USING THE EGS ELECTROMAGNETIC SHOWER SIMULATION PROGRAM FOR SSC CALCULATIONS

David E. Wagoner and Dennis J. Judd
Florida A&M University, Tallahassee, FL 32307

Summary

EGS is an accurate and general program for the simulation of electromagnetic showers. The operation of the EGS system is explained in general and for the particular system used at Fermilab. EGS is compared to GEANT3 whose electromagnetic shower simulation is very similar to that of EGS. EGS (or GEANT3) will be needed in some measure to understand the electromagnetic shower behavior of the SSC detector. The principal drawback of either program is the large amount of CPU time that they require to analyze typical SSC events. A faster empirical approach to electromagnetic shower simulation using EGS (or GEANT3) for its development and testing will be probably be necessary for the majority of SSC events.

Introduction

The EGS Monte Carlo system (EGS3[1] or EGS4[2]) is a program developed at SLAC for the simulation of electromagnetic showers. EGS is becoming the standard electromagnetic shower simulation program in the high energy physics community. This note contains information on the use of the EGS system based on practical experience with running EGS. There are also comments on the applicability of EGS for SSC energies and detectors and is compared to the electromagnetic shower simulation capabilities of GEANT3 and other schemes. In the remaining discussion, when EGS is mentioned it is the EGS3[1] system that is generally being referred to since that is the system the authors have the most experience with. EGS4[2] is a recently released and updated version of EGS and, except where specifically noted, all comments about EGS3 should also apply to EGS4.

The EGS Simulation Program

EGS (Electron-Gamma Shower) is an analog Monte Carlo written to simulate three-dimensional electromagnetic showers in any media. Showers are developed by simulating in as much detail as possible the various electromagnetic shower processes. The probability distributions of the processes are used, so an EGS shower simulation mimics in detail real showers with real fluctuations. EGS itself is geometry independent; the detector geometry is communicated to EGS through a user-written subroutine which EGS calls. Full information about the shower throughout its development is available to the user. EGS is quite straightforward to use with only a minimal amount of overhead and protocol.

EGS can simulate showers over a wide energy range. Accurate shower simulations with EGS3 can be done for photons from 100 keV to 100 GeV and for electrons and positrons from 1.5 MeV to 100 GeV. This energy range is extended in EGS4. EGS4 can accurately simulate photons from 1 keV to 3 TeV and electrons and positrons from 10 keV to 3 TeV. One can also go outside these energy ranges at the expense of a possible loss of accuracy.

EGS code is written in the MORTRAN language (EGS3 in MORTRAN2 and EGS4 in MORTRAN3.) MORTRAN is a free-form FORTRAN-like language with an extended control structure and a very powerful macro facility. The MORTRAN compiler is a pre-processor which inputs a MORTRAN program (EGS in this case) and which outputs a FORTRAN program which can then be compiled and run normally. Options exist in MORTRAN to input straight FORTRAN code, but then one loses especially the power of the macro facility. Since MORTRAN is a non-standard language its use may be difficult on certain computers. For example, EGS3 has been set up to run on the Cybers at Fermilab, however, EGS4 does not run on the Fermilab Cybers because of problems with the MORTRAN3 language on the Cyber. On the other hand both EGS3 and EGS4 run on VAX and IBM machines.

The EGS system itself is a set of MORTRAN subroutines. The user actually runs EGS for his particular detector by writing (usually in MORTRAN) the main program and two user subroutines (HOWFAR and AUSGAB.) The main program first sets up certain variables in EGS common blocks, sets up the user histograms, and calls an EGS subroutine (HATCH) which reads in cross-section data for the various media being used. It then goes into an event loop where an EGS subroutine (SHOWER) is called initiating the simulation of an electron, positron, or photon shower (or also a π^0 in EGS4) with a specified energy, position, and angle. Exiting the event loop, the user can then output the various shower quantities studied or histogrammed.

EGS communicates with the user by calling two user-written subroutines. User subroutine HOWFAR is called to transmit to EGS information about the detector geometry being simulated. The geometry structure of EGS itself is very general and nonspecific. Random step sizes (rather than fixed length steps) are generated by the particle transport routines. When the step size generated by the particle transport is greater than a user specified minimum the user routine HOWFAR is called. In HOWFAR the user must determine whether the proposed step will cross a detector boundary, and if it does what the distance to the boundary is and what the region number of the new region is. The user can also update the minimum step size that EGS will check on the next step, usually as the distance to the nearest boundary. EGS will then transport the particle by the returned (and possibly decreased) step size. However if the original step size was less than the user specified minimum then particle transport will proceed immediately without HOWFAR being called; use of this feature can greatly speed up the simulation by reducing the number of geometry checking calls. Here all the complexity of the geometry is contained in the user-written routine HOWFAR with only a general communication protocol with EGS.

Information about the shower development is communicated through the user-written subroutine AUSGAB. Whenever energy deposition or other interactions occur EGS calls AUSGAB. In AUSGAB the user can examine the various EGS common blocks at that point in the shower development and histogram the various quantities (e.g., the energy deposited at that spatial point or number of particles crossing a particular plane) that are to be studied. AUSGAB is also where detector characteristics (e.g., optical attenuation or chamber efficiency) can be introduced into the simulation. EGS gives the user information only about the intrinsic quantities of the developing shower. It is up to the user to convert these quantities (e.g., energy deposition or charged track length) into the actual quantities coming out of the detector (e.g., integrated charge or number of photoelectrons.)

Running EGS at Fermilab

EGS3 is set up to run on both the Cyber and VAX at Fermilab. A user's guide to running EGS3 on the Cyber is contained in the indirect access file EGSDOC/UN=95069. A similar user's guide for running EGS3 on the VAX at Fermilab can be found in the file (via Fermilab DECnet) FNAL03::USR$ROOT1:[E705.EGS]EGS.DOC . EGS3 files for the VAX can also be found in this subdirectory. As mentioned previously EGS4 does not run on the Fermilab Cyber. The EGS4 for the VAX system can be found in the subdirectory (again via Fermilab DECnet) FNAL03::USR$ROOT1:[E705.EGS4] . The EGS4 system can also be obtained by writing Walter R. Nelson at SLAC.

Timing and Size Considerations of EGS

Since EGS is an analog Monte Carlo the CPU time to simulate an electromagnetic shower depends linearly on the energy of the shower. This CPU time is relatively independent of the material used, but depends strongly on the low energy cutoffs used, the complexity of the geometry, and the depth cutoff of the shower development. The average CPU time to simulate an EGS3 shower through 24 radiation lengths with a simple 12 layer planar geometry using the minimum EGS3 energy cutoffs (0.1 MeV for photons and 1.5 MeV for electrons and positrons) is approximately 0.5 Cyber 175 CPU seconds/GeV/shower.

This time can be reduced substantially if the low energy cutoffs are raised. In EGS when a particle's energy falls below the cutoff that particle is immediately discarded and its energy is entirely deposited at the spatial point where cutoff occurred. So if a user is interested in the low energy tails or leakages from a shower, he should try several test runs with different cutoffs to find the optimum tradeoff of speed vs precision of the simulation.

Another critical timing factor is the depth cutoff of the shower. A job to simulate a shower in the first 4 or 5 radiation lengths for example can run many times faster than one to simulate a shower through 20 or more radiation lengths.

Finally the complexity of the detector geometry can greatly increase the CPU time for a shower. EGS spends a large fraction of its time calling the HOWFAR geometry routine, so this user subroutine should be written as efficiently as possible. Also note that tracking a shower through a magnetic field can increase the CPU time by over an order of magnitude since this usually necessitates limiting the maximum step size in the EGS simulation to a small value resulting in even more calls to HOWFAR.

EGS3 requires a minimum of approximately 57 kwords of memory to run on the Fermilab Cyber. This is approximately 56% of the total memory on the Cyber. This does not include space needed for histogramming or the additional code required to describe complex geometries.

EGS4 (on the VAX) runs with a speed and size comparable to that of EGS3, although EGS4 does run slightly faster. EGS4 also includes an option to run with importance sampling. Using this option can speed up the simulation by factors of 100 to 300.[3] However importance sampling can be used only to determine average shower properties, for example average shower leakage from a detector. If accurate values for shower fluctuations are also required, then one must use the slower analog simulation.

The GEANT3 Electromagnetic Shower Simulator

GEANT3[4] is another widely used electromagnetic shower simulation system. An electromagnetic shower in GEANT3 is simulated in nearly the same analog fashion as one in EGS. Some slight differences do exist between EGS and GEANT3. For example multiple scattering is handled slightly differently and GEANT3 simulates fluctuations in the ionization loss by either sampling a Landau distribution or generating explicitly the delta rays. EGS3 and EGS4 both generate delta rays, although only EGS4 has an option for sampling the Landau distribution.

The two programs have been compared[5] and it is found that the electromagnetic showers from GEANT3 are nearly identical to those from EGS3. The main difference between the programs is that GEANT3 had its sophisticated geometry handling capability built into the simulator from the beginning. GEANT3 is written to simulate accurately electromagnetic showers from 10 keV to 10 TeV. For relatively simple geometries GEANT3 can run between 10% to 70% faster than EGS3 depending on the particular geometry and energy cutoffs used.[5] For more complex geometries this speed comparison depends on how efficient is the EGS user-written subroutine HOWFAR vs the overhead of the GEANT3 geometry structure.

An additional feature of GEANT3 which does not exist in EGS is the ability to simulate the electromagnetic interactions of muons (and other minimum ionizing particles.) At SSC energies the electromagnetic interactions of muons may become important when considering various signals and backgrounds.

Since GEANT3 is a very large program it cannot run on the Cybers at Fermilab as can EGS3. GEANT3 runs on the VAX and is available on the Fermilab VAX Cluster in directory LIB:[LIB.GEANT3] .

In terms of ease of learning and programming between EGS and GEANT3, EGS is preferred by the authors for simple geometries since EGS is much more simple and straightforward than GEANT3 and has very little overhead to deal with. However, for the complex geometry of a real SSC detector GEANT3 is the system of choice because its powerful geometry handling structure greatly eases setting up and simulating a complex detector. In that case spending the time to learn GEANT3 would be worthwhile.

Applicability of EGS at SSC energies

The most important factor to consider in the simulation of SSC showers in the CPU time per event. The time required to simulate electromagnetic showers with EGS (or GEANT3) in a typical SSC event at $\sqrt{s} = 40$ TeV is proportional to the total energy going into photons from π^0 decay. Since this energy is roughly $\frac{1}{3} \times 40$ TeV $\cong$ 13 TeV (neglecting acceptances), the time required for an EGS3 simulation of a typical SSC event would be approximately 6500 Cyber 175 CPU seconds. This is prohibitively slow. GEANT3 might optimistically run only a factor of two faster. This time could also go up or down by an order of magnitude depending on the detector geometry used, the presence of a magnetic field, the cutoffs used, etc.

So although EGS and GEANT3 are powerful simulators, it appears that the use of either in a brute force Monte Carlo simulation of SSC events is out of the question because of the CPU time involved. Most of the electromagnetic showers in SSC Monte Carlos will need to be simulated by some other means if they are to be done within a reasonable time. An approach commonly used, for example in CDF,[6] is to develop a set of empirical formulae or lookup tables which can be processed quickly yet which accurately represent electromagnetic shower development and its fluctuations. Another approach could be a hybrid one; use EGS or GEANT3 to develop the shower in the early part of the detector where fluctuations are most important and where the simulation runs quickly, then use empirical formulae or tables (with information from the EGS or GEANT3 part) to complete the shower simulation quickly. EGS or GEANT3 will of course still be needed in order to develop these formulae and to test their accuracy. Hardware developments, such as vector or parallel processing, may make this CPU time question less critical by being able to increase the sheer computing power that can be applied to this problem. Even then however the efficiency and speed of the simulation cannot be neglected.

As powerful as EGS and GEANT3 are, there are several electromagnetic processes, important for the SSC, that they both do not simulate at present. EGS or GEANT3 do not simulate synchrotron radiation from electrons or positrons moving in a magnetic field and they do not simulate Cherenkov radiation or transition radiation from electrons or positrons traversing various media or interfaces. These processes are crucial to some parts of the SSC detector and presently must be simulated by independent means. It would be desirable for future incarnations of EGS or EGS-like programs to incorporate these additional processes as options into their code.

Conclusions

EGS and GEANT3 are extremely useful tools and will continue to be so in the future. However effort must begin on developing an SSC electromagnetic simulation package which can run much more quickly than do present simulators, yet still give accurate results. It should be emphasized that the amount of time required to develop the eventual simulation system for the SSC will undoubtedly be considerable given the experience of developing the present large simulation programs. Development of a system for the SSC using the present generation of simulation programs as a starting point should begin soon with some central coordination and preferably with involvement from CERN. Use of the latest software concepts and systems, along with the latest hardware ideas, should be investigated and incorporated in general form into the SSC simulation program. New ideas on the fast simulation of high energy showers should be actively pursued.

Acknowledgements

We would like to thank P. LeBrun for valuable discussions during the 1986 Snowmass Summer Study. We thank W. R. Nelson for his help in our understanding of EGS. This work was supported in part by the U.S. Department of Energy under contract DE-ASO5-85ER40-217.

References

1. R. L. Ford & W. R. Nelson, SLAC-210, UC-32 (June 1978).
2. W. R. Nelson, H. Hirayama, & D. W. O. Rogers, SLAC-265, UC-32 (December 1985).
3. H. Hirayama, NIM **220**, 327 (1984).
4. R. Brun, *et al.*, DD/EE/84-1 (CERN 1986).
5. R. Brun, *et al.*, DD/85/1 (CERN 1985).
6. A. Beretvas, private communication, 1986.