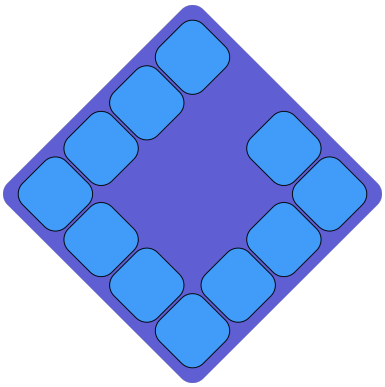
Grout GUI V.1 Documentation

Scott Balchin

27th August 2015

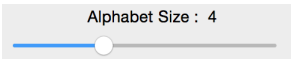


Introduction

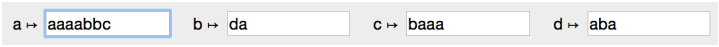
Grout GUI is a user interface that can be used to compute invariants of 1-dimensional substitution tiling spaces. Grout is written in C++, and the GUI was created using the Qt library version 5.4.1. For full functionality a LaTeX installation is required, as Grout will attempt to access PDFLaTeX from the console. Grout can be opened by double clicking the icon in the downloaded folder.

1. Performing and Interpreting Calculations

1.1. **Manually Entering Substitutions.** To manually enter a substitution, first move the alphabet slider to the required number (Grout GUI accepts alphabets up to 8 letters). A substitution can then be typed into the available input boxes, these boxes will only allow you to enter valid letters to reduce human error in entering substitutions. Grout will check that all of the available alphabet has been used before running computations.

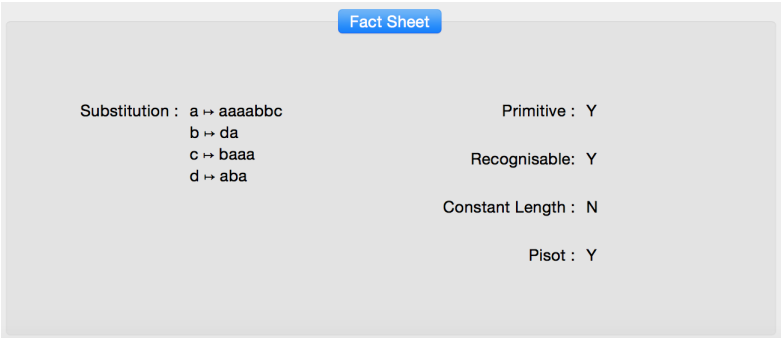


The alphabet slider



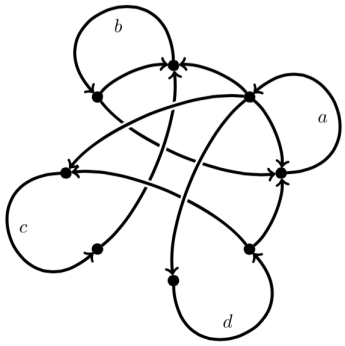
An entered substitution

1.2. **Selecting Output.** Grout always computes a standard set of results which appear in the *fact sheet* page of the results, the *compute* button can be pressed to begin a computation. On this page, it will show whether the substitution is primitive, recognisable, Pisot and constant length. If the recognisable result shows ?, then this means the substitution cannot be determined to be definitely recognisable or not due to limitations in the code.



The fact sheet for the above substitution

All available computations are shown in the scrollable area of tick boxes. To compute a specific result, just tick the relevant box. Mostly all results will display in the results area below the tick boxes, with different tabs for each group of tick boxes. If a substitution is not primitive, then many of the calculations will not be done as the algorithms will not work. The only results that do not display in the results area are the Barge-Diamond and Anderson-Putnam complexes. These complexes are created in Tikz by Grout and then compiled using PDFLaTeX, they will then be opened in the systems default PDF viewer. If PDFLaTeX is not installed, or Grout fails to local to install, then no results for the complexes will be shown. However the Tikz code for them may still be accessed(see section 3.1 for details).

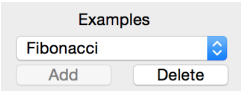


The outputted Barge-Diamond complex

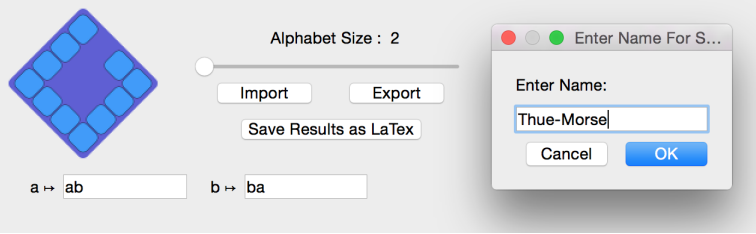
1.3. **Interpreting Results.** All results are easily interpreted apart from the cohomology results. Grout offers three separate cohomology calculations. The Barge-Diamond complex is printed out in terms of the inverse limit of the transpose substitution matrix M^T and any $\mathbb{Z}$ terms that must be quotiented or added. The Anderson-Putnam and Properisation cohomology are given just as matrices, the cohomology of the tiling space is then the inverse limit of this matrix. The cohomology rank is the rank of the cohomology over $\mathbb{Q}$, it is the same for all three cohomology calculations as they give isomorphic results. The only other result that needs explaining is the complexity. This result gives the number of n letter words from $n \geq 2$.

2. The Example List

The example system allows substitutions to be saved and loaded in Grout. To load an example, simply click on the drop down example list and select the required substitution, Grout will then input the substitution into the boxes, without any manual entry required. This is particularly useful for complicated substitutions, or ones that need to be accessed often. To save an example, it must be computed first, this is done so that invalid substitutions are not stored. Clicking on the *add* button brings up a dialogue box which asks for a name for the substitution, keep in mind the program is designed to only work with unique names. To delete an example first select it, then click the *delete* button. The examples are stored in a file called `examples.txt` in the same folder as the executable. The syntax of the stored examples is easy to understand, so examples can be manually entered.



The Fibonacci example loaded in



Adding the Thue-Morse substitution

3. Import and Export Buttons

3.1. **Importing and Exporting Substitutions.** Grout has the option to import and export substitutions as copy and paste-able strings. This allows substitutions to be sent to other people for them to look at without the need of them manually entering it into Grout. Like the examples, the substitution must first be computed before being exported, and the syntax for these strings is the same as the examples, but without an attached name.

The export string for a substitution on 8 letters

3.2. **Exporting to LaTeX.** The final feature of Grout GUI is the ability to export the results to a LaTeX file. After computing a substitution, the *Save Results as LaTeX* button can be clicked, which will bring up a save box, asking for a save location and name for a `.tex` file. A LaTeX file will then automatically be generated, and will include all of the results that have been selected in the GUI. If the Barge-Diamond or Anderson-Putnam complexes have been selected, then the Tikz code for these will also be in the file. These files are generated such that they will compile with no extra work needed.

Substitution Fact Sheet

$$a \mapsto abcda$$
$$b \mapsto ab$$
$$c \mapsto cdbc$$
$$d \mapsto db$$

Primitive : Yes

Recognisable: Yes

Constant Length : No

Pisot : No

Substitution Matrix :

$$\begin{pmatrix} 2 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 2 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$$

Determinant : 4

Eigenvalues : 3.20, 1.13, 1.13, 0.97

Tile Lengths : (4.86, 2.21, 2.66, 1.00)

Tile Frequencies : (0.26, 0.31, 0.21, 0.21)

A portion of an exported LaTeX sheet

Support

If any extra assistance is required, please contact me at `slb85@le.ac.uk`. Please report any bug reports, feature requests or general comments to this address also.

Disclaimer

To my knowledge all of the algorithms implemented in Grout are correct, but there is always a risk there is a bug that has not been spotted.