ИНФОРМАТИКА

2011 январь-март № 1

УДК 004.33.054

С.В. Ярмолик, В.Н. Ярмолик

УПРАВЛЯЕМОЕ СЛУЧАЙНОЕ ТЕСТИРОВАНИЕ

Анализируются управляемые случайные тесты и методы их генерирования. Показывается общность процедур генерирования тестовых векторов управляемых случайных тестов, использующих жадный оптимизационный алгоритм и метрики расстояния между тестовыми наборами. Предлагается метод построения оптимальных управляемых случайных тестов, характеризующихся максимальной полнотой покрытия в сравнении со случайными и управляемыми случайными тестами в силу максимального отличия тестовых наборов. Оптимальные управляемые случайные тесты характеризуются минимальной вычислительной сложностью их генерирования.

Введение

Одной из самых распространенных технологий применения случайного тестирования (*Random Testing*) цифровых устройств и программного обеспечения является метод черного ящика (*Black Box*) [1–4]. В этом случае структура и функциональность объекта тестирования принимаются неизвестными и никак не учитываются при генерировании тестовых наборов. Более того, случайное тестирование не использует информацию, которая доступна в процессе генерирования очередного тестового набора. Эта информация может быть получена из предыдущих тестовых наборов и использована при генерировании очередного тестового воздействия [5]. Развитием случайного тестирования является новый подход, называемый антислучайным (*Antirandom*) тестированием, который основан на том, что каждый последующий тестовый набор формируется с использованием некоторой характеристики либо нескольких характеристик, получаемых на основании предыдущих тестовых наборов [5−7].

Основная идея антислучайного тестирования заключается в том, что в целях достижения более высокого покрытия неисправностей (ошибок), обнаруживаемых тестом, минимизации его сложности и повышения эффективности случайного теста необходимо целенаправленно выбирать очередной тестовый набор в зависимости от ранее сгенерированных наборов. Очевидным примером повышения эффективности случайного теста является простейшее исключение повторяющихся случайных наборов, которые возможны при реализации классического случайного тестирования. В общем случае критерием выбора очередного случайного тестового набора является нахождение максимально отличного (максимально удаленного) тестового набора от ранее сгенерированных наборов. Для количественной оценки отличия (расстояния) текущего тестового набора от предыдущих наборов были предложены методы, основанные на применении расстояния Хэмминга и декартова расстояния [5, 6]. Новый тестовый набор, согласно указанным методам, выбирался таким образом, чтобы метрики различия принимали максимальное значение [5−7]. Этот подход оказался более эффективным по сравнению со случайным тестированием [5, 7]. К сожалению, основным недостатком антислучайного тестирования является его большая вычислительная сложность. Реализация антислучайного тестирования требует перечисления всевозможных входных тестовых воздействий и вычисления расстояния для каждого потенциального кандидата в очередные тестовые наборы [5].

Модификации случайного тестирования, такие, как быстрое антислучайное тестирование (*Fast Antirandom* (*FAR*)) [8], адаптивное случайное тестирование (*Adaptive Random Testing*) [9–11], эволюционное случайное тестирование (*Evolutionary Random Testing*) [12], эффективное случайное тестирование (*Good Random Testing*) [13], ограниченное случайное тестирование (*Restricted Random Testing*) [14, 15], зеркальное случайное тестирование (*Mirror Random Testing*) [16], упорядоченное случайное тестирование (*Orderly Random Testing*) [17] и др., основаны на вычислении характеристик для предыдущих тестовых наборов и также характеризуются большой вычислительной сложностью.

Целью данной работы являются анализ известных методов генерирования управляемых случайных тестов и поиск эффективного метода с точки зрения вычислительной сложности генерирования тестовых наборов.

1. Анализ управляемых случайных тестов

Под управляемыми случайными тестами (*Controlled Random Testing*) в дальнейшем будем понимать случайную тестовую последовательность, в которой очередной тестовый набор формируется с учетом ранее сгенерированных предыдущих наборов. Ключевой особенностью управляемого генерирования случайных тестовых наборов является информация, которая извлекается в виде некоторых характеристик (метрик) из ранее сгенерированных тестовых наборов и используется для генерирования очередного тестового набора [5, 8, 17, 18]. Для всех методов управляемого случайного тестирования, используемого для тестового диагностирования цифровых устройств и программных приложений с N входами и пространством входных наборов, состоящим из 2^N двоичных наборов (векторов), приведем следующие определения [5].

Определение 1. Тест (T) представляет собой множество из $q < 2^N$ тестовых наборов $\{T_0, T_1, T_2, \ldots, T_{q-1}\}$, сгенерированных для обнаружения неисправностей объекта тестирования, где $T_i = t_{i,N-1}, t_{i,N-2}, \ldots, t_{i,2}, t_{i,1}, t_{i,0}$ и $t_{i,l} \in \{0, 1\}$, а N является размером набора в битах.

Определение 2. Управляемым случайным тестом $CRT = \{T_0, T_1, T_2, \ldots, T_{q-1}\}$ является тест T, состоящий из сгенерированных случайным образом тестовых наборов T_i, $i \in \{0, 1, 2, ..., q-1\}$, таких, что T_i удовлетворяет некоторому критерию либо критериям, полученным на основании предыдущих наборов $\{T_0, T_1, T_2, \ldots, T_{i-1}\}$.

1.1. Антислучайные тесты

Одним из первых подходов к управляемому генерированию случайных тестовых наборов является антислучайное тестирование, приведенное в [5]. В предложенном подходе использован тот факт, что если очередной тестовый набор будет сформирован в зависимости от ранее сгенерированных наборов, это позволит достичь большей эффективности теста. Подобная зависимость существенно изменяет структуру вероятностного теста, свойства которого весьма далеки от свойств случайных последовательностей. Поэтому в качестве одного из первых названий управляемых случайных тестов был использован термин «антислучайный тест» (АТ), который соответствует определению 2 [5]. Для того чтобы очередной тестовый набор T_i являлся максимально отличным от предыдущих наборов $T_0, T_1, T_2, \ldots, T_{i-1}$, в качестве критериев использовались расстояние Хемминга (*Hamming Distance*) и декартово расстояние (*Cortesian Distance*) [5]. Данные характеристики определялись для двоичных тестовых наборов T_i и T_j, где расстояние Хэмминга $HD(T_i, T_j)$ (HD) вычислялось как вес $w(T_i \oplus T_j)$ вектора $T_i \oplus T_j$ согласно соотношению

$$HD(T_i, T_j) = w(T_i \oplus T_j) = \sum_{l=0}^{N-1} (t_{i,l} \oplus t_{j,l}) \,. \qquad (1)$$

Декартово расстояние $CD(T_i, T_j)$ (CD) определялось в соответствии с выражением

$$CD(T_i, T_j) = \sqrt{\sum_{l=0}^{N-1} (t_{i,l} - t_{j,l})^2} = \sqrt{\sum_{l=0}^{N-1} \left| t_{i,l} - t_{j,l} \right|} = \sqrt{\sum_{l=0}^{N-1} (t_{i,l} \oplus t_{j,l})} = \sqrt{HD(T_i, T_j)}. \qquad (2)$$

Пример 1. Для двух двоичных наборов $T_\mathrm{i} = 000000$ и $T_\mathrm{j} = 101010$ получим $HD(T_i, T_j) = 3$ и $CD(T_i, T_j) = \sqrt{3} = 1{,}732$.

Очередной тестовый набор T_i формируется таким образом, чтобы он был максимально отличным от всех ранее сгенерированных наборов. В данном случае принимается гипотеза, что для двух тестовых наборов, имеющих минимальное расстояние (1) или (2), количество неисправностей (ошибок), обнаруживаемых вторым набором, будет минимальным и, наоборот, для максимальных значений указанных характеристик обнаруживающая способность второго набора максимальна [5, 6]. Для количества рассматриваемых тестовых наборов более двух при генерировании набора T_i применяются суммарные значения расстояний T_i по отношению к T_0, T_1, T_2, …, T_{i-1} [5]. Тогда для очередного набора T_i суммарное значение расстояний относительно предыдущих наборов T_0, T_1, T_2, …, T_{i-1} вычисляется как

$$THD(T_i) = \sum_{j=0}^{i-1} HD(T_i, T_j), \quad TCD(T_i) = \sum_{j=0}^{i-1} CD(T_i, T_j). \tag{3}$$

Здесь $THD(T_i)$ и $TCD(T_i)$ представляют собой суммарное расстояние Хэмминга (*THD*) и суммарное декартово расстояние (*TCD*) соответственно.

Пример 2. Для антислучайного теста AT = {000, 111, 010, 101} с параметрами $N = 3$ и $q = 4$ значения ранее определенных характеристик приведены в табл. 1.

Таблица 1

Значения $THD(T_i)$ и $TCD(T_i)$ для $T_i = t_{i,2}t_{i,1}t_{i,0}$

i	T_i	$t_{i,2}$	$t_{i,1}$	$t_{i,0}$	$THD(T_i)$	$TCD(T_i)$
0	T_0	0	0	0	–	–
1	T_1	1	1	1	3	1,7320
2	T_2	0	1	0	3	2,4142
3	T_3	1	0	1	6	4,1460

Различают тесты с максимальным суммарным расстоянием Хэмминга (*Maximal Hamming Distance Antirandom Test* (*MHDAT*)) и максимальным декартовым расстоянием (*Maximal Cartesian Distance Antirandom Test* (*MCDAT*)) [5−7]. В приведенном примере AT = {000, 111, 010, 101} является одновременно *MHDAT* и *MCDAT*.

В общем случае для *MHDAT* и *MCDAT* выполняются следующие свойства:

1. Результатом произвольной перестановки бит $t_{i,j}$ одновременно во всех наборах теста *MHDAT* (*MCDAT*) является тест *MHDAT* (*MCDAT*) [5, 6].

2. Результатом инвертирования всех бит наборов теста *MHDAT* (*MCDAT*) является тест *MHDAT* (*MCDAT*) [5, 6].

3. Любой *MHDAT* (*MCDAT*) всегда содержит инверсные наборы, такие, что за набором T_{2k} всегда будет следовать набор $T_{2k+1} = \overline{T}_{2k}$ для k = 0, 1, 2, … [5, 6].

Приведенные свойства позволяют уменьшать сложность генерирования *MHDAT* (*MCDAT*), однако в общем случае, к сожалению, не исключают процедуры перечисления пространства возможных тестовых наборов и вычисления расстояний для каждого потенциального кандидата в тестовые наборы [5, 6]. Даже для улучшенных версий методов генерирования антислучайных тестов вычислительная сложность для реальных значений N является чрезвычайно большой.

1.2. Быстрые антислучайные тесты

Генерирование быстрых антислучайных тестов (*Fast Anti-random* (*FAR*)) основано на вычислении так называемых центроидов (*centroid*), определяемых на основании предыдущих наборов T_0, T_1, T_2, …, T_{i-1} для $T_i = t_{i,N-1}, t_{i,N-2}, …, t_{i,2}, t_{i,1}, t_{i,0}$ [8]. Согласно *FAR*-алгоритму тестовые наборы T_0, T_1, T_2, …, T_{i-1} рассматриваются как двухмерные массивы $i \times N$ бит. Для каждого столбца $l \in \{N-1, N-2, N-3, …, 1, 0\}$ значение l-го элемента c_l центроида $C = c_{N-1}, c_{N-2}, …, c_2, c_1, c_0$ определяется как сумма $t_{i,l} \in \{0, 1\}$, деленная на количество i строк. В результате получается центроид, элементы которого c_l принимают значения в интервале от 0 до 1. Далее *FAR*-алгоритм округляет значения c_l до 0, если $c_l < 0{,}5$, или до 1, если $c_l > 0{,}5$. В случае когда $c_l = 0{,}5$, *FAR*-алгоритм случайным образом равновероятно округляет c_l до 0 или 1 [8]. В результате приведенных преобразований формируется двоичный центроид C_b. На финальном этапе генерируется T_i как результат инвертирования C_b, т. е. $T_i = \overline{C}_b$.

Пример 3. В качестве примера возьмем тестовые наборы $T_0 = 000$, $T_1 = 111$ и $T_2 = 010$, тогда элементы центроида примут следующие значения: $C = c_2, c_1, c_0 = 1/3, 2/3, 1/3 = 0{,}333, 0{,}666, 0{,}333$. Соответствующий бинарный центроид $C_b = 010$, а его инверсное значение 101 и будет тестовым набором $T_3 = 101$. Следует отметить идентичность полученного результата T_3 с аналогичным набором, сгенерированным согласно классической интерпретации антислучайного тестирования (см. пример 2).

Применение вероятностного округления для случая, когда $c_l = 0{,}5$, может привести к ошибкам, снижающим эффективность данного алгоритма. Действительно, для предыдущего примера, когда рассматриваются два набора $T_0 = 000$, $T_1 = 111$, центроид $C = c_2, c_1, c_0 = 0{,}5, 0{,}5, 0{,}5$, тогда очередным набором T_2 может быть любой набор, состоящий из трех бит, в том числе $T_0 = 000$ и $T_1 = 111$.

1.3. Упорядоченные случайные тесты

Концепция упорядоченных случайных тестов впервые была предложена в [17] как антислучайных тестов с полумаксимальным расстоянием (*Semi-Maximum Distance Testing Sequences* (*SMDTS*)), для которых каждый тестовый набор теста имеет свое инверсное значение. Например, $T = \{000, 111, 010, 101\}$ есть *SMDTS*, так как $\{000, 111\}$ и $\{010, 101\}$ являются инверсными значениями тестовых наборов. Для *SMDTS* были предложены новые характеристики для оценки расстояния при генерировании очередного тестового набора T_i [18]. Предложенные в [17, 18] $THD(T)$ и $TCD(T)$ представляют собой суммарное расстояние Хэмминга и суммарное декартово расстояние соответственно для теста T, состоящего из q тестовых наборов, и вычисляются как

$$THD(T) = \sum_{i=1}^{q-1} \sum_{j=0}^{i-1} HD(T_i, T_j), \qquad TCD(T) = \sum_{i=1}^{q-1} \sum_{j=0}^{i-1} CD(T_i, T_j). \tag{4}$$

Здесь $HD(T_i, T_j)$ и $CD(T_i, T_j)$ вычисляются для всех $i \neq j$; $j, j \in \{0, 1, \ldots, q-1\}$.

Процедура генерирования *SMDTS*, содержащего $q = 2k$, $k = 0, 1, 2, \ldots$, наборов, заключается в формировании случайных тестовых наборов с четными индексами. Наборы $T_0, T_2, T_4, \ldots$ генерируются как равновероятные двоичные векторы, состоящие из N бит, а тестовые наборы с нечетными индексами $T_1, T_3, T_5, \ldots$ являются инверсными значениями по отношению к $T_0, T_2, T_4, \ldots$. Тогда $THD(T)$ для $SMDTS = T_0, T_1, T_2, \ldots, T_{2k-2}, T_{2k-1}$ равняется k^2N [18]. Для ранее рассмотренного примера 2 $SMDTS = \{000, 111, 010, 101\}$, $k = 2$ и $N = 3$, тогда $THD(T) = k^2N = 2^2 3 = 12$, что соответствует соотношению (4). В силу того что только половина тестовых наборов достигает максимального расстояния Хэмминга, такие последовательности и получили название тестов с полумаксимальным расстоянием [17, 18].

Тесты с полным максимальным расстоянием (*Total Maximum Distance Test Sequences* (*TMDTS*)) являются дальнейшим развитием упорядоченных случайных тестов [17]. Для их генерирования одновременно используются обе метрики *THD* и *TCD* в соответствии со следующей процедурой [17].

Процедура 1. Построение *TMDTS* для $q=2k$.

1. Первый тестовый набор T_0 генерируется как случайный двоичный N-разрядный вектор.

2. Для получения очередного i-го набора $T_1, T_3, T_5, \ldots, T_{2k-1}$ с нечетным индексом используется соотношение $T_{2k+1} = \overline{T}_{2k}$, которое для $k = 0$ принимает вид $T_1 = \overline{T}_0$.

3. Для получения каждого нового набора $T_2, T_4, T_6, \ldots, T_{2k-2}$ с четным индексом из множества возможных кандидатов в тесты выбирается такой, для которого $TCD(T)$ принимает максимальное значение, причем для вычисления $TCD(T)$ используются ранее сгенерированные тестовые наборы.

4. Этапы 2 и 3 повторяются до тех пор, пока все q наборов не будут сгенерированы.

Очевидно, что наиболее трудоемким является этап 3, который требует большого объема вычислений.

Основным недостатком *TMDTS*-тестов, как и всех упорядоченных случайных тестов, является ограничение их длины. Данный недостаток объясняется тем, что чем больше минимальное расстояние $minCD(T_i, T_j)$, используемое как критерий включения T_i в тест, тем меньше будет количество Q кандидатов в тесты. Это следует из предельной оценки Хэмминга (*Hamming* *bound*) [19], которая для $minHD(T_i, T_j) = 2r + 1$ и $min\mathrm{CD}(T_i, T_j) = \sqrt{2r+1}$ может быть представлена как неравенство

$$Q \le \frac{2^N}{\sum_{l=0}^{r} C_N^l}. \tag{5}$$

Например, в случае, когда $N = 15$ и $minHD(T_i, T_j) = 5 = 2{\times}2{+}1$, декартово расстояние удовлетворяет неравенству $CD(T_i, T_j) \ge \sqrt{5}$. Таким образом, существует не более чем

$$Q \le \frac{2^{15}}{\sum_{l=0}^{2} C_{15}^l} = \frac{2^{15}}{C_{15}^0 + C_{15}^1 + C_{15}^2} = \frac{2^{15}}{1+15+105} = 270 = 2^8$$

возможных кандидатов в тесты с декартовым расстоянием $CD(T_i,T_j)$, большим или равным $\sqrt{5}$. Учитывая инверсные значения наборов, максимальная длина *TMDTS* равняется $q = 2^9$. Однако увеличение декартова расстояния, например до значения $\sqrt{7}$, существенно уменьшает длину *TMDTS*. В этом случае количество наборов *TMDTS* равняется $2^6 = 64$.

2. Оптимальные управляемые случайные тесты

Обобщая многообразные модификации управляемых случайных тестов, следует отметить единообразие в процедуре генерирования очередного тестового набора T_i [5–18]. Во всех случаях критерием является максимальное или минимальное значение некоторой характеристики (характеристик), как правило, имеющей небольшую вычислительную сложность и определяемой на основании предыдущих тестовых наборов $T_0, T_1, T_2, \ldots, T_{i-1}$ [5−10]. К числу таких характеристик относится расстояние Хэмминга и декартово расстояние. Во всех ранее рассмотренных методах находятся лучшее или локально лучшее решения на основании простейших метрик по жадному алгоритму (*Greedy algorithm*), который заключается в принятии локально оптимальных решений на каждом этапе при допущении, что конечное решение также окажется оптимальным. Если глобальная оптимальность алгоритма имеет место практически всегда, жадный алгоритм является предпочтительным по отношению к другим методам оптимизации, например таким, как динамическое программирование. В случае управляемых случайных тестов жадный алгоритм является единственно возможным для практического использования в силу его существенно меньшей вычислительной сложности по сравнению с другими оптимизационными алгоритмами.

Используя метрики, рассмотренные в предыдущем разделе, а именно расстояние Хэмминга $HD(T_i, T_j)$ для тестовых наборов T_i и T_j (1), декартово расстояние $CD(T_i, T_j)$ (2), суммарное расстояние Хэмминга $THD(T_i)$ для следующего набора T_i (3), суммарное декартово расстояние $TCD(T_i)$ (3), суммарное расстояние Хэмминга $THD(T)$ для теста T (4) и суммарное декартово расстояние $TCD(T)$ (4) [5, 8, 17], синтезируем оптимальный управляемый случайный тест.

На первом шаге, следуя концепции черного ящика, генерируем произвольный тестовый набор T_0, состоящий из N бит. Отметим, что в качестве T_0 может быть любой из 2^N тестовых наборов. Не нарушая общности дальнейших рассуждений, предположим, что $T_0 = 000\ldots0$.

В соответствии со всеми ранее приведенными метриками (1)−(4) в качестве второго тестового набора T_1 оптимальным будет инверсное значение T_0, т. е. $T_1 = \overline{T_0}$. Тогда оптимальный управляемый случайный тест (*Optimal Controlled Random Test* (*OCRT*)), состоящий из двух $q = 2$ наборов, принимает вид $OCRT = \{T_0, T_1\}$, где $T_1 = \overline{T_0}$. Для ранее приведенного примера $T_1 = \overline{T}_0 = \overline{000...0} = 111...1$. Оптимальность выбора второго тестового набора T_1 как инверсии первого T_0 подтверждается максимальными значениями всех ранее рассмотренных метрик. Действительно, $HD(T_0,T_1) = THD(T_1) = THD(T) = N$ и $CD(T_0, T_1) = TCD(T_1) = TCD(T) = \sqrt{N}$.

Для получения третьего набора T_2 *OCRT* нельзя использовать характеристики $HD(T_i, T_j)$ и $CD(T_i, T_j)$. Это объясняется тем фактом, что максимизация значений $HD(T_2, T_1)$ и $CD(T_2, T_1)$ приводит к противоречивому результату, а именно $T_2 = T_0$. Поэтому при дальнейших исследованиях данные характеристики не будут использованы. Аналогичное заключение может быть сделано по отношению метрик $THD(T_i)$ и $THD(T)$. Это следует из того факта, что для $OCRT = \{T_0, T_1, T_2\}$, где $T_1 = \overline{T_0}$, любой третий набор T_2 позволяет максимизировать данные характеристики. Действительно, для произвольного T_2 $THD(T_2) = N$ и $THD(T) = 2N$. Следует отметить, что даже для случаев, когда $T_2 = T_0$ и $T_2 = T_1$, указанные метрики принимают максимально возможные значения $THD(T_2) = N$ и $THD(T) = 2N$, что свидетельствует об оптимальности выбора T_2 в обоих случаях.

С учетом приведенного анализа кандидатом в третий набор T_2 $OCRT$ может быть любой набор T_2, который удовлетворяет соотношениям $T_2 \neq T_0$ и $T_2 \neq T_1$. Предположим, что в качестве T_2 выбран набор, для которого $HD(T_i, T_j)$ принимает значение $HD(T_0, T_2)$, тогда $HD(T_1, T_2) = N - HD(T_0, T_2)$. Характеристики, основанные на декартовом расстоянии, принимают значения $TCD(T_i) = CD(T_0, T_2) + CD(T_1, T_2) = \sqrt{HD(T_0, T_2)} + \sqrt{N - HD(T_0, T_2)}$ и $TCD(T) = CD(T_0, T_1) + CD(T_0, T_2) + CD(T_1, T_2) = \sqrt{N} + \sqrt{HD(T_0, T_2)} + \sqrt{N - HD(T_0, T_2)}$. Соответственно $TCD(T_i)$ и $TCD(T)$ принимают максимальное значение при оптимальной величине $HD(T_0, T_2) = N/2$, полученной как решение следующего уравнения:

$$\frac{\partial(\sqrt{H(T_0, T_2)} + \sqrt{N - H(T_0, T_2)})}{\partial(H(T_0, T_2))} = 0 .$$

Приняв допущение, что N является четной величиной, для примера, рассмотренного выше, оптимальным тестовым набором T_2 для двух ранее полученных векторов $T_0 = 000\ldots0000\ldots0$ и $T_1 = 111\ldots1111\ldots1$ будет являться набор, состоящий из $N/2$ бит, принимающих значение 0, и из $N/2$ бит, принимающих значение 1, т. е. $T_2 = 000\ldots0111\ldots1$. Таким образом, для $q = 3$ $OCRT = \{T_0, T_1, T_2\} = \{000\ldots0000\ldots0, 111\ldots1111\ldots1, 000\ldots0111\ldots1\}$.

Для следующего кандидата в тестовые наборы необходимо также достичь максимальных значений $TCD(T_i)$ и $TCD(T)$. С целью максимизации $HD(T_0, T_3)$ и $HD(T_1, T_3)$ необходимо выбирать T_3 из множества наборов, для которых сумма $HD(T_0, T_i) + HD(T_1, T_i)$ принимает максимальное значение, т. е. для кандидата в четвертый тестовый набор должно выполняться равенство $w(T_i) = N/2$. Для получения максимальных значений метрик $TCD(T_3)$ и $TCD(T)$ очевидным и единственным решением является $T_3 = \overline{T_2}$. Для ранее рассмотренного примера получим $T_3 = 111\ldots1000\ldots0$ и $OCRT = \{T_0, T_1, T_2, T_3\} = \{000\ldots0000\ldots0,\ 111\ldots1111\ldots1,\ 000\ldots0111\ldots1,\ 111\ldots1\,000\ldots0\}$.

Аналогичный результат будет получен с использованием FAR-алгоритма [8]. Согласно данному алгоритму, основываясь на трех предыдущих наборах $T_0 = 000\ldots0000\ldots0$, $T_1 = 111\ldots1111\ldots1$ и $T_2 = 000\ldots0111\ldots1$, можно получить центроид $C = c_{N-1}, c_{N-2}, \ldots, c_2, c_1, c_0 = 1/3, 1/3, 1/3, \ldots, 1/3, 2/3, 2/3, 2/3, \ldots, 2/3$, а на его основе − бинарный центроид $C_b = 000\ldots0111\ldots1$ [8]. Наконец, инвертируя значения элементов бинарного центроида, окончательно получим $T_3 = 111\ldots1000\ldots0$. Для набора T_3 декартовы метрики принимают максимальные значения, равные соответственно $TCD(T_3) = \sqrt{N} + 2\sqrt{N/2}$ и $TCD(T) = 2\sqrt{N} + 4\sqrt{N/2}$.

Для процедуры генерирования $OCRT$ с учетом предыдущих этапов может быть использовано следующее условие. На всех последующих этапах генерирования очередного тестового набора необходимо максимизировать значения метрик $TCD(T_i)$ и $TCD(T)$. Тогда для набора T_i с четным значением $i \in \{0, 2, 4, \ldots, 2k-2\}$ должны выполняться равенства

$$maxTCD(T_i) = i \times \sqrt{N/2} ;$$
$$maxTCD(T) = (i/2) \times \sqrt{N} + (i^2/2) \times \sqrt{N/2} , \quad i \in \{0, 2, 4, \ldots, 2k-2\}. \tag{6}$$

Отметим, что приведенные равенства представляют максимально возможные значения метрик $TCD(T_i)$ и $TCD(T)$, которые обеспечиваются путем выбора набора T_i с четным индексом i, такого, что $HD(T_i, T_j) = N/2$ между набором T_i и всеми предыдущими наборами T_j, $j < i$.

В то же время набор с нечетными индексами $i \in \{1, 3, 5, \ldots, 2k-1\}$ является инверсным значением по отношению к предыдущему набору, т. е. $T_i = \overline{T_{i-1}}$, что позволяет максимизировать значения метрик $TCD(T_i)$ и $TCD(T)$:

$$maxTCD(T_i) = \sqrt{N} + (i-1) \times \sqrt{N/2}\,;$$
$$maxTCD(T) = ((i+1)/2) \times \sqrt{N} + ((i^2-1)/2) \times \sqrt{N/2}\,, \quad i \in \{0, 2, 4, \ldots, 2k-2\}. \tag{7}$$

В качестве примера рассмотрим случай, когда $N = 2^m$, тогда количество q наборов $OCRT = \{T_0, T_1, T_2, \ldots, T_{q-1}\}$ равняется $2(m+1)$. Для общего случая количество наборов $OCRT$ определяется как $q = 2(\lceil \log_2 N \rceil + 1)$, а конструктивный алгоритм для генерирования тестовых наборов представлен в [20].

Предполагая, что T_0 = 00000000, $N = 2^m$ и $m = 3$, $OCRT$ будет состоять из $2(m+1) = 2(3+1) = 8$ тестовых наборов (табл. 2). В табл. 2 также представлены соответствующие значения (6) и (7) для $maxTCD(T_i)$ и $maxTCD(T)$.

Таблица 2

Оптимальный управляемый тест для $q = 8$

T_i	$t_{i,7}$	$t_{i,6}$	$t_{i,5}$	$t_{i,4}$	$t_{i,3}$	$t_{i,2}$	$t_{i,1}$	$t_{i,0}$	maxTCD(T_i)	maxTCD(T)
T_0	0	0	0	0	0	0	0	0	–	–
T_1	1	1	1	1	1	1	1	1	$\sqrt{8}$	$\sqrt{8}$
T_2	0	0	0	0	1	1	1	1	$2\sqrt{4}$	$\sqrt{8}+2\sqrt{4}$
T_3	1	1	1	1	0	0	0	0	$\sqrt{8}+2\sqrt{4}$	$2\sqrt{8}+4\sqrt{4}$
T_4	0	0	1	1	0	0	1	1	$4\sqrt{4}$	$2\sqrt{8}+8\sqrt{4}$
T_5	1	1	0	0	1	1	0	0	$\sqrt{8}+4\sqrt{4}$	$3\sqrt{8}+12\sqrt{4}$
T_6	0	1	0	1	0	1	0	1	$6\sqrt{4}$	$3\sqrt{8}+18\sqrt{4}$
T_7	1	0	1	0	1	0	1	0	$\sqrt{8}+6\sqrt{4}$	$4\sqrt{8}+24\sqrt{4}$

Следует отметить, что в результате синтеза оптимального управляемого теста $OCRT$ с помощью жадного алгоритма был получен достаточно тривиальный результат. Такой же результат легко может быть получен на основании классического алгоритма бинарного поиска (*Divide and Conquer algorithm*) [21]. Тест $OCRT$ также может быть получен при применении одного из известных методов генерирования управляемых случайных тестов, а именно при использовании процедуры генерирования антислучайных тестов [5−7], быстрых антислучайных тестов [8], ограниченных случайных тестов [14, 15] и многочисленных их модификаций. Все перечисленные методы используют критерии выбора очередного тестового набора на основе расстояния Хэмминга и декартова расстояния.

В качестве универсальной процедуры генерирования оптимальных управляемых случайных тестов $OCRT$ применим следующий алгоритм.

Процедура 2. Генерирование $OCRT$ для $N = 2^m$.

1. Первоначально генерируется матрица M, содержащая N столбцов и $q = 2(m+1)$ строк, с помощью, например, алгоритма бинарного поиска [21].

Первая строка M_0 матрицы M принимает нулевое значение, вторая строка M_1 состоит из N единиц. Последующие строки матрицы M с четными индексами $i > 0$ представляют собой множество, состоящее из четного количества блоков одинаковой размерности, одна половина которых представляет собой нулевые блоки, а другая половина − единичные. На каждой итерации очередная строка с четным индексом формируется из предыдущей строки для четного i, при этом все блоки предыдущей строки делятся на два блока, первый из которых является нулевым блоком, а второй − единичным. Каждая строка с нечетным индексом формируется как инвертирование элементов предыдущей строки с четным индексом.

2. Основываясь на одном из алгоритмов перестановок [21], столбцы матрицы M перемешиваются для получения матрицы M^* − так называемых векторов масок.

3. Случайным образом формируется вектор P, состоящий из N бит, который принимается как первый набор $T_0 = P$ *OCRT* теста.

4. Каждый последующий набор T_1, T_2, T_3, …, T_{2m+1} определяется как поразрядная сумма вектора P и соответствующей строки матрицы M^*, т. е. $T_i = P \oplus M_i^*$.

5. Этап 4 повторяется до тех пор, пока все $2(m+1)$ наборы не будут сгенерированы.

Следует отметить, что перестановка столбцов матрицы не изменяет значения расстояния Хэмминга между двумя строками (наборами), что следует из соотношения (1).

Пример 4. В качестве примера рассмотрим *OCRT* для $N = 2^m = 2^3$.

1. Матрица M с $N = 8$ столбцами и $q = 2(3 + 1) = 8$ строками строится таким образом, что $M_0 = 00000000$, а $M_1 = 11111111$. Следующая строка с четным индексом M_2 формируется на основании M_0 путем деления M_0 на два равных блока, первый из которых содержит нулевые значения, а второй − единичные, т. е. $M_2 = 00001111$. Строка $M_3 = 11110000$ получается путем инвертирования значений строки M_2. Аналогично формируются две следующие строки $M_4 = 00110011$ и $M_5 = 11001100$ матрицы M. Последние две строки принимают вид $M_6 = 01010101$ и $M_7 = 10101010$ (см. табл. 2).

2. Генерируется матрица M^* векторов масок (табл. 3) как результат перестановок столбцов матрицы M (см. табл. 2).

Таблица 3

Матрица M^* векторов масок для $m = 3$

M_i^*	$t_{i,2}$	$t_{i,6}$	$t_{i,1}$	$t_{i,4}$	$t_{i,3}$	$t_{i,7}$	$t_{i,5}$	$t_{i,0}$
M_0^*	0	0	0	0	0	0	0	0
M_1^*	1	1	1	1	1	1	1	1
M_2^*	1	0	1	0	1	0	0	1
M_3^*	0	1	0	1	0	1	1	0
M_4^*	0	0	1	1	0	0	1	1
M_5^*	1	1	0	0	1	1	0	0
M_6^*	1	1	0	1	0	0	0	1
M_7^*	0	0	1	0	1	1	1	0

3. Генерируется случайный вектор $P = 01111010$, который определяет $T_0 = P = 01111010$.

4. Каждый новый i-й набор T_1, T_2, T_3, …, T_7 получается как сумма $T_i = P \oplus M_i^*$, где M_i^* выбирается из табл. 3.

5. Все восемь наборов для *OCRT* представляются в табл. 4 с соответствующими значениями характеристик $maxTCD(T_i)$ и $maxTCD(T)$.

Таблица 4

Управляемый случайный тест *OCRT* для m=3

T_i	$t_{i,7}$	$t_{i,6}$	$t_{i,5}$	$t_{i,4}$	$t_{i,3}$	$t_{i,2}$	$t_{i,1}$	$t_{i,0}$	$maxTCD(T_i)$	$maxTCD(T)$
T_0	0	1	1	1	1	0	1	0	–	–
T_1	1	0	0	0	0	1	0	1	$\sqrt{8}$	$\sqrt{8}$
T_2	1	1	0	1	0	0	1	1	$2\sqrt{4}$	$\sqrt{8}+2\sqrt{4}$
T_3	0	0	1	0	1	1	0	0	$\sqrt{8}+2\sqrt{4}$	$2\sqrt{8}+4\sqrt{4}$
T_4	0	1	0	0	1	0	0	1	$4\sqrt{4}$	$2\sqrt{8}+8\sqrt{4}$
T_5	1	0	1	1	0	1	1	0	$\sqrt{8}+4\sqrt{4}$	$3\sqrt{8}+12\sqrt{4}$
T_6	1	1	0	1	0	0	0	1	$6\sqrt{4}$	$3\sqrt{8}+18\sqrt{4}$
T_7	0	0	1	0	1	1	1	0	$\sqrt{8}+6\sqrt{4}$	$4\sqrt{8}+24\sqrt{4}$

В данном разделе представлен метод генерирования оптимальных управляемых случайных тестов как обобщение известных алгоритмов, использующих в своей основе жадный алгоритм, а также расстояние Хэмминга и декартово расстояние в качестве критерия выбора очередного тестового набора. В отличие от известных решений, предложенный метод гарантированно обеспечивает максимально возможные значения расстояний $TCD(T_i)$ и $TCD(T)$. Следует отметить, что аналогичный результат может быть получен с использованием известных решений [5−8, 14, 15], однако его достижение не является возможным и требует большого объема вычислений, связанных с определением характеристик расстояния для каждого потенциального кандидата в тестовые наборы. В свою очередь, оптимальные управляемые случайные тесты (*OCRT*) имеют очевидное преимущество по сравнению со всеми известными методами генерирования управляемых случайных тестов. Они характеризуются минимальной вычислительной сложностью в силу отсутствия процедур перечисления всевозможных кандидатов в тестовые наборы и вычисления для них соответствующих характеристик. Для произвольного N наиболее трудоемкой процедурой при генерировании *OCRT* является процедура построения матрицы M, состоящей из $q = 2(m + 1)$ строк, и выполнения перемешивания ее столбцов для получения M^*.

Заключение

В работе проведен анализ методов генерирования управляемых случайных тестов и их модификаций. Показано, что в качестве характеристик, используемых как критерий включения кандидата в наборы управляемого случайного теста, применяются метрики, основанные на расстоянии Хэмминга и декартовом расстоянии. Обосновывается общность процедур генерирования тестовых наборов управляемых случайных тестов, использующих жадный алгоритм оптимизации. Предложен метод построения оптимальных управляемых случайных тестов, характеризующихся максимальной полнотой покрытия и минимальной вычислительной сложностью в сравнении со случайными и управляемыми случайными тестами.

Список литературы

1. Grindal, M. Combination Testing Strategies / M. Grindal, J. Offutt, S.F. Andler // GMU Technical Report ISE-TR-04-05. – George Mason University, USA, 2004. – 32 p.

2. Malaiya, Y.K. The coverage problem for random testing / Y.K. Malaiya, S. Yang // Proc. of ITC. – Las Vegas, USA, 1984. – P. 237–242.

3. Seth, S. A statistical theory of digital circuits testability / S. Seth, V. Agrawal, H. Farhat // IEEE Transactions on Computers. – 1990. – Vol. C-39, № 4. – P. 582–586.

4. Malaiya, Y.K. An examination of fault exposure ratio / Y.K. Malaiya, A. Mayrhauser, P.K. Srimani // IEEE Transactions on Software Engineering. – 1993. – Vol. 19, № 11. – P. 1087–1094.

5. Malaiya, Y.K. Antirandom Testing: Getting the most out of Back-Box Testing / Y.K. Malaiya, S. Yang // Proc. of Sixth Intern. Symposium on Software Reliability Engineering. – Toulouse, France, 1995. – P. 86–95.

6. Antirandom Testing: A Distance-Based Approach / S.H. Wu [et al.] // VLSI Design. – 2008. – № 2. – P. 1–9.

7. Wu, S.Y. Antirandom vs. Pseudorandom Testing / S.H. Wu, Y.K. Malaiya, A.P. Jayasumana // Proc. of IEEE Intern. Conf. on Computer Design (ICCD'98). – Austin, Texas, USA, 1998. – P. 221–223.

8. Fast Antirandom (FAR) Test Generation / A. Mayrhause [et al.] // Proc. of the Third IEEE Intern. High-Assurance System Engineering Symposium. – Washington, D.C., USA, 1998. – P. 262–269.

9. Chen, T.Y. Adaptive Random Testing / T.Y. Chen, H. Leung, I.K. Mak // Proc. of the 9th Asian Computer Science Conf. (ASIAN 2004). – Chiang Mai, Thailand, 2004. – P. 320–329.

10. Zhou, Z.Q. Using Coverage Information to Guide Test Case Selection in Adaptive Random Testing / Z.Q. Zhou // Proc. of the 34th IEEE Computer Software and Applications Conf. Workshops. – Seoul, South Korea, 2010. – P. 208–213.

11. Adaptive Random Test Case Prioritization / B. Jiang [et al.] // Proc. of the IEEE/ACM Intern. Conf. on Automated Software Engineering. – Auckland, New Zealand, 2009. – P. 233–244.

12. Tappenden, A.F. A Novel Evolutionary Approach for Adaptive Random Testing / A.F. Tappenden, J. Miller // IEEE Transaction on reliability. – 2009. – Vol. 58, № 4. – P. 619–632.

13. Chan, K.P. Good Random Testing / K.P. Chan, T.Y. Chen, D. Towey // Proc. of the 9th Ada-Europe Intern. Conf. on Reliable Software Technologies (LNCS). – Palma de Mallorca, Spain, 2004. – P. 200–212.

14. Chan, K.P. Normalized Restricted Random Testing / K.P. Chan, T.Y. Chen, D. Towey // Proc. of the 8th Ada-Europe Intern. Conf. on Reliable Software Technologies (LNCS). – Toulouse, France, 2003. – P. 368–381.

15. Chan, K.P. Restricted Random Testing / K.P. Chan, T.Y. Chen, D. Towey // Proc. of the 7th European Conf. on Software Quality. – Helsinki, Finland, 2002. – P. 321–330.

16. Kuo, F.C. An in-depth study of mirror adaptive random testing / F.C. Kuo // Proc. of the 14th European Conf. on Software Quality. – Kaiserslautern, Germany, 2009. – P. 51–58.

17. Shiyi, Xu. Orderly Random Testing for Both Hardware and Software / Xu Shiyi // Proc. of Pacific Rim Intern. Symposium on Dependable Computing. – Gold Coast, Australia, 2008. – P. 160–167.

18. Shiyi, Xu. Maximum Distance Testing / Xu Shiyi, Chen Jianwen // Proc. of the 11-th IEEE Asian Test Symposium (ATS'02). – Los Alamitos, CA, USA, 2002. – P. 15–20.

19. Hamming, W.R. Error Detecting and Error Correcting Codes / W.R. Hamming // Bell System Tech. Journal. – 1950. – Vol. 26, № 2. – P. 147–160.

20. Das, D. Exhaustive and Near–Exhaustive Memory Testing Techniques and their BIST Implementations / D. Das, M.G. Karpovsky // Journal of Electronic Testing: Theory and Applications. – 1997. – Vol. 10. – P. 215–229.

21. Knuth, D.E. The Art of Computer Programming. Vol. 3: Sorting and Searching. / D.E. Knuth. – 2nd ed. – Massachusetts : Addison-Wesley, 1998. – 730 p.

Поступила 05.01.11

Белорусский государственный университет
информатики и радиоэлектроники,
Минск, П. Бровки, 6
e-mail: yarmolik@cosmostv.by,
yarmolik10ru@yahoo.com

S.V. Yarmolik, V.N. Yarmolik

CONTROLLED RANDOM TESTING

The Controlled Random Tests and methods for their generation have been analyzed and investigated. The similarities of all known controlled random testing approaches are demonstrated. The most important feature of the controlled random tests is Greedy-like procedure for its test pattern generation. The new method for controlled random test generation, called Optimal Controlled Random Testing, have been proposed and analyzed. This method has low computational complexity and maximal faults (errors) coverage due to maximal values of the distances in between test patterns.