Winter 1999

Using character varieties: Presentations, invariants, divisibility and determinants

Jeffrey Allan Hall
University of New Hampshire, Durham

Using character varieties: presentations, invariants, divisibility and determinants

BY

Jeffrey Allan Hall

B.A.,Western Connecticut State University (1993)
M.S.,University of New Hampshire (1994)

Submitted to the University of New Hampshire
in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

in

Mathematics

December 1999

UMI Number: 9953419

This dissertation has been examined and approved.

David Feldman

Director. David Feldman
Associate Professor of Mathematics

Edward Hinson

Edward Hinson
Associate Professor of Mathematics

Kenneth Appel

Kenneth Appel
Professor of Mathematics

Homer Bechtell

Homer Bechtell
Professor of Mathematics

Don Hadwin

Donald Hadwin
Professor of Mathematics

10/9/99

Date

This thesis is dedicated to my grandmother, Margaret Hall, and to the memory of my grandmother, Ethel Smith.

Contents

List of Figures

List of Tables

ABSTRACT

Using character varieties: presentations, invariants, divisibility and determinants

by

Jeffrey Allan Hall

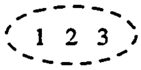
University of New Hampshire, December, 1999

If G is a finitely generated group, then the set of all characters from G into a linear algebraic group is a useful (but not complete) invariant of G. In this thesis, we present some new methods for computing with the variety of $SL_2\mathbb{C}$-characters of a finitely presented group. We review the theory of Fricke characters, and introduce a notion of presentation simplicity which uses these results. With this definition, we give a set of GAP routines which facilitate the simplification of group presentations. We provide an explicit canonical basis for an invariant ring associated with a symmetrically presented group's character variety. Then, turning to the divisibility properties of trace polynomials, we examine a sequence of polynomials $r_n(a)$ governing the weak divisibility of a family of shifted linear recurrence sequences. We prove a discriminant/determinant identity about certain factors of $r_n(a)$ in an intriguing manner. Finally, we indicate how ordinary generating functions may be used to discover linear factors of sequences of discriminants.

Other novelties include an unusual binomial identity, which we use to prove a well-known formula for traces; the use of a generating function to find the inverse of a map $x^n \mapsto f_n(x)$; and a brief exploration of the relationship between finding the

determinants of a parametrized family of matrices and the Smith Normal Forms of the sequence.

Table 0.1 <u>Notation used in the text</u>

χ_ρ	the character afforded by the representation ρ
$\mathbb{C}$ ($\mathbb{Q}$, $\mathbb{Z}$)	the complex numbers (rationals, integers)
$\mathbb{N}$	the natural numbers $\mathbb{N} = \{1, 2, 3. \cdots\}$
F_n	the free group on n letters
$X(G)$	the character variety of a group G
$R(G)$	the coordinate ring of the variety $X(G)$
$L = (P)SL_2\mathbb{C}$	the (projective) special linear group
$H < G$, $H \lhd G$	H is a subgroup of G, H is a normal subgroup of G
$[G : H]$	the index of $H < G$ in G
$[\alpha, \beta]$	for elements $\alpha\,\beta$ of a poset, the interval between α and β, including α and β
$< S >$	the group or algebra generated by the set S, or freely freely generated by S
$[a.b]$	$a^{-1}b^{-1}ab$
G'	the first derived subgroup of G, $G' =< [G, G\} >$
A^B	$B^{-1}AB$
$I < R$, $I \lhd R$	I is a subring of R, I is an ideal of R
(S)	the ideal generated by a set S
$< S \vert R >$	the group freely presented by generators S, with relations R
$\mathbb{V}(I)$	the variety which is defined by an ideal I
$\sqrt{I}$	the radical of an ideal I
$\mathbb{C}[X]^G$	the algebra of invariants of matrix group G acting on the vector space $\mathbb{C}^{\vert X \vert}$
x_i^M	action of matrix M on variable x_i: $x_i^{[a_{ij}]} = \sum_j a_{ji}x_j$
$\prec$	an admissible term order
$in_\prec S$	the set of initial terms with respect to $\prec$ in a set S
$LT_\prec f$	the monic leading term of f with respect to $\prec$
$LC_\prec f$	the leading coefficient of f with respect to $\prec$
σ_n	the n-th elementary symmetric function
$*$	the Reynolds operator: $*f = \frac{1}{n!}\sum_{g \in G} f^g$
S_n	the permutation group on n letters $\{1, 2, \ldots, n\}$. We use the same notation for the group of $n \times n$ permutation matrices
$S_n^{(m)}$	the group S_n, acting on m—element sets
$dist p, q$	the distance between p and q
$S_n^{\{m\}}$	the group S_n, acting on sets with m or fewer elements
$a \vdash n$	a is a partition of integer n (e.g., $(3, 2, 2) \vdash 7$)
	an edge in a graph (an edge between 1 and 2)
	a hyperedge in a hypergraph (a hyperedge on vertices 1, 2, 3)

$\lfloor x \rfloor$	the floor of x (which is the largest integer smaller than x)
$\mathrm{Res}(f, g)$, $\Delta(f)$	the resultant of polynomials f and g; the discriminant of f
$B_t(z)$	the generalized binomial series with parameter t (see [79, chapter 5], where $B_t(z)$ was introduced)
$M\|_j^i$	the minor of matrix M obtained by deleting the i–th row and the j–th column
$P_n(j)$	the number of j–tableaux shape with n or fewer rows
$W(S)$	the "Whittemore variety" of a set of words
$\diamondsuit$	the end of a proof; q.e.d.

Chapter 1

An introduction to the ring of Fricke characters

Boone's construction of a group without solvable word problem in the 1950's was a development which would have delighted Bishop George Berkeley: a natural question about an algebraically important construction turned out to be algorithmically undecidable. There were two natural ways to explore this new territory: to construct new groups and semigroups with unsolvable word problem (respectively conjugacy problem, etc.), and to show that interesting groups and families of groups had solvable word problem (respectively conjugacy problem, etc.) The first approach has yielded a host of interesting (and often discouraging) undecidability results, the second has at least helped delineate the undecidable from the decidable in combinatorial group theory. Neither approach interests us here directly. Instead, our goal is to introduce and examine some algorithms which exploit properties of finitely generated groups, and of objects which are related to the study of representations of finitely presented groups, which are semi-decidable (i.e., if the property is true, then there is a procedure to verify it, whose running time may not be bounded.) These provide us with properties of elements in groups, weaker than the conjugacy problem, which are decidable for all finitely-presented groups.

In this thesis, a *representation* of a group G into a complex linear algebraic group L is a homomorphism $\rho : G \to L$. If G is a topological group, then we require that ρ be continuous. The *character* of G afforded by ρ is the function $\chi_\rho : G \to \mathbb{C}$ defined by $\chi_\rho(g) = \operatorname{tr} \rho(g)$. If L is one-dimensional, then we call the character afforded by ρ *linear*. Our focus will be largely on the representations of finitely presented groups G into $L = SL_2\mathbb{C}$, $PSL_2\mathbb{C}$, or $\mathbb{C}$. In this chapter, all representations and characters will be into $SL_2\mathbb{C}$.

G is said to be a finitely generated group if it is the homomorphic image of a finite-rank free group under some homomorphism $\xi : F_n \to G$. (If G is finitely generated, then there are in general many such homomorphisms.) Fix $\xi : F_n \to G$, a surjective homomorphism from a finite-rank free group to G. Now, any representation $\rho : G \to L$ determines a representation $\xi \circ \rho$ of F_n, and since F_n is free, this representation is defined by the images of the free generators of F_n. Thus, the representation ρ is uniquely determined by the images of a set of generators of G. The condition that ρ

is a homomorphism is the same as the condition that the diagram

$$\begin{array}{ccccc} & & \exists\xi' & & \\ & F_n & \longrightarrow & G' & \\ \xi & \downarrow & \searrow & \downarrow\tau & \\ & G & \longrightarrow & SL_2\mathbb{C} & \\ & & \rho & & \end{array}$$

commutes for any group G' of rank n and any choice of τ; thus, the statement that a set of n $(d \times d)$-matrices are the images $\rho(g_i)$ of a finite set $\{g_i\}$ of generators of G is may be restated as the statement that a set of polynomial relations on the d^2n coordinates of the matrices is satisfied. By the Hilbert basis theorem, we may take this set of polynomials to be finite. The variety defined by these polynomials is called the *representation variety* of G. (Here, and in the future, we refer to the zero set of any set of polynomials over a field to be a "variety." We will not require that the set be irreducible: this is typical in the literature. Some authors might use the term "algebraic set.") We denote by $X(G)$ the quotient of the representation variety, obtained by identifying representations with equal traces. We embed $X(G)$ into the image of the representation variety under the trace map $\mathrm{tr} : SL_2\mathbb{C} \to \mathbb{C}$. This manifold is actually a variety itself, called the *character variety* of G . It is an important invariant of the finitely generated group G.

$X(G)$, our main object of study, has a lesser role in the study of finitely presented groups than its distant cousin, the character table, plays in the study of finite groups. A finite group's irreducible characters have bounded degree; and, by Cayley's theorem, every finite group of order m has a faithful representation of degree m. Thus, knowledge of the representations up to degree m completely determines the structure of the finite group.

But finitely presented groups do not, in general, have solvable word problem. On the other hand, given any set of matrices, and any finite word in this set, we may decide whether the product is the identity matrix by multiplying them together. In other words, the word problem for finite-dimensional matrix groups is algorithmically solvable [93]. The correctness of Buchberger's algorithm now implies:

Proposition 1.1 *The following questions are algorithmically undecidable:*

a) *What is the smallest degree of a finite-dimensional faithful representation of* G*?*

b) *What is the smallest index* $[\mathsf{G} : \ker \rho]$ *among finite-dimensional representations* ρ*?*

c) *Does* G *have a finite-dimensional faithful representation?*

(For an explicit proof, see [93].) The finite-dimensional representation theory of a finitely generated group G is a compromise between our desire to know the structure of a finitely generated group and the reality of the word problem. Despite the unsolvability of the word problem, we may inquire into the behavior of homomorphic images of G in algebraic groups. In fact, although G may not have solvable word problem, its representations necessarily do.

We review the classical theory of $X(G)$.

Theorem 1.2 *Let* $G = < g_1, \ldots, g_n | R >$ *be a finitely presented group. Then:*

a) **(Vogt)** *Let* χ *be an* $SL_2\mathbb{C}$*-character of* G*. Let* $g \in G$*. Then* $\chi(g)$ *is determined by the numbers*

$$\{t_i = \chi g_i\} \cup \{t_{ij} = \chi g_i g_j | 1 \leq i < j \leq n\} \cup \cdots \cup \{t_{123\ldots n} = \chi g_1 g_2 \ldots g_n\}.$$

Indeed, more is true. The values of a character on the singletons, pairs, and triples in the above list suffice to determine the character on all of G*.*

Vogt's theorem lets us index the characters with points

$$(t_1, \ldots, t_n, t_{12}, \ldots, t_{(n-1)n}, t_{123}, \ldots, t_{(n-2)(n-1)n}).$$

(A representation, of course, is determined by its value on generators. Part (a) is a finiteness theorem: characters are uniquely determined by their values on generators, and on ordered products of two and three generators.)

b) **(Horowitz)** *The set of all possible characters is a variety, which is an invariant of the group* G*; i.e., any relation between elements of* $X = \{t_i, t_{ij}, t_{ijk}\}$ *is a polynomial. For example, if* $n = 2$*, then* $X(G)$ *is all of* $\mathbb{C}^3$*.*

Proof: (a) We remind the reader that we are working with 2×2 matrices only. Each of the following identities is a trivial calculation. (1.2 is also a disguised version of the Cayley-Hamilton theorem, but that doesn't concern us here.)

$$\begin{aligned}
\operatorname{tr}(ABC) &= \operatorname{tr} A \operatorname{tr}(BC) + \operatorname{tr} B \operatorname{tr}(AC) + \operatorname{tr} C \operatorname{tr}(AB) - \operatorname{tr} A \operatorname{tr} B \operatorname{tr} C \\
&\quad - \operatorname{tr}(ACB) && (1.1) \\
\operatorname{tr}(AB) &= (\operatorname{tr} A)(\operatorname{tr} B) - \operatorname{tr} A/B && (1.2) \\
\operatorname{tr}(A) &= \operatorname{tr}(A^{-1}) \\
&= \operatorname{tr}(A^{B}). \\
\operatorname{tr}(AB) &= \operatorname{tr}(BA) && (1.3)
\end{aligned}$$

$$\begin{aligned}
\operatorname{tr}(ABCD) &= \frac{1}{2}(\operatorname{tr} A \operatorname{tr}(BCD) + \operatorname{tr} B \operatorname{tr}(CDA) + \operatorname{tr} C \operatorname{tr}(DAB) && (1.4) \\
&\quad + \operatorname{tr} D \operatorname{tr}(CAB) + \operatorname{tr}(AB) \operatorname{tr}(CD) - \operatorname{tr}(AC) \operatorname{tr}(BD) \\
&\quad + \operatorname{tr}(AD) \operatorname{tr}(BC) - \operatorname{tr} A \operatorname{tr} B \operatorname{tr} CD - \operatorname{tr} C \operatorname{tr} D \operatorname{tr} AB \\
&\quad - \operatorname{tr} D \operatorname{tr} A \operatorname{tr} BC - \operatorname{tr} B \operatorname{tr} C \operatorname{tr} DA + \operatorname{tr} A \operatorname{tr} B \operatorname{tr} C \operatorname{tr} D)
\end{aligned}$$

(1.4 is known as "Vogt's relation," the relation 1.1 is known as "Fricke's lemma," and 1.2 is called the "fundamental trace relation.")

Given any word w with more than three generators, we may use 1.4 to express

$\operatorname{tr} w$ using shorter words. Using 1.2, we can express traces of words of length 2 or 3 in terms of traces of length 2 or 3 without using inverses of generators (e.g.,

$$\begin{aligned}
\operatorname{tr} ab^{-1} &= (\operatorname{tr} a)(\operatorname{tr} b^{-1}) - \operatorname{tr} ab \\
&= (\operatorname{tr} a)(\operatorname{tr} b) - \operatorname{tr} ab \\
\operatorname{tr} abc^{-1} &= (\operatorname{tr} ab)(\operatorname{tr} c) - \operatorname{tr} abc
\end{aligned}$$

etc.) By conjugation, traces of words that are mis-ordered cyclically can be re-ordered (e.g., $\operatorname{tr} bca = \operatorname{tr} abc$.) Finally, 1.1 lets us express traces of length 3 words which are mis-ordered non-cyclically.

(b) See [57] or [33].

We will now review in some detail the computational mechanisms with which one may manipulate trace polynomials. First, however, we introduce a somewhat unusual binomial identity.

Theorem 1.3 *For any positive integers* o, l,

$$\sum_{m} \binom{2o}{2m} \binom{m}{l} = 2^{2o-2l} \frac{o}{l!} \prod_{k=1}^{l-1} (2o - l - k) . \tag{1.5}$$

Proof: We proceed via the Wilf-Zeilberger method. [99] The claim of the theorem is readily verified for $o \in \{1,2,3\}$ and $l \in \{1,2,3,4,5,6\}$. We exhibit a recurrence relation which is satisfied by the sequence

$$f_o = \frac{\binom{2o}{2m}\binom{m}{l}}{2^{2o-2l}\frac{o}{l!}\prod_{k=1}^{1..l-1}(2o-l-k)}, \tag{1.6}$$

namely the first-order recurrence

$$\begin{aligned} &2(2l-2o-1)(l-o-1)(k-2o+1)(k+2o)S_o \\ -&(l-2o)(l-2o-1)(k+l-2o-1)+(k+l-2o-1) \;=\; 1 \end{aligned}$$

where S_n is the backwards shift operator $S_n f(n) = f(n-1)$. (This recurrence was found with Zeilberger's implementation of his "creative telescoping" algorithm [99], although it is readily verified by hand.) A similar recurrence relation exists for l. Since we have verified the recurrence for $l = 1$, $o = 1, 2, 3$, by induction the theorem is true for all positive integral m when $l = 1$. Likewise the theorem is true for all integral l, m. ◇

Creative telescoping proofs, like the one above, are straightforward but unenlightening. We note that a proof of this identity by comparing the quotients of successive derivatives of Chebyshev polynomials is possible, but quite tedious, and not terribly insightful. I do not know a purely combinatorial proof of this result.

We use the identity to prove a useful formula, a form of which seems to be well-known among some applied mathematicians [81]:

Theorem 1.4 *Let K be a field. Consider the recurrence relation in* $K[x]$

$$\begin{aligned} T_{n+2} &= xT_{n+1}(x) - T_n(x) \qquad (1.7)\\ T_1 &= x \\ T_0 &= 2 \end{aligned}$$

Then, for $n \geq 1$, $T_n(x)$ is a degree n, monic polynomial, which is either either even or odd, and the coefficient of z^{n-2l} is

$$\frac{(-1)^l}{l!} n \prod_{k=1}^{l-1} (n - l - k) \qquad (1.8)$$

for each $0 < l \leq \lfloor \frac{n}{2} \rfloor$.

Proof: Write c_l for the coefficient of z^{n-2l}. Solving the recurrence relation for T_n by standard techniques, we see that

$$\begin{aligned} c_l &= \frac{1}{2^{n-1}} (-4)^l \sum_{m=-\infty}^{\infty} \binom{n}{2m} \binom{m}{l} \qquad (1.9)\\ &= \frac{1}{2^{n-1}} (-4)^l \sum_{m=l}^{\lfloor \frac{n}{2} \rfloor} \binom{n}{2m} \binom{m}{l} \qquad (1.10) \end{aligned}$$

so, considering odd and even n separately, we have by the identity of the previous theorem

$$c_l = \frac{(-1)^l}{l!} n \prod_{k=1}^{l-1} (n-l-k).$$

$\diamondsuit$

Corollary 1.5 *Let* t *be the trace of a* 2×2 *matrix* $A \in SL_2$. *Then the trace of* A^n *is given by*

$$T_n(t) = t^n + \sum_{l=1}^{\lfloor n/2 \rfloor} c_l t^{n-2l} \qquad (1.11)$$

Proof: By the fundamental trace identity, equation 1.2, the trace of A^n is given by 1.7. $\diamondsuit$

Corollary 1.6 *The character variety of a finite group, or more generally of a finitely generated torsion group of bounded exponent, has zero dimension.*

Proof: If S is the set of all exponents of elements of the group, then any group element's SL_2 trace must satisfy one of $\{T_n(x) | n \in S\}$. Each of these polynomials has a finite solution set, and so each of the coordinates takes on a discrete number of values. $\diamondsuit$

For the reader's convenience, the first few T_n's are collected in table one.

The following formula, due to Jorgensen [75], extends the usefulness of 1.9 to arbitrary words on two letters.

2
x
x^2-2
x^3-3x
x^4-4x^2+2
x^5-5x^3+5x
$x^6-6x^4+9x^2-2$
$x^7-7x^5+14x^3-7x$
$x^8-8x^6+20x^4-16x^2+2$
$x^9-9x^7+27x^5-30x^3+9x$
$x^{10}-10x^8+35x^6-50x^4+25x^2-2$
$x^{11}-11x^9+44x^7-77x^5+55x^3-11x$
$x^{12}-12x^{10}+54x^8-112x^6+105x^4-36x^2+2$
$x^{13}-13x^{11}+65x^9-156x^7+182x^5-91x^3+13x$
$x^{14}-14x^{12}+77x^{10}-210x^8+294x^6-196x^4+49x^2-2$

Table 1.1 Trace polynomials of powers of an element

Proposition 1.7 *Let*

$$w = x^{a_1}y^{b_1}x^{a_2}y^{b_2}\cdots x^{a_k}y^{b_k}$$

be a cyclically reduced word on two letters. Let χ be a character of F_2, the free group on $\{x, y\}$. Write $\alpha+\alpha^{-1}$ for the number χx, and $\beta+\beta^{-1}$ for the number χy. Let $p = \chi(w)$, which by Vogt's identity is a polynomial in variables $\alpha+\alpha^{-1}$, $\beta+\beta^{-1}$, and $\chi(xy)$. Then the degree of $z = \chi(xy)$ in p is the number k of $(a_i, b_i) \neq (0,0)$ and the coefficient of z^k is

$$\prod_i \left(\frac{\alpha^{a_i}-\alpha^{-a_i}}{\alpha-\alpha^{-1}}\right)\prod_i\left(\frac{\beta^{b_i}-\beta^{-b_i}}{\beta-\beta^{-1}}\right). \tag{1.12}$$

We have generally found it more convenient to use Jorgensen's formula in a different form:

$$\prod_i \left(\frac{\alpha^{a_i} - \alpha^{-a_i}}{\alpha - \alpha^{-1}}\right) \prod_i \left(\frac{\beta^{b_i} - \beta^{-b_i}}{\beta - \beta^{-1}}\right) =$$

$$\prod_i \left(\sum_j (\alpha^{a_j - 1} + \alpha^{1-a_j}\right) \prod_i \sum_j (\beta^{b_j - 1} + \beta^{1-b_j})$$

$$= \prod_{i=1}^{k} T_i T_j.$$

We introduce some notation. If w is a word in a free group F_n, and if the character values

$$S = \{t_{w_1} = \operatorname{tr} w_1, t_{w_2} = \operatorname{tr} w_2, \ldots, t_{w_m} = \operatorname{tr} w_m \,|\, w_i \in F_n\}$$

generate the images of each word in F_n under any (SL_2) character χ, then the polynomial in $\mathbb{Z}[S]$ which defines the character image of w is denoted either $\operatorname{tr} w$, or t_w. (Both are standard notations.) Usually, it is convenient order the letters of F_n once and for all, and denote the character images of $x_1, x_2, x_3, \ldots, x_1x_2, \ldots$ etc. by $t_1, t_2, t_3, \ldots, t_{12}, \ldots$ etc. We will furthermore sometimes overload this notation, by considering inverses of generators, i.e., $t_{1^{-1}} = \operatorname{tr} x_1^{-1} (= \operatorname{tr} x_1)$, $t_{12^{-1}} = \operatorname{tr} x_1 x_{2^{-1}} (= t_1 t_2 - t_{12})$; and other powers of generators, e.g. $t_{1^2} = \operatorname{tr}(x_1^2)\ (= t_1^2 - 2)$.

Considering again a representation ρ of a finitely presented group G, we have maps:

$$
\begin{tikzcd}
\ker\psi \arrow[r, hook] \arrow[d] & F_n \arrow[r, "\psi"] \arrow[d] \arrow[dr] & G \arrow[d, "\rho"] \\
\{0_{SL_2}\} \arrow[r, hook] \arrow[d, "tr"] & SL_2\mathbb{C} \arrow[r] \arrow[d, "tr"] & SL_2\mathbb{C} \arrow[d, "tr"] \\
\{2\} \arrow[r, phantom, "\subseteq"] & \mathbb{C} \arrow[r, phantom, "\supseteq"] & \mathbb{C}
\end{tikzcd}
$$

Let I_n be the radical ideal defining the variety $X(G)$. The ring

$$R(F_n) = \mathbb{C}[t_1, t_2, \ldots, t_{12}, \ldots, t_{123}, \ldots]/I_n = \mathbb{C}[X]/I_n$$

is called the *ring of Fricke characters* of the free group $F_n =< g_1, \ldots, g_n >$. (More generally, the ring of Fricke characters $R(G)$ of a finitely generated group G is the coordinate ring of the character variety of G.) Computing inside $R(F_n)$ can be hard for at least four reasons:

1. $R(F_n)$ has $n + \binom{n}{2} + \binom{n}{3} = \frac{n(n^2+5)}{6}$ variables. The average and worst-case running times for most algorithms in computational commutative algebra depend exquisitely on the number of variables in ring presentations; for some illustrative examples, see [116] or [17].

2. $R(F_n)$ has high regularity.

3. The polynomials generating I_n, the irreducible ideal which defines the ideal $X(F_n)$ are the Gonzalez-Montesinos relations, whose definition we recall below. They are very symmetrical. Indeed, they admit an intransitive group of symmetries of order $n!$. Symmetry among the generators of an ideal cause algorithms such as Buchberger's algorithm to do work which does not move it towards its termination condition (for examples, see [115].)

4. The Gonzalez-Montesinos relations have $2\binom{n}{3} + \binom{n-2}{2} + \binom{n-3}{2}$ polynomials comprised of

$$15\binom{n}{3} + 12\binom{n-2}{2} + 24\binom{n}{3} + 10\binom{n-3}{2}$$

(not all distinct) monomials. This imposes a real "book-keeping" cost as n grows large.

For $n < 4$ these are not serious objections to computing inside $R(G)$. For $n = 1, 2$, $I_n = \{0\}$, and so two characters $\overline{f}$, $\overline{g}$ are equal if and only if $f = g$ in $\mathbb{C}[X]$. For $n = 3$, the results of [68] suffice - I_3 is principal over $\mathbb{Z}$,

$$I_3 = (t_{123}^2 - Pt_{123} + Q),$$

where

$$
\begin{aligned}
P &= t_1t_{23} + t_2t_{13} + t_3t_{12} - t_1t_2t_3 \\
Q &= t_1^2 + t_2^2 + t_3^2 + t_{12}^2 + t_{13}^2 + t_{23}^2 + t_{12}t_{13}t_{23} \\
&\quad -t_1t_2t_{12} - t_1t_3t_{13} - t_2t_3t_{23} - 4.
\end{aligned}
$$

$t_{123}^2 - Pt_{123} + Q$ is thus evidently a Grobner basis for I_3 with respect to any term ordering, and so we have many normal-form algorithms for $R(F_3)$. Likewise, since Gonzalez-Acuna and Montesinos-Amilibia have provided an explicit set of polynomials generating I_n, [57], we may in principle find a Grobner basis for $R(F_n)$, and thus a normal-form procedure for $R(I_n)$. But when $n = 4$, this is already a non-trivial calculation, and for larger n, our objections 1-4 above seem formidable.

Our primary object of study is $R(F_n)$. Its defining ideal is generated by the Gonzalez-Montesinos polynomials:

$$
t_{abc}^2 - P_{abc}t_{abc} + Q_{abc} \qquad a < b < c \tag{1.13}
$$

for

$$
\begin{aligned}
P_{abc} &= t_at_{bc} + t_bt_{ac} + t_ct_{ab} - t_at_bt_c \\
Q_{abc} &= t_a^2 + t_b^2 + t_c^2 + t_{ab}^2 + t_{ac}^2 + t_{bc}^2 + t_{ab}t_{ac}t_{bc} \\
&\quad -t_at_bt_{ab} - t_at_ct_{ac} - t_bt_ct_{bc} - 4,
\end{aligned}
$$

$$\begin{vmatrix} t_1^2-4 & 2t_{12}-t_1t_2 & 2t_{1a}-t_1t_a & 2t_{1b}-t_1t_b \\ 2t_{12}-t_1t_2 & t_2^2-4 & 2t_{2a}-t_2t_a & 2t_{2b}-t_2t_b \\ 2t_{1a}-t_1t_a & 2t_{2a}-t_2t_a & t_a^2-4 & 2t_{ab}-t_at_b \\ 2t_{1b}-t_1t_b & 2t_{2b}-t_2t_b & 2t_{ab}-t_at_b & t_b^2-4 \end{vmatrix} \quad 3 \le a < b \le n \tag{1.14}$$

$$\begin{vmatrix} t_1^2-4 & 2t_{12}-t_1t_2 & 2t_{13}-t_1t_3 & 2t_{1a}-t_1t_a \\ 2t_{12}-t_1t_2 & t_2^2-4 & 2t_{23}-t_2t_3 & 2t_{2a}-t_2t_a \\ 2t_{13}-t_1t_3 & 2t_{23}-t_2t_3 & t_3^2-4 & 2t_{3a}-t_3t_a \\ 2t_{1b}-t_1t_b & 2t_{2b}-t_2t_b & 2t_{b3}-t_bt_3 & 2t_{ab}-t_at_b \end{vmatrix} \quad 3 < a < b \le n \tag{1.15}$$

$$(t_{123}-t_{132})(2t_{abc}-t_at_bt_c-t_at_{bc}-t_bt_{ac}-t_ct_{ab}) \tag{1.16}$$

$$-\begin{vmatrix} t_1 & t_{1a} & t_{1b} & t_{1c} \\ t_2 & t_{2a} & t_{2b} & t_{2c} \\ t_3 & t_{3a} & t_{3b} & t_{3c} \\ 2 & t_a & t_b & t_c \end{vmatrix} \quad a < b < c \tag{1.17}$$

We introduce some useful notation. Let G be presented by $< g_1, \ldots, g_n | R >$; call this presentation P. By [57, Section three], the character variety $X(G)$ is the intersection of two other varieties: $X(F_n)$ and

$$V(\,(\operatorname{tr} R') - 2) = V(I_w),$$

where $\operatorname{tr} R' = \{t_r \mid r \in (\{e\} \cup \{g_1, g_2, \dots, g_n\})R\}$ for some arbitrary choice $\{t_r\}$ of trace polynomials in X, one t_r representing each $w \in R'$. We will call the variety $W(P) = V(\,(\operatorname{tr} R)\,)$ a *Whittemore variety* of the presentation $< g_1, \dots, g_n | R >$, which we write $V_w(G) = W(P)$. This is a double abuse of notation: the group G's Whittemore variety is really associated with the presentation $P :< g_1, \dots, g_n | R >$, and we have a choice of trace polynomials for each $r \in R$ whenever $n \geq 3$.

Example 1.8 *We will defer choosing a canonical way of generating* I_w *for a general finitely presented group until the next chapter, but we give here an* ad hoc *example for the group*

$$< f_1, f_2, f_3 \mid f_1 f_2 f_3^{-1}, f_2 f_3 f_1^{-1}, f_3 f_1 f_2^{-1} >$$

which is the finite Fibonacci group $F(2,3)$. *The character variety* $X(F_3)$ *is the variety of the principal ideal*

$$t_{123}^2 - P_{123} t_{123} + Q_{123},$$

where

$$
\begin{aligned}
P &= t_1t_{23} + t_2t_{13} + t_3t_{12} - t_1t_2t_3 \\
Q &= t_1^2 + t_2^2 + t_3^2 + t_{12}^2 + t_{13}^2 + t_{23}^2 + t_{12}t_{13}t_{23} \\
&\quad -t_1t_2t_{12} - t_1t_2t_{13} - t_2t_3t_{23} - 4,
\end{aligned}
$$

as in 1.13 above; while $V_w(F(2,3))$ *may be chosen to be the variety defined by*

$$
\begin{aligned}
&(\operatorname{tr} f_1f_2f_3^{-1}, \operatorname{tr} f_2f_3f_1^{-1}, \operatorname{tr} f_3f_1f_2^{-1}) \qquad (1.18) \\
&\qquad = (\underline{t_3t_{12}} - t_{123}, \underline{t_1t_{23}} - t_{123}, \underline{t_2t_{13}} - t_{123}).
\end{aligned}
$$

Note that each polynomial has its degree-lexicographic leading monomial underlined. Since these terms are relatively prime, the set (1.18) forms a Grobner basis. (A Grobner basis like this one, where the leading terms are relatively prime, is called a structural *Grobner basis. See [117].)*

As the term t_{123} *occurs in none of the leading terms of this graded Grobner basis, the coordinate ring of* $V_w(F(2,3))$ *has codimension of at least one. A coset enumeration (using the system [40], for example) shows that* $|F(2,3)| = 8 < \infty$*, so that* $\dim X(F(2,3)) = \dim X(F_n) \cap V_w = 0$*, and in so particular* $X(F(2,3)) \neq V_w$*, by corollary 1.6.* $\diamondsuit$

When we are calculating V_W *for a group, the nicest sorts of relators which we might encounter are perfect powers, since we have an easy way to write down* $\operatorname{tr} w^n - 2 =$

$T_n(t_w) - 2$ *using corollary 1.5. We can also go backwards; the next theorem tells us how to write* x^n *in terms of the polynomials* $T_n(x)$.

Theorem 1.9 *Define* $\langle\binom{n}{k}\rangle$ *by* $\langle\binom{n}{k}\rangle = \binom{n}{k}$ *if* n *and* k *are integers;* $\langle\binom{n}{k}\rangle = 0$ *otherwise. Define the polynomials* $\tilde{T}_n(x)$ *by*

$$\tilde{T}_n(x) = \begin{cases} 1 & \text{if } n = 0 \\ T_n(x) & \text{if } n > 0. \end{cases}$$

Then

$$x^n = \sum_k \left\langle \binom{n}{\frac{k}{2}} \right\rangle \left(\tilde{T}_{(n-k)}(x)\right).$$

This expression for x^n *as a linear combination of the polynomials* $T_n(x)$ *is unique.*

Proof: The statement of the theorem is equivalent to the statement that the matrix illustrated in figure 1-1 is the inverse of the matrix in figure 1-2. Let N_1 be the $n \times n$ upper-left submatrix of M. Let N_2 be the $n \times n$ upper-left submatrix of of the matrix in figure 1-2. Let α be the a–th row of N_2, β the b–th column of N_1. Index the entries of α, starting at the right, with α_0 the rightmost entry of the row. (Continue this sequence to the left, so that the sequence $\{\alpha_i\}$ runs through all of the binomial coefficients.) Likewise index the entries of β, starting from the top, with β_0 the top entry of the column. The generating function of the sequence $\{\alpha_i\}$ is clearly

$$A_a(z) = (1+z^2)^a z^{n-a-1}.$$

Likewise the generating function of the sequence $\{\beta_i\}$ is

$$B_b(z) = z^b(1-z^2)(1+z^2)^{-b-1}.$$

(Proof: The numbers $N_1(x,y)$ satisfy the recurrence

$$N_1(x,y) = N_1(x-1,y-1) - N_1(x,y-2). \tag{1.19}$$

This is immediate from the defining relation $T_{n+2} = xT_{n+1}(x) - T_n(x)$. Multiply each side of (1.19) by z^n, and sum over all $n \geq 0$. We have

$$\sum_n N_1(x,y)z^n = \sum_n N_1(x-1,y-1)z^n - \sum_n N_1(x,y-2)z^n$$

$$B_x(z) = zB_{x-1}(z) - z^2 B_x(z)$$

$$B_x(z) = \frac{z}{1+z^2} B_{x-1}(z).$$

Since $B_0(z) = \frac{2}{1+z^2} - 1$, we have

$$B_x(z) = \left(\frac{z}{1+z^2}\right)^x \left(\frac{1-z^2}{1+z^2}\right)$$

as claimed.)

Now consider the generating function of the convolution $\alpha * \beta$, which is the product $A_a(z)B_b(z)$:

$$(A_a B_b)(z) = z^{n-1+(b-a)}(1+z^2)^{a-b-1}(1-z^2).$$

We examine the coefficient of z^{n-1} in this power series. There are three cases:

$a < b$ Then $(A_a B_b)(z) = z^{n-1}F(z)$, where $F(z)$ is analytic at $z = 0$, so the coefficient of z^{n-1} in $(A_a B_b)(z)$ is 0.

$a = b$ Then the coefficient of z^{n-1} in $(A_a B_b)(z)$ is obviously 1.

$a > b$ Then $(A_a B_b)(z)$ is a polynomial. The coefficient of z^{n-1} in $(A_a B_b)(z)$ is the coefficient of z^{a-b} in

$$(1+z^2)^{a-b-1}(1-z^2).$$

This coefficient is 0 if $(a-b)$ is odd, and is

$$\binom{a-b-1}{\frac{a-b}{2}} - \binom{a-b-1}{\frac{a-b}{2}-1}$$

$$M = \begin{pmatrix} 1 \\ 0 & 1 \\ -2 & 0 & 1 \\ 0 & -3 & 0 & 1 \\ 2 & 0 & -4 & 0 & 1 \\ 0 & 5 & 0 & -5 & 0 & 1 \\ -2 & 0 & 9 & 0 & -6 & 0 & 1 \\ 0 & -7 & 0 & 14 & 0 & -7 & & \ddots \end{pmatrix}$$

Figure 1-1 The matrix M

$$M^{-1} = \begin{pmatrix} 1 \\ 0 & 1 \\ 1 & 0 & 1 \\ 0 & 3 & 0 & 1 \\ 6 & 0 & 4 & 0 & 1 \\ 0 & 10 & 0 & 5 & 0 & 1 \\ 20 & 0 & 15 & 0 & 6 & 0 & 1 \\ 0 & 35 & 0 & 21 & 0 & 7 & & \ddots \end{pmatrix}$$

Figure 1-2 The inverse of matrix M

if $(a-b)$ is even. By the symmetry property of Pascal's triangle,

$$\binom{a-b-1}{\frac{a-b}{2}} - \binom{a-b-1}{\frac{a-b}{2}-1} = 0$$

when $(a-b-1)$ is odd.

So $(\alpha * \beta)_{n-1}$ is 0 if $a \neq b$, and 1 if $a = b$. But $(\alpha * \beta)_{n-1}$ is precisely the dot product of the $a-th$ row of N_2 with the $b-th$ column of N_1, so the matrices N_1 and N_2 are inverses of each other. ◇

Chapter 2

Simplification of group presentations: dimension theory aids string matching

In this brief chapter, we explore definitions of group presentation simplicity that use geometric information taken from the group presentation. Our approach is firmly experimental.

Suppose that $\mathsf{G} = \langle \mathsf{Y} | \mathsf{R} \rangle$ and $\mathsf{H} = \langle \mathsf{Z} | \mathsf{S} \rangle$ are two finite presentations of groups. Then it is a well-known theorem of Tietze that that G is isomorphic to H if and only if the presentation $\langle \mathsf{Y} | \mathsf{R} \rangle$ may be obtained from $\langle \mathsf{Z} | \mathsf{S} \rangle$ by a finite sequence of "Tietze transformations:"

1. Adding a new generator g, and a new relation $\mathsf{g}w$, where w is any word in Z
2. Deleting a generator g, and a relation $\mathsf{g}w$, where w is a word in $\mathsf{Z} - \{\mathsf{g}\}$, and no other relation in S uses the generator g
3. Adding a relation that is a consequence of other relations in S
4. Deleting a relation that is a consequence of the other relations in S.

(If two presentations $\langle \mathsf{Y} | \mathsf{R} \rangle$ and $\langle \mathsf{Z} | \mathsf{S} \rangle$ of the same group are not finite, then in general we may not find a finite sequence of Tietze transformations transforming one into the other, even if $|Y|, |Z| < \infty$. For example, $\langle a \,|\, a, a^2, a^3, \ldots \rangle$ is clearly a presentation of the trivial group, and no Tietze transformation can ever yield a presentation where there are not infinitely many relators, which uses only one generator. But

$$\langle a, b \,|\, a, b, w_2(a, b), w_2(a, b), \ldots \rangle$$

is such a presentation of the trivial group, so it can't be obtained from $\langle a \,|\, a, a^2, a^3, \ldots \rangle$ in only finitely many steps.)

The search-space of Tietze transforms of a finitely presented group is an important object of study in computational group theory. A typical application of searching through the space of Tietze transforms of a finitely presented group is the problem of simplifying a presentation. What does it mean for one presentation of a finitely presented group to be "simpler" than another? Some typical definitions are that a presentation is simpler if it has fewer generators, or fewer relations, or smaller total relator length. Given a presentation $< Y|R >$, there is a (finite) path of Tietze transformations that transforms $< Y|R >$ into a "simplest" presentation. But, for general finitely presented groups, the isomorphism problem is unsolvable; and thus, if our definition of simplicity admits a unique simplest presentation then the problem of finding this path is algorithmically undecidable.

In practice, then, before attempting to simplify a finite presentation of group G via a sequence of Tietze transformations, one chooses a binary relation $\leq$ on the set of all group presentations, defining $< Y|R >$ to be simpler than $< Z|S >$ if

$< Y|R > \leq < Z|S >$. Our search space is an oriented graph, where the vertices are finite presentations of groups isomorphic to G, and where edges correspond to Tietze transforms. We search through this graph, until we find a locally minimal (or acceptable minimal) element. The complexity of this graph gives the whole theory a very computational flavor. The hardest part of simplifying a presentation by searching the space of Tietze transformations is determining the search-space itself: transformations of type (2) and (4) require a common-substring search, which is much harder than merely transforming a presentation, or comparing the number of generators or relation lengths [113, Section 6.4] [64]. This is a distinctive aspect of this problem, which is often absent in other applications of combinatorial search.

In this chapter, we will explore a new definition of simplicity for a group presentation. Our definition has the advantage that it allows us to avoid common-substring searching when deciding whether to apply Tietze transformations of type (1) and (2). Briefly, we will consider one presentation $P_1 =< x_1, \ldots, x_n | R_1 >$ of a finitely presented group G to be simpler than another presentation $P_2 =< x_1, \ldots, x_m | R_2 >$, if the Gonzalez-Montesinos presentation of the character variety of F_n "contributes" less, via presentation P_1, to the Hilbert polynomial of the ideal $I(X(G))$ than the corresponding presentation of F_m via presentation P_2. (We will clarify what we mean by "contributes less" below.)

Let us first consider a simple example. The triangle group $(2, 3, \infty)$ may be presented as:

$$\Gamma = < g_1, g_2, g_3 | \, g_1^2 = (g_1 g_2)^3 = g_3 g_1^{-1} g_2^2 = e > .$$

Its character variety's Gonzalez-Montesinos presentation is

$$T_1 = \{x_1^2 - 2, x_{12}^3 - 3x_{12}, x_1 x_2, x_{23} - x_2 x_{123} - x_1 x_3 + x_{13}, x_{123}^2 - P x_{123} + Q\}$$

where P and Q are as defined in Chapter 1. Buchberger's criterion states that a set G of polynomials is a Grobner basis if and only if, for each f_1, $f_2 \in \Gamma$,

$$S(f_1, f_2) = \frac{\mathrm{lcm}(\mathrm{in}(f_1), \mathrm{in}(f_2))}{\mathrm{LC}(f_1)\,\mathrm{in}(f_1)} f_1 - \frac{\mathrm{lcm}(\mathrm{in}(f_1), \mathrm{in}(f_2))}{\mathrm{LC}(f_2)\mathrm{LC}(f_2)} f_2$$

reduces to 0 on division by G. If we order, say, by total degree, then T_1 is not a Grobner basis, since

$$S(x_1^2 - 2, x_1 x_2 x_{23} - x_2 x_{123} - x_1 x_3 + x_{13}) = x_1^2 x_3 + x_1 x_2 x_{123} - x_1 x_{13} - 2x_2 x_{23}$$

has remainder $x_1 x_2 x_{123} - x_1 x_{13} - 2x_2 x_{23} - 2x_3$ on division by T_1. A Tietze transformation of type (2), however, yields a presentation

$$\Gamma = < g_1, g_2 | \, g_1^2 = (g_1 g_2)^3 = e >$$

whose Gonzalez-Montesinos presentation is

$$T_2 = \{x_1^2 - 4, x_{12}^3 - 3x_{12}, x_1^2x_2 - x_1x_{12} - 4x_2, x_{12}^2x_1 - x_{12}x_2 - x_1 - x_2,$$
$$x_{12}^2x_2 - x_{12}x_1 - x_2 - x_1\}$$

which is a Grobner basis.

If we wished to use a search algorithm to simplify a group presentation so that its Gonzalez-Montesinos polynomials are closer to being a Grobner basis, then we would need a way to compare two polynomials and determine which is more "Grobner-like." Some ways to measure how far T_1 is from being a Grobner basis include the number of terms in the remainders of the S-polynomials, or the maximum degree of the remainders. But these measures would be computationally difficult to compute, and this would dominate the cost of the search. Moreover, most of the polynomials in the set of Gonzalez-Montesinos polynomials for a presentation will be determinantal or Fibonacci-type polynomials, and there is no obvious way to use our knowledge of the special properties of such polynomials to hasten the computation of these the remainders of their S-polynomials. We will instead take a slightly roundabout approach.

Recall from Chapter 1, corollary 1.5, that if $u = \operatorname{tr} w$, then the coefficient of u^{n-2l} in $\operatorname{tr} w^n$ is

$$\begin{cases} 1, & l = 0 \\ \frac{(-1)^l}{l!} n \prod_{k=1}^{l-1}(n-l-k), & l > 0. \end{cases}$$

This allows us to calculate the Fricke polynomials for relations of the form w^n in Γ efficiently, since the coefficient of u^{n-2l-2} may be found iteratively from the coefficient of u^{n-2l}.

But calculating the Fricke polynomial of each potential new relation would easily dominate a search for a simpler group presentation, and probably be impractical for most interesting examples. Instead, one reasonable approach might be to *evaluate* these polynomials at some integral point. Following the advice of [81], an efficient way to do perform this calculation is to use the relation for Chebyshev polynomials of the first kind

$$2T_n(x)T_m(x) = T_{n+m}(x) + T_{n-m}(x)$$

which implies the relation

$$(\operatorname{tr} x^n)(\operatorname{tr} x^m) = \operatorname{tr} x^{n+m} + \operatorname{tr} x^{n-m}$$

for Fricke polynomials. But the difficulty of guessing a good point (or a good set of points) at which to evaluate the trace polynomials in this manner makes this approach seem unpromising.

In our example above, we saw that Γ had a presentation in which its ideal of character relations' "natural" presentation was a structural Grobner basis. It would be nice if this were the norm, rather than the exception. In general, sets of polynomials whose leading terms with respect to a given term order are relatively prime are rather rare; and likewise, two randomly selected words in a free group will rarely have trace polynomials with relatively prime leading terms. (Of course, relators which arise in practical computations are not at all uniformly selected from the set of all words in a free group: they are, for purely physical reasons, restricted to words of some reasonable length.)

Since isomorphic finitely generated groups have isomorphic character varieties, the dimension of the character variety is invariant under Tietze transformations. For $SL_2\mathbb{C}$ characters, the character variety of $G = \langle Y | R \rangle$, $|Y| < \infty$ is the intersection of the character variety of the free group on $|Y|$ letters, $X(F_{|Y|})$, and the Whittemore variety arising from the relators R, $W(R)$. The automorphisms of G induce morphisms of the character variety $X(G)$. (Since inner automorphisms of G fix characters, we have an injection $\text{Out } G \hookrightarrow \text{Aut } X(G)$.)

Neither of $X(F_{|Y|})$ or $W(R)$ are invariant under automorphisms of G. However, since there exists a smallest n such that G is generated by n elements, the dimension of the smallest such $X(F_n)$ is an invariant of the group G. Likewise, smallest dimension of a Whittemore variety corresponding to a set of relators defining G is a well-defined invariant of G. Let's denote the rank of G by rankG, and the smallest dimension of a Whittemore variety corresponding to a set of relators defining G (which may be on a larger set of generators) by $\dim_w G$. If $G = \langle Y \mid R \rangle$, with $|Y| = \mathrm{rank}G$, and if $\langle Z \mid S \rangle$ is another finite presentation of G, then it is not necessarily true that $W(R) = W(S)$ implies that $|Z| = \mathrm{rank}G$ (as may easily be seen, for example, by the Klein four-group

$$\langle x, y \mid x^2, y^2, (xy)^2 \rangle = \langle x, y, z \mid x^2, y^2, (xy)^2, xyz^{-1} \rangle .)$$

But it is true that, if $\langle Z \mid S \rangle$ is obtained from $\langle Y \mid R \rangle$ by a Tietze transformation of type (1), then $\dim W(R) \leq \dim W(S)$. For, adjoining a new generator to $\langle Y \mid R \rangle$ adds $1 + |Y| + \binom{|Y|}{2}$ new variables to the ideal of character relations, and adding new relation tw for some $w \in Y$ adds the polynomial $p = (\mathrm{tr}\, tw) - 2$. p cannot depend only on traces of words in $\langle Y \rangle$. Furthermore, we have

Proposition 2.1 *Suppose that* $\langle Z \mid S \rangle$ *is obtained from* $\langle Y \mid R \rangle$ *by a Tietze transformation of type (1), adjoining variable* z *to* Y*, where* $z = w$*. Then* $\dim W(R) \leq \dim W(S)$*, and furthermore* $\dim W(R) < \dim W(S)$ *if* $\mathrm{tr}\, z^{-1}w$ *does not involve any of the new variable* $\{t_z, t_{zy}, t_{zxy} \mid x, y \in Y\}$*. In particular, a Tietze transformation of type (2) will always reduce the dimension of a Whittemore variety of a presentation, if the generator* z *deleted only appears in a word of form* $z^{-1}w$*, where* $\mathrm{tr}\, z^{-1}w$ *does not involve each variable* $\{t_z, t_{zy}, t_{zxy} \mid x, y \neq z\}$.

We are now in a position to define (and to try to justify) our definition of "simplicity" of a group presentation. We will say that a presentation of a group is *simpler* than another if its Whittemore variety has smaller dimension. Why is this a reasonable definition of simplicity? First, we should note that, given any presentation P, it is possible to find a sequence of Tietze transformations which reduce P to "simplest" (really "locally simplest") form P′; i.e., any transformation either makes P′ less simple, or keeps it equally simple. Secondly, by the above proposition, a Tietze transformation of type (1) which makes a presentation less simple increases the number of generators. What about transformations of type (3) and (4)?

Theorem 2.2 *Let* $< Y = \{g_1, g_2, \ldots, g_n\} | R >$ *be a finite presentation. Then*

a *A Tietze transformation of type (1) either increases the dimension of the Whittemore variety, or keeps it the same.*

b *A Tietze transformation of type (2) either reduces the dimension of the Whittemore variety, or keeps it the same.*

c *A Tietze transformation of type (3) or type (4) does not change the dimension of the Whittemore variety.*

Proof: Parts **a** and **b** were proved in the last proposition. We will show **c** for Tietze transformation of type (3); this implies the statement for transformations of type (4), since type (4) transformations are inverses of transformations of type (3). Suppose that a new relator r is added, which is the consequence of two relators w, x in R; i.e., up to a cyclic permutation, $r = xw$. Adjoin variables $\operatorname{tr} x, \operatorname{tr} w$ to our set of indeterminates, and denote by I the ideal of character relations arising from the original relators. In particular,

$$I' = (\operatorname{tr} x - 2, \operatorname{tr} w - 2, \operatorname{tr} x g_i - t_i, \operatorname{tr} w g_i - t_i, \dots) \subset I.$$

We proceed by induction on the word-length of x. Suppose $|x| = 1$, say $x = g_1$. Then we have immediately that $\operatorname{tr} r = \operatorname{tr} g_1 w \equiv 2 \bmod I'$. Furthermore, it is easy to see that $\operatorname{tr} g_i r \equiv t_i \bmod I'$, since

$$\begin{aligned}
\operatorname{tr} g_1 r &= \operatorname{tr} g_1^2 w \\
&= t_1 \operatorname{tr} g_1 w - \operatorname{tr} w \\
&\equiv t_1^2 - \operatorname{tr} w \bmod I' \\
&\equiv 4 - 2 = 2 \equiv t_1 \bmod I'
\end{aligned}$$

and for $j \neq 1$,

$$
\begin{aligned}
\operatorname{tr} g_j g_1 w &= t_1 \operatorname{tr} t_j w - \operatorname{tr} w g_j g_1^{-1} \\
&\equiv t_1 t_j - \operatorname{tr} w g_j g_1^{-1} \bmod I' \\
&\equiv t_1 t_j - t_j \operatorname{tr} g_1^{-1} w + \operatorname{tr} g_1^{-1} g_j^{-1} w \bmod I' \\
&\equiv t_{1j} \operatorname{tr} w - \operatorname{tr} g_j g_1 w \bmod I' \\
2 \operatorname{tr} g_j g_1 w &\equiv 2t_j. \bmod I'
\end{aligned}
$$

Now suppose that part c is true when one relator has length $< n$. Let $|x| = n$. By repeated application of the induction hypothesis, and the fundamental trace identity,

$$
\begin{aligned}
\operatorname{tr} g_j x w &= \operatorname{tr} x \operatorname{tr} t_j w - \operatorname{tr} w g_j x^{-1} \\
&\equiv \operatorname{tr} x t_j - \operatorname{tr} w g_j x^{-1} \bmod I'' \\
&\equiv \operatorname{tr} x t_j - t_j \operatorname{tr} x^{-1} w + \operatorname{tr} x^{-1} g_j^{-1} w \bmod I'' \\
&\equiv \operatorname{tr} x g_j \operatorname{tr} w - \operatorname{tr} g_j x w \bmod I'' \\
2 \operatorname{tr} g_j x w &\equiv 2t_j \bmod I'' \\
\operatorname{tr} g_j x w &\equiv t_j \bmod I''
\end{aligned}
$$

where $j \in \{1, 2, \ldots, n\}$ and I'' is generated by I', together with all elements of the form $\operatorname{tr} g_j y v - t_j$, for relators y, v with $|y| < n$. Finally, we note that $\operatorname{tr} g_j x w - 2 \equiv 0$, by substituting "$g_j = e$" in the above calculation. ◇

The simplest example of Theorem 2.2 is the free group of rank two $P_1 = \langle x, y \,|\, \rangle$. Here, the Whittemore variety is unique: $W(P_1) = \mathbb{C}^3$, and so $\dim W(P_1) = 3$. (It is clear that in fact $\dim_W P_1 = 3$.) A Tietze transformation of type (1) gives us the isomorphic group presentation $P_2 =< x, y, z \,|z >$. Now we have

$$\begin{aligned} W(P_2) &= \mathbb{V}(t_z - 2,\ t_{xz} - t_x,\ t_{yz} - t_y,\ t_{z^2} - t_z) \\ &= \mathbb{V}(t_z - 2,\ t_{xz} - t_x,\ t_{yz} - t_y,\ 4 - 2 - 2) \\ &= \mathbb{V}(t_z - 2,\ t_{xz} - t_x,\ t_{yz} - t_y) \end{aligned}$$

and so $\dim W(P_2) = 4$.

If instead we apply a different Tietze transformation of type (1) to P_1, perhaps $P_3 =< x, y, z \,|zy^{-1}x^{-1} >$, then

$$\begin{aligned} W(P_3) &= \mathbb{V}(\operatorname{tr} zy^{-1}x^{-1} - 2,\ \operatorname{tr} xzy^{-1}x^{-1} - t_x,\ \operatorname{tr} yzy^{-1}x^{-1} - t_y,\ \operatorname{tr} z^2y^{-1}x^{-1} - t_z) \\ &= \mathbb{V}(\, t_z t_{xy} - t_{xyz} - 2,\ t_z t_y - t_{yz} - t_x,\ t_{yz} t_{xy} - t_y t_{xyz} + t_{xz} - t_y, \\ &\qquad t_z^2 t_{xy} - t_z t_{xyz} - t_{xy} - t_z \,) \end{aligned}$$

A Grobner basis for the polynomials defining $W(P_3)$, with respect to degree-lexicographic order with $t_x > t_y > t_{yz} > t_{xyz} > t_{xy} > t_z > t_{xz}$ is:

$$\{\underline{t_z t_y} - t_{yz} - t_x, \underline{t_y t_{xyz}} - t_z t_{yz} + t_y - t_z, \underline{t_{xy}} - t_z, \underline{t_z^2} - t_{xyz} - 2,$$
$$\underline{t_x t_{xyz}} + t_x - t_{yz} - t_{xyz} - 2, \underline{t_z t_x} - t_y - t_z, \underline{t_x^2} + t_x t_{yz} - t_y^2 - t_x - t_{yz}\} \ .$$

Inspection of the leading terms, which generate the initial ideal, shows that $\dim W(P_3) = 3$.

This example, as simple as it is, exhibits two omissions in our definition of presentation simplicity. Firstly, we have not decided on a canonical way to choose the Whittemore variety of a presentation. Secondly, and more disturbingly, it seems that the dimension of $W(P)$ is much too coarse a measure of simplicity: after all, P_3 "looks" quite a bit different than P_1, yet they both have the same $\dim W(P)$.

Definition 2.3 *(The canonical Whittemore variety of a presentation) Let* $P = <Y|R>$ *be a presentation. We will choose* $W(P)$ *as follows: For each* $w \in (Y \cup \{e\})R$, *assign variable* $p(w)$ *the value* $\operatorname{tr} w$.

1. *First, we apply the fundamental trace relation repeatedly to make* p *square-free (i.e., we transform* p *into an equivalent Fricke polynomial consisting of the traces of square-free words.) We always choose to eliminate the left-most syllable of length greater than one, and we always cyclically permute the word so that the syllable being permuted is on the right. (Note that this means that, in general, cyclic permutations of a relator may give different trace polynomials to the Whittemore variety. We choose this definition for its ease of computation, as opposed, say, to choosing the longest syllable, or choosing alphabetically.)*

   Example 2.4 *We would transform* $t_{12^23^{-1}4} \mapsto t_{3^{-1}412^2} \mapsto t_{3^{-1}412}t_2 - t_{3^{-1}41}$

2. *We eliminate inverses in words in the same manner:*

Example 2.5 $t_{3^{-1}41} \mapsto t_{413^{-1}} \mapsto t_{41}t_3 - t_{413}$ $(= t_{14}t_3 - t_{134})$

3. *Each word in the trace expression now is a square-free, inverse-free word. For each word of length* $N > 3$*, apply Vogt's identity, with*

$$A = \text{ the first } N - 3 \text{ letters of the word.}$$

Repeat until all the words have length 3.

4. *Finally, apply Fricke's lemma in the unique way, so that each trace is a trace of an ordered word on* 3 *or less letters.*

(Fractional dimension of an ideal) Let $\prec$ be a graded term order on $\mathbb{C}[X]$. Let Γ be the reduced lattice basis of M, a monomial ideal in $\mathbb{C}[X]$. The *fractional dimension of* M is the real number

$$\sum_{l \in A} \min_{p \in \Gamma} \frac{\operatorname{dist}(l, p)}{\operatorname{dist}(l, p) + 1}.$$

Let $\prec$ be a graded term order on $\mathbb{C}[X]$, and I be an ideal in $\mathbb{C}[X]$. then the *fractional dimension of* I *with respect to graded term order* $\prec$ is the fractional dimension of $\mathrm{in}_{\prec} I$.

In the appendix (page 94) we present a GAP package that searches the space of Tietze transforms of a finitely presented group, attempting to minimize the "fractional" dimension of the Whittemore variety. We illustrate this technique with a GAP session.

```
gap> Read( "FPFricke.g" ); Read( "fibs.g" );
fpfricke, Version .899927
```

("fibs.g" defines the generalized Fibonacci groups [73]

$$F(r,n) = \left\langle x_1, x_2, \ldots, x_n \mid \left\{ x_i x_{i+1} \cdots x_{i+r-1} x_{i+r}^{-1} \middle| i \in \{1,2,\ldots,n\} \right\} \right\rangle$$

where the subscripts in the relators are taken mod n.)

```
gap> G := FibonacciGroup( 7,5 );
Group( f.1, f.2, f.3, f.4, f.5 )
gap> P := PresentationFpGroup( G );
<< presentation with 5 gens and 5 rels of total length 40 >>
gap> T := Copy( P );
<< presentation with 5 gens and 5 rels of total length 40 >>
gap> TzWhit( P );
gap> P;
<< presentation with 5 gens and 5 rels of total length 24 >>
```

We have reduced the total size of the relators from 40 characters to 24, by using TzWhit, which tries to reduce the "partial" dimension of the Whittemore variety's defining polynomials coming directly from pairs of generators. Could we do better by considering triples of generators?

```
gap> TzWhitTriples( T );
#I  new generator is _x6
gap> T;
<< presentation with 5 gens and 6 rels of total length 32 >>
gap> U := Copy( P );
<< presentation with 5 gens and 5 rels of total length 24 >>
gap> TzWhitTriples( U );
gap> U;
```

Apparently not! Let's try simplifying $F(7,5)$ with GAP's own presentation simplification function, TzGoGo.

```
V := Copy( P );
gap> TzGoGo( V );
#I  there are 2 generators and 2 relators of total length 54
gap> G;
Group( f.1, f.2, f.3, f.4, f.5 )
gap> W := PresentationFpGroup( G );;
gap> TzGoGo( W );
#I  there are 2 generators and 2 relators of total length 46
```

Can we do better than this, by applying TzWhit?

```
gap> TzWhit( V );
```

```
#I  new generator is _x7
gap> V;
<< presentation with 2 gens and 3 rels of total length 79 >>
gap> TzGoGo( V );
```

Apparently not.

```
gap> TzWhitTriples( V );
gap> V;
<< presentation with 2 gens and 2 rels of total length 79 >>
gap> TzGoGo( V );
gap> V;
<< presentation with 2 gens and 2 rels of total length 79 >>
```

It gets worse:

```
gap> TzWhit( V );
#I  new generator is _x8
gap> V;
<< presentation with 2 gens and 3 rels of total length 104 >>
gap> TzWhitTriples( V );
gap> V;
<< presentation with 2 gens and 2 rels of total length 104 >>
gap> TzGoGo( V );
```

```
#I  there are 2 generators and 2 relators of total length 104
gap> TzWhit( V );
#I  new generator is _x9
gap> V;
<< presentation with 2 gens and 3 rels of total length 129 >>
gap> TzGoGo( V );
#I  there are 2 generators and 2 relators of total length 129
gap> TzWhit( V );
#I  new generator is _x10
gap> V;
<< presentation with 2 gens and 3 rels of total length 154 >>
```

The actual relators here are:

```
gap> G6 := FpGroupPresentation( V );
Group( f.4, _x10 )
gap> G6.relators;
[ f.4^2*_x10^-1*f.4^4*_x10^-1*f.4^5*_x10^-1*f.4^4*_x10^-1*f.4^3*
_x10*f.4*_x10^-1*f.4^4*_x10^-1*f.4^5*_x10^-1*f.4^4*_x1\
0^-1*f.4^6*_x10^-1*f.4^4*_x10^-1*f.4^5*_x10^-1*f.4^4*_x10^-1*f.4^4,
f.4^-3*_x10*f.4^-6*_x10*f.4^-4*_x10*f.4^-5*_x10*f.4^-4*_x10*f.4^-5*_x10*f.
^2*_x10^-1*f.4^4*_x10^-1*f.4^5*_x10^-1*f.\
4^4*_x10^-1*f.4^3*_x10*f.4^-4*_x10*f.4^-6*_x10*f.4^-4*_x10
```

```
*f.4^-5*_x10*f.4^-4*_x10*f.4^-2 ]
```

It would seem from this example (and from numerous other examples) that this definition of group presentation simplicity gives, at best, a minimal advantage over other standard presentation simplification techniques. On the other hand, the computational simplicity of our definition of presentation simplicity is quite appealing.

Chapter 3

Some invariant theory of the symmetric group

In this chapter, we present an algorithm that can be used as part of a normal-form algorithm for $R(F_n)$. Our motivation, roughly, is as follows. We observe that S_n acts on the Gonzalez-Montesinos relations (1.13 through 1.17 on page 16) in the natural way, and that a complete set of orbit representatives has 4 polynomials, for each $n > 4$. Following [115] we may describe the points in $\mathbb{V}(I_n) = X(F_n)$ by examining separately a Grobner basis for these polynomials, and a Grobner basis for the ideal of the orbit variety of $\mathbb{C}^{(n^3+5n)/6}$ modulo this representation of S_n. Although this approach works for small n, it quickly bogs down in the complexity of finding and manipulating the invariant ring of this $\frac{1}{6}(n^3+5n)$-dimensional permutation representation of S_n. But, as we shall see, some careful use of "constructive" Polya theory lets us extend this method considerably. More precisely, we are able to give a practical normal-form algorithm for the invariant ring of these representations of S_n. We do not however claim that this would be a practical method to find the normal form of an an element of $R(G)$, for a general finitely-presented group G.

An $n \times n$ complex matrix $[a_{ij}]$ acts on $\mathbb{C}[x_1, x_2, \ldots, x_n] = \mathbb{C}[X]$ by transforming generators as follows: $x_i^{[a_{ij}]} = \sum_j a_{ji} x_j$. the action being extended to all of $\mathbb{C}[X]$. The fixed points in $\mathbb{C}[X]$ under the action of a complex matrix group G form a $\mathbb{C}$-algebra, denoted by $\mathbb{C}[X]^G$, the "ring of invariants" of G. Rings of invariants of finite groups received much detailed study during this century. We mention particularly that $\mathbb{C}[X]^G$ is finitely generated (the "Hilbert finiteness theorem," see [39, section 1.4.1]) and that $\mathbb{C}[X]^G$ is Cohen-Macaulay (the Hochster-Eagon theorem [66]). When encountering a finitely generated algebra, we naturally ask whether we can list or describe a set of its generators; whether there exists a set of generators with nice properties; and even whether we may write down such a set.

The question of the existence of a procedure for constructing a generating set for $\mathbb{C}[X]^G$ was solved by Noether [97] who found a degree bound for a certain generating set of $\mathbb{C}[X]^G$. She then showed that this implies that there is a finite set of polynomials, whose image under the Reynolds operator generates all of $\mathbb{C}[X]^G$. (We recall the definition of the Reynolds operator below.) In this chapter, we will give a *canonical basis* (in the sense of Robbiano and Sweedler, whose work we briefly survey) for the ring of invariants of a particular representation of the symmetric group. We will also present a very explicit normal-form algorithm that uses this canonical basis.

A "canonical basis" (also called a "sagbi basis") B of an algebra $A \subseteq \mathbb{C}[X]$, with respect to a term order $\prec$, is a generating set for A such that

$$< \mathrm{in}_{\prec} B > = < \mathrm{in}_{\prec} A > .$$

[110]. For example, consider $\mathbb{C}[X]^{S_n}$, the ring of symmetric functions, which are generated by

$$\begin{aligned}
\sigma_1 &= x_1 + x_2 + \cdots + x_n \\
\sigma_2 &= x_1x_2 + x_1x_3 + \cdots + x_{n-1}x_n \\
&\vdots \\
\sigma_n &= x_1x_2 \ldots x_n
\end{aligned}$$

the "elementary symmetric functions." (Here, we view S_n as the group of $n \times n$ permutation matrices; i.e. $\mathbb{C}[X]^{S_n}$ is the ring of invariants of the symmetric group with the obvious action.) The elementary symmetric functions form a canonical basis for $\mathbb{C}[X]^{S_n}$ with respect to degree-lexicographic order (indeed, with respect to any term order [110, an observation attributed to Sturmfels].)

A finite canonical basis for an algebra allows us to write normal forms modulo A, much as a Grobner basis allows us to write normal forms modulo an ideal. (When we write "modulo an algebra A," we mean of course modulo A as a vector space. Also, we will consistently write $<B>$ for the subalgebra generated by B; (B) for the ideal generated by B.) Instead of the familiar division algorithm of Grobner basis theory, in this algebra case we must use the "subduction" (**subalgebra reduction**) algorithm. The subduction algorithm proceeds as follows: Suppose that the algebra $B \subset \mathbb{C}[X]$ is generated by the set $\{f_1, f_2, \ldots, f_n\}$. To reduce $f \in \mathbb{C}[X]$ modulo $\langle B \rangle$, for $B = \{f_1, f_2, \ldots, f_n\}$: [116]

Algorithm 3.1 *Subduction Algorithm. Given* f *and* $B = \{f_1, f_2, \ldots, f_n\}$.

while $f \notin \mathbb{C}$ *and there are non-negative integers* $\{i_1, \ldots, i_r\}$ *so that*

$$\mathrm{in}_{\prec} f = c \cdot \prod_{j=1}^{r} \mathrm{in}_{\prec}(f_j)^{i_j}, \quad c \neq 0 \;\; do \tag{3.1}$$

output $cf_1^{i_1} f_2^{i_2} \cdots f_r^{i_r}$

replace f *by* $f - cf_1^{i_1} f_2^{i_2} \cdots f_r^{i_r}$

od

output f *as the remainder*

When $\{f_1, f_2, \ldots, f_n\}$ is a canonical basis, then this algorithm will always return a remainder of zero for any $f \in B$. Conversely, if a set $\{f_1, f_2, \ldots, f_n\}$ subduces each $f \in B$ to zero then the set $\{f_1, f_2, \ldots, f_n\}$ is a canonical basis [98]. The expressions $cf_1^{i_1} f_2^{i_2} \cdots f_r^{i_r}$ found in this procedure, which we think of as monomials in $\mathbb{C}[\text{"}f_i\text{"}]$, are called *superpositions,* and are analogous to the more familiar S–polynomials of Grobner basis theory.

The permutation group S_n has a natural representations as a group of $2^n \times 2^n$ matrices, and also as a group of $\binom{n}{m} \times \binom{n}{m}$ matrices. We present an explicit canonical basis for the invariant ring of these representations of the symmetric group S_n acting on the *power set* of $\{1, 2, \ldots, n\}$. In other words, we consider the representation by permutation matrices arising from this action of S_n, and let this matrix group act on

$$X_{\binom{n}{m}} := \{x_{12\ldots m}, x_{12\ldots(m-1)(m+1)}, \cdots, x_{(n-m+1)\ldots n}\}$$

by permuting the indices of the elements of $X_{\binom{n}{m}}$:

$$\sigma(x_{j_1,\ldots,j_m}) = x_{\{\sigma j_1,\ldots,\sigma j_m\}}$$

for $\sigma \in S_n$, $1 \leq j_1 < \ldots < j_m \leq n$.

First, let's fix some notation. As is customary, we denote by $S_n^{(m)}$ this representation of S_n acting pointwise on m-element subsets of $\{1 \ldots n\}$. Analogously, by $S_n^{\{m\}}$ we will denote the symmetric group S_n acting on subsets with m or fewer elements. (So, for example,

$$S_n^{(1)} = S_n^{\{1\}} = S_n \,,$$

$$S_n^{(1)} + S_n^{(2)} + \cdots + S_n^{(m)} = S_n^{\{m\}}$$

etc.)

We will henceforth write $\mathbb{C}[X]^{S_n^{(m)}}$ for $\mathbb{C}[X_{\binom{n}{m}}]^{S_n^{(m)}}$, and $\mathbb{C}[X]^{S_n^{\{m\}}}$ for $\mathbb{C}[X_{\binom{n}{m}} \cup \ldots \cup X_{\binom{n}{2}} \cup X_{\binom{n}{1}}]^{S_n^{\{m\}}}$. In [114], Stanley introduced an ingenious scheme for the construction of a set of generators for

$$\mathbb{C}[X]^{S_n^{(m)}}$$

see also [90]. We call a hypergraph with exactly m vertices on each hyperedge an "m-hypergraph". We take the convention that graphs (resp. hypergraphs) are without loops, but may have multiple edges (resp. hyperedges.) Encode each monomial with coefficient 1,

$$X^{\alpha}_{\binom{n}{m}} := \prod_{1 \leq i_1 < \cdots < i_m \leq n} x_{ij\ldots k}^{\alpha(i_1, i_2, \ldots, i_m)},$$

with the m-hypergraph on vertices $\{1, 2, \ldots, n\}$, where there are exactly $\alpha(i_1, i_2, \ldots, i_m)$ hyperedges on the m vertices $1 \leq i_1 < \cdots < i_m \leq n$. Let $* = *_{S_n^{(m)}} : \mathbb{C}[X] \to \mathbb{C}[X]$ be the so-called "Reynolds operator" of the matrix group $S_n^{(m)}$, which averages the action of a group on a polynomial:

$$*f = \frac{1}{n!} \sum_{g \in S_n^{(m)}} f^g.$$

Then $*f \in \mathbb{C}^{S_n^{(m)}}$, and furthermore any f in the ring of invariants may be written uniquely as a $\mathbb{C}$-linear combination of Reynolds operators of monomials. By extending linearly the identification of hypergraphs with vertices from $\{1, 2, \ldots, n\}$ with monomials in $\mathbb{C}[X]$ to the free vector space generated by these graphs, Merris and

Watkins observed:

Proposition 3.2 *The Reynolds operator, restricted to the free vector space* M *generated by a complete set of representatives of non-isomorphic hypergraphs with k hyperedges on vertex set* $\{1, 2, \ldots, n\}$, *is a vector space isomorphism from* M *to the space* $(\mathbb{C}[X]^{S_n^{(m)}})_k = \{f \in \mathbb{C}^{S_n^{(m)}} \mid \deg f = k\}$.

Proof: see [90].

(By the way, Merris and Watkins were motivated by some computational problems in Polya theory. For example, by calculating the permutation character of $G = S_n^{(2)}$, and invoking Molien's theorem, which expresses the Hilbert series of $\mathbb{C}[X]^G$ in terms of the characteristic polynomials of the elements of G, they verified a generating function for a_k^n, the number of nonisomorphic graphs on n vertices with k edges:

$$\begin{aligned} \sum_{k=0}^{\infty} a_k^n z^k &= \frac{1}{n!} \sum_{\sigma \in S_n^{(2)}} \frac{1}{\det(I - z\sigma)} \\ &= \frac{1}{n!} \sum_{\sigma} (1 - z^a)^{-1} (1 - z^b)^{-1} \cdots \end{aligned}$$

where $a, b, \ldots$ are the lengths of disjoint cycles of permutation σ.)

For the convenience of the reader, we briefly recall some classical invariant theory. A recent survey of algorithmic invariant theory can be found in [115]. An important fact about invariant rings is that their degree doesn't depend much on the action of the group:

Theorem 3.3 *Let* $G < GL(\mathbb{C}^d)$ *be a finite matrix group. Then any* $d + 1$ *elements* *of* $\mathbb{C}[x_1, \ldots x_d]^G$ *are* $\mathbb{C}$*-algebraically dependent. There exist* d *independent elements of* $\mathbb{C}[x_1, \ldots x_d]^G$*, thus* $\dim \mathbb{C}[X]^G = d$.

PROOF: This is essentially theorems I and II of chapter XVII of Burnside's treatise [23]. Our proof basically follows his. Another proof may be found in [115, Theorem 2.1]. By the Hilbert finiteness theorem, $\mathbb{C}[x_1, \ldots x_d]^G$ is a finitely generated integral domain. Thus, by the Noether normalization theorem, $\dim \mathbb{C}[x_1, \ldots x_d]^G$, the maximum length of a chain of prime ideals, is the length of any maximal chain of prime ideals. Since G is finite, $\mathbb{C}[x_1, \ldots x_d]^G$ is Cohen-Macaulay. We exhibit a chain of length d, which is the longest possible chain in a Cohen-Macaulay subring of $\mathbb{C}[x_1, \ldots x_d]^G$.

Recall that if $I, J \subset R$ are ideals, $I + J$ is the smallest ideal that contains both I and J. Define

$$\begin{gathered} I_0 = (*x_1) \\ I_1 = (*x_1 x_2) + I_0 \\ \vdots \\ I_{d-1} = (*x_1 \cdots x_d) + I_{d-2} \\ I_d = \mathbb{C}[X]^G. \end{gathered}$$

These ideals evidently are a filtration for $\mathbb{C}[x_1, \ldots x_d]^G$. Recall that the set of *associated primes* $\mathrm{Ass}(I)$ of an ideal $I \subset R$ is the set of prime ideals of R which annihilate some element of the module R/I. For each i, choose minimal $J_i \in \mathrm{Ass}\, I_i$. For $1 \leq i < d$, I_i contains $*x_1 x_2 \cdots x_d \notin I_{i-1}$. Also, I_d contains $1 \notin I_{d-1}$. So we have the maximal chain of prime ideals

$$J_0 \lneq J_1 \lneq \cdots \lneq J_{d-1} \lneq I_d = \mathbb{C}[X]^G$$

and thus $\dim \mathbb{C}[x_1, \ldots, x_d]^G = d$. ◊

(An interesting discussion of this result may be found in [114].)

Given a permutation group, let us say the permutation group $S_n^{(2)}$, then clearly any symmetric polynomial lies in the invariant ring $\mathbb{C}[X]^{S_n^{(2)}}$. The elementary symmetric polynomials

$$\begin{aligned}
\sigma_1 &= x_{12} + x_{13} + \cdots + x_{(n-1)n} \\
&= \sum x_{ij} \quad = *x_{12} \\
\sigma_2 &= x_{12}x_{13} + \cdots \\
&= \frac{1}{2} * (x_{12}x_{13}) + \frac{1}{2} * (x_{12}x_{34}) \\
&\vdots
\end{aligned}$$

generate the symmetric polynomials (and indeed, as we have noted, are a canonical basis for them.) It is well-known that $\mathbb{C}[X]^{S_n^{(2)}}$ is a free module over the ring $< \sigma_1, \ldots >$ (the existence of such a "regular system of parameters" is a popular way to define Cohen-Macaulayness.) We call these homogeneous invariants the *primary invariants* of the group $S_n^{(2)}$, and seek a finite set of invariants, called the *secondary invariants*, which together with the primary invariants generate the whole invariant ring. (Such a set of secondary invariants exists, since invariant rings of finite groups are Cohen-Macaulay.) By corollary 2.7.10 of [115] the set of polynomials

$$S = \{*m \mid m \text{ is a descent monomial of } S_{\binom{n}{2}}\}$$

are a set of secondary invariants of $S_n^{(2)}$. A "descent monomial" is a monic monomial which is associated to a permutation. Since $S_{\binom{n}{2}}$ has $\binom{n}{2}!$ members, this set of secondary invariants is clearly less than optimal. To "pick out" a minimal set of secondary invariants, one may use the Hilbert series of $\mathbb{C}[X]^{S_n^{(2)}}$, which tells us the number of algebraically independent invariants of a given degree. (Hilbert series of invariant rings may be found by a fundamental theorem of Molien, and are usually called *Molien series* in his honor.) The Molien series of G $\mathbb{C}[X]^{S_n^{(2)}}$ is, by Molien's theorem,

$$\begin{aligned}
\frac{1}{n!}\sum_{M\in S_n^{(2)}} \frac{1}{\det(I - zM)} &= \frac{1}{n!}\sum_{M\in S_n^{(2)}} \prod_{i=1}^{\binom{n}{2}} (1 - z^i)^{-l_i(M)} \\
&= \frac{1}{n!}\sum_{\sigma\in S_n} \prod_{i=1}^{\binom{n}{2}} (1 - z^i)^{(2-n)(l_i(\sigma))}
\end{aligned}$$

(where $l_i(M)$ is the number of cycles of length i in M)

$$= \sum_{a \vdash n} \frac{\prod_{i=1}^{\mathrm{length}(a)} (1 - z^{a(i)})^{(n-2)}}{\prod_{i=1}^{\mathrm{length}(a)} (a(i)! i^{a(i)})},$$

$a \vdash n$ denoting that vector a is a partition of integer n.

We do not take this approach here. Instead, we present a canonical subalgebra basis for the ring $\mathbb{C}[X]^{S^{(2)}}$, and a version of the subduction algorithm which uses this (very large) basis implicitly.

Theorem 3.4 *Recall that we have associated labelled hypergraphs with monomials.*

a) Let S be a set of equivalence-class representatives of labelled hypergraphs partitioned by isomorphism on vertices $\{1, 2, \ldots, n\}$, with k m–hyperedges (resp., k hyperedges, each edge on k or fewer vertices.) Then $*S$ is a canonical basis for $\left\langle (\mathbb{C}[X]^{S_n^{(m)}})_k \right\rangle$ (the algebra generated by the set of degree k elements of the ring $\mathbb{C}[X]^{S_n^{(m)}}$) (resp. $\left\langle (\mathbb{C}[X]^{S_n^{[m]}})_k \right\rangle$) with respect to any term order. Furthermore, S contains a regular system of parameters $f \left\langle (\mathbb{C}[X]^{S_n^{(m)}})_k \right\rangle$ (resp. $\left\langle (\mathbb{C}[X]^{S_n^{[m]}})_k \right\rangle$) under the filtration induced by degree grading.

b) The image under $*$ of a set of representatives of all labelled hypergraphs up to isomorphism on vertices $\{1, 2, \ldots, n\}$ with $\binom{n}{m}$ or fewer hyperedges, each on m vertices (resp. m or fewer vertices) is a canonical basis for $\mathbb{C}[X]^{S_n^{(m)}}$ (resp. $\mathbb{C}[X]^{S_n^{[m]}}$) with respect to any term order.

PROOF: Fix any term order $\prec$.

a) Let f be in $(\mathbb{C}[X]^{S_n^{\{m\}}})_k \setminus \{0\}$, for $k > 0$. Then $(\mathrm{in}_\prec f)/LC_\prec(f)$ (where $LC_\prec(f)$ denotes the leading coefficient of f with respect to $\prec$) is a hypergraph with k m-hyperedges, since $\deg \mathrm{in}_\prec f = k$. Since f is in $\mathbb{C}[X]^{S_n^{\{m\}}}$,

$$f - LC(f) \cdot *(\mathrm{in}_\prec f/LC(f)) = f_1$$

is in $\mathbb{C}[X]^{S_n^{\{m\}}}$, and furthermore f_1 is again homogeneous of degree k, and has fewer terms than f (since f is not constant.) Continuing this process, we eventually reach a constant, which is zero since f is homogeneous of positive degree. Thus, the set of monomials associated to the set of hypergraphs with n vertices and k m-hyperedges generate $< (\mathbb{C}[X]^{S_n^{\{m\}}})_k >$ as an algebra. Also, each step in our reduction is a step in the subduction algorithm, thus these hypergraphs are a canonical basis for

$$< (\mathbb{C}[X]^{S_n^{\{m\}}})_k > .$$

If $f \in (\mathbb{C}[X]^{S_n^{\{m\}}})_k \setminus \{0\}$, $k > 0$, then $(\mathrm{in}_\prec f)/LC_\prec(f)$is a hypergraph with k hyperedges, each with m or fewer vertices. The proof proceeds as above.

b) Since $\mathbb{C}[X]^{S_n^{\{m\}}} = \bigoplus_k (\mathbb{C}[X]^{S_n^{\{m\}}})_k$, we have by part (a) an (infinite) canonical basis for $\mathbb{C}[X]^{S_n^{\{m\}}}$: the image under $*$ of all hypergraphs (up to isomorphism) on $\{1 \ldots n\}$, with each hyperedge on m vertices. We would like to express each $f \in (\mathbb{C}[X]^{S_n^{\{m\}}})_D$, $D > \binom{n}{m}$ as an algebraic combination of $\{*f_l \mid f_l \in (\mathbb{C}[X]^{S_n^{\{m\}}})_l, l \leq \binom{n}{m}\}$. Now products of Reynolds operators of simple hypergraphs (hypergraphs without multiple hyperedges) must have multiple hyperedges, thus the set of simple hypergraphs, each of whose Reynolds operator is contained in some $(\mathbb{C}[X]^{S_n^{\{m\}}})_l, l \leq \binom{n}{m}$, contains $\binom{n}{m}$ algebraically independent elements. Statement (b) is now true by theorem 3.3. $\diamondsuit$

Although we are most interested in using the theorem in the case when $m \leq 3$, it is nonetheless unfortunate that these canonical bases grow so quickly (there are *many* 3-hypergraphs on 9 vertices.) For this reason, any computer algorithm which operates on an explicit canonical basis for $\mathbb{C}[X]^{S_n^{\{3\}}}$ will be confined, for practical reasons, to small n. But the subduction algorithm itself only requires that we be able to exhibit elements of the canonical basis satisfying condition 3.1.

Algorithm 3.5 *Algorithm (Subduction for $\mathbb{C}[X]^{S_n^{\{m\}}}$.) Given: f in $\mathbb{C}[X]$,*

Output: a normal form for $f \mod \mathbb{C}[X]^{S_n^{\{m\}}}$, together with a sequence of superpositions

$F := f$.

done := false

repeat

if F is constant then

output F as a term of the expression for f

done := true

elseif there are hypergraphs $\{\Gamma_i\}$ so that:

a) $\mathrm{in}_{\prec} * \Gamma_i = \Gamma_i$

b)each Γ_i *has* $\binom{n}{m}$ *edges or less, and*

c) $\mathrm{in}_{\prec} F = \prod_i \Gamma_i$

then

output $LC(f) \cdot \prod_i * \Gamma_i$ *as a term of the superposition*

$F := f - LC(f) \cdot \prod_i * \Gamma_i$

else

done := true

fi

until done

output F *as a normal form for* f.

We have considerable leeway in our choice of $\{\Gamma_i\}$ at each step. But part (d) of the theorem (together with the theory of canonical bases - see e.g. [98] or [92]) guarantee that the algorithm will terminate with a normal form for $f \mod \mathbb{C}[X]^{S_n^{\{m\}}}$.

As an example, let's look at

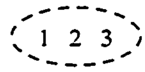
$$f_1 = x_{12}^2 + x_{13}^4 x_{123} + 2 \in \mathbb{C}[X_4] \tag{3.2}$$

with lex term order, $x_1 \succ x_2 \succ \ldots \succ x_{123} \succ \ldots \succ x_{234}$. We have

LT(f_1)/LC(f_1) =  , so we write f_1 as

$$f_1 = *(x_{12}^2) + (f_1 - *(x_{12}^2))$$

$$= *(x_{12}^2) + \underbrace{-x_{13}^2 - x_{14}^2 - x_{34}^2 + x_{13}^4 x_{123} + 2}_{f_2}$$

Now $LT(f_2)/LC(f_2)$ $=$  (where  denotes the hyperedge on vertices $\{1, 2, 3\}$) cannot be written as hypergraphs such that $\mathrm{in}_{\prec} \Gamma = \mathrm{in}_{\prec} *\Gamma$, and thus f_2 is the normal form for $f_1 \mod \mathbb{C}[X_{\binom{n}{2}}, X_{\binom{n}{3}}]^{S_n^{\{m\}}}$.

How may we recognize when an appropriate choice $\{\Gamma_i\}$ exists? In other words, how may we recognize that an appropriate superposition 3.1 is available? (We restrict ourselves to hypergraphs with the same number of vertices on each hyperedge, since all of the hyperedges in the orbit of a hyperedge lie on the same number of vertices.) One reasonable idea is to greedily choose higher-weight hyperedges to form a maximal graph Γ_i so that $\mathrm{in}_{\prec} \Gamma_i = \mathrm{in}_{\prec} *\Gamma_i$ (let's call such a Γ_i an "initial graph" (or "initial hypergraph.") If this can be done, we have reduced the problem to a smaller graph $LT(f_2)/(LC(f_2) \cdot \Gamma_i)$. But for the monomial

$$f = x_{12}^2 x_{13}^2 x_{14}$$

with lex order $\prec$, this greedy algorithm would first choose $\Gamma_1 = x_{12}^2$, and then fail for $f/x_{12}^2 = x_{13}^2 x_{14}$, which is worrisome since f is already an initial graph. (Clearly, on the other hand, if $\prec$ were degree ordering, this greedy algorithm *would* work for f) . Let us examine this situation more closely, using the language of greedoids [83].

A "greedoid" is a sort of generalized matroid. Consider the following five conditions on the ordered pair $(E, \mathcal{G} \subseteq \text{PowerSet}(E))$:

1. $\emptyset \in \mathcal{G}$

2. (Accessibility Axiom) $X \in \mathcal{G}, X \neq \emptyset$ implies that there is $x \in X$ such that $X \setminus \{x\} \in \mathcal{G}$

3. (Exchange Axiom) $X, Y \in \mathcal{G}$ $|X| = |Y| + 1$ implies that there is $x \in X \setminus Y$ such that $Y \cup \{x\} \in \mathcal{G}$.

4. If $A \subset B \in \mathcal{G}$, then $A \in E$.

5. $\mathcal{G}$ is closed under union.

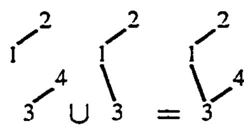
If the pair $(E, \mathcal{G} \subseteq \text{PowerSet}(E))$ satisfies conditions (1) - (3), it is called a (*simple*) *greedoid*. If the pair satisfies conditions (1) - (4), it is called a *matroid*. If the pair satisfies conditions (1) - (3) and (5), it is called an *antimatroid*. The elements of $\mathcal{G}$ are called the *feasible sets* of the greedoid. A maximal feasible set is called a *basis*. If we furthermore allow elements of $\mathcal{G}$ to be multisets, rather than sets, the greedoid is called *non-simple*.

See [22] or [83] for a survey of greedoids and their applications to combinatorial optimization.

The multiplicity-free initial graphs (in other words, the square-free monomials) form a greedoid $\mathcal{G}$ as (hyper-)edge sets of $\{1..n\}$ under $\subseteq$. We declare $\oslash$ to be the initial graph of 1, so that $\oslash \in \mathcal{G}$. The inclusion poset of $\mathcal{G}$, when $m = 2$, and $\prec$ is lex order, $x_{12} \succ x_{13} \succ \cdots$ is shown in figure 3-1.

We have drawn the graphs in 3-1 so that if Γ_1 is to the left of Γ_2, then $\Gamma_1 \prec \Gamma_2$. Note that

1) $\mathcal{G}$ is not a matroid when $n > 2$. $\mathcal{G}$ is not closed under union (e.g.



$\notin \mathcal{G}$.) So $\mathcal{G}$ is not an antimatroid, when $n > 3$.

2) K_n, the complete graph, is the unique basis (maximal feasible set) of $\mathcal{G}$. Thus $\text{rank}(\mathcal{G}) = \text{rank}(K_n) = n!$.

3) By construction, any objective function which chooses $\prec$-greater graphs would be compatible with this greedoid structure, thus the greedy algorithm applied to $\mathcal{G}$ will find a basis which maximizes this function - in other words, K_n.

If $\Gamma \in \mathcal{G}$, then the interval $[\oslash, \Gamma]$ is a greedoid where the greedy algorithm produces Γ, which is the unique basis. Let Γ be a graph. We define a greedoid $\mathcal{G}(\Gamma)$ as follows: If Γ'' is a multiplicity-free initial graph subgraph of Γ, which occurs τ times in Γ, then

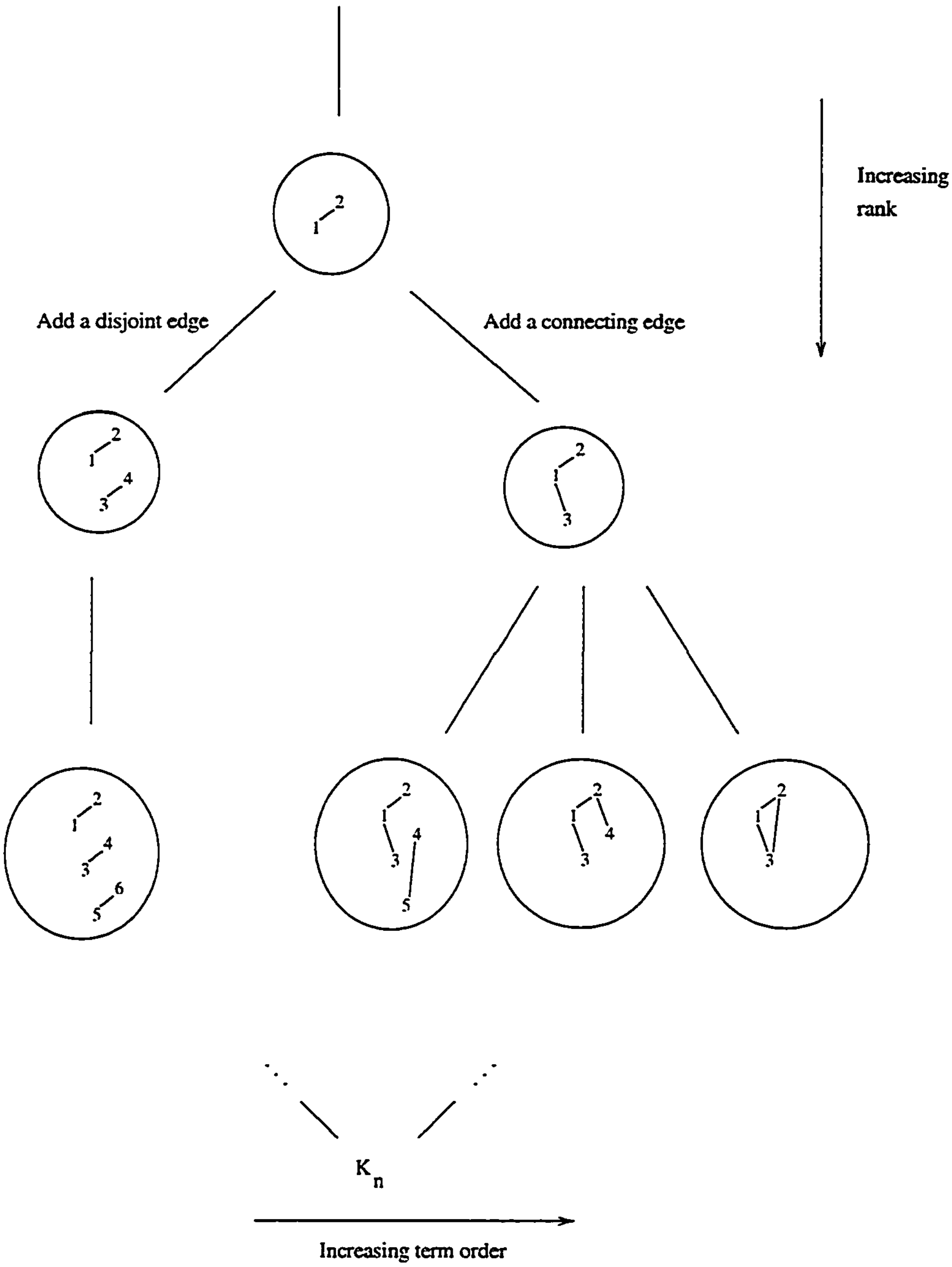


Figure 3-1 Inclusion poset for $\mathcal{G}$

the multiset (subgraph)

$$\{\underbrace{e, e, \ldots, e}_{\tau \text{ times}} | e \text{ an edge of } \Gamma\}$$

is a feasible set of $\mathcal{G}(\Gamma)$, which we call the *initial graph branching matroid* of Γ. (If Γ contains no initial graph, we define $\mathcal{G}(\Gamma) = \oslash$.) Non-simple greedoids maximize compatible objective functions with the greedy algorithm in the same way that simple greedoids do - in this case to provide us with a $\prec$-maximal basis (included initial graph) of Γ. Indeed, this greedy algorithm is just a mutation of Prim's 'visit an unvisited node first' procedure to find a maximal-weight spanning tree in a graph.

The following theorem gives us a termination condition for the subduction algorithm 3.5.

Theorem 3.6 *Let Γ_0 be the basis found by greedily searching for a maximal initial graph subgraph of Γ, in the greedoid $\mathcal{G}(\Gamma)$. If*

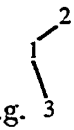
$$\Gamma \setminus \Gamma_0$$

is non-empty and contains no initial graph, then condition 3.1 in the subduction algorithm for $\mathbb{C}[X]^{S_n}$ cannot be satisfied for $F = \Gamma$.

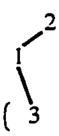
PROOF: Suppose on the contrary that there exist initial graphs $\Gamma_1, \Gamma_2, \ldots \Gamma_l$ so that

$$\Gamma = \Gamma_1 \Gamma_2 \ldots \Gamma_l.$$

Then, since $\prec$ is a term order, Γ is an initial graph, so $\Gamma \in \mathcal{G}(\Gamma)$, and $\mathcal{G}(\Gamma)$ is just the interval $[\oslash, \Gamma]$ in $\mathcal{G}(K_n)$. But then $\Gamma \setminus \Gamma_0 = \oslash$, since $\Gamma$is itself the unique basis in $[\oslash, \Gamma]$. ◇

If we encode a graph by the $\frac{n^2-n}{2}$-tuple consisting of the entries in the upper triangular part of the graph's adjacency matrix, (e.g.  has adjacency matrix

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix},$$

so we write code() $= (1, 1, 0)$), then the task of greedily searching $\mathcal{G}(\Gamma)$ with respect to lex-order $\prec$ is:

repeat

new := largest code (considered as a binary integer) obtained

by changing the first zero which has no ones to the right of

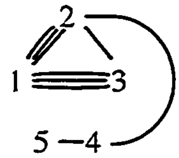
it to one

until new is maximal in Γ and is not isomorphic

to any graph previously considered

The number of operations that must be performed in this loop is dominated by the difficulty of checking each of the graphs constructed for isomorphism with previous graphs. Consider the monomial associated to graph

f =



.

If m contains a spanning initial graph subgraph, then this graph has three vertices (1, 2, and 4) of degree 2, one vertex (2) of degree 3, and one vertex (5) of degree 1. If greedily adding an edge would cause us to exceed one of these degree bounds, then we have found our basis of $\mathcal{G}(\Gamma)$. R. Read [107], [78] has considered this problem in the context of the problem of listing all simple graphs on n vertices up to isomorphism. He showed that, for lex order, it suffices in the above loop to ensure that the code of new is *canonical* - is, maximal under all permutations of the vertices. (He coined the term *orderly algorithm* for this sort of graph-cataloging procedure.)

To greedily search $\mathcal{G}(\Gamma)$, we start with $\oslash$, and add the single edge 1-2. We add edge 1-3, which does not exceed our vertex degree bounds, and which is canonical. We may not add edges 1-4 or 1-5, since degree(1) now equals 2. We add edge 2-3, which is canonical. We add edge 2-4, which is canonical. We may not add 2-5. Adding edge 4-5 completes our search.

Checking a graph for canonicality can take as many as $n!$ comparisons. But for most graphs, when $\preceq$ is lex order, the situation is not that dire –the new graph is by definition canonical.

We summarize our discussion with an efficient normal form algorithm for $\mathbb{C}[X]^{S_n^{(2)}}$.

Algorithm 3.7 *Algorithm: Subduction for* $\mathbb{C}[X]^{S_n^{(2)}}$ *with respect to lexicographic order. Given:* $f \in \mathbb{C}[X]$

Output: A normal form for $f \mod \mathbb{C}[X]^{S_n^{(2)}}$.

F *:=* f

done := false

repeat

if $F \in \mathbb{C}$ *then*

done := true

else (the "orderly algorithm")

new := (0, 0, $\cdots$, 0)

newer := new

success := false

repeat

change the first zero in new, which is to the right of all ones, to one

if this new edge is in F *then*

success := true

new := newer

fi

until success

or

(no edge that would be represented by a position to the right of

rightmost one is in F*)*

if success then

m *:= multiplicity of new in* F

$\mathsf{F} := \mathsf{F} - LC(\mathsf{F}) \cdot (*\mathsf{new})^{\mathsf{m}}$

else

done := true

fi

fi

until done

output F *as a normal form for* f

Chapter 4

Divisibility properties of trace polynomials

4.4.1 The shifted trace polynomials; strong and weak divisibility

In this chapter, we give some divisibility properties of character relations and trace polynomials.

Theorem 4.1 *(horowitz) Let* $w_1 = g_1^k$, $w_2 = g_1^l$, $g_1 \in F_n$. *If* $\gcd(k.l) = 1$, *then*

$$\gcd(t_{w_1} - 2, t_{w_2} - 2) = t_{g_1^{\gcd(k.l)}} - 2.$$

In other words, the sequence of polynomials $\{t_{g_1^i} - 2\}_i$ *is a strong divisibility sequence.*

We begin our proof with a version of Theorem B of [94]

Proposition 4.2 *For* $z \in \mathbb{N}$, *let* $\{f_n\}$ *be the sequence of integers determined by*

$$\begin{aligned} f_{n+2} &= a(f_{n+1} + c) + b(f_n + c) - c \qquad (4.1) \\ f_0 &= 0 \\ f_1 &= z. \end{aligned}$$

Suppose that a, b, c *are pairwise relatively prime.*

Then, for sufficiently large prime d, *there is an integer* k *so that*

$$k|n \Rightarrow d|f_n.$$

Proof: Set $\mathfrak{I} = \{n \mid d|f_n\}$. Since $f_0 = 0$, $\mathfrak{I}$ is not empty. If $\mathfrak{I} \neq \{0\}$, then let n be the element of $\mathfrak{I}$ with smallest absolute value. Then, following [94], there is an integer d' so that $d > d'$ implies

$$f_{n+k} + (-b)^k f_{n-k} \equiv 0 \mod d.$$

Thus, $2n, 3n, \ldots \in \mathfrak{I}$, and likewise $-n, -2n, \ldots \in \mathfrak{I}$. $\diamondsuit$

Corollary 4.3 *(to proposition)* $f_{\gcd(m,n)} = \pm C \gcd(f_m, f_n)$.

Proof: (of theorem) We apply the proposition with $c = 2$, $z = t_1 = \operatorname{tr} g_1$. By the recurrence 1.7, and corollary 1.5, $(\operatorname{tr} x^n) - 2 = f_n(t_x)$ for any word x. Also, $\deg_{t_x}(\operatorname{tr} x^n) = n$, so $\deg_{t_1}(t_{g_1^{\gcd(k.l)}}) \leq \gcd(k.l)$. For an infinite number of primes d, we have by the above corollary

$$\gcd(t_{w_1} - 2|_{t_1=z}, t_{w_2} - 2|_{t_1=z} \equiv \pm(t_{1^{\gcd(k.l)}} - 2)|_{t_1=z} \mod d.$$

Since each of the polynomials $t_{w_1} - 2$, $t_{w_2} - 2$, and $t_{1^{\gcd(k.l)}} - 2$ are monic, the theorem follows. $\diamondsuit$

Let $\{a_n\}$ be a sequence of elements of a unique factorization domain. $\{a_n\}$ is called a **divisibility sequence** [119] if $a_n | a_m$ implies that $n|m$. a_n is called a **strong divisibility sequence** if, in addition $a_{\gcd(m,n)} = \pm \gcd(a_m, a_n)$. Corollary 4.3 states that the shifted trace polynomials f_n are a strong divisibility sequence. For each integer $n \geq 0$, let h_n be the polynomial

$$h_n = \frac{f_n - 2}{\operatorname{lcm}\{f_i - 2 \,|\, i|n\}}.$$

The sequence of polynomials $h_n(x)$ was first considered, in a different context, by Horadam, Loh and Shannon [67], who noted that $f_n(x)$ is a divisibility sequence. For positive integer a, define a sequence of polynomials

$$g_{n+2} = x g_{n+1} - g_n$$

$$\begin{aligned} g_0 &= a \\ g_1 &= x \end{aligned}$$

with associated $H_n = g_n - a$,

$$h_n = \frac{H_n}{\operatorname{lcm}\{H_n \,|\, i|n\}}.$$

	$\operatorname{tr} y^n - 2$
1	$x-2$
2	$(x-2)(x+2)$
3	$(x-2)(x+1)^2$
4	$(x-2)(x+2)x^2$
5	$(x-2)(x^2+x-1)^2$
6	$(x-2)(x+2)(x-1)^2(x+1)^2$
7	$(x-2)(x^3+x^2-2x-1)^2$
8	$(x-2)(x+2)(x^2-2)^2x^2$
9	$(x-2)(x+1)^2(x^3-3x+1)^2$
10	$(x-2)(x+2)(x^2-x-1)^2(x^2+x-1)^2$
11	$(x-2)(x^5+x^4-4x^3-3x^2+3x+1)^2$
12	$(x-2)(x+2)(x-1)^2(x+1)^2(x^2-3)^2x^2$
13	$(x-2)(x^6+x^5-5x^4-4x^3+6x^2+3x-1)^2$
14	$(x-2)(x+2)(x^3-x^2-2x+1)^2(x^3+x^2-2x-1)^2$
15	$(x-2)(x+1)^2(x^2+x-1)^2(x^4-x^3-4x^2+4x+1)^2$

Table 4.1 The shifted trace polynomials

We aim to generalize the results of [67] to consider the notion of "weak divisibility." Let $k \in \mathbb{Z}$, and let $\{a_n\}$ be a sequence in a unique factorization domain. We call $\{a_n\}$ a "k—weak divisibility sequence" if, for all $l > 0$ such that $k|l$, a_k divides a_l. Clearly, divisibility sequences are weak divisibility sequences for all k.

Since we have seen that $a = 2$ implies that H_n is a divisibility sequence, we ask: for what $a \in \mathbb{C}$ is H_n a k—weak divisibility sequence? In the polynomial ring $\mathbb{C}[x, a]$, let $<$ be degree-lexicographic order [31], with $a < x$. The ideal $I = (x - a)$ is principal, and so $\{x - a\}$ is a Grobner basis for I. Let $r_n(x, a) \in \mathbb{C}[x, a]$ be the normal form of $H_n(x, a)$ mod I. If a term of r_n included the variable x to a positive power, then the leading term of r_n would also include x to a positive power, and we could reduce r_n by $x - a$. But this would contradict the assumption that r_n is the normal form of a polynomial. Thus we have

	$\operatorname{tr} y^n$
1	x
2	x^2-2
3	$(x^2-3)x$
4	x^4-4x^2+2
5	$(x^4-5x^2+5)x$
6	$(x^2-2)(x^4-4x^2+1)$
7	$(x^6-7x^4+14x^2-7)x$
8	$x^8-8x^6+20x^4-16x^2+2$
9	$(x^2-3)(x^6-6x^4+9x^2-3)x$
10	$(x^2-2)(x^8-8x^6+19x^4-12x^2+1)$
11	$(x^{10}-11x^8+44x^6-77x^4+55x^2-11)x$
12	$(x^4-4x^2+2)(x^8-8x^6+20x^4-16x^2+1)$
13	$(x^{12}-13x^{10}+65x^8-156x^6+182x^4-91x^2+13)x$
14	$(x^2-2)(x^{12}-12x^{10}+53x^8-104x^6+86x^4-24x^2+1)$
15	$(x^2-3)(x^4-5x^2+5)(x^8-7x^6+14x^4-8x^2+1)x$

Table 4.2 The trace polynomials factored over the integers

$r_n(x,a) \in \mathbb{C}[a]$. The roots of the polynomials r_n thus are the complex numbers a such that H_n is an n—weak divisibility sequence. The first few r_n are listed in table 4.3.

4.4.2 A discriminant identity

We now fix some notation for the rest of the chapter.

Notation 4.4 *We denote by $f_n(x)$ the $(n)^{th}$—degree polynomial*

$$\sum_i (-1)^{\lfloor i/2 \rfloor} \binom{n - \lfloor (i+1)/2 \rfloor}{\lfloor i/2 \rfloor} x^i.$$

We denote by $\alpha_{n,i}$ the coefficient of x^{n-i+1} in f_n. When n is clear, we write α_i instead

$$0$$
$$-2a+a^2$$
$$-2a-a^2+a^3$$
$$-2a^2-a^3+a^4$$
$$2a^2-3a^3-a^4+a^5$$
$$-2a+3a^2+3a^3-4a^4-a^5+a^6$$
$$-2a-3a^2+6a^3+4a^4-5a^5-a^6+a^7$$
$$-4a^2-6a^3+10a^4+5a^5-6a^6-a^7+a^8$$
$$4a^2-10a^3-10a^4+15a^5+6a^6-7a^7-a^8+a^9$$
$$-2a+5a^2+10a^3-20a^4-15a^5+21a^6+7a^7-8a^8-a^9+a^{10}$$

Table 4.3 The remainder polynomials $r_n(a)$

r_1	0
r_2	$a(-2+a)$
r_3	$a(a+1)(-2+a)$
r_4	$a^2(a+1)(-2+a)$
r_5	$a^2(-2+a)(a^2+a-1)$
r_6	$a(a-1)(-2+a)(a+1)(a^2+a-1)$
r_7	$a(a-1)(-2+a)(a+1)(a^3+a^2-2a-1)$
r_8	$a^2(-2+a)(a^2-2)(a^3+a^2-2a-1)$
r_9	$a^2(a+1)(-2+a)(a^2-2)(a^3-3a+1)$
r_{10}	$a(a+1)(-2+a)(a^2+a-1)(-a-1+a^2)(a^3-3a+1)$
r_{11}	$a(-2+a)(a^2+a-1)(-a-1+a^2)(a^5+a^4-4a^3-3a^2+3a+1)$
r_{12}	$a^2(a-1)(-2+a)(a+1)(a^2-3)(a^5+a^4-4a^3-3a^2+3a+1)$
r_{13}	$a^2(a-1)(-2+a)(a+1)(a^2-3)(a^6+a^5-5a^4-4a^3+6a^2+3a-1)$
r_{14}	$a(-2+a)(a^3+a^2-2a-1)(a^3-a^2-2a+1)\ (a^6+a^5-5a^4-4a^3+6a^2+3a-1)$

Table 4.4 The remainder polynomials $r_n(a)$, factored over the integers

of $a_{n,i}$.

It is easy to show by induction that $f_{\lfloor (n-1)/2 \rfloor}(a)$ *divides* $r_n(a)$ *for all integers* $n \geq 0$. *(See table 4.4.)*

We have

$$\alpha_{2m,i} = \begin{cases} (-1)^K \binom{m+k}{2k}, & i = 2K+1 \\ (-1)^K \binom{m+k-1}{2k}, & i = 2K. \end{cases}$$

and an analogous formula for $\alpha_{2m+1,i}$.

We give a surprising formula for the discriminant of the polynomials $f_n(a)$:

Theorem 4.5 *If* $n \geq 1$, *the discriminant of* f_n, $\Delta(f_n)$, *is* $(2n+1)^{n-1}$.

Proof: To avoid an otherwise oppressive notation, we will first assume that n is divisible by 4, and write $n = 2m$. We indicate at the conclusion of the proof the minor changes needed when n is not divisible by 4. Our proof is in three parts. In part one, we show that $(2n+1)^{n-1} \mid \Delta(f_n)$. In part two, we show that $|\Delta(f_n)| \leq (2n+1)^{n-1}$. Finally, in part three, we show that $\Delta(f_n)$ has positive sign.

Part one: Our convention is that the Sylvester matrix of the resultant of two polynomials f, g looks like

$$\begin{pmatrix} \text{Coefficients of } f & \\ & \text{Coefficients of } f \\ & \vdots \\ \text{Coefficients of } g & \\ & \text{Coefficients of } g \\ & \vdots \end{pmatrix}$$

so, since $f_{2m}(x)$ is a polynomial of degree $2m$, the Sylvester matrix S of $\Delta(f_n)$ is:

$$\begin{array}{ccccc|ccccc}
\alpha_1 & \alpha_2 & \alpha_3 & \cdots & \alpha_{n-1} & \alpha_n & \alpha_{n+1} & 0 & \cdots & \\
 & \alpha_1 & \alpha_2 & \cdots & \alpha_{n-2} & \alpha_{n-1} & \alpha_n & \alpha_{n+1} & 0 & \cdots \\
 & & \vdots & & & & & \vdots & & \\
 & & & & \alpha_1 & \alpha_2 & \alpha_3 & \cdots & \cdots & \alpha_{n+1} \\
\hline
n\alpha_1 & (n-1)\alpha_2 & (n-2)\alpha_3 & \cdots & 2\alpha_{n-1} & \alpha_n & 0\alpha_{n+1} & \cdots & & \\
 & n\alpha_1 & (n-1)\alpha_2 & (n-2)\alpha_3 & \cdots & & & & & \\
 & & \vdots & \vdots & & (n-1)\alpha_2 & \cdots & \vdots & & \\
 & & & & 0 & n\alpha_1 & (n-1)\alpha_2 & \cdots & & \alpha_n \\
 & & \multicolumn{2}{c}{\underbrace{\hspace{3cm}}_{n-1}} & & & \multicolumn{2}{c}{\underbrace{\hspace{3cm}}_{n}} & &
\end{array}$$

Transpose the rows of S, so that each of the first $2m - 1$ rows coming from the coefficients of $f_n'(x)$ are followed by the corresponding row from $f_n(x)$

$$S' = \begin{pmatrix} 2m\binom{m}{m} & -(2m-1)\binom{m}{m-1} & -(2m-2)\binom{m-1}{m-1} & (2m-3)\binom{m-1}{m-2} & \cdots \\ \binom{m}{m} & -\binom{m}{m-1} & -\binom{m-1}{m-1} & \binom{m-1}{m-2} & \cdots \\ & 2m\binom{m}{m} & -(2m-1)\binom{m}{m-1} & -(2m-2)\binom{m-1}{m-1} & \cdots \\ & \binom{m}{m} & \binom{m}{m-1} & -\binom{m-1}{m-1} & \cdots \\ & & 2m\binom{m}{m} & -(2m-1)\binom{m}{m-1} & \cdots \\ & & \binom{m}{m} & -\binom{m}{m-1} & \cdots \\ & & & \vdots & \ddots \end{pmatrix} \tag{4.2}$$

Since we have assumed that $n \equiv 0 \bmod 4$, S' is obtained from S by an even number of row exchanges. So the determinant of this matrix is the determinant of S, which is a polynomial in m. An irreducible polynomial $p(m)$ divides the polynomial $(\det S)(m)$ if and only if each root r of this polynomial $p(m)$, when substituted for m in the above matrix, gives a matrix of determinant zero. If $S(r)$ is a matrix whose row rank is less than $2n - 1$, then there is a non-trivial linear combination of the rows which is the zero vector. Let us write down such a linear combination. There are $(n-1)$ triples of consecutive rows which look like

$$\begin{array}{ccc} (2m)\binom{m}{m} & -(2m-1)\binom{m}{m-1} & \cdots \\ \binom{m}{m} & -\binom{m}{m-1} & \cdots \\ (2m+1)\binom{m-1}{m} & \binom{m}{m} & \cdots \end{array} \tag{4.3}$$

We write each of the binomial coefficients as a polynomial as follows: for $k \geq 0$, $\binom{m}{k}$ is identified with the polynomial $\frac{1}{k!}m^{\underline{k}}$, and $\binom{m+k}{m-l}$ is rewritten as $\binom{m+k}{-l-k}$. For $k < 0$, $\binom{m}{k}$ is identified with the polynomial 0. (The reader should be aware that in general these polynomials yield the appropriate values for the binomial coefficients only when m is a positive integer.) In particular, when we write an expression like $\binom{m+k}{m-l}|_{m=a}$ we intend that a be substituted for m in the polynomial $\binom{m+k}{-l-k}$.

We claim that twice the first row in 4.3 plus the second equals the third, when $m = r = -\frac{1}{4}$. This is verified by direct calculation: since , for positive integers a, b :

$$A\binom{a}{b} - B\binom{a-1}{b-1} = 0$$

if and only if $Aa - Bb = 0$, we have

$$\left(2(2m-2i)\binom{m+i}{m-i} + \binom{m+i}{m-i}\right)\Big|_{m=-\frac{1}{4}} = (2m-2i+1)\binom{m+i-1}{m-i-1}\Big|_{m=-\frac{1}{4}}$$

for each i if and only if

$$(4m-4i+1)(m+i)\,|_{m=-\frac{1}{4}} = (2i)(2m-2i+1)\,|_{m=-\frac{1}{4}}$$

which is clearly true. So

$$2(2m-k+1)\alpha_k + \alpha_k = (2m-k+2)\alpha_{k-1}$$

for even k; and similarly for odd k. Thus, $\det(S)(m)$ evaluated at $m = -\frac{1}{4}$ has determinant zero, since adding twice the first row and (-1) times the third row to the second row yields a row of zeroes. But likewise, twice the third row plus the fourth equals the fifth, etc. There are $(n-1)$ rows of zeros in this new matrix; so, $m = -\frac{1}{4}$ is a root of multiplicity at least $(n-1)$:

$$\det S = C\left(m + \frac{1}{4}\right)^K$$

for some $K \geq 2m-1$. In the next part, we will show that $K \leq 2m-1$. (In particular, this means that the determinant is nonzero.) Our method of proof will give us as a bonus that $C|4^k$, so that $4^k = C$.

Part Two: Let $r_1 \geq r_2 \geq \ldots \geq r_n$ be the roots of f_n. By definition, $\Delta(f_n) = \prod_{i\neq j}(r_i - r_j)$. Since the geometric mean of a set of positive numbers is less the arithmetic mean when the numbers are not all equal, we have that

$$
\begin{aligned}
\sqrt[(n-1)n]{\prod_{i\neq j} |(r_i - r_j)|} &< \frac{1}{n(n-1)} \sum_{i\neq j} |r_i - r_j| \\
&\leq \frac{1}{n} \sum_i |r_i|
\end{aligned}
$$

i.e.

$$
|\Delta(f_n)| < \left[\left(\frac{1}{n} \sum_i |r_i| \right)^n \right]^{n-1}. \tag{4.4}
$$

By Graeffe's method from the theory of symmetric functions, $|r_1| \doteq \left|\frac{\alpha_1}{1}\right|$, $|r_2| \doteq \left|\frac{\alpha_2}{\alpha_1}\right|$, etc. In particular,

$$
\begin{aligned}
\sum |r_i| &\leq 2\sum \left| \frac{\alpha_i}{\alpha_{i-1}} \right| \\
&= 2\sum \frac{\binom{m+a}{m-a}}{\binom{m+a-1}{m-a}} + 2\sum \frac{\binom{m+a}{m-a+1}}{\binom{m+a}{m-a}} \\
&= 2\sum \frac{m+a}{2a} + 2\sum \frac{2a}{m-a+1} \\
&\leq 2\int_1^m \frac{m+a}{2a} + 2\int_0^m \frac{2a}{m-a+1} \\
&= (1+\ln m)\, m - 1 + (4\ln(m+1) - 4)\, m + 4\ln(m+1) \\
&< \sqrt[n]{2}\,\sqrt[n]{2n+1} \quad (\text{for } n > 0.)
\end{aligned}
$$

Plugging this last inequality into 4.4 gives us that $|\Delta(f_n)| < 2(2n+1)^{n-1}$. Since the discriminant of f_n is clearly an integer, we have $|\Delta(f_n)| \leq (2n+1)^{n-1}$.

Part Three: We define a new matrix, S'', obtained by adding twice each odd row of S' to any row immediately below it, subtracting each odd row from any row immediately above it, and multiplying each even row by $\frac{1}{(4m+1)}$. S'' looks like

$$S'' = \begin{pmatrix} 2m\binom{m}{m} & -(2m-1)\binom{m}{m-1} & \cdots \\ \frac{(4m+1)\binom{m}{m}}{(4m+1)} & \frac{(-(4m-1)\binom{m}{m-1}-2m\binom{m}{m})}{(4m+1)} & \cdots \\ 0 & 2m\binom{m}{m} & \cdots \\ 0 & \vdots & \end{pmatrix}$$

Clearly $\det S = \det S' = (4m+1)^{2m-1} \det S''$. It is our task to show that $\det S'' = 1$. In light of parts one and two above, we know that $\det S'' = -1, 0,$ or 1. Thus, it suffices to show that $\det S'' > 0$.

Let M be a connected $k \times k$ minor of S'', which does not have a zero on the main diagonal. (As the name implies, a connected minor of a matrix M is a minor which is obtained by deleting rows and columns only at the beginning and end of M.)

For a matrix M, the notation $M|_i^j$ denotes the minor obtained by deleting the i–th row and the j–th column of matrix M. There is a recurrence, originally popularized by C. Dodgson, which relates the determinants of a square matrix with the determinants of its connected minors. Dodgson's determinant identity, [6], [38], states that

$$\det\left((M|_n^n)|_1^1\right)(\det M) = \left(\det M|_1^1\right)\left(\det M|_n^n\right) - \left(\det M|_n^1\right)\left(\det M|_1^n\right). \qquad (4.5)$$

The recurrence 4.5 is non-linear. Clearly, if we specify values for the determinants of connected one and two-dimensional minors of M, the recurrence 4.5 gives a unique value for $\det M$ if and only if there are no "interior" connected minors with zero determinant.

Let $z = \frac{k+1}{4}$. Let M be any $l \times l$ connected minor of M, for $l \leq k$. We use 4.5 to show that the $k \times k$ minor M satisfies the following three properties:

P1 If $k > 2$, then $\det M$ is an integer-valued polynomial (a polynomial which sends integers to integers.)

P2 Suppose $l = 2$; i.e. $\mathsf{M} = \begin{pmatrix} a(m) & b(m) \\ c(m) & d(m) \end{pmatrix}$ is a 2×2 minor of M. Then

$$|a(z)d(z)| > |b(z)c(z)|.$$

P3 $|\det \mathsf{M}|(m)$ is increasing for all $m \geq z$.

Property **(P1)** is an immediate consequence of part one of the proof. We will prove properties **(P2)** and **(P3)** using the identity 4.5. We will first note that they are true for 1-dimensional and 2-dimensional connected minors of the matrix S''.

$$\begin{pmatrix} + & - & - & + & + & - & - & + \\ + & - & - & + & + & - & - & + \\ 0 & + & - & - & + & + & - & - \\ 0 & + & - & - & + & + & - & - \\ 0 & 0 & + & - & - & + & + & - \\ 0 & 0 & + & - & - & + & + & - \\ 0 & 0 & 0 & + & - & - & + & + \\ 0 & 0 & 0 & + & - & - & + & + \end{pmatrix}$$

$$\longrightarrow \begin{pmatrix} - & + & - & + & - & + & - \\ + & + & + & + & + & + & + \\ 0 & - & + & - & + & - & + \\ 0 & + & + & + & + & + & + \\ 0 & 0 & - & + & - & + & - \\ 0 & 0 & + & + & + & + & + \\ 0 & 0 & 0 & - & + & - & + \end{pmatrix}$$

$$\longrightarrow \begin{pmatrix} + & - & - & + & + & - \\ - & - & + & + & - & - \\ 0 & + & - & - & + & + \\ 0 & - & - & + & + & - \\ 0 & 0 & + & - & - & + \\ 0 & 0 & - & - & + & + \end{pmatrix} \longrightarrow \begin{pmatrix} - & - & - & - & - \\ + & + & + & + & + \\ 0 & - & - & - & - \\ 0 & + & + & + & + \\ 0 & 0 & - & - & - \end{pmatrix}$$

$$\longrightarrow \begin{pmatrix} + & - & - & + \\ - & + & + & - \\ 0 & + & - & - \\ 0 & - & + & + \end{pmatrix}$$

$$\rightarrow \begin{pmatrix} + & - & + \\ + & + & + \\ 0 & + & - \end{pmatrix} \rightarrow \begin{pmatrix} + & - \\ + & + \end{pmatrix} \rightarrow \begin{pmatrix} + \end{pmatrix}$$

Figure 4-1 The “sign condensation” of S''

The 2×2 connected minors of S' look like one of

$$\begin{pmatrix} \binom{m+i}{m-1} & \pm\binom{m+i}{m-i-1} \\ \mp(2m+C+1)\binom{m+i-1}{m-i} & (2m+C)\binom{m+i}{m-i} \end{pmatrix}$$

whose determinant is

$$\frac{(m+i)(2m^2 + \text{lower order terms})m!^2}{(2i)!^2(m-i)!^2};$$

$$\begin{pmatrix} \binom{m+i}{m-i-1} & \pm\binom{m+i+1}{m-i-1} \\ \mp(2m+C+1)\binom{m+i}{m-i} & (2m+C)\binom{m+i}{m-i-1} \end{pmatrix}$$

whose determinant is

$$\frac{2(i+1)(-8m^2 + \text{lower order terms})(m+i)!^2}{(2i+2)!^2(m-i-1)!^2(m-i)};$$

$$\begin{pmatrix} (2m+C+1)\binom{m+i-1}{m-i} & \pm(2m+C)\binom{m+i}{m-i} \\ \binom{m+i-1}{m-i} & \pm\binom{m+i}{m-i} \end{pmatrix}$$

whose determinant is

$$\pm\binom{m+i-1}{m-i}\binom{m+i}{m-i}$$

or

$$\begin{pmatrix} (2m+C)\binom{m+i}{m-i} & \pm(2m+C-1)\binom{m+i}{m-i-1} \\ \binom{m+i}{m-i} & \pm\binom{m+i}{m-i-1} \end{pmatrix}$$

whose determinant is

$$\pm\binom{m+i}{m-i}\binom{m+i}{m-i-1}.$$

Examining the 1- and 2-dimensional connected minors, we see that **(P2)** - **(P3)** are satisfied when $l = 1, 2$.

Since the degree of $M|_1^1 M|_n^n$ never equals the degree of $M|_n^1 M|_1^n$, except when the minor M in the interior of S' has a zero on its diagonal, the right-hand side of 4.5 is never identically zero. Thus, the recurrence 4.5 has unique solution, with initial values for the 1×1 and 2×2 connected submatrices of S. In particular, the properties **(P2)** - **(P3)** follow by induction.

Since there are no interior zeros, 4.5 gives the value of $\det M(m)$ for any $m \geq z$. In particular, since z lies to the right of all the zeros and poles of $\det M(m)$, by **(P3)**, the sign of $\det M(z)$ is determined by the signs of the entries of $M(z)$. (For example, the case when $k = 8$ is shown in table 4-1.) Clearly, when M lies in the north-east corner of S'', and k is odd, we have that the sign of $M(z)$ is positive. But when $M = S''$, then $M(z) = \Delta(f_n)$ by construction. Thus $\Delta(f_n) \geq 0$. Combining parts one, two, and three, we now have that

$$\Delta(f_n) = (2n+1)^{n-1}$$

when $n \equiv 0 \bmod 4$.

The case $n \equiv 2 \bmod 4$ is the same as the case $n \equiv 0 \bmod 4$, except that the transformation of S to S' in 4.2 on page 73 changes the sign of the determinant of S. The determinants of $k \times k$ connected minors of S' now alternate in sign, giving the desired positive determinant $(2n+1)^{n-1}$ for S. The case $n \equiv 1 \bmod 4$ is the same as the case $n \equiv 0 \bmod 4$, except that the first two rows of the matrix S' now looks like

$$\begin{pmatrix} (2m+1)\binom{m}{m} & (2m)\binom{m+1}{m} & -(2m-1)\binom{m+1}{m-1} & \cdots \\ \binom{m}{m} & \binom{m}{m-1} & -\binom{m-1}{m-1} & \cdots \\ & & & \end{pmatrix}$$

where $(2m+1) = n$. The case $n \equiv 3 \bmod 4$ is similar to the case $n \equiv 2 \bmod 4$. $\diamondsuit$

Two invaluable resources for determinant identities are the surveys [99] and [7]. For the reader's convenience, we illustrate a "condensation" using the recurrence 4.5 from the second half of the proof, for the Sylvester matrix of polynomial $f_4(x)$ in figure 4-2 on page 92. We note that

- Condition **(P3)** is important, because if there are interior zeros at any step, the difference equation 4.5, together with the $(2n-1)^2+(2n-2)^2$ initial values at level $l=1$ and $l=2$, may not have unique solution.

- This is quite an unusual application of Dodgson's recurrence. More typically, the connected $n \times n$ minors of a matrix $M(m)$ are imbedded in a family of matrices $M_n(a,b)$ - and the recurrence 4.5 becomes an integral recurrence relation with variables a, b, M_n. Proving a determinant identity is now a matter of showing that the family of matrices $M_n(a,b)$ satisfies the recurrence and that an adequate set of initial conditions are satisfied. By using 4.5 instead essentially to bound the degree of $(2n+1)$ in $(\det S)(m)$, we avoided the (usually very hard) problem of finding a useful parameterization of the minors of S.

- Of course, one might try to find a more "mechanical" method of identifying the factors of the determinant of a matrix (part one of the proof.) Here is a brief description of the process which led to the discovery of theorem 4.5. We may write the first two rows of the matrix S' as:

$$\begin{matrix} -1\binom{m}{m} & 2\binom{m}{m-1} & \cdots \\ \binom{m}{m} & -\binom{m}{m-1} & \cdots . \end{matrix}$$

without changing the determinant. Clearly, the first non-zero entry of each even row must cancel the entry immediately above it; so we may assume without loss of generality that the linear combination of rows for which we are searching is obtained by multiplying the matrix $S(\tau)$ by the vector $(1, x_1, 1, x_2, \dots, 1, x_{2m-1}, X)^T$. Since the result is the zero vector, the sequence $[x_i]_i$ satisfies, for each $1 \le N < 2m$

$$(x_N - 1) + (x_{N-1} - 2)\alpha_{N-1} + \cdots + (x_1 - N)\alpha_N = 0$$

$$x_N = \sum_{l=1}^{N-1} (x_l - N + l - 1)\alpha_{N-l+1} = \sum_{l=1}^{N-1} x_l \alpha_{N-l+1} + \sum_{l=1}^{N-1} (N - l + 1)\alpha_{N-l+1}$$

i.e.

$$\sum_{l=1}^{N} x_l \alpha_{N-l+1} = \sum_{k=1}^{N} k\alpha_k. \tag{4.6}$$

In [79] the "generalized binomial series" B_t is introduced. It satisfies, and is well-defined by, the relation

$$\frac{B_t(z)^R}{1 - t + tB_t(z)^{-1}} = \sum_{k \geq 0} \binom{tk + R}{k} z^k.$$

When $t = 1$, this is the binomial series; when $t = 2$, we have the generating function for the Catalan numbers. More generally, we have

$$B_t(z) = \sum_{k\geq 0} \binom{tk+1}{k} \frac{1}{tk+1} z^k.$$

Consider the function

$$\begin{aligned} G_1(z) &= \sum_{k\geq 0} \binom{m+k}{2k} z^k \\ &= \frac{2B_{1/2}(z)^{[m+1]}}{B_{1/2}(z)+1}. \end{aligned}$$

(The second equality is the result of the identity $B_{1/2}(z) = \frac{1}{B_{1/2}(-z)}$.) The even terms of $G_1(z)$ may be extracted as

$$E(z) = \frac{B_{1/2}(z)^{[m+1]} + B_{1/2}(z)^{[-m]}}{B_{1/2}(z)+1}.$$

Likewise the odd terms of the function

$$\begin{aligned} G_2(z) &= \sum_{k\geq 0} \binom{m+k-1}{2k} z^k \\ &= \frac{2B_{1/2}(z)^{[m+\frac{1}{2}]}}{B_{1/2}(z)+1} \end{aligned}$$

are

$$O(z) = \frac{B_{1/2}(z)^{[m+\frac{1}{2}]} - B_{1/2}(z)^{[\frac{1}{2}-m]}}{B_{1/2}(z)+1}.$$

Thus,

$$\begin{aligned}
\sum_k |\alpha_{k+1}| z^k &= \sum_k \binom{m+\lfloor\frac{k+1}{2}\rfloor}{m-\lfloor\frac{k+2}{2}\rfloor} z^k \\
&= E(z) + O(z) \\
&= \frac{1}{B_{1/2}(z)+1}\Big(B_{1/2}(z)^{m+1} + B_{1/2}(z)^{-m} + B_{1/2}(z)^{m+\frac{1}{2}} \\
&\qquad -B_{1/2}(z)^{\frac{1}{2}-m}\Big).
\end{aligned}$$

and

$$\begin{aligned}
G(z) &= \sum_{k\geq 0} \alpha_k z^k \\
&= \frac{1}{z}\left(E(iz) + iO(iz)\right) \\
&= \frac{1}{z}\left(\left(\frac{iz+\sqrt{4-z^2}}{2}\right)^2 + 1\right)^{-1} \cdot \\
&\quad \left(\left(\frac{iz+\sqrt{4-z^2}}{2}\right)^{2m+2} + \left(\frac{iz+\sqrt{4-z^2}}{2}\right)^{-2m}\right. \\
&\quad \left. i\left(\frac{iz+\sqrt{4-z^2}}{2}\right)^{2m+1} + i\left(\frac{iz+\sqrt{4-z^2}}{2}\right)^{1-2m}\right).
\end{aligned}$$

Now the left-hand side of 4.6 is the convolution of the sequences $[x_k]_k$ and $[\alpha_{k-1}]_k$; the right-hand side of 4.6 is the convolution of the sequence $[1]_k$ and the derivative sequence of $[\alpha_k]_k$. So, writing 4.6 in terms of generating functions, we get

$$H(z)(zG(z)) = \frac{1}{1-z}G'(z)$$

$$H(z) = \frac{G'(z)}{G(z)}\frac{1}{z-z^2}$$

where $H(z)$ is the generating function for $[x_k]_k$.

For $M-1 < l$, we must have

$$-\sum(x_k + 2m - M)\alpha_{M-k} = X(2m - M + 1)\alpha_M \tag{4.7}$$

so that $\frac{\sum(x_k+2m-M)\alpha_{M-k}}{(2m-M+1)\alpha_M}$ must be constant. Writing the cases $M = 0, M = 1$ in terms of generating functions in variable z, and substituting $z = 0$, we see by inspecting special cases that this happens only when $m = -\frac{1}{4}$. S, an integral matrix, clearly **has** an integral determinant, so each of the equations 4.7 must be satisfied for some rational $m \in \mathbb{C}$. We have discovered that $(m+\frac{1}{4})$ is a linear factor (and, apparently, the only linear factor) of $\det S$. The force of theorem 4.5 is that S has no other linear factors.

- At the risk of diverging even further from the topic of this thesis, it is reasonable to ask: if we have an $n \times n$ matrix M, whose $(i-j)$—th entry is some function of i, j, and n, how could we hope to find a matrix N, with the same determinant as M, so that the recurrence 4.5 for $\det N$ has unique solution for a given set of initial values? I.e., if we have a determinant identity which we wish to prove, how could we guess a transformation of M which yields a matrix for which Dodgson's determinant recurrence is useful?

One possibility, if you are a graduate student with a lot of free time on your hands, is to try every determinant-preserving matrix transformation that you can think of until you find something that works. We very briefly sketch a more systematic plan of attack here.

Suppose that we have an integral matrix M, and that we have successfully used the recurrence

$$(\det M) = \frac{\left(\det M|_1^1\right)\left(\det M|_n^n\right) - \left(\det M|_n^1\right)\left(\det M|_1^n\right)}{\left(\det M|_n^n|_1^1\right)}$$

to evaluate $\det M$ (so that there are no interior zeros.) Then each connected minor was integral, and if we kept track of the determinants of the connected minors, then we could write down the Smith normal form of M, and the Hermite normal for of M. (This is actually a practical algorithm for finding the Smith and Hermite normal forms of integral matrices. The advantages of this method are its easy scalability to parallel machines, and the fact that the integers in intermediate calculations don't grow very fast as $n \to \infty$. The disadvantage of this method is that, if an interior zero is found, then one must start over, applying an elementary row or column operation to eliminate the zero. For a discussion of the technique, see [25] or [8].) On the other hand, a transformation of M to a matrix M' which changes the sort of possible parameterizations $M(a, b)$ must not respect the Smith normal form of M - otherwise, we could have gone from M to M' with an $SL_2\mathbb{Z}$ transformation.

If the determinant of an $n \times n$ matrix has prime factorization $p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$, then there are

$$P(M) = P_n(e_1)P_n(e_2)\cdots P_n(e_k)$$

possible choices for the invariant factors of M (where $P_n(j)$ is the number of j—tableaux shape with n or fewer rows – i.e. the number of partitions of j into n or fewer parts.) Recall that a *Smith Matrix* is a matrix in Smith normal form: a diagonal matrix s.t. $a_{ii} | a_{(i+1),(i+1)}$ for each diagonal entry $a_{(i+1),(i+1)}$, $i > 0$. There are clearly $P(M)$ Smith matrices with the same determinant as M. Each such matrix $\overline{M}$ which is $GL_2\mathbb{Q}$ similar to M, but only one them is $SL_2\mathbb{Z}$ similar to M. Writing down each of these matrices, and the $GL_2\mathbb{Q}$ matrices which turn it into M, is purely mechanical. If $M_n(m)$ is family of square matrices, each with parameter m and dimension n, then we get a family of possible transforms of $M_n(m)$ amenable to analysis with Dodgson's recurrence 4.5 by choosing some n', m', and finding the second-order recurrence relation which is satisfied by each $\overline{M_{n'}(m')}$, $\overline{M_{n'+1}(m')}$, and $\overline{M_{n'+2}(m')}$. Repeating the process with another

m'' gives another family of transforms of $M_n(m)$ - and any possible transform of $M_n(m)$ at these values of n must work for both m' and m''.

Here is a contrived but illustrative example. Consider the $n \times n$ matrix A_n defined by

$$a_{ij} = \begin{cases} 0, & i \neq j \\ 1, & i = j < n \\ \prod_{1 \leq p < q \leq n}(q-p), & i = j = n. \end{cases}$$

The first few values of $|A_n|$ are 1, 1, 2, $12 = 2^2 3$, $288 = 2^5 3^2$, $34560 = 2^8 3^3 5$. A_n is already in Smith normal form. For $i < j < n$, divide the term $(j-i)$ from the $n-$th row, and multiply the $j-$th row by $(j-i)$. We have GL_n-transformed A_n into an integral matrix with the same determinant, but with the following Smith normal form:

$$V'_n = \begin{pmatrix} 1 & & & & & \\ & 2-1 & & & & \\ & & (3-2)(3-1) & & & \\ & & & \ddots & & \\ & & & & \prod_{i<j}(j-i) & \\ & & & & & \ddots \end{pmatrix}.$$

By judiciously adding rows and columns, it is easy to SL_n-transform V' into the matrix

$$V_n = \begin{pmatrix} 1 & 1 & 1 & \dots \\ 1 & 2 & 3 & \\ 1^2 & 2^2 & 3^2 & \\ \vdots & & & \ddots \end{pmatrix}$$

which is a Vandermonde matrix whose determinant is $\prod_{1\leq p<q\leq n}(q-p)$. So we have discovered an unsurprising formula for the determinant of A_n. (We have also discovered an obscure fact about the Smith normal form of these Vandermonde matrices.)

- Returning from this long digression to the matter at hand, one might ask what the Smith normal forms of the Sylvester matrices of the f_n's look like. Actually tracing the divisors of $(\det \widetilde{S}')(m)$ for connected minors $\widetilde{S}'$ of S' yields:

Theorem 4.6 *Let S be the Smith normal form of the Sylvester matrix for $\Delta(f_n) = Res(f_n, f_n')$. Then S is a diagonal matrix, consisting of "1"'s in the first n diagonal positions, and "$(2n-1)$"'s in the remaining $n-1$ positions.*

$$
S = \begin{pmatrix}
1 & -2 & -3 & 1 & 1 & 0 & 0 \\
4 & -6 & -6 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & -2 & -3 & 1 & 1 \\
0 & 1 & -2 & -3 & 1 & 1 & 0 \\
0 & 4 & -6 & -6 & 1 & 0 & 0 \\
0 & 0 & 4 & -6 & -6 & 1 & 0 \\
0 & 0 & 0 & 4 & -6 & -6 & 1
\end{pmatrix}
$$

$$
\longrightarrow \begin{pmatrix}
-(2) & (2)(3) & -(3) & 1 & 0 & 0 \\
(2)^2 & (2)^3(3) & (2)^3(3) & (7) & 0 & 0 \\
0 & -(2) & (2)(3) & -(3) & 1 & 0 \\
0 & (2)^2 & (2)^3(3) & (2)^3(3) & (7) & 0 \\
0 & 0 & -(2) & (2)(3) & -(3) & 1 \\
0 & 0 & (2)^2 & (2)^3(3) & (2)^3(3) & (7)
\end{pmatrix}
$$

$$
\longrightarrow \begin{pmatrix}
(2)^2(3)^2 & -(2)^3(3)^2 & -(3)^2(5) & 0 & 0 \\
-(2) & -(2)^5 & (19) & (7) & 0 \\
0 & (2)^2(3)^2 & -(2)^3(3)^2 & -(3)^2(5) & 0 \\
0 & -(2) & -(2)^5 & (19) & (7) \\
0 & 0 & (2)^2(3)^2 & -(2)^3(3)^2 & -(3)^2(5)
\end{pmatrix}
$$

$$
\longrightarrow \begin{pmatrix}
-(2)(3)^3 & -(3)^2(13) & -(3)^2(5) & 0 \\
(2)^2(3)^2 & (2)(3)^3(5) & (3)^2(13) & 0 \\
0 & -(2)(3)^3 & -(3)^2(13) & -(3)^2(5) \\
0 & (2)^2(3)^2 & (2)(3)^3(5) & (3)^2(13)
\end{pmatrix}
$$

$$
\longrightarrow \begin{pmatrix}
(2)^2(3)^4 & -(3)^4 & 0 \\
-(2)(3)^3 & (3)^3(13) & (3)^2(13) \\
0 & (2)^2(3)^4 & -(3)^4
\end{pmatrix}
\longrightarrow \begin{pmatrix}
(3)^4(5) & -(3)^4 \\
(2)^2(3)^4 & (3)^4(7)
\end{pmatrix}
$$

$$
\longrightarrow \left((3)^6 \right)
$$

Figure 4-2 Dodgson condensation for the Sylvester matrix of $f_4(x)$.

Appendix: Program listing

```
## fpfricke  ver. 0.91
##
## Some routines for simplifying group presentations
## To accompany Chapter 4.
##
##
## Jeffrey Hall, 1997-1998.
##
## Written for GAP, v. 3.4.4  (Shonert et.al.)
##
## All monomials are ordered by degree-lex order:  triples before pairs
## before singletons.  Other graded monomial orderings would be easy
## to implement.
##
##
## Some handy GAP functions for simplifying presentations, and reducing
## words in free groups, are:
##    RelatorRepresentatives is an (undocumented -- beware!) function which
##    cyclically reduces a list of words in abstract generators.
##    RelatorRepresentatives( IdWord ) = [ ]    (the empty list)
##
##    MostFrequentGeneratorFpGroup returns the most frequent generator
##    of a word
##
##    TzMostFrequentPairs( <Tietze record>, <n> ) returns a list of lists
##    for each ofthe n  most frequently occuring relator subwords of the
##    form g1^e*g2^f,  e,f = -1 or 1, g1 <> g2.
##    The format returned is:
##       [ frequency, a,b, x]
##    where x=0 for e=1,f=1
##          x=1 for e=1, f=-1
##          x=2 for e=-1, f=1
##          x=3 for e=f=-1
##
##
##    ReducedRrsWord freely reduces a word
##
Print( "fpfricke, Version .91\n" );

NUMPAIRSCHKTzWHIT := 20;
NUMTRIPLESCHKTzWHIT := 20;

emptyVectors := [ [0], [0,0,0], [0,0,0,0,0,0,0] ];

RememberMonabc := [];
```

```
RememberMonabc[3] := []; RememberMonabc[3][1] := [];
RememberMonabc[3][1][2] := []; RememberMonabc[3][1][2][3] := 7;

##
##  setRememberMonabc sets up the n'th row of the RememberMonabc table for
function Monabc

setRememberMonabc := function( n )

local count, i,j,k;

  if not IsBound( RememberMonabc[n] ) then
    count := n + Binomial( n, 2 );
    RememberMonabc[n] := [];

    for i in [1..n] do
       RememberMonabc[n][i] := [];
      for j in [(i+1)..n] do
        RememberMonabc[n][i][j] := [];
        for k in [(j+1)..n] do
          count := count + 1;
          RememberMonabc[n][i][j][k] := count;
        od;
      od;
    od;
   fi;
end;



##
## MonomialVector returns a vector of zeroes, whose length is
## n + C(n, 2) + C(n,3)


MonomialVector := function( n )

local l, i, c;

  if IsBound( emptyVectors[n] ) then
    return ShallowCopy( emptyVectors[n] );
  else
      c :=  (n + Binomial(n,2) +Binomial(n,3));
      l := [];
      l[c] := 0;
```

```
      for i in [1..c] do
        l[i] := 0;
      od;
      emptyVectors[n] := ShallowCopy( l );
      return l;

  fi;
end;


##  set up the remember tables

for i in [4..45] do
  MonomialVector( i );
  setRememberMonabc( i );
od;



##
##  Sorted returns the sorted version of its single argument, without
##  changing the argument.

Sorted := function( l )

local L;

  L := ShallowCopy( l );
  Sort( L );
  return L;
end;


##
##  ourRelatorRepresentatives cyclically reduces a list of relators

ourRelatorRepresentatives := function( l )

  if l = [IdWord] then
    return [IdWord];
  else
    return RelatorRepresentatives( l );
  fi;
end;
```

```
##
##  positionInSet returns the position of  l or l^-1 in L,
##  where L is a sorted list without holes of length len.
##
##  No checking is done to see whether l<>[] and length(l) = 1.

positionInSet := function( l,L, len )

local left, right, middle, tmp;

  left := 1; right := len;

  while left + 1 < right do
    middle := QuoInt( right-left, 2 ) + left;
    tmp := L[middle];
    if tmp < l then
      left := middle;

    elif tmp > l^-1 then      # By convention, l < l^-1
      right := middle;

    else
       left := middle;
       right := middle;
    fi;
  od;

  if L[left] < l then
    return right;
  else
    return left;
  fi;

end;


##
##  positionInList returns the position of the first occurence of
##  l or l^-1 in L
##  L need not be sorted, and may have holes.


positionInList := function( l,L )
```

```
  return  Minimum( Position( L, l ),
                   Position( L, l^-1 ) );
end;


##
##  LettersInWord returns a list of the letters in the word, in order
##  of first appearance.

LettersInWord := function( w, varList )

local W, thisLetter, l;

  W := Copy( w );  l := [];

  while (W <> IdWord) do

    thisLetter := Subword( W, 1, 1);
    if not (thisLetter in varList) then
      thisLetter := thisLetter^-1;
    fi;
    Add( l, thisLetter );
    W := EliminatedWord( W, thisLetter, IdWord );
  od;

  return l;
end;


##
## IsSortedWord returns true if w is sorted lexicographically

IsSortedWord := function( w )

local count, tmp, tmp1, tmp2, len;

  len := LengthWord( w );
  tmp1 := Subword( w, 1, 1);
  tmp := true;
  count := 2;

  while (count <= len) and tmp do
    tmp2 :=  Subword( w, count, count);
    tmp := tmp and (tmp1 <= tmp2);
    tmp1 := tmp2;
```

```
    count := count+1;
  od;

  return tmp;
end;


##
##  LettersInWordList returns a list of the letters in the words of a
##  list, in order of first appearance.

LettersInWordList  := function( l, varlist )

local L, W, thisLetter, li, len, i, j;

  L := Copy(l); li := []; len := Length( l );

  for i in [1..len]  do

    W := L[i];

    while (W <> IdWord) do

      thisLetter := Subword( W, 1, 1);
      if not (thisLetter in varlist) then
        thisLetter := thisLetter^-1;
      fi;
      Add( li, thisLetter );
      W := EliminatedWord( W, thisLetter, IdWord );
      for j in [i..len] do
        L[j] := EliminatedWord( L[j], thisLetter, IdWord );
      od;

     od;
  od;

  return li;
end;

##
##  Monab( a,b, VARLIST, n) returns the position of the coordinate of
tr_{ab}
##  where a, b are variables in VARLIST, a<>B, where n = |VARLIST|
##  No type- or bound-checking is performed.
```

```
Monab := function( a, b, VARLIST, n )
local A,B;

  A := positionInSet( a, VARLIST, n );
  B := positionInSet( b, VARLIST, n );

  return A*(n+1) -(A^2 + A)/2 + B - A;

end;


##
##  Monabc( a,b,c, VARLIST, n) returns the position of the coordinate of
##    tr_{abc},
##    a < b < c
##  where a, b are variables in VARLIST, a<>B, where n = |VARLIST|
##  No type- or bound-checking is performed.

Monabc := function( a, b, c, VARLIST, n)

local A,B,C;

   A := positionInSet( a, VARLIST, n );
   B := positionInSet( b, VARLIST, n );
   C := positionInSet( c, VARLIST, n );

  setRememberMonabc( n );
  return RememberMonabc[n][A][B][C];

end;


##
##  PosFirstNonInverse returns the position of the first generator
##  in w which is not raised to a negative power.
##  w is assumed to be square-free
##
##  If w has only positive exponents, returns false

PosFirstNonInverse := function( w, varlist )

local count;

  for count in [1..LengthWord( w )] do
    if not (Subword( w, count, count) in varlist) then
```

```
      return count;
    fi;
  od;
  return false;
end;


##
##  leqDLEX returns true if  m1 <= m2  with degree-lex order
##
##  m1, m2 are monomials, represented as a list of integers
##  No type-checking is performed.


leqDLEX := function( m1, m2 )

local s1, s2;

  s1 := Sum(m1);
  s2 := Sum(m2);

  return ( (s1 < s2)
              or
           ( (s1 = s2) and  m1 <= m2 ) );
end;

##
##  MaxMonPair returns the largest of m1, m2, with respect
##  to leqDLEX order

MaxMonPair := function( m1, m2 )

  if leqDLEX(m1, m2) then
    return m2;
  else
    return m1;
  fi;
end;


##
##  MaxMonTrip returns the largest of m1, m2, m3

MaxMonTrip := function( m1, m2, m3 )
  return MaxMonPair( MaxMonPair( m1, m2), m3 );
```

```
end;


##
##  MaxMon returns the largest monomial in its argument, which
##  must be a list.
##  If the argument is empty, returns "false"

MaxMon := function( l )

local max, i;

  max := false;
  for i in l do
    max := MaxMonPair( max, i);
  od;

  return max;
end;



##
##  Forward reference of function LTFrickeChar1

LTFrickeChar1 := function( w,       # a word
                           varList, # the variables
                           m );
end;


##
##  LTFrickeCharSquareFree returns the lead monomial of a normal form of
##  the Fricke character (trace polynomial) of square-free word w,
##  multiplied by the monomial m.
##  w is assumed to be cyclically reduced.

LTFrickeCharSquareFree := function( w,       # a word
                                    VARLIST, # the variables
                                    m )

local len, numvars, M, letters, count, ES,
      tmp, a, b, c, max, maxPos, W;

  len := LengthWord( w );
```

```
M := ShallowCopy( m );
numvars := Length( VARLIST );

if len = 1 then
  tmp := positionInSet( Subword( w, 1, 1), VARLIST, len );
  M[tmp] := M[tmp] + 1;
  return M;

elif len = 0 then
  return M;

elif len = 2 then

  # two cases: something like "ab" or something like "ab^-1"
  # the leading term of trace(ab) is just t_{ab}
  # the leading term of trace(ab^-1) is t_a * t_b

  a := Subword(w,1,1);
  b := Subword(w,2,2);


  if (a in VARLIST) and (b in VARLIST) then

    tmp := Sorted( [ a,b ] );
    tmp := Monab( tmp[1], tmp[2], VARLIST, numvars );
    M[tmp] := M[tmp] + 1;
    return M;
  else
    tmp :=  positionInSet( a, VARLIST, numvars );
    M[tmp] := M[tmp] + 1;
    tmp :=  positionInSet( b, VARLIST, numvars );
    M[tmp] := M[tmp] + 1;
    return M;
  fi;

elif len = 3 then

  a :=  Subword(w,1,1);
  b :=  Subword(w,2,2);
  c :=  Subword(w,3,3);

  # 4 cases: "abc", or "acb", or "abc^-1", or "acb^-1"

  ES := 0;
```

```
if a in VARLIST then
  ES := ES + 1;
else
  ES := ES -1;
fi;
if b in VARLIST then
  ES := ES + 1;
else
  ES := ES -1;
fi;
if c in VARLIST then
  ES := ES + 1;
else
  ES := ES -1;
fi;

if  ES < 0 then
  return LTFrickeCharSquareFree( w^-1, VARLIST, m);

elif  ES = 3 then

  if IsSortedWord( w ) then   # "abc" case

    tmp := Monabc( a, b, c, VARLIST, numvars);

    M[tmp] := M[tmp] + 1;
    return M;

  else     #"acb" case

    tmp := positionInSet( a, VARLIST, numvars );
    M[tmp] := M[tmp] + 1;
    tmp := positionInSet( b, VARLIST, numvars );
    M[tmp] := M[tmp] + 1;
    tmp := positionInSet( c, VARLIST, numvars );
    M[tmp] := M[tmp] + 1;
    return M;
  fi;

else     # "abc^-1"  ... one letter has exponent -1, the others +1
         #
         # so that LT( trace( abc^-1 ) ) = tr(ab)tr(c)

  if ExponentSumWord( w, a) < 0 then
    tmp := a; a := c; c := tmp;
```

```
      elif ExponentSumWord( w, b) < 0 then
        tmp := b; b := c; c := tmp;

      fi;

      tmp := Sorted( [a,b ] );
      tmp := Monab( tmp[1], tmp[2], VARLIST, numvars );
      M[tmp] := M[tmp] + 1;

      tmp := positionInSet( c, VARLIST, numvars );
      M[tmp] := M[tmp] + 1;
      return M;
    fi;

  else # len >= 4, so there is the possibility of repeated letters

    # get rid of repeated letters


    for count in [1..len-1] do

      tmp :=  Subword(w, count, count);
      tmp :=  Minimum( PositionWord( w, tmp, count+1 ),
                       PositionWord( w, tmp^-1, count+1 ));

      if tmp <> false then

        a := Subword( w, 1, count);
        b := ourRelatorRepresentatives( ReducedRrsWord( [Subword( w,
                                                                   count+1,
                                                                   tmp )]
))[1];

        if tmp = len then
          c := IdWord;
        else
          c := Subword( w, tmp+1, len);
        fi;
        tmp := ourRelatorRepresentatives( ReducedRrsWord( [c*a] ))[1];

        return LTFrickeChar1( tmp,
                              VARLIST,
                              m )
                 +
```

```
              LTFrickeCharSquareFree( b,
                                      VARLIST,
                                      m );
      fi;

    od;


      # if w survives the for loop, then w is square-free,
      # with no repeated letters

      tmp := false;
      count := 0;
      max := Subword( w, 1, 1);
      maxPos := 1;

      while (tmp = false) and count < len do
        count := count + 1;
        a := Subword( w, count, count);
        if not (a in VARLIST) then
          tmp := count;
        fi;
        if a < max then
          maxPos := count;
          max := a;
        fi;
      od;

      if tmp = false then    # all of the exponents of w are 1

        if maxPos = len then
          W := max*Subword( w, 1, maxPos-1);

        elif maxPos > 1 then
          W := max*Subword( w, maxPos +1, len)*Subword( w, 1, maxPos-1);
        else
          W := w;
        fi;

        M :=  MonomialVector( numvars );


        # Now, if f = max, and b,c,d are the last letters of W, and a is
everything
```

```
        # between, then by Vogt's identity, the leading term of t_w = t_W is
one of:
        #
        #           t_{fac}*t_b*t_d
        #           t_{fa}*t_b*t_{cd}
        #           t_{fad}*t_b*t_c
        # since these are the terms of highest degree, when w=(fabcd) is a
word with
        # positive exponents and no repeated letters.


        return (
         m
           +
         MaxMonTrip( LTFrickeCharSquareFree( Subword(W,1,len-3)*Subword(W,len-1,len-1),
                                            VARLIST, M)
               +
                    LTFrickeCharSquareFree( Subword(W,len-2,len-2),
                                            VARLIST, M)
                      +
                    LTFrickeCharSquareFree( Subword(W,len,len),
                                            VARLIST, M),
# i.e.,  t_{fac}*t_b*t_d

                    LTFrickeCharSquareFree( Subword(W,1,len-3),
                                            VARLIST, M)
               +
                    LTFrickeCharSquareFree( Subword(W,len-2,len-2),
                                            VARLIST, M)
                      +
                    LTFrickeCharSquareFree( Subword(W,len-1,len-1)*Subword(W,len,len),
                                            VARLIST, M),
# i.e. t_{fa}*t_b*t_{cd}

                    LTFrickeCharSquareFree( Subword(W,1,len-3)*Subword(W,len,len),
                                            VARLIST, M)
               +
                    LTFrickeCharSquareFree( Subword(W,len-2,len-2),
                                            VARLIST, M
                      +
                    LTFrickeCharSquareFree( Subword(W,len-1,len-1),
                                            VARLIST, M) )
# i.e. t_{fad}*t_b*t_c

                   )
```

```
                  );


      else  # finally, we have the case where w is square-free, without
repeated
            # generators, but with at least one exponent -1 at position tmp

        a := Subword(w,tmp,tmp);

        return  LTFrickeCharSquareFree( EliminatedWord( w,
                                                        a,
                                                        a^-1 ),
                                        VARLIST,
                                        m );
      fi;

  fi;

end;    # LTFrickeCharSquareFree



##
##  LTFrickeChar1 returns the lead monomial of a normal form of
##  the Fricke character (trace polynomial) of word w, multiplied
##  by the monomial m
##  w is assumed to be cyclically reduced.

LTFrickeChar1 := function( w,       # a word
                           VARLIST, # the variables
                           m )

local M, W, c, tmp, squareFreePart, len, numvars;

  M := ShallowCopy( m );
  squareFreePart := IdWord;
  W := Copy( w );
  len := LengthWord(W);
  numvars := Length( VARLIST );

  while (W <> IdWord) do

    tmp := Subword(W, 1, 1);
    if (len > 1)
         and
```

```
      (tmp = Subword(W, 2, 2))
    then
      c := positionInSet( tmp, VARLIST, numvars );
      M[c] := M[c] + 1;

    else
      squareFreePart := squareFreePart*tmp;
    fi;

    W := tmp^-1 * W;
    len := len -1;
  od;

  return LTFrickeCharSquareFree( squareFreePart, VARLIST, M );

end;  # LTFrickeChar1


##
##  LTFrickeChar returns the lead monomial of a normal form of
##  the Fricke character (trace polynomial) of word w

LTFrickeChar := function( w,       # a word
                          varList) # the variables

  return LTFrickeChar1( ourRelatorRepresentatives( ReducedRrsWord([w]) )[1],
                        Set(varList),
                        MonomialVector( Length( varList ) ) );

end;


##
##  nudgeSet increases its argument S by 1, where S, a boolean list,
##  is considered to be a binary integer written with least significant
##  digit first, with false=0, true=1.  In case of overflow, nudgeSet
##  changes nothing, and returns "false"; otherwise "true" is returned.

nudgeSet := function( S )

local p, carry, lenS;

  lenS := Length( S );

  if SizeBlist( S ) = 0 then
```

```
    return false;
  else
    carry := true;
    p := 1;
    while carry and p < lenS do
      if S[p] then
        carry := true;
        S[p] := false;
        p := p + 1;
      else
        S[p] := true;
        carry := false;
        return true;
      fi;
    od;
  fi;
end;


##
##  DimMonomialIdeal returns the degree of the Hilbert polynomial of
##  the ideal generated by the monomials in list L,  The ideal <L> is
##  assumed to be over a field of characteristic zero.
##  We do this by actually looking at each of the subsets of the
##  variables, in order to use GAP 3.4 kernel functions in preference
##  to library functions.
##  As the number of variables grows, then the issue of avoiding library
##  functions in favor of internal GAP functions becomes moot, of course.

DimMonomialIdeal := function( L )

local l, S, least, M, nummons, numvars, count, covers, p, truesies,
noOverflow;

  l := Set( L );  # get rid or duplicates - might not always be useful
  if l = [] then
    return false;
  fi;

  nummons := Length( l );
  numvars := Length( l[1] );
  least := numvars;

  # for each monomial in l, make a boolean list, with entry "true" in
  # the place of each variable used in the monomial
```

```
  M := [];
  for count in [1..nummons] do
    M[count] := List( l[count],
                      i -> i > 0 );
    IsBlist( M[count] );
  od;


  # Search through the subsets of [1..numvars]
  truesies := List( [1..numvars], i -> true );
  IsBlist( truesies );
  S := ShallowCopy( truesies );
  S[1] := false;
  IsBlist( S );
  least := numvars;

  noOverflow := true;
  while noOverflow do
    covers := true;
    count := 1;

    while covers and count <= nummons do
      # Oddly, it's quicker to use the kernel function SizeBlist, than
      # to find the OR of the elements of S AND B.
      covers := covers
                  and
                SizeBlist( ( IntersectionBlist( S, M[count] ) ) ) > 0;
      count := count + 1;
    od;

    if covers then     # found a set of variables in each of the monomials

      least := Minimum( least, SizeBlist( S ) );
      noOverflow := nudgeSet( S );

    else               # ignore large sets
      repeat
        noOverflow := nudgeSet( S );
      until (not noOverflow) and SizeBlist( S ) > least;
    fi;
  od;

  return numvars - least;

end; # DimMonomialIdeal
```

```
##
## Calculates the part of the squared "fractional dimension"
## contributed by the coordinate axis i in monomial ideal M
## Returns 2^15-1 if the fractional dimension contributed is 1

FracDimSqrForAxis := function(M, l)

local least, m, tmp;

  least := false;

  for m in M do
    tmp := m[l];
    if tmp < least then
      least := tmp;
    fi;
  od;

  if least <> false then
    return least;
  else
    return 0;
  fi;

end;


##
## Calculates the "fractional dimension" of a monomial ideal

FracDimSqrPerAxis := function( M )

local i, tmp;

  if M = [] or (not IsList( M )) then
    return false;
  fi;

  tmp := [];
  for i in [1..Length(M[1])] do
    tmp[i] := FracDimSqrForAxis(M, i);
  od;
```

```
  return tmp;

end;


##
##  IsStructGP tests M to see if it is pairwise relatively prime

IsStructGB := function( M )

  return false = PositionProperty( Sum ( List( M,
                                              m -> List( m,
                                                         i -> SignInt(i)
                                                       )
                                              )
                                       ),
                                   i -> i > 1
                                  );
end;



##
##  TzWhit tries to simplify the presentation P, by searching the space
##  of Tietze transforms of P, to minimize the dimension of the monomial
##  ideal generated by the leading terms of the variety of the ideal of
##  SL_2 character relations which come from the relations of P, as
##  presented by the function LTFrickeChar.
##
##  TzWhit uses, and is modelled upon, the low- and high- level
##  functions for Tietze transformations in GAP.


TzWhit := function( arg ) # arg = [P,
                          #         numtries, (default is 10)
                          #         useMostFreqP (default is false)  ];


local tietze, count, rels, relators, RealRelators, m, n, tmp, FD, pp,
      A, B, LeadingMonomials, p, pairs, pairs1, couples, x, g, elims,
      left, right, i, j, current, total,
      P, numtries, useMostFreqP  ;

  if Length( arg ) = 3 then    # this flag is undocumented, and indeed
hardly ever works.
```

```
    useMostFreqP := arg[3];
  else useMostFreqP := false;
  fi;
  if  Length( arg ) = 2 then numtries := arg[2];
  else numtries := 10;
  fi;
  P := arg[1];


  #TzFindCyclicJoins( P );  # do some preprocessing, and run consistency
checks

  tietze := P.tietze;
  n := tietze[TZ_NUMGENS];
  if n < 2 then
    return;     # nothing to do to P
  fi;
  rels := tietze[TZ_RELATORS];
  m := tietze[TZ_NUMRELS];

  RealRelators := List( [1..m],
                     j -> TzWord( tietze, rels[j] ) );  # the presentation
relators


  # relators is a list of the group relators, plus the relators times left-
  # multiplied by each generator

  relators := [];
  for tmp in tietze[TZ_GENERATORS] do
    Append( relators, tmp*RealRelators );
  od;


  LeadingMonomials := List( relators,
                             r -> LTFrickeChar( r,
                                                 Copy( tietze[TZ_GENERATORS] )
) );

  FD := FracDimSqrPerAxis( LeadingMonomials );

  # TzMostFrequentPairs returns a list of lists for each of the
  # NUMPAIRSCHKTzWHIT most frequently occuring length 2 subwords (a^eb^f)
  # in the relators.  The format returned is:
  #     [ frequency, a,b, x]
```

```
# where x=0 for e=1,f=1
#       x=1 for e=1, f=-1
#       x=2 for e=-1, f=1
#       x=3 for e=f=-1

if useMostFreqP then

  pairs := TzMostFrequentPairs( P, NUMPAIRSCHKTzWHIT, useMostFreqP );

  # Pick out those squares which have the same exponent
  # Order them by the increase in fractional codimension which each
  # might contribute if they are replaced by a new generator

  couples := [];
  IsSet( couples );
  for p in pairs do
    if p[4] = 0 or p[4] = 3 then
      pp :=  Sorted( p{[2,3]} );
      A := pp[1];
      B := pp[2];
      AddSet( couples,
              Concatenation( [FD[A*(n+1) -(A^2 + A)/2 + B - A ]],
                             pp
                           ) );
    fi;
  od;
else
# Choose the NUMPAIRSCHKTzWHIT smallest-frac-codim pairs, order them by
the
# increase in frac. codim. each might contribute if they were replaced.
# This avoids string matching. We use a modified quicksort selection
method
# (see e.g. Sedgewick, "Algorithms", Addison--Wesley, 1983)

  pairs := [];
  pairs1 := [];
  total := 0;
  for A in [1..n-1] do  #chuck out the zeros
    for B in [(A+1)..n] do
      tmp := FD[A*(n+1) -(A^2 + A)/2 + B - A ];

      if tmp <> 0 then
        if tmp = 1 then
          Add( pairs1,
               [tmp, A, B] );
```

```
      else
         Add( pairs,
             [tmp, A, B] );
         total := total + tmp;
      fi;
    fi;
  od;
od;

couples := [];
right := Length( pairs );
if right < NUMPAIRSCHKTzWHIT then
  couples := Concatenation( pairs,
                            pairs1{[1..Minimum( NUMPAIRSCHKTzWHIT,
                                               Length( pairs1 )
                                             )
                                  ]} );
elif right = 0 then
  couples := [];
else
  left := 1;
  i := 0;  j := right;
  current := QuoInt( total*2*NUMPAIRSCHKTzWHIT, tmp^2 );
            # e.g. a weighted average of pairs
            # best partition strategy if the values of FD[pairs]
            # were uniformly distributed
  while left < right do
    repeat
      repeat
        i := i + 1;
      until pairs[i][1] <= current;
      repeat
        j := j - 1;
      until pairs[j][1] >= current;
      tmp := pairs[i];
      pairs[i] := pairs[j];
      pairs[j] := tmp;
    until j <= i;

    pairs[j] := pairs[i];
    pairs[i] := pairs[right];
    pairs[right] := tmp;

    if i >= NUMPAIRSCHKTzWHIT then
      right := i -1;
```

```
      fi;
      if i <= NUMPAIRSCHKTzWHIT then
        left := i + 1;
      fi;
      current := pairs[right][1];
      i := left - 1;
      j := right;
    od;

    couples := pairs{[1..NUMPAIRSCHKTzWHIT]};
  fi;
fi;

elims := [];
for p in couples{[1..Minimum( numtries, Length( couples ) )]} do

  # Add a new generator
  AddGenerator( P );
  x := P.generators[ Length( P.generators ) ];

  # Add relation x^-1*a*b
  AddRelator( P,
              x^-1*TzWord( tietze, [p[2], p[3]] ) );

  # choose the generator in {a,b} contributing the least fract. codim.

  elims := [];
  if FD[p[2]] < FD[p[3]] then
    AddSet( elims, p[3] );
  else
    AddSet( elims, p[2] );
  fi;

od;

# replace each ab or (ab)^-1 by the generators introduced
P.searchSimultaneous := Maximum( 20, Length( couples )+10 );
TzCheckRecord( P );
TzSearch( P );

# for each (a,b) eliminate the generator with the largest FracDimSqr

for g in elims do
  TzEliminateGen(P , g);
od;
```

```
end;  # TzWhit



##
## TzWhitTriples is like TzWhit, but for triples rather than pairs.

TzWhitTriples := function( arg ) # arg = [P,
                                 #        numtries, (default is 10) ]


local tietze, count, rels, relators, m, n, tmp, FD, pp,
      A, B, C, LeadingMonomials, p, triples, triples1, triplets, x, g,
      elims, left, right, i, j, current, total,
      P, numtries;


  if  Length( arg ) = 2 then numtries := arg[2];
  else numtries := 10;
  fi;
  P := arg[1];

  TzFindCyclicJoins( P );  # do some preprocessing, and run consistency
checks

  tietze := P.tietze;
  n := tietze[TZ_NUMGENS];
  if n < 3 then
    return;     # nothing to do to P
  fi;
  rels := tietze[TZ_RELATORS];
  m := tietze[TZ_NUMRELS];

  relators := List( [1..m],
                    j -> TzWord( tietze, rels[j] ) );

  LeadingMonomials := List( relators,
                            r -> LTFrickeChar( r,
                                               Copy( tietze[TZ_GENERATORS] )
) );

  FD := FracDimSqrPerAxis( LeadingMonomials );
```

```
  # Choose the NUMTRIPLESCHKTzWHIT smallest-frac-codim triples, order them
by the
  # increase in frac. codim. each might contribute if they were replaced.
  # This avoids string matching. We use a modified quicksort selection
method
  # (see e.g. Sedgewick, "Algorithms", Addison--Wesley, 1983)

  triples := [];
  triples1 := [];
  total := 0;
  setRememberMonabc( n );

  for A in [1..n-2] do  #chuck out the zeros
    for B in [(A+1)..n] do
      for C in [(B+1)..n] do
        tmp := FD[ RememberMonabc[n][A][B][C] ];

        if tmp <> 0 then
          if tmp = 1 then
            Add( triples1,
                 [tmp, A, B, C] );
          else
             Add( triples,
                 [tmp, A, B, C] );
             total := total + tmp;
          fi;
        fi;
      od;
    od;
  od;

  triplets := [];
  right := Length( triples );
  if right < NUMTRIPLESCHKTzWHIT then
    triplets := Concatenation( triples,
                             triples1{[1..Minimum( NUMTRIPLESCHKTzWHIT,
                                                   Length( triples1 )
                                                 )
                                          ]} );
  elif right = 0 then
    triplets := [];
  else
    left := 1;
    i := 0;  j := right;
    current := QuoInt( total*2*NUMTRIPLESCHKTzWHIT, tmp^2 );
```

```
              # e.g. a weighted average of triples
              # best partition strategy if the values of FD[triples]
              # were uniformly distributed
  while left < right do
    repeat
      repeat
        i := i + 1;
      until triples[i][1] <= current;
      repeat
        j := j - 1;
      until triples[j][1] >= current;
      tmp := triples[i];
      triples[i] := triples[j];
      triples[j] := tmp;
    until j <= i;

    triples[j] := triples[i];
    triples[i] := triples[right];
    triples[right] := tmp;

    if i >= NUMTRIPLESCHKTzWHIT then
      right := i -1;
    fi;
    if i <= NUMTRIPLESCHKTzWHIT then
      left := i + 1;
    fi;
    current := triples[right][1];
    i := left - 1;
    j := right;
  od;

  triplets := triples{[1..NUMTRIPLESCHKTzWHIT]};
fi;

elims := [];
for p in triplets{[1..Minimum( numtries, Length( triplets ) )]} do

  # Add a new generator
  AddGenerator( P );
  x := P.generators[ Length( P.generators ) ];

  # Add relation x^-1*a*b*c
  AddRelator( P,
              x^-1*TzWord( tietze, [p[2], p[3], p[4]] ) );
```

```
    # choose the generator in {a,b,c} contributing the least fract. codim.

    elims := [];
    if FD[p[2]] <= FD[p[3]] then
      if  FD[p[3]] <= FD[p[4]] then
        AddSet( elims, p[4] );
      else
         AddSet( elims, p[3] );
      fi;
    elif  FD[p[2]] <= FD[p[4]] then
       AddSet( elims, p[4] );
    else
       AddSet( elims, p[2] );
    fi;

  od;

  # replace each abc or (abc)^-1 by the generators introduced
  P.searchSimultaneous := Maximum( 20, Length( triplets )+10 );
  TzCheckRecord( P );
  TzSearch( P );

  # for each (a,b,c) eliminate the generator with the largest FracDimSqr

  for g in elims do
    TzEliminateGen(P , g);
  od;

end;  # TzWhitTriples
```

Bibliography

[1] *Microsymposium: Graph theory in chemistry.* Inst. für Strahlenchemie, Max-Planck-Inst. Kohlenforschung, Mülheim, 1976. Held at the Institut für Strahlenchemie, Max-Planck-Institut für Kohlenforschung, Mülheim, May 26–28, 1975, Match No. 1 (1975). Reprinted in 1976.

[2] William Abikoff, Keneth Appel, and Paul Shupp. Lifting surface groups to sltwoc. In *Kleinian groups and related topics*, volume 971 of *Lecture Notes in Math.*, pages 1–5. Springer-Verlag, 1985.

[3] Scot Adams. Representation varieties of arithmetic groups and polynomial periodicity of betti numbers. *Israel Journal of Mathematics*, 88:73–124, 1994.

[4] William W. Adams and Philippe Loustaunau. *An introduction to Grobner bases.* Graduate Studies in Mathematics. American Mathematical Society, 1994.

[5] Roger C. Alperin and Claudio Procesi. Some representations of groups of automorphisms of a free group. *Journal of Algebra*, 181(1):16–25, 1996.

[6] Tweodros Ambdeberhan and Shalosh B. Ekhad. A condensed condensation proof of a determinant identity conjectured by greg kuperberg and jim propp. *Journal of Combinatorial Theory, Series A*, 78(1):169–170, 1997.

[7] George Andrews and W. H. Burge. Determinant identities. *Pacific Journal of Mathematics*, 158:1–14, 1993.

[8] Edwin H. Bareiss. Sylvester's identity and multistep integer-perserving Gaussian elimination. *Mathematics of Computation*, 22(103):565–578, 1968.

[9] G. Baumslag, editor. *Algorithms and classification in combinatorial group theory*, volume 23 of *Mathematical Sciences Research Institute Publications*. Springer-Verlag, New York, 1992. Papers from the Workshop on Algorithms, Word Problems and Classification in Combinatorial Group Theory held in Berkeley, California, January 1989.

[10] Gilbert Baumslag. A survey of groups with a single defining relation. In *Proceedings of groups—St. Andrews 1985*, volume 121 of *London Mathematical Society Lecture Note Series*, pages 30–58, Cambridge, 1986. Cambridge Univ. Press.

[11] Gilbert Baumslag. *Topics in combinatorial group theory*. Lectures in Mathematics ETH Zürich. Birkhäuser Verlag, Basel, 1993.

[12] Gilbert Baumslag, Dion Gildenhuys, and Ralph Strebel. Algorithmically insoluble problems about finitely presented solvable groups, Lie and associative algebras. III. *Journal of Pure and Applied Algebra*, 54(1):1–35, 1988.

[13] Gilbert Baumslag and Charles F. Miller III. Some odd finitely presented groups. *The Bulletin of the London Mathematical Society*, 20(3):239–244, 1988.

[14] Gilbert Baumslag, John W. Morgan, and Peter B. Shalen. Generalized triangle groups. *Mathematical Proceedings of the Cambridge Philosophical Society*, 102(1):25–31, 1987.

[15] Gilbert Baumslag and Peter B. Shalen. Affine algebraic sets and some infinite finitely presented groups. In *Essays in group theory*, volume 8 of *Math. Sci. Res. Inst. Publ.*, pages 1–14. Springer-Verlag, New York, 1987.

[16] Dave Bayer and David Mumford. What can be computed in algebraic geometry? In *Computational algebraic geometry and commutative algebra (Cortona, 1991)*, Sympos. Math., XXXIV, pages 1–48. Cambridge Univ. Press, Cambridge, 1993.

[17] Dave Bayer and Mike Stillman. Computation of Hilbert functions. *Journal of Symbolic Computation*, 14(1):31–50, 1992.

[18] David Bayer and Michael Stillman. A criterion for detecting $\mathfrak{m}$-regularity. *Inventiones Mathematicae*, 87(1):1–11, 1987.

[19] David Bayer and Michael Stillman. On the complexity of computing syzygies. *Journal of Symbolic Computation*, 6(2-3):135–147, 1988. Computational aspects of commutative algebra.

[20] A.F. Beardon. Some remarks on non-discrete groups. *Communications in Mathematical Physics*, 163(3):605–627, 1994.

[21] Ch. Benecke, R. Grund, R. Hohberger, A. Kerber, R. Laue, and Th. Wieland.

Chemical isomerism, a challenge for algebraic combinatorics and for computer science. In *Applied algebra, algebraic algorithms and error-correcting codes (Paris, 1995)*, volume 948 of *Lecture Notes in Comput. Sci.*, pages 4–20. Springer-Verlag, Berlin, 1995.

[22] Anders Björner and Günter M. Ziegler. Introduction to greedoids. In *Matroid applications*, pages 284–357. Cambridge Univ. Press, Cambridge, 1992.

[23] William Burnside. *The Theory of Groups of Finite Order.* Cambridge, 1902. Reprint Dover 1955.

[24] David Eisenbud C. Deconcini and Claudio Procesi. Young diagrams and determinantal varieties. *Inventiones Mathematicae*, 56:129–65, 1980.

[25] Henri Cohen. *A Course in Computational Number Theory.* Graduate Texts in Mathematics. Springer-Verlag, 1993.

[26] Paula Beazley Cohen and Juergen Wolfart. Modular embeddings for some non-arithmetic fuchsian groups. *Acta Arithmetica*, 56(2):93–110, 1990.

[27] Paula Beazley Cohen and Juergen Wolfart. Dessins de Grothendieck et variétés de Shimura. *Comptes rendus de l'Academie des sciences. Serie I Mathematique*, 315(Deuxieme Semestre):1025–1028, 1992.

[28] J.H. Conway, R.T. Curtis, S.P. Norton, R.A. Parker, and R.A. Wilson. *Atlas of finite groups.* Clarendon Press, Oxford, 1985.

[29] Robert Cori and Antonio Machì. Maps, hypermaps and their automorphisms: a survey i. *Expositiones Mathematicae*, 10:403–27, 1992.

[30] Jean-Marc Couveignes and Louis Granboulan. Dessins from a geometric point of view. In Leila Schneps, editor, *The Grothendieck theory of dessins d'enfants*, number 200 in London Mathematical Society Lecture Note Series, pages 79–113. Cambridge University Press, 1994.

[31] David Cox, John Little, and Donal O'Shea. *Ideals, varieties, and algorithms : an introduction to computational algebraic geometry and commutative algebra.* Springer-Verlag, 2nd edition, 1997.

[32] H.S.M. Coxeter and W.O.J. Moser. *Generators and Relations for Discrete Groups*, volume 14 of *Ergeb. Math. Grenzgeb.* Springer-Verlag, 4th edition, 1957.

[33] Marc Culler and Peter B. Shalen. Varieties of group repesentations and splittings of 3-manifolds. *Annals of Mathematics*, 117:109–46, 1983.

[34] Marc Culler and Karen Vogtmann. A group theoretic criterion for property fa. *Proceedings of the American Mathematical Society*, 124(3):677–683, 1996.

[35] Charles W. Curtis and Irving Reiner. *Methods of Representation Theory (with Applications to Finite Groups and Orders), Volume 1.* John Wiley & Sons, 1981.

[36] D. D. Long D. Cooper, M. Culler and P. B. Shalen. Plane curves associated to character varieties of 3-manifolds. Preprint.

[37] John D. Dixon. High speed computation of group characters. *Numerical Mathematics*, 10:446–450, 1967.

[38] C. L. Dodgson. Condensation of deteminants, being a new and brief method for computing their arithmetic values. *Proceedings of the Royal Society, Series A*, 15:150–155, 1866.

[39] David Eisenbud. *Commutative Algebra.* Springer-Verlag, 1994.

[40] Martin Schoenert et al. *GAP - Groups, Algorithms, and Programming.* Lehrstuhl D fur Mathematik, Rheinisch Westfalische Technische, Hochschule, Aachen, Germany, 5th ed., release 3.4.4 edition, 1995.

[41] David V. Feldman. Counting subgroups of free groups. December 1996.

[42] B. Fine and G. Rosenberger. A note on generalized triangle groups. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 56:233–244, 1986.

[43] Benjamin Fine. *Algebraic theory of the Bianchi groups*, volume 129 of *Monographs and Textbooks in Pure and Applied Mathematics.* Marcel Dekker Inc., New York, 1989.

[44] Benjamin Fine. Subgroup presentations without coset representatives. In *Topology and combinatorial group theory (Hanover, NH, 1986/1987; Enfield, NH, 1988)*, volume 1440 of *Lecture Notes in Math.*, pages 59–73. Springer-Verlag, Berlin, 1990.

[45] Benjamin Fine, Frank Levin, Frank Roehl, and Gerhard Rosenberger. The generalized tetrahedron groups. In *Geometric group theory (Columbus, OH, 1992)*, volume 3 of *Ohio State Univ. Math. Res. Inst. Publ.*, pages 99–119. de Gruyter, Berlin, 1995.

[46] Benjamin Fine, Frank Levin, and Gerhard Rosenberger. Free subgroups and decompositions of one-relator products of cyclics. I. The Tits ernative. *Archiv der Mathematik (Basel)*, 50(2):97–109, 1988.

[47] Benjamin Fine, Frank Levin, and Gerhard Rosenberger. Faithful complex representations of one relator groups. *New Zealand Journal of Mathematics*, 26(1):45–52, 1997.

[48] Benjamin Fine and Morris Newman. The normal subgroup structure of the Picard group. *Transactions of the American Mathematical Society*, 302(2):769–786, 1987.

[49] Benjamin Fine, Frank Roehl, and Gerhard Rosenberger. A Freiheitssatz for certain one-relator amalgamated products. In *Combinatorial and geometric*

group theory (Edinburgh, 1993), volume 204 of *London Math. Soc. Lecture Note Ser.*, pages 73–86. Cambridge Univ. Press, Cambridge, 1995.

[50] Benjamin Fine and Gerhard Rosenberger. Complex representations and one-relator products of cyclics. In *Geometry of group representations (Boulder, CO, 1987)*, volume 74 of *Contemp. Math.*, pages 131–147. Amer. Math. Soc., Providence, RI, 1988.

[51] Benjamin Fine and Gerhard Rosenberger. On groups of special NEC type. A Freiheitssatz and related results for a class of multi-relator groups. In *Geometric topology (Haifa, 1992)*, volume 164 of *Contemp. Math.*, pages 17–34. Amer. Math. Soc., Providence, RI, 1994.

[52] Edward Formanek. *The polynomial identities and invariants of* $n \times n$ *matrices*, volume 78 of *CBMS Regional Conference Series in Mathematics*. Published for the Conference Board of the Mathematical Sciences, Washington, DC, 1991.

[53] Edward Formanek and Claudio Procesi. The automorphism group of a free group is not linear. *Journal of Algebra*, 149(2):494–499, 1992.

[54] Robert Fricke and Felix Klein. *Vorlesungen uber die Theorie der automorphen Functionen*, volume 1. Teubner, 1897-1912.

[55] William Goldman. Moduli spaces. Summary of a series of talks presented at the Korea National University of Education July 11, 1994, 1994.

[56] William Goldman. Ergodic theory on moduli spaces. *preprint*, 1995.

[57] F. González-Acuña and José María Montesinos-Amilibia. On the character variety of group representations in sl(2, c) and psl(2, c). *Mathematische Zeitschrift*, 214(4):627–652, 1993.

[58] R. William Gosper. Decision procedure for indefinite hypergeometric summation. *Proceedings of the National Academy of Sciences of the United States of America*, 75:40–42, 1978.

[59] Louis Granboulan. Construction d'une extension régulière de group de galois $\mathfrak{m}_{24}$. *preprint*, 1994.

[60] Daniel H. Greene and Donald E. Knuth. *Mathematics for the Analysis of Algorithms*. Progress in Computer Science. Birkhauser, Boston, 1982.

[61] Larry C. Grove. *Groups and Characters*. John Wiley and Sons, 1997.

[62] R. Grund, A. Kerber, and R. Laue. Construction of discrete structures, especially isomers. *Discrete Applied Mathematics*, 67:115–126, 1996.

[63] George Havas and Jin Xian Liam. A new problem in string searching. *preprint*, pages 1–9, 1998.

[64] George Havas and Mark Ollila. Applications of substring searching methods to group presentations. In *Proceedings of the Sixteenth Australian Computer Sci-

ence Conference, Brisbane 1993, pages 587–593. Australian Computer Science Communications, 1993.

[65] George Havas and Edmund F. Robertson. Application of computational tools for finite groups. In *Computational Support for Discrete Mathematics, DIMACS 1992*, volume 15, pages 29–32. American Mathematical Society, 1994.

[66] Melvin Hochster and J.L. Roberts. Rings of invariants of reductive groups acting on regular rings are cohen-macaulay. *Advances in Mathematics*, 13, 1974.

[67] A. F. Horadam, R. P. Loh, and A. G. Shannon. Divisibility properties of some Fibonacci-type sequences. In *Combinatorial mathematics, VI (Proc. Sixth Austral. Conf., Univ. New England, Armidale, 1978)*, pages 55–64. Springer-Verlag, 1979.

[68] Robert D. Horowitz. Characters of free groups represented in the two-dimensional special linear group. *Communications in Pure and Applied Mathematics*, 25:635–649, 1972.

[69] Robert D. Horowitz. Induced automorphisms on Fricke characters of free groups. *Transactions of the American Mathematical Society*, 208:41–50, 1975.

[70] I.M. Isaacs. *Character Theory of Finite Groups*, volume 69 of *Pure and Applied Mathematics*. Academic Press (Reprinted Dover), Berlin, Heidelberg, New York, 1976.

[71] Joseph P. S. Kung J. Desarmenian and Gian-Carlo Rota. Invariant theory, young bitableaux, and combinatorics. *Advances in Mathematics*, 27:63–92, 1978.

[72] D.L. Johnson. *Presentation of Groups.* London Mathematical Society Lecture Note Series. Cambridge University Press, Cambridge, 1976.

[73] D.L. Johnson. *Presentations of Groups.* Cambridge University Press, Cambridge, 1990.

[74] Gareth Jones and David Singerman. Maps, hypermaps and triangle groups. In Leila Schneps, editor, *The Grothendieck theory of dessins d'enfants*, number 200 in London Mathematical Society Lecture Note Series, pages 115–145. Cambridge University Press, 1994.

[75] Troels Jørgensen. Traces in 2-generator subgroups of sl(2, **c**). *Proceedings of the American Mathematical Society*, 84(3):339–343, 1982.

[76] G. Kemper. The invar package for calculating rings of invariants. Maple package, 1993.

[77] Adalbert Kerber. *Representations of permutation groups. II.* Springer-Verlag, Berlin, 1975. Lecture Notes in Mathematics, Vol. 495.

[78] Adalbert Kerber. *Algebraic combinatorics via finite group actions.* Bibliographisches Institut, Mannheim, 1991.

[79] Ronald L. Graham Donald E. Knuth and Oren Patashnik. *Concrete Mathematics*. Addison-Wesley, corrected 2nd edition edition, 1995.

[80] Fulton Koehler. Bounds for the moduli of the zeros of a polynomial. *Proceedings of the American Mathematical Society*, 5:414–419, 1954.

[81] Wolfram Koepf. Efficient computation of Chebyshev polynomials in computer algebra. 1996.

[82] Wolfram Koepf. Algorithms for the indefinite and definite summation. preprint, Maple share library, 1997.

[83] Bernhard Korte, Laszlo Lovasz, and Rainer Schrader. *Greedoids*. Number 4 in Algorithms and Combinatorics. Springer-Verlag, Berlin, 1991.

[84] S.A. Linton. Constructing matrix representations of finitely presented groups. *Journal of Symbolic Computation*, 12:427–438, 1991.

[85] Alexander Lubotzky and Andy R. Magid. *Varieties of representations of finitely presented groups*. Number 336 in Memoirs of the AMS. American Mathematical Society, 1985.

[86] Roger C. Lyndon and Paul E. Schupp. *Combinatorial Group Theory*. Springer-Verlag, 1977.

[87] Jr. Marshall Hall. *The Theory of Groups*. Macmillan Co., New York, 1959.

[88] Brendan McKay. Practical graph isomorphism. *Congressus Numeratium*, 30:45–87, 1981.

[89] John McKay. The construction of the character table of a finite group from generators and relations. In *Computational problems in abstract algebra*, pages 89–100, London, New York, 1970. (Oxford, 1967), Pergamon Press.

[90] Russell Merris and William Watkins. An invariant theory approach to graph enumeration. *Linear Algebra and its Applications*, 61:277–285, 1984.

[91] Gerhard O. Michler. Some problems in computational representation theory. *Journal of Symbolic Computation*, 9:571–582, 1990.

[92] J. Lyn Miller. Analogs of grobner bases in polynomial rings over a ring. *Journal of Symbolic Computation*, 21(2):139–153, 1996.

[93] Charles F. Miller III. Decision problems for groups—survey and reflections. In *Algorithms and classification in combinatorial group theory (Berkeley, CA, 1989)*, volume 23 of *Math. Sci. Res. Inst. Publ.*, pages 1–59. Springer-Verlag, New York, 1992.

[94] H. R. Morton. Fibonacci-like sequences and greatest common divisors. *American Mathematical Monthly*, 102(8):731–734, 1995.

[95] Morris Newman. The smith normal form. *Linear Algebra and its Applications*, 254:367–381, 1997.

[96] W. Nickel, A. Niemeyer, and M. Schönert. GAP *Getting started and reference manual.* U. Aachen, Lehrstuhl D für Mathematik, 1988.

[97] Emmy Noether. Der endlichkeitssatz der invarianten enlicher gruppen. *Mathische Annalen*, pages 77–89, 1916.

[98] Francois Ollivier. Canonical bases: Relations with standard bases, finiteness conditions and application to tame automorphisms. In Teo Mora and C. Traverso, editors, *Effective Methods in Algebraic Geometry*, volume 94 of *Progress in Mathematics*, pages 379–400. Birkhauser, Boston, 1991.

[99] Marko Petkovšek, Herbert S. Wilf, and Doron Zeilberger. A = B. A K Peters, Wellesley, MA, 1996.

[100] M. Pohst and H. Zassenhaus. *Algorithmic algebraic number theory*, volume 30 of *Encyclopedia of Mathematics and its applications.* Cambridge, revised edition, 1997.

[101] G. Polya. *Combinatorial enumeration of groups, graphs, and chemical compounds.* Springer-Verlag, 1987. Edited by Ronald Read.

[102] C. Procesi. Computing with 2×2 matrices. *Journal of Algebra*, 87(2):342–359, 1984.

[103] Claudio Procesi. Finite dimensional representations of algebras. *Israel Journal of Mathematics*, 19:169–182, 1974.

[104] Claudio Procesi. The invariants of $n \times n$ matrices. *Bulletin of the American Mathematical Society*, 82(6):891–892, 1976.

[105] R.C. Read. The graph isomorphism disease. *Journal of Graph Theory*, 1:339–363, 1977.

[106] Ronald Read. Every one a winner, or how to avoid isomorphism search when cataloguing combinatorial configurations. *Annals of Discrete Mathematics*, 2:107–120, 1978.

[107] Ronald Read. A survey of graph generation techniques. In Kevin McAvarey, editor, *Combinatorial Mathematics, VII*, volume 884 of *Lecture Notes in Mathematics*, pages 77–89. Springer-Verlag, 1981.

[108] Darren Redfern. *The Maple handbook.* Springer-Verlag, New York, 1993.

[109] Paulo Ribenboim. *The Book of Prime Number Records.* Springer-Verlag, 1988.

[110] Lorenzo Robbiano and Moss Sweedler. Subalgebra bases. In *Commutative algebra (Salvador, 1988)*, volume 1430 of *Lecture Notes in Math.*, pages 61–87. Springer, Berlin, 1990.

[111] Kyoji Saito. Algebraic representation of the teichmuller spaces. In Leila Schneps, editor, *The Grothendieck theory of dessins d'enfants*, number 200 in London Mathematical Society Lecture Note Series, pages 47–78. Cambridge University Press, 1994.

[112] W.R. Scott. *Group Theory.* Dover Publications, Inc., Mineola, N.Y., 1964.

[113] Charles C. Sims. *Computation with finitely presented groups*, volume 48 of *Encyclopedia of Mathematics and its Applications.* Cambridge University Press, Cambridge, 1994.

[114] Richard P. Stanley. Invariants of finite groups and their applications to combinatorics. *Bull. Amer. Math. Soc. (N.S.)*, 1(3):475–511, 1979.

[115] Bernd Sturmfels. *Algorithms in invariant theory.* Texts and Monographs in Symbolic Computation. Springer-Verlag, Vienna, 1993.

[116] Bernd Sturmfels. *Gröbner bases and convex polytopes*, volume 8 of *University Lecture Series.* American Mathematical Society, Providence, RI, 1996.

[117] Bernd Sturmfels and Markus Wiegelmann. Structural Gröbner basis detection. *Applicable Algebra in Engineering, Communication and Computing*, 8(4):257–263, 1997.

[118] Charles R. Traina. Trace polynomials for two generator subgroups of sl(2,c). *Proceedings of the American Mathematical Society*, 79(3):369–372, 1980.

[119] Morgan Ward. Linear divisibility sequences. *Transactions of the American Mathematical Society*, 41:276–286, 1937.

[120] Neil White, editor. *Theory of matroids*, volume 26 of *Encyclopedia of Mathematics and its Applications.* Cambridge University Press, Cambridge, 1986.

[121] Neil White, editor. *Combinatorial geometries*, volume 29 of *Encyclopedia of Mathematics and its Applications.* Cambridge University Press, Cambridge, 1987.

[122] Neil White, editor. *Matroid applications*, volume 40 of *Encyclopedia of Mathematics and its Applications.* Cambridge University Press, Cambridge, 1992.

[123] Alice Whittemore. On representations of the group of Listing's knot by subgroups of sl(2, C). *Proceedings of the American Mathematical Society*, 40:378–382, 1973.

[124] Alice Whittemore. On special linear characters of free groups of rank $n \geq 4$. *Proceedings of the American Mathematical Society*, 40:383–388, 1973.

[125] Herbert S. Wilf. What is an answer? *The American Mathematical Monthly*, 89(5):289–292, 1982.

[126] Herbert S. Wilf and Doron Zeilberger. An algorithmic proof theory for hypergeometric (ordinary and "q") multisum/integral identities. *Inventiones Mathematicae*, 108(3):575–633, 1992.

[127] Abraham Karass Wilhelm Magnus and Donald Solitar. *Combinatorial Group Theory: Presentations of Groups in Terms of Generators and Relations.* Dover Publications, 2nd ed. edition, 1976. First edition published General Publishing Co., Toronto.

[128] Stephen Wolfram. *Mathematica: a system for doing mathematics by computer*, 1988.

[129] Doron Zeilberger. Dodgson's determinant-evaluation rule proved by two-timing men and women. *Electronic Journal of Mathematics*, 4(2), 1997. R22, Wilf Festschrift Volume.