PHOKHARA

version 2.0, December 2002

Abstract

The radiative return offers the unique possibility for a measurement of the cross section of electron–positron annihilation into hadrons over a wide range of energies [1]. The large luminosity of present ϕ- and B-factories easily compensates for the additional factor of α due to the emission of a hard photon. PHOKHARA is a Monte Carlo generator which simulates this process. It includes next-to-leading order corrections to initial state radiation (ISR) from virtual and real photon emission. In the present version the simulation has been extended from large photon angles into the collinear region, where the cross section is enhanced by large logarithms. In addition, final states with four pions are implemented and final state radiation (FSR) for muon and pion pair production is included in leading order.

1 Introduction

The first version of PHOKHARA [2] incorporates ISR only for $\pi^+\pi^-\gamma(\gamma)$ and $\mu^+\mu^-\gamma(\gamma)$ as final states. It exhibits, however, a modular structure that simplifies the implementation of additional hadronic modes or the replacement of the current(s) of the existing modes. The program provides predictions either at leading order (LO) or at next-to-leading order (NLO). In the former case only single photon events are generated. In the latter, both events with one or two photons are generated at random.

The first version of the program was designed to simulate configurations with photons emitted at relatively large angles, $\theta^2 \gg m_e^2/s$. In this case it is legitimate to drop terms proportional to m_e^2, an assumption that leads to a considerable simplification of the virtual corrections [3]. Analytical results for these virtual corrections, that are also valid in the small angle region [4], are now included in the new version of PHOKHARA [5]. The validity of the program is therefore extended into this small angle region.

Final state radiation, FSR, is also considered at LO for pion and muon pair production, as well as final states with four pions [6].

2 The program

PHOKHARA is written is FORTRAN 77. All the real variables and functions are defined `double precision`. The present distribution consist of the following files:

- **phokhara_2.0.f** : the main program.
- **phokhara_2.0.inc** : defines some variables and `COMMON` blocks which are used by most of the subroutines in the main program.
- **input_2.0.dat** : all the constants and specific parameters needed for the generation are given through this file. The values of these input parameters can be varied by the user.
- **seed.dat** : contains the seed used to initialize `RM48`, the double precision random number generator from the CERN Program library CERNLIB. After each generation run a new seed is stored in this file.

The program has to be linked with the CERNLIB library; e.g.

```
f77 phokhara_2.0.f `cernlib mathlib`
```

Further information and updates of the program can be find in the following web address: `http://cern.ch/grodrigo/phokhara`

3 Input file

The interaction of the user with the program is made through the file **input_2.0.dat**. It defines some physical constants and the specific parameters needed for the generation. The values of these parameters can be changed by the user. Some new variables have been included.

`nges` – number of events which shall be generated. The number of events accepted and returned in the output file depends on the kinematical constraints, the energy of the collision and the generated final state. Typically, the acceptance rate varies between 30% and a few per cent.

`nm` – number of events used to scan the integrand and find its maximum. A preliminary scan is maid to find the maximum(s) of the integrand(s) before the true generation starts. The value of the maximum used in the generation phase is slightly greater than this approximated maximum.

`outfile` – name of the output file where the four-momenta of the particles of the accepted events are stored.

`iprint` – whether the four-momenta of the generated events is printed (`iprint=1`) or not (`iprint=0`) through subroutine `writeevent` in the output file `outfile`.

`nlo` – whether the program should provide predictions at LO (`nlo=0`) or NLO (`nlo=1`).

`w` – energy cutoff of soft photon emission, normalized to the centre-of-mass energy. The physical result is independent of its value. Recommended value $w = 10^{-4}$.

`pion` – which final state channel shall be simulated:

`pion=0` : $\mu^+\mu^-$
`pion=1` : $\pi^+\pi^-$
`pion=2` : $2\pi^0\pi^+\pi^-$
`pion=3` : $2\pi^+2\pi^-$

`fsr` – only ISR is simulated (`fsr=0`), FSR is included at LO without ISR–FSR interference (`fsr=1`) or with ISR–FSR interference (`fsr=2`).

`tagged` – at least one photon is tagged (`tagged=0`) inside the angular region defined by the angular cuts `phot1cut` and `phot2cut`. In the untagged mode (`tagged=1`), the hadrons are tagged and the angular and energy cuts are applied on the missing energy-momentum.

The next set of parameters defines physical constants: coupling constants, masses, and decay widths. The following values are used:

$1/\alpha$ = 137.03599976 – fine structure constant

m_e = 0.51099906 $\cdot 10^{-3}$ GeV – electron mass

m_μ = 0.1056583568 GeV – muon mass

m_τ = 1.77703 GeV – tau mass

$m_{\pi^\pm}$ = 0.13956995 GeV – charge pion mass

m_{π^0} = 0.1349766 GeV – neutral pion mass

m_ρ = 0.7685 GeV – ρ-meson mass

Γ_{ee} = 0.677 $\cdot 10^{-5}$ GeV – partial decay width for $\rho \to e^+e^-$

Γ_ρ = 0.1507 GeV – total decay width of the ρ-meson

$m_{\rho'}$ = 1.37 GeV – ρ'-meson mass

Γ_ρ = 0.51 GeV – total decay width of the ρ'-meson

$m_{\omega'}$ = 0.78194 GeV – ω-meson mass

Γ_ω = 0.00843 GeV – total decay width of the ω-meson

For the parametrization of the pion form factor [10], two more constants are needed:

a = 0.00185

b = -0.145

These two constants are given by the fits from reference [10]. Other parametrizations can be introduced in the program from subroutine `PionFormFactor`.

The remaining set of parameters defines the specific experimental settings:

`E` – centre-of-mass energy (GeV).

`q2min` – minimal squared invariant mass of the system formed by the hadrons and the tagged photon (GeV^2).

`q2_max_c` – maximal squared invariant mass of the hadronic/muonic system (GeV^2).

`gmin` – minimal energy of the tagged photon (GeV).

`phot1cut` – lower cut on the azimuthal angle of the tagged photon (degrees).

`phot2cut` – upper cut on the azimuthal angle of the tagged photon (degrees).

`pi1cut` – lower cut on the azimuthal angle of the muons or pions (degrees).

`pi2cut` – upper cut on the azimuthal angle of the muons or pions (degrees).

All the kinematical cuts are given in the centre-of-mass system of the initial particles. The azimuthal angles are defined with respect to the positron momentum.

The program offers the possibility of presenting various differential distributions as histograms. If this option is used, the name of the output file where the histograms are stored and the attributes of each histogram must be given.

`fname` – name of the output file where the histograms are stored.

`title(i)` – title of histogram `i`

`xlow(i)` – lower edge in x for histogram `i`

`xup(i)` – upper edge in x for histogram `i`

`bins(i)` – number of bins for histogram `i`

where `i=1,...,10`.

4 Output

PHOKHARA presents the output information in several forms and saves it in different files.

The four-momenta of the particles of the accepted events are stored in the file given by `outfile`. The format of the output is determined by the subroutine `writeevent(pion)` and can be changed by the user. All the momenta are given in the centre-of-mass system of the colliding electron and positron.

The subroutine `inithisto` books the histograms being based on the information given by the input file. The subroutine `endhisto(fname)` fills the histograms at the end of the generation run and save the result in the output file given by `fname`. These two subroutines use `HBOOK` to create and manipulate the histograms. A later manipulation of the histograms is possible with `PAW`. For accuracy reasons – `HBOOK` works only in single precision – the histogram information is stored in the intermediate steeps of the calculation in the matrix `histo(i,j)`, where `i` identifies the histogram number and `j` the bin. Only at the end of the generation run it is transferred to the histograms. The user can therefore use its favourite histograming tool simply modifying the subroutines `inithisto` and `endhisto`.

By default only the Q^2 distribution, where Q^2 is the squared invariant mass of the hadronic (muonic) system, is calculated. The contribution from single photon events is stored in histogram `1`. The contribution from two photon events is stored in histogram `2`. The final Q^2 distribution is given by the sum of both results. Other differential distributions can be defined though the subroutine `addiere(wgt,qq,i)`, where `wgt` is the weight of the event, `qq` is the value of Q^2 and `i` is equal to 1 for single photon events and 2 for two photon events. The four-momenta of the events are given by the matrix `momenta(i,0:3)`, where `i=1,...,7`, for the positron, the electron, the two real photons (for single photon events `momenta(4,0:3)` is set to zero), the virtual photon converting into hadrons, the $\pi^+(\mu^+)$ and $\pi^-(\mu^-)$ respectively. It can be used by the user to define other differential distributions.

At the end of the run, PHOKHARA displays also the total number of accepted events, the value of the cross section, the value of the scanned maximum(s) and the biggest value of the integrand(s) found during the Monte Carlo generation. The last should be always smaller than the scanned maximum. If during the generation a value of the integrand is found to be bigger than the scanned maximum, a warning is given. Then, the number of events used for the initial scan should be increased.

5 Forthcoming features

- Inclusion of FSR at NLO.
- Simulation of other exclusive hadronic channels.

References

[1] S. Binner, J. H. Kühn and K. Melnikov, Phys. Lett. **B459** (1999) 279 [hep-ph/9902399].

[2] G. Rodrigo, H. Czyż, J. H. Kühn and M. Szopa, Eur. Phys. J. C **24** (2002) 71 [hep-ph/0112184].

[3] G. Rodrigo, A. Gehrmann-De Ridder, M. Guilleaume and J. H. Kühn, Eur. Phys. J. C **22** (2001) 81 [hep-ph/0106132].

[4] J. H. Kühn and G. Rodrigo, Eur. Phys. J. C **25** (2002) 215 [hep-ph/0204283].

[5] H. Czyż, A. Grzelińska, J. H. Kühn and G. Rodrigo, hep-ph/0212225.

[6] H. Czyż and J. H. Kühn, Eur. Phys. J. C **18** (2001) 497 [hep-ph/0008262].

[7] G. Rodrigo, Acta Phys. Polon. B **32** (2001) 3833 [hep-ph/0111151].

[8] H. Czyż, J. H. Kühn and G. Rodrigo, hep-ph/0211186; hep-ph/0210287; hep-ph/0205097.

[9] J. H. Kühn, Nucl. Phys. Proc. Suppl. **98** (2001) 289 [hep-ph/0101100].

[10] J. H. Kühn and A. Santamaria, Z. Phys. C **48** (1990) 445.