Start to End Simulation of a Neutrino Factory

C.R. Prior
STFC Rutherford Appleton Laboratory, Harwell Science & Innovation Campus, Chilton, Oxfordshire OX11 0QX, UK

1 Introduction

While computer modelling plays a large part in the design of any accelerator facility, the Neutrino Factory is so diverse in the nature of its constituents as to make reliable, accurate simulations harder to perform and yet even more important before any engineering work can go ahead. Each section must be individually modelled in detail to give confidence that it performs to expectations. Then a final start-to-end simulation has to be carried out to demonstrate compatibility and ascertain the output neutrino flux to the long-range detectors.

A schematic layout of the proposed facility, based on the recommendations of the International Scoping Study for a Neutrino Factory [1], is shown in Figure 1.

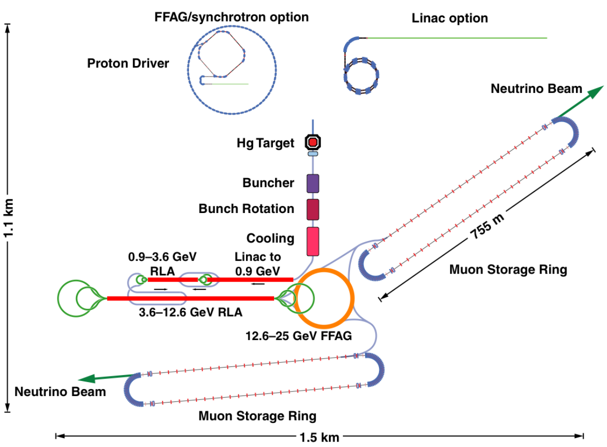

Fig. 1: Schematic layout of a Neutrino Factory

The constituent accelerators are: a high power proton driver, a pion production target, the muon front-end, a series of muon accelerators and finally a muon storage ring which directs the neutrinos from the decay through the Earth to the detectors. The target itself effectively acts as an interface between the proton driver and the muon accelerators. The pions that it produces will have a distribution that is a function of both its geometry and the parameters of the impinging proton beam. The sections that follow the target control and transport the muon beam, and differ widely in their purpose and mode of operation. Thus modelling codes need to treat the large aperture, high field solenoid capture channel; the technically difficult bunching and phase rotation sections; the novel, and as yet untried, ionisation cooling channel, which contains overlapping rf gradients and solenoid fields; a sequence of muon linacs, dog-bone RLAs and FFAG accelerators; and finally specially designed muon decay rings with a variety of geometries. This could well require several different codes and raise important issues such as compatibility, benchmarking, capability, the underlying physics and demand an understanding of each code's strengths and weaknesses. It is unlikely that any one code can achieve the task at the present time, though it may be possible to identify one or more candidates that could be developed to that end in the future.

2 Feasibility

Ideally a start-to-end simulation would start with the beam from the ion source in the proton driver and track it through to the Neutrino Factory target. The pions produced would then be followed as they decay to muons through the remainder of the accelerator complex. However the treatment of a proton beam is very different to the approach adopted for muons. Beams in proton drivers have generally quite small transverse dimensions and can be simulated by paraxial equations of motions with third order chromatic corrections. The muon beam in a Neutrino Factory will be very much larger and tracking will have to include terms to a much higher order to achieve the same degree of accuracy. In the proton accelerator, complex mechanisms such as beam chopping and injection phase space painting have to be modelled, and require specially written codes. Most importantly, high intensity proton beams experience severe space charge effects, and taking this into account means that all simulation particles need to be propagated together in time, so that techniques such as ray-tracing, commonly used for muons, are generally excluded.

A survey of codes suitable for modelling proton accelerators is given in [2] and a spreadsheet comparing capabilities and limitations can be found at [3]. Codes like WARP-3D and GPT can treat the beam coming from the ion source and the initial acceleration. Specialist codes, such as PARMTEQ, could be used for the RFQ. For the main linac, there are many codes available of which IMPACT is probably the most recent and subject to the most sustained development. Detailed calculation in linear accelerators is perfectly feasible because of the limited length that the beam is to be tracked, and many millions of simulation particles can be used to explore transmission, emittance growth and halo formation. Rings, however, are harder to model because of the large number of revolutions to be tracked (maybe 20,000 in a synchrotron for example); and during the injection, trapping and acceleration cycles, the beam is constantly evolving, so that "short cut" techniques may be unreliable. To some extent, individual aspects can be modelled using codes that work separately in longitudinal and transverse phase space. However coupling between all phase planes needs to be included in the final analysis. The code most able to model self-consistently in realistic CPU-time is ORBIT (ORNL) and the Brookhaven version known as SIMBAD. Other codes, such as SIMPSONS, could do the full 3D simulation but would take a great deal of computing time.

There is much experience of modelling accelerators from work on projects such as SNS and J-PARC, and development will carry on independently to provide faster, more accurate, modelling codes for the proton driver. But, so far as the treatment of muons is concerned, it should be acceptable to generate a model distribution from the Neutrino Factory target using a proprietary code like MARS, using as input some predetermined ideas about the impinging proton beam. MARS has built-in data from experimental studies of particle production, such as HARP, and is constantly being updated as further information becomes available.

As regards muons, an important issue in simulations of unstable beams – where particles are lost either through instabilities, interactions or decay – is maintaining a sufficient number of representative particles to obtain meaningful results. The problem does not arise in the proton driver where the main consideration at the outset is to design a machine with extremely low uncontrolled loss levels (roughly in 10^4). In the proton case, an initial 10^6 simulation particles will remain at that level throughout analysis of a well-designed system. In the muon sections of the Neutrino Factory, however, there is considerable loss in the front-end sections, particularly in the ionisation cooling. The particles coming from the target travel in all directions with a wide range of energies and, after the capture, bunching, phase rotation and cooling sections, current optimisations suggest that for every pion generated by the beam on target roughly 0.2 muons are within the acceptance of the subsequent accelerators. Depending on how a code works, random decays could then reduce this number throughout the simulation, so that the number of model particles reaching the muon storage ring, which is where the essential data of the likely neutrino production is calculated, could be fairly small. Ways round this include tracking through sections of the complex one at a time, analysing the beam distribution (via moments etc.) and using regeneration techniques to obtain more particles. This can be made to work well in 2D phase space; however, in 6D phase space, it is difficult to maintain higher order correlations, and therefore unlikely to give entirely satisfactory results. One alternative is to weight particles by a factor representing their likelihood of decay; in this way the number would not reduce, though some thought will need to go into interpreting the statistics at the end. Other tricks include making each particle into a family of particles, possibly with a slight scatter based on the average particle spacing. One could also, after a first run to obtain information, carry out a second run starting with a larger set of particles that are within a region that is not lost dynamically.

Not all codes will work on the same principles, so it will be important in a start-to-end simulation to understand their methods of operation, so that a measure of confidence can be ascribed to results from each code at each stage. Benchmarking is essential, not only via comparison of codes with each other when applied to the same input data set and same beam-line structure, but through comparison of codes against experiment. Thus the MICE experiment [4] may be suitable for testing codes modelling ionisation cooling, and the EMMA non-scaling FFAG ring [5] can be used to benchmark muon acceleration tools. The US Study IIa muon front-end, for which data-sets exist on the web, might be used for code *v.* code comparison. Where there are discrepancies, a thorough check needs to be made to ensure codes are using the same definitions (of, for example, rms emittance) and that these are being calculated in essentially the same way (remembering that some methods of calculation are more susceptible to machine errors than others, especially where multi-particle simulations are concerned). Note too that there will always be slight differences in individual runs because of the randomness of the decays.

3 Codes Available for Muon Tracking

The proton codes mentioned above are not really suitable for handling muon beams in NF-like channels at the present time. Instead several others have been constructed for this purpose in recent years.

3.1 ICOOL

ICOOL [6] is a 3-dimensional tracking program that was originally written to study ionisation cooling of muon beams, but has since developed to cover pion collection, rf phase rotation and acceleration in FFAG rings. The program simulates particle-by-particle propagation through materials and electromagnetic fields. Particles are tracked and regions are described using Serret-Frenet ("accelerator") co-ordinates. The program was written with low energy (1 MeV/c – 1 GeV/c) muons in mind, but tracking of electrons, pions, kaons and protons is also possible. The physics processes available include decays, delta rays, multiple scattering, energy loss and straggling. The structure is aimed at eventual inclusion of proper space-charge interactions, though only a very rudimentary model is available to date.

ICOOL has been widely used for modelling the muon front-end of the Neutrino Factory and Muon Collider. Written in Fortran77, it runs successfully on UNIX, PC and Macintosh platforms, taking about 2 hrs to model the US Study II front-end with 2400 simulation particles [7]. It is now generally considered to be a benchmark for the development of other codes. However, while it provides flexible options for a wide range of studies, it could perhaps be improved a little in terms of user friendliness.

3.2 Muon1

The inception of Muon1 was triggered by a need to model the pion capture channel, where particles can have wide ranging energies and directions of travel (Figure 2). It still requires the inclusion of field maps in order to treat FFAGs in detail but has the potential to model most, if not all, of the muon sections of the Neutrino Factory. It can handle a full non-linearised 3-dimensional simulation, and uses realistic initial pion distributions from the target. Stochastic particle decays are included with multi-particle phase space generation. Solenoid end-fields are treated via 3rd order power expansions, and OPERA-3D field maps with tri-linear interpolation are used for magnets. A typical simulation uses 20k–50k particles. Tracking is performed by 4th order classical Runge-Kutta methods on the 6D phase space. Time-steps are of the order

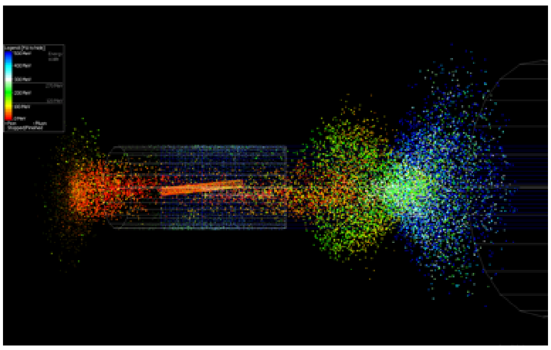

Fig. 2: Muon1 modelling of the pion beam from the Neutrino Factory target

of 0.01 ns.

This code contains an optimiser that overcomes problems of calculating derivatives caused by stochastic noise and the sheer number of dimensions by using a genetic algorithm. Beginning with a random design, the code improves with mutation, interpolation and crossover. It has proved successful in handling front-end problems with up to 137 parameters in the decay channel [8].

3.3 G4MICE

The G4MICE code [9] is based on GEANT4 and was developed specifically to model the MICE ionisation cooling experiment. Further developments allow treatment of other cooling schemes such as ring and dog-bone coolers, snakes and the Guggenheim structure. It is now capable of treating a range of combined accelerator and detector simulation. Most Neutrino Factory accelerator components are included, such as solenoids, arbitrary order multipoles with Enge fringe fields, arbitrary 3D field maps, different shaped absorbers, pillbox cavities and general RF cavities via superfish data. Modelling of FFAG accelerators uses displaced quadrupoles, superimposed multipoles and 3D field maps. The full suite of GEANT4 tracking and physics processes is built in, so that particle-matter interactions and decays can be included in the analysis.

The code integrates a representative set of particles through a given system, specified for example by arbitrary field maps, and uses the results to generate a linear mapping for the main particle tracking. Second order Lie algebra algorithms are partially implemented and the aim is to increase the order of the mapping in future months. There are analysis tools for calculating 2D, 4D and 6D emittances, amplitudes and β-functions. The code can read and write in G4MICE, G4Beamline and ICOOL formats, and uses a graphical interface to aid users. Graphical features include visualisation and animation of accelerator simulations.

3.4 G4Beamline

Also developed from the GEANT4 toolkit, G4Beamline [10] is intended to perform accurate and realistic simulations of beam-lines and related systems using single-particle tracking. Its flexibility has been demonstrated by simulating complex beamlines like the MICE muon beam and the Neutrino Factory Study II SFOFO muon cooling channel. The code is written in C^{++} and runs on Windows, Linux (Intel) and Mac OS X platforms. It pays attention to proper shielding and takes into account unwanted particles such as delta rays and neutrons from vacuum windows.

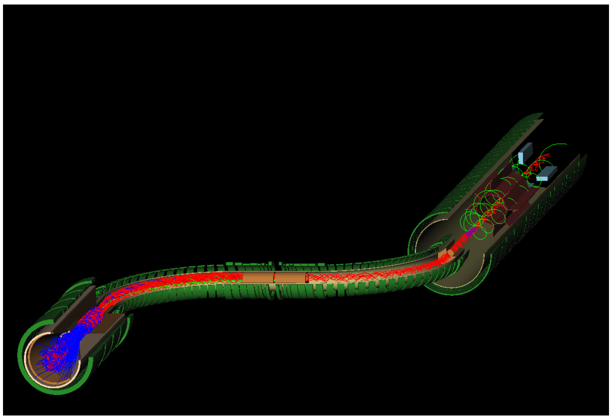

Fig. 3: Graphical layout of the Mu2e target beamline from G4Beamline

G4Beamline includes all the beamline elements used in current muon collider and Neutrino Factory designs: dipoles, quadrupoles, solenoids, pillbox RF cavities, absorbers with shaped windows, helical solenoids, beam pipes, targets, etc. New components can be implemented on demand. It produces an impressive display of graphical output (see Figure 3) and clicking on different elements reveals the data that goes into the structure of the beam at that point. Importantly, it has a user-friendly interface.

3.5 OptiM

The OptiM code, used at Fermilab to model the Tevatron, originated as a linear optics code, similar to MAD but with an integrated GUI. For Neutrino Factory studies it has been used to study beam optics in the initial muon linac and the downstream dog-bone RLAs. The code has an integrated system for optics design, support and measurement analysis. It attempts to be user-friendly and has a system of on-line help.

Magnet strengths are computed from magnet currents using the results of magnet measurements; additional small factors are introduced to match computations with optical measurements (where available). Standard instrumentation (BPMs, Profile monitors, current monitors, scrapers) are included. Computations are carried out in 6D and cover a large set of optical elements, with x-y coupling plus treatment of acceleration.

The full OptiM code is intended to run on an MS-Windows platform but will also work on other platforms without the GUI. It has disadvantages in that it uses Borland C^{++}, which is no longer supported, treats only multipoles of zero length, and ignores non-linearities in combined function magnets. On the other hand, it does not use Taylor expansions and so is more readily able to model the large momentum spreads found in the Neutrino Factory muon beams.

3.6 S-Code

A new code, known provisionally as the S-Code, has been developed by S. Machida at RAL to treat FFAG accelerators in a very general way [11]. Rather than using a reference orbit defined by the positions of the lattice elements, as in a conventional accelerator, the code de-couples the lattice geometry from the particle orbits. The lattice geometry itself is determined (possibly only approximately) by another

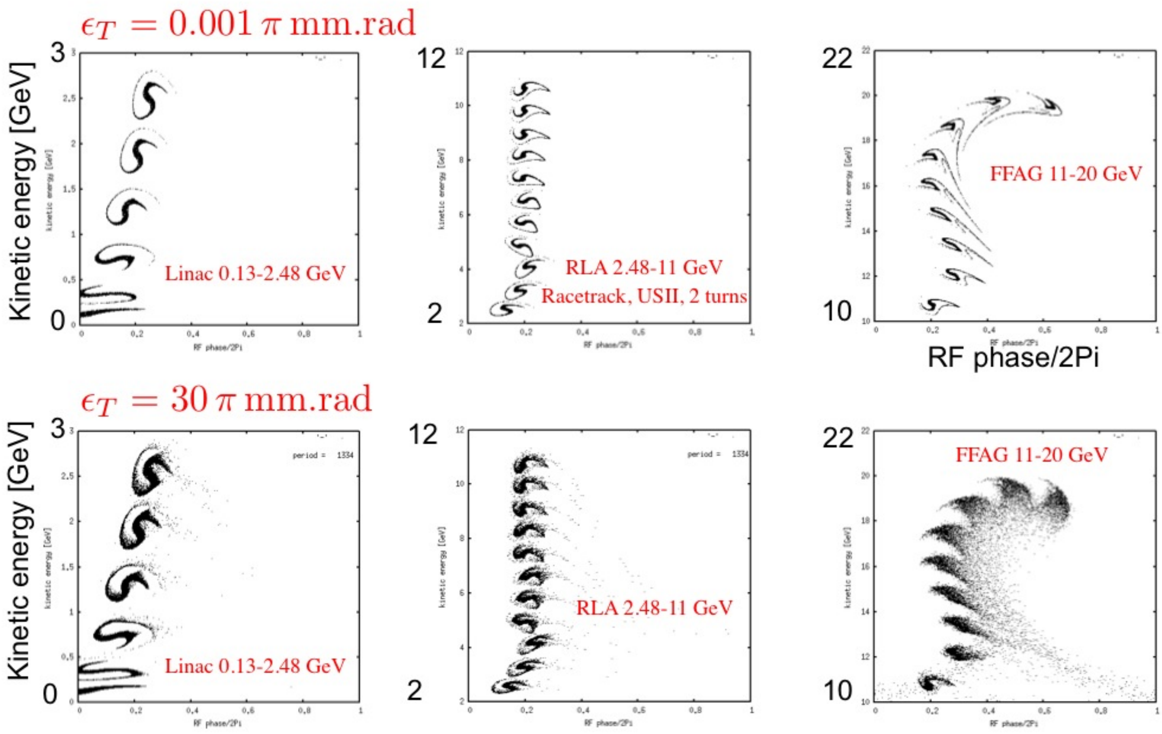

Fig. 4: Simulation of muon acceleration using S-Code (from Nufact'06)

code such as MAD. The new code tracks particles based on the predetermined location and strength of magnetic elements and rf cavities in this lattice, and updates the three components of particle momentum as it progresses. Elements are represented as a collection of thin lenses, which include Enge-type end field regions. Particle trajectories are therefore made up of small straight line segments. A simple model of space-charge is included based on formulae for uniform beams, but development of a more realistic model is in progress.

The S-Code has been used successfully to model EMMA, the electron model of a non-scaling muon FFAG under construction in the U.K. [5]. It also produced the first attempt at a continuous simulation of a muon beam in a Neutrino Factory. Taking a representative particle distribution from the exit of the muon front-end, the code tracked through a simplified linac from 0.13 to 2.48 GeV; then through a racetrack re-circulating linac taken from US Study II, up to 11 GeV; and finally through a non-scaling FFAG from Study IIa to an energy of 20 GeV. The results for longitudinal phase space, comparing a beam with almost zero transverse emittance with a beam with the full NF emittance of 30 π mm.rad, are shown in Figure 4. The difficulties in handling a large emittance beam are clearly illustrated, and the S-Code has since shown in more detail that (even though partial solutions have been found) phase slippage is an issue that can only get worse in going from one FFAG into another.

3.7 Zgoubi

Zgoubi [12] is a ray-tracing code that computes particle trajectories in arbitrary magnetic and/or electric field maps or analytical models. The code is a compendium of numerical recipes for simulation of most types of optical elements encountered in beam optics. Over the years, it has acquired a reputation as a reliable tool for modelling muon beam behaviour, often serving as a benchmark for the development of new codes for study the Neutrino Factory or Muon Collider.

Zgoubi contains a built-in fit procedure, calculates synchrotron radiation, treats spin tracking and and can handle a wide range of Monte Carlo processes. It uses an integration method founded on stepwise resolution of the Lorentz equation using a technique based on Taylor series. The position and velocity of a particle at location $M_2(s+\Delta s)$ after a step Δs are computed from Taylor expansion at location $M_1(s)$. The coefficients in these Taylor expansions involve the derivatives of the velocity, which are drawn from the Lorentz equation and require a knowledge (or modelling) of the magnetic field and its derivatives. The high accuracy of the integration method allows efficient multiturn tracking in periodic machines.

Examples of the use of the code, showing its versatility, are given in [11]. These include simulations of a radial FFAG, a spiral FFAG accelerating protons from 17 to 180 MeV (ignoring space charge), a linear FFAG lattice with serpentine acceleration, and an isochronous muon FFAG based on a pumplet lattice. A recent application has seen a study of muon beam polarisation in a bow-tie storage ring.

4 Work programme

Of all the codes described here, Zgoubi is arguably best able to simulate the Neutrino Factory from the exit of the cooling channel through acceleration to the decay rings. S-Code and possibly G4Beamline would be good candidates for cross-checking. The cooling section itself becomes more difficult with the passage of time because of increasing uncertainties in the physics, along with issues of field representation in the codes. It is then particularly important to know what has been included in the simulations and make clear to others when codes are up-to-date.

Assuming that a single repository has been set up providing read-access, the following are among the necessary conditions for success of the project.

- Ensure there is one single person (the "Co-ordinator") who has responsibility for ensuring co-ordination between all sections of the simulation study. The Co-ordinator can decide who should have write-access to the files in the central repository.
- All codes should be properly documented and user manuals regularly updated.
- The set of MARS files or equivalent for input to the muon front-end should be updated and added to as appropriate. These should allow for changes in target materials and geometry, as well as effects of any alteration in the proton beam.

- Master lattices files should be stored in the central repository and notification sent to the start-to-end simulation group whenever any are revised. Files should be labelled with version numbers and accompanied by a note of changes made, as appropriate. Note that sets of files for tracking may need to comprise more than a single input file to a beam optics code.
- Attention should be paid to interfaces between different codes and, where necessary, conversion modules written so that output from one code can be directly imported to another. The fact that different codes may use different units should also be taken into account. It might be useful to identify a standard format in which data-sets are stored, with modules available to convert files between each one of the codes in use and this standard. Interfaces between codes would then always be via the standard structure.
- All beam distribution files should be stored in the same repository, clearly labelled with time and date and any special characteristics, as appropriate. Uploading to the central repository should be via the Co-ordinator, who has responsibility for assessing the exact status of the data-sets (a master data-set that should be used for all subsequent simulations, or an alternative based on a new idea for study of some special features only, for example).
- It is important to ensure that all files are properly backed up on a regular basis (at least once a week, and preferably every night). Participants may well have their own copies, but this should not be relied on.
- Rigorous attempts should be made at comparison between codes. Given the list above, there are at least two codes capable of modelling different regions of the muon part of Neutrino Factory, and it should be confirmed that they predict similar beam behaviour and output similar quantities to an agreed level of accuracy. Where there are discrepancies, these should be reported to the Co-ordinator and attempts made to track down the source.
- Codes that rely on experimental information (such as the effects of different absorber materials, for example) should be updated as new data becomes available, with users notified via the Co-ordinator.

Experience shows that such a procedure, however prescriptive, is necessary for the proper execution and success of a simulation project. Without a well-defined framework, it is easy to lapse into duplication of files, repetition of work and general confusion among participants. The demands on the Co-ordinator should not be under-estimated: this is an important rôle, carrying responsibility and demanding time. The bottom line is the generation of a start-to-end simulation that is reliable, gives confidence that all aspects have been investigated in detail, provides enough information for engineering and construction, and gives confidence that there are no hidden surprises in store when operation goes ahead.

References

[1] Prior, CR: Proceedings of US Particle Accelerator Conference PAC07, Albuquerque, New Mexico, June 2007. `http://pac07.org/proceedings/PAPERS/TUXKI02.PDF`

[2] Prior, CR: "Space Charge Simulation", Proceedings of the 29th ICFA Advanced Beam Dynamics Workshop on Beam Halo Dynamics, Diagnostics and Collimation, HALO'03, Montauk, New York, pp 32-37, May 2003.

[3] ICFA Beam Dynamics Newsletter **31**, pp 125-130 and Appendix A, August 2003. http://www-bd.fnal.gov/icfabd/Newsletter31.pdf

[4] Kaplan, D and Long, K: "MICE: The International Muon Ionization Cooling Experiment", XXIII International Symposium on Lepton and Photon Interactions at High Energy (LP07), Daegu, Korea, August 2007.

[5] Edgecock, TR: "EMMA - the World's First Non-Scaling FFAG", Proceedings of the XIth European Particle Accelerator Conference, EPAC'08, Genoa, Italy, June 2008.

[6] Fernow, RC: ICOOL User's Guide, http://pubweb.bnl.gov/users/gallardo/www/for_icool/icoolman.html

[7] Fernow, RC: "Recent Developments on the Muon-Facility Design-Code ICOOL", Proceedings of 2005 Particle Accelerator Conference, PAC'05, Knoxville, TN.

[8] Brooks, SJ: "Muon Capture Schemes for the Neutrino Factory", D.Phil. thesis, University of Oxford, UK, 2008.

[9] Rogers, CT: "Simulation of MICE Using G4MICE", Proceedings of European Particle Accelerator Conference, EPAC06, Edinburgh, Scotland, June 2006. http://cern.ch/AccelConf/e06/PAPERS/WEPLS003.PDF

[10] Roberts, T: G4Beamline User's Guide (1.14), http://www.muonsinc.com/g4beamline/G4beamlineUsersGuide.pdf

[11] Prior, CR (Ed.): ICFA Beam Dynamics Newsletter **43**, August 2007. http://www-bd.fnal.gov/icfabd/Newsletter43.pdf

[12] Méot, F and Valero, S: Zgoubi Users' Guide, FERMILAB-TM-2010 (1997). See also, NIM A 427 (1999) 353-356.