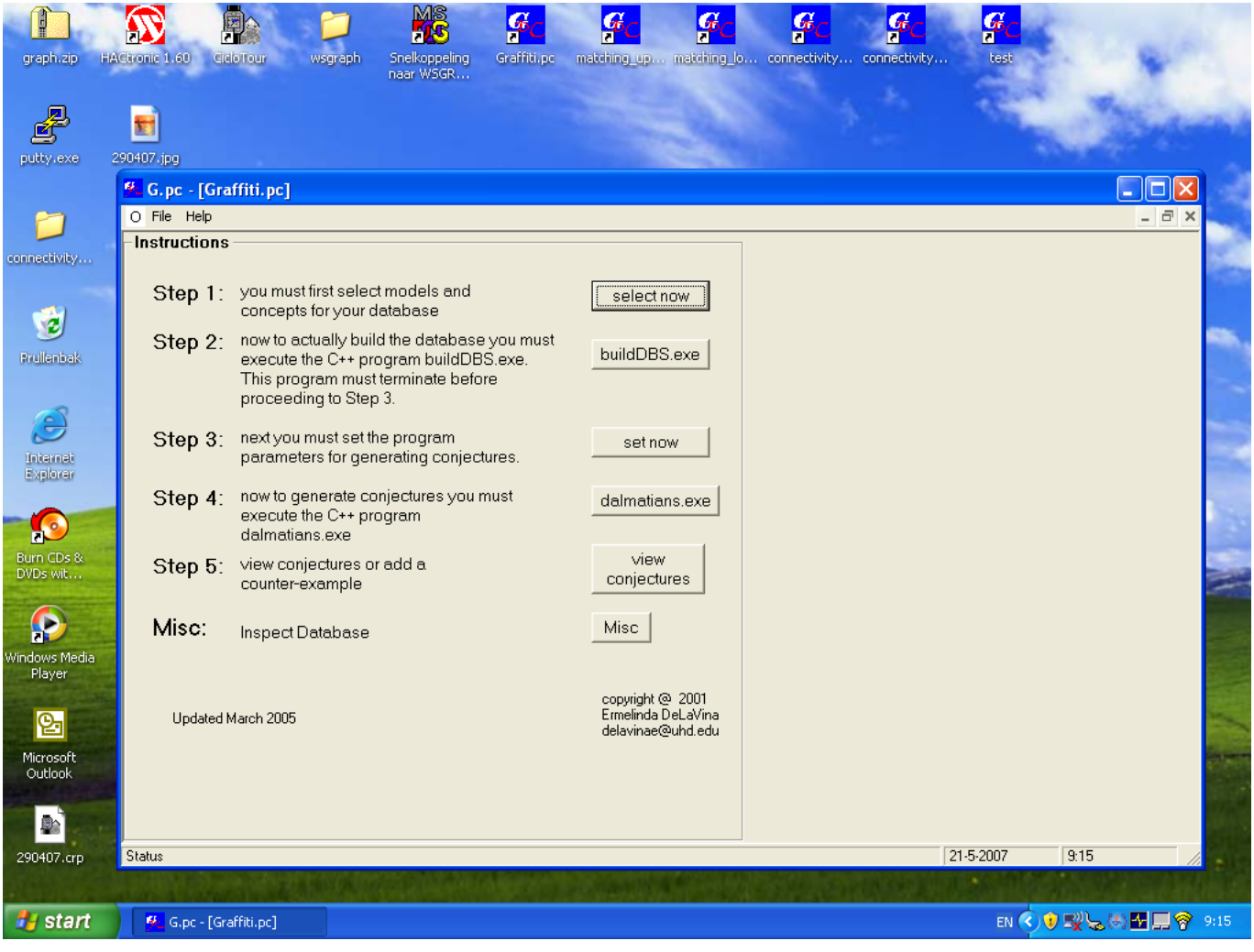
# GrInvIn for Teaching and Research

Gunnar Brinkmann and
Nicolas van Cleemput

Joint work with
Kris Coolsaet, Veerle Fack, Adriaan Peeters
(all University of Ghent, Belgium)

# Motivation:

Teaching with
Graffiti (S. Fajtlovicz)
and
Graffiti.pc (E. DeLaVina).


graph.zip
HACtronic 1.60
CicloTour
wsgraph
Snelkoppeling naar WSGR...
Graffiti.pc
matching_up...
matching_lo...
connectivity...
connectivity...
test
putty.exe
290407.jpg
connectivity...
Prullenbak
Internet Explorer
Burn CDs & DVDs wit...
Windows Media Player
Microsoft Outlook
290407.crp
G.pc - [Graffiti.pc]
File Help
Instructions
Step 1: you must first select models and concepts for your database
select now
Step 2: now to actually build the database you must execute the C++ program buildDBS.exe. This program must terminate before proceeding to Step 3.
buildDBS.exe
Step 3: next you must set the program parameters for generating conjectures.
set now
Step 4: now to generate conjectures you must execute the C++ program dalmatians.exe
dalmatians.exe
Step 5: view conjectures or add a counter-example
view conjectures
Misc: Inspect Database
Misc
Updated March 2005
copyright @ 2001
Ermelinda DeLaVina
delavinae@uhd.edu
Status
21-5-2007
9:15
start
G.pc - [Graffiti.pc]
EN
9:15

# Teaching was done

- on university level:
  USA, Germany, Serbia (Graffiti, Graffiti.pc)
  Belgium (GrInvIn)

- in a course for highschool teachers:
  Germany (Graffiti.pc),
  Belgium (GrInvIn, a short introduction)

- in some lectures for highschool students:
  Belgium (GrInvIn)

## Results of the courses:

- The approach via trying to prove or disprove conjectures is very motivating for students!

- Graffiti.pc would need extensive improvements before it would be suitable for an educational environment.

So **SOMEONE** should develop a software that's more user friendly and has a better design...

Jonathan Berry (mathematician, now Sandia, before that he developed *LINK* while at DIMACS):

If you want good and usable mathematical software, a software engineer must be involved at a **responsible** position!

## Important features for teaching:

- easily adapted to various languages – just translate a language file

- documentation of invariants

## Important features for teaching:

- case sensitive help functions

- runs on every platform:
  unix, gnu/linux, mac, windows

# How can you use it in teaching?

## Rules:

- Choose one **fixed** invariant and other invariants to compare it to.
  e.g.: I want to compare the girth with the diameter, the clique number, the...

- Put a small graph (e.g. $K_3$) into the list.

- Then repeat the following steps:
  - Let the program make a conjecture.
  - * In case the conjecture is correct: Prove it and afterwards (e.g.) remove a non-fixed invariant involved from the invariant list.
    * In case the conjecture is wrong: Give a smallest counterexample, prove minimality and add the counterexample to the list.

# Why does it work so well ?

- Students identify themselves with **their** invariant.
- It has the flair of discovery. . .
- Students see the need to formulate their own lemmas.

# Important

Students want to get a *normal* graph theory course afterwards!

*Demonstration.*

## Ongoing work to support teaching:

- Development of an easy handbook for teachers with given scenarios to *play safe*.
- Development of special teaching versions where the teacher's version has more possibilities than the student's version.
- Making GrInvIn print out a text describing the session (conjectures made, graphs inputted, etc).

## Ongoing work to support teaching:

- Improving the conjecturing engine – especially for teaching.

- Restriction to given classes (e.g. only planar graphs, only cubic graphs).

## Important features for research:

- documentation of how invariants were tested

- easy extensibility

*Demonstration*

## Ongoing work to support research:

- Make a database of *interesting* graphs – that is graphs that already served as counterexamples.
- Include graph generation programs and counterexample finding programs.
- Automatic improvement of conjectures using these routines.
- etc. . . .

A first version is available at

# grinvin.org

## It would be great if you

- would test it.
- use it in teaching (maybe with the next version of the conjecturing engine).
- mail us about problems, suggestions, ideas, etc. That's not lost work – GrInvIn **is continuously improved!**

## It would be great if you

- give a course for highschool teachers in your area and try to convince them that graph theory is **ideal** for teaching **mathematical and logical reasoning** at highschool level.

- were interested in cooperation. . .