

University of Saskatchewan  
Subatomic Physics Internal Report

SPIR 146.1

# Calibration and Use of the 5-Paddle Flux Monitor

Rob Pywell

Revised September, 2015

## Abstract

This document describes how to calibrate and use the 5-paddle photon flux monitor.

## Contents

<b>Contents</b>	<b>1</b>
<b>1 Introduction</b>	<b>2</b>
<b>2 Calibration of the Flux Monitor</b>	<b>3</b>
<b>3 Analysis of the Calibration Data</b>	<b>6</b>
3.1 Extracting The Run Information . . . . .	6
3.2 Calculating the Raw Calibration Factor . . . . .	9
3.3 Corrections using the GEANT4 Simulation . . . . .	12
<b>4 Using The Flux Monitor</b>	<b>18</b>
<b>5 Flux Monitor Check List</b>	<b>21</b>
5.1 Calibrating the Flux Monitor . . . . .	21
5.2 Running an Experiment . . . . .	21
<b>6 Using the Simulation</b>	<b>22</b>
6.1 Predicting the Calibration Factor . . . . .	22
6.2 Finding the Absorption Factors . . . . .	23
6.3 Simulating a High Photon Flux Rate . . . . .	24

# 1 Introduction

The 5-paddle photon flux monitor was developed by the University of Saskatchewan group for use at the High Intensity Gamma Source (HIGS) at the Duke University Free Electron LASER facility. It formed part of the M.Sc. thesis of Octavian Mavrichi[1] and its initial use is describe in the NIMA publication, Pywell et al.[2].

It was designed to intercept only a small fraction of the incident photons. It may be used in a configuration where it is placed before or after the experiment target. In its current configuration it is placed in the collimator hut just after the primary collimator and is therefore before the experiment target.

The monitor works by detecting the recoil electrons following Compton scattering or the electron-positron pairs created in pair production. These are created primarily in a thin converter placed between the second and third scintillator paddles. Currently this converter (also referred to in the simulation as the absorber) is a thin Aluminum sheet. The recoil electrons or positrons are detected by requiring a triple coincidence between the last three scintillator paddles while in anti-coincidence with the paddle just before the converter (referred to as the veto paddle). The veto paddle reduced background from upstream material. The additional paddle is only used in the set-up of the paddle system to determine the threshold of the veto paddle, when optimizing its response of the monitor and checking its operation.

It has been shown that the efficiency of the 5-paddle system can be predicted well by a GEANT4[3] simulation. However we do not rely on the simulation. Rather, the efficiency is measured for each experimental set-up by comparing the response of the monitor to number of photons measured by a large volume NaI detector. This calibration is done at a low rate so that NaI can count all the photons incident upon it.

A package of codes used for the calibration and use of the flux monitor is available under the "Software" tab at

<http://kanga.usask.ca/nucleus>

A copy of the GEANT4 code used to simulate the flux monitor is available at the same location.

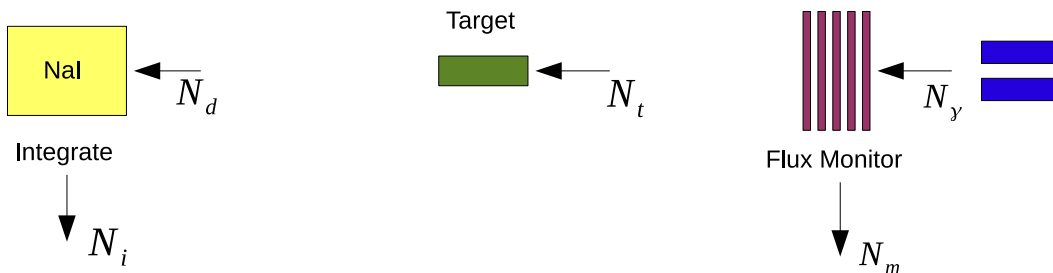


Figure 1: A schematic of the experimental set-up to calibrate the flux monitor. The calibration may be with the experiment target in place, as shown, or removed. The actual experiment would be with the NaI detector moved out of the beam line.

## 2 Calibration of the Flux Monitor

The goal of the calibration is to determine the calibration factor  $f_m$  defined by the relation

$$N_\gamma = f_m N_m \quad (1)$$

where  $N_\gamma$  is the number of gamma rays incident on the flux monitor, and  $N_m$  is the number of counts from the photon flux monitor. Since all experiments only record data during their live time,  $N_m$  is to be considered the quantity measured during the experiment live time and therefore  $N_\gamma$  will be the number of gamma rays that could cause reactions that can be measured by the experiment if they reach the target. Ultimately we need to find the number of gamma rays that reach the target,  $N_t$ . We will use the simulation to find the absorption factor needed to relate  $N_t$  to  $N_\gamma$ .

The experimental arrangement to calibrate the flux monitor is shown in Figure 1. The calibration may be performed with the experiment target in place or not. The preference would be to do the calibration with the target removed, but in some cases it may be difficult to remove the target. The actual experiment would use the same schematic but with the NaI detector removed from the beam line.

We define  $N_d$  to be the number of gamma rays incident on the NaI detector. During the calibration we are not able to measure  $N_d$  directly. What we measure is a NaI spectrum which must be integrated. Because of significant low energy background in the NaI spectrum it is usually integrated over an energy range that avoids most of this background. This integral we will call  $N_i$ . The initial goal of the calibration is to measure the uncorrected or raw calibration factor  $f'_m$  defined by

$$N_i = f'_m N_m. \quad (2)$$

Later we will use the GEANT4 simulation to make corrections to determine  $f_m$ .

An example of a NaI spectrum with the low energy background visible may be seen in SPIR-144[4] figure 3. Another example from the recent deuterium photodisintegration measurement at a photon beam energy of 18 MeV is shown in Figure 2 shows the NaI spectrum taken during the calibration with a 18 MeV gamma ray beam. A charge integrating ADC was used for the NaI signal. The count rate in the NaI was about 6000 Hz, which is

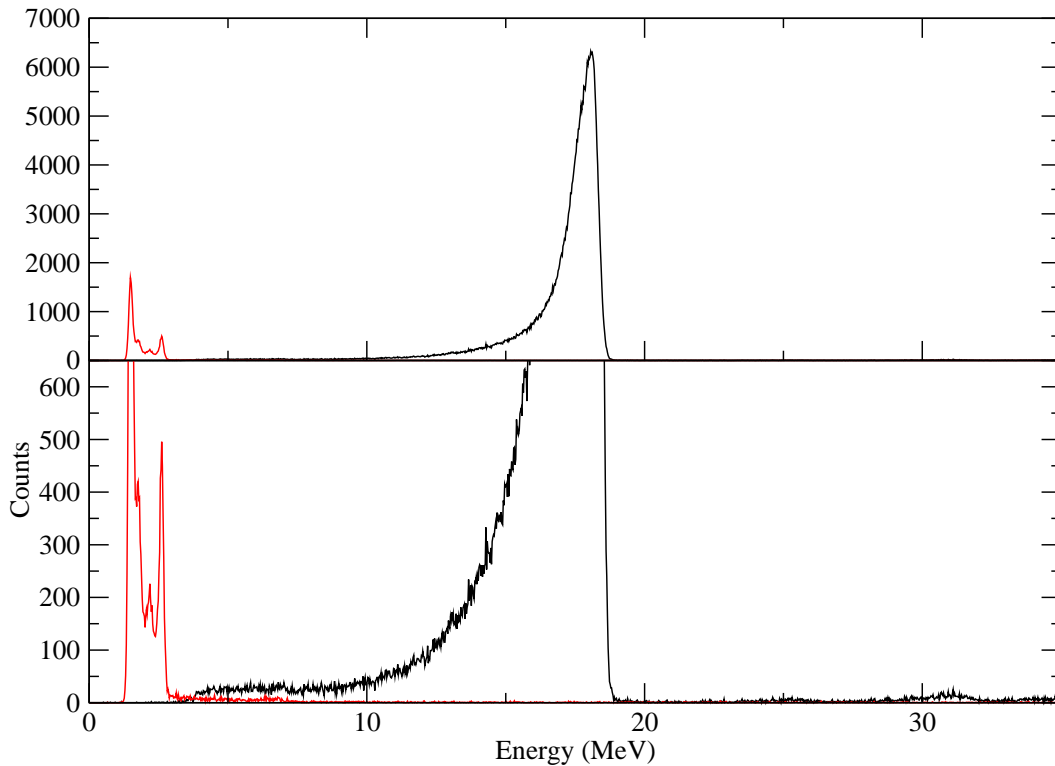


Figure 2: The NaI spectrum taken during the calibration with a 18 MeV gamma ray beam (in black). The count rate in the NaI was about 6000 Hz. The red line is a background spectrum taken with the beam off, and scaled to the same live time as the the spectrum with the beam on.

near the highest rate recommended. At this rate large pulse heights due to pile-up is not a large effect. A hardware threshold reduced the low energy background. In this example the NaI spectrum was integrated from 8 MeV to 19.5 MeV. From this a background spectrum is integrated over the same energy range and subtracted from the NaI spectrum after scaling by the live time. This background correction is essentially negligible provided the integration range does not include the low energy background. This allows one to calculate  $N_i$  in equation 2.

$N_m$  in equation 2 is the number of counts from the flux monitor during the live time. However there is a non-negligible background in the flux monitor. This background is difficult to measure because very long count times would be necessary to reduce the error sufficiently, and because it has been found that background rate can depend on such factors as low long the beam has been off. Therefore it is necessary to measure the background rate indirectly.

The measured number of flux monitor counts during the live time  $T_{live}$  is  $N'_m$  of which some is background. Therefore

$$N_m = N'_m - B_m T_{live} \quad (3)$$

Table 1: Flux monitor calibration runs for the 18 MeV deuterium photodisintegration measurement during the October 2010 running period. (The second calibration set was abbreviated since the beam was due to be turned off.) For set 1 a 10.7 cm long D<sub>2</sub>O target was in the beam while for set 2 a 2.0 cm long target was in place.

Set Number	Beam Energy (MeV)	Run Number	Attenuators In Beam	Flux Monitor Rate (Hz)	NaI Rate (Hz)	Flux Monitor Count	Live Time (s)
1	18	145	1-2-3-4	40	1800	9838	219.3
1	18	146	1-2-3-4-6	12	550	5064	455.6
1	18	147	2-3-4	108	5000	11179	100.7
1	18	148	2-3-4-6	21	1200	5556	233.8
2	18	160	2-4-6	24	1200	6563	291.5
2	18	161	2-3-4	100	5400	7105	147.5
2	18	163	2-3-4-5	10	460	3121	336.6

where  $B_m$  is the background rate in the flux monitor. So

$$f'_m = \frac{N_i}{N'_m - B_m T_{live}} . \quad (4)$$

Rearranging this equation we get

$$\frac{N_i}{T_{live}} = f'_m \frac{N'_m}{T_{live}} - B_m f'_m . \quad (5)$$

Therefore the flux monitor calibration should be done at several different beam rates. Then a plot of the rate in the NaI detector ( $\frac{N_i}{T_{live}}$ ) against the measured rate in the flux monitor ( $\frac{N'_m}{T_{live}}$ ) should be a straight line, the slope of which is  $f'_m$ .

The flux monitor calibration will be done with the copper attenuators placed in the beam so that the beam intensity is reduced to minimize pile-up in the NaI detector. This will be with NaI count rates less than about 6000 Hz. At least 3, preferably 4 different rates should be used. We want to determine  $f'_m$  to better than about 1%. This means that we would like to have a total flux monitor counts in each measurement ( $N'_m$ ) of near 10,000. At the lower flux monitor count rates (around 10 Hz) this can be relaxed. In that case the measurement needs to be long enough to get near 4,000 counts from the flux monitor. These are all at flux monitor count rates low enough so that we may make the assumption that the probability of two photons in a beam bunch is negligible.

Table 1 shows examples of two calibration measurements taken during the 18 MeV deuterium photodisintegration measurement.

## 3 Analysis of the Calibration Data

### 3.1 Extracting The Run Information

The first step in the analysis is to extract the required data from each flux monitor calibration run. I have found it convenient to create a single text file with this information in it. The following is an example of this information for run 161.

Listing of file "runinfo.0161.dat"

```
-----  
RunNumber: 161  
NaI-Integral: 361605.  
5-Paddle(live): 7105  
5-Paddle(real): 23256  
Bunch(live): 8.23020e+08  
Bunch(real): 1.42803e+09  
Live_time: 147.418  
Real_time: 255.941  
Live_fraction: 0.575984  
3-Paddle(live): 0  
3-Paddle(real): 0  
Veto(live): 115780  
Veto(real): 204083  
5-Paddle_rate: 48.1963  
NaI_trigger: 2516.84  
-----
```

This information must be extracted from the data file(s) for the run. If the data was taken using the Coda system root may be used to extract this information. The root routine `flux_analyze.C` is used. It takes as an argument the run number to be analyzed. i.e.

```
> flux_analyze(<runnumber>)
```

This routine first reads a file, `rundata.runs`. This file is essentially a list of run numbers along with the name of a file which contains constants specific to the analysis of that run number. These constants included the pedestals for the paddle ADCs, the energy calibration factors for the NaI ADC, and the integration range to be used for the NaI. `flux_analyze` will read the parameters from this file. An example of this file is given below.

```
# This file shows which runs we have and where  
# information needed for analysis is stored.  
# Energy(MeV) run_number Filename  
#61.0 3003 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat  
#61.0 3019 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat  
61.0 3020 /home/higs/5-paddle/sept2015/run_data/rundata.3020.dat  
#61.0 3021 /home/higs/5-paddle/sept2015/run_data/rundata.3021.dat
```

```

61.0 3043 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3045 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3047 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3048 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3050 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3051 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3052 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3053 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3054 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3055 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3056 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3057 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3058 /home/higs/5-paddle/sept2015/run_data/rundata.3003.dat
61.0 3060 /home/higs/5-paddle/sept2015/run_data/rundata.3060.dat

```

In the current (September 2015) setup, the scaler information is also stored in the root data file and `flux_analyze` will get that information from there. The scaler information needed includes the flux monitor counts and the number of beam bunches in the live time of the calibration run. (The scaler information is also stored in a text file. Earlier versions of `flux_analyze` called a Perl program `getscalers.pl` which searched through the `flux_scaler` file for the scaler information for that run number. This scaler information is then read into `flux_analyze`.) `flux_analyze` will then create histograms, integrate the NaI spectrum, and write files which contain some of the histograms and the `runinfo` file.

The version of `flux_analyze.C` provided in this package was the one used to analyze the flux monitor data for the September 2015 Compton scattering experiment. (This contained several details related to dealing with severe NaI pileup due to the use of a peak sensing ADC and the accelerator setup which was present in an early (June 2009) run. The analysis of this data is described in detail in SPIR-144[4].) A similar setup was used in the September 2015 run but the pile-up problems were much less severe. Nevertheless the code still includes corrections for pileup. The routine will undoubtedly need to be modified for different situations.

There is a significant amount of background that needs to be subtracted from the NaI spectrum before obtaining the NaI integral. A long background run should be taken of the NaI. If this background run is analyzed using `flux_analyze` a file labeled `data.naiE2.xxxx.dat` (where `xxxx` is the run number) is created in the `spectra` subdirectory (if that subdirectory exists). This file contains the NaI spectrum from this background run. If the filename of this background run is given in the `rundata` file read by `flux_analyze` it is used to subtract background from the NaI spectrum. To do the subtraction the live time for the background run is also needed and it too is contained in the `rundata` file.

An example `rundata` file is given below. (This is for the September 2015 run where the beam energy was 61 MeV.)

```

# pedestals hit_threshold
0 105 20
1 120 20

```

```

2 143 20
3 110 20
4 105 20
# NaI pedestal hit_threshold
5 20 10
# NaI calibration
# energy channel
10.144 450.35
61.0 1881
# NaI integration range
20. 70.
# NaI pileup maximum energy
140.
# NaI background livetime (s)
55708.0
# NaI background filename
spectra/data.naiE2.3060.dat

```

If, in this file, the NaI background live time is set to zero a background file is not read and no background subtraction will be done. (This for example would be necessary for generating the background file itself.)

For the example below, for the October 2010 Deuterium photodisintegration experiment, the data was taken using the LUCID data acquisition system where all the event-by-event information and the scaler information is contained in one file. In that case the NaI integration and the generation of the `runinfo` file is accomplished in one step by LUCID.

Not all the information contained in the "runinfo" file is needed for the subsequent analysis to obtain the flux monitor calibration. The necessary quantities are:

```

RunNumber:
NaI-Integral:
5-Paddle(live):
Bunch(live):
Bunch(real):
Real_time:

```

These labels must be written exactly as listed above. There must be white space between the label and the following number. Blank lines and comment lines (lines beginning with #) are ignored in the `runinfo` file. The quantities are defined more fully as follows:

`NaI-Integral` This is the integral of the NaI spectrum between the specified energy limits.

`5-paddle(live)` This is the number of counts from the 5-paddle flux monitor during the live time of the measurement.

`Bunch(live)` This is the number of beam bunches during the live time of the measurement. Often this quantity is prescaled before going into a scaler, but the quantity in the "runinfo" file should have the prescaling factor removed. The scaler must have been gated



with a live-time gate, or alternatively vetoed with a dead-time gate when the data acquisition system is unable to accept events.

**Bunch(real)** This is the number of beam bunches during the real time of the measurement. Again this should have any prescaling factor removed.

**Real\_time** This is the real time of the measurement in seconds. This would normally have been measured by some fixed frequency signal going into a scaler.

The subsequent analysis needs the live time of the measurement. This is calculated from the real time and the ratio of the Bunch(live) and Bunch(real) values since this is would usually be more accurate than using a time oscillator gated by the live time of the measurement.

## 3.2 Calculating the Raw Calibration Factor

The data runs in each measurement set will be fitted to find  $f'_m$ . The Perl program `rates.pl` is used to do the fitting. The `-h` option to `rates.pl` prints a help message. i.e.

```
> rates.pl -h
Usage rates.pl [-p -h] <flux_run_filename
  -p = Print information about each run.
  -h = Print this message and exit.
"flux_run_filename" is the name of a file containing
the data runs to be analyzed.
```

The file that is read from the standard input contains a list of the data runs to be analyzed. An example of this file for the October 2010 run is given below.

Listing of the file "flux.runs"

```
-----
# Flux Monitor Calibration data
# October 2010
# Set Energy runnumber
# (Set = B means a background run for the NaI)
B 18.0 164
1 18.0 145
1 18.0 146
1 18.0 147
1 18.0 148
2 18.0 160
2 18.0 161
2 18.0 163
-----
```

The format of the file is a calibration set number (several runs at different beam rates for the same experimental setup), the photon beam energy, and the run number. If the set number is replaced with "B", the run is a background run for the NaI detector (in this case the beam is off so the energy is meaningless). This background run will be used to make

a correction to the NaI integral for all the runs following it in the file until another "B" is encountered. If no "B" line appears in the file before a line containing a set and run number, then no background correction will be made. Note: If the data was taken with the coda system and the NaI integrals are generated by `flux_analyze.C`, the NaI background is already subtracted so no "B" lines are needed. Comment lines (beginning with #) and blank line are ignored in the file. There are a few options that are hard coded in the program `rates.pl`. These appear at the beginning of `rates.pl` as shown below,

```
#!/usr/bin/perl

$max_pad_rate = 200.0;
$const_back = 0; # if set to 1 the following background rate is used
$const_back_rate = 0.23;
$const_back_rate_err = 0.14;
$runinfodir = "../flux_info";
```

Only data runs where the count rate from the flux monitor is less than `$max_pad_rate` (in Hz) are included in the fitting. It is possible to use a constant background rate for the flux monitor, rather than the background rate being determined from the fit, by setting `$const_back = 1` and the following two variables will then specify the background rate and its uncertainty. The directory where the `runinfo` file for each run are stored is given by `$runinfodir`.

An example of running `rates.pl` is given below.

```
> rates.pl < flux.runs
```

```
Fitting for each set...
Maximum 5-paddle rate = 200 Hz
Set = 1 Energy = 18.0 MeV
  FACTOR f_m' = 44.07 +/- 0.25
  BACKGROUND B = 0.26 +/- 0.14
  Chisq = 4.3, Reduced Chisq = 2.2
Set = 2 Energy = 18.0 MeV
  FACTOR f_m' = 51.68 +/- 0.42
  BACKGROUND B = 1.16 +/- 0.14
  Chisq = 1.6, Reduced Chisq = 1.6
```

In this example we see that the raw calibration factors  $f'_m$  are significantly different for the two sets. This is because different thickness targets were in place for each set.

For each set two files are created. One contains the data that is fitted to find the calibration factor and background. The other contains the fit to the data. These may be plotted to check on the goodness of the fit. Examples of these two files are given below for set 1 from the October 2010 calibration.

Listing of the file "Rate.set1.18.0MeV.dat"

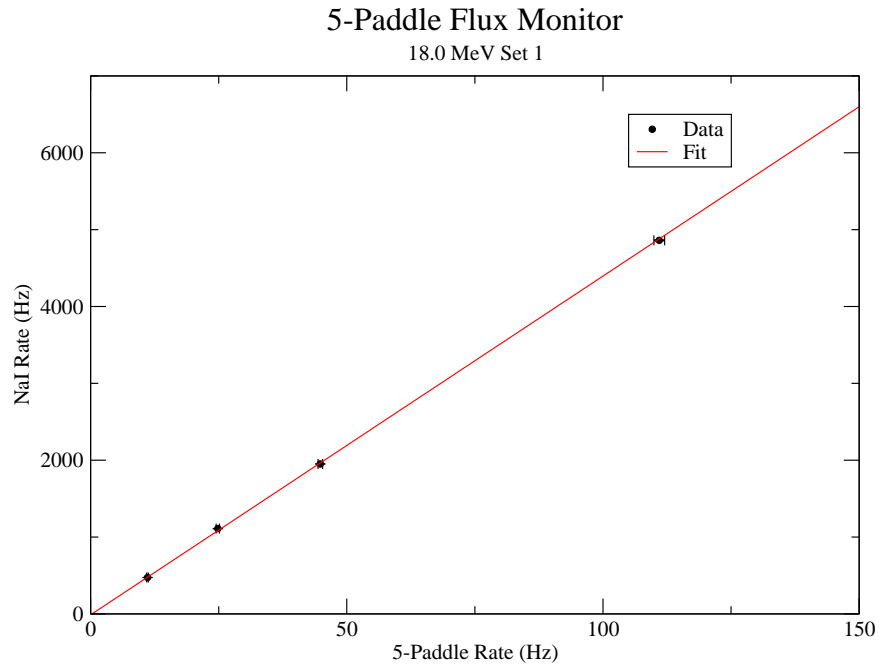


Figure 3: The fit to the flux monitor calibration data for set 1 from the October 2010 run.

```

-----
# Set 1
# Energy = 18.0 MeV
# 5-Paddle Calibration
# <pad5_rate> <nai_rate> <pad5_rate_err> <nai_rate_err>
# Run 145
44.83 1951.21 0.45 3.01
# Run 146
11.11 474.06 0.16 1.03
# Run 147
110.98 4860.83 1.05 7.15
# Run 148
24.83 1110.07 0.33 2.24
-----

```

```

Listing of the file "fit.Rate.set1.18.0MeV.dat"
-----

```

```

0. -11.45819999999
200 8802.54179999879
-----

```

This data is plotted for set 1 and set 2 in figures 3 and 4.

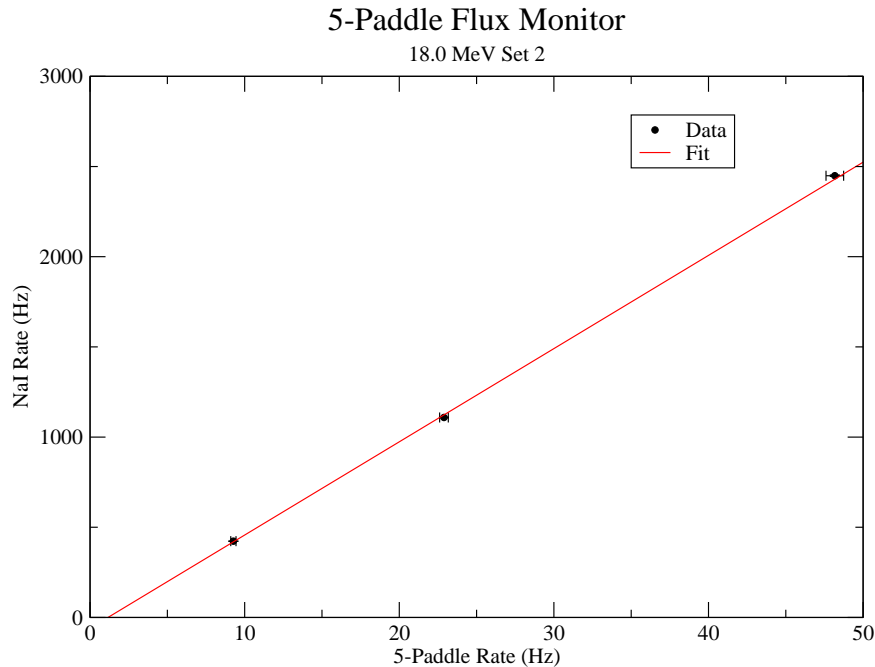


Figure 4: The fit to the flux monitor calibration data for set 2 from the October 2010 run.

### 3.3 Corrections using the GEANT4 Simulation

We now need to use the GEANT4 simulation of the flux monitor to make the corrections necessary to convert  $f_m^l$  to  $f_m$ . We do not use the simulation to predict the efficiency of the flux monitor. Rather, we only use the simulation to calculate the absorption of photons through the flux monitor, air and (if present during the calibration) a target. Therefore it is not necessary to have the detector thresholds of the flux monitor paddles matched to those of the actual detector. However it is important to have the geometry accurately modelled by the GEANT4 simulation.

We use the simulation named `flux`. (The latest version at the time of this report (September 2015) is version 1.6.) This version of the simulation does not write event-by-event data necessary for a complete analysis of the flux monitor, but only determines the number of photons detected by the NaI detector and compares it to the number of photons incident on the flux monitor.

The simulation is controlled by macro files which defines the geometry and other aspects of the simulation. It is convenient to create two separate macro files. One defining the geometry and the other defining the parameters for a particular run.

We give an example below of the files used for the calibration of the flux monitor for the September 2015 run with a liquid helium target. The `DetectorConstruction.cc` source file must be modified for all but the simplest of target configurations. the commands:

```
/flux/detector/setTargetActive true
/flux/detector/setTargetDistance 416 cm
/flux/detector/setTargetThickness 2.0 cm
```

```
/flux/detector/setTargetDiameter 4 cm
/flux/detector/setTargetMaterial D20
```

refer only to the actual active volume of the target and not any container, so anything else that is part of the target system, and is in the path of the beam, must be included in the `DetectorConstruction.cc` file. In that file the target should be defined within the construct

```
if(target_exist)
{
... define target here
}
```

If the command

```
/flux/detector/setTargetActive false
```

is set in a macro file the target will not be constructed. Within your target definition you can use the variables

```
G4Material *target_material; // Target material
G4double target_thickness; // Target thickness
G4double target_diameter; // Target diameter
G4double target_distance; // distance from collimator to target centre
```

which may be changed with commands in a macro file, but you may need to define other variable.

An example of a macro file defining the geometry as used for the September 2015 run is `setgeometry_target_in.mac` and its contents are given below.

```
# Flux monitor paddle options
#-----
/flux/detector/setMonitorDistance 0.5 m
/flux/detector/setAbsMat Aluminum_6061
/flux/detector/setAbsDist 14.25 mm
/flux/detector/setAbsThick 2.297 mm
/flux/detector/setScintWidth 15 cm
/flux/detector/setScintHeight 10 cm
/flux/detector/setScintSeperation 28.5 mm
# ----
/flux/detector/setLayerIndex 0
/flux/detector/setScintThick 2.339 mm
/flux/detector/setThresh 270 keV
/flux/detector/setSigma_Elec 0.03

/flux/detector/setLayerIndex 1
/flux/detector/setScintThick 2.073 mm
/flux/detector/setThresh 100 keV
```

```

/flux/detector/setSigma_Elec 0.03

/flux/detector/setLayerIndex 2
/flux/detector/setScintThick 2.076 mm
/flux/detector/setThresh 81 keV
/flux/detector/setSigma_Elec 0.03

/flux/detector/setLayerIndex 3
/flux/detector/setScintThick 2.352 mm
/flux/detector/setThresh 78 keV
/flux/detector/setSigma_Elec 0.03

/flux/detector/setLayerIndex 4
/flux/detector/setScintThick 2.116 mm
/flux/detector/setThresh 108 keV
/flux/detector/setSigma_Elec 0.03

# ---- NaI options ----
/flux/detector/setNaIActive true
/flux/detector/setNaIDistance 17 m

/flux/detector/setNaIDiameter 23.83 cm
/flux/detector/setNaIThickness 35.56 cm
/flux/detector/setNaIAlThickness 2.5 mm
/flux/detector/setNaIThreshold 100 keV
/flux/detector/setNaIIntegrate_start 20. MeV
/flux/detector/setNaIIntegrate_stop 70. MeV

# ---- Target Options ----
/flux/detector/setTargetActive true
/flux/detector/setTargetDistance 14 m
/flux/detector/setTargetThickness 20. cm
/flux/detector/setTargetDiameter 3.81 cm
/flux/detector/setTargetMaterial LHe

# -- make sure changes are incorporated and print
/flux/detector/update
/flux/detector/print
# -----

```

Note that comment lines (beginning with #) and blank lines are ignored.

A macro file for actually running the simulation is `run_target_in.mac`. This file sets up the energy of the beam and other run specific parameters and starts a run. Its contents are

```
# Set geometry for target in
```

```

/control/execute setgeometry_target_in.mac
# ---- Gun Options ---
# 20 MeV is the default energy
# EnergySpread parameter (in %) will apply to
# last energy added and all those added after
# unless redefined
/flux/gun/removeEnergy 20 MeV

/flux/gun/addEnergy 61.0 MeV
/flux/gun/setEnergySpread 6.78
/flux/gun/setTargetMinimumEnergy 55 MeV

# photon rate of 0 means one photon per geant event
/flux/gun/setPhotonRate 0
#/flux/gun/setPhotonRate 55000.
/flux/gun/setCollDiameter 25.4 mm
/flux/gun/setCollDistance 60 m

# -- make sure changes are incorporated and print
/flux/detector/print
/flux/gun/print

/run/beamOn 40000
# -----

```

The simulation is normally run by typing

```
flux run_target_in.mac
```

where `run_target_in.mac` is the name of a macro file used to control the simulation. Typing simply

```
flux
```

will bring up the QT GUI so that you may check the geometry. More details about running the simulation are given in section `refsec:using-simulation`.

From the simulation we need to find two absorption factors  $C_A$  and  $C_B$  needed to correct  $f'_m$  to find the number of photons incident on the target. The simulation must be set up with the geometry exactly as it is for the calibration of the flux monitor. If the target was in place during the calibration it must be included and made "active". (In the simulation there is an option to make the target disappear by making it not "active".)

The energy integration range for the NaI is set to the same value as used in the analysis of the calibration. The simulation will report the ratio of  $N_{\gamma A}$ , the number of photons emerging from the collimator, and  $N_{iA}$ , the integral of the photons detected by the NaI between the minimum and maximum energies. The ratio of these we define as the absorption factor

$$C_A = \frac{N_{\gamma A}}{N_{iA}}. \quad (6)$$

If the target is in place, the simulation will also record the number of full energy gamma rays hitting the target,  $N_{iB}$ . We define as the absorption factor

$$C_B = \frac{N_{\gamma A}}{N_{iB}}. \quad (7)$$

If the target was not in place during the actual calibration, two simulations will need to be run. One without the target in place to find  $C_A$ , and a second with the target in place to find  $C_B$ .

When we do the actual flux monitor calibration,  $N_i$  is the integral of the NaI spectrum, and so the number of photons emerging from the collimator is

$$N_\gamma = \frac{N_{\gamma A}}{N_{iA}} N_i = C_A N_i \quad (8)$$

where  $N_m$  is the number of counts measured from the flux monitor. Therefore, the true calibration factor is

$$f_m = \frac{N_\gamma}{N_m} = C_A \frac{N_i}{N_m} = C_A f'_m. \quad (9)$$

The number of photons incident on the target position is then

$$N_t = \frac{N_{iB}}{N_{\gamma B}} N_\gamma = \frac{1}{C_B} N_\gamma = \frac{f_m}{C_B} N_m. \quad (10)$$

As an example we find the true calibration factor for set 1 of the October 2010 experiment. For set 1 a 10 cm long D<sub>2</sub>O target was in place during the calibration. Simulation A uses the parameters as listed in the example `runeff.mac` above. The most relevant variables are

```
/flux/detector/setNaIDistance 17 m
/flux/detector/setNaIThreshold 100 keV
/flux/detector/setNaIIntegrate_start 8 MeV
/flux/detector/setNaIIntegrate_stop 19.5 MeV
```

```
# ---- Target Options ----
/flux/detector/setTargetActive true
/flux/detector/setTargetThickness 10.7 cm
/flux/detector/setTargetMaterial D2O
```

The final output for this simulation is

```
End of Run 111 -----
One photon per geant event (bunch)
NaI integration range: 8 to 19.5 MeV
Energy 18 (MeV)
Events 40000 Bunches 40000 Photons 40000
Detected by monitor 673
```



Table 2: Applying the absorption factors to find the true calibration factor using equation 9.

Set Number	Beam Energy (MeV)	$f'_m$	$C_A$	$C_B$	$f_m$
1	18.0	$44.07 \pm 0.25$	$1.344 \pm 0.008$	$1.069 \pm 0.006$	$59.23 \pm 0.49$
2	18.0	$51.68 \pm 0.42$	$1.142 \pm 0.006$	$1.069 \pm 0.006$	$59.02 \pm 0.57$

```

Veto hits 331
Hits on NaI detector 30660
Integral of NaI detector 29767
Minimum Energy for hit on target 17.5 MeV
Hits on target 37568
Integral on target 37427
Absorption factor C_A (photons / integral): 1.344 +/- 0.008 )
Absorption factor C_B (photons / target integral): 1.069 +/- 0.006 )
(Predicted calibration factor (photons / detected): 59.44 +/- 2.29 )

```

Thus the absorption factor  $C_A = 1.344 \pm 0.008$ , and the absorption factor  $C_B = 1.069 \pm 0.006$ .

In the second set the target thickness was 2.0 cm. Using the simulation in the same way we get  $C_A = 1.142 \pm 0.006$  and (of course)  $C_B$  is the same as calculated above. We can now calculate the true calibration factors using equation 9. The results are shown in table 2. We see that, after corrections, the true correction factor is the same, within errors, for the same beam energy. This is as we should expect. We also note that the uncertainty in the correction factor is of the order of 1% which should be the aim of each calibration. We would use the average value of  $f_m = 59.13 \pm 0.51$  for the analysis of data at 18.0 MeV with this experimental set up. We would also use the "flux-monitor-to-target" absorption factor ( $C_B$ ) of  $1.069 \pm 0.006$  to find the number of photons incident on the target.

It may be noted that the GEANT4 simulation predicts that the calibration factor for 18 MeV should be  $59.44 \pm 2.29$ . The startling agreement with our measured factor should be considered somewhat fortuitous since no attempt has been made to ensure that the flux monitor thresholds used in the simulation exactly match the actual values for the flux monitor during the measurement. Nevertheless, this shows that the simulation is good for estimating the the calibration factor. The calibration factor does need to be measured for each experimental setup.

## 4 Using The Flux Monitor

To use the output of the flux monitor in the calculation of a cross section, the number of photons incident on the target during the live time of the experiment is needed. Thus it is necessary to measure the output of the flux monitor during the live time of the experiment. This would usually be done by counting the flux monitor pulses with a live time gated scaler.

The number of photons incident on the flux monitor would, in principle, be given by

$$n_\gamma = f_m N_m \quad (11)$$

where  $N_m$  is the flux monitor counts during the live time. However, the HIGS beam is not a 100% duty factor photon beam. The photon beam comes in bunches at a rate of 5.58 MHz or about 180 ns apart. The dead-time of the flux monitor is of the order 60 ns (about the width of the veto generated by the veto paddle). Therefore photons arriving in separate bunches will always be counted by the scaler if they are detected by the flux monitor. However, more than one photon being detected in the flux monitor in a single bunch will be counted as only one.

The calibration factor determined in the above calibration procedure,  $f_m$ , is measured at very low photon fluxes so that the probability of more than one photon being detected in a single bunch is extremely small. Therefore the measured efficiency, is an absolute efficiency appropriate only at low rates. At high photon rates a correction for multiple hits in a bunch must be made.

The number of photons detected is reduced when two or more photons are detected in a single bunch and are counted as only one. A further reduction occurs when a photon is detected, but there is also a hit in the veto paddle, thus killing the detected photon.

Therefore in order to calculate the correction at high rates the probability that there is a hit in the flux monitor veto paddle in a single bunch is also needed. In the flux monitor publication[2] it was suggested that this information can be obtained from the simulation. However, it is our experience that in the current location of the flux monitor, the background rate in the veto paddle is higher than that predicted by the simulation. Therefore it is necessary to measure the veto paddle counts during the experiment. i.e. the veto paddle hits needs to be counted by a live time gated scaler during each run.

We define,  $N_h = N_\gamma / f_m =$  the number of hits on the flux monitor that have the potential to be counted. So then,  $\mu_h = N_h / B =$  the average number of hits per bunch, where  $B$  is the number of accelerator bunches during the live time of a measurement. If  $P_h(x_h, \mu_h)$  is the probability of  $x_h$  hits in a bunch according to Poisson statistics, then since one or more hits will be counted as one, the probability of a count will be

$$P_h(x_h \geq 1, \mu_h) = 1 - P_h(0, \mu_h) = 1 - e^{-\mu_h}. \quad (12)$$

But this hit may be vetoed if another photon in the bunch causes a hit in the veto paddle. The probability that there is a hit in the veto paddle in a particular bunch is simply

$$P_v = N_v / B \quad (13)$$

where  $N_v$  is the number of veto hits counted during the live time. Therefore the total number of counts from the flux monitor is

$$\begin{aligned} N_m &= B\bar{P}_v P_h(x_h \geq 1, \mu_h) \\ &= B \left(1 - \frac{N_v}{B}\right) \left(1 - e^{-\frac{N_\gamma}{f_m B}}\right) \end{aligned} \quad (14)$$

Therefore, solving for  $N_\gamma$  we find

$$N_\gamma = f_m B \ln \left( \frac{B - N_v}{B - N_v - N_m} \right). \quad (15)$$

The error in  $N_\gamma$  depends on the uncertainties in  $f_m$ ,  $N_m$  and  $N_v$  in the usual way,

$$\delta N_\gamma^2 = \left( \frac{\partial N_\gamma}{\partial f_m} \right)^2 \delta f_m^2 + \left( \frac{\partial N_\gamma}{\partial N_m} \right)^2 \delta N_m^2 + \left( \frac{\partial N_\gamma}{\partial N_v} \right)^2 \delta N_v^2, \quad (16)$$

where it can be shown that

$$\begin{aligned} \frac{\partial N_\gamma}{\partial f_m} &= \frac{N_\gamma}{f_m}, \\ \frac{\partial N_\gamma}{\partial N_m} &= f_m B \frac{1}{(B - N_v - N_m)}, \text{ and} \\ \frac{\partial N_\gamma}{\partial N_v} &= f_m B \frac{N_m}{(B - N_v)(B - N_v - N_m)}. \end{aligned} \quad (17)$$

Once  $N_\gamma$  has been found then the number of gamma rays hitting the target may be found by

$$N_t = N_\gamma / C_B. \quad (18)$$

The correctness of equation 15 has been tested using the GEANT4 simulation. The simulation has the option of tracking multiple photons per GEANT event. The number of photons initiated in a GEANT event (or bunch) is calculated from Poisson statistics given an average photon rate specified by `setPhotonRate`, e.g.

```
/flux/gun/setPhotonRate 30000000.
```

sets the photon rate to  $3.0 \times 10^7$  Hz. When run, the simulation reports the number of 5-paddle flux monitor hits, the number of beam bunches, and the number of veto paddle hits during the live time. Also reported is the actual number of photons simulated. These numbers are entered into equation 15 to calculate the number of photons from the flux monitor output. The results for various photon rates are shown in figure 5. Also shown in the figure is the ratio if the number of photons is calculated using equation 11, i.e. with no rate correction. We see that the rate correction does a good job of calculating the correct number of photons. The simulation does not simulate additional background that may be present in the veto paddle beyond that created by the photon beam itself.

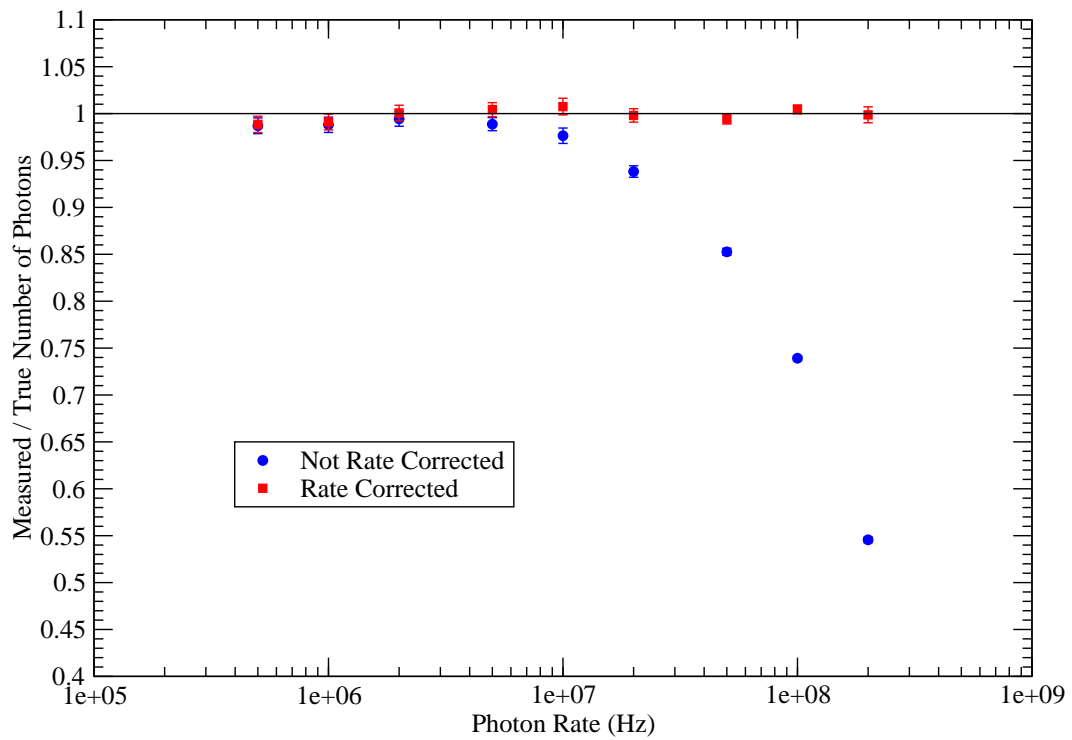


Figure 5: The ratio of the number of photons calculated using the flux monitor output divided by the true number of photons simulated by GEANT4.

## 5 Flux Monitor Check List

### 5.1 Calibrating the Flux Monitor

The following quantities need to be measured for several low photon flux rates.

1. Beam bunches in the live time.
2. Live time.
3. Flux monitor counts in the live time.
4. NaI spectrum in the live time (which will be integrated over a given energy range).
5. A background NaI spectrum (with beam off) and its live time.

Note: It may be more accurate to calculate the live time from the real time and the ratio of the beam bunches in the live time to the measured number of beam bunches in the real time.

### 5.2 Running an Experiment

The following quantities are needed to be able to make the rate correction and calculate the number of photons incident on the target.

1. Beam bunches in the live time.
2. Flux monitor counts in the live time.
3. Flux monitor veto paddle counts in the live time.

## 6 Using the Simulation

The current version of this simulation is `flux-1.6`. Get the package `flux-1.6.tgz` from <http://kanga.usask.ca/nucleus> under the "Software" tab.

Complete instructions on how to install the package is included in the `README` file included in the package.

The simulation makes use of the package `LightOutput` for calculating the light output response of the scintillators in the flux monitor. The latest version of this package is `LightOutput-1.3` which is also available from <http://kanga.usask.ca/nucleus> under the "Software" tab.

After installation the application `flux` should now be available to be run. Typing just

```
> flux
```

will start the application and the QT viewer.

The command `Set Geometry Target In` under `Run` will run the macro `setgeometry_target_in.mac` so that the geometry can be viewed. (The geometry included with the distribution is for the liquid helium target.) so that the geometry can be viewed. You can change which options are available in the QT viewer menu by editing the `gui.mac` file.

Under the `Help` tab in the QT viewer you can obtain descriptions of each of the commands available for controlling the simulation. There are many commands under `/flux/detector` which allow you to change many variables associated with the flux monitor, the NaI detector, and the target. There are also commands under `/flux/gun` which control factors defining the beam in a particular experiment. Commands under `/flux/counters` are more for debugging purposes.

Once you are satisfied that the geometry is correct you can run a calculation by typing

```
> flux <macro file>
```

The application will print the results after execution and also save the results by appending to the file `flux.rundata`.

There are several modes in which the GEANT4 simulation can be run.

### 6.1 Predicting the Calibration Factor

In this mode the target and the NaI detector should both be made not active to speed the simulation. i.e.

```
/flux/detector/setNaIActive false  
/flux/detector/setTargetActive false
```

The simulation should be set to simulate only one photon per GEANT4 event. i.e.

```
/flux/gun/setPhotonRate 0
```

About 10 million events should be run to obtain less than 1% accuracy.

## 6.2 Finding the Absorption Factors

The NaI should be made active, i.e.

```
/flux/detector/setNaIActive true
```

If the target was in place during the calibration measurement it must be made active, i.e.

```
/flux/detector/setTargetActive true
```

As well, its parameters (size, location, material) must be set to the values used. This may mean modifying the `DetectorDescription.cc` part of the code since, at present, only a solid cylindrical target is included with a limited number of materials defined. If the target was not in place during the calibration measurement it must be made not active.

```
/flux/detector/setTargetActive false
```

The simulation can run several photon beam energies at once. The default energy is 20 MeV, so if that is not one of the energies you want it must be removed from the energy list by

```
/flux/gun/removeEnergy 20 MeV
```

For each energy you want to run you must specify the energy, energy spread, and minimum energy needed at the target to be counted. These will be explained below.

The beam energy is set `/flux/gun/addEnergy`. For each energy you need to specify the energy spread. For the simulation this is defined as one-half the difference between the maximum and minimum energy that will pass through the collimator. The PERL script `higs_energy_spread.pl` is included with the GEANT4 package to allow this to be estimated given the accelerator settings and the collimator size. The energy spread should be specified in percent with the command `/flux/gun/setEnergySpread`.

The simulation will count the number of photons reaching the target. We only want to count the number of full-energy photons reaching the target since these are what is important for the experiment. There will be a few degraded energy photons reaching the target. Therefore only photons with energy above a defined minimum should be counted, this is the target integral. The minimum energy for this integral can be set by `/flux/setTargetMinimumEnergy`. This minimum should normally be set to just below the lowest energy allowed by the energy spread parameter. For example, in the September 2015 run at 61 MeV with a 1 inch diameter collimator, the energy spread is calculated to 6.78%. So the energy is set in the simulation with the commands

```
/flux/gun/addEnergy 61.0 MeV  
/flux/gun/setEnergySpread 6.78  
/flux/gun/setTargetMinimumEnergy 55 MeV
```

The NaI detector must be integrated and therefore the integration limits must be set. e.g.

```
/flux/detector/setNaIIntegrate_start 8 MeV  
/flux/detector/setNaIIntegrate_stop 19.5 MeV
```

The simulation should be set to simulate only one photon per GEANT4 event. i.e.

```
/flux/gun/setPhotonRate 0
```

Only about 40,000 events is needed to obtain sub-percent accuracy for the absorption factors.

The simulation will report the absorption factor  $C_A$ . If the target was set to active the simulation will also report the absorption factor  $C_B$ . If the target was not set to active a second simulation must be run with the target set to active to obtain  $C_B$ .

### 6.3 Simulating a High Photon Flux Rate

This may be done by setting the photon flux rate in Hz. e.g.

```
/flux/gun/setPhotonRate 30000000.
```

In this case both the NaI detector and the target may be set to be not active.



## References

- [1] Octavian Mavrighi M.Sc. Thesis *Photon Flux Monitor*, University of Saskatchewan, 2010.
- [2] R.E. Pywell, O. Mavrighi, W.A. Wurtz and R. Wilson, *Photon Flux Monitor for a Mono-energetic Gamma Ray Source*, Nuclear Instr. and Meth. A 606 (2009) 517
- [3] GEANT4 Collaboration, Nucl. Inst. and Meth. A 503 (2003) 250.
- [4] Rob Pywell, SPIR-144, *Calibration of the 5-paddle flux monitor for the  $^{209}\text{Bi}$  Compton Scattering Experiment*,  
Available at [http://nucleus.usask.ca/technical\\_reports/report\\_index.html](http://nucleus.usask.ca/technical_reports/report_index.html)