Advanced Algorithmics (6EAP)

Project proposals

Jaak Vilo

2013 Spring
Key info

• Project = 1-2-3 person teams
• Poster session: May 29th (Wed)
  – 2pm - room L122
  – Poster file (PDF?) has to be uploaded before
• Prerequisite for exam
Expectations:

• Study the problem
• Implement, Evaluate, Compare, Measure, ...
• Your task is to make the project interesting to others: right questions; cool applications; novel ideas; desire to read; materials to complement next year courses.

• Find a clear objective and focus, state it, study it!
• 20-40h / per person
• Report – Poster
Tasks

• Here is a list of some proposals

• You can propose your own.

• Or select some on your own
  – from international competitions
    • e.g. IOI (ACM) olympics finals series
    • implementation challenges from DIMACS, etc.
    • etc.
• Compare some alternative algorithms, and try to improve on them
• Take a problem and try to apply algorithmic problems to solve it
• Take an algorithm from the literature, implement and test
• Find cool ways to study/visualise algorithmic ideas presented in the course.
Succinct data structures

• Test in practice the size and speed for some succinct data structures (binary search trees, heaps ...)

• Could you support some (any) dynamic update operations? Which kind? What would be needed basic operations for updates? (however bad the time is)
Clustering using Differential Evolution

- Define an objective function
- Define a numeric vector representing a clustering
- Optimise the objective function using differential result
- Report the types of clusters discovered, time and convergence, strengths and weaknesses as compared to other standard clustering methods
Combinatorial search (BFS)

- **Optimal** solution from a (or any) state

- “Discover” the short assembly step algorithms

- Provide solutions
Graph layout

• Graph layout
  – “Physical Spring model” with some extra added constraints or specialised nodes for stars, cliques, connection strength, etc.

• Create a nr of criteria and try to minimize nr of crossings, area of graph, etc.
Constrained Spring Embedding Layout

- Define certain areas (or lines, etc) that “attract” nodes. Allow graph to “layout” itself dynamically.
TSP variant, but with physical laws of velocity...
Visit all cities... - physically!

http://cswww.essex.ac.uk/staff/sml/gecco/PTSPComp.html
http://algoval.essex.ac.uk/ptsp/ptsp.html
942, 941
• 652, 652
• 648, 636
Seriation

- [link](http://courses.cs.ut.ee/2009/dm/Main/HW04)
- Serialise matrices
- (2-way)
Biclustering

- Ordering rows and columns to reveal modules/areas of high “coherence”

Query of OCT4 (POU5F1) (210265_X_AT)

StdDev < 0.29

http://eid.ee/b3
Some algorithmic competition

• Test your skills on some algorithmic competition
Finally, 15,000 pages later:

```
-7  260  0
 7  -260  0
 1072 1070  0
-15 -14 -13 -12 -11 -10  0
-15 -14 -13 -12 -11  10  0
-15 -14 -13 -12  11 -10  0
-15 -14 -13 -12  11  10  0
-7  -6  -5  -4  -3  -2  0
-7  -6  -5  -4  -3  2  0
-7  -6  -5  -4  3  -2  0
-7  -6  -5  -4  3  2  0
 185  0
```

Combinatorial search space of truth assignments:

\[ 2^{50000} \approx 3.160699437 \cdot 10^{15051} \]

Current SAT solvers solve this instance in approx. 1 minute!
Your own projects

• Ask a question
• Study literature
• Propose solution
• Implement
• Experiment and report results of experiments