UNIVERSITAS SCIENTIARUM
Revista de la Facultad de Ciencias
PONTIFICIA UNIVERSIDAD JAVERIANA
Vol. 10, No. 1, 15-24

ALGUNAS PROPIEDADES COMBINATORIAS DE LOS POLINOMIOS SIMÉTRICOS Y CÓMPUTO DE POLINOMIOS DE SCHUR OBLICUOS

Fernando Novoa

Departamento de Matematicas, Facultad de Ciencias,
Pontificia Universidad Javeriana,
Cra. 7, No. 43-82, Bogotá, D.C.

fernando.novoa@javeriana.edu.co

RESUMEN

Presentamos algunas rutinas de un programa para el sistema computacional CoCoA, las cuales realizan el cómputo de familias de polinomios de simétricos, entre ellas los polinomios de Schur. Con estas rutinas se pueden comprobar propiedades combinatorias que relacionan tablas de Young y polinomios skew de Schur.

Palabras claves: Polinomios simétricos, polinomios de Schur, polinomios skew de Schur, tablas de Young.

ABSTRACT

We developed some computational routines for the algebraic computational system CoCoA, in order to compute families of symmetric polynomials; in particular, the Schur polynomials. With these routines we can show some properties of the combinations between Young Tableaux and skew Schur polynomials.

Key words: Symmetric polynomials, Schur polynomials skew Schur polynomials, Young tableaux.

INTRODUCCIÓN

Los polinomios de Schur forman una base para el anillo de polinomios. Por lo tanto, los polinomios skew de Schur son combinaciones lineales de los polinomios de Schur y sus coeficientes son conocidos como los coeficientes de Littlewood-Richardson, los cuales permiten contar tablas de Young con algunas propiedades. Son precisamente estas propiedades combinatorias las que queremos poner de manifiesto con el uso de las rutinas de nuestros programas.

En particular, deseamos ver que las ecuaciones

$$S_{\lambda/\mu}(x_1, ..., x_n) = \sum_{\nu} c^{\lambda}_{\mu,\nu} S_{\nu} \tag{1.1}$$

$$S_{\lambda/\mu}(x_1, ..., x_n) = \sum_{T} x^{T} \tag{1.2}$$

se satisfacen y hallar explícitamente el valor de los coeficientes $c^{\lambda}_{\mu,\nu}$ y su relación con el número de las tablas de Young de la forma λ/μ

Inicialmente recordaremos las definiciones y propiedades básicas de estos polinomios, para luego presentar los programas.

DEFINICIONES

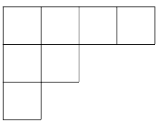
Seanu n entero positivo y $\lambda = (\lambda_1, ..., \lambda_k)$ una partición de n. Entonces se tiene que

$$\sum_{i=1} \lambda_i = n$$

$$\lambda_i \geq ... \geq \lambda_k > 0$$

Cada partición se identifica con un diagrama de Ferre, el cual es un arreglo de cajas alineadas a la izquierda, en donde la fila i contiene λ_i cajas. Si λ es partición n de escribiremos $|\lambda| = n$.

Ejemplo 1. El diagrama de Ferre para la partición es de la forma



Si enumeramos las cajas del diagrama de Ferre de la partición λ y $|\lambda| = n$, con números del conjunto $\{1,...,n\}$de tal forma que las filas y columnas forman sucesiones estrictamente crecientes al leerlas de izquierda a derecha y de arriba abajo respectivamente, entonces llamamos a esta enumeración una tabla estándar. Si las filas son no decrecientes (es decir, permitimos repeticiones por filas), pero las columnas son estrictamente crecientes, decimos que la tabla es semiestándar.

Ejemplo 2. Para la partición (4,2,1) las siguientes tablas son ejemplos de tabla estándar y semiestándar respectivamente.

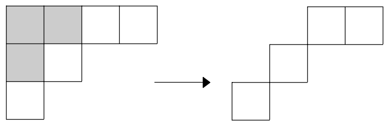
1	3	6	7
2	5		
4			

2	2	3	4
4	5		
6			

Si λ, μ son particiones, decimos que $\mu \leq \lambda$ si para todo i se tiene que $\mu_i \leq \lambda_i$.

Ejemplo 3. Si $\mu = (2,1)$ y $\lambda = (4,2,1)$ entonces $\mu \leq \lambda$.

Si μ, λ son particiones y $\mu \leq \lambda$, entonces podemos definimos el *diagrama* oblicuo λ/μ como el diagrama resultante de suprimir las cajas de μ del diagrama de λ. Así si $\mu = (2,1)$ y $\lambda = (4,2,1)$ entonces el diagrama de λ/μ es de la forma



Una tabla skew será entonces una enumeración de las cajas del diagrama skew y será estándar o semiestándar si conserva las relaciones para filas y columnas dadas anteriormente.

Los polinomios de Schur fueron inicialmente dados por Jacobi y reciben este nombre puesto que fue Schur quien los redescubrió en sus trabajos sobre representaciones polinomiales y caracteres de los grupos lineales $GL(n,C)$ sobre los complejos. Estos polinomios son simétricos y existen diversas formas de definirlos.

Sea $Z[x_1, ..., x_n]$ el anillo de polinomios con coeficientes enteros en las variables $x_1, ..., x_n$, y sea $\alpha = (\alpha_1, ..., \alpha_n)$ una secuencia de enteros. Definimos

$$g_\alpha = \det\left[x_i^{\alpha_j}\right]_{1\leq i,j\leq n}$$

y el *polinomio de Schur* asociado a la partición λ como

$$s_\lambda = \frac{g_{\lambda+\delta}}{g_\delta} \qquad (2.1)$$

donde $\delta = (n\text{-}1, n\text{-}2,...,0)$.

Ejemplo 4. Sea $\lambda = (2,1)$. Entonces el polinomio de Schur $s_\lambda(x_1, x_2)$ se puede calcular así:

$$s_\lambda(x_1, x_2) = {g_{\lambda+\delta}}/{g_\delta} = {g_{(3,1)}}/{g_{(1,0)}} = \frac{\begin{vmatrix} x_1^3 & x_1 \\ x_2^3 & x_2 \end{vmatrix}}{\begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix}} =$$

$$\frac{x_1^3\, x_2 - x_2^3\, x_1}{x_1 - x_2} = x_1^2\, x_2 + x_1\, x_2^2$$

Alternativamente, se pueden definir estos polinomios por medio de la fórmula de Jacobi-Trudi. Para esto, recordemos que si λ es una partición, entonces la partición λ^* es la partición conjugada de λ obtenida al transponer filas y columnas de λ. Denotaremos además por $e_k^n = e_k(x_1,..., x_n)$ o simplemente e_k si el número de variables es claro, al polinomio simétrico elemental

$$e_k = \sum_{i_1<i_2<...<i_k} x_{i_1} x_{i_2} ..., x_{i_k}. \qquad (2.2)$$

Recordemos que un polinomio $f \in \mathbb{Z}[x_1,...x_n]$ se dice simétrico, si para toda permutación $\sigma \in S_n$, $\sigma f = f(x_{\sigma(1)},...x_{\sigma(n)}) = f(x_1,...x_n)$. El conjunto de polinomios simétricos de $\mathbb{Z}[x_1,...x_n]$ lo denotamos por Λ_n. También es bien conocido que el conjunto de polinomios elementales e_λ donde $\lambda = (\lambda_1,...\lambda_k)$ con $k \leq n$ forman una base para los polinomios elementales en Λ_n. Como

$$e_\lambda = e_{\lambda_1} ... e_{\lambda_k} \qquad (2.3)$$

entonces $\Lambda_n = \mathbb{Z}[e_1,...,e_n]$.

Con esto podemos definir el polinomio de Schur para una partición λ como

$$s_\lambda = \det\left[e_{\lambda_i^* - i+j}\right]_{1\leq i,j\leq n} \qquad (2.4)$$

donde λ' y $= (\lambda'_1,...,\lambda'_e)$, $l \leq n$.

Ejemplo 5. Calculemos el polinomio de Schur para la partición $\lambda = (3,2)$ en dos variables, $\lambda = x_1,x_2$.

En este caso $\lambda' = (2,2,1) = (\lambda'_1,\lambda'_2,\lambda'_3)$

y por lo tanto podemos tomar $n = 3$. Así tenemos

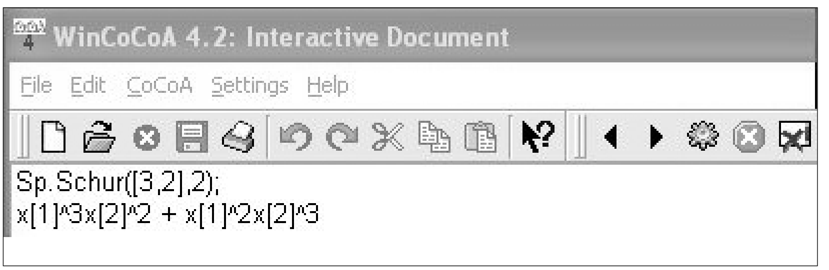
$$s_\lambda(x_1, x_2) = \det \begin{bmatrix} e_2^2 & e_3^2 & e_4^2 \\ e_1^2 & e_2^2 & e_3^2 \\ e_{-1}^2 & e_0^2 & e_1^2 \end{bmatrix}$$

donde se sabe que $e_0 = 1$, $e_p^n = 0$ para $p < 0$ ó $p > n$. Por lo tanto,

$$s_{(3,2)}(x_1, x_2) = \det \begin{bmatrix} x_1x_2 & 0 & 0 \\ x_1+x_2 & x_1x_2 & 0 \\ 0 & 1 & x_1+x_2 \end{bmatrix} =$$

$$(x_1, x_2)^2\,(x_1 + x_2) = x_1^3x_2^2 + x_1^2\,x_2^3$$

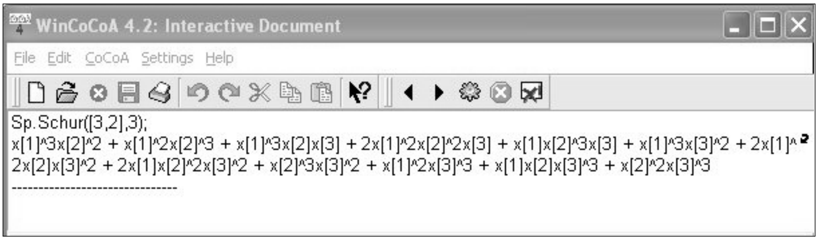
Dicho cómputo lo podemos realizar con el programa **simetricos.pkg** hecho para el sistema computacional algebraico CoCoA, con la rutina **Schur (**λ,k**)** en donde λ es una partición y k es el número de variables, de la siguiente manera:



Para ver otras propiedades generales de los polinomios simétricos y sus principales características se pueden consultar cualquiera de los textos Macdonald 95, Macdonald 91, o Manivel 2001.

Ejemplo 6 Calculemos el polinomio de Schur

$$s_{(3,2)}(x_1, x_2, x_3)$$



y observemos los coeficientes de este polinomio. En primer lugar, todos los coeficientes son positivos, por lo tanto sirven para contar algo. Consideremos el término 2x[1]^2x[2]^2x[3]. Su coeficiente 2 cuenta el número de tablas semi-estándar de la forma $\lambda = (3,2)$ con contenido (2,2,1) es decir que contengan 2 unos, 2 dos y un tres (observe las potencias de las variables). Estas tablas son:

1	1	2
2	3	

1	1	3
2	2	

mientras que el término x[2]^2x[3]^3 indica que solo hay una tabla semi-estándar de la forma con dos 2 y tres 3 y esa tabla es

2	2	3
3	3	

Si T es una tabla que contiene μ_i entradas iguales a i, su contenido se define como la secuencia $(\mu_1,\mu_2,\ldots)$.

En general se tiene que

$$s_\lambda = \sum_T x^{\mu(T)} \tag{2.5}$$

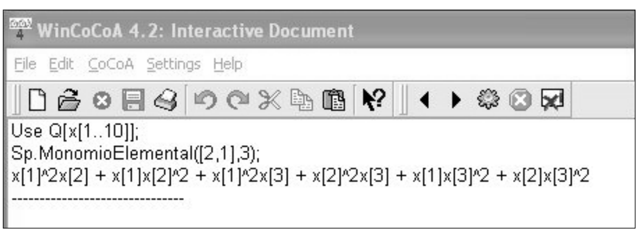
donde la suma corre sobre las tablas semi-estándar de la forma λ y $\mu(T) = (\mu_1,\ldots,\mu_n)$ es el contenido de la tabla T, y $x^{\mu(T)} = x_1^{\mu 1}\ldots x_n^{\mu n}$.

Ahora consideremos los polinomios elementales llamados "monomios" *elementales*. Estos polinomios se definen así: sea $\lambda = (\lambda_1,\ldots,\lambda_k)$ una partición y sea $n \geq k$ entonces

$$m_\lambda\,(x_1,\ldots,x_n) = \sum_{\sigma\in S_n} x_{\sigma(1)}^{\lambda 1}\ldots x_{\sigma(\kappa)}^{\lambda\kappa} \tag{2.6}$$

(sin repetir monomios). Si $n < k$ entonces $m_\lambda\,(x_1,\ldots,x_n) = 0$

Ejemplo 7 El monomio elemental $m_\lambda\,(x_1,x_2,x_3)$ lo calculamos así:



Los monomios elementales forman una base para los polinomios simétricos por lo tanto es natural preguntarse por la descomposición de los miembros de una familia de polinomios simétricos en términos de otra familia de polinomios simétricos que sea base para ese espacio.

Ejemplo 8. Tomemos $\lambda = (3,2)$ como en el ejemplo 6. Los monomios elementales m_π para π partición de 5 calculados con nuestras rutinas son:

```
File  Edit  CoCoA  Settings  Help
U:=$contrib/novoa/rep.Parti2(5);
U;//particiones de 5
[[5], [4, 1], [3, 2], [3, 1, 1], [2, 2, 1], [2, 1, 1, 1], [1, 1, 1, 1, 1]]
-------------------------------
//Todos los monomios elementales para las particiones de 5 en 3 variable
For I:=1 To Len(U) Do
PrintLn 'Para la partición ',U[I],' el  Monomio Elemental en 3 variables es ', Sp.MonomioElemental(U[I],3)
EndFor;
Para la partición [5] el  Monomio Elemental en 3 variables es x[1]^5 + x[2]^5 + x[3]^5
Para la partición [4, 1] el  Monomio Elemental en 3 variables es x[1]^4x[2] + x[1]x[2]^4 + x[1]^4x[3] + x[2]^4x[3] + x[1]x[3]^4
+ x[2]x[3]^4
Para la partición [3, 2] el  Monomio Elemental en 3 variables es x[1]^3x[2]^2 + x[1]^2x[2]^3 + x[1]^3x[3]^2 + x[2]^3x[3]^2 + x[
1]^2x[3]^3 + x[2]^2x[3]^3
Para la partición [3, 1, 1] el  Monomio Elemental en 3 variables es x[1]^3x[2]x[3] + x[1]x[2]^3x[3] + x[1]x[2]x[3]^3
Para la partición [2, 2, 1] el  Monomio Elemental en 3 variables es x[1]^2x[2]^2x[3] + x[1]^2x[2]x[3]^2 + x[1]x[2]^2x[3]^2
Para la partición [2, 1, 1, 1] el  Monomio Elemental en 3 variables es 0
Para la partición [1, 1, 1, 1, 1] el  Monomio Elemental en 3 variables es 0

-------------------------------
```

Veamos como expresamos el polinomio $s_{(3,2)}(x_1,x_2,x_3)$ en términos de estos m_π. Es decir, solucionaremos la ecuación

$$\sum a_\lambda m_\lambda - s_{(3,2)} = 0$$

Con ayuda de rutinas en CoCoA (ver el manual de CoCoA para la función Syz) hallamos que la solución a dicha ecuación con coeficientes enteros:

```
File  Edit  CoCoA  Settings  Help
Alias Sp:=$contrib/novoa/simetricos;//llada al programa que calcula polinomios simetricos
Use Q[x[1..10]];// definición de las variables. Anillo de polinomios
U:=$contrib/novoa/rep.Parti2(5);//particiones de 5
H:=[];//inicialización del vector de monomio elementales en 3 variables
For I:=1 To Len(U) Do
  Append(H,Sp.MonomioElemental(U[I],3));
EndFor;
Append(H,Sp.Schur([3,2],3));
Gens(Syz(H));//generadores del modulo Syzygy
[Vector(0, 0, 0, 0, 0, 1, 0, 0), Vector(0, 0, 0, 0, 0, 0, 1, 0), Vector(0, 0, 1, 1, 2, 0, 0, -1), Vector(0, 0, 0, -x[1]x[2] -
 x[1]x[3] - x[2]x[3], x[1]^2 + x[2]^2 + x[3]^2, 0, 0, 0), Vector(0, 0, 0, -x[1]x[2] - x[1]x[3] - x[2]x[3], x[1]^2 + x[2]^2
+ x[3]^2, 0, 0, 0), Vector(-x[1]x[2] - x[1]x[3] - x[2]x[3], x[1]^2 - x[1]x[2] + x[2]^2 - x[1]x[3] - x[2]x[3] + x[3]^2, x[1]
^2 - 2x[1]x[2] + x[2]^2 - 2x[1]x[3] - 2x[2]x[3] + x[3]^2, 3x[1]^2 + 3x[2]^2 + 3x[3]^2, 2x[1]^2 - 2x[1]x[2] + 2x[2]^2 -
2x[1]x[3] - 2x[2]x[3] + 2x[3]^2, 0, 0, 0), Vector(0, -x[1]x[2] - x[1]x[3] - x[2]x[3], x[1]^2 - x[1]x[2] - x[1]x[3] - x[2]x[
3] + x[3]^2, 2x[1]^2 + 2x[1]x[2] + x[2]^2 + 2x[1]x[3] + 2x[2]x[3] + 2x[3]^2, -2x[1]x[2] - 2x[2]^2 - 2x[1]x[3] - 2x[2]x[
```

De los resultados obtenidos, vemos que la mayoría de vectores no tienen sus componentes en Z. De los tres vectores con componentes enteros, los dos primeros son soluciones triviales que suceden porque

$$m_{(2,1,1,1)}(x_1,x_2,x_3) = m_{(1,1,1,1,1)} = 0$$

y por lo tanto, solo queda como solución no trivial el vector

©0,0,1,1,2,0,0,-1ª

que proporciona la solución con coeficientes enteros para nuestra pregunta.

Es decir,

$$0m_{(5)} + 0m_{(4,1)} + 1m_{(3,2)} + 1m_{(3,1,1)} + 2m_{(2,2,1)} +$$

$$0m_{(2,1,1,1,1)} + 0m_{(1,1,1,1,1)} - 1s_{(3,2)} = 0$$

En general se sabe que si λ es partición de n,

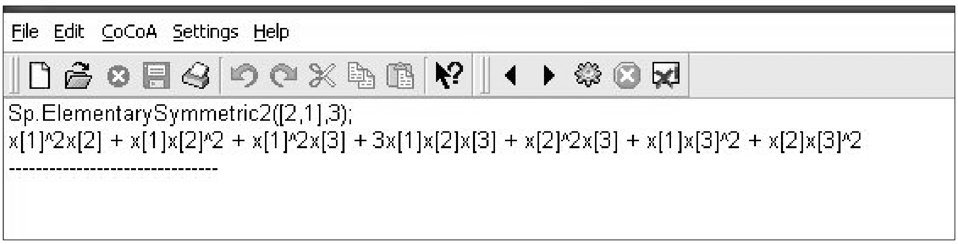
$$s_\lambda = m_\lambda + \sum_{\mu<\lambda} K_{\lambda\mu} m_\mu \qquad (2.7)$$

donde la suma anterior corre sobre las particiones μ de n con $\mu \leq \lambda$ (ordenadas por el orden lexicográfico) y los coeficientes $K_{\lambda\mu}$ representan los *números de Kostka*, que enumeran el número de tablas semiestándar de la forma λ con contenido μ.

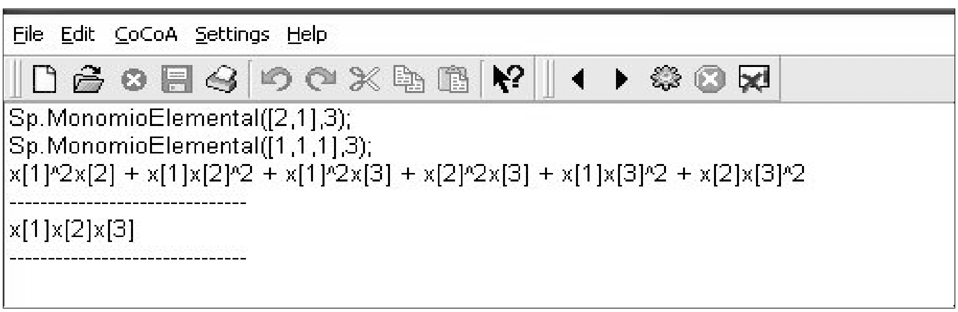
Ejemplo 9. En el ejemplo anterior tenemos que para $\lambda = (3,2)$ y $\mu = (2,2,1)$ el número de Kostka $K_{\lambda\mu} = 2$. Es decir hay dos tablas semiestándar de la forma $\lambda = (3,2)$ y contenido $\mu = (2,2,1)$ y esas tablas son exactamente las del ejemplo 6.

Similares interpretaciones del significado combinatorio de los coeficientes en la descomposición de uno de estos polinomios simétricos respecto a una base de polinomios simétricos siempre son bienvenidas.

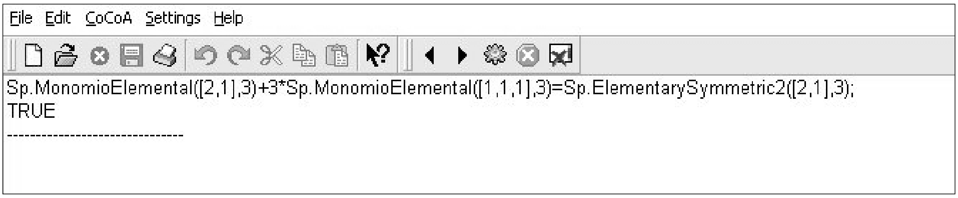
Ejemplo 10. El polinomio elemental $e_{(2,1)}(x_1,\ldots,x_3)$ (ver fórmula 2.3) es:



y los polinomios $m_{(2,1)}$ y $m_{(1,1,1)}$ son:



Entonces (de nuevo gracias a las rutinas en CoCoA) podemos fácilmente ver que:



es decir,

$$m_{(2,1)} + 3m_{(1,1,1)} = e_{(2,1)}$$

La interpretación de estos coeficientes usualmente se hace en términos de matrices con ceros y unos (Manivel 2001).

Antes de definir los polinomios skew de Schur, definamos otra familia de polinomios simétricos llamados polinomios simétricos completos y denotados como h_k.

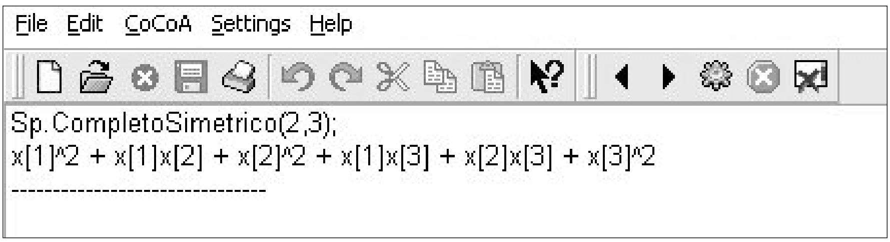
Definimos el r-esimo polinomio simétrico completo en las variables $x_1,\ldots,x_n$, como el polinomio

$$h_r = \sum_{1 \le i_1 \le \cdots \le i_r \le n} x_{i_1} \ldots x_{i_r} \tag{2.8}$$

Similarmente a como se definió en la fórmula 2.3, tenemos que si $\lambda = (\lambda_1,\ldots\lambda_k)$

$$h_\lambda = (h_{\lambda_1},\ldots h_{\lambda_k}) \tag{2.9}$$

Ejemplo 11. El polinomio $h_2\,(x_1,x_2,x_3)$ lo podemos calcular por medio de del programa simetricos.pkg:



POLINOMIOS OBLICUOS DE SCHUR

Sean λ, μ dos particiones. Definimos el polinomio oblicuo de Schur asociado a las particiones λ, μ en las variables $X = x_1,\ldots,x_n$ como

$$s_{\lambda/\mu}(x) = \det\left[e_{\lambda'_i-\mu'_j-i+j}(x)\right]_{1\le i,j\le m}$$

Donde $l(\lambda')\le m$, e_k denota el k-ésimo polinomio elemental simétrico y λ' denota la partición conjugada de λ. Similares definiciones y propiedades de estos polinomios se pueden hallar en (Macdonald 1995). En particular si $\mu \not\subset \lambda$ entonces $s_{\lambda/\mu} = 0$

Ejemplo 11. Sean $\lambda = (2,1)$, $\mu = (1)$, y $X = x_1,x_2$. Note que λ, μ son simétricas, por lo tanto $\lambda = \lambda'$, $\mu = \mu'$. La matriz que determina el polinomio $s_{\lambda/\mu}$ es

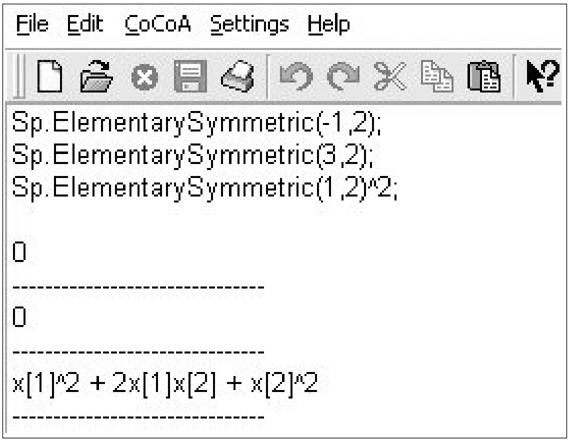
$$\begin{bmatrix} e_1 & e_3 \\ e_{-1} & e_1 \end{bmatrix}$$

por lo tanto

$$s_{\lambda/\mu}(X) = e_1^2(X)$$

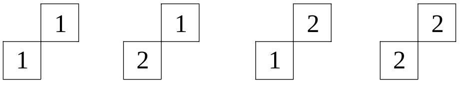
por cuanto $e_{-1}(X) = 0 = e_3(X)$.

Estos cómputos los podemos hacer con el programa simetricos.pkg de la siguiente forma aunque su cálculo a mano es inmediato también:



De esta forma tenemos que $s_{\lambda/\mu}(x) = e_1^2(x) = (x_1 + x_3)^2$.

Adicionalmente, tenemos que las tablas semiestándar de la forma $^{(2,1)}/_{(1)}$ usando los elementos del conjunto $\{1,2\}$ son

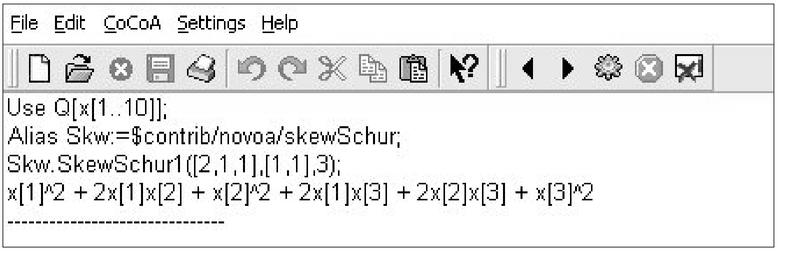


las cuales tienen como monomios asociados x_1^2, x_2^2 y x_1x_2 el cual se repite dos veces, y la relación entre polinomios oblicuos de Schur y tablas oblicuas se hace evidente.

UN PROGRAMA PARA CALCULAR LOS POLINOMIOS SKEW DE SCHUR.

En el ambiente computacional CoCoA haciendo uso de la definición dada en la sección 3 hemos implementado un programa que calcula el polinomio skew de Schur para cualquier número de variables. El programa se ha llamado **skewSchur.pkg** y se usa de la siguiente manera: define primero el número de variables en las que desea trabajar para su anillo de polinomios y use la función *skewSchur*1(λ, μ, n), donde n es el número de variables y $\lambda \geq \mu$ son particiones.

Ejemplo 12. Calculemos el polinomio $s_{\lambda/\mu}$ en tres variables para las particiones $\lambda = (2,1,1)$ $\mu = (1,1)$.



Ejemplo 13. Los polinomios oblicuos de Schur sirven para calcular explícitamente el valor de los coeficientes $c^{\lambda}_{\nu/\mu}$, llamados coeficientes de Littlewood-Richardson de la ecuación 1.1. Estos coeficientes aparecen en diversos problemas de conteo en álgebra, topología, geometría y combinatoria. En particular, estos coeficientes calculan el número de tablas Yamanouchi de cierta forma que tienen un contenido determinado. Más precisamente, $c^{\lambda}_{\nu/\mu}$ cuenta el número de tablas Yamanouchi de la forma λ/μ con contenido ν, en donde una tabla Yamanoichi es una tabla semiestándar cuya palabra $x_1,\ldots, x_k$ satisface que el número de i en $x_s,\ldots, x_k$ es por lo menos igual al número de $i+1$ en $x_s,\ldots, x_k$ para todo i y para todo s, $1\leq s\leq k$-1.

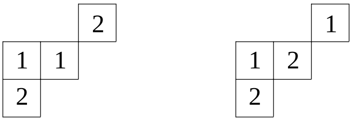


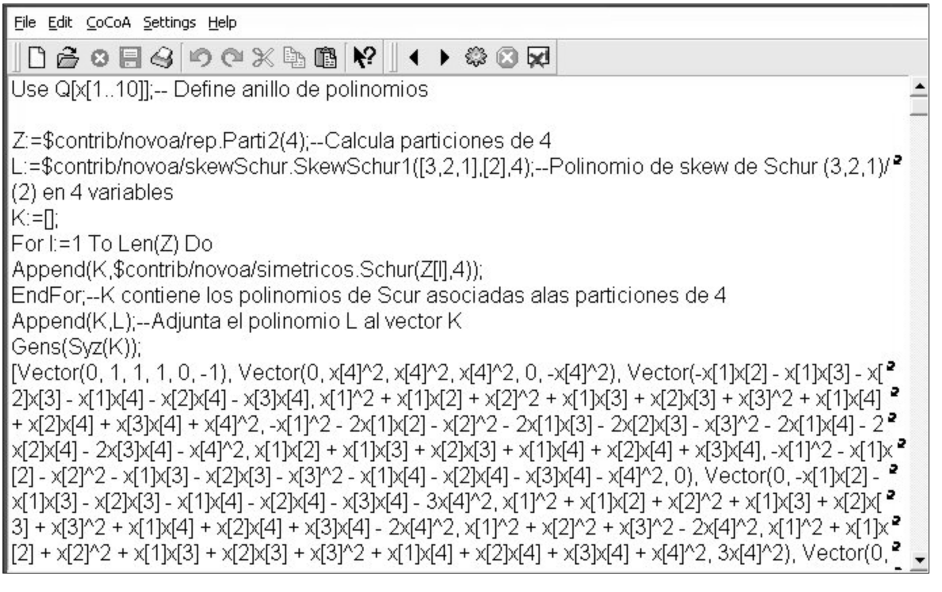
FIGURA 1

La palabra $p(\pi)$ de una tabla π se obtiene al escribir las filas de la tabla una a continuación de otra, de abajo hacia arriba. En el ejemplo, la palabra asociada a la primera tabla en la figura 1 es 2112 mientras que la palabra para la segunda tabla es 2121. La primera palabra no corresponde a una tabla Yamanoichi pues en la subpalabra 2 formada al suprimir los tres caracteres iniciales, no hay por lo menos tantos 1 como 2. En la segunda palabra, tenemos las siguientes subpalabras:

2121
121
21
1

y en todas ellas hay al menos tantos i como i+1, para todo $i\geq 1$.

Según la ecuación 1.1 podemos calcular estos coeficientes simplemente descomponiendo los polinomios oblicuos de Schur en términos de los polinomios de Schur. Sea λ = (3,2,1) y μ = (2). Calculemos los polinomios de Schur en 4 variables y el polinomio $s_{\lambda/\mu}$:



```
File  Edit  CoCoA  Settings  Help
Use Q[x[1..10]];-- Define anillo de polinomios

Z:=$contrib/novoa/rep.Parti2(4);--Calcula particiones de 4
L:=$contrib/novoa/skewSchur.SkewSchur1([3,2,1],[2],4);--Polinomio de skew de Schur (3,2,1)/
(2) en 4 variables
K:=[];
For I:=1 To Len(Z) Do
Append(K,$contrib/novoa/simetricos.Schur(Z[I],4));
EndFor;--K contiene los polinomios de Scur asociadas alas particiones de 4
Append(K,L);--Adjunta el polinomio L al vector K
Gens(Syz(K));
[Vector(0, 1, 1, 1, 0, -1), Vector(0, x[4]^2, x[4]^2, x[4]^2, 0, -x[4]^2), Vector(-x[1]x[2] - x[1]x[3] - x[
2]x[3] - x[1]x[4] - x[2]x[4] - x[3]x[4], x[1]^2 + x[1]x[2] + x[2]^2 + x[1]x[3] + x[2]x[3] + x[3]^2 + x[1]x[4]
+ x[2]x[4] + x[3]x[4] + x[4]^2, -x[1]^2 - 2x[1]x[2] - x[2]^2 - 2x[1]x[3] - 2x[2]x[3] - x[3]^2 - 2x[1]x[4] - 2
x[2]x[4] - 2x[3]x[4] - x[4]^2, x[1]x[2] + x[1]x[3] + x[2]x[3] + x[1]x[4] + x[2]x[4] + x[3]x[4], -x[1]^2 - x[1]x
[2] - x[2]^2 - x[1]x[3] - x[2]x[3] - x[3]^2 - x[1]x[4] - x[2]x[4] - x[3]x[4] - x[4]^2, 0), Vector(0, -x[1]x[2] -
x[1]x[3] - x[2]x[3] - x[1]x[4] - x[2]x[4] - x[3]x[4] - 3x[4]^2, x[1]^2 + x[1]x[2] + x[2]^2 + x[1]x[3] + x[2]x[
3] + x[3]^2 + x[1]x[4] + x[2]x[4] + x[3]x[4] - 2x[4]^2, x[1]^2 + x[2]^2 + x[3]^2 - 2x[4]^2, x[1]^2 + x[1]x
[2] + x[2]^2 + x[1]x[3] + x[2]x[3] + x[3]^2 + x[1]x[4] + x[2]x[4] + x[3]x[4] + x[4]^2, 3x[4]^2), Vector(0,
```

De todas estas soluciones obtenemos una única solución entera (0,1,1,1,0,-1), que según el orden que tenemos de las particiones de 4, representa

$$0s_{(4)} + 1s_{(3,1)} + 1s_{(2,2)} + 1s_{(2,1,1)} + 0s_{(1,1,1,1)} -$$

$$s_{\lambda/\mu} = 0.$$

Finalmente, estas únicas tablas Yamanouchi de la forma con esos contenidos son:

$$\begin{array}{ccc} & & 1 \\ 1 & 1 & \\ 2 & & \end{array} \qquad \begin{array}{ccc} & & 1 \\ 1 & 2 & \\ 2 & & \end{array} \qquad \begin{array}{ccc} & & 1 \\ 1 & 2 & \\ 3 & & \end{array}$$

CONCLUSIONES

Los programas simetricos.pkg y skewSchur.pkg hechos para el sistema computacional CoCoA calculan los polinomios simétricos monomiales, potencias, completos, elementales y de Schur, y los polinomios oblicuos de Schur en varias variables. Los coeficientes enteros que aparecen en la descomposición de uno de estos polinomios como combinación lineal de polinomios de una de estas familias, tienen usualmente una explicación combinatoria y representa en muchos casos el conteo de diversos tipos de tablas de Young de alguna forma, que satisfacen ciertas condiciones. Por medio de esta descomposición directa, podemos calcular estos coeficientes y de esta forma hallar el número de tablas que satisfacen una propiedad. En general, es mucho más complicado obtener fórmulas cerradas para el cómputo del número de estas tablas que hallar los coeficientes de las combinaciones lineales de la descomposición de los polinomios en términos de distintas bases polinomiales.

Otras aplicaciones adicionales de estos programas se presentan en el estudio de funciones de correlación de procesos de Schur y de la medida de Plancherel, dado que las probabilidades en estos procesos son proporcionales a productos de polinomios oblicuos Schur o variaciones de estos.

LITERATURA CITADA

MACDONALD I.G. 1995. Symmetric functions and Hall polynomials. *Oxford Science Publications*.

MACDONALD I.G. 1991. Symmetric functions and Orthogonal polynomials. *University Lecture series* Vol. 2. American Mathematical Society.

MANIVEL L. 2001. Symmetric functions, Schubert polynomials and degeneracy locy. *American Mathematical Society*.

SAGAN B. 2001. The symmetric group. Representations, combinatorial algorithms, and symmetric functions. *Springer-Verlag*. Second edition.

OKOUNKOV A. and RESHETIKIN N. 2003. Correlation function of Schur process with application to local geometry of a random 3-dimensional Young diagram. arXiv:math. CO/0107056 v3.

Recibido: 11-11-2003
Aceptado: 2-02-2005