V.A. Trapeznikov Institute of Control Sciences
Academy of Sciences
RUDN University
Tomsk State University
Institute of Information and Communication
Technologies Bulgarian Academy of Sciences
Research and development company
“Information and networking technologies”

DISTRIBUTED COMPUTER AND COMMUNICATION NETWORKS: CONTROL, COMPUTATION, COMMUNICATIONS (DCCN-2016)

Volume 3
Youth School-Seminar

Proceedings
of the Nineteenth International Scientific Conference

Russia, Moscow, 21–25 November 2016

Under the general editorship
of D.Sc. V. M. Vishnevskiy and D.Sc. K. E. Samouylov

Moscow
2016

Федеральное государственное бюджетное учреждение науки
Институт проблем управления им. В. А. Трапезникова
РОССИЙСКОЙ АКАДЕМИИ НАУК
Федеральное государственное автономное
образовательное учреждение высшего образования
«РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ»
Национальный исследовательский
Томский государственный университет
Институт информационных и телекоммуникационных технологий
БОЛГАРСКОЙ АКАДЕМИИ НАУК
Научно-производственное объединение
«Информационные и сетевые технологии»

РАСПРЕДЕЛЕННЫЕ КОМПЬЮТЕРНЫЕ И ТЕЛЕКОММУНИКАЦИОННЫЕ СЕТИ: УПРАВЛЕНИЕ, ВЫЧИСЛЕНИЕ, СВЯЗЬ (DCCN-2016)

В трех томах

Том 3
Молодежная школа-семинар

Материалы
Девятнадцатой международной научной конференции

Россия, Москва, 21–25 ноября 2016 г.

Под общей редакцией
д.т.н. В.М. Вишневского и д.т.н. К.Е. Самуйлова

Москва
2016

УДК 004.7:004.4.001:621.391:007(063)
ББК 32.973.202:32.968
Р 24

Р 24 **Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2016) = Distributed computer and communication networks: control, computation, communications (DCCN-2016)** : материалы Девятнадцатой международной научной конференции, 21–25 нояб. 2016 г. : в 3 т.; под общ. ред. В. М. Вишневского и К. Е. Самуйлова — М.: РУДН, 2016.
ISBN 978-5-209-07666-7
Т. 3: Молодежная школа-семинар = Youth School-Seminar. — 499 с. : ил.
ISBN 978-5-209-07669-8 (т. 3)

Научная молодежная школа-семинар проводится при финансовой поддержке Российского фонда фундаментальных исследований, Проект № 16-37-10500.

В научном издании представлены материалы Девятнадцатой международной научной конференции «Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь» по следующим направлениям:

— Оптимизация архитектуры компьютерных и телекоммуникационных сетей;
— Управление в компьютерных и телекоммуникационных сетях;
— Оценка производительности и качества обслуживания в беспроводных сетях;
— Аналитическое и имитационное моделирование коммуникационных систем последующих поколений;
— Беспроводные сети 4G/5G и технологии сантиметрового и миллиметрового диапазона радиоволн;
— RFID-технологии и их применение в интеллектуальных транспортных сетях;
— Интернет вещей, носимые устройства, приложения распределенных информационных систем;
— Распределенные системы и системы облачного вычисления, анализ больших данных;
— Вероятностные и статистические модели в информационных системах;
— Теория очередей, теория надежности и их приложения;
— Математическое моделирование высокотехнологичных систем;
— Математическое моделирование и задачи управления.

Сборник материалов конференции предназначен для научных работников и специалистов в области теории и практики построения компьютерных и телекоммуникационных сетей.

Текст воспроизводится в том виде, в котором представлен авторами

Утверждено к печати Программным комитетом конференции

ISBN 978-5-209-07669-8 (т. 3)
ISBN 978-5-209-07666-7

Contents

УДК 004.94+519.62

Реализация винеровского и пуассоновского стохастических процессов в OpenModelica

М. Н. Геворкян[*], А. В. Демидова[*], А. В. Королькова[*], Д. С. Кулябов[*†], Л. А. Севастьянов[*‡]

[*] *Кафедра прикладной информатики и теории вероятностей, Российский университет дружбы народов, ул. Миклухо-Маклая, д.6, Москва, Россия, 117198*

[†] *Лаборатория информационных технологий, Объединённый институт ядерных исследований, ул. Жолио-Кюри 6, Дубна, Московская область, Россия, 141980*

[‡] *Лаборатория теоретической физики, Объединённый институт ядерных исследований, ул. Жолио-Кюри 6, Дубна, Московская область, Россия, 141980*

Аннотация. В открытой реализации компонентно-ориентированного языка моделирования OpenModelica нет средств для моделирования стохастических процессов. В статье рассматривается проблема добавления генераторов случайных распределений, отличных от равномерного, в частности нормального и пуассоновского. Нормальное распределение затем используется для моделирование винеровского случайного процесса, который играет ключевую роль в стохастических дифференциальных уравнениях.

Ключевые слова: псевдослучайные числа, стохастические процессы, процесс Винера, Modelica.

1. Введение

В данной работе рассматривается вопрос генерации нормального, пуассоновского и экспоненциального распределений. Такой выбор мотивирован использованием этих распределений в теории стохастических дифференциальных уравнений. Наиболее общий вид таких уравнений использует два случайных процесса: винеровский и пуассоновский. Винеровский процесс позволяет учесть имплицитную стохастичность моделируемой системы, а пуассоновский процесс — внешнее воздействие. Кратко опишем структуру статьи.

2. Генерирование нормального и пуассоновского распределений

Генераторы псевдослучайных равномерно распределенных чисел являются основой для получения других псевдослучайных последовательностей. Однако, большинство алгоритмов требуют задания случайного числа из интервала $[0, 1]$, в то время как подавляющее большинство генераторов равномерно распределенных псевдослучайных чисел дают последовательность из интервала $[0, m]$, где число m зависит от алгоритма, разрядности операционной системы и процессора.

Для получения чисел из интервала $[0, 1]$ можно поступить двумя способами. Во первых можно нормировать пученную последовательность случайных числе, поделив каждый ее элемент на максимальный элемент последовательности. Такой подход гарантированно даст 1 в качестве одного из случайных чисел. Однако такой способ плох, когда последовательность псевдослучайныхх чисел слишком велика и не умещается в оперативную память. В этом случае лучше использовать второй способ, а именно поделить каждое сгенерированное число на m.

2.1. Генерирование нормального распределения

Метод генерации нормально распределенных псевдослучайных величин предложен в 1958 году Дж. Э. Р. Боксоми П. Мюллером [1] и назван в их честь *преобразованием Бокса-Мюллера.* Метод основан на простом поробразовании, задаваемом двумя формулами. Данное преобразование обычно записывается в двух форматах:

— стандартная форма (она как раз и предложена авторами статьи [1]),

— полярная форма (предложена Дж. Беллом [2] и Р. Кнопом [3]).

Стандартная форма. Пусть x и y — два независимых, равномерно распределенных псевдослучайных числа из интервала $(0, 1)$, тогда числа z_1 и z_2 вычисляются по формуле

$$z_1 = \cos(2\pi y)\sqrt{-2\ln x},\ z_2 = \sin(2\pi y)\sqrt{-2\ln x}$$

и являются независимыми псевдослучайными числами, распределенными по стандартному нормальному закону $\mathcal{N}(0, 1)$ с математическим ожиданием $\mu = 0$ и стандартным среднеквадратичным отклонением $\sigma = 1$.

Полярная форма. Пусть x и y — два независимых, равномерно распределенных псевдослучайных числа из интервала $[-1, 1]$. Вычислим вспомогательную величину $s = x^2 + y^2$, если $s > 1$ и $s = 0$, то значения x и y следует отбросить и проверить следующую пару. Если же $0 < s \geqslant 1$ тогда числа z_1 и z_2 вычисляются по формуле

$$z_1 = x\sqrt{\frac{-2\ln s}{s}},\ z_2 = y\sqrt{\frac{-2\ln s}{s}}$$

и являются независимыми псевдослучайными числами, распределенными по стандартному нормальному закону $\mathcal{N}(0, 1)$.

При компьютерной реализации данного алгоритма предпочтительней использовать полярную форму, так как в этом случае приходится вычислять только одну трансцендентную функцию ln, а не три (ln, sin cos), как в стандартном варианте. Это компенсирует даже тот факт,

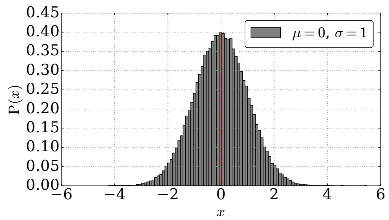

Рис. 1. Нормальное распределение

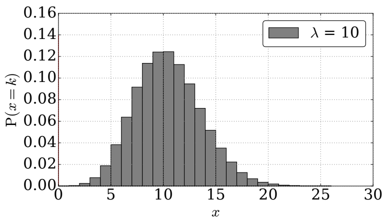

Рис. 2. Распределение Пуассона

что часть исходных равномерно распределенных числе отбрасывается — полярная версия метода все равно работает быстрее. Пример работы алгоритма изображен на рисунке 1

Для получения нормального распределения общего вида из стандартного нормального распределения используют формулу $Z = \sigma \cdot z + \mu$, где $z \sim \mathcal{N}(0, 1)$, а $Z \sim \mathcal{N}(\mu, \sigma)$.

2.2. Генерирование распределения Пуассона

Для генерирования распределения Пуассона существует большое число различных алгоритмов [4–6]. Наиболее простой был предложен Кнутом [7, 8]. Для работы алгоритманеобходимо уметь генерировать равномерные псевдослучайные числа из промежутка $[0, 1]$. Пример работы алгоритма изображен на рисунке 2

3. Определение винеровского и пуассоновского случайных процессов

Случайный кусочно-постоянный процесс $N = \{N_t, 0 \leqslant t \leqslant T\}$ с интенсивностью $\lambda > 0$ называется *процессом Пуассона* (пуассоновским, см. [9]) если выполняются следующие свойства.

1. $P\{N_0 = 0\} = 1$, иначе говоря $N_0 = 0$ почти наверное.
2. N_t — процесс с независимыми приращениями, то есть $\{\Delta N_0, \Delta N_1, \ldots\}$ независимые случайные величины; $\Delta N_{t_i} = N_{t_{i+1}} - N_{t_i}$ и $0 \leqslant t_0 < t_1 < t_2 < \ldots < t_n \leqslant T$; $\Delta N_{t_i} = N_{t_{i+1}} - N_{t_i}$ и $0 \leqslant t_0 < t_1 < t_2 < \ldots < t_n \leqslant T$.
3. Существует число $\lambda > 0$ такое, что для любого приращения $\Delta N_i,\ i = 0, \ldots, n-1$, $E[\Delta N_i] = \lambda \Delta t_i$.
4. Если $P(s) = P\{N_{t+s} - N_t > 2\}$, то $\lim\limits_{s \to 0} \dfrac{P(s)}{s} = 0$.

Случайный процесс $W = \{W_t, 0 \leqslant t \leqslant T\}$ называется скалярным *процессом Винера* (винеровским, см. [9]) если выполняются следующие условия.

1. $P\{W_0 = 0\} = 1$, иначе говоря $W_0 = 0$ почти наверное.
2. W_t — процесс с независимыми приращениями, то есть $\{\Delta W_0, \Delta W_1, \ldots\}$ независимые случайные величины; $\Delta W_{t_i} = W_{t_{i+1}} - W_{t_i}$ и $0 \leqslant t_0 < t_1 < t_2 < \ldots < t_n \leqslant T$.
3. $\Delta W_i = W_{t_{i+1}} - W_{t_i} \sim \mathcal{N}(0, t_{i+1} - t_i)$ где $0 \leqslant t_{i+1} < t_i < T$, $i = 0, 1, \ldots, n-1$.

Из определения следует, что ΔW_i — нормально распределенная случайная величина с математическим ожиданием $\mathbb{E}[\Delta W_i] = \mu = 0$ и дисперсией $\mathbb{D}[\Delta W_i] = \sigma^2 = \Delta t_i$.

Винеровский процесс является моделью *броуновского движения* (хаотического блуждания). Если рассмотреть процесс W_t в те моменты времени $0 = t_0 < t_1 < t_2 < \ldots < t_{N-1} < t_N$ когда он испытывает случайные аддитивные изменения, то непосредственно из определения следует: $W_{t_1} = W_{t_0} + \Delta W_0$, $W_{t_2} = W_{t_1} + \Delta W_1$, $\ldots$, $W(t_N) = W(t_{N-1}) + \Delta W_{n-1}$, где $\Delta W_i \sim \mathcal{N}(0, \Delta t_i)$, $\forall i = 0, \ldots, n-1$.

Запишем W_{t_n} в виде накопленной суммы: $W_{t_n} = W_{t_0} + \sum\limits_{i=0}^{n-1} \Delta W_i$ и учтем, что $\mathbb{E}[\Delta W_i] = 0$ и $\mathbb{D}[\Delta W_i] = \Delta t_i$ можно показать, что сумма нормально распределенных случайных чисел ΔW_i также является нормально распределенным случайным числом:

$$\mathbb{E} \sum_{i=0}^{n-1} \Delta W_i = 0, \quad \mathbb{D} \sum_{i=0}^{n-1} \Delta W_i = \sum_{i=0}^{n-1} \Delta t_i = T, \quad \sum_{i=0}^{n-1} \Delta W_i \sim \mathcal{N}(0, T).$$

Многомерный винеровский процесс $\mathbf{W}_t \colon \Omega \times [t_0, T] \to \mathbb{R}^m$ определяют как случайный процесс составленный из совместно независимых одномерных винеровских процессов $W_t^1, \ldots, W_t^m$. Приращения ΔW_i^α, $\forall \alpha = 1, \ldots, m$ являются совместно независимыми нормально распределенными случайными величинами. С другой стороны вектор ΔW_i^α можно представить как многомерную нормально распределенную случайную величину с вектором математических ожиданий $\mu = 1$ и диагональной матрицей ковариаций.

4. Генерирование пуассоновского и винеровского процессов в OpenModelica

4.1. Генерирование винеровского процесса

Для симулирования одномерного винеровского процесса необходимо сгенерировать N нормально распределенных случайных чисел $\varepsilon_1, \ldots, \varepsilon_N$ и построить их накопленные суммы ε_1, $\varepsilon_1 + \varepsilon_2$, $\varepsilon_1 + \varepsilon_2 + \varepsilon_3$

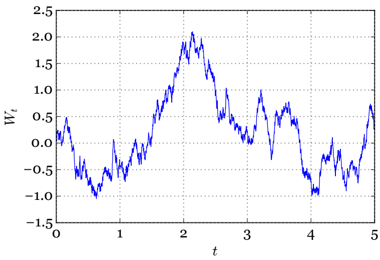

Рис. 3. Винеровский процесс

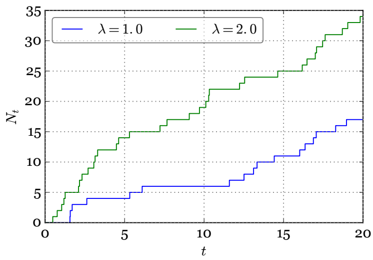

Рис. 4. Пуассоновский процесс

... В результате мы получим *выборочную траекторию* винеровского процесса $W(t)$ см. рис. 3.

В случае многомерного случайного процесса следует сгенерировать уже m последовательностей из N нормально распределенных случайных величин.

4.2. Генерирование пуассоновского процесса

Симулирование пуассоновского процесса во многом аналогично винеровскому, но теперь необходимо сгенерировать последовательность из чисел распределенных по пуассоновскому закону и затем вычислить их накопленную сумму. График процесса изображен на рис. 4. Из графика видно, что пуассоновский процесс представляет собой скачкообразное изменение числа произошедших с течением времени событий. От интенсивности λ зависит среднее количество событий за отрезок времени.

Из за такого характерного поведения пуассоновский процесс также называется скачкообразным, а стохастические дифференциальные уравнения где он участвует в качестве второго ведущего процесса получили названия уравнений со скачками [10]

4.3. Описание примера реализации в OpenModelica

Опишем реализацию генератора нормального распределения и винеровского процесса. Будем предполагать, что генератор равномерно-распределенных случайных величин уже реализован в виде функции `urand`. Для генерирования нормального воспользуемся вышеописанным преобразованием Бокса-Мюллера, а элементы последовательности, задающей винеровский процесс вычислим как кумулятивную сумму нормально-распределенных чисел.

Минимальный рабочий вариант кода приведен ниже.Ключевым моментом является использование оператора `sample(t_0, h)`, который генерирует события через `h` секунд, начиная с момента времени t_0. При каждом срабатывании оператора sample вызывается функция urand, которая возвращает новое случайное число.

```
model generator
  Integer x1, x2;
  Port lg; "Порт для линейного генератора"
  Port normal; "Порт для нормально распределенных чисел"
  Port wiener; "Порт для винеровского процесса"
  Integer m = 429496729; "Значение модуля (маски)"
  Real u1, u2;
initial equation
  x1 = 114561;
  x2 = 148166;
algorithm
  when sample(0, 0.1) then
    x1 := urand(x1);
    x2 := urand(x2);
  end when;
  lg.data[1] := x1 / m;
  lg.data[2] := x2 / m;
  u1 := lcg.data[1];
  u2 := lcg.data[2];
  // нормальный генератор и винеровский процесс
  normal.data[1] := sqrt(-2 * log(u1)) * sin(6.28 * u2);
  normal.data[2] := sqrt(-2 * log(u1)) * cos(6.28 * u2);
  wiener.data[1] := wiener.data[1] + normal.data[1];
  wiener.data[2] := wiener.data[2] + normal.data[2];
end generator;
```

Отметим также применение специальной переменной типа Port, которая служит для соединения различных моделей между собой. В нашем примере мы создали три таких переменных: lg, normal, wiener. Благодаря этому, другие модели могут получить доступ к результату работы нашего генератора.

Ниже представлен минимальный код примера иллюстрирующего соединение моделей между собой. В качестве примера выбрана система из двух обыкновенных дифференциальных уравнений, описывающая осциллятор Ван-дер-Поля–Дуффинга и к ней добавлен стохастический вклад в виде процесса Винера.

$$\dot{x} = y, \; \dot{y} = x(1.0 - x^2) - y + x \cdot W_t.$$

Отметим особо, что это уравнение не является стохастическим. Встроенные в OpenModelica численные методы не позволяют решать стохастические дифференциальные уравнения.

```
model ODE
  Real x, y;
  Port IN;
initial equation
```

```
    x = 2.0;
    y = 0.0;
  equation
    der(x) = y ;
    der(y) = x*(1-x*x) - y + x*IN.data[1];
  end ODE;
  model sim
    generator gen;
    ODE eq;
  equation
    connect(gen.wiener, eq.IN);
  end sim;
```

5. Заключение

В статье были рассмотрены алгоритмы генерации последовательности псевдослучайных чисел, распределенных по нормальному и пуассоновскому закону, а также алгоритмы генерации элементов пуассоновского и винеровского процессов. Данные алгоритмы использовались для реализации вышеперечисленных процессов в среде OpenModelica. Был разобран минимальны работающий пример и указаны некоторые особенности его реализации.

Благодарности

Работа частично поддержана грантами РФФИ № 14-01-00628, 15-07-08795 и 16-07-00556.

Литература

1. Box G. E. P., Muller M. E. A note on the generation of random normal deviates // The Annals of Mathematical Statistics. — 1958. — 06. — Vol. 29, no. 2. — P. 610–611.
2. Bell J. R. Algorithm 334: Normal Random Deviates // Commun. ACM. — 1968. — July. — Vol. 11, no. 7. — P. 498–.
3. Knop R. Remark on algorithm 334 [g5]: Normal Random Deviates // Commun. ACM. — 1969. — May. — Vol. 12, no. 5. — P. 281–.
4. Devroye L. Non-Uniform Random Variate Generation. — New York : Springer-Verlag, 1986.
5. Ahrens J. H., Dieter U. Computer methods for sampling from gamma, beta, poisson and bionomial distributions // Computing. — 1974. — Vol. 12, no. 3. — P. 223–246.
6. Ahrens J. H., Dieter U. Computer generation of poisson deviates from modified normal distributions // ACM Trans. Math. Softw. — 1982. — June. — Vol. 8, no. 2. — P. 163–179.

7. Кнут, Дональд Эрвин. Искусство программирования. — 3 изд. — Москва : Вильямс, 2004. — Т. 2.
8. Knuth D. E. The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms. — Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc., 1997. — Vol. 2.
9. Kloeden P. E., Platen E. Numerical Solution of Stochastic Differential Equations. — 2 edition. — Berlin Heidelberg New York : Springer, 1995.
10. Platen E., Bruti-Liberati N. Numerical Solution of Stochastic Differential Equations with Jumps in Finance. — Heidelberg Dordrecht London New York : Springer, 2010.

UDC 004.94+519.62

Implementation of the Wiener stochastic process in OpenModelica

M. N. Gevorkyan[*], A. V. Demidova[*], A. V. Korolkova[*], D. S. Kulyabov[*†], L. A. Sevastianov[*‡]

[*] *Department of Applied Probability and Informatics*
RUDN University
Miklukho-Maklaya str. 6, Moscow, 117198, Russia
[†] *Laboratory of Information Technologies*
Joint Institute for Nuclear Research
Joliot-Curie 6, Dubna, Moscow region, 141980, Russia
[‡] *Bogoliubov Laboratory of Theoretical Physics*
Joint Institute for Nuclear Research
Joliot-Curie 6, Dubna, Moscow region, 141980, Russia

In an open implementation of component-oriented modeling language OpenModelica there are no tools to simulate stochastic processes. The article considers the problem of adding generators random distributions that are not uniform, in particular the normal and Poisson.. The normal distribution is then used for the simulation of a Wiener random process, which plays a key role in stochastic differential equations.

Keywords: pseudo-random numbers, stochastic processes, Wiener process, Modelica.