

Zastosowanie wybranych generatorów pseudolosowych do symulacji naważania materiałów wsadowych

K. Wańczyk, E. Ziółkowski*

AGH Akademia Górniczo-Hutnicza im. S. Staszica w Krakowie,
Wydział Odlewnictwa, Katedra Inżynierii Procesów Odlewniczych,
30-059 Kraków, ul. Reymonta 23

*Kontakt korespondencyjny: e-mail: ez@agh.edu.pl

Otrzymano 20.11.2014; zaakceptowano do druku 12.12.2014

Streszczenie

W komputerowych programach do symulacji zdarzeń losowych stosuje się generatory liczb pseudolosowych. Przykładem takich zdarzeń może być symulacja naważania materiałów wsadowych dla pieców odlewniczych.

W artykule przedstawiono matematyczne zależności wybranych generatorów pseudolosowych, stosowanych w różnych kompilatorach i programach symulacyjnych. Przedstawiono także graficzną interpretację rozkładu liczb pseudolosowych, która może posłużyć do uproszczonej oceny jakości badanych generatorów. Całość uzupełnia definicja generowania wartości naważek materiałów wsadowych o założonej dyspersji.

Słowa kluczowe: generatory liczb pseudolosowych, symulacja naważania

1. Wprowadzenie

W programach komputerowych realizujących symulację przebiegu procesów technologicznych często występuje konieczność użycia liczb o charakterze losowym. Do losowania liczb stosuje się algorytmy numeryczne zwane generatorami liczb pseudolosowymi [1]. Generatory te obliczają wartości kolejnych liczb, uznawanych jako losowe, na podstawie obliczonej wartości liczby poprzedniej.

W programach komputerowych do symulacji procesów odlewniczych stosuje się generatory liczb pseudolosowych w celu wytworzenia wartości różnych wielkości fizycznych, „przypadkowo” odchylonych od założonych wartości teoretycznych.

2. Generatory liczb pseudolosowych

Algorytmy obliczania wartości pseudolosowych dzieli się zasadniczo na dwie grupy [1]:

- metody bezpośredniego generowania liczb,
- metody generowania „bit po bicie”, a następnie składanie ustalonej liczby bitów w liczbę finalną.

Typowe generatory liczb pseudolosowych wymagają ustalenia wartości początkowych we wzorze matematycznym do obliczania wartości wyznaczonej liczby. Zestaw takich wartości początkowych jest w literaturze często nazywany „ziarnem” lub „zarodkiem” generatora liczb pseudolosowych.

Wartości początkowe algorytmu mogą być zdefiniowane lub zależać od na przykład czasu odczytywanego z zegara czasu rzeczywistego w komputerze.

Generator liczb pseudolosowych powinien charakteryzować się założonym rozkładem statystycznym wyznaczanych liczb (równomierny, Gaussa lub inne) oraz tak zwanym długim okresem ciągu losowanych liczb. Dodatkowo algorytm generujący liczby powinien dawać podobne efekty niezależnie od wprowadzanych wartości początkowych.

Jednym z najstarszych algorytmów generujących liczby pseudolosowe jest liniowy generator kongruencyjny (ang. *LCG – linear congruential generator*), który można przedstawić w postaci

$$x_{i+1} = (a \cdot x_i + c) \bmod m \quad (1)$$

gdzie:

- x_{i+1} – aktualnie wyznaczana $i+1$ liczba pseudolosowa ($i=0, 1, 2, \dots$),
- x_i – poprzednio wyznaczona liczba pseudolosowa,
- a, c, m – całkowitoliczbowe parametry generatora,
- \bmod – operator wyznaczania reszty z dzielenia dwóch liczb całkowitych.

Algorytm realizujący losowanie według wzoru (1) jest powszechnie stosowany w wielu kompilatorach i translatorach różnych języków programowania oraz pakietach obliczeniowych. W tabeli 1 zestawiono wartości parametrów a , c i m dla wybranych narzędzi programistycznych [1÷4].

Tabela 1. Wartości parametrów dla wybranych generatorów LCG

Nazwa pakietu	Parametr generatora LCG		
	a	c	m
Numerical Recipes	1664525	1013904223	2^{32}
ANSI C	1103515245	12345	2^{32}
GNU Compiler Collection	69069	5	2^{32}
Borland C/C++	22695477	1	2^{32}
Microsoft Visual C++	214013	2531011	2^{32}
RANDU	65539	0	2^{31}
Borland DELPHI	134775813	1	2^{32}

Kompilator RAD Studio XE4 posiada dwie funkcje generujące liczby pseudolosowe:

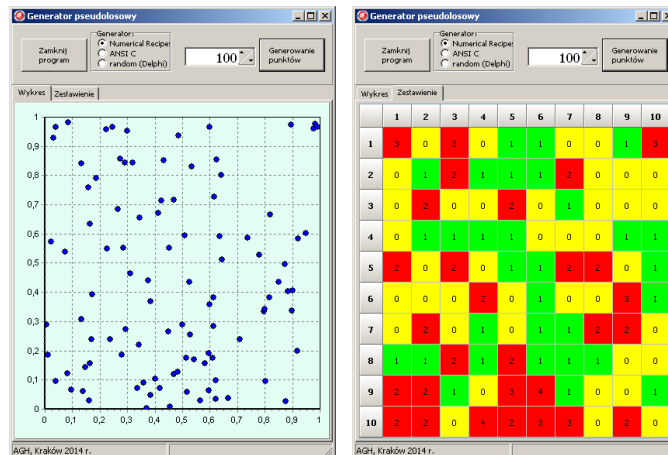
- *random[(Range: integer)]*, której zadaniem jest wyznaczenie liczby rzeczywistej z przedziału [0; 1], gdy ominięto parametr *Range*, lub liczbę całkowitą z przedziału [0; *Range*),
- *RandG(Mean, StdDev: integer)*, która wyznacza zbiór liczb pseudolosowych o rozkładzie Gaussa wokół liczby *Mean*, z odchyleniem standardowym *StdDev*.

Należy zaznaczyć, iż wywołanie w programie komputerowym, stworzonym za pomocą kompilatora Embarcadero RAD Studio funkcji *random* lub *RandG* zawsze będzie generować identyczny zbiór kolejnych liczb pseudolosowych, jeśli nie wykona się na początku tego programu funkcji *Randomize*, która ustawi „zarodka” na podstawie aktualnego stanu zegara czasu rzeczywistego w komputerze.

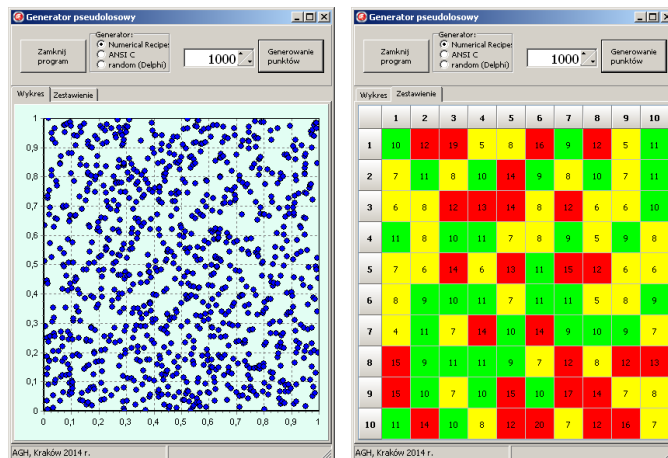
3. Testowanie rozkładów wybranych generatorów liczb pseudolosowych

W celu realizacji pseudolosowego naważania materiałów wsadowych dla pieców odlewniczych, w programie komputerowej symulacji procesu zestawiania wsadu, wykonano testowy program komputerowy, w którym zaimplementowano 3 algorytmy generujące liczby pseudolosowe. Dla każdego z wybranych algorytmów losowano zbiór liczb z przedziału [0; 1] o zadanej ilości tych liczb i przedstawiano w formie graficznej. Dodatkowo zliczano w poszczególnych podobszarach wylosowane liczby i klasyfikowano je jako mieszczące się w przedziale $\pm 10\%$ wartości nominalnej lub nie, co zobrazowano odpowiednimi kolorami na wykresach.

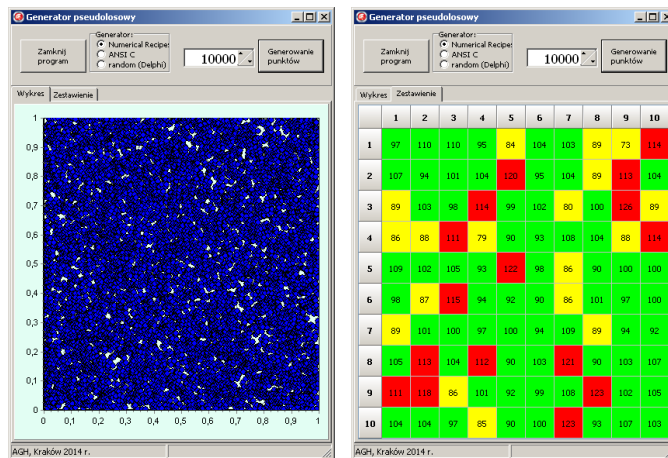
Na rysunku 1 przedstawiono widok okienek testowego programu losującego zbiór 100 liczb według algorytmu LCG ze współczynnikami przyjętymi w bibliotece procedur Numerical Recipes. Na rysunkach 2÷4 przedstawiono widok okienek dla 1000, 10000 i 100000 wylosowanych liczb dla tej samej funkcji.



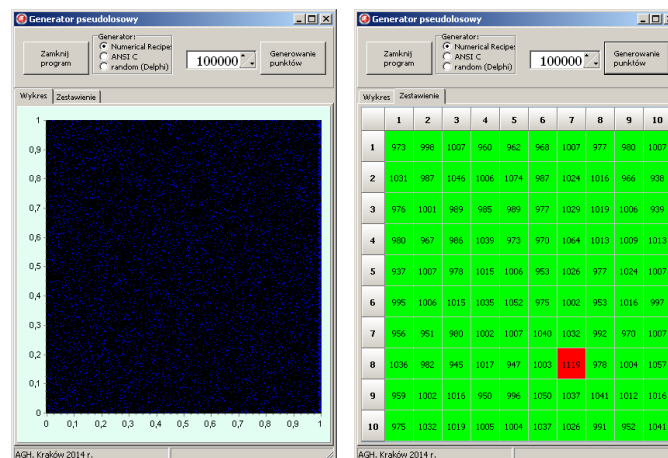
Rys. 1. Widok okienek programu po wylosowaniu zbioru 100 liczb za pomocą algorytmu LCG w bibliotece Numerical Recipes



Rys. 2. Widok okienek programu po wylosowaniu zbioru 1000 liczb za pomocą algorytmu LCG w bibliotece Numerical Recipes



Rys. 3. Widok okienek programu po wylosowaniu zbioru 10000 liczb za pomocą algorytmu LCG w bibliotece Numerical Recipes



Rys. 4. Widok okienek programu po wylosowaniu zbioru 100000 liczb za pomocą algorytmu LCG w bibliotece Numerical Recipes

Z analizy wyników zamieszczonych na rysunkach 1÷4 wynika, że im więcej liczb zostanie wylosowanych przez daną funkcję, tym lepsze będzie pokrycie obszaru, w którym graficznie zostanie pokazany zbiór wygenerowanych liczb. Reguła ta dotyczy także pozostałych badanych generatorów (ANSI C i *random* w Delphi). Oznacza to, że do celów symulacji przypadkowości wystąpienia zakłóceń można z pełnym powodzeniem zastosować wbudowane w stosowany kompilator funkcje generatora pseudolosowego.

4. Symulacja naważania materiałów wsadowych

Naważanie materiałów wsadowych sypkich można zrealizować na przykład za pomocą zależności

$$m_w = m_d + (m_g - m_d) \cdot \text{random}(), \text{ kg} \quad (2)$$

gdzie:

- m_w – masa naważki, kg,
- m_d, m_g – założona odpowiednio dolna i górna wartość naważki, kg,
- $\text{random}()$ – funkcja losująca liczbę z przedziału [0; 1).

W przypadku stosowania monodispersyjnych materiałów wsadowych kawałkowatych pseudolosową naważkę można wyznaczyć z zależności

$$m_w = \text{Round} \left(\frac{m_d + (m_g - m_d) \cdot \text{random}()}{m_k} \right) \cdot m_k, \text{ kg} \quad (3)$$

gdzie m_k , to masa pojedynczego kawałka naważanego materiału wsadowego, kg.

5. Podsumowanie

W komputerowych programach symulacyjnych naważanie materiałów wsadowych lub inne zdarzenia o charakterze przypadkowym można z powodzeniem zrealizować za pomocą generatorów liczb pseudolosowych typu LCG. Z przeprowadzonych testów wynika, że funkcje losujące, zdefiniowane w pakietach Numerical Recipes, ANSI C lub Embarcadero Delphi charakteryzują się bardzo dobrą równomiernością rozkładu i szybkością działania.

Literatura

- [1] Niederreiter, H. (1992). Random Number Generation and Quasi-Monte Carlo Methods. SIAM
- [2] www.embarcadero.com
- [3] www.visualstudio.com
- [4] www.nr.com

Usage of Selected Quasi-Random Generators in Simulations of Charge Materials Weighting

Streszczenie

Quasi-random incidents are implemented in computer simulations using random number generators. A simulation of charge materials weighting for foundry furnaces is an example of such occurrences.

The article presents mathematical equations for selected quasi-random generators, used in different compiler and simulation programs. Graphical interpretation of quasi-random distribution and simplified assessment of tested generators quality is included. The whole paper is supplemented by the definition of generation of charge materials weighting.