## Master Thesis

Generated by Doxygen 1.8.4

Sun Jul 7 2013 15:26:06

# **Contents**

1	Nam	nespace Index	1
	1.1	Namespace List	1
2	Hiera	archical Index	3
	2.1	Class Hierarchy	3
3	Clas	es Index	5
	3.1	Class List	5
4	File	Index	7
	4.1	File List	7
5	Nam	nespace Documentation	9
	5.1	PRPSEvolution Namespace Reference	9
		5.1.1 Detailed Description	9
6	Clas	es Documentation	11
	6.1	$\label{eq:problem} {\sf PRPSEvolution::Permutate::AntennaPermutations} < {\sf N\_MAT, T} > {\sf Struct\ Template\ Reference}  .  .  .$	11
	6.2	PRPSEvolution::Exceptions::Permutation::Calculation Class Reference	11
	6.3	PRPSEvolution::Exeptions::Permutation::Calculation Class Reference	11
	6.4	PRPSEvolution::Constants Struct Reference	12
	6.5	$\label{eq:problem} PRPSE volution:: Positioning:: Coord Container < N, T > Struct \ Template \ Reference \ \dots \dots \dots \dots$	12
	6.6	PRPSEvolution::Exceptions::FileIO::FileNotFound Class Reference	12
	6.7	PRPSEvolution::Exeptions::FileIO::FileNotFound Class Reference	13
	6.8	PRPSEvolution::Exceptions::Permutation::Init Class Reference	13
	6.9	PRPSEvolution::Exeptions::Permutation::Init Class Reference	13
	6.10	PRPSEvolution::Exeptions::FileIO::MalformedInput Class Reference	13
	6.11	PRPSEvolution::Exceptions::FileIO::MalformedInput Class Reference	13
	6.12	$\label{eq:problem} \mbox{PRPSEvolution::Normalizer} < \mbox{N, T} > \mbox{Struct Template Reference}  .  .  .  .  .  .  .  .  .  $	13
		6.12.1 Constructor & Destructor Documentation	14
		6.12.1.1 Normalizer	14
		6.12.2 Member Function Documentation	14
		6 12 2 1 normalize	14

iv CONTENTS

6.13	PRPSE	Evolution::Exceptions::General::NotImplemented Struct Reference	14
	6.13.1	Detailed Description	14
6.14	PRPSE	Evolution::Exceptions::FileIO::Output Class Reference	15
6.15	PRPSE	Evolution::Exeptions::FileIO::Output Class Reference	15
6.16		Evolution::Calibration::performCalibration < N_ANTA, N_CALPOS, T > Struct Template Ref-	
			15
		·	15
6.17		Evolution::Permutate::permuteAntennas< N_ANTA, N_ANTPERM, T > Struct Template Ref	15
			16
		•	16
			16
			16
			17
			17
6.18	PRPSE	Evolution::Solve::PostProcessing Class Reference	17
		Evolution::Solve::PreProcessing < N_ANTA, N_Configs, T, T_Measure > Class Template Ref-	
	erence		17
	6.19.1	Constructor & Destructor Documentation	17
		6.19.1.1 PreProcessing	17
	6.19.2	Member Data Documentation	18
		6.19.2.1 matricesForSolution	18
6.20	PRPSE	Evolution::Solve::ProblemDimensions Struct Reference	18
6.21	PRPSE	Evolution::Solve::Process Class Reference	18
	6.21.1	Detailed Description	18
	6.21.2	Constructor & Destructor Documentation	19
		6.21.2.1 Process	19
	6.21.3	Member Function Documentation	19
		6.21.3.1 findSolution	19
		6.21.3.2 setMinSolutionFitness	19
6.22	PRPSE	Evolution::Solve::solveresult_t< T_Store, T_Return > Struct Template Reference	19
	6.22.1	Detailed Description	19
	6.22.2	Member Data Documentation	20
		6.22.2.1 converged	20
		6.22.2.2 duration	20
		6.22.2.3 fitness	20
		6.22.2.4 iterations	20
			20
6.23		•	20
	6.23.1		20
		6.23.1.1 System	20

CONTENTS

	6.24	PRPSEvolution::Solve::Ueber9000< T > Struct Template Reference	21
		6.24.1 Detailed Description	21
		6.24.2 Constructor & Destructor Documentation	21
		6.24.2.1 Ueber9000	21
		6.24.2.2 Ueber9000	21
		6.24.3 Member Function Documentation	22
		6.24.3.1 fitnessAckley	22
		6.24.3.2 fitnessRosenbrock	22
		6.24.3.3 fitnessSphere	22
		6.24.3.4 PositionVariation	22
		6.24.3.5 WavenumberVariation	22
		6.24.3.6 WholeTomatoeApproach	22
		6.24.3.7 WholeTomatoeApproach	22
7	File I	Documentation	25
	7.1	trunk/include/PRPSError.h File Reference	25
		7.1.1 Detailed Description	25
	7.2	trunk/include/PRPSEvolution.h File Reference	25
		7.2.1 Detailed Description	26
	7.3	trunk/include/PRPSEvolutionFIOExeptions.h File Reference	26
		7.3.1 Detailed Description	27
	7.4	trunk/include/PRPSEvolutionGeneralExeptions.h File Reference	27
		7.4.1 Detailed Description	27
	7.5	trunk/include/PRPSEvolutionSolveExeptions.h File Reference	27
		7.5.1 Detailed Description	27
	7.6	trunk/libCalibration/calib.h File Reference	28
		7.6.1 Detailed Description	28
	7.7	trunk/libNormalizer/normalizer.h File Reference	28
		7.7.1 Detailed Description	29
	7.8	trunk/libPermutate/permutate.h File Reference	29
		7.8.1 Detailed Description	30
	7.9	trunk/libSolve/solve.h File Reference	30
		7.9.1 Detailed Description	31
	7.10	trunk/libSolve/solveresult.h File Reference	31
		7.10.1 Detailed Description	31
	7.11	trunk/test/AntennaConfiguration.cpp File Reference	32
		7.11.1 Detailed Description	32

33

Index

# **Chapter 1**

# Namespace Index

1.1	Namespace List
Here i	s a list of all documented namespaces with brief descriptions:

2 Namespace Index

# **Chapter 2**

# **Hierarchical Index**

## 2.1 Class Hierarchy

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

$PRPSE volution:: Permutate:: Antenna Permutations < N\_MAT, T > \dots \dots$
PRPSEvolution::Constants
PRPSEvolution::Positioning::CoordContainer $< N, T > \dots $ 12
PRPSEvolution::Positioning::CoordContainer< N_ANTA, T >
PRPSEvolution::Positioning::CoordContainer< N_CALPOS, T >
exception
PRPSEvolution::Exceptions::FileIO::FileNotFound
PRPSEvolution::Exceptions::FileIO::MalformedInput
PRPSEvolution::Exceptions::FileIO::Output
PRPSEvolution::Exceptions::General::NotImplemented
PRPSEvolution::Exceptions::Permutation::Calculation
PRPSEvolution::Exceptions::Permutation::Init
PRPSEvolution::Exeptions::FileIO::FileNotFound
PRPSEvolution::Exeptions::FileIO::MalformedInput
PRPSEvolution::Exeptions::FileIO::Output
PRPSEvolution::Exeptions::Permutation::Calculation
PRPSEvolution::Exeptions::Permutation::Init
PRPSEvolution::Normalizer $< N, T > \dots \dots$
PRPSEvolution::Calibration::performCalibration < N ANTA, N CALPOS, T >
PRPSEvolution::Permutate::permuteAntennas< N_ANTA, N_ANTPERM, T >
PRPSEvolution::Solve::PostProcessing
PRPSEvolution::Solve::PreProcessing < N_ANTA, N_Configs, T, T_Measure >
PRPSEvolution::Solve::ProblemDimensions
PRPSEvolution::Solve::Process
PRPSEvolution::Solve::solveresult_t< T_Store, T_Return >
PRPSEvolution::System
PRPSEvolution::Solve::Ueber9000 < T >

**Hierarchical Index** 

# **Chapter 3**

# **Class Index**

## 3.1 Class List

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

PRPSEvolution::Permutate::AntennaPermutations< N_MAT, T >
PRPSEvolution::Exceptions::Permutation::Calculation
PRPSEvolution::Exeptions::Permutation::Calculation
PRPSEvolution::Constants
$\label{eq:problem} PRPSE volution:: Positioning:: Coord Container < N, T > \dots \dots$
PRPSEvolution::Exceptions::FileIO::FileNotFound
PRPSEvolution::Exeptions::FileIO::FileNotFound
PRPSEvolution::Exceptions::Permutation::Init
PRPSEvolution::Exeptions::Permutation::Init
PRPSEvolution::Exeptions::FileIO::MalformedInput
PRPSEvolution::Exceptions::FileIO::MalformedInput
$PRPSE volution:: Normalizer < N, T > \dots \dots$
PRPSEvolution::Exceptions::General::NotImplemented
PRPSEvolution::Exceptions::FileIO::Output
PRPSEvolution::Exeptions::FileIO::Output
$\label{eq:problem} \mbox{PRPSEvolution::Calibration::performCalibration} < \mbox{N\_ANTA, N\_CALPOS, T} > \dots $
$PRPSE volution:: Permutate:: permute Antennas < N\_ANTA, N\_ANTPERM, T > \dots \dots$
PRPSEvolution::Solve::PostProcessing
PRPSEvolution::Solve::PreProcessing < N_ANTA, N_Configs, T, T_Measure >
PRPSEvolution::Solve::ProblemDimensions
PRPSEvolution::Solve::Process
PRPSEvolution::Solve::solveresult_t< T_Store, T_Return >
PRPSEvolution::System
PRPSEvolution::Solve::Ueber9000 < T >

6 Class Index

# Chapter 4

# File Index

## 4.1 File List

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

trunk/include/coords.h
trunk/include/prps.h
trunk/include/PRPSError.h
trunk/include/PRPSEvolution.h
trunk/include/PRPSEvolutionCalibrationExceptions.h
trunk/include/PRPSEvolutionCalibrationExeptions.h
trunk/include/PRPSEvolutionFIOExceptions.h
trunk/include/PRPSEvolutionFIOExeptions.h
trunk/include/PRPSEvolutionGeneralExceptions.h
trunk/include/PRPSEvolutionGeneralExeptions.h
trunk/include/PRPSEvolutionPermutationExceptions.h
trunk/include/PRPSEvolutionPermutationExeptions.h
trunk/include/PRPSEvolutionSolveExceptions.h
trunk/include/PRPSEvolutionSolveExeptions.h
trunk/libCalibration/calib.h
trunk/libNormalizer/normalizer.h
trunk/libPermutate/permutate.h
trunk/libPRPSSystem/prpsevolutionsystem.h
trunk/libSolve/solve.h
trunk/libSolve/solveresult.h
trunk/test/AntennaConfiguration.cpp
trunk/test/AntennaConfiguration.h

8 File Index

## **Chapter 5**

## **Namespace Documentation**

## 5.1 PRPSEvolution Namespace Reference

#### Classes

- struct Normalizer
- struct Constants
- struct System

#### **Enumerations**

enum NormalizatioMethodes { Native, B, CMPLX, RND }

#### **Variables**

- const int ANTENNA\_AMOUNT = 8
- const int EXPECTED\_LINES\_CALIBRATION\_FILE = 4
- const int EXPECTED\_VALUES CALIBRATION\_FILE = ANTENNA AMOUNT
- const int EXPECTED LINES COORD FILE = ANTENNA AMOUNT
- const int EXPECTED\_VALUES\_COORD\_FILE = 3
- const int **EXPECTED\_LINES\_SYSTEM\_INI\_FILE** = 2
- const int MAT\_ROWS = 3
- const int MAT\_COLS = 10
- const int CALIBRATION\_POINTS\_AVAILABLE = 4
- const int EXPECTED\_LINES\_MEASUREMENT\_FILE = ANTENNA\_AMOUNT
- const int EXPECTED\_VALUES\_MEASUREMENT\_FILE = 2
- const int **DATA\_NV** = 65535
- const int **DEFAULT\_MIN\_GROUP\_SIZE** = 4

## 5.1.1 Detailed Description

This file contains structures and classes belonging to the system itself

Names	pace	Docur	mentatior

## **Chapter 6**

## **Class Documentation**

# 6.1 PRPSEvolution::Permutate::AntennaPermutations < N\_MAT, T > Struct Template Reference

**Static Public Member Functions** 

- static void dump\_matrix (NRmatrix< T > mat)
- static void dump\_matrix\_2\_file (std::ofstream &f, NRmatrix< T > mat)

#### **Public Attributes**

- std::array< NRmatrix< T >, N\_MAT > mat
- std::array< std::string, N\_MAT > names

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

• trunk/libPermutate/permutate.h

## 6.2 PRPSEvolution::Exceptions::Permutation::Calculation Class Reference

Inheritance diagram for PRPSEvolution::Exceptions::Permutation::Calculation:

Collaboration diagram for PRPSEvolution::Exceptions::Permutation::Calculation:

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

· trunk/include/PRPSEvolutionPermutationExceptions.h

## 6.3 PRPSEvolution::Exeptions::Permutation::Calculation Class Reference

Inheritance diagram for PRPSEvolution::Exeptions::Permutation::Calculation:

Collaboration diagram for PRPSEvolution::Exeptions::Permutation::Calculation:

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

· trunk/include/PRPSEvolutionPermutationExeptions.h

12 Class Documentation

## 6.4 PRPSEvolution::Constants Struct Reference

#### **Public Member Functions**

Constants (const PRPSEvolution::Constants &c)

#### **Public Attributes**

- · double a\_1
- double a\_2
- · double lambda
- · double f mess
- double c\_0

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

• trunk/libPRPSSystem/prpsevolutionsystem.h

## 6.5 PRPSEvolution::Positioning::CoordContainer < N, T > Struct Template Reference

Collaboration diagram for PRPSEvolution::Positioning::CoordContainer< N, T >:

## **Public Types**

• typedef T value\_type

## **Public Member Functions**

- template<typename T1 > CoordContainer (T1 init)
- T & operator[] (std::size\_t i)

## **Public Attributes**

- T x\_[N]
- T **y**\_[N]
- T **z**\_[N]

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

· trunk/include/coords.h

## 6.6 PRPSEvolution::Exceptions::FileIO::FileNotFound Class Reference

Inheritance diagram for PRPSEvolution::Exceptions::FileIO::FileNotFound:

Collaboration diagram for PRPSEvolution::Exceptions::FileIO::FileNotFound:

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

trunk/include/PRPSEvolutionFIOExceptions.h

## 6.7 PRPSEvolution::Exeptions::FileIO::FileNotFound Class Reference

Inheritance diagram for PRPSEvolution::Exeptions::FileIO::FileNotFound:

Collaboration diagram for PRPSEvolution::Exeptions::FileIO::FileNotFound:

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

· trunk/include/PRPSEvolutionFIOExeptions.h

## 6.8 PRPSEvolution::Exceptions::Permutation::Init Class Reference

Inheritance diagram for PRPSEvolution::Exceptions::Permutation::Init:

Collaboration diagram for PRPSEvolution::Exceptions::Permutation::Init:

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

· trunk/include/PRPSEvolutionPermutationExceptions.h

## 6.9 PRPSEvolution::Exeptions::Permutation::Init Class Reference

Inheritance diagram for PRPSEvolution::Exeptions::Permutation::Init:

Collaboration diagram for PRPSEvolution::Exeptions::Permutation::Init:

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

• trunk/include/PRPSEvolutionPermutationExeptions.h

## 6.10 PRPSEvolution::Exeptions::FileIO::MalformedInput Class Reference

 $Inheritance\ diagram\ for\ PRPSEvolution:: Exeptions:: File IO:: Malformed Input:$ 

Collaboration diagram for PRPSEvolution::Exeptions::FileIO::MalformedInput:

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

• trunk/include/PRPSEvolutionFIOExeptions.h

## 6.11 PRPSEvolution::Exceptions::FileIO::MalformedInput Class Reference

Inheritance diagram for PRPSEvolution::Exceptions::FileIO::MalformedInput:

Collaboration diagram for PRPSEvolution::Exceptions::FileIO::MalformedInput:

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

• trunk/include/PRPSEvolutionFIOExceptions.h

## 6.12 PRPSEvolution::Normalizer < N, T > Struct Template Reference

**Public Member Functions** 

Normalizer (NormalizatioMethodes method)

14 Class Documentation

- std::array< T, N > complexNorm (const std::array< T, N > &p, const std::array< T, N > &a)
- std::array< T, N > randNorm ()
- std::array< T, N > normalize (std::array< T, N > phase, std::array< T, N > amp)

#### **Public Attributes**

· NormalizatioMethodes Method

#### 6.12.1 Constructor & Destructor Documentation

```
6.12.1.1 template<std::size_t N, typename T > PRPSEvolution::Normalizer< N, T >::Normalizer ( NormalizatioMethodes method ) [inline]
```

#### Constructor

#### **Parameters**

	., ,	
in	method	Selects the Normalization function

#### 6.12.2 Member Function Documentation

```
6.12.2.1 template < std::size_t N, typename T > std::array < T, N > PRPSEvolution::Normalizer < N, T >::normalize ( std::array < T, N > phase, std::array < T, N > amp ) [inline]
```

#### Calculates the normalizations

#### **Parameters**

in	phase	The measured phase data
in	атр	The measured amplitude data

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

• trunk/libNormalizer/normalizer.h

## 6.13 PRPSEvolution::Exceptions::General::NotImplemented Struct Reference

#include <PRPSEvolutionGeneralExceptions.h>

Inheritance diagram for PRPSEvolution::Exceptions::General::NotImplemented:

 $Collaboration\ diagram\ for\ PRPS Evolution :: Exceptions :: General :: Not Implemented :: Not Implemented$ 

#### **Public Member Functions**

• const char \* what () const noexcept

## 6.13.1 Detailed Description

Throw this if a Method is not implemented

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

trunk/include/PRPSEvolutionGeneralExceptions.h

## 6.14 PRPSEvolution::Exceptions::FileIO::Output Class Reference

Inheritance diagram for PRPSEvolution::Exceptions::FileIO::Output:

Collaboration diagram for PRPSEvolution::Exceptions::FileIO::Output:

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

· trunk/include/PRPSEvolutionFIOExceptions.h

## 6.15 PRPSEvolution::Exeptions::FileIO::Output Class Reference

Inheritance diagram for PRPSEvolution::Exeptions::FileIO::Output:

Collaboration diagram for PRPSEvolution::Exeptions::FileIO::Output:

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

• trunk/include/PRPSEvolutionFIOExeptions.h

# 6.16 PRPSEvolution::Calibration::performCalibration < N\_ANTA, N\_CALPOS, T > Struct Template Reference

#include <calib.h>

## 6.16.1 Detailed Description

 $template < std::size\_t \ N\_ANTA, \ std::size\_t \ N\_CALPOS, \ typename \ T> struct \ PRPSEvolution::Calibration::performCalibration < N\_ANTA, \ N\_CALPOS, \ T>$ 

This will perform the calibration stuff

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

• trunk/libCalibration/calib.h

# 6.17 PRPSEvolution::Permutate::permuteAntennas< N\_ANTA, N\_ANTPERM, T > Struct Template Reference

#include <permutate.h>

Collaboration diagram for PRPSEvolution::Permutate::permuteAntennas< N\_ANTA, N\_ANTPERM, T >:

## **Public Member Functions**

- permuteAntennas (const PRPSEvolution::Constants c)
- int rCoordFile ()
- int computePermutations (const PRPSEvolution::Constants &co)
- template<std::size\_t NN, std::size\_t MM>
   const NRmatrix< T > computeMatrix (const int ref, const int a1, const int a2, const int a3, const PRPS Evolution::Constants &co)
- NRmatrix< T > compute\_d\_k0\_Mat ()
- void dumpConfigurationsToFile ()
- void dump\_matrices\_2\_file ()

16 Class Documentation

#### **Public Attributes**

- int ref
- PRPSEvolution::Constants systemConstants
- · Positioning::CoordContainer
  - < N\_ANTA, T > AntennaCoordinates
- std::array
  - < AntennaPermutations
  - < N\_ANTPERM, Doub >, N\_ANTA > configurations
- NRmatrix $< T > d_k0_mat$

## 6.17.1 Detailed Description

template < std::size\_t N\_ANTA, std::size\_t N\_ANTPERM, typename T> struct PRPSEvolution::Permutate::permuteAntennas < N\_ANTA, N\_ANTPERM, T>

This will collect some stuff for calculating the permutation of the antennas

### 6.17.2 Member Function Documentation

6.17.2.1 template < std::size\_t N\_ANTA, std::size\_t N\_ANTPERM, typename T > template < std::size\_t NN, std::size\_t MM > const NRmatrix < T > PRPSEvolution::Permutate::permuteAntennas < N\_ANTA, N\_ANTPERM, T > ::computeMatrix ( const int ref, const int a1, const int a2, const int a3, const PRPSEvolution::Constants & co

This method will compute all the possible permutations based on the given reference antenna

### See Also

ref

#### **Parameters**

in	ref	The reference antenna
in	a1	First antenna
in	a2	Second antenna
in	аЗ	Third antenna

6.17.2.2 template < std::size\_t N\_ANTA, std::size\_t N\_ANTPERM, typename T > int PRPSEvolution::Permutate::permuteAntennas < N\_ANTA, N\_ANTPERM, T >::computePermutations ( const PRPSEvolution::Constants & co )

This method handles the computation of the antenna permutations

#### **Parameters**

in	со	Constant structure with the system constants we need

### See Also

PRPSEvolution::Constants

```
6.17.2.3 template < std::size_t N_ANTA, std::size_t N_ANTPERM, typename T > void PRPSEvolution-
::Permutate::permuteAntennas < N_ANTA, N_ANTPERM, T >::dump_matrices_2_file (
)
```

This method will dump all the Antennas to an output file

```
6.17.2.4 template<std::size_t N_ANTA, std::size_t N_ANTPERM, typename T > int PRPSEvolution::Permutate-
::permuteAntennas< N_ANTA, N_ANTPERM, T >::rCoordFile ( )
```

Load the csv-file containing the coordinates and store it into the container.

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

• trunk/libPermutate/permutate.h

## 6.18 PRPSEvolution::Solve::PostProcessing Class Reference

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

• trunk/libSolve/solve.h

# 6.19 PRPSEvolution::Solve::PreProcessing< N\_ANTA, N\_Configs, T, T\_Measure > Class Template Reference

**Public Member Functions** 

 PreProcessing (const std::array< AntennaPermutations< Permutate::MAX\_PERMUTATION\_AMOUNT, Doub >, N\_ANTA > &, const NRmatrix< T > &)

#### **Public Attributes**

- std::vector< NRmatrix< T >> matricesForSolution
- std::vector< NRvector< T >> vectorsForSolution
- $\bullet \ \, {\sf std::vector} {<} \ \, {\sf std::string} > {\sf ConfigurationNames}$

#### 6.19.1 Constructor & Destructor Documentation

6.19.1.1 template < std::size\_t N\_ANTA, std::size\_t N\_Configs, typename T , typename T\_Measure > PRPSEvolution::Solve::PreProcessing < N\_ANTA, N\_Configs, T, T\_Measure >::PreProcessing ( const std::array < AntennaPermutations < Permutate::MAX\_PERMUTATION\_AMOUNT, Doub >, N\_ANTA > & precalculatedMatrices, const NRmatrix < T > &  $d_k$ 0s )

Construct the object an perform neccessary PreProcessing steps.

- 1. Read out the measurements from the given interface (e.g. a file)
- 2. Normalize everything
- 3. Select the matrices for further processing
- 4. Fill the matrices with the information

18 Class Documentation

5. Precalculate the

 $c_{k0}$ 

- -Vector
- 6. Store matrices to make them available in the next steps

#### 6.19.2 Member Data Documentation

6.19.2.1 template < std::size\_t N\_ANTA, std::size\_t N\_Configs, typename T, typename T\_Measure > std::vector < NRmatrix < T > > PRPSEvolution::Solve::PreProcessing < N\_ANTA, N\_Configs, T, T\_Measure > ::matricesForSolution

determines how many Configurations will solved for the Wavenumber

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

• trunk/libSolve/solve.h

## 6.20 PRPSEvolution::Solve::ProblemDimensions Struct Reference

#### **Static Public Attributes**

- static const int WholeTomatoeApproach = 7
- static const int **Sphere** = 10
- static const int Rosenbrock = 15

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

• trunk/libSolve/solve.h

## 6.21 PRPSEvolution::Solve::Process Class Reference

#include <solve.h>

#### **Public Member Functions**

- Process ()
- Process (const Process &p)
- double getLastSolutionFitness ()
- template<typename T >
   T findSolution (const NRmatrix< Doub > &A\_selected, const NRvector< Doub > &b\_selected, ESStrategy strategy, int seed)
- int sq (int i)
- · void setMinSolutionFitness (double value)
- void setSeed (unsigned int value)

## 6.21.1 Detailed Description

Find solutions for the possible matrices

## 6.21.2 Constructor & Destructor Documentation

**6.21.2.1 PRPSEvolution::Solve::Process::Process()** [inline]

a Pointer to the object containing the fitness functions Constructor

#### 6.21.3 Member Function Documentation

6.21.3.1 template<typename T > T PRPSEvolution::Solve::Process::findSolution ( const NRmatrix< Doub > & A\_selected, const NRvector< Doub > & b\_selected, ESStrategy strategy, int seed ) [inline]

## Set the ES-Strategy

#### **Parameters**

in	Strategy	The selected strategy Find a Solution for a given pair of matrices
in	A_selected	The matrix A to use in this solution
in	b_selected	The c_k0' vector for this solution

#### Returns

The solution

6.21.3.2 void PRPSEvolution::Solve::Process::setMinSolutionFitness ( double value ) [inline]

Sets the min. solution fitness we want to achieve.

#### **Parameters**

in	value	The new value for the solution fitness

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

• trunk/libSolve/solve.h

# 6.22 PRPSEvolution::Solve::solveresult\_t< T\_Store, T\_Return > Struct Template Reference

#include <solveresult.h>

#### **Public Attributes**

- T\_Store values
- T Return fitness
- int iterations
- int duration
- bool converged

## 6.22.1 Detailed Description

template<typename T\_Store, typename T\_Return>struct PRPSEvolution::Solve::solveresult\_t< T\_Store, T\_Return>

Stores the final state of a solution

20 Class Documentation

#### 6.22.2 Member Data Documentation

 $6.22.2.1 \quad template < typename \ T\_Store \ , typename \ T\_Return > bool \ PRPSEvolution::Solve::solveresult\_t < T\_Store, \\ T\_Return > ::converged$ 

Indicates whether the build in convergence criterium was applied, or not

 $\begin{array}{ll} \textbf{6.22.2.2} & \textbf{template} < \textbf{typename} \ \textbf{T\_Store} \ , \ \textbf{typename} \ \textbf{T\_Return} > \textbf{int} \ \textbf{PRPSEvolution::Solve::solveresult\_t} < \textbf{T\_Store}, \\ \textbf{T\_Return} > \textbf{::duration} \\ \end{array}$ 

The processing time for this solution

6.22.2.3 template < typename T\_Store , typename T\_Return > T\_Return PRPSEvolution::Solve::solveresult\_t < T\_Store, T\_Return >::fitness

The fitness value

 $\begin{array}{ll} \textbf{6.22.2.4} & \textbf{template} < \textbf{typename} \ \textbf{T\_Store} \ , \ \textbf{typename} \ \textbf{T\_Return} > \textbf{int} \ \textbf{PRPSEvolution::Solve::solveresult\_t} < \textbf{T\_Store}, \\ \textbf{T\_Return} > \textbf{::iterations} \\ \end{array}$ 

The amount of iterations needed for this result

Whrere the result is stored

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

• trunk/libSolve/solveresult.h

## 6.23 PRPSEvolution::System Struct Reference

Collaboration diagram for PRPSEvolution::System:

**Public Member Functions** 

- System (const PRPSEvolution::System &s)
- int rPRPSIniFile ()

**Public Attributes** 

- PRPSEvolution::Constants constants
- std::string fn

## 6.23.1 Constructor & Destructor Documentation

**6.23.1.1** PRPSEvolution::System (const PRPSEvolution::System & s) [inline]

copy constructor

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

• trunk/libPRPSSystem/prpsevolutionsystem.h

## 6.24 PRPSEvolution::Solve::Ueber9000 < T > Struct Template Reference

#include <solve.h>

#### **Public Member Functions**

- Ueber9000 ()
- **Ueber9000** (const **Ueber9000** &me)
- Ueber9000 (const NRmatrix< T > A\_selected, const NRvector< T > c\_k0\_selected)
- double WholeTomatoeApproach (const ChromosomeT< double > &x)
- double WholeTomatoeApproach (const NRmatrix< T > &A, const ChromosomeT< double > &x, const N-Rvector< T > &b)
- double WavenumberVariation (const ChromosomeT< double > &n)
- double PositionVariation (const ChromosomeT< double > &pos)
- double fitnessSphere (const ChromosomeT< double > &c)
- double fitnessRosenbrock (const ChromosomeT< double > &c)
- double fitnessAckley (const std::vector< double > &x)

#### **Public Attributes**

- double(Ueber9000 < double >::\* evaluate )(const ChromosomeT < double > &)
- int Dimension
- NRmatrix< T > A
- NRvector< T > c\_k0

### 6.24.1 Detailed Description

template < typename T> struct PRPSE volution::Solve::Ueber 9000 < T>

Collect the fitness functions. Make sure they are static so we can function-pointer to them.

#### 6.24.2 Constructor & Destructor Documentation

6.24.2.1 template < typename T > PRPSEvolution::Solve::Ueber9000 < T >::Ueber9000 ( ) [inline]

Default constructor

6.24.2.2 template < typename T > PRPSEvolution::Solve::Ueber9000 < T >::Ueber9000 ( const NRmatrix < T > A\_selected, const NRvector < T > c\_k0\_selected) [inline]

Construct Ueber9000 to use the WholeTomatoeApproach as fitness function

#### **Parameters**

in	A_selected	The matrix A for this Solution
in	c_k0_selected	The vector c_k0 for this Solution

22 Class Documentation

#### 6.24.3 Member Function Documentation

6.24.3.1 template<typename T> double PRPSEvolution::Solve::Ueber9000< T>::fitnessAckley ( const std::vector< double > & x ) [inline]

The infamous Ackley-function

6.24.3.2 template<typename T> double PRPSEvolution::Solve::Ueber9000< T>::fitnessRosenbrock ( const ChromosomeT< double > & c ) [inline]

The Rosenbrock implementation

6.24.3.3 template < typename T > double PRPSEvolution::Solve::Ueber9000 < T >::fitnessSphere ( const ChromosomeT < double > & c ) [inline]

This ist the fitness function used in the EA algorithm

6.24.3.4 template < typename T > double PRPSEvolution::Solve::Ueber9000 < T >::PositionVariation ( const ChromosomeT < double > & pos ) [inline]

Approach 3 based on the thoughts of by S. Winter

6.24.3.5 template < typename T > double PRPSEvolution::Solve::Ueber9000 < T >::WavenumberVariation ( const ChromosomeT < double > & n) [inline]

Approach 2 based on the thoughts of S. Winter. Here we want to optimize the wavenumbers

6.24.3.6 template < typename T > double PRPSEvolution::Solve::Ueber9000 < T >::WholeTomatoeApproach ( const ChromosomeT < double > & x ) [inline]

This method basically wraps around the real WholeTomatoeApproach-function. Maps the function so that it can be used with the evaluate-method

### **Parameters**

in	X	The vector x
----	---	--------------

6.24.3.7 template < typename T > double PRPSEvolution::Solve::Ueber9000 < T >::WholeTomatoeApproach ( const NRmatrix < T > & A, const ChromosomeT < double > & x, const NRvector < T > & b) [inline]

This approach will solve calculate the 10x3 matrix described in the Master-Thesis of C.Gnip Basically solves the linear equation

$$r = \mathbf{A}\mathbf{x} - \mathbf{b}$$

### **Parameters**

in	Α	The 10x3 Matrix that ist used in this solution
in	X	The vector containing the variables
in	b	Representing the vector b

#### Returns

The residuum of the equation system representing the "Fitness" of the given Solution in

See Also

Х

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

• trunk/libSolve/solve.h

24 Class Documentation

## **Chapter 7**

## **File Documentation**

## 7.1 trunk/include/PRPSError.h File Reference

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

#### **Variables**

- const int PRPSError::FileIO::okay = 0
- const int PRPSError::FileIO::generalError = -1
- const int PRPSError::FileIO::fnf = -2
- const int PRPSError::FileIO::inputmalformed = -3
- const int PRPSError::okay = 0
- const int PRPSError::general = -1
- const int PRPSError::critical = 10

## 7.1.1 Detailed Description

## Date

2013|Jun|18 This file cantains definitions belonging to the PRPSError-namespace. It is split into subnamespaces for keeping thing nicely small.

## 7.2 trunk/include/PRPSEvolution.h File Reference

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

## **Namespaces**

PRPSEvolution

## **Constant Groups**

• PRPSEvolution

26 File Documentation

#### **Variables**

- const int PRPSEvolution::ANTENNA\_AMOUNT = 8
- const int PRPSEvolution::EXPECTED\_LINES\_CALIBRATION\_FILE = 4
- const int PRPSEvolution::EXPECTED\_VALUES\_CALIBRATION\_FILE = ANTENNA\_AMOUNT
- const int PRPSEvolution::EXPECTED\_LINES\_COORD\_FILE = ANTENNA\_AMOUNT
- const int PRPSEvolution::EXPECTED\_VALUES\_COORD\_FILE = 3
- const int PRPSEvolution::EXPECTED LINES SYSTEM INI FILE = 2
- const int PRPSEvolution::MAT\_ROWS = 3
- const int PRPSEvolution::MAT COLS = 10
- const int PRPSEvolution::CALIBRATION\_POINTS\_AVAILABLE = 4
- const int PRPSEvolution::EXPECTED LINES MEASUREMENT FILE = ANTENNA AMOUNT
- const int PRPSEvolution::EXPECTED\_VALUES\_MEASUREMENT\_FILE = 2
- const int PRPSEvolution::DATA\_NV = 65535
- const int PRPSEvolution::DEFAULT MIN GROUP SIZE = 4

#### 7.2.1 Detailed Description

#### Date

2013|Jun|18 This file collects definitions belonging to the PRPSEvolution-namespace. Especially const. defines.

## 7.3 trunk/include/PRPSEvolutionFIOExeptions.h File Reference

```
#include <iostream>
#include <exception>
```

Include dependency graph for PRPSEvolutionFIOExeptions.h:

## **Classes**

- · class PRPSEvolution::Exeptions::FileIO::FileNotFound
- class PRPSEvolution::Exeptions::FileIO::MalformedInput
- · class PRPSEvolution::Exeptions::FileIO::Output

## **Namespaces**

PRPSEvolution

## **Constant Groups**

PRPSEvolution

#### **Variables**

- PRPSEvolution::Exeptions::FileIO::FileNotFound PRPSEvolution::Exeptions::FileIO::FNFExeption
- PRPSEvolution::Exeptions::FileIO::MalformedInput PRPSEvolution::Exeptions::FileIO::MalformedInput-Exeption
- PRPSEvolution::Exeptions::FileIO::Output PRPSEvolution::Exeptions::FileIO::OutputExeption

## 7.3.1 Detailed Description

Exeptions of file IO

## 7.4 trunk/include/PRPSEvolutionGeneralExeptions.h File Reference

```
#include <iostream>
#include <exception>
#include <string>
Include dependency graph for PRPSEvolutionGeneralExeptions.h:
```

## **Namespaces**

PRPSEvolution

## **Constant Groups**

PRPSEvolution

## 7.4.1 Detailed Description

Contains the exeption used in the PRPSEvolution - Project

## 7.5 trunk/include/PRPSEvolutionSolveExeptions.h File Reference

```
#include <iostream>
#include <exception>
Include dependency graph for PRPSEvolutionSolveExeptions.h:
```

## **Namespaces**

PRPSEvolution

## **Constant Groups**

• PRPSEvolution

## 7.5.1 Detailed Description

Contains the exeption used in the PRPSEvolution - Project

28 File Documentation

## 7.6 trunk/libCalibration/calib.h File Reference

```
#include "nr3/nr3.h"
#include "nr3/svd.h"
#include "../include/coords.h"
#include "../include/PRPSEvolution.h"
#include "../include/PRPSEvolutionCalibrationExceptions.h"
#include "../include/PRPSError.h"
#include "../libPRPSSystem/prpsevolutionsystem.h"
Include dependency graph for calib.h: This graph shows which files directly or indirectly include this file:
```

## Classes

• struct PRPSEvolution::Calibration::performCalibration< N\_ANTA, N\_CALPOS, T >

## **Namespaces**

PRPSEvolution

## **Constant Groups**

PRPSEvolution

## 7.6.1 Detailed Description

Date

2013|Jun|25

## 7.7 trunk/libNormalizer/normalizer.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <iostream>
#include "../include/PRPSEvolutionGeneralExceptions.h"
#include "../include/PRPSEvolutionFIOExceptions.h"
#include <complex>
#include <array>
#include <random>
```

Include dependency graph for normalizer.h: This graph shows which files directly or indirectly include this file:

#### **Classes**

struct PRPSEvolution::Normalizer< N, T >

## **Namespaces**

PRPSEvolution

## **Constant Groups**

• PRPSEvolution

#### **Enumerations**

enum NormalizatioMethodes { Native, B, CMPLX, RND }

## 7.7.1 Detailed Description

Collects normalizations for the input data

## 7.8 trunk/libPermutate/permutate.h File Reference

```
#include <stdio.h>
#include <iterator>
#include <iostream>
#include <algorithm>
#include <array>
#include <string>
#include "../include/coords.h"
#include "../include/PRPSEvolution.h"
#include "../include/PRPSEvolutionPermutationExceptions.h"
#include "../include/PRPSEvolutionFIOExceptions.h"
#include "../include/PRPSEvolutionFIOExceptions.h"
#include "../include/PRPSError.h"
#include "../include/PRPSSystem/prpsevolutionsystem.h"
#include "nr3/nr3.h"
#include "nr3/svd.h"
```

Include dependency graph for permutate.h: This graph shows which files directly or indirectly include this file:

#### Classes

- struct PRPSEvolution::Permutate::AntennaPermutations < N\_MAT, T >
- struct PRPSEvolution::Permutate::permuteAntennas < N\_ANTA, N\_ANTPERM, T >

## **Namespaces**

PRPSEvolution

### **Constant Groups**

• PRPSEvolution

#### **Functions**

- int PRPSEvolution::Permutate::Factorial (int x)
- template<typename Iterator >

bool PRPSEvolution::Permutate::next\_combination (const Iterator first, Iterator k, const Iterator last)

#### **Variables**

const int PRPSEvolution::Permutate::MAX\_PERMUTATION\_AMOUNT = 35

30 File Documentation

## 7.8.1 Detailed Description

Date

2013|Jun|25

## 7.9 trunk/libSolve/solve.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <iterator>
#include <iostream>
#include <algorithm>
#include <array>
#include <chrono>
#include <string>
#include <random>
#include "../include/coords.h"
#include "../include/PRPSEvolution.h"
#include "../include/PRPSEvolutionSolveExceptions.h"
#include "../include/PRPSEvolutionFIOExceptions.h"
#include "../include/PRPSEvolutionGeneralExceptions.h"
#include "../include/PRPSError.h"
#include "../libPermutate/permutate.h"
#include "../libPRPSSystem/prpsevolutionsystem.h"
#include "../libNormalizer/normalizer.h"
#include "nr3/nr3.h"
#include "nr3/svd.h"
#include <EALib/ChromosomeCMA.h>
#include <SharkDefs.h>
#include <EALib/PopulationT.h>
#include <EALib/ObjectiveFunction.h>
#include "solveresult.h"
```

Include dependency graph for solve.h: This graph shows which files directly or indirectly include this file:

## **Classes**

- struct PRPSEvolution::Solve::ProblemDimensions
- struct PRPSEvolution::Solve::Ueber9000< T >
- class PRPSEvolution::Solve::PreProcessing
   N\_ANTA, N\_Configs, T, T\_Measure
- class PRPSEvolution::Solve::Process
- · class PRPSEvolution::Solve::PostProcessing

#### **Namespaces**

PRPSEvolution

## **Constant Groups**

PRPSEvolution

#### **Enumerations**

enum SelectBy { ConditionNumber, Random, AllPossible }

 enum ESStrategy { PRPSEvolution::Solve::OnePlusOne, PRPSEvolution::Solve::MuPlusLambda, PR-PSEvolution::Solve::MuCommaLambda }

## **Variables**

• const int PRPSEvolution::Solve::nConfigsForProcessing = 1

## 7.9.1 Detailed Description

Date

2013|Jun|25

## 7.10 trunk/libSolve/solveresult.h File Reference

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

## Classes

• struct PRPSEvolution::Solve::solveresult\_t< T\_Store, T\_Return >

## **Namespaces**

PRPSEvolution

## **Constant Groups**

PRPSEvolution

## 7.10.1 Detailed Description

32 File Documentation

Date

2013|Jul|5

## 7.11 trunk/test/AntennaConfiguration.cpp File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <array>
#include <iostream>
#include <exception>
#include <fstream>
#include <sstream>
#include <string>
#include <chrono>
#include <future>
#include <thread>
#include "AntennaConfiguration.h"
#include "../libPermutate/permutate.h"
#include "../libPRPSSystem/prpsevolutionsystem.h"
#include "../libCalibration/calib.h"
#include "../libSolve/solve.h"
#include "../libSolve/solveresult.h"
#include "../include/PRPSEvolutionGeneralExceptions.h"
#include <EALib/ChromosomeCMA.h>
Include dependency graph for AntennaConfiguration.cpp:
```

## **Functions**

• int main (int argc, char \*argv[])

#### **Variables**

• const int **SOLUTION\_AMOUNT** = 100

## 7.11.1 Detailed Description

This File contains the main() of the AntennaApp-Project

# Index

computeMatrix	normalize, 14
PRPSEvolution::Permutate::permuteAntennas, 16	Normalizer, 14
computePermutations	PRPSEvolution::Normalizer < N, T >, 13
PRPSEvolution::Permutate::permuteAntennas, 16	PRPSEvolution::Permutate::AntennaPermutations< N-
converged	_MAT, T $>$ , 11
PRPSEvolution::Solve::solveresult_t, 20	PRPSEvolution::Permutate::permuteAntennas
	computeMatrix, 16
dump_matrices_2_file	computePermutations, 16
PRPSEvolution::Permutate::permuteAntennas, 16	dump_matrices_2_file, 16
duration	rCoordFile, 17
PRPSEvolution::Solve::solveresult_t, 20	$\label{eq:problem} \mbox{PRPSEvolution::Positioning::CoordContainer} < \mbox{N, T} > ,$
findSolution	12
PRPSEvolution::Solve::Process, 19	PRPSEvolution::Solve::PostProcessing, 17
fitness	PRPSEvolution::Solve::PreProcessing
PRPSEvolution::Solve::solveresult_t, 20	matricesForSolution, 18
fitnessAckley	PreProcessing, 17
PRPSEvolution::Solve::Ueber9000, 22	PRPSEvolution::Solve::ProblemDimensions, 18
fitnessRosenbrock	PRPSEvolution::Solve::Process, 18
PRPSEvolution::Solve::Ueber9000, 22	findSolution, 19
fitnessSphere	Process, 19
PRPSEvolution::Solve::Ueber9000, 22	setMinSolutionFitness, 19
	PRPSEvolution::Solve::Ueber9000
iterations	fitnessAckley, 22
PRPSEvolution::Solve::solveresult_t, 20	fitnessRosenbrock, 22
	fitnessSphere, 22
matricesForSolution	PositionVariation, 22
PRPSEvolution::Solve::PreProcessing, 18	Ueber9000, 21
naveralina	WavenumberVariation, 22
normalize	WholeTomatoeApproach, 22
PRPSEvolution::Normalizer, 14	PRPSEvolution::Solve::Ueber9000< T >, 21
Normalizer  PRESENDITION: Normalizer 14	PRPSEvolution::Solve::solveresult_t
PRPSEvolution::Normalizer, 14	converged, 20
PRPSEvolution, 9	duration, 20
PRPSEvolution::Constants, 12	fitness, 20
PRPSEvolution::Exceptions::FileIO::FileNotFound, 12	iterations, 20
PRPSEvolution::Exceptions::FileIO::MalformedInput, 13	values, 20
PRPSEvolution::Exceptions::FileIO::Output, 15	PRPSEvolution::Solve::solveresult_t< T_Store, T
PRPSEvolution::Exceptions::General::NotImplemented,	Return >, 19
14	PRPSEvolution::System, 20
PRPSEvolution::Exceptions::Permutation::Calculation,	System, 20
11	PositionVariation
PRPSEvolution::Exceptions::Permutation::Init, 13	PRPSEvolution::Solve::Ueber9000, 22
PRPSEvolution::Exeptions::FileIO::FileNotFound, 13	PreProcessing
PRPSEvolution::Exeptions::FileIO::MalformedInput, 13	PRPSEvolution::Solve::PreProcessing, 17
PRPSEvolution::Exeptions::FileIO::Output, 15	Process
${\tt PRPSEvolution::Exeptions::Permutation::Calculation},$	PRPSEvolution::Solve::Process, 19
11	
PRPSEvolution::Exeptions::Permutation::Init, 13	rCoordFile
PRPSEvolution::Normalizer	PRPSEvolution::Permutate::permuteAntennas, 17

34 INDEX

```
setMinSolutionFitness
     PRPSEvolution::Solve::Process, 19
System
    PRPSEvolution::System, 20
trunk/include/PRPSError.h, 25
trunk/include/PRPSEvolution.h, 25
trunk/include/PRPSEvolutionFIOExeptions.h, 26
trunk/include/PRPSEvolutionGeneralExeptions.h, 27
trunk/include/PRPSEvolutionSolveExeptions.h, 27
trunk/libCalibration/calib.h, 28
trunk/libNormalizer/normalizer.h, 28
trunk/libPermutate/permutate.h, 29
trunk/libSolve/solve.h, 30
trunk/libSolve/solveresult.h, 31
trunk/test/AntennaConfiguration.cpp, 32
Ueber9000
     PRPSEvolution::Solve::Ueber9000, 21
values
     PRPSEvolution::Solve::solveresult_t, 20
WavenumberVariation
     PRPSEvolution::Solve::Ueber9000, 22
WholeTomatoeApproach
     PRPSEvolution::Solve::Ueber9000, 22
```