Cutting And Packing Algorithms Research Framework http://caparf.googlecode.com

Denis Nazarov

Ufa State Technical University of Aviation

12 July 2010

Algorithm development challenges/issues

- Algorithm details are often omitted in articles
- It's hard to get/generate test instances proposed by other authors
- Different system configurations for testing
- Small number of review articles with algorithms comparisons

Desired solution

Consists of

- Algorithms (i.e. lower bounds/upper bounds/exact methods) for solving particular type(s) of problems
- Test data generators

Provides

- Convenient «testing scenario» model
- Various reports generators (for example, results for article in TeX or presentation)
- Easy way to develop your own algorithms using existing methods and compare them with well-known reference algorithms by other authors

Cutting and Packing Algorithms Research Framework

1. CAPARF is an OpenSource project

   hosted at http://code.google.com under GPLv3 licence
2. CAPARF is a cross-platform framework

   it is implemented in Java
3. CAPARF forces unification

   all algorithms for the same type of problem implement the same interface
4. CAPARF potentially can contain a lot of algorithms and test data/generators

Problem type concept

Problem type is defined by `Input` and `Output`.

`Input` is represented by

- ordered list of items to pack
- and, possibly, ordered list of containers into which items are packed

`Output` is represented by

- ordered list of item placements
- objective function

`OutputVerifier` is implemented for each problem type, i.e. `Input` and `Output`.

Algorithm concept

Algorithm can be defined for particular problem type or for some class of problem types.

Algorithm types

1. Lower bounds
2. Upper bounds
3. Exact methods

Interruption of algorithm computations can be safely done by implementing `Interruptible` interface.

Test data generator and scenario concepts

Test data generator provides reference and random `Inputs`.
Scenario is defined for particular problem type.

Scenario components

- Algorithms of the same type to run
- Test data on which algorithms should be run
- Technical constraints (like time limit)

Report generator(s) can be added to produce statistics/results in preferable format.

Contribution

CAPARF can be successful and useful only as a community project.

Possible types of contribution

1. Ideas, suggestions, criticism
2. Feature requests, bug reports
3. Code changes, including but not limited to:
   1. Adding your algorithms, test generators
   2. Changing CAPARF core, adding more functionality

Contacts

For more information visit

http://caparf.googlecode.com

or mail to

caparf-discuss@googlegroups.com