*To appear in ACM Communications in Computer Algebra*

# Accelerating indefinite hypergeometric summation algorithms

Eugene V. Zima
Physics & Computer Science Dept.
Wilfrid Laurier University
Waterloo, Canada, N2L 3C5
ezima@wlu.ca

Let $\mathbb{K}$ be a field of characteristic zero, $x$ an independent variable, $E$ the shift operator with respect to $x$, i.e., $Ef(x) = f(x+1)$ for an arbitrary $f(x)$. Recall that a nonzero expression $F(x)$ is called a hypergeometric term over $\mathbb{K}$ if there exists a rational function $r(x) \in \mathbb{K}(x)$ such that $F(x+1)/F(x) = r(x)$. Usually $r(x)$ is called the rational *certificate* of $F(x)$. The problem of indefinite hypergeometric summation (anti-differencing) is: given a hypergeometric term $F(x)$, find a hypergeometric term $G(x)$ which satisfies the first order linear difference equation

$$(E-1)G(x) = F(x). \tag{1}$$

If found, write $\sum_x F(x) = G(x) + c$, where $c$ is an arbitrary constant.

An important notion widely used in the context of algorithmic summation is the *dispersion set* of polynomials $p(x)$ and $q(x)$, which is the set of positive integers $h$ such that $\deg(\gcd(p(x+h), q(x))) > 0$. Another important notion is the largest element of the dispersion set known as the *dispersion.*

One more piece of standard terminology required here is the notion of *shift equivalence* of polynomials: two polynomials $u(x), v(x) \in \mathbb{K}[x]$ are shift equivalent if there exists $h \in \mathbb{Z}$, such that $u(x+h) = v(x)$.

Finally, following [9], define the *factorial polynomial* (a generalization of the falling factorial) for $p(x) \in \mathbb{K}[x]$ as

$$[p(x)]_k = p(x) \cdot p(x-1) \cdot \ldots \cdot p(x-k+1) \tag{2}$$

for $k > 0$ and $[p(x)]_0 = 1$.

The algorithmic treatment of an indefinite summation problem oriented towards efficient implementation in computer algebra systems starts with a classical series of works by Abramov (beginning with his publication [1] which not only gives a factorization-free algorithm for rational summation, but describes its Lisp implementation), works of Gosper [6, 7], Moenck [9] and Karr [8]. The subject is well understood and developed so it is even more surprising that some of the simplest and most understood algorithms for indefinite summation can still be improved, not only theoretically but also practically.

In [7] Gosper described a decision procedure for the hypergeometric term summation, which is widely adopted and used in computer algebra systems. The algorithm is based on simple observation that if a given hypergeometric term $F(x)$ has a hypergeometric anti-difference $G(x)$ (i.e. if it is Gosper summable), then the terms $G(x)$ and $F(x)$ are *similar*: i.e., there exists $Y(x) \in \mathbb{K}(x)$ such that $G(x) = Y(x)F(x)$ (in other words, the anti-difference is a rational function multiple of the summand). This reduces the original summation problem to the problem of finding a rational solution of

$$Y(x+1)r(x) - Y(x) = 1, \tag{3}$$

where $r(x)$ is the rational certificate of the summand.

In order to solve (3), the rational certificate is transformed to so-called Gosper form [1] , i.e. one finds polynomials $P, Q, R$ such that

$$r(x) = \frac{P(x+1)Q(x)}{P(x)R(x)},$$

[1] Sometimes also called Gosper-Petkovšek form, if an extra conditions on P,Q,R is used in this step. The modern term for this representation, *Polynomial Normal Form*, can be found for example in [2].

where $Q(x)$ and $R(x+h)$ are co-prime for all non-negative integers $h$. This reduces the search for a rational solution of (3) to the search of a polynomial $y(x)$ solving the *key equation*:

$$Q(x)y(x+1) - R(x-1)y(x) = P(x). \tag{4}$$

If $y(x)$ is found, then the rational multiple of the summand is $Y(x) = \frac{R(x-1)y(x)}{P(x)}$ and

$$G(x) = F(x)\frac{R(x-1)y(x)}{P(x)}. \tag{5}$$

It is worth mentioning here that Gosper's algorithm is applicable to an arbitrary hypergeometric term (no matter how simple or complex it is). It can be applied to a polynomial $p(x)$, in which case the key equation becomes $y(x+1) - y(x) = p(x)$, and the result is always affirmative [2]. It can be applied to a quasi-polynomial, which is also always Gosper-summable, or to a rational or quasi-rational function. For example, it finds the anti-difference of the following rational input:

$$\sum_x \frac{-2\,x+999}{(x+1)\,(x-999)\,x\,(x-1000)} = \frac{1}{x\,(x-1000)}, \tag{6}$$

One well-known problem with Gosper's algorithm is that its running time complexity depends at least linearly on the dispersion of the rational certificate of the summand, which can be exponentially large in the bit size of the summand. For example, the Gosper form of the certificate of the summand from (6) involves a polynomial of degree 999. After this form is computed, Gosper's algorithm will look for a polynomial solution of the first-order difference equation with the right-hand side being a polynomial of degree about 1000. If found, this solution and the right-hand side will be used to write down the numerator and the denominator of an indefinite sum, which have a large common factor. This common factor cancels after the final substitution in (5). It is obvious that this dependency of the running time of Gosper's algorithm on the dispersion of the input is potentially removable (as the output in (6) is small). In [4] (see also [11]) it was shown that indeed this dependency is non-essential and a summation algorithm was given with running time polynomial in the input size and linear in the output size (which might or might not depend on the value of the dispersion). But Gosper's algorithm behaves similarly for non-rational and non-quasi-rational hypergeometric terms with large dispersion of the certificate of the summand.

**Example.** Consider the application of Gosper's algorithm to the following hypergeometric summand

$$F(x) = \frac{\left(27\,x^3+819\,x^2+246\,x-194\right)(2\,x)!}{(3\,x+91)\,(3\,x+1)\,(x+1)\,(3\,x+94)\,(3\,x+4)\,(x!)^2}. \tag{7}$$

The rational certificate of $F(x)$ is

$$r(x) = 2\,\frac{(3\,x+1)\,(3\,x+91)\,(2\,x+1)\left(27\,x^3+900\,x^2+1965\,x+898\right)}{(27\,x^3+819\,x^2+246\,x-194)\,(3\,x+7)\,(3\,x+97)\,(x+2)},$$

which has dispersion of the numerator and the denominator equal to 32.

After computing the Gosper-Petkovšek form the key equation becomes

$$4\,\,(x+1/2)\,(x+1/3)\,y\,(x+1) - (x+94/3)\,(x+1)\,y\,(x) = \left(x^3+\frac{91}{3}\,x^2+\frac{82}{9}\,x-\frac{194}{27}\right)\left[x+\frac{88}{3}\right]_{28}$$

with the right-hand side of degree 31. This equation has a polynomial solution $y(x)$ of degree 29:

$$\frac{1}{3}\left[x+\frac{88}{3}\right]_{29},$$

[2] Observe cancellation in (5) of $F(x)$ and $P(x)$, which are both equal to $p(x)$ in this case.

which after substitution in (5) forces the denominator $P(x)$ to cancel completely, and final result of summation is

$$\sum_x \frac{\left(27\,x^3+819\,x^2+246\,x-194\right)(2\,x)!}{(3\,x+91)\,(3\,x+1)\,(x+1)\,(3\,x+94)\,(3\,x+4)\,(x!)^2} = \frac{(2\,x)!}{(3\,x+91)\,(3\,x+1)\,(x!)^2}.$$

**Remark.** Sometimes the polynomial $P(x)$ from (4) is called *the universal denominator*, and in this particular example the universal denominator is pessimistically large. What we describe below is applied to (4) before finding the polynomial solution $y(x)$ in an attempt to cancel extraneous terms in this universal denominator in advance. A similar technique can be applied to the algorithms for solving higher order recurrences that are based on computing the universal denominator as the first step.

Note that Gosper's original approach described in [6] did not involve the computation of the dispersion and the Gosper form. It was an attempt to directly find a rational solution $Y(x)$ of (3) by building a continued fraction that "approximates" $Y(x)$. One of the advantages of this approach (also it was not a complete decision procedure) is that (if successful) it produces the rational multiple $Y(x)$ of the summand $F(x)$ in reduced form. In [12] we presented a partial direct algorithm based on evaluation and rational function interpolation, which (if successful) returns $Y(x)$ in reduced form, and has similar to the original Gosper approach behavior.

In this note we propose another approach to the acceleration of the hypergeometric summation which is solely based on the evaluation (and does not involve interpolation). It is a modification of the Gosper decision procedure [7], and is applicable in the case when a hypergeometric term is summable, but has a very large dispersion of the rational certificate (as in the example above). Although this approach is applicable to an arbitrary hypergeometric term, we discuss it only for essential hypergeometric terms, i.e. for the terms which are hypergeometric, but not rational or quasi-rational (these cases have different efficient algorithms [4, 11] and can be treated separately).

We first note that the left hand side of (2) offers a succinct (most compact) representation of the product in the right hand side for large values of $k$, as it requires $\Theta(\log k)$ bits to represent the polynomial $p(x)\cdot p(x-1)\cdot\ldots\cdot p(x-k+1)$ assuming the degree of $p(x)$ is fixed. The same polynomial would require $\Theta(k\log k)$ bits if represented as in [3]. Also, (2) offers easy-to-implement lazy evaluation rules, such as

$$[p(x)]_k = [p(x)]_{k-1}\cdot p(x-k+1),\quad [p(x+1)]_k = p(x+1)\cdot[p(x)]_{k-1}\,,\quad \text{for } k>0,\quad \textit{etc.}$$

For example, for arbitrary $A$ and $B$,

$$A\cdot[p(x)]_k \pm B\cdot[p(x+1)]_k = [A\cdot p(x-k+1)\pm B\cdot p(x+1)]\cdot[p(x)]_{k-1}\,. \tag{8}$$

In what follows let $[p(x)]_k$ be one of the factors of $P(x)$ in (4). Our approach is based on a succinct representation of the factorial polynomials appearing in the Gosper-Petkovšek form, lazy evaluation of consecutive values of $y(x)$ in (4) and very simple properties of the components of the equation (4):

1. The Polynomial Normal Form (PNF) has a "local" property [10]: PNF of the product of polynomials from different shift equivalence classes is the product of PNFs of those polynomials.
2. Factorial polynomials appear only in the right-hand side of the key equation (4) and they are the only candidates for cancelation. Moreover, the number of these factorial polynomials is bounded by the degrees of the numerator and the denominator of the rational certificate $r(x)$ and does not depend on the value of the dispersion. On the other hand, each term of the form $[p(x)]_k$ contributes the value of $k$ towards the upper bound of the degree of the solution $y(x)$, which can be as large as the value of the dispersion.
3. The term $[p(x)]_k$ vanishes at any root $\alpha$ of $p(x)$ and also at $\alpha+1,\ldots,\alpha+k-1$.

4. If a solution $y(x)$ of (4) is equal to zero at any of $\alpha, \alpha+1, \ldots, \alpha+k$ (where $\alpha$ is a root of $p(x)$), then $y(x)$ is equal to zero at all these points. This means that $[p(x)]_{k+1}$ is a factor of $y(x)$ and the factorial polynomial term $[p(x)]_k$ in $P(x)$ cancels after substituting $y(x)$ into (5).
5. Any shift equivalent to $p(x)$ factor of $Q(x)$ or $R(x-1)$ from (4) provides an initial value for a solution of $y(x)$ at a root $\beta$ of this factor. If neither $Q(x)$ nor $R(x-1)$ contains a factor shift equivalent to $p(x)$, then the term $[p(x)]_k$ is present in the summand $F(x)$.
6. The evaluations required to detect equality or non-equality of $y(x)$ to zero at the consecutive points starting at $\beta$ can be done lazily using (8). The expanded form of $[p(x)]_k$ is not required for this test. Moreover, every nonzero value of $y(x)$, computed at the consecutive points $\beta, \beta+1, \ldots$ (or $\beta, \beta-1, \ldots$) during the test, is represented by a nontrivial factor in the rational certificate $r(x)$.

These properties allow us to incorporate simple and efficient changes into Gosper's decision procedure, which do not worsen the total asymptotic complexity of the procedure, but can lead to tremendous savings in the running time for summable terms with large dispersion of the rational certificate. Returning to the example above, two evaluation points ($x = -91/3$ and $x = -88/3$) are sufficient to find out that the term $\left[x + \frac{88}{3}\right]_{28}$ will cancel, and the substitution of $y(x) = \left[x + \frac{88}{3}\right]_{29} y(x)$ into the key equation gives reduced key equation:

$$4\,(x+1/2)\,(x+1/3)\,(x+91/3)\,y\,(x+1) - (x+94/3)\,(x+4/3)\,(x+1)\,y\,(x) = \left(x^3 + \frac{91}{3}x^2 + \frac{82}{9}x - \frac{194}{27}\right)$$

with the degree of the solution $< 3$. The solution $y(x) = 1/3$ of the last equation produces the desired result in reduced form.

Our prototype implementation in Maple shows practical improvements in the running time for summable essential hypergeometric terms with large dispersion of the rational certificate, compared to the standard Maple summation tools. Combining this with existing [5] and new techniques for detecting non-existence of the hypergeometric sum on early stages of computation will allow further reduction in the running time for non-summable inputs.

## References

[1] S. A. Abramov. The summation of rational functions. *Ž. Vyčisl. Mat. i Mat. Fiz.*, 11:1071–1075, 1971.

[2] S. A. Abramov and M. Petkovšek. Rational normal forms and minimal decompositions of hypergeometric terms. *Journal of Symbolic Computation*, 33(5):521–543, 2002. Computer algebra (London, ON, 2001).

[3] A. Bostan, F. Chyzak, B. Salvy, and T. Cluzeau. Low complexity algorithms for linear recurrences. In *Proceedings of the 2006 International Symposium on Symbolic and Algebraic Computation*, ISSAC '06, pages 31–38, New York, NY, USA, 2006. ACM.

[4] J. Gerhard, M. Giesbrecht, A. Storjohann, and E. V. Zima. Shiftless decomposition and polynomial-time rational summation. In *Proceedings of the 2003 International Symposium on Symbolic and Algebraic Computation*, pages 119–126 (electronic), New York, 2003. ACM.

[5] A. Gheffar. Detecting nonexistence of rational solutions of linear difference equations in early stages of computation. *ACM Comm. Computer Algebra*, 48(3/4):90–97, 2014.

[6] R. W. Gosper, Jr. Indefinite hypergeometric sums in MACSYMA. In *Proceedings of the 1977 MACSYMA Users' Conference*, pages 237–251, 1977.

[7] R. W. Gosper, Jr. Decision procedure for indefinite hypergeometric summation. *Proc. Nat. Acad. Sci. U.S.A.*, 75(1):40–42, 1978.

[8] M. Karr. Summation in finite terms. *Journal of the Association for Computing Machinery*, 28(2):305–350, 1981.

[9] R. Moenck. On computing closed forms for summations. In *Proceedings of the 1977 MACSYMA Users' Conference*, pages 225–236, 1977.

[10] R. Pirastu and V. Strehl. Rational summation and Gosper-Petkovsšek representation. *J. Symb. Comput.*, 20(5-6):617–635, Nov. 1995.

[11] E. V. Zima. Accelerating indefinite summation: Simple classes of summands. *Mathematics in Computer Science*, 7(4):455–472, 2013.

[12] E. V. Zima. Direct indefinite summation. *ACM Commun. Comput. Algebra*, 48(3/4):145–147, Feb. 2015.