SGGI Package

Peter A. Brooksbank

Bucknell University

pbrooksb@bucknell.edu

Version 0.1

February 12, 2020

Contents

CHAPTER 1

Introduction

This documentation describes MAGMA functions that facilitate computation with *string groups generated by involutions* (SGGIs). Functions that compute with these structures already exist in distributed versions of MAGMA [**BCP**]. The purpose of the current package is to supplement—and in some cases improve upon—the existing machinery.

Citing SGGI. To cite the SGGI package, please use the following

Peter A. Brooksbank, *SGGI*, version 0.1, GitHub, 2020. `https://github.com/galois60/SGGI`.

For AMSRefs:

```
\bib{SGGI}{misc}{
   author={Brooksbank, Peter A.},
   title={SGGI},
   publisher={GitHub},
   year={2020},
   edition={version 0.1},
   note={\texttt{https://github.com/galois60/SGGI}},
}
```

1.1. Overview

1.2. Version

CHAPTER 2

String groups generated by involutions (SGGIs)

This package works with SGGIs as a data type. We begin by describing the essential ingredients. A **string group generated by involutions**, or **SGGI** consists of:

(1) a group H; and
(2) a list $h_1, \ldots, h_m$ of involutions of H.

The properties that make it an SGGI are first that $H = \langle h_1, \ldots, h_n \rangle$, and secondly that the involutions satisfy the **string condition**, namely

$$\forall i, j \in [m] \qquad |i - j| > 1 \implies h_i h_j = h_j h_i \tag{1}$$

The integer m is the **rank** of the SGGI. The package uses a data type `SGGI` for these objects.

2.1. Constructing SGGIs

The package works with the two representations of groups that are currently most relevant to computation with SGGIs, namely permutation groups and matrix groups.

A user can specify a SGGI in several related ways.

```
StringGroupGeneratedByInvolutions(S) : SeqEnum[GrpPermElt] -> SGGI
StringGroupGeneratedByInvolutions(S) : SeqEnum[GrpMatElt] -> SGGI
```

Provided S is a list of involutions satisfying (1) this returns the SGGI with group generated by S.

```
StringGroupGeneratedByInvolutions(G, S) : GrpPerm, SeqEnum[GrpPermElt] -> SGGI
StringGroupGeneratedByInvolutions(G, S) : GrpMat, SeqEnum[GrpMatElt] -> SGGI
```

Similar, but checks that the specified G is in fact generated by S. One can build the SGGI on the defining generators of input group G as follows:

```
StringGroupGeneratedByInvolutions(G) : GrpPerm -> SGGI
StringGroupGeneratedByInvolutions(G) : GrpMat  -> SGGI
```

Here are some basic access functions for SGGIs.

```
Group(H) : SGGI -> Grp
```

gives the underlying group of H, and

```
Generators(H) : SGGI -> SeqEnum
```

returns the sequence $h_1, \ldots, h_n$ of involutions generating that group.

```
Rank(H) : SGGI -> RngIntElt
```

is the rank of H, and

```
SchlafliType(H) : SGGI -> SeqEnum[RngIntElt]
```

return the Schlafli symbol of H, namely the sequence $\mathrm{ord}(h_i h_{i+1})$ for $i = 1, \ldots, n-1$.

```
Print(H) : SGGI
```

displays information about H.

2.1.1. Special constructions.

The symmetric group S_{n+1} with generators $(1\ 2), (2\ 3), \ldots, (n\ n{+}1)$ is the most obvious construction of a SGGI of rank n. The corresponding polytope is the simplex:

```
Simplex(n) : RngIntElt -> SGGI
```

Example 2.1. `Simplex`

We build the 5-simplex as an SGGI and access its basic features.

```
> H := Simplex(5);
> Group(H);
Permutation group acting on a set of cardinality 6
    (1, 2)
    (2, 3)
    (3, 4)
    (4, 5)
    (5, 6)
> Rank(H);
5
> SchlafliType(H);
[3, 3, 3, 3]
> Print(H);
SGGI with underlying group type GrpPerm.
```

Evidently, SGGIs also arise naturally as quotients of Coxeter groups.

`ModularReflectionGroup(L, p) : SeqEnum[RngIntElt], RngIntElt -> SGGI`

Given a list $L = [\ell_1, \ldots, \ell_{n-1}]$ of matural numbers such that the reflection group G_0 with Schlafli symbol L is crystallographic, return the reduction of G_0 modulo the specified prime p. (Note, a "0" entry in L is understood as "∞".)

CHAPTER 3

String C-groups

Bibliography

[BCP] Wieb Bosma, John Cannon, and Catherine Playoust, *The Magma algebra system. I. The user language*, J. Symbolic Comput. **3-4** (1997), 235-265.

[BL] Peter A. Brooksbank and Dimitri Leemans, *Rank reduction of string C-group representations*, Proc. Amer. Math. Soc.

Intrinsics