ANL-HEP-CP-86-126
CONF-8606215--33
ANL-HEP-CP--86-126
DE87 004734

REPORT OF THE WORKING GROUP ON DETECTOR SIMULATION

Lawrence E. Price
Argonne National Laboratory, Argonne, IL 60439

Paul Lebrun
Fermi National Accelerator Laboratory, Batavia, IL 60510

G. Alverson	Northeastern Univ.	T. Oshugi	Hiroshima Univ.
Y. Arai	KEK	F. Paige	BNL
A. Beretvas	Rutgers Univ.	A. Savoy-Navarro	Saclay
R. Blair	ANL	P. Shephard	Univ. Pittsburgh
B. Blumenfeld	Johns Hopkins Univ.	T. Sjöstrand	Univ. Lund
D. Cassel	Cornell Univ.	J. Slaughter	Yale Univ.
L. Cremaldi	Univ. Colorado	D. Smith	Univ. California Riverside
Y. Fukui	FNAL	W. Smith	Columbia Univ.
T. Gottschalk	Caltech	H. Spinka	ANL
G. Hanson	SLAC	G. Trilling	LBL
K. Heller	Univ. Minnesota	D. Underwood	ANL
D. Judd	Florida A&M Univ.	F. Verbeure	Univ. of Antwerp
A. Kanofsky	Lehigh Univ.	R. Wagner	ANL
T. Kondo	KEK	D. Wagoner	Florida A&M
S. Linn	Florida State Univ.	E. Wang	LBL
J. Morfin	FNAL	A. Yamashita	Univ. Tsukuba/FNAL

Summary

An ad hoc group at Snowmass reviewed the need for detector simulation to support detectors at the SSC. This report first reviews currently available programs for detector simulation, both those written for single specific detectors and those aimed at general utility. It then considers the requirements for detector simulation for the SSC, with particular attention to enhancements that are needed relative to present programs. Finally, a list of recommendations is given.

Outline

I. Introduction

At this and previous Snowmass summer studies and at other workshops aimed at utilization of the SSC, much work has been done to generate simulated events and to use them to investigate the possibility of extracting specific pieces of physics from the very complex events that are expected. Great care and ingenuity has been exercised by the authors of ISAJET, PYTHIA, etc. to make the best extrapolations possible to the SSC energy regime. The resulting events are then used to determine not only the relative strength of signal and background for a given process, but also resolutions on p_T^{miss} due to neutrinos, for example, and therefore the ability of various cuts to enhance signal relative to background.

In almost all such studies to date, considerably less attention has been paid to the effect that the detector has on knowledge of the event. It is common to put in a hadronic calorimeter resolution of $50\%/\sqrt{(E)}$, less common to put in cracks between detector modules and very uncommon to note that the cracks are probably filled with material different from the calorimeter and to continue to simulate showers in the crack material. It is common to put in an ideal momentum resolution for charged tracks based on the magnetic field and an average tracking spatial resolution, less common to include the decreased resolution and lost tracks that come from broken or inefficient channels and very uncommon to generate hits in realistic tracking cells and to find and fit tracks based on those hits. Such aspects of realistic detector performance, however, are often at least as important to the extraction of physics from events as the intrinsic resolutions simulated by the physics-based event generators.

The major reason for the lack of effort on the treatment of events by detectors is the lack of standard programs that can be used to simulate detector effects with the same ease of use that has been achieved by the physics Monte Carlo programs. And the programs are either lacking or present considerable difficulty of use mainly because of the great degree of variability that is possible in detectors. Indeed, one of the principal uses of a detector simulation program would be to vary the geometry and type of detector subsystems to determine

Work supported in part by the U.S. Department of Energy, Division of High Energy Physics, under contract W-31-109-ENG-38.

the optimum arrangement for producing maximum physics output while keeping the detector in the possible range for constructability and cost. Another obstacle is the considerable amount of effort necessary to analyze realistically simulated events. A drift chamber simulation, for example, requires a good track finding/fitting routine to evaluate its design. Such capabilities are rarely in place even when an experiment starts taking data.

It can be claimed that a detector simulation program keeps track of as much physics as a "physics" Monte Carlo program in order to simulate properly the large number of physical processes that govern the passage of particles through the detector. In addition, a detector simulation program should allow flexible (and easy) specification of a wide variety of geometries, responses, and imperfections of detector elements. This high degree of variability makes a generally usable detector simulation program much more complex than the corresponding physics simulation program. In the latter case, a complex variety of physical processes must be simulated, but variability can generally be restricted to a relatively small number of numeric parameters.

Although no subgroup on detector simulation had been included in the initial planning for Snowmass '86, several participants in the summer study felt that substantial work would be necessary to ensure the availability of suitable detector simulation tools for design of SSC detectors and that the summer study was the appropriate place to begin planning. The result was a self-organized subgroup that (a) reviewed the present situation; (b) began to use current programs to simulate the detector components under discussion at Snowmass; and (c) developed recommendations for providing detector simulation for the SSC.

The results (so far) of this work are presented in this report. Section II discusses presently available programs. Section III presents the considerations we feel are important for providing detector simulation for the SSC, with some emphasis on necessary improvements over present packages. The prospects for new hardware solutions to provide the large amount of computing are discussed in Section IV. Finally, a summary of our recommendations is given in Section V.

II. Current Approaches to Detector Simulation

Of course a large amount of work has already been done in the area of detector simulation. With perhaps only one exception, detector simulation programs have been aimed either at general but limited problems such as showers (electromagnetic or hadronic) or at the simulation of a specific detector which may already be built and about which detailed knowledge is needed to calculate acceptances, efficiencies, and resolutions with which to extract physics from the data. For the former class of programs, it is generally left to the user to code from scratch the routines needed for the specific geometry of interest. In the latter class, the geometry is specifically coded in the program and only small amounts of variability are allowed for.

II.A Detector-Specific Simulators

As examples of simulation programs written for specific detectors, we include brief descriptions of the simulators for CDF and UA1.

CDF. The CDF simulation program takes input tracks from either the ISAJET or Lund JETSET Monte Carlos. Each particle is individually traced through regions of space identified with the different detector components. Each detector component (vertex TPC, central drift chamber, calorimeters, ...) is divided into subvolumes with well defined (and uniform) properties such as material composition, radiation length, density, etc. Heavy use is made of the YBOS memory manager both to implement a data base to describe the separate components of the detector and to organize the output digitizings from each part of the detector.

The data base contains a complete description of both the geometry and the signals generated by each volume (and subvolume) of the detector. The descriptions are references to standard types of geometry and detector types so that detector components can be modified and added rather flexibly. On the other hand, the menus from which elements are chosen are structured specifically to describe CDF and could not be used for another detector without extensive modification.

As the particle is tracked, the information in the data base is used to simulate the usual physical processes suffered by propagating particles: decay, dE/dx, multiple scattering, radiation, conversion, showering, etc.

Calorimeter simulation, potentially the most time-consuming part of a detector simulation program, is handled in the CDF program purely by parametrization.[1] Traversed radiation lengths are summed until the randomly pre-chosen depth at which the shower begins. The centroid of the shower is then tracked effectively as a fictitious neutral particle and a parametrization tuned to fit test beam data is used to distribute the energy of the shower about the path of the centroid. Longitudinal and transverse development of the showers are treated, so that energy sharing between adjacent pads is realistic, as well as the energy deposition in calorimeters with longitudinal segmentation. The effects of cracks, dead areas, finite calorimeter thickness, and fluctuations in shower development are also simulated.

Because its structure is aimed only at simulating one detector and because of the shower parametrization, the CDF simulator is rather fast, simulating an average TeV I event in 25 sec on a VAX 11/780.

UA1. The UA1 Monte-Carlo takes tracks from the ISAJET or EUROJET event generator and passes them through a detector simulation program. The detector simulation program does not use EGS or GEANT in the simulation of electromagnetic and hadronic showers. Instead, a response function or electromagnetic and hadronic showers is used to describe the shower activity in the calorimeters. Although the shower simulation is peculiar to the UA1 experiment, it significantly increases the speed with which the Monte-Carlo simulations are done. The typical SPS collider event takes 30-60 sec. (VAX 11/780 equiv.) to simulate in the UA1 detector.[2]

II.B An Example of a General Purpose Monte Carlo – GEANT3

GEANT3[3] is a general framework for HEP Monte-Carlo calculation. It is a system of detector description and simulation tools. The user can define his geometry using elementary shapes organized in a hierarchical tree structure. He then associates physical properties with the shapes such as magnetic fields and materials. Particles are traced step by step through the detector. The size and outcome of each step is determined by the geometry and the relevant physics processes such as decay, dE/dx, multiple scattering, interaction, bremsstralung, etc. Processes can be turned on or off. There is a framework for and some tools for simulating the detector response as actual digitizations. There are also tools for writing and reading back the data structures so that the simulation can be done in stages.

The "kinematics banks" (the particle track input to the detector simulation) are linked to numerous event generator programs (ISAJET, LUND,...). The electromagnetic shower response in a wide variety of materials has been successfully compared to data and, as discussed below, to EGS. Also, GEANT3 is linked to GHEISHA, a sophisticated hadron shower simulation program.

There is an interactive graphics package for displaying the detector geometry, the particle tracks and the digitizations. The graphics package is linked to the GKS standard, allowing a wide variety of terminal support. Advanced 3D graphics have been produced within GEANT3 using the PIONS system (CERN, UA1) and DI3000 (FERMILAB).

An example of the geometry and graphics capabilities of GEANT3 is given in Fig. 1, which shows an event (generated by ISAJET) in a vertex detector similar to that considered by the working group at Snowmass. The event is a $pp \to t\bar{t}$ with p_T = 500 GeV/c. The vertex detector simulation allows the number of silicon layers, their sizes and their positions to be entered interactively by the user.

The Central Tracking group at Snowmass also used GEANT3 to simulate the detector they were considering. A graphics rendition can be found in the Central Tracking section of these proceedings.[4]

The GEANT3 package has been under development at CERN for a number of years. It has been successfully used for such diverse experimental needs as those of LEP (OPAL,..), hadron colliders (DO, FERMILAB), and the fixed target program at CERN and FERMILAB (E706, E705, E687,..). Because so many groups collaborated to produce the GEANT framework, GEANT3 has become one of the most complete and powerful HEP Monte-Carlo packages. Its advantages are:

1. It has a well-defined data structure and modular architecture which make it very general and very powerful. It is easy to write out and read back geometries, constants and events at various stages. For example one can trace events, getting initially only the location of tracks, write them out, and then try different digitization schemes in subsequent runs.

2. Most relevant physical processes have been provided for and it is easy to add new ones, with the exceptions noted below.

3. It is fairly easy to define the geometry of a detector in a consistent way. It is certainly the most ambitious attempt to date to do this within the HEP community.

4. It has a committed group maintaining it. The existence of a complete manual and a common language are essential in collaborative efforts.

5. It uses graphics to help develope and debug the user's implementation.

GEANT3 has a number of disadvantages, however:

1. Its very generality and comprehensiveness mean that there is a significant learning curve. Once the basic ideas are mastered it is easy to do simple calculations with different geometries. Getting started, however, can be daunting. This is partly a documentation problem; the manual needs a tutorial and simple examples. A fair

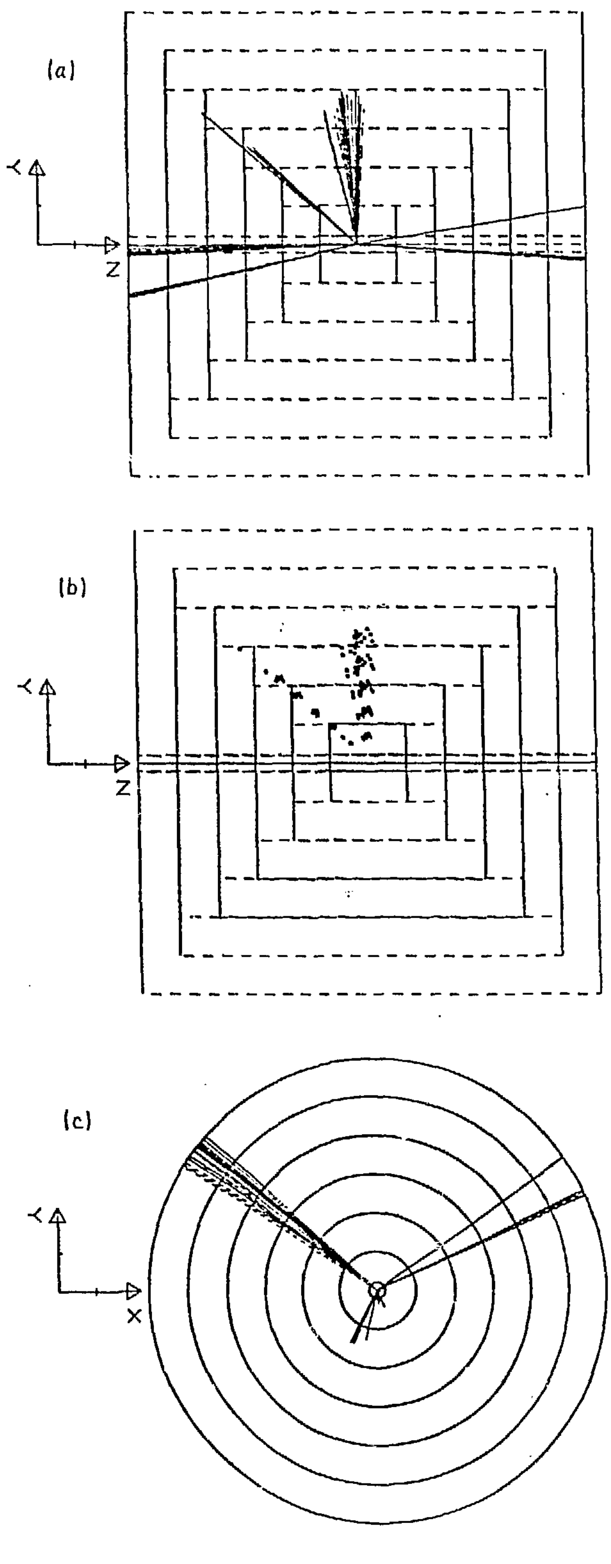

Fig. 1. GEANT3 graphics presentation of $pp \to t\bar{t}$ event in a 5-layer vertex detector at the SSC ($\sqrt{s}$ = 40 TeV). (a) Side view of tracks, (b) side view of hits, (c) end view of tracks. The tick marks on the axes are at 5 cm.

amount of sophistication is necessary to define the parameters of the physical processes to be appropriate for the task at hand. GEANT3 can be VERY slow if the geometry is carelessly defined or the optimization tools ignored or if unnecessary detail is included.

2. The geometry building tools are good but they need to be better. The definition of the detector needs real CAD and/or 3-D solid modeling tools. Also it would be useful (although obviously challenging) to have available generic parameterized detectors as well as elementary geometrical shapes as building blocks.

3. Its very generality and comprehensiveness mean that GEANT3 will always be slower for a given detector than a carefully coded program with a fixed geometry.

4. Not all physical processes are included or easy to add. In particular it is not easy to include Cherenkov detectors or TRD's because they require additional parameters in the data base used to characterize materials.

5. It is not easy to select just the necessary subset of all the available functionality.

Loading and linking such a large program can be tedious at best. This is due in part to its code management system, PATCHY, which is clearly outdated.

II.C Problem Specific General Monte Carlos

II.C.1 Electromagnetic Shower Simulations

EGS. The EGS code system[5,6] developed at SLAC has become the standard program for the simulation of electromagnetic showers. EGS (Electron-Gamma Shower) is an analog Monte Carlo program written to simulate three-dimensional electromagnetic showers in any mixture of 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 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 non-specific. 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.)

Timing and Size Considerations of EGS. Since EGS is an analog Monte Carlo program, 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 3.5 VAX 780 CPU seconds/GeV.

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 be careful to try several test runs with different cutoffs to find the optimum tradeoff of speed vs. precision of the simulation.

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.

The latest version (EGS4) also includes an option to run with importance sampling. Using this option can speed up the simulation by factors of 100 to 300.[7] 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. As described above, GEANT3[3] is a general detector simulation package developed at CERN. It contains code to build a detector geometry out of standard shapes and to simulate essentially all physics processes affecting the passage of particles through the matter of a detector. In this section we concentrate on its electromagnetic shower simulation. 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,[8] 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.[8] 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.

In comparing ease of learning and programming between EGS and GEANT3, EGS is generally preferred for simple geometries since it is much more simple and straight-forward 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.

II.C.2 Hadronic Shower Simulations

For the detailed simulation of hadronic interactions there are two programs widely accepted for use: High Energy Transport Code (HETC) by A. Gabriel[9] and Gamma Hadron Electron Interaction SHower code (GHEISHA) by H. Fesefeldt[10]. Both programs can be run in a stand alone mode that includes a geometry package and an EGS interface for electromagnetic showers. Both programs rely heavily on the available data; in some cases parametric models are used to describe nuclear phenomena where the physics is not well understood.

HETC has been very successful in modeling calorimeters and was used to understand the "compensation" mechanism in Uranium-liquid Argon calorimeters[11]. It is a stand-alone program with its own combinatorial geometry package[12] which is capable of constructing a wide variety of shapes. At this time it is being rewritten to better simulate hadronic showers at SSC energies.

The main features of GHEISHA can be summarized as follows. It can handle all stable or weakly decaying particles including strange baryons. It handles all elements, compounds, and mixtures. It can run as a stand-alone program using EGS for electromagnetic showers or as an option in GEANT3. Hadronic interactions in nuclei are treated as interactions on free nucleons and by an intranuclear cascade model which has one free parameter. Nuclear fission and evaporation are included. Finally, GHEISHA has been carefully compared with the data from many calorimeters[13] and in general the agreement is quite good, though not perfect.

Before these programs can be used to reliably predict predict what will happen at SSC energies, much theoretical and experimental work needs to be done. The energy range that will be of interest will exceed what is possible to obtain at Fermilab. Multi-TeV test beams will be necessary for comparison and input to these programs. At these extreme energies it may be necessary to have theoretical input about the formation of quark-gluon plasmas or other exotic phenomena if they are shown to exist at lower energies.

III. Issues in Detector Simulation Needs for the SSC

The detector simulation group assumed two related tasks at Snowmass '86. One was to provide on short notice some semblance of simulation tools for the use of other groups. The other was to discuss future needs, short term and long term. The material is this section comes both from the discussions and the experience in trying to use GEANT3 and CDFSIM in the Snowmass environment. It is only a start at specifying the strategy for SSC detector simulation.

III.A Program Strategy, Data Structures and Code Management

In the past 10 years software has become a major, often hidden cost, in HEP experiments. Some of the issues in planning and managing large collaborative coding efforts were addressed in the SSC Data Acquisition Workshop.[14] The fundamental considerations are (1) a good data structure, including the ability to read and write that data structure so that it can communicate with a variety of programs, and (2) a program organization and a code management system that will allow flexible configurations depending on the level of detail needed for a particular question. Also the data structure and the program organization have to provide for two very different kinds of users: the experts and the casual users. As mentioned above the heavy overhead associated with GEANT3 is one of its drawbacks. These points are not unique to SSC software: the HEP community faces a real challenge, either to pool resources and produce the needed tools (e.g. CERN's ZEBRA and HISTORIAN) and/or to negotiate licensing agreements on a collaboration-wide (or discipline-wide) basis in order to use commercial products.

III.B User Interface

By user interface, we mean the way in which the user, (particularly the causal user) interacts with an executing program. The GEANT3 provisions are a good start at such an interface. One difficulty in developing these interfaces is the "lowest common denominator" of terminal that is supported. Specific considerations are:

1. The software should be "menu driven" as much as possible.

2. There should be a really excellent graphics based

method to produce detector volumes for the simulation programs. A user should be able to easily specify the size, shape, segmentation, etc. and save the specific design in a detector library. There needs to be a way to specify an attribute for the geometrical volumes that can later be used to change the complexity or level of detail for a particular simulation.

III.C Interfacing to Event Generators

It has been noted above that physics Monte Carlo programs applicable to the SSC are already well developed. Several of these programs currently coexist (ISAJET, PYTHIA, FIELDAJET, GOTTSCHALK, EUROJET) because of the unique strengths possessed by each. It is appropriate that facilities be developed to allow each of these programs, the choice being made ideally by the user for each run, to serve as input of particle tracks (event generator) to the detector simulation package. Such a situation allows the physics Monte Carlos to be maintained by their original authors and interchanged as appropriate, instead of being hardwired into the detector simulation program. In this section, we present the strategies that are needed to allow this flexibility.

It is strongly suggested that such event generators run as the "slave" of another main program; this main program may simply consist of calling the event generator routine, but also might very well be the complete Monte-Carlo. This configuration allows, for instance, the user to generate more then one collision in a given experimental event, without subroutine reorganizations.

Since the CPU requirement for generating a statistically large number of events might be sizeable, an I/O system must be provided, in order to store/fetch the events to/from mass storage. Such an I/O system must produce a compact data file, and be reasonably fast. It is also recommended that this package be linked to the data structure manager, in order to transfer not only the data, but also the structure of that data. This is particularly important for the decay scheme of particles. This linkage has been achieved in the ISAZEB (ISAJET/ZEBRA) package.

It is also important that all relevant information concerning the decays of particles be transferred, even if it is not clear whether the detector can respond to a particular aspect of the decay. For instance, the π^0 life time will, no doubt, be difficult to measure at the SSC as it is at lower energy accelerators, so that one might be tempted to store only the two photons' momentum and not the π^0 itself. But it may be that the experimentalist will be able to distinguish between two photons from a π^0 decay and two randomly chosen photons. Thus, in order to compare detector response to the Monte-Carlo "truth", this information must be available to the experimentalist.

In order to avoid unnecessary conversion, it is desirable to introduced a standard particle numbering scheme to be used by all event generators, accepted by theorists as well as experimentalists. Although superstring theories predict an infinity of elementary particles, a 32-bit number should be adequate to unambiguously define a particle. One of the best models available for such a numbering scheme can be found in ISAJET.

III.D Geometry Specification

The geometry of a typical SSC detector can be extremely complex, involving hundreds of elementary volumes in which particles must be traced. Consequently a large data base containing volume boundaries and characteristics must be created. Such a data structure is fairly complex because, by the nature of the particle tracing problem, the corresponding data base has both RELATIONAL and HIERARCHICAL aspects. Once these requirements are understood, the design of a Monte Carlo geometry data base becomes more feasible.

The relational aspect involves the storage and retrieval of all attributes, or characteristics of a particular volume. Indeed, the geometry information could be stored in a big table, one entry in such a table being an elementary volume. These attributes can be tentatively classified in the following way:

- Dimensions, or boundary locations, and physical shape.
- Graphics attributes: visibility, color, line shape, ...
- Tracking attributes: required spatial accuracy, steps, ...
- Physical parameters: radiation/absorption length, magnetic field.

The volume location data may come from a rough estimate in the first modeling stage but, later, must be able to come from the exact survey of the apparatus. Survey numbers and engineering blueprints are rarely expressed in a RDBMS form[15], leading often to confusion and inaccuracy in the Monte-Carlo.

The physical shape of a volume may be specified as "constant", "fixed", or "programmable". In the first case, each volume shape is entirely described by boundary plane locations; the only available shapes are rectangular parallelepipeds and cylinders. If other shapes are needed, it is necessary for the user to code explicitly for each new volumes the boundary search routine used in the tracing process.

In the second case, the user is allowed to choose among a limited number of volume shapes, the so-called "system shapes". GEANT3, for example, has 13 elementary shape, ranging from the simple box to the polygon.

Introducing "programmable" shapes allows full solid modeling, where volumes are constructed not only from very simple primitives, but also from a set of rules governing the volume intersections, insertions, edges definitions and so forth. Such solid modeling techniques are extensively used in computer graphics, and engineering calculation of material properties (heat propagation, resistance, etc.)[16].

Note that such solid modeling programs use data structures which are hierarchical rather then relational in order to have a geometry which is comprehensible and manageable, where the relative positioning of an elementary volume is specified with respect to a "mother volume" and not within a single global coordinate system. Thus, while building the geometry, the user builds a "geometrical tree", where volumes have a hierarchy, starting from the primary volume, branching off to a subvolume and ending at the definition of an elementary cell of the detector.

Volume representation is one problem; particle tracing is another. This latter task is the core of the Monte-Carlo, and therefore must be extremely well optimized for speed. This optimization leads one to a more subtle hierarchical volume tree where the algorithm loops over neighbors, and establishes if a particular neighbor is the next volume where the particle will be propagated. The order in which such neighbors are considered is crucial to the optimization of such an algorithm. This ordering leads to the need for a second volume hierarchy which is not usually identical to the volume solid modeling one, since it depends on the geometry with respect to the most probable path of particle in the detector. In an SSC detector, for instance, starting from the microstrip vertex detector, the next probable volume along the particle path is within the central tracker, but the micro-vertex volume and the volume within the central tracker are probably located in two different parallel branches in the solid modeling tree.

The data structures used by the program must be general enough and flexible enough to allow easy transition from a global description of the detector with a small amount of detail to a highly detailed one with all relevant tracking process parameters being included correctly.

A full three-dimensional solid modeling Monte Carlo program for HEP is certainly desirable, but will be very expensive, because of the inherent complexity and size of such programs. Also, solid modeling techniques are far from well established, and are relatively new in computing science. It is unclear to what extent the existing programs can be applied to HEP Monte Carlos. In Fig. 2, we show an approximation of the central tracking detector studied at Snowmass '86. There are four tracking modules and a vertex detector covering the central region. It was constructed with the help of a CAD system in a few hours time. Tracks were manually inserted for highlight. By interfacing such a system into a physics package, a powerful design tool could be obtained.

III.E Graphics

For many years, graphics capability has often been considered as a luxury in HEP programs. But, as the size and complexity of the detectors increase, graphics - especially with 3D capability - becomes a mandatory tool to allow easy and quick understanding of detector design issues.

A graphics package deals essentially with the 3 basic elements of any detector simulation Monte-Carlo: (i) the detector itself; (ii) the particle paths through this detector; and (iii) the graphical

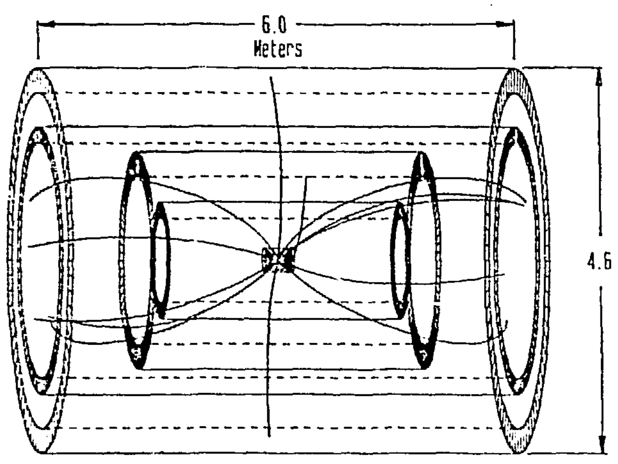

Fig. 2. Three-dimensional presentation of vertex and central tracking detectors generated by a CAD system.

representation of the interaction of the particles with the detector. The detector, its geometry and volume organization, can be thoroughly checked against the geometry data base graphically. Track drawing is a powerful tool for learning quickly about event topologies. The graphic representation of the "hits" is in fact an alternate event display to the online packages, and helps in designing the event reconstruction analysis package.

It is also true that the interactive user menu and the histogram plotting package are important components. But these items are less specific to Monte-Carlo programs, and can be picked up from the other analysis software. An example can be drawn from the CDF user interface package[17].

Given the intrinsic 3-dimensional nature of the detector simulation problem, good 3D capability is required. The user must be able to see the detector from arbitrary views, and to zoom, translate, rotate the image in a 3-dimensional world. We recall the important role of the Megatek workstation in extracting the W and Z events from the UA1 data.

Graphics attributes such as color, line style, and polygon fill are also extremely useful. But even more important is the adequate resolution of the terminal or hard copy. In order to display events with hundreds of tracks, the availability of 1K by 1K pixels is not at all a luxury. Some of us have recently been impressed by the capability of the Evans and Sutherland PS300 workstation, a 3D display device with a resolution equivalent to 8K by 8K pixels.

On the software side, it seems evident that a necessary condition for modern graphics is an international standard. We see that many problems could arise if the present situation continues. CERN is committed to GKS (Graphics Kernel Standard), originally a 2D system, but now being extended to 3D. Unfortunately, this extension is not fully standard yet, and graphics software vendors are reluctant to invest in GKS-3D. Therefore, at Fermilab, both D0 and CDF, followed by many fixed target experiments, are using DI3000, a commercial package which follows the CORE standard, and offers better 3D capability and terminal support than GKS. SLAC is still using it own Unified Graphics system. To conclude, standards are certainly useful, but the graphic community apparently has a hard time living within only one of them partly for commercial reasons, partly because of the tremendous growth rate of and varied demands on computer graphics.

III.F Detector Systems

III.F.1 Tracking

The simulation should contain a package of fundamental physics processes. They should be easily switched off and on in each detector volume. Some of the standard processes which limit resolution and pattern recognition in tracking devices are ionization energy loss, Coulomb scattering, photon conversions, delta-ray production, secondary interactions and decays. In addition, bremsstrahlung, synchrotron radiation, and Compton scattering should be included for use where relevant to detectors.

The charged particles in each event should be propagated through arbitrary electric and magnetic fields. In the simplest case, these would be constant fields, and a fast field swimmer could be used. Interest in fringe field effects will force a full field swim. Step sizes will, of course, be of interest and should be left to user control. The swim should produce a bank of detector hits to be digitized if selected.

Event digitization should occur under some standard format. This will be closely tied to detector geometry. Simplicity dictates that subpackages exist to fill volumes of drift cells of SQUARE, HEX, or STRAW design, scintillating fibers, or silicon microstrip detectors. Stereo angles should be easily specified, and digitization automated. At the digitization stage, the user should have the option to include dead detector channels, noise, crosstalk, and other such effects. It should also be possible to merge events for signal pile-up studies.

The package could contain models for hit sharing in silicon microstrip detectors, pile-up, space charge effects, and resolution smearing in drift cells. Other subpackages that might be useful are models of charge division, cathode strip readouts, signal waveforms, transition radiation detectors, and ring imaging Cerenkov detectors. Signal pile-up effects are best studied by waveform analysis; TRD's and RICH's are mentioned due to their close integration with the tracking detectors.

Effective use of tracking devices, more than most other detector components, requires detailed pattern recognition to organize hits into tracks. For this reason it is particularly true of tracking that the detector simulation work must be closely coupled with development of the software that will analyze the output of the tracking detectors.

III.F.2 Calorimetry

The simulation of electromagnetic and hadronic showers is the most time consuming part of any realistic detector simulation. Typical simulation times for showers in GEANT3[18] are about 30 VAX 780 seconds/GeV of energy deposited for electrons and about 1/3 as much for hadrons. This means roughly 1 event/day for a 'typical' SSC event. As seen in Fig. 3, a program like GEANT must trace all secondary particles created by the showering process, which number in the thousands, through the geometry while

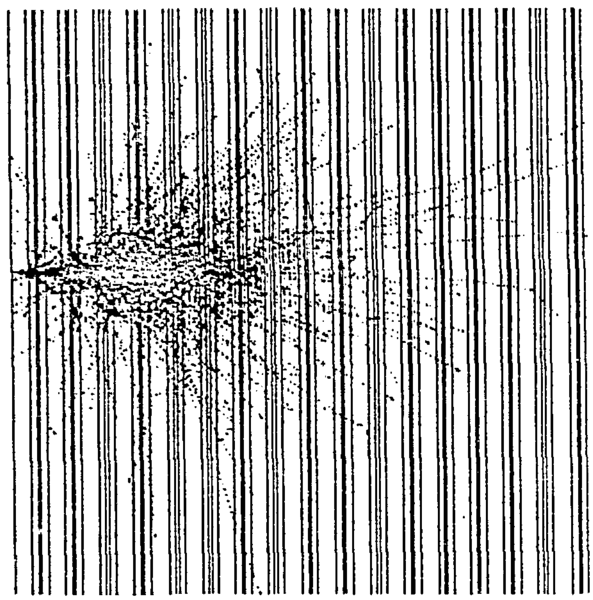

Fig. 3. Sensitive region crossings for a 10-GeV electron shower generated by GEANT3 in a uranium/liquid argon calorimeter with 6 mm uranium plates, 1.6 mm G-10, and 2.3 mm liquid argon gaps.

doing the physics of each particle. The amount of time necessary for these computations is excessively large. Ways to do these computations using hardware are discussed in another section; here we will discuss algorithmic means for reducing the computation time.

All approaches we will discuss in some way degrade the quality of information the simulation will provide. It is up to the individual to determine how detailed a description is necessary to answer the questions asked. Understanding the response of a new calorimeter may indeed require inclusion of the effect of 1 MeV photons; however, a calculation of trigger rates would not use this level of detail. We include a list of techniques that can be used to speed program execution with different effects on accuracy.

a. **Parametrizations**

The main advantage of parametric techniques is that in general they are about 100 times faster than detailed tracing of individual particles. The philosophy is to represent the energy deposition of a shower by an analytic expression for the energy density. The parameters describing the shower shape are usually derived from fits to test data and can describe the average properties of the shower very well[19,20]. To determine the energy deposited in a calorimeter cel', one integrates the energy density over the volume of the cells near the shower axis. In practice this integration is difficult because either the energy density or integration boundaries must be expressed in a coordinate system rotated with respect to the master reference system of the detector. Some form of numerical approximation must be employed. The shower shape is usually parametrized in radiation lengths and absorption lengths so that showers crossing boundaries between different materials can be handled in an approximate way. A shower developing near a crack or detector boundaries of vastly different materials (e.g., beam pipe) will be handled incorrectly. A major shortcoming is the ad-hoc way that fluctuations are incorporated[21]. In general these fluctuations differ greatly from test data. Recently some methods for creating realistic fluctuations have been developed[22,23]. These methods rely on a finely segmented calorimeter in both longitudinal and transverse dimensions to measure correlations. Usually test calorimeters are built with other purposes in mind and it is difficult to extract the information needed to reliably simulate showers in different materials and at a variety of incident angles. The previous criticism of behavior near boundaries also applies to these techniques.

For example, the FNAL E706 group[24] has developed a parametric strategy for speeding shower simulation. The experiment contains a combined liquid-argon electromagnetic and hadronic calorimeter and an iron-scintillator forward calorimeter. Their mechanism for parametrizing showers is based on the assumption that a shower can be described by some distribution of minimum ionizing tracks that cross the sensitive volumes in a detector. Shower development is simulated by populating the sensitive volumes with track segments, in accordance with some prescribed shower shape. Each detector then interprets these crossings as minimum ionizing particles, and digitizes the energies accordingly. These simulated track crossings are stored in the same manner as normal GEANT hits, and consequently the digitization of the event is independent of whether the shower was generated in the normal GEANT manner or simulated through the minimum-ionization mechanism.

b. **Frozen Showers**

This method uses a 'library' of previously simulated showers from mass storage. A natural variant could also make use of test beam data. The algorithm follows the multiplication to some threshold value then replaces the remainder by "frozen showers" chosen randomly from storage. In this way realistic fluctuations are obtained. This necessitates some interpolation and rescaling of the energy. This method seems to work best in homogeneous calorimeters where the angle of incidence does not change the resolution. Since the shower is generated in one medium, its spatial extent must be described by a material-independent scale such as the radiation length for EM showers. If a shower crosses from one medium to the next, the description will only be approximate.

A technique which combines this cutoff philosophy and parametrizations is used by DO. Electromagnetic and hadronic showers are allowed to multiply until electrons and photons are produced with energies below 200 MeV. These showers are then parametrized with a longitudinal distribution only. This speeds the program by factors of between 7-10 depending on the geometry. This parametrization has a detrimental effect on the transverse shape of EM showers. It is well known that the transverse shape of an EM shower can be described by the sum of two exponential distributions. The width of the central "core" is due to Coulomb scattering. The "tails" arise from minimum attenuation photons which can travel long distances before absorption. A cutoff at 200 MeV produces transverse distributions that do not have these tails. The effect of no tails is seen in the reconstruction of shower centroids where energy sharing between adjacent cells aids the position resolution considerably. There is very little effect seen in the shape of hadronic showers.

c. **Geometry Simplification**

It is trivial to deduce that more elements in a geometric description of a detector will cause searches of the representing data structure to take more time. Efficient use of hierarchy and optimal ordering of these elements can help to alleviate this overhead. Factors of two are easily gained by these techniques. An obvious way to cut this time is to reduce the number of elements by choosing an appropriate level of description. Very few geometers have attempted to describe threaded rods in a calorimeter and an equal fraction have not had the nerve to omit such objects altogether. The question again is whether omission of something will change the physics. The beauty of GEANT is that one can insert or omit levels of description easily. Given the potential amounts of CPU time necessary for SSC simulation, it is worthwhile to spend a fair amount of processing time determining what level is necessary.

We close this section with a description of a technique employed by the DO collaboration. DO is a sampling calorimeter made of liquid argon and uranium. The central calorimeter is constructed of trapezoidal modules which provide mechanical support for plates of uranium and G-10 readout boards. These modules are stacked in a cryostat which is filled with liquid argon. An initial design choice was to represent calorimeter modules by trapezia made of a homogeneous material with the correct average properties. While speeding the simulation, this change introduced two "features" that required correction:

- ULA has the property of an almost equal response to electrons and pions. The mechanism of this "compensation" is now a well understood result of suppressing electromagnetic showers. This suppression is due to proportionally more energy being deposited in Uranium than a simple calculation would indicate.
- If a sampling calorimeter is replaced by a homogeneous medium, the calorimeter will have no sampling fluctuations which depend on the square root of the absorber thickness as seen by the particle.

To compensate these effects it is necessary to suppress the electromagnetic signal a priori and also to smear the energy resolution. Figure 4 is a cut view of the DO calorimeter with simplified geometry.

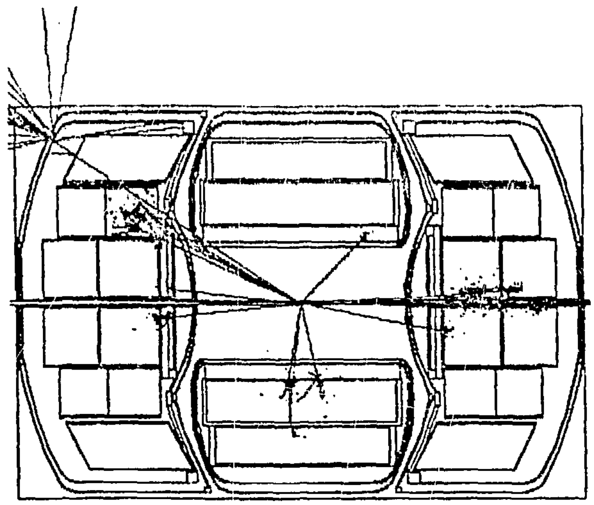

Fig. 4. A rare event in the DO calorimeters, showing several showers and hadronic punch-through.

d. **Cutoffs**

GEANT contains several parameters that allow the user to change the energy below which a particle is not traced. The default settings for these parameters are in general set at a few MeV. For detailed studies these parameters are usually set lower. It is not clear what the effect of increasing these parameters by an order of magnitude would do to the physics of an SSC event. The gain in speed is more than an order of magnitude. We recommend that studies be done to quantify these effects for the standard calorimetric materials available now and in the near future.

III.F.3 Muons

The importance of muons in a high energy search for new physics has been repeatedly stressed. Even at conventional energies, a high p_T muon tag has been instrumental in the discoveries made at the CERN Collider. At the SSC the efficient detection and precision measurement of high energy muons could lead to the discovery of (amongst other things):

1. new higher mass ($M < 6.5$ TeV) W and Z bosons via $W \to \mu\nu$ and $Z \to \mu\mu$;
2. parton substructure with compositeness scale $\Lambda \sim 20$ TeV as well as the existence of leptoquarks;
3. the elusive Higgs.

A complete discussion of the physics that could be performed with a high momentum lepton trigger at the SSC can be found in the Physics section and the Muon

Detector section of these proceedings.

To be able to competently design a muon detector for the SSC, a complete understanding of the phenomena of ultra-high energy muon interactions must be incorporated in a Monte Carlo format. The relative importance of the various processes which a muon undergoes

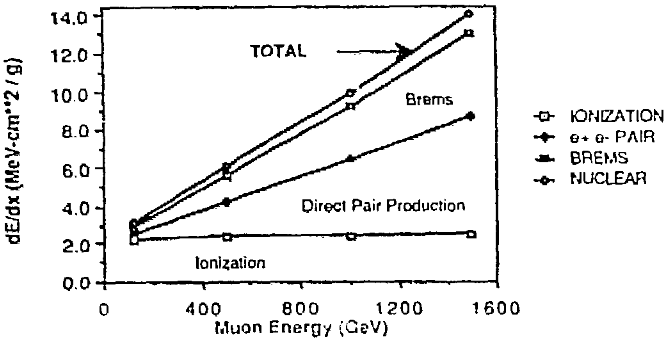

Fig. 5. Energy loss processes for muons in iron as a function of muon energy.

as it passes through matter is shown in Fig. 5. Above a critical energy (330 GeV in iron), the major form of energy loss is no longer ionization. Other processes, in particular direct pair production, become increasingly important with increasing muon energy. The energy loss spectra of all the processes can be found in Reference 25. The energy loss spectrum from pair production can be approximated as:

$$\frac{dN_p}{dxdE} \sim \frac{1}{l_p}\frac{1}{E_p} \qquad 2\,m_e < E_p < 0.1\,E_\mu$$

$$\sim \frac{1}{l_p}\frac{(.01\,E)^2}{E_p^3} \qquad .01\,E_\mu < E_p < E_\mu ,$$

where $l_p \sim 6$ meters in iron. Although the probability for a muon to lose much greater than 1% of its energy through a single loss in 6 m of Fe is less than .005, the probability that it will lose close to 1% of its energy is O(1)! For a 10-TeV muon, this implies a 100 GeV shower. For bremsstrahlung, the energy loss spectrum is:

$$\frac{dN_B}{dxdE_B} \sim \frac{1}{l_B}\frac{1}{E_B} \qquad 0 < E_B < E_\mu; \quad l_B \sim 450 \text{ m (iron)}$$

so that the probability of a single loss greater than $.01\,E_\mu$ is about 1% per meter of iron.

What this means for those attempting to develop a muon trigger at the SSC is that there are large energy electromagnetic showers, associated with the muon, propagating through the absorber. The scope of this problem can be illustrated[26] by the following example. If the average shower range from a directly-produced pair is 0.5 m of Fe, and a "trigger" requires that there be at least 4 clean detectors out of 10, each separated by 0.5 m of Fe, then <u>1 out of 5 good μ's will be lost.</u>

A second type of difficulty that will be encountered in an SSC muon-oriented detector is due to our (still) insufficient understanding of multiple scattering to reliably predict the tails. The danger is that a series of larger angle scatters could cause a low energy muon to be incorrectly measured as having higher energy. An associated problem is the necessity of including[27] the <u>angle-position correlations</u> in the multiple scattering calculations. Tail effects and second order contributions become important at SSC energies.

The most complete simulation program now available for tracking muons through an arbitrary detector is the CERN-based Monte Carlo GEANT3. Step by step multiple scattering is treated in the standard (and correct) way of transforming the particles direction and position into a local reference frame and then randomly generating, via a Gaussian distribution, two deviation angles in the relevant orthogonal projections. Moliere theory is than used to calculate the overall effect of multiple scattering on the particle trajectory. It is done thoroughly up to the point where it is stated[28] that "... the problem of joint angle lateral displacement in the Moliere approximation has not been solved ...". Obviously, effort could be well spent in trying to introduce angle position correlations into the calculations.

The cross section for direct pair production uses a parametrization of the explicit fourth-order QED expressions developed by L. Urban. The error introduced by the parametrization is given as < 8% for muon energy greater than 5 GeV. The applicable range of the parametrization is $E_\mu < 10$ TeV. The actual propagation of the shower uses an approximation that is based on the observations that the shape of the relevant functions are essentially independent of the atomic number of the material; the dominant contribution to the energy loss integral comes from low shower energies; and in this low energy region, the energy loss is flat as a function of the density of material.

Instead of using the explicit Bethe-Heitler formulae for the cross section of bremsstrahlung by muons, L. Urban has used an approximation which he claims is good to ~ 95% for muons of energy up to 10 TeV. The differential cross section for bremsstrahlung is used explicitly in generating the shower energy.

The last contribution is muon-nucleus interactions which is relatively small with respect to direct pair production and bremsstrahlung. The cross section is assumed to be growing at a rate,

$$\sigma = 0.3\,(E/30)^{0.25}\ \mu b \ .$$

The mechanism for generating the hadronic shower is taken directly from the GHEISHA[10,13] Monte Carlo which uses the rather crude approximation of replacing the virtual photon by a real pion of random charge and the same total energy. This may be a relatively safe approximation for calorimetric purposes, but it is certainly incorrect both for the explicit kinematics and particle content of the final shower.

III.F.4. <u>Note on Particle Identifiers</u>

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.

IV. Computer Hardware Requirements for SSC Detector Simulation

A. Needs

We will focus on the hardware necessary to achieve the needed CPU cycles for detector simulation at a large hadron collider. The other hardware aspects of simulations such as mass storage and networking have been discussed elsewhere[29-31]. Software is discussed only where it is hardware dependent. We start from the Report of the Task Force on Detector R&D for the SSC[29], where an estimate is given of the resources needed for "Monte Carlo proposals and code development". They estimate a need for 400 MIPS three years before turn-on increasing to 4000 MIP's at turn on.

An independent determination of the needs can be made using the experience of the DO group whose detector is three years from completion. The DO simulation which uses GEANT with a simplified geometry and a shower parametrization uses 12.0 VAX 780 seconds/GeV of energy deposited. This is about 2400 seconds for the process $W \rightarrow e\nu$. By the end of 1986 10^4 events will have been generated mainly for triggering studies. If we use 2400 seconds/event, it gives a total of 2.4 10^7 VAX 780 seconds. Other studies and code development increase this about to 1 MIPS for the year of 1986.

At the SSC about 6 TeV of energy is deposited in a detector when a 300 GeV Higgs decays to two W's[32]. At present simulation speeds this is 1200 minutes /event. A one year study requiring 10^5 events would use 240 MIPs. It has been estimated that analysis of one event will require 1200 seconds on a VAX 780[30]. If these events were analyzed 10 times during the course of code development, this would require an additional 38 MIPs.

Basically there will be two different requests from the users:

- An interactive system for development and debugging of programs as well as hosting workstations with graphics capabilities. This should be a very fast machine with a turnaround time on the order of minutes to hours.

- A large batch oriented system with a turnaround of hours to days.

If we use 4000 MIPS as what will be available, the best division between the two systems would be one that would optimize programming productivity. Let us assume that Monte Carlo events are generated on the large system and analysis code is developed on the fast machine. One hundred physicists interactively analyzing 1 event/minute would require a 2000 MIP machine. If the 100 physicists each had an 8 MIP workstation, then a 1.2 GIP machine would suffice. Trade-offs are possible here; because, a 2 GIP machine could run batch 3/4 of the time where as 100 workstations might not be as efficient.

Both facilities should be available at least six years before startup because the simulations must go hand-in-hand with the detector development.

B. Possible Solutions

Assuming that no drastic improvements are made on the software front, as might be realized by parametric techniques, the amount of CPU time required becomes enormous. One would like to find ways of achieving the goal other than operating a large number of big mainframe computers, e.g. 200 IBM 3081K's. We summarize the existing hardware developments:

- **Vectorization.** Work is in progress to make use of vector architecture in detector simulations. Portions of the GEANT3 geometry and ray tracing routines have been explicitly vectorized for the CDC Cyber 205. For a small number of widely used geometric configurations, it has been possible to gain a factor of five in speed enhancement between scalar and vector modes. The algorithm exploits the hierarchical structure of GEANT's geometry definition. If a detector has a large number of similar daughter volumes in a common mother volume, the computations of the smallest geometric step that can be taken in the mother before reaching any daughter can be executed in parallel. Most detectors have a great deal of symmetry, and are in general constructed from large numbers of simple shapes, e.g. tubes, boxes, and trapezoids. For the inevitable broken symmetries in a realistic detector, the use of artificial boundaries and stepwise approximations are under investigation by the DO collaboration.

The next generation of supercomputers, such as the ETA-10 have a peak computing power equivalent to 40,000 VAX 780's. Realistic rates are probably a hundred times slower.

Both DO and ALEPH collaborations are interested in using the FSU Cyber 205's and later the ETA-10 for a their Monte Carlo and off-line computations. Much development work is taking place and will continue in the near future. A class of tracking routines has been vectorized is being used for production processing of Fermilab E711 data. The processing time on the CYBER 205 is more than 300 times faster than a scalar program running on a VAX 780.

Very little work has been done on the question of using array processors for high energy physics. This could prove to be an efficient compromise between large and small systems if the code can be adapted.

- **Farms of small processors.** At CERN there is a project underway to use 3081/E emulators for parallel production of GEANT. The system uses an IBM 4361 as a host which controls I/O between nodes and host. Results of this undertaking should be available in the near future.[3]

The DO group is using a prototype of its trigger processor consisting of 15 μVAX-II nodes to run GEANT detector simulations. This system will eventually have 50 nodes. The present system has 5 nodes with 5Mb of memory and 10 nodes with 3Mb of memory. While 3Mb is sufficient for most events, 10-15% of all events simulated require 5Mb because of higher multiplicity. All control and communication is done in high level languages. The throughput of individual nodes is about 90% of a VAX 780.

The Fermilab Advanced Computer Project (ACP)[34] has produced a system which is now being operated by the Fermilab computing department. This system has a 140 node capacity to run batch processing under operator control. A 53-node system was used for production processing of Fermilab E691 data. Attempts are underway to run GEANT on the ACP.

C. Costs

We present here a table of performance/cost for some of the options discussed in the text. The numbers in most cases have been provided by the proponents of particular systems. The reader is

Table 1

Machine	Equivalent VAX 780/ M$	Reference
VAX 780	4	[34]
μVAX-II	143	[35]
ACP-II	420	[34]
ACP-III	500	[34]
Cyber 205	10-60	[33,36]
ETA-10	125-750	[33,36]
3081E	100	[30]

warned that reducing complex systems to one figure of merit can be a dangerous exercise. Cost can vary by as much as a factor of ten depending on the market and the configuration.

V. Recommendations

1. A standard detector simulation package should be developed for use at the SSC. The package can evolve as the detectors go through their stages of development. Thus conceptual design requires a package that emphasizes ease of use and flexibility of detector elements. In the technical specification stage, a package is needed with the ability to incorporate fine details but perhaps with less ease of modification. And finally for physics analysis, one wants the capability of precise specification, including as-built asymmetries and high accuracy in detector responses.

2. Urgent attention should be given to deciding whether to base this package on the existing GEANT3 or to start anew.

3. In order to allow both fairly casual use and flexible reconfiguration of the detector during the conceptual design phase, configurable versions of all common detector components should be provided that do not require each user to build the detector component from primitive geometric shapes and provide code to simulate detector responses.

4. For the writers of detector component packages, a very wide variety of geometric shapes should be available.

5. Effective use should be made of modern computing techniques that physicists usually do not fully use to their advantage. These include a high level user interface through a graphics oriented "user menu" and interactive CAD/CAM graphics techniques in producing the geometry of the detector.

6. It is likely that a fully functional detector simulation package will draw on commercial sources of software, particularly in the area of graphics. It is desirable that the High Energy Physics community agree on standards for graphics and other areas of software so that multiple interfaces to these commercial packages are not required. Arrangements with vendors for discipline-wide licensing of software would facilitate such standardization.

7. Considerable attention must be given to improved algorithms for simulation of showers:

   a. The physics of hadronic showers must be understood more thoroughly, so that hadronic shower simulations are reliable in a wide variety of materials without extensive tuning. This task will require considerable input from the Nuclear Physics community.

   b. Strategies need to be developed to allow shower simulation to proceed at a rate consistent with reasonable computing resources. It will be necessary to provide options that allow trading off speed with accuracy in all details, so that the conceptual designer can accept some degree of approximation in the interest of exploring many configurations, while the final physics analyst can have the ultimate accuracy that he needs (ideally) only once. Perhaps facilities can be provided so that the SSC physicist can quickly understand a few "exact" showers and then set up the corresponding model and run many more events without losing substantial accuracy. This is a non-trivial problem, particularly if it is made possible to move from a general purpose facility to a fast, economical, but rigid, "hardcoded" setup, without loss of information.

8. Standards should be developed for the output of physics Monte Carlo programs, so that several can act interchangeably as event generators for the detector simulation package.

9. Strategies should be developed for allowing the experts who contribute the physics and detector knowledge to be different individuals from those who contribute the computer savvy in coding and managing such a large package. Both areas need crucial improvements if realistic SCC simulations are to be computed. Because so many physicists and programmers have to contribute, advanced code management techniques are mandatory. But this management must not become a burden in getting the work done: for the casual user, it must appear flexible and easy to handle.

10. Since it is unlikely that cleverness in algorithms alone will solve the problem of the large amount of computing that will be necessary to support SSC detectors with adequate simulation packages, substantial effort should be directed to exploiting the new computer architectures for the purposes of detector simulation.

11. Appropriate resources must be provided, starting early in the SSC program, for detector simulation. These resources include both the access to large computers needed to run the programs and the substantial software effort required to provide the packages at the appropriate times. In the early phase, it would be appropriate to fund the development of a detector simulation package as part of the projected program of detector R&D for the SSC.

References

1. J. Freeman and A. Beretvas, these Proceedings.
2. D. Linglin, Proceedings of the Workshop on Triggering, Data Acquisition and Computing For High Energy/High Luminosity Hadron-Hadron Colliders, B. Cox, R. Fenner, and P. Hale eds., p. 338.
3. R. Brun, et al., DD/EE/84-1 (CERN 1986).
4. D. Cassel and G. Hanson, these Proceedings.
5. R. L. Ford and W. R. Nelson, SLAC-210, UC-32 (June 1978).
6. W. R. Nelson, H. Hirayama, and D. W. O. Rogers, SLAC-265, UC-32 (December 1985).
7. H. Hirayama, et al., NIM 220, 327 (1984).

8. R. Brun, et al., DD/85/1 (CERN 1985).
9. A. Gabriel, ORNL TM-9727, Oak Ridge National Lab, (1985).
10. H.C. Fesefeldt, PITHA-report 85-02, RWTH Aachen, FRG.
11. J. Brau and A. Gabriel, NIM A238 (1985) 498.
12. A. Gabriel, ORNL TM-5619, Oak Ridge National Lab, (1977).
13. H.C. Fesefeldt, PITHA-report 86-05, RWTH Aachen, FRG.
14. J. Appel, et al., Proceedings of the Workshop on Triggering, Data Acquisition and Computing For High Energy/High Luminosity Hadron-Hadron Colliders, B. Cox, R. Fenner, and P. Hale eds., p. 315
15. See for instance: R. McClathey, ALEPH-COMPO 85-3, CERN.
16. IEEE COMPUTER GRAPHICS, Vol. 5, No. 4, pps. 23-75.
17. D. R. Quarrie, "UIPACK", the CDF User Interface Package Programmers Manual, CDF note 372 (September 1985).
18. From the E706 calorimeter and a simulation of the D0 test calorimeter.
19. E. Longo and L. Luminari, I.N.F.N. Internal Note 831, (October 1984).
20. R.K. Bock et.al., Nucl. Instr. and Meth. 186, (1981) 533.
21. E. Longo and I. Sestili, Nucl. Instr. and Meth. 129 (1975) 283.
22. M. DePalma et.al., Nucl. Instr. and Meth. 219, (1984) 87.
23. M. DePalma and G. Iaselli, Nucl. Instr. and Meth. A235 (1985) 70.
24. G. Alverson, et al, "Investigation of Direct Photon Production at Fermilab," Proceedings of the 18th International Conference on Cosmic Rays, P.V. Ramana Murthy, ed. Also F. Lobkowicz, et al., "A Large Liquid Argon Photon/Hadron Calorimeter at FNAL," Nucl. Instr. and Meth. A235, 332 (1985).
25. Z. Zupancic, "Physical and Statistical Foundations of TeV Muon Spectroscopy and the NA4 Experience," Transparencies from the SSC/LHC Detection Workshop, University of Wisconsin, p. 37 (April 1985).
26. R. Adair et al., "Advanced Calculation of Radiative Effects for High Energy Muons in Matter," ibid., p. 144.
27. G. Todorova, ibid., p. 150.
28. M. S. Dixit; "Moliere Scattering," ibid., pg. Phys. 352-2.
29. Report of the Task Force on Detector R&D for the SSC, SSC-SR-1021, p. 238.
30. Proceedings of the 1984 Summer Study on the Design and Utilization of the SSC, R. Donaldson and G.Morfin eds., p. 691.
31. Proceedings of the Workshop on Triggering, Data Acquisition and Computing For High Energy/High Luminosity Hadron-Hadron Colliders, B. Cox, R. Fenner, and P. Hale eds., p. 282.
32. "Experimental Observation of the Higgs Boson at Large Hadron Colliders," these Proceedings.
33. C. Georgeopolis, private communication.
34. E.T. Nash, Parallel Session Talk, XXIII International Conference on High Energy Physics.
35. D. Cutts, private communication.
36. We use $5M as the cost of the Cyber205 - $10M for the ETA-10. The range is from running Geant to E711 reconstruction. The ETA-10 was taken to be 25 times a Cyber 205 or 40 GIP's.