Using the inverse distribution function method and the modified superposition method in the NMPUD computer system

D A Cherkashin\(^1\) and A V Voytishek\(^{2,3}\)

\(^1\) 10 EN class, Lyceum No. 130, Uchonyh str., 10, Novosibirsk, 630090, Russia
\(^2\) Laboratory of Stochastic Problems, Institute of Computational Mathematics and Mathematical Geophysics SB RAS, ICM&MG SB RAS, Ac. Lavrentieva ave., 6, Novosibirsk, 630090, Russia
\(^3\) Department of Mathematics and Mechanics, Novosibirsk State University, NSU, Pirogova ul. 2, 630090, Novosibirsk, Russia

E-mail: vav@osmf.sscc.ru

Abstract. This paper presents a computer system for modelling one-dimensional random variables NMPUD, developed in the laboratory of mathematical modelling of Lyceum No. 130 in Novosibirsk. The results of the numerical experiments and the considerations justifying the practicability for using in the NMPUD system: the elementary densities constructed by the technology of sequential (inserted) substitutions, the densities representing weighted sums of elementary densities (which can be simulated using the modified discrete superposition method), the algorithms for a piecewise linear approximation of unknown densities using a given sample, the algorithms of the modified superposition method for computational modelling of random variables with piecewise linear densities, are also presented.

1. The NMPUD computer system

When carrying out scientific research of many real processes and phenomena, it is important to construct an appropriate mathematical model. These models make it possible both to refine the parameters of practical experiments (and sometimes to replace these experiments) and to predict new effects. In turn, the methods of computational mathematics make it possible to perform analytical and to approximate calculations associated with a particular mathematical model on modern computers.

Stochastic (probabilistic) models are an important class of modern mathematical models. They correspond to computational (numerical) algorithms for statistical modelling (or Monte Carlo methods). These algorithms always include the choice of low-cost numerical procedures for simulating sample values of the random parameters being introduced.

The NMPUD (Numerical Modelling of Probabilistic Univariate Distributions) computer system is being developed, starting in 2020, at the laboratory of mathematical modelling of Lyceum No. 130 in Novosibirsk \([1, 2]\). The NMPUD system is mainly used as a useful (and even necessary) tool for selecting the important efficiently simulated distributions and is intended for researchers and students who study, develop and (or) use computer stochastic models to solve important applied problems.

The NMPUD system bank includes the formulas for probability densities
\[
f_{\xi}(u; \lambda), \quad u \in (a, b) \tag{1.1}
\]

(usually elementary – see section 1.4.1 in [3], section 2.6 in [4] and section 2 in this paper) of random variables \( \xi \in (a, b) \) and the corresponding formulas (less often – algorithms) for computational (numerical) modelling (simulation) of sample values \( \xi_0 \) of these random variables of the form

\[
\xi_0 = \psi_{\xi}(\alpha_0; \lambda), \tag{1.2}
\]

where \( \alpha_0 \in U(0,1) \) is a standard random number (see section 1.1.1 in [3] and section 2.4 in [3]), i.e., a sample value of the random variable \( \alpha \in U(0,1) \) uniformly distributed over the interval \((0,1)\); this sample value is implemented on the computer using special routines (referred to in programming languages like \texttt{RAND} or \texttt{RANDOM}).

In formulas (1.1) and (1.2), the letter \( \lambda \) defines a parameter (a set of parameters) of the distribution. The set of parameters \( \lambda \) can include, among other things, the boundaries of the interval \((a, b)\).

The bank also records such an important characteristic of formula (algorithm) (1.2) as the cost of modelling (the average time of one realization of formula (1.2) – in nanoseconds).

To study a particular example of density (1.1) and modelling formula (algorithm) (1.2), the user of the NMPUD system can apply the main page (see figure).

**Figure 1.** The main page of the NMPUD system.

At the bottom of the page (on the left side), the density formula (1.1) and the corresponding distribution interval \((a, b)\) are presented. At the center of the page, one can see the graph of this function (the white curve).

The modelling formula (algorithm) (1.2) is shown (or introduced) at the bottom in the right part of the main page of the system. The histogram is created by repeatedly simulating sample values by the formula and displayed in yellow bars at the center of the page. This process can be observed in stages. The correctness of the modelling formula (algorithm) (1.2) is determined by the visual proximity of the constructed histogram and the density graph (1.1).

The average time of one realization of formula (algorithm) (1.2) in nanoseconds is also shown on the right side of the screen (the large green number).

There exists an instruction for introduction of density formulas (1.1) and modelling formulas (algorithms) (1.2) into the system for carrying out the necessary studies of the computational efficiency of these formulas.
2. Creating the bank of elementary densities for the NMPUD system

The main content of the density bank of the NMPUD system is the so-called elementary densities. We recall the corresponding definitions.

The inverse distribution function method for a random variable \( \xi \in (a, b) \) with a distribution function \( F_\xi(x; \lambda) = P\{\xi < x\} = \int_{-\infty}^{x} f_\xi(u; \lambda) \, du \) is implemented by the formula

\[ \xi_0 = F_\xi^{-1}(\alpha_0; \lambda), \]  

(see, for example, section 1.4.1 in [3] and section 2.5 in [4]); here \( \alpha_0 \in U(0,1) \) is a standard random number.

As an intermediate calculation for the derivation of formula (2.1), we can write down the relation

\[ \int_a^{\xi_0} f_\xi(u; \lambda) \, du = \alpha_0; \]  

(2.2)

here we take into account the fact that for \( u \notin (a, b) \), the function \( f_\xi(u; \lambda) \) from (1.1) is equal to zero.

Density (1.1) is called elementary if equation (2.2) is solvable with respect to the upper limit of integration \( \xi_0 \) in elementary functions. In other words, the solution of equation (2.2) gives formula (1.2), where \( \psi_\xi(v; \lambda) = F_\xi^{-1}(v; \lambda) \) is a relatively simple («programmable») composition of elementary functions (see section 1.4.1 of in [3] and section 2.6 in [4]).

To form a sufficiently large bank of efficiently modelled elementary densities, it is appropriate to use the following technology of sequential (inserted) substitutions.

METHOD 1 (see section 1.4.4 in [3] and section 14.2 in [4]). Let

\[ f_\eta(v; \lambda); \, v \in (c, d), \, \lambda \in \Lambda \]  

(2.3)

be a density of the random variable \( \eta \) with the elementary distribution over the interval \((c, d)\), i.e., from the equation of the form of (2.2) \( \int_c^{\eta_0} f_\eta(v; \lambda) \, dv = \alpha_0 \) for the corresponding sample value \( \eta_0 \) of the random variable \( \eta \) one can get a formula of the form of (1.2) or (2.1): \( \eta_0 = \psi_\eta(\alpha_0; \lambda) \), where \( \psi_\eta(w; \lambda) \) is a simple composition of elementary functions.

Consider a one-to-one transformation given by a monotone (for example, increasing) differentiable function \( \varphi(u) \uparrow \) and mapping the interval \((a, b)\) onto the interval \((c, d)\) (for an increasing function we have \( \varphi(a) = c, \varphi(b) = d \)). We also assume that the function \( \varphi(u) \) itself, its derivative \( \varphi'(u) \) and its inverse \( \varphi^{-1}(v) \) can be represented as simple compositions of elementary functions.

Let the random variable \( \xi \in (a, b) \) have the distribution density

\[ f_\xi(u; \lambda) = f_\eta(\varphi(u); \lambda) \times \varphi'(u), \, u \in (a, b) \]  

(2.4)

(for a decreasing function \( \varphi(u) \), the modulus of the derivative \( \varphi'(u) \) must be used in the latter expression).

Under the above assumptions, it can be argued that the density \( f_\xi(u; \lambda) \) of the form of (2.4) is elementary, i.e. the corresponding equation (2.2) is solvable with respect to \( \xi_0 \) in elementary functions, and more specifically,

\[ \xi_0 = \varphi^{-1}[\psi_\eta(\alpha_0; \lambda)] \]  

(2.5)

(for a decreasing function \( \varphi(u) \), we have \( \xi_0 = \varphi^{-1}[\psi_\eta(\alpha_0'; \lambda)]; \, \alpha_0' = 1 - \alpha_0 \).

Indeed, writing down equation (2.2) for the constructed new density (2.4), we have

\[ \int_a^{\xi_0} f_\eta(\varphi(u); \lambda) \varphi'(u) \, du = \alpha_0 \quad \text{or} \quad \int_{\varphi(a)}^{\varphi(\xi_0)} f_\eta(v; \lambda) \, dv = \alpha_0 \quad \text{or} \quad \varphi(\xi_0) = \psi_\eta(\alpha_0; \lambda) \quad \text{or} \quad \xi_0 = \varphi^{-1}[\psi_\eta(\alpha_0; \lambda)] \]  

(2.6)

(for a decreasing function \( \varphi(u) \), the calculations are similar).

The term technology of sequential (inserted) substitutions for method 1 is related to the fact that the resulting density (2.4) can be taken as initial density (2.3) and another transformation \( \varphi(u) \) can be
performed for constructing a new elementary density (2.4). Such inserted substitutions can be used to obtain an unlimited number of new elementary densities.

During the development of the NMPUD system, the subsystem for generating an unlimited number of densities of the form of (2.4) and corresponding modelling formulas of the form of (2.5) from the initial densities (2.3) using classical elementary functions was elaborated (here the nerdamer and mathjs symbolic algebra packages were used). The only limitation in this process is the time required for the computer implementation of these formulas, since the formulas of the form of (2.5) become more complex as the number of substitutions \( \varphi(u) \) increases. Using the automation of the sequential (inserted) technology we have got over 1000 densities, and only about a hundred of them are simulated faster than 100 ns (this was the criterion for including these densities into the NMPUD system bank).

### 3. The modified superposition method

The formulas of the form of (2.1), (2.5) of the inverse distribution function method are the most commonly used ones in practical calculations (see, for example, [3], [4]).

During the development of the NMPUD system, the possibilities of using probability densities efficiently simulated by other (than the inverse distribution function method (2.1)) methods of computer simulation of random variables were discussed and investigated.

In particular, it turned out that the use of the so-called modified discrete superposition method is effective for a number of important cases.

This method is used when density (1.1) of a random variable \( \xi \in (a, b) \) can be represented by a weighted sum

\[
 f_\xi(u; \lambda) = \sum_{i=1}^{M} p_i f_i(u; \lambda), \quad u \in (a, b); \quad p_i > 0; \quad \sum_{i=1}^{M} p_i = 1,
\]

and \( f_i(u; \lambda) \); \( i = 1, \ldots, M \) are elementary densities of the random variables \( \xi^{(i)} \) having simulating formulas of the form of (2.1): \( \xi^{(i)}_0 = \Psi^{(i)}(\alpha_0; \lambda) \) obtained by solving equations of the form of (2.2):

\[
 \int_{a}^{b} f_i(u; \lambda) \, du = \alpha_0.
\]

**ALGORITHM 1** (see section 1.6.3 in [3] and section 11.2 in [4]).

1. Simulate a standard random number \( \alpha_0 \in U(0,1) \) and using a suitable and most efficient algorithm for simulating the integer-valued discrete random variable \( \eta \) with the distribution \( P[\eta = i] = p_i; \ i = 1, \ldots, M \), find the number \( \eta = m \) such that \( \alpha_0 \in \Delta_m = \left[ \sum_{i=1}^{m-1} p_i; \sum_{i=1}^{m} p_i \right] \).

2. Simulate the sample value \( \xi_0 \) of a random variable \( \xi \in (a, b) \) with distribution density (3.1) by the formula \( \xi_0 = \Psi^{(m)}[\beta(\alpha_0); \lambda] \), where \( \beta(\alpha_0) = (\alpha_0 - \sum_{i=1}^{m-1} p_i)/p_m \).

### 4. The case of a small number of summands

Using the NMPUD computer system, we investigated the efficiency of algorithm 1 (in comparison with the formulas of the inverse distribution function method (2.1)) for the case of a small number \( M \) of terms in sum (3.1).

Let, for example, \( M = 2 \) and

\[
 f_\xi(u; \lambda) = p_1 f_1(u; \lambda) + p_2 f_2(u; \lambda); \quad 0 < p_1 < 1, 0 < p_2 < 1; \quad p_1 + p_2 = 1.
\]

In this case algorithm 1 has the form:

if \( \alpha_0 < p_1 \), then \( \xi_0 = \Psi^{(1)}(\alpha_0/p_1; \lambda) \), else \( \xi_0 = \Psi^{(2)}[(\alpha_0 - p_1)/p_2; \lambda] \); (4.2)

here the functions \( \Psi^{(1)}(w; \lambda) \) and \( \Psi^{(2)}(w; \lambda) \) define formulas of the form of (2.1) and (2.5) for the elementary densities \( f_1(u; \lambda) \) and \( f_2(u; \lambda) \), respectively.

Let us formulate some conclusions from our experiments conducted on a personal computer Acer Aspire 3 with processor: AMD Ryzen 3 2200U with Radeon Vega Mobile Gfx (4 CPUs) ~2.5GHz, 6 GB RAM using the NMPUD computer system.

**CONCLUSION 1.** If the costs for calculations of the functions \( F_\xi^{-1}(w; \lambda) \), \( \Psi^{(1)}(w; \lambda) \) and \( \Psi^{(2)}(w; \lambda) \) are approximately equal, but exceed the cost of «cheap» operations (for example, a comparison operation, or adding two numbers, etc.), then the costs of formulas of the inverse distribution function method (2.1) and the modified superposition algorithms (4.2) are close.
For example, the cost of the modelling formula \( \xi_0 = \sqrt[3]{a_0}; \ a_0 \in U(0,1) \) for the power distribution with the density \( f_\xi(u) = 7u^6, 0 < u < 1 \) is equal to 154.14 nanoseconds, and the cost of the algorithm of the form (4.2): if \( a_0 < \frac{1}{7} \) then \( \xi_0 = \sqrt[3]{7a_0} \), otherwise \( \xi_0 = \frac{1}{6} (7a_0 - 1) \) for the density of the form of (4.1): \( f_\xi(u) = u^6 + \frac{24}{7}u^3, 0 < u < 1 \) is 158.24 nanoseconds (these costs are close).

**CONCLUSION 2.** For densities of the form of (4.1), for which there exist both the formula of the inverse distribution function method (2.1) and algorithm (4.2), the superposition method often turns out to be more efficient.

For example, the cost of simulation of the random variable with the density \( f_\xi(u) = u^7 + \frac{7}{2}u^3, 0 < u < 1 \) according to the formula of the inverse distribution function method \( \xi_0 = \sqrt[3]{\frac{32a_0 - 49}{2}} \) (obtained from relation (2.2), which is equal to a quadratic equation for \( \xi_0 \)) is equal to 153.73 nanoseconds. In turn, the cost of the superposition algorithm (4.2) of the form: if \( a_0 < \frac{1}{8} \) then \( \xi_0 = \sqrt[8]{8a_0} \), else \( \xi_0 = \sqrt[3]{8a_0 - 1} \) is 148.06 nanoseconds.

Even more significant is the example of simulation of the distribution density of the scattering angle cosine for the Rayleigh law on molecular photon scattering in the atmosphere, which has the form \( f_\xi(u) = \frac{3}{8}(1 + u^2), -1 < u < 1 \) (see example 11.1 in [3]). The consideration of equation (2.2) gives a cubic equation and the formula of the inverse distribution function method of the form

\[
\xi_0 = \left( \frac{16a_0^2 - 16a_0 + 5 + 4a_0 - 2}{3\sqrt[3]{16a_0^2 - 16a_0 + 5 + 4a_0 - 2}} \right)^\frac{2}{3} - 1
\]

(this is the Cardano formula). The NMPUD system records the cost of 335.62 nanoseconds for this formula. In turn, the cost of the superposition algorithm (4.2) of the form: if \( a_0 < \frac{3}{4} \), then \( \xi_0 = \frac{8a_0}{3} \), else \( \xi_0 = \frac{3}{8a_0 - 7} \) is 59.20 nanoseconds (i.e. more than five times less than for the inverse distribution function method).

**CONCLUSION 3.** The densities of the form of (4.1) and the modelling algorithms of the form of (4.2) can significantly complete the bank of densities and modelling algorithms of the NMPUD computer system.

5. **On the NMPUD system block for big data processing and modelling piecewise linear densities**

As a part of the development of the NMPUD computer system, the concept of a special block of the system for big data processing was developed. Using the functional content of this block, one can solve several tasks at once.

First, the problem of an operational computer approximation of an unknown one-dimensional distribution density \( f_\xi(u); u \in (a, b) \) of stationary volume sets of sample values \( \{\xi_1, \xi_2, \ldots, \xi_M\} \) at a given error level \( \varepsilon \) will be solved. Here we use the results presented in [5] and related papers, where it was shown that the construction of the polygon of frequencies

\[
f_\xi(u) = v_{i-1} + (u - a_i) \frac{v_i - v_{i-1}}{b_i - a_i}, \quad a_i \leq u < b_i; \quad i = 1, \ldots, M; \quad (5.1)
\]

\(-\infty < a = a_1 < b_1 = a_2 < b_2 < \cdots < b_{M-1} = a_M < b_M = b < +\infty; \)

is practically optimal because of simplicity and efficiency of the computational scheme (5.1) and the possibility of the conditional optimization theory development for this scheme.

In formula (5.1), \( v_i = \frac{k_i}{n|\xi|} \) and \( k_i \) is the number of sample values \( \{\xi_1, \xi_2, \ldots, \xi_n; n \leq n\} \), belonging to the semi-interval \( D_i = \left[ \frac{a_{i-1} + b_{i-1}}{2}, \frac{a_i + b_i}{2} \right); \) and \( |D_i| \) is the length of the semi-interval \( D_i \).
The mentioned conditional optimization theory allows us to define the conditionally optimal parameters \( M = M_{\text{opt}}^{(B,a,b)}(L) \) and \( n = n_{\text{opt}}^{(B,a,b)}(L) \) in the selected Banach space \( B[a,b] \) (most often \( B[a,b] = L_2[a,b] \) and \( B[a,b] = C[a,b] \)) using the following technology.

METHOD 2 (see, for example, section 5.3.7 in [3], section 8.7 in [4] and introduction in [5]). Construct the upper boundary \( U_P(B)(M,n) \) for the error \( \delta(B)(M,n) \) of approximation (5.1) of the density \( f_\hat{x}(u) \); this boundary depends on the parameters \( M \) and \( n \):

\[
\delta(B)(M,n) = \left\| f_\hat{x} - f_\xi \right\|_{B(X)} \leq U_P(B)(M,n).
\]

Suppose this function of two variables to be equal to the value \( L \). Using the equation

\[
U_P(B)(M,n) = L, \tag{5.2}
\]

express one parameter (for example, \( n \)) through another: \( n = Y(u)(M) \). This relation is substituted into the expression for the computational cost \( S(M,n) \) for constructing approximation (5.1) (which also depends on the parameters \( M \) and \( n \)). As a result, we get the function \( S(B,L)(M) \) of one variable \( M \). This function is investigated to a minimum using well-known methods of the mathematical or the numerical analysis. The found values \( M_{\text{opt}}^{(B)}(L) = M_{\text{opt}}^{(B)}(L), n_{\text{opt}}^{(B)}(L) = Y(u) \left[ M_{\text{opt}}^{(B)} \right] \) are defined as conditionally optimal parameters of the numerical approximation (5.1).

The “conditionality” of this optimization technology is due to the fact that on the left-hand side of the equation of the form of (5.2), not the error of the algorithm \( \delta(B)(M,n) \) is used, but its upper boundary \( U_P(B)(M,n) \).

For example, for the so-called \( L_2 \)-approach (here the fulfillment of the inequality (5.2) is assumed in the mean-square sense) the following forms of the conditionally optimal parameters were obtained in [5] using method 2:

\[
M_{\text{opt}}^{(L_2)}(L) = \frac{\sqrt{5}}{2\sqrt{3}}(b - a)^{5/4} \sup_{x \in [a,b]} \left| f_\hat{x}''(x) \right| \times L^{-1/2}, \tag{5.3}
\]

\[
n_{\text{opt}}^{(B)}(L) = \frac{5\sqrt{5}}{8\sqrt{3}}(b - a)^{5/4} \max_{x \in [a,b]} f_\hat{x}(x) \sup_{x \in [a,b]} \left| f_\hat{x}''(x) \right| \times L^{-5/2} \tag{5.4}
\]

(as one can see, when applying method 2, problems arose of preliminary calculation of \( \max_{x \in [a,b]} f_\hat{x}(x) \) – this is a relatively simple task – and \( \sup_{x \in [a,b]} \left| f_\hat{x}''(x) \right| \) – here the elaboration and the application of the corresponding computational scheme is needed).

Second, for the NMPUD users, it will be strongly recommended to use the elementary composite density (5.1) in the case when it is needed to get additional sample values \( \{\xi_1, \xi_2, \ldots, \xi_K\} \), similar in distribution to the initial set \( \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n\} \). Here we use the fact that density (5.1) can be presented in the form of (3.1) with

\[
\begin{align*}
p_i &= \frac{(v_i - v_{i-1}) + v_{i+1} - v_i}{2}; \\
f_i(u; \lambda) &= (A_i u + B_i) \lambda^{(a,b)}(u); \\
A_i &= \frac{2(v_i - v_{i-1})^2}{(v_i + v_{i+1})(b - a)^2}; \\
B_i &= \frac{2(v_i - v_{i-1})^2}{(v_i + v_{i+1})(b - a)^2}; \\
\lambda^{(a,b)}(u) &= \begin{cases} 1 & \text{for } u \in [a_i, b_i]; \\ 0 & \text{for } u \notin [a_i, b_i] \end{cases}
\end{align*} \tag{5.5}
\]

together with the following statement.

STATEMENT 1 (see section 1.6.4 in [3] and section 11.3 in [4]). For a random variable \( \xi \) with the elementary composite density (in particular, with density (3.1), (5.5)), the inverse distribution function method is equal to the modified superposition method (algorithm 1) with the simulation of the number \( m \) by the standard algorithm, consisting of subtracting the quantities \( \sum_{i=1}^{m} p_i \) from \( \alpha_n \) to get the first negative value.

It is easy to obtain formulas of the inverse distribution function method for random variables \( \xi^{(i)} \) with distribution densities \( f_i(u; \lambda) \) from relations (5.5):
\[ \xi^{(i)}_0 = \Psi_i(\alpha_0; \lambda) = \frac{-B_i + \sqrt{(B_i + A_i a_i)^2 + 2A_i a_i}}{A_i}; \quad \alpha_0 \in U(0,1). \]

From this, we get the following efficient algorithm for simulating a sample value \( \xi_0 \) of a random variable \( \xi \) having a piecewise linear distribution density (5.1).

**ALGORITHM 2.** 1. Simulate a standard random number \( \alpha_0 \in U(0,1) \) and using a suitable and most efficient algorithm for simulating the integer-valued discrete random variable \( \eta \) with the distribution \( P(\eta = i) = p_i = \frac{v_i + \eta v_i}{2}, \) find the number \( \eta = m \) such that \( \alpha_0 \in \Delta_m = \left[ \sum_{i=1}^{m-1} p_i; \sum_{i=1}^{m} p_i \right) . \)

2. Simulate the sample value \( \xi_0 \) of the random variable \( \xi \in (a, b) \) with the distribution density (5.1) by the formula

\[ \xi_0 = \Psi_m[\beta(\alpha_0); \lambda] = \frac{-B_m + \sqrt{(B_m + A_m a_m)^2 + 2A_m \beta(\alpha_0)}}{A_m}, \]

where \( \beta(\alpha_0) = \frac{(\alpha_0 - \sum_{i=1}^{m-1} p_i)}{p_m}. \)

We must take into account that for the approximations of the densities \( f_{\xi}(u); u \in (a, b) \) by the sample \( \{\xi_1, \xi_2, \ldots, \xi_M\} \) the number \( M \) of the semi-intervals \( [a_i, b_i) \) in formula (5.1) is sufficiently large. Thus, for simulation of the number \( m \) at the first step of algorithm 2, we should apply the quantile method or the Walker method (see sections 1.3.3, 1.3.4 in [3] and sections 10.6, 10.7 in [4]) instead of the standard algorithm with subtraction of sums \( \sum_{i=1}^{m} p_i \) from \( \alpha_0 \).

There is also an idea of using the polygon of frequencies of the form (3.1), (5.5), in which the probabilities \( p_i \) are equal to each other: \( p_1 = p_2 = \ldots = p_M = 1/M \). This can significantly increase the efficiency of algorithm 2, but it makes it more difficult for finding the required conditionally optimal parameters (5.3) and (5.4). The practicability of using the identical probabilities requires a separate study.

All the described ideas are taken into account when developing the corresponding special block of the NMPUD computer system.

6. Conclusion

This paper presents the computer system for modelling one-dimensional random variables NMPUD, developed in the laboratory of mathematical modelling of Lyceum No. 130 in Novosibirsk. We have presented the results of numerical experiments and the considerations justifying the practicability of using in the NMPUD system:

- the elementary densities constructed by the technology of sequential (inserted) substitutions;
- the densities representing weighted sums of elementary densities, which can be simulated using the modified discrete superposition method;
- the algorithms for the piecewise linear approximation of unknown densities using a given sample;
- the algorithms of the modified superposition method for the computational simulating of random variables with piecewise linear densities.

Acknowledgements

This work was supported by state contract with ICM&MG SB RAS (0251-2021-0002)

References

[1] Vasiliev T V, Postovalov Ya S and Cherkashin D A 2020 Project of a computer system for selection and research of modelled probability distributions Proceedings of the 58-th International Student Conference “Student and Scientific and Technological Progress”. Mathematics (Novosibirsk: NSU Publishing and Printing Center) p 112 [in Russian]

[2] Voytishek A V, Postovalov Ya S and Cherkashin D A 2021 System of numerical modelling of one-dimensional random variables NMPUD: formation of a density bank, automation of mathematical calculations and applications Proceedings of the XIX International Conference
named after A. F. Terpugov “Information Technologies and Mathematical Modelling” (Tomsk: NTL Publishing House) pp 363–368 [in Russian]

[3] Mikhailov G A and Voytishek A V 2020 *Statistical Modelling. Monte Carlo Methods* (Moscow: Yurayt Publishing House) [in Russian]

[4] Voytishek A V 2018 *Lectures on Numerical Monte Carlo Methods* (Novosibirsk: NSU Publishing and Printing Center) [in Russian]

[5] Bulgakova T E 2020 *Optimization of Functional Computational Statistical Estimates and Algorithms* Dissertation ... candidate of physical and mathematical sciences (RAS, Siberian Department, ICMMG Novosibirsk) [in Russian]