DOI: 10.18372/2310-5461.59.17944
УДК 004.056

Ю. В. Щербина, канд. техн. наук, доцент,
Національний університет «Одеська юридична академія»
orcid.org/0000-0003-3885-6747
e-mail: shcherbinayura53@gmail.com;

Н. Ф. Казакова, д-р техн. наук, професор,
Одеський державний екологічний університет
orcid.org/0000-0003-3968-4094
e-mail: kaz2003@ukr.net;

О. О. Фразе-Фразенко, канд. техн. наук, доцент,
Одеський державний екологічний університет
orcid.org/0000-0002-2288-8253
e-mail: frazenko@gmail.com;

О. А. Лаптєв, д-р техн. наук, снс,
Факультет інформаційних технологій
Київський національний університету імені Тараса Шевченка
orcid.org/0000-0002-4194-402X
e-mail: olaptiev@knu.ua;

А. В. Собчук доктор філософії
Державний університет телекомунікацій
orcid.org/0000-0003-3250-3799
e-mail: anri.sobchuk@gmail.com

ВИБІР ДЖЕРЕЛА ВИПАДКОВОСТІ ДЛЯ КОМП'ЮТЕРНОГО МОДЕЛЮВАННЯ

Вступ

В процесі проектування складних систем виконують натурні експерименти, які дають можливість оцінити їх поведінку у певних умовах експлуатації, але зазвичай, це вимагає витрат значних матеріальних і фінансових ресурсів, що не завжди можливо. Зважаючи на це, попередньо використовують комп'ютерне моделювання, яке засновується на програмній реалізації математичної моделі, що описує взаємодію складових частин системи між собою та зовнішнім середовищем експлуатації. Це дає можливість зробити попередній вибір параметрів системи та прогноз її поведінки в різних умовах.

На даний момент існує велика кількість математичних моделей, що дозволяють реалізувати комп'ютерну імітацію гіпотетичних реальних явищ або систем стохастичної природи.

Проблема полягає у тому, що спочатку спираючись на методи математичної статистики, необхідно зробити обґрунтований вибір такої моделі, яка найбільш точно описує поведінку реальної системи, що підлягає дослідженню [1].

Наступною проблемою є імітація поведінки системи як випадкового процесу. У якості джерела такої випадковості, зазвичай використовують генератори псевдовипадкових чисел, яких, також, на даний момент розроблено багато варіантів і, деякі з них вбудовані у відомі середовища програмування. Обраний генератор має забезпечувати задану адекватність поведінки реальної системи у певних умовах експлуатації і створеного комп'ютерного віртуального процесу [2].

Аналіз останніх досліджень і публікацій

Застосування генераторів випадкових чисел у ймовірнішому моделюванні систем потребує проведення великої кількості випробовувань. Для цього потрібні довгі послідовності випадкових чисел. Кількість випадкових чисел, які реалізуються для формування одного випробовування коливається у досить широких межах. Тому наявність простих та економічних способів формування послідовності випадкових чисел багато у чому визначає можливість практичного використання генераторів випадкових чисел. Генератори випадкових чисел використовують у моделюванні фактично з того моменту, як обчислювальна техніка стала доступною для наукових досліджень. Реалізація кожного такого генератора передбачає наявність джерела випадковості, у якості якого може бути використане джерело реального фізичного процесу, або рекурентний обчислювальний арифметичний процес. В обох випадках є свої проблеми. Генератори, що використовують реальні фізичні процеси, нестабільні в роботі і не забезпечують на виході стаціо-

нарний потік рівномірно розподілених випадкових чисел. Їх реалізація є дорогою і вимагає додаткової корекції вихідного потоку. Що стосується арифметичних генераторів, то вони, з теоретичної точки зору, взагалі, не здатні генерувати випадкових чисел. Послідовності на їх виходах циклічно повторюються з періодом T. Тим не менш, саме вони застосовуються у більшості випадків, зважаючи на те, що, зазвичай, їх період повторення багатократно перевищує величину конкретної реалізації.

Найбільш повне описання і оцінку досліджень арифметичних генераторів дав Д. Кнут у своїй роботі «Мистецтво програмування» Д. Кнут [2]. Основний висновок, який ним було зроблено, полягає в тому, що вони мало придатні для потреб моделювання, оскільки послідовності на їх виходах не задовольняють вимогам рівномірності розподілення чисел.

Подальше широке застосування випадкових чисел в криптографічних перетвореннях змусило дослідників створити нові генератори, які вирішували цю проблему, але таке рішення завжди забезпечувалось за рахунок значного ускладнення алгоритму [3]. Такі генератори, попри хорошу рівномірність вихідного потоку, забезпечують її на бінарному рівні, а для потреб моделювання, зазвичай необхідний потік рівномірно розподілених дійсних чисел. Щоб його сформувати використовують додаткові математичні перетворення, що руйнують таку рівномірність [4].

Враховуючи описані особливості сучасних генераторів ПВЧ, зусилля фахівців, що займаються моделюванням стохастичних процесів, націлені на пошук простих алгоритмів, що вимагають обмежених обчислювальних ресурсів і забезпечують задану рівномірність вихідного числового потоку методами його додаткових перетворень [5].

Зробивши аналіз існуючих рішень, прийшли до висновку, що невирішена проблема пошуку ефективного джерела випадковості для комп'ютерного моделювання стохастичних процесів.

Тому завдання формування послідовності випадкових чисел при математичному моделюванні е актуальним, а вибір простих та економічних способів формування послідовності випадкових чисел є актуальним науковим завданням.

Виклад основного матеріалу

Зважаючи на невирішеність проблеми пошуку ефективного джерела випадковості для комп'ютерного моделювання стохастичних процесів, головним завданням є пошук економічних, з точки зору використання обчислювальних ресурсів, генератора випадкових чисел. З урахуванням цього, ціллю роботи є розроблення способу постоброблення вихідної послідовності на виході простого генератора псевдовипадкових чисел.

Основні зусилля розробників націлені на створення генераторів ПВЧ, призначених для криптографічних перетворень. Відповідно і тестові пакети для перевірки якості числових послідовностей на їх виходах, виконують аналіз потоку на бінарному рівні [6, 7, 8]. Причина полягає в тому, головною криптографічною операцією є гамування. У ситуації з моделюванням, доводиться розділяти бінарну послідовність на байти та перетворювати бінарний потік у послідовність дійсних чисел, що змінюються у певних межах. Причому, як уже було вказано, рівномірність бітового потоку не переходить автоматично на потік дійсних чисел.

У роботі [2], показано що зробити простий ідеальний генератор практично не можливо, і тому ідея постоброблення як для генераторів реальних випадкових чисел, так і для генераторів ПВЧ залишається актуальною.

В окремих ситуаціях, коли немає необхідності регенерації числового потоку, у якості джерела випадковості, використовують реальні фізичні шумоподібні процеси, а щоб позбавитись проблем, породжених їх нестабільністю, також використовують методи постоброблення.

У роботі [9], вперше було використано постоброблення, запропоновано об'єднувати пару бітів, від різних генераторів за принципом: якщо біти співпадають (00 або 11), то біти скасовуються, комбінації бітів 01 відповідає 0-й вихідний біт, а комбінації 10 відповідає 1-й вихідний біт. Окремо було зроблене зауваження про труднощі генерації випадкових десяткових чисел.

У роботі [8], з'ясовано, що на даний момент можна виділити чотири наступні методики постоброблення:

– спеціальні прості коректори (Ad hoc simple correctors);

– відбілювання з використанням хеш-функцій (Whitening with hash functions);

– алгоритми видалення (Extractor algorithms);

– використання еластичних функцій (Resilient functions).

Загальною вимогою до всіх способів постоброблення є мінімізація ресурсів, для їх реалізації.

В комп'ютерному моделюванні найбільше розповсюдження отримали методи корекції та екстракції, які вилучають із загального числового потоку ту її частину, що має найбільшу ентропію так вказано у роботі [10].

В роботі [11] вводиться поняття зміщення вихідного розподілення чисел, під яким розуміється величина , яка обчислюється як:

$$e = \frac{1}{2}(P(i=1) - P(i=0)) \quad (1)$$

де $P(x_i = 1) = \frac{1}{2} + e$ і $P(x_i = 0) = \frac{1}{2} - e$, а x_1 це незалежні біти, що належать послідовності $x_1, x_2, ..., x_i$. Наявність цієї величини дозволяє оцінювати ступень несиметричності потоку на виході генератора. У разі бінарної послідовності ця величина має дорівнювати 0.5, а величина e повинна наближатись до нуля.

Щоб побудувати якісний генератор ПВЧ для потреб моделювання, необхідно, по-перше, виконати екстракцію чисел з вихідного потоку, що містять найбільшу ентропію, по-друге, реалізувати ефективний алгоритм такого відбору та математично його обґрунтувати.

У якості джерела випадковості краще обирати достатньо простий генератор, який не споживає занадто великих обчислювальних ресурсів. В роботах [2–5] описано велику кількість реалізацій генератора, побудованого на основі використання чисел Фібоначчі. Основу їх роботи можливо описати наступним виразом:

$$X_n = (X_{n-24} + X_{n-55}) \, mod \, m, \quad n \geq 55, \quad (2)$$

де m – парне число, а $X_0, ..., X_{54}$ довільні цілі числа, при чому не всі вони парні.

Довжина періоду послідовності на виході такого генератора складає $2^{q-1}(2^{55}-1)$, де q – це розрядність регістра мікропроцесора, а $m = 2q$. Враховуючи необхідність заповнення на початку роботи пам'яті генератора 55-тю почат-ковими числами і, того факту, що числа 24 і 55 – це числа Фібоначчі, його вихідні послідовності називають послідовностями Фібоначчі з запізненням. Генератори Фібоначчі вважали найкращими джерелами випадковості наприкінці минулого сторіччя через їх швидкодію та найбільший період повторення. Крім того, вони добре працюють з дійсними числами.

В роботі [12] запропоновано економічний, з точки зору обчислювальних ресурсів, генератор ПВЧ, відомий під назвою Xorshift. Пізніше було розроблення декілька його ефективних модифікацій. Цей генератор уявляє собою певну кількість регістрів зі зворотними зв'язками (LFSR), конфігурація яких визначається видом утворюючих поліномів невисокого ступеню.

Хороші показники генераторів типу Xorshift визначаються використанням елементарних комп'ютерних операцій (додавання і зсуви), які забезпечують простоту і ефективність реалізації необхідних перетворень над числами.

Суть алгоритму Xorshift полягає в тому, що він використовує набір усіх ненульових бінарних векторів 1×32 із Z, а f розглядається як лінійне перетворення над Z, виражене невиродженою бінарною матрицею T розміру 32×32. З урахуванням цього, можна стверджувати, що послідовність чисел на виході генератора буде виглядати як yT, yT^2, yT^3, ..., лише у разі, коли порядок T дорівнює 2^{32}-1 у групі не вироджених бінарних матриць розміром 32×32 і послідовність має період 2^{32}-1 [13]. Марсальєю було показано, що формування матричного добутку yT можна реалізувати, якщо порядок:

$$T = (I + a)(I + b)(I + c), \quad (3)$$

Також було показано, що для отримання максимально періоду необхідні матриці, які реалізують наступні типи зсувів:

$$y^{\uparrow} = y \ll 13; \; y^{\uparrow} = y \ll 17; \; y^{\uparrow} = y < 5.$$

Варіант реалізація описаного алгоритму мовою С може виглядати так:

```
x = 123456789
y = 362436069
z = 521288629
w = 88675123
t = x ^ ((x << 11) & 0xFFFFFFFF);
x = y;
y = z;
z = w;
w = (w ^ (w >> 19)) ^ (t ^ (t >> 8));
```

Рис. 1. Приклад реалізація описаного алгоритму мовою С

Після запуску генератора задаються початкові числа Після запуску генератора задаються початкові числа та w, що визначають внутрішній стан генератора. Кожне наступне число $w = \{N_0, N_1, ..., N_{31}\}$ формується як комбінація розрядів чисел x та попереднім значенням w так, як це показано у наведеному фрагменті коду. Після цього відбувається зсув чисел та $w \to z$. Саме ці зсуви і підвищують «випадковість» молодших розрядів чисел.

Враховуючи простоту і невибагливість його генератора Xorshift до ресурсних витрат, його можна вважати основним кандидатом на використання у якості джерела випадковості у комп'ютерному моделюванні.

Математичне моделювання показують, що запропонована Марсальєю остання модифікація генератора Xorshift128, не відповідає вимогам моделювання і потік випадкових чисел на його виході вимагає додаткового постоброблення через те, що нерівномірність чисел на його виході повністю переноситься на процес, що є ціллю моделювання.

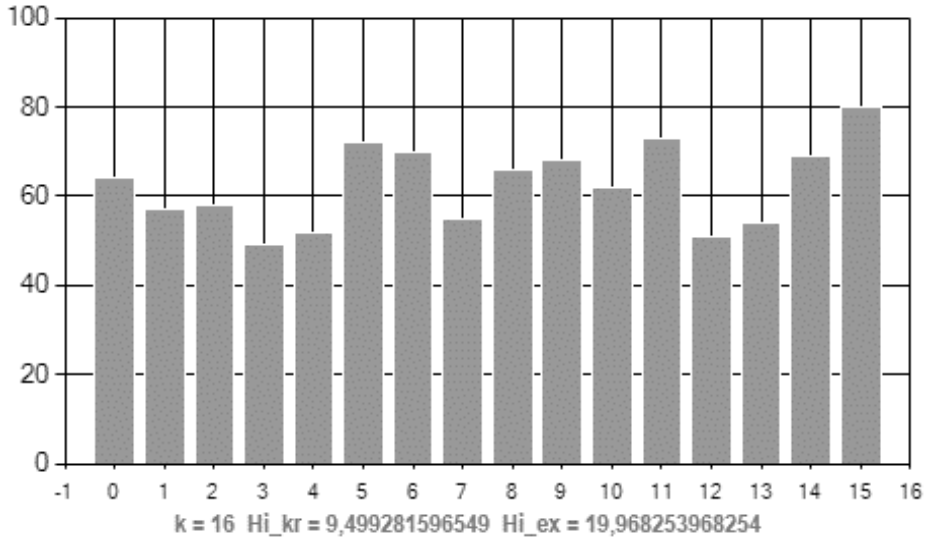

Рис. 2. Гістограма розподілення ПВЧ отриманих за допомогою функції Xorshift128

На рис. 2, наведено 16-сегментну гістограму послідовності у 1000 позитивних дійсних випадкових 4-байтових чисел, на виході генератора Xorshift128. Як видно, нерівномірність вихідної послідовності видно неозброєним оком.

Після проведеного дослідження пропонується ефективний метод постоброблення, який дає змогу суттєво зменшити зміщення вихідної послідовності. Його суть полягає в тому, що, враховуючи тип розподілення, розмір вибірки N, та кількості інтервалів k, обчислюється кількість чисел, що попадають у кожний інтервал гістограми. Для рівномірного розподілення ці величини мають співпадати і дорівнюють величині $\frac{N_i}{k}$. Математичне очікування величини m_i, що попадають у кожний сегмент гістограми $x_{\min} \le x_i < x_{\max}$, дорівнює величині $x_{\min} = \frac{(x_{\min} + x_{\max})}{2}$, а сума чисел S_i, які має попасти в i-й інтервал, буде приблизно дорівнювати величині $S_{\bar{i}} = N_{\bar{i}} m_i$. Якщо сума чисел, які дійсно попали в i-й сегмент гістограми S_i, то кожного разу вона буде відрізнятись від очікуваної величини $S_{\bar{i}}$.

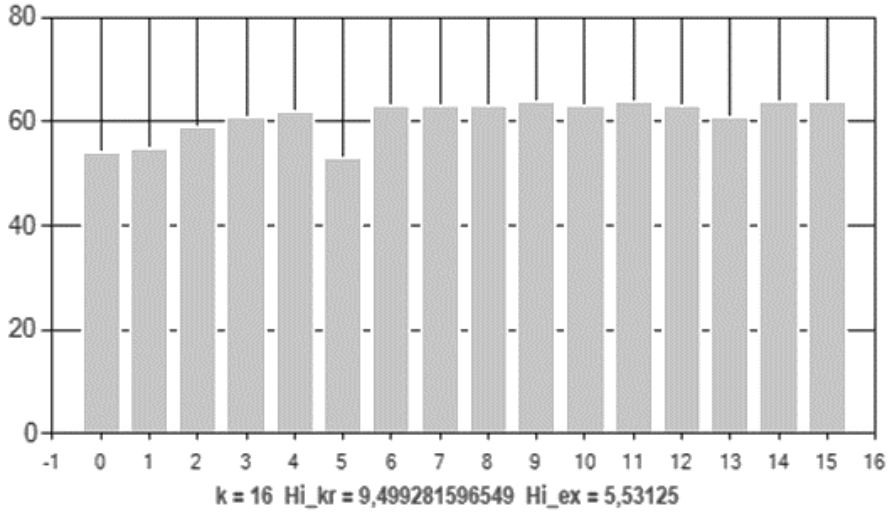

Рис. 3. Гістограма розподілення чисел з виходу генератора Xorshift128 після постоброблення

Описаний метод постоброблення виконує екстракцію “зайвих” чисел, що попадають в кожний i-й сегменти гістограми. На рис. 3 показано, що зміщення розподілення вихідного потоку зменшується до прийнятного рівню.

У проведеному математичному моделюванні оцінка рівномірності числових потоків виконувалась з використанням χ^2-критерію Пірсона. Це обумовлено тим , що показник χ^2-критерію

Пірсона легко обчислюється на основі параметрів гістограми.

Проведені дослідження доводять, що генератор Дж. Марсальї Xorshift128 має перевагу у виборі джерела випадковості для комп'ютерного моделювання, з точки зору обчислювальних ресурсів.

Оцінку та корекцію вихідного потоку краще виконувати на рівні числового потоку дійсних чисел, що безпосередньо використовуються у моделюванні. А розмір інтервалу розподілення варіаційного ряду, у разі несиметричних розподілень краще визначати за формулою Фрідмана-Діаконіса, яка дає меншу його величину, і це підвищує достовірність оцінки.

Висновки

У результаті проведеного дослідження комбінації простого алгоритму програмного способу формування псевдовипадкових чисел та способу додаткового їх постоброблення, які складають джерело випадковості комп'ютерної моделі стохастичного процесу, було запропоновано, з точки зору обчислювальних ресурсів, для проведення математичного моделювання, з метою формування псевдовипадкових чисел, використовувати генератор Дж. Марсальї Xorshift128.

Оцінку та корекцію вихідного потоку пропонується виконувати на рівні числового потоку дійсних чисел, що безпосередньо використовуються у моделюванні.

Оцінка якості числового потоку виконувалась з використанням показника χ^2 -критерію Пірсона вибір якого обумовлений саме тим, що його легко обчислити на основі параметрів гістограми. Розмір інтервалу розподілення варіаційного ряду, у разі несиметричних розподілень за доцільно визначати за формулою Фрідмана-Діаконіса, яка дає меншу його величину, і це підвищує достовірність оцінки.

ЛІТЕРАТУРА

[1] Mark A. Pinsky, Samuel Karlin. An Introduction to Stochastic Modeling, Fourth Edition., Academic Press, 2010. URL: https://faculty.ksu.edu.sa/sites/default/files/an_introd_to_stoch_modeling_4th_ed.pdf. (access date 15/07/2023)

[2] Knuth, D. E. The Art of Computer Programming. Volume 2. Seminumerical Algorithms. 3rd edition / D. E. Knuth. – Boston, Mass, USA : Addison-Wesley, Longman Publishing, 1997. – 762 p. ISBN 0-201-89683-4. URL: https://www.pdfdrive.com/art-of-computer-programming-knuth-vol-v2-e57538699.html. (access date 18/07/2023)

[3] Bruce Schneier. Applied Cryptography, Second Edition: Protocols, Algorthms, and Source Code in C. 1996. Wiley Computer Publishing, John Wiley & Sons, Inc. URL: https://dut.edu.ua/uploads/l_1134_27449793.pdf (access date 15/07/2023)

[4] Lemire Daniel. Fast Random Integer Generation in an Interval. ACM Transactions on Modeling and Computer SimulationVolume 29. Issue 1. Article No.: 3 pp 1–12 URL: https://arxiv.org/pdf/1805.10941.pdf (access date 15/07/2023)

[5] Tin Ni Ni Kyaw, Akio Tsuneda. Generation of chaos-based random bit sequences with prescribed auto-correlations by post-processing using linear feedback shift registers. DOI: 10.1587/nolta.8.224

[6] Mario Stipčević, True Random Number Generators. Open Problems in Mathematics and Computational Science, Open Problems in Mathematics and Computational Science 275–315) doi:10.1007/978-3-319-10683-0_12

[7] J. von Neumann. Various techniques for use in connection with random digits. Applied Math Series, Notes by G. E. Forsythe, in National Bureau of Standards, Vol. 12, 36–38, 1951. URL: https://mcnp.lanl.gov/pdf_files/nbs_vonneumann.pdf. (access date 15/07/2023)

[8] Trevisan L. Extractors and Pseudorandom Generators 1999. Journal of the ACM URL: http://theory.stanford.edu/~trevisan/pubs/extractor-full.pdf. (access date 25/07/2023)

[9] Siew-Hwee Kwok, Yen-Ling Ee, Guanhan Chew, Kanghong Zheng, Khoongming Khoo, Chik-How Tan. A Comparison of Post-Processing Techniques for Biased Random Number Generators. WISTP 2011: Information Security Theory and Practice. Security and Privacy of Mobile Devices in Wireless Communication. PP. 175–190. URL: https://link.springer.com/content/pdf/10.1007/978-3-540-74619-5_9.pdf. (access date 15/07/2023)

[10] Yurii Shcherbina, Nadiia Kazakova, Oleksii Fraze-Frazenko. Using the Xorshift generator to simulate stochastic processes. Processing, transmission and security of information – 5 December 2022. URL: https://www.engineerxxi.ath.eu/publikacja/processing-transmission-and-security-of-information-2022/.

[11] George Marsaglia. Xorshift RNGs. 2003. DOI:10.18637/jss.v008.i14.

[12] Laptiev, O., Sobchuk, V.,Subach, I., Barabash, A., Salanda, I. The Method of Detecting Radio Signals Using the Approximation of Spectral Function. CEUR Workshop Proceedings, 2022, 3384, pp. 52– 61.

[13] Valentyn Sobchuk , Iryna Zelenska and Oleksandr Laptiev. Algorithm for solution of systems of singularly perturbed differential equations with a differential turning point. Bulletin of the Polish Academy of Sciences Technical Sciences, Vol.71, No 3, 2023, Article number: e145682. DOI: 10.24425/bpasts. 2023.145682

Щербина Ю. В., **Казакова Н. Ф.**, **Фразе-Фразенко О. О.**, **Лаптєв О. А.**, **Собчук А. В.**

ВИБІР ДЖЕРЕЛА ВИПАДКОВОСТІ ДЛЯ КОМП'ЮТЕРНОГО МОДЕЛЮВАННЯ

У статті розглядаються проблеми вибору джерела випадковості для комп'ютерного моделювання стохастичних процесів, що використовується для дослідження характеристик потоків подій безпеки в розподілених комп'ютерних мережах, на етапі проектування складних автоматизованих систем та процесів, які мають місце в управлінні виробництвом та інфраструктурними об'єктами. Складовою частиною комп'ютерної моделі є джерело випадковості, яке формує рівномірно розподілений потік випадкових цілих або дійсних чисел. Воно повинно формувати потік рівномірно розподілених чисел і, в той же час, бути економічним з точки зору обчислювальних ресурсів. В роботі надано аналіз простих генераторів псевдовипадкових чисел, в алгоритмі яких використовуються прості комп'ютерні операції. До складу таких генераторів віднесені генератор Фібоначчі з запізненням та запропонований Дж. Марсальєю генератор Xorshift128. Відзначено, що будь-яка нерівномірність розподілення чисел на виході генератора, суттєво впливає на якість процесу, який підлягає моделюванню. На основі результатів проведених досліджень існуючих способів постоброблення вихідних послідовностей, зроблено висновок про те, для забезпечення ефективності алгоритму формування потоку рівномірно роз-поділених псевдовипадкових чисел, процедури додаткового оброблення повинні бути достатньо економічними з точки зору задіяних методів обчислення. Оцінка нерівномірності розподілення числового потоку виконувалась з використанням показника хі-квадрат Пірсона. Для корекції вихідного числового потоку запропоновано і обґрунтовано спосіб екстракції з нього тої його частини, ентропія якої найбільша. Також, обґрунтовано параметри гістограми, що дають хороші результати оцінки вихідного розподілення. Показано, що комбінація простого і економічного генератора псевдовипадкових чисел в сукупності з постобробленням дає хороші результати при мінімальних обчислювальних ресурсах

Ключові слова: Моделювання, стохастичний процес, генератор Xorshift, метод оберненої функції, критерій хі-квадрат Пірсона, постобробка чисельного потоку.

Shcherbyna Y., Kazakova N., Fraze-Frazenko O., Laptiev O., Sobchuk A.

SELECTION OF RANDOMNESS SOURCE FOR COMPUTER SIMULATION

The main purpose of evolutionary optimization is to find a combination of parameters (independent variables) that would help maximize or minimize the qualitative, quantitative, and probabilistic characteristics of the problem. Recently, integrated optimization methods have become very common, borrowing the basic principles of their work from wildlife. Researchers are experimenting with different types of representations, for example, evolutionary and genetic algorithms use selection methods and genetic operators. A large number of algorithms based on the swarm method are known.

The artificial bee colony is an optimization method that mimics the behavior of bees, a specific application of cluster intelligence, the main feature of which is that it does not need to understand specific information about the problem, you just need to optimize the problem. Comparing inferiority with the help of the local optimization behavior of each person with an artificial bee finally leads to the appearance in the group of a global optimal value with a higher rate of convergence.

The paper considers the method of solving the optimization problem based on modeling the behavior of the bee colony. Description of the model of the behavior of intelligence agents and forage agents, search mechanisms, and selection of positions in a given neighborhood. The general structure of the optimization process is given. Graphical results are also presented, which prove the possibility of the bee colony method to optimize the results, i.e. from all multiple sources of information, the bee colony method by optimization can significantly limit the number of information sources, identify a narrow range of sources that may be false information. Which in the future will allow you to more accurately identify sources with false information and block them.

Keywords: Modeling, stochastic process, Xorshift generator, inverse function method, Pearson's chi-square test, numerical stream post-processing.

Стаття надійшла до редакції 02.08.2023 р.
Прийнято до друку 11.10.2023 р.