

```

gap> g:= SymmetricGroup( 4 )
Sym( [ 1 .. 4 ] )
gap> tbl:= CharacterTable( g ); HasIrr( tbl );
15 : betti(t,Weights=>{1,0})
false
0 1 2 3 4
o5 = total: 1 4 13 14 4
0: 1 . . . .
1: . 2 2 4 2
2: . 2 5 6 .
3: . . 4 . 2
4: . . . 4 .
5: . . 2 . .
gap> tblmod2:= CharacterTable( tbl, 2 );
BrauerTable( Sym( [ 1 .. 4 ] ), 2 )
gap> tblmod2 = CharacterTable( tbl, 2 );
true
gap> tblmod2 = BrauerTable( tbl, 2 );
true
gap> tblmod2 = BrauerTable( tbl, 2 );
true
o5 : BrauerTable( "M", 2 )
16 : betti(t,Weights=>{0,1})
0 1 2 3 4
o6 = total: 1 4 13 14 4
0: 1 . . . .
1: . 2 . . 2
2: . 2 . . 2
3: . 4 . . 2
4: . . . 4 .
5: . . 2 . .
gap> libtbl:= CharacterTable( "M" );
CharacterTable( "M" )
gap> CharacterTableRegular( libtbl, 2 );
BrauerTable( "M", 2 )
gap> BrauerTable( libtbl, 2 );
fail
ring r1 = 32003,(x,y,z),ds;
int a,b,c,t=11,5,3,0;
poly f = x^a+y^b+z^(3*c)+x^(c+2)*y^(c-1)+x^(c-2)*y^c*(y^2+t*x)^2;
gap> CharacterTable( "Symmetric", 4 );
CharacterTable( "Sym(4)" )
gap> ComputedBrauerTables( tbl );
[ , BrauerTable( Sym( [ 1 .. 4 ] ), 2 ) ]
option(noprot);
timer=1;
ring r2 = 32003,(x,y,z),dp;
poly f=imap(r1,f);
ideal j=jacob(f);
vdim(std(j));
==> 536
vdim(std(j+f));
==> 195
timer=0; // reset timer
o6 : BettiTally
17 : t1 = betti(t,Weights=>{1,1})
o7 = total: 0 1 2 3 4
1 4 13 14 4
0: 1 . . . .
1: . . . . .
2: . . . . .
3: . 2 . . .
4: . . . . .
5: . 2 . . .
6: . . 1 . .
7: . . 8 6 .
8: . . 4 8 4
o7 : BettiTally
18 : peek t1
o8 = BettiTally{(0, {0, 0}, 0) => 1 }
(1, {2, 2}, 4) => 2
(1, {3, 3}, 6) => 2
(2, {3, 7}, 10) => 2
(2, {4, 4}, 8) => 1
(2, {4, 5}, 9) => 4
(2, {5, 4}, 9) => 4
(2, {7, 3}, 10) => 2
(3, {4, 7}, 11) => 4
(3, {5, 5}, 11) => 4
(3, {7, 4}, 11) => 4
(4, {5, 7}, 12) => 2
(4, {7, 5}, 12) => 2

```

# Journal of Software for Algebra and Geometry

## Matroids: a Macaulay2 package

JUSTIN CHEN

# Matroids: a Macaulay2 package

JUSTIN CHEN

**ABSTRACT:** We give an overview of the Macaulay2 package `Matroids.m2`, which introduces functionality to create and compute with matroids into Macaulay2. Examples highlighting the use of many functions in the package are provided, including applications of matroids to other areas.

**INTRODUCTION.** A matroid is a combinatorial object which abstracts the notions of (linear algebraic, graph-theoretic) independence. Since their introduction by Whitney [1935], matroids have found diverse applications in combinatorics, graph theory, optimization, and algebraic geometry, in addition to being studied as interesting objects in their own right.

We describe here the Macaulay2 package `Matroids.m2`, which is available at <https://github.com/jchen419/Matroids-M2>. For the reader already familiar with matroids, it provides capabilities to form matroids from a matrix, graph, or ideal; convert between various representations of matroids; create and detect existence of minors; compute Tutte polynomials and Chow rings; as well as applications of matroids to polyhedral and algebraic geometry, commutative algebra, optimization, and even group theory. Each will in turn be illustrated with examples. Virtually all notation and results mentioned below can be found in [Oxley 2011].

One striking feature of matroids is the multitude of distinct ways to define them. This variety of equivalent — or *cryptomorphic* — ways to characterize matroids is a great strength of matroid theory, and one of the reasons for its ubiquity. From the perspective of this package, the key definition is via bases:

**Definition.** Let  $E$  be a finite set, and  $\emptyset \neq \mathcal{B} \subseteq 2^E$  a set of subsets of  $E$ . The pair  $(E, \mathcal{B})$  is a *matroid* if for any  $B_1, B_2 \in \mathcal{B}$  and  $b_1 \in B_1 \setminus B_2$ , there exists  $b_2 \in B_2 \setminus B_1$  with  $B_1 \setminus \{b_1\} \cup \{b_2\} \in \mathcal{B}$ .

The set  $E$  is called the *ground set* of the matroid  $M = (E, \mathcal{B})$ , and  $\mathcal{B}$  is the set of *bases* of  $M$ . All bases have the same cardinality, called the *rank* of  $M$ . Any subset of a basis is an *independent set*. A subset of  $E$  that is not independent is *dependent*.

---

*MSC2010:* primary 05-04, 05B35; secondary 05C31, 52B40.

*Keywords:* matroids, circuits, Tutte polynomial.

`Matroids.m2` version 0.9.7

The minimal (with respect to inclusion) dependent sets are *circuits*. It is easy to see that any of bases, independent sets, dependent sets, and circuits determines the others.

As any subset of an independent set is independent, the set of independent sets of a matroid forms a simplicial complex on  $E$ , called the *independence complex* of  $M$ , denoted by  $\Delta_M$ . Via Stanley–Reisner theory,  $\Delta_M$  corresponds to a squarefree monomial ideal  $I_{\Delta_M} := \langle \prod_{i \in C} x_i \mid C \text{ circuit} \rangle$ , inside a polynomial ring  $k[x_i \mid i \in E]$  (since faces of  $\Delta_M$  are independent sets, the minimal nonfaces are precisely the minimal dependent sets, i.e., circuits). We call  $I_{\Delta_M}$  the (circuit) ideal of  $M$ : internally, many algorithms in this package work directly with this ideal, to exploit Macaulay2's facility with monomial ideals.

**A FIRST EXAMPLE.** The most basic way to create a matroid is by specifying the ground set and list of bases:

```
i1 : needsPackage "Matroids";
i2 : M = matroid({a,b,c,d},{a,b},{a,c})
o2 = a matroid of rank 2 on 4 elements
o2 : Matroid
```

This creates a matroid of rank 2 on the ground set  $\{a, b, c, d\}$  with two bases. We can peek at the matroid to see more of its internal structure:

```
i3 : peek M
o3 = Matroid{bases => {set {0, 1}, set {0, 2}}}
      cache => CacheTable{...2...}
      groundSet => set {0, 1, 2, 3}
      rank => 2
```

Two things should be noticed: first, `groundSet` is a set of integers  $\{0, \dots, 3\}$  (instead of the given list  $\{a, b, c, d\}$ ). Second, the bases consist of a list of subsets of `groundSet`. This convention is by design: internally, the ground set is always identified with the set  $\{0, \dots, |E| - 1\}$ , and all sets associated to the structure of the matroid are subsets of the ground set. One should think of the integers in `groundSet` as *indices* of the actual elements, so 0 is the index of the first element (in this case  $a$ ), 1 is the index of the second element, etc.

The actual elements of the user-inputted ground set are not lost though; they have been cached in the `CacheTable`, and can be accessed by using indices as subscripts on  $M$ , or all at once with an asterisk:

```
i4 : (M_3, M_{0,1}, M_{set{1,2}}, M_*)
o4 = (d, {a, b}, {b, c}, {a, b, c, d})
```

So far, no attempt has been made to check that  $M$  is actually a matroid. We verify this now using the method `isWellDefined` (which internally checks the circuit elimination axiom), and also give a nonexample.

```
i5 : (isWellDefined M, isWellDefined matroid({a,b,c,d},{a,b},{c,d}))
o5 = (true, false)
```

We can obtain plenty of matroid-theoretic information for this example. Recall:

**Definition.** A *loop* in  $M$  is a 1-element circuit, and a *coloop* in  $M$  is an element contained in every basis. For  $A \subseteq E$ , the *rank* of  $A$  is the size of the largest independent subset of  $A$ , and the *closure* of  $A$  is  $\bar{A} := \{x \in E \mid \text{rank}(A) = \text{rank}(A \cup \{x\})\}$ . A *flat* of  $M$  is a closed subset, i.e.,  $A = \bar{A}$ . A *hyperplane* of  $M$  is a flat of rank equal to  $\text{rank } M - 1$ .

```
i6 : (rank M, rank(M, set{0,3}))
o6 = (2, 1)
i7 : (circuits M, independentSets(M, 1))
o7 = ({set {1, 2}, set {3}}, {set {0}, set {1}, set {2}})
i8 : (loops M, coloops M, closure(M, set{2,3}), hyperplanes M)
o8 = ({3}, {0}, set {1, 2, 3}, {set {0, 3}, set {1, 2, 3}})
i9 : flats M -- sorted by increasing size
o9 = {set {3}, set {0, 3}, set {1, 2, 3}, set {0, 1, 2, 3}}
i10 : fVector M -- number of flats of rank i, for 0 <= i <= rank M
o10 = HashTable{0 => 1}
          1 => 2
          2 => 1
```

CONSTRUCTING TYPES OF MATROIDS. The simplest family of matroids is the family of *uniform* matroids, where the set of bases equals all subsets of a fixed size:

```
i11 : U = uniformMatroid(2,4); bases U
o12 = {set {0, 1}, set {0, 2}, set {1, 2}, set {0, 3}, set {1, 3}, set {2, 3}}
```

Another family of fundamental importance is the class of *linear* matroids, which arise naturally from a matrix. The columns of the matrix form the ground set, and a set of column vectors is declared independent if they are linearly independent in the vector space spanned by the columns.

```
i13 : A = matrix{{0,4,-1,6},{0,2/3,7,1}}; MA = matroid A; representationOf MA
o15 = | 0 4   -1 6 |
      | 0 2/3 7  1 |
```

An abstract matroid  $M$  is called *representable* or *realizable* over a field  $k$  if  $M$  is *isomorphic* to a linear matroid over  $k$ , where an *isomorphism* of matroids is a bijection between ground sets that induces a bijection on bases. We verify that the matroid  $M$  we started with is isomorphic to  $MA$ , hence is representable over  $\mathbb{Q}$ :

```
i16 : areIsomorphic(M, MA)
o16 = true
```

A matroid can also be constructed by specifying its circuit ideal, which we do for the same  $M$  above. Here two matroids are considered equal if they have the same set of bases and same size ground sets; or, equivalently, the identity permutation is an isomorphism between them.

```
i17 : R = QQ[x,y,z,w]; MI = matroid ideal(y*z, w)
o18 = a matroid of rank 2 on 4 elements
i19 : M == MI
o19 = true
```

An important class of representable matroids (over any field) is the class of *graphic* matroids, derived from a graph. If  $G$  is an (undirected) graph, then the graphic matroid  $M(G)$  has ground set equal to the edge set of  $G$ , and circuits given by cycles in  $G$ , including loops and parallel edges.

```
i20 : K5 = completeGraph 5; M5 = matroid K5
o21 = a matroid of rank 4 on 10 elements
i22 : #bases M5 -- == n^(n-2) for M(K_n), by Cayley's theorem
o22 = 125
```

In this package, the graphic matroid is created by specifying circuits. This can be done for an abstract matroid as well, using the optional argument `EntryMode => "circuits"` in the constructor function. Regardless of the value of `EntryMode`, the bases are automatically computed upon creation. We recreate the matroid  $M$  from before, by specifying its circuits (note the similarity with specifying the circuit ideal):

```
i23 : M == matroid({a,b,c,d},{b,c},{d}}, EntryMode => "circuits")
o23 = true
```

Certain common matroids are close to uniform, in the sense that relatively few subsets of size  $\text{rank } M$  are dependent, so the set of *nonbases* (= dependent sets of size  $\text{rank } M$ ) can also be specified:

```
i24 : nb = {{0,2,4},{1,3,4},{1,2,5},{0,3,5},{0,1,6},{2,3,6},{4,5,6}}/set;
i25 : F7 = matroid(toList(0..6), nb, EntryMode => "nonbases")
o25 : a matroid of rank 3 on 7 elements
i26 : (#bases F7, #circuits F7)
o26 = (28, 14)
```

A few specific matroids of theoretical importance are also built-in. Currently these are  $F_7$ ,  $F_7^-$ ,  $V_8$ ,  $V_8^+$ ,  $AG(3, 2)$ ,  $R_{10}$ , and the Pappus and non-Pappus matroids. A library of all matroids on up to eight elements is included as well:

```
i27 : F7 == specificMatroid "fano"
o27 = true
i28 : L7 = allMatroids 7 -- non-isomorphic matroids on 7 elements
o28 = {a matroid of rank 0 on 7 elements, a matroid of rank 1 on 7 elements, ...
i29 : (#L7, #flatten apply(6, allMatroids))
o29 = (306, 70)
```

One can also construct a new matroid from smaller ones by taking *direct sums*: if  $M_1 = (E_1, \mathcal{B}_1)$ ,  $M_2 = (E_2, \mathcal{B}_2)$  are matroids, then their direct sum is

$$M_1 \oplus M_2 := (E_1 \sqcup E_2, \{B_1 \sqcup B_2 \mid B_1 \in \mathcal{B}_1, B_2 \in \mathcal{B}_2\}).$$

A matroid that cannot be written as a direct sum of nonempty matroids is called *connected*. Every matroid is a direct sum of connected matroids, its *connected components*, which are unique up to rearrangement:

```
i30 : S = U ++ matroid completeGraph 3
o30 = a matroid of rank 4 on 7 elements
```

```

i31 : C = components S
o31 = {a matroid of rank 2 on 4 elements, a matroid of rank 2 on 3 elements}
i32 : S == C#0 ++ C#1 and C#0 == U and C#1 == matroid completeGraph 3
o32 = true

```

**DUALITY AND MINORS.** One of the most important features of matroid theory is the existence of a duality. It is straightforward to check that if  $M = (E, \mathcal{B})$  is a matroid, then  $\{E \setminus B \mid B \in \mathcal{B}\}$  is the set of bases of a matroid on  $E$ , called the *dual matroid* of  $M$ , denoted by  $M^*$ .

```

i33 : D = dual M; (bases M, bases D)
o34 = ({set {0, 1}, set {0, 2}}, {set {2, 3}, set {1, 3}})
i35 : M == dual D
o35 = true

```

Virtually any matroid-theoretic property or operation can be enriched by considering its dual version—for instance, loops of  $M^*$  are coloops of  $M$ , and circuits of  $M^*$  are complements of hyperplanes of  $M$  (this is in fact how the method `hyperplanes` works). Another operation is deletion, which dualizes to contraction:

**Definition.** Let  $M = (E, \mathcal{B})$  be a matroid, and  $S \subseteq E$ . The *restriction* of  $M$  to  $S$ , denoted  $M|_S$ , is the matroid on  $S$  with bases  $\{B \cap S \mid B \in \mathcal{B}, |B \cap S| = \text{rank } S\}$ . The *deletion* of  $S$ , denoted  $M \setminus S$ , is the restriction of  $M$  to  $E \setminus S$ . The *contraction* of  $M$  by  $S$ , denoted  $M/S$ , is defined as  $(M^* \setminus S)^*$ .

```

i36 : N1 = M \ set{3}; (N1_*, bases N1)
o37 = ({a, b, c}, {set {0, 1}, set {0, 2}})
i38 : N2 = M / set{1}; (N2_*, bases N2)
o39 = ({a, c, d}, {set {0}})

```

A *minor* of  $M$  is any matroid which can be obtained from  $M$  by a sequence of deletions and contractions. It is a fact that any minor of  $M$  is of the form  $(M/X) \setminus Y$  for disjoint subsets  $X, Y \subseteq E$ .

```

i40 : minorM5 = minor(M5, set{9}, set{3,5,8}) -- contracts {9}, then deletes {3,5,8}
o40 = a matroid of rank 3 on 6 elements
i41 : (minorM5_*, #bases minorM5)
o41 = ({set {0, 1}, set {0, 2}, set {0, 3}, set {1, 2}, set {1, 4}, set {2, 3}}, 16)

```

Minors can be used to describe many important classes of matroids. For example, a class  $\mathcal{M}$  of matroids is said to be *minor-closed* if every minor of a matroid in  $\mathcal{M}$  is again in  $\mathcal{M}$ . The classes of uniform,  $k$ -representable (for any field  $k$ ), and graphic matroids are all minor-closed. Various classes of matroids can be characterized by their *forbidden* or *excluded* minors: namely the matroids not in the class, but with every proper minor in the class.

**Theorem 1 (Tutte 1958a; 1958b; 1959).** *Let  $M$  be a matroid. We denote by  $U_{2,4}$  the uniform matroid of rank 2 on 4 elements, and by  $F_7$  the Fano matroid.*

- (i)  $M$  is binary (= representable over  $\mathbb{F}_2$ ) if and only if  $M$  has no  $U_{2,4}$  minor (i.e., no minor of  $M$  is isomorphic to  $U_{2,4}$ ).

- (ii)  $M$  is regular (= representable over any field) if and only if  $M$  has no  $U_{2,4}$ ,  $F_7$ , or  $F_7^*$  minor.
- (iii)  $M$  is graphic if and only if  $M$  has no  $U_{2,4}$ ,  $F_7$ ,  $F_7^*$ ,  $M(K_5)^*$ , or  $M(K_{3,3})^*$  minor.

We illustrate this by verifying that  $M(K_5)$  is regular (alternatively, note that for any graph  $G$ , the signed incidence matrix of any orientation of  $G$  represents  $M(G)$  over any field):

```
i42 : any({U, F7, dual F7}, forbidden -> hasMinor(M5, forbidden))
o42 = false
```

Every minor of  $M$  is in fact of the form  $(M/I) \setminus I^*$ , where  $I, I^*$  are disjoint,  $I$  is independent, and  $I^*$  is *coindependent* (= independent in  $M^*$ ). Such a minor has rank equal to that of  $M/I$ , which is equal to  $\text{rank } M - |I|$ . Thus checking existence of a minor  $N$  in  $M$  can be realized as a two-step process, where the first step contracts independent sets of  $M$  of a fixed size down to the rank of  $N$ , and the second step deletes coindependent sets down to the size of  $N$ .

```
i43 : M4 = matroid completeGraph 4; hasMinor(M5, M4)
o44 = true
i45 : minorM5 == M4
o45 = true
```

Finally, the *Tutte polynomial*  $T_M(x, y)$  of a matroid is an invariant which is universal with respect to satisfying a *deletion-contraction recurrence*. It is a bivariate polynomial over  $\mathbb{Z}$  which can be defined by the relation

$$T_M(x, y) = T_{M \setminus e}(x, y) + T_{M/e}(x, y), \quad e \in E \text{ not a loop or coloop,}$$

with the initial condition  $T_M(x, y) = x^a y^b$  if  $M$  consists of  $a$  coloops and  $b$  loops. Any numerical invariant of matroids which satisfies a (weighted) deletion-contraction recurrence is an evaluation of the Tutte polynomial, up to a scale factor. For instance, the number of bases is equal to  $T_M(1, 1)$ :

```
i46 : tuttePolynomial M5
o46 = y^6 + 4y^5 + x^4 + 5xy^3 + 10y^4 + 6x^3 + 10xy^2 + 15xy^2 + 15y^3 + 11x^2 + 20xy + 15y^2...
i47 : tutteEvaluate(M5, 1, 1)
o47 = 125
```

For graphic matroids, the Tutte polynomial contains a wealth of information about the graph; e.g., the Tutte polynomial specializes to the chromatic polynomial. Even evaluations at specific points contain nontrivial information: e.g.,  $T_{M(G)}(2, 1)$  counts the number of spanning forests in  $G$ , and  $T_{M(G)}(2, 0)$  counts the number of acyclic orientations of  $G$ .

```
i48 : (tutteEvaluate(M5, 2, 1), tutteEvaluate(M5, 2, 0), factor chromaticPolynomial K5)
o48 = (291, 120, (x)(x - 4)(x - 3)(x - 2)(x - 1))
```

**CONNECTIONS.** We now present some connections of matroids to other areas of mathematics. First, polyhedral geometry: let  $M = ([n], \mathcal{B})$  be a matroid on  $\{1, \dots, n\}$ . In Euclidean space  $\mathbb{R}^n$  with standard basis  $\{e_1, \dots, e_n\}$ , define the matroid polytope  $P_M$  by taking the convex hull of the indicator vectors of the bases of  $M$ :

$$P_M := \text{conv}\left(\sum_{i \in B} e_i \mid B \in \mathcal{B}\right).$$

The matroid polytope can be created as follows:

```
i49 : needsPackage "Polyhedra"; P = convexHull basisIndicatorMatrix M4
o50 = {ambient dimension => 6
      }
      dimension of lineality space => 0
      dimension of polyhedron => 5
      number of facets => 16
      number of rays => 0
      number of vertices => 16
o50 : Polyhedron
```

A theorem of Gelfand, Goresky, MacPherson, and Serganova [Gelfand et al. 1987] classifies the subsets  $B \subseteq 2^{[n]}$  which are the bases of a matroid on  $[n]$  in terms of the polytope  $P_M$ .

Next is optimization: let  $E$  be a finite set, and  $\mathcal{I} \subseteq 2^E$  a set of subsets that is downward closed: if  $X \in \mathcal{I}$  and  $Y \subseteq X$ , then  $Y \in \mathcal{I}$ . Let  $w$  be a weight function on  $E$ , i.e., a function  $w : E \rightarrow \mathbb{R}$ , extended to  $w : 2^E \rightarrow \mathbb{R}$  by setting  $w(X) := \sum_{x \in X} w(x)$ . Consider the optimization problem (\*) of finding a maximal member of  $\mathcal{I}$  of maximum weight, with respect to  $w$ . One attempt to solve (\*) is to apply the greedy algorithm: namely, after having already selected elements  $\{x_1, \dots, x_i\}$ , choose an element  $x_{i+1} \in E$  of maximum weight such that  $\{x_1, \dots, x_i, x_{i+1}\} \in \mathcal{I}$ , and repeat. It turns out that the greedy algorithm will work if and only if  $\mathcal{I}$  is the set of independent sets of a matroid:

**Theorem 2 [Borůvka 1926].** *Let  $E$  be a finite set, and  $\mathcal{I} \subseteq 2^E$ . Then  $\mathcal{I}$  is the set of independent sets of a matroid on  $E$  if and only if  $\mathcal{I}$  is downward closed and for all weight functions  $w : E \rightarrow \mathbb{R}$ , the greedy algorithm successfully solves (\*).*

A solution to (\*) provided by the greedy algorithm can be obtained using the method `maxWeightBasis` (the weight function  $w$  is specified by its list of values on  $E$ ):

```
i51 : w = {0, log(2), 4/3, 1, -4, 2, pi_RR}; maxWeightBasis(F7, w)
o52 = set {3, 5, 6}
```

Another application to optimization comes from the operation of *matroid union*: if  $M_1, M_2$  are matroids with independent sets  $\mathcal{I}_1, \mathcal{I}_2$ , then the independent sets of the union are of the form  $I_1 \cup I_2$ , where  $I_1 \in \mathcal{I}_1, I_2 \in \mathcal{I}_2$  (and thus coincides with the direct sum if the ground sets are disjoint).

```
i53 : matroid({a,b,c,d}, {{a},{b},{c}}) + matroid({a,b,c,d}, {{b},{c},{d}}) == U
o53 : true
i54 : F7 + F7 == uniformMatroid(6, 7)
o54 : true
```

Matroid union is an important operation in combinatorial optimization, and is closely related to transversal and matching problems: a matroid is *transversal* if and only if it is a union of rank 1 matroids, and *gammoids* (a class of matroids defined from vertex paths in directed graphs) are the minor-closure of the transversal matroids.

One can also find connections to group theory via the method `getIsos`, which computes all isomorphisms between two matroids. Many interesting groups can be realized as automorphism groups of small matroids:

```
i55 : aut = getIsos(F7, F7)
o55 : {{0, 1, 2, 3, 4, 5, 6}, {1, 0, 2, 3, 4, 6, 5}, {0, 2, 1, 3, 5, 4, 6}, {2, 0, 1, ...
i56 : #aut
o56 : 168
```

The above output is an explicit permutation representation of  $\text{Aut}(\mathbb{P}_{\mathbb{F}_2}^2) = \text{PGL}(3, \mathbb{F}_2)$  as a subgroup of  $S_7$ . For a larger example, the automorphism group of the Steiner system  $S(5, 6, 12)$  is the Mathieu group  $M_{12}$ , a sporadic simple group of order  $95040 = 2^6 \cdot 3^3 \cdot 5 \cdot 11$ . This in turn is also equal to the automorphism group of the realizable matroid associated to a particular  $6 \times 12$  matrix over  $\mathbb{F}_3$  ([Oxley 2011], p. 367), and a high-performance computing cluster took just under 2 hours to compute the entire permutation representation of this group inside  $S_{12}$ .

For an application to commutative algebra: matroids are closely related to the Cohen–Macaulay property, for symbolic powers of squarefree monomial ideals. Indeed, from [Terai and Trung 2012] we know that if  $I$  is a squarefree monomial ideal, then  $I$  is the circuit ideal of a matroid if and only if every symbolic power  $I^{(n)}$  is Cohen–Macaulay, for  $n \geq 1$  (in fact, this is equivalent to requiring just  $I^{(3)}$  to be Cohen–Macaulay). As one can quickly check whether an ideal is the ideal of a matroid, this can give a quick proof that a particular symbolic power is Cohen–Macaulay:

```
i57 : M6 = matroid completeGraph 6; L = (irreducibleDecomposition ideal M6)/(P -> P^3);
i59 : try ( alarm 10; I3 = intersect L; ) -- doesn't finish in 10 seconds
i60 : time isWellDefined M6
      -- used 0.359306 seconds
o60 : true
```

Last but not least is algebraic geometry; in particular the emerging field of combinatorial Hodge theory. For a matroid  $M$  on ground set  $E$  with no loops, one can define a Chow ring associated to  $M$ : for a field  $k$ , set

$$R := k[x_F \mid F \text{ proper, nonempty flat}]/(I_1 + I_2),$$

$$I_1 := \left( \sum_{i_1 \in F} x_F - \sum_{i_2 \in F} x_F \mid i_1, i_2 \in E \text{ distinct} \right),$$

$$I_2 := (x_F x_{F'} \mid F, F' \text{ incomparable}),$$

where  $F, F'$  run over all nonempty proper flats of  $M$ . Then  $R$  is a standard graded Artinian  $k$ -algebra of Castelnuovo–Mumford regularity  $r := \text{rank } M - 1$ . A result of Adiprasito, Katz, and Huh [Adiprasito et al. 2018] states that  $R$  is a Poincaré duality algebra (in particular, is Gorenstein) and has the strong Lefschetz property: for general  $l \in R_1$  and  $j \leq r/2$ , multiplication by  $l^{r-2j}$  is an isomorphism  $R_j \xrightarrow{\sim} R_{r-j}$ . We illustrate the Gorenstein property for the Vamos matroid (which is a smallest matroid not realizable over any field), and conclude by computing the dual socle generator or *volume polynomial* (which generates the Macaulay inverse system of  $R$ ) for  $M(K_4)$ :

```

i61 : V = specificMatroid("vamos"); (rank V, #V.groundSet, #bases V, #flats V)
o62 = (4, 8, 65, 79)
i63 : I = idealChowRing V; apply(0..<rank V, i -> hilbertFunction(i, I))
o63 : Ideal of QQ[x_{7}, x_{6}, x_{5}, x_{4}, x_{3}, x_{0}, x_{2}, x_{1}, x_{6,7}, x_{5,7}, ...
o64 = (1, 70, 70, 1)
i65 : cogeneratorChowRing M4
o65 = 2t^2_{5} + 2t^2_{4} + 2t^2_{3} + 2t^2_{2} + 2t^2_{1} + 2t^2_{0} - 2t t_{5} - 2t t_{0,5} + ...

```

ACKNOWLEDGEMENTS. This project was partially supported by grant DMS-1001867 from the NSF. The author is grateful to June Huh for explaining the connection to the Chow ring of a matroid. The author would also like to thank David Eisenbud and Daniel Grayson for advice with Macaulay2, Joe Kileel for enlightening discussions, Aaron Dall for software testing, Chris Eur for suggesting many valuable improvements, and the referee for helpful comments.

SUPPLEMENT. The [online supplement](#) contains version 0.9.7 of `Matroids.m2`.

## REFERENCES.

- [Adiprasito et al. 2018] K. Adiprasito, J. Huh, and E. Katz, “Hodge theory for combinatorial geometries”, *Ann. of Math.* (2) **188**:2 (2018), 381–452. [MR](#) [Zbl](#)
- [Borůvka 1926] O. Borůvka, “O jistém problému minimálním”, *Práce moravské přírodovědecké společnosti* **3**:3 (1926), 37–58.
- [Gelfand et al. 1987] I. M. Gelfand, R. M. Goresky, R. D. MacPherson, and V. V. Serganova, “Combinatorial geometries, convex polyhedra, and Schubert cells”, *Adv. in Math.* **63**:3 (1987), 301–316. [MR](#)
- [Oxley 2011] J. Oxley, *Matroid theory*, 2nd ed., Oxford Graduate Texts in Mathematics **21**, Oxford University Press, 2011. [MR](#) [Zbl](#)
- [Terai and Trung 2012] N. Terai and N. V. Trung, “Cohen–Macaulayness of large powers of Stanley–Reisner ideals”, *Adv. Math.* **229**:2 (2012), 711–730. [MR](#) [Zbl](#)
- [Tutte 1958a] W. T. Tutte, “A homotopy theorem for matroids, I”, *Trans. Amer. Math. Soc.* **88** (1958), 144–160. [MR](#) [Zbl](#)
- [Tutte 1958b] W. T. Tutte, “A homotopy theorem for matroids, II”, *Trans. Amer. Math. Soc.* **88** (1958), 161–174. [MR](#) [Zbl](#)
- [Tutte 1959] W. T. Tutte, “Matroids and graphs”, *Trans. Amer. Math. Soc.* **90** (1959), 527–552. [MR](#) [Zbl](#)
- [Whitney 1935] H. Whitney, “On the abstract properties of linear dependence”, *Amer. J. Math.* **57**:3 (1935), 509–533. [MR](#) [Zbl](#)

RECEIVED: 6 Sep 2015

REVISED: 26 Sep 2018

ACCEPTED: 27 Sep 2018

JUSTIN CHEN: Department of Mathematics, Georgia Institute of Technology, Atlanta, GA, United States

[jchen646@gatech.edu](mailto:jchen646@gatech.edu)





<i>Strongly stable ideals and Hilbert polynomials</i>	1
Davide Alberelli and Paolo Lella	
<i>DiffAlg: a Differential algebra package</i>	11
Manuel Dubinsky, César Massri, Ariel Molinuevo and Federico Quallbrunn	
<i>Matroids: a Macaulay2 package</i>	19
Justin Chen	
<i>Computing quasidegrees of A-graded modules</i>	29
Roberto Barrera	
<i>An algorithm for enumerating difference sets</i>	35
Dylan Peifer	
<i>Hyperplane arrangements in CoCoA</i>	43
Elisa Palezzato and Michele Torielli	
<i>Numerical implicitization</i>	55
Justin Chen and Joe Kileel	
<i>Random Monomial Ideals: a Macaulay2 package</i>	65
Sonja Petrović, Despina Stasi and Dane Wilburne	
<i>Calculations involving symbolic powers</i>	71
Ben Drabkin, Eloísa Grifo, Alexandra Seceleanu and Branden Stone	
<i>The gfanlib interface in Singular and its applications</i>	81
Anders Jensen, Yue Ren and Hans Schönemann	