Neutral Particle Spectrometer Geant4 Simulation Guide for Hall C DVCS

Ho San KO[1],*

[1]Laboratoire de Physique des 2 infinis Irène Joliot-Curie

October 25, 2020

This document is a short guide on how to run a Neutral Particle Spectrometer (NPS) Geant4 simulation with a DVCS event generator and a photon reconstruction software. There are also short descriptions of the necessary classes. Currently, the kinematic setting exists only for the proposed Hall C DVCS. For any questions or comments, please send an email to hosanko@jlab.org.

*E-mail : hosanko@jlab.org

Contents

1 Running a simulation

1.1 Environment setting in JLab/ifarm and getting the code

The Geant4 library of this simulation is based on a version 4.10.03.p03. The environment configuration in JLab/ifarm is done by 'source site/12gev_phys/softenv.csh 2.3' (or 'softenv.sh' depending on the preference of the user). This configuration uses a Geant4 version 4.10.03.p02. Detailed configurations are shown in Fig. 1. Additional environment settings are necessary for the DVCS event generator and the photon reconstruction to work: see Fig. 2.

The codes for the simulations are on GitHub (`https://github.com/gboon18/HallC_NPS/`) and the directory for the simulation with the DVCS event generator and the photon reconstruction is 'DVCS_evt_gen/'.

1.2 Configuration and compiling

This section presents one of the ways to configure a Makefile and compile using 'ccmake'. You can skip this section if you know how to configure and compile the code yourself.

In the directory 'DVCS_evt_gen/', the subdirectory 'DVCS/' has source codes and 'macros/' has example macros that might be useful to some users. Make a build directory (let us name it 'build/') inside the 'DVCS_evt_gen/'. In the 'build/', configure the source with 'ccmake', by typing 'ccmake ../DVCS/'. If you set the environment 'source site/12gev_phys/softenv.csh 2.3' correctly according to the JLab user guide, all the environments needed for the simulation will be configured automatically: see Fig. 3. Type 'c' to configure and 'g' to generate the 'Makefile'. After that, type 'make' to make executable 'DVCS'.

1.3 Running a simulation

There are 4 arguments to input when running the simulation:

```
                        J E F F E R S O N   L A B
------------------------------------------------------------------------------
This computer is owned by the Federal Government or is connected to a
network owned by the Federal Government.  It is for authorized use only.
Users have no explicit or implicit expectation of privacy.

Any or all uses of this system and all files on this system may be intercepted,
monitored, recorded, copied, audited, inspected, and disclosed to authorized
site, Department of Energy, and law enforcement personnel, as well as
authorized officials of other agencies, both domestic and foreign. By using
this system, the user consents to such interception, monitoring, recording,
copying, auditing, inspection, and disclosure at the discretion of authorized
site or Department of Energy personnel.

Unauthorized or improper use of this system may result in administrative
disciplinary action and civil and criminal penalties. By continuing to use
this system you indicate your awareness of and consent to these terms and
conditions of use. LOG OFF IMMEDIATELY if you do not agree to the conditions
stated in this warning.
------------------------------------------------------------------------------
CentOS Linux release 7.7.1908



> Common Environment Version: <2.3>  (Wed October 3 2018)
> Running as hosanko on ifarm1801.jlab.org
> OS Release:    Linux_CentOS7.7.1908-x86_64-gcc4.8.5
> JLAB_ROOT set to:     /site/12gev_phys
> JLAB_SOFTWARE set to: /site/12gev_phys/2.3/Linux_CentOS7.7.1908-x86_64-gcc4.8.5

> BANKS         version:  1.4
> CCDB          version:  1.06.02
> CLHEP         version:  2.4.0.4
> EVIO          version:  5.1
> GEANT4        version:  4.10.04.p02
> GEMC          version:  2.7
> JANA          version:  0.8.0
> MLIBRARY      version:  1.3
> MYSQL         installed in  /site/12gev_phys/2.3/Linux_CentOS7.7.1908-x86_64-gcc4.8.5/mysql/lib
> QT            version:  5.10.1
> ROOT          version:  6.14.04
> SCONS         version:  1.7
> XERCESC       version:  3.2.2

>> For documentation on the environment please visit https://data.jlab.org
```

Figure 1: JLab/ifarm environment configuration. The configuration is done by 'source /site/12gev_phys/softenv.csh 2.3'.

```
#ROOT ## needed (especially) when compiling the NPS photon reconstruction software
setenv ROOTSYS /apps/root/5.34.36/root
setenv LD_LIBRARY_PATH ${ROOTSYS}/lib:${LD_LIBRARY_PATH}
setenv PATH ${ROOTSYS}/bin:${PATH}
#NPS software environment
setenv NPS_SOFT /work/hallc/nps/hosan/NPS_SOFT
setenv LD_LIBRARY_PATH ${NPS_SOFT}:${LD_LIBRARY_PATH}
setenv PATH ${NPS_SOFT}:${PATH}
#DVCS event generator for NPS environment #No need to do this but just in case.
setenv DVCS_EVENT_GEN /work/hallc/nps/hosan/dvcs_gen
setenv LD_LIBRARY_PATH ${DVCS_EVENT_GEN}:${LD_LIBRARY_PATH}
setenv PATH ${DVCS_EVENT_GEN}:${PATH}
#analyzer
setenv ANALYZER /work/hallc/nps/hosan/analyzer-1.5.22
setenv LD_LIBRARY_PATH ${ANALYZER}:${LD_LIBRARY_PATH}
setenv PATH ${ANALYZER}:${PATH}
```

Figure 2: Additional environment settings for the DVCS event generator and the photon reconstruction to work. C shell based.

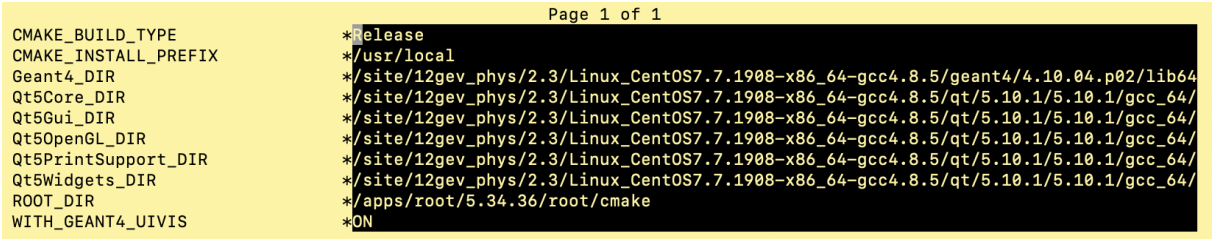

Figure 3: ccmake configuration

	Energy Dependence at fixed (Q^2, x_B)												Low-x_B				High-Q^2		
x_B	0.36					0.50			0.60				0.2				0.36	0.50	0.60
Q^2 (GeV)^2	3.0			4.0		3.4		4.8	5.1			6.0	2.0			3.0	5.5	8.1	10
k (GeV)	6.6*	8.8	11	8.8*	11	8.8	11	11	6.6	8.8*	11	11	6.6	8.8	11	11	11		
k' (GeV)	2.2	4.4	6.6	2.9	5.1	5.2	7.4	5.9	2.1	4.3	6.5	5.7	1.3	3.5	5.7	3.0	2.9	2.4	2.1
θ_{Calo} (deg)	11.7	14.7	16.2	10.3	12.4	20.2	21.7	16.6	13.8	17.8	19.8	17.2	6.3	9.2	10.6	6.3	7.9	8.0	8.0
D_{Calo} (m)	3	3	3	4	3	3	3	3	3	3	3	3	6	4	4	6	4	4	4
I_{beam} (μA)	28	28	28	50	28	28	28	28	28	28	28	28	11	5	50	11	50	50	50
N_{evt} (10^5)	1.5	8.8	8.2	2.1	7.9	7.3	11	5.1	0.2	0.2	2.7	2.6	3.5	3.6	64	3.4	6.1	0.8	0.4
$\sigma_{M_X^2}$ (GeV^2)	0.13	0.13	0.12	0.15	0.15	0.09	0.09	0.11	0.09	0.09	0.09	0.09	0.17	0.17	0.17	0.22	0.19	0.15	0.13
Days	1	2	1	1	3	3	2	5	5	1	5	10	1	1	1	1	5	5	12
run number:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

Figure 4: Run numbers. Figure extracted from the proposal PR12-13-010 [1].

- run number,
- name of the macro file,
- target type (0 for hydrogen, 1 for deuterium),
- target type (0 for proton, 1 for neutron, 2 for deuteron).

There exist 19 run numbers from 0 to 18 depending on the kinematic settings in the proposal PR12-13-010 [1]. The run numbers are given as shown in Fig. 4. In order to use the photon reconstruction software, one needs to upload the kinematic settings (beam energy, HMS/NPS angles, etc.) on MySQL server. For now, only the DVCS kinematic settings proposed in PR12-13-010 (Fig. 4) are available on the server. One can, however, select the desired target between hydrogen to deuterium in those DVCS settings.

Here is an example command to run a simulation: ’./DVCS 2 macros/test.mac 1 1’. This will run kinematic settings ’2’ with a macro file called ’test.mac’ in ’macros/’ directory and use

neutron target in liquid deuterium.

Once the simulation runs, it will ask for the name of the output file and 3 random numbers for the Geant4 random engine. The output file is in ROOT format and can be designated anywhere the user wants (e.g. /u/home/hosanko/output.root).

For the simulation to run, the user needs to give several inputs which are already in the macro file (see "macros/test.mac" for an example):

- /testhadr/Physics QBBC (not necessary to run but recommended for better background generation)
- /run/initialize
- /run/beamOn 1000 (make 1000 initial electron beam events)

2 Simulation structure

2.1 DVCS.cc

The simulation is based on the Geant4 example 'AnaEx02'. This main code, located in the 'DVCS/' directory, initializes and runs the simulation.

2.2 DetectorConstruction

Located in the 'DVCS/src/' directory, the geometries of the simulation are defined in this code. There are Sensitive Detectors (ConstructSDandField), in which the user can collect desired information within the logical volume of choice. Also, the parameters needed for the optical photons, such as refractive index, are described here.

The basic geometries are the target (15 cm in length and 2.5 cm in radius) and its chamber, downstream beam pipes, and the NPS. The target is the geometrical center of the simulation. The NPS consists of a sweeper magnet and a calorimeter. The angular difference between the

sweeper magnet and the NPS is 4° or 5.5° depending on the kinematic settings. The default is 5.5°. However, for some kinematic settings, the magnet center will be tilted to 4° for the calorimeter to reach a smaller angle: see [2, 3] for more detail. The center of the magnet is positioned 1.6 m from the target.

The magnetic field of the sweeper magnet is generated by the 'SimpleField' class in 'ConstructSDandField' class. The 'SimpleField' in 'DVCS/src/' uses a magnetic field table of a 1 cm grid. The center of the magnetic field table is also similarly positioned as the sweeper magnet. The angular difference between the magnetic field table center and the NPS is 4.1° or 5.6° depending on the kinematic settings. The center of the magnetic field is positioned 1.57 m from the target.

The calorimeter consists of 30×36 $PbWO_4$ crystals (20.5×20.5×200.5 mm^3) each of them wrapped with a VM2000 wrapper of a thickness of 65 μm. The carbon fiber frame is placed at the front- and backside of each of the crystal with a width of 2 cm and a thickness of 0.5 mm. There are glasses (20.5×20.5×20.5 mm^3) attached to the backside of each of the crystal to simulate the PMT when the optical photon physics simulation is at work.

2.3 PhysicsList

The user can set physics processes. Although it is not recommended to change the physics process from 'QBBC', the user can use a command '/testhadr/Physics <name of the physics process>' to change it. The macro 'test.mac' uses 'QBBC'. The argument will be input through 'PhysicsListMessenger'. The lines for the optical photon physics are commented out because the optical photon physics increases the computation time immensely. If one needs the optical photon physics, uncomment the lines near '// Needed for optical photon physics' in 'DVCS/src/PhysicsList.cc'. To record the number of optical photons collected at the PMTs, several lines in 'HistoManager' class should also be uncommented: see the next section.

2.4 HistoManager

It creates a tree of a ROOT based output file and saves the events. There are 2 trees:

- 't' saves the deposited energy per crystal per event (beam electron).
- 't_dvcs' saves the generated (at the vertex) and reconstructed information of the electron, photon, etc. of the event.
- 'evtNb' in both trees 't' and 't_dvcs' records the event number, i.e. which beam electron it was, to allow users to match the event in 't' and 't_dvcs'

If one wants to record the number of optical photons collected at the PMT, uncomment the lines below '//temporarily disabled, no op physics' in 'DVCS/src/HistoManager.cc'

2.5 PrimaryGeneratorAction

It uses a 'G4ParticleGun' class with a DVCS event generator class 'TGenDVCS'. It generates scattered electron and DVCS photon per event (beam electron), according to the kinematic setting.

2.6 RunAction

It sets the random seeds and book the output file and saves them at the end of the run (at the end of the whole events).

2.7 EventAction

It gets the informations the user needs, such as energy deposited in each crystal, from the detectors and gives that information to HistoManager. The EventAction also gathers information from such as the PrimaryGeneratorAction, B5HadCalorimeterHit (collects deposited energy

in each crystal), and SteppingAction (identify scattered electron) to, for example, reconstruct DVCS photon and send them to HistoManager.

2.8 B5HadCalorimeterSD and B5HadCalorimeterHit, i.e. Sensitive Detector

These classes are based on Geant4 'example/basic/B5/' and one of several Sensitive Detectors (SDs). These SD and Hit classes collect information of the particle within the volume the user has set. 'B5HadCalorimeterSD' and 'B5HadCalorimeterHit' classes are one of them. The logical volume of the $PbWO_4$ crystals is set as a Sensitive Detector to these classes through the 'ConstructSDandField' in the 'DetectorConstruction' class. It collects the deposited energy, the number of optical photons created in the crystal.

They are one of many Sensitive Detectors in the simulation. The basic structure is the same for all the Sensitive Detectors.

3 Short descriptions of t_dvcs branches

Here are short descriptions of t_dvcs branches. It is recommended for the user to cross-check what is written here.

- evtNb: A sequential number assigned to each beam electron. Starts from 0.
- clust_ene: Total energy of one cluster. Clusters of each event (i.e. beam electron) on the calorimeter are calculated using the clustering algorithm (i.e. photon reconstruction software).
- clust_x and clust_y: x and y position of the cluster on the calorimeter.
- clust_W2: Missing mass squared of the event. $(p_b+p_p-p_e-p_g)^2$, where p_b denotes the 4-momentum of the initial beam electron (only has z-direction momentum component), p_p

represents the 4-momentum of the initial proton (stationary in the lab frame), p_e shows the 4-momentum of the scattered electron, and p_g stands for the 4-momentum of the photon calculated by the clustering algorithm.

- clust_size: Shows the number of blocks (crystals) used in the clustering algorithm.

- psf: phase space factor calculated from the DVCS event generator. $psf = \Delta Q^2 \Delta x_B \Delta t \Delta \phi_e$. It must be noted that $\Delta\phi$ (2π) is not included in the simulation. Therefore, it should originally be $psf = \Delta Q^2 \Delta x_B \Delta t \Delta \phi \Delta \phi_e$.

- RIE_px, py, and pz: Momenta x, y, and z of the initial electron. It is a fancy name, but it is just a beam energy initially set (by the kinematic setting). It only has a z component (= beam energy) and the others are set to zero.

- GIE_px, py, and pz: Initial electron momenta x, y, and z at the vertex. Radiative corrections, such as pre-vertex external Bremsstrahlung, are applied. It only has a z component (= beam energy) and the others are set to zero.

- RSE_px, py, and pz: Detected scattered electron momenta x, y, and z. SteppingAction class detects electron when it passes through the opening of the HMS and records its momentum and energy.

- GSE_px, py, and pz: Scattered electron momenta x, y, and z at the vertex.

- RP_px, py, and pz: Reconstructed momenta x, y, and z of the photon. They are reconstructed by the clustering algorithm. Vertex resolution smearing due to the nominal vertex resolution of the HMS is also taken into account. It is set to $\sigma_{90^\circ} = 1.0$. The resolution smearing is done with Gaussian random ($\sigma_{vertex} = \frac{\sigma_{90^\circ}}{\sin\theta_{HRS}}$).

- RP_px, py, and pz: Photon momenta x, y, and z at the vertex.

- RV_z: Smeared vertex z position. This is the one that takes into account the HMS nominal vertex resolution which is mentioned in item "RP_px"

- GV_z: Vertex z poisition.

- Rt and Gt: Reconstructed t and t calculated at the vertex, respectively. For t calculation, information of the initial and the scattered electron and the photon is needed. Reconstructed t uses the value RIE_px (py & pz), RSE_px (py & pz), and RP_px (py & pz). Vertex t uses the value GIE_px (py & pz), GSE_px (py & pz), and GP_px (py & pz).

- RxB and GxB: Reconstructed x_B and x_B calculated at the vertex, respectively. For x_B calculation, information of the initial and the scattered electron is needed. Reconstructed x_B uses the value RIE_px (py & pz) and RSE_px (py & pz). Vertex x_B uses the value GIE_px (py & pz) and GSE_px (py & pz).

- RQ2 and GQ2: Reconstructed Q^2 and Q^2 calculated at the vertex, respectively. For Q^2 calculation, information of the initial and the scattered electron is needed. Reconstructed Q^2 uses the value RIE_px (py & pz) and RSE_px (py & pz). Vertex Q^2 uses the value GIE_px (py & pz) and GSE_px (py & pz).

- Rphi and Gphi: Reconstructed ϕ (angle between the leptonic and the hadronic planes) and ϕ calculated at the vertex, respectively. For ϕ calculation, information of the initial and the scattered electron and the photon is needed. Reconstructed ϕ uses the value RIE_px (py & pz), RSE_px (py & pz), and RP_px (py & pz). Vertex ϕ uses the value GIE_px (py & pz), GSE_px (py & pz), and GP_px (py & pz).

- X_sum, X_diff, and X_BH: These are unpolarized exclusive photon electroproduction cross section, polarized exclusive photon electroproduction cross section, and Bethe-Heitler process cross section. The cross sections are calculated using a code written by

H. Moutarde. Its original formulas are written by P. Guichon and M. Vanderhaegen. For more, a user can look into TGenDVCS class and TGVKelly class in "/work/hallc/nps/hosan/dvcs_gen" directory.

- Rr_val, Gr_val: Reconstructed r value and r value calculated at the vertex. r value is calculated using Hall A HRS R-function. It needs to be modified for the Hall C HMS acceptance.

References

[1] *PR12-13-010.* https://hallaweb.jlab.org/experiment/DVCS/documents/proposals/E12-13-010_proposal.pdf

[2] *NPS ERR Report Responses* by Tanja Horn. NPS ERR. https://wiki.jlab.org/cuawiki/images/2/26/NPS_ERR_Report_Responses_v3.pdf

[3] *NPS: Mechanical Structures and Installation* by Steven Lessiter. NPS ERR. https://wiki.jlab.org/cuawiki/images/6/61/NPS_ERR_2019_-_Mechanical-Rev_B.pdf