XLI Студенческая международная заочная научно-практическая конференция «Молодежный научный форум: технические и математические науки»

ЛИНЕЙНЫЙ КОНГРУЭНТНЫЙ ГЕНЕРАТОР ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ И МЕТОД РАСКРЫТИЯ ЕГО ПАРАМЕТРОВ

Андросова Татьяна Евгеньевна

студент 4 курса, кафедра геоинформатики и информационной безопасности, Самарский университет, РФ, г. Самара

Курочкин Владислав Михайлович

студент 4 курса, кафедра геоинформатики и информационной безопасности, Самарский университет, РФ, г. Самара

Болдырев Артем Сергеевич

студент 4 курса, кафедра геоинформатики и информационной безопасности, Самарский университет, РФ, г. Самара

Чернов Роман Вячеславович

студент 4 курса, кафедра геоинформатики и информационной безопасности, Самарский университет, РФ, г. Самара

В работе рассматривается линейный конгруэнтный генератор псевдослучайных чисел, описан способ нахождении параметров генератора, а также разобран пример работы алгоритма их поиска.

Линейный конгруэнтный генератор.

Линейный конгруэнтный генератор (ЛКГ) является простейшим генератором псевдослучайных чисел. Алгоритм его работы заключается в следующем:

1. Выбираются числа-параметры a, m и c. Свойства, которыми должны обладать данные числа будут рассмотрены ниже.

2. Выбирается некоторое число $s_0 < m$.

3. Все последующие числа находятся по формуле $s_n = (a*s_{n-1} + c)\ mod\ m$.

Последовательность, выдаваемая ЛКГ, называется линейной конгруэнтной последовательностью, максимальный период которой равен m. Ее период максимален в том случае, когда параметры a, m и c обладают следующими свойствами:

1. Числа c и m взаимно просты.

2. Число a-1 кратно p для каждого простого p, являющегося делителем m.

3. Число a-1 кратно 4, если m кратно 4.

Данный генератор псевдослучайных чисел (ГПСЧ) не является криптостойким, поэтому он не подходит для использования в криптографических алгоритмах, но все же находит свое применение для задач моделирования случайных процессов.

Андросова Т.Е., Курочкин В.М., Болдырев А.С. [и др.] Линейный конгруэнтный генератор псевдослучайных чисел и метод раскрытия его параметров // Молодежный научный форум: Технические и математические науки: электр. сб. ст. по мат. XLI междунар. студ. науч.-практ. конф. № 1(41).
URL: https://nauchforum.ru/archive/MNF_tech/1(41).pdf *(дата обращения: 08.09.2025)*

Нахождение параметров ЛКГ.

В данном разделе мы рассмотрим алгоритмы нахождения параметров *a*, *c* и *m* по известной последовательности выданных генератором чисел.

Нахождение параметра *m*.

Обозначим через t_{n+1} разность чисел s_{n+1} и s_n. Тогда получаем:

$t_{n+1} = s_{n+1} - s_n = (as_n - c) - (as_{n-1} - c) = as_n - as_{n-1} = at_n \ mod \ m$

$t_{n+2} = a^2 t_n \ mod \ m$

$t_{n+2}\, t_n - t_{n+1}^2 = a^2 t_n * t_n - (at_n)^2 = 0 \ (mod \ m)$

Следовательно, число $u_n = |t_{n+2}\, t_n - t_{n+1}^2|$ делится на *m* без остатка. Из теории чисел известно, что, если *u* и *v* – случайно выбираемые целые числа, то вероятность того, что НОД(u, v) = 1, равна $6/\pi^2 = 0.60793$. В нашем случае такую же вероятность будет иметь случай, когда НОД(u_i, u_j) = НОД(m*x, m*y) = m * 1 = m. При этом, чем больше мы имеем чисел вида $u_n = |t_{n+2}\, t_n - t_{n+1}^2|$, тем эта вероятность быстрее будет стремиться к 1, так как НОД(u_1,..,u_n) = НОД(u_1, НОД(u_2,..,u_n)) = ... = НОД(u_1, НОД(u_2, ..., НОД(u_{n-1}, u_n)...)) и НОД(u_1,...,u_{n-1}, 1) = 1.

Нахождение параметра *a*.

Возьмем 4 подряд выданных генератором числа s_k, s_{k+1}, s_{k+2}, s_{k+3}. Введем следующие обозначения: $x = s_{k+2} - s_k$, $b = s_{k+3} - s_{k+1}$.

В результате получаем:

$x = as_{k+1} + c - s_k = a^2 s_k + ac + c - s_k = s_k(a^2-1) + c(a+1) = (a+1)(as_k - s_k + c) = (a+1)(s_{k+1} - s_k)$.

$b = as_{k+2} + c - as_k - c = a(a^2 s_k + ac + c) - as_k = a(a^2 s_k + ac + c - s_k) = a(s_k(a^2-1) + c(a+1)) = a(a+1)(as_k - s_k + c) = a(a+1)(s_{k+1} - s_k)$.

Тогда, найдя обратный к *x* элемент по модулю *m*:

$x^{-1} = (a+1)^{-1}(s_{k+1} - s_k)^{-1}$,

получим следующее выражение:

$b * x^{-1} = a(a+1)(s_{k+1} - s_k) * (a+1)^{-1}(s_{k+1} - s_k)^{-1} = a \ (mod \ m)$

Нахождение параметра *c*.

Взяв из предыдущего пункта числа, например, s_{k+3} и s_{k+2}, найдем *c*:

$s_{k+3} - as_{k+2} = as_{k+2} + c - as_{k+2} = c \ (mod \ m)$

Усовершенствование генератора.

Линейный конгруэнтный метод генерации псевдослучайных чисел используется во многих компиляторах известных языков программирования. Однако, кроме шагов алгоритма, рассмотренных выше, добавляется еще один шаг, в ходе которого из результата берутся только часть битов.

Так, например, в компиляторах Borland C/C++, Watcom C, Microsoft Visual/Quick C/C++ отбрасываются младшие 16 бит и один старший.

Пример.

Рассмотрим пример вычисления параметров генератора. Сгенерируем псевдослучайную последовательность, выбрав случайные параметры. Например, $a = 52$, $c = 65$, $m = 71$, $s_0 = 7$. Получим последовательность

[3, 8, 55, 14, 12, 50, 38, 53, 52]

Тогда массив с числами t_n будет выглядеть следующим образом:

[5, 47, -41, -2, 38, -12, 15, -1]

Массив с числами u_n:

[2414, 1775, 1562, 1420, 426, 213]

Факторизуем все числа u_n. Получим следующие результаты: $2414 = 2 * 17 * 71$, $1775 = 5^2 * 71$, $1562 = 2 * 11 * 71$, $1420 = 2^2 * 5 * 71$, $426 = 2 * 3 * 71$, $213 = 3 * 71$.

Вычисляем параметр m: m = НОД(2414, 1775, 1562, 1420, 426, 213) = 71.

Для нахождения параметра a возьмем последние 4 числа последовательности: 50, 38, 53, 52. Тогда $x = 53 – 50 = 3$, $b = 52 – 38 = 14$. Найдем обратный к x элемент x^{-1}: $3^{-1} = 24$ (mod 71), так как $3 * 24 = 1$ (mod 71). В итоге получаем, что $a = b*x^{-1} = 14 * 24 = 52$ (mod 71).

Параметр c получается равным $52 – 52 * 53 = – 2704 = 65$ (mod m).

Таким образом, мы нашли параметры генератора с помощью приведенного выше алгоритма. Найденные параметры совпадают с теми, которые использовались при генерации последовательности.

Заключение.

В статье был рассмотрен линейный конгруэнтный генератор псевдослучайных чисел. Данный генератор не является криптографически стойким, что было показано путем составления алгоритма нахождения параметров генератора по известной последовательности, полученной на выходе генератора. Для проверки правильности алгоритма был рассмотрен пример нахождения параметров генератора, которые задавались самостоятельно.

Список литературы:

1. Мао В. Современная криптография: теория и практика: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 768 с.

2. Смарт Н. Криптография, Москва: Техносфера , 2005. – 528 с.

3. Фергюсон Н., Шнайер Б. Практическая криптография: Пер. с англ. – Издательский дом «Вильямс», 2004. – 432 с.

4. Cracking a linear congruential generator – [Электронный ресурс] – Режим доступа: http://security.stackexchange.com/questions/4268/cracking-a-linear-congruential-generator. – Information Security Stack Exchange. – (Дата обращения: 25.12.2016).