November 26, 2003

A General Algorithm for the MacMahon Omega Operator

Guo-Niu HAN

Institut de Recherche Mathématique Avancée,
Université Louis Pasteur et CNRS,
7, rue René-Descartes
67084 Strasbourg Cedex, France
guoniu@math.u-strasbg.fr

AMS Subject Classification. — 05A17, 05A19, 05E05, 15A15, 68W30

Keywords. — partition analysis, partitions, Omega operator, diophantine inequalities, algorithm

Abstract. — In his famous book "Combinatory Analysis" MacMahon introduced Partition Analysis ("Omega Calculus") as a computational method for solving problems in connection with linear diophantine inequalities and equations. The technique has recently been given a new life by G. E. Andrews and his coauthors, who had the idea of marrying it with the tools of to-day's Computer Algebra.

The theory consists of evaluating a certain type of rational function of the form $A(\lambda)^{-1}B(1/\lambda)^{-1}$ by the MacMahon Ω operator. So far, the case where the two polynomials A and B are factorized as products of polynomials with two terms has been studied in details. In this paper we study the case of arbitrary polynomials A and B. We obtain an algorithm for evaluating the Ω operator using the coefficients of those polynomials without knowing their roots. Since the program efficiency is a persisting problem in several-variable polynomial Calculus, we did our best to make the algorithm as fast as possible. As an application, we derive new combinatorial identities.

Résumé. — Dans son célèbre livre "Combinatory Analysis", MacMahon a introduit l'Analyse des Partitions ("Omega Calculus") comme une méthode calculatoire pour résoudre des problèmes combinatoires liés aux systèmes linéaires des équations et des inéqualités diophantiennes. Cette technique a été récemment revitalisée par G. E. Andrews et ses coauteurs, qui ont eu l'idée de la marier avec les outils du calcul formel moderne.

Cette théorie consiste à évaluer un certain type de fonction rationnelle de forme $A(\lambda)^{-1}B(1/\lambda)^{-1}$ par l'opérateur Ω de MacMahon. Jusqu'à présent, le cas qui a été beaucoup étudié est celui où les deux polynômes A et B se factorisent explicitement en un produit de polynômes à deux termes. Dans cet article, nous étudions le cas où A et B sont deux polynômes quelconques. Nous obtenons ainsi un algorithme pour évaluer l'opérateur Ω à partir des coefficents des polynômes A et B, sans nécessairement connaître leurs racines. Puisque l'efficacité est un problème persistant dans le calcul des polynômes à plusieurs variables, nous nous sommes efforcés de rendre l'algorithme le plus rapide possible. Comme application de cette étude, nous obtenons quelques nouvelles identités combinatoires.

1. Introduction

In his famous book "Combinatory Analysis" [M] MacMahon introduced Partition Analysis ("Omega Calculus") as a computational method for solving problems in connection with linear diophantine inequalities and equations. The technique has recently been given a new life by Andrews [Pa1,Pa2], first in his study of the *lh-partitions* ("lecture-hall partitions") introduced by Bousquet-Mélou and Eriksson [BE], then, in a professional Ω operator study [Pa3–Pa9]: construction of the algorithms, implementation within Computer Algebra softwares and applications to combinatorial identity proving.

For the definition of Ω we fix two alphabets $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_r\}$ and $\Gamma = \{p, q, x, y, z, x_1, x_2, \ldots\}$. The expression

$$F := \sum_{s_1=-\infty}^{\infty} \cdots \sum_{s_r=-\infty}^{\infty} A_{s_1,\ldots,s_r} \lambda_1^{s_1} \cdots \lambda_r^{s_r}$$

is a Laurent formal series in Λ and the coefficients $A_{s_1,\ldots,s_r}$ are ordinary formal series in Γ. The MacMahon operator $\Omega_{\geq} := \Omega_{\geq}[\Lambda, \Gamma]$ is defined by

$$\underset{\geq}{\Omega} F := \sum_{s_1=0}^{\infty} \cdots \sum_{s_r=0}^{\infty} A_{s_1,\ldots,s_r}.$$

In contrast with the classical definition [Pa1-Pa9] that treats the functions analytically, we add a second alphabet Γ that enables us to work formally.

Recall that a *partition* (resp. *lh-partition*) of n is a sequence of integers $(b_1, b_2, \ldots, b_j)$ such that $n = b_1 + b_2 + \cdots + b_j$ and $0 < b_1 \leq b_2 \leq \cdots \leq b_j$ (resp. $0 \leq \frac{b_1}{1} \leq \frac{b_2}{2} \leq \cdots \leq \frac{b_j}{j}$). Bousquet-Mélou and Eriksson have proved that the number of lh-partitions $(b_1, b_2, \ldots, b_j)$ of n is equal to number of partitions of n, whos parts are odd integers at most equal to $2j - 1$ [BE].

Let us illustrate the Ω operator technique for the lh-partitions of order $j = 3$. First, the generating function for the lh-partitions is:

$$\sum_{0 \leq \frac{b_1}{1} \leq \frac{b_2}{2} \leq \frac{b_3}{3}} q^{b_1+b_2+b_3} = \underset{\geq}{\Omega} \sum_{b_1,b_2,b_3 \geq 0} q^{b_1+b_2+b_3} \lambda_1^{2b_3-3b_2} \lambda_2^{b_2-2b_1}.$$

As a formal series in q the right-hand side is equal to:

$$\underset{\geq}{\Omega} \frac{1}{(1 - \frac{q}{\lambda_2^2})(1 - \frac{q\lambda_2}{\lambda_1^3})(1 - q\lambda_1^2)}.$$

For evaluating this kind of Ω expressions MacMahon has given a list of particular cases [M, vol. 1, pp. 102–103]. Andrews, Paule and Riese have studied the general expression and derived the following recurrence relation [Pa3].

Theorem 1 ("Fundamental Recurrence"). — *Let n and m be two nonnegative integers and a an arbitrary integer. For $n \geq 2$ we have :*

$$\underset{\geq}{\Omega} \frac{\lambda^a}{(1-x_1\lambda)(1-x_2\lambda)\cdots(1-x_n\lambda)(1-\frac{y_1}{\lambda})(1-\frac{y_2}{\lambda})\cdots(1-\frac{y_m}{\lambda})}$$
$$= \frac{P_{n,m,a}(x_1,\ldots,x_n;y_1,\ldots,y_m)}{\prod_{i=1}^n(1-x_i)\cdot\prod_{i=1}^n\prod_{j=1}^m(1-x_iy_j)},$$

where

$$P_{n,m,a}(x_1,\ldots,x_n;y_1,\ldots,y_m) = \frac{1}{x_n - x_{n-1}}$$
$$\cdot \Big\{ x_n(1-x_{n-1})\cdot\prod_{j=1}^m(1-x_{n-1}y_j)\cdot P_{n-1,m,a}(x_1,\ldots,x_{n-2},x_n;y_1,\ldots,y_m)$$
$$- x_{n-1}(1-x_n)\cdot\prod_{j=1}^m(1-x_ny_j)\cdot P_{n-1,m,a}(x_1,\ldots,x_{n-2},x_{n-1};y_1,\ldots,y_m)\Big\}.$$

This fundamental recurrence has allowed Riese to develop a Computer Algebra package for `Mathematica`, called `Omega1` [Om]. In the case where the denominator involves factors of the form $(1-x\lambda^r)$, those factors must be decomposed as

$$(1-x\lambda^r) = \prod_{j=0}^{r-1}(1-e^{2\pi ij/r}x^{1/r}\lambda)$$

before applying Theorem 1. As mentioned in [Pa3], this method is very costly algorithmically, because the final result is to be simplified for eliminating the irrational coefficients $e^{2\pi i/r}$. To solve the problem the same authors, in a subsequent paper [Pa6], have studied the following expression

$$\underset{\geq}{\Omega} \frac{\lambda^a}{(1-x_1\lambda^{j_1})\cdots(1-x_n\lambda^{j_n})(1-\frac{y_1}{\lambda^{k_1}})\cdots(1-\frac{y_m}{\lambda^{k_m}})}$$

and constracted an algorithm based on a generalized recurrence. This gave rise to the `Omega2` package [Om].

When reading Theorem 1 above we were struck by two observations: (a) the Ω expression is symmetric in $X=\{x_1,\ldots,x_n\}$ and $Y=\{y_1,\ldots,y_m\}$; (b) the polynomial $P_{n,m,a}$ is calculated by a relation of "divided-difference" type [L1]. The natural question arises:

Problem 1. — *How can we express the polynomials $P_{n,m,a}(X,Y)$ by means of the elementary symmetric functions $e_r(X)$ and $e_r(Y)$?*

As indicated by Lascoux, such a problem is relevant both in the theory of symmetric functions and in the domain of algorithmics. A significant example can be found in the papers [L2, H, HK]. The difficulties met by `Omega1` will be overcome in our approach, as we construct an algorithm that is more general than `Omega2`. This is the main object of this paper. More precisely, our purpose is to solve the following problem:

PROBLEM 2. — *Let*

$$A(t) = 1 + a_1 t + a_2 t^2 + \cdots + a_n t^n = \prod_{i=1}^{n} (1 - x_i t),$$
$$B(t) = 1 + b_1 t + b_2 t^2 + \cdots + b_m t^m = \prod_{i=1}^{m} (1 - y_i t)$$

be two polynomials and

$U(t)$ be a Laurent polynomial

with $x_i \neq 1$ et $x_i y_j \neq 1$ for all i, j. Calculate the following expression Ω:

$$\underset{\geq}{\Omega} \frac{U(\lambda)}{A(\lambda) B(1/\lambda)}$$

by using only the expansions $1 + a_1 t + a_2 t^2 + \cdots + a_n t^n$ and $1 + b_1 t + b_2 t^2 + \cdots + b_m t^m$ without having to determine and use the roots x_i and y_i.

Theorem 2 below can be regarded as the first step in our solution of Problem 2. Notice that the expression found for Ω is somehow more explicit than the form given in Theorem 1. Throughout the paper and in particular in the statement of Theorem 2 the notations of Problem 2 have been kept.

THEOREM 2. — *If* $\deg(U) \leq n - 1$, *then:*

$$\underset{\geq}{\Omega} \frac{U(\lambda)}{A(\lambda) B(1/\lambda)} = \sum_{i=1}^{n} \frac{x_i^{n-1} U(1/x_i)}{(1 - x_i) B(x_i) \prod_{j \neq i} (x_i - x_j)}.$$

The formula holds even when the x_j's are not all distinct.

In the above summation the roots y_j do not appear, but the roots x_j are still present, so that half of our goal has been fulfilled. To eliminate the x_j's in the sum we make use of *two methods*. The first one, that was indicated by Habsieger [Ha], is to use the Lagrange interpolation formula (section 2). The algorithm is elegant and simple, but becomes inefficient, for instance when the degree of the polynomial $A(t)$ is large. The algorithm involves the Euclidean division of two polynomials, that is known to have a complexity of high order when the coefficients of the two polynomials are polynomials in several variables.

The second method is realized in two steps: first, re-express the i-th summand in Theorem 2 in a form that depends only on x_i (section 3); then, symmetrize the new summands as rational functions in one variable (section 4). This method involves the resultant of two polynomials. As the resultant is a multiplicative operator, this method is much faster than the first one when $A(t)$ is not an irreducible polynomial.

In section 5 we give a global description of our algorithm, that keeps the advantages of the two methods. It has been implemented as a Maple package `GenOmage` that is freely available on the web[1]. Concerning the efficiency we do not claim that `GenOmega` is faster than `Omega2`, because we did not effectuate a complete testing for a large number of examples in various environments. However, in our preliminary test `GenOmega` is often faster than `Omega2` by a factor of 3 to 6. As an application of our study we derive two new combinatorial identities.

2. Proof of Theorem 2

Let $f(t)$ be polynomial in t with constant term equal to 1. For each integer i (positive or negative) the complete homogeneous symmetric function $h_i(f)$ is defined by

$$\frac{1}{f(t)} = \sum_i h_i(f)t^i.$$

Let $a \le n-1$; with the notation of Problem 2 we have

$$\begin{aligned}\underset{\geq}{\Omega}\, \frac{\lambda^a}{A(\lambda)B(1/\lambda)} &= \underset{\geq}{\Omega} \sum_{k,j} h_k(A)h_j(B)\lambda^{a+k-j} \\ &= \sum_{k,j,a+k-j\geq 0} h_k(A)h_j(B) \\ &= \sum_k \Big[h_k(A) \sum_{j\leq a+k} h_j(B)\Big].\end{aligned}$$

But with $C(t) := (1-t)B(t)$,

$$\frac{1}{C(t)} = \frac{1}{B(t)(1-t)} = \sum_i h_i(B)t^i \sum_{j\geq 0} t^j = \sum_{\ell\geq 0}\Big[\sum_{i\leq \ell} h_i(B)\Big]t^\ell,$$

i.e., $h_\ell(C) = \sum_{i=0}^{\ell} h_i(B)$. We then have

$$\underset{\geq}{\Omega}\, \frac{\lambda^a}{A(\lambda)B(1/\lambda)} = \sum_k h_k(A)h_{a+k}(C).$$

[1] `http://www-irma.u-strasbg.fr/~guoniu/software`

Let $X = \{x_1, x_2, \ldots, x_n\}$. Recall that for every $k \geq -(n-1)$ (see for examples [Ma,LS])

$$h_k(A) = \begin{vmatrix} x_1^{k+n-1} & \cdots & x_n^{k+n-1} \\ x_1^{n-2} & \cdots & x_n^{n-2} \\ \cdots & \cdots & \cdots \\ x_1^1 & \cdots & x_n^1 \\ x_1^0 & \cdots & x_n^0 \end{vmatrix} / \Delta(X),$$

where $\Delta(X)$ is the Vandermonde determinant. When multiplying the first row by $h_{a+k}(C)$, we get:

$$\Delta(X) \cdot \underset{\geq}{\Omega} \frac{\lambda^a}{A(\lambda)B(1/\lambda)} = \sum_{k \geq -(n-1)} \begin{vmatrix} x_1^{k+n-1} h_{a+k}(C) & \cdots \\ x_1^{n-2} & \cdots \\ \cdots & \cdots \\ x_1^0 & \cdots \end{vmatrix}$$

$$= \begin{vmatrix} x_1^{n-1-a} \sum_{k \geq -(n-1)} x_1^{a+k} h_{a+k}(C) & \cdots \\ x_1^{n-2} & \cdots \\ \cdots & \cdots \\ x_1^0 & \cdots \end{vmatrix} = \begin{vmatrix} \dfrac{x_1^{n-1-a}}{C(x_1)} & \cdots \\ x_1^{n-2} & \cdots \\ \cdots & \cdots \\ x_1^0 & \cdots \end{vmatrix}.$$

Notice that the two conditions $a \leq n-1$ and $k \geq -(n-1)$ imply that $a + k \leq 0$, a condition that is needed in the last step. Now by expanding the determinant according to the first row we get:

$$\underset{\geq}{\Omega} \frac{\lambda^a}{A(\lambda)B(1/\lambda)} = \frac{1}{\Delta(X)} \sum_{i=1}^{n} \frac{x_i^{n-1-a}}{C(x_i)} (-1)^{i+1} \Delta(X \setminus x_i)$$
$$= \sum_{i=1}^{n} \frac{x_i^{n-1}/x_i^a}{(1-x_i)B(x_i)\prod_{j \neq i}(x_i - x_j)}.$$

In other words, Theorem 2 is proved in the case $U(t) = t^a$ and $a \leq n-1$. The linear combination of all those particular cases provides a complete proof of Theorem 2.

Habsieger [Ha] has given another proof of Theore 2 by using the partial fraction decomposition of a rational function. Moreover, he explained to me that further calculations could be made by means of the Lagrange interpolation formula.

Theorem 3. — *If the Laurent polynomial U has degree* $\deg(U) \leq n-1$ *and low degree* $\operatorname{ldeg}(U) \leq m-1$, *then there exists a unique polynomial $D(t)$ of degree smaller than than n such that*

$$D(t) \equiv \frac{U(t)}{B(1/t)} \bmod A(t).$$

Moreover,

$$\underset{\geq}{\Omega}\, \frac{U(\lambda)}{A(\lambda)B(1/\lambda)} = \frac{D(1)}{A(1)}.$$

Proof. — As $\gcd(t^m B(1/t), A(t)) = 1$, Bézout's Theorem implies that there exists a polynomial $C(t)$ such that $C(t)\cdot t^m B(1/t) + K(t)\cdot A(t) = 1$, i.e., $C(t) \equiv \frac{1}{t^m B(1/t)} \bmod A(t)$. Then the polynomial $D(t)$ is the remainder of the division of $t^m U(t)C(t)$ by $A(t)$. For the second part it suffices to verify that

$$\sum_{i=1}^{n} \frac{x_i^{n-1}U(1/x_i)}{B(x_i)} \frac{\prod_{j\neq i}(1-x_j t)}{\prod_{j\neq i}(x_i - x_j)} = D(t).$$

This follows from the Lagrange interpolation formula. ∎

3. Evaluation of the summands of Theorem 2

The purpose of this section is to eliminate all the x_j ($j \neq i$) and keep only x_i in the i-th summand of Theorem 2. For an arbitrary polynomial $A(t)$ we consider the partial fraction decomposition

$$\frac{1}{A(t)} = \frac{U_1(t)}{A_1(t)} + \cdots + \frac{U_\ell(t)}{A_\ell(t)}.$$

As all the $A_i(t)$'s are powers of square-free polynomials, it suffices to study the case where $A(t)$ itself is a power of a square-free polynomial. First, consider the case where the polynomial $A(t)$ is square-free, i.e., all the x_j ($1 \leq j \leq n$) are distinct.

Lemma 4. — *If* $\deg(U) \leq n-1$ *and if* $x_i \neq x_j$ *for all* $i \neq j$, *then*

$$\underset{\geq}{\Omega}\, \frac{U(\lambda)}{A(\lambda)B(1/\lambda)} = \sum_{i=1}^{n} \frac{x_i^{n-1}U(1/x_i)}{(x_i - 1)B(x_i)\bar{A}(1/x_i)x_i^{n-1}},$$

where $\bar{A}(t) = t\frac{\partial}{\partial t}A(t)$. *Moreover, if* $\operatorname{ldeg}(U) \leq m-1$, *then the numerator* $N(x_i)$ *and the denominator* $D(x_i)$ *of the above summand are two polynomials in* x_i *such that* $\deg(N(x_i)) < \deg(D(x_i))$.

Proof. — By Theorem 2

$$\prod_{j\neq i}(x_i - x_j) = \frac{\prod_{j=1}^{n}(t - x_j)}{t - x_i}\Big|_{t\to x_i} = \frac{t^n A(1/t)}{t - x_i}\Big|_{t\to x_i}$$

$$= \frac{\frac{\partial}{\partial t} t^n A(1/t)}{\frac{\partial}{\partial t}(t - x_i)}\Big|_{t\to x_i} = -x_i^{n-1}\bar{A}(1/x_i).$$

If $\operatorname{ldeg}(U) \leq m-1$, then $\deg(N(x_i)) = n - 1 + \operatorname{ldeg}(U) \leq n + m - 2$ and $\deg(D(x_i)) = m + n$. ∎

Next consider the case of the power of a square-free polynomial.

Theorem 5. — *If* $\deg(U) \leq n-1$ *and if* $A(t) = W(t)^k$ *with* $W(t) = (1-x_1t)(1-x_2t)\cdots(1-x_st)$ *such that* $x_i \neq x_j$ *for* $1 \leq i < j \leq s$*, then :*

$$\underset{\geq}{\Omega} \frac{U(\lambda)}{A(\lambda)B(1/\lambda)} = \sum_{i=1}^{s} \frac{1}{(k-1)!} \Big[\Big(\frac{\partial}{\partial t}\Big)^{k-1} \frac{t^{n-1}U(1/t)}{(1-t)B(t)t^n} \Big(\frac{t-x_i}{W(1/t)}\Big)^k \Big] \Big|_{t\to x_i}.$$

Moreover, if $\operatorname{ldeg}(U) \leq m-1$*, then the* i*-th summand is a rational fraction, whose numerator* $N(x_i)$ *and denominator* $D(x_i)$ *are two polynomials in* x_i *such that* $\deg(N(x_i)) < \deg(D(x_i))$*.*

The proof of Theorem 4 is based on the following Lemma.

Lemma 6. — *Let* $f(t)$ *be a function and* $X = \{x_1, x_2, \ldots, x_k\}$ *be an alphabet. Then*

$$\sum_{i=1}^{k} \frac{f(x_i)}{\prod_{j\neq i}(x_i - x_j)}\Big|_{x_1,x_2,\ldots,x_k\to t} = \frac{1}{(k-1)!}\Big(\frac{\partial}{\partial t}\Big)^{k-1} f(t).$$

Proof. — By linearity it suffices to take $f(t) = t^\alpha$, so that

$$\sum_{i=1}^{k} \frac{x_i^\alpha}{\prod_{j\neq i}(x_i - x_j)} = \frac{1}{\Delta(X)} \begin{vmatrix} x_1^\alpha & \cdots & x_k^\alpha \\ x_1^{k-2} & \cdots & x_k^{k-2} \\ \cdots & \cdots & \cdots \\ x_1^2 & \cdots & x_k^2 \\ x_1 & \cdots & x_k \\ 1 & \cdots & 1 \end{vmatrix} = h_{\alpha-k+1}(X).$$

Therefore,

$$\sum_{i=1}^{k} \frac{f(x_i)}{\prod_{j\neq i}(x_i - x_j)}\Big|_{x_1,x_2,\ldots,x_k\to t} = h_{\alpha-k+1}(t,t,\ldots,t)$$

is the coefficient of $u^{\alpha-k+1}$ in $1/(1-ut)^k$. $\square$

Proof of the Theorem 5. — Let $x_1 = x_{11} = x_{12} = x_{1k}$, $x_2 = x_{21} = x_{22} = x_{2k}$, $\cdots$, $x_s = x_{s1} = x_{s2} = x_{sk}$. Also let $F(t) := \frac{t^{n-1}U(1/t)}{(1-t)B(t)}$. It follows from Theorem 2 that

$$\begin{aligned} \underset{\geq}{\Omega} \frac{U(\lambda)}{A(\lambda)B(1/\lambda)} &= \sum_{i=1}^{s}\sum_{j=1}^{k} \frac{F(x_{ij})}{\prod_{(i',j')\neq(i,j)}(x_{ij} - x_{i'j'})} \\ &= \sum_{i=1}^{s}\sum_{j=1}^{k} \frac{F(t)}{\prod_{i'\neq i, j'=1,\ldots,k}(t - x_{i'j'}) \prod_{j'\neq j}(t - x_{ij'})}\Big|_{t\to x_{ij}} \\ &= \sum_{i=1}^{s}\sum_{j=1}^{k} \frac{F(t)}{\prod_{i'\neq i}(t - x_{i'})^k \prod_{j'\neq j}(x_{ij} - x_{ij'})}\Big|_{t\to x_i}. \end{aligned}$$

From the previous Lemma we have

$$\begin{aligned}
&\sum_{j=1}^{k} \frac{F(t)}{\prod_{i'\neq i}(t-x_{i'})^k \prod_{j'\neq j}(x_{ij}-x_{ij'})}\Big|_{x_{ij}\to t}\Big|_{t\to x_i}\\
=&\frac{1}{(k-1)!}\Big(\frac{\partial}{\partial t}\Big)^{k-1}\frac{F(t)}{\prod_{i'\neq i}(t-x_{i'})^k}\Big|_{t\to x_i}\\
=&\frac{1}{(k-1)!}\Big(\frac{\partial}{\partial t}\Big)^{k-1}\frac{F(t)(t-x_i)^k}{\prod_{i'}(t-x_{i'})^k}\Big|_{t\to x_i}\\
=&\frac{1}{(k-1)!}\Big[\Big(\frac{\partial}{\partial t}\Big)^{k-1}\frac{F(t)}{t^n}\Big(\frac{t-x_i}{W(1/t)}\Big)^k\Big]\Big|_{t\to x_i}.
\end{aligned}$$

Now if $\mathrm{ldeg}(U)\leq m-1$, then $N_I := t^{n-1}U(1/t)(t-x_i)^k$ is a polynomial in t whose degree is smaller than $n+m-2+k$ and $D_I := (1-t)B(t)t^nW(1/t)^k$ is a polynomial in t of degree $n+m+1$. Then we can write

$$\Big(\frac{\partial}{\partial t}\Big)^{k-1}\frac{N_I}{D_I}=\frac{N_F}{D_F}.$$

When the operator $\frac{\partial}{\partial t}$ acts on a rational fraction, it increases the difference between the degree of the denominator and the degree of the numerator. Thus

$$\deg(D_F)-\deg(N_F)\geq(n+m+1)-(n+m-2+k)+(k-1)=2.$$

The substitution of t by x_i cannot be immediately made either on N_I/D_I or on N_F/D_F, because $W(1/x_i)=0$. Notice that x_i is a zero of both N_F and D_F of the same degree $2k-1$. Consequently, we have

$$\Big(\frac{\partial}{\partial t}\Big)^{k-1}\frac{N_I}{D_I}\Big|_{t\to x_i}=\frac{(\frac{\partial}{\partial t})^{2k-1}N_F\Big|_{t\to x_i}}{(\frac{\partial}{\partial t})^{2k-1}D_F\Big|_{t\to x_i}}.$$

The substitution $t\to x_i$ on $(\frac{\partial}{\partial t})^{2k-1}N_F$ does not increase the degree in x_i, because all the x_i in this expression occur in factors of the form $t-x_i$. Finally,

$$\deg\Big((\frac{\partial}{\partial t})^{2k-1}D_F\Big|_{t\to x_i}\Big)-\deg\Big((\frac{\partial}{\partial t})^{2k-1}N_F\Big|_{t\to x_i}\Big)\geq 2. \quad\square$$

4. Symmetrize a rational function

Thanks to the results obtained in section 3 the sum in the identity of Theorem 2 is of the form $\sum_{i=1}^{n} f(x_i)$, where each summand is a rational function that depends on a single variable. The next step is to calculate that sum by using the expansion $A(t) = 1+a_1t+\cdots+a_nt^n$ without having to determine and use the roots x_i. This will be done in Proposition 8.

LEMMA 7. — *Let $K(t) = k_0 + k_1t + k_2t^2 + \cdots + k_mt^m + \cdots + k_\ell t^\ell$ with $0 \le m \le \ell - 1$ such that $K(x_i) \neq 0$ for each $1 \le i \le n$. Then*

$$\sum_{i=1}^{n} \frac{x_i^m}{K(x_i)} = \frac{\frac{\partial}{\partial u}R}{R}\Big|_{u\to k_m},$$

where R is the resultant of the two polynomials $K(t) + (u - k_m)t^m$ and $t^nA(1/t)$.

Proof. — Let $\hat{K}(t) := K(t) + (u - k_m)t^m$. Because

$$R = \hat{K}(x_1)\hat{K}(x_2)\cdots\hat{K}(x_n),$$

we have

$$\frac{\partial}{\partial u}\log R = \frac{\frac{\partial}{\partial u}R}{R} = \sum_{i=1}^{n}\frac{\frac{\partial}{\partial u}\hat{K}(x_i)}{\hat{K}(x_i)} = \sum_{i=1}^{n}\frac{x_i^m}{\hat{K}(x_i)}.$$

Thus

$$\frac{\frac{\partial}{\partial u}R}{R}\Big|_{u\to k_m} = \sum_{i=1}^{n}\frac{x_i^m}{\hat{K}(x_i)}\Big|_{u\to k_m} = \sum_{i=1}^{n}\frac{x_i^m}{K(x_i)}. \quad \square$$

In the case where the numerator is not a monomial we can apply Lemma 7 several times by linearity. But to avoid the calculation of the resultant several times, we prefer to use the following Proposition, that can be proved in the same way as the Lemma.

PROPOSITION 8. — *Let $K(t) = k_0 + k_1t + k_2t^2 + \cdots + k_\ell t^\ell$ with $K(x_i) \neq 0$ for each $1 \le i \le n$. If $S(t) = s_{\alpha_1}t^{\alpha_1} + s_{\alpha_2}t^{\alpha_2} + \cdots + s_{\alpha_m}t^{\alpha_m}$ with $0 \le \alpha_1 < \alpha_2 < \cdots < \alpha_m \le \ell - 1$, then*

$$\sum_{i=1}^{n}\frac{S(x_i)}{K(x_i)} = \frac{(s_{\alpha_1}\frac{\partial}{\partial u_1} + \cdots + s_{\alpha_m}\frac{\partial}{\partial u_m})R}{R}\Big|_{u_1\to k_{\alpha_1},\cdots,u_m\to k_{\alpha_m}},$$

where R is the resultant of the two polynomials $K(t) + (u_1 - k_{\alpha_1})t^{\alpha_1} + (u_2 - k_{\alpha_2})t^{\alpha_2} + \cdots + (u_m - k_{\alpha_m})t^{\alpha_m}$ and $t^nA(1/t)$.

Remark. — Jouanolou [J] has shown me an elegant formula for symmetrizing a rational function in one variable:

$$\sum_{i=1}^{n} f(x_i) = \mathrm{tr} f(M),$$

where M is the companion matrix of the polynomial $t^n A(1/t)$. However, the formula, although very simple, did not enable me to build a faster algorithm, because it needs find the Bézout coefficients of two polynomials and the evaluation of the polynomial value of a matrix.

5. The Algorithm

We describe our algorithm for solving Problem 2 in the following five steps.

(A1) First, we can suppose that $A(t) \neq 1$ and $B(t) \neq 1$. It is easy to evaluate the Ω expression when $A(t) = 1$ or $B(t) = 1$ (see for example [Pa3, Lemmas 2.1–2.2]).

(A2) Then, we can suppose that $A(t)$ is of form $W(t)^k$ where $W(t)$ is an irreducible polynomial. If it is not, we use the partial fraction decomposition

$$\frac{1}{A(t)} = \frac{U_1(t)}{A_1(t)} + \cdots + \frac{U_\ell(t)}{A_\ell(t)}$$

and the linearity of the Ω operator.

(A3) We can also suppose that the numerator $U(t)$ satisfies $\deg(U) < n$ and $\mathrm{ldeg}(U) < m$. If not, we decompose the Laurent polynomial $U(t) = U^+(t) + U^-(t)$ where $U^+(t)$ (resp. $U^-(t)$) is the "positive part" (resp. "negative part") of $U(t)$. Notice that $U^+(t)$ and $U^-(1/t)$ are two polynomials. We can find four polynomials Q_1, Q_2, R_1, R_2 satisfying

$$\begin{aligned} U^+(t) &= Q_1(t)A(t) + R_1(t), \quad \deg(R_1) < n \\ U^-(1/t) &= Q_2(t)B(t) + R_2(t), \quad \deg(R2) < m. \end{aligned}$$

Let $R(t) = R_1(1) + R_2(1/t)$ be a new Laurent polynomial with $\deg(R) < n$ and $\mathrm{ldeg}(R) < m$. We have

$$U(t) = Q_1(t)A(t) + Q_2(1/t)B(1/t) + R(t)$$

and

$$\underset{\geq}{\Omega} \frac{U(t)}{A(t)B(1/t)} = \underset{\geq}{\Omega} \frac{Q_1(t)}{B(1/t)} + \underset{\geq}{\Omega} \frac{Q_2(1/t)}{A(t)} + \underset{\geq}{\Omega} \frac{R(t)}{A(t)B(1/t)}.$$

Because the first two terms are easy to evaluate (see step (A1)), it suffices to evaluate the third term, which satisfies $\deg(R) < n$ et $\mathrm{ldeg}(R) < m$.

(A4) If $A(t)$ is an irreducible polynomial, i.e., $k = 1$ in step (A2), we use Theorem 3. Let the polynomial $C(t)$ be the Bézout coefficient in $C(t) \cdot t^m B(1/t) + K(t) \cdot A(t) = 1$, and the polynomial $D(t)$ be the rest of the division of $t^m U(t)C(t)$ by $A(t)$. Then

$$\underset{\geq}{\Omega} \frac{U(t)}{A(t)B(1/t)} = \frac{D(1)}{A(1)}.$$

(A5) Now we consider the case $k \geq 2$ in step (A2). From Theorem 5 we have

$$\underset{\geq}{\Omega} \frac{U(t)}{A(t)B(1/t)} = \sum_{i=1}^{n} \frac{S(x_i)}{K(x_i)},$$

where $S(t)$ and $K(t)$ are two polynomials satisfying $\deg(S) < \deg(K)$. Then we calculate the latter sum by Proposition 8.

6. Applications

Because our algorithm is more general than the algorithm used in `Omega` package [Pa3,Pa6], the examples shown in those papers are also appropriate examples for the `GenOmega` package. In this section we study an example that illustrates the difference between `Omega` and `GenOmega`. Consider the following Ω expression

$$\underset{\geq}{\Omega} \frac{1}{(1 + x\lambda + y\lambda^5)(1 + a/\lambda)}.$$

Putting this expression in the `Omega` package returns the error message:

$1/((1 + x\lambda + y\lambda^5)(1 + a/\lambda))$ `is not a valid input.`

Putting it in the `GenOmega` package gives the correct result:

$$\frac{1 + ya - ya^2 + ya^3 - ya^4}{(1 + x + y)(1 - xa - ya^5)}.$$

In the same manner the Ω expression

$$\underset{\geq}{\Omega} \frac{1}{(1 - x\lambda - x\lambda^2)(1 - y/\lambda - y/\lambda^2)}$$

provides interesting combinatorial identities, as shown in the next proposition.

Proposition 9. — *For each integer $n \geq 1$ we have*

$$\sum_{\substack{i,j,k \\ 4k+j-i\geq 2n \\ n\geq k\geq 0}} \binom{k}{i}\binom{n-k}{j} = \frac{(3n+4)2^n + 2(-1)^n}{6},$$

$$\sum_{\substack{i,j,k \\ 4k+j-i\geq 4n \\ 2n\geq k\geq 0}} (-1)^k \binom{k}{i}\binom{2n-k}{j} = 2^{2n-1} - (-3)^{n-1}.$$

The initial values of the above sums, denoted by h_n and g_n, respectively, are:

n	:	1	2	3	4	5	6	7	8	$\cdots$
h_n	:	2	7	17	43	101	235	533	1195	$\cdots$
g_n	:	1	11	23	155	431	2291	7463	34955	$\cdots$

Proof. — More generally we calculate:

$$\begin{aligned}
F(x,y) &:= \sum_{k,m,i,j\geq 0, 2m+j\geq 2k+i} \binom{m}{i}\binom{k}{j} x^m y^k \\
&= \underset{\geq}{\Omega} \sum_{m,k,i,j\geq 0} \binom{m}{i}\binom{k}{j} x^m y^k \lambda^{2m-i-2k+j} \\
&= \underset{\geq}{\Omega} \sum_{m,k\geq 0} (x\lambda + x\lambda^2)^m (y/\lambda + y/\lambda^2)^k \\
&= \underset{\geq}{\Omega} \frac{1}{(1-x\lambda - x\lambda^2)(1-y/\lambda - y/\lambda^2)} \\
&= \frac{1+x^2y}{(1-2x)(1-3xy-x^2y-xy^2)}
\end{aligned}$$

Specializing $y = x$ or $y = -x$ yields:

$$\begin{aligned}
F(x,x) &= \frac{1/3}{1+x} + \frac{1/2}{(1-2x)^2} + \frac{1/6}{1-2x}, \\
F(x,-x) &= \frac{1}{6} + \frac{1/2}{1-2x} + \frac{1/3+x}{1+3x^2}.
\end{aligned}$$

By comparing the coefficient of x^n we obtain Proposition 9. $\square$

Notice that the polynomial $1 - 3xy - x^2y - xy^2$ with two variables can not be factorized, but in the case of $y = x$ or $y = -x$ it has very simple factors.

Acknowledgements. — I wish to thank P. Paule, L. Habsieger, J.-P. Jouanolou, A. Lascoux and D. Foata for many useful suggestions.

References

[BE] M. Bousquet-Mélou, K. Eriksson. — Lecture hall partitions, *The Ramanujan Journal*, **1** (1997), pp. 101–111.

[H] G.-N. Han. — Généralisation de l'identité de Scott sur les permanents, *Linear Alg. Appl.*, **311** (2000), pp. 25–34.

[Ha] L. Habsieger. — Private commucation, 2002.

[HK] G.-N. Han, Ch. Krattenthaler. — Rectangular Scott-type permanents, *43e Séminaire Lotharingien de Combinatoire*, [B43g], 2000, 25 pages.

[J] J.P. Jouanolou. — Private commucation, 2002.

[L1] A. Lascoux. — *Notes on Interpolation in one and several variables*, 40 pages. Available via the web `http://igm.univ-mlv.fr/~al`.

[L2] A. Lascoux. — Square-ice Enumeration, in "The Andrews Festschrift. Seventeen Papers on Classical Number Theory and Combinatorics"(D. Foata and G.N. Han eds.), *Springer*, 2001, pp. 335–348.

[LS] A. Lascoux, M.-P. Schützenberger. — *Formulaire raisonné de fonctions symétriques*. — Publication LITP, Université Paris 7, 1985.

[M] P.A. MacMahon. — *Combinatory Analysis*, 2 vols., Cambridge University Press, Cambridge, 1915–1916, (Reprinted: Chelsea, New York, 1960).

[Ma] I.G. Macdonald. — *Symmetric functions and Hall polynomials,* second edition. Clarendon Press, Oxford, 1995.

[Om] G.E. Andrews, P. Paule, A. Riese. — Omega package for Mathematica. Available via the web:
`http://www.risc.uni-linz.ac.at/research/combinat/risc/software`.

[Pa1] G.E. Andrews. — MacMahon's partition analysis I: The lecture hall partition theorem, *Progr. Math.*, **161** (1998), pp. 1-22.

[Pa2] G.E. Andrews. — MacMahon's partition analysis II: Fundamental theorems, *Ann. Comb.*, **4** (2000), pp. 327-338.

[Pa3] G.E. Andrews, P. Paule, A. Riese. — MacMahon's partition analysis III: The Omega Package, *European J. Combin.*, **22** (2001), pp. 887-904.

[Pa4] G.E. Andrews, P. Paule. — MacMahon's partition analysis IV: Hypergeometric Multisums, in "The Andrews Festschrift. Seventeen Papers on Classical Number Theory and Combinatorics"(D. Foata and G.N. Han eds.), *Springer*, 2001, pp. 189–208.

[Pa5] G.E. Andrews, P. Paule, A. Riese, V. Strehl. — MacMahon's partition analysis V: Bijections, recursions, and magic squares, in "Algebraic Combinatorics and Applications" (A. Betten and al., eds.), *Lecture Notes in Math.*, pp. 1-39, Springer, 2001.

[Pa6] G.E. Andrews, P. Paule, A. Riese. — MacMahon's partition analysis VI: A New Reduction Algorithm, *Ann. Comb.*, **5** (2001), pp. 251-270.

[Pa7] G.E. Andrews, P. Paule, A. Riese. — MacMahon's partition analysis VII: Constrained Compositions, in "q-Series with Applications to Combinatorics, Number Theory, and Physics" (B.C. Berndt and K. Ono, eds.), Contemp. Math., Vol. **291**, pp. 11-27, *Amer. Math. Soc.*, 2001.

[Pa8] G.E. Andrews, P. Paule, A. Riese. — MacMahon's partition analysis VIII: Plane Partition Diamonds, *Adv. in Appl. Math.*, **27** (2001), pp. 231-242.

[Pa9] G.E. Andrews, P. Paule, A. Riese. — MacMahon's partition analysis IX: k-Gon Partitions, *Bull. Austral. Math. Soc.*, **64** (2001), pp. 321-329.