Parallel pagerank

0.1

Generated by Doxygen 1.8.15

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 3

Data Structure Documentation

3.1 NTWPR_CRS Struct Reference

A Compressed Row Storage struct.

```
#include <ntwpr_wg.h>
```

Data Fields

- uint32_t edge_num
- uint32_t node_num
- double ∗ val
- uint32_t ∗ col_ind
- uint32_t ∗ row_ptr

3.1.1 Detailed Description

A Compressed Row Storage struct.

Used to save web graph tables in RAM. See CRS

3.1.2 Field Documentation

3.1.2.1 col_ind

```
uint32_t* NTWPR_CRS::col_ind
```

A vector containing the columns of the val's values.

3.1.2.2 edge_num

```
uint32_t NTWPR_CRS::edge_num
```

The number of non zero elements (edges)

3.1.2.3 node_num

```
uint32_t NTWPR_CRS::node_num
```

The number of nodes of the graph.

3.1.2.4 row_ptr

```
uint32_t* NTWPR_CRS::row_ptr
```

A pointer to the val vector indicating the start of a matrix' row.

3.1.2.5 val

```
double* NTWPR_CRS::val
```

The non zero values of the matrix.

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

- /home/ecat/nikos/parallel/source/helpers/ntwpr_wg.h

3.2 NTWPR_WGFile Struct Reference

A struct containing a web-graph file and the graphs info.

```
#include <ntwpr_wg.h>
```

Data Fields

- uint32_t edge_num
- uint32_t node_num
- FILE * edge_data

3.2.1 Detailed Description

A struct containing a web-graph file and the graphs info.

Used as a "middle" save format. This API uses this type to compute the pagerank of the nodes. Any dataset's pagerank vector can be computed by extracting such struct from the dataset.

3.2.2 Field Documentation

3.2.2.1 edge_data

```
FILE* NTWPR_WGFile::edge_data
```

The edges represented as node pairs

3.2.2.2 edge_num

```
uint32_t NTWPR_WGFile::edge_num
```

The number of the graph's edges.

3.2.2.3 node_num

```
uint32_t NTWPR_WGFile::node_num
```

The number of the graph's nodes.

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

- /home/ecat/nikos/parallel/source/helpers/ntwpr_wg.h

Chapter 4

File Documentation

4.1 /home/ecat/nikos/parallel/source/helpers/ntwpr_wg.c File Reference

File containing the implementations of ntwpr_wg.h.

```
#include "ntwpr_wg.h"
```

Functions

- NTWPR_CRS ∗ NTWPR_load2crs (NTWPR_WGFile ∗restrict NTWPR_in_fp)

  Loads a web graph in memory using the CRS data structure.
- NTWPR_WGFile ∗ NTWPR_WGfopen (const char path[static 1])

  Creates a new WGFile at path.
- int NTWPR_WGfclose (NTWPR_WGFile ∗wgfile)

  Closes the NTWPR_WGFile stream and frees memory.
- bool NTWPR_WGFile_Reset (NTWPR_WGFile ∗const NTWPR_WGF)

  Reset the NTWPR_WGF struct to read the edges again.
- void NTWPR_expfm (NTWPR_WGFile ∗restrict wgfp, const char exp_path[static 1], uint32_t NTWPR_↩ node_num)

  Exports the full graph matrix of a NTWPR_WGFile to the file at exp_path.
- void NTWPR_SU2WG (const char SU_WGD_path[static 1], const char exp_path[static 1], uint32_t NTWP↩ R_node_num)

  Converts web graph files from the format Stanford U. (SU) used to the NTWPR_WGFile' edge_data file format.

4.1.1 Detailed Description

File containing the implementations of ntwpr_wg.h.

Author

Katomeris Nikolaos, 8551, ngkatomer@auth.gr

Date

09-08-2018

4.1.2 Function Documentation

4.1.2.1 NTWPR_expfm()

```
void NTWPR_expfm (
            NTWPR_WGFile *restrict NTWPR_WGF,
            const char exp_path[static 1],
            uint32_t NTWPR_node_num )
```

Exports the full graph matrix of a NTWPR_WGFile to the file at *exp_path*.

Should be used with caution and for relatively small *NTWPR_node_num*. (NTWPR_node_num < 10000)

Parameters

NTWPR_WGF	The input NTWPR_WGFile pointer.
exp_path	The output file's path.
NTWPR_node_num	The number of nodes that will be in the graph matrix.

4.1.2.2 NTWPR_load2crs()

```
NTWPR_CRS* NTWPR_load2crs (
            NTWPR_WGFile *restrict NTWPR_in_fp )
```

Loads a web graph in memory using the CRS data structure.

Parameters

NTWPR_in← _fp	The input web graph data.

Returns

NTWPR_CRS∗ pointing to the loaded CRS or null.

4.1.2.3 NTWPR_SU2WG()

```
void NTWPR_SU2WG (
            const char SU_WGD_path[static 1],
            const char exp_path[static 1],
            uint32_t NTWPR_node_num )
```

Converts web graph files from the format Stanford U. (SU) used to the NTWPR_WGFile' edge_data file format.

Extra: The output can be loaded into MATLAB with: fread(file_id, [2,inf], 'uint32')';

Parameters

SU_WGD_path	The input file's path..
exp_path	The output file's path.
NTWPR_node_num	The number of nodes that will be in the graph matrix. (Enter 0 to get all the nodes of the file.)

4.1.2.4 NTWPR_WGfclose()

```
int NTWPR_WGfclose (
            NTWPR_WGFile * wgfile )
```

Closes the NTWPR_WGFile stream and frees memory.

Parameters

wgfile	The file to be closed.

Returns

int Returns 0 if succussful. On failure EOF is returned.

4.1.2.5 NTWPR_WGFile_Reset()

```
bool NTWPR_WGFile_Reset (
            NTWPR_WGFile *const NTWPR_WGF )
```

Reset the *NTWPR_WGF* struct to read the edges again.

Parameters

NTWPR_WGF	The NTWPR_WGFile struct.

Returns

true If successful.
false If an error occured.

4.1.2.6 NTWPR_WGfopen()

```
NTWPR_WGFile* NTWPR_WGfopen (
            const char path[static 1] )
```

Creates a new WGFile at *path*.

The WGFile should be then closed with NTWPR_WGfclose()

Parameters

path	The path of the file.

Returns

NTWPR_WGFile∗ The struct containing the file created.

4.2 /home/ecat/nikos/parallel/source/helpers/ntwpr_wg.h File Reference

Web-Graph import/export functions.

```
#include <stdio.h>
#include <stdbool.h>
#include <stdlib.h>
#include <stdint.h>
```

Data Structures

- struct NTWPR_CRS

  A Compressed Row Storage struct.
- struct NTWPR_WGFile

  A struct containing a web-graph file and the graphs info.

Macros

- #define NTWPR_WG_H 1

  Macro used to not redifine header definitions when this file is included to other sources.
- #define DF_SU_LINE_COMMENT '#'

  The character used to indicate comment lines in the Stanford's (SU) data files.
- #define NTWPR_invalid_fp_exit(fp)

  Checks for null file pointers returning debug message.

Typedefs

- typedef struct NTWPR_CRS NTWPR_CRS

  A Compressed Row Storage struct.
- typedef struct NTWPR_WGFile NTWPR_WGFile

  A struct containing a web-graph file and the graphs info.

Functions

- NTWPR_WGFile ∗ NTWPR_WGfopen (const char path[static 1])

  Creates a new WGFile at path.
- int NTWPR_WGfclose (NTWPR_WGFile ∗wgfile)

  Closes the NTWPR_WGFile stream and frees memory.
- bool NTWPR_WGFile_Reset (NTWPR_WGFile ∗const NTWPR_WGF)

  Reset the NTWPR_WGF struct to read the edges again.
- NTWPR_CRS ∗ NTWPR_load2crs (NTWPR_WGFile ∗restrict NTWPR_in_fp)

  Loads a web graph in memory using the CRS data structure.
- void NTWPR_SU2WG (const char SU_WGD_path[static 1], const char exp_path[static 1], uint32_t NTWP↩ R_node_num)

  Converts web graph files from the format Stanford U. (SU) used to the NTWPR_WGFile' edge_data file format.
- void NTWPR_expfm (NTWPR_WGFile ∗restrict NTWPR_WGF, const char exp_path[static 1], uint32_t NT↩ WPR_node_num)

  Exports the full graph matrix of a NTWPR_WGFile to the file at exp_path.

4.2.1 Detailed Description

Web-Graph import/export functions.

Currently supported data:

- Files containing sorted edges of web graphs like the ones found at: Stanford Large Network Dataset Collection

Author

Katomeris Nikolaos

Date

09-08-2018

4.2.2 Macro Definition Documentation

4.2.2.1 DF_SU_LINE_COMMENT

```
#define DF_SU_LINE_COMMENT '#'
```

The character used to indicate comment lines in the Stanford's (SU) data files.

4.2.2.2 NTWPR_invalid_fp_exit

```
#define NTWPR_invalid_fp_exit(
                fp )
```

Value:

```
do {                                                                    \
    if (!fp)                                                            \
    {                                                                   \
        fprintf(stderr,                                                 \
            "%s:%d: Invalid file pointer.\n", __FILE__, __LINE__);   \
        exit(EXIT_FAILURE);                                             \
    }                                                                   \
} while(false)                                                          \
```

Checks for null file pointers returning debug message.

Parameters

fp	The file pointer to check for null.

4.2.3 Typedef Documentation

4.2.3.1 NTWPR_CRS

```
typedef struct NTWPR_CRS NTWPR_CRS
```

A Compressed Row Storage struct.

Used to save web graph tables in RAM. See CRS

4.2.3.2 NTWPR_WGFile

```
typedef struct NTWPR_WGFile NTWPR_WGFile
```

A struct containing a web-graph file and the graphs info.

Used as a "middle" save format. This API uses this type to compute the pagerank of the nodes. Any dataset's pagerank vector can be computed by extracting such struct from the dataset.

4.2.4 Function Documentation

4.2.4.1 NTWPR_expfm()

```
void NTWPR_expfm (
            NTWPR_WGFile *restrict NTWPR_WGF,
            const char exp_path[static 1],
            uint32_t NTWPR_node_num )
```

Exports the full graph matrix of a NTWPR_WGFile to the file at *exp_path*.

Should be used with caution and for relatively small *NTWPR_node_num*. (NTWPR_node_num < 10000)

Parameters

NTWPR_WGF	The input NTWPR_WGFile pointer.
exp_path	The output file's path.
NTWPR_node_num	The number of nodes that will be in the graph matrix.

4.2.4.2 NTWPR_load2crs()

```
NTWPR_CRS* NTWPR_load2crs (
             NTWPR_WGFile *restrict NTWPR_in_fp )
```

Loads a web graph in memory using the CRS data structure.

Parameters

NTWPR_in↩_fp	The input web graph data.

Returns

NTWPR_CRS∗ pointing to the loaded CRS or null.

4.2.4.3 NTWPR_SU2WG()

```
void NTWPR_SU2WG (
             const char SU_WGD_path[static 1],
             const char exp_path[static 1],
             uint32_t NTWPR_node_num )
```

Converts web graph files from the format Stanford U. (SU) used to the NTWPR_WGFile' edge_data file format.

Extra: The output can be loaded into MATLAB with: fread(file_id, [2,inf], 'uint32')';

Parameters

SU_WGD_path	The input file's path..
exp_path	The output file's path.
NTWPR_node_num	The number of nodes that will be in the graph matrix. (Enter 0 to get all the nodes of the file.)

4.2.4.4 NTWPR_WGfclose()

```
int NTWPR_WGfclose (
             NTWPR_WGFile * wgfile )
```

Closes the NTWPR_WGFile stream and frees memory.

Parameters

wgfile	The file to be closed.

Returns

int Returns 0 if succussful. On failure EOF is returned.

4.2.4.5 NTWPR_WGFile_Reset()

```
bool NTWPR_WGFile_Reset (
             NTWPR_WGFile *const NTWPR_WGF )
```

Reset the *NTWPR_WGF* struct to read the edges again.

Parameters

NTWPR_WGF	The NTWPR_WGFile struct.

Returns

true If successful.
false If an error occured.

4.2.4.6 NTWPR_WGfopen()

```
NTWPR_WGFile* NTWPR_WGfopen (
             const char path[static 1] )
```

Creates a new WGFile at *path*.

The WGFile should be then closed with NTWPR_WGfclose()

Parameters

path	The path of the file.

Returns

NTWPR_WGFile∗ The struct containing the file created.

4.3 /home/ecat/nikos/parallel/source/helpers/test.c File Reference

Test script used for development.

```
#include <stdio.h>
#include <stdlib.h>
#include "ntwpr_wg.h"
```

Functions

- int **main** (int argc, char ∗argv[argc+1])

4.3.1 Detailed Description

Test script used for development.

Author

Katomeris Nikolaos, 8551, ngkatomer@auth.gr

Date

09-08-2018

Index