоков. Современным аппаратом решения задач диагностики и прогнозирования являются методы нелинейной динамики, фрактального анализа, вейвлет-анализа, нечеткая логика, специальные искусственные сети и другие современные математические модели. Но опыт использования этих методов в управлении ОДС практически отсутствует. В отличие от классических методов, этот инструментарий позволяет выявить скрытые характеристики и закономерности поведения финансовых потоков предприятия. Обоснование инструментария прогнозирования должно включать совокупность рассматриваемых далее этапов анализа временных рядов, выбора методов формирования прогнозных моделей и их реализации.

Условием прогнозирования является определение характеристик и свойств временных рядов денежных потоков и их подготовка (предпрогнозный анализ). Для повышения точности прогноза зашумленного временного ряда разработана комплексная система предпрогнозного анализа и обработки временных рядов ежедневных денежных потоков, основанная на комплексном использовании инструментария предпрогнозного анализа (фрактального анализа, базирующегося на алгоритме R/S-анализа, и вейвлет-анализа). Система включает четыре блока.

1. Оценка временных рядов ежедневных изменений денежных потоков и выявление трендов, оценка наличия стационарности традиционными методами математической статистики. Основными задачами, решаемыми на этом этапе, являются исследование закона распределения и определение характеристик временных рядов.

С помощью пакета *Statistica 6.1* осуществлен анализ денежных потоков предприятия ОАО «АВС» классическими методами статистики. Расчеты статистических показателей и характеристик временных рядов ряда предприятий, проверка гипотезы о предполагаемом законе распределения позволили сделать вывод о том, что распределения вероятностей временных рядов потоков денежных средств предприятия не являются нормальными. Временные ряды денежных потоков сильно зашумлены и краткосрочны, слабовыраженный тренд воспринимается как случайные колебания. Функции распределений несимметричны относительно средней величины, выражены «толстые» правосторонние хвосты распределений, высокие пики и для случайных величин денежных потоков не выполняется «правило трех сигм». Выявлено, что исследуемые временные ряды являются нестационарными. В силу этого базирующиеся на аппарате математической статистики эконометрические методы анализа и прогнозирования, применяемые для данного вида временных рядов, оказываются неадекватными. Требуется разработка методов краткосрочного прогнозирования, адаптированных для этих условий.

2. Исследование временных рядов непрерывным вейвлет-преобразованием для оценки возможности прогнозирования временного ряда на основе специализированных пакетов прикладных программ. Проводится анализ и определяются: периодичность, тренд, локальные особенности, необходимый объем исходной выборки.

Для реализации непрерывного вейвлет-преобразования использована программа *Matlab 7* и применены вейвлеты Гаусса и Добеши. Спектрограммы сигнала денежного оттока, синтезированных гауссовым вейвлетом порядка 1 и Добеши порядка 1, представлены на рис. 1.

На графиках денежного оттока выделены моменты изменения поведения и структуры денежного оттока, определены его области непредсказуемости. Отток денежных средств является планируемым и более предсказуемым.

3. Предпрогнозная обработка исходных данных дискретным вейвлет-преобразованием для очищения временных рядов от шума с целью подготовки их для использования в качестве входных данных для обучения нейронной сети и построения качественного прогноза инструментарием нейросетевых технологий. Вейвлет-преобразование позволяет оптимизировать количество обрабатываемых данных без потери наиболее существенной информации и является необходимым условием для применения моделей прогнозирования.

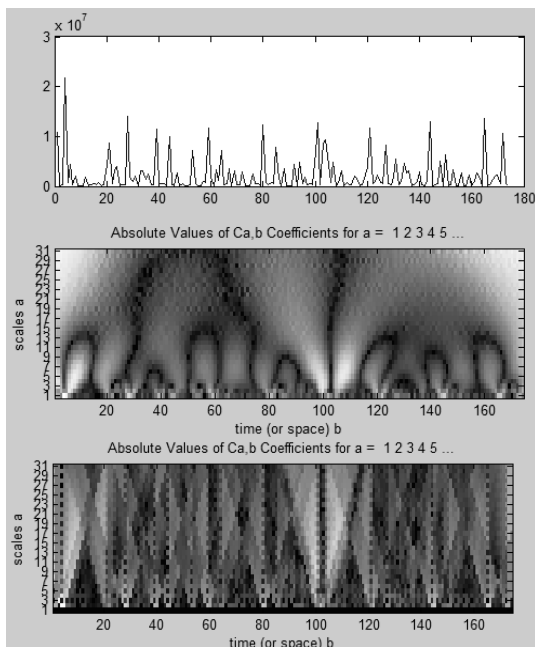


Рис. 1. Спектрограмма сигнала ежедневного оттока денежных средств предприятия ОАО «АВС»

После проведения ряда экспериментов в виде механизма обработки для очистки зашумленного сигнала в работе обоснованно используется короткий вейвлет *Добеши* 3-го порядка, 2 уровня разложения в программе *Deductor ver. 4.3* (рис. 2), в которой для удаления шума и сглаживании данных в программе уже автоматически установлены тип пороговой обработки и критерий расчета самого порога. При обработке сформировалась сглаженная версия исходного сигнала - отфильтровывался не только шум, но и некоторые локальные особенности (выбросы) исходного сигнала и полезная информация. Поэтому дополнительно использовались другие вейвлеты. После серий экспериментов в пакете *Wavelet Toolbox* системы *Matlab 7* (рис. 3) для очищения от шума временных рядов потоков денежных средств был выбран короткий вейвлет *Добеши* 3-го порядка, 2 уровня разложения и параметры, влияющие на качество шумоподавления сигнала.

4. Получение дополнительной информации о свойствах исследуемых временных рядов инструментарием фрактального анализа с целью оценки возможностей надежного прогнозирования. Метод нормирования размаха Херста позволил выявить особенности поведения ди-

намики в рассматриваемых временных рядах денежных потоков, получить такие характеристики, как трендоустойчивость и долговременная память, оказывающие существенное влияние на точность прогнозных оценок потоков денежных средств.

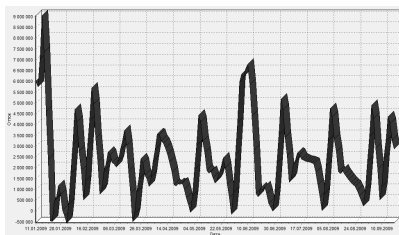


Рис. 2. Временной ряд оттока денежных средств предприятия после очищения от шума с помощью вейвлет-преобразования в программе Deductor ver.4.3

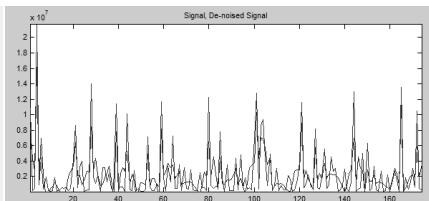


Рис. 3. Исходный временной ряд оттока денежных средств предприятия и после очищения от шума в системе Matlab 7

С помощью применения алгоритма адаптивного метода последовательного R/S-анализа временных рядов осуществлен анализ циклов и оценка глубины памяти. В результате применения метода последовательного R/S – анализа временных рядов, как на эталонных (искусственно-составленных) рядах, так и на модельной серии экспериментальных временных рядов Херста, представлена информация о динамике точек «срыва с тренда» R/S – траектории временных рядов денежных потоков (рис. 4). На рисунке 5 представлено геометрическое представление нечеткого множества глубины памяти для временного ряда оттока денежных средств предприятия (где $\mu(l)$ - это значения функции принадлежности «глубины l » нечеткому множеству).

На основании результатов фрактального анализа установлено наличие в исследуемых временных рядах потоков денежных средств предприятия трендоустойчивости, долговременной памяти и квазициклов. В исследуемых рядах денежных потоков обоих предприятий присутствует циклическая компонента, т.е. ряды демонстрируют наличие устойчивого «эффекта памяти». Глубина памяти конкретного временного ряда не является фиксированным числом; ее величина меняется вдоль рассматриваемого временного ряда, т.е. для различных его отрезков она является различной. Преимущественная размерность глубины памяти временного ряда притока и оттока денежных средств равна семи, что позволяет сделать вывод о построении прогноза для притока и оттока на основании данных предшествующей недели.

Полученные оценки глубины памяти дают предпрогнозную информацию для выбора и построения прогнозной модели.

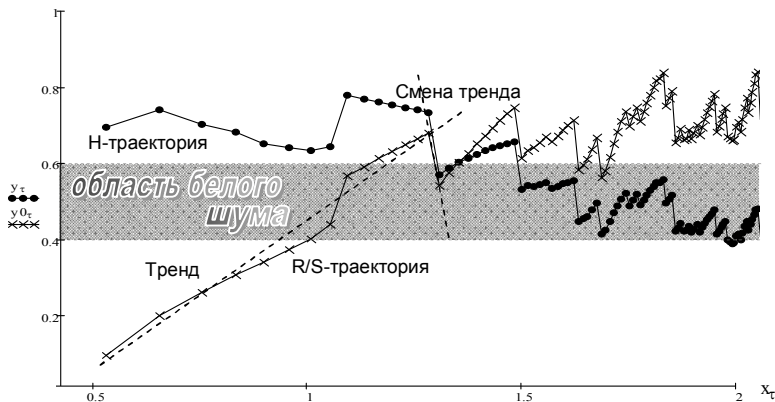


Рис. 4. - N- и R/S-траектории для временного ряда ежедневного денежного притока предприятия ОАО «ABC»

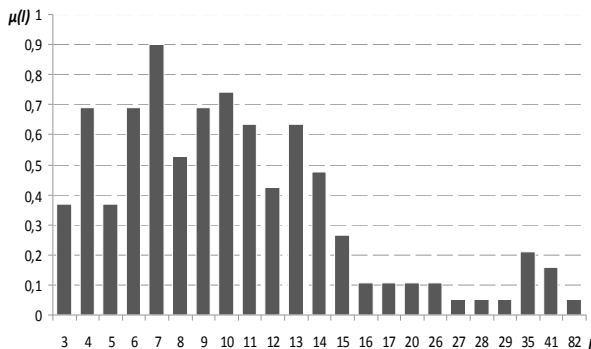


Рис. 5. Геометрическое представление нечеткого множества глубины памяти для ВР оттока денежных средств

В полной мере ко всем методам прогнозирования относятся проблемы оценки точности моделей и обоснование (выбор) наиболее адекватной. При обосновании модели прогнозирования мы исходим из следующего условия: наиболее адекватной является модель, обеспечивающая минимальную среднюю относительную ошибку отклонения за последние n точек исходного ряда ($n = 5$), предшествующих точке (периоду) прогнозирования (минимум ошибки аппроксимации *MAPE*). При этом совокупность методов и моделей прогнозирования ОДС,

рекомендуемая для отдельных предприятий, не является универсальной. Необходимым условием формирования такой модели является проведение исследований, содержание и последовательность которых приведена ниже.

1. Формирование исходной информации для анализа состояния и ожидаемой динамики денежных потоков. Для обоснования рационального представления исходной информации исследованы 3 варианта ее предоставления:

- временные ряды денежного притока ($PR1$) и денежного оттока ($OT1$) предприятия (рис. 1) без предварительной обработки;

- временные ряды денежного притока ($PR2$) и денежного оттока ($OT2$) предприятия (рис. 3), очищенные от высокочастотного шума с помощью вейвлет-преобразования с использованием короткого вейвлета 3-го порядка, 2 уровня разложения для сохранения полезной информации, распределения которых не подчиняются нормальному закону. Для подавления шумов фактических данных применялась система *Matlab7* с мягким режимом обработки сигнала для сохранения исходных данных. После серий экспериментов были выбраны параметры, влияющие на качество шумоподавления сигнала. Для шумоподавления оценивался спектральный состав шумовой компоненты, выбирался тип пороговой обработки детализирующих коэффициентов и критерий расчета самого порога;

- временные ряды денежного притока ($PR3$) и денежного оттока ($OT3$) предприятия (рис. 2), сглаженные (очищенные от шума и случайных выбросов) при помощи короткого вейвлета 3-го порядка, 2 уровня разложения, распределения которых являются гауссовскими. Для подавления шумов фактических данных применялась программа *Deductor ver.4.3*. При удалении шума и сглаживания данных в программе автоматически установлены параметры выбора порогового уровня фона (оценки дисперсии шума) и критерий его расчета.

Жесткие статистические предположения о свойствах временных рядов ограничивают возможности методов математической статистики. В последнее время для сложных, нестационарных временных рядов, характерных для потоков денежных средств предприятия, адекватным аппаратом для решения задач диагностики и прогнозирования стали применяться нейронные сети, которые являются самообучающимися системами и позволяют вытянуть скрытые в потоке информации закономерности. В соответствии с этим нами использован нейросетевой инструментарий в качестве модели прогнозирования.

В качестве сети для построения прогноза временных рядов денежных потоков с использованием нейросетевых технологий после

многочисленных экспериментов был выбран многослойный персептрон с алгоритмом обучения сети методом обратного распространения ошибки в нейропакете *Statistica Neural Networks*.

С учетом предпрогнозных характеристик временных рядов, полученных с помощью фрактального и вейвлет-анализа, принимаются в качестве параметров временных окон: $n = 5$ – количество значений, по которым строится прогноз (учитывая доминирующую размерность квазициклов); $m = 5$ – горизонт прогноза (ежедневный прогноз выполняется непрерывно на пять дней в соответствии с принципами, заложенными в модели); $s = 1$ – шаг сдвига окон. Наилучшие прогнозные оценки были получены при построении нейросетевого прогноза для исходных данных временных рядов потоков денежных средств, сглаженных и очищенных от шума, случайных выбросов, нелинейных искажений посредством инструментария дискретного вейвлет-преобразования (в программе *Deductor*).

В ходе исследований найдены типы нейронных структур для 3 видов данных исследуемых временных рядов денежных потоков. На рисунке 6 представлена архитектура нейронной сети *MLP* для временных рядов ежедневных денежных притоков (*PR3*) и оттоков (*OT3*) предприятия (очищенных от шума с помощью вейвлет-преобразования с автоматической подбором и установкой в программе *Deductor* шумоподавляющих параметров).

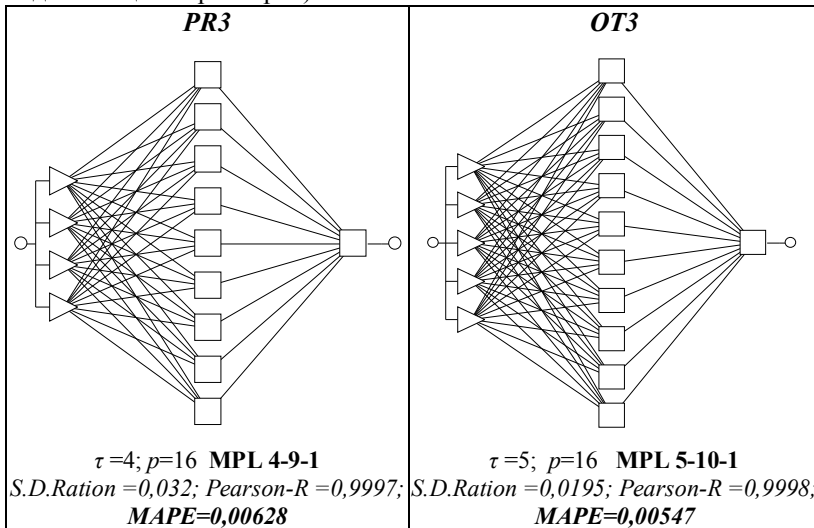


Рис. 6. Архитектура *MLP*-нейросети для временных рядов потоков денежных средств предприятия

Прогнозируемые временные ряды (ретроспективные значения), построенные с помощью нейронных сетей, имеют очень близкие значения к исходному ряду (практически полностью совпадают) и имитируют его незначительные изменения на протяжении временной оси. Ошибка аппроксимации $MAPE = 0,628\%$ по притоку, $MAPE = 0,547\%$ по оттоку денежных средств. В таблице 1 представлены ретроспективные (за 5 дней) и прогнозные (за 5 дней) значения денежного притока и оттока (предварительно очищенные от шума с помощью вейвлет-преобразования с использованием системы Deductor).

Таблица 1. Ретроспективные и прогнозные значения денежного притока и оттока, тыс. руб.

Дни	Приток ДС, тыс. руб.	PRЗ		Отток ДС, тыс. руб.	ОТЗ	
		Прогноз притока ДС, тыс. руб.	Относит. ошибка, ε		Отток притока ДС, тыс. руб.	Относит. ошибка, ε
1	2736,652	2734,161	0,00091	2404,208	2420,873	0,006932
2	2638,016	2605,76	0,012227	3187,769	3204,372	0,005208
3	2008,326	2003,219	0,002543	3566,475	3560,538	0,001665
4	1102,179	1089,449	0,01155	3352,475	3311,5	0,012222
5	441,4118	439,572	0,004168	2773,836	2777,6	0,001357
		MAPE	0,00628		MAPE	0,005477
1		429,5769			2267,465	
2		1128,548			2182,225	
3		2254,709			2570,199	
4		3354,841			3175,666	
5		4041,594			3609,653	

Результаты применения моделей прогнозирования показывают, что к выбору методов предпрогнозной обработке исходных данных и параметрам шумоподавления необходимо подходить наиболее тщательным образом, учитывая компромисс между точностью построения прогноза и высокого уровня сглаживания данных, что несет за собой потерю информации об исходном ряде.

В целом нейронные сети показали высокие результаты качества прогноза и получения прогнозных значений. В соответствии с этим сделан вывод о том, что наиболее надежным инструментарием прогнозирования для исследованных в работе временных рядов являются нейросетевые технологии – многослойный персептрон с алгоритмом обучения сети методом обратного распространения ошибки.

На основе проведенного исследования выявлено:

1. С помощью инструментария фрактального анализа, базирующегося на алгоритме R/S-анализа временных рядов и непрерывного вейвлет-преобразования сигнала временных рядов, могут быть получены предпрогнозные характеристики исходной информации и информация о возможности прогнозирования временных рядов денежных потоков.

На основании полученных результатов фрактального анализа адаптированного метода последовательного R/S-анализа временных рядов установлено наличие в исследуемых временных рядах потоков денежных средств предприятия трендоустойчивости, долговременной памяти и квазициклов.

Использование дискретного вейвлет-преобразования для обработки данных от шума и случайных искажений и подготовки для более качественного прогноза ОДС является необходимым условием для применения моделей прогнозирования.

2. Анализ динамики денежных потоков нейросетевыми методами в комбинации с подходами, развитыми в экономфизике, методов нелинейной динамики позволяет исследовать её на способность быть прогнозируемой, выделить периоды возможности построения более качественного прогноза и области непредсказуемости.

При выборе архитектуры сети и способа прогнозирования необходима глубокая предварительная обработка данных, используя методы теории сложных систем: фрактальный анализ, вейвлет-анализ, методы нелинейной динамики. Предварительный анализ позволяет оптимизировать параметры нейросети, определить горизонт прогноза и провести сравнение качества предсказаний для временных рядов с различными видами исходных данных.

3. Жесткие статистические предположения о свойствах временных рядов потоков денежных средств, ограничивают возможности методов математической статистики и применяемые методы прогнозирования для данного вида временных рядов оказываются неадекватными. Поэтому для получения прогнозных значений временных рядов ежедневных потоков денежных средств предложено использование инструментария нейросетевых технологий. В ходе исследования различных модификаций архитектуры многослойных перцептронов нейронных моделей и в качестве алгоритма обучения сетей и построения наиболее надежного и точного прогноза доказана целесообразность использования метода обратного распространения ошибки в нейропакете *Statistica Neural Networks*.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Добеши Я. Десять лекций по вейвлетам. М.: РХД, 2001.
2. Петерс Э. Фрактальный анализ финансовых рынков: Применение теории Хаоса в инвестициях и экономике. – М.: Интернет-трейдинг, 2004. – 304с.
3. Петерс Э. Хаос и порядок на рынках капитала. Новый аналитический взгляд на циклы, цены и изменчивость рынка/ пер. с англ. – М.: Мир. 2000. – 333 с. ил.
4. Смоленцев Н.К. Основы теории вейвлетов. Вейвлеты в MATLAB – ДМК Пресс, 2005. – 304 с. Тихонов, Э.Е. Методы прогнозирования в условиях рынка: Учебное пособие / Тихонов Э.Е. – Невинномысск, 2006. – 221 с.

В.А. РЕВУЦКИЙ

Рязанский государственный радиотехнический университет

АЛГОРИТМ ОБНАРУЖЕНИЯ И ОЦЕНКИ ПАРАМЕТРОВ СВЕРТОЧНЫХ ПОМЕХОУСТОЙЧИВЫХ КОДОВ

Обоснован алгоритм обнаружения и оценки параметров сверточных помехоустойчивых кодов (ПК), применяемых в системах спутниковой связи (ССпС). Показано, что предложенный алгоритм обеспечивает правильное обнаружение и оценку параметров рассматриваемых ПК в 81 % случаев при значениях вероятности битовой ошибки в анализируемой двоичной последовательности (АДП) не более 10^{-3} .

Введение. На сегодняшний день в современных ССпС используется множество различных стандартов связи, в состав которых входят ПК различного типа и параметров [1...3].

Возможны ситуации, когда необходимо организовать сеанс радиосвязи, не имея полной информации о компонентах используемого стандарта связи. При этом одной из неизвестных составляющих стандарта спутниковой связи могут быть тип и параметры используемого ПК. Данным положением обусловлена актуальность разработки алгоритма обнаружения и оценки параметров сверточных ПК, устойчивого к ошибкам в канале связи. Рассматриваемый алгоритм выполняет обнаружение сверточного ПК в АДП и, если код обнаружен, выполняет процедуры по оценке параметров кодера.

Обоснование схемы предложенного алгоритма. Учитывая, что помимо системы помехоустойчивого кодирования в состав стандарта СпС входят и другие составляющие [1], исходными данными для рассматриваемого алгоритма является двоичная последовательность, со-

держащая известный несоставной ПК. При этом имеются сведения о положении и виде последовательностей синхронизации по ПК и параметрах перемежения.

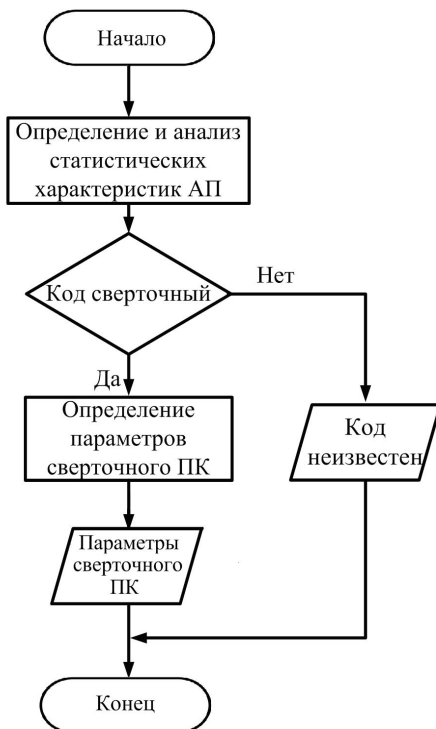


Рис. 1. Алгоритм обнаружения и оценки параметров сверточных ПК

Общая структура предложенного алгоритма соответствует процедуре поэтапного устранения неопределенности о компонентах стандарта СпС, входящих в состав системы ПК (рисунок 1).

Процедура распознавания типа ПК основана на оценке критерия статистической зависимости между сегментами АДП, содержащей сверточный код или коды других типов, в том числе псевдослучайные последовательности. При этом в [2..4] показано, что последовательности сверточных ПК характеризуются наиболее сильной статистической зависимостью, оцениваемой по критерию:

$$K_1 = 2^{2-H} / V(\{R_{\max}\}), \quad (1)$$

где $\{R_{\max}\}$ множество отсчетов $W(N_i, N_j)$ объемом $V(\{R_{\max}\})$, превышающих ее среднее значение $E(W(N_i, N_j))$ и $H = n_{\text{сун}} m_{\text{сун}}$ - длина сегментов АДП, определяемая из гипотетических параметров $n_{\text{сун}}, m_{\text{сун}}$ сверточного ПК.

При оценке параметров m и n сверточного ПК возможны следующие подходы:

- анализ поведения значения критерия K_1 от интервала между сегментами АДП для некоторого фиксированного $n_{\text{сун}}$;
- анализ зависимости среднего минимального расстояния Хемминга – $d_{\min, \text{ср.}}$ в $\{R_{\max}\}$ от $m_{\text{сун}}$ для некоторого фиксированного $n_{\text{сун}}$.

Экспериментальная часть. Проведены эксперименты по проверке предложенного алгоритма обнаружения и оценки параметров сверточных ПК на основе его многократных испытаний. Так в эксперименте использовались сверточные коды со следующими параметрами m, n, k : 3.3.1; 5.3.1; 8.3.1; 3.2.1; 5.2.1; 7.2.1; 3.3.2; 5.3.2; 7.3.2, блочные коды БЧХ с n, k параметрами: 15.5; 31.11; 15.7; 31.16; 15.11; 31.21 и коды Рида-Соломона с аналогичными параметрами.

Так экспериментально доказано [5], что при разбиение АДП на сегменты, равные различному значению длины кодового ограничения и дальнейшей оценке закона распределения вероятности совместного появления сегментов определенного вида - $W(N_i, N_j)$, критерий статистической связи между сегментами K_1 при наличии сверточного ПК в АДП не опускается ниже порога $K_{1, \text{пор}} = 7, 8$.

Также экспериментально доказано, что каждый из приведенных подходов по оценке параметров сверточных ПК позволяет сузить первичный список гипотетических параметров $A_0 = \{m_{\text{сун}1}, n_{\text{сун}1}, k_{\text{сун}1}; m_{\text{сун}2}, n_{\text{сун}2}, k_{\text{сун}2}; \dots; m_{\text{сун}l}, n_{\text{сун}l}, k_{\text{сун}l}\} \Rightarrow A_1$.

По результатам проведенных экспериментов можно сделать общий вывод об устойчивости разработанного алгоритма.

Вывод. Полученный алгоритм обнаружения и оценки параметров сверточных ПК позволяет обнаружить сверточный код из гипотетического списка A_0 и оценить его параметры с общей вероятностью ошибки классификации $P_{\text{ош. кл}} \leq 0.19$, если вероятность битовой ошибки в АДП $P_{\text{ош. б}} \leq 10^{-3}$.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Proximity-1 Space Link Protocol—Coding and Synchronization Sublayer. Blue Book. Issue April 2003. Электронный ресурс. Режим доступа: <http://public.ccsds.org/publications/SLS.aspx>
2. Скляр Б. Цифровая связь, Пер. с англ. – М. : Издательский дом “Вильямс”, 2003. – 1104 с.
3. Блейхуд Р. Теория и практика кодов, контролирующих ошибки. М.: Мир, 1986 – 576 с.
4. Никитин Г.И., Сверточные коды. – СПб.:СПбГУАП, 2001. – 80с.
5. Ревуцкий В.А., Алгоритм классификации помехоустойчивых кодов, Межвузовский сборник «Информационные технологии» РГРТУ – 2011, С. 126-130

С.В. СКВОРЦОВ, М.А. КОЗЛОВ

Рязанский государственный радиотехнический университет

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМА СОРТИРОВКИ ДЛЯ МНОГОЯДЕРНЫХ ПРОЦЕССОРОВ

Рассматривается парадигма функционального программирования для автоматизации распараллеливания алгоритма сортировки слиянием с помощью языка Clojure.

Конкурентное программирование связано с управлением доступом к общим ресурсам из потоков, в свою очередь параллельное программирование - связано с использованием нескольких ядер (процессоров). На сегодняшний день для конкурентного программирования есть вполне отточенные техники и приемы. Так же существует некоторое количество библиотек к различным языкам, чтобы помочь им управлять многопоточностью, например: для Java есть библиотека Concurrency, для C есть pthread, OpenMP [2].

В данной статье мы затронем параллельное программирование, реализующее параллелизм на уровне обработки данных и покажем как с помощью современных языков можно достичь большей производительности в алгоритме сортировке слиянием [1]. Для этого воспользуемся функциональную парадигму, реализованную в современном языке программирования Clojure - это современный диалект Лиспа, так же он является языком общего назначения с поддержкой разработки в интерактивном режиме и упрощающий поддержку параллельного программирования [4].

Для исследования предложенного подхода написана программа,

которая выполняет сортировку данных представленных в виде связанного однонаправленного списка. Для обработки списка будет использоваться функция `map`. Во многих языках программирования `map` — функция высшего порядка, которая применяет данную функцию к каждому элементу списка, возвращая список результатов. При рассмотрении в функциональной форме она часто называется "применить ко всем". В нашем случае, применительно к сортировке, данная функция будет применять функцию слияния к каждой части исходного списка. Мы же хотим сделать так, чтобы метод "разделяй и властвуй" мог работать в параллельном режиме, для этого будем использовать функцию `rmap`, что значит `parallel map`. `Rmap` подобна `map`, только выполняется параллельно. Данная функция полезна тогда, когда нам нужно максимально увеличить производительность несмотря на накладные расходы связанные с затратами оперативной памяти [4].

Изменения в классическом алгоритме слияния коснулись рекурсивного спуска во время которого выполняется деление списка на подсписки. При спуске каждая итерация слияния выделяется в отдельную нить и ожидает выполнения. Как только останутся единственные элементы в сливаемых подсписках происходит процесс возврата и каждая пара подсписков сливается в отдельном потоке. Тем самым мы исключаем время простоя и получаем, что задействуются все свободные ядра [1].

Clojure пользуется API JVM для работы с многопоточными приложениями, но в отличие от Java он использует его более эффективно за счет функциональной парадигмы. Clojure обеспечивает не изменчивость (`immutable`) данных, тем самым исключая проблемы блокировки ресурсов [4].

Преимущество которые дает нам данная техника программирования:

- программирование становится более простым;
- единый поток управления программой;
- неявная синхронизация.

Для исследования предложенного алгоритма были проведены эксперименты на двухядерном и четырехядерном процессорах фирмы Intel под управлением операционной системы Linux, версия JDK 1.6.0.31, версия Clojure 1.3. Исходные данные для сортировки формировались случайным образом и были представлены в виде целочисленных списков достаточно большой длины (1000, 10000, 100000, 1000000 элементов). Сравнение соответствующих программ сортировки выполнялось по средним значениям времени работы,

полученным по 10 опытам для каждого размера массива. При этом для сравнения был реализован последовательный (однопоточный) алгоритм сортировки слиянием. Для измерения времени выполнения использовалась функция `time` из стандартной библиотеки языка Clojure.

Таблица 1. Результаты испытаний

Алгоритмы сортировки	Количество элементов			
	1000	10000	100000	1000000
Parallel merge sort (2-яд/4-яд)	388/312	938/793	1732/1504	15780/13221
Merge sort (2-яд/4-яд)	504/423	1679/1544	2145/2031	21503/19529

После проведенных испытаний можно сделать вывод, что функциональное программирование способно составить конкуренцию классическому подходу в параллельном, многоядерном программировании. С помощью новых парадигм возможно увеличить быстродействие классических алгоритмов обработки данных, что показано на примере алгоритма сортировки слиянием.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Робертс Д., Эхтер Ш. Многоядерное программирование. – СПб.: Питер, 2010. – 320 с.
2. Богачев К. Ю. Основы параллельного программирования. – М.: БИНОМ. Лаборатория знаний, 2010. – 342 с.
3. Камерон Хьюз, Трейси Хьюз Параллельное и распределенное программирование с использованием C++. – М.: Вильямс, 2004 г. – 672 с.
4. Michael Fogus, Chris Houser The Joy of Clojure: Thinking the Clojure Way. Manning Publications; 1 edition (April 4, 2011), 300 pages.

Н.В. СКВОРЦОВ, С.В. СКВОРЦОВ, В.И. ХРЮКИН

Рязанский государственный радиотехнический университет

**РАЗРАБОТКА ПРОГРАММНЫХ СРЕДСТВ
ТЕОРЕТИКО-ВЕРОЯТНОСТНЫХ РАСЧЕТОВ
В ЕДИНОЙ СИСТЕМЕ ДОПУСКОВ И ПОСАДОК**

Рассматриваются вопросы реализации расчетов параметров механических соединений с учетом вероятностного характера размеров компонентов.

При решении задачи расчета параметров соединений деталей и узлов электромеханических компонентов вычислительной техники наиболее часто используется детерминистический подход, предполагающий применение расчетных формул единой системы допусков и посадок (ЕСДП) [1]. Эта система соответствует международному стандарту ISO 286.

При таком подходе отклонения размеров от номинальных считаются предельными, а допуск посадки определяется как сумма допусков отверстия и вала, составляющих соединение. Здесь вал — термин, условно применяемый для обозначения наружных элементов деталей, включая и нецилиндрические элементы; отверстие — термин, условно применяемый для обозначения внутренних элементов деталей, включая и нецилиндрические элементы; допуск размера вала (отверстия) T_d (T_D) - разность между наибольшим и наименьшим предельными размерами или алгебраическая разность между верхним и нижним отклонениями [1]. В этом случае обеспечивается полная взаимозаменяемость изделий. Однако при таком способе расчетов ужесточаются требования к точности размеров, что приводит к удорожанию конструкций.

Существует возможность, позволяющая снизить требования на точность изготовления деталей и узлов практически без ухудшения параметров их соединений. Для этого используется теоретико-вероятностный подход. Он основан на предположении, что поля допусков подчиняются определенным законам распределения. Так при нормальном законе распределения полей допусков вала и отверстия, что характерно для серийного производства изделий, распределение поля допуска соединения будет подчиняться нормальному закону. При равномерном законе распределения допусков вала и отверстия, что характерно для индивидуального производства изделий — закону Симпсона. Это дает возможность использования стандартных соотношений ЕСДП для расчета допусков и посадок с учетом поправочных

коэффициентов. Тогда допуск соединения в соответствии с [1] будет определяться по формуле:

$$TS = \frac{1}{k_s} \sqrt{k_d^2 \cdot Td^2 + k_D^2 \cdot TD^2},$$

где k_d , k_D , k_s – поправочные коэффициенты для допусков вала, отверстия и их соединения соответственно.

Значения этих коэффициентов для указанных распределений [1]: $k=1$ для нормального закона распределения; $k=1,73$ для равномерного закона распределения; $k=1,22$ для закона распределения Симпсона.

В этом случае, например, для нормального закона распределения допуски вала и отверстия можно увеличить приблизительно в 1,4 раза по сравнению с допусками, обеспечивающими полную взаимозаменяемость, при одном и том же (фиксированном) допуске соединения. При этом только у 0,27% всех соединений возникает возможность появления брака. Очевидно, что такой подход позволит существенно снизить стоимость изготовления деталей и сборки узлов.

Однако использование коэффициентного метода расчета имеет ограничения. Он применим только в двух случаях: при нормальном законе распределения полей допусков вала и отверстия; при равномерном законе распределения полей допусков вала и отверстия и соизмеримости величин этих допусков. При других обстоятельствах возникает необходимость в определении коэффициента k_s , который не будет совпадать ни с одним из указанных выше значений. То же самое касается коэффициентов k_d , k_D , если поля допусков вала или отверстия не подчиняются рассмотренным выше законам распределения.

Для вычисления допуска соединения в самом общем случае, т.е. при произвольных законах распределения полей допусков вала и отверстия и при отсутствии ограничений на величины этих допусков, предлагается использовать методы вероятностного моделирования.

В этом случае задачу можно сформулировать следующим образом. Имеется случайная величина (с.в.) X с исходной плотностью распределения $f(x)$. Следует произвести ее функциональное преобразование $Y=\varphi(X)$ так, чтобы с.в. Y имела заданное распределение $G(y)$, соответствующее распределению поля допуска вала или отверстия.

При практической реализации [2] принимают исходную с.в. X распределенной с постоянной плотностью на интервале $(0, 1)$. Затем подвергают с.в. X функциональному преобразованию

$$y = G^{-1}(x),$$

где $G^{-1}(x)$ – функция, обратная требуемой $G(y)$.

В этом случае величина допуска соединения будет определяться как сумма значений допусков вала и отверстия, полученных в соответ-

ствии с представленным выше способом. Многократно реализуя вычисление значений допуска соединения, можно определить его максимальное значение. Используя эту величину, можно найти реальные значения предельных зазоров или натягов в соединении.

Обобщая вышеизложенное, можно предложить следующий порядок выполнения расчетов параметров соединений.

1. Анализируются технологические процессы изготовления деталей и сборки узлов.

2. В случае необходимости обеспечения полной взаимозаменяемость следует использовать для расчета стандартные формулы ЕСДП.

3. При отсутствии вышеуказанного ограничения расчет производится коэффициентным методом, если распределения полей допусков вала и отверстия подчиняются нормальному или равномерному закону одновременно. В последнем случае также требуется, чтобы вал и отверстие выполнялись по одному качеству.

4. Если эти требования к законам распределения не выполняются, следует использовать вероятностные методы расчета параметров соединений.

В соответствии с предложенной методикой была разработана информационно-справочная среда по ЕСДП. Она предназначена для предоставления пользователю справочной и расчетной информации о размерах сопрягаемых деталей, а также о допусках и посадках гладких цилиндрических и плоских соединений. Помимо прямой задачи – определению параметров соединения деталей по условному обозначению полей допусков вала и отверстия, в среде реализуется обратная задача - по величинам предельных зазоров или натягов производится выбор стандартных посадок. Кроме того программа предоставляет справочную информацию по посадкам подшипников качения в соответствии со стандартом ISO 9001-2008 (ГОСТ 3325-85).

Разработанная система имеет стандартный Windows-интерфейс и тем самым обеспечивает простоту восприятия и понимания. Поэтому работу с системой может осуществлять любой непрофессиональный пользователь, обладающий знаниями в предметной области и минимальными навыками работы с компьютером.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Якушев А.И., Воронцов Л.Н., Федотов Н.М.. Взаимозаменяемость, стандартизация и технические измерения. – М.: Машиностроение, 1987. – 352 с.

2. Вентцель Е.С., Овчаров Л.А. Теория вероятностей и ее инженерные приложения. – М.: Высшая школа, 2000. – 480 с.

О.В. СКРИПКИНА, Ф.Н. ГУСЕЙНОВА

Рязанский государственный радиотехнический университет

ИСПОЛЬЗОВАНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ПРИ СДАЧЕ БУХГАЛТЕРСКОЙ (ФИНАНСОВОЙ) ОТЧЕТНОСТИ В КОНТРОЛИРУЮЩИЕ ОРГАНЫ

Рассматриваются информационные технологии, с помощью которых осуществляется сдача бухгалтерской (финансовой) отчетности через интернет. Дается сравнительная характеристика систем электронного защищенного документооборота.

В настоящее время все экономические субъекты могут сдавать бухгалтерскую (финансовую) и налоговую отчетность в контролирующие органы в электронном виде без дублирования на бумаге, используя обычный доступ в Интернет. Наряду с этим налогоплательщики имеют возможность без посещения инспекции получать необходимые информационные и распорядительные материалы из налоговых органов, а также выписки о состоянии расчетов с бюджетами разных уровней.

Преимущества электронного способа сдачи отчетности для организаций очевидны. Во-первых, программа контролирует правильность заполнения высылаемых организацией документов на соответствие формату ФНС. Во-вторых, вся информация между организацией и инспекцией передается в зашифрованном виде, что, в отличие от бумажных носителей, обеспечивает ее полную безопасность и конфиденциальность. В-третьих, один бухгалтер со своего рабочего места может отправлять отчетность в налоговые инспекции от лица всех филиалов компании, даже в инспекции, располагающиеся в других городах России. В-четвертых, при использовании системы происходит автоматическая выгрузка данных из бухгалтерских программ в программу формирования электронной отчетности. Кроме того, формы отчетности и реквизиты инспекций и счетов, необходимые для налоговых вычетов, обновляются своевременно. Следует отметить, что каждый документ, обращающийся в системе электронного документооборота, обязательно заверяется электронной цифровой подписью уполномоченного сотрудника организации, что делает его юридически эквивалентным бумажному документу.

У контролирующих органов так же есть преимущества. Присланные отчитывающейся организацией сведения автоматически разносятся на лицевые счета, что исключает необходимость в переносе данных, как следствие, исчезают связанные с этим опечатки. Представленная информация обрабатывается быстрее, что делает процесс

выверки платежей более оперативным. Кроме того, исключается потеря времени на стояние бухгалтеров в очередях в налоговой инспекции, а также расширяются рамки сдачи налоговой отчетности, поскольку налогоплательщик может отправить файлы деклараций и других отчетных документов до 24 часов последнего дня сдачи отчетности.

Сегодня практически во всех организациях ведение бухгалтерского учета автоматизировано. Уже невозможно представить бухгалтера, стоящего в очереди в налоговую инспекцию для сдачи отчетности, заполненной от руки. В ближайшем будущем уже будет невозможно представить бухгалтера просто стоящего в очереди для сдачи отчетности. Ведь затраты на подключение системы электронного защищенного документооборота незначительные. Размер ежемесячной абонентской платы сопоставим с издержками на бумагу для распечатки документов. Решение остается за налогоплательщиком, но эти незначительные для большинства хозяйствующих субъектов затраты окупаются сэкономленными часами рабочего времени бухгалтера.

В настоящее время практически в каждом регионе действуют несколько специализированных операторов, предлагающих различные решения по безбумажному и бесконтактному документообороту организаций и налоговых органов. Всего в стране сдают отчетность через Интернет порядка 770000 налогоплательщиков. Рассмотрим системы, применяемые для сдачи отчетности через Интернет.

Система «Такском-Спринтер» основана на технологии, разработанной компанией «Такском», являющейся системным интегратором проекта, совместно с ведущим разработчиком бухгалтерских программ - фирмой «1С». Все формы отчетности, в том числе обновления, разрабатываются и тестируются совместно с «1С». Разработка, апробация и внедрение технологии осуществлялись с 2000 года в Москве, а затем в Московской области, Татарстане и ряде других регионов. В настоящее время система представления налоговой отчетности по каналам связи «Такском-Спринтер» представлена практически во всех регионах, где вместе с компанией «Такском» ее внедрением занимаются свыше 70 организаций-партнеров. В Москве более 80% налогоплательщиков, сдающих отчетность в электронном виде, используют эту популярную и проверенную систему. Основными достоинствами системы «Такском-Спринтер», отличающими ее от других систем, являются следующие:

- это первая технология, которая была предложена для массового представления налоговой отчетности в электронном виде по каналам связи;

- предлагаемые средства эффективно и надежно работают как при хорошем, так и при плохом качестве подключения абонентов к сети Интернет.

- в организации устанавливается собственное программное обеспечение для формирования, отправки и хранения отчетности на рабочем месте. В Интернет выход необходим только на несколько секунд для отправки отчетности и получения подтверждения спецоператора связи о прохождении отчетности.

Особенность технологии *системы «Контур-Экстерн»* заключается в том, что из всех представленных на рынке она единственная предоставляет абоненту возможность вводить все данные без установки какого-либо специального программного обеспечения подготовки данных на компьютере бухгалтера.

Все обновление клиентского программного обеспечения при изменениях в законодательстве в системе происходит автоматически. Компания СКБ «Контур» была образована в 1988 году в Екатеринбурге и, по сути, является ровесницей IT-индустрии в стране. Информация в системе «Контур-Экстерн» циркулирует с высокой скоростью, что ведет к повышению качества и эффективности ее учета и обработки.

Преимущества системы Контур-Экстерн:

- более современная и совершенная технология (веб-технология) на основе которой разработана данная система;

- возможность проводить электронную сверку с Пенсионным фондом РФ. С ФНС сейчас проводится сверка практически во всех системах;

- после отправки отчетности через интернет в контролирующие инстанции организация мгновенно получает подтверждение спецоператора связи о том, что отчетность отправлена.

Помимо, этих основных преимуществ, система электронной отчетности Контур-Экстерн способна представлять отчетность через интернет в контролирующие органы на всей территории Российской Федерации.

Клиентами системы являются многие государственные и муниципальные учреждения РФ, крупнейшие организации и банки. В становлении и развитии системы «Контур-Экстерн» активное участие принимают государственные контролирующие органы.

Система «Астрал Отчет» является доступным, простым в использовании решением, позволяющим безопасно и оперативно передавать конфиденциальные данные посредством глобальной сети Интернет.

«Астрал Отчет» позволяет:

- сдавать отчетность через интернет в ФНС, ПФР, Росстат, Фонд социального страхования, не покидая своего рабочего места;
- исключить потерю времени на пребывание в очередях в контролирующих органах при сдаче отчетности;
- расширить временные рамки представления отчетности до 24.00 последнего дня сдачи отчетности;
- автоматически получать обновление форм отчетности и программного обеспечения непосредственно на рабочее мест;
- заполнять формы отчетности непосредственно в программе «Астрал-Отчет».

К достоинствам данного программного комплекса относятся простота установки (не требуется выезд специалиста) и отсутствие первичной настройки. Все данные об организации вводятся сотрудником «Астрал-Отчет».

Как следует из вышеизложенного, у каждой системы есть свои преимущества. Хозяинствующий субъект самостоятельно выбирает систему электронного защищенного документооборота, отвечающую всем его требованиям.

На современном этапе развития общества в Российской Федерации в процессы повседневной деятельности активно внедряются новые высокопроизводительные информационно-телекоммуникационные технологии. В этих условиях использование телекоммуникационных систем для обеспечения электронного документооборота между государственными контролирующими органами и хозяйствующими субъектами приобретает первостепенное значение.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Федеральный закон от 06.04.2011 № 63-ФЗ "Об электронной подписи».
2. Мишустина М.В. Налоговые декларации в электронном виде: решения и перспективы>//Российский налоговый курьер. 2011. N4.
3. Информационные технологии управления / под ред. Черкасова Ю.М. – М.: ИНФРА–М, 2010. – С.7.

В.В. СОЛДАТОВ, О.В. СОЛДАТОВА

Рязанский государственный радиотехнический университет

**МОНИТОРИНГ КРИТИЧЕСКИ ВАЖНЫХ И/ИЛИ
ПОТЕНЦИАЛЬНО ОПАСНЫХ ОБЪЕКТОВ, СМЕЩЕНИЙ
(ДЕФОРМАЦИЙ) И ЧАСТОТЫ КОЛЕБАНИЙ КРИТИЧЕСКИХ
ТОЧЕК КОНСТРУКЦИЙ ЗДАНИЙ, СООРУЖЕНИЙ
НА ОСНОВЕ КОСМИЧЕСКИХ СНИМКОВ**

В статье рассмотрены общие принципы построения 3D объектов, ГИС МОНИТОРИНГ, МЕТЕОР-1 КОММУНИКАЦИИ, ВЫСОТНЫЕ ЗДАНИЯ.

Основная цель данной статьи - решение проблемы мониторинга критически важных и/или потенциально опасных объектов, смещений (деформаций) и частоты колебаний критических точек конструкций зданий, сооружений, инженерных коммуникаций (объектов энергетики, гидротехнических сооружений, трубопроводов, мостов, высотных зданий и т.д.).

Основные решаемые задачи:

- сбор, обработка, хранение данных, поступающих от датчиков системы ГЛОНАСС/GPS и других измерительных устройств, их визуализация;
- определение предельно допустимых значений смещений критических точек конструкций;
- формирование отчетов о состоянии объектов мониторинга до должностных лиц, ответственных за их безопасную эксплуатацию, поддержку принятия решений должностными лицами;
- моделирование последствий природно-техногенных ситуаций, связанных с объектами мониторинга и окружающими их территориями;
- архивация данных;
- отображение на модели объекта датчиков состояния (движения, объема, пожарной тревоги, камер видеонаблюдения и т.д.) и их показаний;
- визуализация показаний любого выбранного датчика;
- свободное перемещение в трехмерном пространстве, просмотр территории и объекта в трехмерном виде;
- регистрация и просмотр информации об объектах в трехмерном виде.

Состав:

- специальное программное обеспечение «РЕКОД»;
- специальное программное обеспечение «РЕКОД-Модель»;

- трехмерная модель объекта мониторинга и окружающего пространства.



Рис. 1. Поэтажный план здания

Особенностью решения аналитических задач в комплексе является сбор, систематизация, аналитическая и математическая обработка пространственной картографической и графической информации.

Для решения аналитических задач используются:

- визуальное исследование (картографический и графический анализ);
- аналитические преобразования (применение заранее заданного оператора к имеющимся географическим данным для вычисления нового представления);
- правдоподобный вывод (нахождение оператора аналитического преобразования).

Картографические операции обеспечивают выявление образа пространственных и пространственно-временных свойств слоя, взаимных пространственно-временных свойств нескольких географических слоев и картографическое управление анализом данных.

Преимущества:

- наглядность отображения информации;
- простота и удобство использования;
- быстрое время реагирования диспетчеров на изменение ситуации;

- комплекс разработан в соответствии с требованиями, предъявляемыми к созданию, производству и эксплуатации космических комплексов.

Заключение

1. Повышение удобства использования систем безопасности и мониторинга важных объектов;
2. Сокращение времени реагирования на срабатывание датчиков систем безопасности до 1-2 секунд.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Еремеев В.В., Козлов Е.П. Повышение точности и надежности координатной привязки снимков от геостационарных космических систем наблюдения земли по электронным картам //Вестник РГРТУ. – Вып.28. 2009. – С. 6 – 12.
2. <http://www.ntsomz.ru/zakaz/data>
3. www.npkrekod.ru
4. Р.Гонсалес, Р. Вудс Цифровая обработка изображений Москва: техносфера, 2006г.
5. Цифровая обработка сигналов и ее применения: Материалы докладов 12 Международной конференции. – Москва, 2009. – Т. VI. – С. 256-259.

А.И. ТАГАНОВ, Д.В. ГИЛЬМАН

Рязанский государственный радиотехнический университет

АСПЕКТЫ НЕЧЕТКОЙ ИДЕНТИФИКАЦИИ И АНАЛИЗА ПРОЦЕССНЫХ РИСКОВ ПО СТАДИЯМ ЖИЗНЕННОГО ЦИКЛА ПРОЕКТА

Рассматриваются методы формализации задач идентификации и анализа процессных рисков по стадиям жизненного цикла проекта программных изделий в условиях нечеткости исходных данных.

В работе рассмотрены аспекты и возможные подходы к решению задач идентификации и анализа процессных рисков в условиях нечеткости по стадиям жизненного цикла (ЖЦ) программного проекта (ПП). Указанные задачи относятся к классу слабоструктурированных задач и требуют разработки соответствующих подходов и методов их решения с учетом факторов неопределенности и нечеткости [1-3].

Аспекты задачи идентификации процессных рисков

Идентификация процессных рисков ПП представляет собой спе-

циальный процесс, ориентированный на выявление потенциальных рисков процессов и документирование их характеристик [3]. Область определения задачи, связанной с идентификацией процессных рисков может быть представлена кортежем:

$$\langle Q, H, R, G, A, L \rangle, \quad (1)$$

где Q - заданное множество исходных данных (описание внешней среды проекта, описание внутренней среды проекта, описание отклонений в процессе реализации проекта, описание альтернатив реализации проекта и др.); H - заданная модель характеристик качества проектного процесса; $R = \{R_1, R_2, \dots, R_n\}$ - искомое множество процессных рисков; G - граф связности элементов множества R ; A - множество альтернатив реагирования на идентифицированное множество R ; L - набор методов и средств, ориентированных на определение множеств R , A и построение графа G .

Успешность решения задач с областью определения (1) во многом зависит от используемого набора методов, входящих в состав множества L . В инженерной практике такой набор методов обычно включает в себя: метод мозгового штурма, метод Дельфи, метод построения дерева решений, анализ SWOT, анализ контрольных списков, метод графического отображения моделей рисков с помощью диаграмм и др. [2, 3]. Однако указанные методы являются слабо формализованными, трудоемкими и соответственно субъективными по отношению к качеству принимаемых решений в условиях нечеткости проектных данных по стадиям ЖЦ программного проекта. Для решения задачи с областью определения (1) предлагается взять за основу известные подходы [2 - 4], которые ориентированы на идентификацию рисков качества продукта, а не качества процесса. Это позволяет на формализованной основе построить соответствующие адаптированные алгоритмы и далее инструментальные средства для поддержки процесса идентификации рисков качества процессов ПП.

Аспекты задачи анализа процессных рисков

Анализ процессных рисков ПП заключается в ранжировании по важности всех идентифицированных рисков анализируемого процесса с последующим выделением по заданным критериям некоторой совокупности рисков, которые передаются на следующие этапы планирования и мониторинга этих рисков.

В условиях нечеткости исходных данных, предлагаемый здесь подход к решению указанной задачи нечеткого анализа процессных рисков предполагает решение двух слабоструктурированных задач:

Задача 1 - определение вектора степеней влияния идентифици-

рованных процессных рисков на обобщенный (интегральный) критерий качества процесса;

Задача 2 - оптимизация состава контролируемых процессных рисков качества, передаваемых на следующие этапы, связанные с планированием и мониторингом рисков процессов проекта.

Область определения первой слабоструктурированной задачи, связанной с анализом рисков, может быть представлена следующим кортежем:

$$\langle R(t), H(t), f_s, P_s, G(t), W(R(t)), Z \rangle, \quad (2)$$

где $R(t)$ - множество идентифицированных процессных рисков проекта на момент времени t ; $H(t)$ - модель характеристик качества процесса ПП на текущем этапе проекта; f_s - отображение $R(t)$ в $H(t)$, определяющее оценки влияния последствий $R(t)$ на характеристики качества процесса проекта; P_s - структура предпочтений специалистов (экспертов) по процессным рискам; $G(t)$ - граф связности процессных рисков; $W(R(t))$ - искомый вектор степеней влияния последствий $R(t)$ на обобщенный критерий качества процесса ПП; $Z = \{Z1 \div Z5\}$ - набор предлагаемых алгоритмов для определения вектора $W(R(t))$ на основе представленных на определенном этапе проекта исходных данных.

Для решения задачи 1 с областью определения (2) предлагается исследовать следующие подходы и способы построения набора алгоритмов $Z = \{Z1 \div Z5\}$:

- алгоритм $Z1$, позволяющий определять (по стадиям ЖЦ проекта) вектор $W(R(t))$ на основе следующих исходных данных: модель $H(t)$ задана множеством независимых характеристик качества процесса ПП; $R(t)$ содержит множество независимых рисков; f_s и P_s представлены экспертным способом на основе количественных оценок;

- алгоритм $Z2$, отличающийся от $Z1$ тем, что экспертные оценки специалистов по проектным рискам заданы вербальным способом;

- алгоритм $Z3$, отличающийся от $Z1$ и $Z2$ тем, что модель $H(t)$ представлена деревом характеристик качества процесса ПП;

- алгоритм $Z4$, позволяющий на основе данных, полученных при использовании одного из алгоритмов $Z1, Z2, Z3$, учитывать наличие однозначной связности процессных рисков $R(t)$ по стадиям ЖЦ проекта;

- алгоритм Z_5 , отличающийся от Z_4 тем, что дополнительно позволяет учитывать субъективную связность процессных рисков $R(t)$ ПП.

В основу обоснованного построения рассматриваемых алгоритмов $Z = \{Z_1 \div Z_5\}$ предлагаются прикладные методы теории нечетких множеств (ТНМ), модифицированный метод анализа иерархий и многокритериальные модели принятия решений в условиях неопределенности [1, 2, 4].

Вторая задача, связанная с определением совокупности контролируемых процессных рисков $R(t)^* \subset R(t)$ по критерию их наибольшего суммарного влияния на обобщенный показатель качества процесса ПП, имеет следующую область определения:

$$\langle R(t), W(R(t)), P_r(t), B \rangle, \quad (3)$$

где $R(t)$ и $W(R(t))$ - соответственно заданное множество процессных рисков ПП и заданный вектор степеней влияния этих рисков на обобщенный критерий качества процесса ПП; $P_r(t)$ - структура предпочтений экспертов по ресурсам проекта, которая ставит в соответствие каждому риску из $R(t)$ требуемые ресурсы на управление этим риском; $B = \{B_1, B_2\}$ - набор алгоритмов, необходимый для решения задачи по определению $R(t)^*$ с учетом заданных условий и ограничений.

Для решения слабоструктурированной задачи 2 с областью определения (3) предлагаются два подхода, рассматривающие построение алгоритмов определения $R(t)^* \subset R(t)$:

- первый подход связан с построением алгоритма B_1 , ориентированного на решение задачи с областью определения (3) в том случае, когда в исходных данных присутствуют условия нечеткости процессных данных и отсутствуют жесткие ресурсные ограничения на управление процессными рисками ПП;

- второй подход отражает построение алгоритма B_2 , отличающегося от B_1 возможностью решения указанной задачи, при наличии в исходных данных заданных предпочтений экспертов по ресурсам, выделенных в проекте для управления процессными рисками.

Для построения алгоритмов B_1 и B_2 предлагается исследовать подходы, основывающиеся на использовании положений ТНМ и методов многокритериального выбора альтернатив в условиях неопределенности. Опыт применения указанных теорий для задач анализа рисков программных изделий позволяет адаптировать известные методы и

алгоритмы [2, 4] к задачам анализа рисков качества процессов ПП. Сформированный такими способами оптимизированный состав $R(t)^*$ используется далее на этапе мониторинга и сокращения процессных рисков качества процессов ПП.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Аверкин А.Н., Батыршин И.З., Блишун А.Ф., Силов В.Б., Тарасов В.Б. Нечеткие множества в моделях управления и искусственно-го интеллекта / под. ред. Д.А.Поспелова. - М.:Наука, 1986. – 312 с.

2. Гуров В.С., Корячко В.П., Таганов А.И, Таганов Р.А. Теория и практика снижения рисков проектов сложных программных систем по характеристикам качества // Труды VII Всероссийской научно-методической конференции «Телематика-2010». – С.-П., 2010. – Том 2. – С. 389-390.

3. Липаев В.В. Анализ и сокращение рисков проектов сложных программных средств. - М.: СИНТЕГ, 2005. – 224 с.

4. Таганов А.И., Таганов Р.А. Формальные методы поддержки процесса управления рисками качества проекта: Учебное пособие / Под ред. проф. Корячко В.П. Рязан. гос. радиотехн. акад. – Рязань:РГРТА, 2003. – 74 с.

А.И. ТАГАНОВ, Д.В. ГИЛЬМАН

Рязанский государственный радиотехнический университет

ФОРМАЛИЗАЦИЯ ЗАДАЧ МОНИТОРИНГА ПРОЦЕССНЫХ РИСКОВ ПО СТАДИЯМ ЖИЗНЕННОГО ЦИКЛА ПРОЕКТА

Рассматриваются способы формализации задач мониторинга и сокращения процессных рисков по стадиям жизненного цикла проекта программных изделий в условиях нечеткости исходных данных.

Функциональное назначение этапа мониторинга и сокращения процессных рисков программного проекта заключается в решении двух слабоструктурированных задач [2, 3]. Первая задача связана с определением по ходу программного проекта рисков ситуаций подобных тем, которые зафиксированы в массиве контролируемых процессных рисков $R(t)^*$. Решение второй задачи заключается в определении для каждого выявленного процессного риска адекватной альтернативы реагирования (смягчения риска).

Постановка задачи

Область определения первой задачи - идентификации рисков на этапе мониторинга может быть представлена кортежем [3]:

$$\langle S, w(t), f, h, P_s(t), D \rangle, \quad (1)$$

где $S = \{S_1, S_2, \dots, S_N\}$ - набор эталонных рисков ситуаций проекта, соответствующих $R(t)^*$, которые хранятся в библиотеке контролируемых рисков; $w(t)$ - текущая рисковая ситуация проекта; f - отображение $w(t)$ в S , характеризующее степень их соответствия; h - порог достоверности нечеткого соответствия эталонной и текущей ситуации проекта; $P_s(t)$ - структура предпочтений экспертов по рискам; D - набор алгоритмов, позволяющих определять соответствие текущей ситуации проекта $w(t)$ (с некоторым порогом достоверности нечеткого соответствия h) ситуациям из набора эталонных ситуаций S .

Для решения первой задачи с областью определения (1) могут быть адаптированы и использованы методы и алгоритмы, представленные в [3,4], которые основываются на использовании и построении расплывчатых ситуационных моделей принятия решений в условиях нечеткости.

Область решения второй задачи, связанной с определением оптимальной альтернативы реагирования на идентифицированное рисковое событие может, быть представлена кортежем:

$$\langle A(R_i), R_i, G, P_a, U \rangle, \quad (2)$$

где $A(R_i) = \{A_1(R_i), \dots, A_p(R_i)\}$ - множество альтернатив реагирования на идентифицированное рисковое события R_i ; G - отображение $A(R_i)$ на R_i ; P_a - структура предпочтений экспертов по рискам; $U = \{U1, U2, U3\}$ - набор требуемых алгоритмов для решения задачи определения наиболее предпочтительной (рациональной) альтернативы реагирования A^* на идентифицированное рисковое событие R_i согласно P_a .

Решение задачи

Для решения задачи с областью определения (2) в статье рассматриваются способы построения алгоритмов $U = \{U1, U2, U3\}$, основывающиеся на использовании положений теории нечетких множеств и решении задачи в условиях нечеткости исходных данных. Эти способы отличаются тем, что критерии важности альтернатив реагирова-

ния на проектные риски рассматриваются в терминах предпочтительности альтернатив на заданном множестве альтернатив $A(R_i) = A$. В этом случае для любой пары альтернатив реагирования $A_i, A_j \in A$ под значениями $\mu_{J_W}(A_i, A_j)$ понимается степень предпочтения « A_i не хуже A_j » в записи $A_i \geq A_j$. Тогда решение указанной задачи заключается в выборе наиболее предпочтительной альтернативы из множества A , на котором задано нечеткое отношение предпочтения J_W .

Первый способ (алгоритм U1) решения поставленной задачи, основывается на использовании опыта одного эксперта по рискам. Считается, что эксперт по рискам в терминах предпочтительности определяет матрицу предпочтительности альтернатив реагирования на идентифицированное рисковое событие R_i . Далее в соответствии с предлагаемым способом для нечеткого отношения предпочтения J_W строится нечеткое отношение строгого предпочтения J_W^S , определяемое функцией принадлежности:

$$\mu_{J_W^S} = \begin{cases} \mu_{J_W}(A_i, A_j) - \mu_{J_W}(A_j, A_i), \\ \text{если } \mu_{J_W}(A_i, A_j) > \mu_{J_W}(A_j, A_i); \\ 0, \text{ если } \mu_{J_W}(A_i, A_j) \leq \mu_{J_W}(A_j, A_i). \end{cases} \quad (3)$$

После этого вычисляется нечеткое подмножество недоминируемых альтернатив реагирования $A_R^{nd} \subset A$, ассоциированное с J_W и включающее те альтернативы, которые не доминируют ни над какими другими альтернативами реагирования, определяемыми функцией принадлежности:

$$\mu_{J_W}^{nd}(A_i) = 1 - \max_{A_j \in A} \mu_{J_W^S}(A_j, A_i). \quad (4)$$

Для любой альтернативы реагирования $A_j \in A$ значение $\mu_{J_W}^{nd}(A_j)$ понимается как степень недоминируемости этой альтернативы, то есть степень, с которой A_j не доминирует ни над одной из альтернатив реагирования множества A ; $\mu_{J_W}^{nd}(A_i) = \alpha$ означает, что никакая альтернатива A_j не может быть лучше A_i со степенью доминирования α ; иначе говоря, может доминировать над другими альтернативами, но со степенью не выше $1 - \alpha$. Рациональным будем счи-

тать выбор альтернатив, имеющих по возможности большую степень принадлежности множеству $A_{J_W}^{nd}$. И, следовательно, за оптимальную примем альтернативу A^* , для которой значение $\mu_{J_W}^{nd}(A^*)$ максимально:

$$A^* = \arg \max_{A_j \in A} \mu_{J_W}^{nd}(A_j).$$

Если наибольшую степень недоминируемости имеет не одна, а несколько альтернатив, то менеджер по рискам может выбрать одну из них, исходя из каких-либо иных соображений, либо расширить круг экспертов для формирования исходных данных задачи и решить поставленную задачу на основе второго и третьего способов, которые рассмотрены далее по тексту.

Второй способ (алгоритм $U2$) учитывает возможности привлечения группы менеджеров по рискам для задания предпочтительности альтернатив реагирования на идентифицированное рисковое событие и позволяет получить более объективное решение задачи с областью определения (2). В этом варианте на множестве всевозможных альтернатив реагирования $A = \{A_1, A_2, \dots, A_{N_A}\}$ каждый эксперт по рискам задает свое видение важности альтернатив реагирования в терминах предпочтительности J_{W_k} . Главный менеджер (руководитель проекта) по-разному может относиться к мнению каждого эксперта по рискам, что может находить отражение в весовых коэффициентах λ_k , (где $0 \leq \lambda_k \leq 1$, $\sum \lambda_k = 1$), соответствующих каждому из специалистов.

В этих условиях для определения оптимальной альтернативы реагирования на идентифицированное рисковое событие, в соответствии с предлагаемым вторым способом (алгоритмом $U2$), необходимо для каждого отношения предпочтения J_{W_k} построить нечеткое отношение строгого предпочтения $J_{W_k}^S$, функция принадлежности которого определяется по формуле (3). Далее построить свертку P отношений предпочтения, как пересечение нечетких отношений строгого предпочтения экспертов, по формуле:

$$P = \bigcap J_{W_k}^S(A_i, A_j) = \min\{\mu(A_i, A_j)\}. \quad (5)$$

Таким образом, получается новое нечеткое отношение нестрогого предпочтения. Далее на основе P можно построить отношение строгого предпочтения $P^S = P / P^T$ с функцией принадлежности μ_P^S :

$$\mu(J_W^S, A_i, A_j) = \begin{cases} \mu(J_W^S, A_i, A_j) - \mu(J_W^S, A_j, A_i), \\ \text{если } \mu(J_W^S, A_i, A_j) > \mu(J_W^S, A_j, A_i) \\ 0, \text{ если } \mu(J_W^S, A_i, A_j) \leq \mu(J_W^S, A_j, A_i) \end{cases}.$$

Следующим шагом следует определить множество недоминируемых альтернатив $U(P^S, nd)$ с функцией принадлежности:

$$\mu^S(nd, A_i) = 1 - \max_j \{\mu_P^S(nd, A_i)\}. \quad (6)$$

На следующем шаге необходимо построить выпуклую свертку Q отношений J_{W_k} , которая определяется как $Q = \sum \lambda_k J_{W_k}$ с функцией принадлежности:

$$\mu_Q(A_i, A_j) = \sum \lambda_k \mu(A_i, A_j). \quad (7)$$

Для нечеткого отношения предпочтения Q аналогично P^S строится отношение строгого предпочтения Q^S и множество недоминируемых альтернатив $U(Q^S, nd)$. Из формул (5)-(7) можно сделать вывод, что множества $U(R^S, nd)$ и $U(Q^S, nd)$ несут дополняющую друг друга информацию о недоминируемости альтернатив реагирования.

Далее рассматривается пересечение полученных множеств $U(R^S, nd)$ и $U(Q^S, nd)$ с функцией принадлежности:

$$\mu^{nd}(A_i) = \min(\mu_P^{nd}(A_i), \mu_Q^{nd}(A_i)).$$

Рациональным считается выбор той альтернативы A^* из A , для которой значение $\mu^{nd}(A^*)$ максимально:

$$A^* = \arg \max \mu^{nd}(A_i), A_i \in A.$$

Третий способ (алгоритм $U3$) решения задачи в постановке (2) характеризуется тем, что значимость мнений экспертов по рискам оценивается при помощи нечеткого отношения предпочтения J_N , заданного на множестве экспертов E с функциями принадлежности $\mu_{J_N}(e_k, e_l)$, $e_k, e_l \in E$, значения которых обозначают степень предпочтения мнения эксперта e_k по сравнению с мнением эксперта e_l . В данном случае для решения поставленной задачи необходимо аналогично второму способу для каждого J_{W_k} построить $J_{W_k}^S$ и A_k^{nd} .

Далее для ясности введем обозначение $\mu_k^{nd}(A_i) = \mu_\Phi(A_i, e_k)$ и тем самым зададим нечеткое соответствие Φ между множествами A и E .

На следующем шаге построим свертку Γ в виде композиции соответствий $\Gamma = \Phi^T J_N \Phi$, причем результирующее отношение Γ определяется как произведение Φ^T , J_N , Φ . То есть, получается единое результирующее отношение, определенное с учетом информации об относительной важности нечетких отношений предпочтения J_{W_k} . С отношением Γ ассоциируются отношение Γ^S и множество A_r^{nd} . Корректируется множество A_r^{nd} до множества $A_r''^{nd}$ с функцией принадлежности:

$$\mu_{\Gamma^S}^{nd}(A_i) = \min(\mu_{\Gamma}^{nd}(A_i), \mu_{\Gamma}(A_i, A_i)).$$

Выбирается та альтернатива, для которой значение функции принадлежности, скорректированного нечеткого множества $A_r''^{nd}$ недоминируемых альтернатив максимально. Достоинством рассмотренных способов (алгоритмов $U = \{U1, U2, U3\}$) решения задачи с областью определения (5) в условиях нечеткости процессных данных является простота их программной реализации.

Заключение

Рассмотренные в работе задачи и методы нечеткой идентификации и нечеткого мониторинга процессных рисков качества процессов проекта программных изделий, расширяют области эффективного применения формальных методов для решения основных задач управления процессными рисками проектов программных изделий в условиях нечеткости исходных данных. При этом достаточно простая программная реализация предложенных методов позволяет эффективное использование созданного специального инструментария в составе интегрированных CASE- технологий.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. ANSI/PMI 99-001-2004. Руководство к Своду знаний по управлению проектами. Четвертое издание (Руководство РМВОК).
2. Липаев В.В. Анализ и сокращение рисков проектов сложных программных средств. – М.: СИНТЕГ, 2005. – 224 с.

3. Мелихов А.Н., Бернштейн Л.С., Коровин С.Я. Расплывчатые ситуационные модели принятия решений. – Таганрог: ТРТИ, 1986. – 92 с.

4. Таганов А.И. Методика анализа и сокращения рисков проектов сложных программных систем по характеристикам качества // Вестник РГРТУ. – Рязань, 2010. – Вып. 30. – С. 77-82.

5. Таганов А.И. Применение нечетких множеств для формализации процессов анализа и идентификации важности рисков программного проекта // Системы управления и информационные технологии. – Воронеж: Научная книга, 2007. – № 4(30). – С. 46-51.

6. Фатрелл Р.Т., Шафер Д.Ф., Шафер Л.И. Управление программными проектами: достижение оптимального качества при минимальных затратах: пер. с англ. – М.: Вильямс, 2003. – 1136 с.

А.И. ТАГАНОВ, Д.В. ГИЛЬМАН

Рязанский государственный радиотехнический университет

ЗАДАЧИ И АЛГОРИТМЫ СТРУКТУРНОЙ ОПТИМИЗАЦИИ ПРОЦЕССОВ ПРОЕКТА ПО КРИТЕРИЯМ РИСКА

Рассматриваются постановки задач структурной оптимизации процессов проекта по критериям риска и подходы к построению алгоритмов их решения.

Для структурной оптимизации процессов проекта по критериям риска в работе рассматривается подход, основанный на генерации различных вариантов выполнения каждого заданного процесса с последующим отбором оптимальных вариантов по критериям минимально риска и времени исполнения проекта. В качестве научно-методической базы для постановки соответствующих задач и построения требуемых алгоритмов их решения предлагается использовать подходы и методы оптимизации алгоритмических процессов по критериям надежности в условиях нечеткости [1, 2, 3].

Постановка задач оптимизации

В условиях нечеткости проектных данных задачи оптимизации процессных рисков связаны с оптимальным выбором способов реализации операторов и логических условий, входящих в состав заданных процессов проекта, что приводит к необходимости сравнения нечетких чисел и соответственно определения соответствующих правил [3].

Правило 1. Если заданы нечеткие числа $\tilde{a} = \bigcup_{\alpha \in [0,1]} (\underline{a}_\alpha, \bar{a}_\alpha)$ и $\tilde{b} = \bigcup_{\alpha \in [0,1]} (\underline{b}_\alpha, \bar{b}_\alpha)$, то условия равенства и неравенства этих чисел определим так:

$\tilde{a} = \tilde{b}$, если $\forall \alpha \in [0,1]: \underline{a}_\alpha = \underline{b}_\alpha$ и $\bar{a}_\alpha = \bar{b}_\alpha$;

$\tilde{a} < \tilde{b}$, если $\begin{cases} \forall \alpha \in [0,1]: \underline{a}_\alpha \leq \underline{b}_\alpha & \text{и} & \bar{a}_\alpha \leq \bar{b}_\alpha \\ \exists \alpha \in [0,1]: \underline{a}_\alpha < \underline{b}_\alpha & \text{или} & \bar{a}_\alpha < \bar{b}_\alpha \end{cases}$.

Правило 2. Если нечеткое число $\tilde{a} = \bigcup_{\alpha \in [0,1]} (\underline{a}_\alpha, \bar{a}_\alpha)$ задано на k_a α -уровнях, а нечеткое число $\tilde{b} = \bigcup_{\alpha \in [0,1]} (\underline{b}_\alpha, \bar{b}_\alpha)$ на k_b α -уровнях, то:

$\tilde{a} < \tilde{b}$, если $\frac{1}{2k_a} \sum_{\alpha \in [0,1]} (\underline{a}_\alpha + \bar{a}_\alpha) < \frac{1}{2k_b} \sum_{\alpha \in [0,1]} (\underline{b}_\alpha + \bar{b}_\alpha)$;

$\tilde{a} = \tilde{b}$, если $\frac{1}{2k_a} \sum_{\alpha \in [0,1]} (\underline{a}_\alpha + \bar{a}_\alpha) = \frac{1}{2k_b} \sum_{\alpha \in [0,1]} (\underline{b}_\alpha + \bar{b}_\alpha)$.

Оптимальный выбор операторов и условий

Далее постановку задач оптимального выбора способов реализации операторов и логических условий, заданного процесса проекта по критериям времени и риска правильного выполнения, будем рассматривать из предположения, что исходные данные представлены в виде нечетких чисел.

Пусть известен исходный вариант структуры процесса проекта, в котором каждый i -ый оператор (A_i) может быть реализован n_i различными способами, а каждое j -ое логическое условие (ω_j), – n_j способами.

$$A_i = \{A_{i_1}, A_{i_2}, \dots, A_{i_{n_i}}\};$$

$$\omega_j = \{\omega_{j_1}, \omega_{j_2}, \dots, \omega_{j_{n_j}}\}.$$

Наличие различных способов выполнения операторов и логических условий можно рассматривать в общем случае как следствие имеющейся у проектировщика возможности управления структурой и параметрами процесса.

Предполагается, что параметры риска операторов и условий заданы нечеткими числами.

Пусть $X = \{x_1, x_2, \dots, x_n\}$ – вектор управляемых переменных, которыми являются номера способов реализации операторов и логических условий;

$\tilde{P}(X)$ и $\tilde{T}(X)$ – нечеткие вероятность риска (правильного выполнения) и время выполнения процесса, в случае, когда управляемые переменные заданы вектором X ;

$\tilde{P}_{ДОП}$ – наименьшее допустимое значение величины $\tilde{P}(X)$;

$\tilde{T}_{ДОП}$ – наибольшее допустимое значение величины $\tilde{T}(X)$.

Задача оптимизации процесса проекта состоит в выборе такого вектора X , чтобы:

(а) $\tilde{T}(X) \rightarrow \min$ и $\tilde{P}(X) \geq \tilde{P}_{ДОП}$ – прямая постановка задачи;

(б) $\tilde{P}(X) \rightarrow \max$ и $\tilde{T}(X) \leq \tilde{T}_{ДОП}$ – обратная постановка задачи.

Алгоритм структурной оптимизации состоит в отсечении перспективных вариантов реализации операторов и логических условий, получаемых каждым шагом укрупнения структуры процесса. При этом будем использовать свойства алгоритмических процессов [2, 3], интерпретируемые в терминологии процессных рисков.

Свойство 1. Последовательное применение правил укрупнения структуры процесса проекта за конечное число шагов преобразует исходный процесс к эквивалентному оператору.

Свойство 2. Вероятность правильного выполнения заданного процесса проекта возрастает по мере снижения рисков элементов процесса, т.е. возрастания вероятностей правильного выполнения этих элементов.

Свойство 3. Время выполнения рассматриваемого процесса проекта возрастает по мере возрастания времени выполнения элементов процесса.

Пусть \tilde{p}_A^1 и \tilde{t}_A – нечеткие вероятности правильного выполнения оператора A и время его выполнения;

$\tilde{k}_\omega^{11}(\tilde{k}_\omega^{00})$ и \tilde{t}_ω – нечеткие вероятности отсутствия ошибок первого и второго рода при выполнении условия ω и время его выполнения.

Предложение 1. Если A_1 и A_2 – два варианта реализации оператора A , причем $\tilde{p}_{A_1}^1 > \tilde{p}_{A_2}^1$ и $\tilde{t}_{A_1} < \tilde{t}_{A_2}$, то вариант A_2 не может входить в оптимальный АППП.

Предложение 2. Если ω_1 и ω_2 – два варианта выполнения логического условия ω , причем $\tilde{k}_{\omega_1}^{11} > \tilde{k}_{\omega_2}^{11}$, $\tilde{k}_{\omega_1}^{00} > \tilde{k}_{\omega_2}^{00}$ и $\tilde{t}_{\omega_1} < \tilde{t}_{\omega_2}$, то вариант ω_2 не может входить в оптимальный вариант структурного построения процесса проекта.

Применение введенных предложений 1 и 2 позволяет сформулировать алгоритмы структурной оптимизации процессов проекта по критериям риска.

Алгоритм структурной оптимизации в прямой постановке

Шаг 1. Установить счетчик процессных структур в единицу, $m:=4$.

Шаг 2. Найти процессную структуру с номером m .

Шаг 3. Отбросить неперспективные варианты выполнения операторов, входящих в m -ю структуру, пользуясь предложением 1

Шаг 4. Отбросить неперспективные варианты выполнения логических условий, входящих в m -ю структуру, пользуясь предложением 2

Шаг 5. Вычислить характеристики всех вариантов выполнения m -ой структуры, пользуясь формулами нечетких моделей надежности алгоритмических структур [3], интерпретированных в терминах риска.

Шаг 6. Отбросить неперспективные варианты выполнения m -ой структуры, пользуясь предложением 1.

Шаг 7. Проверить условие: "исходный процесс укрупнен до эквивалентного оператора?". Если "да", то перейти к шагу 8, иначе установить $m:=m+1$ и перейти к шагу 2.

Шаг 8. Отбросить варианты реализации процесса, для которых $\tilde{P}(X) < \tilde{P}_{\text{Доп}}$. Из оставшихся выбрать тот, у которого $\tilde{T}(X) \rightarrow \min$. Развернуть выбранный вариант до уровня операторов и логических условий и записать оптимальный вариант искомого процесса проекта.

Алгоритм структурной оптимизации в обратной постановке

Для построения этого алгоритма введем понятие неальтернативных элементов процесса проекта.

Определение 3. Неальтернативными элементами процесса проекта назовем, такие операторы и условия, которые при реализации процесса выполняются с вероятностью единица.

Алгоритм структурной оптимизации состоит в следующем.

Шаг 1. Установить счетчик процессных структур в единицу, $m:=1$.

Шаг 2. Найти структуру с номером m .

Шаг 3. Отбросить неперспективные варианты выполнения операторов, входящих в m -ю структуру, пользуясь предложением 1. Из оставшихся неальтернативных операторов отбросить те, которые превышают ограничение по времени.

Шаг 4. Отбросить неперспективные варианты выполнения логических условий, входящих в m -ю структуру, пользуясь предложением 2. Из оставшихся неальтернативных логических условий отбросить те, которые превышают ограничение по времени.

Шаг 5. Вычислить характеристики всех вариантов выполнения m -й структуры, пользуясь формулами нечетких моделей надежности алгоритмических структур [3], интерпретированных в терминах риска.

Шаг 6. Отбросить неперспективные варианты выполнения m -й структуры, пользуясь предложением 1.

Шаг 7. Проверить условие: "исходный вариант процесса проекта укрупнен до эквивалентного оператора?" Если "да", то перейти к шагу 8, иначе установить $m := m + 1$ и перейти к шагу 2.

Шаг 8. Отбросить варианты процесса, для которых $\tilde{T}(X) > \tilde{T}_{доп}$. Из оставшихся выбрать вариант, у которого $\tilde{P}(X) \rightarrow \max$. Развернуть выбранный вариант структуры до уровня операторов и логических условий и записать оптимальный структуры процесса проекта по критериям риска.

Выводы

Рассмотренные в работе постановки задач структурной оптимизации заданных процессов проекта по критериям риска, а также подходы к построению алгоритмов их решения с использованием положений теории надежностной оптимизации в условиях нечеткости позволяют реализовать на практике требуемый инструментарий для поддержки процессов сокращения рисков проекта программных изделий.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Гуров В.С., Корячко В.П., Таганов А.И., Таганов Р.А. Теория и практика снижения рисков проектов сложных программных систем по характеристикам качества // Труды VII Всероссийской научно-методической конференции «Телематика-2010». – С.-П., 2010. – Том 2. – С. 389-390.

2. Гильман Д.В. Использование нечеткого вероятностного графа для оценки показателя надежности проекта // В сб. трудов Всероссийской научно-технической конференции «Новые информационные технологии в научных исследованиях и в образовании». – Рязань: РГРТУ, 2010.

3. Ротштейн А.П., Штовба С.Д. Нечеткая надежность алгоритмических процессов. – Винница: Континент, 1997. – 142 с.

А.В. ТОВПЕКО

Фиалиал ФГУП «ГНПРКЦ «ЦСКБ-Прогресс»-«ОКБ «Спектр»

ОСОБЕННОСТИ УПРАВЛЕНИЯ ПРОЕКТОМ СИСТЕМЫ АНАЛИЗА ТЕЛЕИЗМЕРЕНИЙ

Предметом настоящей статьи являются вопросы оптимального планирования и координации работ по разработке системы анализа измерительной информации (ИИ) при мониторинге состояния сложных технических комплексов. Представленные здесь положения могут распространяться на различные сферы техники, где используются телеизмерения, но изложенные результаты были получены и апробированы при работах преимущественно в сфере космической телеметрии. Системы телеизмерений являются специфическим классом инженерных задач, и, рассматривая их характерные свойства, можно сформулировать специализированные правила управления проектом, востребованные на всех стадиях жизненного цикла от информационного обследования до опытной эксплуатации системы.

Если проанализировать существующие подходы по тематике управления проектом, то большинство теорий представляют управленческие решения как компромисс между требованиями заказчика и интересами разработчика. Традиционно указывается, что требования и условия во многом противоречивы, что неизбежно приводит к проблеме выбора в треугольнике «цена – время – качество» и к зависящим от него дилеммам – универсальность или эффективность, быстроедействие или надёжность, дружелюбность к пользователю или простота программирования. Универсальные правила инженерии программного обеспечения обусловлены объективными причинами экономического и технического характера. Они действительно распространяются на все программные проекты, в том числе на комплексы обработки телеизмерений. Однако, выделив общие черты и проблемы в задачах сбора и анализа измерительной информации, возможно дополнить общие подходы новыми правилами, характерными для специфики нашей предметной области. Предварительно необходимо сформулировать пять ключевых свойств, накладывающие отпечаток на проблематику управления проектом:

1) **наличие многошаговой технологической последовательности** – жёсткое разделение специальных задач извлечения параметров из первичных входных форматов и достаточно универсальных инструментов вторичной обработки. Регистрация, сбор, обработка и анализ телеизмерений представляют собой многоуровневый технологический процесс с независимыми друг от друга стадиями вычислительного процесса. Существует возможность создания унифицированных форматов для того или иного этапа преобразования, что обеспечивает инвариантность модулей последующих шагов обработки от предшествующих. Такие функции, как декоммутация и компоновка кодов телеметрии или тарирование датчиков выносятся в отдельные составные части, чтобы поддержка тех или иных телеметрических систем или преобразователей была сосредоточена в отдельной конструктивной единице системы анализа, которую можно разрабатывать, сопровождать и автономно тестировать независимо от других. Применительно к управлению проектом это свойство означает возможность сократить сроки разработки за счёт параллельного программирования для нескольких подсистем большим количеством исполнителей при условии согласования интерфейсов. Данное свойство часто имеет место и в других сферах с тем отличием, что разделение вычислительного процесса на независимые шаги обычно не лежит на поверхности – его приходится либо выявлять, либо вносить искусственно, в то время как в системах анализа телеизмерений декомпозиция составных частей очевидна;

2) **неполная исходная постановка задачи на предобработку, визуализацию и документирование конечных результатов.** Если система анализа ИИ создаётся одновременно с объектом контроля, всегда существует риск информационной и программной несовместимости. Данный вопрос имеет два аспекта – с одной стороны несовместимость может быть вызвана форматом, информативностью или способом передачи данных, отличными от ожидаемых. С другой стороны, созданной графической среды может быть недостаточно для представления или вторичной математической обработки какого-либо параметра. Таким образом, адаптивность программной системы и простота её функционального наращивания выходят на первый план. Если в системах другого назначения можно пожертвовать оптимальной модульной структурой, обеспечивающей гибкую техническую поддержку, то в системах анализа ИИ это будет явным недостатком, который проявит себя по мере развития объекта контроля и повлечёт дополнительные затраты. Данные издержки могут быть изначально предотвращены, если архитектура системы разработана корректно, и ввод новой функ-

ции достигается внесением в состав нового модуля без модификации ранее созданного задела. Согласование входных и выходных протоколов до разработки кажется очевидным обязательным этапом, но риск несовместимости не исчезает. Причины следующие:

- на начало разработки заказчик может не обладать всей полнотой данных по объекту контроля или по внешним системам-потребителям результатов анализа;

- в процессе разработки выясняется, что соблюдение условий обмена с системой анализа приводит к более дорогим последствиям, чем отступление от них. Например, потребовалось многократно увеличить количество телеметрических параметров или перейти от высокоскоростной ВОЛС на неустойчивую радиосвязь;

- существуют объективные причины, по которой некоторые информационные сечения и способы приёма-передачи не могут быть утверждены заблаговременно – независимо от того, разрабатывается ли система кооперацией нескольких предприятий, либо группой из двух программистов. Фактические системные ограничения быть завуалированы и проявляться только в процессе разработки. Например, в таких задачах, как репортаж телеизмерений (представление в реальном времени) или оперативная оценка качества потребуется ряд экспериментов, прежде чем будет реализовано полноценное решение, отвечающее всем требованиям;

- если тестирование осуществлялось на математической модели в отсутствие объекта контроля на момент сдачи системы – риск несовместимости обусловлен риском некорректности программного имитатора и неполноты тестов.

3) **принципиальные отличия критериев для функций оперативной и апостериорной обработки ИИ.** Приоритет эффективности перед полнотой данных в реальном времени неприемлем для анализа после сеанса, где надёжность и целостность выводимых параметров предпочтительнее быстроедействия. Несмотря на то, что пользователю могут быть удобны единые инструменты обработки, данная особенность накладывает требования на организацию вычислительного процесса. На практике задачи оперативного экспресс-анализа и полной отложенной обработки часто реализуются двумя различными подсистемами;

4) **наличие в средствах визуализации дизайнерской части, независимой от алгоритмизации и программирования.** Помимо тривиальных диаграмм и таблиц конечным результатом анализа являются графические модели технического состояния. При современном подходе к машинной графике – это анимированные трёхмерные моде-

ли и мнемосхемы объекта. Векторная графика и текстуры должны быть основаны на общепринятых форматах, создаваемых в типовых редакторах. Например, разработка интерпретатора формата COLLADA и рисование моделей в стандартной САПР (3DS Max, Maya, Blender) более универсально и менее трудоёмко, чем ручное программирование объекта по отдельным графическим примитивам. Из наличия «художественной» составляющей в процессе разработки следует два вывода – о возможности распараллеливания работы дизайнера и программиста, а также о стремлении к универсальности программы отображения, когда смена или дополнение объекта контроля заключаться в правке технологического задания и смене графической модели с сохранением неизменного унифицированного «вычислительного ядра»;

5) пользователь стремится создавать собственные алгоритмы анализа без прямого участия разработчика системы, что требует наличия языка предметной области в программах. По своему содержанию анализ ИИ является творческой задачей. В идеале, эксперт предметной области при выполнении своих обязанностей не должен быть ограничен фиксированным набором инструментов. Пользователю-анализатору должен быть предоставлен инструментарий для гибкого моделирования, преобразования и отображения – некий язык предметной области, предоставляющий свободу выбора операций над массивами первичных измерений вида «время – значение – достоверность». Это может быть интеграция среды графического (LabVIEW) или матричного (MATLAB) программирования, либо специальная разработка – например, адаптация языка VBA к среде графического просмотра. Следует отличать пятое свойство – наличие языка предметной области – от второго – обязательности функционального наращивания, так как в данном случае речь о программировании силами пользователя без участия программиста-профессионала, что обязывает учитывать требования простоты. По возможности, язык-интерпретатор для пользователя должен быть не императивным, а функциональным (возможно, с поддержкой объектно-ориентированного подхода), не быть строго типизированным и иметь простой интуитивно-понятный синтаксис, позволяющий не только быстро разрабатывать сценарии, но и делающий их доступным для чтения и понимания другими специалистами (непрограммистами). В частности, таким критериям отвечает интерпретатор Lua, имеющий простой синтаксис и массу синтаксических приёмов, сокращающих тексты сценариев на нём. Императивный язык C++ с низкоуровневыми конструкциями, адресной арифметикой и необходимостью следить за распределением памяти, очевидно, не может служить лингвистическим обеспечением для среды графического

анализа ИИ – операция, в которой на языке С++ потребуются десятки строк кода, на Lua может быть выражена несколькими операторами. Внедрение интерпретаторов также открывает интересные возможности для разработчика системы – единожды реализовав полноценную среду интерпретации, функциональное наращивание инструментов анализа может осуществляться созданием библиотеки сценариев, что менее трудоёмко динамически вызываемых библиотек функций при «подлинном» программировании загрузочных модулей.

Таковы пять технических особенностей систем анализа ИИ, которые в процессе управления проектом накладываются на пять типовых предпочтений разработчика, которые независимы от предмета разработки и практически всегда актуальны:

1) **минимизация труда при сохранении качественного уровня продукции в заданные сроки.** По сути это стремление извлечь наибольшую прибыль, уменьшая себестоимость. Данный критерий – экономический – должен иметь наибольший приоритет;

2) расширение предыдущего пункта – **обеспечение простоты технического сопровождения.** Это сводится к применению таких подходов к архитектуре системы, при которых добавление новых функций, аналогичных реализованным ранее, не приводило к существенным затратам в доработке и развёртывании очередной версии. Качественное программное обеспечение должно быть масштабируемым – как по функциональным характеристикам, так и по пропускной способности и количеству входов/выходов;

3) **применение универсальных подходов с расчётом на перспективу** – это обеспечение возможности заимствования данной разработки в других проектах. Универсальное решение не всегда является простейшим в исполнении, и системы анализа ИИ не исключение – второй пользовательский критерий отчасти конфликтует с первым, если не принимать во внимание возможный экономический эффект от внедрения практичного решения в будущем;

4) **анализ программного проекта в контексте прочих заказов** (по возможности – всех). Если рассматривать предприятие в целом, то в интересах организации найти такой подход к разработке, который обеспечит должный уровень унификации и заимствования в среднем, а не в конкретном проекте. Это может быть использование готовых решений из других работ – не всегда безупречных с точки зрения объективных критериев качества, но обеспечивающих ту или иную совместимость с другими системами. И напротив, это может быть трудоёмкая разработка идеальной рабочей среды с планом по будущему внедрению на других объектах;

5) **начинать разработку при наличии достаточных исходных данных и требований назначения.** Закономерное требование исполнителя, легко реализуемое для сбора и первичной обработки ИИ, но не всегда доступное применительно к анализу и представлению конечных результатов. Универсальность, функциональная завершенность среды графического анализа и документирования в предметной области анализа телеизмерений приобретают очень важный смысл. Жизнеспособность системы в течение длительного периода эксплуатации должна быть основана на возможности пользователя реализовать собственную методику без перекомпиляции специального программного обеспечения.

Архитектура системы и распределение задач исполнителей должны осуществляться с учётом двух возможных ситуаций, продиктованных исходными контрактными условиями. Первый вариант – когда необходимо провести разработку в кратчайшие сроки с максимальным использованием труда как можно большего количества сотрудников. Тогда изначально целесообразно разделить задачи на несколько независимых ветвей, которые могут создаваться и отлаживаться независимо друг от друга. Доработка по результатам общего (комплексного) тестирования не может осуществляться абсолютно независимо и появляется некоторый минимум по срокам, который не изменяется при увеличении рабочей группы – действует правило Брукса, что привлечение новых исполнителей на поздней стадии разработки не ускоряет, а усугубляет проект. Однако на начальных стадиях корректная декомпозиция задач большой системы анализа обеспечивает практически прямую зависимость сроков от количества программистов. Второй вариант – когда количество исполнителей ограничено двумя-тремя специалистами и на первый план выходит критерий эффективности разработки с возможными уступками в полноте и качестве функционала. Здесь может быть востребован противоположный подход, направленный не на разделение составных частей, а на объединение функций в единых конструктивных единицах системы с целью максимально сократить объёмы исходного кода и содержание тестов. Однако, такое объединение тоже должно планироваться исходя из специфики системы анализа, то есть решения, угрожающие необходимому и достаточному уровню возможности наращивания и унификации модулей, должны пресекаться.

Чтобы перейти к прикладным вопросам архитектуры системы и планирования разработки, рассмотрим технологический процесс обработки и анализа ИИ в самом общем виде. Обработка телеизмерений является многоуровневым процессом приёма, преобразования и анали-

за циклического потока данных, на каждом шаге которого в строгой последовательности выполняются операции извлечения кадров, подкадров и кодов, которые в конечном итоге приводятся к параметрам, подлежащим математической обработке.

Если рассмотреть технологическую цепь от поступления первичной информации до вывода результата, данные трансформируются в следующие формы – исходный дискретный поток, блок данных телеметрической системы, кадр устройства коммутации датчиков, код измерения, значение контролируемого процесса в физических единицах и, на заключительном шаге, массивы точек на графиках. Отличительным свойством следует считать возможность разбить составной процесс на стадии, имеющие зависимые друг от друга информационные сечения.

Циркулирующие в комплексе-обработчике потоки образуют граф, вершины которого представляют модули преобразования, а связи между ними – измерительные потоки и технологические задания, прошедшие тот или иной уровень обработки. В самом общем виде он представлен в форме потоковой модели (рис. 1). Анализ измерений сосредоточен в комплексе графического представления с подключённой средой интерпретатора. Рис. 2 показывает зависимость четырёх ключевых составных частей системы от внешних условий и друг от друга. Сплошные линии показывают жёсткую принципиальную зависимость, пунктирные – косвенную, не влияющую на обменные информационные сечения, но определяющие специфику некоторых технических деталей.

Таким образом, возможны два типовых плана разработки (сетевые план-графики представлены на рис. 3 и 4). Представлена независимость дизайнерской разработки, средств диагностирования и основных функциональных программ. Внутри стадии программирования имеются отдельные ветви по обработке и визуализации. Жирным выделены критические маршруты. В первом случае плана разрабатывается сложная графическая среда с интерпретатором, во втором варианте предполагается итеративная доработка группы модулей по «спиральной модели». Если первый является предпочтительным с точки зрения долгосрочной перспективы и полноты функциональных характеристик, то второй ориентирован на сжатые сроки и простоту реализации. С точки зрения привлечения наибольшего количества исполнителей для одновременной работы оба способа равноправны.

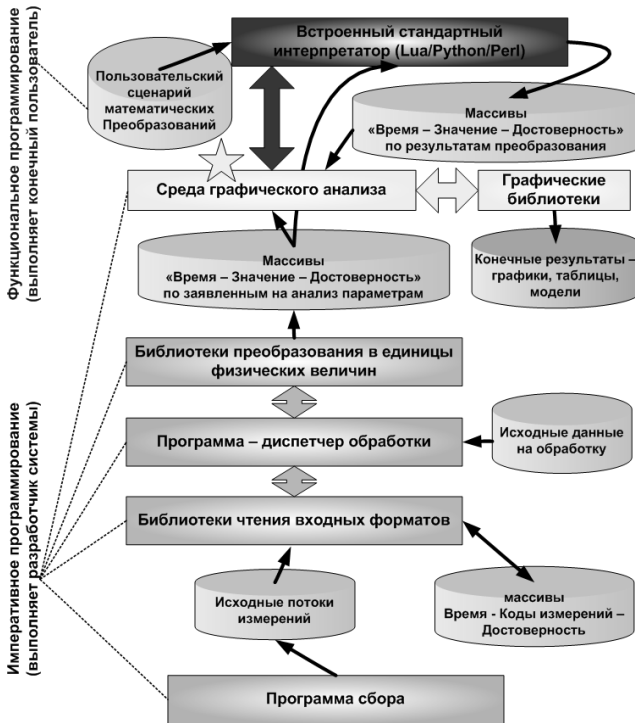


Рис. 1. Общий вид потоковой модели системы анализа ИИ



Рис. 2. Зависимость составных частей системы

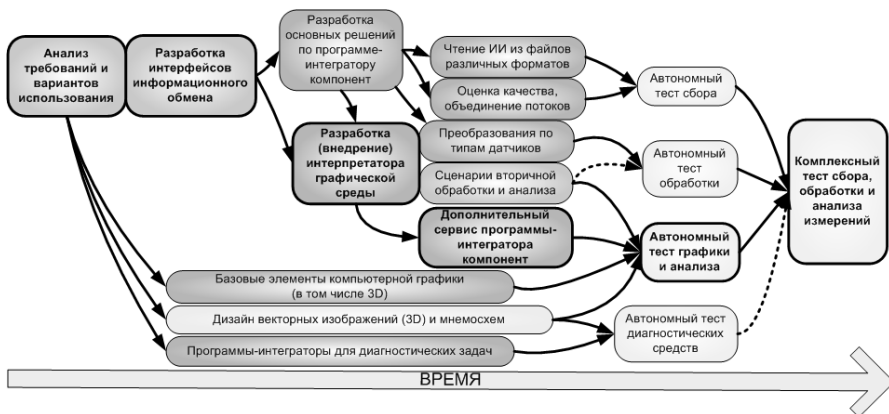


Рис. 3. Сетевой план-график разработки системы с внедрением языка интерпретатора (вторичная обработка реализуется сценариями)

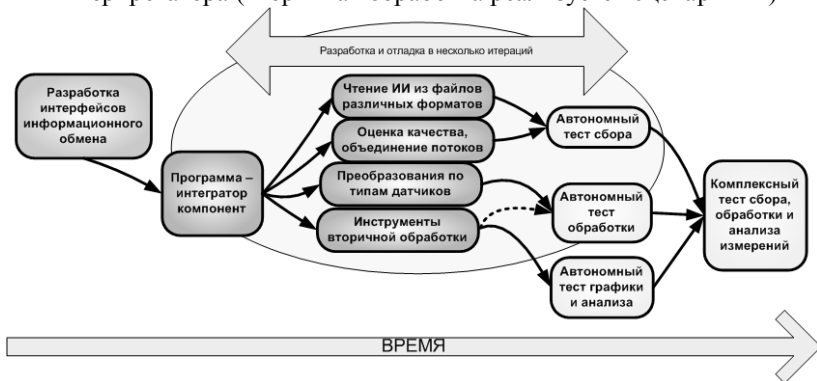


Рис. 4. Второй вариант сетевого план-графика со средствами анализа в виде загрузочных модулей на императивном языке

Различие между вариантами обусловлены основополагающим решением о роли библиотек обработки и головной графической среды:

- первый вариант (рис. 3) имеет трудоёмкое начало и эффективное продолжение. План заключается в качественной проработке системных решений, внедрении интерпретатора и реализации части модулей на простом языке. Ярко выражены две очереди программирования – начальная версия интегрирующей среды запуска и остальных загрузочных модулей и сценариев. Основная масса труда вложена на начальном этапе в объединяющую программу, разработка остальных

компонентов сравнительно проста и автономное тестирование не трудоёмко;

- второй вариант (рис. 4) предполагает одновременную разработку библиотек обработки и анализа, имеет более сложную стадию проверки – на практике тесты могут оказаться не параллельными, а последовательными. Иными словами, сэкономленные на постановке задач и обобщающей программе усилия приводят к сложностям разработки и проверки взаимодействия модулей. В то же время данный метод предпочтителен для сжатых сроков, так как внедрение интерпретатора и разработка интерфейса может являться более трудоёмким.

Представим математическое выражение выбора между вариантами плана. Пусть для первого варианта плана $T_{И1}$ – время на разработку (внедрение) интерпретатора в среду графического анализа, T_{M1} – время разработки зависимых модулей на императивном языке (так как программирование осуществляется одновременно по автономным направлениям, это максимальный период одной из подзадач), $T_{MИ1}$ – время создания сценариев анализа на интерпретаторе, а $T_{план1}$ – время на принятие системных решений и согласований форматов обмена. Тогда общее время разработки составит

$$T_{общ1} = T_{план1} + T_{И1} + \max(T_{M1}, T_{MИ1}) = T_{план1} + T_{И1} + T_{M1}$$

$$\text{Аналогично, для второго варианта плана } T_{общ2} = T_{план2} + T_{И2} +$$

T_{M2}

При этом $T_{план1} \gg T_{план2}$ и $T_{И1} > T_{И2}$, но $T_{M2} > T_{M1} \gg T_{MИ1}$

Преимущества первого варианта очевидны при следующем:

- если любое действие по обработке может быть выражено на языке интерпретатора, и вторая очередь программирования затрагивает только сценарии на простом языке;
- когда возможность реализации пользовательских алгоритмов без перекомпиляции среды является обязательным;
- если имеется возможность качественного выбора системных решений (больше времени на планирование).

Если рассматривать процесс в целом, с учётом автономного и комплексного тестирования, то $(T_{общ1} + T_{тест1}) > (T_{общ2} + T_{тест2})$, несмотря на то, что в отличие от первого варианта $T_{тест2}$ представляет собой не максимум длительности по одному из тестов, а сумму $T_{тест2} = T_{компл.тест2} + \sum T_{авт.тест2}$. Здесь следует заметить, что время тестирования сопоставимо со временем разработки в обеих ситуациях. На практике $0.8T_{общ1} < T_{тест1} < 2.5T_{общ1}$ (значительное превышение времени тестирования над разработкой обусловлено тем, что подвергается проверке сам язык интерпретации – полнота программного интерфейса) и $0.3T_{общ2} < T_{тест2} < 1.2T_{общ2}$.

На практике, как правило $T_{\text{общ1}} \approx 1.5T_{\text{тест1}}$ и $T_{\text{общ2}} \approx 0.5T_{\text{тест2}}$.

Все указанные выше периоды времени являются нечёткими величинами с мерой принадлежности около 0.9.

Этап комплексного тестирования является непредсказуемым. Риски, присущие системной интеграции компонентов связаны с тем, что зависимость от информационных сечений проявляется не на стадии планирования, а различных этапах разработки – на нескольких уровнях технологического процесса. Разработка исчерпывающих спецификаций требований на форматы и интерфейсы может дать обратный результат – «ужесточение» условий информационной совместимости приводит к сложностям в разработке модулей. Кроме того, особенность режима реального времени – жёсткое требование ко времени реакции – не позволяет судить о готовности программного продукта в целом, до тех пор пока не отработана вся технологическая цепочка. Возможно, по результатам комплексного теста потребуется пересмотр организации вычислительного процесса, несмотря на отдельные положительные результаты автономных тестов.

Также необходимо учитывать, что апробация и приёмка системы анализа измерений имеет важную особенность – методы проверки должны разрабатываться до или одновременно с программированием модулей. Если объект испытаний существует и имеется возможность проверять и отлаживать на натуральных измерительных потоках, то это не представляет проблемы. В противном случае необходимо создание аппаратно-программных имитаторов или макетов объекта, т.е. остро встаёт вопрос об объективности тестирования и корректности имитационных потоков. Данная ситуация увеличивает время разработки в 1.5-2 раза и повышает риск информационной несовместимости.

Выводы по проектированию систем сбора и анализа измерительной информации следующие:

- проводить декомпозицию системы в зависимости от стадий технологического процесса – деление составных частей обусловлено изначальной постановкой задачи на сбор/обработку/анализ;
- внедрять интерпретаторы простых языков программирования, что способствует широким возможностям в эксплуатации и простоте функционального наращивания;
- с целью оптимизации трудоёмкости и сроков разработки придерживаться изложенных типовых моделей плана.

Н.С. ФРОЛОВА, С.В. СКВОРЦОВ

Рязанский государственный радиотехнический университет

СИСТЕМА АНАЛИЗА И ОТОБРАЖЕНИЯ ПАРАМЕТРОВ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА ЗАЩИТЫ НА ЭЛЕКТРОСТАНЦИИ

Рассматривается система анализа и отображения параметров технологического процесса защиты позволяющая легко и быстро проанализировать информацию, полученную из архива.

Автоматизация электростанции — одно из самых важных и актуальных направлений технического перевооружения. Уровень автоматизации электростанции сегодня — это не только технический вопрос (ведение режима, исключение отказов и повреждений оборудования, повышение ресурса и т. д.), но и вопрос экономический, от уровня, решения которого зависит эффективность производства энергии (ее себестоимость), а значит конкурентоспособность на рынке.

Технический уровень ранее установленных систем контроля и управления на станции Рязанской ГРЭС был крайне низок, как по качеству технических средств, так и по объему и уровню функциональности. Работоспособность применяемых устройств и изделий системы контроля и управления поддерживалась наличием широкой номенклатуры соответствующих запасных частей. При этом многие заводы-поставщики этих устройств сегодня прекратили выпуск запасных частей к ним как морально и физически устаревших. Поэтому на станции «Рязанская ГРЭС» был создан современный программно-технический комплекс (ПТК) Квинт [1] позволяющий автоматизировать отдельные группы оборудования, отдельные агрегаты и производство в целом.

Квинт ориентирован на выполнение следующих функций: информационных; управляющих; вспомогательных и сервисных; автоматизацию проектирования АСУ ТП; информационная связь с резидентными ПТК

В дополнение к ПТК Квинт была разработана автоматизированная система анализа и отображения параметров технологического процесса защиты «Прекращение поступления питательной воды в котёл». Разработанная программа позволяет представить архивную информацию в удобной для анализа форме. Она используется в двух случаях — для анализа нештатной (в частности, аварийной) ситуации и для подготовки и распечатки протоколов и ведомостей. Анализируемая информация может предоставляться в виде таблиц и графиков.

В ходе написания программы была создана база данных (БД) в которой может храниться и накапливаться информация поступающая с разных датчиков. В БД отражены следующие сведения: об объектах; о типе объекта; о полном имени датчика; об идентификаторах; о номерах; о марках; о флагах; о контроллерах; о АЦП датчиков; об анализируемых параметрах.

В программе предусмотрена настройка БД, которая позволяет подключаться к любому другому серверу с необходимой информацией.

Вход в систему производится отдельно для пользователя и администратора. Чтобы войти в программу пользователю необходимо зарегистрироваться, после успешной регистрации произойдет вход систему (рис.1). Ему будет доступно окно в котором он может проанализировать информацию, полученную из архива. Выбрав нужный период месяца, значения отобразятся в виде графика, после чего можно распечатать или сделать отчет в удобной для пользователя форме (word, excel). Так же, для удобства может быть рассчитан максимум и минимум каждого параметра. Вход администратора произойдет при вводе пароля, после чего ему будет доступна база данных архива, в которой он может добавлять, редактировать и удалять данные, доступные для этой категории пользователей системы (рис. 2).

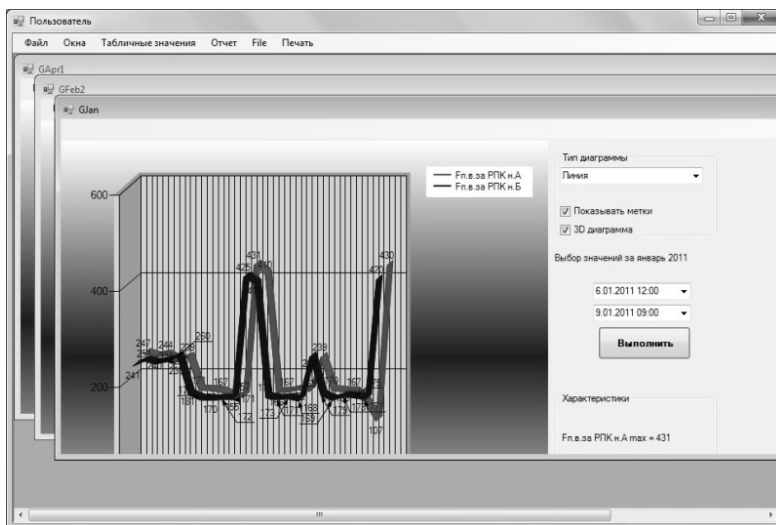


Рис. 1. Интерфейс пользователя

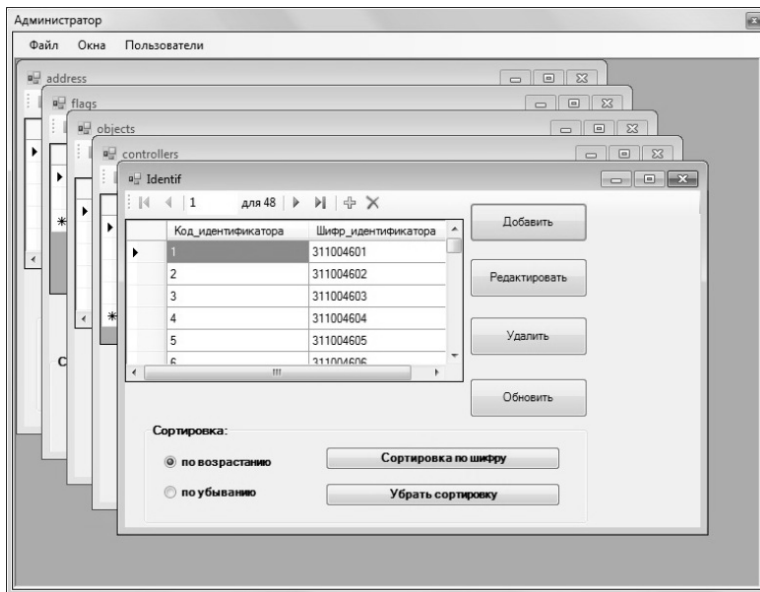


Рис. 2. Интерфейс администратора

Структура автоматизированной информационной системы представляет собой приложение, написанное на языке высокого уровня C# в среде программирования Microsoft Visual Studio 2008. Это приложение, которое с помощью технологии ADO.NET взаимодействует с разработанной БД «Станция анализа», реализовано на языке Transact – SQL в системе управления базами данных Microsoft SQL Server 2008.

Таким образом, разработка и внедрение автоматизированной информационной системы анализа и отображения параметров технологического процесса защиты «Прекращение поступления питательной воды в котёл» с применением ПТК Квинт является перспективным проектом, обеспечивающим ряд дополнительных возможностей для обслуживающего персонала электростанции.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Комплекс программно-технический Квинт. Руководство по применению и эксплуатации (СИКТ.421457.031.РЭ). – М.: НИИ «Теплоприбор», 2009. – 120с.

И.А. ЦВЕТКОВ

Рязанский государственный радиотехнический университет

ИССЛЕДОВАНИЕ ФУНКЦИЙ АЛЬТЕРНАТИВНОГО ОДИНОЧНОГО ВЫБОРА

Предлагается понятие функции альтернативного одиночного выбора; доказываются утверждения об условиях Чернова и возможных конфигурациях условий Чернова для этих функций, их числе для заданного множества вариантов; определяется число таких функций с этими конфигурациями для 2-х, 3-х и 4-элементных множеств вариантов.

Введение. В теории выбора вариантов [1] изучаются *функции выбора*. Для конечного множества вариантов с двумя или более элементами функция выбора C ставит в соответствие каждому аргументу X — *непустому* подмножеству множества вариантов — значение $C(X)$, являющееся подмножеством (возможно, пустым) X .

Функция выбора может, в частности, рассматриваться как модель выбора объектов. Например, если всего есть два объекта: a , b и из них выбирают b , при наличии только a не выбирают ничего, при наличии только b выбирают b , то множество вариантов — это $\{a, b\}$, значения функции выбора C таковы: $C(\{a, b\}) = \{b\}$, $C(\{a\}) = \emptyset$, $C(\{b\}) = \{b\}$.

В теории выбора вариантов исследованы функции *одиночного* выбора [1], все значения которых — 1-элементные подмножества. В работе [3] были предложены и в работе [2] исследованы функции *одиночно-пустого* выбора, все значения которых — пустые или 1-элементные подмножества, причем хотя бы одно значение — пустое подмножество и хотя бы одно значение — 1-элементное подмножество.

В настоящей статье предлагается понятие функции *альтернативного одиночного* выбора, являющейся частным случаем функции одиночно-пустого выбора. Доказываются утверждения об условиях Чернова, возможных конфигурациях условий Чернова (кратко — конфигурациях условий) для функций альтернативного одиночного выбора, о числе всех этих функций для заданного множества вариантов. Определено число функций альтернативного одиночного выбора с каждой из возможных конфигураций условий для 2-х, 3-х и 4-элементных множеств вариантов.

Обозначения и определения. Символы \subseteq и \subset обозначают нестрогое и строгое включения соответственно. Буквой n обозначается натуральное число.

Функция выбора C на конечном n -элементном (где $n \geq 2$) множестве вариантов A задана, если для каждого *непустого* $X \subseteq A$ определено значение функции $C(X) \subseteq X$ [1].

Функцию выбора C на множестве вариантов A называем *функцией альтернативного одиночного выбора*, если $C(X) = \emptyset$ и $|C(Y)| = 1$ для всех таких $X \subset A$ и $Y \subseteq A$, что $|X| = 1$ и $|Y| > 1$. Когда нет альтернатив выбора (при $|X| = 1$), значение $C(X)$ такой функции — пустое подмножество (ничего не выбирается). Когда *есть альтернатива* (при $|Y| > 1$), значение $C(Y)$ такой функции — 1-элементное подмножество (*одиночный* выбор). Поэтому применяем термин «функция альтернативного одиночного выбора» (выбор является одиночным при наличии альтернативы).

Обозначим I_n , $n \geq 2$, множество всех функций альтернативного одиночного выбора на n -элементном множестве вариантов.

Для функции выбора C на множестве вариантов A выполняются условия Чернова: 1) условие *наследования*, если и только если для любых непустых $X \subseteq A$ и $Y \subset A$ при $Y \subset X$ имеет место $Y \cap C(X) \subseteq C(Y)$; 2) условие *константности*, если и только если для любых непустых $X \subseteq A$ и $Y \subset A$ при $Y \subset X$ из $C(X) = \emptyset$ следует $C(Y) = \emptyset$, а из $Y \cap C(X) \neq \emptyset$ следует $Y \cap C(X) = C(Y)$; 3) условие *согласия*, если и только если для любых непустых $X \subset A$ и $Y \subset A$ имеет место $C(X) \cap C(Y) \subseteq C(X \cup Y)$; 4) условие *независимости* от отбрасывания отвергнутых вариантов (кратко — условие независимости), если и только если для любых непустых $X \subseteq A$ и $Y \subset A$ при $C(X) \subseteq Y \subset X$ имеет место $C(X) = C(Y)$ [1].

Называем *конфигурацией условий* строку из четырех символов, каждый из которых равен 0 или 1 и соответствует невыполнению (символ 0) и выполнению (символ 1) условий Чернова в порядке слева направо: наследования, константности, согласия, независимости. Например, конфигурация условий 0010 означает, что условие согласия выполняется, условия наследования, константности и независимости не выполняются.

Символ \blacklozenge показывает окончание доказательства.

Полученные результаты. Для функций альтернативного оди-
ночного выбора автором доказаны следующие утверждения.

Утверждение 1. Для любой функции из I_n при каждом $n \geq 2$ не выполняется условие наследования.

Доказательство. Пусть множество вариантов A такое, что $|A| = n$, $n \geq 2$. Выберем произвольную функцию выбора $C \in I_n$. Покажем, что для функции C на A не выполняется условие наследования.

Пусть $X = A$. Тогда $C(X) = \{t\}$, где t — произвольный (в силу произвольности функции $C \in I_n$) элемент из A . Пусть $Y = \{t\}$. Поскольку $C \in I_n$, получим $C(Y) = \emptyset$, так как $Y \subset A$, $|Y| = 1$. Тогда $Y \cap C(X) = \{t\}$. При $Y \subset X$ не имеет места $Y \cap C(X) \subseteq C(Y)$. Следовательно, для функции C на A не выполняется условие наследования.

Функция $C \in I_n$ выбрана произвольно, поэтому для любой функции из I_n при каждом $n \geq 2$ не выполняется условие наследования. ♦

Утверждение 2. Для любой функции из I_n при каждом $n \geq 2$ не выполняется условие константности.

Доказательство. В силу утверждения 1 для любой функции из I_n при каждом $n \geq 2$ не выполняется условие наследования. Тогда согласно [1, с. 34] для любой функции из I_n при каждом $n \geq 2$ не выполняется условие константности. ♦

Утверждение 3. Для любой функции из I_n при каждом $n \geq 2$ не выполняется условие независимости.

Доказательство. Пусть множество вариантов A такое, что $|A| = n$, $n \geq 2$. Выберем произвольную функцию выбора $C \in I_n$. Покажем, что для функции C на A не выполняется условие независимости.

Пусть $X = A$. Тогда $C(X) = \{t\}$, где t — произвольный (в силу произвольности функции $C \in I_n$) элемент из A . Пусть $Y = \{t\}$. Поскольку $C \in I_n$, получим $C(Y) = \emptyset$, так как $Y \subset A$, $|Y| = 1$. При $C(X) \subseteq Y \subset X$ не имеет места $C(X) = C(Y)$. Следовательно, для функции C на A не выполняется условие независимости.

Функция $C \in I_n$ выбрана произвольно, поэтому для любой функции из I_n при каждом $n \geq 2$ не выполняется условие независимости. ♦

Утверждение 4. При каждом $n \geq 3$ существует функция из I_n , для которой не выполняется условие согласия.

Доказательство. Пусть множество вариантов A такое, что $\{a, b, c\} \subseteq A$, $|A| = n$, $n \geq 3$. Рассмотрим функцию выбора $C \in I_n$, для которой $C(\{a, b\}) = C(\{a, c\}) = \{a\}$, $C(\{a, b, c\}) = \{b\}$. Покажем, что для функции C на A не выполняется условие согласия.

Пусть $X = \{a, b\}$, $Y = \{a, c\}$. Тогда $C(X) \cap C(Y) = \{a\}$, $C(X \cup Y) = C(\{a, b, c\}) = \{b\}$. При $X \subset A$ и $Y \subset A$ не имеет места $C(X) \cap C(Y) \subseteq C(X \cup Y)$. Следовательно, для функции C на A не выполняется условие согласия. ♦

Пример. Не выполняется условие согласия для такой функции $C_1 \in I_3$ на множестве вариантов $A = \{a, b, c\}$, что $C_1(\{a, b\}) = C_1(\{a, c\}) = \{a\}$, $C_1(\{b, c\}) = C_1(\{a, b, c\}) = \{b\}$, $C_1(\{a\}) = C_1(\{b\}) = C_1(\{c\}) = \emptyset$.

Утверждение 5. При каждом $n \geq 2$ существует функция из I_n , для которой выполняется условие согласия.

Доказательство. Обозначим множество вариантов $A = \{a_1, a_2, \dots, a_n\}$, где $n \geq 2$.

Определим функцию выбора C следующим образом: $C(V) = \emptyset$ для всех таких $V \subset A$, что $|V| = 1$; $C(W) = \{a_i\}$ для всех таких $W \subseteq A$, что $|W| > 1$, где i — минимальное значение элементов множества $\{k : a_k \in W\}$. Тогда $C \in I_n$.

Покажем, что для функции C на A выполняется условие согласия.

Рассмотрим для непустых $X \subset A$ и $Y \subset A$ пять случаев.

Случай 1. $|X| = 1$, $|Y| = 1$. По определению функции C получим $C(X) = C(Y) = \emptyset$. Тогда $C(X) \cap C(Y) = \emptyset$, поэтому имеет место $C(X) \cap C(Y) \subseteq C(X \cup Y)$.

Случай 2. $|X|=1$, $|Y|>1$. По определению функции C получим $C(X)=\emptyset$. Тогда $C(X)\cap C(Y)=\emptyset$, поэтому имеет место $C(X)\cap C(Y)\subseteq C(X\cup Y)$.

Случай 3. $|X|>1$, $|Y|=1$. По определению функции C получим $C(Y)=\emptyset$. Тогда $C(X)\cap C(Y)=\emptyset$, поэтому имеет место $C(X)\cap C(Y)\subseteq C(X\cup Y)$.

Случай 4. $|X|>1$, $|Y|>1$, $C(X)\cap C(Y)=\emptyset$. Тогда имеет место $C(X)\cap C(Y)\subseteq C(X\cup Y)$.

Случай 5. $|X|>1$, $|Y|>1$, $C(X)\cap C(Y)\neq\emptyset$. По определению функции C имеем $C(X)=C(Y)=\{a_i\}$, где i — минимальное значение элементов каждого из множеств $\{k: a_k \in X\}$ и $\{l: a_l \in Y\}$. При этом минимальным значением элементов множества $\{s: a_s \in X\cup Y\}$ также является i . Следовательно, по определению функции C получим $C(X\cup Y)=\{a_i\}$. Тогда имеет место $C(X)\cap C(Y)\subseteq C(X\cup Y)$.

Анализ случаев 1–5 позволяет заключить, что для любых непустых $X\subset A$ и $Y\subset A$ имеет место $C(X)\cap C(Y)\subseteq C(X\cup Y)$. Следовательно, для функции C на A выполняется условие согласия. ♦

Пример. Выполняется условие согласия для функции $C_2 \in I_3$ на множестве вариантов $A=\{a_1, a_2, a_3\}$, где

$$\begin{aligned} C_2(\{a_1, a_2\}) &= C_2(\{a_1, a_3\}) = C_2(\{a_1, a_2, a_3\}) = \{a_1\}, \\ C_2(\{a_2, a_3\}) &= \{a_2\}, \\ C_2(\{a_1\}) &= C_2(\{a_2\}) = C_2(\{a_3\}) = \emptyset. \end{aligned}$$

Утверждение 6. При каждом $n \geq 3$ любая функция из I_n имеет одну из двух конфигураций условий: 0000 или 0010, причем существует функция из I_n с каждой из этих конфигураций.

Доказательство. В силу утверждений 1–3 для любой функции из I_n при каждом $n \geq 3$ не выполняются условия наследования, константности, независимости. Следовательно, при каждом $n \geq 3$ в I_n нет функций с конфигурациями условий, отличными от 0000 и 0010.

В силу утверждения 4 при каждом $n \geq 3$ существует функция из I_n , для которой не выполняется условие согласия. Следовательно, при

каждом $n \geq 3$ в I_n существует функция с конфигурацией условий 0000.

В силу утверждения 5 при каждом $n \geq 3$ существует функция из I_n , для которой выполняется условие согласия. Следовательно, при каждом $n \geq 3$ в I_n существует функция с конфигурацией условий 0010. ♦

Утверждение 7. При каждом $n \geq 2$ имеет место $|I_n| = \prod_{k=2}^n k C_n^k$.

Доказательство. Для любого 1-элементного подмножества n -элементного множества вариантов есть ровно одно значение (пустое подмножество) функций из I_n . Для любого k -элементного подмножества n -элементного множества вариантов при $2 \leq k \leq n$ есть ровно k различных значений функций из I_n . Следовательно, по комбинаторному правилу произведения для всех C_n^k k -элементных подмножеств n -элементного множества вариантов имеем ровно $k C_n^k$ различных наборов значений функций из I_n , где $2 \leq k \leq n$. Тогда по этому правилу получим $|I_n| = \prod_{k=2}^n k C_n^k$. ♦

Имеем $|I_2| = 2$, $|I_3| = 24$, $|I_4| = 20736$. У обеих функций из I_2 конфигурация условий 0010. В I_3 есть по 12 функций с конфигурациями условий 0000 и 0010. В I_4 20400 и 336 функций имеют конфигурации условий 0000 и 0010 соответственно (определено перебором на ЭВМ всех функций из I_4).

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Айзерман М.А., Алескеров Ф.Т. Выбор вариантов: основы теории. – М.: Наука, 1990. – 240 с.
2. Цветков И.А. Условия Чернова для функций одиночно-пустого выбора // Вестник РГРТУ. — 2012. – № 1 (39). – Ч. 2. – С. 57–64.
3. Цветков И.А. Функции одиночно-пустого выбора // Перспективы развития гуманитарных и технических систем: материалы Всероссийской науч. конф. – Ч. 2. – Таганрог: изд-во ТТИ ЮФУ, 2011. – С. 77–79.

Е.А. ЦЫРУЛЬНИКОВА

Рязанский государственный технический университет

НЕРАЗРУШАЮЩИЙ КОНТРОЛЬ ТРУБОПРОВОДА*Рассматриваются методы диагностики состояния трубопровода*

Повышение надежности и безопасности эксплуатации магистральных трубопроводов требует разработки надежных методов и средств оперативного контроля их состояния в процессе эксплуатации и профилактического осмотра. Значительная протяженность действующих коммуникаций, труднодоступность их пролегания требует использования бесконтактных методов неразрушающего контроля, таких как:

- Радиоволновый метод основан на регистрации изменений параметров электромагнитных волн радиодиапазона, взаимодействующих с объектом контроля. Недостатком метода является сравнительно низкая разрешающая способность устройств, реализующих этот метод, обусловленная малой глубиной проникновения радиоволн в металлы.

- Тепловой метод основан на регистрации изменений тепловых или температурных полей контролируемых объектов. Он применим к объектам из любых материалов. Недостатком этого метода контроля является необходимость использования контактных устройств, что затрудняет процессы автоматизации при непрерывных измерениях и контроле движущихся объектов.

- Оптический метод основан на наблюдении или регистрации параметров оптического излучения, взаимодействующего с объектом контроля. Не зависит от материала объекта. Недостатками метода являются узкий диапазон контролируемых параметров, жесткие требования к состоянию окружающей среды и чистоте поверхности изделия.

- Радиационный метод основан на регистрации и анализе проникающего ионизирующего излучения после взаимодействия его с объектом контроля. К недостаткам относятся повышенные требования к технике безопасности, сложность, дороговизна и громоздкость аппаратуры, а также ограничения, связанные со сравнительно небольшими толщинами объекта контроля.

- Неразрушающий контроль методом проникающих веществ основан на явлении капиллярного проникновения хорошо смачивающих пробных веществ (жидкости) в полость дефектов объема контроля. Метод требует значительных временных затрат на пропитку и проявление индикаторной жидкости, что приводит к снижению производительности контроля и сложности его автоматизации.

- Магнитный метод основан на анализе взаимодействия магнитного поля с контролируемым объектом. Как правило, его применяют для контроля объектов из ферромагнитных материалов.

- Электрический метод основан на регистрации параметров электрического поля, взаимодействующего с контролируемым объектом, или поля, возникающего в контролируемом объекте в результате внешнего воздействия. Недостатками методов являются необходимость контакта с объектом контроля, жесткие требования к чистоте поверхности изделия, трудности автоматизации процесса измерения и зависимость результатов измерения от состояния окружающей среды.

- В акустическом методе для обнаружения местоположения и размера дефектов используются звуковые волны. Широко применяют благодаря ряду преимуществ: подходит практически для любого материала; волны легко вводятся в объект контроля, хорошо распространяются в металлах, бетоне и других материалах; эффективны при выявлении дефектов с малым раскрытием, чувствительны к изменению структуры и физико-механических свойств материалов. Использование различных типов волн (продольных, поперечных, поверхностных, нормальных и других) расширяет возможности акустических методов.

Для поиска течей на действующих трубопроводах без вскрытия грунта предлагаю использовать акустический метод диагностики. Конструктивным решением является течеискатель, а именно акустическая локация. Устройство преобразует вибрации грунта или колебаний газовой среды (воздуха) в электрические сигналы, частотной и амплитудной селекции этих сигналов. Непосредственного контакта датчика с объектом при этом не требуется. Новизна конструкции в использовании рiс-контроллера, что позволяет уменьшить габариты и масса конструкции и применить современные технологии.

В настоящее время в числе первоочередных задач, стоящих перед нефтегазотранспортными предприятиями и теплотсетями России, выступает решение проблемы предупреждения отказов и продления срока службы трубопроводов. Использование акустической локации сократит количество ремонтных работ и предотвратит утечки веществ в грунт.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Сударикова Е.В. Неразрушающий контроль в производстве: Учебное пособие. Ч.1. – СПб.: ГУАП. 2007. – 137 с.
2. Ермолов И.Н., Алешин Н.П., Потапов А.И. Акустические методы контроля. Книга 2. – М.: Высшая школа. 1991. – 283 с.