

ChainCounting User Guide

Version 1.0

Derrick Stolee
University of Nebraska-Lincoln
s-dstolee1@math.unl.edu

February 27, 2012

Abstract

The *ChainCounting* software generates formulas for the number of chains in width-two posets built from small configurations. Then, these formulas are evaluated on many inputs to find which numbers are represented as the number of chains in these posets.

1 Acquiring *ChainCounting*

The latest version of *ChainCounting* and its documentation is available online as part of the *SearchLib* collection at the address

<http://www.math.unl.edu/~s-dstolee1/SearchLib/>

ChainCounting is made available open-source under the GPL 3.0 license.

To compile *ChainCounting*, use a terminal to access the `ChainCounting/src/` folder and type `make`. The executables will be placed in `ChainCounting/bin/`

1.1 Acquiring Necessary Libraries

There is one *SearchLib* project used by *ChainCounting*.

1. *TreeSearch* is a project in *SearchLib* that abstracts the structure of a backtrack search in order to allow for parallelization. *TreeSearch* is available on the same web site as *ChainCounting*. Consult the *TreeSearch* documentation [2] for details about the arguments and execution processes.

1.2 Full Directory Structure

For proper compilation, place the different dependencies in the following directory structure:

- `SearchLib/` – The *SearchLib* collection.
 - `ChainCounting/` – The *ChainCounting* project.
 - * `bin/` – The final binaries are placed here.
 - * `docs/` – This folder contains documentation.
 - * `src/` – Contains source code. Compilation occurs here.
 - `TreeSearch/` – A support project from *SearchLib*.

2 Execution

The *ChainCounting* project uses a single executable: `chains.exe`. This executable evaluates a given formula $f_C(\mathbf{a}; \mathbf{b})$ for some configuration C of a certain size. These formulas are hard-coded into the source files, but they were generated automatically using the methods described in [1].

```
chains.exe [TreeSearch args] -N # -r # [--cliquer]
```

- `-N #` specifies the number n of vertices to use. All uniquely K_r -saturated graphs of order n will be generated.
- `-r #` specifies the value of r to use when searching for uniquely K_r -saturated graphs.
- `--cliquer` is an option that specifies to use the *cliquer* library in the pruning steps of the search. If not specified, the search uses a tabulation method.

References

- [1] E. Kupin, B. Reiniger, D. Stolee, Counting chains in width-two posets, *in preparation*, (2012).
- [2] D. Stolee, TreeSearch user guide, available at <http://www.github.com/derrickstolee/TreeSearch/> 2011.