НОВИ ЕЛЕМЕНТИ В МЕТОДА МАКЛАРЕН – МАРСАЛЬИ

Тянев Д. С.

Резюме: Разглежда се проблемът за алгоритмично генериране на псевдо-случайни числа по стандартен закон на равномерно разпределение. Анализира се същността на един от утвърдените методи за реализация на базова случайна величина – методът на Макларен-Марсальи. Въвежда се нова интерпретация на неговата функцията за избор на елемент от FIFO-буфера. Въз основа на направените изводи се предлага усъвършенстване на алгоритмичната схема чрез вграждане на по-подходяща хеш-функция. Новата алгоритмична схема е експериментирана и оценена чрез критерия на Колмогоров.

1. Увод

Компютърното моделиране и методът Монте-Карло се основават на избран модел на базова случайна величина (БСВ), апроксимираща едномерен стандартен закон на разпределение, с функция на плътността

(1) $$P(x) = \begin{cases} 1, & x \in [0,1); \\ 0, & x \notin [0,1). \end{cases}$$

Качеството на компютърния модел на БСВ се определя преди всичко от степента на равномерност на генерираните числа във всяка извадка $\{x_i\}$ с произволен обем $i = \overline{1,n}$. Това качество се осигурява от периода на генерираната последователност и от подходящо избраните начални стойности на параметрите в алгоритмичната схема, която я реализира.

Известно е [1, 2 и др.], че увеличаването на периода на генерираната последователност е необходимото условие за постигане на желаното качество, но то не е достатъчно. По тази причина много лансирани модели, например [3], водят до различни резултати в едно и също приложение [4].

2. Избор на метод

Основен критерий за избор на модел на БСВ е периодът на генерираната числова последователност. В това отношение един от доказалите се без конкуренция и горещо препоръчвани методи е този на Макларен-Марсальи [5].

Методът се основава на две независими случайни последователности $X : \{x_i\}$ и $Y : \{y_i\}$. В помощен буфер V се съхраняват k на брой числа от последователността X. Изходната последователност $Z : \{z_i\}$ се получава от избрани елементи на помощния буфер, т.е.

(2) $$z_i := V[j] .$$

Изборът на j-тия елемент от буфера V се осъществява с помощта на следната функция

(3) $$j = \lfloor k.y_i \rfloor .$$

Имайки предвид (1) и (3), то

(4) $$j \in [0, k-1] .$$

Съществуват две схеми за заместване на текущо избрания елемент от буфера V. Според първата [1], след присвояването (2) се извършва непосредственото заместване:

(5) $$V[j] := x_i ,$$

при което j-тият елемент от буфера V се подменя с текущото число от последователността X .

Втората схема [6] предлага заместването

(6) $$V[j] := x_{i+k} .$$

при което *j*-тият елемент от буфера *V* не се подменя с текущото число, а с числото, което последователността X генерира след *k* стъпки, т.е. x_{i+k} .

3. Предложения

Предложение А:

Основен момент в разгледания метод е динамичното изменение на съдържанието на буфера *V*. Случайният избор при четене и записване на неговите елементи се определя от комбинирането на качествата на използваните две алгоритмични схеми. Изборът на текущ елемент от буфера се постига чрез функцията (3), чието поведение е такова, че същата може да бъде определена като хеш функция от вида:

(7) $$h(R) = R \bmod \omega \ ,$$

където за *R* може да се приеме цялото число

(8) $$R = \lfloor (y_i)^{-1} \rfloor .$$

Тъй като за всяко y_i е необходимо да се определи коректното *j,* то следва че

(9) $$\omega = k \ .$$

Хеш функцията от вида (7) е твърде елементарна и нейната способност да “разсейва” избора на текущ елемент в пространството на буфера *V* е ограничена. Ето защо е желателно да се засили тази й способност, което ще се отрази положително на изходната последователност Z. Така за хеш функцията тук се предлага мултипликативната зависимост:

(10) $$j = h(R) = (P.R) \bmod k \ ,$$

където *P* е подходящо избрано голямо и взаимно просто с *k* число [7],[8].

Предложение Б:

Разглежданият тук метод няма никакви предварителни изисквания към генераторите на последователностите X и Y , включвани в неговата схема. Това ни дава основание в реализацията на алгоритмичната схема на метода Макларен-Марсальи да използваме за реализиране на случайната последователност Y , генераторът, предложен в [9]. Алгоритмичната схема на този генератор използва едномерен масив в качеството му на FIFO-буфер. Наличието на този буфер поражда идеята за неговото съвместно използване от двете алгоритмични схеми. Така след това предложение може да се твърди, че буферът *V* в реализацията на метода Макларен-Марсальи е FIFO-буфер.

4. Експеримент

Водени от разбирането, че не съществуват критерии, които да гарантират получаването на абсолютно случайна последователност, оценката на получаваните тук последователности е извършена само в смисъла на статистическия критерий на Колмогоров [6].

Убедени сме още, че при всеки краен набор от статистически критерии, последователността, която успешно ги удовлетворява, може да се окаже съвършено неприемлива за някои конкретни приложения. Така винаги ще съществува вероятност, че и най-щателните изследвания на една последователност не са в състояние да изявят абсолютно всички нейни свойства. В такъв случай е най-добре да сравняваме генераторите на случайни последователности, комбинирайки няколко основни статистически критерия с особените изисквания на конкретното приложение, които те следва да удовлетворяват.

При генерирането на случайни последователности чрез метода Макларен-Марсальи бяха реализирани и двете предложения, направени по-горе. Използваното в реализацията просто число е $P = 2147483647$. Поведението на така получения генератора беше сравнявано в условията на различни по обем извадки, вариращи от 500 до 1000000 числа, с генератора на последователността X . Получаваните оценки за критерия на Колмогоров при ниво на значимост $\varepsilon_0 = 0{,}05$ за двете извадки (съответно $K_{(3)}$ и $K_{(10)}$) бяха стойности, които удовлетворяваха отношението

$$K_{(3)} < K_{(10)}, \tag{11}$$

за повече от 70% от случаите. Това ни дава основание да твърдим, че предложената функция се е отразила положително на метода.

На основание на оценките за същия критерий беше проведено и сравнение на схемите за заместване на текущия елемент. Уверено може да изтъкнем, че схемата за заместване (6) е по-добра от схемата (5). Следва де признаем обаче, че скоростта на генериране по тази схема е значително по-ниска, което се дължи на пропусканите *k-1* числа преди достигане на числото x_{i+k}.

Литература

[1]. Knuth D.E., *The Art of Computer Programming. Seminumerical Algorithms, vol. 2,* 2nd ed., 1981, Addison Wesley, Reading, MA.

[2]. Niederreiter H., *Random Number Generation and Quasi-Monte Carlo Methods,* SIAM, Philadelphia, PA, 1992.

[3]. Matsumoto M., Nishimura T., *Mersenn Twister: a 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator,* ACM Transactions on Modeling and Computer Simulation, vol. 8, N 1, 1998, p. 3-30.

[4]. Тянев Д. С., *Компютърно генериране на случайни вектори с гарантиране на статистическите параметри,* списание “Електротехника и електроника”, брой 5-6, 1999 год., стр. 37-41.

[5]. MacLaren M.D., Marsaglia G., *Uniform Random Number Generators,* Journal of the Association for Computing Machinery, vol. 12, N 1, Jan. 1965.

[6]. Харин, Степанова, *Практикум на ЭВМ по математической статистике,* Минск, Издательство “Университетское”, 1987 год.

[7]. Knuth D.E., *The Art of Computer Programming. Seminumerical Algorithms, vol. 3,* 2nd ed., 1981, Addison Wesley, Reading, MA.

[8]. Graham R.L., Knuth D.E., Patashnik O., *Conckrete Mathematics – a Fondation for Computer Science,* 2nd ed., 1998, Addison Wesley, Reading.

[9]. Rabiner L.R., Gold B., *Theory and application of digital signal processing*, Prentice-Hall, 1975.

[10]. http://random.mat.sbg.ac.at/