Cross-section generation

Denisa Apetroaei

January 2019

Abstract

The focus of this project is to produce a simulated particle event, in this case two-to-two processes using Monte Carlo. This is also used to calculate the cross-section integral. There are 4 main goals: transforming the two-to-two partonic cross-sections given differentially in $\hat{t}$ to differential cross-sections in $\cos\hat{\theta}$ and integrating them, using the accept-and-reject method to generate two-body final state events, calculating the PDF for a given particle and calculating the total cross-section.

Two-body final state events were successfully generated for the differential cross-sections for a light quark scattering on a gluon, gluon scattering into a quark/anti-quark pair , Higgs production from gluon fusion and total cross-section was calculated using data read from the input file *pdf.dat*. Quark scattering and gluon scattering presented problems as $\cos\hat{\theta}$ approaches -1 and 1 because of the limit going towards infinity. This was avoided by looking at the maximum of the function ignoring the infinite values. Also the Higgs production differential cross-section is a constant so the probability of generating these events is always 1.

1 Introduction

Monte Carlo generators play an important role in particle physics. They provide a method to connect perturbative quantum field theory with empirically driven phenomenological models, provide a method to efficiently integrate high dimension integrals, and can output particle events similar to what would be seen in colliders[1]. In this project it will be used for the generation of the hard process, producing a simulated particle event for two-to-two processes and calculating the total cross-section.
This is calculated as a function of the three Mandelstam variables: s,t and u.

$$\begin{aligned} s &= (p_{(1)} + p_{(2)})^2 = (p_{(3)} + p_{(4)})^2 \\ t &= (p_{(1)} - p_{(3)})^2 = (p_{(4)} - p_{(2)})^2 \\ u &= (p_{(1)} - p_{(4)})^2 = (p_{(3)} - p_{(2)})^2 \end{aligned} \tag{1}$$

2 Goal 1

In this project there are three main events to be looked at. The first goal is to transform the two-to-two partonic cross-sections that are given differentially in $\hat{t}$ to differential equations in $\cos\hat{\theta}$. It is required to change their form and integrate the partonic cross-sections for: light quark scattering on a gluon, gluon scattering into a quark/anti-quark pair and Higgs production from gluon fusion. This is done in the class $functions()$. It contains the method $req\_for\_eq()$ which takes as inputs $\hat{s}$ and $\cos\hat{\theta}$ and sets the variables needed in two of the three functions. Initialisation was not used, as it would be expected, because some of the methods need to be called later, $f(x, y)$, as inputs to other functions.

2.1 Light quark scattering on a gluon

The partonic cross-section for a light quark scattering on a gluon, $qg \to qg$, is:

$$\frac{d\sigma(\hat{s}, \hat{t}, \hat{u})}{d\hat{t}} = \frac{\pi {\alpha_s}^2 (\hat{s}^2 + \hat{u}^2)(9\hat{s}\hat{u} - 4\hat{t}^2)}{9\hat{s}^3\hat{t}^2\hat{u}} \tag{2}$$

By multiplying with $\frac{d\hat{t}}{d\cos(\hat{\theta})} = \frac{\hat{s}}{2}$ and replacing $\hat{u}$ and $\hat{t}$ with their form from Eq. 2 the final form of the equation is:

$$\frac{d\sigma}{d\cos\hat{\theta}} = \frac{(\cos\hat{\theta}^2 + 2\cos\hat{\theta} + 2)(2\cos\hat{\theta}^2 + 5\cos\hat{\theta} + 2)}{9\hat{s}^4(\cos\hat{\theta}^3 - \cos\hat{\theta}^2 - \cos\hat{\theta} + 1)} \tag{3}$$

This equation tends towards infinity as $\cos\hat{\theta} \to 1$ and towards -infinity when $\cos\hat{\theta} \to -1$. When integrated over $[-1, 1]$ the integral is divergent its primitive being:

$$\sigma(\cos\hat{\theta}) = -\frac{(\cos\hat{\theta} - 1)\ln\left(\left|\cos\hat{\theta} + 1\right|\right) - 4\cos\hat{\theta}(\cos\hat{\theta} + 11) - 117\ln\left(\left|\cos\hat{\theta} - 1\right|\right) + 90}{36\hat{s}^4(\cos\hat{\theta} - 1)} + C \tag{4}$$

Integrated over $[-0.99, 0.99]$ Eq.4 would equal $\frac{233.813441}{\hat{s}^4}$.
This function is defined in the method *quark_scattering*() which takes as inputs $\hat{s}$ and $\cos\hat{\theta}$.

2.2 Gluon scattering into a quark/anti-quark pair

The partonic cross-section for a gluon scattering into a quark/anti-quark pair is:

$$\frac{d\sigma(\hat{s}, \hat{t}, \hat{u})}{d\hat{t}} = \frac{\pi{\alpha_s}^2(\hat{t}^2 + \hat{u}^2)(4\hat{s}^2 - 9\hat{t}\hat{u})}{24\hat{s}^4\hat{t}\hat{u}} \tag{5}$$

Applying the chain-rule and replacing the expressions for $\hat{t}$ and $\hat{u}$ the equation becomes:

$$\frac{d\sigma}{d\cos\hat{\theta}} = \frac{(1 - \cos\hat{\theta})^2 + (1 + cos\hat{\theta})^2(4\hat{s}^2 - 9\frac{\hat{s}^2}{4}(1 - \cos\hat{\theta}^2))}{24\hat{s}^4(1 - \cos\hat{\theta}^2)} \tag{6}$$

This equation tends towards infinity as $\cos\hat{\theta} \to 1$ and $\cos\hat{\theta} \to -1$. When integrated over $[-1, 1]$ the integral is divergent its primitive being:

$$\sigma(\cos\hat{\theta}) = \frac{8\ln\left(\left|\cos\hat{\theta} + 1\right|\right) - \cos\hat{\theta}(\hat{s}^2(3\cos\hat{\theta}^2 + 9\cos\hat{\theta} + 25) + 4) - 32\hat{s}\ln\left(\left|\cos\hat{\theta} - 1\right|\right)}{96\hat{s}^4} + C \tag{7}$$

Integrated over $[-0.99, 0.99]$ Eq.6 would equal $\frac{1.188166}{\hat{s}^2} + \frac{0.358608}{\hat{s}^4}$.
This function is defined in the method *gluon_scattering*() which takes as inputs $\hat{s}$ and $\cos\hat{\theta}$.

2.3 Higgs production from gluon fusion

The partonic cross-section for Higgs production from gluon fusion is:

$$\frac{d\sigma(\hat{s},\hat{t},\hat{u})}{d\hat{t}} = \frac{\pi B_{gg} B_{f\bar{f}} \Gamma^2}{8(\Gamma^2 m_H^2 + (\hat{s} - m_H^2)^2)} \tag{8}$$

After applying the chain-rule this equation becomes:

$$\frac{d\sigma(\hat{s})}{d\cos\hat{\theta}} = \frac{\pi B_{gg} B_{f\bar{f}} \Gamma^2}{8(\Gamma^2 m_H^2 + (\hat{s} - m_H^2)^2)} \frac{\hat{s}}{2} \tag{9}$$

For the partonic cross-section for Higgs production from gluon fusion the result varies with different constants for the branching fraction of the Higgs boson into gluons, the branching fraction of the Higgs boson into a fermion pair, and the width of the Higgs. They were chosen for the Higgs mass[2] at rest $= 125.09 GeV/c^2$.

$$\begin{aligned} B_{gg} &= 0.0857 GeV/c^2 \\ B_{ff} &= 0.065 GeV/c^2 \\ \gamma &= 0.00407 GeV/c^2 \\ m_H &= 125.09 GeV/c^2 \end{aligned} \tag{10}$$

Integrated over $[-1, 1]$, the value of the function becomes:

$$\frac{d\sigma(\hat{s})}{d\cos\hat{\theta}} = \frac{\pi B_{gg} B_{f\bar{f}} \Gamma^2}{8(\Gamma^2 m_H^2 + (\hat{s} - m_H^2)^2)} \frac{\hat{s}}{2}(1-(-1)) = \frac{\pi B_{gg} B_{f\bar{f}} \Gamma^2}{8(\Gamma^2 m_H^2 + (\hat{s} - m_H^2)^2)} \hat{s} \tag{11}$$

This function is defined in the method *higgs_production*() which takes as inputs $\hat{s}$ and $\cos\hat{\theta}$.

2.4 Implementation of Goal 1

In the python code a class $functions()$ is created. Method $req\_for\_eq(self, s, cos\_theta)$ updates variables when a certain function is called. It is not defined as initialisation method because it is desired to be updated everytime a function is called as a method, not as a class.

For each function there is a method that takes as input the value of s and $\cos\theta$. The values for minimum and maximum are searched for in the method *max_min_value* which takes as an input the name of the function and the value of s. It returns the value of the maximum. The method integrate takes as values the function and a value of s and it calculates the integral using the quad function of the Scipy library.

3 Goal 2

Class *event_gen*() generates two-body final state events when $\hat{s}$ is given together with the name of the function and the number of efents that want to be generated. It uses the hit and miss method, implemented in method *hit_and_miss*() and it gives as outputs, when method *print_output*() is called, the final Momentum fourvectors in a text file. It contains in its title the name of the function used and the value of s. In order for the class to generate and print method *generate*() needs to be called. This has been split this was because later on in PDF() class events are generated by calling this functions but printed in a different manner.
For the output Momentum Fourvectors[3]:

$$\begin{aligned} p_3 = (E_3, q\sin\theta\cos\phi, q\sin\theta\sin\phi, q\cos\theta) \\ p_4 = (E_4, -q\sin\theta\cos\phi, -q\sin\theta\sin\phi, -q\cos\theta) \\ q = \sqrt{E_1^2 - m_3^2} \end{aligned} \tag{12}$$

From Eq.1:

$$s + t + u = m_1^2 + m_2^2 + m_3^2 + m_4^2 = 0 \tag{13}$$

From the conservation of energy and momentum:

$$(2E_3)^2 = s \tag{14}$$

so:

$$E_3 = \frac{\sqrt{s}}{2}$$

Together with Eq.12 it can be seen that $q = \frac{\sqrt{s}}{2}$. So the output vectors become:

$$\begin{aligned} p_3 = (\frac{\sqrt{s}}{2}, \frac{\sqrt{s}}{2}\sin\theta\cos\phi, \frac{\sqrt{s}}{2}\sin\theta\sin\phi, \frac{\sqrt{s}}{2}\cos\theta) \\ p_4 = (\frac{\sqrt{s}}{2}, -\frac{\sqrt{s}}{2}\sin\theta\cos\phi, -\frac{\sqrt{s}}{2}\sin\theta\sin\phi, -\frac{\sqrt{s}}{2}\cos\theta) \end{aligned} \tag{15}$$

In this algorithm ϕ is generated as a random number in the interval $[0, 2\pi]$ after it is decided if the event will be generated successfully.

The output file also contains the probability of success of the generation. Given s, the hit and miss method selects a random $\cos\theta$ and calculates the value of the cross-section derivative in that point. This calls the previously defined functions in Goal 1. The maximum of the given function is defined in the initialisation of the generator class such that a random number in between 0 and the maximum is chosen. What is tested here is if a choosen random value under the function curve is less then the previously chosen value of the function.
At this step the probability depends on the step with which the function is defined. For the first two functions, quark scattering and gluon scattering, it was discussed how the function is divergent. As it approaches 1, it grows towards infinity. The order of the step defines the order of the maximum value of the function. If the maximum is very high the range of possible values for the random number between 0 and maximum increases drastically while the $\cos\theta$ has a fixed range of 2. If the maximum value is let to be infinity, it can be impossible to generate an event. This is avoided by excluding the infinite values when looking for the maximum.
The step chosen in the code is 10^{-2}. For a gluon-scattering at $s = 520GeV/c^2$ and $N = 10$ events wanted, the probability of success is 0.078. For the same initial conditions in quark-scattering the probability of success is 0.013.

For a step of 10^{-1} the probability for gluon-scattering grows to 0.19 and for quark-scattering grows to 0.13 for $N = 10$ events generated.
The probability of getting a value for $cos\theta$ is $\frac{1}{2N_{total}}$ and the probability of getting a random $d\sigma$ is $\frac{1}{max(d\sigma)N_{total}}$ where N_{total} is the number of subintervals on $[0, 1]$ of the $random.uniform()$ function in Python.

Because $\sigma(\cos\theta)$ is calculated from the randomly chosen $\cos\theta$ the probability of getting a set value is the same, $\frac{1}{2N_{total}}$. When the maximum value of $d\sigma$ is bigger than 2, it can be seen how it becomes harder and harder for a randomly chosen f to be less than $f(\cos\theta)$ where $\cos\theta$ is chosen randomly.

The maximum value of the function depends on the step because of the open interval on which *numpy.arange* operates. The bigger the step, the further away from the value of 1 the array generation stops and so, it is further from an infinite value.

The cross-section is integrated using the Monte-Carlo method, as previously used in Problem Set 2 (Eq. 16) and is printed on the second line in the output file.

$$\int_a^b f(x)dx =< f > (a-b), \tag{16}$$

where $< f >$ is the average value of the function.

4 Goal 3

A class is created to read data from PDF data file *pdf.dat* . It takes as inputs the value for x,Q and PID and returns the PDF for the parton with a momentum fraction x and at an energy scale Q. Each line contains $f(x, Q)$ for each of the partons with PID -5, -4, -3, -2, -1, 1, 2, 3, 4, 5, 21.
First, the file is read line by line in the method $read\_and\_populate\_with\_content()$ and a list is created with values for x and Q. A list of lists, the equivalent of the (x,Q) grid, contains the data for each PID. The Particle ID is given as an input together with a x and Q when class is called. A method called $get\_pdf\_matrix\_based\_on\_pid()$ gets the index of the PID from a list containing them and using this index it gets the specific lists of lists from the lists of lists of lists with the data.
This model is similar to a 3x3 matrix. One dimension is for x, one for Q and one for the PID.
Next, using the $interp2d$ from the Scipy package, the interpolation is used on x,Q and the list with data for the specific PID. It then outputs $f(x, Q)$ where x and Q are given by the user.
Interpolation is needed due to the non-continous character of the information.This is done in method $interpolate\_function()$ X is given with a step of 10^{-6} taking values in the interval $[0, 1]$. Q increases as a power series of 2 from 1 to 512. If the input x and Q are not in the file, but in between 2 values, the function should generate what the value of the PDF should be. The output is given through the method $get()$.

5 Goal 4

Putting together Goal 1,2 and 3 a function is defined, *gen_from_pdf*() which takes as inputs the name of the function to be used for $d\sigma$, number of steps for integration and PID. It randomly generates values for x_1 and x_2. Using these values it calculates $\hat{s}$ as:

$$\hat{s} = x_1 x_2 s$$

S is chosen to be 512 GeV/c^2, which is the maximum Q in the *pdf.dat* input file for x_1 and x_2 taking values from 0 to 1. This way s can cover the whole Q range.
The hit and miss method of the *event_gen* class is then called to generate one event for the calculated $\hat{s}$. The output vectors are printed in a text file together with the s value for which they were generated. At the end of the file the total cross-section is also printed.
The total cross-section integral (Eq.17) can be factorised if the energy scale is independent of $\hat{\theta}$.

$$\sigma = \int_{x_1=0}^{x_1=1} dx_1 \int_{x_2=0}^{x_2=1} dx_2 \int_{\cos\hat{\theta}=-1}^{\cos\hat{\theta}=1} d\cos\hat{\theta} f_1(x_1,Q) f_2(x_2,Q) \frac{d\hat{\sigma}(\hat{s},\cos\hat{\theta})}{d\cos\hat{\theta}} \quad (17)$$

Eq. 17 becomes:

$$\sigma = \int_{x_1=0}^{x_1=1} f_1(x_1,Q)dx_1 \int_{x_2=0}^{x_2=1} f_2(x_2,Q)dx_2 \int_{\cos\hat{\theta}=-1}^{\cos\hat{\theta}=1} d\cos\hat{\theta} \frac{d\hat{\sigma}(\hat{s},\cos\hat{\theta})}{d\cos\hat{\theta}} \quad (18)$$

Each of the three integrals in Eq.18 can be treated separately using the Monte Carlo integration method (Eq.16). When x_1 and x_2 are sampled they are used to generate the PDF using the class created at Goal 3. Q is set as $\hat{s}$. Then a $\cos\hat{\theta}$ is sampled and the function is calculated for it and $\hat{s}$.
The product of the three integrals calculated using Monte Carlo is returned as the total cross-section.

6 Conclusion

The aims of this project were to tranform functions for the derivative of cross section, to integrate them and use them to generate particle events; to return PDF for a given parton and to calculate the total cross-section using the defined classes. They were achieved and the speed and acquracy of the results depend on the magnitude of the step taken and on how intervals for $\cos\hat{\theta}$ are defined. It can be investigated further how the value of Q set as $\hat{s}$ is chosen and more functions can be defined for generation of different events. The current project limits to quark scattering, gluon scattering and Higgs production. For the first two, the probabilities of generation are low, but for Higgs production it is always one, as expected from the fact that the derivative of cross-section is a constant.

References

[1] `https://canvas.bham.ac.uk/courses/35168/pages/2-cross-section-generation?module_item_id=1082290`

[2] `https://twiki.cern.ch/twiki/bin/view/LHCPhysics/CERNYellowReportPageBR`

[3] `http://bolvan.ph.utexas.edu/~vadim/Classes/11f/STU.pdf`