Skåne Automatic Monte Carlo
User manual

Rickard Cronholm (rickard.cronholm@skane.se)

release 0.5.2

Introduction

This user manual describes the installation, configuration and usage of a software for Monte Carlo simulation framework developed at Skåne university hospital for Python. Please report any problems with this software to *rickard.cronholm@skane.se*

This software uses *pydicom* [Mason] by Darcy Mason.

1 An automatic framework for Monte Carlo simulations

According to a recent publication, approximately 40 institutions are in the process of developing a system for Monte Carlo–based pre–treatment quality assurance [Popescu et al., 2015]. This project set out to build a system that is dynamical, flexible, lightweight, portable and requiring as few manual interactions as possible. The system designed and built under this project will hereafter be refereed to as SAMC (Skåne Automatic Monte Carlo).

Dynamical The wrokflow of SAMC enables linear accelerates of different manufacturers and types to be handled simultaneously within the same work flow, without requiring any changes.

Flexible Nothing in SAMC is hard coded. All major parameters are determined by the SAMC configuration file, which is a plain text file. Minor parameters are determined intrinsically at run time and are dependant only on the input data.

Lightweight The total file size of SAMC is on the order of kilobytes.

Portable SAMC is written eqclusively in python and as such is portable between different operating systems. The input required to initiate an instance of SAMC are pure DICOM files, which ensures that the system is portable between different treatment planning systems.

Table 1: The required python packages for SAMC.

Package	Version
numpy	$\geq$ 1.7.2
scipy	$\geq$ 0.9
pydicom	$\geq$ 0.9.8, <1.0[1]
progressBar	$\geq$ 2.3
psutil	$\geq$ 4.0.0
scikit-image	$\geq$ 0.12.3
Cython	$\geq$ 0.23
Six	$\geq$ 1.4
Matplotlib	$\geq$ 1.1.0
NetworkX	$\geq$ 1.8
Pillow	$\geq$ 1.7.8
dask[array]	$\geq$ 0.5.0

Manual interactions A system requiring only two manual interactions was built; export of DICOM files to SAMC and import to the TPS from SAMC. This can be acheived in various ways e.g. making the network containing the clinincal data available to SAMC. In this project a concious descision was made to not make the clinical data available to SAMC. Therefore the interactions in this project involves manually moving files between the TPS and SAMC.

1.1 Prerequirements of SAMC

SAMC generates input files for and executes simulations of EGSnrc [Kawrakow et al.] and its user codes [Rogers et al., a,b]. Therefore a requirement of SAMC is a working installation of EGSnrc (version $\geq$ V4-2.4.0). The simulation in SAMC starts from a pre generated phase space files scored above the first collimating device of the linear accelerator, therefore such a phase space is required. Moreover, SAMC requires a template input files for the part of the linear accelerator head below the $phsp_{aboveJaw}$ as well as for DOSXYZnrc.

SAMC requires python (version $\geq$ 2.6) and the packages listed in table 1.

1.2 Configuration of SAMC

The configuration of SAMC is divided in two parts as described in the following sections.

1.2.1 The SAMC configuration file

This plain text file contains all major parameters of SAMC such as file paths, calibration curve and binning schemes for CTC_auto, the number of worker threads to devide

[1]SAMC can be used with pydicom *geq*1.0 if the module is imported using: **import pydicom as dicom**.

simulations between etc.

1.2.2 The input file templates

SAMC treats every simulation as it were a a fully dynamic delivery, meaning allowing simultaneous movement of block collimators, MLC, gantry rotation, collimator rotation, table rotation and ISO–center displacement during delivery. Each of these movements are described in the Monte Carlo input files either by references to sequence files (see [Rogers et al., a]) or in plain text (see [Rogers et al., b]).

SAMC requires a template input file describing the geometry of the linear accelerator head below the $phsp_{aboveJaw}$. In the template file any input parameters related to dynamical information is replaced with a container that will be overwritten at run time, while keeping the constant information such as the materials used etc. This allows SAMC to facilitate simulations of linear accelerators of different manufacturers and types simultaneously within the framework.

1.2.3 The treatment machine specifc control file

This is a plain text files which, which must have the same file name as the DICOM tag TreatmentMachineName for the machine in question. The file contains the name of the defined treatment machine and its respecitve energies and fluence modes. For each energy and fluence mode a calibration factor converting from dose per incident particle to GyMU^{-1} is specified (cross ref to label seq:absCalib in MC_model_comm_valid.tex). Additionally a conversion factor for each treatment machine, energy, fluence mode and defined material is specified.

Monte Carlo reports dose to medium while TPS algorithms can report either dose to water or dose to medium [Knöös et al., 2006]. In order to allow an intercomparison between MC, reporting Dm, and a TPS, reporting Dw, a conversion method between them is needed. An analytical procedure based on Bragg–Gray cavity theory was proposed by Siebers et al. [2000]. They concluded that Spencer–Attix mass restricted collision stopping–power ratios can be used as conversion factors between the two quantities. Moreover, they concluded that a single correction factor can be used for each particular media throughout the eld for a given photon beam energy. The exception from this is for air, but as we are rarely interested in dose to air in clinical considerations and because that the conversion is done locally for each voxel this can be disregarded from. Rather than using the anaylitical approach suggested the EGSnrc code SPRRZnrc [Rogers et al., c] was used to compute Spencer–Attix mass restricted collision stopping–power for each media, linear accelerator and nominal beam energy used in the proeject. The conversion from dose to medium to dose to water for the beam quality, Q, and medium, m, can be conducted using the following equation:

$$D^w_{xyz} = D^m_{xyz} \times SPR^w_m\left(Q, m\right), \tag{1}$$

where D^w_{xyz} and D^m_{xyz} is the dose to voxel xyz in dosetowater and dosetomedium, respectively and $SPR^w_m\left(Q, m\right)$ is the SPRRZnrc simulated SpencerAttix mass restricted

collision stoppingpower ratio.

Additionaly parameters defined in the global *samc.conf* may be overwritten by defining them in the machine specific configuration file.

Given below is an exaple of a treatment machine specifc control file for a TrueBeam linear accelerator.

```
TreatmentMachineName TB01
TreatmentMachineType TB
absCalib 6X_STANDARD 1258215058492.0500
absCalib 10X_STANDARD 618247026345.6080
absCalib 6X_NON_STANDARD 3375593446091.8721
mlcLib BEAM_SYNCVMLC
templateType TB
template SYNCVMLC_TBTrueNDS
nLeafs 60
SCD 54.14956
SAD 100
leafRadius 8.00
physLeafOffset 0.0005816

Dose-to-media  to dose-to-water conversion factors
TB01 6X_STANDARD 170C700ICRU 0.0000
TB01 6X_STANDARD Rohacell51A 0.0000
TB01 6X_STANDARD AIR700OUT 0.0000
TB01 6X_STANDARD H2O700ICRU 1.0000
TB01 6X_STANDARD AU700ICRU 1.000
TB01 6X_STANDARD AIR521ICRU 1.1194
TB01 6X_STANDARD LUNG521ICRU 0.9980
```

The specific information include which MLC library to use during simulation, the specific template input file as well as data for the source–collimation–distance, source–axis–distance, the number of MLC leaf pairs etc.

1.3 Building blocks of SAMC

SAMC consists of a numbe of modules that are intended to be run sequentially. The connection between them is handled by the fact that the output a module becomes input for the next. In the following sections each module is described. The python dicom library pydicom [Mason] is used in SAMC.

1.3.1 DaemonA

DaemonA triggers on DICOM RT Plan files located in the directory as specified by the SAMC configuration file. The operations of daemonA are summarized below.

1. Checks that DICOM tag TreatmentMachineName of the DICOM RT Plan file correlates to one of the configured treatment machines in SAMC.

   If False: Termination

2. Generates and instance within SAMC that is given a unique timeStamp, which simply is the date and time of creation. Throughout its lifetime in SAMC the instance and its auxillary files will be identified by the timeStamp.

3. Determines which $phsp_{aboveJaw}$ to use depending on treatment machine name, nominal energy and fluence mode.

4. Determines the template input files to use as basis depending on treatment machine type and the treatment machine specifc control file.

5. Extract all plan specific data from the DICOM RT Plan file (e.g. MU per beam, beam limiting device positions per control point, gantry, collimator and couch angles per control point.).

6. Writes Monte Carlo input files based on templates and the plan specific data.

7. Complutes the calibration factor per simulation, which is the number of MU multiplied by the number of fractions, back scatter correction factor and the calibration factor for the treatment machine, energy and fluence mode.

8. Checks whether or not DICOM CT and DICOM RT Struct were exists fo the case in question.

   If True: Executes CTC_auto, which generates a Monte Carlo compliant phantom, writes the reference to it in the DOSXYZnrc input file.

   If False: Refeers to the precomputed Monte Carlo compliant phantom defined in the SAMC configuration file.

9. Generates a presimulation tar file for the timeStamp.

1.3.2 DaemonB

The presimulation tar file is untared and each file is copied to the path as defined in the SAMC configuration file. Initiates BEAMnrc simulation(s) using the number of workerthreads as specified by the SAMC configuration file.

1.3.3 DaemonC

The trigger of daemonC is completed BEAMnrc simulations that a) are located in a path defined by the SAMC configuration file and b) have a filename corresponding to a valid timeStamp. The operations of daemonC are summarized below.

1. Initiatie the fortran code *addphsp* that concatenates the IAEA phase space files generated by each workerthred to a single IAEA phase space file.

2. Verifies the checksum of the IAEA phase space file

   If False Reinitiates the BEAMnrc simulation and terminates.

3. Computes the number of particles required to simulate in DOSXYZNrc in order to obtain a defined statistical uncertainty.

4. Initiates DOSXYZnrc simulation(s) using the number of workerthreads as specified by the SAMC configuration file.

1.3.4 DaemonD

The trigger of dameonD is 3ddose files resulting from DOSXYZnrc simulations that a) are located in the path defined by the SAMC configuration file and b) have a filename corresponding to a valid timeStamp. If all DOSXYZnrc simulations of the timeStamp instance are finalized deamonD cleans the systems form all auxillary files (e.g. BEAMnrc input files, phase space files etc) and creates a post simulation tar file, named in accordance with the timeStamp instance, containing the 3ddose file(s).

1.3.5 DaemonE

The trigger of daemonE is post simulation tar files as created by daemonD. The operations of daemonE are

1. Untar the post simulation tar file.

2. Generate two new DICOM RT Plan file by copying the one exported from the TPS and replacing the DICOM tag SOPInstanceUID.

3. For each 3ddose file:

   (a) Match it to the corresponding TPS exported DICOM RT Dose file.

   (b) Convert to dose in Gy, using the factor derived in daemonA.

   (c) Generate a new dose matrix, where the dose to medium as reported by Monte Carlo is converted to dose to water.

   (d) Crop the dose matrices to the shape of the original DICOM RT Dose.

   (e) Rotate the dose matrices so that they are in accordance with the DICOM coordinate system for the patient orientation in question.

   (f) Write DICOM RT Dose files (both in dose to medium and dose to water) using a copy of the TPS exported DICOM RT Dose file while updating SOPInstanceUID and references to the corresponding SAMC generated DICOM RT Plan file.

Table 2: The binning scheme for voxels located inside the external body outline.

Media name	Upper CT number boundary	Corresponding density [gcm^{-3}]
AIR521ICRU	50	0.0324
LUNG521ICRU	460	0.4571
adipose	1000	1.0252
muscle	1100	1.0926
bone1	1400	1.2946
bone2	1700	1.4967
bone3	2000	1.6987
bone4	2300	1.9008
bone5	2600	2.1028

By maintaining all DICOM tag data apart from the SOPInstance UID, and references to it, the SAMC generated DICOM files will be directly importable to the treatment planning system. Moreover, they will automatically connect to the proper patient, study and series. This means that the DICOM RT plan and RT Dose will be connected to the primary image and structure set as well as all other image and structe sets that are regeistered to them. Thus the MC computed dose can be superimposed on any image data registered to the series of the patient in question.

1.3.6 CTC_auto

Although not inherently a module of SAMC, a description of CTC_auto is given here since it is used by SAMC as a recource. The objective of CTC_auto is to generate a Monte Carlo compliant voxalized phantom based on the DICOM data. CTC_auto is an based on CTC_ask [Ottosson and Behrens, 2011], with the fundamental differences that it is written in python, is interaction–free and inclueds the treatment couch if it is included as a structure in the DICOM RT Struct file.

1.3.6.1 Configuration of CTC_auto

CTC_auto requires a calibration curve between CT pixel value and physical density (figure 1) and a binning scheme for each structure type. The binning scheme used in this project, for voxels located inside the external body outline, is given in table 2. Both input types are described in plain text files.

1.3.6.2 CTC_auto workflow

The input to CTC_auto are DICOM RT Plan, DICOM RT Dose, DICOM RT Struct and DICOM CT files. The use of each respective file type is described below.

DICOM RT Plan The Image Patient Orientation tag is used to rotate the CT data

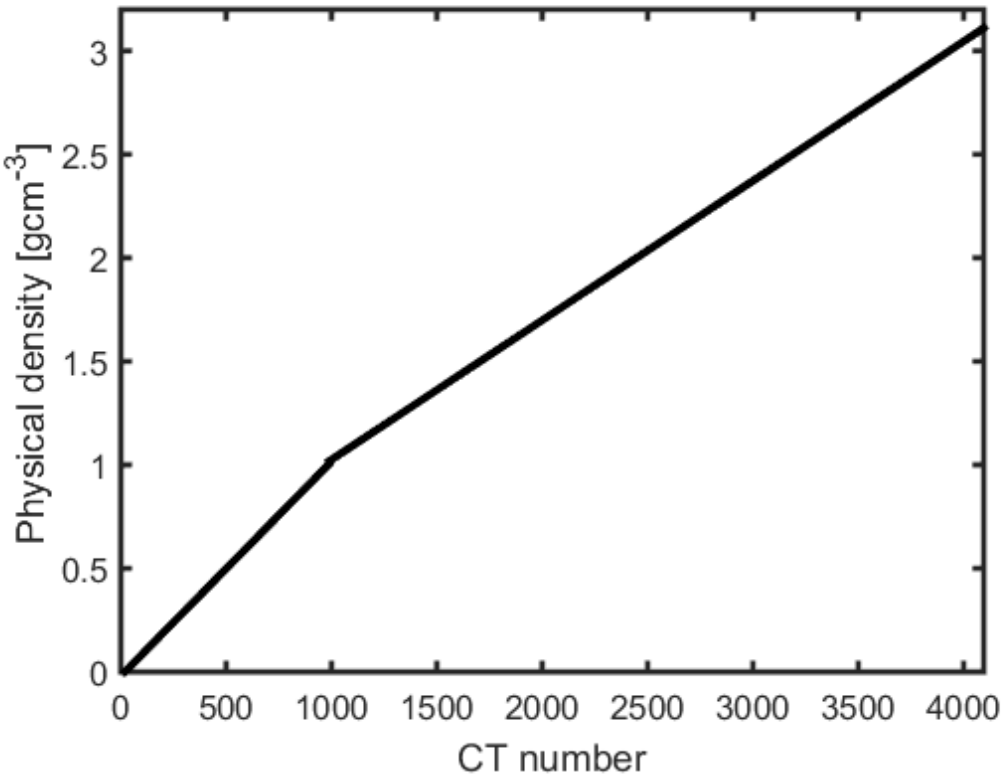

Figure 1: The calibration curve between CT pixel value and physical density used in this project.

matrix in order to be in compliance with the coordinate system of DOSXYZnrc [Rogers et al., b].

DICOM RT Dose The grid of the dose matrix is used as a base for the resulting phantom. This ensures that the DICOM RT Dose files generated by SAMC are on the same coordinates as the ones exported from TPS. This ensures that no interpolation is required in order to compare dose matrices between the two ssytems.

DICOM RT Struct The countours are used to define regions of the CT matrix that obey to different binning schemes depending on the type of the structure. For instance, pixels outsid of the external contour of a patient will obey to different binning schemes than those inside the external contour. This is highly flexible in CTC_auto and each structure type can be assigned a unique binning scheme.

DICOM CT The pixel data is converted density and matrices using calibraion curves and binning schemes respectively.

The pixel data of each CT file is read, sorted according to slice location and appended

to a CT matrix. The grid of the DICOM RT Dose file is read and if required expanded in order to include structure outlines of any requested structure types (e.g. the treatment couch, the external body outline etc.). The dose grid is expanded by maintaining the original coordinates. The CT matrix is interpolated onto the dose grid using a slice–wise bicubic interpolation. The CT matrix is converted to a density[2] matrix using a global calibration curve. Each outline in the DICOM RT Struct file of a type of interest (this is user definable) is converted to a boolean matrix of the same shape as the dose grid with each element is set to *True* or *False* depending on wheather or not the voxel is encompassed by the structure in question. A user defined list of priority is employed in order to ensure that each voxel belongs to only one strucutre. The CT matrix is converted to a material matrix per structure using the user defined binning scheme per structure type. A final material matrix is computed by concatenating the material matrix for each structure. The Image Patient Orientation tag of the DICOM RT Plan file is used to rotate the density and material matrices so that they are in compliance with the coordinate system of DOSXYZnrc. Finally the coordinates, material and density matrices are written as an Monte Carlo compliant phantom [Rogers et al., b].

The coordinates of the DICOM RT Dose grid is written to a file, which is used by daemonE (section 1.3.5) to crop the Monte Carlo generated 3ddose file to the shape of the original DICOM RT Dose.

Inclusion of the treatment couch and the ability to handle any possible patient orientation are both unprecedented features of a code converting DICOM data to phantoms to be used in Monte Carlo simulations.

1.4 Acheiving automation

Each module of SAMC can be run manually as well as independently of the other modules. Acheiving automation of SAMC is dependandt on two criteria i) that each module when executed leaves a trigger for the next module and ii) that the modules are executed automatically by the system. The first criterion is met by design of the code of each module, while the second is met by the use of *cron* under a Linux operating system. Cron is a daemon that executes scheduled. Commands executing each module are added to the crontables of the user under which SAMC is installed.

1.5 Manual execution

Each module can be executed manually by the following command:

```
./daemonX.sh
```

where X is substituted for the deamon denomination [A-E] in question.

[2]It should be noted that Monte Carlo uses the physical density as oppose to many TPS that use the relative electron density.

References

D. Mason. pydicoml. https://github.com/darcymason/pydicom. Accessed: 2016-03-10.

I. A. Popescu, P. Atwal, J. Lobo, J. Lucido, and B. M. C. McCurdy. Patient-specific qa using 4d monte carlo phase space predictions and epid dosimetry. *Journal of Physics: Conference Series*, 573(1):012004, 2015. URL http://stacks.iop.org/1742-6596/573/i=1/a=012004.

I. Kawrakow, E. Mainegra-Hing, D. W. O. Rogers, F. Tessier, and B. R. B. Walters. The EGSnrc Code System: Monte Carlo Simulation of Electron and Photon Transport. http://nrc-cnrc.github.io/EGSnrc/doc/pirs701-egsnrc.pdf. Accessed: 2016-03-10.

D. W. O. Rogers, B. R. B. Walters, and I. Kawrakow. BEAMnrc Users Manual. http://nrc-cnrc.github.io/EGSnrc/doc/pirs509a-beamnrc.pdf, a. Accessed: 2016-03-10.

D. W. O. Rogers, B. R. B. Walters, and I. Kawrakow. DOSXYZnrc Users Manual. http://nrc-cnrc.github.io/EGSnrc/doc/pirs794-dosxyznrc.pdf, b. Accessed: 2016-03-10.

T. Knöös, E. Wieslander, L. Cozzi, C. Brink, A. Fogliata, D. Albers, H. Nystrom, and S. Lassen. Comparison of dose calculation algorithms for treatment planning in external photon beam therapy for clinical situations. *Phys Med Biol*, 51(22):5785–5807, Nov 2006.

J. V. Siebers, P. J. Keall, A. E. Nahum, and R. Mohan. Converting absorbed dose to medium to absorbed dose to water for Monte Carlo based photon beam dose calculations. *Phys Med Biol*, 45(4):983–995, Apr 2000.

D. W. O. Rogers, I. Kawrakow, J. P. Seuntjens, E. Mainegra-Hing, and B. R. B. Walters. NRC User Codes for EGSnrc. http://nrc-cnrc.github.io/EGSnrc/doc/pirs702-egsnrc-codes.pdf, c. Accessed: 2016-03-10.

R. O. Ottosson and C. F. Behrens. CTC-ask: a new algorithm for conversion of CT numbers to tissue parameters for Monte Carlo dose calculations applying DICOM RS knowledge. *Phys Med Biol*, 56(22):N263–274, Nov 2011.