

BBBREM – Monte Carlo simulation of radiative Bhabha scattering in the very forward direction

R. Kleiss

NIKHEF-H, P.O. Box 41882, 1009 DB Amsterdam, The Netherlands

H. Burkhardt

SL Division, CERN, 1211 Geneva 23, Switzerland

Abstract

A fast and simple Monte Carlo program is presented that simulates single Bremsstrahlung in Bhabha scattering, $e^+e^- \rightarrow e^+e^-\gamma$, without constraints on scattering angles. This allows the study of this process at arbitrarily small, or even vanishing, scattering angles. Experimental cuts can be imposed on an event-by-event basis, allowing for detailed studies of the process as a limitation to beam lifetimes, or a luminosity-measuring device, in e^+e^- storage rings. As an application, we show that the easy introduction of a cut-off parameter, corresponding to the characteristic distance between particles in the e^\pm bunches, gives a reduced cross section that is in good agreement with observation.

submitted to Computer Physics Communications

Program Summary

Title of program: BBBREM

Program obtainable from: R. Kleiss, NIKHEF-H, P.O. Box 41882, 1009 DB Amsterdam, The Netherlands, t30@nikhef.nikhef.nl

Computers: systems supporting standard FORTRAN 77

Programming language used: FORTRAN 77

Memory required: about 200kb

number of bits per word: 32

Subprograms used: none

Number of lines in distributed program: 386

Keywords: Bhabha scattering, Bremsstrahlung, radiative processes, forward scattering, collinear singularities, Monte Carlo simulation, experimental cuts, beam lifetimes, luminosity monitoring

Nature of physical problem: Radiative Bhabha scattering, $e^+e^- \rightarrow e^+e^-\gamma$, has a very large cross section at small scattering angles, and plays various rôles in existing and future e^+e^- colliders. It can be an important background to several two-photon scattering processes; it forms the major ingredient in the finite lifetime of colliding beams; and it is a possible process by which the luminosity can be measured, by observing electrons or photons emerging at zero scattering angle. Accurate knowledge of its cross section is therefore important.

Method of solution: Due to the extremely singular structure of the matrix elements, and the possibility of complicated or unusual experimental cuts, a straightforward integration of the cross section over the allowed phase space is impractical. We therefore construct a Monte Carlo algorithm that generates (random) events in phase space, with a distribution that matches the actual

cross section as closely as possible. These events are assigned a *weight* which corrects for discrepancies between the actual and the approximate matrix elements: the average value of the weight in a sample of generated events is the Monte Carlo estimate of the cross section. Since each generated event is a complete description of the momenta of the produced particles, any conceivable experimental cut can be implemented, in addition to the single a-priori constraint, namely, a minimum value for the energy of the Bremsstrahlung photon. This value can be set (to essentially arbitrarily small values) by the user. By setting the weight of events that fail a particular set of cuts to zero, one obtains the cross section under those cuts.

Typical running time: about 185 μsec per generated event on SUN SPARC 10; about 40 μsec per generated event on IBM 9000.

Unusual features of the program: none

Long Write-Up

1 Introduction

The process of radiative Bhabha scattering,

$$e^-(p_1) e^+(q_1) \rightarrow e^-(p_2) e^+(q_2) \gamma(k) , \quad (1)$$

where we have indicated the various momenta, is expected to play an important rôle in several physical problems. In the first place, it is the main process by which electrons are lost upon the collisions of e^\pm beams, and dominates the beam lifetime of LEP [1]. Secondly, it may be an important background to processes where single photons or π^0 's are observed at small angles, such as at the DAΦNE collider [2]. One therefore intends to install tagging devices that can, in principle, observe electrons and photons emitted at zero scattering angle. Finally, this last device can also serve as a luminosity monitor

due to the very large event rate. It is obvious that the various cross sections related to this process have to be known accurately.

Various authors have already treated the process (1), using a variety of approximations to the matrix element. Since scattering over very small angles completely dominates the process, Bremsstrahlung off the positron and off the electron are essentially independent, and we use only the two Feynman diagrams where the electron is seen to radiate the photon. On the other hand, when all the particles' trajectories are essentially collinear, the electron mass m cannot be neglected with impunity. The result of the CALKUL collaboration [3], where the leading correction terms of order m^2/E^2 are included (where E is the beam energy), is also inapplicable since it assumes that the electron mass is negligible with respect to t , the momentum transfer of the positron. Of the early publications, the work of Altarelli *et al.* must be mentioned. They computed the total cross section [4] and various differential distributions [5]. Similar results were obtained by a number of Russian authors [6]. With the possibility of actually measuring the process (1) at small or zero angle, these analytic, totally inclusive, results are no longer adequate, and the use of approximate matrix elements somewhat doubtful. In [7] a version of the matrix element, appropriate to small-angle scattering, and correct up to truly negligible terms, was described in detail. In two recent publications, and Italian collaboration has presented results for cross sections obeying a variety of angular and energy cuts [8], and described the integration program used to obtain these [9] (the matrix element employed in [8] has, in fact, been checked to be completely identical to that of [7]).

In this paper we describe another approach to the study of (1), namely that of a full-fledged Monte Carlo simulation. The user generates any desired number of *events*, that is, sets of momenta of the three produced particles. Their distribution in phase space is matched as closely as possible to the exact distribution. The remaining factor is included in each event's *weight*. The average of the weights in the sample is the Monte Carlo estimate of the cross section. This is a very flexible approach since *any* experimental cut can be imposed by simply setting the weight of events that fall outside the cuts to zero. Moreover, since the momenta are explicitly given, the produced particles can be tracked by other simulation programs, either into a detector, or in the accelerator. A good check is provided by the fact that those cuts that are provided for in [9] must also yield the cross section calculated by

that program, by completely different means¹.

2 The Monte Carlo

We shall now describe our approach to the cross section of Eq.(1). We start by defining the necessary variables. We shall use the following invariants:

$$\begin{aligned} s &= (p_1^\mu + q_1^\mu)^2 = 4E^2 \quad , \quad s_1 = (p_2^\mu + q_2^\mu)^2 \quad , \\ t &= (q_1^\mu - q_2^\mu)^2 \quad , \quad \Delta_1 = p_1^\mu k_\mu \quad , \quad \Delta_2 = p_2^\mu k_\mu \quad . \end{aligned} \quad (2)$$

Together with the azimuthal orientation of the event around the beam axis, these serve to completely specify all the momenta. In terms of these invariants, the correct form of the matrix element, squared and summed/averaged over the particle spins, is given by

$$\begin{aligned} \frac{1}{4} \sum_{\text{spins}} |\mathcal{M}|^2 &= e^6 M \quad , \\ M &= \frac{1}{\Delta_1 \Delta_2 |t|} \left[s^2 + s_1^2 + (s + t - 2\Delta_2)^2 + (s_1 + t + 2\Delta_1)^2 \right] \\ &\quad - \frac{4m^2}{\Delta_1^2 \Delta_2^2} \left[(s - 2m^2)\Delta_1 - (s_1 - 2m^2)\Delta_2 - 2\Delta_1 \Delta_2 \right]^2 \\ &\quad - \frac{8m^2}{\Delta_1 \Delta_2 t^2} \left[\Delta_1^2 + \Delta_2^2 \right] \quad , \end{aligned} \quad (3)$$

where e is the electron charge. The distribution of the generated events over the phase space shall be proportional the *approximate* matrix element

$$M_{\text{app}} = \frac{4s^2 C_1 C_2}{\Delta_1 \Delta_2 |t|} \quad , \quad (4)$$

where C_1 and C_2 are defined later on. This simple form captures the essentials of the process dynamics, and also, owing to clever definitions of the $C_{1,2}$, leads to a simple result when integrated over the whole phase space. Denoting this

¹A small discrepancy was actually observed, due to the fact that in [9] the electromagnetic coupling constant α used was $1/(138.318)$ instead of $1/(137.036)$.

integral (or approximate total cross section) by σ_{app} , we then define the weight of a generated event by

$$\text{weight} = \frac{M}{M_{\text{app}}} \sigma_{\text{app}} \quad , \quad (5)$$

for those events that satisfy the desired cuts, and zero otherwise. Note here, that only one cut is always present, namely a lower limit on the photon energy. This lower limit is given by κE , where $0 < \kappa < 1$, and serves to avoid the infrared singularity lurking at $k^0 = 0$.

We shall deal with two Lorentz frames: the laboratory frame, and the R -frame, that is, the rest frame of $R^\mu = p_2^\mu + k^\mu$. The most convenient picture of the process (1) is then the emission of a virtual photon by the positron (which is best described in the lab frame), followed by ‘Compton’ scattering of the virtual photon and the electron (which finds its easiest description in the R -frame). We therefore have as ‘natural’ invariants the following set:

- the components of the positron momentum transfer $q^\mu = q_1^\mu - q_2^\mu$, namely $t = q^\mu q_\mu$, $y = q^0/E$, and ϕ , the azimuthal angle of \vec{q}_2 around the beam axis. These are all defined in the lab frame.
- the scattering polar angle θ_γ and azimuthal angle ϕ_γ of the photon with respect to the incoming electron momentum. These are defined in the R -frame, with $c_\gamma = \cos \theta_\gamma$.

In terms of these variables, the phase space integration element can be written as

$$\begin{aligned} d\Phi &\equiv d^4 p_2 \delta(p_2^2 - m_2^2) d^4 q_2 \delta(q_2^2 - m^2) d^4 k \delta(k^2) \delta^4(p_1 + q_1 - p_2 - q_2 - k) \\ &= \frac{w^2 - m^2}{32w^2} dy dt d\phi dc_\gamma d\phi_\gamma \quad , \end{aligned} \quad (6)$$

where $w^2 = R^\mu R_\mu = sy + m^2$. The only nontrivial phase space bounds are those on t : we have $t_{\min} \leq t \leq t_{\max}$, with

$$\begin{aligned} t_{\min} &= 2m^2 - 2Eq_2^0 - 2 \left[(m^2 - Eq_2^0)^2 - m^2(E - q_2^0)^2 \right]^{1/2} \quad , \\ t_{\max} &= m^2 sy^2 / t_{\min} \quad . \end{aligned} \quad (7)$$

Now, y is typically very small, of order m^2/E^2 ; therefore, $|t_{\max}|$ can become *extremely* small.

We may now write the invariants $\Delta_{1,2}$ in terms of the chosen variables:

$$\begin{aligned}
\Delta_2 &= sy/2 \ , \\
\Delta_1 &= sy\lambda(w^2, m^2, t)(\epsilon + v_\gamma)/(4w^2) \ , \\
\lambda(w^2, m^2, t) &= [(w^2 + m^2 - t)^2 - 4w^2m^2]^{1/2} \ , \\
\epsilon &= (w^2 + m^2 - t)/\lambda(w^2, m^2, t) - 1 \ , \\
v_\gamma &= 1 - c_\gamma \ .
\end{aligned} \tag{8}$$

Now, the approximate differential cross section is

$$d\sigma_{\text{app}} = \frac{\alpha^3 C_1 C_2}{\pi^2 |t| \lambda(w^2, m^2, t) y (\epsilon + v_\gamma)} dy dt dc_\gamma d\phi_\gamma d\phi \ . \tag{9}$$

The angular integrals are now trivial, and we find

$$d\sigma_{\text{app}} = 4\alpha^3 \log\left(\frac{2+\epsilon}{\epsilon}\right) \frac{C_1 C_2}{|t| y \lambda(w^2, m^2, t)} dy dt \ . \tag{10}$$

we now choose C_1 so as to simplify this expression:

$$C_1 \equiv \left[\log\left(1 + \frac{sy}{m^2}\right) \right] \left[\log\left(\frac{2+\epsilon}{\epsilon}\right) \right]^{-1} \ ; \tag{11}$$

this approximation is justified when $|t|$ is very small. We then have

$$d\sigma_{\text{app}} = 4\alpha^3 \log\left(1 + \frac{sy}{m^2}\right) \frac{C_2}{|t| y \lambda(w^2, m^2, t)} dy dt \ . \tag{12}$$

Now we do the t integral; owing to our choice of C_1 , it gives the fairly simple result

$$\begin{aligned}
d\sigma_{\text{app}} &= \frac{4\alpha^3 C_2}{sy^2} \log\left(1 + \frac{sy}{m^2}\right) [J(t_{\max}) - J(t_{\min})] dy \ , \\
J(t) &= \log\left(\frac{\lambda(w^2, m^2, t) + sy - t}{\lambda(w^2, m^2, t) - sy - t}\right) \ .
\end{aligned} \tag{13}$$

It is now time to choose C_2 :

$$C_2 = 2 \log\left(\frac{s}{m^2}\right) [J(t_{\max}) - J(t_{\min})]^{-1} \ , \tag{14}$$

which goes to 1 in the typical case where y is small. The final integral is best performed in terms of the variable $z \equiv sy/m^2$:

$$d\sigma_{\text{app}} = \frac{8\alpha^3}{m^2} \log\left(\frac{s}{m^2}\right) \frac{\log(1+z)}{z^2} , \quad (15)$$

and, taking the upper limit on z to infinity for simplicity, we finally have

$$\sigma_{\text{app}} = \frac{8\alpha^3}{m^2} \log\left(\frac{s}{m^2}\right) \left[\log\left(\frac{1+z_0}{z_0}\right) - \frac{\log(1+z_0)}{z_0} \right] , \quad (16)$$

where z_0 is the lower limit on z . In practice, we want to impose a lower limit on k^0 rather than on z , or y , since q_2^0 will usually be extremely close to its maximum. However, for fixed k^0 , the value of y is bounded by

$$y \geq \frac{m^2}{s} \left(\frac{k^0}{E - k^0} \right) \left(\frac{2}{1 + \sqrt{1 - \frac{m^2}{E(E - k^0)}}} \right) , \quad (17)$$

which is reached when \vec{p}_2 and \vec{k} are collinear and opposite to \vec{q}_2 . So, unless κ is very close to 1, we have a lower bound

$$z \geq z_0 \equiv \frac{\kappa}{1 - \kappa} , \quad (18)$$

and we shall use this in the event generator. Note, however, that upon generating events with this lower bound, we unavoidably also get events with $k^0 < \kappa E$. These we discard by putting their weight to zero: even without additional experimental cuts, therefore, the generated event sample will contain a modest fraction of events with zero weight.

We are now in a position to generate the events. This is done by applying mappings in the order opposite to that of the integration, namely, first z , distributed according to Eq.(15), then t , according to Eq.(12), and finally the angular variables, of which only v_γ is not completely trivial. One minor remark is in order here: the generation algorithm assumes an infinite upper limit on z . Very occasionally, therefore, a y value will occur that is unphysically large. This is immediately indicated by t_{min} becoming complex, in which case the event weight is put to zero and the event skipped. After having obtained all the phase space variables, we can construct the

momenta of the outgoing particles. This is in principle straightforward, but is made tricky by the enormous Lorentz boost necessary to go from the lab frame to the R -frame and back. An ample collection of numerical pitfalls occurs, which we avoid by using expressions in which the major cancellations (from order 1 down to order m^2/E^2) have been performed by hand. Finally, we calculate M and M_{app} , also with an eye to the numerical stability. For the computation of M in an numerically stable manner, we refer to [7]. In addition, since $|t|$ is usually very small, t^2 will occasionally lead to underflow problems. We therefore evaluate t^2M and t^2M_{app} instead. Finally, we remark that ordinary double precision (`REAL*8`) suffices for our numerical work; extended precision (`REAL*16`), such as employed in [9], is not necessary.

3 Running BBBREM

The Monte Carlo has a very limited number of input parameters, read at initialization from the standard input unit. They are, in order:

1. `ROOTS` the total CMS energy of the incoming e^+e^- system, in GeV. Any realistic value is allowed, from around 1 GeV (such as for the DAΦNE machine) up to hundreds of GeV for NLC/CLIC.
2. `RKO` the photon energy cutoff fraction κ . In realistic cases, κ will be of the order of percents, but it can be put much lower (although not to zero). When `RKO` increases, the fraction of internally rejected events (see the discussion above) increases slightly. `RKO=1` is not allowed.
3. `NEVENT` the requested number of generated events. This includes events that come out with zero weight.
4. `NRAN` a flag determining the source of (pseudo-)random numbers used. In principle, any reliable such source may be substituted instead of the routine `RANDOM`, which occurs at 8 places in the generator code. We provide two alternatives:
 - `NRAN=1` A *very* crude, low-period multiplicative congruential algorithm. Its only merit is portability, and it should only be used for checking the program against the test run mentioned below.

- **NRAN=2** A simple additive quasi-random number algorithm. We have used this to obtain the numerical results mentioned below. It is, however, not strictly portable since different rounding procedures on different machines lead to different sequences. The results below should, however, be reproduced to within statistical accuracy.

The input parameters **ROOTS** and **RKO** are transferred to the event generator subroutine **BCUBE** by common

```
COMMON/EXPERC/ ROOTS,RKO
```

and **NRAN** is transferred to the random number source **RANDOM** by common

```
COMMON/RANCHN/ NRAN
```

When called the first time, **BCUBE** initializes, and prints the result for σ_{app} , in millibarn. Note that this is *not* intended as an accurate numerical approximation to the actual cross section! Upon each call, **BCUBE** fills the common

```
COMMON/LABMOM/ P1(4),P2(4),Q1(4),Q2(4),QK(4),D1,D2,T,WEIGHT
```

which contains the essential information on the event: the four-momenta p_1^μ , p_2^μ , q_1^μ , q_2^μ and k^μ (the fourth component being the energy), all in GeV; the invariants Δ_1 , Δ_2 , and t , all in GeV^2 , whose computation, directly from the returned momenta, is numerically unstable; and, finally, the event weight, in millibarn. As given, the main program yields, after the generation of **NEVENT** events, the following significant numbers:

1. **RANCHK** the result of one last call to **RANDOM**. This serves as a check on the portability of the program, since it depends both on the actual performance of the random number generator, and the *number* of random numbers that was actually used — this number is variable due to various decision and rejection steps.
2. **SMC** the computed cross section, *i.e.* the average weight.
3. **SER** the estimated error on the value of **SMC**, defined in the standard way, using the variance of the weight distribution.

In addition, some information on the weight distribution is given: **W1**, the sum of the weights; **W2**, the sum of their squares; **WNIL**, the number of events with zero weight; **WNEG**, the number of events with *negative* weight: these should not occur, but if they do this would indicate rounding problems, or the use of an inappropriate form for M (for instance, negative weights arise if the form of ref.[5] is used, although the cross section is still essentially correct); and, finally, **WMAX**, the maximum weight occurring in the generation. Concerning this last number it should be remarked that usually **WMAX** is quite a bit larger than **SMC**: the weight distribution has a low tail. This can be traced back to small values of C_1 . In principle, the t integral in Eq.(12) can also be performed exactly, without the benefit of C_1 , leading to a slight complication in the algorithm. This *might* be desirable in such cases as necessitate the use of *unweighted* events, since high values of **WMAX/SMC** reduce the efficiency for unweighted events; for instance, if the produced momenta are to be used for tracking in machine studies. There, however, the tracking itself takes so much time that a somewhat inefficient unweighted-event generation is not a problem. We have therefore decided for the simpler algorithm. In any case, the high-weight tail does not prohibit the attainment of a small Monte Carlo error estimate.

4 Results

We shall now discuss some results obtained with **BBBREM**. In the first place, we reproduce here the output of a test run with the following choice of parameters: **ROOTS=91.2d0**, **RK0=0.01d0**, **NEVENT=10000**, and **NLAN=1**. It is:

```

*****
*** bbbrem : beam-beam bremsstrahlung ***
*** authors r. kleiss and h. burkhardt ***
*****
total energy                91.200000 gev
minimum photon energy       0.010000 * beam energy
approximate cross section   0.627848D+03 millibarn
random: very crude random numbers chosen
***** cross section evaluation *****

```

```

random number check      0.1471576690674
total # of events        10000.
sum of weights           0.323102D+07
sum of (weight**2)s      0.390034D+10
max weight occurred      0.221525D+05
# of zero weights        1805.
# of negative weights     0.
computed cross section   0.323102D+03 millibarn
                        +/-      0.534452D+01 millibarn
*****

```

We now turn to a more physical application of BBBREM, concerning the lifetimes of e^+e^- colliding beams. As discussed in [1], the process (1) is actually the dominant process by which electrons and positrons are lost from the beams in storage rings such as LEP: an energy loss of an electron/positron due to bremsstrahlung that exceeds the so-called r.f. bucket half-height s_b cannot be made up for by the r.f. system, and the particle will be lost. At LEP and most other e^+e^- machines, s_b is typically about 10^{-2} . One can therefore calculate, from first principles, the corresponding cross section: at $\sqrt{s} = 91.2$ GeV, corresponding to LEP running at the Z^0 peak, one finds then $\sigma \sim 320$ mb, both from BBBREM and analytical calculations such as [5]. Actual measurement [1] finds, however, a cross section of about 210 mb. As discussed in detail in [1], we ascribe this reduction to a density effect: inside each bunch, the electromagnetic fields from all electrons (say) overlap in such a way that each electron has a finite interaction range, in contrast to the usual infinite range. This range is of the order of half the average distance d between two electrons in the bunch (at rest!), which is [1] about $3.3 \mu\text{m}$, corresponding to a momentum-transfer squared $(\hbar c/d)^2 = 3.56 \cdot 10^{-21} \text{ Gev}^2$. Denoting this number by t_c , we may then impose a cutoff on the momentum transfers allowed in the process (1), in two ways:

- hard cutoff: events with $|t| < t_c$ are forbidden. In the Monte Carlo program this just means putting their weight to zero.
- soft cutoff: the photon mediating the positron scattering is assigned an effective mass $\hbar c/d$, leading to an effective exponentially suppressed electromagnetic potential of the electrons. In the Monte Carlo, this is

very simply implemented by multiplying the event weight with a factor $t^2/(t - t_c)^2$.

Note that it would be very hard to implement either of these two cut-off procedures in any way but that of Monte Carlo. Below we present results for four different values of `ROOTS`: 1.02 GeV, corresponding to DAΦNE at the Φ -meson peak, 60 GeV, appropriate to TRISTAN, 91.2 GeV, corresponding to LEP 1 at the Z^0 peak, and 190 GeV, typical for LEP200. In each case we have used `RK0=0.01`, `NEVENT=1000000`, and `NRAN=2`. The cross sections are given in millibarn, for each of the three cases: no cutoff, hard cutoff, soft cutoff.

<code>ROOTS</code> (GeV)	no cutoff	hard cutoff	soft cutoff
1.02	208.1	198.7	194.6
60	308.6	210.2	204.0
91.2	318.9	210.1	203.9
190	336.8	209.8	203.7

The estimated error is 0.4 millibarn for DAΦNE, and 0.5 millibarn in the other cases. We see that the LEP 1 cross sections with the cutoff are in much better agreement with the observation. For more details about this physical, rather than computational, issue we refer to [1].

References

- [1] H. Burkhardt, Proceedings of the third Workshop on LEP Performance, J. Poole, ed., CERN SL/93-19 (DI); R. Kleiss and H. Burkhardt, in preparation.
- [2] The DAΦNE Physics Handbook, L. Maiani, G. Pancheri, and N. Paver, eds.; G. Alexander *et al.*, preprint LNF-93/030 (P); KLOE collaboration, report LNF-92/019 (R).
- [3] F.A. Berends *et al.*, Nucl. Phys. **B206** (1993) 61.
- [4] G. Altarelli and F. Bucella, Nuovo Cim. **34** (1964) 1337.
- [5] G. Altarelli and B. Stella, Lett. al Nuovo Cim. **9** (1974) 416.
- [6] V.N. Baier, V.S. Fadin and V.A. Khoze, Sov. Phys. JETP **24** (1967) 760; **26** (1968) 1238.
- [7] R. Kleiss, Phys. Lett. **B318** (1993) 217.
- [8] M. Greco, G. Montagna, O. Nicrosini and F. Piccinini, Phys. Lett. **B318** (1993) 635.
- [9] G. Montagna, F. Piccinini and O. Nicrosini, Comp. Phys. Comm. **78** (1993) 155.