Monte Carlo Tools in Particle Physics

Master Laboratory Course (2 Days)

Prof. Dr. Wolfgang Kilian Tobias Striegl

Winter Term 2024/2025

Contents

1 First Day **3**
1.1 Prerequisites . . . 3
1.1.1 The Standard Model of Particle Physics . . . 3
1.1.2 Monte Carlo Techniques . . . 4
1.1.3 WHIZARD . . . 6
1.1.4 Install WHIZARD . . . 7
1.2 Exercises . . . 9

2 Second Day **11**
2.1 Prerequisites . . . 11
2.1.1 Phase Space and Kinematic Observables . . . 11
2.1.2 The Electroweak Interaction and Its Feynman Rules . . . 13
2.1.3 Feynman Diagrams and Perturbation Theory . . . 14
2.2 Exercises . . . 16

Introduction

During this laboratory course, you will be introduced to the fundamentals of modern particle physics by developing a basic understanding of general-purpose Monte Carlo event generators (MCEGs) [1].

In particular, you will be introduced to the `WHIZARD` [2] event generator, which is a sophisticated tool for automatic simulations of particle collisions and decays.

Before any simulations, a basic understanding of particle physics and Monte Carlo methods is required. This will be covered in chapters 1.1.1 and 1.1.2. Furthermore, the *Standard Model of particle physics* (SM) will be presented in Chapter 1.1.1, focusing on the *electroweak force* later in Chapter 2.1.2. The basics of Monte Carlo integration and its application to event generation will be discussed in Chapter 1.1.2.

You will be introduced to the concepts of particle scattering and particle decays. Moreover, you will learn how such processes are calculated in *quantum field theory* (QFT) in Chapter 2.1.1. From this, you can see the need for MCEGs such as `WHIZARD` , a MCEG that can calculate such objects numerically as the analytic expressions grow factorially.

Since installing `WHIZARD` , or any other particle physics software, is somewhat complex, the installation of `WHIZARD` is part of the preparation for this course.

On the first day (part 1), you will study and manipulate a given simulation setup in order to understand the syntax of `WHIZARD` 's scripting language, `sindarin` , and develop a basic understanding of particle reaction simulations.

The second day will be reserved for writing your own simulation scripts. More details will be provided in part 2.

1 First Day

1.1 Prerequisites

1.1.1 The Standard Model of Particle Physics

The *Standard Model of Particle Physics* (SM) [3–5] is a relativistic quantum field theory that describes three of the four known fundamental interactions. It includes three generations of *quarks* and *leptons*, both of which are fermions. Together, they are the constituents of all visible matter. Furthermore, the SM contains *vector bosons* that act as the mediators of all represented fundamental interactions. Finally, one *scalar boson*, the Higgs boson h, is included in the Standard Model. It plays a role in the unification of the weak and electromagnetic interactions and gives masses to all SM particles.

The *strong interaction* is responsible for binding quarks to hadrons, such as neutrons and protons, as well as for the cohesion of the atomic nucleus. This interaction is mediated by the so-called *gluons* g, which are massless bosons. This interaction will not be part of this course.

The well-known *electromagnetic interaction*, which is mediated by the massless *photon* γ, is also included in the Standard Model. It mediates interactions between electrically charged particles.

The *weak interaction* is the force behind the radioactive decays of matter. Its mediators are the massive $W^{\pm}$ and Z bosons. In comparison to the electron, these bosons are very heavy, with $m_W = 80.379 \pm 0.012$ GeV[1][2] and $m_Z = 91.1876 \pm 0.0021$ GeV. Moreover, the Higgs boson's mass is about the same order of magnitude, $m_h = 125.15 \pm 0.17$ GeV.

The electromagnetic and weak interactions are unified in the *electroweak interaction*, in which the Higgs boson h plays an important role. In this sophisticated mechanism, h is expanded around its vacuum expectation value[3] and gives mass to all SM particles. Quarks, leptons, as well as the gauge bosons of the weak interaction, are affected by it. This course revolves around this interaction. Further information about it will be given in 2.1.2. The Higgs boson also interacts with other particles and couples to all SM particles proportional to their masses.

The *gravitational interaction*—the last of the four fundamental interactions—is not included in the Standard Model.

The SM's charged leptons —the electron e^-, the muon μ^-, and the tau τ^-—each have corresponding anti-particles with opposite charge. Their anti-particles are the positron e^+, the anti-muon μ^+, and the anti-tau τ^+. The masses of the charged leptons vary over several orders of magnitude, from the electron $m_e = 0.5109989461 \pm 0.0000000031$ keV, to the muon $m_\mu = 105.6583745 \pm 0.000002$ MeV, and finally to the tau $m_\tau = 1776.86 \pm 0.12$ MeV.

Each charged lepton comes with a corresponding neutrino (ν_e, ν_μ, ν_τ). Neutrinos are electrically neutral particles that interact only via the weak force.They all have very low masses, $m_\nu < 2$ eV. Like all other SM fermions, the neutrinos also have corresponding anti-particles ($\bar{\nu}_e$, $\bar{\nu}_\mu$, $\bar{\nu}_\tau$).

The second group of SM fermions consists of the quarks, which come in six different *flavors*.These include the up u, strange s, and bottom b quarks, all having a fractional charge of $2/3$.

[1] Note that we use natural units in particle physics, so the speed of light c, the Planck constant $\hbar$, the gravitational constant G, and the Boltzmann constant k_b are all set to $c = \hbar = G = k_b = 1$. This leads to mass being expressed in energy units. In SI units, mass would have $[m] = \frac{\text{GeV}}{c^2}$. In these units, the mass of the proton is $m_p \approx 1$ GeV.

[2] All numeric values can be found in the *Particle Data Book* (PDB) [6].

[3] You may think of this as the expansion around its degenerate minimal values.

The remaining three quarks, with charge $^{-1}/_{3}$, are the down d, charm c, and top t. The quark masses also vary over several orders, from $m_u \approx 2$ MeV to $m_t = 172.76 \pm 0.3$ GeV, which is the heaviest particle in the Standard Model. Each quark has an associated anti-particle that has almost the same properties but the opposite electric charge.

The SM is primarily used to calculate so-called *cross sections* of particle collisions and *decay widths* of particles. Both are Lorentz-invariant measures of the probability for a particle reaction to occur. Calculating them involves solving often complicated multidimensional integrals in the relativistic phase space 2.1.1. In this course, you will use Monte Carlo techniques explained in Chapter 1.1.2 to solve those integrals.

Particle Family	Particles	C	EM	weak	strong
Charged Leptons	e^-, μ^-, τ^-	-1	x	x	
Neutrinos	ν_e, ν_μ, ν_τ	0		x	
Up Quarks	u, c, t	$^{2}/_{3}$	x	x	x
Down Quarks	d, s, b	$^{-1}/_{3}$	x	x	x

Table 1: Information about the interactions SM fermions are involved in. C is the fermion's charge, and 'x' indicates involvement in this interaction. Note that each fermion has a corresponding anti-fermion with the opposite charge.

1.1.2 Monte Carlo Techniques

Monte Carlo: A First Example

As a first example, we apply a Monte Carlo approach to calculate the value of the irrational number π. For that, we consider a circle with radius $^{r}/_{2}$ inside a square with length r. The ratio R is defined as

$$\mathrm{R} = 4 \cdot \frac{A_{\text{circle}}}{A_{\text{quad}}} = 4 \cdot \frac{\left(\frac{r}{2}\right)^2 \pi}{r \cdot r} = \pi \quad . \tag{1.1}$$

Since this ratio R is independent of the radius r, we can choose $r = 1$ from now on.

Now we generate N pairs of random numbers x_i and y_i with $x_i, y_i \in [0, 1]$. Let M be the number of pairs that match the condition

$$x_i^2 + y_i^2 \leq 1 \quad . \tag{1.2}$$

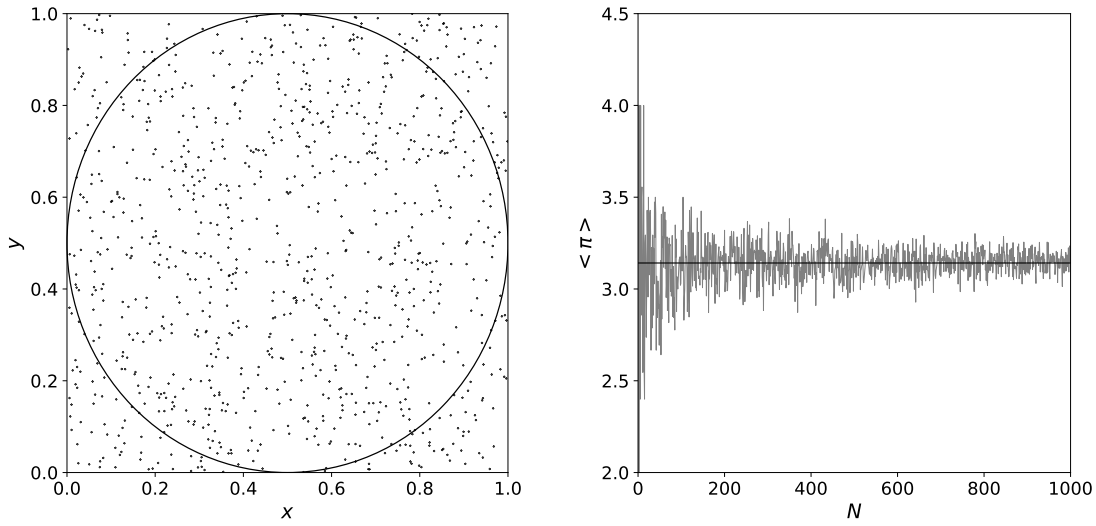
For a visualization of this condition, see Figure 1(a).

Hence, M is the number of points inside the circle, and N is the total number of points in the square.

According to 1.1, π can be estimated by

$$\langle \pi \rangle = 4 \cdot \frac{M}{N} \quad . \tag{1.3}$$

The convergence of this algorithm is shown in Figure 1(b).



(a) Scattering of the random numbers

(b) Convergence of the hint-or-miss algorithm

Figure 1: (a) Distribution of $N = 1000$ random number pairs (x_i, y_i). (b) Estimation $\langle \pi \rangle$ as a function of the number of iterations of the presented algorithm N. The gray line corresponds to the estimation, while the black line corresponds to the real value of π.

Monte Carlo Integration

Since evaluating—especially multidimensional—integrals can be a tedious task, various deterministic and non-deterministic methods have been developed to calculate them. One of these is the so-called *Monte Carlo* integration [7]. This is a non-deterministic numeric technique that uses random numbers to approximate the value of an integral.Due to its non-deterministic nature, the outcome of each integration is different.

First, consider an n-dimensional real integral I of the regular function $\mathrm{f}(\vec{x}) < \mathrm{f}_{\max}$:

$$\mathrm{I} = \int_{\Omega} \mathrm{d}\vec{x}\, \mathrm{f}(\vec{x})\,, \quad \Omega \subset \mathbb{R}^n \quad . \tag{1.4}$$

Obviously,

$$\mathrm{I} < \mathrm{f}_{\max} \cdot \int_{\Omega} \mathrm{d}\vec{x} := \mathrm{V} \quad . \tag{1.5}$$

Now consider each $\vec{x}_i \in \Omega$ as a vector of (pseudo) random numbers[4]. Therefore, we can approximate

$$\mathrm{I} \approx \mathrm{P}_N := \frac{\mathrm{V}}{N} \sum_i^N \mathrm{f}(\vec{x_i}) = \mathrm{V} \cdot \mathrm{E[f]} \quad . \tag{1.6}$$

Due to the *law of large numbers*,

$$\mathrm{I} = \lim_{N \to \infty} \mathrm{P}_N \quad . \tag{1.7}$$

Therefore, the convergence of the integral is ensured for a sufficiently large number N.
By using some properties of the variance $V[P_N] = \sigma^2_{P_N}$, you can show that

$$\sigma_{\mathrm{P}_N} \sim \frac{1}{\sqrt{N}} \quad . \tag{1.8}$$

Note that the accuracy of the integration is dependent on the number of iterations N, and the scaling of its standard deviation σ_{P_N} can be estimated by $\frac{1}{\sqrt{N}}$.
However, this cannot be understood as a strict error bound since a particular realization of $\mathrm{P_N}$ may not cover all important features of an integral.
In contrast to many other integration methods, the error of the Monte Carlo integration does not scale with the integral's number of dimensions. This may be its most significant advantage over deterministic integration methods.

1.1.3 WHIZARD

`WHIZARD` [5] [2] is a general-purpose MCEG designed for the efficient calculation of multi-particle scattering cross sections and simulated event samples.
The program automatically computes complete *tree-level* matrix elements, integrates them over phase space, and evaluates distributions of (kinematic) observables like the invariant mass or the scattering angle.
There is no conceptual limit on the process complexity; using current hardware, the program has successfully been applied to scattering processes with up to eight particles in the final state.
The Standard Model of particle physics, as well as many other models, has been implemented.
`WHIZARD` 1.1.4 will be used for all calculations in this course. You will be guided through the installation of `WHIZARD` in this chapter and will run and investigate a first example on the first day.

[4] Pseudo random numbers are produced algorithmically from so-called Markov chains and therefore depend on an initial value called the *seed*.
[5] `https://whizard.hepforge.org/`

1.1.4 Install WHIZARD

Since installing WHIZARD might be a non-trivial task—especially for someone not familiar with Linux—the installation is explained in this chapter. You are expected to install WHIZARD on your own laptop as preparation for the first experiment so that at least one participant has a functional WHIZARD installation on a notebook. If the installation is not possible on your computer for some reason, or if you don't have a notebook, you will find WHIZARD installed on a computer at the university, and you can also do the installation on your own there.

However, installing WHIZARD is part of this course, so you should at least try it on your own. For further information on the WHIZARD installation, see.[6]

Note that running WHIZARD on Windows is not supported. Therefore, you need to set up a Linux installation first. You can set up a VirtualBox with a Linux distribution; Ubuntu LTS or Debian, for example, would be sufficient.[7][8] If you are running Windows 10, you might alternatively set up its Linux subsystem (WSL) supported by Microsoft and install WHIZARD there[9]. Furthermore, you can install a second OS on your computer and run it in dual boot.

The packages required for WHIZARD on Debian are at least:

- build-essential
- unzip
- tar
- make
- gfortran
- libglib2.0-dev
- gcc and g++
- libtool
- autoconf
- graphviz
- cm-super
- ocaml
- ocamlweb
- python-is-python3
- context
- latex, e.g., texlive and texlive-metapost if plots from WHIZARD are desired.

[6] https://whizard.hepforge.org/manual/manual003.html
[7] Download https://www.virtualbox.org/wiki/Downloads
[8] Set up VirtualBox with Ubuntu: https://www.wikihow.com/Install-Ubuntu-on-VirtualBox
[9] Choose manual WSL install: https://docs.microsoft.com/en-us/windows/wsl/install-win10

Depending on your Linux distribution, some further packages might be required. The name of your package manager depends on your Linux distribution. On Debian, its name is apt-get. To install WHIZARD on your own computer, you can download it here[10].

Open a terminal on your computer and go to the directory where the installation file is located. The file is called whizard-3.x.x.tar.gz. The file ending tar.gz means that it's a compressed bundle of files. It consumes less space and is easier to transport. First, you need to *untar* it.

```
<path>/$ tar -xzvf whizard-3.x.x.tar.gz
```

This command will unpack the files into a folder called whizard-3.x.x.In it, you will find a file called makefile.

You can think of it as a recipe for building WHIZARD . It contains all the necessary steps to build the functional program.

Create the folders build and install next.

```
<path>/$ mkdir build install
```

The install folder is where you will find the executable WHIZARD after the complete installation.The build folder will be used to install WHIZARD from within it. Now change to it.

```
<path>/$ cd build
```

Inside the build folder, run:

```
<path>/build/$ ../whizard-3.x.x/configure --prefix=<path>/install/
```

This will set up everything for the installation. You don't need any additional options for this course, but if you would like to use WHIZARD later in a more advanced setting, you might specify something here. See the manual for more details here[11]. The --prefix option is used to specify your installation directory. Otherwise, WHIZARD would be installed directly into the build folder.

Next, you need to install the program.

```
<path>/build/$ make && make install && make check
```

This will execute all three commands after the previous command has fully executed without any errors. make will prepare the installation, make install will perform the installation, and make check will verify if everything is installed correctly. Note that this will take some time. If make check does not encounter any failures (xfails are fine), you have installed WHIZARD successfully. "'

[10] https://whizard.hepforge.org/downloads/
[11] https://whizard.hepforge.org/manual/manual003.html

1.2 Exercises

After the installation is complete, you will find a folder /share/whizard/examples inside the `install` folder. The examples all end with `.sin`. They are called `sindarin` files since `sindarin` is `WHIZARD` own scripting language. More information about `WHIZARD` 's scripting language `sindarin` can be found here[12].

Now, create a new folder for your first `WHIZARD` run and copy `Z-lineshape.sin` into it. Open the file Z-lineshape.sin and study the commands. You might understand a few lines without knowing anything about `sindarin` .

Then, run `WHIZARD` inside your newly created folder and take a look at the output files.

```
<path>/test_run/$ <installation-path-of-whizard>/bin/whizard Z-lineshape.sin
```

Find the results of your calculation. Does the size of the errors meet your expectations? Find the *seed* of your run. Study the file and try to determine the meaning of each command.

Complete the following tasks, each starting from `Z-lineshape.sin` without making any changes:

A1. First, add the error value to your output data, then remove the integration of `realcorr`. Now lower the number of iterations for `bornproc` to $2 : 250 : "gw", 1 : 100$, and then further to $1 : 10 : "gw", 1 : 5$. Document your results for both runs. Compare your results.

A2. Remove all `cuts` from the `sindarin` file.

A3. Again, remove the integration of `realcorr` and set the Z-Boson's mass to $M_Z = 100$ GeV. Adapt `scan` and `integration` to these changes.

A4. Remove the integration of `realcorr` again and set the Z-Boson's width to $W_Z = 5$ GeV. Adapt `scan` and `integration` to these changes.

Document and describe your results in your report. Do your results meet your expectations? Does the error of your results align with your expectations? Furthermore, describe the meaning of the changes you made in the `sindarin` files and adapt the plot description accordingly. Make sure that your `sindarin` files don't contain artifacts from previous runs and comment on them reasonably and meaningfully.

[12] https://whizard.hepforge.org/manual/manual005.html#sec66

```
# We choose our favourite model
model = SM

# Define incomming particle beam
beams = e1, E1

# Define some particle containers for the cuts
alias lep = e1:E1:e2:E2
alias prt = lep:A

# These are the two processes we want to compare
process bornproc = e1, E1 => e2, E2
process realcorr = e1, E1 => e2, E2, gamma

# Compile model and process information
compile

# This is a cut on the phase space. If 'true' the matrix element will be set to zero

cuts = all E >= 100 MeV [prt]
   and all abs (cos(Theta)) <= 0.99 [prt]
   and all M2 >= (1 GeV)^2 [prt, prt]

# Define title and labels as global variables that will be used in the plot:
$description = "A WHIZARD 3.0 Example"
$x_label = "$\sqrt{s}$/GeV"
$y_label = "$\sigma(s)$/pb"
x_min = 88 GeV
x_max = 95 GeV

# Allocate one plot
$title = "The Z Lineshape in $e^+e^-\to\mu^+\mu^-$"
plot lineshape_born

# Allocate another plot
$title = "The Z Lineshape in $e^+e^-\to\mu^+\mu^-\gamma$"
plot lineshape_realcorr {$draw_options = "withcolor blue"}

# Compute the cross sections for different sqrts
# with smaller steps around the peak
scan sqrts = ((88.0 GeV => 90.0 GeV /+ 0.5 GeV),
              (90.1 GeV => 91.9 GeV /+ 0.1 GeV),
              (92.0 GeV => 95.0 GeV /+ 0.5 GeV)) {
  integrate (bornproc)    { iterations = 2:1000:"gw", 1:2000 }
  record lineshape_born     (sqrts, integral (bornproc) / 1000)
  integrate (realcorr)    { iterations = 5:3000:"gw", 2:5000 }
  record lineshape_realcorr (sqrts, integral (realcorr) / 1000)
}

# Combine the plots to one graph
$title = "The Z Lineshape in $e^+e^-\to\mu^+\mu^-(\gamma)$"
graph g1 = lineshape_born & lineshape_realcorr { $draw_options = "withcolor blue" }

compile_analysis { $out_file = "Z-lineshape.dat" }
```

2 Second Day

2.1 Prerequisites

2.1.1 Phase Space and Kinematic Observables

As you already know, all possible states of a system correspond to a specific point in phase space. Since particle physics is a relativistic theory, its phase space is spanned by the 4-vectors of positions $Q = (t, \vec{x})$ and momenta $P = (E, \vec{p})$. However, calculations turn out to be most conveniently performed in momentum space. Therefore, we define incoming particles $A_m(P_m)$ and outgoing particles $B_n(K_n)$ in a particular particle reaction. Of course, all possible trajectories in particle physics are constrained by momentum and energy conservation.[13]

An arbitrary particle reaction can be written as

$$A_1(P_1), \ldots, A_m(P_m) \to B_1(K_1), \ldots, B_n(K_n) \quad . \tag{2.1}$$

In practice, however, only particle decays 2.2 and collisions 2.3 are considered:

$$A(P) \to B_1(K_1), \ldots, B_n(K_n) \tag{2.2}$$

$$A_1(P_1), A_2(P_2) \to B_1(K_1), \ldots, B_n(K_n) \tag{2.3}$$

The cross section σ and decay width Γ both are Lorentz-invariant measures for the probabilities of particle reaction rates and can be calculated from a corresponding matrix element $\mathcal{M}$.

The matrix element $\mathcal{M}$ itself contains all information corresponding to a particular process. Therefore, all formulas in this chapter for 2.2 and 2.3 are universal and valid for all particle reactions and do not depend on a particular theory.

According to *Fermi's golden rule*, we can calculate these probabilities knowing the phase space and a matrix element $\mathcal{M}$.

$$\mathrm{d}P(A_1, A_2, \ldots \to B_1, B_2, \ldots) = |\langle B_1, B_2, \ldots | A_1, A_2, \ldots \rangle|^2 \tag{2.4}$$

$$= (2\pi)^4 \delta^{(4)} \left(\sum_i P_i - \sum_i K_i \right) |\mathcal{M}|^2 \prod_i \frac{\mathrm{d}^3 \vec{k_i}}{(2\pi)^3 2E_i} \prod_j \frac{\mathrm{d}^3 \vec{p_j}}{(2\pi)^3 2E_j}. \tag{2.5}$$

Note that the delta distribution $\delta^{(4)}$ reflects the overall momentum and energy conservation, and the factor $(2\pi)^4$ comes from its normalization.

The decay rate Γ and cross section σ will both be presented in their differential form since their total values can always be calculated by integrating over all variables.

In this course, `WHIZARD` will be used to calculate the cross section σ since it is able to handle the integration of phase space and matrix elements properly in a numerical way.

[13] `https://pdg.lbl.gov/2021/web/viewer.html?file=%2F2021/reviews/rpp2020-rev-kinematics.pdf`

Decay Width

In its differential form, the decay rate Γ is given by

$$\mathrm{d}\Gamma(P, K_1, \dots) = \frac{(2\pi)^4}{2M} \cdot |\mathcal{M}|^2 \cdot \delta^{(4)} \left(P - \sum_i K_i \right) \prod_i \frac{\mathrm{d}^3 \vec{p_i}}{(2\pi)^3 2E_i} \quad . \tag{2.6}$$

The factor $2M$ can be obtained from the phase space by using 4-momentum conservation $M^2 = P^2 = (K_1 + K_2)^2$ and some properties of the delta distribution. Note that M is the mass of the decaying particle.

For a $1 \to 2$ decay in the decaying particle's rest frame, one obtains

$$\mathrm{d}\Gamma(P, K) = \frac{1}{32\pi^2} \cdot |\mathcal{M}|^2 \cdot \frac{|\vec{k}|}{M^2} \mathrm{d}\cos\theta \mathrm{d}\varphi \quad . \tag{2.7}$$

Branching Ratios

For a decaying particle A and a particular decay channel $A \to B_1, \dots, B_n$, the so-called *branching ratios* $\mathcal{B}$ are often listed, where

$$\mathcal{B} = \frac{\Gamma_{A \to B_1, \dots, B_n}}{\Gamma_{A \to X}} \quad . \tag{2.8}$$

In this notation, X symbolizes the sum of all possible decay channels; therefore, $\mathcal{B} \in [0, 1]$.

Cross Section and Luminosity

Since it differs only by one more incoming particle in the phase space, the cross section σ has a similar form to the decay rate. Again, $\delta^{(4)}$ reflects the overall momentum and energy conservation, and the different prefactor can be obtained in a similar way:

$$\mathrm{d}\sigma(P_1, P_2, K_1, \dots) = \frac{(2\pi)^4}{\sqrt{(P_1 \cdot P_2)^2 - (m_1 m_2)^2}} \cdot |\mathcal{M}|^2 \cdot \delta^{(4)} \left(P_1 + P_2 - \sum_i K_i \right) \prod_i \frac{\mathrm{d}^3 \vec{p_i}}{(2\pi)^3 2E_i} \tag{2.9}$$

$$\approx \frac{(2\pi)^4}{4s} \cdot |\mathcal{M}|^2 \cdot \delta^{(4)} \left(P_1 + P_2 - \sum_i K_i \right) \prod_i \frac{\mathrm{d}^3 \vec{p_i}}{(2\pi)^3 2E_i} \quad . \tag{2.10}$$

Here $s = (P_1 + P_2)^2$ is one of the three so-called *Mandelstam variables*, which are a convenient set of Lorentz-invariant variables to parameterize the kinematics in $2 \to 2$ scattering processes. Note that $\sqrt{s}$ is approximately the same as the collider's energy E for $s \gg m_i^2$. In the center of mass frame (CMS),

$$\sqrt{s} = \sqrt{(P_1 + P_2)^2} \approx \sqrt{2\, P_1 \cdot P_2} = E_{\mathrm{CMS}} \quad . \tag{2.11}$$

For our purposes, it is furthermore reasonable to assume $s \gg m_i^2$. Therefore, for a $2 \to 2$ scattering, one obtains by using the delta distribution:

$$\mathrm{d}\sigma(P_1, P_2, K_1, K_2) = \frac{1}{64\pi^2 s} \cdot |\mathcal{M}|^2 \cdot \mathrm{d}\cos\theta \mathrm{d}\varphi \quad . \tag{2.12}$$

To calculate the number of events N for a certain type of interaction derived from the cross section σ in a real particle collider setting, the luminosity L is also necessary. In contrast to the cross section, which contains all information for particle reactions, the luminosity reflects the properties of the particle collider, such as the collision rate of the initial state particles. Therefore,

$$N = \int \mathrm{d}t L(t)\sigma(t) \quad . \tag{2.13}$$

Assuming neither the cross section nor the luminosity has an explicit time t dependence,

$$N = T \cdot L \cdot \sigma \tag{2.14}$$

where T is the running time of the collider.

Angular Distribution

The distribution for any (kinematic) observable can be derived from 2.6 or 2.12 by elementary differential operations and integrating out the other dependencies.

One of them is the angular distribution. Due to the conservation of energy and momentum, the only nontrivial distribution for a $2 \to 2$ process is the distribution of the scattering angle θ, the angle between a final state particle and the beam axis. Therefore,

$$\frac{\mathrm{d}\sigma}{\mathrm{d}\theta} = \frac{\sin\theta}{32\pi s} \cdot |\mathcal{M}|^2 \quad . \tag{2.15}$$

For further information on the kinematics of particle colliders, see [8].

2.1.2 The Electroweak Interaction and Its Feynman Rules

As mentioned in 1.1.1, one of the fundamental forces of nature is the *electroweak interaction*. In a simplistic way, it can be understood as a unification of the *electromagnetic* and the *weak* forces.

In the Standard Model, this unification is derived by the so-called *Higgs mechanism*, which also generates the masses for all particles.

In contrast to the electromagnetic interaction, which is mediated by the massless photon γ, the weak interaction is mediated by the massive vector bosons $W^\pm$ and Z and the massive *Higgs* boson h.

The electroweak interaction conserves the electromagnetic charge as well as the lepton number, both in total and for each generation separately. The charges and lepton numbers of all leptons and the electroweak mediators can be found in Table 2.

Particle	C	L_e	L_μ	L_τ
$e^\pm$	±1	∓1	0	0
$\mu^\pm$	±1	0	∓1	0
$\tau^\pm$	±1	0	0	∓1
$\nu_e/\bar{\nu}_e$	0	∓1	0	0
$\nu_\mu/\bar{\nu}_\mu$	0	0	∓1	0
$\nu_\tau/\bar{\nu}_\tau$	0	0	0	∓1
$W^\pm$	±1	0	0	0
Z,h,γ	0	0	0	0

Table 2: Information about the quantum numbers of all leptons and the mediators of the electroweak interaction. C is the charge and L_i is the lepton number in each generation.

2.1.3 Feynman Diagrams and Perturbation Theory

Even if it is usually not possible to calculate the matrix element of an interacting theory exactly, one can approach the solution perturbatively. This is where the so-called *Feynman rules* of the theory come into play. Each Feynman diagram is a pictorial representation of a mathematical expression contributing to the matrix element. It is an intuitive way to get all contributions to a certain order of perturbation theory for a particular process.

There are three types of elements a Feynman diagram can be built from. First, there are vertices. They can be understood as an interaction of several particles at a point in spacetime. Each of them is proportional to an order of the coupling constant of the theory it was derived from. For this course, we oversimplify it by calling the coupling constant just α_{EW}. In reality, there is more than one coupling constant involved, and one has to be careful to get all contributions to obtain a certain precision.

The electroweak interaction involves vertices shown in Figure 2.

The different line styles involving bosons have no meaning other than to distinguish vector bosons from the scalar Higgs h. However, for fermions, the direction of the arrow distinguishes a fermion from its anti-fermion, which becomes especially important when momentum is assigned to the particles.

Using Table 2, you might check that each of those vertices conserves charge C and the lepton number in each generation L_i. This does not happen by accident. It is an important feature of the Feynman rule expansion. Therefore, it's obvious that diagrams built from these vertices also conserve C and L_i.

The other sets of expressions you need to know to calculate a matrix element are rules for the lines themselves. An internal line is called a propagator. The lines corresponding to a final or initial state particle are called external lines and have different expressions than the propagators. For this course, we won't go any deeper into that.

To build a diagram corresponding to a correction of a certain order in perturbation theory, you just need to connect two ends with the same particle together. The initial and final state

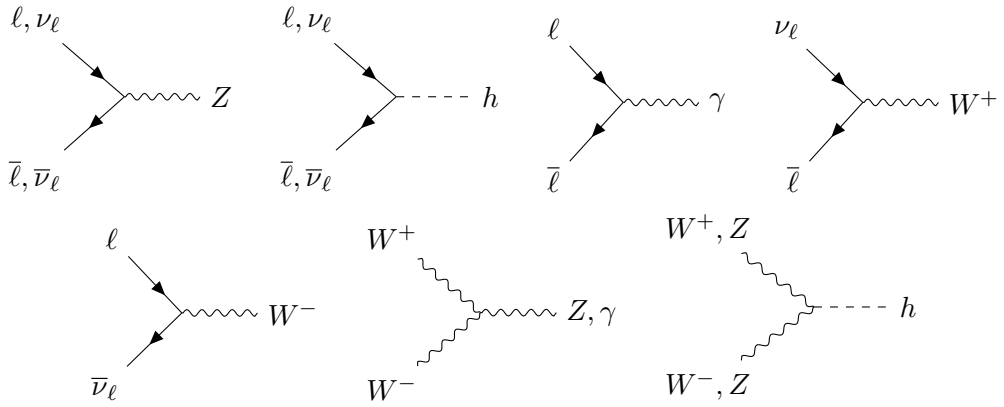


Figure 2: Some vertices of the electroweak interaction. Note that there are also vertices coupling four bosons to each other, but for now we ignore them. You might find them on your own using Table 2.

particles are the left and right-sided loose ends of the diagram.

An exemplary contribution to the matrix element of $e^+e^- \to W^+W^-$ is shown in Figure 3. Of course, there are several more contributing diagrams to this process. You might find the other diagrams yourself.[14]

Of course, you can draw diagrams with more than two vertices, but they would contribute to a different order in perturbation theory. For now, just say that the order in perturbation theory corresponds to the number of vertices in the diagram. However, in a real calculation, it's different since, for example, a vertex itself might have a higher order in the coupling constant.

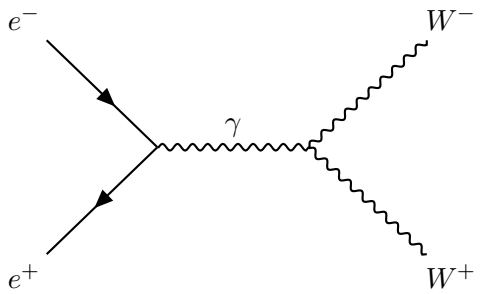


Figure 3: One of the Feynman diagrams contributing to the matrix element of $e^+e^- \to W^+W^-$.

[14] If you want to draw Feynman diagrams in your protocol, see the TIKZ page for LaTeX [9].

2.2 Exercises

Now the goal is to analyze

$$e^+e^- \to W^+W^- \to \ell_1\ell_2\ell_3\ell_4, \quad \ell_i \in \{e, \mu, \tau, \nu_e, \nu_\mu, \nu_\tau\} \quad . \tag{2.16}$$

Therefore, you will simulate two different approximations of the process 2.16:

- For this analysis, you can assume an idealized detector that can detect all final state particles, including neutrinos, under all circumstances. Therefore, there are no blind spots, and all non-identical particles can be distinguished from each other.
- Due to the high collision energy $\sqrt{s}$, lepton masses can safely be neglected since $m_\ell \ll \sqrt{s}$.
- You will find the latest values of masses m and widths w [15] of $W^\pm$, Z, and H in the Particle Data Book (PDG)[16].

Before calculating any processes, think about what subset of final state leptons are possible from a decaying W-Boson pair. Choose your final leptons accordingly for each task.

B1. **Sub-Process Integration**: First, consider $e^+e^- \to W^+W^- \to \ell_1\ell_2\ell_3\ell_4$ and use `WHIZARD` to *scan* over the process energy $\sqrt{s}$ again. As a simplification, use the branching ratios for W^+ and W^- from the PDB instead of calculating the full decays. Adjust the lower bound to the intermediate state. The upper bound should be $\sqrt{s} = 4 \cdot M_W$. The chosen energy steps should provide insight into all relevant phenomena of the process. Discuss your results.

B2. **Hard-Process Integration**: Now consider $e^+e^- \to \ell_1\ell_2\ell_3\ell_4$ and use `WHIZARD` to *scan* over the process energy $\sqrt{s}$. Choose the upper and lower bounds the same as in the previous task. Again, the chosen energy steps should provide insight into all relevant phenomena of the process. To obtain finite results, it is necessary to apply a cut on the kinematics. For this, use $|\cos\theta| < 0.985$. Compare your results with those from the previous task.

B3. **Angular Distribution**: Now consider $e^+e^- \to W^+W^-$ and use `WHIZARD` 's `simulate` command to produce the θ-distribution of W^+ and W^- with $\theta \in [0°, 180°]$. Use `WHIZARD` to generate a histogram[17] of this distribution. The histogram should be normalized to the total cross section of the process. For this, use `$sample_normalization="sigma/N"`. A bin width of 20° will be sufficient. $\sqrt{s}$ should be about the energy of the cross section's maximum from task B2.. If you don't produce enough events in the first run, you can combine the data from several histograms.

[15] An unstable particle in a scattering process shows up as a resonance in the cross section, which can be estimated via a relativistic generalization of the *Breit-Wigner* formula. Therefore, near the resonance, $\mathcal{M} \sim (P^2 - m^2 + \mathrm{i}mw)^{-1}$. For this distribution, the width can be derived.

[16] `https://pdg.lbl.gov/2021/download/db2020.pdf`

[17] An example of how to generate a histogram can be found in `W-endpoint.sin`, which is located in the `examples` folder.

Refinements

If all prior tasks are completed and there is still enough time, you might proceed with the following tasks. Nevertheless, proceeding won't give you any additional or hidden bonus points. Therefore, it's possible to achieve the highest grade in the course without doing any of these.

R1. **Angular Cuts**: Redo tasks B1. and B2. under the assumption that no particles can be detected within a range of $\theta_b = 20°$ around the beam axis. For this, you need to set a `cut` on the matrix element. Compare your results with those of the previous tasks.

R2. **Background**: Analyze the background of the process $e^+e^- \to W^+W^- \to \ell_1\ell_2\ell_3\ell_4$. List all two-boson intermediate states and carefully consider which can be neglected. Compare your results. Redo tasks B1. and B3. for the remaining processes.

Report

- Document the *seeds* used for each run with `WHIZARD` and include them in the description of each of your plots.
- Your protocol must contain your `sindarin` files as well as your full `WHIZARD` data as an appendix.
- Your `sindarin` files are part of your protocol, so ensure that they are well-structured and commented in a way that aids understanding.
- Make sure that your plots have proper x- and y-axis labels, as well as a reasonable title.
- Describe your observations. Do they match your expectations?
- Discuss the error values of your simulations. Do their values seem reasonable?

References

[1] S. Alioli et al., *Monte Carlo event generators for high energy particle physics event simulation*, 1902.01674.

[2] W. Kilian, T. Ohl and J. Reuter, *WHIZARD: Simulating Multi-Particle Processes at LHC and ILC*, *Eur. Phys. J. C* **71** (2011) 1742, [0708.4233].

[3] M. D. Schwartz, *Quantum Field Theory and the Standard Model*. Cambridge University Press, 2014.

[4] M. E. Peskin and D. V. Schroeder, *An Introduction to quantum field theory*. Addison-Wesley, Reading, USA, 1995.

[5] M. Thomson, *Modern particle physics*. Cambridge University Press, New York, 2013.

[6] Particle Data Group collaboration, P. Zyla et al., *Review of Particle Physics*, *PTEP* **2020** (2020) 083C01.

[7] S. Weinzierl, *Introduction to Monte Carlo methods*, hep-ph/0006269.

[8] T. Han, *Collider phenomenology: Basic knowledge and techniques*, in *Theoretical Advanced Study Institute in Elementary Particle Physics: Physics in $D \geq 4$*, 8, 2005, hep-ph/0508097, DOI.

[9] J. Ellis, *TikZ-Feynman: Feynman diagrams with TikZ*, *Comput. Phys. Commun.* **210** (2017) 103–123, [1601.05437].