

The SAGE Graph Theory Project

Robert L. Miller

March 23, 2007

What's already out there?

- Nauty
 - Computes automorphism groups of graphs, and produces canonical labellings
 - Does *nothing* else, but is *very* good at what it does

What's already out there?

- Nauty
 - Computes automorphism groups of graphs, and produces canonical labellings
 - Does *nothing* else, but is *very* good at what it does

The problem: Licensing (emphasis mine)

Copyright (1984-2004) Brendan McKay. All rights reserved. Permission is hereby given for use and/or distribution with the exception of sale for profit or *application with nontrivial military significance* . You must not remove this copyright notice, and you must document any changes that you make to this program. This software is subject to this copyright only, irrespective of any copyright attached to any package of which this is a part.

Absolutely no guarantees or warranties are made concerning the suitability, correctness, or any other aspect of this program. Any use is at your own risk.

What's already out there?

- NetworkX, A GPL Python library
Included as an optional package in SAGE
`sage -i networkx-0.32`

What's already out there?

- NetworkX, A GPL Python library
Included as an optional package in SAGE
`sage -i networkx-0.32`
- GRAPE, an optional extension of GAP
`sage -i gap_packages-4.4.7.4_4.p1`

What's already out there?

- NetworkX, A GPL Python library
Included as an optional package in SAGE
`sage -i networkx-0.32`
- GRAPE, an optional extension of GAP
`sage -i gap_packages-4.4.7.4_4.p1`
- Magma, which is neither open nor free
But it is FAST!

What's already out there?

- NetworkX, A GPL Python library
Included as an optional package in SAGE
`sage -i networkx-0.32`
- GRAPE, an optional extension of GAP
`sage -i gap_packages-4.4.7.4_4.p1`
- Magma, which is neither open nor free
But it is FAST!
- Both Magma and GRAPE interface nauty

Is that all?

- Very nauty

Designed to be compatible with nauty

Extends its functionality to study cliques in $G(n, p)$ and chromatic number

Is that all?

- Very nauty
 - Designed to be compatible with nauty
 - Extends its functionality to study cliques in $G(n, p)$ and chromatic number
- An assortment of very specialized programs and “works in progress” :

Is that all?

- Very nauty
 - Designed to be compatible with nauty
 - Extends its functionality to study cliques in $G(n, p)$ and chromatic number
- An assortment of very specialized programs and “works in progress” :
 - 'laplacian.mpl' (extends Maple), Gato, GeoSteiner, GraphViz, plantri and fullgen, GUESS, CAIDA's Walrus, GraphThing, GRIN, DSATUR, LEDA, LINK, ABACUS, and dozens of other acronyms

What about Steve Wolfram?

To be fair, we should also list

What about Steve Wolfram?

To be fair, we should also list

- Mathematica, which only provides plotting support (and a few basic algorithms) on its own

What about Steve Wolfram?

To be fair, we should also list

- Mathematica, which only provides plotting support (and a few basic algorithms) on its own
- Combinatorica, which extends Mathematica and is free for download (but you have to buy the book if you want documentation)

What about Steve Wolfram?

To be fair, we should also list

- Mathematica, which only provides plotting support (and a few basic algorithms) on its own
- Combinatorica, which extends Mathematica and is free for download (but you have to buy the book if you want documentation)
- Maple, which provides a modest assortment of functionality and a decent plotting package

What about Steve Wolfram?

To be fair, we should also list

- Mathematica, which only provides plotting support (and a few basic algorithms) on its own
 - Combinatorica, which extends Mathematica and is free for download (but you have to buy the book if you want documentation)
 - Maple, which provides a modest assortment of functionality and a decent plotting package
- We'll return to these later...

So what is the state of graph theory computation?

- There is no authority on graph theory computation
- Each program emphasizes different aspects of the field, no program is comprehensive
- With the exception of Magma, the fastest programs out there are already available as optional packages in SAGE

Car parts

- The software survey

`http://sage.math.washington.edu:9001/graph_survey`

Car parts

- The software survey

`http://sage.math.washington.edu:9001/graph_survey`

Did we miss anything??

Car parts

- The software survey
`http://sage.math.washington.edu:9001/graph_survey`
Did we miss anything??
- NetworkX

Drag racing

- The benchmarks
`http://sage.math.washington.edu:9001/graph_benchmark`

Drag racing

- The benchmarks
`http://sage.math.washington.edu:9001/graph_benchmark`
- A sample: computing the diameter of fifteen random graphs in $G(100, .45)$:

Drag racing

- The benchmarks
http://sage.math.washington.edu:9001/graph_benchmark
- A sample: computing the diameter of fifteen random graphs in $G(100, .45)$:

Magma 0.00085 s

Drag racing

- The benchmarks
http://sage.math.washington.edu:9001/graph_benchmark
- A sample: computing the diameter of fifteen random graphs in $G(100, .45)$:

Magma	0.00085 s	
NetworkX	0.1263 s	(factor of 149)

Drag racing

- The benchmarks
http://sage.math.washington.edu:9001/graph_benchmark
- A sample: computing the diameter of fifteen random graphs in $G(100, .45)$:

Magma	0.00085 s	
NetworkX	0.1263 s	(factor of 149)
GRAPE	0.157 s	(factor of 1.25)

Drag racing

- The benchmarks
http://sage.math.washington.edu:9001/graph_benchmark
- A sample: computing the diameter of fifteen random graphs in $G(100, .45)$:

Magma	0.00085 s	
NetworkX	0.1263 s	(factor of 149)
GRAPE	0.157 s	(factor of 1.25)
Combinatorica	5.2114 s	(factor of 33)

Drag racing

- The benchmarks
http://sage.math.washington.edu:9001/graph_benchmark
- A sample: computing the diameter of fifteen random graphs in $G(100, .45)$:

Magma	0.00085 s	
NetworkX	0.1263 s	(factor of 149)
GRAPE	0.157 s	(factor of 1.25)
Combinatorica	5.2114 s	(factor of 33)
Maple	19.5256 s	(factor of 3.75)

A SAGE Graph Class

- Architecture
 - NetworkX model: graphs are stored as "a node-centric dictionary of dictionaries" – each node is essentially a list of its neighbors

This demonstrates the elegance (and speed) of Python

A SAGE Graph Class

- Architecture
 - NetworkX model: graphs are stored as "a node-centric dictionary of dictionaries" – each node is essentially a list of its neighbors
 - This demonstrates the elegance (and speed) of Python
 - What is Magma doing differently?

A SAGE Graph Class

- Architecture
 - NetworkX model: graphs are stored as "a node-centric dictionary of dictionaries" – each node is essentially a list of its neighbors

This demonstrates the elegance (and speed) of Python

 - What is Magma doing differently?
 - What is best for SAGE?
 - Build a Graph class with built-in utilization of NetworkX
 - Provide Graph-native interfaces to GRAPE and Magma
 - Graph methods for everything in SAGE that has a natural graph associated to it, e.g. Cayley graphs

Features of the Graph Class

- Algorithms
 - A great way for undergraduates to get involved with SAGE: a student can take an algorithm, research it thoroughly, and implement it— one-week projects to quarter courses

Features of the Graph Class

- Algorithms
 - A great way for undergraduates to get involved with SAGE: a student can take an algorithm, research it thoroughly, and implement it— one-week projects to quarter courses
- Visualization

Features of the Graph Class

- Algorithms
 - A great way for undergraduates to get involved with SAGE: a student can take an algorithm, research it thoroughly, and implement it– one-week projects to quarter courses
- Visualization
 - Ray casting with Alex Clemesha

Features of the Graph Class

- Algorithms
 - A great way for undergraduates to get involved with SAGE: a student can take an algorithm, research it thoroughly, and implement it— one-week projects to quarter courses
- Visualization
 - Ray casting with Alex Clemesha
 - Interactive Java applet for working hands-on in the notebook?

Another car part: CAIDA's Walrus (GPL)

Yet another: Jython (GPL)

Feedback

The most important time for feedback is right now.