

## МЕТОДЫ ГЕНЕРИРОВАНИЯ КВАЗИСЛУЧАЙНЫХ ТЕСТОВЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Самкевич Ю. А.

Ярмолик В. Н. – д.т.н., профессор

Аппаратная и программная сложность современных вычислительных систем порождает крайне актуальную проблему исчерпывающего тестирования. Вероятность наличия множества неисправностей и ошибок, как в аппаратной, так и в программной частях системы определяет необходимость разработки наиболее эффективных и производительных методов и алгоритмов тестирования. Одним из таких методов является метод квазислучайного тестирования, использующий в своей основе квазислучайные тестовые последовательности.

Основные недостатки вероятностного тестирования заключаются в том, что при тестировании данным методом не используется доступная информация о предыдущих тестовых наборах, что приводит к неравномерному покрытию тестами модели тестирования и как результат увеличивается время тестирования и длина тестовых последовательностей. Так же случайное тестирование является неэффективным, если плотность обнаруженных неисправностей оказывается низкой [1]. Поэтому в качестве альтернативного решения для тестирования вычислительных систем предлагается использование так называемого квазислучайного тестирования. Данный метод тестирования основан на идее антислучайного тестирования и предполагает в качестве обобщенных тестовых воздействий использование квазислучайных последовательностей.

Квазислучайной последовательностью чисел, называется последовательность, любой последовательный участок которой хорошо распределен (более равномерно по сравнению с псевдослучайной последовательностью). Квазислучайные последовательности получили широкое применение в приложениях математики и теории чисел. В частности в задачах численных методов квазислучайные последовательности применяются в реализации алгоритма Монте-Карло, для вычисления несобственных интегралов. Такие алгоритмы получили название квази-Монте-Карло, их применение позволяет достичь меньших вычислительных погрешностей и более быстрой сходимости [1]. В русскоязычной литературе такие последовательности получили название ЛПТ-последовательностей. В англоязычной литературе их называют последовательностями с малым дисбалансом (low-discrepancy sequence) [2].

Квазислучайные последовательности имеют хорошее распределение, что сказывается на эффективности покрытия области входных значений вычислительных систем. Квазислучайное тестирование, как и при использовании его в методе квази-Монте-Карло, позволяет достичь большей полноты покрытия при меньших длинах тестовых последовательностей. В качестве методов генерирования квазислучайных тестовых последовательностей предлагается использование методов Корпута, Халтона, а так же метода Соболя и его модификаций.

Последовательности Соболя, основанные на использовании двоичной системы координат, являются широко востребованными для современных приложений в основном из-за простоты реализации на ЭВМ алгоритмов их генерирования. Значение координат  $n$ -го элемента последовательности Соболя вычисляется как поразрядная сумма по модулю два до  $w = \log_2 n$  операндов, в зависимости от количества ненулевых компонентов двоичного представления  $b_{w-1}(n)b_{w-2}(n) \dots b_1(n)b_0(n)$  числа  $n$  [3]. При использовании экономичного способа Антонова-Салеева, основанного на представлении числа  $n$  в виде двоичного кода Грея, количество операндов может быть снижено до одного. Таким образом  $n$ -ый элемент последовательности Соболя рассчитывается по формуле:

$$A(n) = A(n-1) \oplus V_i; n = \overline{0, 2^w - 1}, i = \overline{0, w-1},$$

где  $V_i$  – так называемые порождающие числа.

В общем случае, модифицированные порождающие числа  $V_i$  можно представить в виде нижней треугольной матрицы с единичной главной диагональю [3]:

$$V = \begin{bmatrix} b_{w-1}(0) & b_{w-2}(0) & b_{w-3}(0) & \dots & b_0(0) \\ b_{w-1}(1) & b_{w-2}(1) & b_{w-3}(1) & \dots & b_0(1) \\ b_{w-1}(2) & b_{w-2}(2) & b_{w-3}(2) & \dots & b_0(2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{w-1}(m-1) & b_{w-2}(m-1) & b_{w-3}(m-1) & \dots & b_0(m-1) \end{bmatrix}$$

Для доказательства эффективности применения последовательностей квазислучайных чисел в приложениях тестирования был проведен ряд экспериментов. Задачей экспериментов было показать, что для выявления неисправностей в вычислительной системе, длина квазислучайной последовательности заметно меньше чем псевдослучайной. Эксперименты проводились над разными типами моделей. В качестве метода генерирования квазислучайных последовательностей был использован эффективный метод Антонова-Салеева, для генерирования последовательности Соболя. Результаты экспериментов представлены в таблице 1.

№	Количество экспериментов	Тип тестовой модели	Средняя длина квазислучайной последовательности	Средняя длина псевдослучайной последовательности
1	100	Точечная	309	323
2	100	Точечная	437	500
3	100	Точечная	424	420
4	100	Узкополосная	35857	35789
5	100	Узкополосная	253456	252987
6	100	Узкополосная	9145	9378
7	100	Блочная	15167	16754
8	100	Блочная	16564	18143
9	100	Блочная	16572	17143

Таблица 1 – Сравнение квазислучайных и псевдослучайных последовательностей в зависимости от типа тестовой модели

Из приведенных выше данных видно, что длина квазислучайной тестовой последовательности на 7-10% меньше чем псевдослучайной. Наибольшую эффективность квазислучайные последовательности показали при тестировании блочной тестовой модели.

Список использованных источников:

1. Ярмолик, С.В. Квазислучайное тестирование вычислительных систем. / С.В. Ярмолик, В.Н. Ярмолик // Информатика. – 2013. - № 3 – С. 65-81.
2. Иванюк, А.А. Проектирование встраиваемых цифровых устройств и систем / А.А. Иванюк. – Минск : Бестпринт, 2012. – 338 с.
3. Ярмолик, В. Н. Адресные последовательности для многократного тестирования ОЗУ / В. Н. Ярмолик, С. В. Ярмолик // Информатика. - 2014. - N 2. - С. 124-136.

## ИНСТРУМЕНТАЛЬНОЕ ПРОГРАММНОЕ СРЕДСТВО ДЛЯ АВТОМАТИЗАЦИИ РАЗРАБОТКИ МОДУЛЕЙ РАСШИРЕНИЯ ПРИЛОЖЕНИЙ MICROSOFT OFFICE

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Коваль Д. А.*

*Сурков К. А. – ассистент*

Задача по автоматизации разработки модулей расширения приложений Microsoft Office (Excel, Word, PowerPoint) является актуальной в сфере разработки программного обеспечения, поскольку на сегодняшний день на рынке существует лишь одно по-настоящему мощное средство – Visual Studio Tools For Office. Целью разработки является программное средство CommonOffice, представляющее собой набор библиотек программирования, позволяющих автоматизировать процесс разработки модулей расширения приложений Microsoft Office на языках программирования C++ и C#, а так же интегрировать разработанные модули в один комбинированный модуль.

Принципы, положенные в основу поиска технического решения и проектирования программного средства, сформировались в условиях реальной необходимости такого программного средства, а именно: во время разработки коллективами программистов системы модулей расширения, были обнаружены следующие проблемы:

- децентрализованность прикладных интерфейсов программирования
- сложность во взаимодействии модулей
- дублирование кода
- дублирование данных
- проблемы с производительностью и потребляемой хостом оперативной памятью.

Приложения Microsoft Office (Excel, Word, PowerPoint) предоставляют возможности для своего расширения посредством COM интерфейсов. Таким образом, с точки зрения хост приложений, разрабатываемое программное средство представляет собой COM плагин. Для интеграции в Office приложения в качестве COM плагина, необходимо разработать COM компонент реализующий IDTExtensibility2 COM интерфейс, экспортируемый Office приложениями, и зарегистрировать этот компонент в качестве COM плагина