ON DIFFERENT APPROACHES TO COMPUTE THE CHERN CLASSES OF A TENSOR PRODUCT OF TWO VECTOR BUNDLES

OLEKSAND IENA

ABSTRACT. We compare four different approaches to compute the Chern classes of a tensor product of two vector bundles in terms of the Chern classes of the factors.

1. INTRODUCTION, SOME DEFINITIONS AND NOTATIONS

1.1. **Motivation.** The computation of the Chern classes of a tensor product of two vector bundles in terms of the Chern classes of the factors does not provide any theoretical difficulties. In practice, however, different formulas and approaches can be very different from the point of view of the time needed to obtain the result.

Our aim is to compare the efficiency of four different approaches. More precisely, we shall compare the corresponding implementations from [3]. We compare the elimination approach, the approach using the multiplicativity of the Chern character, the Lascoux's formula [6], and the Manivel's formula [8].

1.2. **Chern classes and the Chern character.** Let X be a complex manifold of dimension N. Let E be a complex vector bundle of rank r on X. For a non-negative integer k, the k-th Chern class $c_k = c_k(E)$ of E is the k-th elementary symmetric polynomial

$$c_k = e_k(a_1, \dots, a_r) \tag{1}$$

in the so-called Chern roots $a_1, \dots, a_r$ of E, which are elements of degree 1 in a certain commutative graded ring. The higher Chern classes $c_i = c_i(E)$, for $i > r$, are defined to be zero. The total Chern class $c = c(E)$ of E is defined to be

$$c = (1 + a_1) \cdot (1 + a_2) \dots (1 + a_r) = c_0 + c_1 + c_2 + \dots + c_r.$$

The Chern character $\mathrm{ch} = \mathrm{ch}(E)$ is defined by $\mathrm{ch} = \sum_{i=1}^{r} \exp(a_i)$. In other words,

$$\mathrm{ch} = r + \mathrm{ch}_0 + \mathrm{ch}_2 + \dots + \mathrm{ch}_N,$$

where ch_k is the sum of k-th powers of the Chern roots multiplied by $\frac{1}{k!}$

$$\mathrm{ch}_k = \frac{1}{k!} \cdot p_k(a_1, \dots, a_r), \quad p_k(a_1, \dots, a_r) = \sum_{i=1}^{r} a_i^k.$$

Since ch_k is a symmetric polynomial of degree k in the Chern roots $a_1, \dots, a_r$, one can express them as polynomials in the Chern classes with rational coefficients, hence

$$\mathrm{ch}_k = \mathrm{ch}_k(c_1, \dots, c_k) \in \mathbb{Q}[c_1, \dots c_r].$$

2010 *Mathematics Subject Classification.* 13P99.

Key words and phrases. Chern classes, tensor product of vector bundles, symbolic computations.

For a given partition $I = (i_1, \dots, i_m)$, $0 \leqslant i_1 \leqslant i_2 \leqslant \dots \leqslant i_m$, the Schur polynomial of E can be defined in terms of the Chern classes by the second Jacobi-Trudi (second determinantal) formula

$$S_I = S_I(E) = \det \begin{pmatrix} c_{j_1} & c_{j_2+1} & \cdots & c_{j_n+n-1} \\ c_{j_1-1} & c_{j_2} & \cdots & c_{j_n+n-2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{j_1-n+1} & c_{j_2-n+2} & \cdots & c_{j_n} \end{pmatrix}, \tag{2}$$

where $\tilde{I} = (j_1, \dots, j_n)$ is the partition dual (conjugate) to I.

2. Chern classes of tensor products of vector bundles

Let E be a vector bundle of rank r with Chern roots $a_1, \dots, a_r$. Let F be a vector bundle of rank s with Chern roots $b_1, \dots, b_s$.

The set of the Chern roots of the tensor product $E \otimes F$ coincides with the set of sums

$$a_i + b_j, \quad 1 \leqslant i \leqslant r, \quad 1 \leqslant j \leqslant s.$$

The total Chern class of $E \otimes F$ equals

$$c(E \otimes F) = \prod_{1 \leqslant i \leqslant r, 1 \leqslant j \leqslant s} (1 + a_i + b_j). \tag{3}$$

As the Chern classes of $E \otimes F$ are elementary symmetric polynomials in $a_i + b_j$, they can be expressed as polynomials in the Chern classes of E and F.

There are different approaches and formulas for computation of the Chern classes of $E \otimes F$.

2.1. **Elimination.**
Eliminating a_i and b_j from (3) using (1), one gets the expression for $c(E \otimes F)$ in terms of the Chern classes of E and F.

2.2. **Multiplicativity of the Chern character and the Newton identities.**
One can compute $c(E \otimes F)$ using the equality $\mathrm{ch}(E \otimes F) = \mathrm{ch}(E) \cdot \mathrm{ch}(F)$, and the Newton's identities:

$$p_{k+1} = c_1 \cdot p_k - c_2 \cdot p_{k-1} + \dots + (-1)^k (k+1) c_{k+1}, \quad k \in \mathbb{Z}_{\geqslant 0}. \tag{4}$$

2.3. **Lascoux's formula.**
A. Lascoux in [6] provides an explicit formula for $c(E \otimes F)$ in terms of Schur polynomials of E and F:

$$c(E \otimes F) = \sum (I, J) \cdot S_J(E) \cdot S_{\mathbf{C}_r \tilde{I}}(F), \tag{5}$$

where the sum runs over the partitions $J = (j_1, j_2, \dots, j_r)$, $I = (i_1, i_2, \dots, i_r)$, $0 \leqslant j_\nu \leqslant i_\nu \leqslant s$, $(I, J) = \det \left(\binom{i_\mu + \mu - 1}{j_\nu + \nu - 1} \right)_{1 \leqslant \mu, \nu \leqslant m}$, and $\mathbf{C}_r \tilde{I}$ denotes the complement of $\tilde{I}$ with respect to r.

2.4. **Manivel's formula.**
A variation of the Lascoux's formula is given by L. Manivel in [8]. The coefficients (I, J) turn out to be polynomials in the ranks r and s. They can be computed in terms of the content polynomials of skew partitions and Littlewood-Richardson coefficients. The Manivel's formula reads as

$$c(E \otimes F) = \sum_{I, J} P_{I,J}(r, s) \cdot S_I(E) \cdot S_J(F), \tag{6}$$

with

$$P_{I,J}(r, s) = \sum_K c^{\tilde{K}}_{\tilde{I}, J} \cdot (r | K - I) \cdot (s | \tilde{K} - J) / h(K),$$

where $c_{\tilde{I},J}^{\tilde{K}}$ are Littlewood-Richardson coefficients (cf. [7, p. 142]), which can be non-zero only if $I \subset K$ and $J \subset \tilde{K}$, $h(K)$ is the product of all hook-lengths of K, and the polynomials $(r|K-I)$, $(s|\tilde{K}-J)$ (in r and s respectively) are the quotients of the content polynomials of the corresponding partitions (cf. [7, p. 15]).

3. Comparison of different approaches

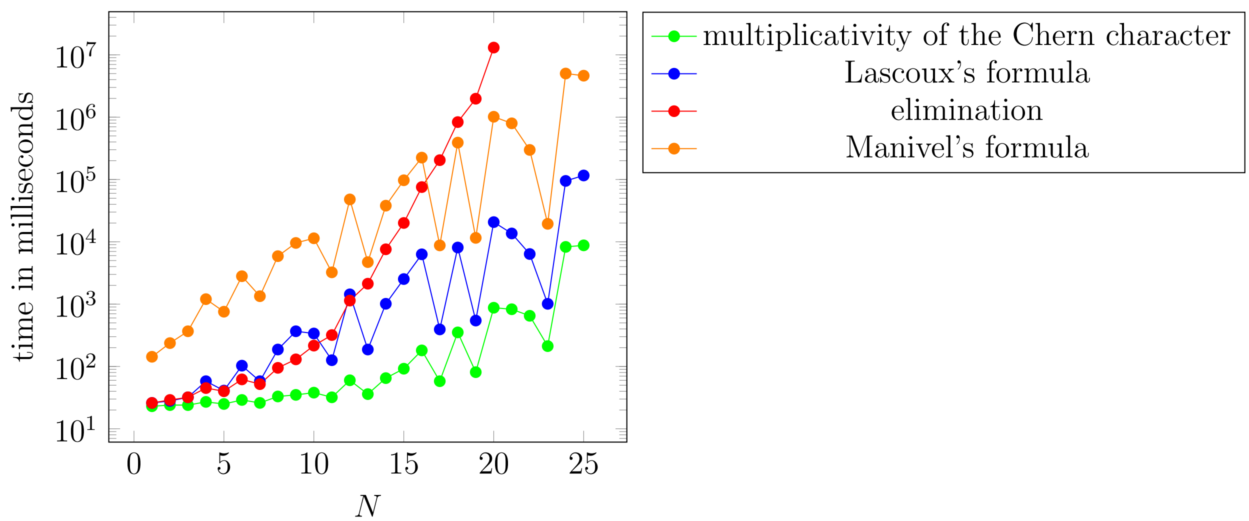
The approaches mentioned above have been implemented in the library `chern.lib` [3] for the computer algebra system Singular [2]. An informal description of the library is presented in [5].

The implementation of the Manivel's formula uses the Littlewood-Richardson calculator by Anders Koch [1] (via the interface [4] for Singular).

The diagram below shows the times needed to compute the Chern classes of all possible tensor products of vector bundles E and F such that $\operatorname{rank}(E \otimes F) = N$.

Unsurprisingly, the elimination approach turns out to be the slowest one (for big N). The approach using the multiplicativity of the Chern character is the fastest one. The Lascoux's formula is the second fastest approach. Its modification by Manivel is the second slowest one. For small N using the Manivel's formula is even more time consuming that elimination.

Computation of the Chern classes of tensor product:
performed on Intel(R) Xeon(R) CPU E7-4850 @ 2.000Ghz
with 188.9GB of RAM



Appendix A. The code in Singular

We provide here the Singular code used to test different implementations.

```
LIB "chern.lib";
// procedure that computes all pairs (a, b) with ab=n
proc PRD(int n){...}
// the test procedure that computes the Chern classes
// of all tensor products of rank N
proc TST(int N)
{
```

```
  ring r= 0,(c(1..N), C(1..N)), dp;
  int j, m, n, sz;
  list L, c, C;
  L=PRD(N); // pairs (m, n) with mn=N
  sz=size(L);
  for(j=1; j<= sz; j++)
  {
    m=L[j][1];
    n=L[j][2];
    "";
    // uncomment the required method to be tested:
    // for elimination
    //chProdE(list( c(1..m) ), list( C(1..n) ) );
    //
    // for the multiplicativity of the Chern character
    //chProd( m, list( c(1..m) ),  n, list( C(1..n) ) );
    //
    // for Lascoux's formula
    //chProdL( m, list( c(1..m) ),  n, list( C(1..n) ) );
    //
    // for Manivel's formula
    chProdM( m, list( c(1..m) ),  n, list( C(1..n) ) );
  }
}
TST(4); // running a test for N=4 for Manivel's formula
$
```

References

[1] Anders Buch. Littlewood-Richardson calculator. `https://www.math.rutgers.edu/~asbuch/lrcalc/`, 2014.

[2] Wolfram Decker, Gert-Martin Greuel, Gerhard Pfister, and Hans Schönemann. Singular 4-0-2 — A computer algebra system for polynomial computations. `http://www.singular.uni-kl.de`, 2015.

[3] Oleksandr Iena. chern.lib, a singular library for symbolic computations with Chern classes. `http://hdl.handle.net/10993/21949`, June 2015.

[4] Oleksandr Iena. lrcalc.lib, a singular interface to the Littlewood-Richardson calculator by Anders Buch. `http://hdl.handle.net/10993/22928`, December 2015.

[5] Oleksandr Iena. On symbolic computations with Chern classes: remarks on the library chern.lib for singular. `http://hdl.handle.net/10993/22395`, 2015.

[6] Alain Lascoux. Classes de Chern d'un produit tensoriel. *C. R. Acad. Sci. Paris Sér. A-B*, 286(8):A385–A387, 1978.

[7] I. G. Macdonald. *Symmetric functions and Hall polynomials. Reprint of the 1998 2nd edition.* Oxford: Oxford University Press, reprint of the 1998 2nd edition edition, 2015.

[8] L. Manivel. Chern classes of tensor products. *ArXiv e-prints*, November 2010.