

Выбор способа тестирования как решение многокритериальной задачи

Е.Ю. Галимова, А.Н. Коваленко

Высшая школа печати и медиатехнологий Санкт-Петербургского государственного университета промышленных технологий и дизайна, г. Санкт-Петербург, Российская Федерация

Аннотация: Задача выбора метода тестирования программного модуля ставится и рассматривается как задача многокритериальной оптимизации. С точки зрения оптимизации задача имеет вырожденный вид, поскольку множество допустимых решений в ней конечно и мало. А именно, метод тестирования отдельного программного модуля может быть автоматизированным, ручным или смешанным. Эта формальная простота компенсируется значительным количеством критериев и ступенчатым алгоритмом их вычисления, основанным на обработке экспертных оценок. Оценки даются как в группе программистов – создателей программного продукта, так и в группе системных тестировщиков. Предложен алгоритм решения поставленной задачи оптимизации, основанный на методе групповой линейной свертки. Итоговое значение свертки соответствует одному из трех возможных методов тестирования. Алгоритм может быть распространён на другие задачи дискретной оптимизации с конечным количеством элементов допустимого множества.

Ключевые слова: ручное тестирование ПП, автоматизированное тестирование ПП, полуавтоматизированное тестирование ПП, метод выбора, задача многокритериальной оптимизации, алгоритм решения многокритериальной задачи.

Введение

Тестирование является трудоемким и дорогостоящим этапом разработки программного продукта (ПП), но необходимым для обеспечения качества. Под качеством понимается степень соответствия присущих ПП характеристик заявленным требованиям [1]. Правильное планирование позволяет сократить затраты. На этапе планирования тестирования встает вопрос выбора способа тестирования между ручным, автоматизированным и смешанным. В данной работе предлагается решать вопрос выбора как многокритериальную задачу [2]. Поскольку ПП обладает целым рядом характеристик, авторы отказались от использования методов однокритериальной оптимизации [3], был выбран метод линейной свертки. Вопрос выбора способа тестирования является актуальным, так как в

нынешний век глобальной автоматизации ручное тестирование продолжает активно применяться [4].

Основная часть

Постановка задачи о формальной оценке ПП для целей тестирования

Пусть объект G описывается критериями H_1, H_2, \dots, H_k , которым можно приписать числовые значения h_1, h_2, \dots, h_k . Пусть по совокупности значений $\{h_1, h_2, \dots, h_k\}$ определяется некоторая характеристика R_G , позволяющая формально ответить на некоторый вопрос, касающийся объекта G . Пусть эта характеристика R_G дискретна, и набор её возможных значений конечен.

В данной работе объект G – это модуль ПП. Возможно, что это ПС целиком, если оно оценивается как достаточно простое для тестирования.

В данной работе критерии H_1, H_2, \dots, H_k – это помещённые в подразделе 2.2 вопросы о модуле тестируемого ПС. Обозначим этот набор (множество) через M .

Характеристика R_G связывается (должна указывать) на выбор автоматизированного или ручного способа тестирования ПП или на перенесение решения этого вопроса на более сложные алгоритмы, в том числе, на экспертное решение.

Таким образом, в рассматриваемой задаче может быть три решения (три альтернативы):

$R1_G$ – использование для данного модуля ПП автоматизированного тестирования;

$R2_G$ – использование для данного модуля ПП ручного тестирования;

R3_G – указание на некоторый пограничный случай между выбором автоматизированного и ручного методов тестирования.

В данной работе предлагается алгоритм, позволяющий формальным образом определять некоторую величину **P**, однозначно указывающую на выбор между **R1_G**, **R2_G**, **R3_G**.

Такая задача, строго говоря, является задачей многокритериальной оптимизации с критериями **H₁**, **H₂**, ..., **H_k**, но представляет собой вырожденный случай, поскольку допускает прямое вычисление всех возможных значений величины **P**. Тем не менее, она решается на основе идей, используемых для исследования многокритериальных задач.

Входным материалом в задачу являются два вида экспертных оценок, что порождает двухстадийный процесс формирования оценок **h₁**, **h₂**, ..., **h_k**. Поскольку экспертные оценки по определению дискретны, то формально решаемая задача относится к классу дискретных задач оптимизации.

Другой особенностью является деление вопросов множества **M** на три группы. Этот акт позволяет учитывать обобщенный опыт тестирования ПП. В алгоритме оптимизации такой факт учитывается при вычислении величины **P**.

Исходная информация в задаче состоит:

а) из ответов (экспертных оценок) $\beta = (\beta_1, \beta_2, \dots, \beta_k)$, даваемых в группе программистов – создателей данного ПП. Вопросы из множества **M** сформулированы именно под ответ «да» или «нет». В соответствии с этим величина β_i , $i=1, 2, \dots, k$ представляет собой **1** или **0**. Такая информация не является обременительной для создателя ПП. Вся сложность уходит в дальнейшую обработку.

б) из экспертных оценок $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_k)$ (называемых в работе весами), которые даются в группе тестирования.

Наличие двух типов экспертной информации делает решение задачи более объективным, поскольку работа будет идти не с реальными физическими величинами, а с искусственными бальными оценками.

За основу решения задачи многокритериальной оптимизации берётся метод свёртки [5].

Разновидность метода свёртки – групповая свёртка – используется в связи с тем, что критерии (вопросы) H_1, H_2, \dots, H_k делятся на группы, как это и сделано в подразделе 2.2. А именно, $H^I = (H_1, H_2, \dots, H_{k_1})$ – это первая группа вопросов. В них веса $\alpha_1, \alpha_2, \dots, \alpha_{k_1}$, близкие к 1, ведут преимущественно к автоматизированному тестированию. Во второй группе критериев $H^{II} = (H_{k_1+1}, H_{k_1+2}, \dots, H_{k_2})$ веса $\alpha_{k_1}, \alpha_{k_1+1}, \dots, \alpha_{k_2}$, близкие к 1, ведут преимущественно к ручному способу тестированию. В третьей, в целом нейтральной группе критериев $H^{III} = (H_{k_2+1}, H_{k_2+2}, \dots, H_k)$, ответы близкие к 1, ведут преимущественно к выбору автоматизированного вида тестирования. Естественно, $k_1 \leq k_2 \leq k$. Такие оценки наиболее удобно давать эксперту. Их вид далее учитывается в алгоритме обработки.

Алгоритм решения многокритериальной задачи

Ход решения многокритериальной задачи включает себя ряд действий, которые изобразим в виде следующей схемы:



Рис. 1. – Схема решения многокритериальной задачи

Разберём приведённую схему.

Ячейка 1. Оценки $\beta_1, \beta_2, \dots, \beta_k$ (ответы на вопросы \mathbf{M}) даются в группе создателей ПП, поскольку программисты лучше всех знают особенности созданного ими ПП. Как уже указывалось, для облегчения работы программистов оценки имеет вид **1** («да») и **0** («нет») [6]. При этом в оценке не учитывается деление критериев на три группы ($\mathbf{R1}_G$, $\mathbf{R2}_G$, $\mathbf{R3}_G$). Дальнейшее усложнение оценок будет идти при обработке исходных массивов на 4-й и 5-й стадиях алгоритма.

Ячейка 2. Эксперт (обычно руководитель тестирования) снабжает оценки $\beta_1, \beta_2, \dots, \beta_k$ для всех модулей данного ПП весами. Эти веса $\alpha_1, \alpha_2, \dots, \alpha_k$ даются из промежутка $[0, 1]$ и отражают сложность организации тестирования по критерию H_i для данного модуля. Рекомендуется назначать веса в виде небольшого дискретного набора, например, $[0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1]$. Причиной этого является сложность в выборе, например, между весами 0.3 или 0.4, в то время как интервал 0.25 достаточно удобен для экспертного назначения.

Выбор весов тестировщиком зависит от деления всех критериев H_1, H_2, \dots, H_k на группы H^I, H^{II}, H^{III} . В группе H^I выбор делается по принципу – удобно ли будет с точки зрения данного критерия использовать автоматизированное тестирование (оценка 1); достаточно удобно (оценка $\frac{3}{4}$); есть сомнения, что удобно (оценка $\frac{1}{2}$); есть сильные сомнения, что удобно (оценка $\frac{1}{4}$); совершенно неудобно (оценка 0). В группе H^{II} принцип обратный - удобно ли будет с точки зрения данного критерия использовать ручное тестирование (оценка 0); достаточно удобно (оценка $\frac{1}{4}$); есть сомнения, что удобно (оценка $\frac{1}{2}$); есть сильные сомнения, что удобно (оценка $\frac{3}{4}$); совершенно неудобно (оценка 1). В группе H^{III} принцип такой же, как и в первой группе.

Ячейка 3. При наличии неясностей с назначением оценки β_i для конкретного вопроса H_i тестировщик обращается в группу программистов, и, после обсуждений, оценка β_i может быть изменена.

Ячейка 4. Оценки $\beta_1, \beta_2, \dots, \beta_k$ напрямую неудобны для дальнейшей обработки, поскольку оценка $\beta_j = 0$ при её умножении на вес всегда даёт нуль, и, тем самым, вес как бы не учитывается. Предлагается отделить оценку от нуля (а, заодно, и от единицы) заменой чисел 0 и 1 на числа $\frac{1}{4}$

и на $\frac{3}{4}$. То есть, если величина $\beta_i = 0$, $i=1, \dots, k$, то она заменяется на $b_i = \frac{1}{4}$. Если же величина $\beta_i = 1$, то она заменяется на $b_i = \frac{3}{4}$. Таким образом, в целом, имеем переход

$$\beta_1, \beta_2, \dots, \beta_k \rightarrow b_1, b_2, \dots, b_k.$$

Ячейка 5. Веса $\alpha_1, \alpha_2, \dots, \alpha_k$ для единообразия дальнейшей обработки по групповым правилам заменяются на веса a_1, a_2, \dots, a_k . Эти правила зависят от группы критериев. А именно:

Для первой группы критериев $H^I = (H_1, H_2, \dots, H_{k1})$, ориентированной на выбор автоматизированного тестирования, веса $\alpha_1, \alpha_2, \dots, \alpha_{k1}$ необходимо только нормировать по группе – разделить на $k1$:

$$a_i = \alpha_i / k1 \text{ для } i=1, \dots, k1.$$

Для второй группы критериев $H^{II} = (H_{k1+1}, H_{k1+2}, \dots, H_{k2})$, ориентированной на выбор ручного тестирования, необходимо веса $\alpha_{k1+1}, \alpha_{k1+2}, \dots, \alpha_{k2}$ инвертировать: $\alpha_i \rightarrow (1 - \alpha_i)$ и поменять в них знак. В итоге имеем

$$a_i = (\alpha_i - 1) / k2 \text{ для } i=k1+1, \dots, k2$$

Для третьей, нейтральной группы критериев $H^{III} = (H_{k2+1}, H_{k2+2}, \dots, H_k)$, необходимо центрирование оценок $\alpha_{k2+1}, \alpha_{k2+2}, \dots, \alpha_k$. То есть нейтральной (нулевой) должна стать оценка $\frac{1}{2}$. С учётом количества критериев в этой группе $(k - k1 - k2)$ это достигается переходом

$$a_i = (\alpha_i - \frac{1}{2}) / (k - k1 - k2) \text{ для } i= k2+1, \dots, k.$$

В итоге замен во всех трёх группах критериев имеем переход

$$\alpha_1, \alpha_2, \dots, \alpha_k \rightarrow a_1, a_2, \dots, a_k.$$

Ячейка 6. Подготовительная работа, проведённая в ячейках 4 и 5, позволяет единообразно вычислять групповую свёртку по формуле

$$P = a_1b_1 + a_2b_2 + \dots + a_kb_k.$$

Указанная величина всегда находится в пределах $-1 \leq P \leq 1$. Математическое ожидание такой свёртки на множестве всех возможных исходных оценок $\beta_1, \beta_2, \dots, \beta_k$ и весов $\alpha_1, \alpha_2, \dots, \alpha_k$ есть 0. Это позволяет говорить о том, что положительное значение оценки P говорит о предпочтительности автоматизированного тестирования, а отрицательное – о предпочтительности ручного тестирования.

Ячейка 7. Назначаются величины P_1 и P_2 , определяющие вид тестирования. По построенной выше части алгоритма можно было бы обойтись без ограничительных величин P_1 и P_2 и предложить использовать автоматизированное тестирования при $P \geq 0$ и ручное при $P < 0$. Но для большей аргументированности решений предлагается отодвинуть границы от 0 и ввести величины $P_1 \geq 0$ и $P_2 \leq 0$. Это и приводит к третьему случаю, когда формализованное решение признаётся недостаточно обоснованным. В качестве конкретных значений P_1 и P_2 предлагается взять симметричные значения $P_1 = 0.2$ и $P_2 = -0.2$.

Ячейка 8. Выбор вида тестирования $R1_G$, $R2_G$, или $R3_G$ для конкретного модуля или всего ПП делается по значению функции P .

При $P \geq P_1$, рекомендуется автоматизированный метод тестирования

При $P \leq P_2$, рекомендуется ручной метод тестирования

При $P_2 < P < P_1$, формальным способом выбора пользоваться не рекомендуется, решение принимается экспертно или требует использования дополнительных моделей.

Пример решения поставленной задачи

Пусть в решаемой задаче семь критериев H_1, H_2, \dots, H_7 , четыре первых из них традиционно считаются ведущими к автоматизированному тестированию ($R1_G$), , два последующих – к необходимости ручного тестирования ($R2_G$), и последний нейтрален ($R3_G$). В группе программистов для данного модуля по критериям были даны оценки

$$(\beta_1, \beta_2, \dots, \beta_7) = (1, 0, 1, 1, 0, 1, 0).$$

Напомним, это означает, что на вопросы №2, №5 и №7 программисты ответили «нет». Пусть представитель тестировщиков усомнился в оценке №5, и после обсуждения «0» был изменён на «1». В итоге, с учётом действий в ячейках 3 и 4, массив \mathbf{b} имеет вид

$$(\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_7) = (3/4, 1/4, 3/4, 3/4, 3/4, 3/4, 1/4).$$

Пусть веса, данные тестировщиками, имеют вид

$$(\alpha_1, \alpha_2, \dots, \alpha_7) = (1/2, 0, 3/4, 1/2, 1/2, 1, 0).$$

Их обработка в ячейке 5 даст ненормированный массив

$$(1/2, 0, 3/4, 1/2, -1/2, 0, -1/2).$$

Нормированный массив \mathbf{a} будет иметь вид

$$(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_7) = (1/8, 0, 3/16, 1/8, -1/4, 0, -1/2)$$

После вычислений имеем $\mathbf{P} = \mathbf{a}_1\mathbf{b}_1 + \mathbf{a}_2\mathbf{b}_2 + \dots + \mathbf{a}_7\mathbf{b}_7 = 1/64 \approx 0.016$.

Для рекомендованных $P_1 = 0.2$, $P_2 = -0.2$ имеем указание на альтернативу $R3_G$, то есть, рекомендацию, несмотря на то, что величина $P \approx 0.016$ положительна, не принимать формальное решение о выборе для данного модуля автоматизированного вида тестирования.

Связи с классической задачей многокритериальной оптимизации

Поставленная задача о формальном нахождении способа тестирования является неклассической задачей дискретной оптимизации [7]. Главным отличием является узость множества альтернатив – их всего три. Это использование при тестировании данного модуля автоматизированного метода решения задачи ($R1_G$), ручного метода ($R2_G$) или рассмотрение более сложной ситуации, включающей более сложное рассмотрение, в том числе, комбинирование методов ($R3_G$). Напомним, что в классической задаче такое множество обычно бесконечно, в т.ч. часто непрерывно, и главной проблемой является перемещение к точкам допустимого множества (альтернативам), дающим более выгодное значение обобщённого критерия. Здесь же решение задачи оптимизации состоит в вычислении некоторой величины (у нас – групповой свёртка $P = P^I + P^{II} + P^{III}$), значение которой указывает на наилучшую альтернативу. Альтернативы $R1_G$, $R2_G$, $R3_G$ характеризуются величинами P^I , P^{II} , P^{III} , а окончательный выбор достигается их суммированием.

Известное и весьма полезное в классических задачах множество Парето здесь состоит из всех рассматриваемых альтернатив: $R1_G$, $R2_G$, $R3_G$.

В классической задаче многокритериальной оптимизации критерии (у нас элементы множества M) задаются на множестве альтернатив. Здесь же исходными являются множества оценок $\beta_1, \beta_2, \dots, \beta_k$ и весов $\alpha_1, \alpha_2, \dots, \alpha_k$, которые характеризуют элементы множества M . То есть в задачу вводится дополнительный нижний уровень информации. Зато здесь не требуются стандартные для классической задачи методы сокращения количества альтернатив (до одной или нескольких), поскольку их всего три: $R1_G$, $R2_G$,

R3_G . Требуется лишь процедура их оценки, которая и предложена выше. Это и позволяет назвать данную задачу вырожденной.

Выводы

Использование различных вариантов методов свёртки – наиболее распространённые приём решения многокритериальных задач [8]. В данной задаче метод обосновывается тем, что свёртка объединяет мнение программистов и тестировщиков, а также тем, что позволяет удобно вводить в задачу экспертное мнение. Напомним, что входные величины в задаче – это искусственные баллы, а не объективные числовые оценки.

Вопросы межпрофессионального взаимодействия IT-специалистов являются важными в наши дни [9], так как основу экономики развитых стран составляют многофункциональные проекты [10], поэтому предложенный авторами алгоритм взаимодействия разработчиков ПП и тестировщиков ПП актуален и может быть широко применим.

Литература

1. Галимова Е. Ю. Процесс тестирования программного обеспечения как неотъемлемая часть его жизненного цикла, 61-я Международная молодежная научно-техническая конференция «Молодежь. Наука. Инновации», сборник статей, ноябрь 2013, Владивосток, с. 70-72.

2. T'Kindt Vincent, Multicriteria Scheduling Problems // Multiple Criteria Optimization, Springer Science & Business Media, 2002, pp 445-491.

3. Орловский Н. М. Решение задачи однокритериальной оптимизации процесса планирования действий экипажа российского сегмента международной космической станции на основе генетического алгоритма // Инженерный вестник Дона, 2013, №3 URL: ivdon.ru/ru/magazine/archive/n3y2013/1776

4. Галимова Е. Ю. Преимущества ручного подхода к тестированию программного обеспечения, III Международный научный форум "Наука в исследованиях молодых", сборник статей, май 2013, Новосибирск, с. 39 – 42.

5. Мухлисуллина Д. Т., Моор Д. А. Анализ эффективности различных свёрток критериев оптимальности в задаче многокритериальной оптимизации // «Наука и образование», 2010, №4, с. 46 URL: technomag.edu.ru/doc/141623.html

6. Галимова Е. Ю., Коваленко А. Н. «Применение алгоритма многокритериальной оптимизации при выборе между ручным и автоматизированным тестированием». Сборник докладов 63-й Международной молодежной научно-технической конференции «Молодежь. Наука. Инновации», Владивосток, 17-20 ноября 2015, том 1. Изд-во Мор. Гос. Ун-т., 2015. – 356 с.

7. Ковалев М. М. Дискретная оптимизация (целочисленное программирование) - М., Едиториал УРСС, 2003, с. 7-8.

8. Ногин В. Д. Линейная свертка критериев в многокритериальной оптимизации, «Искусственный интеллект и принятие решений», № 4, 2014, с. 73-82.

9. Gerald M. Weinberg, Perfect software: And other Illusions about testing, Dorset House Publishing Company, 2008, 200 p.

10. Лежнева М. С. Технология формирования готовности будущих IT-специалистов к межпрофессиональному взаимодействию // Инженерный вестник Дона, 2013, №1 URL: ivdon.ru/ru/magazine/archive/n1y2013/1563

References

1. Galimova E. Ju. 61-ja Mezhdunarodnaja molodezhnaja nauchno-technicheskaja konferencija «Molodezh'. Nauka. Innovacii», sbornik statej, nojabr' 2013, Vladivostok, pp. 70-72.



2. T'Kindt Vincent, Multicriteria Scheduling Problems. Multiple Criteria Optimization, Springer Science & Business Media, 2002, pp. 445-491.
3. Orlovskij N. M. Inženernyj vestnik Dona (Rus), 2013, №3 URL: ivdon.ru/ru/magazine/archive/n3y2013/1776
4. Galimova E. Ju. III Mezhdunarodnyj nauchnyj forum "Nauka v issledovaniyah molodyh", sbornik statej, maj 2013, Novosibirsk, pp. 39 – 42.
5. Muhlisullina D. T., Moor D. A. «Nauka i obrazovanie», 2010, №4, p. 46 URL: technomag.edu.ru/doc/141623.html
6. Galimova E. Ju., Kovalenko A. N. «Primenenie algoritma mnogokriterial'noj optimizacii pri vybore mezhdu ruchnym i avtomatizirovannym testirovaniem» [Application of multi-criteria optimization in selecting between manual and automated testing]. Sbornik dokladov 63-j Mezhdunarodnoj molodezhnoj nauchno-tehnicheskoy konferencii “Molodezh'. Nauka. Innovacii”, Vladivostok, 17-20 nojabrja 2015, tom 1. Izd-vo Mor. Gos. Un-t., 2015. 356 p.
7. Kovalev M. M. Diskretnaja optimizacija (celochislennoe programmirovanie) [Discrete optimization (integer programming)] - M., Editorial URSS, 2003, pp. 7-8.
8. Nogin V. D. «Iskusstvennyj intellekt i prinjatие reshenij», № 4, 2014, pp. 73-82.
9. Gerald M. Weinberg, Perfect software: And other Illusions about testing, Dorset House Publishing Company, 2008, 200 p.
10. Lezhneva M. S. Inženernyj vestnik Dona (Rus), 2013, №1 URL: ivdon.ru/ru/magazine/archive/n1y2013/1563