Example Analysis Using `AmpTools` for γKK

Kei Moriya*
Department of Physics, Indiana University, Bloomington, IN 47403

March 10, 2020

Abstract

This is documentation for an example of using the `AmpTools` package for the analysis of $J/\psi(\psi') \to \gamma KK$. All programs are documented, and usage is shown.

1 Introduction

This document illustrates how to quickly start doing analyses with the use of the `AmpTools` package. The example we will use is $J/\psi(\psi') \to \gamma KK$, and we are interested in a mass-independent analysis. In reality, the above reaction will be restricted to states of $0^{++}, 2^{++}, 4^{++}, \cdots$ for $KK = K_S^0 K_S^0$, but nontheless, for illustration purposes we will show an example of how the decay will look if we have the KK system going to a spin-1 state.

This tutorial should have been downloaded together with the `AmpTools` package. Since this tutorial is based on the Dalitz plot analysis tutorial also included, it is recommended that the Dalitz plot tutorial be worked out first, to familiarize yourself with the `AmpTools` package.

The overall directory structure is set up as follows. The directories gammaKKAmp, gammaKKDataIO, and gammaKKPlot include all of the necessary files to create the libraries used by this example. The top-level Makefile will create all of these libraries. The directory gammaKKExe contains all of the programs that need to be run in this example tutorial, and these programs are *not* included in the top-level Makefile. Go to this directory and type make to create the executables. The directory run will contain configuration files needed for generating physics events and fitting. Also included is a file run.sh that can be used to run all example executables at once.

2 Preliminaries

To run the tutorial, you will need the packages `ROOT`, `CLHEP`, `AmpTools` set up correctly. Make sure the environment variables `ROOTSYS`, `CLHEP_INCLUDE_DIR`, `AMPTOOLS` are set correctly.

*kmoriya@indiana.edu

The final variable to set is `GAMMAKK`, which can be set by sourcing the file setup_gammaKK.csh in the top directory.

Once these are set up, type make in the top directory to create all of the necessary libraries. Next, move to the gammaKKExe directory and type make to create all of the executables. Finally, move to the run directory and run the script run.sh to run all of the executables in order.

For the file run.sh, if you uncomment the definition of the variable `VERBOSE` at the top of the file, the script will stop at the end of each stage, so that you can understand what the output was, rather than have everything fly by on your screen at once.

3 Directory Structure

As explained in the Introduction, the tutorial is structured so that the source files for all libraries are contained in gammaKKAmp, gammaKKDataIO, and gammaKKPlot, while the executables are in gammaKKExe.

3.1 Libraries

Here we will view the files that create the libraries.

- gammaKKDataIO

  This directory contains the files gammaKKDataReader and gammaDataWriter, which allows the user to read in and write out data to and from files. The DataReader and DataWriter formats are specified so that the user has complete freedom in choosing how the data is formatted.

  For gammaKKDataReader, the data is read in in the format of 4-vectors for each event from a TTree that contains each component of all 4-vectors.

  For gammaKKDataWriter, the function
  `writeEvent( const Kinematics & kin)`
  will read in each event. The Kinematics object is used throughout the `AmpTools` package, and contains a `std::vector<HepLorentzVector>` for that event. The components of the 4-vectors are written out into a TTree, which can in turn be read in by a gammaKKDataReader.

- gammaKKPlot

  This directory contains only gammaKKPlotGenerator, which is used to plot the results of a fit done in this tutorial. The gammaKKPlotGenerator class inherits from PlotGenerator in `AmpTools`, and will implement the virtual function `fillProjections`. The executables that will use this class are plotResults and fitAmplitudesMassBins in gammaKKExe.

- gammaKKAmp

  This directory contains all of the necessary tools needed to generate physics distributions. Some, like clebschGordan and wignerD are rather obvious. NBodyPhaseSpaceFactory will generate events based on phase space, and return a `std::vector <HepLorentzVector>` for that event.

  The other files in this directory are for generating specific amplitudes that involve physics. For this tutorial we will use two amplitudes, gammaKKHelicityAmp and MultipoleAmps. gammaKKHelicityAmp will take in as arguments

  - J: spin of the KK resonance
  - μ: helicity of the KK resonance ($|\mu| \leq J$)
  - M: the z-projection of the spin of the initial state J/ψ or ψ'
  - λ: the final helicity of the photon ($\lambda = \pm 1$)

  The helicity amplitudes are very useful in that given a reaction, it is a simple exercise to write down what that amplitude is. On the other hand, there is no a priori relation between the strengths of the different helicity amplitudes (save those related by parity), so it is difficult to have a physical intuition of the relative strengths.

  The class MultipoleAmps will take in as arguments

  - J: spin of the KK resonance
  - J_γ: total angular momentum of the photon
  - M: the z-projection of the spin of the initial state J/ψ or ψ'
  - λ: the final helicity of the photon ($\lambda = \pm 1$)

  Note that due to conservation of angular momentum, $|J - J_\gamma| \leq 1$, where 1 is the spin of the initial J/ψ or ψ'. Since the helicity of the photon is always $\lambda = \pm 1$ due to its having no mass, J_γ can never be 0. These put restrictions on what values J_γ can take for a specific reaction For example, for $J = 0$, $J_\gamma = 1$ only, for $J = 1$, $J_\gamma = 1, 2$, for $J = 2$, $J_\gamma = 1, 2, 3$, and so on.

  The strength of the multipole amplitudes are related to the helicity amplitudes by a unitary transformation that conserves the overall strength of the reaction. Although the multipole basis is rather complicated due to its containing many sums and coefficients, physics models can give arguments that the lower multipoles will dominate in certain reactions, and in this case it is useful to have amplitudes where a hierarchy is expected.

  Further details on the relation of helicity and multipole amplitudes will be given in a later section, as this will give many insights into how the physics works.

3.2 Executables

Here we will view the executables that will be used.

- generatePhaseSpace

  A class that will generate any number of phase space events for a given reaction. This program will make use of gammaKKDataWriter to write out the event 4-vectors into a ROOT file using a TTree. The arguments taken in are

  - output ROOT file name
  - number of events

  For the actual generation of phase space events, the class NBodyPhaseSpace contained in the directory gammaKKAmp is used.

- generatePhysics

  A class that is very similar to generatePhaseSpace in that it generates events according to a flat phase space distribution, but will accept or reject these events according to a set of amplitudes specified by the user.

  The program will take in as arguments

  - configuration file name
  - output ROOT file name
  - number of events to sample from
  - random seed

  The configuration file is parsed, and the appropriate amplitude and intensity is calculated for each event. For this to be processed properly, each amplitude that is used must be registered within the AmplitudeMananger object. Once this is done, the intensity of each event will be calculated, and the maximum intensity is saved. Since we want the phase space distribution to be filled with each event's weight proportional to its intensity, we will throw a random number for each event and keep the event only if the intensity is higher than the maximum intenstiy multiplied by the random number.

  At the end of the program, the program will report how many events have been retained for analysis. The kept events are written into a ROOT file.

- toyAcceptance

  We would like to simulate having a toy Monte Carlo system that has finite acceptance across different regions of phase space. This program will just read in the 4-vectors of each event, and write out the event with a probability that is proportional to the acceptance at that point. The efficiency can be whatever the user chooses. In this tutorial we have an acceptance that oscillates with the polar angles of the particle in the original frame.

The program will take in as arguments

  - input ROOT file name
  - output ROOT file name

- plotData

  This is a simple program that will take in a ROOT file containing a TTree created by gammaKKDataReader, and make some basice plots. Currently, there are plots for the invariant masses of the combination of particles, a Dalitz plot, and a plot for the γ polar angle in the rest frame.

- fitAmplitudesATI

  The program that will do the heavy work of fitting a given data file with the amplitudes that are specified in the configuration file that is read in. The program reads in

  - configuration file name

  and parses the information to do the fit.

  The configuration file will need the following information. First, the definition of a fit name must be given, followed by the reaction name. The fit name will be used to create the .fit file which contains the fit result. Next, the configuration file must specify which files are to be used for the generated phase space Monte Carlo distribution, the accepted Monte Carlo events, and the data file. The normintfile name will be used to create a normalization integral file which contains the correlation of the different amplitudes in the fit.

  If the fit successfully converges, the files will be written out, and the results examined.

- plotResults

  This program will use the fit results to create a ROOT file containing the histograms that are specified in gammaKKPlotGenerator. The histograms will be for the data, the accepted Monte Carlo events, and the generated Monte Carlo events, based on the fit result.

- fitAmplitudesATIMassBins and fitAmplitudesATISplitMassBins

  These programs are designed to do a mass-independent fit to the spectrum of $J/\psi \to \gamma K_S K_S$, where the data is binned in the invariant mass of the $K_S K_S$ system, and a fit is done independently for each mass bin.

  Since it is sometimes useful to be able to seed the next mass bin's fit by the previous mass bin's fit result, these two programs show examples of how to save the current fit results, and put them back into the next fit. In both cases, information necessary for binning in $K_S K_S$ mass is given in the same cfg file that will be used in the fit. The keyword "binning" is used with three arguments,

- the number of mass bin
- the minimum value
- the maximum value

and this information is extracted within the programs using the class ConfigurationInfo.

fitAmplitudesATIMassBins is designed to read in all of the ROOT files necessary (data, generated Monte Carlo events, accepted Monte Carlo events), and for each mass bin, it will loop over the entire file to determine how many events fit in that mass bin. This is done by passing in three arguments into the gammaKKDataReader, where the first argument is the file name, and the second and third arguments are the mass ranges of the $K_S K_S$ system.

As the entire ROOT files are read in and the $K_S K_S$ invariant mass is calculated for each event in each mass bin, the program is rather slow. An example script of using this program is provided in the directory gammaKKExe/massBins1.

The alternative to this method is to split the necessary ROOT files into $K_S K_S$ mass bins before starting the fits, and this is what fitAmplitudesATISplitMassBins does. We first need to use the program splitByMass to split each of the ROOT files into however many bins are necessary. Then, for the actual fitting, only the necessary ROOT files for that mass bin are required to be read in.

This method is the quicker of the two, but the downside is that you end up with many more data files in your directory. An example script using this program is contained in the directory gammaKKExe/massBins2.

4 Running the Programs

In the run directory, a script run.sh has been created that can be used to run all of the above executables in the correct order, showing how they were intended to be used. The variable `VERBOSE` can be uncommented, so that the script will stop at the end of each stage. Here we view what the program does, and what the results are.

0. make

   First make all of the executables in gammaKKExe to make sure they are up to date.

1. generate phase space events

   Using generatePhaseSpace, we generate 10M phase space events for the reaction $J/\psi \to \gamma K_S^0 K_S^0$.

2. apply toy acceptance to the Monte Carlo events

   Using toyAcceptance, we apply our acceptance to the generated events to obtain our acceptend Monte Carlo event sample.

3. create plots of Dalitz distribution for the Monte Carlo events

   We use plotData to create plots of the above distributions. The plots are saved in the figures directory.

4. generate physics events

   We generate 1M physics events according to the physics that we specify in our configuration file. We can use the helicity amplitudes or the multipole amplitudes.

5. apply toy acceptance to the data

6. create plots of Dalitz distribution for the data

7. fit the data

   Use the program fitAmplitudesATI, along with a configuration file to do the fit.

8. make ROOT files of the fit results

   Use plotResults to create ROOT files containing the fit results.

9. plot the fit results

   Use a ROOT script plotRootFile.C to create figures out of the ROOT file created above. The script plotRootFile.C just opens the ROOT file taken in as an argument, and plots the data, acceptend Monte Carlo, and generated Monte Carlo events. For the Monte Carlo events, the events are divided into contributions from different amplitudes.

5 Amplitudes

In this section we discuss some of the nuances of the helicity and multipole amplitudes. To simplify the examples, we will start with the assumption that there is only one resonance contribution to the reaction.

5.1 Helicity Amplitudes

First we begin with definition of the helicity amplitudes. For our reaction of $J/\psi \to \gamma KK$, if the helicities of each particle is given as M, λ, μ, respectively, then we can write down the amplitude of that decay as

$$A(M, \lambda, \mu) = N_J D^J_{\mu,0}(\hat{q}) N_1 D^1_{M,\mu-\lambda}(\hat{p}) A_\mu, \tag{1}$$

where J is the spin of the KK resonance, $\hat{q}$ is the direction of one of the kaons in the KK rest frame (no rotation from the original rest frame), and $\hat{p}$ is the direction of the KK system in the initial frame, where the z-axis defines the J/ψ spin projection M. N_j is a normalization constant given by $N_j = \sqrt{(2j+1)/4\pi}$, and A_μ gives the dynamical strength of this amplitude, where all dependencies on angle have been taken out.

For our example, to obtain the intensity of the photon in the initial frame, we would have to calculate

$$I = \sum_{M,\lambda} \left| \sum_{\mu} A(M,\lambda,\mu) \right|^2 \tag{2}$$

$$= \sum_{M,\lambda} N_J^2 N_1^2 \sum_{\mu,\mu'} D^J_{\mu,0}(\hat{q}) D^{J*}_{\mu',0}(\hat{q}) D^1_{Mu,\mu-\lambda}(\hat{p}) D^{1*}_{M,\mu'-\lambda}(\hat{p}) A_\mu A^*_{\mu'}. \tag{3}$$

Note that if we are not interested in the angles of the breakup of the KK system, then we can integrate over the directions $\hat{q}$, and use the orthogonality of the Wigner D functions to obtain

$$I' = \int \mathrm{d}\hat{q} \sum_{M,\lambda} \left| \sum_{\mu} A(M,\lambda,\mu) \right|^2 \tag{4}$$

$$= \sum_{M,\lambda} N_1^2 \sum_{\mu} \left| D^1_{M,\mu-\lambda}(\hat{p}) \right|^2 \tag{5}$$

$$= N_1^2 \sum_{M,\lambda,\mu} \left| D^1_{M,\mu-\lambda}(\hat{p}) \right|^2 |A_\mu|^2 . \tag{6}$$

This is a rather simple quantity to calculate for the lower values of J. Note that for each of the amplitudes that are given by $A(M,\lambda,\mu)$, the different amplitudes do not interfere with each other.

For $J = 0$, we have $\mu = 0$, so that there is only one strength A_0 for the amplitudes.

$$I'(J=0) \propto \sum_{M,\lambda} \left| D^1_{M,-\lambda}(\hat{p}) \right|^2 \tag{7}$$

$$= \left| D^1_{1,-1}(\hat{p}) \right|^2 + \left| D^1_{1,+1}(\hat{p}) \right|^2 + \left| D^1_{-1,-1}(\hat{p}) \right|^2 + \left| D^1_{-1,+1}(\hat{p}) \right|^2 \tag{8}$$

$$= 1 + \cos^2\theta_\gamma. \tag{9}$$

Therefore, for example in $J/\psi \to \gamma\chi_{c0}$, the distribution is completely specified as $1 + \cos^2\theta_\gamma$.

For $J = 1$, we have three different amplitudes for $\mu = \pm 1, 0$. The photon distribution will be

$$I'(J=1) \propto \sum_{M,\lambda,\mu} \left| D^1_{M,\mu-\lambda}(\hat{p}) \right|^2 |A_\mu|^2 . \tag{10}$$

Here we come to the realization that, as the helicity basis does not specify the relative strengths of each amplitude, there is nothing more we can do without some kind of dynamics to specify the decay.

However there is one simplification that we can do, as parity ensures that $A(M, \lambda, \mu)$ is related to $A(M, -\lambda, -\mu)$ by a phase of ± 1. Hence, the final intensity is

$$I'(J=1) \propto \sum_{M,\lambda,\mu=0,+1} \left|D^1_{M,\mu-\lambda}(\hat{p})\right|^2 |A_\mu|^2 \tag{11}$$

$$\propto \left\{\left|D^1_{1,0}(\hat{p})\right|^2 + \left|D^1_{1,0}(\hat{p})\right|^2\right\} |A_1|^2 + \left\{\left|D^1_{1,-1}(\hat{p})\right|^2 + \left|D^1_{1,+1}(\hat{p})\right|^2\right\} |A_0|^2 \tag{12}$$

$$= (1 - \cos^2\theta_\gamma) |A_1|^2 + \frac{1+\cos^2\theta_\gamma}{2} |A_0|^2 \tag{13}$$

As can be calculated from the definition of the multipole amplitudes, for a pure E1 transition in $\psi' \to \gamma\chi_{c1}$, the helicity amplitudes are given as $A_1 = A_0 = a_1/\sqrt{2}$, where a_1 is the E1 amplitude. In this case the above expression simplifies to

$$I'(J=1) \propto 1 - \frac{1}{3}\cos^2\theta_\gamma. \tag{14}$$

The decay for $J = 2$ can also be treated in a similar way, noting that now there are three independent strengths, A_2, A_1, and A_0. The transformation to the multipole basis allows the determination of these strengths, and in the case of E1 dominated $\psi' \to \gamma\chi_{c2}$,

$$A_2 = \sqrt{2}A_1 = \sqrt{6}A_0. \tag{15}$$

Then the photon angular distribution is given as

$$I'(J=1) \propto 1 + \frac{1}{13}\cos^2\theta_\gamma. \tag{16}$$

5.2 Multipole Amplitudes

Now that we have some intuitive feel of how to calculate the helicity amplitudes, we move on to the multipole amplitudes. The multipole amplitudes are specified for each combination of M, λ, and J_γ, where J_γ is the angular momentum of the photon system as

$$a(M,\lambda,J_\gamma) = \sqrt{\frac{2J_\gamma+1}{2\times 1+1}} \left[\sum_\mu D^1_{M,\mu-\lambda}(\hat{q}) D^J_{\mu,0}(\hat{p}) CG(J,\mu-\lambda|J_\gamma,-\lambda;j,\mu)\right] a_{\lambda,J_\gamma} \tag{17}$$

$$\equiv \sqrt{\frac{2J_\gamma+1}{2\times 1+1}} c_{J_\gamma} a_{\lambda,J_\gamma} \tag{18}$$

where $CG(J, M|j_1, m_1; j_2, m_2)$ is a standard Clebsch-Gordan coefficient coupling spins j_1 and j_2 to J. The part in square brackets can be considered an angle-dependent coefficient for the dynamical strength a_{λ,J_γ}, which depends on M and λ, and is summed over all μ

values. As the photon is massless, its helicity values are constrained to ± 1, so that J_γ can never be 0.

Again, we calculate the photon angular distribution. The total intenstity is

$$I = \sum_{M,\lambda} \left| \sum_{J_\gamma} a(M, \lambda, J_\gamma) \right|^2 \tag{19}$$

$$= \sum_{M,\lambda} \sum_{J_\gamma, J'_\gamma} \frac{\sqrt{(2J_\gamma + 1)(2J'_\gamma + 1)}}{3} c_{J_\gamma} c^*_{J'_\gamma}(\hat{p}) a_{\lambda,J_\gamma} a^*_{\lambda,J'_\gamma}. \tag{20}$$

If we integrate over the KK breakup angles $\hat{q}$, the only part that has a dependence on $\hat{q}$ are the c_{J_γ} terms, and we have

$$C_{J_\gamma,J'_\gamma} \equiv \int \mathrm{d}\hat{q} c_{J_\gamma} c^*_{J'_\gamma} \tag{21}$$

$$= \int \mathrm{d}\hat{q} \sum_{\mu,\mu'} D^1_{M,\mu-\lambda}(\hat{p}) D^J_{\mu,0}(\hat{q}) D^{1*}_{M,\mu'-\lambda}(\hat{q}) D^{J*}_{\mu',0}(\hat{q})$$
$$\times CG(1, \mu - \lambda | J_\gamma, -\lambda; J, \mu) CG(1, \mu' - \lambda | J'_\gamma, -\lambda; J, \mu'). \tag{22}$$

Again we use the orthogonality of the Wigner D functions, but now we have

$$C_{J_\gamma,J'_\gamma} = \int \mathrm{d}\hat{q} \sum_{\mu,\mu'} D^1_{M,\mu-\lambda}(\hat{p}) D^J_{\mu,0}(\hat{q}) D^{1*}_{M,\mu'-\lambda}(\hat{q}) D^{J*}_{\mu',0}(\hat{q})$$
$$\times CG(1, \mu - \lambda | J_\gamma, -\lambda; J, \mu) CG(1, \mu' - \lambda | J'_\gamma, -\lambda; J, \mu') \tag{23}$$

$$\propto \sum_{\mu} \left| D^1_{M,\mu-\lambda}(\hat{p}) \right|^2 CG(1, \mu - \lambda | J_\gamma, -\lambda; J, \mu) CG(1, \mu - \lambda | J'_\gamma, -\lambda; J, \mu) \tag{24}$$

so that

$$I' \equiv \int \mathrm{d}\hat{q} I \tag{25}$$

$$= \sum_{J_\gamma, J'_\gamma} \frac{\sqrt{(2J_\gamma + 1)(2J'_\gamma + 1)}}{3} \sum_{M,\lambda}$$
$$\times \left[\sum_{\mu} \left| D^1_{M,\mu-\lambda}(\hat{p}) \right|^2 CG(1, \mu - \lambda | J_\gamma, -\lambda; J, \mu) CG(1, \mu - \lambda | J'_\gamma, -\lambda; J, \mu) \right] a_{\lambda,J_\gamma} a^*_{\lambda,J'_\gamma} \tag{26}$$

The important feature of the above equation in contrast to Eq. 6 is that while the helicity amplitudes were individually squared, and then summed, the multipole amplitudes will have an interference term between them.

This is illustrated in the examples below. Take the case of $J = 1$, which would correspond to $\psi' \to \gamma\chi_{c1}$ (the case of $J = 0$ is exactly the same as for the helicity angles, with the multipole strength a_{λ,J_γ} being equal to A_0).

In this case we can divide I' into a sum over $(J_\gamma, J'_\gamma) = (1,1),(1,2),(2,2)$ where the $(1,1)$ and $(2,2)$ are for the E1 and M2 terms, and $(1,2)$ term is the interference term between them.

For $(J_\gamma, J'_\gamma) = (1,1)$,

$$I_{11} = \sum_{M,\lambda}\left[\sum_{\mu}\left|D^1_{M,\mu-\lambda}(\hat{p})\right|^2 CG(1,\mu-\lambda|1,-\lambda;J,\mu)^2\right] a_{\lambda,1}a^*_{\lambda,1} \tag{27}$$

If we assume parity conservation, it can be shown that the amplitude a_{λ,J_γ} is related to $a_{-\lambda,J_\gamma}$ by

$$a_{\lambda,J_\gamma} = (-)^{J_\gamma - 1} a_{-\lambda,J_\gamma} \tag{28}$$

for our case of $\psi' \to \gamma\chi_{c2}$. This simplifies the sum, and we have

$$I'_{11} \propto \left(1 - \frac{1}{3}\cos^2\theta_\gamma\right)|a_1|^2, \tag{29}$$

where a_1 is the E1 amplitude.

Similarly, a calculation can be carried out for the interference term, which yields

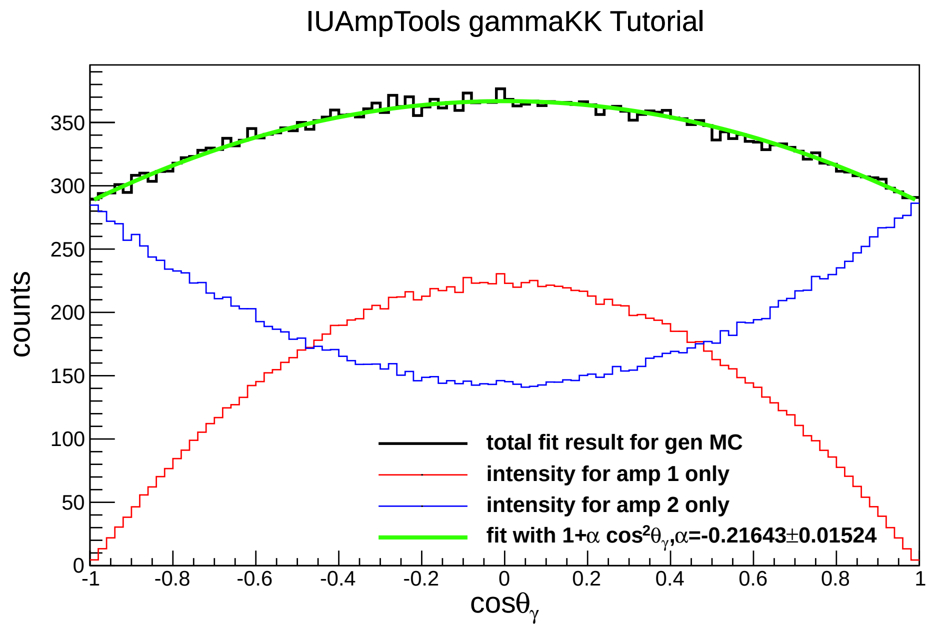
$$I'_{12} \propto \left(3\cos^2\theta_\gamma - 1\right) a_1 a_2, \tag{30}$$

where a_2 is the M2 amplitude. The intensity due to M2 only (I'_{22}) actually has the same angular form as the E1 intensity, so we observe that any deviation from the form of $1 - \cos^2\theta_\gamma/3$ signals an interference of the E1 and M2 amplitudes. Note that however, if we integrate the total intenstiy I' over all KK angles $\hat{p}$, the interference term becomes 0.

5.3 Examples

In the current tutorial, there are some examples of generating and fitting with the helicity and multipole amplitudes. The classes gammaKKHelicityAmp and MultipoleAmps in gammaKKAmp give each of these amplitudes so that simple calculations can be done. These examples are in a separate script called run_spin1_E1dominated.sh, which will run all of the possible combinations to pull out the same results.

Figures 1 and 2 show the results of generating and fitting data samples with all combinations of helicity amplitudes and multipole amplitudes. Note how the helicity amplitudes add together incoherently to create the overall distribution, while the multipole amplitude has a much small intensity for M2, but the interference with E1 contributes significantly to the distribution. In all cases the angular distribution of the photon is fit well to a form of $1 + \alpha\cos^2\theta_\gamma$, with $\alpha = -0.241379$ predicted by theory for E1/M2 = 0.05.



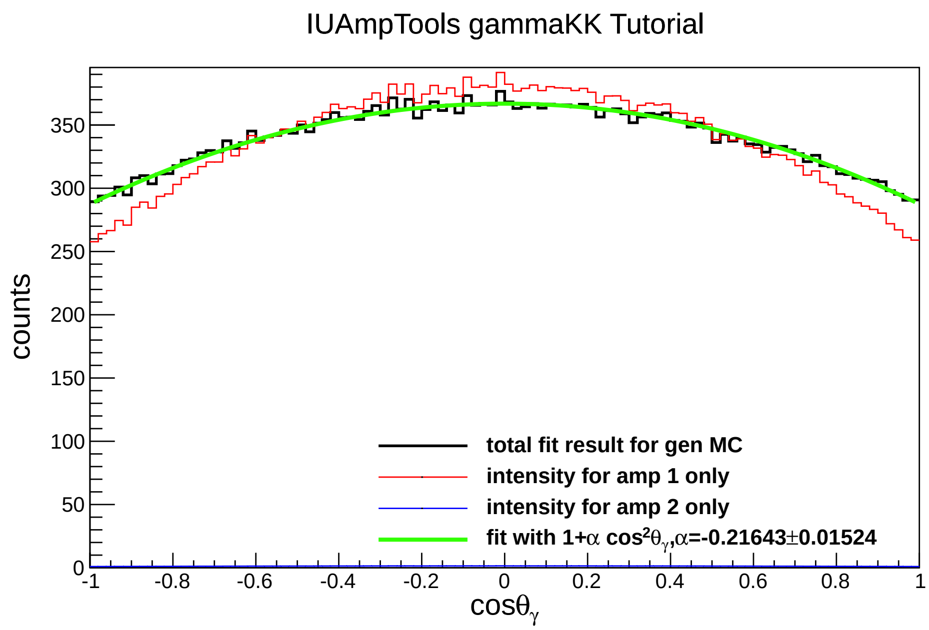


Figure 1: The angular distributions obtained for a mixture of helicity amplitudes, fit with helicity amplitudes (top) and multipole amplitudes (bottom). In both cases the correct parameter α for the distribution $1+\alpha\cos^2\theta_\gamma$ is extracted from a fit, although the ampitudes contribute in significantly different ways. Note the dashed blue curve at the bottom for M2 in the bottom plot. The difference between the intensity of amp 1 (E1) and the total distribution is the interference between E1 and M2. In the example above, the ratio is $A_1/A_0 = 0.95/1.05$ where A_μ is the strength of the helicity amplitude for μ.

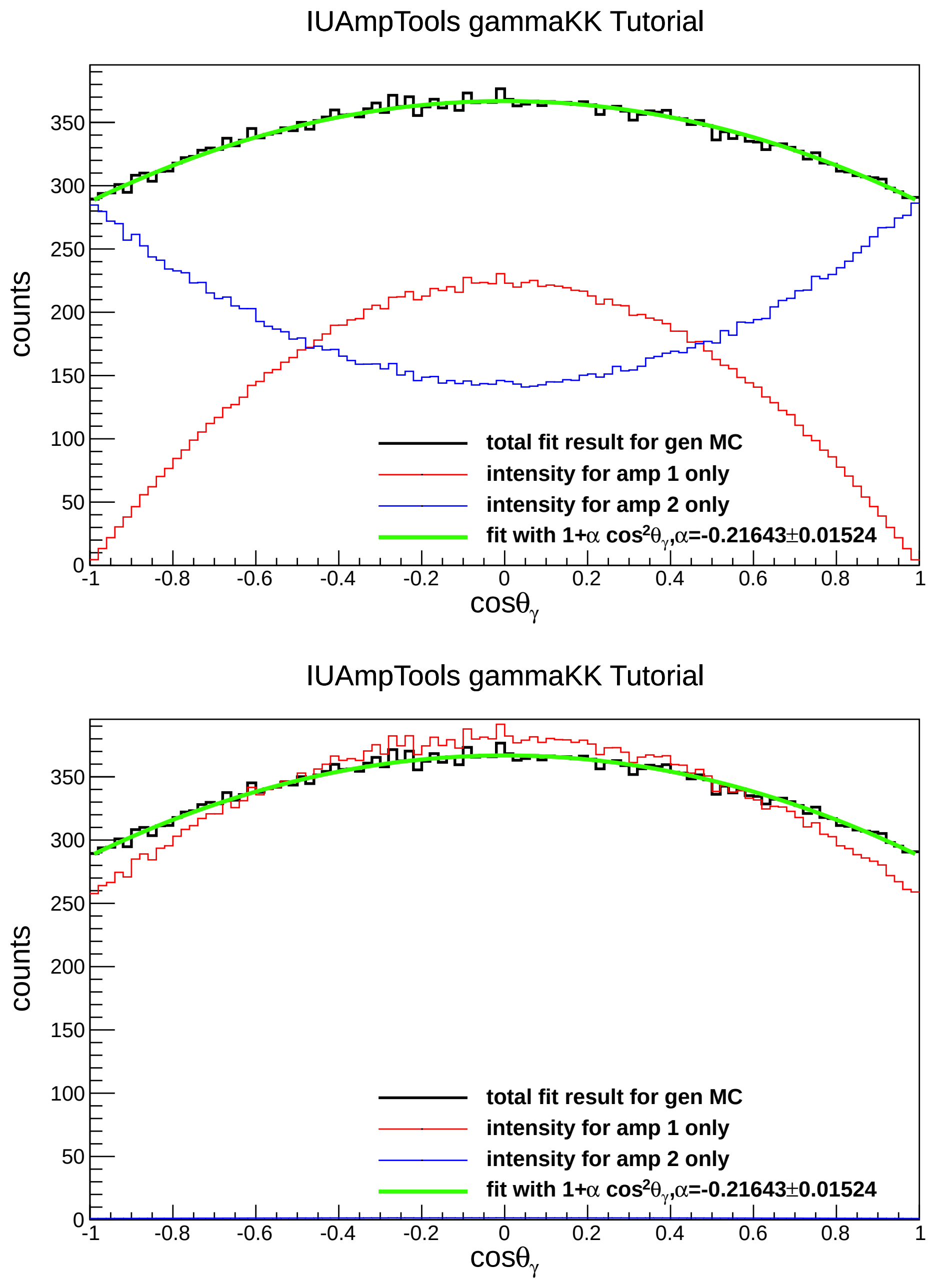


Figure 2: The angular distributions obtained for a mixture of E1 and M2 amplitudes, fit with helicity amplitudes (top) and multipole amplitudes (bottom). In both cases the correct parameter α for the distribution $1 + \alpha \cos^2 \theta_\gamma$ is extracted from a fit, although the ampitudes contribute in significantly different ways. Note the dashed blue curve at the bottom for M2 in the bottom plot. The difference between the intensity of amp 1 (E1) and the total distribution is the interference between E1 and M2. In the example above, the ratio is M2/E1= 0.05.

6 Conclusion

This document summarizes the usage of the tutorial to do a mass-independent amplitude analysis of $J/\psi \to \gamma KK$ using `AmpTools`. The scripts included should allow the user to understand what is going on inside each step of the program.