

B- $\rightarrow$ DK, D- $\rightarrow$ KK $\pi\pi$ , binned fit

Generated by Doxygen 1.9.0



<b>1 Class Index</b>	<b>1</b>
1.1 Class List	1
<b>2 File Index</b>	<b>3</b>
2.1 File List	3
<b>3 Class Documentation</b>	<b>5</b>
3.1 Amplitude Class Reference	5
3.1.1 Detailed Description	5
3.1.2 Constructor & Destructor Documentation	5
3.1.2.1 Amplitude()	5
3.1.3 Member Function Documentation	5
3.1.3.1 operator()	5
3.2 Bin Class Reference	6
3.2.1 Detailed Description	6
3.2.2 Constructor & Destructor Documentation	6
3.2.2.1 Bin()	6
3.2.3 Member Function Documentation	6
3.2.3.1 AddEvent()	6
3.2.3.2 GetEvents()	7
3.2.3.3 GetNumberEvents()	7
3.3 BinList Class Reference	8
3.3.1 Detailed Description	8
3.3.2 Constructor & Destructor Documentation	8
3.3.2.1 BinList()	8
3.3.3 Member Function Documentation	8
3.3.3.1 AddEvent() [1/2]	9
3.3.3.2 AddEvent() [2/2]	9
3.3.3.3 GetBin()	9
3.3.3.4 GetEvents()	10
3.3.3.5 LoadTTree()	10
3.3.3.6 NumberBins()	10
3.3.3.7 Predict()	11
3.4 CPPParameters Class Reference	11
3.4.1 Detailed Description	11
3.4.2 Constructor & Destructor Documentation	11
3.4.2.1 CPPParameters()	12
3.4.3 Member Function Documentation	12
3.4.3.1 GetCPPParameters()	12
3.4.3.2 GetError()	12
3.4.3.3 SetError()	13
3.5 DDecayParameters Class Reference	13
3.5.1 Detailed Description	14

3.5.2 Constructor & Destructor Documentation	14
3.5.2.1 DDecayParameters() [1/2]	14
3.5.2.2 DDecayParameters() [2/2]	14
3.5.3 Member Function Documentation	15
3.5.3.1 Getc()	15
3.5.3.2 GetK()	15
3.5.3.3 GetKbar()	15
3.5.3.4 Gets()	16
3.5.3.5 saveCSV()	16
3.6 Event Class Reference	16
3.6.1 Detailed Description	17
3.6.2 Constructor & Destructor Documentation	17
3.6.2.1 Event() [1/3]	17
3.6.2.2 Event() [2/3]	17
3.6.2.3 Event() [3/3]	17
3.6.3 Member Function Documentation	18
3.6.3.1 GetEvent()	18
3.6.3.2 GetEventVector()	18
3.6.3.3 GetInvMass2()	18
3.6.3.4 GetInvMass3()	19
3.7 EventList Class Reference	19
3.7.1 Detailed Description	19
3.7.2 Constructor & Destructor Documentation	20
3.7.2.1 EventList()	20
3.7.3 Member Function Documentation	20
3.7.3.1 AddEvent()	20
3.7.3.2 GetEvents()	20
3.7.3.3 NumberEvents()	21
3.8 Fitter Class Reference	21
3.8.1 Detailed Description	21
3.8.2 Constructor & Destructor Documentation	21
3.8.2.1 Fitter()	21
3.8.3 Member Function Documentation	22
3.8.3.1 DoFit()	22
3.9 Generator Class Reference	22
3.9.1 Detailed Description	22
3.9.2 Constructor & Destructor Documentation	22
3.9.2.1 Generator()	22
3.9.3 Member Function Documentation	23
3.9.3.1 Generate()	23
3.10 Likelihood Class Reference	23
3.10.1 Detailed Description	23

3.10.2 Constructor & Destructor Documentation	23
3.10.2.1 Likelihood()	23
3.10.3 Member Function Documentation	24
3.10.3.1 operator()()	24
3.11 PhaseSpaceParameterisation Class Reference	24
3.11.1 Detailed Description	25
3.11.2 Constructor & Destructor Documentation	25
3.11.2.1 PhaseSpaceParameterisation()	25
3.11.3 Member Function Documentation	25
3.11.3.1 NumberOfBins()	25
3.11.3.2 WhichBin()	25
<b>4 File Documentation</b>	<b>27</b>
4.1 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/Amplitude.h File Reference	27
4.2 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/Bin.h File Reference	27
4.3 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/BinList.h File Reference	27
4.4 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/CPPParameters.h File Reference	28
4.5 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/DDecayParameters.h File Reference	28
4.6 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/Event.h File Reference	28
4.7 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/EventList.h File Reference	28
4.8 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/FitGamma.h File Reference	29
4.9 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/Fitter.h File Reference	29
4.10 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/Generator.h File Reference	29
4.11 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/HadronicParameters.h File Reference	29
4.12 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/Likelihood.h File Reference	29
4.13 /data/lhcb/users/tat/KKpipi_Binned_Fit/include/PhaseSpaceParameterisation.h File Reference	30
4.14 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/Amplitude.cpp File Reference	30
4.15 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/Bin.cpp File Reference	30
4.16 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/BinList.cpp File Reference	30
4.17 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/CMakeLists.txt File Reference	31
4.18 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/CPPParameters.cpp File Reference	31
4.19 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/DDecayParameters.cpp File Reference	31
4.20 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/Event.cpp File Reference	31
4.21 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/EventList.cpp File Reference	31
4.22 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/Fitter.cpp File Reference	32
4.23 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/Generator.cpp File Reference	32
4.24 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/Likelihood.cpp File Reference	32
4.25 /data/lhcb/users/tat/KKpipi_Binned_Fit/src/PhaseSpaceParameterisation.cpp File Reference	32
<b>Index</b>	<b>33</b>



# Chapter 1

## Class Index

### 1.1 Class List

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

<a href="#">Amplitude</a>	5
<a href="#">Bin</a>	6
<a href="#">BinList</a>	8
<a href="#">CParameters</a>	11
<a href="#">DDecayParameters</a>	13
<a href="#">Event</a>	16
<a href="#">EventList</a>	19
<a href="#">Fitter</a>	21
<a href="#">Generator</a>	22
<a href="#">Likelihood</a>	23
<a href="#">PhaseSpaceParameterisation</a>	24





## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

/data/lhcb/users/tat/KKpipi_Binned_Fit/include/Amplitude.h	27
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/Bin.h	27
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/BinList.h	27
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/CPPParameters.h	28
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/DDecayParameters.h	28
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/Event.h	28
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/EventList.h	28
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/FitGamma.h	29
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/Fitter.h	29
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/Generator.h	29
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/HadronicParameters.h	29
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/Likelihood.h	29
/data/lhcb/users/tat/KKpipi_Binned_Fit/include/PhaseSpaceParameterisation.h	30
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/Amplitude.cpp	30
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/Bin.cpp	30
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/BinList.cpp	30
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/CPPParameters.cpp	31
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/DDecayParameters.cpp	31
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/Event.cpp	31
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/EventList.cpp	31
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/Fitter.cpp	32
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/Generator.cpp	32
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/Likelihood.cpp	32
/data/lhcb/users/tat/KKpipi_Binned_Fit/src/PhaseSpaceParameterisation.cpp	32



## Chapter 3

# Class Documentation

### 3.1 Amplitude Class Reference

```
#include <Amplitude.h>
```

#### Public Member Functions

- [Amplitude](#) (const std::string &Damplitude, const std::string &DBARamplitude)
- std::complex< double > [operator\(\)](#) (const std::vector< double > &event, int conj)

#### 3.1.1 Detailed Description

[Amplitude](#) is a class that loads the shared libraries generated by AmpGen and calculates the amplitude of events  
Definition at line 13 of file Amplitude.h.

#### 3.1.2 Constructor & Destructor Documentation

##### 3.1.2.1 Amplitude()

```
Amplitude::Amplitude (
    const std::string & Damplitude,
    const std::string & DBARamplitude )
```

Constructor that loads the shared library for D and Dbar decay amplitudes  
Definition at line 10 of file Amplitude.cpp.

#### 3.1.3 Member Function Documentation

##### 3.1.3.1 operator()

```
std::complex< double > Amplitude::operator() (
    const std::vector< double > & event,
    int conj )
```

Overload () operator to easily access amplitude

**Parameters**

<i>event</i>	Vector of four-momenta of event
<i>conj</i>	Set to +1 for $D^0$ decay and -1 for $\overline{D}^0$ decay

Definition at line 17 of file Amplitude.cpp.

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

- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Amplitude.h](#)
- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Amplitude.cpp](#)

## 3.2 Bin Class Reference

```
#include <Bin.h>
```

### Public Member Functions

- [Bin](#) ()
- void [AddEvent](#) ([Event](#) event, int charge)
- int [GetNumberEvents](#) (int charge) const
- [EventList](#) [GetEvents](#) (int charge)

### 3.2.1 Detailed Description

[Bin](#) is a class for a bin in phase space

Definition at line 11 of file Bin.h.

### 3.2.2 Constructor & Destructor Documentation

#### 3.2.2.1 Bin()

```
Bin::Bin ( )
```

Default constructor that creates an empty [EventList](#)

Definition at line 7 of file Bin.cpp.

### 3.2.3 Member Function Documentation

#### 3.2.3.1 AddEvent()

```
void Bin::AddEvent (
    Event event,
    int charge )
```

Function for adding an event

## Parameters

<i>event</i>	<a href="#">Event</a> to add
<i>charge</i>	+1 for B+, -1 for B-

Definition at line 10 of file Bin.cpp.

### 3.2.3.2 GetEvents()

```
EventList Bin::GetEvents (
    int charge )
```

Function for betting [EventList](#) object

## Parameters

<i>charge</i>	+1 for B+, -1 for B-
---------------	----------------------

## Returns

eventlist [EventList](#) object

Definition at line 27 of file Bin.cpp.

### 3.2.3.3 GetNumberEvents()

```
int Bin::GetNumberEvents (
    int charge ) const
```

Function for getting number of events in this bin

## Parameters

<i>charge</i>	+1 for B+, -1 for B-
---------------	----------------------

## Returns

Number of events in this bin

Definition at line 18 of file Bin.cpp.

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

- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Bin.h](#)
- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Bin.cpp](#)

### 3.3 BinList Class Reference

```
#include <BinList.h>
```

#### Public Member Functions

- [BinList](#) ([PhaseSpaceParameterisation](#) php)
- void [AddEvent](#) ([Event](#) event, int charge)
- void [AddEvent](#) ([Event](#) event, int charge, int maxevents)
- void [LoadTTree](#) (TTree \*tree, int charge)
- int [NumberBins](#) ()
- std::vector< int > [GetEvents](#) (int charge) const
- [Bin GetBin](#) (int i)
- void [Predict](#) (const [DDecayParameters](#) &ddparameters, const [CPPParameters](#) &cppparameters, std::vector< double > &BplusEvents, std::vector< double > &BminusEvents, int totalBplus, int totalBminus)

#### 3.3.1 Detailed Description

[BinList](#) is a class that contains all the bins in phase space [BinList](#) also loads the input data and puts it in their respective bins

Definition at line 17 of file [BinList.h](#).

#### 3.3.2 Constructor & Destructor Documentation

##### 3.3.2.1 [BinList\(\)](#)

```
BinList::BinList (
    PhaseSpaceParameterisation php )
```

Constructor that takes a [PhaseSpaceParameterisation](#) object and creates the bins

##### Parameters

<i>php</i>	A <a href="#">PhaseSpaceParameterisation</a> object that defines the bins in the 5D phase space
------------	---

Definition at line 14 of file [BinList.cpp](#).

#### 3.3.3 Member Function Documentation

### 3.3.3.1 AddEvent() [1/2]

```
void BinList::AddEvent (
    Event event,
    int charge )
```

Function for adding an event to the correct bin

#### Parameters

<i>event</i>	Event object to be added to the correct bin
<i>charge</i>	+1 for B+, -1 for B-

Definition at line 17 of file BinList.cpp.

### 3.3.3.2 AddEvent() [2/2]

```
void BinList::AddEvent (
    Event event,
    int charge,
    int maxevents )
```

Function for adding an event to the correct bin, if the number of events in that bin is less than the maximum

#### Parameters

<i>event</i>	Event object to be added to the correct bin
<i>charge</i>	+1 for B+, -1 for B-
<i>maxEvents</i>	Maximum number of events in each bin

Definition at line 21 of file BinList.cpp.

### 3.3.3.3 GetBin()

```
Bin BinList::GetBin (
    int i )
```

Function for getting Bin object

#### Parameters

<i>i</i>	Bin number
----------	------------

**Returns**

[Bin](#) object

Definition at line 65 of file BinList.cpp.

**3.3.3.4 GetEvents()**

```
std::vector< int > BinList::GetEvents (
    int charge ) const
```

Function for getting the number of events in each bin

**Parameters**

<i>charge</i>	+1 for B+, -1 for B-
---------------	----------------------

**Returns**

A vector of the number of events in each bin

Definition at line 57 of file BinList.cpp.

**3.3.3.5 LoadTTree()**

```
void BinList::LoadTTree (
    TTree * tree,
    int charge )
```

Function for loading events from input data into their respective bins

**Parameters**

<i>tree</i>	A ROOT TTree in the AmpGen format containing all the input data events
<i>charge</i>	+1 for B+, -1 for B-

Definition at line 28 of file BinList.cpp.

**3.3.3.6 NumberBins()**

```
int BinList::NumberBins ( )
```

Function for getting number of bins

Definition at line 53 of file BinList.cpp.



### 3.3.3.7 Predict()

```
void BinList::Predict (
    const DDecayParameters & ddparameters,
    const CParameters & cpparameters,
    std::vector< double > & BplusEvents,
    std::vector< double > & BminusEvents,
    int totalBplus,
    int totalBminus )
```

Function for calculating the number of events in each bin, given the D decay parameters and the CP parameters

#### Parameters

<i>ddparameters</i>	A <a href="#">DDecayParameters</a> object that describes the D meson decay
<i>cpparameters</i>	A <a href="#">CParameters</a> object that describes the CP violation in the B meson decay
<i>BplusEvents</i>	Vector of predicted number of B+ events
<i>BminusEvents</i>	Vector of predicted number of B- events
<i>totalBplus</i>	Total number of B+ events
<i>totalBminus</i>	Total number of B- events

Definition at line 69 of file BinList.cpp.

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

- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/BinList.h](#)
- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/BinList.cpp](#)

## 3.4 CParameters Class Reference

```
#include <CParameters.h>
```

### Public Member Functions

- [CParameters](#) (double xplus, double xminus, double yplus, double yminus)
- void [GetCParameters](#) (double &xplus, double &xminus, double &yplus, double &yminus) const
- void [SetError](#) (double xplus, double xminus, double yplus, double yminus)
- void [GetError](#) (double &xplus, double &xminus, double &yplus, double &yminus) const

### 3.4.1 Detailed Description

[CParameters](#) is a class that contains the CP parameters x and y

Definition at line 9 of file CParameters.h.

### 3.4.2 Constructor & Destructor Documentation

### 3.4.2.1 CParameters()

```
CParameters::CParameters (
    double xplus,
    double xminus,
    double yplus,
    double yminus )
```

Constructor that takes the CP parameters, x and y

#### Parameters

<i>xplus</i>	r_Bcos(delta_B + gamma) for Bplus decays
<i>xminus</i>	r_Bcos(delta_B - gamma) for Bminus decays
<i>yplus</i>	r_Bsin(delta_B + gamma) for Bplus decays
<i>yminus</i>	r_Bsin(delta - gamma) for Bminus decays

Definition at line 5 of file CParameters.cpp.

## 3.4.3 Member Function Documentation

### 3.4.3.1 GetCParameters()

```
void CParameters::GetCParameters (
    double & xplus,
    double & xminus,
    double & yplus,
    double & yminus ) const
```

Function for getting CP parameters

#### Parameters

<i>xplus</i>	r_Bcos(delta_B + gamma) for Bplus decays
<i>xminus</i>	r_Bcos(delta_B - gamma) for Bminus decays
<i>yplus</i>	r_Bsin(delta_B + gamma) for Bplus decays
<i>yminus</i>	r_Bsin(delta - gamma) for Bminus decays

Definition at line 8 of file CParameters.cpp.

### 3.4.3.2 GetError()

```
void CParameters::GetError (
    double & xplus,
```

```
double & xminus,  
double & yplus,  
double & yminus ) const
```

Function for getting CP parameter errors

#### Parameters

<i>xplus</i>	xplus error
<i>xminus</i>	xminus error
<i>yplus</i>	yplus error
<i>yminus</i>	yminus error

Definition at line 22 of file CParameters.cpp.

#### 3.4.3.3 SetError()

```
void CParameters::SetError (  
    double xplus,  
    double xminus,  
    double yplus,  
    double yminus )
```

Function for setting CP parameter errors

#### Parameters

<i>xplus</i>	xplus error
<i>xminus</i>	xminus error
<i>yplus</i>	yplus error
<i>yminus</i>	yminus error

Definition at line 15 of file CParameters.cpp.

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

- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/CParameters.h](#)
- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/CParameters.cpp](#)

## 3.5 DDecayParameters Class Reference

```
#include <DDecayParameters.h>
```

## Public Member Functions

- [DDecayParameters](#) (const [PhaseSpaceParameterisation](#) &psp, const double &mass\_parent, const double \*mass\_decay, int events)
- [DDecayParameters](#) (std::string filename)
- void [saveCSV](#) (std::string filename) const
- std::vector< double > [GetK](#) () const
- std::vector< double > [GetKbar](#) () const
- std::vector< double > [Getc](#) () const
- std::vector< double > [Gets](#) () const

### 3.5.1 Detailed Description

[DDecayParameters](#) is the class that calculates and stores the parameters describing the  $D^0$  and  $\bar{D}^0$  decay. These parameters only need to be calculated once because they only depend on the amplitude model.

Definition at line 14 of file [DDecayParameters.h](#).

### 3.5.2 Constructor & Destructor Documentation

#### 3.5.2.1 [DDecayParameters\(\)](#) [1/2]

```
DDecayParameters::DDecayParameters (
    const PhaseSpaceParameterisation & psp,
    const double & mass_parent,
    const double * mass_decay,
    int events )
```

Constructor that takes in a [PhaseSpaceParameterisation](#) object and calculates the D decay parameters in each bin

Parameters

<i>psp</i>	<a href="#">PhaseSpaceParameterisation</a> object
<i>events</i>	Number of events in each bin for Monte Carlo integration

#### 3.5.2.2 [DDecayParameters\(\)](#) [2/2]

```
DDecayParameters::DDecayParameters (
    std::string filename )
```

Constructor that takes in the D meson hadronic parameters from a comma separated CSV file, in the order i K\_i Kbar\_j c\_i s\_i. First line is assumed to be column names

#### Parameters

<i>filename</i>	Filename of file with D meson hadronic parameters
-----------------	---

Definition at line 72 of file DDecayParameters.cpp.

### 3.5.3 Member Function Documentation

#### 3.5.3.1 Getc()

```
std::vector< double > DDecayParameters::Getc ( ) const
```

Function for getting cosine of the strong phase

##### Returns

c Vector of cosine of the strong phases

Definition at line 113 of file DDecayParameters.cpp.

#### 3.5.3.2 GetK()

```
std::vector< double > DDecayParameters::GetK ( ) const
```

Function for getting fractional yield K<sub>i</sub>

##### Returns

K Vector of fractional yields of D0 events

Definition at line 106 of file DDecayParameters.cpp.

#### 3.5.3.3 GetKbar()

```
std::vector< double > DDecayParameters::GetKbar ( ) const
```

Function for getting fractional yield K<sub>i</sub>

##### Returns

K Vector of fractional yields of DBAR0 events

Definition at line 109 of file DDecayParameters.cpp.

### 3.5.3.4 Gets()

```
std::vector< double > DDecayParameters::Gets ( ) const
```

Function for getting sine of the strong phase

#### Returns

s Vector of sine of the strong phases

Definition at line 117 of file DDecayParameters.cpp.

### 3.5.3.5 saveCSV()

```
void DDecayParameters::saveCSV (
    std::string filename ) const
```

Function for saving K\_i, Kbar\_i, c\_i and s\_i to a CSV file

#### Parameters

<i>filename</i>	Filename of file to save D meson hadronic parameters
-----------------	--

Definition at line 97 of file DDecayParameters.cpp.

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

- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/DDecayParameters.h](#)
- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/DDecayParameters.cpp](#)

## 3.6 Event Class Reference

```
#include <Event.h>
```

### Public Member Functions

- [Event](#) ()
- [Event](#) (std::vector< double > p)
- std::vector< double > [GetEventVector](#) ()
- [Event](#) (const std::vector< TLorentzVector > &p)
- double [GetInvMass2](#) (int particle1, int particle2) const
- double [GetInvMass3](#) (int particle1, int particle2, int particle3) const
- std::vector< double > [GetEvent](#) () const

### 3.6.1 Detailed Description

[Event](#) is a class for storing the four-momenta of daughter particles in a  $D \rightarrow KK\pi\pi$  decay.

Definition at line 12 of file Event.h.

### 3.6.2 Constructor & Destructor Documentation

#### 3.6.2.1 Event() [1/3]

```
Event::Event ( )
```

Default constructor for D to K+ K- pi+ pi- event with zero momentum

Definition at line 7 of file Event.cpp.

#### 3.6.2.2 Event() [2/3]

```
Event::Event (
    std::vector< double > p )
```

Constructor that takes a vector of four-momenta

Parameters

$p$	Four-momenta in the form (E, px, py, pz), in the order K+ K- pi+ pi-
-----	--

Definition at line 11 of file Event.cpp.

#### 3.6.2.3 Event() [3/3]

```
Event::Event (
    const std::vector< TLorentzVector > & p )
```

Constructor that takes a vector of four-momenta

Parameters

$p$	Vector of TLorentzVector objects, in the order K+ K- pi+ pi-
-----	--

Definition at line 18 of file Event.cpp.

### 3.6.3 Member Function Documentation

#### 3.6.3.1 GetEvent()

```
std::vector< double > Event::GetEvent ( ) const
```

Function for getting vector of four-momenta of event

##### Returns

Vector of four-momenta

Definition at line 42 of file Event.cpp.

#### 3.6.3.2 GetEventVector()

```
std::vector< double > Event::GetEventVector ( )
```

Returns the four-momenta of daughter particles as a vector

##### Returns

Four-momenta of daughter particles in the form (E, px, py, pz), in the order K+ K- pi+ pi-

Definition at line 14 of file Event.cpp.

#### 3.6.3.3 GetInvMass2()

```
double Event::GetInvMass2 (
    int particle1,
    int particle2 ) const
```

Function for getting invariant mass of two particles

##### Parameters

<i>particle1</i>	Particle 0(K+), 1(K-), 2(pi+), 3(pi-)
<i>particle2</i>	Particle 0(K+), 1(K-), 2(pi+), 3(pi-)

##### Returns

Returns invariant mass of given particles



Definition at line 28 of file Event.cpp.

#### 3.6.3.4 GetInvMass3()

```
double Event::GetInvMass3 (
    int particle1,
    int particle2,
    int particle3 ) const
```

Function for getting invariant mass of three particles

##### Parameters

<i>particle1</i>	Particle 0(K+), 1(K-), 2(pi+), 3(pi-)
<i>particle2</i>	Particle 0(K+), 1(K-), 2(pi+), 3(pi-)
<i>particle3</i>	Particle 0(K+), 1(K-), 2(pi+), 3(pi-)

##### Returns

Returns mass of given particles

Definition at line 35 of file Event.cpp.

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

- /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/Event.h
- /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/Event.cpp

## 3.7 EventList Class Reference

```
#include <EventList.h>
```

### Public Member Functions

- [EventList](#) ()
- void [AddEvent](#) ([Event](#) event)
- int [NumberEvents](#) () const
- std::vector< [Event](#) > [GetEvents](#) ()

#### 3.7.1 Detailed Description

[EventList](#) is a class that contains all events in a sample

Definition at line 13 of file EventList.h.

## 3.7.2 Constructor & Destructor Documentation

### 3.7.2.1 EventList()

```
EventList::EventList ( )
```

Default constructor that creates an empty [EventList](#)

Definition at line 7 of file EventList.cpp.

## 3.7.3 Member Function Documentation

### 3.7.3.1 AddEvent()

```
void EventList::AddEvent (
    Event event )
```

Function that adds an [Event](#) to the [EventList](#)

Parameters

<i>event</i>	New <a href="#">Event</a> object to be added to the <a href="#">EventList</a>
--------------	---

Definition at line 10 of file EventList.cpp.

### 3.7.3.2 GetEvents()

```
std::vector< Event > EventList::GetEvents ( )
```

Function that returns the vector of [Event](#) objects

Returns

Vector of [Event](#) objects

Definition at line 18 of file EventList.cpp.

### 3.7.3.3 NumberEvents()

```
int EventList::NumberEvents ( ) const
```

Function that returns total number of events in this [EventList](#)

Definition at line 14 of file EventList.cpp.

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

- /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/[EventList.h](#)
- /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/[EventList.cpp](#)

## 3.8 Fitter Class Reference

```
#include <Fitter.h>
```

### Public Member Functions

- [Fitter](#) ([BinList](#) binlist, [DDecayParameters](#) ddparameters)
- void [DoFit](#) ([CPPParameters](#) &cparameters)

### 3.8.1 Detailed Description

[Fitter](#) is a class for maximising the likelihood and obtaining the CP violation parameters for a B meson decay

Definition at line 13 of file Fitter.h.

### 3.8.2 Constructor & Destructor Documentation

#### 3.8.2.1 Fitter()

```
Fitter::Fitter (
    BinList binlist,
    DDecayParameters ddparameters )
```

Constructor that takes in a [BinList](#) object of input data and D meson decay parameters

#### Parameters

<i>binlist</i>	Input data events
<i>ddecayparameters</i>	Parameters describing the D meson decay

Definition at line 11 of file Fitter.cpp.

### 3.8.3 Member Function Documentation

#### 3.8.3.1 DoFit()

```
void Fitter::DoFit (
    CParameters & cpparameters )
```

Function for doing fit and returning the CP violation parameters (by reference)

##### Parameters

<i>cpparameters</i>	Initial guess of CP violation parameters, function replaces these with the fitted parameters
---------------------	--

Definition at line 14 of file Fitter.cpp.

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

- /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/Fitter.h
- /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/Fitter.cpp

## 3.9 Generator Class Reference

```
#include <Generator.h>
```

### Public Member Functions

- [Generator](#) (const double &mass\_parent, const Double\_t \*mass\_decay, Int\_t particles)
- std::vector< TLorentzVector > [Generate](#) ()

#### 3.9.1 Detailed Description

[Generator](#) is a class that generates uniformly distributed events in phase space, assuming the parent particle is at rest

Definition at line 13 of file Generator.h.

#### 3.9.2 Constructor & Destructor Documentation

##### 3.9.2.1 Generator()

```
Generator::Generator (
    const double & mass_parent,
    const Double_t * mass_decay,
    Int_t particles )
```

Constructor that takes in the particle masses and sets up phase space

## Parameters

<i>mass_parent</i>	Mass of parent particle
<i>mass_decay</i>	mass of decay particles
<i>particles</i>	Number of particles in the final state

Definition at line 9 of file Generator.cpp.

### 3.9.3 Member Function Documentation

#### 3.9.3.1 Generate()

```
std::vector< TLorentzVector > Generator::Generate ( )
```

Function that generates a random unweighted event

Definition at line 15 of file Generator.cpp.

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

- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Generator.h](#)
- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Generator.cpp](#)

## 3.10 Likelihood Class Reference

```
#include <Likelihood.h>
```

### Public Member Functions

- [Likelihood](#) ([BinList](#) bins, [DDecayParameters](#) ddparameters)
- double [operator\(\)](#) (const double \*cpparameters)

#### 3.10.1 Detailed Description

[Likelihood](#) is a class for calculating the likelihood, given an [EventList](#) of input data and a set of D meson decay parameters and CP violation parameters in B meson decays () operator is overloaded to make the likelihood function easily accessible

Definition at line 13 of file Likelihood.h.

### 3.10.2 Constructor & Destructor Documentation

#### 3.10.2.1 Likelihood()

```
Likelihood::Likelihood (
    BinList bins,
    DDecayParameters ddparameters )
```

Constructor that takes in an [BinList](#) object with input data and a [DDecayParameters](#) object

## Parameters

<i>events</i>	<a href="#">BinList</a> object with the input data
<i>ddecayparameters</i>	A <a href="#">DDecayParameters</a> object with the parameters for the D meson decay

Definition at line 10 of file Likelihood.cpp.

### 3.10.3 Member Function Documentation

#### 3.10.3.1 operator()

```
double Likelihood::operator() (
    const double * cparameters )
```

Operator overload of () to easily access the likelihood function

## Parameters

<i>cparameters</i>	A <a href="#">CParameters</a> object with the CP violation parameters for the B meson decay
--------------------	---

## Returns

$-2 \cdot \ln(L)$ , where L is the likelihood function

Definition at line 13 of file Likelihood.cpp.

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

- /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/[Likelihood.h](#)
- /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/[Likelihood.cpp](#)

## 3.11 PhaseSpaceParameterisation Class Reference

```
#include <PhaseSpaceParameterisation.h>
```

### Public Member Functions

- [PhaseSpaceParameterisation](#) ()
- int [WhichBin](#) (const [Event](#) &event)
- int [NumberOfBins](#) ()

### 3.11.1 Detailed Description

[PhaseSpaceParameterisation](#) is a class that contains the information about how phase space is divided into bins. [PhaseSpaceParameterisation](#) contains a very coarse and arbitrary binning of phase space. A more sophisticated binning can be added by adding a new class that inherits from [PhaseSpaceParameterisation](#).

Definition at line 13 of file `PhaseSpaceParameterisation.h`.

### 3.11.2 Constructor & Destructor Documentation

#### 3.11.2.1 PhaseSpaceParameterisation()

```
PhaseSpaceParameterisation::PhaseSpaceParameterisation ( )
```

Default constructor

Definition at line 7 of file `PhaseSpaceParameterisation.cpp`.

### 3.11.3 Member Function Documentation

#### 3.11.3.1 NumberOfBins()

```
int PhaseSpaceParameterisation::NumberOfBins ( )
```

Function that returns the number of bins in the binning scheme

Returns

Number of bins

Definition at line 23 of file `PhaseSpaceParameterisation.cpp`.

#### 3.11.3.2 WhichBin()

```
int PhaseSpaceParameterisation::WhichBin (
    const Event & event )
```

Function that determines which bin an event belongs to

**Parameters**

<i>event</i>	The event we want to determine the bin of
--------------	---

**Returns**

[Bin](#) number

Definition at line 10 of file PhaseSpaceParameterisation.cpp.

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

- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/PhaseSpaceParameterisation.h](#)
- [/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/PhaseSpaceParameterisation.cpp](#)



## Chapter 4

# File Documentation

### 4.1 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/Amplitude.h File Reference

```
#include <vector>
#include <string>
#include <complex>
```

#### Classes

- class [Amplitude](#)

### 4.2 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/Bin.h File Reference

```
#include "EventList.h"
#include "Event.h"
```

#### Classes

- class [Bin](#)

### 4.3 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/BinList.h File Reference

```
#include <vector>
#include "PhaseSpaceParameterisation.h"
#include "Event.h"
#include "Bin.h"
#include "DDecayParameters.h"
#include "CParameters.h"
```

## Classes

- class [BinList](#)

## 4.4 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/CParameters.h File Reference

## Classes

- class [CParameters](#)

## 4.5 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/DDecayParameters.h File Reference

```
#include <vector>
#include <string>
#include "PhaseSpaceParameterisation.h"
```

## Classes

- class [DDecayParameters](#)

## 4.6 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/Event.h File Reference

```
#include <vector>
#include "TLorentzVector.h"
```

## Classes

- class [Event](#)

## 4.7 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/EventList.h File Reference

```
#include "Event.h"
#include "TTree.h"
#include <vector>
```

## Classes

- class [EventList](#)

## 4.8 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/FitGamma.h File Reference

## 4.9 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/Fitter.h File Reference

```
#include "BinList.h"
#include "DDecayParameters.h"
#include "CParameters.h"
```

## Classes

- class [Fitter](#)

## 4.10 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/Generator.h File Reference

```
#include <vector>
#include "TLorentzVector.h"
#include "TGenPhaseSpace.h"
```

## Classes

- class [Generator](#)

## 4.11 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/HadronicParameters.h File Reference

## 4.12 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/Likelihood.h File Reference

```
#include "BinList.h"
#include "DDecayParameters.h"
```

## Classes

- class [Likelihood](#)

### 4.13 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/include/PhaseSpaceParameterisation.h File Reference

```
#include "Event.h"
```

## Classes

- class [PhaseSpaceParameterisation](#)

### 4.14 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/Amplitude.cpp File Reference

```
#include <vector>
#include <string>
#include <complex>
#include <algorithm>
#include <dlfcn.h>
#include "Amplitude.h"
```

### 4.15 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/Bin.cpp File Reference

```
#include "Bin.h"
#include "EventList.h"
#include "Event.h"
```

### 4.16 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/BinList.cpp File Reference

```
#include <vector>
#include <functional>
#include "Bin.h"
#include "BinList.h"
#include "PhaseSpaceParameterisation.h"
#include "Event.h"
#include "TTree.h"
#include "DDecayParameters.h"
#include "CParameters.h"
#include "TMath.h"
```

## 4.17 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/CMakeLists.txt File Reference

## 4.18 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/CPPParameters.cpp File Reference

```
#include "CPPParameters.h"
```

## 4.19 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/DDecayParameters.cpp File Reference

```
#include <algorithm>
#include <vector>
#include <complex>
#include <iostream>
#include <string>
#include <fstream>
#include <sstream>
#include "DDecayParameters.h"
#include "BinList.h"
#include "Generator.h"
#include "TLorentzVector.h"
#include "Event.h"
#include "EventList.h"
#include "Bin.h"
#include "TMath.h"
#include "Amplitude.h"
```

## 4.20 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/Event.cpp File Reference

```
#include "Event.h"
#include "TLorentzVector.h"
#include "TMath.h"
```

## 4.21 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/EventList.cpp File Reference

```
#include <vector>
#include "EventList.h"
#include "Event.h"
```

## 4.22 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/Fitter.cpp File Reference

```
#include "Fitter.h"
#include "BinList.h"
#include "DDecayParameters.h"
#include "CParameters.h"
#include "Likelihood.h"
#include "Minuit2/Minuit2Minimizer.h"
#include "Math/Functor.h"
```

## 4.23 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/Generator.cpp File Reference

```
#include <vector>
#include "Generator.h"
#include "TGenPhaseSpace.h"
#include "TLorentzVector.h"
#include "TRandom3.h"
```

## 4.24 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/Likelihood.cpp File Reference

```
#include "Likelihood.h"
#include "DDecayParameters.h"
#include "CParameters.h"
#include "BinList.h"
#include "TMath.h"
#include "Math/PdfFuncMathCore.h"
```

## 4.25 /data/lhcb/users/tat/KKpipi\_Binned\_Fit/src/PhaseSpace↵ Parameterisation.cpp File Reference

```
#include <vector>
#include "PhaseSpaceParameterisation.h"
#include "Event.h"
```

# Index

[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Amplitude.h](#), [BinList](#), [8](#), [9](#)  
[27](#) [EventList](#), [20](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Bin.h](#), [27](#) [Amplitude](#), [5](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/BinList.h](#), [Amplitude](#), [5](#)  
[27](#) [operator\(\)](#), [5](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/CPPParameters.h](#),  
[28](#) [Bin](#), [6](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/DDecayParameters.h](#), [AddEvent](#), [6](#)  
[28](#) [Bin](#), [6](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Event.h](#), [GetEvents](#), [7](#)  
[28](#) [GetNumberEvents](#), [7](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/EventList.h](#), [BinList](#), [8](#)  
[28](#) [AddEvent](#), [8](#), [9](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/FitGamma.h](#), [BinList](#), [8](#)  
[29](#) [GetBin](#), [9](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Fitter.h](#), [GetEvents](#), [10](#)  
[29](#) [LoadTTree](#), [10](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Generator.h](#), [NumberBins](#), [10](#)  
[29](#) [Predict](#), [10](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/HadronicParameters.h](#),  
[29](#) [CPPParameters](#), [11](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/Likelihood.h](#), [CPPParameters](#), [11](#)  
[29](#) [GetCPPParameters](#), [12](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/include/PhaseSpaceParameterisation.h](#),  
[30](#) [GetError](#), [12](#)  
[30](#) [SetError](#), [13](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Amplitude.cpp](#), [DDecayParameters](#), [13](#)  
[30](#) [DDecayParameters](#), [14](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Bin.cpp](#), [30](#) [GetC](#), [15](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/BinList.cpp](#), [GetK](#), [15](#)  
[30](#) [GetKbar](#), [15](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/CMakeLists.txt](#), [Gets](#), [15](#)  
[31](#) [saveCSV](#), [16](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/CPPParameters.cpp](#), [DoFit](#)  
[31](#) [Fitter](#), [22](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/DDecayParameters.cpp](#),  
[31](#) [Event](#), [16](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Event.cpp](#), [Event](#), [17](#)  
[31](#) [GetEvent](#), [18](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/EventList.cpp](#), [GetEventVector](#), [18](#)  
[31](#) [GetInvMass2](#), [18](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Fitter.cpp](#), [32](#) [GetInvMass3](#), [19](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Generator.cpp](#), [EventList](#), [19](#)  
[32](#) [AddEvent](#), [20](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/Likelihood.cpp](#), [EventList](#), [20](#)  
[32](#) [GetEvents](#), [20](#)  
[/data/lhcb/users/tat/KKpipi\\_Binned\\_Fit/src/PhaseSpaceParameterisation.cpp](#), [GetInvMass2](#), [18](#)  
[32](#) [GetInvMass3](#), [19](#)  
[AddEvent](#) [Fitter](#), [21](#)  
[Bin](#), [6](#) [DoFit](#), [22](#)  
[Fitter](#), [21](#)

- Generate
  - Generator, [23](#)
- Generator, [22](#)
  - Generate, [23](#)
  - Generator, [22](#)
- GetBin
  - BinList, [9](#)
- Getc
  - DDecayParameters, [15](#)
- GetCPPParameters
  - CPPParameters, [12](#)
- GetError
  - CPPParameters, [12](#)
- GetEvent
  - Event, [18](#)
- GetEvents
  - Bin, [7](#)
  - BinList, [10](#)
  - EventList, [20](#)
- GetEventVector
  - Event, [18](#)
- GetInvMass2
  - Event, [18](#)
- GetInvMass3
  - Event, [19](#)
- GetK
  - DDecayParameters, [15](#)
- GetKbar
  - DDecayParameters, [15](#)
- GetNumberEvents
  - Bin, [7](#)
- Gets
  - DDecayParameters, [15](#)
- Likelihood, [23](#)
  - Likelihood, [23](#)
  - operator(), [24](#)
- LoadTTree
  - BinList, [10](#)
- NumberBins
  - BinList, [10](#)
- NumberEvents
  - EventList, [20](#)
- NumberOfBins
  - PhaseSpaceParameterisation, [25](#)
- operator()
  - Amplitude, [5](#)
  - Likelihood, [24](#)
- PhaseSpaceParameterisation, [24](#)
  - NumberOfBins, [25](#)
  - PhaseSpaceParameterisation, [25](#)
  - WhichBin, [25](#)
- Predict
  - BinList, [10](#)
- saveCSV
  - DDecayParameters, [16](#)
- SetError
  - CPPParameters, [13](#)
- WhichBin
  - PhaseSpaceParameterisation, [25](#)