Journal of Software for Algebra and Geometry

Software for computing conformal block divisors on $\overline{M}_{0,n}$

DAVID SWINARSKI

vol 8 2018

JSAG **8** (2018), 81–86
dx.doi.org/10.2140/jsag.2018.8.81

The Journal of Software for
Algebra and Geometry

Software for computing conformal block divisors on $\overline{M}_{0,n}$

DAVID SWINARSKI

ABSTRACT: We introduce the packages `LieTypes.m2` and `ConformalBlocks.m2` for Macaulay2. `LieTypes.m2` contains basic types for working with Lie algebras and Lie algebra modules. `ConformalBlocks.m2` computes ranks and first Chern classes of vector bundles of conformal blocks on $\overline{M}_{0,n}$.

1. INTRODUCTION. The moduli stacks $\overline{\mathcal{M}}_{g,n}$ of Deligne–Mumford stable n-pointed curves of genus g are central objects of study in algebraic geometry and mathematical physics. The WZW model of conformal field theory can be interpreted as defining vector bundles on $\overline{\mathcal{M}}_{g,n}$ whose fibers are the so-called vector spaces of conformal blocks. These vector bundles were first constructed by Tsuchiya, Ueno [2008], and Yamada; their ranks are computed by the famous Verlinde formula.

We omit the lengthy full definition of conformal blocks (see the references [Beauville 1996] and [Ueno 2008]) and instead merely describe the input required to specify a conformal block. Let $\mathfrak{g}$ be a simple Lie algebra, and let ℓ be a positive integer called the *level*. Choose a set of simple roots for the root system associated to $\mathfrak{g}$, and let θ be the highest root. Let $(-,-)$ denote the Killing form, normalized so that $(\theta,\theta)=2$.

Proposition 1.1. *Let g and n be nonnegative integers satisfying $3g-3+n\geq 0$. Let ℓ be a positive integer. Let $\vec{\lambda}=(\lambda_1,\ldots,\lambda_n)$ be an n-tuple of weights with $(\lambda_i,\theta)\leq\ell$ for each $i=1,\ldots,n$. For each such triple $(\mathfrak{g},\ell,\vec{\lambda})$, we may construct a vector bundle $\mathbb{V}(\mathfrak{g},\ell,\vec{\lambda})$ on $\overline{\mathcal{M}}_{g,n}$, called the vector bundle of conformal blocks.*

In 2008, Fakhruddin gave formulas for the Chern classes of these vector bundles [Fakhruddin 2012]. We will refer to the first Chern class of a conformal block bundle as a *conformal block divisor*. The package `ConformalBlocks.m2` implements some of Fakhruddin's main formulas in the genus 0 case.

Several quantities from representation theory appear in Fakhruddin's formulas, and the earliest version of `ConformalBlocks.m2` contained several functions for

MSC2010: primary 14D21; secondary 14D22, 81T40.
Keywords: conformal blocks, fusion product, moduli of curves.
ConformalBlocks.m2 version 2.4
LieTypes.m2 version 0.5

representation theory calculations. At the suggestion of Grayson and Stillman, these were moved into a separate package, `LieTypes.m2`.

2. THE LIETYPES.M2 PACKAGE. The `LieTypes.m2` package defines two new classes, `LieAlgebra` and `LieAlgebraModule`; objects of these classes are hash tables. Currently, only simple Lie algebras over $\mathbb{C}$ are implemented. (Volunteers who would like to extend the functionality of this package are invited to contact the author.) Simple Lie algebras over $\mathbb{C}$ are specified by their rank and root system type. Irreducible Lie algebra modules are specified by their underlying Lie algebra and highest weight, and a general Lie algebra module is specified by the multiplicities of the irreducible submodules it contains.

The `LieTypes.m2` package contains several functions implementing basic Lie algebra data, such as the Cartan matrix. The documentation within the package contains references for formulas and/or sources of reference data for each of these functions. This package uses Macaulay2's combinatorial and linear algebra functions.

2.1. ***Tensor coefficients and fusion coefficients.*** One notable feature of the package `LieTypes.m2` is that it computes tensor product decompositions and fusion product decompositions for all irreducible root system types.

Let V_λ denote the irreducible $\mathfrak{g}$-module with highest weight λ. Define the tensor product coefficients $N_{\lambda,\mu}^{\nu}$ by

$$V_\lambda \otimes V_\mu = \bigoplus V_\nu^{\oplus N_{\lambda,\mu}^{\nu}}.$$

The `LieTypes.m2` package uses the Racah–Speiser algorithm for computing tensor product coefficients [Di Francesco et al. 1997, 13.5.2].

In type A (that is, $\mathfrak{g} = \mathfrak{sl}_k$), the tensor product coefficients are the Littlewood–Richardson coefficients. These coefficients have been previously implemented in other Macaulay2 packages (e.g., `SchurRings.m2`).

The fusion product $\otimes_\ell$ is a product for integrable level ℓ modules over an affine Lie algebra $\hat{g}$. The fusion coefficients $N_{\lambda,\mu}^{(\ell)\nu}$ are defined by the decomposition of the fusion product, and can be computed using the Kac–Walton algorithm (see [Di Francesco et al. 1997, § 16.2.2]). The Kac–Walton algorithm is closely related to the Racah–Speiser algorithm for tensor products, and it is defined entirely using the combinatorics of the root system of the underlying finite-dimensional Lie algebra. Therefore, we can abuse notation and use the Kac–Walton algorithm to define a product $\otimes_\ell$ on Lie algebra modules as well as affine Lie algebra modules.

Fusion coefficients have previously been implemented in KAC and Magma; but, to the author's knowledge, the implementation in `LieTypes.m2` in Macaulay2 is the first free, open-source implementation of fusion coefficients.

As an example, let $\mathfrak{g} = \mathfrak{sl}_3$. Let ω_1 and ω_2 be the fundamental dominant weights, and $\lambda = 2\omega_1 + \omega_2 = (2, 1)$, $\mu = \omega_1 + 2\omega_2 = (1, 2)$. The calculation below shows

that the tensor product $V_{(2,1)} \otimes V_{(1,2)}$ contains two copies of $V_{(1,1)}$, while the level 3 fusion product $V_{(2,1)} \otimes_3 V_{(1,2)}$ contains one copy of $V_{(1,1)}$. The information computed by the tensor product and fusion product functions is sufficient to determine the characters of these products, though characters are not implemented in this version of `LieTypes.m2`.

```
i1 : loadPackage("LieTypes");
i2 : sl_3=simpleLieAlgebra("A",2)
o2 = Simple Lie algebra, type A, rank 2
o2 : LieAlgebra
i3 : U=irreducibleLieAlgebraModule({2,1},sl_3);
i4 : V=irreducibleLieAlgebraModule({1,2},sl_3);
i5 : W=irreducibleLieAlgebraModule({1,1},sl_3);
i6 : tensorCoefficient(U,V,W)
o6 = 2
i7 : fusionCoefficient(U,V,W,3)
o7 = 1
```

3. THE CONFORMALBLOCKS.M2 PACKAGE. The `ConformalBlocks.m2` package implements some of Fakhruddin's formulas for conformal block divisors on the moduli space of pointed genus 0 curves $\overline{M}_{0,n}$. Its three main functions compute

(1) the rank of a conformal block bundle,

(2) the intersection number of a conformal block divisor with an F-curve,

(3) the divisor class of the symmetrization of a conformal block divisor.

The version of this package described here uses Macaulay2's combinatorial and linear algebra functions.

Some references for divisors and curves on $\overline{M}_{0,n}$ include [Keel and McKernan 2013; Keel 1992; Arap et al. 2012]. The boundary $\Delta = \partial\overline{M}_{0,n}$ (that is, the locus parametrizing nodal curves) consists of irreducible components Δ_I. These span $\mathrm{Pic}(\overline{M}_{0,n}, \mathbb{Q})$. Moreover, the symmetrizations of the classes Δ_I yield a basis $\{B_2, \ldots, B_{\lfloor n/2 \rfloor}\}$ of $\mathrm{Pic}(\overline{M}_{0,n}, \mathbb{Q})^{S_n}$. The `ConformalBlocks.m2` package implements S_n-symmetric divisors in a new class called `SymmetricDivisorM0nbar`. Divisors may be entered/viewed as linear polynomials in the classes B_i. For instance, the divisor $B_2 + B_3 + 2B_4$ on $\overline{M}_{0,8}$ could be created with the command `symmetricDivisorM0nbar(8,B_2+B_3+2*B_4)`. There are methods, for the `SymmetricDivisorM0nbar` class, for creating and comparing divisors, as well as addition, negation, scalar multiplication, and printing.

We will also be interested in certain combinatorially defined curves in the moduli space called F-curves. These are denoted F_{I_1,I_2,I_3,I_4}, where $I_1 \sqcup I_2 \sqcup I_3 \sqcup I_4$ is a partition of $\{1, \ldots, n\}$ into four nonempty subsets. Averaging such a curve with its S_n translates gives a symmetric curve class; if $\#I_1 = a, \#I_2 = b, \#I_3 = c, \#I_4 = d$, we write $F_{a,b,c,d}$ for this class. The classes $\{F_{j,1,1,n-j-2}\}_{j=1}^{\lfloor n/2 \rfloor - 1}$ form an ordered basis of $H_2(\overline{M}_{0,n}, \mathbb{Q})^{S_n}$.

3.1. ***Ranks of conformal block bundles.*** The function `conformalBlockRank` in `ConformalBlocks.m2` computes ranks of conformal block bundles recursively using propagation and factorization (see [Beauville 1996, Corollary 2.4 and page 84]). We abbreviate $r_{\vec{\lambda}} = \operatorname{rank} \mathbb{V}(\mathfrak{g}, \ell, \vec{\lambda})$ if this will cause no confusion.

In practice, propagation means that if one of the weights is zero, we may drop it. Specifically, let $\vec{\lambda} = (\lambda_1, \ldots, \lambda_n)$, and suppose that $\lambda_n = 0$. Then $\mathbb{V}(\mathfrak{g}, \ell, \vec{\lambda}) = \pi_n^* \mathbb{V}(\mathfrak{g}, \ell, \hat{\lambda})$, where $\hat{\lambda} = (\lambda_1, \ldots, \lambda_{n-1})$ and $\pi_n : \overline{M}_{0,n} \to \overline{M}_{0,n-1}$ is the map forgetting the n-th marked point. In particular, $r_{\vec{\lambda}} = r_{\hat{\lambda}}$.

The factorization rules for conformal block bundles refer to a specific direct sum decomposition of each fiber. We merely state the consequence of factorization for ranks: Let $\vec{\mu} \cup \vec{\nu}$ be a partition of the vector $\vec{\lambda} = (\lambda_1, \ldots, \lambda_n)$ into two vectors, each of length at least 2. Then

$$r_{\vec{\lambda}} = \sum_{\beta \in P_\ell} r_{\vec{\mu} \cup \beta} r_{\vec{\nu} \cup \beta^*}.$$

Here $*$ denotes the involution on the root system given by $-w_0$, where w_0 is the longest word in the Weyl group. Formulas for the action of this involution for the simple Lie algebras are given in [Di Francesco et al. 1997, page 511] and implemented in `LieTypes.m2` with the `starInvolution` function.

To seed the recursion, we must know the ranks of conformal block bundles for n=3. We get these from the fusion coefficients by $\operatorname{rank} \mathbb{V}(\mathfrak{g}, \ell, (\lambda, \mu, \nu)) = N_{\lambda,\mu}^{(\ell)\nu^*}$.

As an example, we compute $\operatorname{rank} \mathbb{V}(\mathfrak{sl}_2, 3, (\omega_1, \ldots, \omega_1))$ on $\overline{M}_{0,8}$:

```
i8 : loadPackage("ConformalBlocks");
i9 : sl_2=simpleLieAlgebra("A",1);
i10 : V=conformalBlockVectorBundle(sl_2,3,{{1},{1},{1},{1},{1},{1},{1},{1}},0);
i11 : conformalBlockRank(V)
o11 = 13
```

3.2. ***Intersection numbers with F-curves.*** Fakhruddin uses factorization to express intersection numbers of $c_1\mathbb{V}(\mathfrak{g}, \ell, \vec{\lambda})$ with an F-curve in terms of degrees of conformal blocks on $\overline{M}_{0,4} \cong \mathbb{P}^1$ and ranks of conformal blocks on $\overline{M}_{0,n'}$ with $n' < n$ [Fakhruddin 2012, Proposition 2.7]. This formula is implemented in the function `FCurveDotConformalBlockDivisor`.

```
i12 : w={{1},{1},{1},{1},{1},{1}};
i13 : V=conformalBlockVectorBundle(sl_2,1,w,0)
o13 = V
o13 : Conformal block vector bundle on M-0-6-bar
i14 : conformalBlockRank(V)
o14 = 1
i15 : FCurveDotConformalBlockDivisor({{1,2,3},{4},{5},{6}},V)
o15 = 1
i16 : FCurveDotConformalBlockDivisor({{1,2},{3,4},{5},{6}},V)
o16 = 0
```

Line o14 tells us that the vector bundle $\mathbb{V}(\mathfrak{sl}_2, 1, (\omega_1, \ldots, \omega_1))$ is a line bundle. The intersection numbers computed in o15 and o16 allow us to give a geometric interpretation of this divisor (see [Alexeev et al. 2014, Theorem 7.2] for details): Let $f : M_{0,6}/S_6 \xrightarrow{\cong} H_2$ be the map which identifies a smooth genus 2 curve with the branch points of its g^1_2. This extends to a map $f : \overline{M}_{0,6}/S_6 \xrightarrow{\cong} \overline{H}_2$ using the theory of admissible covers. By comparing the intersection numbers computed above to those of the pullback $f^*\lambda$ of the λ class on $\overline{M}_2$, we see that $\mathbb{V}(\mathfrak{sl}_2, 1, (\omega_1, \ldots, \omega_1))$ is a multiple of $f^*\lambda$.

3.3. ***Divisor classes of symmetric or symmetrized bundles.*** The S_n-symmetric divisors play an important role in the study of the birational geometry of $\overline{M}_{0,n}$. In addition, they are much easier to study, since $\dim \mathrm{Pic}(\overline{M}_{0,n}, \mathbb{Q}) = 2^{n-1} - \binom{n}{2} - 1$ while $\dim \mathrm{Pic}(\overline{M}_{0,n}, \mathbb{Q})^{S_n} = \lfloor n/2 \rfloor - 1$.

Fakhruddin [2012, Corollary 3.6] gives a formula for computing the symmetrization $\sum_{\sigma \in S_n} c_1\mathbb{V}(\mathfrak{g}, \ell, \sigma\vec{\lambda})$ of a conformal block divisor over its S_n-translates. This is implemented in the function `symmetrizedConformalBlockDivisor` for an arbitrary n-tuple of weights $\vec{\lambda}$. This function can also be used and is even faster if the set of weights is already S_n-symmetric.

In the example below, we compute $c_1\mathbb{V}(\mathfrak{sl}_6, 1, (\omega_2, \ldots, \omega_2))$ for $n = 6$:

```
i17 : sl_6=simpleLieAlgebra("A",5);
i18 : w2={0,1,0,0,0};
i19 : V=conformalBlockVectorBundle(sl_6,1,apply(6, i -> w2),0);
i20 : D=symmetrizedConformalBlockDivisor(V)
o20 = 288*B  + 864*B
           2        3
o20 : S_6-symmetric divisor on M-0-6-bar
i21 : coefficientList D
o21 = {288, 864}
o21 : List
i22 : coefficientList scale D
o22 = {1, 3}
o22 : List
```

We see that $c_1\mathbb{V}(\mathfrak{sl}_6, 1, (\omega_2, \ldots, \omega_2))$ is a multiple of $B_2 + 3B_3$. The pullback to $\overline{M}_{0,6}$ of the distinguished polarization on the GIT quotient $(\mathbb{P}^1)^6 /\!/ \mathrm{SL}_2$ with the symmetric linearization is also a multiple of $B_2 + 3B_3$; GIT divisors of this form are studied in [Alexeev and Swinarski 2012].

ACKNOWLEDGEMENTS. I am grateful to Valery Alexeev and Angela Gibney, who helped me learn about conformal blocks. I am also grateful to Mike Stillman and Dan Grayson for expert Macaulay2 programming advice, and to Greg Smith and the anonymous referees for several suggestions for improving the packages, their documentation, and this article. I would also like to thank Amelia Taylor and Frank Moore for inviting me to Macaulay2 workshops at Colorado College and

Wake Forest University where this package was completed; these workshops were partially supported by the National Science Foundation under Grant No. 0964128 and the National Security Agency under Grant No. H98230-10-1-0218. This work was also partially supported by the University of Georgia's NSF VIGRE grant DMS-03040000.

SUPPLEMENT. The online supplement contains version 0.5 of LieTypes.m2 and version 2.4 of ConformalBlocks.m2.

REFERENCES.

[Alexeev and Swinarski 2012] V. Alexeev and D. Swinarski, "Nef divisors on $\overline{M}_{0,n}$ from GIT", pp. 1–21 in *Geometry and arithmetic*, edited by C. Faber et al., EMS Ser. Congr. Rep. **8**, Eur. Math. Soc., Zürich, 2012. MR Zbl

[Alexeev et al. 2014] V. Alexeev, A. Gibney, and D. Swinarski, "Higher-level $\mathfrak{sl}_2$ conformal blocks divisors on $\overline{M}_{0,n}$", *Proc. Edinb. Math. Soc.* (2) **57**:1 (2014), 7–30. MR Zbl

[Arap et al. 2012] M. Arap, A. Gibney, J. Stankewicz, and D. Swinarski, "sl_n level 1 conformal blocks divisors on $\overline{M}_{0,n}$", *Int. Math. Res. Not.* **2012**:7 (2012), 1634–1680. MR

[Beauville 1996] A. Beauville, "Conformal blocks, fusion rules and the Verlinde formula", pp. 75–96 in *Proceedings of the Hirzebruch 65 Conference on Algebraic Geometry* (Ramat Gan, 1993), edited by M. Teicher, Israel Math. Conf. Proc. **9**, Bar-Ilan Univ., Ramat Gan, 1996. MR Zbl

[Di Francesco et al. 1997] P. Di Francesco, P. Mathieu, and D. Sénéchal, *Conformal field theory*, Springer, 1997. MR Zbl

[Fakhruddin 2012] N. Fakhruddin, "Chern classes of conformal blocks", pp. 145–176 in *Compact moduli spaces and vector bundles*, edited by V. Alexeev et al., Contemp. Math. **564**, Amer. Math. Soc., Providence, RI, 2012. MR Zbl

[Keel 1992] S. Keel, "Intersection theory of moduli space of stable n-pointed curves of genus zero", *Trans. Amer. Math. Soc.* **330**:2 (1992), 545–574. MR Zbl

[Keel and McKernan 2013] S. Keel and J. McKernan, "Contractible extremal rays on $\overline{M}_{0,n}$", pp. 115–130 in *Handbook of moduli: Vol. II*, edited by G. Farkas and I. Morrison, Adv. Lect. Math. (ALM) **25**, International Press, Somerville, MA, 2013. MR Zbl

[Ueno 2008] K. Ueno, *Conformal field theory with gauge symmetry*, Fields Institute Monographs **24**, American Mathematical Society, Providence, RI, 2008. MR Zbl

RECEIVED: 16 Feb 2014 REVISED: 23 Jun 2018 ACCEPTED: 2 Aug 2018

DAVID SWINARSKI:

dswinarski@fordham.edu

Department of Mathematics, Fordham University, New York, United States

JOURNAL OF SOFTWARE FOR ALGEBRA AND GEOMETRY vol 8, no 1, 2018

1948-7916(2018)8:1;1-F