

# ПРИМЕНЕНИЕ МЕТРИК СЛОЖНОСТИ В ПРОЦЕССЕ ТЕСТИРОВАНИЯ ПРОГРАММНЫХ СИСТЕМ

## APPLICATION OF DIFFICULTY METRICS IN THE PROCESS OF TESTING SOFTWARE SYSTEMS

**E. Galimova  
S. Belov**

*Summary.* The article discusses the main metrics for assessing the qualitative characteristics of a software system, describes the mechanism for forming a metric basis of complexity. The authors have developed an approach to modeling the process of software development and testing using affine subspaces and a metric basis. The dimension of the intersection of affine subspaces acts as the resulting evaluation criterion.

*Keywords:* complexity metrics, quality of software systems, software testing, affine shell, affine subspace.

**Галимова Екатерина Юрьевна**

Ассистент, Санкт-Петербургский государственный университет промышленных технологий и дизайна  
galim81@mail.ru

**Белов Сергей Валерьевич**

К.т.н., доцент, Астраханский государственный технический университет  
ssbelov@yandex.ru

*Аннотация.* В статье рассмотрены основные метрики оценки качественных характеристик программной системы, описан механизм формирования метрического базиса сложности. Авторами разработан подход к моделированию процесса разработки и тестирования программного обеспечения с помощью аффинных подпространств и метрического базиса. В роли результирующего оценочного критерия выступает размерность пересечения аффинных подпространств.

*Ключевые слова:* метрики сложности, качество программных систем, тестирование программного обеспечения, аффинная оболочка, аффинное подпространство.

### Введение

**П**отребность различных отраслей современного общества в сложных программных системах, растет быстрее, чем технологии их разработки и тестирования [1]. Происходит постоянный рост количества программных модулей, входящих в состав одной программной системы. С увеличением сложности увеличиваются и затраты на тестирование. Специфика процесса тестирования зависит, в том числе, и от предполагаемой сферы применения исследуемой ПС [2–5]. Тенденции развития современной индустрии информационных технологий показывают, что с каждой версией программного продукта, разрабатываемого одной и той же организацией, его сложность возрастает. Следовательно, требуется постоянное усовершенствование процессов разработки и тестирования. Предлагается

подход для оценки динамики изменения сложности создаваемых программных систем, основанный на метрическом базисе.

### Формирование метрического базиса программной системы

В процессе анализа предметной области выявлен набор метрик, наиболее полно характеризующий сложность программной системы [6]. Из них сформирован метрический базис сложности  $M = \{m_1, m_2, \dots, m_n\}$ , где  $m_i$  — величина  $i$ -ой метрики (таблица 1).

Исследуемая программная система  $P$  будет иметь набор координат  $(p_1, p_2, \dots, p_n)$  относительно метрического базиса сложности. В зависимости от целей проводимого исследования, можно применять подмножество

Таблица 1. Метрический базис сложности программной системы

Обозначение в базисе	Название	Формула
$m_1$	Цикломатическая сложность программы	$M = E - N + 2P$
$m_2$	Высота дерева наследования	$DIT = \max(path)$
$m_3$	Количество детей класса	NOC = number of children
$m_4$	Количество откликов на класс	$RFC = N_m + K$
$m_5$	Нехватка связности в методах	$LCOM2 = 1 - \frac{sum(MA)}{M \cdot A}$
$m_6$	Средний размер класса	$CS = \sum_{i=1}^N (C_i + S_i)$
$m_7$	Количество операций, переопределенных подклассом	NOO = Overridden Operations
$m_8$	Количество операций, добавленных подклассом	NOA = Added Operations
$m_9$	Индекс специализации	$SI = \frac{NOO \cdot U}{M_{total}}$
$m_{10}$	Метрика Джилба	$JM = \frac{CL}{n}$
$m_{11}$	Мера Вудворта	WM = number of intersections of arcs
$m_{12}$	Метрика Пивоварского	$N(G) = M^* + \sum_{i=1}^n P_i$
$m_{13}$	Количество методов на класс	$N_{obj} = \frac{TN_{meth}}{TN_{class}}$
$m_{14}$	Средняя сложность метода	$AMC = \frac{\sum_{i=1}^n Q_i}{N_{total}}$
$m_{15}$	Среднее количество сообщений на операцию	$OS_{avg} = \frac{\sum_{i=1}^n N_{of} \cdot M_i}{N_{of} \cdot OP}$
$m_{16}$	Среднее количество параметров на операцию	$NP_{avg} = \frac{\sum_{i=1}^n N_{par_i}}{N_{op}}$
$m_{17}$	Уровень комментированности	$F = \frac{N_{com}}{N_{str}}$
$m_{18}$	Модифицированная цикломатическая сложность	$M^* = E^* - N^* + 2 \cdot P^*$
$m_{19}$	Метрика Чепина	$Q = P + 2 \cdot M + 3 \cdot C + 0,5 \cdot T$
$m_{20}$	Уровень программной реализации Холстеда	$L = \frac{V^*}{V}$
$m_{21}$	Метрика связности объектов	$D_{of} C = \frac{N_{Fan-In}}{N_{all}}$

$D \subset M$ , содержащее необходимую для решения текущей задачи выборку метрик.

### Особенности аффинных пространств

Пусть  $V$  — векторное пространство (его элементы обозначим строчными латинскими буквами). Пусть  $T$  — множество элементов, называемых точками (их обозначим прописными латинскими буквами). Точка является неопределяемым понятием, так как не может быть определена через другие объекты. Каждой упорядоченной паре точек  $A, B$  соответствует один и только один вектор  $\vec{r} \subset V$ . При этом один и тот же вектор может соответствовать нескольким парам точек [7]. Связь между точками и векторами реализуется в виде операции откладывания векторов. Выполняются аксиомы Вейля [8]. Тогда совокупность точек и векторов образуют аффинное пространство  $\Upsilon^n$ . Аксиоматической базой аффинной геометрии является множество точек  $\{T\}$  (основное), множество чисел  $\{R\}$  и множество векторов  $\{V\}$  (вспомогательные). Если аффинные пространства имеют одинаковую размерность, они изоморфны. Следовательно, их удобно изучать с использованием арифметической модели  $T(R_n)$ .

Рассмотрим систему линейных неоднородных уравнений:

$$\begin{cases} b_1 r_1 + b_2 r_2 + \dots + b_{1n} r_n = a_1 \\ b_2 r_1 + b_2 r_2 + \dots + b_{2n} r_n = a_2 \\ \dots\dots\dots \\ b_{s1} r_1 + b_{s2} r_2 + \dots + b_{sn} r_n = a_s \end{cases}$$

Пусть ранг матрицы ее коэффициентов  $w$  и  $k = n - w$ . Тогда множество всех точек, принадлежащих  $\Upsilon^n$  и имеющих координаты, удовлетворяющие данной системе уравнений, называется  $k$ -мерным подпространством (или плоскостью). Обозначим его  $\Psi$ . Пусть  $l$ -мерное подпространство  $\Psi'$  определяется следующей системой линейных неоднородных уравнений:

$$\begin{cases} d_{11} r_1 + d_{12} r_2 + \dots + d_{1n} r_n = c_1 \\ d_{21} r_1 + d_{22} r_2 + \dots + d_{2n} r_n = c_2 \\ \dots\dots\dots \\ d_{q1} r_1 + d_{q2} r_2 + \dots + d_{qn} r_n = c_q \end{cases}$$

Для определения пересечения этих подпространств строится система уравнений, объединяющая обе вышеуказанные системы уравнений. Следовательно, пересечение тоже является некоторым подпространством. Если объединенная система несовместна, пересечение будет пустым. Однородная система линейных уравнений определяет подпространство, проходящее через начальную точку аффинного пространства.

### Применение метрического базиса в процессе оценки качества программной системы

Для анализа процесса тестирования в организации смоделируем его при помощи аффинных подпространств. Рассмотрим линейку программных продуктов. Для каждой версии ПС (обозначим ее  $P_i$ ) строится метрический базис. Множество ПС  $\langle P_0, P_1, \dots, P_m \rangle$  с координатами относительно метрического базиса образуют аффинную оболочку или аффинное подпространство, для которого начальной точкой является метрический базис первой версии ПС ( $P_0$ ), а направляющее подпространство  $U$  сформировано векторами  $\overline{P_0 P_1}, \overline{P_0 P_2}, \dots, \overline{P_0 P_n}$ . Составим матрицу:

$$M = \begin{pmatrix} \overline{P_0 P_1} \\ \overline{P_0 P_2} \\ \vdots \\ \overline{P_0 P_n} \end{pmatrix}$$

Пусть полученный  $Rg M = \dim U = k$ . Далее выделим базис. Явное задание аффинной оболочки запишем параметрически в виде:

$$\begin{pmatrix} P_1 \\ P_2 \\ \vdots \\ P_n \end{pmatrix} = \begin{pmatrix} P_{01} \\ P_{02} \\ \vdots \\ P_{0n} \end{pmatrix} + \sum_{i=1}^k s_i \begin{pmatrix} P_{i1}^* \\ P_{i2}^* \\ \vdots \\ P_{in}^* \end{pmatrix}, s_i \in R.$$

Сформируем неявное задание подпространства  $U$  в виде однородной системы линейных алгебраических уравнений. Найдем ранг матрицы системы ( $rang C$ ).

Для двух линеек программных продуктов вычисляется размерность пересечения соответствующих подпространств по формуле Грассмана:

$$\dim(U_m \cap U_{m+1}) = \dim(U_m) + \dim(U_{m+1}) - \dim(U_m + U_{m+1})$$

Если  $U_m$  есть аффинная гиперплоскость и  $\dim(U_{m+1}) = q$ , то  $\dim(U_m \cap U_{m+1}) = q - 1$

Размерность пересечения от 7 и выше свидетельствует о стабильных показателях качества процесса создания программных систем в организации.

Если метрический базис сложности исследуемой ПС превосходит допустимую величину, необходимо учитывать сложность в алгоритме оценки стоимости тестирования ПС, который рассмотрен в работе [9].

## Заключение

В настоящее время процесс тестирования является важным этапом жизненного цикла программных систем. Важно соблюсти баланс между сложностью

разрабатываемого программного обеспечения и его качеством. В дальнейшем, при принятии решения о необходимости усовершенствования процесса тестирования, рекомендуется применять методику, изложенную в работе [10].

## ЛИТЕРАТУРА

1. Lyu M.R. Handbook on Software Reliability Engineering. McGraw-Hill, USA and IEEE Computer Society Press, Los Alamitos, California, USA, 1996.
2. Galimova E. Features of software testing in the development of geographic information systems // Proceedings of XVIII Scientific Forum "Ural Mining Decade" <https://doi.org/10.1051/e3sconf/202017702008>
3. Galimova E. Discussion of cloud CAD software testing approach // IOP Conf. Series: Material Science and Engineering, Vol. 919 <https://doi.org/10.1088/1757-899X/919/5/052034>
4. Галимова Е.Ю. Исследование подходов к тестированию программных продуктов, применяемых в кредитно-финансовой сфере // Управление качеством в образовании и промышленности: сборник статей Всероссийской научно-технической конференции (21–22 мая 2020 г., г. Севастополь). — Севастополь: ФГАОУ ВО «Севастопольский государственный университет», 2020. — С. 505–509.
5. Галимова Е.Ю. Особенности тестирования экспертных систем, разрабатываемых для нужд спорта // Материалы IV Всероссийской научно-практической конференции «День спортивной информатики» 4–5 декабря 2020 года. — Москва, 2021. — С. 116–118.
6. Корниенко А.А., Диасамидзе С.В. Подход к структурированию метрик сложности для выявления дефектов безопасности программного обеспечения // Известия ПГУПС, 2016, № 3, с. 421–429.
7. Головина Л.И. Линейная алгебра и некоторые ее приложения. — 4-е изд., испр. — М.: Наука, Главная редакция физико-математической литературы, 1985–392 с.
8. Игнатьев И.Г. Аналитическая геометрия. Часть II. Аффинные и евклидовы пространства. Учебное пособие. II семестр. — Казань: ТГГПУ, 2013, — 188 с.
9. Галимова Е.Ю. Метод оценки стоимости развертывания процесса тестирования программных систем // Modern science. Москва, 2020. — № 8–2. — С. 279–282.
10. Галимова Е.Ю., Белов С.В. Модель информационно-советующей системы поддержки принятия решения при выборе способа тестирования программного обеспечения // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. — 2020. — № 3. — С. 52–60.

© Галимова Екатерина Юрьевна (galim81@mail.ru), Белов Сергей Валерьевич (ssbelov@yandex.ru).

Журнал «Современная наука: актуальные проблемы теории и практики»



Санкт-Петербургский государственный университет промышленных технологий и дизайна