Tabu Graph Coloring

Generated by Doxygen 1.8.13

Contents

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 2

Class Documentation

2.1 ArgList Struct Reference

Public Attributes

- unsigned int **nColors**
- unsigned int **nIterations**
- unsigned int **tabuSize**
- unsigned int **nNeighbours**

The documentation for this struct was generated from the following file:

- include/utils.h

2.2 ColorClass Class Reference

This class represents one color or class of vertices - e.t. it keeps a set of vertices belonging to the same color. It might be empty.

```
#include <colorClass.h>
```

Public Member Functions

- **ColorClass** (const Graph &_g)
- **ColorClass** (const ColorClass &nC)
- ColorClass & **operator=** (const ColorClass &c)
- void updateConflicts ()

    Recalculate number of conflicts in a class.
- void updateCost ()

    Recalculate current cost function value based on the precalculated number of conflicts.
- int conflictsChangeAfterRemoval (unsigned int v) const

    Calculate change in number of conflicts in a class if vertex v is removed without really removing it.
- int conflictsChangeAfterAdding (unsigned int v) const

Calculate change in number of conflicts in a class if vertex v is added without really adding it.

- int costChangeAfterRemoval (unsigned int v) const

  Calculate cost function value change for the class if vertex v is removed without really removing it.
- int costChangeAfterAdding (unsigned int v) const

  Calculate cost function value change for the class, if vertex v is added without really adding it.
- void addVertex (unsigned int v)

  Add a new vertex to the class.
- int getCost () const

  Get current cost based on a current number of conflicts.
- int getNConflicts () const

  Get current number of conflicts in a class.
- bool isEmpty () const

  Is class empty.
- size_t nVertices () const

  Get number of vertices in a class.
- unsigned int getVertex (unsigned int index) const

  Find vertex with index.
- std::list< unsigned int > getVertices () const

  Obtain list of vertices belonging to the class.

Static Public Member Functions

- static void performMove (ColorClass &from, ColorClass &to, unsigned int v)

  Static method used to move one vertex from one class to another.

2.2.1 Detailed Description

This class represents one color or class of vertices - e.t. it keeps a set of vertices belonging to the same color. It might be empty.

2.2.2 Member Function Documentation

2.2.2.1 conflictsChangeAfterAdding()

```
int ColorClass::conflictsChangeAfterAdding (
            unsigned int v ) const
```

Calculate change in number of conflicts in a class if vertex v is added without really adding it.

Parameters

v	A vertex to be added.

Returns

Change in number of conflicts.

2.2.2.2 conflictsChangeAfterRemoval()

```
int ColorClass::conflictsChangeAfterRemoval (
            unsigned int v ) const
```

Calculate change in number of conflicts in a class if vertex v is removed without really removing it.

Parameters

v	A vertex to be removed

Returns

Change in number of conflicts

2.2.2.3 costChangeAfterAdding()

```
int ColorClass::costChangeAfterAdding (
            unsigned int v ) const
```

Calculate cost function value change for the class, if vertex v is added without really adding it.

Parameters

v	Vertex to be added

Returns

Change in cost function value.

2.2.2.4 costChangeAfterRemoval()

```
int ColorClass::costChangeAfterRemoval (
            unsigned int v ) const
```

Calculate cost function value change for the class if vertex v is removed without really removing it.

Parameters

v	A vertex to removed.

Returns

Change in cost function value

2.2.2.5 getCost()

```
int ColorClass::getCost ( ) const
```

Get current cost based on a current number of conflicts.

Returns

Value of cost function.

2.2.2.6 getVertex()

```
unsigned int ColorClass::getVertex (
             unsigned int index ) const
```

Find vertex with index.

Parameters

index	Index of a vertex to be returned

Returns

Vertex

2.2.2.7 getVertices()

```
std::list<unsigned int> ColorClass::getVertices ( ) const
```

Obtain list of vertices belonging to the class.

Returns

list of vertices belonging to the class

2.2.2.8 nVertices()

```
size_t ColorClass::nVertices ( ) const
```

Get number of vertices in a class.

Returns

Number of vertices in a class.

2.2.2.9 performMove()

```
static void ColorClass::performMove (
            ColorClass & from,
            ColorClass & to,
            unsigned int v )  [static]
```

Static method used to move one vertex from one class to another.

Parameters

from	ColorClass from which a vertex should be taken
to	ColorClass to which a vertex should be moved
v	vertex

This method updates number of conflicts in the 'from' and 'to' classes. It also updates the cost function and performs the vertex move.

The documentation for this class was generated from the following file:

- include/colorClass.h

2.3 TabuSearch Class Reference

A class, which implements the tabu search algorithm.

```
#include <tabu.h>
```

Public Member Functions

- **TabuSearch** (unsigned int nIterations, unsigned int tabuSize, size_t kColors, unsigned int nNeighbours, const Graph &ng)
- int getCost () const

  Get current best cost function value.
- unsigned int getChromaticNumber () const

  Get curent best chromatic number found.

- unsigned int getNumberOfConflicts () const

    Get number of conflicts in a currently best solution.
- Solution getSolution () const

    Get currently best solution.
- void optimize (bool verbose)

    This method performs tabu search algorithm.

2.3.1 Detailed Description

A class, which implements the tabu search algorithm.

2.3.2 Member Function Documentation

2.3.2.1 getSolution()

```
Solution TabuSearch::getSolution ( ) const
```

Get currently best solution.

Returns

Best solution found - a vector of ColorClass

2.3.2.2 optimize()

```
void TabuSearch::optimize (
            bool verbose )
```

This method performs tabu search algorithm.

Parameters

verbose	If set to true it prints all cost function values

The documentation for this class was generated from the following file:

- include/tabu.h