An Overview of GEANT-4's Geometry

John Apostolakis and the RD44 collaboration

IT Division, CERN, CH-1211 Geneva 23, Switzerland.

We describe the key geometrical entities in Geant 4, which are used to define a detector geometry and propagate tracks through it. An important new feature allows the interchange of geometries with CAD tools. We describe organisational and optimisation methods developed to handle these and traditional geometries.

Key words: Simulation, Geometry, Boundary Represented Solids, Tracking, CAD systems

Geant 4 is a next generation detector simulation package currently being developed by an international HEP collaboration[1]. It has been designed using object oriented methods and implemented in C++. An overview of its current status can be found in another contribution to this conference [2]. This paper presents an overview of the geometrical components of the project.

The role of Geometry part of Geant 4 is to give the simulation user the ability to describe the geometrical structure of a detector and to allow the simulation system to propagate particles efficiently in this model detector. The Geometrical model of Geant 4 borrows concepts from previous simulation packages and adds advances in key areas and refinements.

Important considerations in the design were the need for performant navigation and the requirement to exchange detector geometries with Computer Aided Design (CAD) systems via the ISO 'Standard for the Exchange of Product Model Data' (STEP). This new feature allows, for the first time, the comparison between simulations of the 'exact' engineering descriptions of detectors and the more traditional, simpler, geometry descriptions used today.

The work described here has been done by the Geant-4 Geometry Working Group and includes, among others, the contributions of V. Grichine, P. Kent, S. Giani and J. Sulkimo.

1 Detector Representation

A detector's geometry is described by listing the different elements it contains and specifying their positions and orientations. Geant-3 [3] employed the concepts of a logical and a physical volume. A logical volume is a detector element of a certain shape which can hold other volumes inside it and can have other attributes. A physical volume is a logical volume positioned with respect to an enclosing (logical) volume. Structures of the detector that are repeated can be described as one logical volume and placed in several places. This provides great memory saving for complex structures.

These concepts are retained by Geant 4, and refined by separating the concept of a logical volume's shape and dimensions and placing it in a separate entity, that we name a "solid". Solids with simple shapes, including rectilinear boxes, trapezoids and spherical and cylindrical sections and shells, are available directly as solid objects, as part of the Constructed Solid Geometry (CSG). More complex volumes are defined by their bounding surfaces, which can be planes, second order surfaces or higher order B-spline surfaces[6]. They are called Boundary Representations (BREPs). This variety of shapes allowed us to implement the ability to import and export detector geometries using the ISO STEP standard for CAD systems.

A detector can be described by utilising a hierarchy of volumes, each of which can contain smaller volumes. Or it can be specified by listing all the volumes one by one, with no volume containing another. The former way has been utilised successfully to provide performant simulations, e.g. in Geant-3 [3]. The latter, 'flat', geometry is required to interface to CAD systems, in which a hierarchy is not provided. In this case a point can belong to one and only one volume.

Furthermore utility programs to translate geometries described in Geant-3 are being provided as well, to aid in the future migration of detector models and simulation programs.

2 Propagation in detector model.

The second task of the geometry is to handle the geometrical propagation of tracks and find their intersection with volume boundaries. It must do this also in the case of charged particles in a magnetic field. This is the geometrical or "pure" tracking and it consists of the three steps:

(i) Identifying the solid that the particle is inside.

(ii) Calculating the intersection distance to volumes inside of the current solid, and the distance to the boundary of the current solid.
(iii) Calculating the minimum step, and using it to move the particle.

Geant-3 utilises small pushes to change a volume at a boundary. These produce errors in the case of photon reflection and inefficiency when volume boundaries are not coincident - due to the need for a series of small steps. Geant-4's propagation methods were designed to overcome these limitations and to provide accuracy and efficiency. To minimise the number of geometrical calculations, tracks are propagated exactly to boundaries and their state is stored: whether it is on a boundary, whether it is exiting the current volume, etc.

Furthermore boundaries are imbued with a very small but finite thickness to take into account the roundoff and accumulated errors of floating point arithmetic. With this "tolerant" geometry[5] a particle exiting at a boundary ignores an intersection with the current volume if it is closer than the tolerance and the momentum is outward. The tolerance is chosen to be very small compared to detector features but much larger than the expected arithmetic errors.

Thus we have reduced the number of steps a particle must take to traverse a detector and reused the knowledge of a particle's state between states. In order to efficiently traverse a detector model geometry, however, it is critical to reduce the number of intersections of candidate volumes made in each step. As intersections are potentially costly operations, we seek an optimisation method to minimise the number of them required.

2.1 Tracking Optimisation

In tracking a particle's progress in a detector, a track can encounter any one of several detector parts at each step. Different methods have been used to lower the number of candidate volumes to be tested for intersection, some of them inspired by techniques used in ray tracing.

Geant 3.21 used its own scheme of virtual divisions[3] in which mother volumes (ones that contain other volumes) are divided evenly along one axis into sections. Each section stores pointers to sub-volumes it contains, compressed by bunching together common lists. This works well in a hierarchical detector description, in which the daughter volumes are fewer and structured. However it will fail for flat geometries, because a large number (for example ten thousand) of sub-volumes exists and the level of detail in different regions of the model detector varies greatly.

Influenced by this method and the voxel technique (in which space is recursively subdivided into octants), we have devised a new technique, which we call 'smart voxels'[4]. For each mother volume, a one dimensional virtual division is performed. The best axis is chosen by using a heuristic and equal subdivisions are gathered into single bins. Each division that contains too many volumes is independently refined by applying another division using a second Cartesian axis. A third subdivision along the remaining axis can be done also, if required. A simple example is illustrated in figure 1.

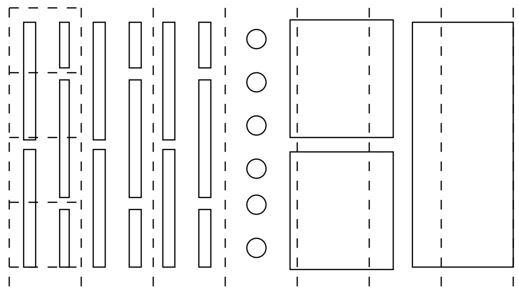

Fig. 1. Smart Voxels: A mother volume with an initial division along the horizontal axis, with some voxels' independent vertical divisions, are shown.

Smart voxels are computed at initialisation time, and require small memory and computing resources. At tracking time searching is done in the hierarchy of virtual divisions. A particle's most refined division is found first; its boundaries and the detector elements it contains are interrogated for intersection. If the particle leaves this voxel, the division and voxel it will enter can be found simply. Thus this method for pure tracking was found to be more performant than the navigation method of Geant 3.21, without the need to tune the detector description as in previous versions of Geant. In particularly difficult geometries a performance increase of over a factor of two was observed versus tuned Geant 3.21 descriptions, and an order of magnitude over previous versions like Geant 3.15.

2.2 Navigation in Magnetic Field

Charged particles moving in a magnetic field do not follow linear trajectories between interactions. In a uniform field their trajectories are helical, while in non-uniform fields they are curves that cannot be described analytically.

Propagation thus involves two tasks: first to calculate the trajectory (numerically if the field is non-uniform) and then to find its intersection with volume boundaries. For propagating the particle we solve its motion using a fourth order Runge-Kutta method. To intersect volume boundaries we split the curved path into sections and approximate each section by its chord. Although the tracking is not exact, by placing a constraint on the sections we ensure that a user defined accuracy is attained[7].

3 Summary

The geometry modeller of Geant 4 is capable of describing a large variety of shapes and can import detector descriptions from CAD systems that support the STEP interface standard. We started from well-tested principles for describing a model detector geometry and navigating in it. We have refined and greatly extended them to cope with a greater number and different organisation of detector volumes now possible. In doing this we have benefited greatly from the power and flexibility of object oriented methods.

We have made the first release of Geant-4 at this conference, but this is limited to the close collaborators of the project. During the coming year we will be working towards testing it, measuring and improving its performance and further enhancing it, in the expectation of a public pre-release in about a year's time.

References

[1] Geant 4: Simulation For The Next Generation Of HEP Experiments, Simone Giani and the DRDC RD44 project, Computing for High Energy Physics 1995.

[2] Geant 4: a world-wide collaboration to build Object Oriented HEP simulation software, Simone Giani, Computing for High Energy Physics 1997.

[3] Geant – Detector Description and Simulation Tool, CERN Program Library Long Writeup W5013, Application Software Group, IT Division, CERN.

[4] Pure Tracking and Geometry in GEANT 4, P. Kent, April 1995, (unpublished).

[5] Minimising Precision Problems in GEANT 4 Geometry, P.Kent, April 1995, Geant-4 working note.

[6] GEREP, a Boundary Representation Modeller proposal for Geant-4, J. Sulkimo and J. Vuoskoski, CN Division Internal Report, CERN.

[7] Particle propagation in a magnetic field in Geant-4, J.Apostolakis, S.Giani and V.Grichine, CERN, in preparation.