qpp

0.1

Generated by Doxygen 1.8.7

Mon Oct 20 2014 11:11:09

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Chapter 5

Namespace Documentation

5.1 qpp Namespace Reference

Namespaces

- ct
- internal
- types

Classes

- class DiscreteDistribution
- class DiscreteDistributionAbsSquare
- class Exception
- class Gates
- class NormalDistribution
- class Qudit
- class RandomDevices
- class Singleton
- class States
- class Timer
- class UniformIntDistribution
- 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)
- constexpr std::complex< double > operator""_i (unsigned long long int x)
- constexpr std::complex< double > operator""_i (long double x)
- 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 gconcurrence (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 > &subsys, 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< 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< 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)

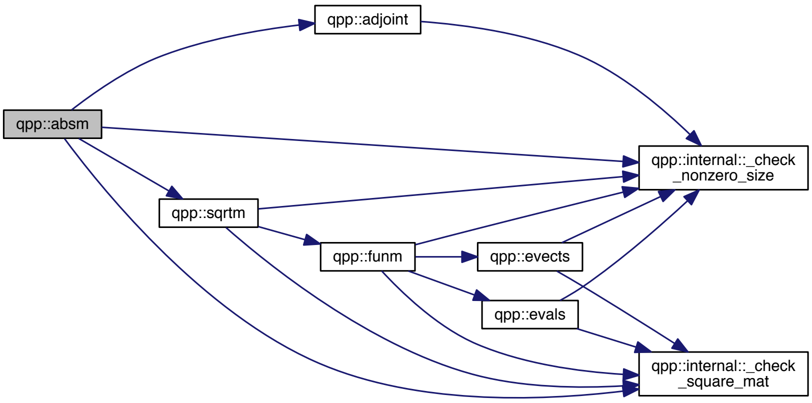
Variables

- const RandomDevices & rdevs = RandomDevices::get_instance()
- const Gates & gt = Gates::get_instance()
- const States & st = States::get_instance()

5.1.1 Function Documentation

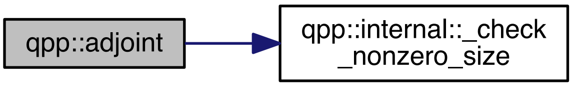
5.1.1.1 template<typename Derived > types::cmat qpp::absm (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::adjoint
qpp::absm
qpp::internal::_check _nonzero_size
qpp::sqrtm
qpp::funm
qpp::evects
qpp::evals
qpp::internal::_check _square_mat

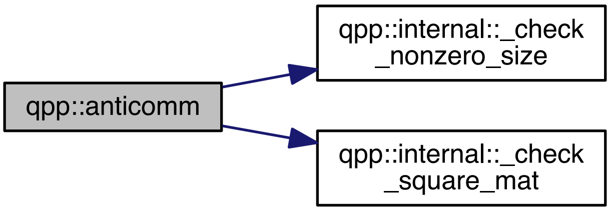
5.1.1.2 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::adjoint (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:



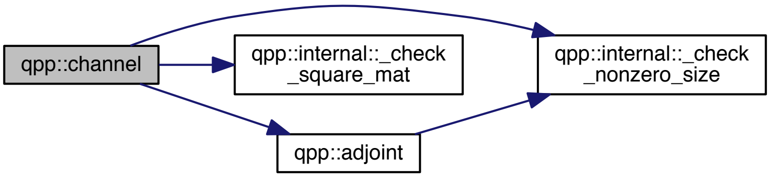
5.1.1.3 template<typename Derived1 , typename Derived2 > types::DynMat<typename Derived1::Scalar> qpp::anticomm (const Eigen::MatrixBase< Derived1 > & *A*, const Eigen::MatrixBase< Derived2 > & *B*)

Here is the call graph for this function:


qpp::anticomm
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

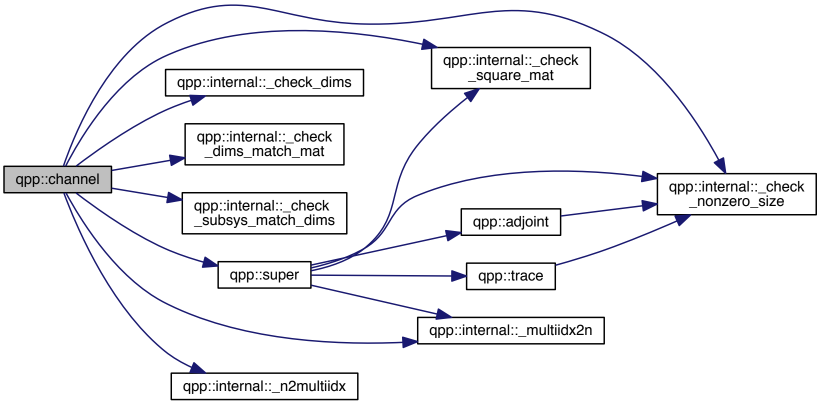
5.1.1.4 template<typename Derived > types::cmat qpp::channel (const Eigen::MatrixBase< Derived > & *rho*, const std::vector< types::cmat > & *Ks*)

Here is the call graph for this function:


qpp::channel
qpp::internal::_check _square_mat
qpp::internal::_check _nonzero_size
qpp::adjoint

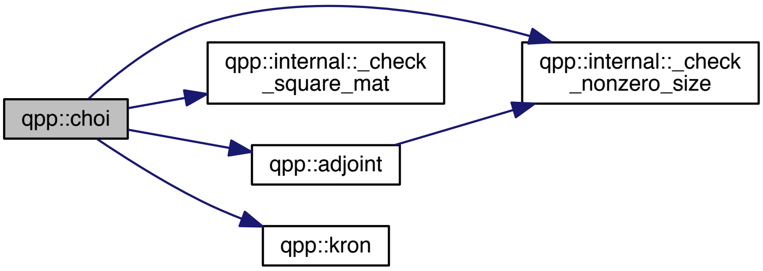
5.1.1.5 template<typename Derived > **types::cmat** qpp::channel (const Eigen::MatrixBase< Derived > & *rho*, const std::vector< types::cmat > & *Ks*, const std::vector< std::size_t > & *subsys*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::channel
qpp::internal::_check_dims
qpp::internal::_check_dims_match_mat
qpp::internal::_check_subsys_match_dims
qpp::super
qpp::internal::_n2multiidx
qpp::internal::_check_square_mat
qpp::adjoint
qpp::trace
qpp::internal::_multiidx2n
qpp::internal::_check_nonzero_size

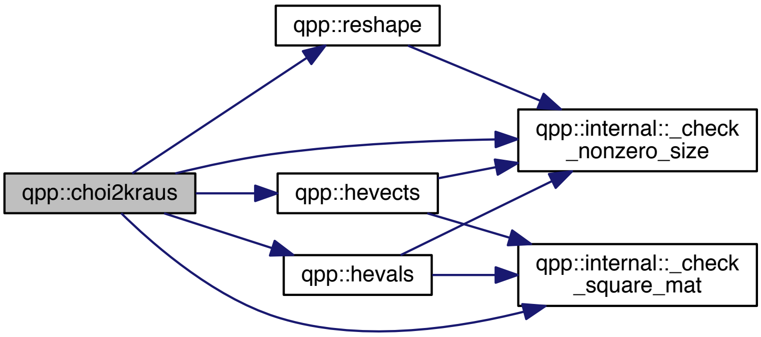
5.1.1.6 **types::cmat** qpp::choi (const std::vector< types::cmat > & *Ks*)

Here is the call graph for this function:


qpp::choi
qpp::internal::_check_square_mat
qpp::adjoint
qpp::kron
qpp::internal::_check_nonzero_size

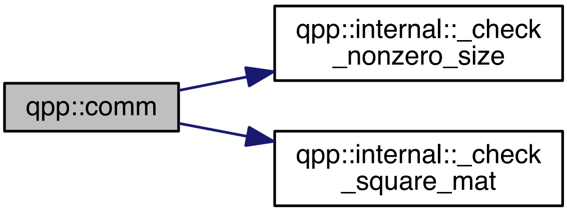
5.1.1.7 std::vector< **types::cmat** > qpp::choi2kraus (const types::cmat & *A*)

Here is the call graph for this function:


qpp::reshape
qpp::internal::_check _nonzero_size
qpp::choi2kraus
qpp::hevects
qpp::hevals
qpp::internal::_check _square_mat

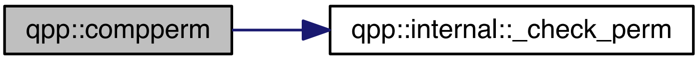
5.1.1.8 template<typename Derived1 , typename Derived2 > **types::DynMat**<typename Derived1::Scalar> qpp::comm (const Eigen::MatrixBase< Derived1 > & *A*, const Eigen::MatrixBase< Derived2 > & *B*)

Here is the call graph for this function:


qpp::internal::_check _nonzero_size
qpp::comm
qpp::internal::_check _square_mat

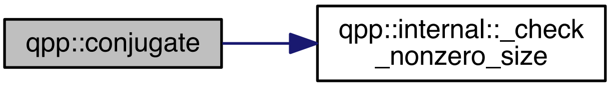
5.1.1.9 std::vector<std::size_t> qpp::compperm (const std::vector< std::size_t > & *perm,* const std::vector< std::size_t > & *sigma*)

Here is the call graph for this function:



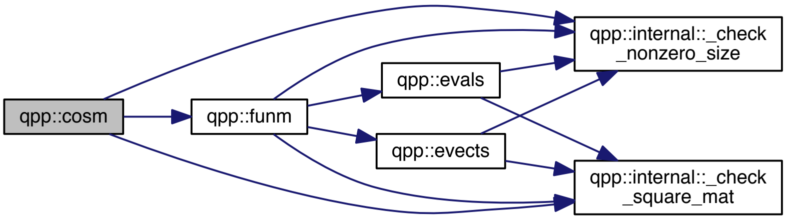
5.1.1.10 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::conjugate (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:



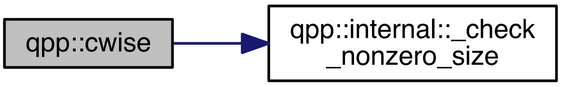
5.1.1.11 template<typename Derived > types::cmat qpp::cosm (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:



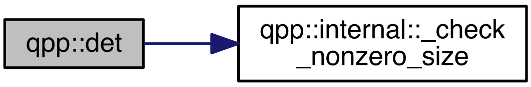
5.1.1.12 template<typename OutputScalar , typename Derived > types::DynMat<OutputScalar> qpp::cwise (const Eigen::MatrixBase< Derived > & *A*, OutputScalar(∗)(const typename Derived::Scalar &) *f*)

Here is the call graph for this function:



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

Here is the call graph for this function:



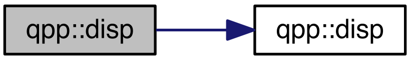
5.1.1.14 template<typename T > void qpp::disp (const T & *x*, const std::string & *separator*, const std::string & *start* = "[", const std::string & *end* = "]", std::ostream & *os* = `std::cout`)

5.1.1.15 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`)

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

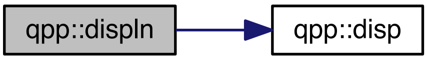
5.1.1.17 void qpp::disp (const types::cplx *c*, double *chop* = `ct::chop`, std::ostream & *os* = `std::cout`)

Here is the call graph for this function:



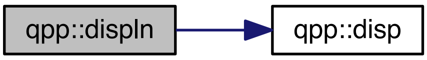
5.1.1.18 template<typename T > void qpp::displn (const T & *x*, const std::string & *separator*, const std::string & *start* = "[", const std::string & *end* = "]", std::ostream & *os* = std::cout)

Here is the call graph for this function:



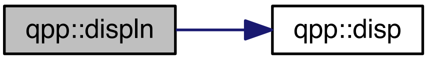
5.1.1.19 template<typename T > void qpp::displn (const T ∗ *x*, const std::size_t *n*, const std::string & *separator*, const std::string & *start* = "[", const std::string & *end* = "]", std::ostream & *os* = std::cout)

Here is the call graph for this function:



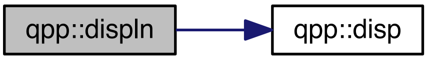
5.1.1.20 template<typename Derived > void qpp::displn (const Eigen::MatrixBase< Derived > & *A*, double *chop* = ct::chop, std::ostream & *os* = std::cout)

Here is the call graph for this function:



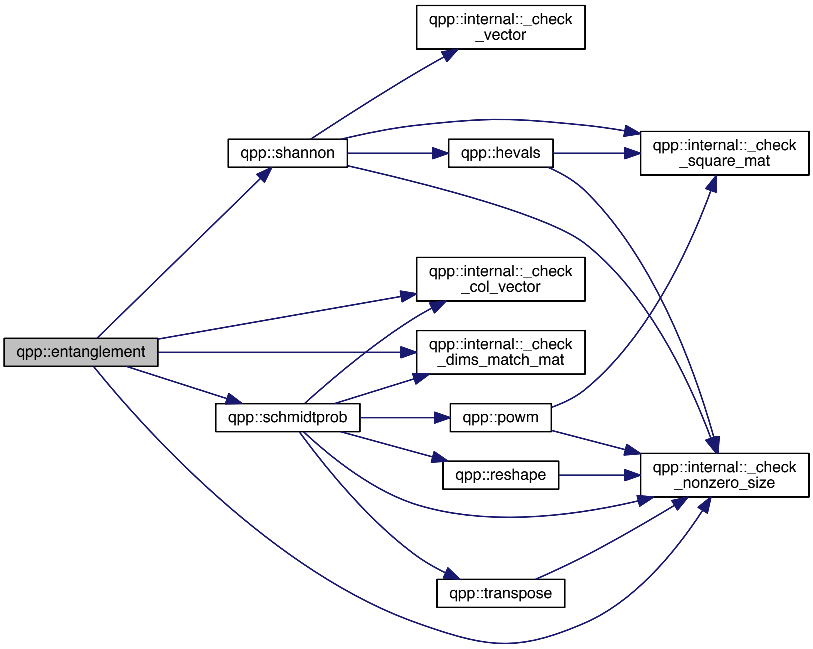
5.1.1.21 void qpp::displn (const types::cplx *c*, double *chop* = `ct::chop`, std::ostream & *os* = `std::cout`)

Here is the call graph for this function:



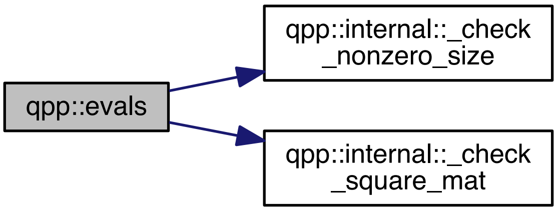
5.1.1.22 template<typename Derived > double qpp::entanglement (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::internal::_check_vector
qpp::shannon
qpp::hevals
qpp::internal::_check_square_mat
qpp::internal::_check_col_vector
qpp::entanglement
qpp::internal::_check_dims_match_mat
qpp::schmidtprob
qpp::powm
qpp::reshape
qpp::internal::_check_nonzero_size
qpp::transpose

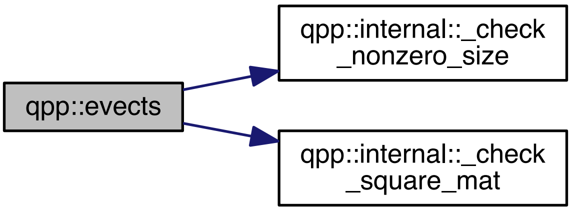
5.1.1.23 template<typename Derived > **types::cmat** qpp::evals (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::evals
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

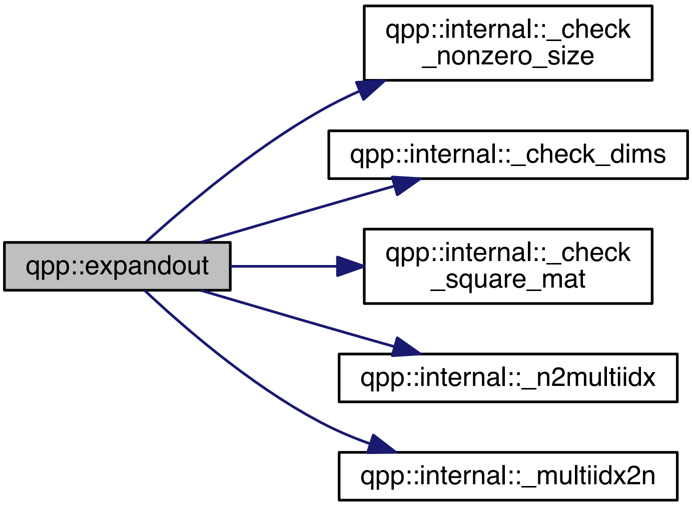
5.1.1.24 template<typename Derived > **types::cmat** qpp::evects (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::evects
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

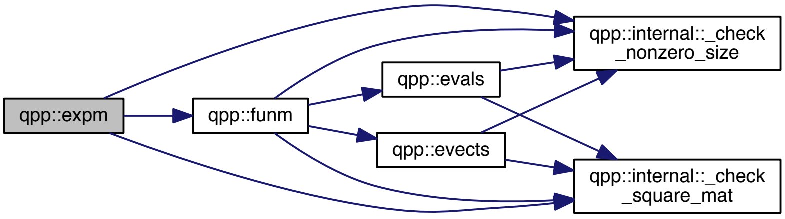
5.1.1.25 template< typename Derived > types::DynMat< typename Derived::Scalar> qpp::expandout (const Eigen::MatrixBase< Derived > & *A*, std::size_t *pos*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::internal::_check _nonzero_size
qpp::internal::_check_dims
qpp::expandout
qpp::internal::_check _square_mat
qpp::internal::_n2multiidx
qpp::internal::_multiidx2n

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

Here is the call graph for this function:


qpp::internal::_check _nonzero_size
qpp::evals
qpp::expm
qpp::funm
qpp::evects
qpp::internal::_check _square_mat

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

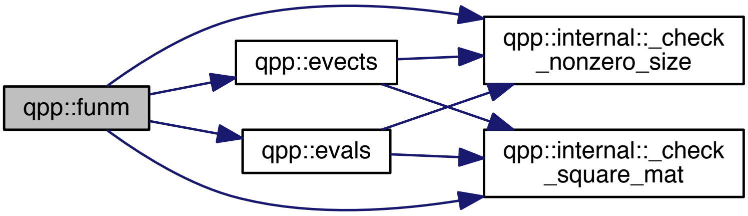
Parameters

A	input matrix
f	function pointer

Returns

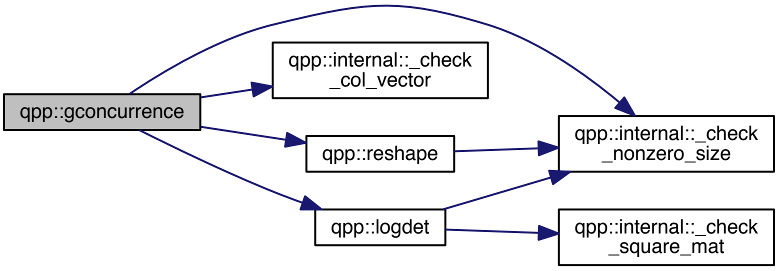
types::cmat

Here is the call graph for this function:


qpp::funm
qpp::evects
qpp::evals
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

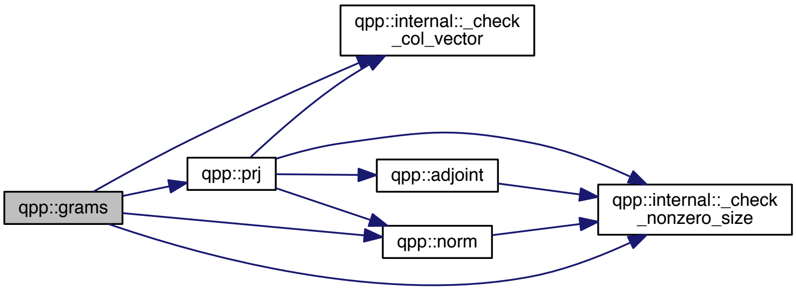
5.1.1.28 template< typename Derived > double qpp::gconcurrence (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::gconcurrence
qpp::internal::_check _col_vector
qpp::reshape
qpp::logdet
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

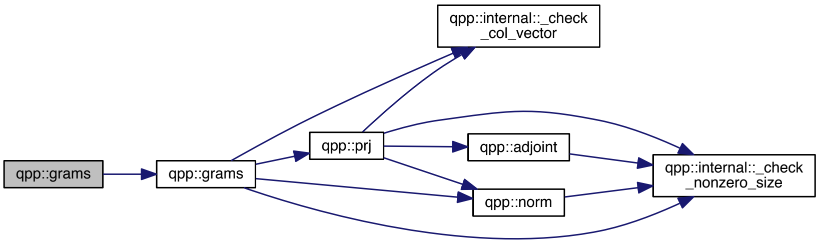
5.1.1.29 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::grams (const std::vector< Derived > & *Vs*)

Here is the call graph for this function:



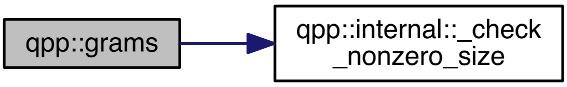
5.1.1.30 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::grams (const std::initializer_list< Derived > & *Vs*)

Here is the call graph for this function:



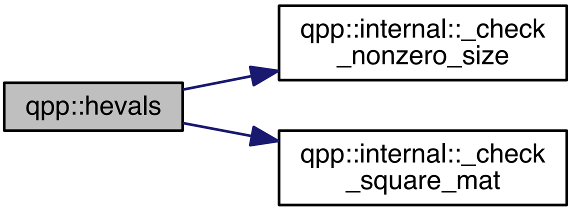
5.1.1.31 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::grams (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:



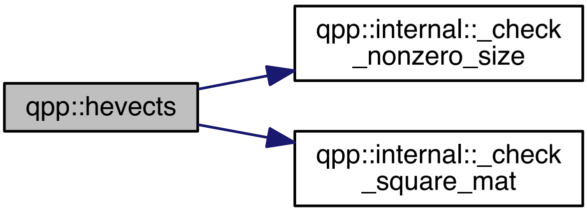
5.1.1.32 template<typename Derived > types::dmat qpp::hevals (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::hevals
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

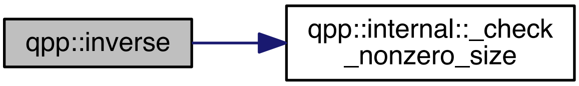
5.1.1.33 template<typename Derived > types::cmat qpp::hevects (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::hevects
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

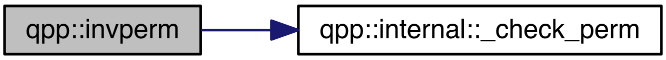
5.1.1.34 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::inverse (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:



5.1.1.35 std::vector<std::size_t> qpp::invperm (const std::vector< std::size_t > & *perm*)

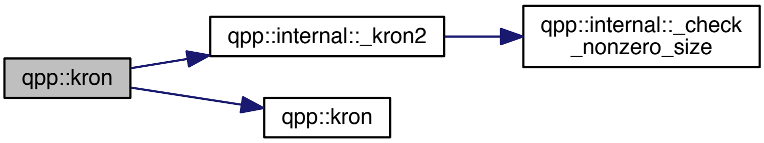
Here is the call graph for this function:



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

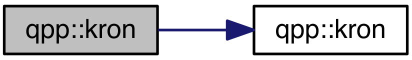
5.1.1.37 template<typename T , typename... Args> **types::DynMat**<typename T::Scalar> qpp::kron (const T & *head,* const Args &... *tail*)

Here is the call graph for this function:



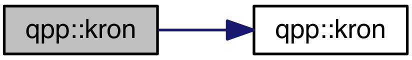
5.1.1.38 template<typename Derived > **types::DynMat**<typename Derived::Scalar> qpp::kron (const std::vector< Derived > & *As*)

Here is the call graph for this function:



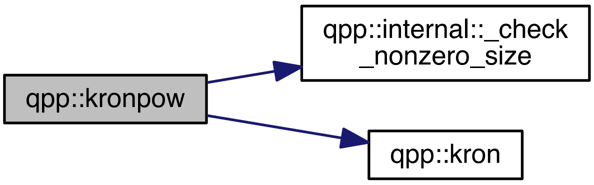
5.1.1.39 template<typename Derived > **types::DynMat**<typename Derived::Scalar> qpp::kron (const std::initializer_list< Derived > & *As*)

Here is the call graph for this function:



5.1.1.40 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:



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

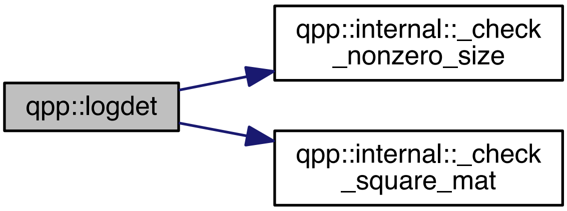
5.1.1.42 template<typename Derived > Derived qpp::loadMATLABmatrix (const std::string & *mat_file,* const std::string & *var_name*)

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

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

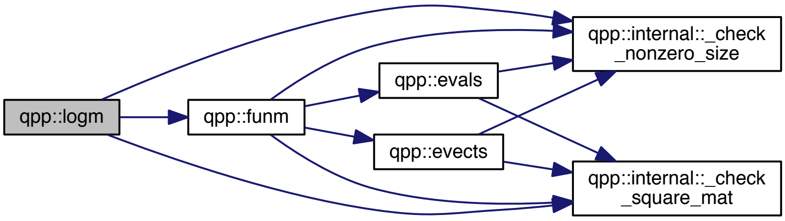
5.1.1.45 template< typename Derived > Derived::Scalar qpp::logdet (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::logdet
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

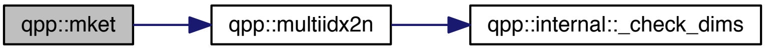
5.1.1.46 template< typename Derived > types::cmat qpp::logm (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::logm
qpp::funm
qpp::evals
qpp::evects
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

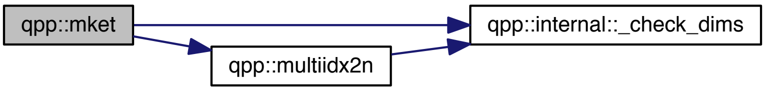
5.1.1.47 types::ket qpp::mket (const std::vector< std::size_t > & *mask*)

Here is the call graph for this function:



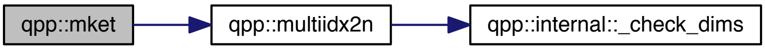
5.1.1.48 types::ket qpp::mket (const std::vector< std::size_t > & *mask*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:



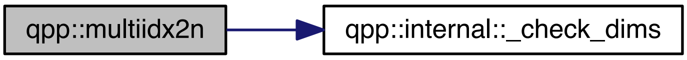
5.1.1.49 types::ket qpp::mket (const std::vector< std::size_t > & *mask*, std::size_t *d*)

Here is the call graph for this function:



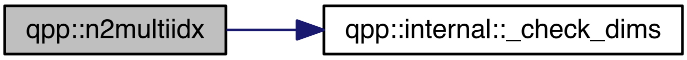
5.1.1.50 std::size_t qpp::multiidx2n (const std::vector< std::size_t > & *midx*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:



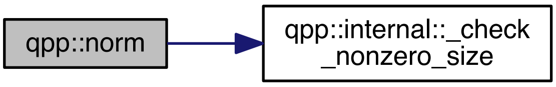
5.1.1.51 std::vector<std::size_t> qpp::n2multiidx (std::size_t *n*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:



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

Here is the call graph for this function:

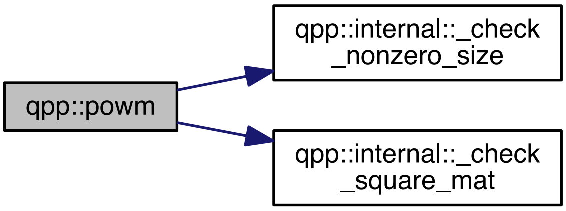


5.1.1.53 constexpr std::complex<double> qpp::operator""_i (unsigned long long int *x*)

5.1.1.54 constexpr std::complex<double> qpp::operator""_i (long double *x*)

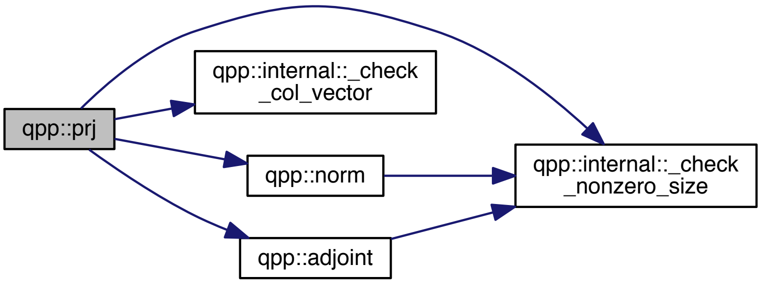
5.1.1.55 template< typename Derived > **types::DynMat**<typename Derived::Scalar> qpp::powm (const Eigen::MatrixBase< Derived > & *A*, std::size_t *n*)

Here is the call graph for this function:


qpp::powm
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

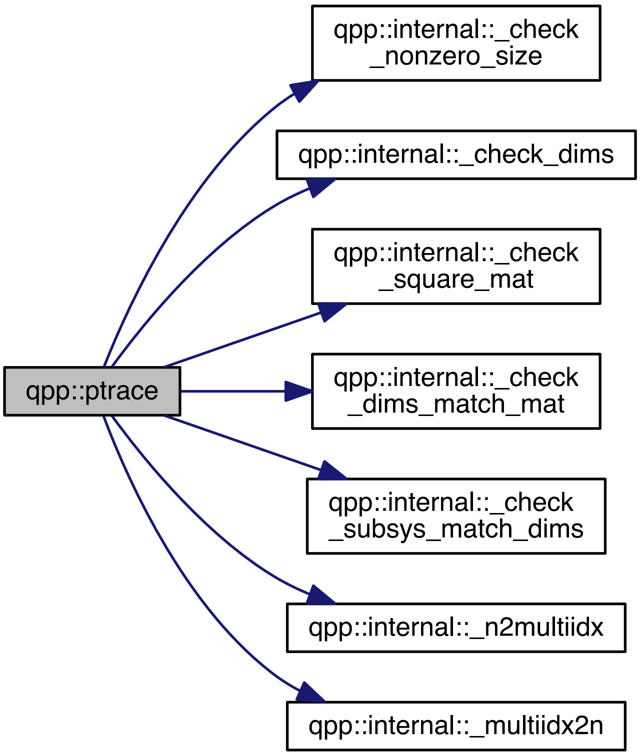
5.1.1.56 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::prj (const Eigen::MatrixBase< Derived > & *V*)

Here is the call graph for this function:



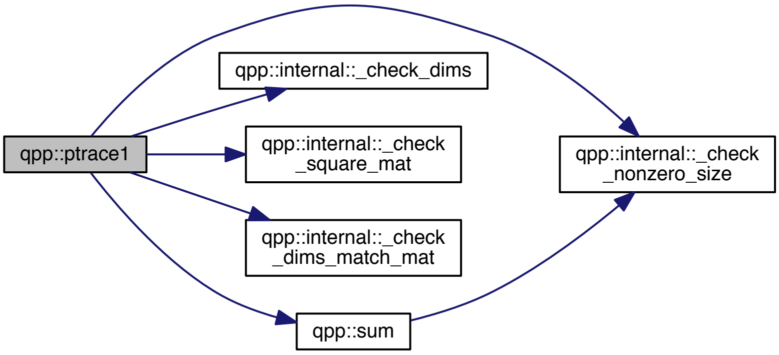
5.1.1.57 template<typename Derived > **types::DynMat**<typename Derived::Scalar> qpp::ptrace (const Eigen::MatrixBase< Derived > & *A,* const std::vector< std::size_t > & *subsys,* const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::internal::_check_nonzero_size
qpp::internal::_check_dims
qpp::internal::_check_square_mat
qpp::ptrace
qpp::internal::_check_dims_match_mat
qpp::internal::_check_subsys_match_dims
qpp::internal::_n2multiidx
qpp::internal::_multiidx2n

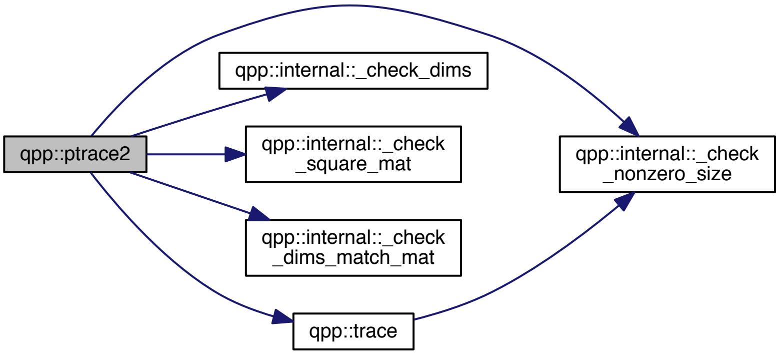
5.1.1.58 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::ptrace1 (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::ptrace1
qpp::internal::_check_dims
qpp::internal::_check_square_mat
qpp::internal::_check_dims_match_mat
qpp::sum
qpp::internal::_check_nonzero_size

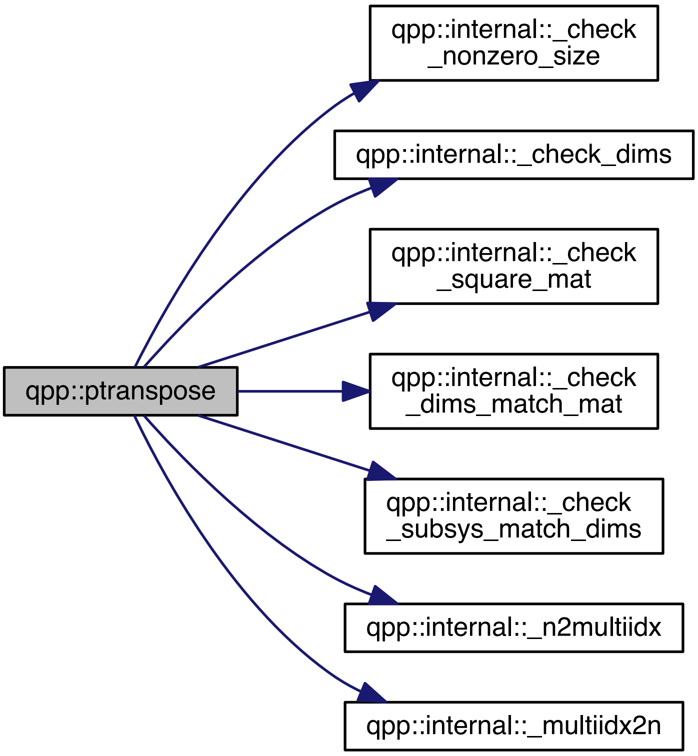
5.1.1.59 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::ptrace2 (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::ptrace2
qpp::internal::_check_dims
qpp::internal::_check_square_mat
qpp::internal::_check_dims_match_mat
qpp::trace
qpp::internal::_check_nonzero_size

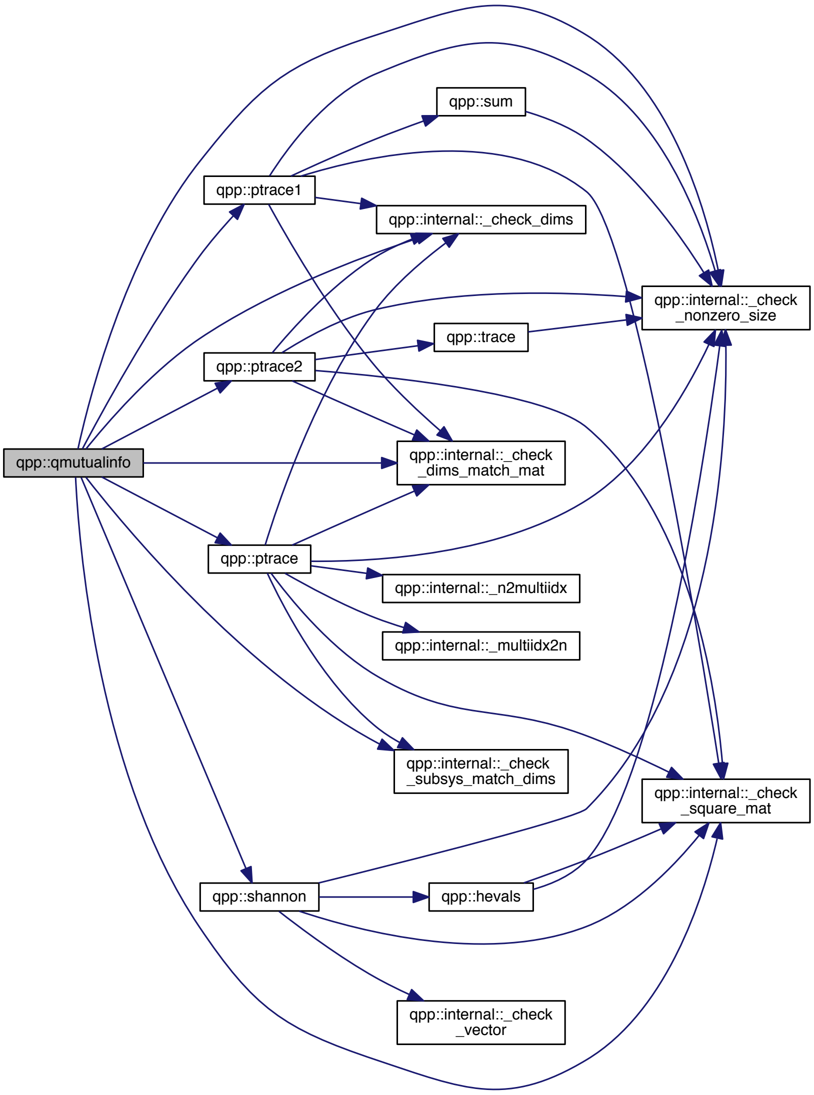
5.1.1.60 template< typename Derived > **types::DynMat**< typename Derived::Scalar> qpp::ptranspose (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *subsys*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::internal::_check _nonzero_size
qpp::internal::_check_dims
qpp::internal::_check _square_mat
qpp::ptranspose
qpp::internal::_check _dims_match_mat
qpp::internal::_check _subsys_match_dims
qpp::internal::_n2multiidx
qpp::internal::_multiidx2n

5.1.1.61 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:

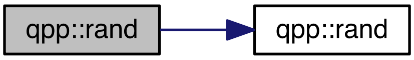

qpp::qmutualinfo
qpp::sum
qpp::ptrace1
qpp::internal::_check_dims
qpp::internal::_check _nonzero_size
qpp::trace
qpp::ptrace2
qpp::internal::_check _dims_match_mat
qpp::ptrace
qpp::internal::_n2multiidx
qpp::internal::_multiidx2n
qpp::internal::_check _subsys_match_dims
qpp::internal::_check _square_mat
qpp::shannon
qpp::hevals
qpp::internal::_check _vector

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

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

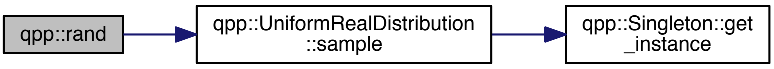
5.1.1.64 template<> types::cmat qpp::rand (std::size_t *rows*, std::size_t *cols*, double *a*, double *b*)

Here is the call graph for this function:



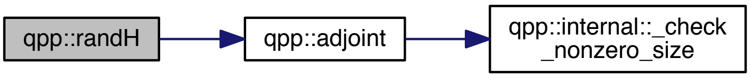
5.1.1.65 double qpp::rand (double *a* = 0, double *b* = 1)

Here is the call graph for this function:


qpp::rand
qpp::UniformRealDistribution ::sample
qpp::Singleton::get _instance

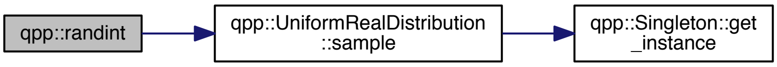
5.1.1.66 types::cmat qpp::randH (std::size_t *D*)

Here is the call graph for this function:



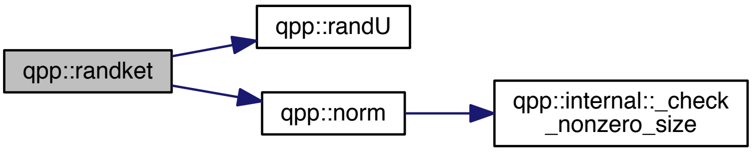
5.1.1.67 long long qpp::randint (long long *a*, long long *b*)

Here is the call graph for this function:


qpp::randint
qpp::UniformRealDistribution ::sample
qpp::Singleton::get _instance

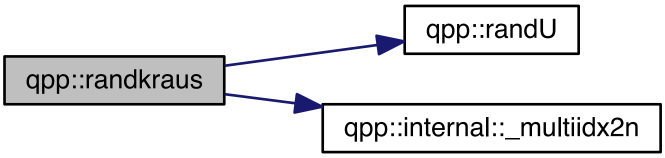
5.1.1.68 types::ket qpp::randket (std::size_t *D*)

Here is the call graph for this function:



5.1.1.69 std::vector< types::cmat > qpp::randkraus (std::size_t *n*, std::size_t *D*)

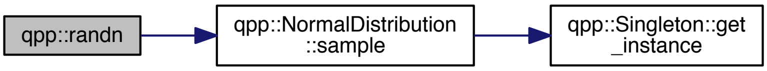
Here is the call graph for this function:



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

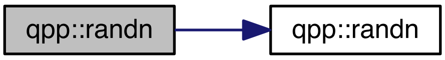
5.1.1.71 template<> types::dmat qpp::randn (std::size_t *rows*, std::size_t *cols*, double *mean*, double *sigma*)

Here is the call graph for this function:


qpp::randn
qpp::NormalDistribution ::sample
qpp::Singleton::get _instance

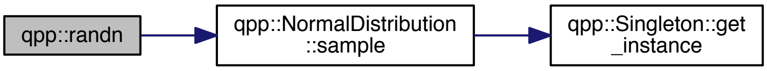
5.1.1.72 template<> types::cmat qpp::randn (std::size_t *rows*, std::size_t *cols*, double *mean*, double *sigma*)

Here is the call graph for this function:



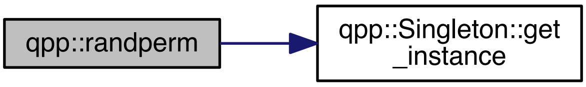
5.1.1.73 double qpp::randn (double *mean* = 0, double *sigma* = 1)

Here is the call graph for this function:


qpp::randn
qpp::NormalDistribution ::sample
qpp::Singleton::get _instance

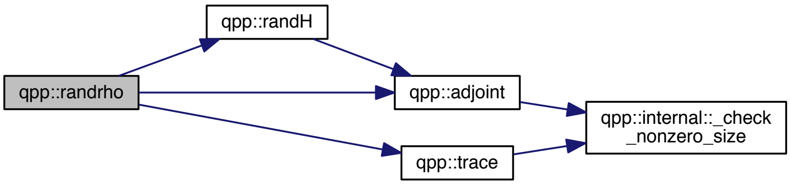
5.1.1.74 std::vector<std::size_t> qpp::randperm (std::size_t *n*)

Here is the call graph for this function:



5.1.1.75 types::cmat qpp::randrho (std::size_t *D*)

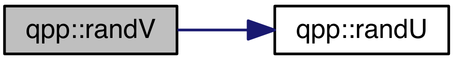
Here is the call graph for this function:



5.1.1.76 types::cmat qpp::randU (std::size_t *D*)

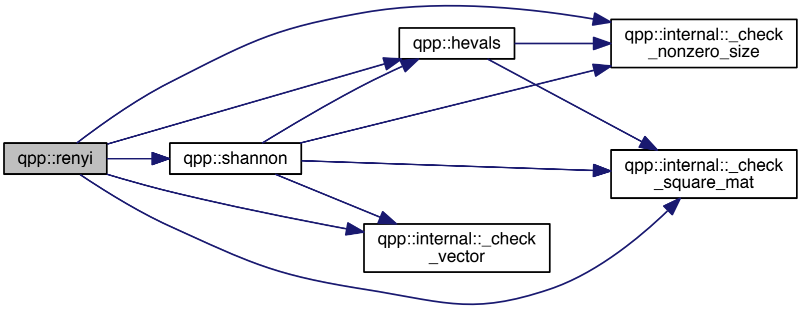
5.1.1.77 types::cmat qpp::randV (std::size_t *Din,* std::size_t *Dout*)

Here is the call graph for this function:



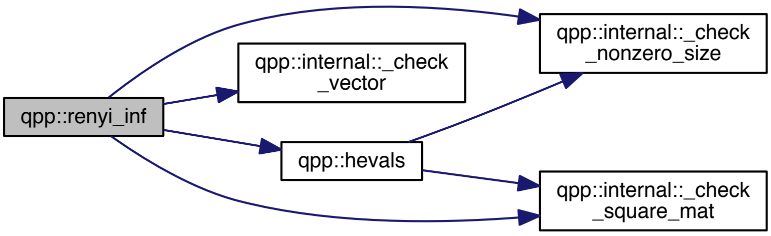
5.1.1.78 template<typename Derived > double qpp::renyi (const double *alpha,* const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::hevals
qpp::internal::_check _nonzero_size
qpp::renyi
qpp::shannon
qpp::internal::_check _square_mat
qpp::internal::_check _vector

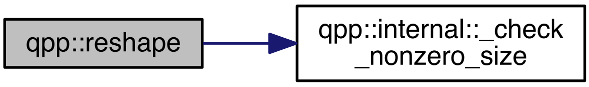
5.1.1.79 template<typename Derived > double qpp::renyi_inf (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::renyi_inf
qpp::internal::_check _vector
qpp::hevals
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

5.1.1.80 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:

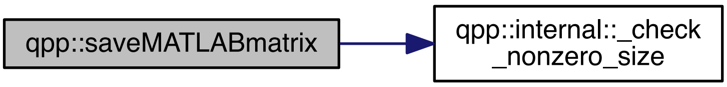


5.1.1.81 template<typename Derived > void qpp::save (const Eigen::MatrixBase< Derived > & *A,* const std::string & *fname*)

5.1.1.82 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*)

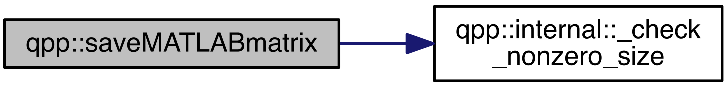
5.1.1.83 template<> void qpp::saveMATLABmatrix (const Eigen::MatrixBase< types::dmat > & *A,* const std::string & *mat_file,* const std::string & *var_name,* const std::string & *mode*)

Here is the call graph for this function:



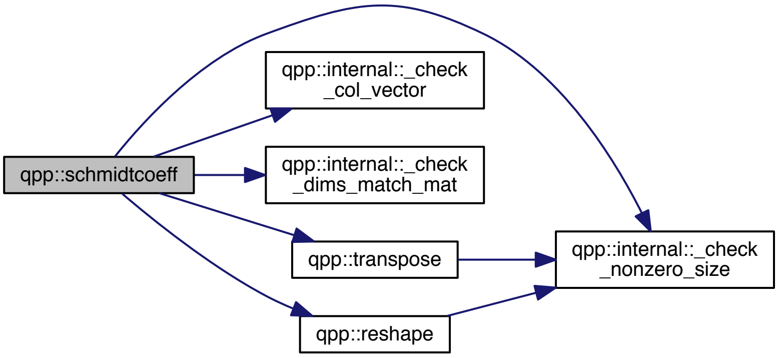
5.1.1.84 template<> void qpp::saveMATLABmatrix (const Eigen::MatrixBase< typename types::cmat > & *A*, const std::string & *mat_file*, const std::string & *var_name*, const std::string & *mode*)

Here is the call graph for this function:



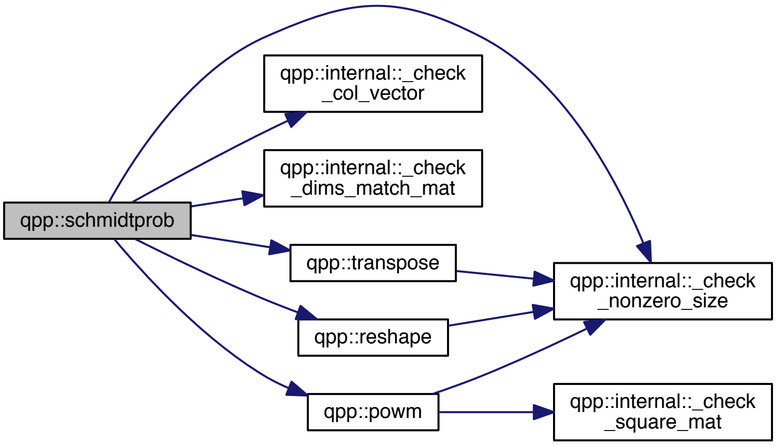
5.1.1.85 template<typename Derived > types::cmat qpp::schmidtcoeff (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::schmidtcoeff
qpp::internal::_check _col_vector
qpp::internal::_check _dims_match_mat
qpp::transpose
qpp::reshape
qpp::internal::_check _nonzero_size

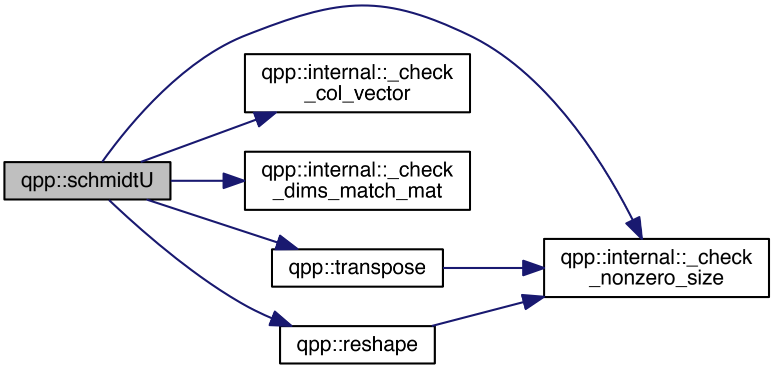
5.1.1.86 template<typename Derived > types::cmat qpp::schmidtprob (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::schmidtprob
qpp::internal::_check _col_vector
qpp::internal::_check _dims_match_mat
qpp::transpose
qpp::reshape
qpp::powm
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

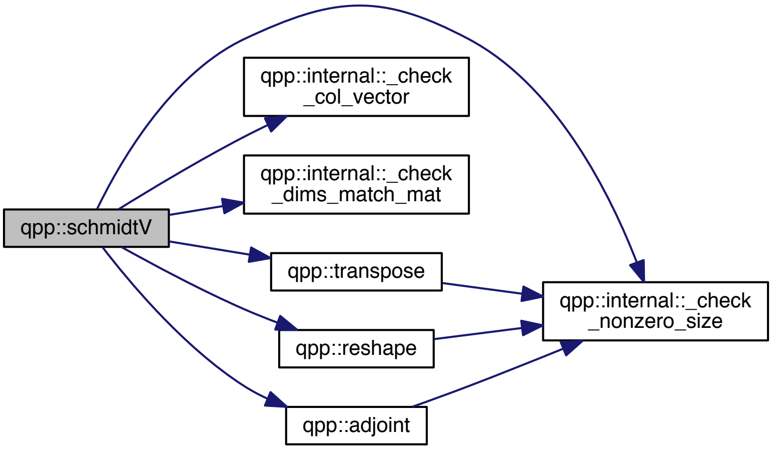
5.1.1.87 template<typename Derived > types::cmat qpp::schmidtU (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::schmidtU
qpp::internal::_check _col_vector
qpp::internal::_check _dims_match_mat
qpp::transpose
qpp::reshape
qpp::internal::_check _nonzero_size

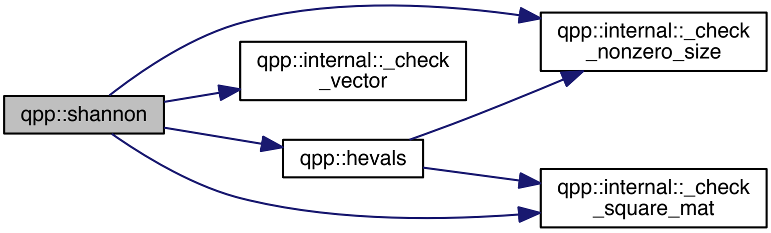
5.1.1.88 template<typename Derived > types::cmat qpp::schmidtV (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::schmidtV
qpp::internal::_check_col_vector
qpp::internal::_check_dims_match_mat
qpp::transpose
qpp::reshape
qpp::adjoint
qpp::internal::_check_nonzero_size

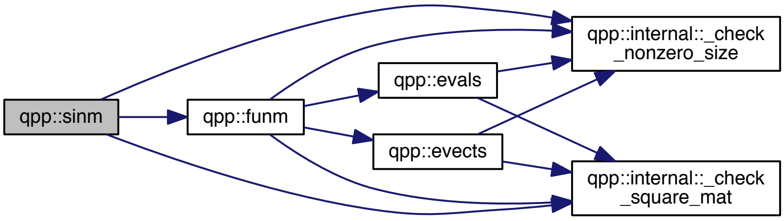
5.1.1.89 template<typename Derived > double qpp::shannon (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::shannon
qpp::internal::_check_vector
qpp::hevals
qpp::internal::_check_nonzero_size
qpp::internal::_check_square_mat

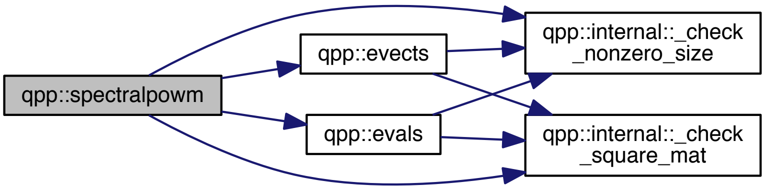
5.1.1.90 template<typename Derived > types::cmat qpp::sinm (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::sinm
qpp::funm
qpp::evals
qpp::evects
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

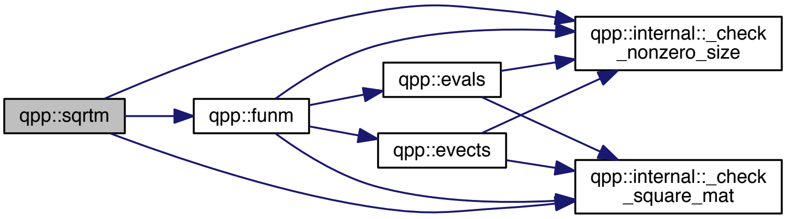
5.1.1.91 template<typename Derived > types::cmat qpp::spectralpowm (const Eigen::MatrixBase< Derived > & *A*, const types::cplx *z*)

Here is the call graph for this function:


qpp::spectralpowm
qpp::evects
qpp::evals
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

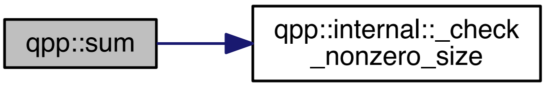
5.1.1.92 template<typename Derived > types::cmat qpp::sqrtm (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:


qpp::sqrtm
qpp::funm
qpp::evals
qpp::evects
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

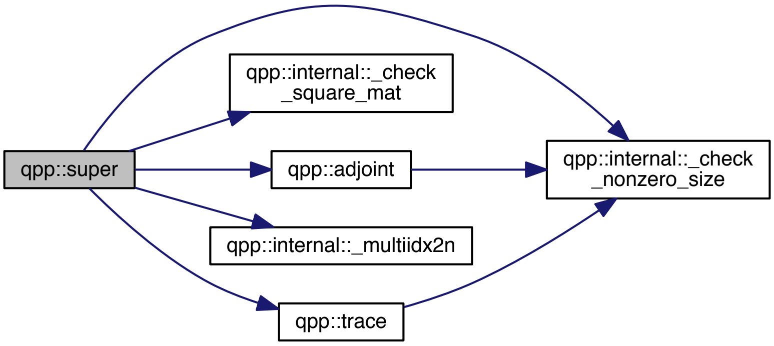
5.1.1.93 template<typename Derived > Derived::Scalar qpp::sum (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:



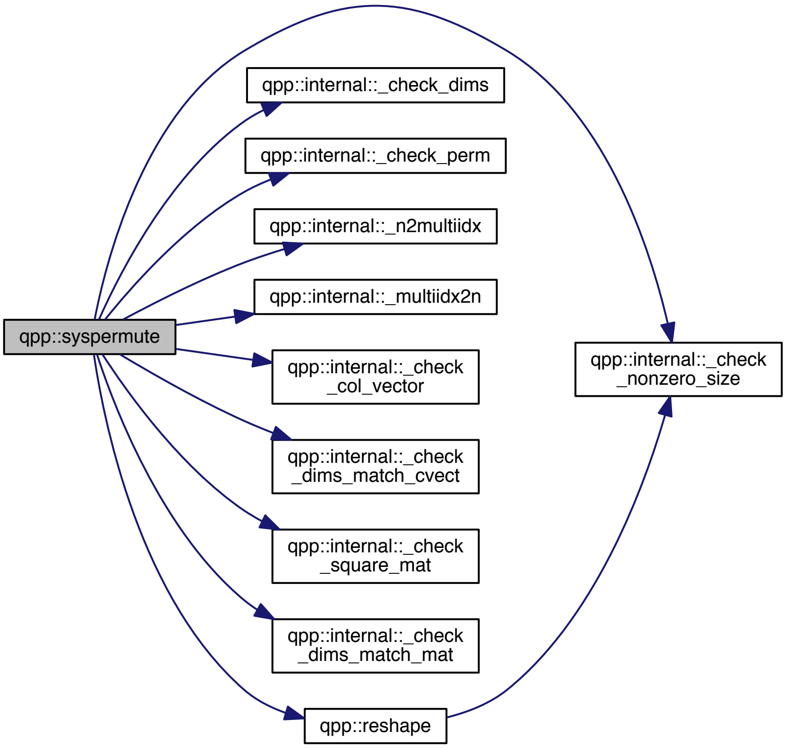
5.1.1.94 types::cmat qpp::super (const std::vector< types::cmat > & *Ks*)

Here is the call graph for this function:


qpp::super
qpp::internal::_check _square_mat
qpp::adjoint
qpp::internal::_multiidx2n
qpp::trace
qpp::internal::_check _nonzero_size

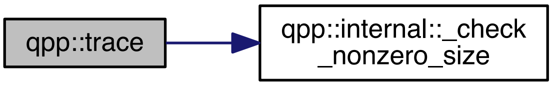
5.1.1.95 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::syspermute (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *perm*, const std::vector< std::size_t > & *dims*)

Here is the call graph for this function:


qpp::syspermute
qpp::internal::_check_dims
qpp::internal::_check_perm
qpp::internal::_n2multiidx
qpp::internal::_multiidx2n
qpp::internal::_check _col_vector
qpp::internal::_check _dims_match_cvect
qpp::internal::_check _square_mat
qpp::internal::_check _dims_match_mat
qpp::reshape
qpp::internal::_check _nonzero_size

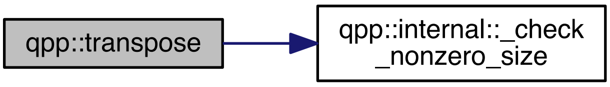
5.1.1.96 template<typename Derived > Derived::Scalar qpp::trace (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:



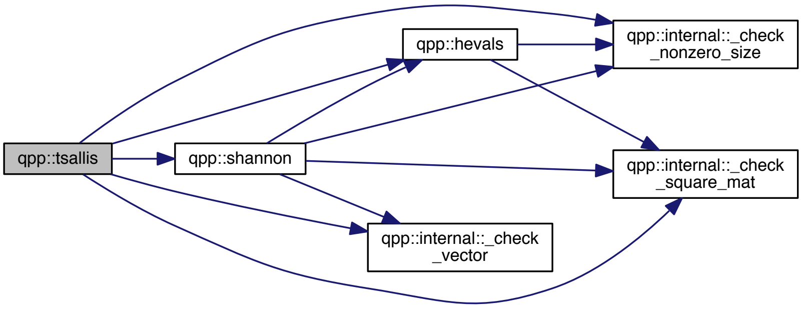
5.1.1.97 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::transpose (const Eigen::MatrixBase< Derived > & *A*)

Here is the call graph for this function:



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

Here is the call graph for this function:


qpp::tsallis
qpp::shannon
qpp::hevals
qpp::internal::_check_nonzero_size
qpp::internal::_check_square_mat
qpp::internal::_check_vector

5.1.2 Variable Documentation

5.1.2.1 const Gates& qpp::gt = Gates::get_instance()

5.1.2.2 const RandomDevices& qpp::rdevs = RandomDevices::get_instance()

5.1.2.3 const States& qpp::st = States::get_instance()

5.2 qpp::ct Namespace Reference

Functions

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

Variables

- constexpr double chop = 1e-10
- constexpr double eps = 1e-12

- constexpr std::size_t maxn = 64
- constexpr double pi = 3.141592653589793238462643383279502884
- constexpr 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 constexpr double qpp::ct::chop = 1e-10

5.2.2.2 constexpr double qpp::ct::ee = 2.718281828459045235360287471352662497

5.2.2.3 constexpr double qpp::ct::eps = 1e-12

5.2.2.4 constexpr std::size_t qpp::ct::maxn = 64

5.2.2.5 constexpr 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*)

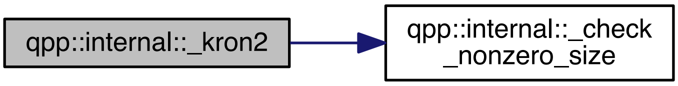
5.3.1.10 template<typename Derived > bool qpp::internal::_check_square_mat (const Eigen::MatrixBase< Derived > & *A*)

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:



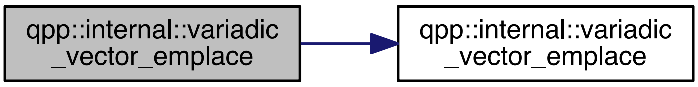
5.3.1.14 std::size_t qpp::internal::_multiidx2n (const std::size_t ∗ *midx,* std::size_t *numdims,* const std::size_t ∗ *dims*)

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:


qpp::internal::variadic_vector_emplace
qpp::internal::variadic_vector_emplace

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]

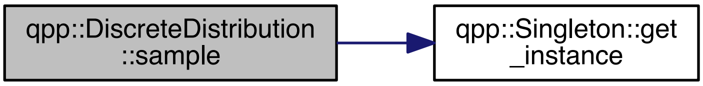
6.1.1.3 qpp::DiscreteDistribution::DiscreteDistribution (std::vector< double > *weights*) [inline]

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]

Here is the call graph for this function:


qpp::DiscreteDistribution ::sample
qpp::Singleton::get _instance

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)
- template<typename Derived >
  DiscreteDistributionAbsSquare (const Eigen::MatrixBase< Derived > &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]

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

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

6.2.1.4 template< typename Derived > qpp::DiscreteDistributionAbsSquare::DiscreteDistributionAbsSquare (const Eigen::MatrixBase< Derived > & *V*) [inline]

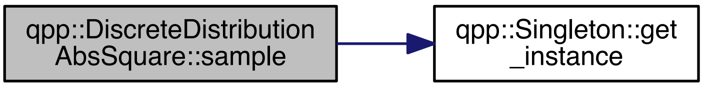
6.2.2 Member Function Documentation

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]

Here is the call graph for this function:


qpp::DiscreteDistribution AbsSquare::sample
qpp::Singleton::get _instance

6.2.3 Member Data Documentation

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

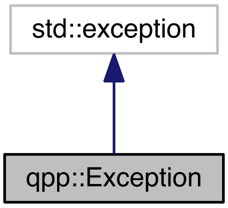
The documentation for this class was generated from the following file:

- include/classes/stat.h

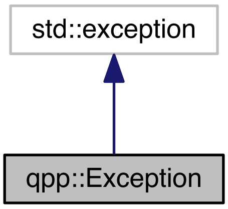
6.3 qpp::Exception Class Reference

```
#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 noexceptoverride

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

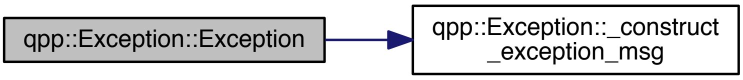
UNDEFINED_TYPE

CUSTOM_EXCEPTION

6.3.2 Constructor & Destructor Documentation

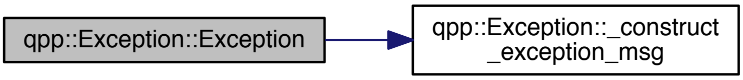
6.3.2.1 qpp::Exception::Exception (const std::string & *where*, const Type & *type*) [inline]

Here is the call graph for this function:


qpp::Exception::Exception
qpp::Exception::_construct _exception_msg

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

Here is the call graph for this function:


qpp::Exception::Exception
qpp::Exception::_construct _exception_msg

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]

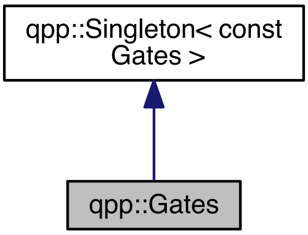
The documentation for this class was generated from the following file:

- include/classes/exception.h

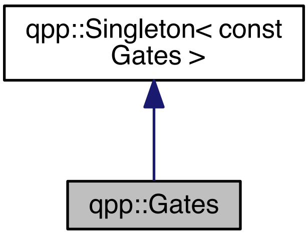
6.4 qpp::Gates Class Reference

```
#include <gates.h>
```

Inheritance diagram for qpp::Gates:



Collaboration diagram for qpp::Gates:



Public Member Functions

- 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 Id (std::size_t D) const
- template<typename Derived1 , typename Derived2 >
  types::DynMat< typename
  Derived1::Scalar > applyCTRL (const Eigen::MatrixBase< Derived1 > &state, const Eigen::MatrixBase< Derived2 > &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
- 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

Public Attributes

- types::cmat Id2 { types::cmat::Identity(2, 2) }
- types::cmat H { types::cmat::Zero(2, 2) }
- types::cmat X { types::cmat::Zero(2, 2) }
- types::cmat Y { types::cmat::Zero(2, 2) }
- types::cmat Z { types::cmat::Zero(2, 2) }
- types::cmat S { types::cmat::Zero(2, 2) }
- types::cmat T { types::cmat::Zero(2, 2) }
- types::cmat CNOTab { types::cmat::Identity(4, 4) }
- types::cmat CZ { types::cmat::Identity(4, 4) }
- types::cmat CNOTba { types::cmat::Zero(4, 4) }
- types::cmat SWAP { types::cmat::Identity(4, 4) }
- types::cmat TOF { types::cmat::Identity(8, 8) }
- types::cmat FRED { types::cmat::Identity(8, 8) }

Private Member Functions

- Gates ()

Friends

- class Singleton< const Gates >

Additional Inherited Members

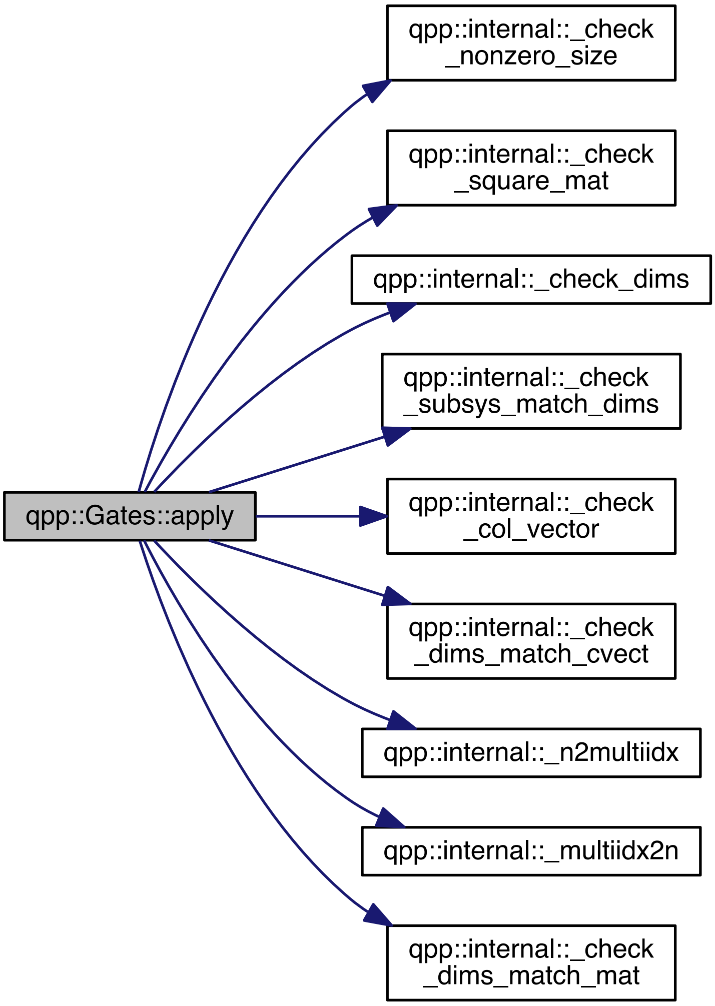
6.4.1 Constructor & Destructor Documentation

6.4.1.1 qpp::Gates::Gates ( ) [inline], [private]

6.4.2 Member Function Documentation

6.4.2.1 template<typename Derived1 , typename Derived2 > types::DynMat<typename Derived1::Scalar> qpp::Gates::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 [inline]

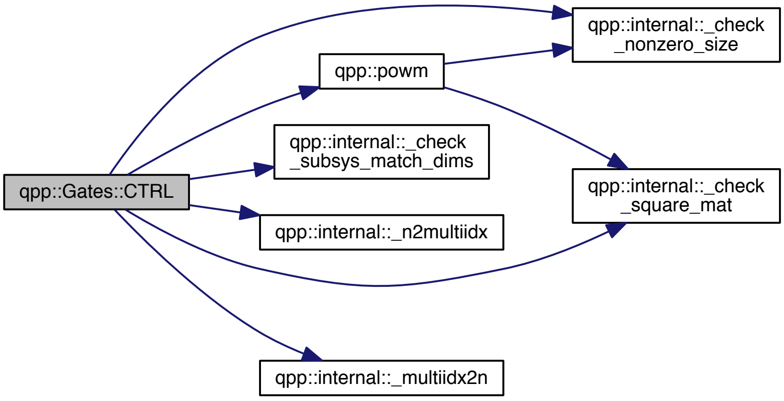
Here is the call graph for this function:


qpp::internal::_check_nonzero_size
qpp::internal::_check_square_mat
qpp::internal::_check_dims
qpp::internal::_check_subsys_match_dims
qpp::Gates::apply
qpp::internal::_check_col_vector
qpp::internal::_check_dims_match_cvect
qpp::internal::_n2multiidx
qpp::internal::_multiidx2n
qpp::internal::_check_dims_match_mat

6.4.2.2 template<typename Derived1 , typename Derived2 > types::DynMat<typename Derived1::Scalar> qpp::Gates::applyCTRL (const Eigen::MatrixBase< Derived1 > & *state*, const Eigen::MatrixBase< Derived2 > & *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 [inline]

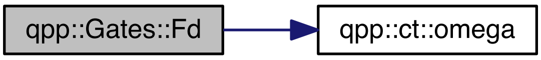
6.4.2.3 template<typename Derived > types::DynMat<typename Derived::Scalar> qpp::Gates::CTRL (const Eigen::MatrixBase< Derived > & *A*, const std::vector< std::size_t > & *ctrl*, const std::vector< std::size_t > & *subsys*, std::size_t *n*, std::size_t *d* = 2) const [inline]

Here is the call graph for this function:


qpp::Gates::CTRL
qpp::internal::_check _nonzero_size
qpp::powm
qpp::internal::_check _subsys_match_dims
qpp::internal::_check _square_mat
qpp::internal::_n2multiidx
qpp::internal::_multiidx2n

6.4.2.4 types::cmat qpp::Gates::Fd (std::size_t *D*) const [inline]

Here is the call graph for this function:

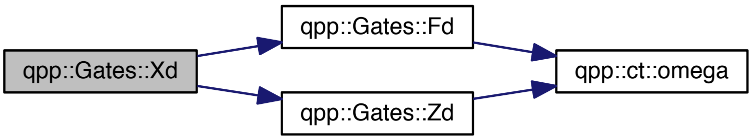


6.4.2.5 template<typename Derived = Eigen::MatrixXcd> Derived qpp::Gates::Id (std::size_t *D*) const [inline]

6.4.2.6 types::cmat qpp::Gates::Rn (double *theta*, std::vector< double > *n*) const [inline]

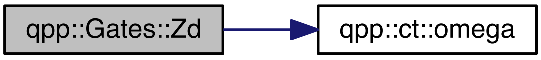
6.4.2.7 types::cmat qpp::Gates::Xd (std::size_t *D*) const [inline]

Here is the call graph for this function:



6.4.2.8 types::cmat qpp::Gates::Zd (std::size_t *D*) const [inline]

Here is the call graph for this function:



6.4.3 Friends And Related Function Documentation

6.4.3.1 friend class Singleton< const Gates > [friend]

6.4.4 Member Data Documentation

6.4.4.1 types::cmat qpp::Gates::CNOTab { types::cmat::Identity(4, 4) }

6.4.4.2 types::cmat qpp::Gates::CNOTba { types::cmat::Zero(4, 4) }

6.4.4.3 types::cmat qpp::Gates::CZ { types::cmat::Identity(4, 4) }

6.4.4.4 types::cmat qpp::Gates::FRED { types::cmat::Identity(8, 8) }

6.4.4.5 types::cmat qpp::Gates::H { types::cmat::Zero(2, 2) }

6.4.4.6 types::cmat qpp::Gates::Id2 { types::cmat::Identity(2, 2) }

6.4.4.7 types::cmat qpp::Gates::S { types::cmat::Zero(2, 2) }

6.4.4.8 types::cmat qpp::Gates::SWAP { types::cmat::Identity(4, 4) }

6.4.4.9 types::cmat qpp::Gates::T { types::cmat::Zero(2, 2) }

6.4.4.10 types::cmat qpp::Gates::TOF { types::cmat::Identity(8, 8) }

6.4.4.11 types::cmat qpp::Gates::X { types::cmat::Zero(2, 2) }

6.4.4.12 types::cmat qpp::Gates::Y { types::cmat::Zero(2, 2) }

6.4.4.13 types::cmat qpp::Gates::Z { types::cmat::Zero(2, 2) }

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

- include/classes/gates.h

6.5 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

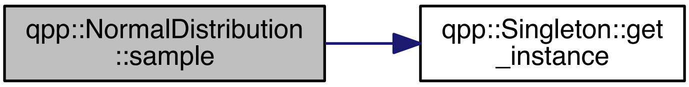
6.5.1 Constructor & Destructor Documentation

6.5.1.1 qpp::NormalDistribution::NormalDistribution (double *mean* = 0, double *sigma* = 1) [inline]

6.5.2 Member Function Documentation

6.5.2.1 double qpp::NormalDistribution::sample ( ) [inline]

Here is the call graph for this function:


qpp::NormalDistribution ::sample
qpp::Singleton::get _instance

6.5.3 Member Data Documentation

6.5.3.1 std::normal_distribution qpp::NormalDistribution::_d [protected]

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

- include/classes/stat.h

6.6 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

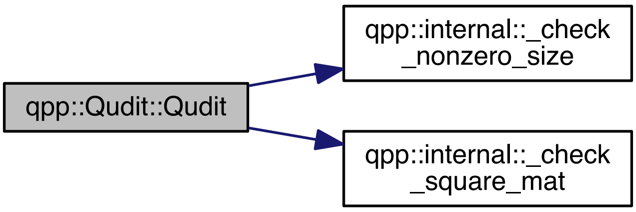
Private Attributes

- types::cmat _rho
- std::size_t _D

6.6.1 Constructor & Destructor Documentation

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

Here is the call graph for this function:


qpp::Qudit::Qudit
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat

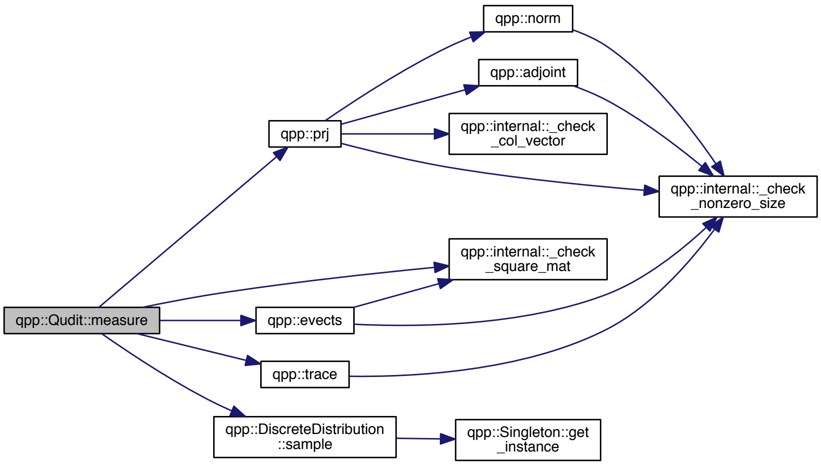
6.6.2 Member Function Documentation

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

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

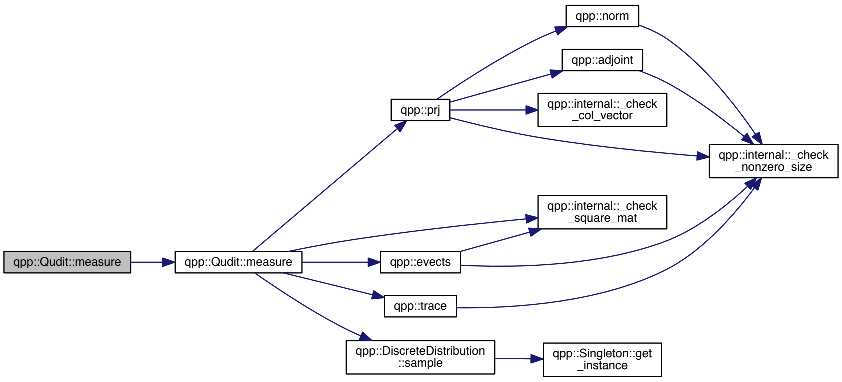
6.6.2.3 std::size_t qpp::Qudit::measure (const types::cmat & *U*, bool *destructive* = `false`) `[inline]`

Here is the call graph for this function:


qpp::norm
qpp::adjoint
qpp::prj
qpp::internal::_check _col_vector
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat
qpp::Qudit::measure
qpp::evects
qpp::trace
qpp::DiscreteDistribution ::sample
qpp::Singleton::get _instance

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

Here is the call graph for this function:


qpp::norm
qpp::adjoint
qpp::prj
qpp::internal::_check _col_vector
qpp::internal::_check _nonzero_size
qpp::internal::_check _square_mat
qpp::Qudit::measure
qpp::Qudit::measure
qpp::evects
qpp::trace
qpp::DiscreteDistribution ::sample
qpp::Singleton::get _instance

6.6.3 Member Data Documentation

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

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

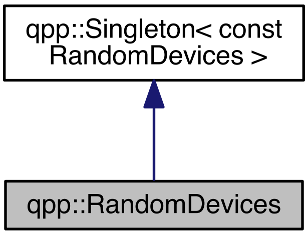
The documentation for this class was generated from the following file:

- include/classes/qudit.h

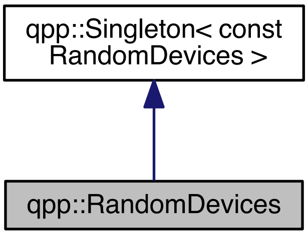
6.7 qpp::RandomDevices Class Reference

`#include <randevs.h>`

Inheritance diagram for qpp::RandomDevices:



Collaboration diagram for qpp::RandomDevices:



Public Attributes

- std::random_device _rd
- std::mt19937 _rng

Private Member Functions

- RandomDevices ()

Friends

- class Singleton< const RandomDevices >

Additional Inherited Members

6.7.1 Constructor & Destructor Documentation

6.7.1.1 qpp::RandomDevices::RandomDevices ( ) [inline], [private]

6.7.2 Friends And Related Function Documentation

6.7.2.1 friend class Singleton< const RandomDevices > [friend]

6.7.3 Member Data Documentation

6.7.3.1 std::random_device qpp::RandomDevices::_rd

6.7.3.2 std::mt19937 qpp::RandomDevices::_rng [mutable]

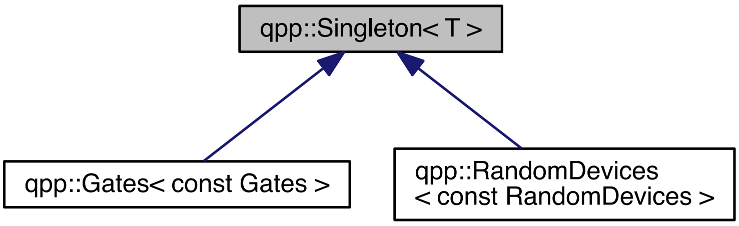
The documentation for this class was generated from the following file:

- include/classes/randevs.h

6.8 qpp::Singleton< T > Class Template Reference

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

Inheritance diagram for qpp::Singleton< T >:



Static Public Member Functions

- static T & get_instance ()

Protected Member Functions

- Singleton ()=default
- virtual ∼Singleton ()
- Singleton (const Singleton &)=delete
- Singleton & operator= (const Singleton &)=delete

6.8.1 Constructor & Destructor Documentation

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

6.8.1.2 template<typename T> virtual qpp::Singleton< T >::∼Singleton ( ) [inline], [protected], [virtual]

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

6.8.2 Member Function Documentation

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

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

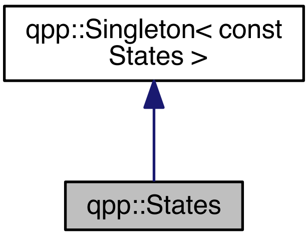
The documentation for this class was generated from the following file:

- include/classes/singleton.h

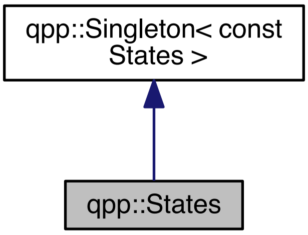
6.9 qpp::States Class Reference

```
#include <states.h>
```

Inheritance diagram for qpp::States:



Collaboration diagram for qpp::States:



Public Attributes

- types::ket x0 { types::ket::Zero(2) }
- types::ket x1 { types::ket::Zero(2) }
- types::ket y0 { types::ket::Zero(2) }
- types::ket y1 { types::ket::Zero(2) }
- types::ket z0 { types::ket::Zero(2) }
- types::ket z1 { types::ket::Zero(2) }
- types::cmat px0 { types::cmat::Zero(2, 2) }
- types::cmat px1 { types::cmat::Zero(2, 2) }
- types::cmat py0 { types::cmat::Zero(2, 2) }
- types::cmat py1 { types::cmat::Zero(2, 2) }
- types::cmat pz0 { types::cmat::Zero(2, 2) }
- types::cmat pz1 { types::cmat::Zero(2, 2) }
- types::ket b00 { types::ket::Zero(4) }
- types::ket b01 { types::ket::Zero(4) }
- types::ket b10 { types::ket::Zero(4) }
- types::ket b11 { types::ket::Zero(4) }
- types::cmat pb00 { types::cmat::Zero(4, 4) }
- types::cmat pb01 { types::cmat::Zero(4, 4) }
- types::cmat pb10 { types::cmat::Zero(4, 4) }
- types::cmat pb11 { types::cmat::Zero(4, 4) }
- types::ket GHZ { types::ket::Zero(8) }
- types::ket W { types::ket::Zero(8) }
- types::cmat pGHZ { types::cmat::Zero(8, 8) }
- types::cmat pW { types::cmat::Zero(8, 8) }

Private Member Functions

- States ()

Friends

- class Singleton< const States >

Additional Inherited Members

6.9.1 Constructor & Destructor Documentation

6.9.1.1 qpp::States::States ( ) [inline], [private]

6.9.2 Friends And Related Function Documentation

6.9.2.1 friend class **Singleton**< const **States** > [friend]

6.9.3 Member Data Documentation

6.9.3.1 **types::ket** qpp::States::b00 { types::ket::Zero(4) }

6.9.3.2 **types::ket** qpp::States::b01 { types::ket::Zero(4) }

6.9.3.3 **types::ket** qpp::States::b10 { types::ket::Zero(4) }

6.9.3.4 types::ket qpp::States::b11 { types::ket::Zero(4) }

6.9.3.5 types::ket qpp::States::GHZ { types::ket::Zero(8) }

6.9.3.6 types::cmat qpp::States::pb00 { types::cmat::Zero(4, 4) }

6.9.3.7 types::cmat qpp::States::pb01 { types::cmat::Zero(4, 4) }

6.9.3.8 types::cmat qpp::States::pb10 { types::cmat::Zero(4, 4) }

6.9.3.9 types::cmat qpp::States::pb11 { types::cmat::Zero(4, 4) }

6.9.3.10 types::cmat qpp::States::pGHZ { types::cmat::Zero(8, 8) }

6.9.3.11 types::cmat qpp::States::pW { types::cmat::Zero(8, 8) }

6.9.3.12 types::cmat qpp::States::px0 { types::cmat::Zero(2, 2) }

6.9.3.13 types::cmat qpp::States::px1 { types::cmat::Zero(2, 2) }

6.9.3.14 types::cmat qpp::States::py0 { types::cmat::Zero(2, 2) }

6.9.3.15 types::cmat qpp::States::py1 { types::cmat::Zero(2, 2) }

6.9.3.16 types::cmat qpp::States::pz0 { types::cmat::Zero(2, 2) }

6.9.3.17 types::cmat qpp::States::pz1 { types::cmat::Zero(2, 2) }

6.9.3.18 types::ket qpp::States::W { types::ket::Zero(8) }

6.9.3.19 types::ket qpp::States::x0 { types::ket::Zero(2) }

6.9.3.20 types::ket qpp::States::x1 { types::ket::Zero(2) }

6.9.3.21 types::ket qpp::States::y0 { types::ket::Zero(2) }

6.9.3.22 types::ket qpp::States::y1 { types::ket::Zero(2) }

6.9.3.23 types::ket qpp::States::z0 { types::ket::Zero(2) }

6.9.3.24 types::ket qpp::States::z1 { types::ket::Zero(2) }

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

- include/classes/states.h

6.10 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)

6.10.1 Constructor & Destructor Documentation

6.10.1.1 qpp::Timer::Timer ( ) [inline]

6.10.2 Member Function Documentation

6.10.2.1 double qpp::Timer::seconds ( ) const [inline]

6.10.2.2 void qpp::Timer::tic ( ) [inline]

6.10.2.3 void qpp::Timer::toc ( ) [inline]

6.10.3 Friends And Related Function Documentation

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

6.10.4 Member Data Documentation

6.10.4.1 std::chrono::steady_clock::time_point qpp::Timer::_end [protected]

6.10.4.2 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

6.11 qpp::UniformIntDistribution Class Reference

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

Public Member Functions

- UniformIntDistribution (int a=0, int b=1)
- int sample ()

Protected Attributes

- std::uniform_int_distribution _d

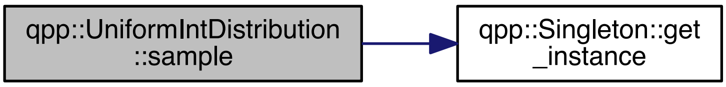
6.11.1 Constructor & Destructor Documentation

6.11.1.1 qpp::UniformIntDistribution::UniformIntDistribution (int *a* = 0, int *b* = 1) [inline]

6.11.2 Member Function Documentation

6.11.2.1 int qpp::UniformIntDistribution::sample ( ) [inline]

Here is the call graph for this function:


qpp::UniformIntDistribution ::sample
qpp::Singleton::get _instance

6.11.3 Member Data Documentation

6.11.3.1 std::uniform_int_distribution qpp::UniformIntDistribution::_d [protected]

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

- include/classes/stat.h

6.12 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

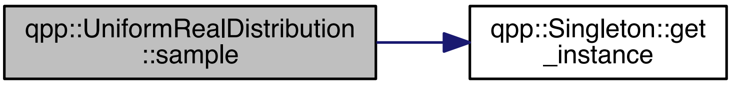
6.12.1 Constructor & Destructor Documentation

6.12.1.1 qpp::UniformRealDistribution::UniformRealDistribution (double *a* = 0, double *b* = 1) [inline]

6.12.2 Member Function Documentation

6.12.2.1 double qpp::UniformRealDistribution::sample ( ) [inline]

Here is the call graph for this function:



6.12.3 Member Data Documentation

6.12.3.1 std::uniform_real_distribution qpp::UniformRealDistribution::_d [protected]

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

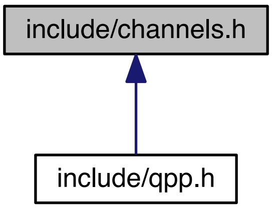
- include/classes/stat.h

Chapter 7

File Documentation

7.1 include/channels.h File Reference

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



Namespaces

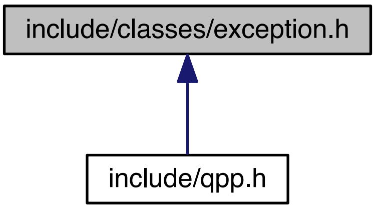
- qpp

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

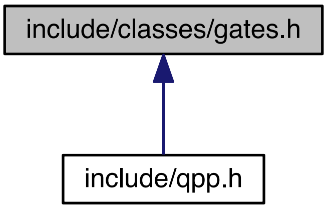
- class qpp::Exception

Namespaces

- qpp

7.3 include/classes/gates.h File Reference

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



Classes

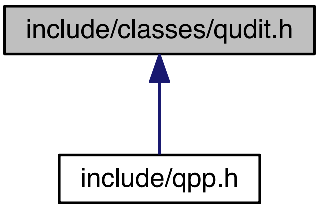
- class qpp::Gates

Namespaces

- qpp

7.4 include/classes/qudit.h File Reference

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



Classes

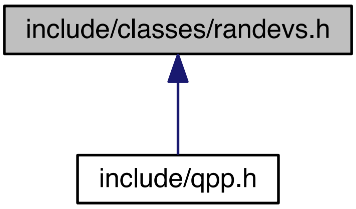
- class qpp::Qudit

Namespaces

- qpp

7.5 include/classes/randevs.h File Reference

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



Classes

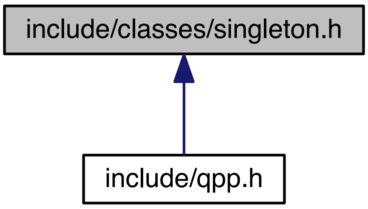
- class qpp::RandomDevices

Namespaces

- qpp

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>\
{\
    friend class Singleton<const Foo>;
```

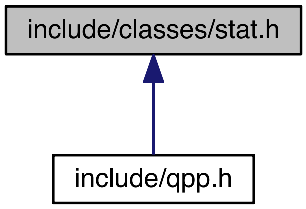
7.6.1.2 #define CLASS_SINGLETON(*Foo*)

Value:

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

7.7 include/classes/stat.h File Reference

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



Classes

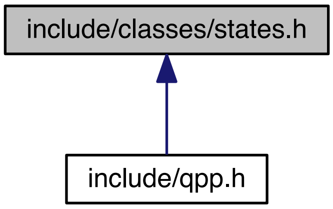
- class qpp::NormalDistribution
- class qpp::UniformRealDistribution
- class qpp::UniformIntDistribution
- class qpp::DiscreteDistribution
- class qpp::DiscreteDistributionAbsSquare

Namespaces

- qpp

7.8 include/classes/states.h File Reference

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



Classes

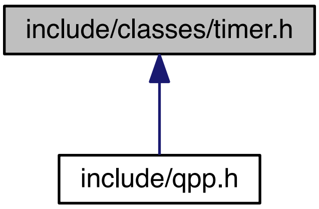
- class qpp::States

Namespaces

- qpp

7.9 include/classes/timer.h File Reference

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



Classes

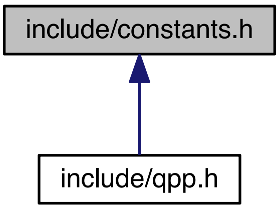
- class qpp::Timer

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

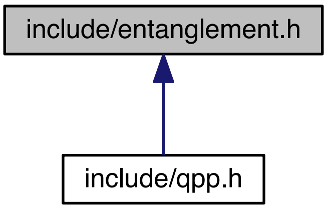
- constexpr std::complex< double > qpp::operator""_i (unsigned long long int x)
- constexpr std::complex< double > qpp::operator""_i (long double x)
- std::complex< double > qpp::ct::omega (std::size_t D)

Variables

- constexpr double qpp::ct::chop = 1e-10
- constexpr double qpp::ct::eps = 1e-12
- constexpr std::size_t qpp::ct::maxn = 64
- constexpr double qpp::ct::pi = 3.141592653589793238462643383279502884
- constexpr 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

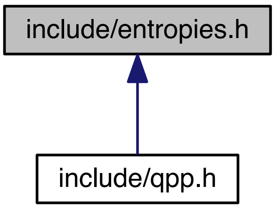
Functions

- 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::gconcurrence (const Eigen::MatrixBase< Derived > &A)

7.12 include/entropies.h File Reference

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



Namespaces

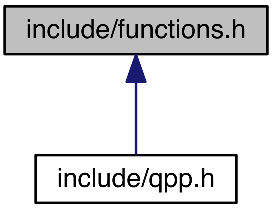
- qpp

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

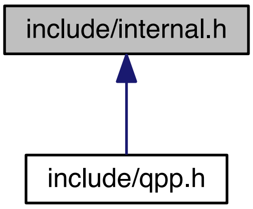
- template<typename Derived >
  types::DynMat< typename
  Derived::Scalar > qpp::transpose (const Eigen::MatrixBase< Derived > &A)
- template<typename Derived >
  types::DynMat< typename
  Derived::Scalar > qpp::conjugate (const Eigen::MatrixBase< Derived > &A)
- template<typename Derived >
  types::DynMat< 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::evects (const Eigen::MatrixBase< Derived > &A)
- template<typename Derived >
  types::dmat qpp::hevals (const Eigen::MatrixBase< Derived > &A)
- template<typename Derived >
  types::cmat qpp::hevects (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::spectralpowm (const Eigen::MatrixBase< Derived > &A, const types::cplx z)
- template<typename Derived >
  types::DynMat< typename
  Derived::Scalar > qpp::powm (const Eigen::MatrixBase< Derived > &A, std::size_t n)
- template<typename OutputScalar , typename Derived >
  types::DynMat< OutputScalar > qpp::cwise (const Eigen::MatrixBase< Derived > &A, Output↩
  Scalar(∗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

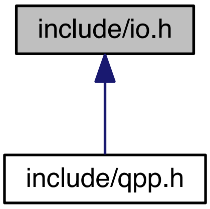
- qpp
- qpp::internal

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

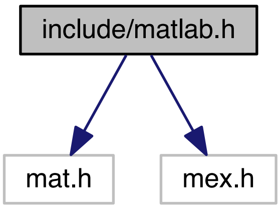
Functions

- 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

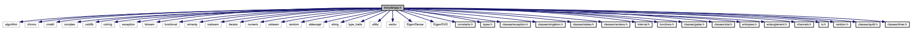
Functions

- 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< 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 <exception>
#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

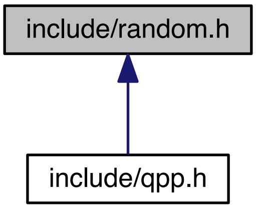
- qpp

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

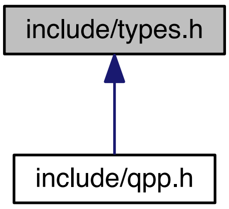
- qpp

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