Rising Greatest Factorial Factorization for Gosper's Algorithm

2 authors:

Husam L. Saad
University of Basrah

Mohammed Abdullah
University of Basrah

Rising Greatest Factorial Factorization for Gosper's Algorithm

Husam L. Saad ***Mohammed Kh. Abdullah***

Department of Mathematics, College of Science, Basrah University, Basrah, Iraq.

Hus6274@hotmail.com

Abstract

In this paper we define the "rising greatest factorial factorization" (RGFF) of polynomials. It is a canonical form representation which can be viewed as an analogue to the greatest factorial factorization (GFF) V.Z. Gathen, etc. (1999) and P.Paule (1995), but with a positive integer shifts instead of negative integer shifts. We give lemma to compute the RGFF of any polynomial. We use this canonical representation and greatest common devisor (gcd) concept to give an approach for Gosper's algorithm J.C. Lafon (1983), P. Lisoněk (1993), Y. Man (1993) and M. Petkovšek, etc. (1999).

Key words: Gosper's algorithm, hypergeometric solution, greatest factorial factorization.

1. Introduction

Let $\mathbb{N}$ be the set of natural numbers, $\mathbb{K}$ be the field of characteristic zero, $\mathbb{K}(n)$ be the field of rational functions of n over $\mathbb{K}$, $\mathbb{K}[n]$ be the ring of polynomials of n over $\mathbb{K}$, if $p(n) \in \mathbb{K}[n]$ is a non zero polynomial we will denote its leading coefficient by $lc(p(n))$, a nonzero polynomial $p(n) \in \mathbb{K}[n]$ is said to be monic if $lc(p(n)) = 1$, $gcd(p, q)$ denotes the greatest common devisor for any polynomials $p, q \in \mathbb{K}[n]$. We assume that the gcd always takes a value as a monic polynomial. The pair $\langle f, g\rangle$ $f, g \in \mathbb{K}[n]$ is called the reduced form of a rational function$r(n)$ if $r(n) = \frac{f}{g}$, g monic and $gcd(f, g) = 1$.

A nonzero term t_n is called a hypergeometric term over $\mathbb{K}$ if there exists a rational function $r(n) \in \mathbb{K}(n)$ such that

$$r(n) = \frac{t_{n+1}}{t_n} \cdot$$

For any monic polynomial $p(n) \in \mathbb{K}[n]$ and $m \in \mathbb{N}$, the m^{th} rising factorial $[p(n)]^{\overline{m}}$ of $p(n)$ is defined as

$$[p(n)]^{\overline{m}} = \prod_{i=0}^{m-1} E^i p(n),$$

where E denote the shift operator defined as $Ep(n) = p(n + 1)$. Note that $[p(n)]^{\overline{0}} = 1$.

In many parts of mathematics and computer science some expressions like $S_n = \sum_{k=0}^{n-1} t_k$ (called indefinite hypergeometric summation), arise in a natural way, for instance in combinatorics or complexity analysis. Usually one is interested in finding a solution for such an expression, Gosper's algorithm is an automatic procedure for evaluating these kinds of sums, provided such an expres sion exists see for example [3,4,5,6,9,11,12]. Given a hypergeometric term t_n and suppose that there exists a hypergeometric term z_n satisfying

$$z_{n+1} - z_n = t_n. \qquad (1.1)$$

In 1978 [10] Gosper developed an algorithm for finding the sum $S_n = \sum_{k=0}^{n-1} t_k$ depends on finding at first the hypergeometric term z_n that satisfies (1.1).

In 1992 [11], Petkovšek used the Gosper-Petkovšek representation, or GP representation, for short, to give an approach for Gosper algorithm. In 1994 [9], Petkovšek gave a derivation for Gosper's algorithm. In 1995 [8], Paule and Strehl gave a derivation of Gosper's algorithm by using the GP representation. In 1995 [7], equipped with the Greatest Factorial Factorization (GFF), Paule presented a new approach to indefinite hypergeometric summation which leads to the same algorithm as Gosper's, but in a new setting. In 2005 [2], Chen and Saad presented a simplified version for Gosper's algorithm by using GP representation. In 2008 [1], Chen et al found a convergence property for the gcd of the raising factorial and falling factorial. Based on this property, they presented an approach for Gosper's algorithm.

2. Rising Greatest Factorial Factorization

In this section "rising greatest factorial factorization" of polynomial is introduced. It is a canonical form representation of polynomial which is can be defined as follows:

Definition 2.1.

We say that $\langle p_1, p_2, \cdots, p_k \rangle_r$, $p_i \in \mathbb{K}[n]$ is a RGFF-form of a monic polynomial $p(n) \in \mathbb{K}[n]$ if the following conditions hold:

(RGFF1) $p(n) = [p_1]^{\overline{1}} [p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}$,

(RGFF2) each $p_i(n)$ monic, and $k > 0$ implies $\deg(p_k) > 0$,

(RGFF3) $i \le j \implies \gcd\left([p_i]^{\overline{i}}, E^{-1} p_j\right) = 1 = \gcd([p_i]^{\overline{i}}, E^{j} p_j)$.

2.1 Computing the RGFF-form

The following lemma is the rule to compute the RGFF of a polynomial $p(n) \in \mathbb{K}[n]$ which is depend on finding gcd between the polynomial $p(n)$ and it's shift Ep.

Lemma 2.1.

Let $p(n) \in \mathbb{K}[n]$ be monic polynomial with RGFF-form$\langle p_1, p_2, \cdots, p_k \rangle_r$. Then

$$\mathrm{RGFF}(\gcd(p, E^{-1}p)) = \langle p_2, p_3, \cdots, p_k \rangle_r \quad \text{and} \quad p_1(n) = \frac{p(n)}{[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}}\,.$$

Proof:

$$\gcd(p, E^{-1}p) = \gcd([p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}, E^{-1}([p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}})$$

$$= p_2 \cdot p_3 \cdot Ep_3 \cdot \cdots \cdot p_k \cdots E^{k-2}p_k$$

$$\cdot \gcd(p_1 \cdot Ep_2 \cdot E^2p_3 \cdots E^{k-1}p_k, E^{-1}p_1 \cdot E^{-1}p_2 \cdots \cdot E^{-1}p_k)$$

From RGFF3 we can easily prove that

$$\gcd\left(p_1 \cdot Ep_2 \cdot E^2p_3 \cdots E^{k-1}p_k, E^{-1}p_1 \cdot E^{-1}p_2 \cdots \cdot E^{-1}p_k\right) = 1$$

Then

$$\gcd(p, E^{-1}p) = p_2 \cdot p_3 \cdot Ep_3 \cdot \cdots \cdot p_k \cdots E^{k-2}p_k$$

$= \langle p_2, p_3, \cdots, p_k \rangle_r$.

Hence

$$\frac{p(n)}{[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}} = \frac{[p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}}{[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}} = [p_1]^{\overline{1}} = p_1(n)$$

Algorithm 2.1. RGFF

INPUT: A monic polynomial $p(n) \in \mathbb{K}[n]$;

OUTPUT: The RGFF-form of $p(n)$ ($\mathrm{RGFF}(p)$).

If $p(n) = 1$ then $\mathrm{RGFF}(p) = \langle \quad \rangle_r$

Otherwise , let $\langle p_2, p_3, \cdots, p_k \rangle_r = \mathrm{RGFF}$ $(\gcd(p, E^{-1}p))$ then:

$$\mathrm{RGFF}(p) = \langle \frac{p(n)}{[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}}, p_2, p_3, \cdots, p_k \rangle$$

Example 2.1.

Compute the RGFF of the monic polynomial

$$p(n) = n^6 + 5n^5 + 5n^4 - 5n^3 - 6n^2$$

Solution. We can write $p(n)$ as $p(n) = (n-1)n^2(n+1)(n+2)(n+3)$. We start with computing $q_1 = \gcd(p, E^{-1}p)$ yielding

$$q_1 = (n-1)n(n+1)(n+2)$$

We continue with q_1 and compute $q_2 = \gcd(q_1, E^{-1}q_1)$ yieldin

$$q_2 = (n-1)n(n+1)$$

Then

$$q_3 = \gcd(q_2, E^{-1}q_2) = n(n-1)$$

and

$$q_4 = \gcd(q_3, E^{-1}q_3) = n-1$$

It is clearly that

$$q_5 = \gcd(q_4, E^{-1}q_4) = 1.$$

Now we can compute RGFF(p) starting with a list containing the last nontrivial gcd which is $q_4 = n-1$, hence

$$\mathrm{RGFF}(q_4) = n-1$$

At this point we use Lemma 2.1. on $p = q_3$ yields

$$\mathrm{RGFF}(q_3) = \langle \frac{n(n-1)}{[n-1]^{\overline{2}}}, n-1 \rangle_r = \langle 1, n-1 \rangle_r$$

again on $p = q_2$ yields

$$\mathrm{RGFF}(q_2) = \langle \frac{n(n-1)(n+1)}{[n-1]^{\overline{3}}}, 1, n-1 \rangle_r = \langle 1,1, n-1 \rangle_r$$

and for $p = q_1$ we have

$$\mathrm{RGFF}(q_1) = \langle \frac{n(n-1)(n+1)(n+2)}{[n-1]^{\overline{4}}}, 1,1, n-1 \rangle_r = \langle 1,1,1, n-1 \rangle_r.$$

Finally we can compute the RGFF for p as

$$\mathrm{RGFF}(p) = \langle \frac{(n-1)n^2(n+1)(n+2)(n+3)}{[n-1]^{\overline{5}}}, 1,1,1, n-1 \rangle_r = \langle n, 1,1,1, n-1 \rangle_r$$

2.2 Fundamental RGFF Lemma

The "gcd-shift" i.e., the gcd of a polynomial $p(n)$ and its shift $Ep(n)$, play a basic role in hypergeometric summation. By using the Fundamental RGFF Lemma, we can compute $\gcd(p, Ep)$ from the RGFF-form of $p(n)$. Also it is a basic result in our approach.

Lemma 2.2. (Fundamental RGFF Lemma)

Given a monic polynomial $p(n) \in \mathbb{K}[n]$ with RGFF-form $\langle p_1, p_2, \cdots, p_k\rangle_r$ then

$$\gcd(p, Ep) = E([p_1]^{\overline{0}}[p_2]^{\overline{1}} \cdots [p_k]^{\overline{k-1}})$$

Proof. The case $k = 0$ is trivial. For $k > 0$,

$$\begin{aligned}\gcd(p, Ep) &= \gcd([p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}, E([p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}))\\ &= \gcd([p_1]^{\overline{1}} \cdots [p_{k-1}]^{\overline{k-1}} p_k E[p_k]^{\overline{k-1}}, E([p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_{k-1}]^{\overline{k-1}}) E^k p_k E[p_k]^{\overline{k-1}})\\ &= E[p_k]^{\overline{k-1}} \gcd([p_1]^{\overline{1}} \cdots [p_{k-1}]^{\overline{k-1}} p_k, E([p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_{k-1}]^{\overline{k-1}}) E^k p_k)\end{aligned}$$

From RGFF3 we get

$$\gcd\left([p_i]^{\overline{i}}, E^k p_k\right) = 1 \qquad \forall 1 \leq i \leq k$$

and

$\gcd(p_k, E[p_i]^{\overline{i}}) = E\gcd(E^{-1}p_k, [p_i]^{\overline{i}}) = 1$ for $i \leq k$

also

$$\gcd\left(p_k, E^k p_k\right) \big| \gcd([p_k]^{\overline{k}}, E^k p_k) = 1$$

Hence

$$\gcd(p, Ep) = E[p_k]^{\overline{k-1}} \gcd([p_1]^{\overline{1}} \cdots [p_{k-1}]^{\overline{k-1}}, E([p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_{k-1}]^{\overline{k-1}})).$$

The rest follows from applying the induction hypothesis.

The Fundamental RGFF Lemma tell us that from the RGFF-form of $p(n)$, i.e. RGFF(p)= $\langle p_1, p_2, \cdots, p_k\rangle_r$, one directly can extract the RGFF-form of its "gcd-shift", i.e. RGFF $(\gcd(p, Ep) = E\langle p_2, p_3, \cdots, p_k\rangle_r)$.

Example2.2.

Let $p(n) = n^6 + 5n^5 + 5n^4 - 5n^3 - 6n^2$, then from Example 2.1 we have $RGFF(p) = \langle n, 1,1,1, n-1\rangle_r$ one immediately gets by Lemma 2.2. that

RGFF($gcd(p, Ep)$) = $E\langle 1,1,1, n-1\rangle_r = \langle 1,1,1, n\rangle_r$.

The following lemma is very important for our approach for Gosper's Algorithm:

Lemma 2.3.

Let $\langle p_1, p_2, \cdots, p_k\rangle_r$ be the RGFF-form of the monic polynomial $g(n) \in \mathbb{K}[n]$. Then

(1) $g_o(n) = \frac{g(n)}{gcd(g,Eg)} = p_1 \cdot p_2 \cdots p_k$

(2) $g_1(n) = \frac{Eg(n)}{gcd(g,Eg)} = Ep_1 \cdot E^2 p_2 \cdots E^k p_k$.

Proof:

From the Fundamental RGFF Lemma we get

$$g_o(n) = \frac{g(n)}{gcd(g, Eg)} = \frac{[p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}}}{E([p_2]^{\overline{1}}[p_3]^{\overline{2}} \cdots [p_k]^{\overline{k-1}})} = p_1 \cdot p_2 \cdots p_k$$

$$g_1(n) = \frac{Eg(n)}{gcd(g,Eg)} = \frac{E([p_1]^{\overline{1}}[p_2]^{\overline{2}} \cdots [p_k]^{\overline{k}})}{E([p_2]^{\overline{1}}[p_3]^{\overline{2}} \cdots [p_k]^{\overline{k-1}})} = Ep_1 \cdot E^2 p_2 \cdots E^k p_k$$

3. An Approach for Gosper's Algorithm

In this section we consider Gosper's algorithm equipped with RGFF concepts to present algebraically motivated approach to the problem. Given a hypergeometric term t_n and suppose that there exists a hypergeometric term z_n satisfying (1,1)

The ratio

$$\frac{z_n}{t_n} = \frac{z_n}{z_{n+1} - z_n} = \frac{1}{\frac{z_{n+1}}{z_n} - 1}.$$

is clearly a rational function of n. Let

$$y(n) = \frac{z_n}{t_n} \cdot$$

Then equation (1.1) can be written as

$$r(n) \cdot y(n+1) - y(n) = 1, \tag{3.1}$$

where $r(n) = \frac{t_{n+1}}{t_n}$ is an unknown rational function of n. Hence we need to find rational solutions of (3.2). Let $\langle a, b\rangle, \langle f, g\rangle$ be the reduced form of $r(n)$ and $y(n)$, respectively, then (3.1) becomes

$$a(n) \cdot \frac{f(n+1)}{g(n+1)} - b(n) \cdot \frac{f(n)}{g(n)} = b(n) \tag{3.2}$$

Vice versa any rational solution $y(n) \in \square(n)$ of equation (1.1) gives rise to a hypergeometric solution of equation (1.1). This means that finding hypergeometric solutions z_n of (3.1) is equivalent to finding rational solutions $y(n)$ of (3.1). In case such a solution $y(n) \in \square(n)$ exist, assume we know $g(n)$ or multiple $V(n) \in \square[n]$ of $g(n)$. From (3.2) we get

$$a(n) \cdot \frac{EU}{EV} - b(n) \cdot \frac{U}{V} = b(n)$$

Hence the problem reduces further to finding a polynomial solution $U(n) \in \square[n]$ of the resulting difference equation with polynomial coefficients,

$$a(n) \cdot V \cdot EU - b(n) \cdot EV \cdot U = b(n) \cdot V \cdot EV \cdot \tag{3.3}$$

Note that $U = f \cdot \frac{V}{g} \cdot$ Let $g_i(n) = \frac{E^i g(n)}{\gcd(g, Eg)}$ $i \in \{0,1\}$. Then equation (3.2) is equivalent to

$$a(n) \cdot g_0 \cdot Ef - b(n) \cdot g_1 \cdot f = b(n) \cdot g_0 \cdot g_1 \cdot \gcd(g, Eg) \cdot \tag{3.4}$$

Now, if $\langle p_1, p_2, \cdots, p_k\rangle_r$, $k > 0$ is the RGFF-form of $g(n)$, it follows from $\gcd(f, g) = 1 = \gcd(g_0, g_1)$ and the Fundamental RGFF Lemma that

$$\begin{aligned} g_o(n) &= p_1 \cdot p_2 \cdots p_k | b(n) \\ g_1(n) &= Ep_1 \cdot E^2 p_2 \cdots E^k p_k | a(n) \end{aligned} \tag{3.5}$$

It follows that

$$p_1 \gcd(p_2 \cdots p_k, \cdot Ep_2 \cdots E^{k-1} p_k) \big| \gcd(E^{-1} a(n), b(n),$$

Hence

$$p_1 \big| \gcd(E^{-1}a(n), b(n))$$

by the same way we can get that

$$p_i \big| \gcd(E^{-i}a(n), b(n), \quad i = 1,2,\cdots,k$$

Now we can compute a multiple $V = [P_1]^{\bar{1}}[P_2]^{\bar{2}} \cdots [P_m]^{\bar{m}}$ of $g(n)$. If $P_1 = \gcd(E^{-1}a(n), b(n))$ then obviously $p_1 | P_1$ Indeed, we shall see below that by exploiting RGFF-form one can extract iteratively p_i- multiples P_i such that $E^i P_i | a(n)$ and $P_i | b(n)$:

Algorithm 3.1. RVMULT

INPUT: The reduced form $\langle □, □ \rangle$, of $□(□) \in □(□)$.

OUTPUT: Polynomials $□_1 □_2 \cdots □_□$ such that $□ = [□_1]^{\bar{1}}[□_2]^{\bar{2}} \cdots [□_□]^{\bar{□}}$ is a multiple of the reduced denominator $□(□)$ of $□(□) \in □(□)$.

(i) Compute $□ = \min\{□ \in \mathbb{N} | \gcd\left(□^{-□}□(□), □(□)\right) = 1 \ \forall □ > □ , \ □ \in \mathbb{Z}\} \cdot$

(ii) Set $□_0 = □$, $□_0 = □$ and compute for □ from 1 to □:

$$□_□ = \gcd(□^{-□}□_{□-1}(□), □_{□-1}(□)), □_□ = □_{□-1} \big| E^{-i}P_i$$

$$□_□ = □_{□-1} \big| E^{-i}P_i,$$

$$□_□ = □_{□-1} | P_i.$$

From equation (3.2) we get

$$a(n) \cdot g(n) \cdot f(n+1) - b(n) \cdot g(n+1) \cdot f(n) = b(n) \cdot g(n) \cdot g(n+1) \qquad (3.6)$$

The next step is to set $□(□) = □(n)$

in equation (3.6). If equation (3.6) can be solved for $f(n) \in \mathbb{K}[n]$ then

$$z_n = \frac{f(n)}{g(n)} \cdot t_n \qquad (3.7)$$

is a hypergeometric solution of (1.1), otherwise no hypergeometric solution of (1.1) exists.

Example 3.1.

Evaluate the sum$S_n = \sum_{k=0}^{n} \frac{k^2 4^k}{(k+1)(k+2)}$.

Solution:

Let $t_n = \frac{n^2 4^n}{(n+1)(n+2)}\cdot$

The term ratio

$$r(n) = \frac{t_{n+1}}{t_n} = \frac{4(n+1)^3}{n^2(n+3)}$$

is a rational function of n. The choice $a(n) = 4(n+1)^3$, $b(n) = n^2(n+3)$ satisfies that $\langle a, b\rangle$, is the reduced form of the rational function $r(n)$. Let $\langle p_1, p_2, \cdots, p_k\rangle_r$ be the RGFF-form of $g(n)$. From the Algorithm RVMULT we get

$$P_1 = n^2 \text{and } P_2 = P_3 = \cdots = P_k = 1\cdot$$

Hence $g(n) = V(n) = P_1(n) = n^2\cdot$

From equation (3.6) we get $4(n+1)\cdot f(n+1) - (n+3)\cdot f(n) = n^2\cdot(n+3)\cdot$

The polynomial $f(n) = \frac{1}{3}(n^2 - 4)$ is a solution to the above equation. By (3.7), we have

$$z_n = \frac{f(n)}{g(n)}\cdot t_n = \frac{4^n(n-2)}{3(n+1)}\cdot$$

Hence from (1.1) we have

$$S_n = \sum_{k=0}^{n} t_k = \frac{4^{n+1}(n-1)}{3(n+2)} + \frac{2}{3}\cdot$$

References

W.Y.C. Chen; P. Paule and H.L. Saad, (2008) Converging to Gosper's algorithm, Advance in Applied Mathematics, 41 351–364.

W.Y.C. Chen and H.L. Saad,(2005) On Gosper-Petkovšk representation of rational functions, J. Symbolic computation, 40 955-963.

V.Z. Gathen, J. and J. Gerhard, (1999) Modern Computer Algebra, Cambridge University Press, 1999.

J.C. Lafon,(1983) Summation in finite terms, in Computer Algebra and Symbolic Computation, 2nd ed. 71 -77, Wien: Springer-Verlag.

P. Lisoněk, P. Paule and V. Strehl ,(1993) Improvement of the degree setting in Gosper’s algorithm, J. Symbolic Computation, 16 (1993) 243-258 .

Y. Man,(1993) On computing closed forms for indefinite summations, J. Symbolic Computation, 16 (1993) 355 – 376.

P. Paule,(1995) Greatest factorial factori-zeation and symbolic summation, J. Symbolic computation, 20 (1995) 235-268.

P. Paule and V. Strehl,(1995) Symbolic summation – some recent development, RISC Linz Report Series 95-11, Johannes Kepler University, Linz. Computer Algebra in Science and Engineering – Algorithms, Systems and Applications, J. Fleischer, J. Grabmeier, F. Hehl, W. Küchlin (eds.), World Scientific, Singapore.

M. Petkovšek,(1994) A generalization of Gosper's algorithm, Discrete Math., 134 (1994) 125- 131.

M. Petkovšek; H.S. Wilf and D. **Zeilberger**(1996), A=B, A.K. Peters.

M. Petkovšek(1992), Hypergeometric solutions of linear recurrences with polynomial coefficients, J. Symbolic Computation, 14 (1992) 243-264.

C. Weixlbaumer(2001), Solutions of Difference Equations with Polynomial Coefficients, Diploma thesis, RISC Linz: Johannes KeplerUniversität, A4040 Linz, Austria, 2001.

التحليل للعامل الأعظم المتصاعد و خوارزمية گوسبر

حسام لوتي سعد **محمد خلف عبدالله**

قسم الرياضيات، كلية العلوم، جامعة البصرة، العراق

Hus6274@hotmail.com

الخلاصة

في هذا البحث نعرف التحليل للعامل الأعظم المتصاعد "rising greatest factorialfactorization" (RGFF)الذي يمثل صيغة تمثيل قياسي لمتعددات الحدود الذي يمكن ان يعطى بشكل مماثل الى التحليل للعامل الأعظم" greatest factorial factorization"(GFF) ولكن باستخدام الاعداد الصحيحة الموجبة بدلا من الاعداد الصحيحة السالبة كأسس للمؤثر (E). نعطي مأخوذة لكيفية حساب RGFF لأي متعددة حدود . نعطي أسلوبا لخوارزمية گوسبر باستخدام صيغة تمثيل قياسي هذه لمتعددات الحدود والعامل المشترك الأكبر" greatest common deviser " (gcd).