УДК 621.391:004.73

А.А. ЛАВДАНСКИЙ
Черкасский государственный технологический университет

ОЦЕНКА КАЧЕСТВА ГЕНЕРАТОРОВ ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ ПО ВЕЛИЧИНЕ ОШИБКИ ВОСПРОИЗВЕДЕНИЯ ЗАКОНА РАСПРЕДЕЛЕНИЯ

Рассматриваются наиболее часто применяемые в практических целях алгоритмы генерации псевдослучайных чисел. Производится их сравнение на основе критерия оценки ошибки воспроизведения закона распределения дискретной случайной величины. Показано, что исследуемые генераторы имеют высокое качество генерируемой последовательности.

Ключевые слова: ошибка воспроизведения закона распределения дискретной случайной величины, сравнение генераторов псевдослучайных чисел, линейный конгруэнтный метод, вихрь Мерсенна.

ARTEM ALEKSANDROVICH LAVDANSKIY
Cherkasy State Technological University

QUALITY ASSESSMENT OF PSEUDORANDOM NUMBER GENERATORS BY MAGNITUDE OF PLAYBACK ERROR OF THE DISTRIBUTION

Abstract – A pseudorandom number generator (PRNG) is an algorithm for generating a sequence of numbers that approximates the properties of random numbers. Pseudorandom number generators are used in different fields of science such as cryptography, simulations, procedural generation and others.

The most frequently used in practical algorithms for generate pseudorandom numbers are examined in this paper. Their comparison is made on the basis of playback error of the distribution of a discrete random variable. Three different pseudorandom number generators are examined: linear congruential generator in Java, subtractive random number generator algorithm in C#, Mersenne twister in PHP. Also a random generator with source of entropy is examined (/dev/urandom in Linux).

All of the examined number generators have good quality of generated output and can be used for practical purposes.

Keywords: playback error of the distribution of a discrete random variable, comparing pseudorandom number generators, linear congruential generator, Mersenne twister

Введение

В настоящее время существует достаточно большое количество алгоритмов формирования псевдослучайных последовательностей чисел, имеющих свои достоинства и недостатки и используемых в различных приложениях и средах программирования. Каждый из языков программирования, как правило, имеет свою реализацию генератора псевдослучайных чисел (ГПСЧ) в стандартных библиотеках. Это накладывает некоторые ограничения на выбор среды программирования при разработке программного обеспечения, чувствительного к качеству стандартного ГПСЧ.

Для оценки качества ГПСЧ чаще всего используют следующие основные характеристики: период повторения, корреляция между словами последовательности, равномерность распределения чисел в некотором диапазоне, равномерность распределения k-грамм, величина ошибки воспроизведения закона распределения дискретной случайной величины. Комплексная оценка по этим параметрам и определяет качество ГПСЧ. Повсеместное использование стандартных средств формирования ГПСЧ делает актуальной задачу определения качества выходной последовательности ГПСЧ.

Целью данной работы является сравнение качества стандартных средств генерации ГПСЧ по величине ошибки воспроизведения закона распределения.

Выделение не решенных ранее частей общей проблемы

К настоящему времени разработано множество методов тестирования псевдослучайных последовательностей. Все методы тестирования ГПСЧ можно разделить на две группы: графические и статистические. Графические тесты (автокорреляционная функция, спектральный тест, равномерность распределения чисел и др.) отображают результаты в виде гистограмм и графиков, характеризующих свойства исследуемой последовательности, но не дают количественной оценки. Статистические же тесты дают возможность выполнить численную оценку качества ГПСЧ. Статистические тесты обычно объединяются в пакеты тестирования (среди них можно выделить тесты DIEHARD, тесты NIST и др.). Одной из статистических оценок является оценка ошибки воспроизведения закона распределения дискретной случайной величины [1].

К настоящему времени произведено множество тестов ГПСЧ с помощью различных методик (в частности, с помощью тестов Джорджа Марсальи ("diehard") результаты которых приведены в [2]), однако оценка ошибки воспроизведения закона распределения для наиболее используемых ГПСЧ ранее не выполнялась. В отличие от существующих статистических тестов (например, критерия хи-квадрат, всех тестов "diehard" и др.), которые дают вероятностную оценку качества ГПСЧ, определение ошибки воспроизведения закона распределения дискретной случайной величины позволяет численно оценить качество генерируемой последовательности (а также произвести сравнение с другими ГПСЧ). Следует учитывать, что оценка ошибки воспроизведения закона распределения является лишь одним из параметров

качества последовательности, который показывает степень соответствия закона распределения тестируемой последовательности заданному закону. Поэтому данную оценку следует использовать только в комплексе с другими методами оценки качества псевдослучайных последовательностей.

Постановка задачи

Задачей данного исследования является оценка ошибки воспроизведения закона распределения дискретной случайной величины, порожденной генератором псевдослучайных чисел, для наиболее часто используемых алгоритмов генерации в разных языках программирования.

Решение задачи

Для решения задачи были взяты наиболее широко используемые пакеты разработки программного обеспечения, такие как:

- .NET Framework (версия 4.5.50709) – класс Random();
- PHP (версия 5.4.4) – функция mt_rand();
- Java (версия 1.7.0_25-b17) – класс Random().

Кроме того, выполнялось сравнение полученных результатов оценки выбранных генераторов с системным генератором случайных чисел с источником внешней энтропии семейства операционных систем Linux /dev/urandom (версия ядра 3.0.5-25).

Особенностью данного генератора является полная непредсказуемость выходного потока символов. Принцип работы генератора состоит в следующем. В специальный буфер собираются "шумы" устройств. Под шумами устройств понимают данные от устройств компьютера (тайминги нажатий клавиш на клавиатуре, движения мыши, тайминги прерываний винчестера и т.д.). Для получения выходного потока случайных символов из буфера извлекается 512 бит данных, над которыми применяется алгоритм хеширования SHA-1, результат которого и выдается на выход. Более подробно алгоритм работы генератора случайных чисел семейства операционных систем Linux /dev/urandom описан в [3].

Следует отметить, что для оценки ошибки воспроизведения закона распределения дискретной случайной величины выбраны ГПСЧ с разным принципом работы. Так, ГПСЧ в среде программирования .NET Framework построен на основе аддитивного генератора, описанного в [4]. Период ГПСЧ, выполненного по такой схеме, составляет 2^{55}-1. ГПСЧ в Java построен на использовании линейного конгруэнтного метода с периодом 2^{48} [5]. На основе вихря Мерсенна выполнен ГПСЧ, используемый в языке PHP. Его период равен 2^{19937}-1 [6].

Сравнение ГПСЧ проводилось на основе оценки ошибки воспроизведения закона распределения дискретной случайной величины для множества целых чисел отрезка $\left[0, N_1 - 1\right]$, приведенной в [1]:

$$\xi = \frac{1}{2V} \sum_{x=0}^{N_1-1} \left|\Delta n(x)\right|, \tag{1}$$

где V – объем выборки;

$\Delta n(x) = n_0(x) - n(x)$;

$n_0(x)$ – количество повторений символа x в теоретическом потоке;

$n(x)$ – количество повторений символа x в эмпирическом потоке.

В соответствии с [1], выходной поток каждого ГПСЧ можно представить в виде композиции двух потоков: потока символов с теоретическим законом распределения и мешающего потока с неизвестным законом распределения. Композиция этих потоков определяет эмпирическое распределение выходной последовательности ГПСЧ. Таким образом, ошибка воспроизведения закона распределения дискретной случайной величины (1) указывает количество мешающих символов, изменивших некоторый символ, на единицу объема выборки [1].

При практическом использовании ГПСЧ в значение ошибки воспроизведения закона распределения дискретной случайной величины как аддитивные коэффициенты входят статистическая и конструктивная составляющие.

Статистическая составляющая ошибки воспроизведения закона распределения дискретной случайной величины обусловлена некратностью периода повторения последовательности, производимой ГПСЧ, выборке. Рассмотрим внутренний алгоритм генерации слов последовательности ГПСЧ до преобразовании области определения. Слова, порожденные внутренним алгоритмом ГПСЧ, принадлежат интервалу [0, T-1] (где T – период повторения ГПСЧ), при этом каждое слово интервала встречается только один раз. В таком случае ошибка воспроизведения будет минимальна. Очевидно, что при выборке, некратной периоду T, некоторые слова будут встречаться реже (или чаще) других слов интервала [0, T-1], что и определяет статистическую составляющую ошибки.

Конструктивная составляющая ошибки воспроизведения закона распределения дискретной случайной величины может быть обусловлена конструкцией генератора или проявляться при преобразовании случайной величины.

Следует учитывать, что минимизировать статистическую составляющую ошибки воспроизведения закона распределения дискретной случайной величины в работе не удалось, так как генерация и

исследование выборок размером в 2^{48} (период ГПСЧ в стандартных библиотеках языка программирования Java) и более байт в данной работе не представилось возможным. Кроме того, каждый ГПСЧ имеет свою конструктивную величину ошибки, обусловленную внутренним алгоритмом генерации и преобразования области определения слов последовательности.

В целях исследования с помощью каждого из ГПСЧ сгенерировано бинарные файлы, содержащие множество целых чисел диапазона [0, 255] с размером выборки от 2^{16} до 2^{32} байт и кратной 2^{16} байт. По этим выборкам определялась статистика распределения дискретной случайной величины и ошибка воспроизведения равномерного закона распределения в соответствии с формулой (1).

Следует отметить, что критерии оценки значения ошибки воспроизведения закона распределения дискретной случайной величины должны определяться применительно к условиям задачи, в которой планируется использовать ГПСЧ.

Полученные результаты

Результаты сравнения генераторов по критерию ошибки воспроизведения закона распределения дискретной случайной величины представлены в табл.1.

Таблица 1

Ошибка воспроизведения закона распределения для тестируемых генераторов

Выборка	.NET	PHP	Java	/dev/urandom
2^{16}	0,0239944458	0,0248336792	0,0244369507	0,0248870850
2^{17}	0,0186119080	0,0189094543	0,0179100037	0,0158157349
2^{18}	0,0127582550	0,0123310089	0,0120220184	0,0135955811
2^{19}	0,0089941025	0,0094995499	0,0076856613	0,0094470978
2^{20}	0,0059075356	0,0060381889	0,0056862831	0,0065135956
2^{21}	0,0043914318	0,0043108463	0,0042927265	0,0041775703
2^{22}	0,0030566454	0,0030938387	0,0029314756	0,0029478073
2^{23}	0,0022252202	0,0020441413	0,0020925403	0,0023204088
2^{24}	0,0013916790	0,0016477406	0,0015237033	0,0014507771
2^{25}	0,0010771006	0,0010732263	0,0009723753	0,0010671020
2^{26}	0,0007315651	0,0007950142	0,0005810633	0,0007509738
2^{27}	0,0005552359	0,0006021895	0,0004854910	0,0005859062
2^{28}	0,0003655534	0,0004040431	0,0003316347	0,0003970563
2^{29}	0,0002554553	0,0002666740	0,0002322616	0,0002693329
2^{30}	0,0001946460	0,0002006642	0,0001725969	0,0001914827
2^{31}	0,0001383310	0,0001284091	0,0001107145	0,0001451354
2^{32}	0,0001028968	0,0000927021	0,0000803099	0,0001020837

Из таблицы 1 видно, что для ГПСЧ величина ошибки воспроизведения закона распределения дискретной случайной величины примерно равна 10^{-3} уже при выборке в 2^{25}. Данное утверждение означает наличие не более одного мешающего символа в выборке объемом 1000 слов. Для максимальной выборки в 2^{32} слов ошибка воспроизведения достигает 10^{-4}, что соответствует наличию не более одного мешающего символа в выборке объемом 10000 слов.

Поскольку результаты оценки ошибки воспроизведения, приведенные в таблице 1, включают статистическую составляющую ошибки воспроизведения (так как выборка не кратна периоду), дальнейшее увеличение выборки приводит к уменьшению значения статистической составляющей ошибки. Конструктивная составляющая ошибки воспроизведения закона распределения дискретной случайной величины постоянна и не зависит от объема выборки, но отличается для каждого из генераторов. Таким образом, можно предполагать дальнейшее уменьшение значения ошибки для всех представленных генераторов по мере увеличения выборки. Минимальное значение ошибки будет достигнуто при равенстве объема выборки периоду для каждого из генераторов.

Выводы

Анализ количественной оценки ошибки воспроизведения закона распределения дискретной случайной величины подтвердил высокое качество выходных последовательностей для всех представленных в работе ГПСЧ.

Литература

1. Фауре Э.В. Оценка точности воспроизведения закона распределения дискретной случайной величины при ее преобразовании / Э.В. Фауре, А.С. Береза, Е.А. Ярославская // Вісник Хмельницького національного університету. – 2012. – №5. – С. 176–182.

2. Random Number Generator Results [Електронний ресурс]. – Режим доступу : http://www.cacert.at/cgi-bin/rngresults .

3. Linux kernel sources [Електронний ресурс]. – Режим доступу : http://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/tree/drivers/char/random.c .

4. Кнут Д. Э. Искусство программирования. Том 2. Получисленные алгоритмы / Дональд Э. Кнут. – М. : Вильямс, 2007. – 832 с.

5. Java Class Random [Електронний ресурс]. – Режим доступу : http://docs.oracle.com/javase/6/docs/api/java/util/Random.html

6. Matsumoto M., Nishimura T. Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator, M. Matsumoto. ACM Trans. Model. Comput. Simulat.,1998,V.8, pp. 3–30.

References

1. E.V. Faure "Evaluation of reproduction accuracy of the distribution law of discrete random variable with its transformation" / E.V. Faure, A.S. Bereza, E.A. Yaroslavskaya. Visnyk Khmelnytskoho natsionalnoho universytetu. Technical science. Khmelnitsky. 2012. Volume 193. Issue 5. pp. 176-182.

2. Random Number Generator Results (Visited 10.10.13) http://www.cacert.at/cgi-bin/rngresults .

3. Linux kernel sources (Visited 10.10.13) http://git.kernel.org/cgit/linux/kernel/git/torvalds/linux.git/tree/drivers/char/random.c .

4. Knuth Donald, Art of Computer Programming, Seminumerical Algorithms (3rd Edition), Massachusetts: Addison-Wesley, 1997.

5. Java Class Random (Visited 10.10.13) http://docs.oracle.com/javase/6/docs/api/java/util/Random.html

6. M. Matsumoto "Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator", M. Matsumoto, T. Nishimura, ACM Trans. Model. Comput. Simulat.,1998,V.8, pp.3-30.

Рецензія/Peer review : 12.12.2013 р. Надрукована/Printed :9.2.2014 р.
Рецензент: д.т.н., проф., зав. кафедри комп'ютерних систем
Черкаського державного технологічного університету, Середенко В.М.