

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

УДК 004.05

Финашина
Елизавета Андреевна

**МЕТОДЫ И СРЕДСТВА ДЛЯ АВТОМАТИЗАЦИИ И ПОДДЕРЖКИ
РЕГРЕССИОННОГО ТЕСТИРОВАНИЯ ВЕБ-ПРИЛОЖЕНИЙ**

АВТОРЕФЕРАТ

на соискание степени магистра информатики и вычислительной техники
по специальности 1-40 81 01 – Информатика и технологии разработки
программного обеспечения

Научный руководитель
Матвейчук Наталья Михайловна
кандидат физико-математических
наук

Минск 2018

ВВЕДЕНИЕ

Проблема качества программного обеспечения становится сегодня все более острой, особенно по мере расширения использования информационных технологий и роста сложности ПО. Высокое качество продуктов дает разработчикам не только конкурентные преимущества и кредит доверия клиентов, но и облегчает сопровождение и развитие ПО.

Как бы хорошо программные продукты не были разработаны первоначально, они подвержены неизбежным изменениям. Контроль над изменениями – критический фактор для сохранения полезности программ.

Стандартный процесс разработки программного обеспечения хорошо работает в случае добавления к системе новых функциональных возможностей; в других аспектах он требует определённой адаптации. Для повторной проверки корректности функциональных возможностей, например, унаследованных из предыдущей версии, в процессе разработки необходимо использовать регрессионное тестирование, то есть повторное тестирование части программы, зависящей от внесённых изменений. Регрессионное тестирование гарантирует, что внесённые в код изменения корректны и не воздействуют неблагоприятно на другие блоки программы. Регрессионные системные тесты могут рассматриваться как частичные требования к новым версиям системы.

Регрессионное тестирование может быть отнесено к методам профилактического сопровождения, применяемым в ходе модификации программного изделия. Профилактическое сопровождение с применением методов регрессионного тестирования не только устраняет необходимость в дальнейшем сопровождении при исправлении дефектов, но и делает возможным экономию проектных ресурсов за счёт их более гибкого и эффективного использования на этапах сопровождения, интеграционного и системного тестирования. Это позволяет улучшить способность программного изделия адаптироваться к любым изменениям требований без потери гарантированного качества при входе на последнюю фазу жизненного цикла программы.

Важным различием между регрессионным тестированием и тестированием в процессе разработки является наличие в процессе регрессионного тестирования готового набора тестов, доступного для повторного использования. Одна из стратегий регрессионного тестирования допускает повторный запуск всех этих тестов, но такой метод потребляет чрезмерно много времени и ресурсов. Альтернативный метод, выборочное повторное тестирование, выбирает из старого набора тесты, которые считает необходимым для тестирования изменённой программы.

Во время, когда система подвергается изменению, трудно определить, какие тесты подлежат модификации и как именно они должны быть изменены. Большинство коммерческих средств автоматизации тестирования обеспечивает не более чем возможность сохранять существующие тесты и запускать их повторно после каждого изменения. У них отсутствует возможность какого-либо автоматического отбора тестов. Методики выборочного регрессионного тестирования пока ещё не нашли своего воплощения в промышленных программных средствах, доступных рядовому пользователю.

Настоящая работа посвящена разработке системы для автоматизации и поддержки регрессионного тестирования, а так же созданию методики, позволяющей производить оптимизацию тестового набора для проведения регрессионного тестирования.

В ходе работы будет создан программный продукт, автоматизирующий применение разработанной методики. Её применение позволит сократить пакет регрессионных тестов на этапах поддержки и сопровождения программного продукта.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы исследования

Регрессионное тестирование относится к необходимым методам профилактического сопровождения и применяется в ходе процесса разработки и модификации программного продукта. Однако, такой род деятельности является крайне ресурсоемким и, как следствие, дорогостоящим. Это обусловлено необходимостью проводить регрессионное тестирование в случае внесения даже малейших изменений в код программы, в то время, как процесс регрессионного тестирования может включать в себя исполнение достаточно большого количества тестов на скорректированной версии программы. И несмотря на то, что усилия, требуемые для внесения небольших изменений, как правило, минимальны, они могут требовать достаточно больших усилий для проверки качества измененной программы.

Тем не менее, надежная и эффективная разработка и сопровождение программного обеспечения невозможны без регрессионного тестирования. Одним из очевидных решений здесь является автоматизация процесса тестирования, помогающая ускорить создание продукта и улучшить его качество. Автоматизация тестирования становится наиболее рентабельным и удобным, когда применяется к регрессионным тестам.

Таким образом, разработка системы для автоматизации и поддержки регрессионного тестирования веб-приложений является актуальной на сегодняшний день.

Степень разработанности проблемы

У большинства современных средств автоматизации тестирования отсутствует возможность какого-либо автоматического отбора тестов. На данный момент методики выборочного регрессионного тестирования не нашли своего воплощения в промышленных программных средствах.

Анализ практического опыта проведения регрессионного тестирования выявил множество недостатков в методах оптимизации регрессионного тестового набора.

Существующие подходы к регрессионному тестированию не учитывают последних достижений в этой области, оставляя без внимания такие возможности, как уменьшение объёма тестируемой программы, методы упорядочения и функции предсказания целесообразности отбора тестов, так что

пользователю не всегда понятны основания, которыми руководствуются при проведении выборочного регрессионного тестирования.

При разработке системы поддержки регрессионного тестирования для оптимизации тестового набора был применен новый метод минимизации на основе редукции тестового пакета. Данный метод помогает сократить и выявить из исходного тестового набора подмножество тестов для повторного проведения регрессионного тестирования.

Цель и задачи исследования

Целью данной работы является оптимизация процесса отбора тестов для проведения регрессионного тестирования на основе метода редукции тестового набора и его программная поддержка.

Для достижения поставленной цели необходимо:

1 Обосновать рациональность внедрения автоматизированного тестирования для выполнения регрессионных тестов на основе оценки эффективности.

2 Сформировать новый метод, позволяющий производить оптимизацию тестового набора на основе анализа существующих методов и средств регрессионного тестирования ПО.

3 Определить требования к разрабатываемой системе поддержки регрессионного тестирования, позволяющей автоматизировать процесс отбора тестов.

Область исследования

Содержание диссертационной работы соответствует образовательному стандарту высшего образования второй ступени (магистратуры) специальности 1-40 81 01 «Информатика и технологии разработки программного обеспечения».

Теоретическая и методологическая основа исследования

В основу диссертации легли исследования зарубежных и отечественных ученых в области организации процесса регрессионного тестирования и автоматизации тестирования. При решении поставленных задач были проанализированы существующие методы и средства для автоматизации регрессионного тестирования.

Для разработки нового метода отбора тестов были использованы знания логического проектирования, применялись комбинаторные алгоритмы дискретной математики.

В качестве инструментальных средств использовались объектно-ориентированный язык программирования Java, MySQL Server, сервер Apache Tomcat, фреймворки Hibernate и Spring.

Информационная база исследования сформирована на основе данных, опубликованных в журналах технического направления, а также архивов статистических данных компании EPAM Systems.

Научная новизна

Научная новизна заключается в создании нового метода минимизации тестового набора для проведения регрессионного тестирования с аналитическими элементами на основе задач комбинаторики.

Теоретическая значимость диссертации заключается в описании процесса организации отбора регрессионных тестов в рамках проектной деятельности.

Практическая значимость результатов исследования заключается в разработанном методе поддержки регрессионного тестирования, которая позволит оптимизировать процесс отбора подмножества тестов для повторного тестирования, а также предоставлять аналитические данные для пользователей.

Основные положения, выносимые на защиту

1 Метод, позволяющий производить оптимизацию тестового набора для проведения регрессионного тестирования на основе редукции тестового пакета.

2 Алгоритм метода редукции тестового набора.

3 Система для поддержки регрессионного тестирования, автоматизирующая применение разработанной методики минимизации регрессионного тестового набора.

Апробация диссертации и информация об использовании ее результатов

Основные теоретические результаты и законченные этапы диссертационной работы отражены в материалах 53-й научно-технической конференции аспирантов, магистрантов и студентов БГУИР, URL: <https://libeldoc.bsuir.by/handle/123456789/13102> и в научном журнале

Публикации

Изложенные в диссертации основные положения и выводы опубликованы в 2 печатных работах, представленные в виде двух докладов:

1 Минимизация объема тестовых наборов для проведения регрессионного тестирования.

2 Поиск минимального безусловного диагностического теста как метод редукции регрессионного тестового набора.

Общий объем публикаций по теме диссертации составляет 7 страниц.

Структура и объем работы

Диссертация состоит из введения, общей характеристики работы, трех глав с краткими выводами по каждой главе, заключения, библиографического списка и приложений.

Во введении рассмотрено современное состояние проблемы минимизации регрессионного пакета и средств автоматизации тестирования обеспечивающих возможность автоматического отбора тестов, определены основные цели и задачи исследований.

В первой главе раскрыты понятия качества программного продукта и регрессионного тестирования на стадиях сопровождения и поддержки программного обеспечения. Рассмотрены виды регрессионного тестирования, особенности тестирования веб-приложений и необходимость использования автоматизации для выполнения регрессионного тестирования.

Во второй главе была разработана функциональная модель организации процесса тестирования программного обеспечения, были рассмотрены подходы к автоматизации процесса тестирования и существующие средства автоматизации тестирования. Были проанализированы существующие методы выборочного регрессионного тестирования. Представлен новый метод отбора регрессионных тестов – метод поиска минимального безусловного диагностического теста как метод редукции регрессионного тестового набора. Метод был апробирован на реальных данных. Результаты использования данного подхода к минимизации тестового набора также представлены в этой главе. Был сделан вывод о целесообразности и эффективности применения данного метода оптимизации регрессионного тестового набора.

Третья глава посвящена постановке задачи, формированию требований к разрабатываемой системе. Чтобы сделать процесс разработки понятным, были построены диаграммы последовательности системы, диаграмма развертывания, диаграмма компонентов системы, диаграмма вариантов использования и обобщенный алгоритм работы приложения. В заключение, было подробно расписано руководство по работе с разработанной системой.

В приложении представлены публикации и графический материал в виде презентации.

Общий объем диссертационной работы составляет 90 страниц. Из них 79 страниц основного текста, 50 иллюстраций, 3 таблицы, библиографический список из 43 наименований, список собственных публикаций соискателя из 2 наименований, 2 приложения.

Библиотека БГУИР

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении рассмотрено современное состояние проблемы минимизации регрессионного набора и средств автоматизации тестирования обеспечивающих возможность автоматического отбора тестов, определены основные цели и задачи исследований. Определены основные направления исследований, а также дается обоснование актуальности темы диссертации.

В общей характеристике работы показана актуальность проводимых исследований, степень разработанности проблемы, сформулированы цель и задачи диссертации, обозначена область исследований, научная (теоретическая и практическая) значимость исследований, а также апробация работы.

В первой главе представлены различные формулировки определения регрессионного тестирования. В данной работе рассматривается определение регрессионного тестирования, как повторного тестирования программы с использованием тестов, отобранных из исходного множества тестов.

В этой главе описаны основные цели и задачи регрессионного тестирования.

Одна из целей регрессионного тестирования состоит в том, чтобы после того, как в предварительно протестированную программу внесены изменения, провести повторное тестирование, основываясь только на информации об изменениях. Проблема оптимального регрессионного тестирования состоит в определении минимального числа тестов, необходимых для повторной проверки правильности изменённой программы на фазе сопровождения.

Другая цель регрессионного тестирования состоит в проверке изменённой программы для достижения уверенности в отсутствии неблагоприятного воздействия результатов исправления ошибок и новых функциональных возможностей, включённых в очередную версию программы, будь то новые свойства или изменённые существующие. Иными словами, регрессионное тестирование имеет целью восстановление уверенности в том, что программа функционирует в соответствии со своей спецификацией, и что изменения не привели к внесению новых ошибок в ранее протестированный код.

Регрессионное тестирование также может выполняться с целью:

- 1 Проверять исправление дефектов или гарантировать, что программа работает согласно спецификации.
- 2 Поддерживать уровень надёжности изменённой программы.
- 3 Уменьшать издержки, то есть сокращать затраты на тестирование.

Для достижения этих целей могут использоваться две стратегии. Одна из них, называемая методом повторного прогона всех тестов, требует применения всего множества исходных тестов. Другая, называемая выборочной, отбирает

подмножество исходного набора тестов, учитывая изменения в программе (это может быть достигнуто с помощью методов минимизации, безопасных или основанных на покрытии методов).

Рассмотрены также какие бывают типы изменений.

По типу изменяемого объекта изменения делятся на три основных класса: изменения потока управления, изменения потока данных, и изменения, не затрагивающие ни потока управления, ни потока данных.

Типы регрессионного тестирования.

По типам модификаций выделяют три типа сопровождения программного обеспечения:

– Корректирующее сопровождение, называемое обычно исправлением ошибок, выполняется в ответ на обнаружение ошибки, не требующей изменения спецификации требований. При корректирующем сопровождении производится диагностика и корректировка дефектов в программном обеспечении с целью поддержания системы в работоспособном состоянии.

– Адаптация системы в ответ на требования изменения данных или сред выполнения составляет адаптивное сопровождение, которое применяется, когда существующая система улучшается или расширяется, а спецификация требований изменяется с целью реализации новых функций.

– Усовершенствующее (прогрессивное) сопровождение включает любую обработку с целью повышения эффективности работы системы или её сопровождения.

Выделяют два типа регрессионного тестирования:

– Прогрессивное регрессионное тестирование предполагает модификацию технического задания. В большинстве случаев при этом к системе программного обеспечения будут добавлены новые модули.

– При корректирующем регрессионном тестировании техническое задание не изменяется. Модифицируются только некоторые операторы программы и, возможно, некоторые конструкторские решения.

Во второй главе произведен обзор существующих средств для автоматизации тестирования и анализ методов выборочного регрессионного тестирования. Помимо этого, в данной главе подробно описывается новый метод редукции тестового набора для выполнения регрессионного тестирования, основанный на поиске минимального безусловного диагностического теста путем решения комбинаторной задачи.

Многие комбинаторные оптимизационные задачи сводятся к задаче о кратчайшем покрытии, которая ставится следующим образом. Пусть даны некоторое множество $A = \{a_1, a_2, \dots, a_n\}$ и совокупность его подмножеств B_1, B_2, \dots, B_m , т.е. $B_i \subseteq A$, $i = 1, 2, \dots, m$, причём $B_1 \cup B_2 \cup \dots \cup B_m = A$.

Требуется среди данных подмножеств выделить такую совокупность $B_{i_1}, B_{i_2}, \dots, B_{i_k}$, с минимальным k , чтобы каждый элемент из A хотя бы в одно из B_{i_j} ($j = 1, 2, \dots, k$), т.е. $B_{i_1} \cup B_{i_2} \cup \dots \cup B_{i_k} = A$.

Практический интерес представляет задача нахождения минимального диагностического теста, которую можно свести к задаче о кратчайшем покрытии. Для этого сначала следует получить матрицу различий, строки которой соответствуют парам строк диагностической матрицы D и показывают, какими компонентами отличаются строки в этих парах. При этом используется покомпонентная операция сложения по модулю два, выполняемая над всеми парами строк матрицы D . Затем надо найти для полученной матрицы различий кратчайшее покрытие, только не строчное, а столбцовое, т.е. состоящее из некоторых столбцов, покрывающих в совокупности все строки данной матрицы. Множество признаков, соответствующих столбцам из найденного покрытия, будет искомым.

Рассмотрим полученную матрицу с точки зрения тестового набора регрессионного тестирования. Положим, строки этой матрицы, соответствующие элементам множества A – это множество всех функциональных модулей программы, которые должна подвергнуться проверке, а столбцы – соответствующие элементам множества – тесты. Тогда полученная матрица – и есть минимальный тестовый набор для регрессионного тестирования.

Рассмотрим на небольшом примере работу данного алгоритма. Пусть дана матрица Z (рисунок 1).

1	2	3	4	5	6	7	8	
1	0	0	1	1	0	1	1	a
0	0	1	1	0	1	0	1	b
0	1	0	1	0	0	0	0	c
0	0	0	1	1	1	0	1	d
0	0	0	1	1	0	1	1	e
1	0	1	0	1	0	1	1	f

Рисунок 1 – Матрица Z

Составим для матрицы Z матрицу различий, строки которой соответствуют парам строк диагностической матрицы и показывают, какими компонентами отличаются строки в этих парах. При этом будем использовать покомпонентную операцию сложения по модулю два, выполняя её над всеми парами строк диагностической матрицы.

Для рассматриваемого примера матрица различий (рисунок 2) примет следующий вид:

1	2	3	4	5	6	7	8	
1	0	1	0	1	1	1	0	ab
1	1	0	0	1	0	1	1	ac
1	0	0	0	0	1	1	0	ad
1	0	0	0	0	0	0	0	ae
0	0	1	1	0	0	0	0	af
0	1	1	0	0	1	0	1	bc
0	0	1	0	1	0	0	0	bd
0	0	1	0	1	1	1	0	be
1	0	0	1	1	1	1	0	bf
0	1	0	0	1	1	0	1	cd
0	1	0	0	1	0	1	1	ce
1	1	1	1	1	0	1	1	cf
0	0	0	0	0	1	1	0	de
1	0	1	1	0	1	1	0	df
1	0	1	1	0	0	0	0	ef

Рисунок 2 – Матрица различий для матрицы Z

К данной матрице можно применить правила редукции, т.е. удалить из неё поглощающие строки и поглощаемые столбцы. Булев вектор b поглощает булев вектор a , если a имеет единицы везде, где имеет единицы b . Поглощающая строка удаляется только в том случае, когда поглощаемая ею строка остаётся в матрице. Аналогичное справедливо и в отношении удаляемых столбцов. В результате удаления поглощающих строк получаем матрицу, изображенную на рисунке 3:

1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	0	1	0	1	0	0	0
0	1	0	0	1	1	0	1
0	1	0	0	1	0	1	1
0	0	0	0	0	1	1	0

Рисунок 3 – Матрица после удаления поглощающих строк

Последующее удаление поглощаемых столбцов приводит нас к матрице, представленной на рисунке 4:

1	3	4	6	7
1	0	0	0	0
0	1	0	0	0
0	1	1	0	0
0	0	1	1	0
0	0	1	0	1
0	0	0	1	1

Рисунок 4 – Матрица после удаления поглощающих столбцов

В матрице вновь появляется поглощающая строка, которую, следовательно, можно удалить. Кратчайшее покрытие матрицы изображено на рисунке 5.

1	3	5	6	7
1	0	0	0	0
0	1	0	0	0
0	0	1	1	0
0	0	1	0	1
0	0	0	1	1

Рисунок 5 – Кратчайшее покрытие матрицы

Одним из кратчайших столбцовых покрытий данной матрицы является совокупность столбцов 1, 3, 5 и 6. Соответствующая совокупность признаков является минимальной для заданной диагностической матрицы.

Составим алгоритм по шагам метода редукции тестового набора.

Шаг 1. Составить диагностическую матрицу строки которой соответствуют модулям программы, а столбцы тестам.

Шаг 2. Составить матрицу различий для исходной диагностической матрицы. Используется покомпонентная операция сложения по модулю два, выполняемая над всеми парами строк диагностической матрицы.

Шаг 3. Применить минимаксный алгоритм к матрице различий. С помощью него находим кратчайшее столбцовое покрытие матрицы различий.

Множество признаков, соответствующих столбцам из найденного покрытия и будет результативным тестовым набором. Графическое

представление алгоритма метода редукции тестового набора изображено на рисунке 6.

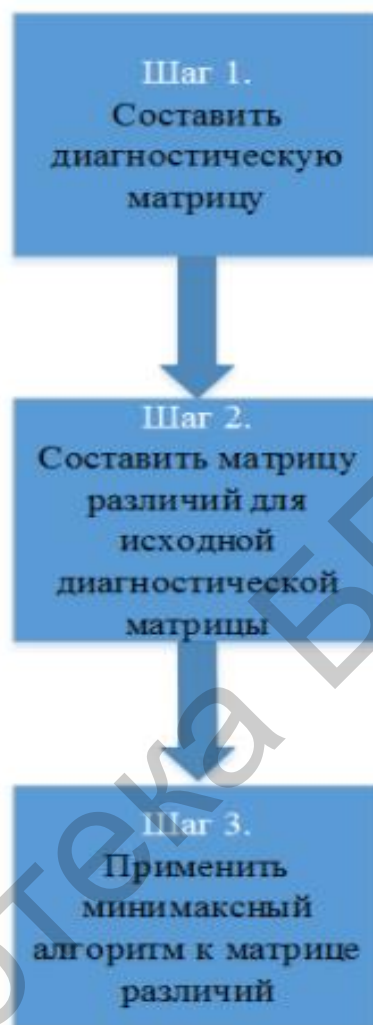


Рисунок 6 – Алгоритм по шагам метода редукции тестового набора

Во второй главе также предоставлены данные апробирования вышеизложенного метода.

В третьей главе в контексте регрессионного тестирования ставится задача создания системы для автоматизации процесса отбора тестового пакета методом редукции тестового набора.

Система разработана с применением вышеописанного метода минимизации регрессионных тестов, позволяющая эффективно сократить предоставленный изначально набор тестов.

Рассмотрена архитектура создаваемой системы. Произведен обзор методов решения поставленных задач, а так же разработана информационная модель

базы данных, основанная на правиле 3NF (англ. Third normal form – третья нормальная форма).

Выполнены модели представления системы в виде UML моделей, наглядно отражающие различные аспекты системы. Для этого использовались следующие диаграммы: последовательности, классов, состояния, компонентов, развертывания, вариантов использования. А так же составлено пошаговое руководство пользователя.

Библиотека БГУИР

ЗАКЛЮЧЕНИЕ

Основные научные результаты диссертации

По итогам проведенной работы можно сделать выводы:

1 Рассмотрены понятия и виды регрессионного тестирования. Обоснована необходимость применения автоматизации для регрессионного тестирования, основанная на оценке эффективности.

2 На основе анализа существующих методов и средств регрессионного тестирования ПО был сформирован новый метод оптимизации регрессионного тестового набора. Метод был апробирован на реальных данных и доказал эффективность его использования, так как способен выбирать тесты, обнаруживающих ошибки.

3 Разработана система поддержки регрессионного тестирования, автоматизирующая новый метод отбора тестов и позволяющая оптимизировать выполнение регрессионных тестов.