

# CaGe

A virtual environment for studying  
some special classes of plane graphs

N. Van Cleemput   G. Brinkmann   O. Delgado–Friedrichs  
S. Liskén   A. Peeters

Combinatorial Algorithms and Algorithmic Graph Theory  
Department of Applied Mathematics and Computer Science  
Ghent University



# Outline

- 1 Introduction
  - What is CaGe?
  - Example
  - Brief history
- 2 CaGe in detail
  - Features
  - The generators
  - The embedders
  - The rest
- 3 Demo
- 4 The future (of CaGe)



# What is CaGe?

a graphical user interface for a set of commandline generators and embedders

GUI

written in Java

generators and embedders

written in C and Java (any language will do)



# User interface

Having a user interface  
 $\neq$   
Being a user friendly program



# Example

## CGF

CGF is a generator for cubic plane graphs with given faces.

```
cgf -g 0 -output stdout -logfile stderr -save  
0 -no_recover -topdown -outputmem 0 0 -v 30  
-vs 20 -f 5 112-12u -f 6 -mapcon 3
```

generates all 3-connected cubic plane graphs with 20 to 30 vertices, 12 pentagons and any number of hexagons.



# And now with CaGe

general cubic plane graphs   fullerenes   cubic plane graphs with given faces   bipartite cubic plane graphs

minimum number of vertices   maximum number of vertices

20   30  
4   250   4   250   ☐ min = max

Face Type

5  
3   6   10   15   20   25   30   35   40  

included face types:

☒ 5-gons   ☒ limits   12   12

☒ 6-gons   ☐ limits

☐ graphs with connectivity number 1  
☐ graphs with connectivity number 2  
☒ graphs with connectivity number 3



# Output

CGF and other generators in CaGe output a binary format, e.g. planar code.

☐ Pre-filter graphs

---

☒ 3D representation    ☒ Viewer    ☐ File    ☐ Pipe  
☒ Jmol Viewer    ☐ text viewer

---

☒ 2D representation    ☒ Viewer    ☐ File    ☐ Pipe  
☒ TwoView    ☐ text viewer

---

☐ Adjacency information    ☐ File    ☐ Pipe



# Timeline

## Some large steps...

Mid 90's	First version
2001	Java 1.1 version
March 2007	Mac OS X supported
June 2009	Java 5 version

## ... and a lot of small steps

- new generators and new embedders added
- user interface improved
- documented and cleaned up the code
- bugs fixes



# Features

- visualization of the options
- checking of constraints on the parameters
- automatic selection of best suited generator
- easy browsing through the results of a generator
- embedding in 2D and 3D
- ...



# Generators

## 3-regular plane graphs

- Fullerenes
- 3-regular plane graphs with given face sizes
- Bipartite 3-regular plane graphs
- General 3-regular plane graphs

## 4-regular plane graphs

- 4-regular plane graphs with given face sizes
- General 4-regular plane graphs

## General plane graphs

# Generators

## Triangulations

- Triangulations with given vertex degrees
- Eulerian triangulations
- General triangulations
- Triangulations of the disk

## Quadrangulations

- Quadrangulations with given vertex degrees
- General quadrangulations



# Generators

## Planar Polycyclic Hydrocarbons

- by formula (e.g.  $C_{31}H_{17}$ )
- by boundary formula (e.g. 222322223)
- by number of hexagons

## Tubes and cones

- nanotubes
- nanocones



# Embedders

- *embed*  
A general purpose embedder for fast embedding of planar graphs in the plane, on spheres and on tubes
- *NanotubeEmbedder*  
Specialized for nanotubes: aware of the cap of the nanotube
- *NanoconeEmbedder*  
Specialized for nanocones: aware of the number of pentagons and thus also of the aperture angle



# Output

## Three types of output after embedding

- adjacency information
  - text
- 3D representation
  - Jmol
  - Rasmol (when available)
  - text
- 2D representation (Schlegel diagram)
  - TwoView
  - text



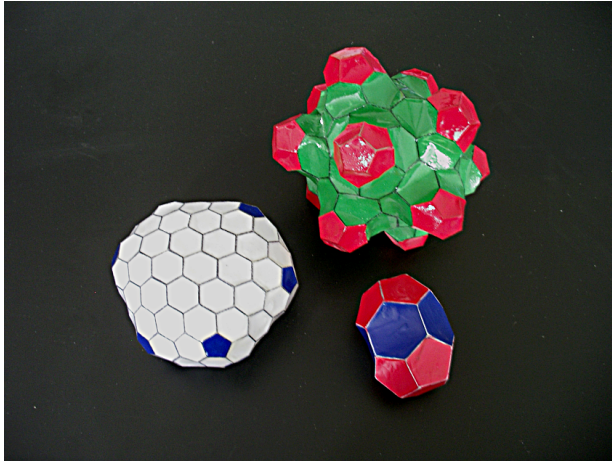
# Output

## Other output options

- send graphs through filter first
- send output to a file
- send output to a pipe
- create unfolding of 3D representation



# Unfolding of 3D representation





# Command-line magic

## Expert mode

Still possible to use unsupported generator/embedder features  
(or even custom embedders)

	generator	<input type="text" value="cgf -g 0 -output stdout -logfile stderr -save 0 -no_recover -ti"/>
generator/embedders	2D embedder	<input type="text" value="embed"/>
	3D embedder	<input type="text" value="embed -d3 -it"/>



# Command-line magic

## External generator

Read graphs from a file or from a custom generator but still use wizard-like configuration for embedding

Generator Options: external generator

☒ generator command:

☐ Input from file:

☒ create new embedding (ignore old coordinates)  
☐ skip embedding (keep old coordinates if provided)  
☐ refine embedding (start with old coordinates if provided)

The expected form of the graphs is

☐ rather flat  
☐ rather spherical  
☐ rather tubular

embed iteration count (% of default value)

10 100 1000

☒ edit embed commands



# DEMO



# Future work

- Improve embedders for non-fullerene-like structures
- Add more classes of graphs
- ... (suggestions?)



# What can you do for CaGe?

- Use it!
- More important: Give feedback
- Programmers: contact us about adding programs to CaGe



## People that contributed to CaGe

Simon Buelens

Gunnar Brinkmann

Olaf Delgado-Friedrichs

Andreas Dress

Thomas Harmuth

Sebastian Lisen

Alexander Lust

Brendan McKay

Adriaan Peeters

Nico Van Cleemput

Ulrike von Nathusius



# Thanks for your attention

Visit our websites:

<http://caagt.ugent.be/CaGe>

<http://www.math.uni-bielefeld.de/~CaGe>

