

УДК 519.6

# Приближённый итерационный алгоритм моделирования негауссовских векторов с заданными одномерными распределениями и ковариационной матрицей\*

М.С. Акентьева<sup>1,2</sup>, Н.А. Каргаполова<sup>1,2</sup>, В.А. Огородников<sup>1,2</sup>

<sup>1</sup>Институт вычислительной математики и математической геофизики Сибирского отделения Российской академии наук, просп. Акад. Лаврентьева, 6, Новосибирск, 630090

<sup>2</sup>Новосибирский национальный исследовательский государственный университет (НГУ), ул. Пирогова, 2, Новосибирск, 630090

E-mails: m.akenteva@ngs.ru (Акентьева М.С.), nkargapolova@gmail.com (Каргаполова Н.А.), ova@osmf.sccc.ru (Огородников В.А.)

Английская версия этой статьи печатается в журнале “Numerical Analysis and Applications” № 4, Vol. 16, 2023.

Акентьева М.С., Каргаполова Н.А., Огородников В.А. Приближённый итерационный алгоритм моделирования негауссовских векторов с заданными одномерными распределениями и ковариационной матрицей // Сиб. журн. вычисл. математики / РАН. Сиб. отд.-ние.— Новосибирск, 2023.— Т. 26, № 4.— С. 345–356.

В работе предложен новый итерационный метод моделирования негауссовских случайных векторов с заданными одномерными распределениями и ковариационной матрицей. Проведено сравнение предложенного алгоритма с другим итерационным алгоритмом моделирования негауссовских векторов, основанным на переупорядочивании выборки независимых случайных величин с заданными одномерными распределениями. Численные исследования показали, что алгоритмы являются фактически эквивалентными в плане точности воспроизведения заданной ковариационной матрицы при моделировании, однако предложенный алгоритм оказался более эффективным по использованию памяти и, во многих случаях, менее трудоёмким.

DOI: 10.15372/SJNM20230401

EDN: VGQRFJF

**Ключевые слова:** негауссовские случайные процессы, численное стохастическое моделирование, одномерные распределения, ковариационная матрица.

Akenteva M.S., Kargapolova N.A., Ogorodnikov V.A. An approximate iterative algorithm for modeling of non-Gaussian vectors with given marginal distributions and a covariance matrix // Siberian J. Num. Math. / Sib. Branch of Russ. Acad. of Sci.—Novosibirsk, 2023.— Vol. 26, № 4.— P. 345–356.

A new iterative method for the modeling of non-Gaussian random vectors with given marginal distributions and covariance matrix is proposed in this paper. The algorithm is compared with another iterative algorithm for the modeling of non-Gaussian vectors, which is based on reordering a sample of independent random variables with given marginal distributions. Our numerical studies show that both algorithms are equivalent in terms of the accuracy of reproducing the given covariance matrix, but the proposed algorithm turns out to be more efficient in terms of memory usage and, in many cases, is faster than the other one.

---

\*Работа выполнена при поддержке гранта Российского научного фонда (проект № 21-71-00007) <https://rscf.ru/project/21-71-00007/>.

**Keywords:** *non-Gaussian stochastic processes, stochastic modeling, marginal distributions, covariance matrix.*

---

## 1. Введение

С активным развитием вычислительной техники в математическом моделировании помимо детерминированного стал широко использоваться стохастический подход к построению моделей. При численном стохастическом моделировании реальный исследуемый процесс описывается некоторым случайным процессом. Затем генерируется большое количество реализаций этого случайного процесса, которые используются для изучения его различных вероятностных характеристик.

Следует отметить, что случайный процесс определяется всеми своими конечномерными распределениями. Однако на практике, как правило, известны не все эти распределения. В связи с этим приходится моделировать процесс, у которого известна лишь часть вероятностных характеристик. Такими характеристиками чаще всего являются одномерные распределения процесса и его корреляционная функция. Эти характеристики наиболее надёжно оцениваются по выборке реальных данных, объём которой обычно ограничен и может быть достаточно малым.

Задача моделирования негауссовских случайных процессов является более сложной по сравнению с аналогичной задачей для гауссовских процессов. В то же время многие процессы, наблюдаемые в природе, являются, очевидно, негауссовскими. Таким образом, представляется актуальной задача построения эффективных алгоритмов моделирования негауссовских случайных процессов с заданными одномерными распределениями и корреляционной функцией. Перечислим некоторые известные алгоритмы моделирования негауссовских случайных процессов.

Поскольку гауссовские случайные процессы являются хорошо изученным классом случайных процессов, и для их моделирования разработано достаточное количество эффективных алгоритмов, представляется разумной идея строить алгоритмы моделирования негауссовских процессов на основе различных преобразований гауссовских процессов. Так, например, широко используется метод обратной функции распределения [1]. В рамках этого метода негауссовский процесс с заданной корреляционной функцией получается в результате специального нелинейного преобразования вспомогательного гауссовского процесса. Корреляционная функция вспомогательного процесса выбирается таким образом, чтобы преобразованный процесс имел заданную корреляционную функцию. В случае, когда требуемый вспомогательный гауссовский процесс существует, метод обратной функции распределения позволяет точно воспроизвести заданные одномерные распределения и корреляционную функцию случайного процесса. В противном случае можно попытаться подобрать гауссовский процесс таким образом, чтобы корреляционная функция моделируемого процесса оказалась близкой к заданной. Способы выбора корреляционной функции (или соответствующей спектральной плотности в стационарном случае) такого гауссовского процесса предлагаются в работах [5–7]. В [9] рассмотрен другой алгоритм моделирования негауссовского процесса. Вспомогательный гауссовский процесс определяется неизвестными параметрами, а нелинейное преобразование принадлежит некоторому заданному параметрическому семейству функций. Все неизвестные параметры находятся методом максимального правдоподобия с использованием данных наблюдений.

Существуют методы моделирования негауссовских процессов, не опирающиеся на моделирование гауссовских процессов. Так, в [3, 8] предлагаются алгоритмы, основанные на разложении Карунена–Лозва. В [10] представлен итерационный алгоритм, принимающий в качестве начальных данных выборку случайных векторов с заданными одномерными распределениями и независимыми компонентами. На каждой итерации алгоритма элементы выборки переставляются специальным образом так, чтобы её выборочная ковариационная матрица становилась ближе к заданной.

В данной работе предлагается новый итерационный метод моделирования негауссовских случайных векторов с заданными одномерными распределениями и ковариационной матрицей. На каждой итерации этого метода используется линейное преобразование выборки из работы [10], а также нелинейное преобразование, похожее на преобразование из метода обратных функций распределения. Проводится сравнение предложенного алгоритма с итерационным алгоритмом из [10]. Представлены результаты численных исследований точности воспроизведения ковариационной матрицы рассмотренными алгоритмами. Обсуждается влияние специального выбора начальных данных на скорость сходимости методов.

## 2. Итерационный алгоритм моделирования негауссовских векторов, основанный на перестановке элементов выборки

Рассматриваемый далее алгоритм моделирования случайных векторов был в настоящем виде впервые сформулирован в работе [10], однако в [2] отмечается, что последовательность действий, выполняемая на каждой итерации, была ранее использована в [4] в качестве самостоятельного алгоритма. Вопросы, касающиеся теоретического обоснования сходимости алгоритма из работы [10], остаются открытыми.

Приведём краткое описание приближённого метода моделирования случайных векторов с заданными одномерными распределениями и ковариационной матрицей, предложенного в работе [10]. Входными данными алгоритма является выборка случайных векторов, компоненты которых независимы между собой и имеют заданные одномерные распределения. За несколько итераций алгоритма эта выборка путём перестановки её элементов преобразуется в выборку случайных векторов с корреляциями, близкими к заданным. Одномерные распределения при этих преобразованиях не меняются. Нужно заметить, что моделирование случайных векторов с заданными одномерными распределениями и независимыми компонентами является относительно простой задачей, поскольку существует достаточное количество алгоритмов моделирования случайных величин с различными распределениями.

Итак, пусть требуется моделировать случайный вектор  $\vec{X}_n = (X_1, \dots, X_n)^\top$ , компонента  $X_j$  которого имеет функцию одномерного распределения  $F_j(x)$ ,  $j = 1, \dots, n$ , а ковариационная матрица  $\vec{X}_{(n)}$  равна  $R$ . Приближённый алгоритм решения этой задачи состоит в следующем.

**Алгоритм 1.** Пусть  $\vec{X}_n^{(0)} = (X_1^{(0)}, \dots, X_n^{(0)})^\top$  — случайный вектор с независимыми компонентами, и пусть компонента  $X_j^{(0)}$  имеет функцию одномерного распределения  $F_j(x)$ ,  $j = 1, \dots, n$ .

1. Моделируются  $M$  независимых реализаций каждой случайной величины  $X_j^{(0)}$ ,  $j = 1, \dots, n$ , с функцией распределения  $F_j(x)$ , и из полученных реализаций формируется матрица  $Y = (\vec{y}^{(1)}, \dots, \vec{y}^{(n)})$  размерности  $M \times n$ . Здесь через  $\vec{y}^{(j)}$ ,  $j = 1, \dots, n$ ,

обозначен  $j$ -й столбец матрицы  $Y$ . Этот столбец состоит из  $M$  независимых реализаций случайной величины  $X_j^{(0)}$ . В строках матрицы  $Y$  находятся реализации случайного вектора, компоненты которого независимы и имеют одномерные распределения  $F_1(x), \dots, F_n(x)$ .

2. Для ковариационной матрицы  $R$  находится разложение Холецкого:

$$R = CC^T.$$

3. Используя строки матрицы  $Y$ , оценивается выборочная ковариационная матрица  $\tilde{R}$  размерности  $n \times n$ . Здесь и в дальнейшем используется стандартная несмещённая оценка ковариационной матрицы случайного вектора.

4. Для ковариационной матрицы  $\tilde{R}$  находится разложение Холецкого:

$$\tilde{R} = \tilde{C}\tilde{C}^T.$$

5. С использованием полученных разложений строится новая матрица  $\tilde{Y}$ :

$$\tilde{Y} = Y(\tilde{C}^T)^{-1}C^T.$$

6. Элементы каждого столбца  $\vec{y}^{(j)}$ ,  $j = 1, \dots, n$ , переставляются в том же порядке по величине, в каком они стоят в столбце  $\vec{y}^{(j)}$ , т.е. для каждого  $i = 1, \dots, M$  находится в столбце  $\vec{y}^{(j)}$   $k$ -й по величине элемент. Обозначим его через  $\tilde{y}_k^{(j)}$ . Пусть  $l$  — номер строки, в которой он стоит. Затем находится  $k$ -й по величине элемент в столбце  $\vec{y}^{(j)}$ , который обозначим  $y_k^{(j)}$ . Элемент  $y_k^{(j)}$  переставляется в строку  $l$ , т.е. в ту строку, в которой стоит  $k$ -й по величине элемент столбца  $\vec{y}^{(j)}$ .
7. Шаги 3–7 повторяются до тех пор, пока не будет выполнено какое-либо условие остановки. Например, пока не выполнится условие  $\|R - \tilde{R}\| < \varepsilon$ , где  $\varepsilon$  — некоторая заданная малая величина.

Заметим, что при перестановке элементов столбца  $\vec{y}^{(j)}$  на шаге 6 распределение элементов этого столбца не изменяется. Легко показать, что оценка ковариационной матрицы выборки случайных векторов, находящихся в строках матрицы  $\tilde{Y}$ , равна заданной ковариационной матрице  $R$ . Поэтому на шаге 6, переставляя элементы столбцов матрицы  $Y$  в таком же порядке, в котором они стоят в соответствующих столбцах матрицы  $\tilde{Y}$ , из эвристических соображений можно ожидать, что и оценка ковариационной матрицы  $\tilde{R}$  строк матрицы  $Y$  будет после таких перестановок более близкой к заданной матрице  $R$ . Таким образом, в результате работы алгоритма в строках матрицы  $Y$  будут сформированы реализации случайного вектора с заданными одномерными распределениями и выборочной ковариационной матрицей, близкой к заданной матрице  $R$ .

Практика показывает, что для получения достаточно хорошей точности воспроизведения заданной ковариационной матрицы зачастую достаточно всего нескольких итераций алгоритма 1.

### 3. Новый приближённый итерационный алгоритм моделирования негауссовских векторов

В данной работе предлагается новый приближённый итерационный алгоритм моделирования случайных векторов  $\vec{X}_n = (X_1, \dots, X_n)^T$  размерности  $n$  с заданными одномерными распределениями  $F_j(x)$ ,  $j = 1, \dots, n$ , и ковариационной матрицей  $R$ . Предложенный алгоритм для своей работы требует в два раза меньше памяти, чем алгоритм 1,

и, как показывают численные эксперименты, во многих случаях является более быстроедейственным. Начальными данными алгоритма является выборка объёма  $M$  случайных векторов размерности  $n$ , компоненты которых имеют равномерное распределение на  $(0, 1)$ . Эти начальные данные представлены в виде матрицы размерности  $M \times n$ . На каждой итерации алгоритма используются два специальных преобразования выборки случайных векторов: первое позволяет преобразовать выборку таким образом, чтобы её выборочная ковариационная матрица стала равна заданной матрице  $R$ , а второе — получить требуемые одномерные распределения компонент случайных векторов (выборочная ковариационная матрица после этого преобразования будет уже отличной от матрицы  $R$ ). Алгоритм можно сформулировать в следующем виде.

### Алгоритм 2.

1. На первом шаге алгоритма строится матрица начального приближения  $Y^{(0)} = \left( y_{ij}^{(0)} \right)$  размерности  $M \times n$ , элементы которой представляют собой независимые реализации случайной величины с равномерным распределением на  $(0, 1)$ .

2. Для ковариационной матрицы  $R$  находится разложение Холецкого:

$$R = CC^T.$$

3. На каждой итерации алгоритма ( $k = 0, 1, \dots$ ) выполняются следующие операции:

- 3.1 Используя строки матрицы  $Y^{(k)}$ , оценивается выборочная ковариационная матрица  $\tilde{R}^{(k)}$  размерности  $n \times n$ .

- 3.2 Для ковариационной матрицы  $\tilde{R}^{(k)}$  находится разложение Холецкого:

$$\tilde{R}^{(k)} = \tilde{C}^{(k)} \left( \tilde{C}^{(k)} \right)^T.$$

- 3.3 С использованием полученных разложений строится новая матрица  $\tilde{Y}^{(k)} = \left( \tilde{y}_{ij}^{(k)} \right)$ :

$$\tilde{Y}^{(k)} = Y^{(k)} \left( \left( \tilde{C}^{(k)} \right)^T \right)^{-1} C^T.$$

Аналогично алгоритму 1, выборочная ковариационная матрица строк  $\tilde{Y}^{(k)}$  равна заданной матрице  $R$ .

- 3.4 Используя компоненты  $j$ -го столбца матрицы  $\tilde{Y}^{(k)}$ , оцениваем эмпирическую функцию распределения  $\hat{F}_j^{(k)}(x)$ .

- 3.5 Находим очередное приближение  $Y^{(k+1)}$  по формуле

$$y_{ij}^{(k+1)} = F_j^{-1} \left( \hat{F}_j^{(k)} \left( \tilde{y}_{ij}^{(k)} \right) \right).$$

- 3.6 Шаги 3.1–3.6 повторяются до тех пор, пока не будет выполнено какое-либо условие остановки. Например, пока не выполнится условие  $\|R - \tilde{R}^{(k)}\| < \varepsilon$ .

В результате работы алгоритма в строках матрицы  $Y_k$ ,  $k = k_{\text{end}}$ , будут сформированы реализации случайного вектора, одномерные распределения и выборочная ковариационная матрица которых будут близки к заданным.

Преобразование, используемое на шаге 3.5, похоже на нелинейное преобразование из метода обратной функции распределения. Отличие состоит в том, что вместо функции стандартного нормального распределения в алгоритме 2 используется эмпирическая

функция распределения, поэтому одномерные распределения моделируемых случайных векторов воспроизводятся лишь приближённо. Чем точнее оценка эмпирической функции распределения, тем точнее будут воспроизводиться одномерные распределения. Поскольку по выборке большего объёма эмпирическую функцию можно оценить с большей точностью, точность воспроизведения одномерных распределений также будет увеличиваться.

Следует отметить, что на первом шаге алгоритма 2 можно использовать выборку из какого-либо другого распределения. Вопрос о влиянии выбора начального распределения на результат работы алгоритма требует отдельного исследования.

#### 4. Численное исследование алгоритмов

Прежде всего отметим, что реализации векторов, полученные как с помощью алгоритма 1, так и с помощью алгоритма 2, являются зависимыми, однако численные эксперименты показывают, что с увеличением количества реализаций корреляции между компонентами разных векторов уменьшаются. В таблице 1 приведён пример оценки коэффициента корреляции между первой компонентой первой реализации вектора и первой компонентой второй реализации.

**Таблица 1.** Оценка коэффициента корреляции между первой компонентой первого вектора в выборке и первой компонентой второго вектора

Кол-во реализаций $M$	Алгоритм 1	Алгоритм 2
10	-0.11	-0.11
$10^2$	-0.042	-0.010
$10^3$	-0.0032	-0.0036

В этом численном эксперименте моделировались стационарные случайные векторы размерности 3 с экспоненциальными одномерными распределениями (плотность  $f(x) = e^{-x}$ ) и корреляционной матрицей

$$R = \begin{pmatrix} 1 & 0.904 & 0.818 \\ 0.904 & 1 & 0.904 \\ 0.818 & 0.904 & 1 \end{pmatrix}.$$

##### 4.1. Сравнение алгоритмов по требуемым затратам времени и памяти

Имеет место следующая простая лемма.

**Лемма.** Алгоритм 2 требует для своей работы в два раза меньше памяти по сравнению с алгоритмом 1.

Справедливость леммы становится очевидной, если сравнить шаги 5 и 6 алгоритма 1 и шаги 3.3 и 3.5 алгоритма 2. При программной реализации алгоритма 1 требуется одновременно хранить в памяти матрицы  $Y$  и  $\tilde{Y}$ . В алгоритме 2 матрица  $\tilde{Y}^{(k)}$  может быть записана на место матрицы  $Y^{(k)}$ .

Трудоёмкость, точность воспроизведения ковариаций и другие свойства алгоритмов 1 и 2 будут исследованы численно. Для проведения численных экспериментов рассмотрим 4 различных одномерных распределения с плотностями  $f_1(x), \dots, f_4(x)$ :

- экспоненциальное распределение  $f_1(x) = \lambda \exp(-\lambda x)$ ,  $\lambda = 1$ ,

- бета-распределение  $f_2(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)}$ ,  $\alpha = 10.6$ ,  $\beta = 13.06$ ,
- логнормальное распределение  $f_3(x) = \frac{1}{\sqrt{2\pi\sigma x}} \exp\left(-\frac{[\ln x - \mu]^2}{2\sigma^2}\right)$ ,  $\mu = 0$ ,  $\sigma^2 = 1$ ,
- равномерное распределение  $f_4(x) = 1$ , если  $x \in (0, 1)$ ,  $f(x) = 0$  иначе;

и 4 корреляционные функции:

- $r_1(x_1, x_2) = \exp\{-|x_1 - x_2|\}$ ,
- $r_2(x_1, x_2) = \exp\{-(x_1 - x_2)^2\}$ ,
- $r_3(x_1, x_2) = \exp(-|x_1 - x_2|) \cos 4\pi(x_1 - x_2)$ ,
- $r_4(x_1, x_2) = \exp\{-(x_1 - x_2)^2\} \cos 4\pi(x_1 - x_2)$ .

Таким образом, получается 16 различных пар  $r_i, f_j$ .

В каждом численном эксперименте будем моделировать стационарный случайный процесс с корреляционной функцией  $r_i$  и плотностью одномерного распределения  $f_j$  в узлах сетки на интервале  $[0, 10)$  с шагом  $h = 0.1$ . Случайный процесс на рассматриваемом интервале моделируется в виде случайного вектора с заданными одномерным распределением и корреляционной матрицей. Для моделирования используются алгоритмы 1 и 2.

По корреляционным функциям  $r_i, i = 1, \dots, 4$ , строятся корреляционные матрицы  $R_i$ . Отметим, что матрицы, построенные по корреляционным функциям  $r_2$  и  $r_4$  на рассматриваемой сетке, являются плохо обусловленными. В данной работе в численных экспериментах матрицы  $R_2$  и  $R_4$  были заменены на близкие к ним корреляционные матрицы  $\hat{R}_2 = (1 - \varepsilon)R_2 + \varepsilon E$  и  $\hat{R}_4 = (1 - \varepsilon)R_4 + \varepsilon E$ , где  $E$  — единичная матрица соответствующей размерности, а  $\varepsilon$  выбрано равным  $10^{-5}$ . Далее для единообразия обозначений будем считать, что  $R_2 = \hat{R}_2, R_4 = \hat{R}_4$ .

Время работы алгоритма 1 практически не зависит от одномерных распределений моделируемого вектора. Быстродействие алгоритма 2, напротив, существенно зависит от трудоёмкости вычисления обратной функции одномерного распределения, которое требуется моделировать. Как показывает табл. 2, для всех рассмотренных плотностей распределения, кроме бета-распределения, алгоритм 2 оказался более быстродейственным, чем алгоритм 1.

**Таблица 2.** Среднее время (в секундах) выполнения одной итерации для случая корреляционной функции  $r_1$

Плотность распределения	Алгоритм 1	Алгоритм 2
$f_1$	5.2	2.9
$f_2$	5.7	58.0
$f_3$	5.4	3.3
$f_4$	5.8	2.6

#### 4.2. Сравнение алгоритмов по точности воспроизведения корреляционной матрицы

В табл. 3 приведены числа обусловленности корреляционных матриц  $R_i, i = 1, \dots, 4$ , а в табл. 4 и 5 — минимальное значение относительной ошибки  $\frac{\|R_i - \hat{R}_i\|}{\|R_i\|}$ , полученной за 10 итераций работы алгоритмов для различных комбинаций корреляционных матриц  $R_i$  и

одномерных плотностей распределений  $f_j$  ( $i, j = 1, \dots, 4$ ). Здесь  $\tilde{R}_i$  — корреляционная матрица, оценённая по модельным реализациям (здесь и всюду далее используется евклидова норма матрицы). Размерность  $n$  матрицы  $R_i$  во всех численных экспериментах меньше  $M$ , что обеспечивает положительную определённую  $R_i$ . В экспериментах, если не указано иное, моделировалось  $M = 10^5$  реализаций случайных векторов. Заметим, что относительная ошибка не всегда убывает монотонно с ростом номера итерации, и минимум не обязательно достигается на последней итерации.

**Таблица 3.** Числа обусловленности корреляционных матриц

$r_i(x_1, x_2)$	$\text{cond}(R_i)$
$\exp\{- x_1 - x_2 \}$	374
$\exp\{-(x_1 - x_2)^2\}$	$1.73 \times 10^6$
$\exp(- x_1 - x_2 ) \cos 4\pi(x_1 - x_2)$	123
$\exp\{-(x_1 - x_2)^2\} \cos 4\pi(x_1 - x_2)$	$8.67 \times 10^5$

**Таблица 4.** Минимальная относительная ошибка  $\frac{\|R_i - \tilde{R}_i\|}{\|R_i\|}$  за 10 итераций работы алгоритмов

Распределение	$r_1$		$r_2$	
	Алгоритм 1	Алгоритм 2	Алгоритм 1	Алгоритм 2
$f_1$	$1.7 \times 10^{-5}$	$8.3 \times 10^{-6}$	$2.8 \times 10^{-3}$	$2.8 \times 10^{-3}$
$f_2$	$4.8 \times 10^{-6}$	$1.5 \times 10^{-6}$	$2.2 \times 10^{-4}$	$7.5 \times 10^{-4}$
$f_3$	$3.4 \times 10^{-5}$	$1.7 \times 10^{-5}$	$2.1 \times 10^{-2}$	$7.0 \times 10^{-3}$
$f_4$	$1.2 \times 10^{-5}$	$1.1 \times 10^{-5}$	$2.2 \times 10^{-3}$	$1.8 \times 10^{-3}$

**Таблица 5.** Минимальная относительная ошибка  $\frac{\|R_i - \tilde{R}_i\|}{\|R_i\|}$  за 10 итераций работы алгоритмов

Распределение	$r_3$		$r_4$	
	Алгоритм 1	Алгоритм 2	Алгоритм 1	Алгоритм 2
$f_1$	$1.4 \times 10^{-1}$	$1.4 \times 10^{-1}$	$1.8 \times 10^{-1}$	$1.8 \times 10^{-1}$
$f_2$	$5.3 \times 10^{-6}$	$4.7 \times 10^{-6}$	$1.0 \times 10^{-3}$	$2.0 \times 10^{-3}$
$f_3$	$2.8 \times 10^{-1}$	$2.8 \times 10^{-1}$	$3.5 \times 10^{-1}$	$3.4 \times 10^{-1}$
$f_4$	$7.7 \times 10^{-6}$	$5.5 \times 10^{-6}$	$1.5 \times 10^{-2}$	$1.5 \times 10^{-2}$

Для существования случайного процесса с заданной ковариационной функцией  $r(t, s)$  и функцией одномерного распределения  $F(t)$  требуется, чтобы  $r(t, s)$  и  $F(t)$  удовлетворяли некоторым условиям совместности, в частности, необходимо, чтобы выполнялось (см. [1])

$$r(t, s) \in [A_{F_t F_s}, B_{F_t F_s}],$$

$$A_{F_t F_s} = \int_0^1 F_t^{-1}(\alpha) F_s^{-1}(1 - \alpha) d\alpha, \quad B_{F_t F_s} = \int_0^1 F_t^{-1}(\alpha) F_s^{-1}(\alpha) d\alpha.$$

В случае стационарного процесса  $B_{F_t F_s} = B_F = 1$ . Для симметричных распределений  $A_{F_t F_s} = -1$ . Среди рассматриваемых вариантов сочетаний корреляционных функций и одномерных распределений не все пары  $r_i, f_j$  ( $i, j = 1, \dots, 4$ ) удовлетворяют указанному необходимому условию совместности ковариаций и одномерных распределений. Условие нарушается для пар  $(r_3, f_1)$ ,  $(r_3, f_3)$ ,  $(r_4, f_1)$ ,  $(r_4, f_3)$ . Этим обстоятельством объясня-

ется большая ошибка  $\frac{\|R_i - \tilde{R}_i\|}{\|R_i\|}$  в табл. 5 для указанных пар корреляционных функций и одномерных распределений.

В целом из табл. 4 и 5 видно, что относительная ошибка  $\frac{\|R_i - \tilde{R}_i\|}{\|R_i\|}$  существенно зависит от корреляционной матрицы. Так, ошибка оказывается больше для корреляционных матриц с большим числом обусловленности ( $R_2$  и  $R_4$ ). В большинстве случаев алгоритмы 1 и 2 дают близкую погрешность воспроизведения корреляционной матрицы, но, как указано выше, алгоритм 2 является более экономичным по затратам памяти и, в большинстве случаев, по времени моделирования.

Также исследовалась зависимость точности воспроизведения корреляционной матрицы от размера выборки случайных векторов. Согласно численным экспериментам, в случае хорошо обусловленной корреляционной матрицы, совместной с одномерными распределениями, увеличение числа реализаций случайных векторов позволяет получить более высокую точность воспроизведения корреляционной матрицы. Соответствующие результаты для  $(r_1, f_1)$  и  $(r_1, f_2)$  приведены в табл. 6.

**Таблица 6.** Зависимость относительной ошибки  $\frac{\|R_1 - \tilde{R}_1\|}{\|R_1\|}$  после 10 итераций от количества реализаций для случая корреляционной матрицы  $R_1$

Количество реализаций	$f_1$		$f_2$	
	Алгоритм 1	Алгоритм 2	Алгоритм 1	Алгоритм 2
$2 \times 10^2$	$3.3 \times 10^{-2}$	$1.1 \times 10^{-2}$	$3.1 \times 10^{-3}$	$3.0 \times 10^{-3}$
$10^3$	$1.0 \times 10^{-3}$	$4.6 \times 10^{-4}$	$3.7 \times 10^{-4}$	$1.7 \times 10^{-4}$
$10^4$	$1.3 \times 10^{-4}$	$6.1 \times 10^{-5}$	$3.0 \times 10^{-5}$	$1.7 \times 10^{-5}$
$10^5$	$1.7 \times 10^{-5}$	$8.3 \times 10^{-6}$	$6.1 \times 10^{-6}$	$1.4 \times 10^{-6}$
$10^6$	$1.8 \times 10^{-6}$	$9.2 \times 10^{-7}$	$1.0 \times 10^{-6}$	$5.9 \times 10^{-7}$

### 4.3. Численное исследование скорости сходимости алгоритмов в зависимости от выбора начального приближения

В алгоритмах 1 и 2 в качестве начального приближения  $Y^{(0)}$  берутся реализации случайных векторов с независимыми компонентами. Однако для ускорения сходимости этих алгоритмов в качестве  $Y^{(0)}$  можно взять выборку случайных векторов с ковариационной матрицей, близкой к заданной ковариационной матрице  $R$ . Рассмотрим следующий специальный способ выбора начального приближения  $Y^{(0)}$ , состоящий из 2-х шагов:

1. Смоделируем  $M$  реализаций гауссовского вектора размерности  $n$  с ковариационной матрицей  $R$ .
2. Компонента  $y_{ij}^{(0)}$  начального приближения  $Y^{(0)}$  находится с помощью преобразования  $y_{ij}^{(0)} = F_j^{-1}(\Phi(z_{ij}^{(0)}))$ , где  $\Phi(\cdot)$  — функция стандартного нормального распределения,  $z_{ij}^{(0)}$  —  $j$ -я компонента  $i$ -й реализации гауссовского вектора.

В табл. 7 и 8 приведена зависимость относительной ошибки от номера итерации для случая, когда в качестве начальных данных бралась выборка случайных векторов с независимыми компонентами как на шаге 1 алгоритма 2 (столбец 1) и случая, когда начальные данные моделировались вышеуказанным специальным способом (столбец 2).

Согласно данным в табл. 7 и 8, можно заключить, что для обоих алгоритмов специальный выбор начальных данных позволяет сократить вычисления на одну итерацию. Такое ускорение может оказаться ощутимым, если учесть, что алгоритмы уже за

сравнительно небольшое число итераций достигают хорошей точности воспроизведения корреляционной матрицы.

Численные эксперименты показывают, что для корреляционных матриц с большим числом обусловленности (как в случае матриц  $R_2$  и  $R_4$ ) не наблюдается монотонное убывание относительной ошибки с ростом номера итерации (см. табл. 8). В таких случаях следует остановить работу алгоритмов, когда после нескольких итераций относительная ошибка вновь начинает расти.

**Таблица 7.** Зависимость относительной ошибки  $\frac{\|R_1 - \tilde{R}_1\|}{\|R_1\|}$  от номера итерации в случае корреляционной матрицы  $R_1$  и одномерного распределения  $f_1$

Номер итерации	Алгоритм 1		Алгоритм 2	
	1	2	1	2
0	$9.5 \times 10^{-1}$	$8.8 \times 10^{-2}$	$9.5 \times 10^{-1}$	$8.7 \times 10^{-2}$
1	$9.6 \times 10^{-2}$	$4.7 \times 10^{-3}$	$8.6 \times 10^{-2}$	$4.4 \times 10^{-3}$
2	$7.3 \times 10^{-3}$	$3.2 \times 10^{-4}$	$5.2 \times 10^{-3}$	$2.8 \times 10^{-4}$
3	$6.2 \times 10^{-4}$	$5.4 \times 10^{-5}$	$3.6 \times 10^{-4}$	$4.3 \times 10^{-5}$
4	$9.5 \times 10^{-5}$	$2.5 \times 10^{-5}$	$4.5 \times 10^{-5}$	$1.8 \times 10^{-5}$
5	$3.6 \times 10^{-5}$	$1.8 \times 10^{-5}$	$1.8 \times 10^{-5}$	$1.2 \times 10^{-5}$
6	$2.4 \times 10^{-5}$	$1.6 \times 10^{-5}$	$1.2 \times 10^{-5}$	$1.0 \times 10^{-5}$
7	$2.0 \times 10^{-5}$	$1.5 \times 10^{-5}$	$9.8 \times 10^{-6}$	$9.2 \times 10^{-6}$
8	$1.9 \times 10^{-5}$	$1.4 \times 10^{-5}$	$9.0 \times 10^{-6}$	$8.7 \times 10^{-6}$
9	$1.8 \times 10^{-5}$	$1.4 \times 10^{-5}$	$8.5 \times 10^{-6}$	$8.4 \times 10^{-6}$
10	$1.7 \times 10^{-5}$	$1.3 \times 10^{-5}$	$8.3 \times 10^{-6}$	$8.2 \times 10^{-6}$

**Таблица 8.** Зависимость относительной ошибки  $\frac{\|R_2 - \tilde{R}_2\|}{\|R_2\|}$  от номера итерации в случае корреляционной матрицы  $R_2$  и одномерного распределения  $f_1$

Номер итерации	Алгоритм 1		Алгоритм 2	
	1	2	1	2
0	$9.4 \times 10^{-1}$	$5.5 \times 10^{-2}$	$9.4 \times 10^{-1}$	$5.5 \times 10^{-2}$
1	$5.6 \times 10^{-2}$	$2.5 \times 10^{-3}$	$5.3 \times 10^{-2}$	$2.3 \times 10^{-3}$
2	$2.8 \times 10^{-3}$	$2.2 \times 10^{-3}$	$2.8 \times 10^{-3}$	$1.5 \times 10^{-3}$
3	$7.7 \times 10^{-3}$	$3.2 \times 10^{-3}$	$1.9 \times 10^{-2}$	$2.1 \times 10^{-3}$
4	$1.2 \times 10^{-2}$	$5.1 \times 10^{-3}$	$2.7 \times 10^{-2}$	$3.0 \times 10^{-3}$
5	$1.7 \times 10^{-2}$	$7.9 \times 10^{-3}$	$3.7 \times 10^{-2}$	$4.7 \times 10^{-3}$
6	$2.3 \times 10^{-2}$	$1.2 \times 10^{-2}$	$4.7 \times 10^{-2}$	$7.4 \times 10^{-3}$
7	$2.9 \times 10^{-2}$	$1.8 \times 10^{-2}$	$5.7 \times 10^{-2}$	$1.1 \times 10^{-2}$
8	$3.4 \times 10^{-2}$	$2.4 \times 10^{-2}$	$6.8 \times 10^{-2}$	$1.6 \times 10^{-2}$
9	$3.8 \times 10^{-2}$	$3.1 \times 10^{-2}$	$7.9 \times 10^{-2}$	$2.2 \times 10^{-2}$
10	$4.3 \times 10^{-2}$	$3.8 \times 10^{-2}$	$9.2 \times 10^{-2}$	$2.7 \times 10^{-2}$

## 5. Заключение

В работе предложен новый приближённый алгоритм моделирования негауссовских случайных векторов с заданными одномерными распределениями и ковариационной матрицей. Проведённые численные эксперименты показали, что этот алгоритм фактически эквивалентен по точности воспроизведения ковариационной матрицы алгоритму из работы [10], однако является в два раза более эффективным по использованию памяти, а также более быстродейственным для случаев, когда обратная функция одномерного распределения является легко вычисляемой. Численно показано, что выбор специальных начальных данных позволяет сократить вычисления на одну итерацию.

В дальнейшем планируется исследовать возможность модификации рассмотренных алгоритмов для моделирования конкретных классов случайных векторов: стационарных, периодически-коррелированных и т. д. Ковариационные матрицы в этом случае являются тёплицевыми или блочно-тёплицевыми, для которых существуют специальные алгоритмы разложения Холецкого и обращения, позволяющие сократить вычисления по сравнению с общим случаем.

## Литература

1. **Пригарин С.М.** Методы численного моделирования случайных процессов и полей. — Новосибирск: Изд-во ИВМиМГ СО РАН, 2005.
2. **Hong H.P., Xiao M.Y.** Simulating nonhomogeneous non-Gaussian field by using iterative rank-dependent reordering versus translation process-based procedure // *Mathematical Problems in Engineering*. — 2022. — Vol. 2022. — Article ID 2700498. — <https://doi.org/10.1155/2022/2700498>.
3. **Huang S.P., Phoon K.K., Quek S.T.** Digital simulation of non-Gaussian stationary processes using Karhunen–Loeve expansion // *Proc. 8th ASCE specialty conf. on probabilistic mechanics and structural reliability*. — 2000. — P. 1–5.
4. **Iman R.L., Conover W.J.** A distribution-free approach to inducing rank correlation among input variables // *Communications in Statistics-Simulation and Computation*. — 1982. — Vol. 11, № 3. — P. 311–334.
5. **Kim H., Shields M.D.** Modeling strongly non-Gaussian non-stationary stochastic processes using the iterative translation approximation method and Karhunen–Loeve expansion // *Computers & Structures*. — 2015. — Vol. 161. — P. 31–42.
6. **Shields M.D., Deodatis G., Bocchini P.** A simple and efficient methodology to approximate a general non-Gaussian stationary stochastic process by a translation process // *Probabilistic Engineering Mechanics*. — 2011. — Vol. 26, № 4. — P. 511–519.
7. **Shields M.D., Deodatis G.** Estimation of evolutionary spectra for simulation of non-stationary and non-Gaussian stochastic processes // *Computers & Structures*. — 2013. — Vol. 126. — P. 149–163.
8. **Tong M.N., Zhao Y.G., Zhao Z.** Simulating strongly non-Gaussian and non-stationary processes using Karhunen–Loeve expansion and L-moments-based Hermite polynomial model // *Mechanical Systems and Signal Processing*. — 2021. — Vol. 160, № 2. — Article ID 107953. — <https://doi.org/10.1016/j.ymsp.2021.107953>.
9. **Yan Y., Jeong J., Genton M.G.** Multivariate transformed Gaussian processes // *Japanese J. of Statistics and Data Science*. — 2020. — Vol. 3, № 1. — P. 129–152.
10. **Zheng Z., Dai H., Wang Y., Wang W.** A sample-based iterative scheme for simulating non-stationary non-Gaussian stochastic processes // *Mechanical Systems and Signal Processing*. — 2021. — Vol. 151. — Article ID 107420. — <https://doi.org/10.1016/j.ymsp.2020.107420>.

*Поступила в редакцию 29 марта 2023 г.*

*После исправления 18 мая 2023 г.*

*Принята к печати 05 сентября 2023 г.*

## Литература в транслитерации

1. **Prigarin S.M.** Metody chislenngo modelirovaniya sluchainykh protsessov i polei. — Novosibirsk: Izd-vo IVMiMG SO RAN, 2005.
2. **Hong H.P., Xiao M.Y.** Simulating nonhomogeneous non-Gaussian field by using iterative rank-dependent reordering versus translation process-based procedure // *Mathematical Problems in Engineering*. — 2022. — Vol. 2022. — Article ID 2700498. — <https://doi.org/10.1155/2022/2700498>.

3. **Huang S.P., Phoon K.K., Quek S.T.** Digital simulation of non-Gaussian stationary processes using Karhunen–Loeve expansion // Proc. 8th ASCE specialty conf. on probabilistic mechanics and structural reliability. — 2000. — P. 1–5.
4. **Iman R.L., Conover W.J.** A distribution-free approach to inducing rank correlation among input variables // Communications in Statistics-Simulation and Computation. — 1982. — Vol. 11, № 3. — P. 311–334.
5. **Kim H., Shields M.D.** Modeling strongly non-Gaussian non-stationary stochastic processes using the iterative translation approximation method and Karhunen–Loeve expansion // Computers & Structures. — 2015. — Vol. 161. — P. 31–42.
6. **Shields M.D., Deodatis G., Bocchini P.** A simple and efficient methodology to approximate a general non-Gaussian stationary stochastic process by a translation process // Probabilistic Engineering Mechanics. — 2011. — Vol. 26, № 4. — P. 511–519.
7. **Shields M.D., Deodatis G.** Estimation of evolutionary spectra for simulation of non-stationary and non-Gaussian stochastic processes // Computers & Structures. — 2013. — Vol. 126. — P. 149–163.
8. **Tong M.N., Zhao Y.G., Zhao Z.** Simulating strongly non-Gaussian and non-stationary processes using Karhunen–Loeve expansion and L-moments-based Hermite polynomial model // Mechanical Systems and Signal Processing. — 2021. — Vol. 160, № 2. — Article ID 107953. — <https://doi.org/10.1016/j.ymssp.2021.107953>.
9. **Yan Y., Jeong J., Genton M.G.** Multivariate transformed Gaussian processes // Japanese J. of Statistics and Data Science. — 2020. — Vol. 3, № 1. — P. 129–152.
10. **Zheng Z., Dai H., Wang Y., Wang W.** A sample-based iterative scheme for simulating non-stationary non-Gaussian stochastic processes // Mechanical Systems and Signal Processing. — 2021. — Vol. 151. — Article ID 107420. — <https://doi.org/10.1016/j.ymssp.2020.107420>.