

Тема 4. МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ФАКТОРОВ

Имитационная модель позволяет исследовать поведение различных систем с учетом влияния случайных факторов. Эти факторы в зависимости от их природы могут быть отражены в модели как случайные события, случайные величины (дискретные или непрерывные), или как случайные функции (процессы).

В основе всех методов и приемов моделирования случайных факторов лежит использование базовой случайной величины -- случайных чисел, имеющих равномерное распределение на интервале $[0; 1]$.

4.1. Построение датчиков БСВ

4.1.1. Датчики БСВ

Базовой случайной величиной (БСВ) в статистическом моделировании называют непрерывную случайную величину Z , равномерно распределенную на интервале $(0,1)$. Ее плотность распределения вероятностей имеет вид:

$$f(z) = \begin{cases} 1, & z \in (0,1); \\ 0, & z \notin (0,1). \end{cases}$$

Математическое ожидание $M[z]$ и дисперсия $D[z]$ БСВ составляют

$$M[z] = \frac{1}{2},$$
$$D[z] = \frac{1}{12},$$

соответственно.

БСВ моделируется на ЭВМ с помощью *датчиков* БСВ. Датчик БСВ - это устройство или программа, выдающая по запросу одно или несколько независимых значений z_1, \dots, z_n БСВ.

Датчики БСВ могут быть трех типов: табличные, физические и программные.

Табличный датчик БСВ - это просто таблица случайных чисел. Основным недостатком такого датчика - ограниченное количество случайных чисел в таблицах. А в статистическом эксперименте часто требуется не ограниченное заранее их количество.

Физический датчик БСВ - это специальное радиоэлектронное устройство в ЭВМ, содержащее источник электронного шума. Шум преобразуется в случайные числа с распределением. Недостатки физического датчика БСВ: невозможность повторения каких-либо ранее полученных реализаций z_1, \dots, z_n без их предварительной записи в память ЭВМ, схемная нестабильность и сложность тиражирования датчика.

Программный датчик БСВ обычно вычисляет значения z_1, z_2, \dots , по какой-либо рекуррентной формуле типа

$$z_i = f(z_n),$$

при заданном стартовом значении z_0 .

Заданное значение z_0 полностью определяет всю последовательность реализаций z_1, z_2, \dots , поэтому z часто называют *псевдослучайной* величиной. Но ее статистические свойства идентичны свойствам "чисто случайной" последовательности, что и обеспечивает успех статистического моделирования.

Программный датчик БСВ имеет следующие преимущества: простота создания датчика, простота применения, простота тиражирования, надежность, быстрдействие, высокая точность достижения необходимых статистических

свойств, сравнивая с точностью представления вещественных чисел, компактность, повторяемость, когда это нужно, любых последовательностей случайных значений без их предварительного запоминания.

В дальнейшем мы будем рассматривать только программные датчики БСВ.

Имея датчик БСВ Z , можно промоделировать любые случайные факторы: непрерывные или дискретные случайные величины (как простые, так и многомерные), случайные события, случайные процессы и поля и т.д. Для этого достаточно соответствующим образом преобразовать последовательность z_1, z_2, \dots . Поэтому БСВ Z и называют базовой.

Теоретически в качестве базовой можно было бы взять почти любую случайную величину. Использование СВ Z с равномерным распределением обусловлено технологическими соображениями: простотой и экономичностью датчика, простотой преобразования Z в другие случайные факторы, относительной простотой тестирования датчика.

4.1.2. Метод середины квадрата

Метод середины квадрата предложен для получения псевдослучайных чисел Д. фон Нейманом в 1946 г. Один из вариантов этого метода заключается в следующем.

1. Возьмем произвольное n -разрядное число.
2. Возведем полученное число в квадрат и, если необходимо, добавим к результату слева нули до $2n$ -разрядного числа.
3. Возьмем четыре цифры из середины $2n$ -разрядного в качестве нового случайного n -разрядного числа.
4. Если нужны еще случайные числа, то перейдем к пункту 2.

Например, если взять в качестве начального числа 1994, то из него получается следующая последовательность псевдослучайных чисел: 9760 2576 6357 4114 9249 5440 5936 2360 5696 4444 7491 1150 3225 4006 0480 2304 3084 5110 1121 2566 ...

Сам по себе метод середины квадрата не получил широкого распространения, так как выдает "больше чем нужно малых значений". Но открытый в нем принцип используется во многих, если не во всех, более поздних датчиках БСВ. Этот принцип состоит в вырезании нескольких цифр из результата какой-либо операции над числами.

4.1.3. Мультипликативный конгруэнтный метод

Так называемый мультипликативный конгруэнтный датчик БСВ задается двумя параметрами: модулем m и множителем k . Обычно это достаточно большие целые числа.

При заданных m, k числа z_1, z_2, \dots , вычисляются по рекуррентной формуле:

$$A_i = (kA_{i-1}) \bmod m, \quad i = 1, 2, \dots, \quad (6.1)$$

$$z_i = A_i / m,$$

где m - модуль,

k - множитель,

A_0 - начальное значение,

\bmod - операция вычисления остатка от деления kA_{i-1} на m .

Таким образом, A_1 определяется как остаток от деления kA_0 на m ; A_2 - как остаток от деления kA_1 на m и т.д. Поскольку все числа A_i - это остатки от деления на m , то $0 \leq A_i < m$. Разделив последнее неравенство на m , видим, что $0 \leq A_i / m < 1$, т. е. $0 \leq z_i < 1$.

Из неравенства $0 \leq A_i < m$ вытекает также, что датчик (6.1) дает периодическую последовательность A_i . Действительно, число всех возможных остатков от 0 до $m - 1$ равно m и, рано или поздно, на каком-то шаге i обязательно появится значение A_i , уже встречавшееся ранее. С этого момента последовательность A_i "зациклится".

Длина периода T будет не больше $m - 1$. Например, если встретится остаток $A_i = 0$, то далее, согласно (4.1), будет $A_{i+1} = 0$, $A_{i+2} = 0$, ..., т.е. длина периода $T = 1$. Ненулевых же остатков в интервале $0 \leq A_i < m$ всего $m - 1$, и, если все они войдут в период, будет $T = m - 1$. Это имеет место, например, при $m = 13$, $k = 7$; в этом случае ряд A_i выглядит так:

1, 7, 10, 5, 9, 11, 12, 6, 3, 8, 4, 2, 1, 7,
 \underline{\hspace{10em}}/

$$T = m - 1 = 12$$

Поскольку в качестве случайной можно использовать лишь подпоследовательность A_i внутри одного периода, то параметры датчика выбирают так, чтобы длина периода T была максимальной. С учетом ограничения $T \leq m - 1$ модуль m берут максимально возможным. Чтобы упростить вычисление остатков по (6.1), для двоичных ЭВМ часто берут $m = 2^n$. Рекомендуется также брать достаточно большой множитель k , причем взаимно простой с m .

В можно найти подробные рекомендации по выбору параметров m , k и начального значения A_0 . Заметим, однако, что в настоящее время не известны правила, которые гарантировали бы высокое качество датчика без его специального статистического тестирования.

Датчик (4.1) называют мультипликативно-конгруэнтным потому, что он использует две основные операции - умножение (англ. multiplication) и вычисление остатка (в теории чисел - получение конгруэнтного числа). Можно было бы поэтому перевести его название и как "множительно-остатковый датчик".

В качестве примера рассмотрим таблицу параметров датчиков, предлагаемых в некоторых публикациях и программных продуктах.

Место использования датчика (программный продукт или публикация)	модуль m	множитель k
Язык моделирования СИМУЛА	2^{35}	5^{16}
Пакеты LLRANDOM, IMSL	$2^{31} - 1$ (простое число)	16807
Язык моделирования SIMSCRIPT	$2^{31} - 1$	63036001

6.2. Характеристики датчиков базовых случайных величин

Практика показывает, что результаты имитационного моделирования существенно зависят от качества используемых последовательностей псевдослучайных чисел. Поэтому используемые в имитационной модели генераторы случайных чисел должны пройти тесты на пригодность. Основные анализируемые характеристики генерируемых датчиком последовательностей:

- равномерность;
- стохастичность (случайность);
- независимость.

Рассмотрим методы проведения такого анализа, наиболее часто применяемые на практике.

4.2.1. Тестирование равномерности

Обозначим равномерное распределение вероятностей на интервале $(0,1)$ через $R[0,1]$. Тогда утверждение, что БСВ Z имеет распределение $R[0,1]$, можно кратко записать в виде $z \sim R[0,1]$.

С помощью статистических тестов проверяют два свойства датчика, делающих его точной моделью идеальной БСВ, - это *равномерность* распределения чисел Z_i , выдаваемых датчиком на интервале $(0,1)$, и их статистическая *независимость*. При этом числа z_i рассматривают как реализации некоторой **СВ.**, т.е. как статистическую выборку.

Достаточно простым методом проверки равномерности распределения является частотный тест. Он основан на законе больших чисел и выполняется по следующему алгоритму.

1. Разобьем интервал $(0,1)$ на K равных отрезков (например, $K = 10$).
2. Сгенерируем n чисел z_1, \dots, z_n с помощью тестируемого датчика БСВ (например, $n = 100$).
3. Подсчитаем, сколько чисел попало в каждый из k отрезков, т.е. найдем числа попаданий n_1, \dots, n_k .
5. Рассчитаем относительные частоты попаданий в отрезки:

$$\hat{p}_1 = \frac{n_1}{n}, \dots, \hat{p}_k = \frac{n_k}{n}$$

5. Построим гистограмму частот $\hat{p}_1, \dots, \hat{p}_k$ на K отрезках интервала $(0,1)$.
6. Повторим действия (2) - (5) для большего значения n (например, для $n = 10000$).
7. Оценим по полученным гистограммам сходимость каждой частоты \hat{p}_i к вероятности $p = 1/K$ того, что БСВ попадет в i -й отрезок. Согласно закону больших чисел должно быть

$$\hat{p}_i \xrightarrow{P} \frac{1}{K}, \quad n \rightarrow \infty \quad (6.2.)$$

Это значит, что высоты столбиков во второй гистограмме должны в целом быть ближе к уровню $1/K$, чем в первой.

Тестирование датчика на равномерность можно совместить с оцениванием **математического ожидания** m^* и дисперсии S^2 . Оценки m^* и S^2 рассчитываются соответственно по формулам:

$$m^* = \frac{1}{n} \sum_{i=1}^n z_i \quad (6.3)$$

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (z_i - m^*)^2 \quad (6.4)$$

С ростом n оценки m^* и S^2 должны сходиться по вероятности к точным значениям $M(z) = 1/2$, $D(z) = 1/12 = 0.08333...$.

6.2.2. Тестирование стохастичности

Рассмотрим один из основных методов проверки – метод комбинаций. Суть его сводится к следующему. Выбирают достаточно большую последовательность случайных чисел x_i и для нее определяют вероятность появления в каждом из x_i ровно j единиц. При этом могут анализироваться как все разряды числа, так и только l старших. Теоретически закон появления j единиц в l разрядах двоичного числа может быть описан как биномиальный закон распределения (исходя из независимости отдельных разрядов).

Тогда при длине выборки N ожидаемое число появлений случайных чисел x_i с j единицами в проверяемых l :

$$n_j = N \cdot C_l^j p_1^j p_0^{l-j} = 2^{-l} N \cdot C_l^j$$

Для полученной последовательности определяется эта же характеристика. Проверка соответствия реального значения теоретическому выполняется с помощью одного из статистических критериев согласия.

4.2.3. Тестирование независимости

Простейшую проверку статистической независимости реализаций z_1, z_2, \dots , можно осуществить, оценивая корреляцию между числами z_i и z_{i+s} , отстоящими друг от друга на шаг $s > 1$.

Для вывода формулы, по которой можно рассчитать коэффициент корреляции чисел z_i и z_{i+s} , рассмотрим две произвольные **с.в.** x, y . Коэффициент корреляции определяется для них формулой:

$$R(x, y) = \frac{M(xy) - M(x)M(y)}{\sqrt{D(x)D(y)}} \quad (6.5)$$

С ростом n оценка R^l должна приближаться к нулю, в противном случае датчик БСВ не отвечает требованию независимости.

Конечно, если R' сходится к нулю, то это еще не гарантирует наличие независимости, но все же один из тестов оказывается успешно выдержанным. При желании всегда можно продолжить испытания датчика другими методами.

Еще одна важная характеристика датчика СЧ — **длина отрезка периодичности L** . Если в основу работы датчика положен мультипликативный метод, то оценить L несложно: она определяется величиной константы M .

4.3. Случайные события и их имитация

4.3.1. Имитация случайного события

Пусть некоторое событие A происходит с вероятностью P_A . Требуется воспроизвести факт наступления события A . Поставим в соответствие событию A событие B , состоящее в том, что x меньше либо равно P_A , где x здесь и в дальнейшем – случайное число (СЧ) с равномерным на интервале $(0,1)$ законом распределения. Вычислим вероятность события B :

$$P(B) = \int_0^{P_A} 1 dy = P_A$$

Таким образом, события A и B являются равновероятными. Отсюда следует процедура имитации факта появления события A . Она сводится к проверке неравенства X_A меньше, либо равно P_A , а алгоритм заключается в следующем:

1. С помощью датчика случайных чисел (СЧ) получают СЧ X ;
2. Проверяют выполнение неравенства X меньше, либо равно P_A ;
3. Если оно выполняется, то событие A – произошло, если нет – то произошло \bar{A}

4.3.2. Имитация сложного события

Имитация сложного события, состоящего, например, из двух независимых элементарных событий A и B , заключается в проверке неравенств:

$$\left. \begin{array}{l} x_1 \leq P_A \\ x_2 \leq P_B \end{array} \right\},$$

где P_A и P_B – вероятности событий A и B , а x_1 и x_2 – СЧ с равномерным законом распределения.

В зависимости от исхода проверки неравенств (аналогично алгоритму 4.2.1.) делается вывод какой из вариантов:

$AB, A\bar{B}, \bar{A}B, \bar{A}\bar{B}$ имеет место.

4.3.3. Имитация сложного события, состоящего из зависимых событий.

В случае, когда сложное событие состоит из элементарных зависимых событий А и В имитация сложного события производится с помощью проверки следующих неравенств:

$$\left. \begin{matrix} x_1 \leq P_A \\ x_2 \leq P_{\bar{B}/A} \end{matrix} \right\} \quad \left. \begin{matrix} x_1 > P_A \\ x_2 \leq P_{\bar{B}/A} \end{matrix} \right\} \quad \left. \begin{matrix} x_1 \leq P_A \\ x_2 > P_{\bar{B}/A} \end{matrix} \right\} \quad \left. \begin{matrix} x_1 > P_A \\ x_2 > P_{\bar{B}/A} \end{matrix} \right\}$$

В зависимости от того, какая из этих четырех систем неравенств выполняется, делается вывод о том, какой из этих четырех возможных исходов $AB, A\bar{B}, \bar{A}B, \bar{A}\bar{B}$ имеет место.

В качестве исходных данных задаются P_A , P_B и условная вероятность $P_{B/A}$, вероятность $P_{\bar{B}/A}$ может быть вычислена. По формуле полной вероятности:

$$P(A) = P(A/B) \cdot P(B) + P(A/\bar{B}) \cdot P(\bar{B}),$$

где

$$P(\bar{B}) = 1 - P(B), \text{ отсюда легко выразить } P(A/\bar{B})$$

4.3.4. Имитация событий, составляющих полную группу

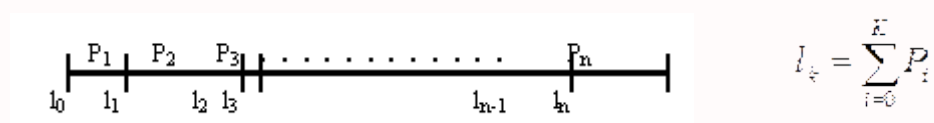
Пусть событие A_i ($i=1,n$) составляют полную группу, тогда их вероятности P_i , таковы что:

$$\sum_{i=1}^n P_i = 1$$

Имитация факта появления одного из событий A_i ($i=1,n$) сводится к проверке следующих неравенств:

$$\sum_{i=0}^{K-1} P_i \leq x < \sum_{i=0}^K P_i, \quad K = \overline{1, n}, \quad P_0 = 0$$

Выполнение К-го неравенства эквивалентно выполнению события A_K . Описанный алгоритм называют иногда алгоритмом "розыгрыша по жребию". Его можно интерпретировать как установление номера К-го отрезка длиной P_K , на который пало СЧ x , при условии разбиения отрезка единичной длины на отрезки с длинами P_1, P_2, \dots, P_n (рис 4.3.)



6.4. Имитация непрерывных случайных величин

6.4.1. Метод обратной функции

Пусть непрерывная случайная величина Y задана своим законом распределения:

$$F_Y(y) = \int_{-\infty}^y f_Y(y) dy,$$

где $f(y)$ – плотность распределения вероятностей, а $F(y)$ – функция распределения вероятностей. Доказано, что случайная величина

$$\xi = \int_{-\infty}^{\xi} f(y) dy = F(y)$$

распределена равномерно на интервале $(0,1)$.

Отсюда следует, что искомое значение y может быть определено из уравнения:

$$x = \int_{-\infty}^y f(y) dy \quad (6.8)$$

которое эквивалентно уравнению:

$$x = F(y) \quad (6.9)$$

где y – значение случайной величины Y , а x – значение СВ X . Решение уравнения (4.9) можно записать в общем виде через обратную функцию

$$y = F^{-1}(x)$$

Основной недостаток метода заключается в том, что интеграл (4.8) не всегда является берущимся, а уравнение (4.9) не всегда решается аналитическими методами.

4.4.2. Метод Неймана (режекции)

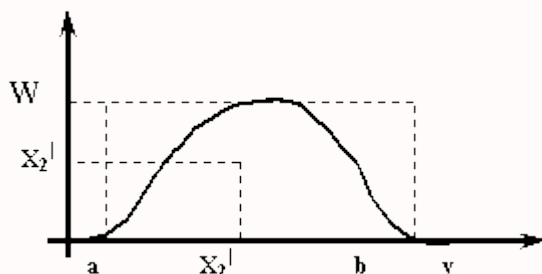
Метод Неймана, так же как метод обратной функции, является методом, позволяющим получить значения СВ в соответствии с заданным законом распределения. Этот метод является достаточно универсальным он применим для моделирования всех СВ, значения которых не выходят за пределы ограниченного интервала (a,b) , а также для СВ, законы распределения которых можно аппроксимировать усеченными.

Метод Неймана состоит в следующем:

С помощью датчика случайных чисел получают пару чисел, распределенных равномерно на $(0,1)$ x_1 и x_2 .

Путем преобразований (по методу обратной функции получают два числа x_1^* и x_2^* , равномерно распределенных соответственно на интервалах (a,b) и $(0,w)$, то есть

$$x_1^* = a + x_1(b - a) \text{ и } x_2^* = x_2' W, \text{ где } W = \max f_n(y)$$



Из точек с координатами x_1^* и x_2^* выбирают те, которые попали "под колокол" функции $f(x)$, то есть те точки, для которых $f(x_1^*) < x_2^*$.

Если условие выполнено, то искомое значение y полагают равным x_1^* .

4.4.3. Алгоритм получения значения нормально распределенной случайной величины.

Нормальное распределение является наиболее часто встречающимся. Функция плотности распределения вероятностей для него имеет вид:

$$f(y) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y-m)^2}{2\sigma^2}\right)$$

где m – математическое ожидание, а σ^2 – дисперсия. Согласно центральной предельной теореме теории вероятностей

$$X = \sum_{i=1}^k \xi_i$$

распределена асимптотически нормально, если ξ_i распределены одинаково.

Для практического получения значений X в качестве ξ_i выбирают равномерно распределенные случайные величины. При этом наиболее часто используют преобразование

$$y = \left(\frac{1}{\sqrt{k/12}} \right) \sum_{i=1}^k x_i - \frac{k}{2} \quad (4.10)$$

где x_i – равномерно распределенные на $(0,1)$ случайные числа. При $k=12$ формула приобретает вид наиболее удобной для расчетов, но она дает достаточно точные

результаты уже для $k=3,4$. Формула (4.10) верна для центрированной ($m=0$) и нормированной ($\sigma=1$) случайной величины.

Для получения y^* , распределенного нормально с произвольными m и σ , пользуются дополнительно преобразованием

$$y^* = m + \sigma y \quad (4.11)$$

4.5. Алгоритмы получения значений систем случайных величин (случайных векторов).

4.5.1. Метод аналитических преобразований.

Пусть системы непрерывных случайных величин (x_1, x_2, \dots, x_n) задана условными законами распределения x_i ($i=1, n$). По теореме умножения плотностей распределения: совместная функция плотности распределения вероятностей

$$f(x_1, x_2, \dots, x_n) = f_1(x_1) f_2(x_2/x_1) f_3(x_3/x_1 x_2) \dots f_n(x_n/x_1, x_2, \dots, x_{n-1}).$$

Для системы двух случайных величин (x_1, x_2) , алгоритм получения вектора ее значений сводится к следующему:

Вычисление частной функции плотности для x_1 :

$$f_1(x_1) = \int_{-\infty}^{+\infty} f(x_1, x_2) dx_2$$

Получение значения x_1 в соответствии с $f_1(x_1)$ согласно любому методу, например, одному из описанных в предыдущем разделе.

Вычисление частной функции плотности для второй компоненты x_2 системы. Она может быть получена на основании теоремы умножения законов распределения:

$$f_2(x_2/x_1) = \frac{f(x_1, x_2)}{f_1(x_1)}$$

Получение x_2 – значения СВ X_2 любым известным методом в соответствии с найденным законом ее распределения.

Алгоритм может быть обобщен для любого n . Однако, практические работы, выполняемые по этому методу, связаны с большими вычислительными трудностями, за исключением тех редких случаев, когда интегралы берутся. Поэтому разработаны другие методы, позволяющие решать задачу получения значений системы непрерывных случайных величин.

4.5.2. Метод разложения по координатным случайным величинам.

Пусть СНСВ задана в рамках теории корреляций: математическими ожиданиями компонент (m_1, m_2, \dots, m_n) и матрицей корреляционных моментов:

$$k = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{bmatrix}, \quad k_{ij} = k_{ji}, k_{ii} = D(X_i)$$

Доказано, что $x_i, (i=\overline{1, n})$ можно получить с помощью их разложения по координатам СВ x_i :

$$\begin{aligned} X_1 &= C_{11}X_1 + M_1 \\ X_2 &= C_{12}X_1 + C_{22}X_2 + M_2 \\ &\vdots \\ X_n &= C_{1n}X_1 + C_{2n}X_2 + \dots + M_n \end{aligned} \quad (4.12)$$

где x_i - некоррелированные, центрированные, нормированные нормально распределенные СВ.

Коэффициенты $c_j, (j = \overline{1, n}, i \leq j)$ могут быть достаточно просто получены решением системы уравнений:

$$k_{ij} = c_{i1}c_{j1} + c_{i2}c_{j2} + \dots + c_{i\overline{i}}c_{j\overline{i}} \quad (i \leq j; i, j = \overline{1, n}) \quad (4.13)$$

Алгоритм получения значений СНСВ сводится к следующему:

- q Решение системы нелинейных уравнений (4.13).
- q Получение n значений y_i нормированных, центрированных СВ, распределенных нормально.
- q Вычисление x_i $i=(1,...,n)$ значений СВ, образующих систему непрерывных случайных величин в соответствии с (4.12).

4.5.3. Алгоритм получения значений системы дискретных случайных величин

Дискретный двумерный вектор СВ задается двумерным законом распределения,

- а) матрицей вероятностей $\|P_{ij}\|, i = \overline{1, n}, j = \overline{1, m}$, где P_{ij} – вероятность совместного появления i -ого и j -ого значений соответственной первой и второй компоненты, причем:

$$\sum_{i=1}^N \sum_{j=1}^M P_{ij} = 1.$$

б) двумя векторами возможных значений первой и второй компоненты $\{A_i\}, \{B_i\}$, $i = 1, n, j = 1, m$.

Для получения значений двумерной дискретной системы случайных величин вычисляют ряд распределения и функцию распределения составляющей X:

$$P_{X_i} = \sum_{j=1}^m P_{ij} \quad \text{и} \quad F(x_i) = \sum_{k=0}^{i-1} P_{X_k}$$

и условные ряды и функции распределения составляющей Y:

$$P_{Yj|X_i} = P_{ij} / P_{X_i} \quad \text{и} \quad F_{Yj|X_i}(y_j) = \sum_{k=0}^{j-1} P_{Yk|X_i}$$

Формирование случайной величины происходит в два этапа. На первом этапе разыгрывается значение составляющей X. Если ξ - равномерно распределенное случайное число из интервала (0,1) такое, что

$$F(x_k) < \xi \leq F(x_{k+1}),$$

то считают, что X компонента двумерной дискретной случайной величины получила k-ое значение.

Сформированное значение составляющей X определяет условный закон распределения составляющей Y

$$F_{Yj|X_k}(y_j) = \sum_{k=0}^{j-1} P_{Yk|X_k},$$

которая формируется аналогично составляющей X.