2. Кристофидес Н. Теория графов. Алгоритмический подход. – М.: Мир, 1978 г.
3. Эдоус М. Методы принятия решений. – М.: Юнити, 1997 г.

УДК 511.176+510.57

ФОРМУЛА ОБЩЕГО ЧЛЕНА ВОЗВРАТНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ И НЕКОТОРЫЕ АСПЕКТЫ ЕЕ ПРИМЕНЕНИЯ

Е.Ю. Грувер, В.В. Штепин

Резюме. В данном исследовании получена формула общего члена возвратной последовательности k-го порядка в матричной форме. Получена формула частной суммы ряда, составленного из членов данной последовательности. Приведен листинг алгоритма для нахождения n-го члена на языке C++. Описаны примеры задач, в которых использование формулы значительно упрощает их решение.

Ключевые слова: возвратная последовательность, общий член, геноматрица.

Возвратной последовательностью $\{x_n\}$ называется линейное реккурентное соотношение вида $x_{n+k} = a_0 x_n + ... + a_{k-1} x_{n+k-1},\ k \geq 1,\ a_{k-1} \neq 0$ – то есть, линейная зависимость x_n от предыдущих k членов. Возвратные последовательности широко используются при построении псевдослучайных чисел, в развивающейся сейчас отрасли кибернетики – моделировании нейронных сетей, в различных генетических алгоритмах и т.п. Самые простые и известные еще со школы примеры – арифметическая/геометрическая прогрессии и числа Фибоначчи. Классическим методом поиска формулы общего члена является построение характеристического уравнения последовательности: $t_k = a_0 + a_1 t + ... + a_{k-1} t^{k-1}$. Если $t_0,\ t_1,\ ...\ t_{k\text{-}1}$ – различные действительные корни характеристического уравнения, то справедлива формула: $x_n = c_0 t_0^n + c_1 t_1^n + ... + c_{k-1} t_{k-1}^n$, где c_i – константы, которые определяются при подстановке начальных значений ($n = 0..k-1$). В случае если характеристическое уравнение имеет комплексно сопряженные корни, в формуле появляются слагаемые вида $c_i n^m t_i^n$ (то есть, t_i^n умножается не просто на константу c_i, а на многочлен некоторой степени m от n). Таким образом, для нахождения общего члена возвратной последовательности необходимо решить уравнение k-ой степени, а затем решить систему k линейных алгебраических уравнений (СЛАУ) для определения c_i. При k<5 эта задача разрешима при помощи формул Кардано и метода Феррари. Но, как видно из теоремы Абеля-Руффини [1], не существует явного метода (закрытой формулы) для поиска решения уравнения степени выше четвертой в радикалах. Поэтому при k>4 данная задача в общем случае может быть разрешена только с использованием численных методов, что неминуемо ведет к потере точности (не существует общих методов решения задачи без использования ЧМ). А если коэффициенты и начальные значения исходной последовательности были целыми числами, то и вовсе к ошибке – ведь корни характеристического уравнения могут быть иррациональными числами, которые не могут храниться в компьютере в явном виде, а их возведение в степень приведет к еще большей погрешности. Поэтому поиск других формул общего члена играет большую роль при изучении возвратных последовательностей. Этому и посвящена первая часть данной работы. Был обобщен метод геноматриц, применяемый ранее только для последовательностей второго порядка, который заключался в следующем. Если возвратная последовательность имеет вид $x_{n+2} = p x_{n+1} + q x_n$, то справедливо равенство: $\begin{pmatrix} p & q \\ 1 & 0 \end{pmatrix}^n \begin{pmatrix} x_1 \\ x_0 \end{pmatrix} = \begin{pmatrix} x_{n+1} \\ x_n \end{pmatrix}$. В работе была создана и доказана аналогичная формула для k-го порядка. Во второй части приведен код на языке программирования C++, реализующий эту формулу. В третьей части работы при помощи формулы было получено выражение для частной суммы ряда из членов последовательности. Также указаны различные области применения формулы и приведены примеры задач, в которых применение формулы значительно упрощает их решение.

I. Формула общего члена

Определение. Следом матрицы $A = \| a_{ij} \| = \begin{pmatrix} a_{0\,0} & \dots & a_{0\,m} \\ \dots & \dots & \dots \\ a_{n-1\,0} & \dots & a_{n-1\,m-1} \end{pmatrix}$ называется функция $trA = \sum\limits_{i=0}^{\min(n,m)-1} a_{ii}$ (от англ. trace - след). Если A – квадратная матрица, то ее след – сумма элементов на главной диагонали. Если A – вектор-строка или вектор-столбец, то след – элемент a_{00}. Здесь и далее используется 0-индексация.

Определение. Геноматрицей возвратной последовательности k-го порядка $(x_n): x_{n+k} = a_0 x_n + \ldots + a_{k-1} x_{n+k-1},\ k \geq 1,\ a_{k-1} \neq 0$ будем называть матрицу $G\ (k \times k)$:

$$G = \| g_{ij} \| = \begin{pmatrix} a_{k-1} & a_{k-2} & \dots & a_1 & a_0 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix}, \text{ то есть } g_{ij} = \begin{cases} a_{k-1-j},\ i = 0 \\ 1,\ j = i-1 \\ 0,\ иначе \end{cases}.$$

Также введем обозначения: $X_n = (x_{n+k-1} \ldots x_n)^t$, $X = X_0 = (x_{k-1} \ldots x_0)^t$.

Лемма 1. Верно следующее утверждение: $G\,X_n = X_{n+1}$. Или в явном виде:

$$\begin{pmatrix} a_{k-1} & a_{k-2} & \dots & a_1 & a_0 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_{n+k-1} \\ x_{n+k-2} \\ \dots \\ x_{n+1} \\ x_n \end{pmatrix} = \begin{pmatrix} x_{n+k} \\ x_{n+k-1} \\ \dots \\ x_{n+2} \\ x_{n+1} \end{pmatrix}.$$

Доказательство. Достаточно заметить, что скалярное произведение первой строки G и вектор-столбца X_n есть x_{n+k}, а любой другой i-ой строки и X_n – элемент x_{n+k+i}:

$$\sum_{j=0}^{k-1} g_{ij} \cdot x_{n+k-1-j} = \sum_{j=0..k-1}^{j \neq i-1} (0 \cdot x_{n+k-1-j}) + 1 \cdot x_{n+k-1-(i-1)} = x_{n+k-i},\ i \neq 0;$$

$$\sum_{j=0}^{k-1} g_{0j} \cdot x_{n+k-1-j} = \sum_{j=0}^{k-1} a_{k-1-j} \cdot x_{n+k-1-j} = x_{n+k}.$$

Теорема 1. $\forall n \in \mathbb{N} \cup \{0\}$ верно следующее соотношение: $G^n X = X_n$. В явном виде:

$$\begin{pmatrix} a_{k-1} & a_{k-2} & \dots & a_1 & a_0 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix}^n \cdot \begin{pmatrix} x_{k-1} \\ x_{k-2} \\ \dots \\ x_1 \\ x_0 \end{pmatrix} = \begin{pmatrix} x_{n+k-1} \\ x_{n+k-2} \\ \dots \\ x_{n+1} \\ x_n \end{pmatrix}.$$

Для доказательства достаточно воспользоваться леммой 1 и методом математической индукции (ММИ) по n, поскольку $G^{n+1}X = GG^nX = GX_n$.

Следствие из Теоремы 1 (формула общего члена): $x_{n+k-1} = tr\,(G^nX)$.

Таким образом, получена новая формула общего члена возвратной последовательности произвольного порядка. Заметим, что для применения этой формулы не нужны никакие дополнительные вычисления – ни решение многочленного уравнения k-го порядка, ни решение СЛАУ. Только возведение матрицы в степень, что является довольно простой задачей. Следует оценить удобство использования этой формулы. Прежде всего – это линейность, сохранение множества (если все числа a_i принадлежали одному полю K, то и результирующие матрицы будут над тем же полем) – что избавляет от необходимости следить за точностью вычислений и сводит погрешность на ноль, возможность вычисления при помощи бинарного возведения в степень за $O(logN)$ [2], возможность с той же асимптотикой считать частные суммы ряда (используя формулу геометрической прогрессии), многие другие.

II. Реализация формулы на языке C++

Приведенный ниже код в самой простой форме реализует поиск n-го члена возвратной последовательности и выводит первые двадцать чисел Фибоначчи. Рассмотрим асимптотику этого алгоритма. Умножение матриц происходит за $O(k^3)$ операций (в оптимальном варианте за $O(k^{2,3727})$ – алгоритм Вирджинии Уильямс [3], являющийся обобщением алгоритма Штрассена), всего выполняется $O(\log n)$ таких умножений (асимптотика бинарного возведения в степень). Итоговая асимптотика – $O(k^3 \log n)$. Если n достаточно большое относительно k (в частности, при $\frac{n}{\log n} > k^2$) – это решение отработает заметно быстрее, чем классические алгоритмы.

Следует также заметить, что при близких $n / \log n$ и k^2 можно сохранить и быструю асимптотику, и матричную форму. Для этого следует возводить матрицу в степень именно так, как это происходит в доказательстве теоремы 1 – умножая текущее произведение на исходную матрицу слева. При этом возможно упростить умножение: при вычислении произведения $G\,G^{n-1}$ достаточно все строки G^{n-1} , начиная со второй, сдвигать на одну вниз, а вычислять как скалярное произведение только первую. Таким образом, асимптотика составит $O(k\,n)$.

```
#include<vector>
#defineVvector
#define VI V<int>
#define VII V<VI >
#define pb push_back
#define sz size()
#define FOR(i,l,r) for(int i=l;i<r;++i)
#define all FOR(t,0,k)
#define alli FOR(i,0,k)
#define allj FOR(j,0,k)
using namespace std;

struct matrix {
	VII m; int k;
	matrix (int l) {k=l; all
m.pb(VI(k));}
	matrix () {matrix z(1); *this=z;}
	matrix (VII y) {matrix z(y.sz); all
alli z.m[t][i]=y[t][i]; *this=z;}
	matrix operator* (matrix z) {
		matrix c(z.k);
		all alli allj
		c.m[t][i]+=m[t][j]*z.m[j][i];
	return c; }
```

```
};

matrix bin_pow (matrix a, int n) {
	int k=a.k; matrix res(k);
	all res.m[t][t]=1;
	while (n) {
		if (n&1) res=res*a;
	a=a*a; n>>=1; }
return res; }

int n_th (VI a, VI x, int n) {
	int k=a.sz; matrix G(k);
	all G.m[0][t]=a[k-1-t];
	FOR(i,1,k) G.m[i][i-1]=1;
	matrix GN=bin_pow(G,n);
	int ans=0;
	all ans+=GN.m[0][t]*x[k-1-t];
return ans; }

int main () {
	VI a; a.pb(1), a.pb(1);
	VI x; x.pb(0), x.pb(1);
	FOR(t,0,20)
printf("%d\n",n_th(a,x,t));
return 0;
```

III. Формула частной суммы и некоторые свойства геноматрицы

Определение. Матрица A называется невырожденной, если существует обратная к ней. Матрица невырожденная тогда и только тогда, когда ее детерминант ненулевой.

Лемма 2. Пусть дана квадратная матрица A $k \times k$ и единичная матрица E того же порядка. Тогда если матрица (A-E) невырожденная, то справедливо следующее тождество: $E + A + A^2 + ... + A^{n-1} = (A^n - E)(A - E)^{-1}$.

Доказательство леммы 2 очевидно следует из формулы геометрической прогрессии и невырожденности матрицы (A-E).

Лемма 3. Пусть G – геноматрица возвратной последовательности k-го порядка: $\{x_n\} : x_{n+k} = a_0 x_n + ... + a_{k-1} x_{n+k-1},\ k \geq l,\ a_{k-1} \neq 0$. Тогда следующие три утверждения попарно эквивалентны:

1) (G-E) – невырожденная;
2) $\sum_{i=0}^{k-1} a_i \neq 1$;
3) 1 не является собственным значением матрицы G.

Доказательство. Покажем, что $D = \det(G - E) = (-1)^k \left(1 - \sum_{i=0}^{k-1} a_i\right)$. Тогда все три пункта равносильны условию $D \neq 0$: матрица невырожденная тогда и только тогда, когда ее определитель ненулевой, второе неравенство получается умножением D на коэффициент $(-1)^n$, а третье условие эквивалентно следующему тождеству: $D = \det(G - E) = \det((G - \lambda E)(1)) = P(1)$, где $P(\lambda)$ – характеристический многочлен.

Введем следующее обозначение: $\Delta_l = \det \begin{pmatrix} a_{l-1} & a_{l-2} & a_{l-3} & ... & a_0 \\ 1 & -1 & 0 & ... & 0 \\ 0 & 1 & -1 & ... & 0 \\ - & - & - & - & - \\ 0 & ... & 0 & 1 & -1 \end{pmatrix},\ l = 0..k$.

$$\det(G - E) = \det \begin{pmatrix} a_{k-1} - 1 & a_{k-2} & a_{k-3} & ... & a_0 \\ 1 & -1 & 0 & ... & 0 \\ 0 & 1 & -1 & ... & 0 \\ - & - & - & - & - \\ 0 & ... & 0 & 1 & -1 \end{pmatrix} = (a_{k-1} - 1) \det \begin{pmatrix} -1 & 0 & ... & 0 \\ 1 & -1 & ... & 0 \\ - & - & - & - \\ 0 & ... & 1 & -1 \end{pmatrix} -$$

$$-\det \begin{pmatrix} a_{k-2} & a_{k-3} & ... & a_1 & a_0 \\ 1 & -1 & 0 & ... & 0 \\ 0 & 1 & -1 & ... & 0 \\ - & - & - & - & - \\ 0 & ... & 0 & 1 & -1 \end{pmatrix} = (a_{k-1} - 1)(-1)^{k-1} - \Delta_{k-1} = \left\{ \begin{matrix} \textit{аналогично раскладываем} \\ \textit{определитель } \Delta_{k-1} \end{matrix} \right\} =$$

$$= (a_{k-1} - 1)(-1)^{k-1} - \left((a_{k-2})(-1)^{k-2} - \Delta_{k-2}\right) = (-1)^{k-1}(a_{k-1} + a_{k-2} - 1) + \Delta_{k-2} = ... =$$

$$= (-1)^{k-1} \left(\sum_{i=0}^{k-1} a_i - 1 \right) = (-1)^k \left(1 - \sum_{i=0}^{k-1} a_i \right)$$

Таким образом лемма 3 доказана.

Теорема 2. Пусть G – геноматрица возвратной последовательности k-го порядка $x_{n+k} = a_0 x_n + ... + a_{k-1} x_{n+k-1},\ k \geq 1,\ a_{k-1} \neq 0$ и $\det(G - E) \neq 0$. Тогда верно тождество:

$$\sum_{i=0}^{n-1} x_{i+k-1} = tr((G^n - E)\cdot(G-E)^{-1}) .$$

Доказательство теоремы 2 очевидно следует из аддитивности следа матрица и леммы 3.

Рассмотрим теперь область применения возвратных последовательностей в целом и полученных в данной работе формул в частности. Самое распространенное – это генерация псевдопростых чисел по формуле $x_{n+1} = ax_n + b \mod m$. Несмотря на кажущуюся простоту, этот алгоритм весьма эффективен – такого распределения оказывается недостаточно только при тяжелых алгоритмах шифрования, таких как PGP. При этом достаточно редко необходимы все n чисел, обычно достаточно только одного. Многие подобные задачи были неразрешимы в общем случае. Однако с использованием полученных в работе формул достаточно перейти к последовательности $y_n : y_n = x_n - \frac{b}{1-a}$ и модифицировать умножение матриц так, чтобы элементами произведения были не сами скалярные произведения, а остатки при делении их на модуль *m*. Возвратные последовательности нашли применение в разработке генетических алгоритмов и теории искусственных нейронных сетей [4]. Самые эффективные и сложные из них (такие как сети Джордана и Элмана [5]) используют предыдущие значения сигналов/мутаций, поэтому дальнейшие исследования реккурентных соотношений принципиально важны для развития этой отрасли науки. Многие задачи, связанные с численными вычислениями интегралов, также могут быть упрощены с использованием выведенных в данной работе формул.

IV. Выводы

В данной работе были получены формулы общего члена и частной суммы возвратной последовательности произвольного порядка, которые значительно упрощают решение многих задач в различных областях математики и позволяют производить компьютерные вычисления, используя меньше ресурсов. Планируется дальнейшее исследование с целью обобщения результатов работы на нелинейные реккурентные соотношения – прежде всего, полиномиальные.

СПИСОК ЛИТЕРАТУРЫ

1. http://mathworld.wolfram.com/AbelsImpossibilityTheorem.html
2. Кормен, Томас Х., Лейзерсон, Чарльз И., Ривест, Рональд Л., Штайн, Клифорд Глава 28. Работа с матрицами // Алгоритмы: построение и анализ = Introduction to Algorithms. — 2-е издание. — М.: «Вильямс», 2005. — С. 833 - 839.
3. http://www.cs.berkeley.edu/~virgi/matrixmult.pdf
4. Петухов С.В. Матричная генетика, алгебры генетического кода, помехоустойчивость. – Ижевск: НИЦ «Регулярная и хаотическая динамика», 2008.
5. Elman, J.L. Finding structure in time. // Cognitive Science. — 1990. — С. 179-211.