

qpp
0.1

Generated by Doxygen 1.8.7

Sat Jul 26 2014 01:23:43

Contents

1	Namespace Index	1
1.1	Namespace List	1
2	Hierarchical Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	File Index	7
4.1	File List	7
5	Namespace Documentation	9
5.1	qpp Namespace Reference	9
5.1.1	Function Documentation	14
5.1.1.1	absm	14
5.1.1.2	adjoint	15
5.1.1.3	anticomm	15
5.1.1.4	apply	16
5.1.1.5	channel	17
5.1.1.6	channel	17
5.1.1.7	choi	18
5.1.1.8	choi2kraus	18
5.1.1.9	comm	19
5.1.1.10	compperm	19
5.1.1.11	conjugate	19
5.1.1.12	cosm	20
5.1.1.13	CTRL	20
5.1.1.14	cwise	21
5.1.1.15	det	21
5.1.1.16	disp	21
5.1.1.17	disp	21
5.1.1.18	disp	21

5.1.1.19	disp	21
5.1.1.20	displn	22
5.1.1.21	displn	22
5.1.1.22	displn	22
5.1.1.23	displn	23
5.1.1.24	entanglement	23
5.1.1.25	evals	24
5.1.1.26	evecs	24
5.1.1.27	expandout	25
5.1.1.28	expm	25
5.1.1.29	Fd	26
5.1.1.30	funm	26
5.1.1.31	Gates	26
5.1.1.32	gconcurrency	27
5.1.1.33	grams	27
5.1.1.34	grams	28
5.1.1.35	grams	28
5.1.1.36	hevals	28
5.1.1.37	hevecs	29
5.1.1.38	ld	29
5.1.1.39	inverse	29
5.1.1.40	invperm	29
5.1.1.41	kron	30
5.1.1.42	kron	30
5.1.1.43	kron	30
5.1.1.44	kron	30
5.1.1.45	kronpow	31
5.1.1.46	load	31
5.1.1.47	loadMATLABmatrix	31
5.1.1.48	loadMATLABmatrix	31
5.1.1.49	loadMATLABmatrix	31
5.1.1.50	logdet	31
5.1.1.51	logm	32
5.1.1.52	mket	32
5.1.1.53	mket	32
5.1.1.54	mket	33
5.1.1.55	multiidx2n	33
5.1.1.56	n2multiidx	33
5.1.1.57	norm	34
5.1.1.58	powm	34

5.1.1.59	prj	35
5.1.1.60	ptrace	36
5.1.1.61	ptrace1	37
5.1.1.62	ptrace2	37
5.1.1.63	ptranspose	38
5.1.1.64	qmutualinfo	39
5.1.1.65	rand	39
5.1.1.66	rand	39
5.1.1.67	rand	40
5.1.1.68	rand	40
5.1.1.69	randH	40
5.1.1.70	randint	41
5.1.1.71	randket	41
5.1.1.72	randkraus	41
5.1.1.73	randn	41
5.1.1.74	randn	42
5.1.1.75	randn	42
5.1.1.76	randn	42
5.1.1.77	RandomDevices	42
5.1.1.78	randperm	42
5.1.1.79	randrho	43
5.1.1.80	randU	43
5.1.1.81	randV	43
5.1.1.82	renyi	43
5.1.1.83	renyi_inf	44
5.1.1.84	reshape	44
5.1.1.85	Rn	44
5.1.1.86	save	44
5.1.1.87	saveMATLABmatrix	44
5.1.1.88	saveMATLABmatrix	45
5.1.1.89	saveMATLABmatrix	45
5.1.1.90	schmidtcoeff	45
5.1.1.91	schmidtprob	46
5.1.1.92	schmidtU	46
5.1.1.93	schmidtV	47
5.1.1.94	shannon	47
5.1.1.95	sinm	48
5.1.1.96	spectralpwm	48
5.1.1.97	sqrtn	48
5.1.1.98	States	49

5.1.1.99	sum	49
5.1.1.100	super	49
5.1.1.101	syspermute	50
5.1.1.102	trace	50
5.1.1.103	transpose	51
5.1.1.104	tsallis	51
5.1.1.105	Xd	51
5.1.1.106	Zd	52
5.1.2	Variable Documentation	52
5.1.2.1	_rng	52
5.1.2.2	b00	52
5.1.2.3	b01	52
5.1.2.4	b10	52
5.1.2.5	b11	52
5.1.2.6	CNOTab	52
5.1.2.7	CNOTba	52
5.1.2.8	CZ	52
5.1.2.9	FRED	52
5.1.2.10	GHZ	52
5.1.2.11	gt	52
5.1.2.12	H	52
5.1.2.13	pb00	52
5.1.2.14	pb01	52
5.1.2.15	pb10	52
5.1.2.16	pb11	52
5.1.2.17	pGHZ	52
5.1.2.18	pW	52
5.1.2.19	px0	52
5.1.2.20	px1	52
5.1.2.21	py0	53
5.1.2.22	py1	53
5.1.2.23	pz0	53
5.1.2.24	pz1	53
5.1.2.25	rdevs	53
5.1.2.26	S	53
5.1.2.27	st	53
5.1.2.28	SWAP	53
5.1.2.29	T	53
5.1.2.30	TOF	53
5.1.2.31	W	53

5.1.2.32	X	53
5.1.2.33	x1	53
5.1.2.34	Y	53
5.1.2.35	y0	53
5.1.2.36	y1	53
5.1.2.37	Z	53
5.1.2.38	z0	53
5.1.2.39	z1	53
5.2	qpp:: Namespace Reference	53
5.2.1	Function Documentation	54
5.2.1.1	omega	54
5.2.2	Variable Documentation	54
5.2.2.1	chop	54
5.2.2.2	ee	54
5.2.2.3	eps	54
5.2.2.4	ii	54
5.2.2.5	maxn	54
5.2.2.6	pi	54
5.3	qpp::internal Namespace Reference	54
5.3.1	Function Documentation	55
5.3.1.1	_check_col_vector	55
5.3.1.2	_check_dims	55
5.3.1.3	_check_dims_match_cvect	55
5.3.1.4	_check_dims_match_mat	55
5.3.1.5	_check_dims_match_rvect	55
5.3.1.6	_check_eq_dims	55
5.3.1.7	_check_nonzero_size	55
5.3.1.8	_check_perm	55
5.3.1.9	_check_row_vector	55
5.3.1.10	_check_square_mat	55
5.3.1.11	_check_subsys_match_dims	55
5.3.1.12	_check_vector	55
5.3.1.13	_kron2	55
5.3.1.14	_multiidx2n	55
5.3.1.15	_n2multiidx	55
5.3.1.16	variadic_vector_emplace	55
5.3.1.17	variadic_vector_emplace	56
5.4	qpp::types Namespace Reference	56
5.4.1	Typedef Documentation	56
5.4.1.1	bra	56

5.4.1.2	cmat	56
5.4.1.3	cplx	56
5.4.1.4	dmat	56
5.4.1.5	DynMat	56
5.4.1.6	ket	56
6	Class Documentation	57
6.1	qpp::DiscreteDistribution Class Reference	57
6.1.1	Constructor & Destructor Documentation	57
6.1.1.1	DiscreteDistribution	57
6.1.1.2	DiscreteDistribution	57
6.1.1.3	DiscreteDistribution	57
6.1.2	Member Function Documentation	57
6.1.2.1	probabilities	57
6.1.2.2	sample	57
6.1.3	Member Data Documentation	57
6.1.3.1	_d	57
6.2	qpp::DiscreteDistributionAbsSquare Class Reference	58
6.2.1	Constructor & Destructor Documentation	58
6.2.1.1	DiscreteDistributionAbsSquare	58
6.2.1.2	DiscreteDistributionAbsSquare	59
6.2.1.3	DiscreteDistributionAbsSquare	59
6.2.1.4	DiscreteDistributionAbsSquare	59
6.2.2	Member Function Documentation	60
6.2.2.1	cplx2weights	60
6.2.2.2	probabilities	60
6.2.2.3	sample	60
6.2.3	Member Data Documentation	60
6.2.3.1	_d	60
6.3	qpp::Exception Class Reference	60
6.3.1	Member Enumeration Documentation	61
6.3.1.1	Type	61
6.3.2	Constructor & Destructor Documentation	62
6.3.2.1	Exception	62
6.3.2.2	Exception	62
6.3.3	Member Function Documentation	62
6.3.3.1	_construct_exception_msg	62
6.3.3.2	what	62
6.3.4	Member Data Documentation	62
6.3.4.1	_custom	63

6.3.4.2	_msg	63
6.3.4.3	_type	63
6.3.4.4	_where	63
6.4	qpp::NormalDistribution Class Reference	63
6.4.1	Constructor & Destructor Documentation	63
6.4.1.1	NormalDistribution	63
6.4.2	Member Function Documentation	63
6.4.2.1	sample	63
6.4.3	Member Data Documentation	63
6.4.3.1	_d	63
6.5	qpp::Qudit Class Reference	63
6.5.1	Constructor & Destructor Documentation	64
6.5.1.1	Qudit	64
6.5.2	Member Function Documentation	64
6.5.2.1	getD	64
6.5.2.2	getRho	64
6.5.2.3	measure	64
6.5.2.4	measure	65
6.5.3	Member Data Documentation	65
6.5.3.1	_D	65
6.5.3.2	_rho	65
6.6	qpp::Singleton< T > Class Template Reference	65
6.6.1	Constructor & Destructor Documentation	65
6.6.1.1	Singleton	65
6.6.1.2	~Singleton	65
6.6.1.3	Singleton	66
6.6.2	Member Function Documentation	66
6.6.2.1	get_instance	66
6.6.2.2	operator=	66
6.7	qpp::Timer Class Reference	66
6.7.1	Constructor & Destructor Documentation	66
6.7.1.1	Timer	66
6.7.2	Member Function Documentation	66
6.7.2.1	seconds	66
6.7.2.2	tic	66
6.7.2.3	toc	66
6.7.3	Friends And Related Function Documentation	66
6.7.3.1	operator<<	66
6.7.4	Member Data Documentation	66
6.7.4.1	_end	67

6.7.4.2	<code>_start</code>	67
6.8	<code>qpp::UniformRealDistribution</code> Class Reference	67
6.8.1	Constructor & Destructor Documentation	67
6.8.1.1	<code>UniformRealDistribution</code>	67
6.8.2	Member Function Documentation	67
6.8.2.1	<code>sample</code>	67
6.8.3	Member Data Documentation	67
6.8.3.1	<code>_d</code>	67
7	File Documentation	69
7.1	<code>include/channels.h</code> File Reference	69
7.2	<code>include/classes/exception.h</code> File Reference	70
7.3	<code>include/classes/gates.h</code> File Reference	70
7.4	<code>include/classes/qudit.h</code> File Reference	71
7.5	<code>include/classes/randevs.h</code> File Reference	72
7.6	<code>include/classes/singleton.h</code> File Reference	73
7.6.1	Macro Definition Documentation	73
7.6.1.1	<code>CLASS_CONST_SINGLETON</code>	73
7.6.1.2	<code>CLASS_SINGLETON</code>	73
7.7	<code>include/classes/stat.h</code> File Reference	74
7.8	<code>include/classes/states.h</code> File Reference	74
7.9	<code>include/classes/timer.h</code> File Reference	75
7.10	<code>include/constants.h</code> File Reference	76
7.11	<code>include/entanglement.h</code> File Reference	77
7.12	<code>include/entropies.h</code> File Reference	78
7.13	<code>include/functions.h</code> File Reference	78
7.14	<code>include/internal.h</code> File Reference	81
7.15	<code>include/io.h</code> File Reference	82
7.16	<code>include/matlab.h</code> File Reference	83
7.17	<code>include/qpp.h</code> File Reference	84
7.18	<code>include/random.h</code> File Reference	85
7.19	<code>include/types.h</code> File Reference	86
	Index	87

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

qpp	9
qpp::ct	53
qpp::internal	54
qpp::types	56

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

qpp::DiscreteDistribution	57
qpp::DiscreteDistributionAbsSquare	58
exception	
qpp::Exception	60
qpp::NormalDistribution	63
qpp::Qudit	63
qpp::Singleton< T >	65
qpp::Timer	66
qpp::UniformRealDistribution	67

Chapter 3

Class Index

3.1 Class List

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

qpp::DiscreteDistribution	57
qpp::DiscreteDistributionAbsSquare	58
qpp::Exception	60
qpp::NormalDistribution	63
qpp::Qudit	63
qpp::Singleton< T >	65
qpp::Timer	66
qpp::UniformRealDistribution	67

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

include/channels.h	69
include/constants.h	76
include/entanglement.h	77
include/entropies.h	78
include/functions.h	78
include/internal.h	81
include/io.h	82
include/matlab.h	83
include/qpp.h	84
include/random.h	85
include/types.h	86
include/classes/exception.h	70
include/classes/gates.h	70
include/classes/qudit.h	71
include/classes/randevs.h	72
include/classes/singleton.h	73
include/classes/stat.h	74
include/classes/states.h	74
include/classes/timer.h	75

Chapter 5

Namespace Documentation

5.1 qpp Namespace Reference

Namespaces

- [ct](#)
- [internal](#)
- [types](#)

Classes

- class [DiscreteDistribution](#)
- class [DiscreteDistributionAbsSquare](#)
- class [Exception](#)
- class [NormalDistribution](#)
- class [Qudit](#)
- class [Singleton](#)
- class [Timer](#)
- class [UniformRealDistribution](#)

Functions

- [types::cmat super](#) (const std::vector< [types::cmat](#) > &Ks)
- [types::cmat choi](#) (const std::vector< [types::cmat](#) > &Ks)
- std::vector< [types::cmat](#) > [choi2kraus](#) (const [types::cmat](#) &A)
- template<typename Derived >
[types::cmat channel](#) (const Eigen::MatrixBase< Derived > &rho, const std::vector< [types::cmat](#) > &Ks)
- template<typename Derived >
[types::cmat channel](#) (const Eigen::MatrixBase< Derived > &rho, const std::vector< [types::cmat](#) > &Ks,
const std::vector< std::size_t > &subsys, const std::vector< std::size_t > &dims)
- [Gates](#) ()
- [types::cmat Rn](#) (double theta, std::vector< double > n) const
- [types::cmat Zd](#) (std::size_t D) const
- [types::cmat Fd](#) (std::size_t D) const
- [types::cmat Xd](#) (std::size_t D) const
- template<typename Derived = Eigen::MatrixXcd>
Derived [ld](#) (std::size_t D) const

- `template<typename Derived1 , typename Derived2 >`
`types::DynMat< typename`
`Derived1::Scalar > apply` (const Eigen::MatrixBase< Derived1 > &state, const Eigen::MatrixBase< Derived2 > &A, const std::vector< std::size_t > &subsys, const std::vector< std::size_t > &dims) const
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > CTRL` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &ctrl, const std::vector< std::size_t > &subsys, std::size_t n, std::size_t d=2) const
- `RandomDevices ()`
- `States ()`
- `template<typename Derived >`
`types::cmat schmidtcoeff` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`types::cmat schmidtU` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`types::cmat schmidtV` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`types::cmat schmidtprob` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`double entanglement` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`double gconcurrency` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`double shannon` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`double renyi` (const double alpha, const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`double renyi_inf` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`double tsallis` (const double alpha, const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`double qmutualinfo` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &subsys, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > transpose` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > conjugate` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > adjoint` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > inverse` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`Derived::Scalar trace` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`Derived::Scalar det` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`Derived::Scalar logdet` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`Derived::Scalar sum` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`double norm` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat evals` (const Eigen::MatrixBase< Derived > &A)

- `template<typename Derived >`
`types::cmat evects` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::dmat hevals` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat hevects` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat funm` (const Eigen::MatrixBase< Derived > &A, `types::cplx`(*f)(const `types::cplx` &))
- `template<typename Derived >`
`types::cmat sqrtm` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat absm` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat expm` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat logm` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat sinm` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat cosm` (const Eigen::MatrixBase< Derived > &A)
- `template<typename Derived >`
`types::cmat spectralpowm` (const Eigen::MatrixBase< Derived > &A, const `types::cplx` z)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > powm` (const Eigen::MatrixBase< Derived > &A, std::size_t n)
- `template<typename OutputScalar , typename Derived >`
`types::DynMat< OutputScalar > cwise` (const Eigen::MatrixBase< Derived > &A, OutputScalar(*f)(const
`typename Derived::Scalar &))`
- `template<typename T >`
`types::DynMat< typename T::Scalar > kron` (const T &head)
- `template<typename T , typename... Args>`
`types::DynMat< typename T::Scalar > kron` (const T &head, const Args &...tail)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > kron` (const std::vector< Derived > &As)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > kron` (const std::initializer_list< Derived > &As)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > kronpow` (const Eigen::MatrixBase< Derived > &A, std::size_t n)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > reshape` (const Eigen::MatrixBase< Derived > &A, std::size_t rows, std::size_t cols)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > syspermute` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t >
&perm, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > ptrace1` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > ptrace2` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > ptrace` (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &sub-
sys, const std::vector< std::size_t > &dims)

- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > ptranspose (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t >`
`&subsys, const std::vector< std::size_t > &dims)`
- `template<typename Derived1 , typename Derived2 >`
`types::DynMat< typename`
`Derived1::Scalar > comm (const Eigen::MatrixBase< Derived1 > &A, const Eigen::MatrixBase< Derived2`
`> &B)`
- `template<typename Derived1 , typename Derived2 >`
`types::DynMat< typename`
`Derived1::Scalar > anticomm (const Eigen::MatrixBase< Derived1 > &A, const Eigen::MatrixBase< De-`
`derived2 > &B)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > prj (const Eigen::MatrixBase< Derived > &V)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > expandout (const Eigen::MatrixBase< Derived > &A, std::size_t pos, const std::vector<`
`std::size_t > &dims)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > grams (const std::vector< Derived > &Vs)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > grams (const std::initializer_list< Derived > &Vs)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > grams (const Eigen::MatrixBase< Derived > &A)`
- `std::vector< std::size_t > n2multiidx (std::size_t n, const std::vector< std::size_t > &dims)`
- `std::size_t multiidx2n (const std::vector< std::size_t > &midx, const std::vector< std::size_t > &dims)`
- `types::ket mket (const std::vector< std::size_t > &mask)`
- `types::ket mket (const std::vector< std::size_t > &mask, const std::vector< std::size_t > &dims)`
- `types::ket mket (const std::vector< std::size_t > &mask, std::size_t d)`
- `std::vector< std::size_t > invperm (const std::vector< std::size_t > &perm)`
- `std::vector< std::size_t > compperm (const std::vector< std::size_t > &perm, const std::vector< std::size_t`
`> &sigma)`
- `template<typename T >`
`void disp (const T &x, const std::string &separator, const std::string &start="[", const std::string &end="]",`
`std::ostream &os=std::cout)`
- `template<typename T >`
`void displn (const T &x, const std::string &separator, const std::string &start="[", const std::string &end="]",`
`std::ostream &os=std::cout)`
- `template<typename T >`
`void disp (const T *x, const std::size_t n, const std::string &separator, const std::string &start="[", const std::←`
`::string &end="]", std::ostream &os=std::cout)`
- `template<typename T >`
`void displn (const T *x, const std::size_t n, const std::string &separator, const std::string &start="[", const`
`std::string &end="]", std::ostream &os=std::cout)`
- `template<typename Derived >`
`void disp (const Eigen::MatrixBase< Derived > &A, double chop=ct::chop, std::ostream &os=std::cout)`
- `template<typename Derived >`
`void displn (const Eigen::MatrixBase< Derived > &A, double chop=ct::chop, std::ostream &os=std::cout)`
- `void disp (const types::cplx c, double chop=ct::chop, std::ostream &os=std::cout)`
- `void displn (const types::cplx c, double chop=ct::chop, std::ostream &os=std::cout)`
- `template<typename Derived >`
`void save (const Eigen::MatrixBase< Derived > &A, const std::string &fname)`

- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > load` (const std::string &fname)
- `template<typename Derived >`
`Derived loadMATLABmatrix` (const std::string &mat_file, const std::string &var_name)
- `template<>`
`types::dmat loadMATLABmatrix` (const std::string &mat_file, const std::string &var_name)
- `template<>`
`types::cmat loadMATLABmatrix` (const std::string &mat_file, const std::string &var_name)
- `template<typename Derived >`
`void saveMATLABmatrix` (const Eigen::MatrixBase< Derived > &A, const std::string &mat_file, const std::string &var_name, const std::string &mode)
- `template<>`
`void saveMATLABmatrix` (const Eigen::MatrixBase< typename `types::dmat` > &A, const std::string &mat_file, const std::string &var_name, const std::string &mode)
- `template<>`
`void saveMATLABmatrix` (const Eigen::MatrixBase< typename `types::cmat` > &A, const std::string &mat_file, const std::string &var_name, const std::string &mode)
- `template<typename Derived >`
`Derived rand` (std::size_t rows, std::size_t cols, double a=0, double b=1)
- `template<>`
`types::dmat rand` (std::size_t rows, std::size_t cols, double a, double b)
- `template<>`
`types::cmat rand` (std::size_t rows, std::size_t cols, double a, double b)
- `double rand` (double a=0, double b=1)
- `long long randint` (long long a, long long b)
- `template<typename Derived >`
`Derived randn` (std::size_t rows, std::size_t cols, double mean=0, double sigma=1)
- `template<>`
`types::dmat randn` (std::size_t rows, std::size_t cols, double mean, double sigma)
- `template<>`
`types::cmat randn` (std::size_t rows, std::size_t cols, double mean, double sigma)
- `double randn` (double mean=0, double sigma=1)
- `types::cmat randU` (std::size_t D)
- `types::cmat randV` (std::size_t Din, std::size_t Dout)
- `std::vector< types::cmat > randkraus` (std::size_t n, std::size_t D)
- `types::cmat randH` (std::size_t D)
- `types::ket randket` (std::size_t D)
- `types::cmat randrho` (std::size_t D)
- `std::vector< std::size_t > randperm` (std::size_t n)

Variables

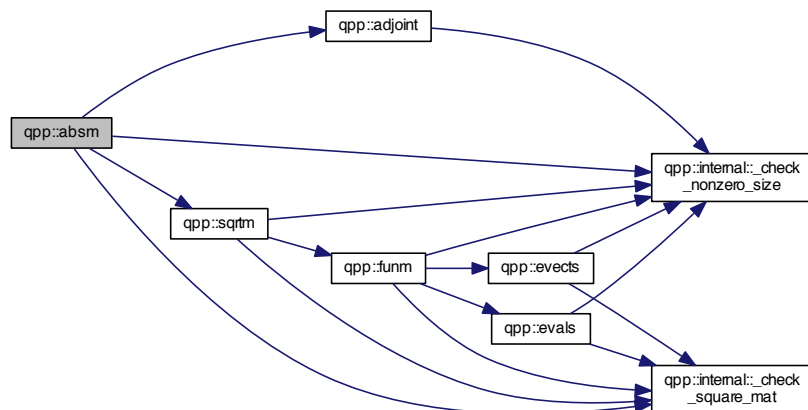
- `CLASS_CONST_SINGLETON(Gates)`
`public types::cmat H`
- `types::cmat X`
- `types::cmat Y`
- `types::cmat Z`
- `types::cmat S`
- `types::cmat T`
- `types::cmat CNOTab`
- `types::cmat CZ`
- `types::cmat CNOTba`
- `types::cmat SWAP`
- `types::cmat TOF`

- `types::cmat FRED`
- `CLASS_CONST_SINGLETON(RandomDevices)`
public `std::mt19937 _rng`
- `CLASS_CONST_SINGLETON(States)`
public `types::ket x1`
- `types::ket y0`
- `types::ket y1`
- `types::ket z0`
- `types::ket z1`
- `types::cmat px0`
- `types::cmat px1`
- `types::cmat py0`
- `types::cmat py1`
- `types::cmat pz0`
- `types::cmat pz1`
- `types::ket b00`
- `types::ket b01`
- `types::ket b10`
- `types::ket b11`
- `types::cmat pb00`
- `types::cmat pb01`
- `types::cmat pb10`
- `types::cmat pb11`
- `types::ket GHZ`
- `types::ket W`
- `types::cmat pGHZ`
- `types::cmat pW`
- `const RandomDevices & rdevs = RandomDevices::get_instance()`
- `const Gates & gt = Gates::get_instance()`
- `const States & st = States::get_instance()`

5.1.1 Function Documentation

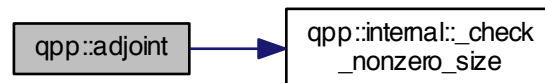
5.1.1.1 `template<typename Derived > types::cmat qpp::absm (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



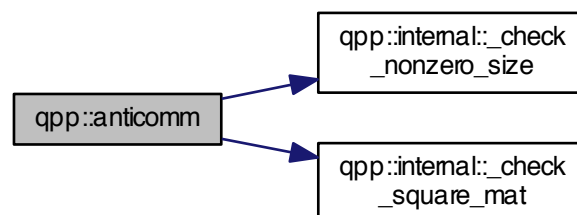
5.1.1.2 `template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::adjoint (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



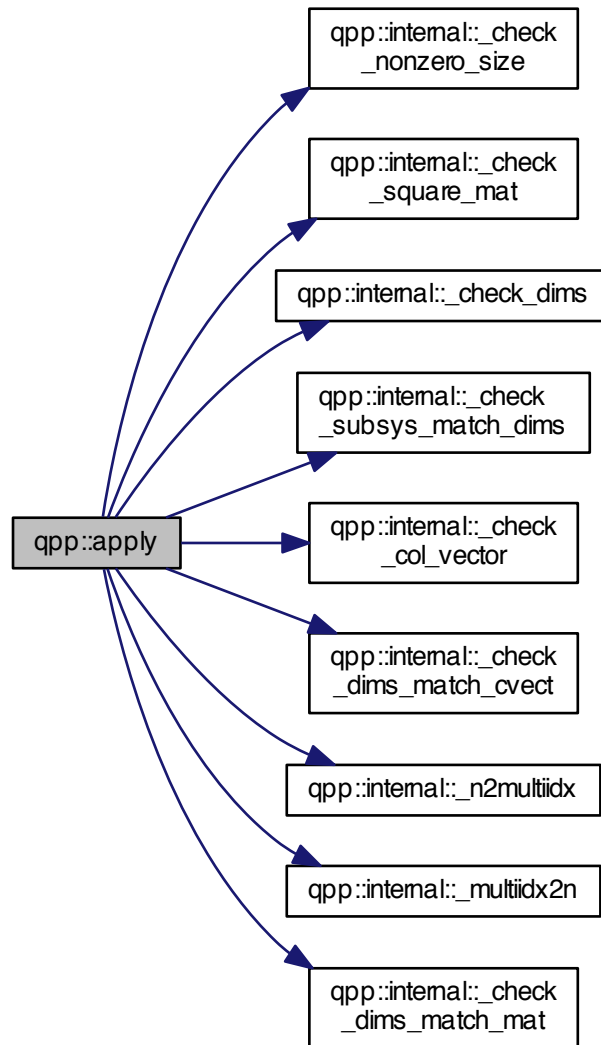
5.1.1.3 `template<typename Derived1 , typename Derived2 > types::DynMat<typename Derived1::Scalar> qpp::anticomm (const Eigen::MatrixBase< Derived1 > & A, const Eigen::MatrixBase< Derived2 > & B)`

Here is the call graph for this function:



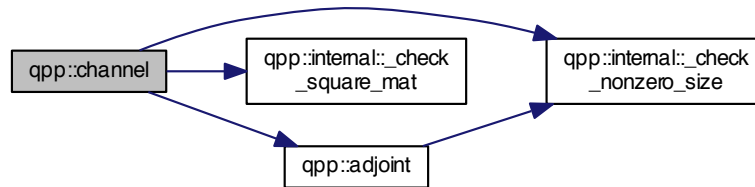
5.1.1.4 `template<typename Derived1 , typename Derived2 > types::DynMat<typename Derived1::Scalar> qpp::apply
(const Eigen::MatrixBase< Derived1 > & state, const Eigen::MatrixBase< Derived2 > & A, const std::vector<
std::size_t > & subsys, const std::vector< std::size_t > & dims) const`

Here is the call graph for this function:



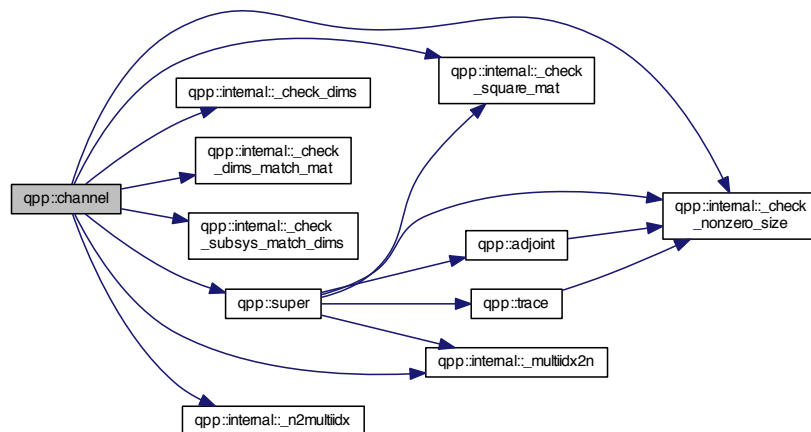
5.1.1.5 `template<typename Derived > types::cmat qpp::channel (const Eigen::MatrixBase< Derived > & rho, const std::vector< types::cmat > & Ks)`

Here is the call graph for this function:



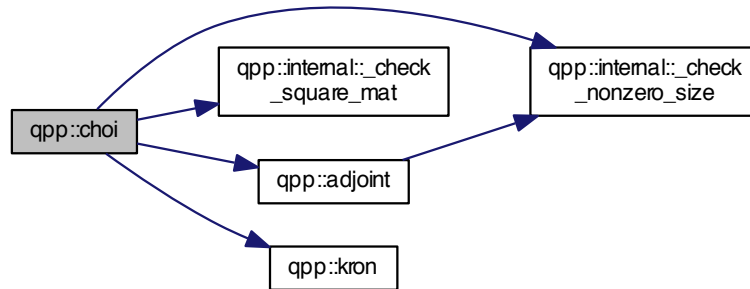
5.1.1.6 `template<typename Derived > types::cmat qpp::channel (const Eigen::MatrixBase< Derived > & rho, const std::vector< types::cmat > & Ks, const std::vector< std::size_t > & subsys, const std::vector< std::size_t > & dims)`

Here is the call graph for this function:



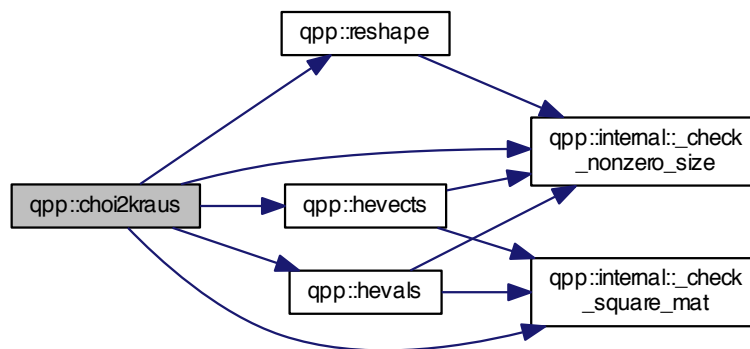
5.1.1.7 `types::cmat qpp::choi (const std::vector< types::cmat > & Ks)`

Here is the call graph for this function:



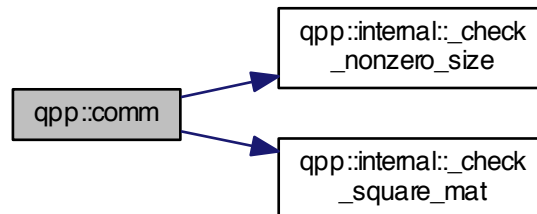
5.1.1.8 `std::vector<types::cmat> qpp::choi2kraus (const types::cmat & A)`

Here is the call graph for this function:



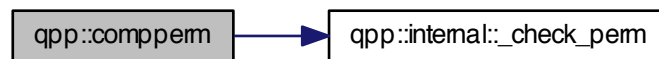
5.1.1.9 `template<typename Derived1 , typename Derived2 > types::DynMat<typename Derived1::Scalar> qpp::comm (const Eigen::MatrixBase< Derived1 > & A, const Eigen::MatrixBase< Derived2 > & B)`

Here is the call graph for this function:



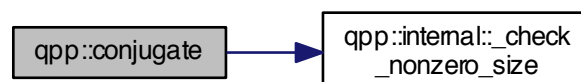
5.1.1.10 `std::vector<std::size_t> qpp::compperm (const std::vector< std::size_t > & perm, const std::vector< std::size_t > & sigma)`

Here is the call graph for this function:



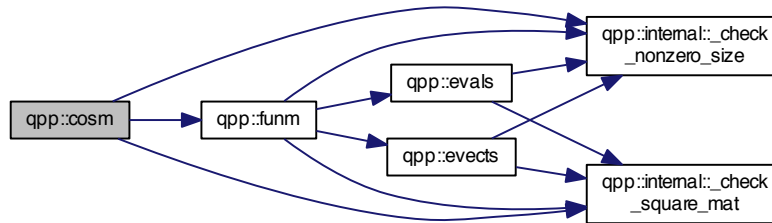
5.1.1.11 `template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::conjugate (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



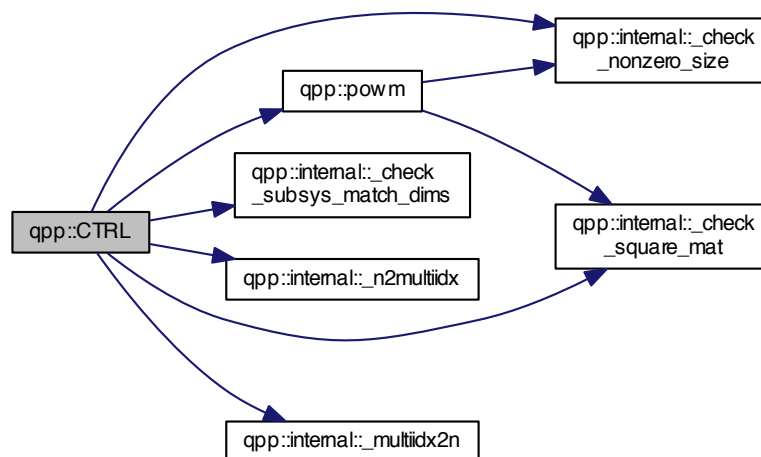
5.1.1.12 `template<typename Derived> types::cmat qpp::cosm (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



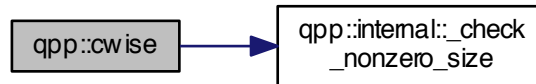
5.1.1.13 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::CTRL (const Eigen::MatrixBase< Derived > & A, const std::vector< std::size_t > & ctrl, const std::vector< std::size_t > & subsys, std::size_t n, std::size_t d = 2) const`

Here is the call graph for this function:



5.1.1.14 `template<typename OutputScalar , typename Derived > types::DynMat<OutputScalar> qpp::cwise (const Eigen::MatrixBase< Derived > & A, OutputScalar*)(const typename Derived::Scalar &) f)`

Here is the call graph for this function:



5.1.1.15 `template<typename Derived > Derived::Scalar qpp::det (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



5.1.1.16 `template<typename T > void qpp::disp (const T & x, const std::string & separator, const std::string & start = " [", const std::string & end = "] ", std::ostream & os = std::cout)`

5.1.1.17 `template<typename T > void qpp::disp (const T * x, const std::size_t n, const std::string & separator, const std::string & start = " [", const std::string & end = "] ", std::ostream & os = std::cout)`

5.1.1.18 `template<typename Derived > void qpp::disp (const Eigen::MatrixBase< Derived > & A, double chop = ct::chop, std::ostream & os = std::cout)`

5.1.1.19 `void qpp::disp (const types::cplx c, double chop = ct::chop, std::ostream & os = std::cout)`

Here is the call graph for this function:



5.1.1.20 `template<typename T> void qpp::displn (const T & x, const std::string & separator, const std::string & start = " [", const std::string & end = "]" , std::ostream & os = std::cout)`

Here is the call graph for this function:



5.1.1.21 `template<typename T> void qpp::displn (const T * x, const std::size_t n, const std::string & separator, const std::string & start = " [", const std::string & end = "]" , std::ostream & os = std::cout)`

Here is the call graph for this function:



5.1.1.22 `template<typename Derived> void qpp::displn (const Eigen::MatrixBase< Derived > & A, double chop = ct::chop, std::ostream & os = std::cout)`

Here is the call graph for this function:



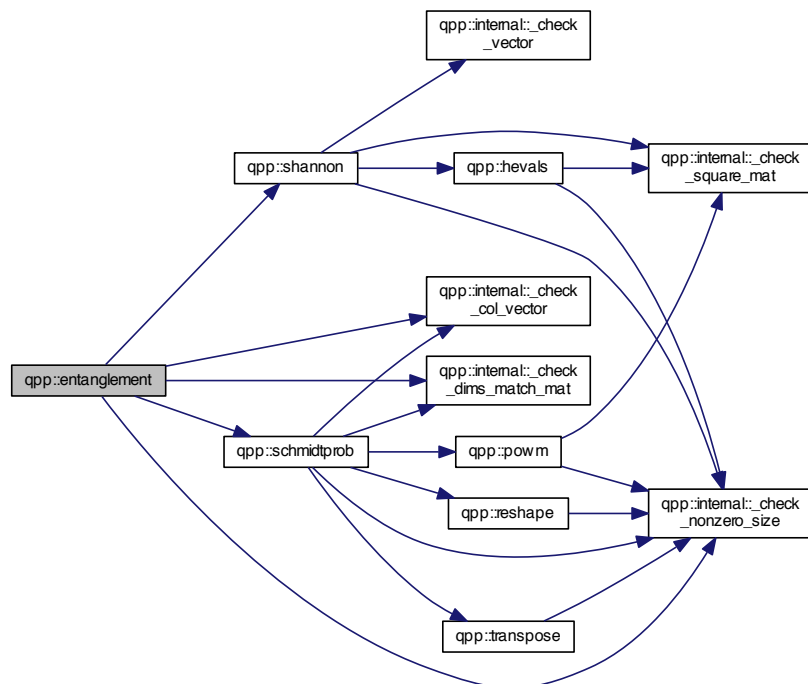
5.1.1.23 `void qpp::displn (const types::cplx c, double chop = ct:::chop, std::ostream & os = std:::cout)`

Here is the call graph for this function:



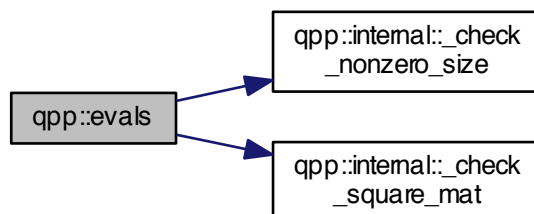
5.1.1.24 `template<typename Derived> double qpp::entanglement (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



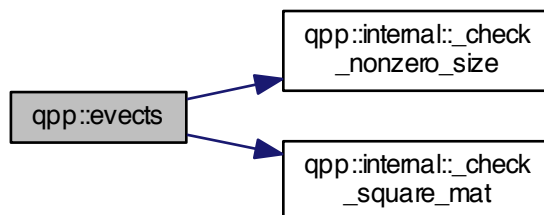
5.1.1.25 `template<typename Derived > types::cmat qpp::evals (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



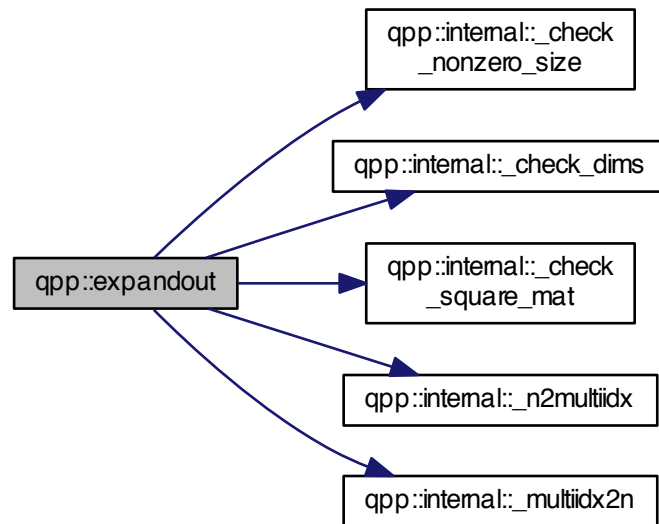
5.1.1.26 `template<typename Derived > types::cmat qpp::evecs (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



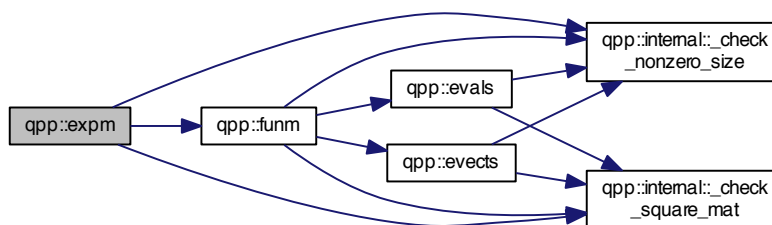
5.1.1.27 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::expandout (const Eigen::MatrixBase< Derived> & A, std::size_t pos, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



5.1.1.28 `template<typename Derived> types::cmat qpp::expm (const Eigen::MatrixBase< Derived> & A)`

Here is the call graph for this function:



5.1.1.29 `types::cmat qpp::Fd (std::size_t D) const`

Here is the call graph for this function:



5.1.1.30 `template<typename Derived > types::cmat qpp::funm (const Eigen::MatrixBase< Derived > & A, types::cplx*)(const types::cplx &) f)`

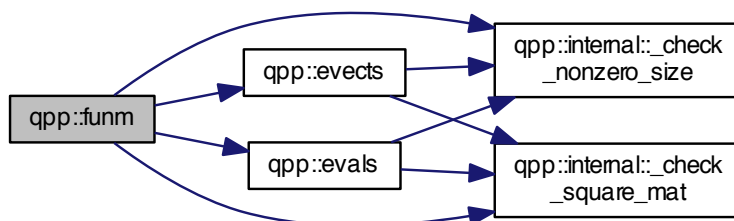
Parameters

<i>A</i>	input matrix
<i>f</i>	function pointer

Returns

[types::cmat](#)

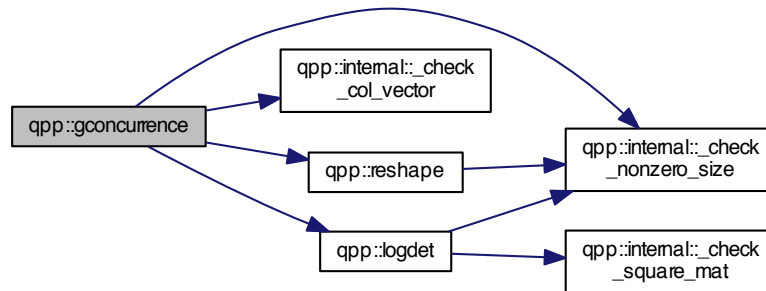
Here is the call graph for this function:



5.1.1.31 `qpp::Gates () [private]`

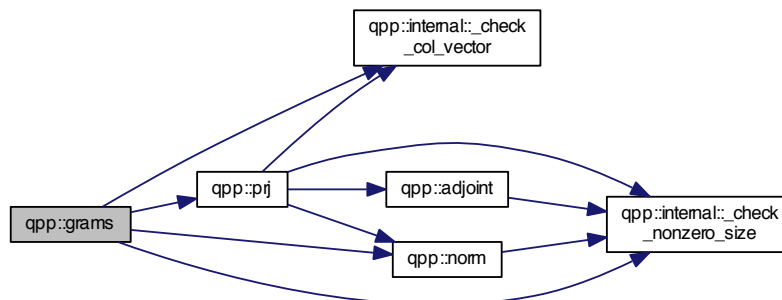
5.1.1.32 `template<typename Derived> double qpp::gconcurrency (const Eigen::MatrixBase< Derived> & A)`

Here is the call graph for this function:



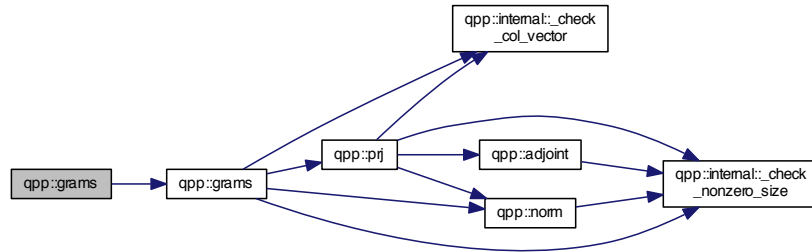
5.1.1.33 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::grams (const std::vector< Derived> & Vs)`

Here is the call graph for this function:



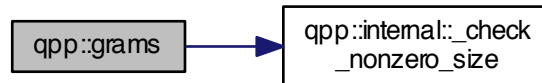
5.1.1.34 `template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::grams (const std::initializer_list< Derived > & Vs)`

Here is the call graph for this function:



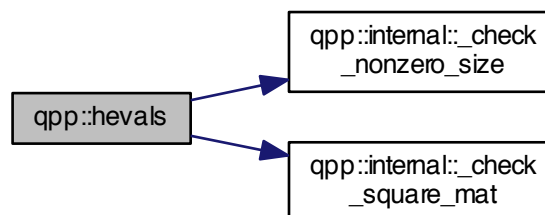
5.1.1.35 `template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::grams (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



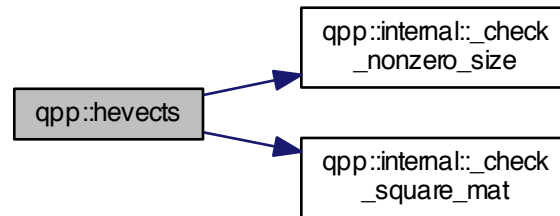
5.1.1.36 `template<typename Derived > types::dmat qpp::hevals (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



5.1.1.37 `template<typename Derived> types::cmat qpp::hevects (const Eigen::MatrixBase< Derived > & A)`

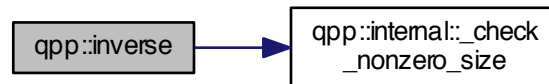
Here is the call graph for this function:



5.1.1.38 `template<typename Derived = Eigen::MatrixXcd> Derived qpp::ld (std::size_t D) const`

5.1.1.39 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::inverse (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



5.1.1.40 `std::vector<std::size_t> qpp::invperm (const std::vector< std::size_t > & perm)`

Here is the call graph for this function:



5.1.1.41 `template<typename T> types::DynMat<typename T::Scalar> qpp::kron (const T & head)`

5.1.1.42 `template<typename T, typename... Args> types::DynMat<typename T::Scalar> qpp::kron (const T & head, const Args &... tail)`

Here is the call graph for this function:



5.1.1.43 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::kron (const std::vector<Derived> & As)`

Here is the call graph for this function:



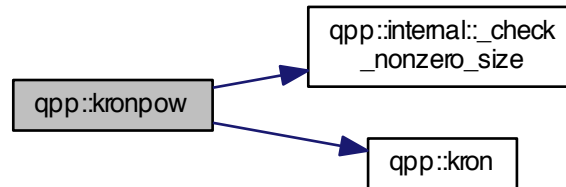
5.1.1.44 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::kron (const std::initializer_list<Derived> & As)`

Here is the call graph for this function:



5.1.1.45 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::kronpow (const Eigen::MatrixBase< Derived> & A, std::size_t n)`

Here is the call graph for this function:



5.1.1.46 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::load (const std::string & fname)`

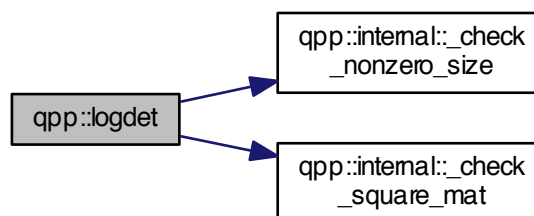
5.1.1.47 `template<typename Derived> Derived qpp::loadMATLABmatrix (const std::string & mat_file, const std::string & var_name)`

5.1.1.48 `template<> types::dmat qpp::loadMATLABmatrix (const std::string & mat_file, const std::string & var_name)`

5.1.1.49 `template<> types::cmat qpp::loadMATLABmatrix (const std::string & mat_file, const std::string & var_name)`

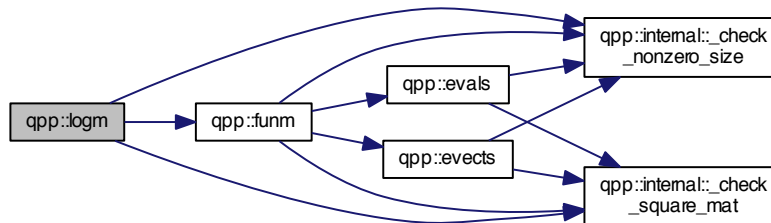
5.1.1.50 `template<typename Derived> Derived::Scalar qpp::logdet (const Eigen::MatrixBase< Derived> & A)`

Here is the call graph for this function:



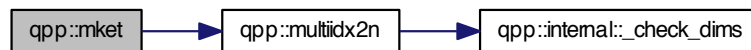
5.1.1.51 `template<typename Derived> types::cmat qpp::logm (const Eigen::MatrixBase< Derived> & A)`

Here is the call graph for this function:



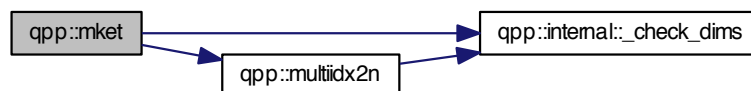
5.1.1.52 `types::ket qpp::mket (const std::vector< std::size_t> & mask)`

Here is the call graph for this function:



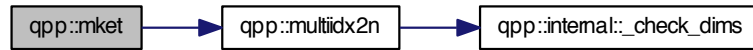
5.1.1.53 `types::ket qpp::mket (const std::vector< std::size_t> & mask, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



5.1.1.54 `types::ket qpp::mket (const std::vector< std::size_t > & mask, std::size_t d)`

Here is the call graph for this function:



5.1.1.55 `std::size_t qpp::multiidx2n (const std::vector< std::size_t > & midx, const std::vector< std::size_t > & dims)`

Here is the call graph for this function:



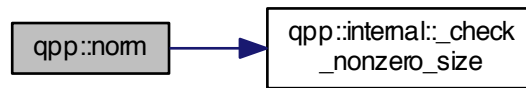
5.1.1.56 `std::vector<std::size_t> qpp::n2multiidx (std::size_t n, const std::vector< std::size_t > & dims)`

Here is the call graph for this function:



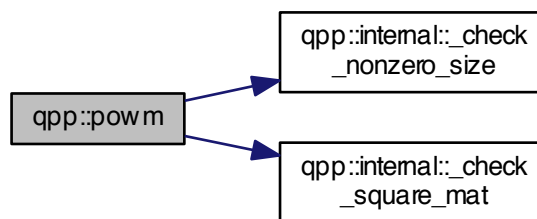
5.1.1.57 `template<typename Derived> double qpp::norm (const Eigen::MatrixBase< Derived> & A)`

Here is the call graph for this function:



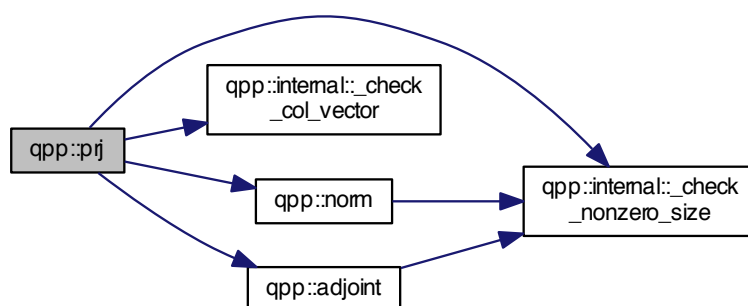
5.1.1.58 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::powm (const Eigen::MatrixBase< Derived> & A, std::size_t n)`

Here is the call graph for this function:



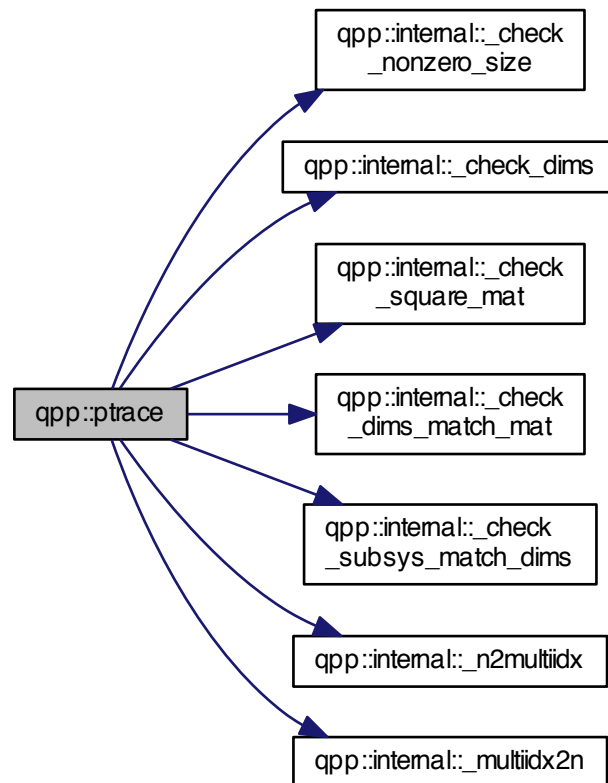
5.1.1.59 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::prj (const Eigen::MatrixBase<Derived> & V)`

Here is the call graph for this function:



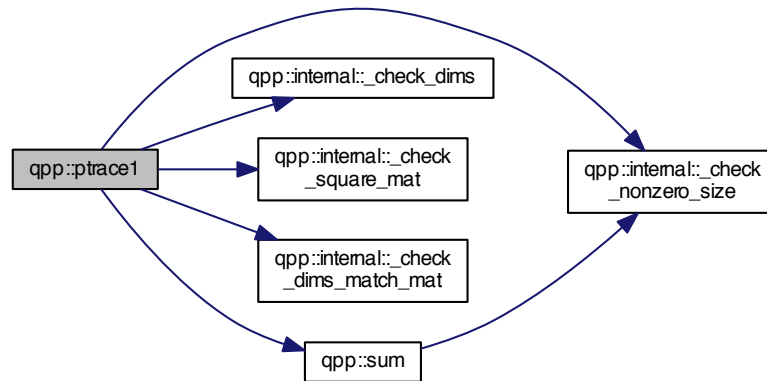
5.1.1.60 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::ptrace (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & subsys, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



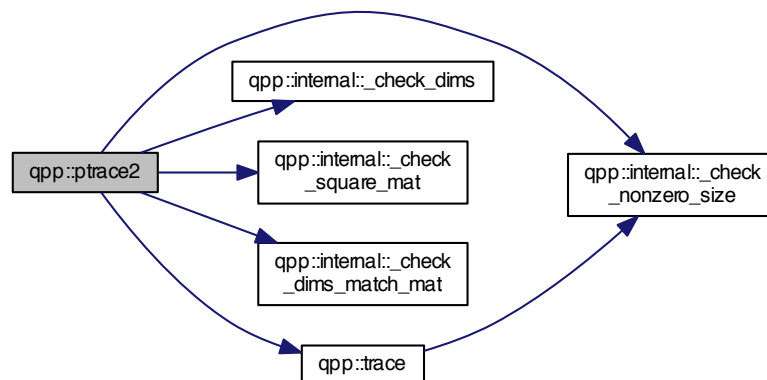
5.1.1.61 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::ptrace1 (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



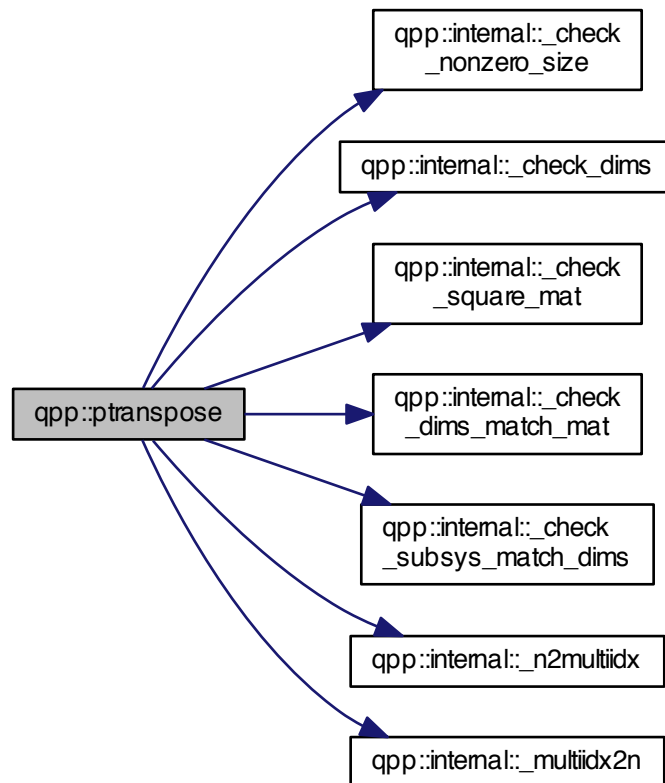
5.1.1.62 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::ptrace2 (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



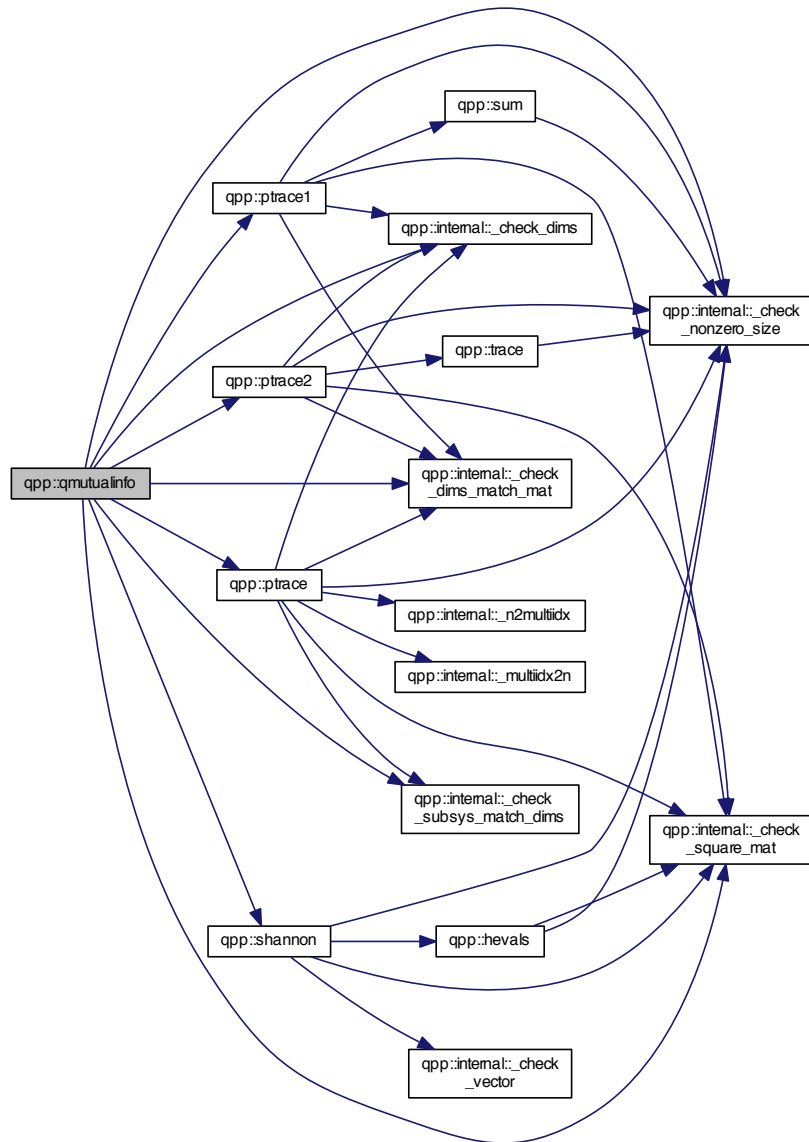
5.1.1.63 `template<typename Derived> types::DynMat<typename Derived::Scalar> qpp::ptranspose (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & subsys, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



5.1.1.64 `template<typename Derived> double qpp::qmutualinfo (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & subsys, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



5.1.1.65 `template<typename Derived> Derived qpp::rand (std::size_t rows, std::size_t cols, double a = 0, double b = 1)`

5.1.1.66 `template<> types::dmat qpp::rand (std::size_t rows, std::size_t cols, double a, double b)`

5.1.1.67 `template<> types::cmat qpp::rand (std::size_t rows, std::size_t cols, double a, double b)`

Here is the call graph for this function:



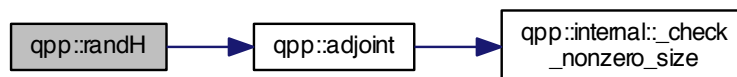
5.1.1.68 `double qpp::rand (double a = 0, double b = 1)`

Here is the call graph for this function:



5.1.1.69 `types::cmat qpp::randH (std::size_t D)`

Here is the call graph for this function:

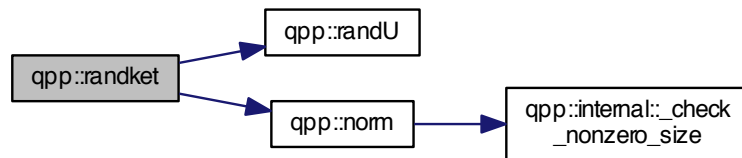


5.1.1.70 `long long qpp::randint (long long a, long long b)`

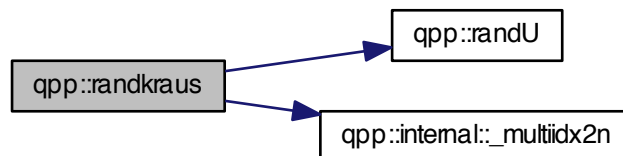
Here is the call graph for this function:

5.1.1.71 `types::ket qpp::randket (std::size_t D)`

Here is the call graph for this function:

5.1.1.72 `std::vector<types::cmat> qpp::randkraus (std::size_t n, std::size_t D)`

Here is the call graph for this function:

5.1.1.73 `template<typename Derived> Derived qpp::randn (std::size_t rows, std::size_t cols, double mean = 0, double sigma = 1)`

5.1.1.74 `template<> types::dmat qpp::randn (std::size_t rows, std::size_t cols, double mean, double sigma)`

Here is the call graph for this function:



5.1.1.75 `template<> types::cmat qpp::randn (std::size_t rows, std::size_t cols, double mean, double sigma)`

Here is the call graph for this function:



5.1.1.76 `double qpp::randn (double mean = 0, double sigma = 1)`

Here is the call graph for this function:

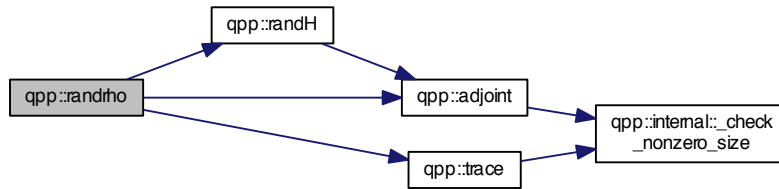


5.1.1.77 `qpp::RandomDevices () [private]`

5.1.1.78 `std::vector<std::size_t> qpp::randperm (std::size_t n)`

5.1.1.79 `types::cmat qpp::randrho (std::size_t D)`

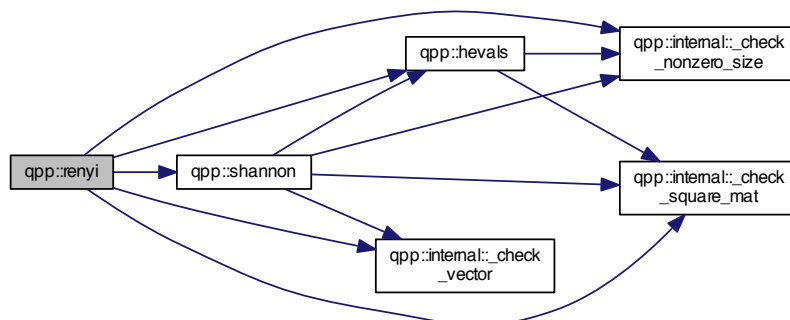
Here is the call graph for this function:

5.1.1.80 `types::cmat qpp::randU (std::size_t D)`5.1.1.81 `types::cmat qpp::randV (std::size_t Din, std::size_t Dout)`

Here is the call graph for this function:

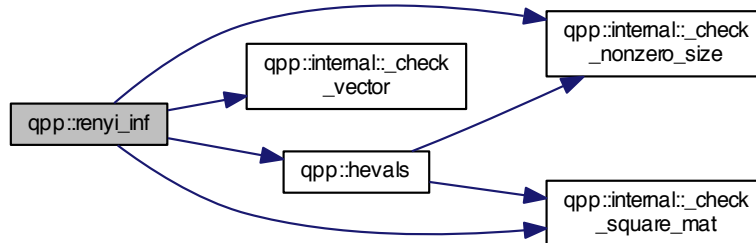
5.1.1.82 `template<typename Derived> double qpp::renyi (const double alpha, const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



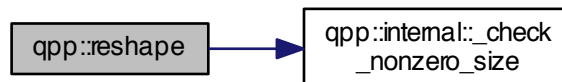
5.1.1.83 `template<typename Derived > double qpp::renyi_inf (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



5.1.1.84 `template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::reshape (const Eigen::MatrixBase< Derived > & A, std::size_t rows, std::size_t cols)`

Here is the call graph for this function:



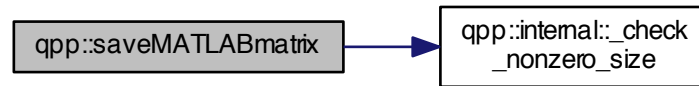
5.1.1.85 `types::cmat qpp::Rn (double theta, std::vector< double > n) const`

5.1.1.86 `template<typename Derived > void qpp::save (const Eigen::MatrixBase< Derived > & A, const std::string & fname)`

5.1.1.87 `template<typename Derived > void qpp::saveMATLABmatrix (const Eigen::MatrixBase< Derived > & A, const std::string & mat_file, const std::string & var_name, const std::string & mode)`

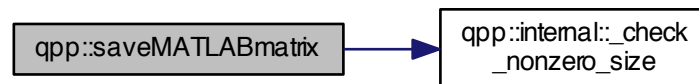
5.1.1.88 `template<> void qpp::saveMATLABmatrix (const Eigen::MatrixBase< typename types::dmat > & A, const std::string & mat_file, const std::string & var_name, const std::string & mode)`

Here is the call graph for this function:



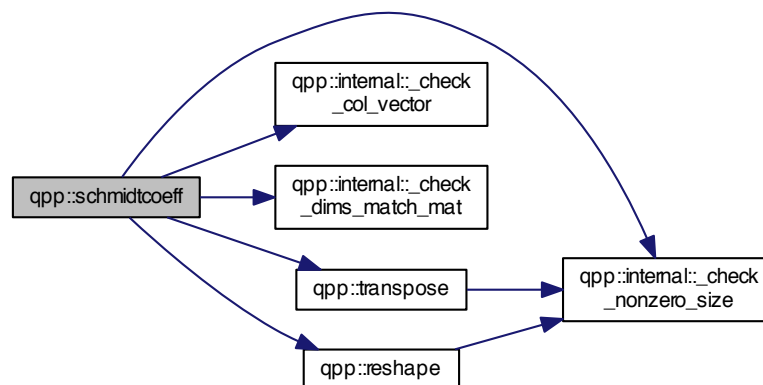
5.1.1.89 `template<> void qpp::saveMATLABmatrix (const Eigen::MatrixBase< typename types::cmat > & A, const std::string & mat_file, const std::string & var_name, const std::string & mode)`

Here is the call graph for this function:



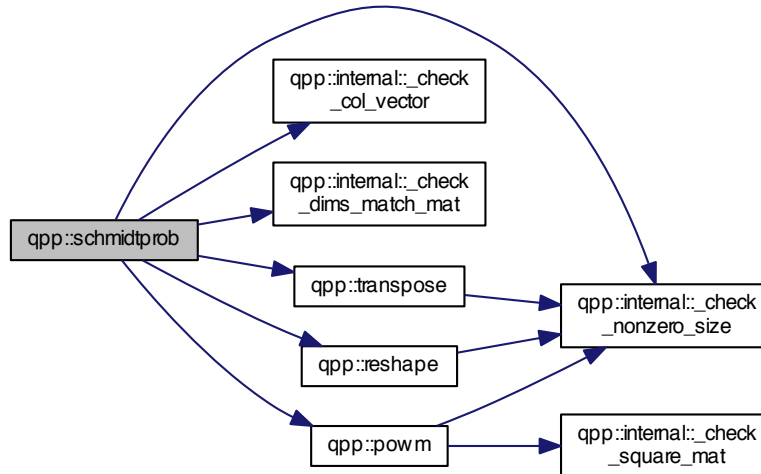
5.1.1.90 `template<typename Derived > types::cmat qpp::schmidtcoeff (const Eigen::MatrixBase< Derived > & A, const std::vector< std::size_t > & dims)`

Here is the call graph for this function:



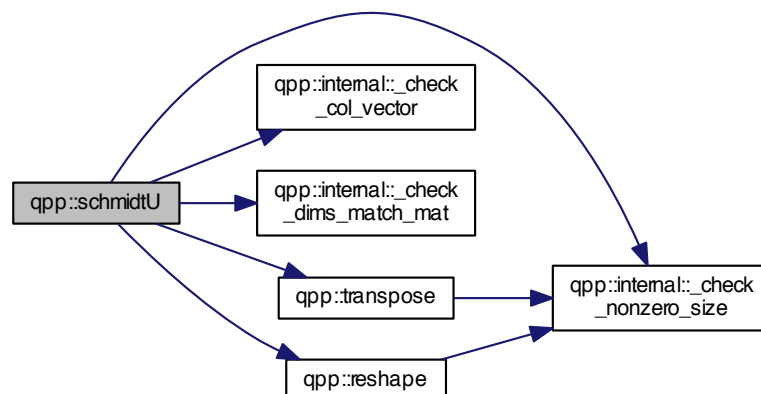
5.1.1.91 `template<typename Derived> types::cmat qpp::schmidtprob (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



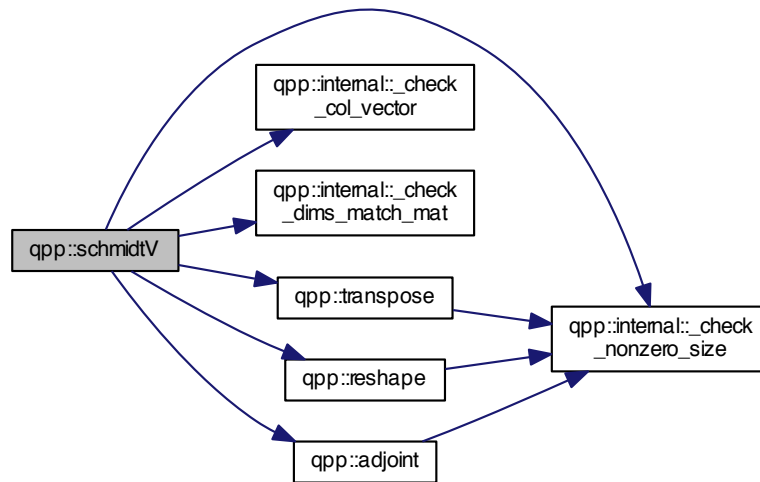
5.1.1.92 `template<typename Derived> types::cmat qpp::schmidtU (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



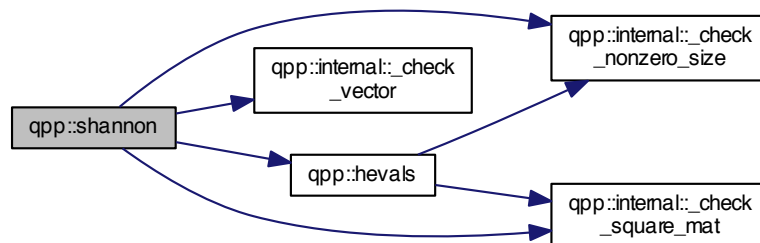
5.1.1.93 `template<typename Derived> types::cmat qpp::schmidtV (const Eigen::MatrixBase< Derived> & A, const std::vector< std::size_t> & dims)`

Here is the call graph for this function:



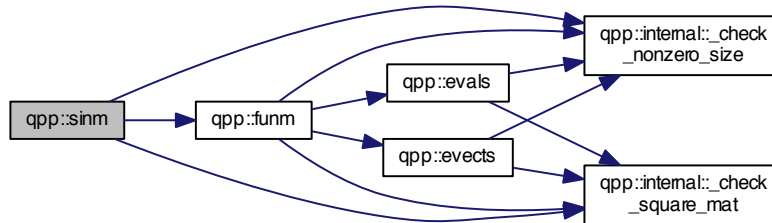
5.1.1.94 `template<typename Derived> double qpp::shannon (const Eigen::MatrixBase< Derived> & A)`

Here is the call graph for this function:



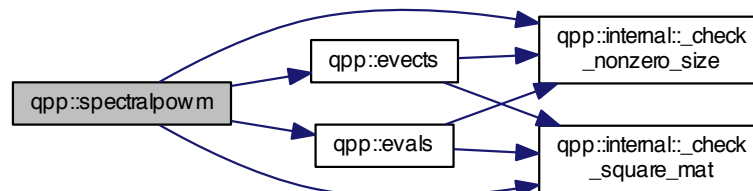
5.1.1.95 `template<typename Derived> types::cmat qpp::sinm (const Eigen::MatrixBase< Derived> & A)`

Here is the call graph for this function:



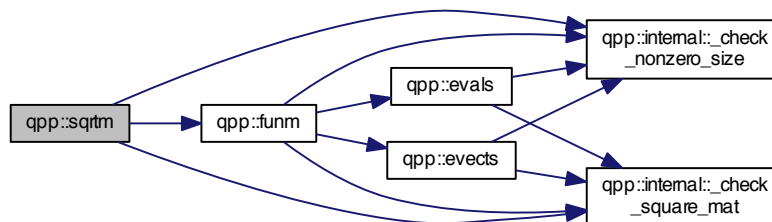
5.1.1.96 `template<typename Derived> types::cmat qpp::spectralpowm (const Eigen::MatrixBase< Derived> & A, const types::cplx z)`

Here is the call graph for this function:



5.1.1.97 `template<typename Derived> types::cmat qpp::sqrtm (const Eigen::MatrixBase< Derived> & A)`

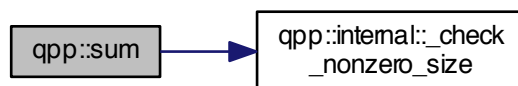
Here is the call graph for this function:



5.1.1.98 `qpp::States () [private]`

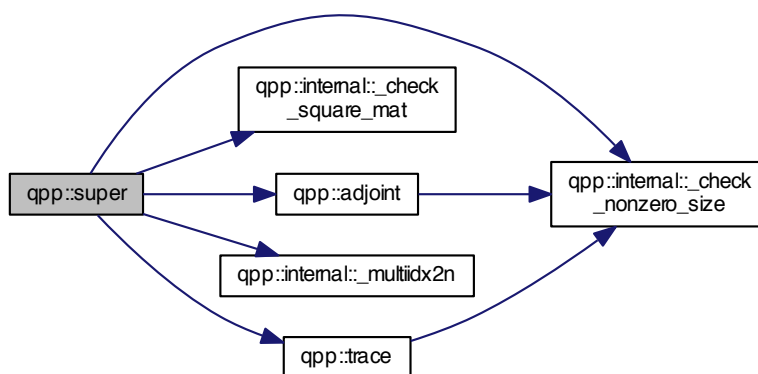
5.1.1.99 `template<typename Derived> Derived::Scalar qpp::sum (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



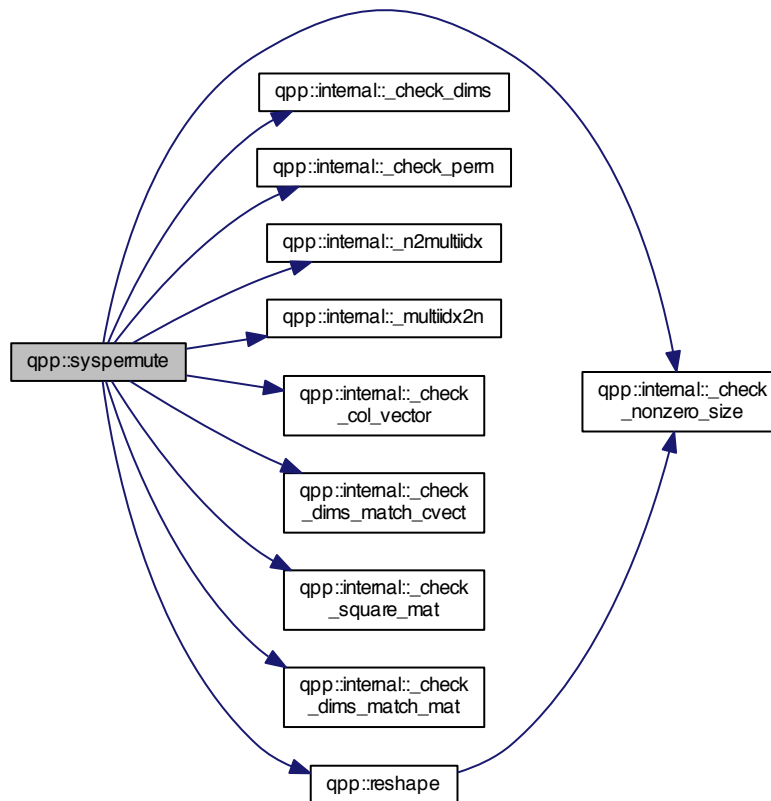
5.1.1.100 `types::cmat qpp::super (const std::vector< types::cmat > & Ks)`

Here is the call graph for this function:



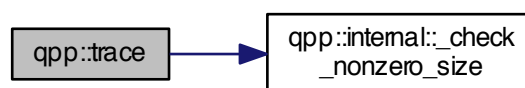
5.1.1.101 `template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::syspermute (const Eigen::MatrixBase< Derived > & A, const std::vector< std::size_t > & perm, const std::vector< std::size_t > & dims)`

Here is the call graph for this function:



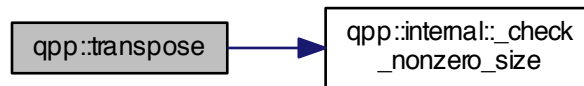
5.1.1.102 `template<typename Derived > Derived::Scalar qpp::trace (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



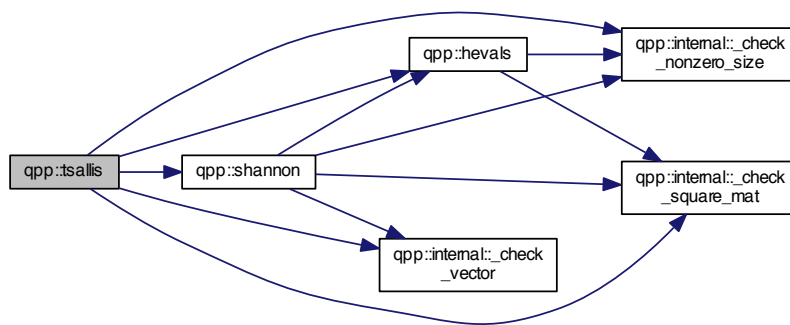
5.1.1.103 `template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::transpose (const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



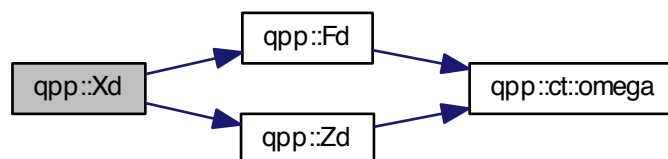
5.1.1.104 `template<typename Derived > double qpp::tsallis (const double alpha, const Eigen::MatrixBase< Derived > & A)`

Here is the call graph for this function:



5.1.1.105 `types::cmat qpp::Xd (std::size_t D) const`

Here is the call graph for this function:



5.1.1.106 `types::cmat qpp::Zd (std::size_t D) const`

Here is the call graph for this function:



5.1.2 Variable Documentation

5.1.2.1 `CLASS_CONST_SINGLETON (RandomDevices) public std::mt19937 qpp::_rng [mutable]`

5.1.2.2 `types::ket qpp::b00`

5.1.2.3 `types::ket qpp::b01`

5.1.2.4 `types::ket qpp::b10`

5.1.2.5 `types::ket qpp::b11`

5.1.2.6 `types::cmat qpp::CNOTab`

5.1.2.7 `types::cmat qpp::CNOTba`

5.1.2.8 `types::cmat qpp::CZ`

5.1.2.9 `types::cmat qpp::FRED`

5.1.2.10 `types::ket qpp::GHZ`

5.1.2.11 `const Gates& qpp::gt = Gates::get_instance()`

5.1.2.12 `CLASS_CONST_SINGLETON (Gates) public types::cmat qpp::H`

5.1.2.13 `types::cmat qpp::pb00`

5.1.2.14 `types::cmat qpp::pb01`

5.1.2.15 `types::cmat qpp::pb10`

5.1.2.16 `types::cmat qpp::pb11`

5.1.2.17 `types::cmat qpp::pGHZ`

5.1.2.18 `types::cmat qpp::pW`

5.1.2.19 `types::cmat qpp::px0`

5.1.2.20 `types::cmat qpp::px1`

- 5.1.2.21 `types::cmat qpp::py0`
- 5.1.2.22 `types::cmat qpp::py1`
- 5.1.2.23 `types::cmat qpp::pz0`
- 5.1.2.24 `types::cmat qpp::pz1`
- 5.1.2.25 `const RandomDevices& qpp::rdevs = RandomDevices::get_instance()`
- 5.1.2.26 `types::cmat qpp::S`
- 5.1.2.27 `const States& qpp::st = States::get_instance()`
- 5.1.2.28 `types::cmat qpp::SWAP`
- 5.1.2.29 `types::cmat qpp::T`
- 5.1.2.30 `types::cmat qpp::TOF`
- 5.1.2.31 `types::ket qpp::W`
- 5.1.2.32 `types::cmat qpp::X`
- 5.1.2.33 `CLASS_CONST_SINGLETON (States) public types::ket qpp::x1`
- 5.1.2.34 `types::cmat qpp::Y`
- 5.1.2.35 `types::ket qpp::y0`
- 5.1.2.36 `types::ket qpp::y1`
- 5.1.2.37 `types::cmat qpp::Z`
- 5.1.2.38 `types::ket qpp::z0`
- 5.1.2.39 `types::ket qpp::z1`

5.2 qpp::ct Namespace Reference

Functions

- `std::complex< double > omega (std::size_t D)`

Variables

- `const double chop = 1e-10`
- `const double eps = 1e-12`
- `const std::size_t maxn = 64`
- `const std::complex< double > ii = { 0, 1 }`
- `const double pi = 3.141592653589793238462643383279502884`
- `const double ee = 2.718281828459045235360287471352662497`

5.2.1 Function Documentation

5.2.1.1 `std::complex<double> qpp::ct::omega (std::size_t D)`

5.2.2 Variable Documentation

5.2.2.1 `const double qpp::ct::chop = 1e-10`

5.2.2.2 `const double qpp::ct::ee = 2.718281828459045235360287471352662497`

5.2.2.3 `const double qpp::ct::eps = 1e-12`

5.2.2.4 `const std::complex<double> qpp::ct::ii = { 0, 1 }`

5.2.2.5 `const std::size_t qpp::ct::maxn = 64`

5.2.2.6 `const double qpp::ct::pi = 3.141592653589793238462643383279502884`

5.3 qpp::internal Namespace Reference

Functions

- `void _n2multiidx (std::size_t n, std::size_t numdims, const std::size_t *dims, std::size_t *result)`
- `std::size_t _multiidx2n (const std::size_t *midx, std::size_t numdims, const std::size_t *dims)`
- `template<typename Derived >`
`bool _check_square_mat (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`bool _check_vector (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`bool _check_row_vector (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`bool _check_col_vector (const Eigen::MatrixBase< Derived > &A)`
- `template<typename T >`
`bool _check_nonzero_size (const T &x)`
- `bool _check_dims (const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`bool _check_dims_match_mat (const std::vector< std::size_t > &dims, const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`bool _check_dims_match_cvect (const std::vector< std::size_t > &dims, const Eigen::MatrixBase< Derived > &V)`
- `template<typename Derived >`
`bool _check_dims_match_rvect (const std::vector< std::size_t > &dims, const Eigen::MatrixBase< Derived > &V)`
- `bool _check_eq_dims (const std::vector< std::size_t > &dims, std::size_t dim)`
- `bool _check_subsys_match_dims (const std::vector< std::size_t > &subsys, const std::vector< std::size_t > &dims)`
- `bool _check_perm (const std::vector< std::size_t > &perm)`
- `template<typename Derived1 , typename Derived2 >`
`types::DynMat< typename`
`Derived1::Scalar > _kron2 (const Eigen::MatrixBase< Derived1 > &A, const Eigen::MatrixBase< Derived2`
`> &B)`
- `template<typename T >`
`void variadic_vector_emplace (std::vector< T > &)`
- `template<typename T , typename First , typename... Args>`
`void variadic_vector_emplace (std::vector< T > &v, First &&first, Args &&...args)`

5.3.1 Function Documentation

5.3.1.1 `template<typename Derived > bool qpp::internal::_check_col_vector (const Eigen::MatrixBase< Derived > & A)`

5.3.1.2 `bool qpp::internal::_check_dims (const std::vector< std::size_t > & dims)`

5.3.1.3 `template<typename Derived > bool qpp::internal::_check_dims_match_cvect (const std::vector< std::size_t > & dims, const Eigen::MatrixBase< Derived > & V)`

5.3.1.4 `template<typename Derived > bool qpp::internal::_check_dims_match_mat (const std::vector< std::size_t > & dims, const Eigen::MatrixBase< Derived > & A)`

5.3.1.5 `template<typename Derived > bool qpp::internal::_check_dims_match_rvect (const std::vector< std::size_t > & dims, const Eigen::MatrixBase< Derived > & V)`

5.3.1.6 `bool qpp::internal::_check_eq_dims (const std::vector< std::size_t > & dims, std::size_t dim)`

5.3.1.7 `template<typename T > bool qpp::internal::_check_nonzero_size (const T & x)`

5.3.1.8 `bool qpp::internal::_check_perm (const std::vector< std::size_t > & perm)`

5.3.1.9 `template<typename Derived > bool qpp::internal::_check_row_vector (const Eigen::MatrixBase< Derived > & A)`

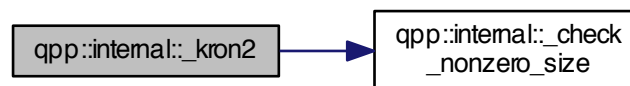
5.3.1.10 `template<typename Derived > bool qpp::internal::_check_square_mat (const Eigen::MatrixBase< Derived > & A)`

5.3.1.11 `bool qpp::internal::_check_subsys_match_dims (const std::vector< std::size_t > & subsys, const std::vector< std::size_t > & dims)`

5.3.1.12 `template<typename Derived > bool qpp::internal::_check_vector (const Eigen::MatrixBase< Derived > & A)`

5.3.1.13 `template<typename Derived1, typename Derived2 > types::DynMat<typename Derived1::Scalar> qpp::internal::_kron2 (const Eigen::MatrixBase< Derived1 > & A, const Eigen::MatrixBase< Derived2 > & B)`

Here is the call graph for this function:



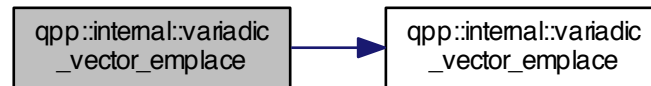
5.3.1.14 `std::size_t qpp::internal::_multiidx2n (const std::size_t * midx, std::size_t numdims, const std::size_t * dims)`

5.3.1.15 `void qpp::internal::_n2multiidx (std::size_t n, std::size_t numdims, const std::size_t * dims, std::size_t * result)`

5.3.1.16 `template<typename T > void qpp::internal::variadic_vector_emplace (std::vector< T > &)`

5.3.1.17 `template<typename T , typename First , typename... Args> void qpp::internal::variadic_vector_emplace (std::vector< T > & v, First && first, Args &&... args)`

Here is the call graph for this function:



5.4 qpp::types Namespace Reference

Typedefs

- using `cplx` = `std::complex< double >`
- using `cmat` = `Eigen::MatrixXcd`
- using `dmat` = `Eigen::MatrixXd`
- using `ket` = `Eigen::Matrix< cplx, Eigen::Dynamic, 1 >`
- using `bra` = `Eigen::Matrix< cplx, 1, Eigen::Dynamic >`
- template<typename Scalar > using `DynMat` = `Eigen::Matrix< Scalar, Eigen::Dynamic, Eigen::Dynamic >`

5.4.1 Typedef Documentation

5.4.1.1 using `qpp::types::bra` = `typedef Eigen::Matrix<cplx, 1, Eigen::Dynamic>`

5.4.1.2 using `qpp::types::cmat` = `typedef Eigen::MatrixXcd`

5.4.1.3 using `qpp::types::cplx` = `typedef std::complex<double>`

5.4.1.4 using `qpp::types::dmat` = `typedef Eigen::MatrixXd`

5.4.1.5 template<typename Scalar > using `qpp::types::DynMat` = `typedef Eigen::Matrix<Scalar, Eigen::Dynamic, Eigen::Dynamic>`

5.4.1.6 using `qpp::types::ket` = `typedef Eigen::Matrix<cplx, Eigen::Dynamic, 1>`

Chapter 6

Class Documentation

6.1 qpp::DiscreteDistribution Class Reference

```
#include <stat.h>
```

Public Member Functions

- `template<typename InputIterator >`
`DiscreteDistribution` (`InputIterator first`, `InputIterator last`)
- `DiscreteDistribution` (`std::initializer_list< double > weights`)
- `DiscreteDistribution` (`std::vector< double > weights`)
- `std::size_t sample` ()
- `std::vector< double > probabilities` () `const`

Protected Attributes

- `std::discrete_distribution`
`< std::size_t > _d`

6.1.1 Constructor & Destructor Documentation

6.1.1.1 `template<typename InputIterator > qpp::DiscreteDistribution::DiscreteDistribution (InputIterator first, InputIterator last)` `[inline]`

6.1.1.2 `qpp::DiscreteDistribution::DiscreteDistribution (std::initializer_list< double > weights)` `[inline]`

6.1.1.3 `qpp::DiscreteDistribution::DiscreteDistribution (std::vector< double > weights)` `[inline]`

6.1.2 Member Function Documentation

6.1.2.1 `std::vector<double> qpp::DiscreteDistribution::probabilities () const` `[inline]`

6.1.2.2 `std::size_t qpp::DiscreteDistribution::sample ()` `[inline]`

6.1.3 Member Data Documentation

6.1.3.1 `std::discrete_distribution<std::size_t> qpp::DiscreteDistribution::_d` `[protected]`

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

- [include/classes/stat.h](#)

6.2 qpp::DiscreteDistributionAbsSquare Class Reference

```
#include <stat.h>
```

Public Member Functions

- `template<typename InputIterator >`
[DiscreteDistributionAbsSquare](#) (InputIterator first, InputIterator last)
- [DiscreteDistributionAbsSquare](#) (std::initializer_list< [types::cplx](#) > amplitudes)
- [DiscreteDistributionAbsSquare](#) (std::vector< [types::cplx](#) > amplitudes)
- [DiscreteDistributionAbsSquare](#) (const [types::cmat](#) &V)
- `std::size_t` [sample](#) ()
- `std::vector< double >` [probabilities](#) () const

Protected Member Functions

- `template<typename InputIterator >`
`std::vector< double >` [cplx2weights](#) (InputIterator first, InputIterator last) const

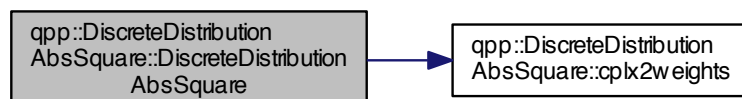
Protected Attributes

- `std::discrete_distribution`
`< std::size_t >` [_d](#)

6.2.1 Constructor & Destructor Documentation

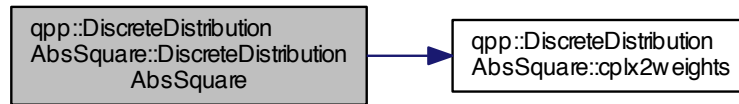
6.2.1.1 `template<typename InputIterator > qpp::DiscreteDistributionAbsSquare::DiscreteDistributionAbsSquare (InputIterator first, InputIterator last)` `[inline]`

Here is the call graph for this function:



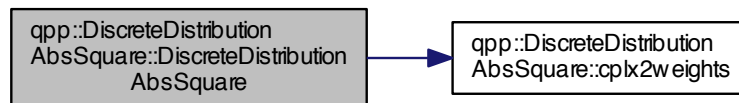
6.2.1.2 `qpp::DiscreteDistributionAbsSquare::DiscreteDistributionAbsSquare (std::initializer_list< types::cplx > amplitudes) [inline]`

Here is the call graph for this function:



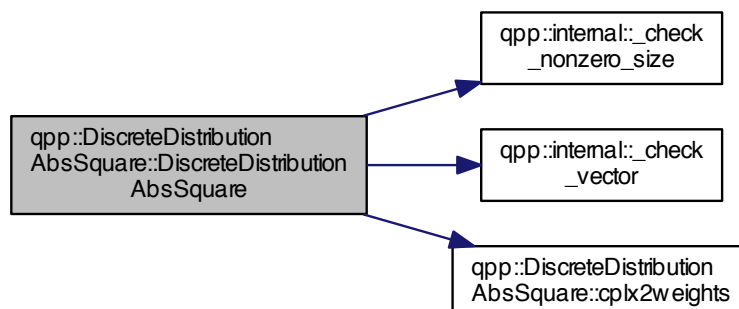
6.2.1.3 `qpp::DiscreteDistributionAbsSquare::DiscreteDistributionAbsSquare (std::vector< types::cplx > amplitudes) [inline]`

Here is the call graph for this function:



6.2.1.4 `qpp::DiscreteDistributionAbsSquare::DiscreteDistributionAbsSquare (const types::cmat & V) [inline]`

Here is the call graph for this function:



6.2.2 Member Function Documentation

6.2.2.1 `template<typename InputIterator > std::vector<double> qpp::DiscreteDistributionAbsSquare::cplx2weights (InputIterator first, InputIterator last) const` `[inline]`, `[protected]`

6.2.2.2 `std::vector<double> qpp::DiscreteDistributionAbsSquare::probabilities () const` `[inline]`

6.2.2.3 `std::size_t qpp::DiscreteDistributionAbsSquare::sample ()` `[inline]`

6.2.3 Member Data Documentation

6.2.3.1 `std::discrete_distribution<std::size_t> qpp::DiscreteDistributionAbsSquare::_d` `[protected]`

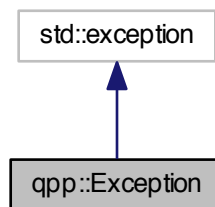
The documentation for this class was generated from the following file:

- `include/classes/stat.h`

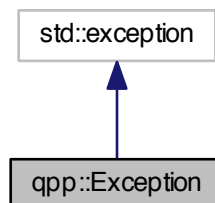
6.3 qpp::Exception Class Reference

```
#include <exception.h>
```

Inheritance diagram for `qpp::Exception`:



Collaboration diagram for `qpp::Exception`:



Public Types

- enum [Type](#) {
[Type::UNKNOWN_EXCEPTION](#) = 1, [Type::ZERO_SIZE](#), [Type::MATRIX_NOT_SQUARE](#), [Type::MATRIX_NOT_CVECTOR](#),
[Type::MATRIX_NOT_RVECTOR](#), [Type::MATRIX_NOT_VECTOR](#), [Type::MATRIX_NOT_SQUARE_OR_CVECTOR](#), [Type::MATRIX_NOT_SQUARE_OR_RVECTOR](#),
[Type::MATRIX_NOT_SQUARE_OR_VECTOR](#), [Type::DIMS_INVALID](#), [Type::DIMS_NOT_EQUAL](#), [Type::DIMS_MISMATCH_MATRIX](#),
[Type::DIMS_MISMATCH_CVECTOR](#), [Type::DIMS_MISMATCH_RVECTOR](#), [Type::DIMS_MISMATCH_VECTOR](#), [Type::SUBSYS_MISMATCH_DIMS](#),
[Type::PERM_INVALID](#), [Type::NOT_QUBIT_GATE](#), [Type::NOT_QUBIT_SUBSYS](#), [Type::NOT_BIPARTITE](#),
[Type::OUT_OF_RANGE](#), [Type::TYPE_MISMATCH](#), [Type::UNDEFINED_TYPE](#), [Type::CUSTOM_EXCEPTION](#) }

Public Member Functions

- [Exception](#) (const std::string &where, const [Type](#) &type)
- [Exception](#) (const std::string &where, const std::string &custom)
- virtual const char * [what](#) () const noexcept override

Private Member Functions

- std::string [_construct_exception_msg](#) ()

Private Attributes

- std::string [_where](#)
- std::string [_msg](#)
- [Type](#) [_type](#)
- std::string [_custom](#)

6.3.1 Member Enumeration Documentation

6.3.1.1 enum qpp::Exception::Type [strong]

Enumerator

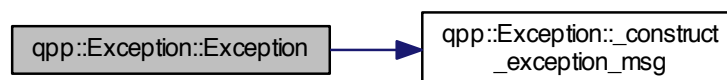
UNKNOWN_EXCEPTION
ZERO_SIZE
MATRIX_NOT_SQUARE
MATRIX_NOT_CVECTOR
MATRIX_NOT_RVECTOR
MATRIX_NOT_VECTOR
MATRIX_NOT_SQUARE_OR_CVECTOR
MATRIX_NOT_SQUARE_OR_RVECTOR
MATRIX_NOT_SQUARE_OR_VECTOR
DIMS_INVALID
DIMS_NOT_EQUAL
DIMS_MISMATCH_MATRIX
DIMS_MISMATCH_CVECTOR

DIMS_MISMATCH_RVECTOR
DIMS_MISMATCH_VECTOR
SUBSYS_MISMATCH_DIMS
PERM_INVALID
NOT_QUBIT_GATE
NOT_QUBIT_SUBSYS
NOT_BIPARTITE
OUT_OF_RANGE
TYPE_MISMATCH
UNDEFINED_TYPE
CUSTOM_EXCEPTION

6.3.2 Constructor & Destructor Documentation

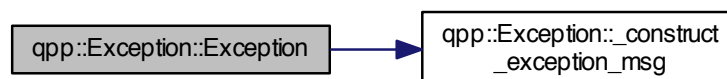
6.3.2.1 `qpp::Exception::Exception (const std::string & where, const Type & type)` `[inline]`

Here is the call graph for this function:



6.3.2.2 `qpp::Exception::Exception (const std::string & where, const std::string & custom)` `[inline]`

Here is the call graph for this function:



6.3.3 Member Function Documentation

6.3.3.1 `std::string qpp::Exception::_construct_exception_msg ()` `[inline]`, `[private]`

6.3.3.2 `virtual const char* qpp::Exception::what () const` `[inline]`, `[override]`, `[virtual]`, `[noexcept]`

6.3.4 Member Data Documentation

6.3.4.1 `std::string qpp::Exception::_custom` [private]

6.3.4.2 `std::string qpp::Exception::_msg` [private]

6.3.4.3 `Type qpp::Exception::_type` [private]

6.3.4.4 `std::string qpp::Exception::_where` [private]

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

- [include/classes/exception.h](#)

6.4 qpp::NormalDistribution Class Reference

```
#include <stat.h>
```

Public Member Functions

- [NormalDistribution](#) (double mean=0, double sigma=1)
- double [sample](#) ()

Protected Attributes

- `std::normal_distribution _d`

6.4.1 Constructor & Destructor Documentation

6.4.1.1 `qpp::NormalDistribution::NormalDistribution (double mean = 0, double sigma = 1)` [inline]

6.4.2 Member Function Documentation

6.4.2.1 `double qpp::NormalDistribution::sample ()` [inline]

6.4.3 Member Data Documentation

6.4.3.1 `std::normal_distribution qpp::NormalDistribution::_d` [protected]

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

- [include/classes/stat.h](#)

6.5 qpp::Qudit Class Reference

```
#include <qudit.h>
```

Public Member Functions

- [Qudit](#) (const [types::cmat](#) &rho=States::get_instance().pz0)
- `std::size_t measure` (const [types::cmat](#) &U, bool destructive=false)
- `std::size_t measure` (bool destructive=false)
- [types::cmat getRho](#) () const
- `std::size_t getD` () const

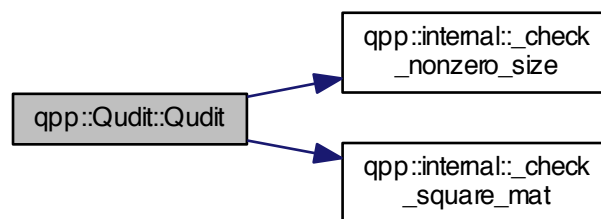
Private Attributes

- [types::cmat_rho](#)
- [std::size_t_D](#)

6.5.1 Constructor & Destructor Documentation

6.5.1.1 `qpp::Qudit::Qudit (const types::cmat & rho = States::get_instance() . pz0) [inline]`

Here is the call graph for this function:



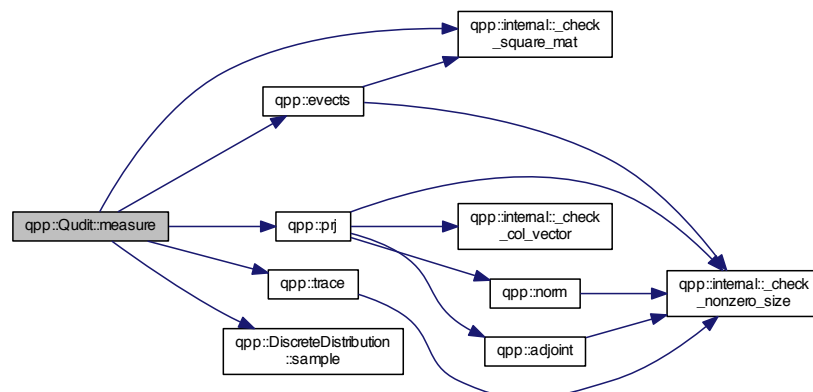
6.5.2 Member Function Documentation

6.5.2.1 `std::size_t qpp::Qudit::getD () const [inline]`

6.5.2.2 `types::cmat qpp::Qudit::getRho () const [inline]`

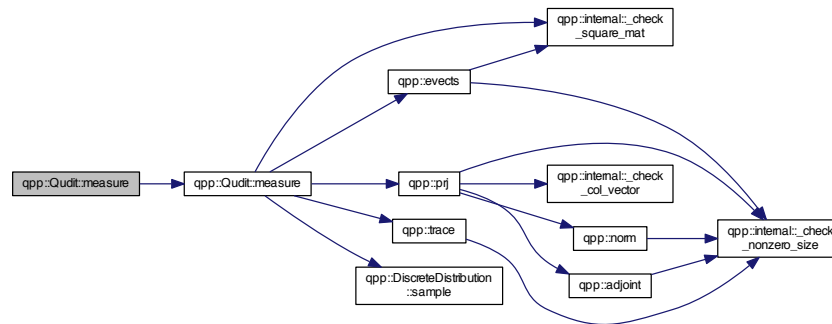
6.5.2.3 `std::size_t qpp::Qudit::measure (const types::cmat & U, bool destructive = false) [inline]`

Here is the call graph for this function:



6.5.2.4 `std::size_t qpp::Qudit::measure (bool destructive = false) [inline]`

Here is the call graph for this function:



6.5.3 Member Data Documentation

6.5.3.1 `std::size_t qpp::Qudit::_D [private]`

6.5.3.2 `types::cmat qpp::Qudit::_rho [private]`

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

- [include/classes/qudit.h](#)

6.6 qpp::Singleton< T > Class Template Reference

```
#include <singleton.h>
```

Static Public Member Functions

- static [T](#) & [get_instance](#) ()

Protected Member Functions

- [Singleton](#) ()=default
- virtual [~Singleton](#) ()=default
- [Singleton](#) (const [Singleton](#) &)=delete
- [Singleton](#) & [operator=](#) (const [Singleton](#) &)=delete

6.6.1 Constructor & Destructor Documentation

6.6.1.1 `template<typename T> qpp::Singleton< T >::Singleton () [protected], [default]`

6.6.1.2 `template<typename T> virtual qpp::Singleton< T >::~~Singleton () [protected], [virtual], [default]`

6.6.1.3 `template<typename T> qpp::Singleton< T>::Singleton (const Singleton< T> &) [protected], [delete]`

6.6.2 Member Function Documentation

6.6.2.1 `template<typename T> static T& qpp::Singleton< T>::get_instance () [inline], [static]`

6.6.2.2 `template<typename T> Singleton& qpp::Singleton< T>::operator= (const Singleton< T> &) [protected], [delete]`

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

- `include/classes/singleton.h`

6.7 qpp::Timer Class Reference

```
#include <timer.h>
```

Public Member Functions

- `Timer ()`
- `void tic ()`
- `void toc ()`
- `double seconds () const`

Protected Attributes

- `std::chrono::steady_clock::time_point _start`
- `std::chrono::steady_clock::time_point _end`

Friends

- `std::ostream & operator<< (std::ostream &os, const Timer &rhs)`

6.7.1 Constructor & Destructor Documentation

6.7.1.1 `qpp::Timer::Timer () [inline]`

6.7.2 Member Function Documentation

6.7.2.1 `double qpp::Timer::seconds () const [inline]`

6.7.2.2 `void qpp::Timer::tic () [inline]`

6.7.2.3 `void qpp::Timer::toc () [inline]`

6.7.3 Friends And Related Function Documentation

6.7.3.1 `std::ostream& operator<< (std::ostream & os, const Timer & rhs) [friend]`

6.7.4 Member Data Documentation

6.7.4.1 `std::chrono::steady_clock::time_point qpp::Timer::_end` [protected]

6.7.4.2 `std::chrono::steady_clock::time_point qpp::Timer::_start` [protected]

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

- [include/classes/timer.h](#)

6.8 qpp::UniformRealDistribution Class Reference

```
#include <stat.h>
```

Public Member Functions

- [UniformRealDistribution](#) (double *a*=0, double *b*=1)
- double [sample](#) ()

Protected Attributes

- `std::uniform_real_distribution _d`

6.8.1 Constructor & Destructor Documentation

6.8.1.1 `qpp::UniformRealDistribution::UniformRealDistribution (double a = 0, double b = 1)` [inline]

6.8.2 Member Function Documentation

6.8.2.1 `double qpp::UniformRealDistribution::sample ()` [inline]

6.8.3 Member Data Documentation

6.8.3.1 `std::uniform_real_distribution qpp::UniformRealDistribution::_d` [protected]

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

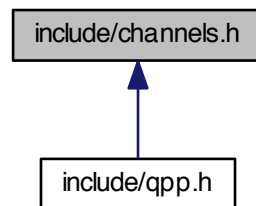
- [include/classes/stat.h](#)

Chapter 7

File Documentation

7.1 include/channels.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

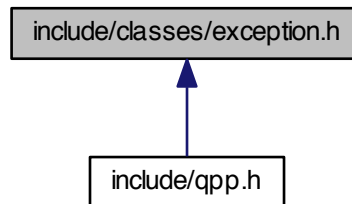
- [qpp](#)

Functions

- `types::cmat qpp::super (const std::vector< types::cmat > &Ks)`
- `types::cmat qpp::choi (const std::vector< types::cmat > &Ks)`
- `std::vector< types::cmat > qpp::choi2kraus (const types::cmat &A)`
- `template<typename Derived >
types::cmat qpp::channel (const Eigen::MatrixBase< Derived > &rho, const std::vector< types::cmat > &Ks)`
- `template<typename Derived >
types::cmat qpp::channel (const Eigen::MatrixBase< Derived > &rho, const std::vector< types::cmat > &Ks,
const std::vector< std::size_t > &subsys, const std::vector< std::size_t > &dims)`

7.2 include/classes/exception.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

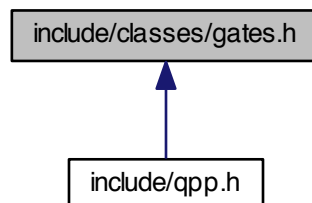
- class [qpp::Exception](#)

Namespaces

- [qpp](#)

7.3 include/classes/gates.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- [qpp](#)

Functions

- [qpp::Gates](#) ()

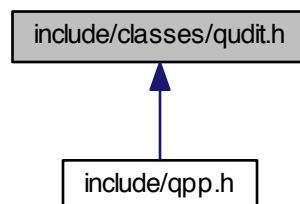
- `types::cmat qpp::Rn` (double theta, `std::vector< double > n`) const
- `types::cmat qpp::Zd` (`std::size_t D`) const
- `types::cmat qpp::Fd` (`std::size_t D`) const
- `types::cmat qpp::Xd` (`std::size_t D`) const
- `template<typename Derived = Eigen::MatrixXcd>`
`Derived qpp::ld` (`std::size_t D`) const
- `template<typename Derived1 , typename Derived2 >`
`types::DynMat< typename`
`Derived1::Scalar > qpp::apply` (const `Eigen::MatrixBase< Derived1 > &state`, const `Eigen::MatrixBase<`
`Derived2 > &A`, const `std::vector< std::size_t > &subsys`, const `std::vector< std::size_t > &dims`) const
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > qpp::CTRL` (const `Eigen::MatrixBase< Derived > &A`, const `std::vector< std::size_t >`
`&ctrl`, const `std::vector< std::size_t > &subsys`, `std::size_t n`, `std::size_t d=2`) const

Variables

- `CLASS_CONST_SINGLETON`(Gates)
`public types::cmat qpp::H`
- `types::cmat qpp::X`
- `types::cmat qpp::Y`
- `types::cmat qpp::Z`
- `types::cmat qpp::S`
- `types::cmat qpp::T`
- `types::cmat qpp::CNOTab`
- `types::cmat qpp::CZ`
- `types::cmat qpp::CNOTba`
- `types::cmat qpp::SWAP`
- `types::cmat qpp::TOF`
- `types::cmat qpp::FRED`

7.4 include/classes/qudit.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

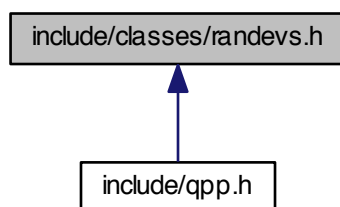
- class `qpp::Qudit`

Namespaces

- [qpp](#)

7.5 include/classes/randevs.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- [qpp](#)

Functions

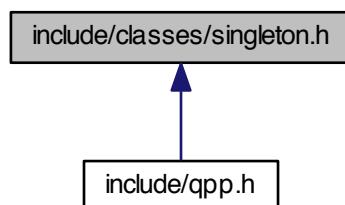
- [qpp::RandomDevices](#) ()

Variables

- [CLASS_CONST_SINGLETON](#)(RandomDevices)
public std::mt19937 [qpp::_rng](#)

7.6 include/classes/singleton.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

- class [qpp::Singleton< T >](#)

Namespaces

- [qpp](#)

Macros

- #define [CLASS_SINGLETON\(Foo\)](#)
- #define [CLASS_CONST_SINGLETON\(Foo\)](#)

7.6.1 Macro Definition Documentation

7.6.1.1 #define CLASS_CONST_SINGLETON(Foo)

Value:

```
class Foo: public Singleton<const Foo>\n{\n    friend class Singleton<const Foo>;
```

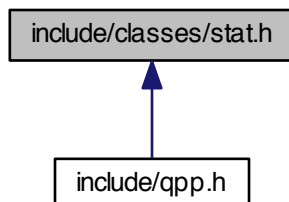
7.6.1.2 #define CLASS_SINGLETON(Foo)

Value:

```
class Foo: public Singleton<Foo>\n{\n    friend class Singleton<Foo>;
```

7.7 include/classes/stat.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

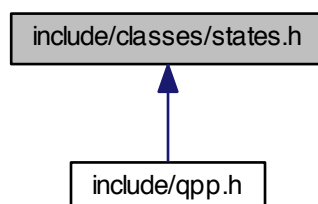
- class [qpp::NormalDistribution](#)
- class [qpp::UniformRealDistribution](#)
- class [qpp::DiscreteDistribution](#)
- class [qpp::DiscreteDistributionAbsSquare](#)

Namespaces

- [qpp](#)

7.8 include/classes/states.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- [qpp](#)

Functions

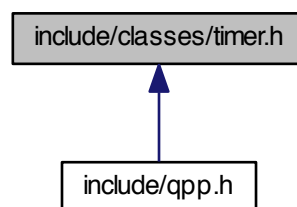
- [qpp::States](#) ()

Variables

- [CLASS_CONST_SINGLETON](#)(States)
public types::ket [qpp::x1](#)
- types::ket [qpp::y0](#)
- types::ket [qpp::y1](#)
- types::ket [qpp::z0](#)
- types::ket [qpp::z1](#)
- types::cmat [qpp::px0](#)
- types::cmat [qpp::px1](#)
- types::cmat [qpp::py0](#)
- types::cmat [qpp::py1](#)
- types::cmat [qpp::pz0](#)
- types::cmat [qpp::pz1](#)
- types::ket [qpp::b00](#)
- types::ket [qpp::b01](#)
- types::ket [qpp::b10](#)
- types::ket [qpp::b11](#)
- types::cmat [qpp::pb00](#)
- types::cmat [qpp::pb01](#)
- types::cmat [qpp::pb10](#)
- types::cmat [qpp::pb11](#)
- types::ket [qpp::GHZ](#)
- types::ket [qpp::W](#)
- types::cmat [qpp::pGHZ](#)
- types::cmat [qpp::pW](#)

7.9 include/classes/timer.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

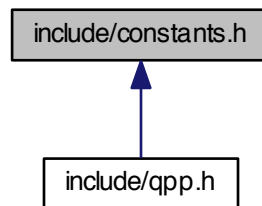
- class [qpp::Timer](#)

Namespaces

- [qpp](#)

7.10 include/constants.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- [qpp](#)
- [qpp::ct](#)

Functions

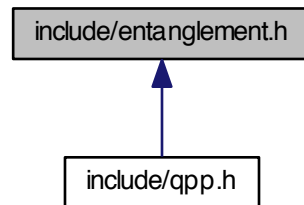
- `std::complex< double > qpp::ct::omega (std::size_t D)`

Variables

- `const double qpp::ct::chop = 1e-10`
- `const double qpp::ct::eps = 1e-12`
- `const std::size_t qpp::ct::maxn = 64`
- `const std::complex< double > qpp::ct::ii = { 0, 1 }`
- `const double qpp::ct::pi = 3.141592653589793238462643383279502884`
- `const double qpp::ct::ee = 2.718281828459045235360287471352662497`

7.11 include/entanglement.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

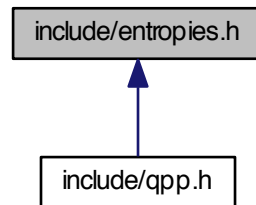
- [qpp](#)

Functions

- `template<typename Derived >`
`types::cmat qpp::schmidtcoeff (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`types::cmat qpp::schmidtU (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`types::cmat qpp::schmidtV (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`types::cmat qpp::schmidtprob (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`double qpp::entanglement (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`double qpp::gconcurrency (const Eigen::MatrixBase< Derived > &A)`

7.12 include/entropies.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

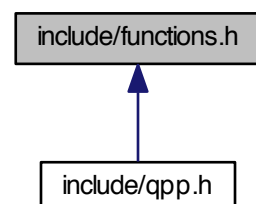
- [qpp](#)

Functions

- `template<typename Derived >`
`double qpp::shannon (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`double qpp::renyi (const double alpha, const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`double qpp::renyi_inf (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`double qpp::tsallis (const double alpha, const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`double qpp::qmutualinfo (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &subsys, const std::vector< std::size_t > &dims)`

7.13 include/functions.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- [qpp](#)

Functions

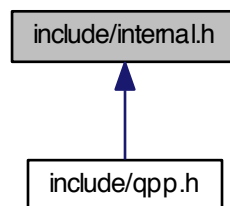
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > qpp::transpose (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > qpp::conjugate (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > qpp::adjoint (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > qpp::inverse (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`Derived::Scalar qpp::trace (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`Derived::Scalar qpp::det (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`Derived::Scalar qpp::logdet (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`Derived::Scalar qpp::sum (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`double qpp::norm (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::evals (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::evecs (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::dmat qpp::hevals (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::hevecs (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::funm (const Eigen::MatrixBase< Derived > &A, types::cplx(*f)(const types::cplx &))`
- `template<typename Derived >`
`types::cmat qpp::sqrtm (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::absm (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::expm (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::logm (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::sinm (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::cosm (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`types::cmat qpp::spectralpwm (const Eigen::MatrixBase< Derived > &A, const types::cplx z)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > qpp::pwm (const Eigen::MatrixBase< Derived > &A, std::size_t n)`

- `template<typename OutputScalar , typename Derived >`
`types::DynMat< OutputScalar > qpp::cwise (const Eigen::MatrixBase< Derived > &A, OutputScalar(*)f)(const typename Derived::Scalar &))`
- `template<typename T >`
`types::DynMat< typename T::Scalar > qpp::kron (const T &head)`
- `template<typename T , typename... Args>`
`types::DynMat< typename T::Scalar > qpp::kron (const T &head, const Args &...tail)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::kron (const std::vector< Derived > &As)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::kron (const std::initializer_list< Derived > &As)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::kronpow (const Eigen::MatrixBase< Derived > &A, std::size_t n)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::reshape (const Eigen::MatrixBase< Derived > &A, std::size_t rows, std::size_t cols)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::syspermute (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &perm, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::ptrace1 (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::ptrace2 (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::ptrace (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &subsys, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::ptranspose (const Eigen::MatrixBase< Derived > &A, const std::vector< std::size_t > &subsys, const std::vector< std::size_t > &dims)`
- `template<typename Derived1 , typename Derived2 >`
`types::DynMat< typename Derived1::Scalar > qpp::comm (const Eigen::MatrixBase< Derived1 > &A, const Eigen::MatrixBase< Derived2 > &B)`
- `template<typename Derived1 , typename Derived2 >`
`types::DynMat< typename Derived1::Scalar > qpp::anticomm (const Eigen::MatrixBase< Derived1 > &A, const Eigen::MatrixBase< Derived2 > &B)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::prj (const Eigen::MatrixBase< Derived > &V)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::expandout (const Eigen::MatrixBase< Derived > &A, std::size_t pos, const std::vector< std::size_t > &dims)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::grams (const std::vector< Derived > &Vs)`

- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > qpp::grams (const std::initializer_list< Derived > &Vs)`
- `template<typename Derived >`
`types::DynMat< typename`
`Derived::Scalar > qpp::grams (const Eigen::MatrixBase< Derived > &A)`
- `std::vector< std::size_t > qpp::n2multiidx (std::size_t n, const std::vector< std::size_t > &dims)`
- `std::size_t qpp::multiidx2n (const std::vector< std::size_t > &midx, const std::vector< std::size_t > &dims)`
- `types::ket qpp::mket (const std::vector< std::size_t > &mask)`
- `types::ket qpp::mket (const std::vector< std::size_t > &mask, const std::vector< std::size_t > &dims)`
- `types::ket qpp::mket (const std::vector< std::size_t > &mask, std::size_t d)`
- `std::vector< std::size_t > qpp::invperm (const std::vector< std::size_t > &perm)`
- `std::vector< std::size_t > qpp::compperm (const std::vector< std::size_t > &perm, const std::vector< std::size_t > &sigma)`

7.14 include/internal.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- [qpp](#)
- [qpp::internal](#)

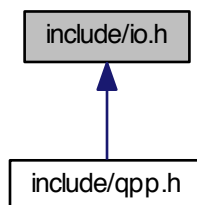
Functions

- `void qpp::internal::_n2multiidx (std::size_t n, std::size_t numdims, const std::size_t *dims, std::size_t *result)`
- `std::size_t qpp::internal::_multiidx2n (const std::size_t *midx, std::size_t numdims, const std::size_t *dims)`
- `template<typename Derived >`
`bool qpp::internal::_check_square_mat (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`bool qpp::internal::_check_vector (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`bool qpp::internal::_check_row_vector (const Eigen::MatrixBase< Derived > &A)`
- `template<typename Derived >`
`bool qpp::internal::_check_col_vector (const Eigen::MatrixBase< Derived > &A)`
- `template<typename T >`
`bool qpp::internal::_check_nonzero_size (const T &x)`
- `bool qpp::internal::_check_dims (const std::vector< std::size_t > &dims)`

- `template<typename Derived >`
`bool qpp::internal::_check_dims_match_mat (const std::vector< std::size_t > &dims, const Eigen::Matrix<↵`
`Base< Derived > &A)`
- `template<typename Derived >`
`bool qpp::internal::_check_dims_match_cvect (const std::vector< std::size_t > &dims, const Eigen::Matrix<↵`
`Base< Derived > &V)`
- `template<typename Derived >`
`bool qpp::internal::_check_dims_match_rvect (const std::vector< std::size_t > &dims, const Eigen::Matrix<↵`
`Base< Derived > &V)`
- `bool qpp::internal::_check_eq_dims (const std::vector< std::size_t > &dims, std::size_t dim)`
- `bool qpp::internal::_check_subsys_match_dims (const std::vector< std::size_t > &subsys, const std::↵`
`::vector< std::size_t > &dims)`
- `bool qpp::internal::_check_perm (const std::vector< std::size_t > &perm)`
- `template<typename Derived1 , typename Derived2 >`
`types::DynMat< typename`
`Derived1::Scalar > qpp::internal::_kron2 (const Eigen::MatrixBase< Derived1 > &A, const Eigen::Matrix<↵`
`Base< Derived2 > &B)`
- `template<typename T >`
`void qpp::internal::variadic_vector_emplace (std::vector< T > &)`
- `template<typename T , typename First , typename... Args>`
`void qpp::internal::variadic_vector_emplace (std::vector< T > &v, First &&first, Args &&...args)`

7.15 include/io.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- [qpp](#)

Functions

- `template<typename T >`
`void qpp::disp (const T &x, const std::string &separator, const std::string &start="[" , const std::string &end="]",`
`std::ostream &os=std::cout)`
- `template<typename T >`
`void qpp::displn (const T &x, const std::string &separator, const std::string &start="[" , const std::string`
`&end="]", std::ostream &os=std::cout)`

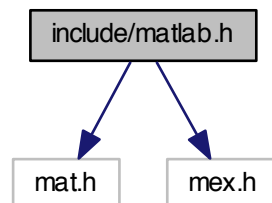
- `template<typename T >`
`void qpp::disp (const T *x, const std::size_t n, const std::string &separator, const std::string &start="[" , const std::string &end="]", std::ostream &os=std::cout)`
- `template<typename T >`
`void qpp::displn (const T *x, const std::size_t n, const std::string &separator, const std::string &start="[" , const std::string &end="]", std::ostream &os=std::cout)`
- `template<typename Derived >`
`void qpp::disp (const Eigen::MatrixBase< Derived > &A, double chop=ct::chop, std::ostream &os=std::cout)`
- `template<typename Derived >`
`void qpp::displn (const Eigen::MatrixBase< Derived > &A, double chop=ct::chop, std::ostream &os=std::cout)`
- `void qpp::disp (const types::cplx c, double chop=ct::chop, std::ostream &os=std::cout)`
- `void qpp::displn (const types::cplx c, double chop=ct::chop, std::ostream &os=std::cout)`
- `template<typename Derived >`
`void qpp::save (const Eigen::MatrixBase< Derived > &A, const std::string &fname)`
- `template<typename Derived >`
`types::DynMat< typename Derived::Scalar > qpp::load (const std::string &fname)`

7.16 include/matlab.h File Reference

```
#include "mat.h"
```

```
#include "mex.h"
```

Include dependency graph for matlab.h:



Namespaces

- [qpp](#)

Functions

- `template<typename Derived >`
`Derived qpp::loadMATLABmatrix (const std::string &mat_file, const std::string &var_name)`
- `template<>`
`types::dmat qpp::loadMATLABmatrix (const std::string &mat_file, const std::string &var_name)`
- `template<>`
`types::cmat qpp::loadMATLABmatrix (const std::string &mat_file, const std::string &var_name)`
- `template<typename Derived >`
`void qpp::saveMATLABmatrix (const Eigen::MatrixBase< Derived > &A, const std::string &mat_file, const std::string &var_name, const std::string &mode)`

- `template<>`
`void qpp::saveMATLABmatrix (const Eigen::MatrixBase< typename types::dmat > &A, const std::string &mat_file, const std::string &var_name, const std::string &mode)`
- `template<>`
`void qpp::saveMATLABmatrix (const Eigen::MatrixBase< typename types::cmat > &A, const std::string &mat_file, const std::string &var_name, const std::string &mode)`

7.17 include/qpp.h File Reference

```
#include <algorithm>
#include <chrono>
#include <cmath>
#include <complex>
#include <cstdlib>
#include <cstring>
#include <fstream>
#include <functional>
#include <iomanip>
#include <iostream>
#include <iterator>
#include <numeric>
#include <ostream>
#include <random>
#include <stdexcept>
#include <string>
#include <type_traits>
#include <utility>
#include <vector>
#include <Eigen/Dense>
#include <Eigen/SVD>
#include "constants.h"
#include "types.h"
#include "classes/exception.h"
#include "classes/singleton.h"
#include "classes/states.h"
#include "classes/randevs.h"
#include "internal.h"
#include "functions.h"
#include "classes/gates.h"
#include "classes/stat.h"
#include "entropies.h"
#include "entanglement.h"
#include "channels.h"
#include "io.h"
#include "random.h"
#include "classes/qudit.h"
#include "classes/timer.h"
Include dependency graph for qpp.h:
```



Namespaces

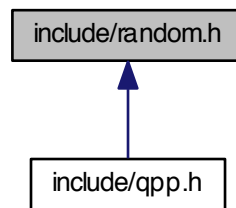
- `qpp`

Variables

- const RandomDevices & [qpp::rdevs](#) = RandomDevices::get_instance()
- const Gates & [qpp::gt](#) = Gates::get_instance()
- const States & [qpp::st](#) = States::get_instance()

7.18 include/random.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

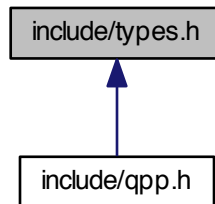
- [qpp](#)

Functions

- template<typename Derived >
Derived [qpp::rand](#) (std::size_t rows, std::size_t cols, double a=0, double b=1)
- template<>
types::dmat [qpp::rand](#) (std::size_t rows, std::size_t cols, double a, double b)
- template<>
types::cmat [qpp::rand](#) (std::size_t rows, std::size_t cols, double a, double b)
- double [qpp::rand](#) (double a=0, double b=1)
- long long [qpp::randint](#) (long long a, long long b)
- template<typename Derived >
Derived [qpp::randn](#) (std::size_t rows, std::size_t cols, double mean=0, double sigma=1)
- template<>
types::dmat [qpp::randn](#) (std::size_t rows, std::size_t cols, double mean, double sigma)
- template<>
types::cmat [qpp::randn](#) (std::size_t rows, std::size_t cols, double mean, double sigma)
- double [qpp::randn](#) (double mean=0, double sigma=1)
- types::cmat [qpp::randU](#) (std::size_t D)
- types::cmat [qpp::randV](#) (std::size_t Din, std::size_t Dout)
- std::vector< types::cmat > [qpp::randkraus](#) (std::size_t n, std::size_t D)
- types::cmat [qpp::randH](#) (std::size_t D)
- types::ket [qpp::randket](#) (std::size_t D)
- types::cmat [qpp::randrho](#) (std::size_t D)
- std::vector< std::size_t > [qpp::randperm](#) (std::size_t n)

7.19 include/types.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- [qpp](#)
- [qpp::types](#)

Typedefs

- using [qpp::types::cplx](#) = `std::complex< double >`
- using [qpp::types::cmat](#) = `Eigen::MatrixXcd`
- using [qpp::types::dmat](#) = `Eigen::MatrixXd`
- using [qpp::types::ket](#) = `Eigen::Matrix< cplx, Eigen::Dynamic, 1 >`
- using [qpp::types::bra](#) = `Eigen::Matrix< cplx, 1, Eigen::Dynamic >`
- template<typename Scalar >
using [qpp::types::DynMat](#) = `Eigen::Matrix< Scalar, Eigen::Dynamic, Eigen::Dynamic >`

Index

absm
 qpp, 14
adjoint
 qpp, 14
anticomm
 qpp, 15
apply
 qpp, 15

b00
 qpp, 52
b01
 qpp, 52
b10
 qpp, 52
b11
 qpp, 52

CUSTOM_EXCEPTION
 qpp::Exception, 62
channel
 qpp, 16, 17
choi
 qpp, 17
choi2kraus
 qpp, 18
comm
 qpp, 18
compperm
 qpp, 19
conjugate
 qpp, 19
cosm
 qpp, 19
cwise
 qpp, 20

DIMS_INVALID
 qpp::Exception, 61
DIMS_MISMATCH_CVECTOR
 qpp::Exception, 61
DIMS_MISMATCH_MATRIX
 qpp::Exception, 61
DIMS_MISMATCH_RVECTOR
 qpp::Exception, 61
DIMS_MISMATCH_VECTOR
 qpp::Exception, 62
DIMS_NOT_EQUAL
 qpp::Exception, 61
det
 qpp, 21
disp
 qpp, 21
displn
 qpp, 21, 22

entanglement
 qpp, 23
evals
 qpp, 23
evects
 qpp, 24
expandout
 qpp, 24
expm
 qpp, 25

Fd
 qpp, 25
funm
 qpp, 26

Gates
 qpp, 26
gconcurrency
 qpp, 26
grams
 qpp, 27, 28
gt
 qpp, 52

H
 qpp, 52
hevals
 qpp, 28
hevects
 qpp, 28

ld
 qpp, 29
inverse
 qpp, 29
invperm
 qpp, 29

kron
 qpp, 29, 30
kronpow
 qpp, 30
load

- qpp, 31
- logdet
 - qpp, 31
- logm
 - qpp, 31
- MATRIX_NOT_CVECTOR
 - qpp::Exception, 61
- MATRIX_NOT_RVECTOR
 - qpp::Exception, 61
- MATRIX_NOT_SQUARE
 - qpp::Exception, 61
- MATRIX_NOT_SQUARE_OR_CVECTOR
 - qpp::Exception, 61
- MATRIX_NOT_SQUARE_OR_RVECTOR
 - qpp::Exception, 61
- MATRIX_NOT_SQUARE_OR_VECTOR
 - qpp::Exception, 61
- MATRIX_NOT_VECTOR
 - qpp::Exception, 61
- mket
 - qpp, 32
- multiidx2n
 - qpp, 33
- n2multiidx
 - qpp, 33
- NOT_BIPARTITE
 - qpp::Exception, 62
- NOT_QUBIT_GATE
 - qpp::Exception, 62
- NOT_QUBIT_SUBSYS
 - qpp::Exception, 62
- norm
 - qpp, 33
- OUT_OF_RANGE
 - qpp::Exception, 62
- PERM_INVALID
 - qpp::Exception, 62
- pb00
 - qpp, 52
- pb01
 - qpp, 52
- pb10
 - qpp, 52
- pb11
 - qpp, 52
- powm
 - qpp, 34
- prj
 - qpp, 34
- ptrace
 - qpp, 35
- ptrace1
 - qpp, 36
- ptrace2
 - qpp, 37
- ptranspose
 - qpp, 37
- px0
 - qpp, 52
- px1
 - qpp, 52
- py0
 - qpp, 52
- py1
 - qpp, 53
- pz0
 - qpp, 53
- pz1
 - qpp, 53
- qmutualinfo
 - qpp, 38
- qpp, 9
 - absm, 14
 - adjoint, 14
 - anticomm, 15
 - apply, 15
 - b00, 52
 - b01, 52
 - b10, 52
 - b11, 52
 - channel, 16, 17
 - choi, 17
 - choi2kraus, 18
 - comm, 18
 - compperm, 19
 - conjugate, 19
 - cosm, 19
 - cwise, 20
 - det, 21
 - disp, 21
 - displn, 21, 22
 - entanglement, 23
 - evals, 23
 - evects, 24
 - expandout, 24
 - expm, 25
 - Fd, 25
 - funm, 26
 - Gates, 26
 - gconcurrency, 26
 - grams, 27, 28
 - gt, 52
 - H, 52
 - hevals, 28
 - hevects, 28
 - Id, 29
 - inverse, 29
 - invperm, 29
 - kron, 29, 30
 - kronpow, 30
 - load, 31
 - logdet, 31
 - logm, 31

- mket, [32](#)
- multiidx2n, [33](#)
- n2multiidx, [33](#)
- norm, [33](#)
- pb00, [52](#)
- pb01, [52](#)
- pb10, [52](#)
- pb11, [52](#)
- powm, [34](#)
- prj, [34](#)
- ptrace, [35](#)
- ptrace1, [36](#)
- ptrace2, [37](#)
- ptranspose, [37](#)
- px0, [52](#)
- px1, [52](#)
- py0, [52](#)
- py1, [53](#)
- pz0, [53](#)
- pz1, [53](#)
- qmutualinfo, [38](#)
- rand, [39](#), [40](#)
- randint, [40](#)
- randket, [41](#)
- randkraus, [41](#)
- randn, [41](#), [42](#)
- randperm, [42](#)
- randrho, [42](#)
- rdevs, [53](#)
- renyi, [43](#)
- reshape, [44](#)
- Rn, [44](#)
- S, [53](#)
- save, [44](#)
- schmidtcoeff, [45](#)
- schmidtprob, [46](#)
- shannon, [47](#)
- sinm, [47](#)
- spectralpowm, [48](#)
- sqrtn, [48](#)
- st, [53](#)
- States, [48](#)
- sum, [49](#)
- super, [49](#)
- syspermute, [49](#)
- T, [53](#)
- trace, [50](#)
- transpose, [50](#)
- tsallis, [51](#)
- W, [53](#)
- X, [53](#)
- x1, [53](#)
- Xd, [51](#)
- Y, [53](#)
- y0, [53](#)
- y1, [53](#)
- Z, [53](#)
- z0, [53](#)
- z1, [53](#)
- Zd, [51](#)
- qpp::Exception
 - CUSTOM_EXCEPTION, [62](#)
 - DIMS_INVALID, [61](#)
 - DIMS_MISMATCH_CVECTOR, [61](#)
 - DIMS_MISMATCH_MATRIX, [61](#)
 - DIMS_MISMATCH_RVECTOR, [61](#)
 - DIMS_MISMATCH_VECTOR, [62](#)
 - DIMS_NOT_EQUAL, [61](#)
 - MATRIX_NOT_CVECTOR, [61](#)
 - MATRIX_NOT_RVECTOR, [61](#)
 - MATRIX_NOT_SQUARE, [61](#)
 - MATRIX_NOT_SQUARE_OR_CVECTOR, [61](#)
 - MATRIX_NOT_SQUARE_OR_RVECTOR, [61](#)
 - MATRIX_NOT_SQUARE_OR_VECTOR, [61](#)
 - MATRIX_NOT_VECTOR, [61](#)
 - NOT_BIPARTITE, [62](#)
 - NOT_QUBIT_GATE, [62](#)
 - NOT_QUBIT_SUBSYS, [62](#)
 - OUT_OF_RANGE, [62](#)
 - PERM_INVALID, [62](#)
 - SUBSYS_MISMATCH_DIMS, [62](#)
 - TYPE_MISMATCH, [62](#)
 - UNDEFINED_TYPE, [62](#)
 - UNKNOWN_EXCEPTION, [61](#)
 - ZERO_SIZE, [61](#)
- rand
 - qpp, [39](#), [40](#)
- randint
 - qpp, [40](#)
- randket
 - qpp, [41](#)
- randkraus
 - qpp, [41](#)
- randn
 - qpp, [41](#), [42](#)
- randperm
 - qpp, [42](#)
- randrho
 - qpp, [42](#)
- rdevs
 - qpp, [53](#)
- renyi
 - qpp, [43](#)
- reshape
 - qpp, [44](#)
- Rn
 - qpp, [44](#)
- S
 - qpp, [53](#)
- SUBSYS_MISMATCH_DIMS
 - qpp::Exception, [62](#)
- save
 - qpp, [44](#)
- schmidtcoeff
 - qpp, [45](#)

schmidtprob
 qpp, [46](#)
shannon
 qpp, [47](#)
sinm
 qpp, [47](#)
spectralpowm
 qpp, [48](#)
sqrtm
 qpp, [48](#)
st
 qpp, [53](#)
States
 qpp, [48](#)
sum
 qpp, [49](#)
super
 qpp, [49](#)
syspermute
 qpp, [49](#)

T
 qpp, [53](#)
TYPE_MISMATCH
 qpp::Exception, [62](#)
trace
 qpp, [50](#)
transpose
 qpp, [50](#)
tsallis
 qpp, [51](#)

UNDEFINED_TYPE
 qpp::Exception, [62](#)
UNKNOWN_EXCEPTION
 qpp::Exception, [61](#)

W
 qpp, [53](#)

X
 qpp, [53](#)
x1
 qpp, [53](#)
Xd
 qpp, [51](#)

Y
 qpp, [53](#)
y0
 qpp, [53](#)
y1
 qpp, [53](#)

Z
 qpp, [53](#)
z0
 qpp, [53](#)
z1
 qpp, [53](#)

ZERO_SIZE
 qpp::Exception, [61](#)
Zd
 qpp, [51](#)