

In particle and nuclear physics, Monte Carlo simulations are used for a wide variety of purposes, including prediction and interpretation of results, studies of the performance of detectors and their components, as well as corrections of measured data to obtain accurate outcomes that are independent of experimental or instrumental effects. A detailed coverage of all the types of simulations used by particle and nuclear physicists is thus well beyond the scope of this text. We focus our discussion, in §14.1, on basic techniques used in the simulation of simple physics observables, such as momentum spectra or correlation functions, while in §14.2, we present simple techniques used in the simulation of detector performance and data correction for smearing and efficiency effects.

14.1 Event Generators

Simulations of particle production by elementary particles or nuclei collisions are based on phenomenological and theoretical models of varying complexity. A large variety of models predicated on a wide range of assumptions are described in the literature and used as practical event generators by theoreticians and experimentalists to describe and interpret measured results or analyze the performance of particle detectors. Commonly used event generators for the simulation of proton–proton collisions include JETSET [174], PYTHIA [175], HERWIG [66], PHOJET [52], EvtGen [133], and several others. There is an even greater variety of models for the generation and simulation of heavy-ion collisions. Early models used in the simulation of AGS, SPS, and RHIC collisions include HIJING [188], RQMD [176], URQMD [29], AMPT [137], VENUS [191], and EPOS [192]. More recent developments for the study of the interaction of jets within the medium produced in heavy-ion collisions include Martini [169], Jewel [198], Cujet [61], YaJEM [164], and many others. As noted earlier, we restrict the scope of our discussion to a small selection of phenomenological and theoretically motivated models, which illustrate how more complex generators are built and can be used to carry out simple simulations and analyses.

14.1.1 Basic Particle Generators

Basic simulations of particle production in elementary particle collisions often assume that produced particles are uncorrelated and have a kinematical distribution determined by a specific PDF. The choice of PDF used in a particular simulation is dictated by the goals of

the study and the needs for realistic reproduction of the attributes of collisions of interest. We focus our discussion on a few illustrative examples.

Isotropic Exponential Distribution

We first consider the simulation of isotropic particle production according to an exponential momentum distribution

$$\frac{dN}{dp} = \frac{1}{\lambda} e^{-p/\lambda} \quad (14.1)$$

over the momentum range $[p_{\min}, p_{\max}]$.

The generation of random momentum vectors requires three random numbers, r_1 , r_2 , and r_3 , in the range $[0, 1]$. Isotropic emission is achieved by generating polar and azimuth angles according to

$$\begin{aligned} \phi &= 2\pi r_1 \\ \cos \theta &= 2r_2 - 1, \end{aligned} \quad (14.2)$$

whereas the generation of particle momenta according to the exponential distribution, Eq. (14.1), can be accomplished using the inversion technique, Eq. (13.22), introduced in §13.3.3:

$$p = -\lambda \ln(r_3 \exp(-p_{\max}/\lambda) + (1 - r_3) \exp(-p_{\min}/\lambda)), \quad (14.3)$$

where $p_{\min}$ and $p_{\max}$ define the range over which the momentum spectrum is to be generated.

Power-Law Distribution

Particle production, particularly at high momentum, can often be described as a power law over a restricted momentum range.

Strictly speaking, a power-law distribution is a function characterized by a scale invariance. For instance, a function of the type $f(x) = ax^k$ is invariant in form under scaling of the argument x by a constant factor c :

$$f(cx) = a(cx)^k = c^k f(x) \propto f(x). \quad (14.4)$$

Indeed, scaling the argument by a factor c only multiplies the amplitude of the function by a constant value. This behavior is in principle readily identified graphically, with data, by plotting the logarithm of the dependent variable f as a function of the logarithm of the independent variable x . The end result is a straight line with a slope equal to the exponent of the power law:

$$\ln(f) = \ln(ax^k) = k \ln(x) + \ln(a). \quad (14.5)$$

Approximate power-law behaviors have been identified in several areas of science (e.g., scale of earthquakes, sizes of moon craters, wealth of individuals, etc.). In particle physics, one observes that the production of high-momentum particles can be described with a

steeply decreasing power law originating from the approximate self-similar behavior associated with parton fragmentation.

Power-law probability distributions are often extended by adding a slowly varying coefficient $S(x)$:

$$f(x) \propto S(x)x^{-\alpha}, \quad (14.6)$$

where $\alpha > 1$, and $S(x)$ is required to satisfy $\lim_{x \rightarrow \infty} S(tx)/S(x) = 1$ for an arbitrary constant $t > 0$. Obviously, for $S(x) = c$, with c a constant, the power law holds for all positive values of x . If $S(x)$ is not exactly constant but varies very slowly, it may be possible to find a minimum value $x_{\min}$ beyond which the power law takes the form

$$p_{\text{pl}}(x) = \frac{\alpha - 1}{x_{\min}} \left(\frac{x}{x_{\min}} \right)^{-\alpha}, \quad (14.7)$$

where the coefficient $(\alpha - 1)/x_{\min}$ defines the PDF's normalization.

Moments μ_n of the power-law function $p_{\text{pl}}(x)$ above the cutoff value $x_{\min}$ are obtained by direct integration. One finds (see Problem 14.1):

$$\mu_n = \int_{x_{\min}}^{\infty} x^n p_{\text{pl}}(x) dx = \frac{\alpha - 1}{\alpha - 1 - n} x_{\min}^n, \quad (14.8)$$

which is strictly defined only for $n < \alpha - 1$. One readily verifies that moments $n \geq \alpha - 1$ indeed diverge. For instance, for $\alpha < 2$, the mean and all higher moments are infinite while for $2 < \alpha < 3$, the mean exists but all other moments (including the variance) do not.

Power-law behaviors are typically observed over a finite range of the independent variable x beyond which they break down. For instance, in particle physics a strict power-law behavior as a function of the momentum (or transverse momentum) of the produced particles would imply that particles with arbitrarily large momenta can be produced. Such a behavior is clearly impossible because it would violate conservation of energy: given a system with a specific collision energy, there is an energy bound beyond which individual particles cannot be produced. This problem can be solved or at the very least suppressed with the introduction of a power law with exponential cutoff:

$$f_{\text{plc}}(x) \propto \left(\frac{x}{x_{\min}} \right)^{-\alpha} e^{-\lambda x}, \quad (14.9)$$

where the exponential $e^{-\lambda x}$ dominates the power law at very large values of x . Strictly speaking, this function does not globally scale as and is not asymptotically a power law, but it does exhibit approximate scaling behavior between the minimum $x_{\min}$ and an upper cutoff determined by the exponential.

Generation of random numbers according to the power law Eq. (14.7) is readily achieved with the transformation method (§13.3.3) since the function $f_{\text{pl}}(x)$ is integrable. One finds (see Problem 14.2) that random continuous values of x distributed according to $f_{\text{pl}}(x)$ can be obtained with

$$x = x_{\min} (1 - r)^{\frac{1}{1-\alpha}}, \quad (14.10)$$

where r is a uniformly distributed random number in the range $[0, 1]$. Note that for application in high energy physics, one may substitute either the momentum p or transverse momentum p_T of the particle for x , as we discuss in the next section.

Flat Rapidity Distribution

The production of particles in high-energy collisions, either proton–proton or nucleus–nucleus, is hardly isotropic and is typically better described with an approximately flat rapidity distribution – particularly in the central rapidity region, i.e., $y \approx (y_{\text{beam}} + y_{\text{target}})/2$. The invariant cross section Eq. (8.73) may then be modeled according to a function of the form

$$E \frac{d^3\sigma}{dp^3} = \frac{d^2\sigma}{\pi dp_T^2 dy} = kf(p_T), \quad (14.11)$$

where $k = d\sigma/dy$ is assumed to be constant in the kinematical range of interest, and $f(p_T)$ is a suitably chosen model for the transverse momentum spectrum of produced particles. Depending on the p_T range of interest, $f(p_T)$ can be chosen to be an exponential function (as in §14.1.1), a power-law distribution (§14.1.1), a Maxwell–Boltzmann momentum distribution (§14.1.1) or other appropriate functions. For illustrative purposes, let us consider the generation of particles with a flat rapidity distribution and a power-law distribution in p_T , which should be suitable (approximately) for the simulation of high p_T particles or jets.

The azimuth angle is assumed to be distributed uniformly and is generated according to

$$\phi = 2\pi r_1, \quad (14.12)$$

where r_1 is a uniformly distributed random number in the range $[0, 1]$. The rapidity distribution is assumed to be flat in a range $[y_{\min}, y_{\max}]$ and is thus generated with

$$y = y_{\min} + r_2 (y_{\max} - y_{\min}), \quad (14.13)$$

where $r_2 \in [0, 1]$. Finally, the p_T of the particle is produced according to

$$p_T = p_{T,\min} (1 - r_3)^{\frac{1}{1-\alpha}}, \quad (14.14)$$

with $r_3 \in [0, 1]$, a preset minimum transverse momentum $p_{T,\min}$, and a suitably chosen value of α . The energy and moment components of the particle may then be obtained according to Eqs. (8.39 and 8.41):

$$E = m_T \cosh y, \quad (14.15)$$

$$p_x = p_T \cos \phi, \quad (14.16)$$

$$p_y = p_T \sin \phi, \quad (14.17)$$

$$p_z = m_T \sinh y, \quad (14.18)$$

with $m_T = \sqrt{p_T^2 + m^2}$, where m is the mass of the particle.

In certain situations, it might be desirable to produce particles according to a flat pseudorapidity, η , distribution. One then generates η according to

$$\eta = \eta_{\min} + r_2 (\eta_{\max} - \eta_{\min}). \quad (14.19)$$

The polar angle θ of the particle is obtained by inverting Eq. (8.31)

$$\theta = 2 \tan^{-1} [\exp(-\eta)], \quad (14.20)$$

and the momentum components are

$$p_x = p_T \cos \phi \sin \theta, \quad (14.21)$$

$$p_y = p_T \sin \phi \sin \theta, \quad (14.22)$$

$$p_z = p_T \cos \theta, \quad (14.23)$$

while the energy is

$$E = \sqrt{p_x^2 + p_y^2 + p_z^2 + m^2}. \quad (14.24)$$

Maxwell–Boltzmann Momentum Distribution

The Maxwell–Boltzmann distribution, introduced in §3.16, describes the momentum distribution of molecules of nonrelativistic systems in (near) thermodynamic equilibrium. It is often used to model thermalized particle production in heavy-ion collisions.

It is convenient to express the Maxwell–Boltzmann distribution as

$$f_{\text{MB}}(\vec{p}) = \left(\frac{1}{2\pi mkT} \right)^{3/2} \exp \left(-\frac{p_x^2 + p_y^2 + p_z^2}{2mkT} \right), \quad (14.25)$$

which may also be written

$$f_{\text{MB}}(\vec{p}) = \frac{1}{N} \frac{d^3N}{p^2 dp d\cos\theta d\phi} = f_{1D}(p_x|T) f_{1D}(p_y|T) f_{1D}(p_z|T), \quad (14.26)$$

with

$$f_{1D}(p_i|T) = \left(\frac{1}{2\pi mkT} \right)^{1/2} \exp \left(-\frac{p_i^2}{2mkT} \right). \quad (14.27)$$

One can then generate momenta distributed according to a Maxwell–Boltzmann distribution using three random Gaussian deviates p_x , p_y , and p_z determined by

$$P(p_i) \propto \exp \left(-\frac{p_i^2}{2\sigma^2} \right), \quad (14.28)$$

with $\sigma = \sqrt{mkT}$. The generation of p_x , p_y , and p_z as Gaussian deviates produces an isotropic distribution. In order to obtain particles with a flat rapidity distribution and a Maxwell–Boltzmann transverse profile, it suffices to generate p_x and p_y according to the preceding PDF and the rapidity according to Eq. (14.13). The p_z component of the momentum and the energy are then obtained with Eqs. (14.18) and (14.15), respectively.

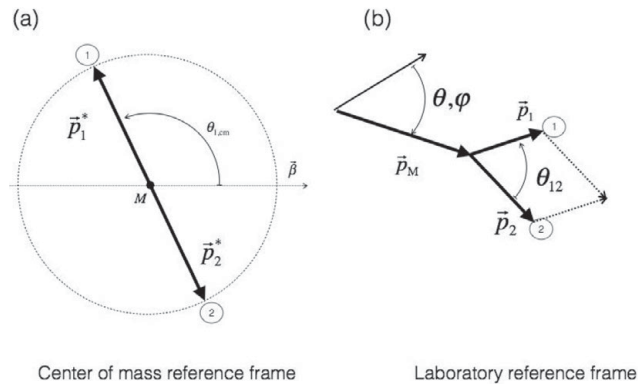


Fig. 14.1

Definition of kinematic variables required for the simulation of two-body decays. (a) Rest frame of the parent particle. (b) Laboratory frame.

14.1.2 Simulation of Resonance Decays

Corrections for detection efficiencies, acceptance, and other instrumental effects require that one can accurately model the decay of short-lived particles and take into account both kinematical and instrumental effects. We first consider the decay of narrow-width resonances, which can be represented by a fixed mass value. Decay simulations involving finite widths are considered next, as are examples of three-body decay that may be represented as a succession of two-body decays.

Fixed-Mass Two-Body Decays

The generation of two-body decays involves (1) the generation of the two daughter particles in the rest frame of the parent, (2) generation of the kinematic parameters of the parent in the laboratory reference frame, and (3) boost of the two daughter particles according to the speed and direction of the parent. The relevant variables are defined in Figure 14.1.

In the rest frame of the parent, the two daughter particles have momenta of equal magnitude, p^* , but in opposite directions. It is thus sufficient to generate the momentum vector of the first particle $\vec{p}_1^*$, and the momentum of the second particle is simply $\vec{p}_2^* = -\vec{p}_1^*$. The magnitude of the momentum is determined by the mass M of the parent, as well as the masses m_1 and m_2 of the daughter particles according to Eq. (8.165):

$$p^* = \frac{1}{2M} \{ [M^2 - (m_1 - m_2)^2] [M^2 - (m_1 + m_2)^2] \}^{1/2}. \quad (14.29)$$

The generation of the polar and azimuth angles of particle 1 are carried out according to

$$\phi_1^* = 2\pi r_1, \quad (14.30)$$

$$\cos \theta_1^* = 1 - 2r_2, \quad (14.31)$$

where r_1 and r_2 represent random numbers in the range $[0, 1]$ generated for each decay. The CM momentum $\vec{p}_1^*$ and energy E_1^* of particle 1 is thus

$$p_{1,x}^* = p^* \sin \theta_1^* \cos \phi_1^*, \quad (14.32)$$

$$p_{1,y}^* = p^* \sin \theta_1^* \sin \phi_1^*, \quad (14.33)$$

$$p_{1,z}^* = p^* \cos \theta_1^*, \quad (14.34)$$

$$E_1^* = \sqrt{(p^*)^2 + m_1^2}, \quad (14.35)$$

and for particle 2, one has

$$p_{2,x}^* = -p_{1,x}^*, \quad (14.36)$$

$$p_{2,y}^* = -p_{1,y}^*, \quad (14.37)$$

$$p_{2,z}^* = -p_{1,z}^*, \quad (14.38)$$

$$E_2^* = \sqrt{(p^*)^2 + m_2^2}. \quad (14.39)$$

Boosting the daughter particles in the laboratory frame requires knowledge of the direction and speed of the parent particle. In turn, generation of the momentum vector of the parent $\vec{p}_P$ requires a specific production scenario or model such as those discussed in prior sections. Let us here assume that the momentum of the parent $\vec{p}_P$ is known. Its velocity vector may then be calculated according to

$$\vec{\beta}_P = \frac{\vec{p}_P}{E_P}. \quad (14.40)$$

The daughter particles $i = 1, 2$ can thus be boosted in the laboratory frame (Problem 14.3) according to

$$\vec{p}_i^{\text{lab}} = \vec{p}_i^* + [(\gamma_P - 1)\vec{p}_i^* \cdot \hat{\beta}_P + \gamma_P E_i^*] \hat{\beta}_P, \quad (14.41)$$

where $\gamma = (1 - \beta_P^2)^{-1/2}$.

Finite-Width Resonance Two-Body Decays

While energy-momentum conservation constrains the energy and momenta of particles produced in elementary particle collisions, it does not uniquely specify the mass of short-lived particles, which can then nominally be produced event by event with values determined by a Breit–Wigner distribution (§3.15). Simulations of the decays of short-lived (i.e., finite width) resonances thus requires generation of a mass value M for the parent according to

$$f_{BW}(M|M_0, \Gamma) = \frac{1}{\pi} \frac{\Gamma/2}{\Gamma^2/4 + (M - M_0)^2}, \quad (14.42)$$

where M_0 and Γ are the nominal mass and width of the parent particle, respectively. Note that since the Breit–Wigner distribution has no a priori bounds, imposing an artificial upper-value cut to account for kinematical limitations encountered in finite beam energy experiments may be required. A lower cut must de facto be imposed to account for the

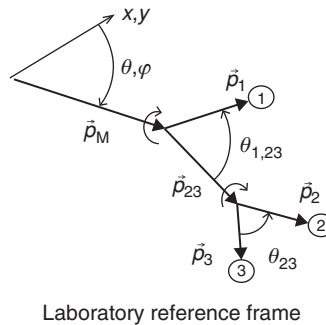


Fig. 14.2 Three-body decays as sequential two-body decays. In general, the two decays do not occur in the same plane.

finite masses of the particles and the need for $|\vec{p}^*| > 0$. With the mass in hand, the generation of daughter particles proceeds in the same manner as for a fixed mass covered in the previous paragraph.

Three-Body Decays Involving Sequential Two-Body Decays

A number of three-body decays may be described as a sequence of two-body decays, as illustrated in Figure 14.2.

Simulations of the decay of a parent particle in terms of a sequence of two-body decays proceeds quite similarly as the simpler case of a single two-body decay. Given the mass M of the parent particle, one generates the first decay into particles of mass m_1 and m_{23} according to the two-body decay algorithm. If the mass m_{23} is not fixed, one must first determine its value according to an appropriate PDF such as a Breit–Wigner distribution. Once the momentum $\vec{p}_{23}$ is determined, one can then proceed to decay the particle into particles 2 and 3 using the two-body decay algorithm.

14.1.3 Simple Event Generators

The generation of collision events with large numbers of particles of different species requires specific assumptions be made concerning the relative probabilities of emission of each type of particle and their momentum distributions. In this section, we first describe a rudimentary technique that neglects energy and momentum conservation, and in the next section, we show how one can modify the technique to achieve particle generation that conserves energy and momentum event by event.

For illustrative purposes, let us assume the number m of particles in an event can be modeled ab initio with a PDF $P_m(m)$ which we choose to be uniform in the range $[m_{\min}, m_{\max}]$, while the production of n different particle species of mass m_i , charge q_i , are specified by probabilities p_i , with $i = 1, \dots, n$ such that

$$\sum_{i=1}^n p_i = 1. \quad (14.43)$$

Let us further assume that all particle species can be characterized by a flat rapidity distribution in the range $[y_{\min}, y_{\max}]$ and Maxwell–Boltzmann distributions in the transverse plane, with temperatures T_i .

Before one begins with the generation of particles, it is convenient to define the cumulative sum of probabilities P_i , for $i = 0, \dots, n$, according to

$$P_0 = 0 \quad \text{for } i = 0, \quad (14.44)$$

$$P_i = \sum_{k=1}^i p_k \quad \text{for } 1 \leq i \leq n, \quad (14.45)$$

with $P_1 = p_1$, $P_2 = p_1 + p_2$, and so on, up to $P_n = 1$ by virtue of Eq. (14.43).

The generation of events can then be accomplished according to the following algorithm:

1. For each event, . . .
 - a. Determine the multiplicity (number of particles) of the event on the basis of uniform PDF $P_m(m)$:

$$m = \text{int} [m_{\min} + r_1 (m_{\max} - m_{\min})], \quad (14.46)$$

where the notation $\text{int} [O]$ indicates one needs to round the number O to the nearest integer.

- b. For each particle,
 - i. Generate a random number r_2 in the range $[0, 1]$ to determine the species (mass m_i , charge q_i , and emission temperature T_i) of the particle. The species of the particle is specified by the integer i that satisfies $P_{i-1} \leq r_2 < P_i$.
 - ii. Generate the momentum vector of the particle assuming a mass m_i and temperature T_i with a flat rapidity distribution and transverse Maxwell–Boltzmann distribution based on the techniques presented in earlier sections.

Note that if some of the produced particles are short-lived resonances, one can apply the techniques presented in §14.1.2 to simulate their decay and the generation of daughter particles.

The aforementioned event-generating technique is quite simple but somewhat ad hoc. It provides a simple method for fast generation of particles and simple modeling of the production of particles in elementary collisions, but given it is based on simplistic and purely phenomenological models, it should not be expected to provide a very accurate account of particle production cross sections and correlations. Although it can be readily modified to use a more realistic multiplicity distribution or particle momentum distributions, it features an intrinsic flaw that cannot be fixed by substitutions of more realistic PDFs: the events produced do not have a specific total energy or net momentum and, as such, do not satisfy laws of energy–momentum conservation. This may be appropriate as long as the particles generated are meant to represent only a fraction of all observable particles produced by actual collisions. But if the generated particles are meant to be representative of whole events, one must use a slightly modified technique to ensure that the total momentum and the total energy of the particles are produced with predefined values; in other words, the

particle generation must obey laws of energy–momentum conservation. It is in principle also necessary to account for the conservation of other conserved quantities (quantum numbers) such as electric charge, strangeness, baryon number, and so on. A particle generation technique that accounts for global energy and momentum conservation is presented in the next section. Better modeling of collisions still can be achieved with computer codes based on comprehensive theoretical frameworks of particle interaction and particle production, such as those already cited [29, 52, 61, 66, 133, 137, 169, 174, 175, 176, 188, 191, 192, 198].

14.1.4 Multiparticle Generation with Energy–Momentum Conservation

We saw in §8.2.2 that the cross section of a process producing n particles is determined by the square of the transition amplitude $|\mathfrak{M}|$ and an n -body phase factor dR_n (8.79). It is thus convenient to define the n -body phase integral R_n as follows:

$$R_n = \int_{4n} \delta^{(4)} \left(P - \sum_{j=1}^n p_j \right) \prod_{i=1}^n \delta(p_i^2 - m_i^2) d^4 p_i, \quad (14.47)$$

where P is the total 4-momentum of the n -body system, while p_i and m_i are the 4-momenta and masses of the produced particles, respectively. The differential cross section in terms of a certain kinematical parameter α (particle momentum, emission angle, etc.) may then be written

$$\frac{d\sigma}{d\alpha} = \frac{d}{d\alpha} (|\mathfrak{M}|^2 R_n). \quad (14.48)$$

One can equally sample the entire n -particle phase space by assuming that $|\mathfrak{M}| = 1$. This shall produce differential cross sections (spectra) that are determined exclusively by the phase-space of the n outgoing particles, that is, cross sections corresponding to a uniform n -body phase space. One notes that by virtue of the factor $\delta^{(4)}(P - \sum_{j=1}^n p_j)$, generation according to R_n shall automatically conserve both energy and momentum. Additionally, given that

$$\delta(p_i^2 - m_i^2) d^4 p_i = \frac{p_i^2}{E_i} dp_i d\cos\theta_i d\phi_i, \quad (14.49)$$

it should also produce particles in proportion to the density of states: events with a higher density of states should be more probable. It is also worth mentioning that R_n leads to an intrinsic n -body particle correlation determined solely by energy–momentum conservation. This type of correlation is to be distinguished from those implied by the production process embodied in the amplitude $|\mathfrak{M}|$. Separating these two sources of correlation experimentally is unfortunately a rather nontrivial task. Finally, consider that if a flat spectrum in n -body phase-space is not appropriate, one can always assign a weight to generated events based on the desired transition amplitude $|\mathfrak{M}|$ after an n -body event has been produced according to R_n .

The notion of using a Monte Carlo method to calculate the R_n integral was first discussed by Kopylov [130], while Srivastava and Sudarshan [177] derived the covariant form,

Eq. (14.47), which lends itself to a recursive calculation of the integral. We here follow the M -generator algorithm presented by James [113].

The integral Eq. (14.47) may be written

$$R_n(P; m_1, \dots, m_n) = \int_{4n} \left[\delta^{(4)} \left(P - p_n - \sum_{j=1}^{n-1} p_j \right) \prod_{i=1}^n \delta(p_i^2 - m_i^2) d^4 p_i \right] \times \delta(p_n^2 - m_n^2) d^4 p_n, \quad (14.50)$$

where the expression in square brackets is $R_{n-1}(P - p_n; m_1, \dots, m_{n-1})$. One can thus indeed calculate $R_n(P; m_1, \dots, m_n)$ recursively

$$R_n(P; m_1, \dots, m_n) = \int_{4n} R_{n-1}(P - p_n; m_1, \dots, m_{n-1}) \delta(p_n^2 - m_n^2) d^4 p_n. \quad (14.51)$$

Making use of Eq. (14.49), Eq. (14.51) may be written

$$R_n(P; m_1, \dots, m_n) = \int_{4n} R_{n-1}(P - p_n; m_1, \dots, m_{n-1}) \frac{d^3 p_n}{2E_n}, \quad (14.52)$$

which in principle provides a basis for the calculation of R_n since it is expressed in terms of a Lorentz invariant and can thus be calculated recursively in arbitrary frames of reference. The problem resides in the choice of efficient bounds of the momentum p_n . It turns out to be more efficient and practical to modify the preceding recursion formula to obtain an expression in terms of particle masses. This is accomplished by noting that

$$\delta^{(4)} \left(P - \sum_{j=1}^n p_j \right) = \int \delta^{(4)} \left(P - P_l - \sum_{j=l+1}^n p_j \right) \delta^{(4)} \left(P_l - \sum_{k=1}^l p_k \right) d^4 P_l, \quad (14.53)$$

which enables us to write

$$R_n(P; m_1, \dots, m_n) = \int \delta^{(4)} \left(P - P_l - \sum_{j=l+1}^n p_j \right) \prod_{j=l+1}^n \delta(p_j^2 - m_j^2) d^4 p_j \times \int \delta^{(4)} \left(P_l - \sum_{k=1}^l p_k \right) \prod_{j=1}^l \delta(p_j^2 - m_j^2) d^4 p_j d^4 P_l. \quad (14.54)$$

Noting that

$$1 = \int_0^\infty \delta(P_l^2 - M_l^2) dM_l^2, \quad (14.55)$$

one can then express R_n as

$$\begin{aligned}
 R_n(P; m_1, \dots, m_n) &= \int_0^\infty \left[\int \delta^{(4)} \left(P - P_l - \sum_{j=l+1}^n p_j \right) \right. \\
 &\quad \times \prod_{j=l+1}^n \delta(p_j^2 - m_j^2) \delta(P_l^2 - M_l^2) d^4 p_j d^4 P_l \\
 &\quad \left. \times \int \delta^{(4)} \left(P_l - \sum_{k=1}^l p_k \right) \prod_{j=1}^l \delta(p_j^2 - m_j^2) d^4 p_j \right] dM_l^2,
 \end{aligned} \quad (14.56)$$

which gives us

$$\begin{aligned}
 R_n(P; m_1, \dots, m_n) &= \int_0^\infty R_{n-l+1}(P; M_l, m_{l+1}, \dots, m_n) \\
 &\quad \times R_l(P_l; m_1, \dots, m_l) dM_l^2.
 \end{aligned} \quad (14.57)$$

One can show that repeated applications of this “splitting” relation, starting with $l = 2$, yields the recurrence relation

$$R_n = \int dM_{n-1}^2 \dots \int dM_2^2 \prod_{i=1}^{n-1} R_2(M_{i+1}; m_i, m_{i+1}), \quad (14.58)$$

where

$$R_2(M_{i+1}; M_i, m_{i+1}) = \frac{2\pi}{M_{i+1}} \sqrt{M_{i+1}^2 + \left(\frac{M_i^2 - m_{i+1}^2}{M_{i+1}} \right)^2 - 2(M_{i+1}^2 + m_{i+1}^2)}. \quad (14.59)$$

Transforming the integrals dM^2 into $2M dM$, one obtains

$$R_2 = \frac{1}{2m_1} \int \int \prod_{i=1}^{n-1} 2M_i R_2(M_{i+1}; M_i, m_{i+1}) dM_{n-1} \dots dM_2, \quad (14.60)$$

which is pictorially represented in Figure 8.12.

One can then proceed to the generation of n -body processes as if they were a succession of two-body decays. One only needs to apply a two-body phase-space factor for each “decay vertex.” The boundaries of integration are a tricky issue. Nominally, one might be tempted to write

$$M_{j-1} + m_j < M_j < M_{j+1} - m_{j+1}. \quad (14.61)$$

But this implies the boundaries of j th integral depend on other integrals, which leads to incorrect sampling of the phase space in a Monte Carlo integration. The masses can, however, be chosen according to the less restrictive condition

$$\sum_{i=1}^j m_i < M_j < M_n - \sum_{i=j+1}^n m_i, \quad (14.62)$$

using the generating technique

$$M_j = r_j \left(M_n - \sum_{i=j+1}^n m_i \right) + \sum_{i=1}^j m_i, \quad (14.63)$$

where r_j are random numbers in the range $[0, 1]$. One verifies that the M_j generated with the preceding expression will satisfy the boundaries Eq. (14.61) provided

$$0 < r_1 < \dots < r_j < \dots < r_{n-2} < 1. \quad (14.64)$$

For the generation of an event with n particles, it thus suffices to generate $n - 2$ random numbers and sort them in ascending order for the calculation of the masses using Eq. (14.63). However, one more step is required at each decay vertex. Indeed the aforementioned condition specifies the mass M_i but it does not dictate the direction of the produced pair. One must then randomly choose, for each decay, a direction for the mass M_j .

$$\phi_j = 2\pi r_j', \quad (14.65)$$

$$d \cos \theta_j = -1 + 2 \times r_j'', \quad (14.66)$$

where r_j' and r_j'' are random numbers in the range $[0, 1]$. It is important to note that isotropic emission is only expected in the rest frame of the mass M_j . To obtain the required Lorentz invariance, one must generate two-body decays successively in the rest frame of each mass M_j . This means one must successively Lorentz-transform each momentum into the group of particles preceding it.

This algorithm was first made available as a program named GenBod in CERN software libraries but is now available within the ROOT framework [59] as class TGenPhaseSpace.

14.1.5 Correlated Particles Generators

A variety of techniques may be used to generate correlated particles. At the outset, note that the production of daughter particles resulting from a two- or three-body decay and boosted in the laboratory frame according to the speed and direction of their parent particle produce correlated particles: for a high-velocity parent particle, the daughter particles tend to be separated by a small angle that decreases for increasing velocity (momentum) of the parent. Similarly, the production of a finite number m of particles with the multiparticle generation technique presented in §14.1.4 also results in net or global correlations between all particles produced [54, 55]. Correlations shall also result from conservation laws such as (electric) charge, strangeness, baryon number conservation, and so on.

Kinematically correlated particles may be generated if they can be produced by means of hierarchical processes, such as sequential two-body decays or successive particle generation accounting individually for energy and momentum conservation. Correlations may also be achieved, for instance, by shifting (§14.1.5) or boosting the momentum of groups of correlated particles [162]. In general, it may be cumbersome or technically difficult to impart elaborated correlations between generated particles. Fortunately, it is always possible to generate particles independently and assign events or particle n -tuplets a weight according to a correlation function ansatz (§14.1.5).

Anisotropic Particle Generation

Anisotropic flow, relative to an event plane, may be represented according to a Fourier decomposition

$$\frac{1}{N} \frac{dN}{d\phi} \propto 1 + 2 \sum_{n=1}^{n_{\max}} v_n \cos(n(\phi - \Psi_n)) \quad (14.67)$$

where Ψ_n represent the orientation of the event plane of order n (see §11.1.1).

Generation of random angles according to the preceding expression with the inversion method is not possible because its integral cannot be inverted, while the use of the acceptance/rejection method is somewhat inefficient. On the other hand, the integrated histogram method is simple and reasonably efficient. It suffices to produce a finely binned histogram of the distribution Eq. (14.67) and apply the histogram method presented in §13.3.5.

In some situations, one may wish to introduce flow artificially after the fact, that is, after particles have been generated by some third-party event generator. Use of the acceptance/rejection method would be a bad choice in this case because it would change the integrated particle production cross section (i.e., the integral of the momentum distribution). While the aforementioned Fourier decomposition cannot be achieved from *scratch*, a reasonable approximation may be obtained by shifting particles in the transverse plane according to

$$\begin{aligned} p_x' &= p_x \times \left(1 + 2 \sum_{n=1}^{n_{\max}} v_n \cos(n\Psi_n) \right), \\ p_y' &= p_y \times \left(1 + 2 \sum_{n=1}^{n_{\max}} v_n \sin(n\Psi_n) \right), \end{aligned} \quad (14.68)$$

where the event plane angles Ψ_n are chosen randomly *event by event* in the range $[0, 2\pi]$. The flow coefficients v_n can be arbitrary constants or even functions of the transverse momentum of the particles. The above “shift” of the pair (p_x, p_y) does not change the integral of the momentum distribution, provided that the angles Ψ_n are generated as uniform deviates in the range $[0, 2\pi]$, since integrals of the sine and cosine functions in this range vanish. The shift also conserves momentum in the transverse plane. Indeed, if the original events were generated with momentum conservation, that is, such that

$$\sum_{i=1}^m p_{x,i} = 0, \quad (14.69)$$

$$\sum_{i=1}^m p_{y,i} = 0, \quad (14.70)$$

where m is the number of particles in the event, then a random shift of all particles of an event according to a specific set of angles Ψ_n and fixed v_n coefficients (i.e., common to all

particles of an event) yields

$$\sum_{i=1}^m \vec{p}_{x,i}' = \left(\sum_{i=1}^m p_{x,i} \right) \times \left(1 + 2 \sum_{n=1}^{n_{\max}} v_n \cos(n\Psi_n) \right) = 0, \quad (14.71)$$

$$\sum_{i=1}^m \vec{p}_{y,i}' = \left(\sum_{i=1}^m p_{y,i} \right) \times \left(1 + 2 \sum_{n=1}^{n_{\max}} v_n \sin(n\Psi_n) \right) = 0, \quad (14.72)$$

and thus conserves momentum in the transverse plane. Note, however, that this simple factorization breaks down if the flow coefficients are functions of the momentum of the particles. The introduction of flow after the fact, with this technique, thus does not strictly conserve momentum if the coefficients v_n are functions of p_T and/or rapidity. It may prove adequate, nonetheless, for simulations of the performance of large detectors that include particle losses and resolution smearing.

Simulation of Correlations with Weights

Simulations of correlated particles production can be achieved by applying an ad hoc weight to particle pairs, n -tuplets, or events after their production based on some function of the particle momenta and energies. This enables the simulation of correlated particles based on the generation of particles using simple algorithms that nominally produce independent and uncorrelated particles.

Let us consider, as an example, the generation of particles with a peaked correlation in relative azimuth and rapidity. One can generate events with uncorrelated particles as in §14.1.3. Correlation may, however, be simulated with the use of weights. For instance, in order to simulate a two-particle correlations of the form

$$w(\Delta\eta, \Delta\phi) = 1 + A \frac{1}{2\pi\sigma_\eta\sigma_\phi} \exp\left(-\frac{\Delta\eta^2}{2\sigma_\eta^2}\right) \exp\left(-\frac{\Delta\phi^2}{2\sigma_\phi^2}\right), \quad (14.73)$$

it suffices to generate independent particles with, say, flat rapidity distributions and assign each pair a weight $w(\Delta\eta, \Delta\phi)$. To simulate a two-particle cumulant, one may generate two sequences of events with identical multiplicity distributions. Pairs of the first sequence are given the weight $w(\Delta\eta, \Delta\phi)$ to simulate “real” events, $\rho_2(\Delta\eta, \Delta\phi)$, while pair of the second sequence are given a unit weight and simulate “mixed” events, $\rho_1 \otimes \rho_1(\Delta\eta, \Delta\phi)$. One next calculates the ratio, bin by bin, of the $\rho_2(\Delta\eta, \Delta\phi)$ and $\rho_1 \otimes \rho_1(\Delta\eta, \Delta\phi)$ histograms, and subtract one, to obtain a normalized cumulant, as illustrated in Figure 14.3.

Similarly, Hanbury-Brown Twiss (HBT) type correlations can be simulated with a weight of the form

$$w(\vec{p}_1, \vec{p}_2) \propto \exp\left(-\frac{q^2}{2\sigma_{\text{HBT}}^2}\right), \quad (14.74)$$

with $q^2 = (\vec{p}_1 - \vec{p}_2)^2$ for pairs of identical particles (e.g., π^+ or π^-) with momenta $\vec{p}_1$ and $\vec{p}_2$. The width σ_{HBT} of the correlation functions is known to be inversely proportional to the size of the emitting source size r .

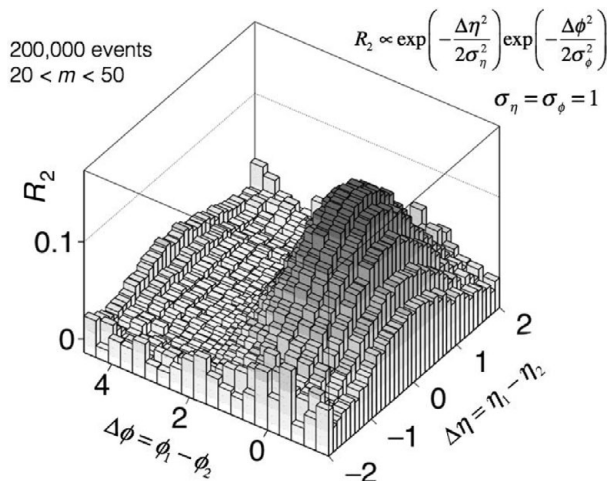


Fig. 14.3

Simulation of a two-particle correlation function using the weight method. Particle pairs were given the weight shown for real events and unit weight for mixed events. The simulation included 200,000 events with a multiplicity of particles in the range $20 < m \leq 50$. The cumulant is calculated as $\rho_2(\Delta\eta, \Delta\phi) / \rho_1 \otimes \rho_1(\Delta\eta, \Delta\phi) - 1$.

Quite obviously, arbitrarily complex and more sophisticated correlation functions can be modeled using this weight technique.

14.2 Detector Simulation

Detector simulations are commonly used to (1) understand the performance of a detection system, (2) determine the acceptance, efficiency, and resolutions of specific measurements, and (3) obtain response matrices that can be used toward the correction of simulated data. Such tasks can be best accomplished with detailed and comprehensive studies of the detector performance carried out within detector simulation environments such as that provided by the computer code GEANT, introduced in §14.2.4. But since the use of GEANT requires quite a bit of setup and computer coding, and considering that GEANT simulations can be rather CPU-intensive, performance studies are often conducted with **fast simulators** involving simple and sometimes rather primitive models of a detector's response.

We introduce basic exemplary techniques for the simulation of the effects of finite efficiency, detector resolution, and acceptance determination in §14.2.1, and provide a simple example of a calculation of a response matrix for jet measurements in §14.2.3.

14.2.1 Efficiency, Resolution, and Acceptance Simulators

Efficiency Simulation

The efficiency of a detection system towards measurement of a specific type of particle (or range of particle species) may be obtained by means of detailed simulations based on a

realistic particle production model and a detailed simulation of the detector performance. Alternatively, one can also embed simulated tracks (or other relevant objects) into actual events and determine what fraction of the embedded tracks are actually recovered by the track reconstruction software.

Let us here assume that the detection efficiency for a specific particle species is known and can be parameterized with a model $\epsilon(\eta, \phi, p_T)$. In some cases, it might be possible to factorize the efficiency into a product of three functions $\epsilon(\eta, \phi, p_T) = \epsilon(\eta)\epsilon(\phi)\epsilon(p_T)$. It may even be possible to assume that dependencies on rapidity and azimuth are negligible and treat $\epsilon(\eta)$ and $\epsilon(\phi)$ as constants. One can then use this model to carry out fast simulations of the effects of the limited detection efficiency on momentum spectra, invariance mass spectra, correlation functions, or more complex objects such as jets.

While the effect of detection can in principle be trivially determined by multiplying the efficiency by the production cross section, it is often more convenient to carry out a simulation involving the production model considered and a model $\epsilon(\eta, \phi, p_T)$ of the efficiency. This has the added benefit of providing a track-by-track account of detector resolution effects. In such simulations, it can often be assumed that the detection efficiency for two or more particles factorizes and that one can apply the function $\epsilon(\eta, \phi, p_T)$ for each particle independently. A simulation may be based on a simple home-brewed particle generator or more sophisticated and theoretically motivated event generators whose events are either generated on the fly (event-by-event) or read from stored files. A typical simulation thus proceeds as follows:

1. Generate or read an event from file.
2. Efficiency/smearing: For each particle of an event
 - a. Decide whether the particle should be accepted.
 - b. Smear kinematical parameters of the particle (if needed/desired).
 - c. Store smeared parameters.
3. Analysis: Carry out the required analysis of the generated event based on accepted (and smeared) particles. Optionally carry out the analysis on all generated and unsmeared particles to obtain a “perfect detection” reference.

The decision whether to accept a particle is based on three steps: (1) given the (unsmeared) kinematical parameters of the particle (η, ϕ, p_T) , calculate the efficiency of detection $\epsilon(\eta, \phi, p_T)$; (2) generate a random number r with a uniform distribution in the range $[0, 1]$; and (3) accept the particle if $r \leq \epsilon(\eta, \phi, p_T)$.

Figure 14.4 presents an example of the application of an efficiency function $\epsilon(p_T)$ on a parent distribution $f(p_T) = \lambda^{-1} \exp(-p_T/\lambda)$. It shows, in particular, that the measured transverse momentum mean $\langle p_T \rangle$ can be significantly altered by the detector's efficiency dependence on transverse momentum, $\epsilon(p_T)$.

Resolution Smearing

Smearing of kinematical parameters is based on models of the detector response. For instance, to simulate instrumental effects (smearing) on a transverse momentum measurement, one requires a PDF $f(p_T|\vec{\theta})$ describing fluctuations of the measured p_T determined

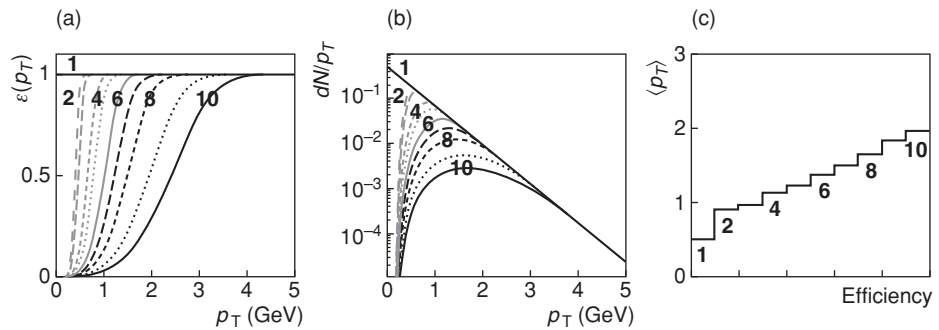


Fig. 14.4

Illustration of the effect of efficiency function $\epsilon(p_T)$ on a p_T spectrum and measured mean $\langle p_T \rangle$. (a) Selected efficiency response curves. (b) Product of the efficiency curves by the input spectrum ($f(p_T) = \lambda^{-1} \exp(-p_T/\lambda)$, with $\lambda = 0.5$). (c) Average p_T obtained with the selected efficiency curves.

by some set of parameters $\vec{\theta}$. The parameters $\vec{\theta}$ are here assumed to be known a priori. In practice, they could be obtained from either detailed simulations of the detector performance (§14.2.4) or an explicit measurement. Often times, but not always, the smearing response of the detection system may be assumed to be (approximately) Gaussian. Specification of the smearing function $f(p_T|\vec{\theta})$ thus reduces to the expression of the mean and root mean square (rms) of a Gaussian as a function of the true p_T of the particles. If the mean is null, the measurement can be considered without bias, and only fluctuations need considering. However, note that given particles lose energy (and thus momentum) as they traverse a detector, the track reconstruction software must compensate for such losses (see §9.2). The problem arises that at a given momentum, energy losses depend on the mass (or particle identification, PID) of the particle, which is unknown a priori. Corrections are thus typically made assuming the PID of the most abundantly produced particle species (e.g., pions). This invariably leads to a bias (i.e., a systematic shift) in the reconstruction of the momentum of particles. Such a shift should ideally be accounted for in simulations of the performance of detection system. Thus for each simulated species production, not only should the rms be provided as a function of the transverse momentum of the particles but also a function estimating the momentum bias achieved in a typical event reconstruction. Additionally, note that the reconstruction of charged particle tracks yield Gaussian fluctuations in the track curvature (C), which is inversely proportional to the p_T of the particle. Fluctuations in p_T are thus generally non-Gaussian. However, for simplicity's sake, we here illustrate a smearing simulation procedure assuming Gaussian fluctuations.

We assume the fluctuations in momentum measurements are determined by a Gaussian PDF with mean $\mu \equiv \mu(p_{T,0})$ and rms $\sigma \equiv \sigma(p_{T,0})$ where $p_{T,0}$ is the true momentum of the particles. The bias $\mu \equiv \mu(p_{T,0})$ is typically smallest for minimum ionizing particles and grows approximately in inverse relation of the momentum at small momenta and proportionally to the momentum at higher momenta. It could thus be modeled according to

$$\mu(p_{T,0}) = a_{-1}p_{T,0}^{-1} + a_0 + a_1p_{T,0}, \quad (14.75)$$

where the coefficients a_n must be suitably fitted to represent the actual measurement bias of each particle species. Energy loss fluctuations typically scale with the average energy loss. Since energy losses have a $1/\beta^2$ dependence at low momentum and a logarithmic rise at higher momenta, one may model the momentum dependency of the fluctuations accordingly. However, note that the p_T resolution becomes exceedingly poor for very small curvature because the finite granularity of the hit detectors intrinsically limits the position resolution. One can thus in general model the rms with a low-order power series, such as

$$\sigma(p_{T,0}) = b_{-2}p_{T,0}^{-2} + b_{-1}p_{T,0}^{-1} + b_0 + b_1p_{T,0} + b_2p_{T,0}^2, \quad (14.76)$$

where the coefficients b_n must also be suitably fitted to represent the actual fluctuations for each measured particle species.

Assuming the coefficients a_n and b_n are known, either from a detailed simulation or actual measurements, one can thus carry out (fast) simulations of the detector response (smearing) according to the algorithm presented earlier in this section. Smearing requires calculation track by track of both $\mu(p_{T,0})$ and $\sigma(p_{T,0})$ with Eqs. (14.75, 14.76). Smeared track momenta are then obtained based on

$$p_T = p_{T,0} + r_G[\mu(p_{T,0}), \sigma(p_{T,0})], \quad (14.77)$$

where $r_G[\mu(p_{T,0}), \sigma(p_{T,0})]$ represents Gaussian deviates with mean $\mu(p_{T,0})$ and rms $\sigma(p_{T,0})$ calculable according to the algorithm presented in §13.3.7.

A fast simulator including smearing may, for instance, be used to study the effects of smearing on momentum spectra (Figure 14.5), the reconstruction of short-lived decaying particles based on the invariant mass technique, or jet measurements.

14.2.2 Kinematic Acceptance of Two-Body Decays

Monte Carlo simulations based on a simple particle generator provide a quick and easy method to determine the acceptance of a detector toward measurements of momentum spectra, particularly those of short-lived resonances. Let us consider, as an example, simulations of the decay of kaons K_s^0 into a pair $\pi^+ + \pi^-$ detected by invariant mass reconstruction (§8.5.1).

We will use the algorithm presented in sections §§14.1.1, 14.1.3, and 14.2.1 to generate K_s^0 with a Maxwell–Boltzmann distribution in transverse momentum and a uniform pseudorapidity in the range $|\eta| < 3$. Kaons are decayed into $\pi^+ + \pi^-$ pairs exclusively. Pions are assumed detectable with a 100% efficiency if in the ranges $0.2 < p_T \leq 1.5$ and $|\eta| < 2$. However, their momenta are smeared according to Eq. (14.76) using resolution parameters of curve 2 in Figure 14.5.

The simulated invariant mass spectra and K_s^0 detection efficiency obtained with these detection conditions are shown in Figure 14.6. One finds that the pion acceptance dramatically shapes the K_s^0 detection efficiency and acceptance.

More generally, one can use k_B functions $B_i^{\min/\max}(\eta, \phi, p_T)$, $i = 1, \dots, k_B$ to define the boundaries within which a specific particle species is considered detectable. For instance,

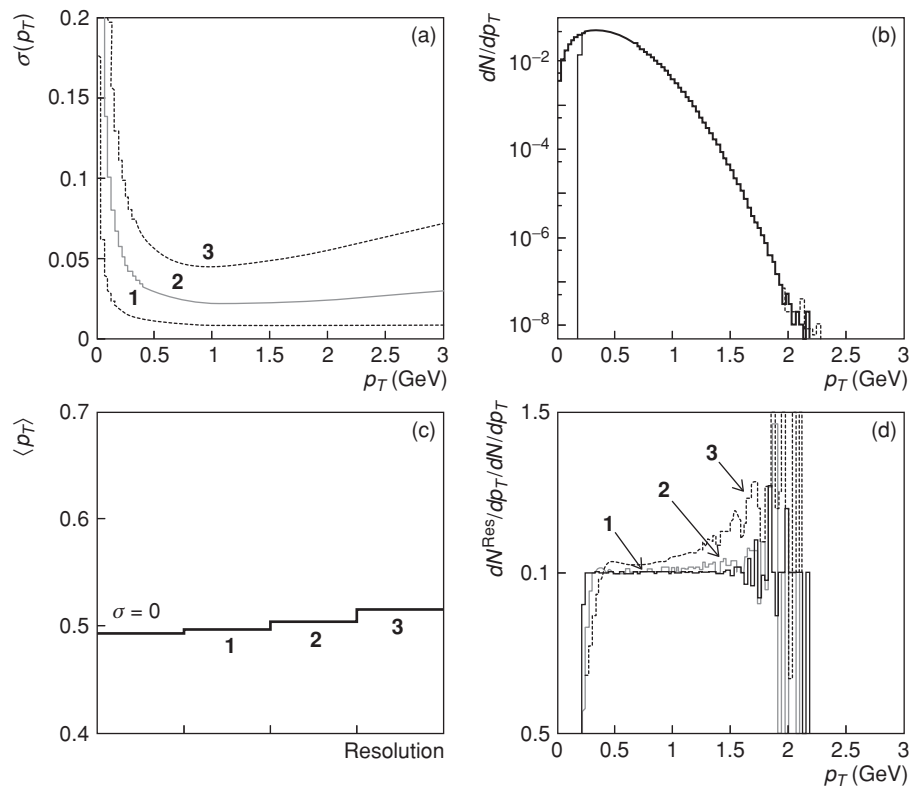


Fig. 14.5

Illustration of the effect of resolution smearing on p_T spectra. (a) Resolution responses used in the simulations. (b) Simulation of p_T spectrum using a Maxwell–Boltzmann distribution and smearing parameters shown in (a). (c) Average p_T obtained with perfect and finite resolution. (d) Ratio of the smeared spectra to the original spectrum. Note that a p_T threshold of 0.2 GeV has been used to illustrate the joint effects of finite resolution and acceptance.

as illustrated in Figure 14.7, a set of functions $p_{T,i}^{\min/\max}(\eta)$ can express the minimum and maximum p_T values detectable by an apparatus. A particle with momentum vector $\vec{p} = (\eta, \phi, p_T)$ would be considered detectable if and only if it satisfies the conditions

$$p_{T,i}^{\min}(\eta) \leq p_T \leq p_{T,i}^{\max}(\eta), \quad \text{for all } i = 1, \dots, k \quad (14.78)$$

14.2.3 Response Matrix Determination

We saw in §12.3 that the unfolding of measured distributions requires knowledge of a response matrix describing the efficiency and smearing imparted on a measured signal by instrumental effects. The determination of a response matrix typically requires simulation of the effects of the instrumentation of the measured particle kinematical parameters, and their impact on other quantities such as detected particle multiplicities, resonance decays, or measurements of jets and their properties. Such studies are best conducted based on detailed simulations of the detector performance (§14.2.4) but can often also be accomplished

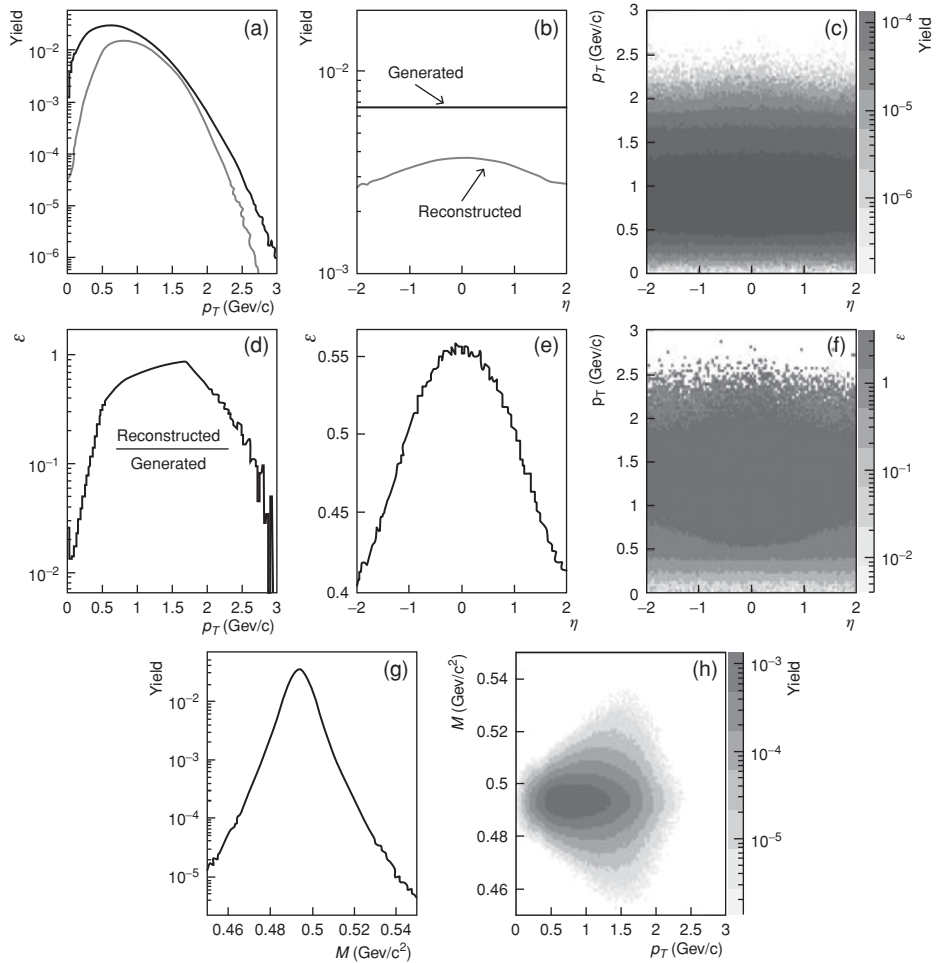


Fig. 14.6

Simulation of the decay and reconstruction of $K_S^0 \rightarrow \pi^+ + \pi^-$ with the invariant mass technique. (a) Generated (solid line) and reconstructed (dashed line) Maxwell-Boltzmann transverse momentum (p_T) spectra. (b) Generated and reconstructed pseudorapidity distributions. (c) Reconstructed distribution in p_T vs. η . (d) Acceptance and efficiency vs. p_T . (e) Efficiency vs. η ; efficiency vs. p_T and η ; (g) K_S^0 reconstructed mass spectrum. (h) Mass vs. reconstructed p_T . Pions are assumed detectable with a 100% efficiency if in the ranges $0.2 < p_T \leq 1.5$ and $|\eta| < 2$. Pion momentum resolution as in curve 2 of Figure 14.5.

using fast simulators such as those already discussed in previous sections. We here illustrate the determination of a response matrix toward corrections of measured charged particle multiplicity spectra.

For the sake of simplicity, we assume the particle multiplicity amounts to an average value $\langle m \rangle$, which we subtract ab initio from the simulated produced and measured distributions. The apparatus is assumed to have an efficiency such that measured multiplicities are, on average, 2.5 units lower than the actual values, and smeared with a resolution of 0.2

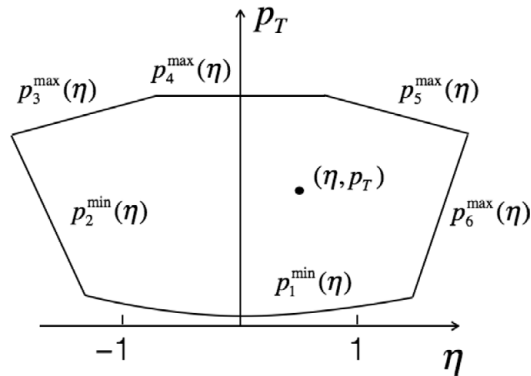


Fig. 14.7 Acceptance definition with k_B boundary functions $p_{T,i}^{\min/\max}(\eta, \phi), i = 1, \dots, 6$.

units. The response is simulated with a Gaussian distribution

$$P(m_m|m_t) = \frac{1}{\sqrt{2\pi}\sigma_m} \exp\left(-\frac{[m_m - m_t - \mu_m]^2}{2\sigma_m^2}\right), \quad (14.79)$$

with $\mu_m = 2.5$, $\sigma_m = 0.2$, and where m_t and m_m represent the *true* and *measured* multiplicities, respectively. One million events were generated to obtain the response matrix displayed in Figure 14.8, which was used in §§12.3.7 and 12.3.8 to present examples of unfolding with the SVD and Bayesian methods, respectively.

More generally, one might wish to account for specificities of the apparatus resolution of particle momenta as well as a more detailed description of its acceptance. The simulation could then proceed as follows:

1. Create a response matrix histogram to store event values of produced and measured multiplicities noted N and n , respectively.

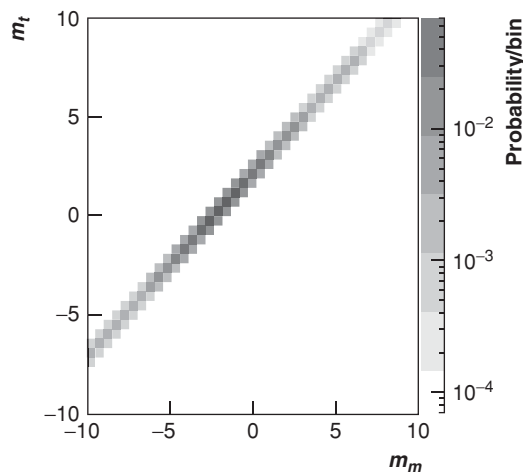


Fig. 14.8 Response matrix used for examples of SVD and Bayesian discussed in §§12.3.7 and 12.3.8.

2. Select a large number of events N_{ev} to be produced.
3. Define or select the function $\epsilon(\eta, \phi, p_T)$ to be used in the simulation of detection efficiency.
4. Define or select the conditions $B_i(\eta, \phi, p_T)$, $i = 1, k$ to be used in the simulation of the experimental acceptance.
5. For each event,
 - a. Determine the total multiplicity M of the event randomly according to an appropriate distribution in the multiplicity range or interest (e.g., $10 \leq M < 1,000$).
 - b. Set particle counters N and n to zero,
 - c. Generate an event with M particles according to appropriate rapidity and transverse momentum distributions (e.g., flat rapidity and Maxwell–Boltzmann p_T spectrum).
 - i. Generate particles according to chosen production model,
 - ii. For each generated particle,
 - (a) Determine if the particle is within the rapidity, azimuthal angle, and p_T boundaries of the experiment.
 - (b) If it is, increment true detected multiplicity N by one unit.
 - (c) Use the efficiency function $\epsilon(\eta, \phi, p_T)$ to determine if the particle is detected.
 - (d) If it is, smear the momentum and rapidity of the particle and determine whether the smeared momentum and rapidity fall within the kinematical boundaries of the experiment.
 - (e) If it does, increment the measured multiplicity n by one unit.
 - d. Increment the response function histogram according to the event's (N, n) .
6. Normalize the response function histogram

14.2.4 Detailed Detector Simulations Using GEANT

The de facto standard in particle and nuclear physics for comprehensive simulations of detector performance is the program GEANT [14]. Specialized computer codes such as EGS 5 (Electron Gamma Shower)[105] and FLUKA [32] are also available to simulate the development of electromagnetic and hadronic showers within materials.

GEANT provides a comprehensive computing environment for simulating the passage of particles through matter. It includes components for modeling detector geometry and material properties, the propagation of particles through electric or magnetic fields, as well as the modeling of particle energy loss and deposition in detector materials. GEANT can thus be used to simulate the detection of arbitrarily complex collisions and study detector performance attributes such as detection acceptance, efficiency, momentum resolution, and much more. GEANT, now in its fourth version, provides tools to simulate with high precision a vast array of particle interactions within detector materials and sensors, including electromagnetic, hadronic and optical processes, as well as the decay of long-lived particles. This is accomplished through modeling of electromagnetic and hadronic processes over a wide energy range starting, in some cases, from 250 eV and extending in others to the TeV energy range. Written in C++, GEANT was designed and constructed to expose the physics models utilized, handle complex geometries, and enable its easy adaptation for

optimal use in different applications. GEANT is the result of a worldwide collaboration of physicists and software engineers. It has a wide community of users and finds applications in particle physics, nuclear physics, accelerator design, space engineering, and medical physics. A detailed discussion of the Monte Carlo techniques and methods used in GEANT is well beyond the scope of this textbook but can be found in various publications (see ref. [14] and references therein).

While GEANT enables accurate and detailed simulations of detector performance, its application for certain tasks such as the description of hadronic showers in calorimeter can be rather fastidious and slow. Various groups have thus designed **fast simulators** to achieve the same goals as GEANT but requiring only a small fraction of its computing time, albeit with perhaps slightly less accuracy [17, 98].

Exercises

- 14.1 Derive the expression (14.8) of the moments of the power-law distribution.
- 14.2 Verify the expression (14.10) for the generation of random numbers according to a power law of the form (14.7).
- 14.3 Derive the expression (14.41) for the boost of a vector $\vec{p}^*$ in an arbitrary direction and velocity $\vec{\beta}_p$. Hint: Decompose the vector $\vec{p}^*$ in terms of components parallel and perpendicular to the boost direction and use the boost formula, Eq. (8.17), to carry the boost in the parallel direction.
- 14.4 Write a Monte Carlo program to calculate the acceptance and detection efficiency of K_s^0 decaying into $\pi^+ + \pi^-$. Assume the K_s^0 are produced with a Maxwell–Boltzmann distribution with $T = 0.40$ GeV. Consider the acceptance for a detector capable of identifying pions ($\pi^\pm$) in the momentum range $0.2 \leq p \leq 1.5$ GeV. Plot the acceptance and efficiency as a function of transverse momentum p_T and pseudorapidity η .
- 14.5 Write a Monte Carlo program using the weight method to simulate the production of correlated pions according to the HBT effect assuming source sizes of 2, 5, and 15 fm. Additionally assume the pions are produced with a Maxwell–Boltzmann distribution with $T = 0.40$ GeV.