

ATSuite C++

v1.0

Generated by Doxygen 1.8.12



# Contents

<b>1</b>	<b>File Index</b>	<b>1</b>
1.1	File List . . . . .	1
<b>2</b>	<b>File Documentation</b>	<b>3</b>
2.1	/Users/atanet/PhD/dev/ATSuite/cpp/atgraph.hpp File Reference . . . . .	3
2.1.1	Typedef Documentation . . . . .	4
2.1.1.1	SpMat . . . . .	4
2.1.1.2	SpMatCSR . . . . .	4
2.1.1.3	Tri . . . . .	4
2.1.2	Function Documentation . . . . .	4
2.1.2.1	addCol2Col(SpMat *T, int j_src, int j_dst) . . . . .	4
2.1.2.2	addCol2ColTriplet(SpMat *T, int jSrc, int jDst) . . . . .	4
2.1.2.3	addRow2Row(SpMat *T, int i_src, int i_dst) . . . . .	4
2.1.2.4	addRow2RowTriplet(SpMat *T, int iSrc, int iDst) . . . . .	4
2.1.2.5	addRow2RowTriplet(SpMat *T, int iSrc, int iDst, SpMatCSR *TCSR) . . . . .	4
2.1.2.6	array2igraph(FILE *f, int N, igraph_t *dst_graph) . . . . .	4
2.1.2.7	EigenSparse2Pajek(FILE *f, SpMatCSR *P) . . . . .	4
2.1.2.8	getModularity(SpMat *T) . . . . .	4
2.1.2.9	igraph2EigenSparse(igraph_t *srcGraph) . . . . .	4
2.1.2.10	pajek2EigenSparse(FILE *f) . . . . .	4
2.1.2.11	pajek2igraph(FILE *f, igraph_t *dst_graph) . . . . .	4
2.1.2.12	pajek2igraphNoVertices(FILE *f, igraph_t *dst_graph, int vertex_id0) . . . . .	4
2.1.2.13	pajek2igraphSym(FILE *f, igraph_t *dst_graph) . . . . .	4
2.1.2.14	scalProdInner(SpMat *T, int outer, double coef) . . . . .	4

2.1.2.15	scalProdOuter(SpMat *T, int inner, double coef)	4
2.2	/Users/atanet/PhD/dev/ATSuite/cpp/atgraph_alglib.hpp File Reference	4
2.2.1	Macro Definition Documentation	5
2.2.1.1	plog2p	5
2.2.2	Typedef Documentation	5
2.2.2.1	spAlg	5
2.2.3	Function Documentation	5
2.2.3.1	entropy(real_1d_array *dist)	5
2.2.3.2	entropyRate(spAlg *T, real_1d_array *dist)	5
2.2.3.3	getMin(real_1d_array *v, int *arg)	5
2.2.3.4	getMin(integer_1d_array *v, int *arg)	5
2.2.3.5	getNNZ(spAlg *s)	5
2.2.3.6	igraph2HashTable(igraph_t *srcGraph)	5
2.2.3.7	pajek2HashTable(FILE *f)	6
2.2.3.8	setCol(real_2d_array *a, real_1d_array *col, int j)	6
2.2.3.9	setCol(integer_2d_array *a, integer_1d_array *col, int j)	6
2.2.3.10	setConstant(real_1d_array *v, double constant)	6
2.2.3.11	setConstant(integer_1d_array *v, int constant)	6
2.2.3.12	setLinSpaced(real_1d_array *v, double low, double high)	6
2.2.3.13	setLinSpaced(integer_1d_array *v, int low, int high)	6
2.2.3.14	setRow(real_2d_array *a, real_1d_array *row, int i)	6
2.2.3.15	setRow(integer_2d_array *a, integer_1d_array *row, int i)	6
2.3	/Users/atanet/PhD/dev/ATSuite/cpp/atio.hpp File Reference	6
2.3.1	Typedef Documentation	7
2.3.1.1	SpMatCSC	7
2.3.1.2	SpMatCSR	7
2.3.1.3	Tri	7
2.3.2	Function Documentation	7
2.3.2.1	Compressed2AR(FILE *)	7
2.3.2.2	Compressed2EdgeList(FILE *)	7

2.3.2.3	Compressed2EdgeList(FILE *, FILE *)	7
2.3.2.4	Compressed2Eigen(FILE *)	7
2.3.2.5	CSC2AR(FILE *)	7
2.3.2.6	CSC2CSR(SpMatCSC *)	7
2.3.2.7	CSR2CSC(SpMatCSR *)	7
2.3.2.8	Eigen2AR(SpMatCSC *)	8
2.3.2.9	Eigen2AR(SpMatCSR *)	8
2.3.2.10	Eigen2ARSym(SpMatCSC *)	8
2.3.2.11	Eigen2ARSym(SpMatCSR *)	8
2.3.2.12	Eigen2Compressed(FILE *, SpMatCSC *)	8
2.3.2.13	Eigen2Compressed(FILE *, SpMatCSR *)	8
2.3.2.14	Eigen2Pajek(FILE *, SpMatCSR *)	8
2.3.2.15	Eigen2Triplet(SpMatCSC *)	8
2.3.2.16	Eigen2Triplet(SpMatCSR *)	8
2.3.2.17	fprintfEigen(FILE *, SpMatCSR *)	8
2.3.2.18	fprintfEigen(FILE *fp, SpMatCSR *T, const char *format)	8
2.3.2.19	lineCount(FILE *)	8
2.3.2.20	pajek2AR(FILE *)	8
2.3.2.21	pajek2Eigen(FILE *)	8
2.4	/Users/atanet/PhD/dev/ATSuite/cpp/atmath.hpp File Reference	8
2.4.1	Macro Definition Documentation	9
2.4.1.1	plog2p	9
2.4.2	Typedef Documentation	9
2.4.2.1	SpMatCSC	9
2.4.2.2	SpMatCSCBool	9
2.4.2.3	SpMatCSR	9
2.4.2.4	SpMatIntCSR	9
2.4.3	Function Documentation	10
2.4.3.1	any(SpMatCSCBool *)	10
2.4.3.2	argmax(SpMatCSC *)	10

2.4.3.3	<a href="#">condition4Entropy(SpMatCSC *)</a>	10
2.4.3.4	<a href="#">cwiseGT(SpMatCSC *, double)</a>	10
2.4.3.5	<a href="#">cwiseLT(SpMatCSC *, double)</a>	10
2.4.3.6	<a href="#">entropy(VectorXd *)</a>	10
2.4.3.7	<a href="#">entropyRate(SpMatCSC *, VectorXd *)</a>	10
2.4.3.8	<a href="#">entropyRate(MatrixXd *, VectorXd *)</a>	10
2.4.3.9	<a href="#">getColSum(SpMatCSR *)</a>	10
2.4.3.10	<a href="#">getColSum(SpMatCSR *, gsl_vector *)</a>	10
2.4.3.11	<a href="#">getColSum(SpMatCSC *)</a>	10
2.4.3.12	<a href="#">getRowSum(SpMatCSR *)</a>	10
2.4.3.13	<a href="#">getRowSum(SpMatCSR *, gsl_vector *)</a>	10
2.4.3.14	<a href="#">getRowSum(SpMatIntCSR *)</a>	10
2.4.3.15	<a href="#">getSum(SpMatCSR *)</a>	10
2.4.3.16	<a href="#">lowlevelTransition(SpMatCSC *, VectorXd *, VectorXi *, MatrixXd *, VectorXd *)</a>	10
2.4.3.17	<a href="#">max(SpMatCSC *)</a>	10
2.4.3.18	<a href="#">min(SpMatCSC *)</a>	10
2.4.3.19	<a href="#">normalizeRows(SpMatCSR *, gsl_vector *)</a>	10
2.4.3.20	<a href="#">normalizeVector(gsl_vector *)</a>	10
2.4.3.21	<a href="#">sumVectorElements(gsl_vector *)</a>	10
2.4.3.22	<a href="#">toAndStochastic(SpMatCSR *)</a>	10
2.4.3.23	<a href="#">toAndStochastic(SpMatCSC *)</a>	11
2.4.3.24	<a href="#">toLeftStochastic(SpMatCSR *)</a>	11
2.4.3.25	<a href="#">toRightStochastic(SpMatCSC *)</a>	11
2.4.3.26	<a href="#">toRightStochastic(SpMatCSR *)</a>	11
2.5	<a href="#">/Users/atanet/PhD/dev/ATSuite/cpp/atspectrum.hpp File Reference</a>	11
2.5.1	<a href="#">Function Documentation</a>	11
2.5.1.1	<a href="#">getEigValNonSym(ARluNonSymMatrix&lt; double, double &gt; *, double *, double *, int, const char *, double, int, int, bool, double *)</a>	11
2.6	<a href="#">/Users/atanet/PhD/dev/ATSuite/cpp/ODESolvers.hpp File Reference</a>	11
2.6.1	<a href="#">Function Documentation</a>	12
2.6.1.1	<a href="#">cuspEuler(gsl_vector *, double, double, double)</a>	12

2.6.1.2	<a href="#">cuspField(gsl_vector *, double, double)</a>	12
2.6.1.3	<a href="#">generateCuspEuler(gsl_vector *, double, double, double, double, int, double)</a>	12
2.6.1.4	<a href="#">generateLorenzRK4(gsl_vector *, double, double, double, double, double, int, double)</a>	12
2.6.1.5	<a href="#">lorenzField(gsl_vector *, double, double, double)</a>	12
2.6.1.6	<a href="#">lorenzRK4(gsl_vector *, double, double, double, double)</a>	12
2.7	<a href="#">/Users/atanet/PhD/dev/ATSuite/cpp/SDESolvers.hpp File Reference</a>	12
2.7.1	<a href="#">Function Documentation</a>	12
2.7.1.1	<a href="#">additiveWienerField(double, gsl_vector *)</a>	12
2.7.1.2	<a href="#">cuspAdditiveWienerEM(gsl_vector *, double, double, gsl_vector *, double, double)</a>	12
2.7.1.3	<a href="#">generateCuspAdditiveWienerEM(gsl_vector *, double, double, gsl_matrix *, double, double, double, int, double)</a>	12
2.7.1.4	<a href="#">generateCuspAdditiveWienerEM(gsl_vector *, double, gsl_vector *, gsl_matrix *, double, double, double, int, double)</a>	13
2.7.1.5	<a href="#">generateLorenzLinearWienerEM(gsl_vector *, double, double, double, gsl_matrix *, double, double, double, int, double)</a>	13
2.7.1.6	<a href="#">linearWienerField(gsl_vector *, double, gsl_vector *)</a>	13
2.7.1.7	<a href="#">lorenzLinearWienerEM(gsl_vector *, double, double, double, gsl_vector *, double, double)</a>	13
2.8	<a href="#">/Users/atanet/PhD/dev/ATSuite/cpp/transferOperator.hpp File Reference</a>	13
2.8.1	<a href="#">Typedef Documentation</a>	14
2.8.1.1	<a href="#">SpMatCSC</a>	14
2.8.1.2	<a href="#">SpMatCSR</a>	14
2.8.1.3	<a href="#">triplet</a>	14
2.8.1.4	<a href="#">tripletUInt</a>	14
2.8.1.5	<a href="#">tripletUIntVector</a>	14
2.8.1.6	<a href="#">tripletVector</a>	14
2.8.2	<a href="#">Function Documentation</a>	14
2.8.2.1	<a href="#">filterTransitionMatrix(SpMatCSR *, gsl_vector *, gsl_vector *, double)</a>	14
2.8.2.2	<a href="#">filterTransitionMatrix(SpMatCSR *T, gsl_vector *rowCut, gsl_vector *colCut, double alpha, int norm)</a>	14
2.8.2.3	<a href="#">getBoxMembership(gsl_vector *, const std::vector&lt; gsl_vector * &gt; *)</a>	14

2.8.2.4	<code>getGridMembership(const gsl_matrix *, const gsl_matrix *, const std::vector&lt; gsl_vector * &gt; *)</code> . . . . .	14
2.8.2.5	<code>getGridMembership(const gsl_matrix *, const std::vector&lt; gsl_vector * &gt; *, const size_t)</code> . . . . .	14
2.8.2.6	<code>getGridMembership(const gsl_matrix *, const std::vector&lt; gsl_vector * &gt; *)</code> . . . . .	14
2.8.2.7	<code>getGridMembership(gsl_vector_uint *, const size_t)</code> . . . . .	14
2.8.2.8	<code>getGridRect(size_t, size_t, double, double)</code> . . . . .	14
2.8.2.9	<code>getGridRect(gsl_vector_uint *, gsl_vector *, gsl_vector *)</code> . . . . .	14
2.8.2.10	<code>getTransitionMatrix(const gsl_matrix_uint *, const size_t, SpMatCSR *, SpMatCSR *, gsl_vector *, gsl_vector *)</code> . . . . .	14
2.8.2.11	<code>getTransitionMatrix(const gsl_matrix *, const gsl_matrix *, const std::vector&lt; gsl_vector * &gt; *, SpMatCSR *, SpMatCSR *, gsl_vector *, gsl_vector *)</code> . . . . .	14
2.8.2.12	<code>getTransitionMatrix(const gsl_matrix *, const std::vector&lt; gsl_vector * &gt; *, const size_t tauStep, SpMatCSR *, SpMatCSR *, gsl_vector *, gsl_vector *)</code> . . . . .	14
2.8.2.13	<code>writeGridRect(FILE *, std::vector&lt; gsl_vector * &gt; *, bool)</code> . . . . .	14
2.9	<code>/Users/atantet/PhD/dev/ATSuite/cpp/transferOperatorTest.hpp</code> File Reference . . . . .	15
2.9.1	Typedef Documentation . . . . .	15
2.9.1.1	<code>SpMatCSC</code> . . . . .	15
2.9.1.2	<code>SpMatCSR</code> . . . . .	15
2.9.1.3	<code>triplet</code> . . . . .	15
2.9.1.4	<code>tripletVector</code> . . . . .	15
2.9.2	Function Documentation . . . . .	15
2.9.2.1	<code>getShuffledCountMatrix(SpMatCSR *, gsl_vector *, gsl_rng *)</code> . . . . .	15
2.9.2.2	<code>getShuffledRow(SpMatCSR *, size_t, size_t, tripletVector *, gsl_rng *)</code> . . . . .	15
2.9.2.3	<code>getSurrogateSpectrum(gsl_vector_int *, size_t, size_t, gsl_matrix *, gsl_matrix *, const char *, double, int, int, bool, double *)</code> . . . . .	15
2.9.2.4	<code>getSurrogateSpectrumFromCount(SpMatCSR *, gsl_rng *, gsl_matrix *, gsl_matrix *, const char *, double, int, int, bool, double *)</code> . . . . .	15
	<b>Index</b> . . . . .	<b>17</b>



# Chapter 1

## File Index

### 1.1 File List

Here is a list of all files with brief descriptions:

/Users/atanet/PhD/dev/ATSuite/cpp/atgraph.hpp . . . . .	3
/Users/atanet/PhD/dev/ATSuite/cpp/atgraph_alglib.hpp . . . . .	4
/Users/atanet/PhD/dev/ATSuite/cpp/atlo.hpp . . . . .	6
/Users/atanet/PhD/dev/ATSuite/cpp/atmath.hpp . . . . .	8
/Users/atanet/PhD/dev/ATSuite/cpp/atspectrum.hpp . . . . .	11
/Users/atanet/PhD/dev/ATSuite/cpp/ODESolvers.hpp . . . . .	11
/Users/atanet/PhD/dev/ATSuite/cpp/SDESolvers.hpp . . . . .	12
/Users/atanet/PhD/dev/ATSuite/cpp/transferOperator.hpp . . . . .	13
/Users/atanet/PhD/dev/ATSuite/cpp/transferOperatorTest.hpp . . . . .	15



## Chapter 2

# File Documentation

### 2.1 /Users/atantet/PhD/dev/ATSuite/cpp/atgraph.hpp File Reference

```
#include <cstdio>
#include <cstdlib>
#include <stack>
#include <vector>
#include <iostream>
#include <igraph/igraph.h>
#include <Eigen/Dense>
#include <Eigen/Sparse>
#include <atio.hpp>
```

#### Typedefs

- typedef Eigen::SparseMatrix< double > [SpMat](#)
- typedef Eigen::SparseMatrix< double, RowMajor > [SpMatCSR](#)
- typedef Eigen::Triplet< double > [Tri](#)

#### Functions

- int [pajek2igraph](#) (FILE \*f, igraph\_t \*dst\_graph)
- int [pajek2igraphNoVertices](#) (FILE \*f, igraph\_t \*dst\_graph, int vertex\_id0)
- int [pajek2igraphSym](#) (FILE \*f, igraph\_t \*dst\_graph)
- [SpMat](#) \* [pajek2EigenSparse](#) (FILE \*f)
- void [EigenSparse2Pajek](#) (FILE \*f, [SpMatCSR](#) \*P)
- [SpMat](#) \* [igraph2EigenSparse](#) (igraph\_t \*srcGraph)
- int [array2igraph](#) (FILE \*f, int N, igraph\_t \*dst\_graph)
- void [addCol2Col](#) ([SpMat](#) \*T, int j\_src, int j\_dst)
- void [addCol2ColTriplet](#) ([SpMat](#) \*T, int jSrc, int jDst)
- void [addRow2Row](#) ([SpMat](#) \*T, int i\_src, int i\_dst)
- void [addRow2RowTriplet](#) ([SpMat](#) \*T, int iSrc, int iDst)
- void [addRow2RowTriplet](#) ([SpMat](#) \*T, int iSrc, int iDst, [SpMatCSR](#) \*TCSR)
- void [scalProdInner](#) ([SpMat](#) \*T, int outer, double coef)
- void [scalProdOuter](#) ([SpMat](#) \*T, int inner, double coef)
- double [getModularity](#) ([SpMat](#) \*T)

## 2.1.1 Typedef Documentation

2.1.1.1 `typedef Eigen::SparseMatrix<double> SpMat`

2.1.1.2 `typedef Eigen::SparseMatrix<double, RowMajor> SpMatCSR`

2.1.1.3 `typedef Eigen::Triplet<double> Tri`

## 2.1.2 Function Documentation

2.1.2.1 `void addCol2Col ( SpMat * T, int j_src, int j_dst )`

2.1.2.2 `void addCol2ColTriplet ( SpMat * T, int jSrc, int jDst )`

2.1.2.3 `void addRow2Row ( SpMat * T, int i_src, int i_dst )`

2.1.2.4 `void addRow2RowTriplet ( SpMat * T, int iSrc, int iDst )`

2.1.2.5 `void addRow2RowTriplet ( SpMat * T, int iSrc, int iDst, SpMatCSR * TCSR )`

2.1.2.6 `int array2igraph ( FILE * f, int N, igraph_t * dst_graph )`

2.1.2.7 `void EigenSparse2Pajek ( FILE * f, SpMatCSR * P )`

2.1.2.8 `double getModularity ( SpMat * T )`

2.1.2.9 `SpMat* igraph2EigenSparse ( igraph_t * srcGraph )`

2.1.2.10 `SpMat* pajek2EigenSparse ( FILE * f )`

2.1.2.11 `int pajek2igraph ( FILE * f, igraph_t * dst_graph )`

2.1.2.12 `int pajek2igraphNoVertices ( FILE * f, igraph_t * dst_graph, int vertex_id0 )`

2.1.2.13 `int pajek2igraphSym ( FILE * f, igraph_t * dst_graph )`

2.1.2.14 `void scalProdInner ( SpMat * T, int outer, double coef )`

2.1.2.15 `void scalProdOuter ( SpMat * T, int inner, double coef )`

## 2.2 /Users/atantet/PhD/dev/ATSuite/cpp/atgraph\_alglib.hpp File Reference

```
#include <cstdio>
#include <cstdlib>
#include <stack>
#include <vector>
#include <iostream>
#include <igraph/igraph.h>
#include "linalg.h"
```

## Macros

- `#define plog2p(x) ( (x) > 0.0 ? (x) * log(x) / log(2) : 0.0 )`

## Typedefs

- typedef sparsematrix `spAlg`

## Functions

- `spAlg * pajek2HashTable (FILE *f)`
- `spAlg * igraph2HashTable (igraph_t *srcGraph)`
- `void setConstant (real_1d_array *v, double constant)`
- `void setConstant (integer_1d_array *v, int constant)`
- `void setLinSpaced (real_1d_array *v, double low, double high)`
- `void setLinSpaced (integer_1d_array *v, int low, int high)`
- `void setRow (real_2d_array *a, real_1d_array *row, int i)`
- `void setRow (integer_2d_array *a, integer_1d_array *row, int i)`
- `void setCol (real_2d_array *a, real_1d_array *col, int j)`
- `void setCol (integer_2d_array *a, integer_1d_array *col, int j)`
- `double getMin (real_1d_array *v, int *arg)`
- `int getMin (integer_1d_array *v, int *arg)`
- `int getNNZ (spAlg *s)`
- `double entropy (real_1d_array *dist)`
- `double entropyRate (spAlg *T, real_1d_array *dist)`

### 2.2.1 Macro Definition Documentation

2.2.1.1 `#define plog2p( x ) ( (x) > 0.0 ? (x) * log(x) / log(2) : 0.0 )`

### 2.2.2 Typedef Documentation

2.2.2.1 typedef sparsematrix `spAlg`

### 2.2.3 Function Documentation

2.2.3.1 `double entropy ( real_1d_array * dist )`

2.2.3.2 `double entropyRate ( spAlg * T, real_1d_array * dist )`

2.2.3.3 `double getMin ( real_1d_array * v, int * arg )`

2.2.3.4 `int getMin ( integer_1d_array * v, int * arg )`

2.2.3.5 `int getNNZ ( spAlg * s )`

2.2.3.6 `spAlg* igraph2HashTable ( igraph_t * srcGraph )`

2.2.3.7 `spAlg* pajek2HashTable ( FILE * f )`

2.2.3.8 `void setCol ( real_2d_array * a, real_1d_array * col, int j )`

2.2.3.9 `void setCol ( integer_2d_array * a, integer_1d_array * col, int j )`

2.2.3.10 `void setConstant ( real_1d_array * v, double constant )`

2.2.3.11 `void setConstant ( integer_1d_array * v, int constant )`

2.2.3.12 `void setLinSpaced ( real_1d_array * v, double low, double high )`

2.2.3.13 `void setLinSpaced ( integer_1d_array * v, int low, int high )`

2.2.3.14 `void setRow ( real_2d_array * a, real_1d_array * row, int i )`

2.2.3.15 `void setRow ( integer_2d_array * a, integer_1d_array * row, int i )`

## 2.3 /Users/atantet/PhD/dev/ATSuite/cpp/atio.hpp File Reference

```
#include <cstdlib>
#include <cstdio>
#include <vector>
#include <Eigen/Dense>
#include <Eigen/Sparse>
#include "arlnsmat.h"
#include "arlssym.h"
#include <gsl/gsl_vector.h>
#include <gsl/gsl_matrix.h>
```

### Typedefs

- `typedef Eigen::SparseMatrix< double, Eigen::ColMajor > SpMatCSC`
- `typedef Eigen::SparseMatrix< double, Eigen::RowMajor > SpMatCSR`
- `typedef Eigen::Triplet< double > Tri`

## Functions

- void [Eigen2Pajek](#) (FILE \*, [SpMatCSR](#) \*)
- void [Eigen2Compressed](#) (FILE \*, [SpMatCSC](#) \*)
- void [Eigen2Compressed](#) (FILE \*, [SpMatCSR](#) \*)
- [SpMatCSC](#) \* [pajek2Eigen](#) (FILE \*)
- [ARluNonSymMatrix](#)< double, double > \* [pajek2AR](#) (FILE \*)
- [ARluNonSymMatrix](#)< double, double > \* [CSC2AR](#) (FILE \*)
- [ARluNonSymMatrix](#)< double, double > \* [Eigen2AR](#) ([SpMatCSC](#) \*)
- [ARluNonSymMatrix](#)< double, double > \* [Eigen2AR](#) ([SpMatCSR](#) \*)
- [ARluSymMatrix](#)< double > \* [Eigen2ARSym](#) ([SpMatCSC](#) \*)
- [ARluSymMatrix](#)< double > \* [Eigen2ARSym](#) ([SpMatCSR](#) \*)
- [ARluNonSymMatrix](#)< double, double > \* [Compressed2AR](#) (FILE \*)
- [SpMatCSR](#) \* [Compressed2Eigen](#) (FILE \*)
- [gsl\\_matrix](#) \* [Compressed2EdgeList](#) (FILE \*)
- void [Compressed2EdgeList](#) (FILE \*, FILE \*)
- [SpMatCSR](#) \* [CSC2CSR](#) ([SpMatCSC](#) \*T)
- [SpMatCSC](#) \* [CSR2CSC](#) ([SpMatCSR](#) \*T)
- vector< [Tri](#) > [Eigen2Triplet](#) ([SpMatCSC](#) \*)
- vector< [Tri](#) > [Eigen2Triplet](#) ([SpMatCSR](#) \*)
- void [fprintfEigen](#) (FILE \*, [SpMatCSR](#) \*)
- size\_t [lineCount](#) (FILE \*)
- void [fprintfEigen](#) (FILE \*fp, [SpMatCSR](#) \*T, const char \*format)

### 2.3.1 Typedef Documentation

2.3.1.1 typedef [Eigen::SparseMatrix](#)<double, [Eigen::ColMajor](#)> [SpMatCSC](#)

2.3.1.2 typedef [Eigen::SparseMatrix](#)<double, [Eigen::RowMajor](#)> [SpMatCSR](#)

2.3.1.3 typedef [Eigen::Triplet](#)<double> [Tri](#)

### 2.3.2 Function Documentation

2.3.2.1 [ARluNonSymMatrix](#)< double, double > \* [Compressed2AR](#) ( FILE \* f )

2.3.2.2 [gsl\\_matrix](#) \* [Compressed2EdgeList](#) ( FILE \* f )

2.3.2.3 void [Compressed2EdgeList](#) ( FILE \* src, FILE \* dst )

2.3.2.4 [SpMatCSR](#) \* [Compressed2Eigen](#) ( FILE \* f )

2.3.2.5 [ARluNonSymMatrix](#)< double, double > \* [CSC2AR](#) ( FILE \* f )

2.3.2.6 [SpMatCSR](#) \* [CSC2CSR](#) ( [SpMatCSC](#) \* T )

2.3.2.7 [SpMatCSC](#) \* [CSR2CSC](#) ( [SpMatCSR](#) \* T )

2.3.2.8 `ARluNonSymMatrix< double, double > * Eigen2AR ( SpMatCSC * TEigen )`

2.3.2.9 `ARluNonSymMatrix< double, double > * Eigen2AR ( SpMatCSR * TEigenCSR )`

2.3.2.10 `ARluSymMatrix< double > * Eigen2ARSym ( SpMatCSC * TEigen )`

2.3.2.11 `ARluSymMatrix< double > * Eigen2ARSym ( SpMatCSR * TEigenCSR )`

2.3.2.12 `void Eigen2Compressed ( FILE * f, SpMatCSC * P )`

2.3.2.13 `void Eigen2Compressed ( FILE * f, SpMatCSR * P )`

2.3.2.14 `void Eigen2Pajek ( FILE * f, SpMatCSR * P )`

2.3.2.15 `vector< Tri > Eigen2Triplet ( SpMatCSC * T )`

2.3.2.16 `vector< Tri > Eigen2Triplet ( SpMatCSR * T )`

2.3.2.17 `void fprintfEigen ( FILE *, SpMatCSR * )`

2.3.2.18 `void fprintfEigen ( FILE * fp, SpMatCSR * T, const char * format )`

2.3.2.19 `size_t lineCount ( FILE * fp )`

2.3.2.20 `ARluNonSymMatrix< double, double > * pajek2AR ( FILE * f )`

2.3.2.21 `SpMatCSC * pajek2Eigen ( FILE * f )`

## 2.4 /Users/atantet/PhD/dev/ATSuite/cpp/atmath.hpp File Reference

```
#include <cstdio>
#include <cstdlib>
#include <vector>
#include <list>
#include <gsl/gsl_vector.h>
#include <gsl/gsl_vector_int.h>
#include <gsl/gsl_matrix.h>
#include <Eigen/Dense>
#include <Eigen/Sparse>
```

### Macros

- `#define plog2p(x) ( (x) > 0.0 ? (x) * log(x) / log(2) : 0.0 )`



## Typedefs

- typedef SparseMatrix< double, RowMajor > [SpMatCSR](#)
- typedef SparseMatrix< double, ColMajor > [SpMatCSC](#)
- typedef SparseMatrix< bool, ColMajor > [SpMatCSCBool](#)
- typedef SparseMatrix< int, RowMajor > [SpMatIntCSR](#)

## Functions

- double [entropy](#) (VectorXd \*)
- double [entropyRate](#) (SpMatCSC \*, VectorXd \*)
- double [entropyRate](#) (MatrixXd \*, VectorXd \*)
- void [toRightStochastic](#) (SpMatCSC \*)
- void [toRightStochastic](#) (SpMatCSR \*)
- void [toLeftStochastic](#) (SpMatCSR \*)
- void [toAndStochastic](#) (SpMatCSR \*)
- void [toAndStochastic](#) (SpMatCSC \*)
- gsl\_vector \* [getRowSum](#) (SpMatCSR \*)
- void [getRowSum](#) (SpMatCSR \*, gsl\_vector \*)
- gsl\_vector\_int \* [getRowSum](#) (SpMatIntCSR \*)
- gsl\_vector \* [getColSum](#) (SpMatCSR \*)
- void [getColSum](#) (SpMatCSR \*, gsl\_vector \*)
- gsl\_vector \* [getColSum](#) (SpMatCSC \*)
- double [getSum](#) (SpMatCSR \*)
- double [sumVectorElements](#) (gsl\_vector \*)
- void [normalizeVector](#) (gsl\_vector \*)
- void [normalizeRows](#) (SpMatCSR \*, gsl\_vector \*)
- void [condition4Entropy](#) (SpMatCSC \*)
- SpMatCSCBool \* [cwiseGT](#) (SpMatCSC \*, double)
- SpMatCSCBool \* [cwiseLT](#) (SpMatCSC \*, double)
- bool [any](#) (SpMatCSCBool \*)
- double [max](#) (SpMatCSC \*)
- double [min](#) (SpMatCSC \*)
- vector< int > \* [argmax](#) (SpMatCSC \*)
- void [lowlevelTransition](#) (SpMatCSC \*, VectorXd \*, VectorXi \*, MatrixXd \*, VectorXd \*)

### 2.4.1 Macro Definition Documentation

2.4.1.1 `#define plog2p( x ) ( (x) > 0.0 ? (x) * log(x) / log(2) : 0.0 )`

### 2.4.2 Typedef Documentation

2.4.2.1 `typedef SparseMatrix<double, ColMajor> SpMatCSC`

2.4.2.2 `typedef SparseMatrix<bool, ColMajor> SpMatCSCBool`

2.4.2.3 `typedef SparseMatrix<double, RowMajor> SpMatCSR`

2.4.2.4 `typedef SparseMatrix<int, RowMajor> SpMatIntCSR`

### 2.4.3 Function Documentation

2.4.3.1 `bool any ( SpMatCSCBool * T )`

2.4.3.2 `vector< int > * argmax ( SpMatCSC * T )`

2.4.3.3 `void condition4Entropy ( SpMatCSC * T )`

2.4.3.4 `SpMatCSCBool * cwiseGT ( SpMatCSC * T, double ref )`

2.4.3.5 `SpMatCSCBool * cwiseLT ( SpMatCSC * T, double ref )`

2.4.3.6 `double entropy ( VectorXd * dist )`

2.4.3.7 `double entropyRate ( SpMatCSC * T, VectorXd * dist )`

2.4.3.8 `double entropyRate ( MatrixXd * T, VectorXd * dist )`

2.4.3.9 `gsl_vector * getColSum ( SpMatCSR * T )`

2.4.3.10 `void getColSum ( SpMatCSR * T, gsl_vector * colSum )`

2.4.3.11 `gsl_vector * getColSum ( SpMatCSC * T )`

2.4.3.12 `gsl_vector * getRowSum ( SpMatCSR * T )`

2.4.3.13 `void getRowSum ( SpMatCSR * T, gsl_vector * rowSum )`

2.4.3.14 `gsl_vector_int * getRowSum ( SpMatIntCSR * T )`

2.4.3.15 `double getSum ( SpMatCSR * T )`

2.4.3.16 `void lowlevelTransition ( SpMatCSC * highT, VectorXd * highDist, VectorXi * member, MatrixXd * lowT, VectorXd * lowDist )`

2.4.3.17 `double max ( SpMatCSC * T )`

2.4.3.18 `double min ( SpMatCSC * T )`

2.4.3.19 `void normalizeRows ( SpMatCSR * T, gsl_vector * rowSum )`

2.4.3.20 `void normalizeVector ( gsl_vector * v )`

2.4.3.21 `double sumVectorElements ( gsl_vector * v )`

2.4.3.22 `void toAndStochastic ( SpMatCSR * T )`

2.4.3.23 void toAndStochastic ( SpMatCSC \* *T* )

2.4.3.24 void toLeftStochastic ( SpMatCSR \* *T* )

2.4.3.25 void toRightStochastic ( SpMatCSC \* *T* )

2.4.3.26 void toRightStochastic ( SpMatCSR \* *T* )

## 2.5 /Users/atantet/PhD/dev/ATSuite/cpp/atspectrum.hpp File Reference

```
#include "arlnsmat.h"
#include "arlsnsym.h"
```

### Functions

- void [getEigValNonSym](#) (ARluNonSymMatrix< double, double > \* *P*, double \* *EigValReal*, double \* *EigValImag*, int *nev*, const char \* *which*, double *tol* = 0., int *maxit* = 0, int *ncv* = 0, bool *AutoShift* = true, double \* *resid* = NULL )

### 2.5.1 Function Documentation

2.5.1.1 void [getEigValNonSym](#) ( ARluNonSymMatrix< double, double > \* *P*, double \* *EigValReal*, double \* *EigValImag*, int *nev*, const char \* *which*, double *tol* = 0., int *maxit* = 0, int *ncv* = 0, bool *AutoShift* = true, double \* *resid* = NULL )

## 2.6 /Users/atantet/PhD/dev/ATSuite/cpp/ODESolvers.hpp File Reference

```
#include <gsl/gsl_vector.h>
#include <gsl/gsl_matrix.h>
```

### Functions

- gsl\_matrix \* [generateCuspEuler](#) (gsl\_vector \*, double, double, double, double, int, double)
- gsl\_vector \* [cuspEuler](#) (gsl\_vector \*, double, double, double)
- gsl\_vector \* [cuspField](#) (gsl\_vector \*, double, double)
- gsl\_matrix \* [generateLorenzRK4](#) (gsl\_vector \*, double, double, double, double, double, int, double)
- gsl\_vector \* [lorenzRK4](#) (gsl\_vector \*, double, double, double, double)
- gsl\_vector \* [lorenzField](#) (gsl\_vector \*, double, double, double)

## 2.6.1 Function Documentation

2.6.1.1 `gsl_vector * cuspEuler ( gsl_vector * state, double r, double h, double dt )`

2.6.1.2 `gsl_vector * cuspField ( gsl_vector * state, double r, double h )`

2.6.1.3 `gsl_matrix * generateCuspEuler ( gsl_vector * state, double r, double h, double length, double dt, int sampling, double spinup )`

2.6.1.4 `gsl_matrix * generateLorenzRK4 ( gsl_vector * state, double rho, double sigma, double beta, double length, double dt, int sampling, double spinup )`

2.6.1.5 `gsl_vector * lorenzField ( gsl_vector * state, double rho, double sigma, double beta )`

2.6.1.6 `gsl_vector * lorenzRK4 ( gsl_vector * state, double rho, double sigma, double beta, double dt )`

## 2.7 /Users/atantet/PhD/dev/ATSuite/cpp/SDESolvers.hpp File Reference

```
#include <cstdio>
#include <cstdlib>
#include <cmath>
#include <ODESolvers.hpp>
```

### Functions

- `gsl_matrix * generateCuspAdditiveWienerEM (gsl_vector *, double, double, gsl_matrix *, double, double, double, int, double)`
- `gsl_matrix * generateCuspAdditiveWienerEM (gsl_vector *, double, gsl_vector *, gsl_matrix *, double, double, double, int, double)`
- `gsl_vector * cuspAdditiveWienerEM (gsl_vector *, double, double, gsl_vector *, double, double)`
- `gsl_matrix * generateLorenzLinearWienerEM (gsl_vector *, double, double, double, gsl_matrix *, double, double, double, int, double)`
- `gsl_vector * lorenzLinearWienerEM (gsl_vector *, double, double, double, gsl_vector *, double, double)`
- `gsl_vector * additiveWienerField (double, gsl_vector *)`
- `gsl_vector * linearWienerField (gsl_vector *, double, gsl_vector *)`

## 2.7.1 Function Documentation

2.7.1.1 `gsl_vector * additiveWienerField ( double Q, gsl_vector * noiseSample )`

2.7.1.2 `gsl_vector * cuspAdditiveWienerEM ( gsl_vector * state, double r, double h, gsl_vector * noiseSample, double Q, double dt )`

2.7.1.3 `gsl_matrix * generateCuspAdditiveWienerEM ( gsl_vector * state, double r, double h, gsl_matrix * noiseSamples, double Q, double length, double dt, int sampling, double spinup )`

- 2.7.1.4 `gsl_matrix * generateCuspAdditiveWienerEM ( gsl_vector * state, double r, gsl_vector * hTransient, gsl_matrix * noiseSamples, double Q, double length, double dt, int sampling, double spinup )`
- 2.7.1.5 `gsl_matrix * generateLorenzLinearWienerEM ( gsl_vector * state, double rho, double sigma, double beta, gsl_matrix * noiseSamples, double Q, double length, double dt, int sampling, double spinup )`
- 2.7.1.6 `gsl_vector * linearWienerField ( gsl_vector * state, double Q, gsl_vector * noiseSample )`
- 2.7.1.7 `gsl_vector * lorenzLinearWienerEM ( gsl_vector * state, double rho, double sigma, double beta, gsl_vector * noiseSample, double Q, double dt )`

## 2.8 /Users/atatet/PhD/dev/ATSuite/cpp/transferOperator.hpp File Reference

```
#include <iostream>
#include <vector>
#include <gsl/gsl_vector.h>
#include <gsl/gsl_vector_uint.h>
#include <gsl/gsl_matrix.h>
#include <gsl/gsl_matrix_uint.h>
#include <Eigen/Dense>
#include <Eigen/Sparse>
#include <omp.h>
#include "atmath.hpp"
```

### Typedefs

- typedef Eigen::Triplet< double > [triplet](#)
- typedef std::vector< [triplet](#) > [tripletVector](#)
- typedef Eigen::Triplet< size\_t > [tripletUInt](#)
- typedef std::vector< [tripletUInt](#) > [tripletUIntVector](#)
- typedef Eigen::SparseMatrix< double, Eigen::ColMajor > [SpMatCSC](#)
- typedef Eigen::SparseMatrix< double, Eigen::RowMajor > [SpMatCSR](#)

### Functions

- void [getTransitionMatrix](#) (const gsl\_matrix\_uint \*, const size\_t, [SpMatCSR](#) \*, [SpMatCSR](#) \*, gsl\_vector \*, gsl\_vector \*)
- void [getTransitionMatrix](#) (const gsl\_matrix \*, const gsl\_matrix \*, const std::vector< gsl\_vector \* > \*, [SpMatCSR](#) \*, [SpMatCSR](#) \*, gsl\_vector \*, gsl\_vector \*)
- void [getTransitionMatrix](#) (const gsl\_matrix \*, const std::vector< gsl\_vector \* > \*, const size\_t tauStep, [SpMatCSR](#) \*, [SpMatCSR](#) \*, gsl\_vector \*, gsl\_vector \*)
- gsl\_matrix\_uint \* [getGridMembership](#) (const gsl\_matrix \*, const gsl\_matrix \*, const std::vector< gsl\_vector \* > \*)
- gsl\_matrix\_uint \* [getGridMembership](#) (const gsl\_matrix \*, const std::vector< gsl\_vector \* > \*, const size\_t)
- gsl\_vector\_uint \* [getGridMembership](#) (const gsl\_matrix \*, const std::vector< gsl\_vector \* > \*)
- gsl\_matrix\_uint \* [getGridMembership](#) (gsl\_vector\_uint \*, const size\_t)
- int [getBoxMembership](#) (gsl\_vector \*, const std::vector< gsl\_vector \* > \*)
- void [filterTransitionMatrix](#) ([SpMatCSR](#) \*, gsl\_vector \*, gsl\_vector \*, double)
- std::vector< gsl\_vector \* > \* [getGridRect](#) (size\_t, size\_t, double, double)
- std::vector< gsl\_vector \* > \* [getGridRect](#) (gsl\_vector\_uint \*, gsl\_vector \*, gsl\_vector \*)
- void [writeGridRect](#) (FILE \*, std::vector< gsl\_vector \* > \*, bool)
- void [filterTransitionMatrix](#) ([SpMatCSR](#) \*T, gsl\_vector \*rowCut, gsl\_vector \*colCut, double alpha, int norm)

## 2.8.1 Typedef Documentation

2.8.1.1 `typedef Eigen::SparseMatrix<double, Eigen::ColMajor> SpMatCSC`

2.8.1.2 `typedef Eigen::SparseMatrix<double, Eigen::RowMajor> SpMatCSR`

2.8.1.3 `typedef Eigen::Triplet<double> triplet`

2.8.1.4 `typedef Eigen::Triplet<size_t> tripletUInt`

2.8.1.5 `typedef std::vector<tripletUInt> tripletUIntVector`

2.8.1.6 `typedef std::vector<triplet> tripletVector`

## 2.8.2 Function Documentation

2.8.2.1 `void filterTransitionMatrix ( SpMatCSR * , gsl_vector * , gsl_vector * , double )`

2.8.2.2 `void filterTransitionMatrix ( SpMatCSR * T, gsl_vector * rowCut, gsl_vector * colCut, double alpha, int norm )`

2.8.2.3 `int getBoxMembership ( gsl_vector * X, const std::vector< gsl_vector * > * gridBounds )`

2.8.2.4 `gsl_matrix_uint * getGridMembership ( const gsl_matrix * initState, const gsl_matrix * finalStates, const std::vector< gsl_vector * > * gridBounds )`

2.8.2.5 `gsl_matrix_uint * getGridMembership ( const gsl_matrix * states, const std::vector< gsl_vector * > * gridBounds, const size_t tauStep )`

2.8.2.6 `gsl_vector_uint * getGridMembership ( const gsl_matrix * states, const std::vector< gsl_vector * > * gridBounds )`

2.8.2.7 `gsl_matrix_uint * getGridMembership ( gsl_vector_uint * gridMemVect, const size_t tauStep )`

2.8.2.8 `std::vector< gsl_vector * > * getGridRect ( size_t dim, size_t nx, double xmin, double xmax )`

2.8.2.9 `std::vector< gsl_vector * > * getGridRect ( gsl_vector_uint * nx, gsl_vector * xmin, gsl_vector * xmax )`

2.8.2.10 `void getTransitionMatrix ( const gsl_matrix_uint * gridMem, const size_t N, SpMatCSR * P, SpMatCSR * Q, gsl_vector * initDist, gsl_vector * finalDist )`

2.8.2.11 `void getTransitionMatrix ( const gsl_matrix * initState, const gsl_matrix * finalStates, const std::vector< gsl_vector * > * gridBounds, SpMatCSR * P, SpMatCSR * Q, gsl_vector * initDist, gsl_vector * finalDist )`

2.8.2.12 `void getTransitionMatrix ( const gsl_matrix * states, const std::vector< gsl_vector * > * gridBounds, const size_t tauStep, SpMatCSR * P, SpMatCSR * Q, gsl_vector * initDist, gsl_vector * finalDist )`

2.8.2.13 `void writeGridRect ( FILE * gridFile, std::vector< gsl_vector * > * gridBounds, bool verbose = false )`

## 2.9 /Users/atanet/PhD/dev/ATSuite/cpp/transferOperatorTest.hpp File Reference

```
#include <vector>
#include <cmath>
#include <gsl/gsl_vector.h>
#include <gsl/gsl_vector_int.h>
#include <gsl/gsl_matrix.h>
#include <gsl/gsl_rng.h>
#include <gsl/gsl_randist.h>
#include <Eigen/Dense>
#include <Eigen/Sparse>
#include "atmath.hpp"
#include "atio.hpp"
#include "atspectrum.hpp"
#include "transferOperator.hpp"
```

### Typedefs

- typedef Eigen::Triplet< double > [triplet](#)
- typedef std::vector< [triplet](#) > [tripletVector](#)
- typedef Eigen::SparseMatrix< double, Eigen::ColMajor > [SpMatCSC](#)
- typedef Eigen::SparseMatrix< double, Eigen::RowMajor > [SpMatCSR](#)

### Functions

- void [getSurrogateSpectrum](#) (gsl\_vector\_int \*, size\_t, size\_t, gsl\_matrix \*, gsl\_matrix \*, const char \*, double, int, int, bool, double \*)
- void [getSurrogateSpectrumFromCount](#) (SpMatCSR \*, gsl\_rng \*, gsl\_matrix \*, gsl\_matrix \*, const char \*, double, int, int, bool, double \*)
- SpMatCSR \* [getShuffledCountMatrix](#) (SpMatCSR \*, gsl\_vector \*, gsl\_rng \*)
- void [getShuffledRow](#) (SpMatCSR \*, size\_t, size\_t, tripletVector \*, gsl\_rng \*)

### 2.9.1 Typedef Documentation

2.9.1.1 typedef Eigen::SparseMatrix<double, Eigen::ColMajor> [SpMatCSC](#)

2.9.1.2 typedef Eigen::SparseMatrix<double, Eigen::RowMajor> [SpMatCSR](#)

2.9.1.3 typedef Eigen::Triplet<double> [triplet](#)

2.9.1.4 typedef std::vector<triplet> [tripletVector](#)

### 2.9.2 Function Documentation

2.9.2.1 SpMatCSR \* [getShuffledCountMatrix](#) ( SpMatCSR \* *C*, gsl\_vector \* *nTransPerRow*, gsl\_rng \* *r* )

2.9.2.2 void [getShuffledRow](#) ( SpMatCSR \* *C*, size\_t *iRow*, size\_t *nTrans*, tripletVector \* *Ts*, gsl\_rng \* *r* )

2.9.2.3 void [getSurrogateSpectrum](#) ( gsl\_vector\_int \* *gridMem*, size\_t *N*, size\_t *tauStep*, gsl\_matrix \* *EigValRealDist*, gsl\_matrix \* *EigValImagDist*, const char \* *which* = "LM", double *tol* = 0., int *maxit* = 0, int *ncv* = 0, bool *AutoShift* = true, double \* *resid* = NULL )

2.9.2.4 void [getSurrogateSpectrumFromCount](#) ( SpMatCSR \* *C*, gsl\_rng \* *r*, gsl\_matrix \* *EigValRealDist*, gsl\_matrix \* *EigValImagDist*, const char \* *which* = "LM", double *tol* = 0., int *maxit* = 0, int *ncv* = 0, bool *AutoShift* = true, double \* *resid* = NULL )





# Index

/Users/atanet/PhD/dev/ATSuite/cpp/ODESolvers.hpp, 11  
/Users/atanet/PhD/dev/ATSuite/cpp/SDESolvers.hpp, 12  
/Users/atanet/PhD/dev/ATSuite/cpp/atgraph.hpp, 3  
/Users/atanet/PhD/dev/ATSuite/cpp/atgraph\_alglib.hpp, 4  
/Users/atanet/PhD/dev/ATSuite/cpp/atio.hpp, 6  
/Users/atanet/PhD/dev/ATSuite/cpp/atmath.hpp, 8  
/Users/atanet/PhD/dev/ATSuite/cpp/atspectrum.hpp, 11  
/Users/atanet/PhD/dev/ATSuite/cpp/transferOperator.hpp, 13  
/Users/atanet/PhD/dev/ATSuite/cpp/transferOperatorTest.hpp, 15

addCol2Col  
  atgraph.hpp, 4  
addCol2ColTriplet  
  atgraph.hpp, 4  
addRow2Row  
  atgraph.hpp, 4  
addRow2RowTriplet  
  atgraph.hpp, 4  
additiveWienerField  
  SDESolvers.hpp, 12  
any  
  atmath.hpp, 10  
argmax  
  atmath.hpp, 10  
array2igraph  
  atgraph.hpp, 4  
atgraph.hpp  
  addCol2Col, 4  
  addCol2ColTriplet, 4  
  addRow2Row, 4  
  addRow2RowTriplet, 4  
  array2igraph, 4  
  EigenSparse2Pajek, 4  
  getModularity, 4  
  igraph2EigenSparse, 4  
  pajek2EigenSparse, 4  
  pajek2igraph, 4  
  pajek2igraphNoVertices, 4  
  pajek2igraphSym, 4  
  scalProdInner, 4  
  scalProdOuter, 4  
  SpMat, 4  
  SpMatCSR, 4  
  Tri, 4  
  atgraph\_alglib.hpp  
    entropy, 5  
    entropyRate, 5  
    getMin, 5  
    getNNZ, 5  
    igraph2HashTable, 5  
    pajek2HashTable, 5  
    plog2p, 5  
    setCol, 6  
    setConstant, 6  
    setLinSpaced, 6  
    setRow, 6  
    spAlg, 5  
  atio.hpp  
    CSC2AR, 7  
    CSC2CSR, 7  
    CSR2CSC, 7  
    Compressed2AR, 7  
    Compressed2EdgeList, 7  
    Compressed2Eigen, 7  
    Eigen2ARSym, 8  
    Eigen2AR, 7, 8  
    Eigen2Compressed, 8  
    Eigen2Pajek, 8  
    Eigen2Triplet, 8  
    fprintfEigen, 8  
    lineCount, 8  
    pajek2AR, 8  
    pajek2Eigen, 8  
    SpMatCSC, 7  
    SpMatCSR, 7  
    Tri, 7  
  atmath.hpp  
    any, 10  
    argmax, 10  
    condition4Entropy, 10  
    cwiseGT, 10  
    cwiseLT, 10  
    entropy, 10  
    entropyRate, 10  
    getColSum, 10  
    getRowSum, 10  
    getSum, 10  
    lowlevelTransition, 10  
    max, 10  
    min, 10  
    normalizeRows, 10  
    normalizeVector, 10  
    plog2p, 9

- SpMatCSCBool, 9
- SpMatCSC, 9
- SpMatCSR, 9
- SpMatIntCSR, 9
- sumVectorElements, 10
- toAndStochastic, 10
- toLeftStochastic, 11
- toRightStochastic, 11
- atspectrum.hpp
  - getEigValNonSym, 11
- CSC2AR
  - atio.hpp, 7
- CSC2CSR
  - atio.hpp, 7
- CSR2CSC
  - atio.hpp, 7
- Compressed2AR
  - atio.hpp, 7
- Compressed2EdgeList
  - atio.hpp, 7
- Compressed2Eigen
  - atio.hpp, 7
- condition4Entropy
  - atmath.hpp, 10
- cuspidAdditiveWienerEM
  - SDESolvers.hpp, 12
- cuspidEuler
  - ODESolvers.hpp, 12
- cuspidField
  - ODESolvers.hpp, 12
- cwiseGT
  - atmath.hpp, 10
- cwiseLT
  - atmath.hpp, 10
- Eigen2ARSym
  - atio.hpp, 8
- Eigen2AR
  - atio.hpp, 7, 8
- Eigen2Compressed
  - atio.hpp, 8
- Eigen2Pajek
  - atio.hpp, 8
- Eigen2Triplet
  - atio.hpp, 8
- EigenSparse2Pajek
  - atgraph.hpp, 4
- entropy
  - atgraph\_alglib.hpp, 5
  - atmath.hpp, 10
- entropyRate
  - atgraph\_alglib.hpp, 5
  - atmath.hpp, 10
- filterTransitionMatrix
  - transferOperator.hpp, 14
- fprintfEigen
  - atio.hpp, 8
- generateCuspidAdditiveWienerEM
  - SDESolvers.hpp, 12
- generateCuspidEuler
  - ODESolvers.hpp, 12
- generateLorenzLinearWienerEM
  - SDESolvers.hpp, 13
- generateLorenzRK4
  - ODESolvers.hpp, 12
- getBoxMembership
  - transferOperator.hpp, 14
- getColSum
  - atmath.hpp, 10
- getEigValNonSym
  - atspectrum.hpp, 11
- getGridMembership
  - transferOperator.hpp, 14
- getGridRect
  - transferOperator.hpp, 14
- getMin
  - atgraph\_alglib.hpp, 5
- getModularity
  - atgraph.hpp, 4
- getNNZ
  - atgraph\_alglib.hpp, 5
- getRowSum
  - atmath.hpp, 10
- getShuffledCountMatrix
  - transferOperatorTest.hpp, 15
- getShuffledRow
  - transferOperatorTest.hpp, 15
- getSum
  - atmath.hpp, 10
- getSurrogateSpectrum
  - transferOperatorTest.hpp, 15
- getSurrogateSpectrumFromCount
  - transferOperatorTest.hpp, 15
- getTransitionMatrix
  - transferOperator.hpp, 14
- igraph2EigenSparse
  - atgraph.hpp, 4
- igraph2HashTable
  - atgraph\_alglib.hpp, 5
- lineCount
  - atio.hpp, 8
- linearWienerField
  - SDESolvers.hpp, 13
- lorenzField
  - ODESolvers.hpp, 12
- lorenzLinearWienerEM
  - SDESolvers.hpp, 13
- lorenzRK4
  - ODESolvers.hpp, 12
- lowlevelTransition
  - atmath.hpp, 10
- max
  - atmath.hpp, 10

- min
  - atmath.hpp, 10
- normalizeRows
  - atmath.hpp, 10
- normalizeVector
  - atmath.hpp, 10
- ODESolvers.hpp
  - cuspeuler, 12
  - cuspeulerField, 12
  - generateCuspeuler, 12
  - generateLorenzRK4, 12
  - lorenzField, 12
  - lorenzRK4, 12
- pajek2AR
  - atio.hpp, 8
- pajek2Eigen
  - atio.hpp, 8
- pajek2EigenSparse
  - atgraph.hpp, 4
- pajek2HashTable
  - atgraph\_alglib.hpp, 5
- pajek2igraph
  - atgraph.hpp, 4
- pajek2igraphNoVertices
  - atgraph.hpp, 4
- pajek2igraphSym
  - atgraph.hpp, 4
- plog2p
  - atgraph\_alglib.hpp, 5
  - atmath.hpp, 9
- SDESolvers.hpp
  - additiveWienerField, 12
  - cuspeulerAdditiveWienerEM, 12
  - generateCuspeulerAdditiveWienerEM, 12
  - generateLorenzLinearWienerEM, 13
  - linearWienerField, 13
  - lorenzLinearWienerEM, 13
- scalProdInner
  - atgraph.hpp, 4
- scalProdOuter
  - atgraph.hpp, 4
- setCol
  - atgraph\_alglib.hpp, 6
- setConstant
  - atgraph\_alglib.hpp, 6
- setLinSpaced
  - atgraph\_alglib.hpp, 6
- setRow
  - atgraph\_alglib.hpp, 6
- spAlg
  - atgraph\_alglib.hpp, 5
- SpMat
  - atgraph.hpp, 4
- SpMatCSCBool
  - atmath.hpp, 9
- SpMatCSC
  - atio.hpp, 7
  - atmath.hpp, 9
  - transferOperator.hpp, 14
  - transferOperatorTest.hpp, 15
- SpMatCSR
  - atgraph.hpp, 4
  - atio.hpp, 7
  - atmath.hpp, 9
  - transferOperator.hpp, 14
  - transferOperatorTest.hpp, 15
- SpMatIntCSR
  - atmath.hpp, 9
- sumVectorElements
  - atmath.hpp, 10
- toAndStochastic
  - atmath.hpp, 10
- toLeftStochastic
  - atmath.hpp, 11
- toRightStochastic
  - atmath.hpp, 11
- transferOperator.hpp
  - filterTransitionMatrix, 14
  - getBoxMembership, 14
  - getGridMembership, 14
  - getGridRect, 14
  - getTransitionMatrix, 14
  - SpMatCSC, 14
  - SpMatCSR, 14
  - triplet, 14
  - tripletUInt, 14
  - tripletUIntVector, 14
  - tripletVector, 14
  - writeGridRect, 14
- transferOperatorTest.hpp
  - getShuffledCountMatrix, 15
  - getShuffledRow, 15
  - getSurrogateSpectrum, 15
  - getSurrogateSpectrumFromCount, 15
  - SpMatCSC, 15
  - SpMatCSR, 15
  - triplet, 15
  - tripletVector, 15
- Tri
  - atgraph.hpp, 4
  - atio.hpp, 7
- triplet
  - transferOperator.hpp, 14
  - transferOperatorTest.hpp, 15
- tripletUInt
  - transferOperator.hpp, 14
- tripletUIntVector
  - transferOperator.hpp, 14
- tripletVector
  - transferOperator.hpp, 14
  - transferOperatorTest.hpp, 15
- writeGridRect

transferOperator.hpp, [14](#)