Proceedings of the Twenty-Fifth EGS Users' Meeting in Japan

August 6 - 7, 2018.

KEK, Tsukuba, Japan

Edited by

Y. Namito, H.Iwase and H. Hirayama

High Energy Accelerator Research Organization

FOREWARD

The Twenty-fifth EGS Users' Meeting in Japan was held at High Energy Accelerator Research Organization (KEK) from August 6 to 7. The meeting has been hosted by the Radiation Science Center. More than 30 participants attended the meeting.

The meeting was divided into two parts. Short course on EGS was held at the first half of the workshop using EGS5 code. In the later half, 5 talks related EGS were presented. The talk covered the wide fields, like the medical application and the calculation of various detector responses *etc.* These talks were very useful to exchange the information between the researchers in the different fields.

Finally, we would like to express our great appreciation to all authors who have prepared manuscript quickly for the publication of this proceedings.

Yoshihito Namito
Hiroshi Iwase
Hideo Hirayama
Radiation Science Center
KEK, High Energy Accelerator Research Organization

CONTENTS

Running EGS5 with the computer cluster PIs

S. TSUJI

Kawasaki Medical School, Kurashiki 701-0192, Japan

Abstract

Combining 24 units of single board computers Banana Pi and Raspberry Pi together to create a computer cluster "parallel integrated system PIs". I ran the Monte Carlo simulation EGS5 with this PIs. Describe concrete construction method, computing capacity and problems of this PIs.

1   Introduction

There are two single board computers Raspberry Pi and Banana Pi. The model of Raspberry Pi is Raspberry Pi 3 Model B, this specification is the ARM Cortex-A 53 of 4 core CPU, 1.0 GB memory. Banana Pi's model is Banana Pi BPI-M3, this specification is the A83T ARM Cortex A7 of 8 core CPU, 2.0 GB memory and 8 GB eMMC storage. They can be loaded with Linux. For the distribution of Linux, Raspberry Pi is RASPBIAN STRETCH WITH DESK TOP, Banana Pi is Ubuntu 16.04 mate. They can be installed the latest version of gfortran and run simulation of EGS5. In addition, it is possible to install the MPI and run EGS5 at the MPI in a single board computer [1]. I have developed a computer cluster "parallel integrated system" (hereafter PIs) by combining these single board computers (1 Raspberry Pi and 23 Banana Pies) [2]. I concretely describe the construction method of PIs, computational capacity and a problem.

2   Construction of PIs

2.1   System diagram

The host name of one Raspberry Pi node is "raspberrypi24". Also, Banana Pi nodes allocate host names from "bananapi00" to "bananapi22". Fig. 1 is a system diagram of PIs. The raspberrypi24 node is used for exchange with outside via network and Banana Pi nodes (bananapi00~bananapi22) are used for calculation. "/home/pi" directory of each node is made common by NFS (Network File System). Time is set by NTP (Network Time Protocol). The raspberrypi24 node is synchronized with the outside and the bananapi00 ~ bananapi22 nodes are synchronized with the raspberrypi24 node.

2.2   Setting of each node (single board computer)

Download each node OS image and burning them on microSD card. Since the Banana Pi node is equipped with 8 GB eMMC, when used as a slave node, burning to eMMC is sufficient. After installing the OS, it is necessary to install the latest version of gfortran and MPICH [3] on each node. Details are listed in reference [1].

2.3   Assembling PIs

When full calculating a single board computer, the chips get extremely hot. As shown in the Fig. 2, it is necessary to put a heat sink on the main chip of each board of PIs. Furthermore, the whole device must be cooled with a fan (shown in Fig. 4). As shown in the Fig. 3, 12 boards are connected
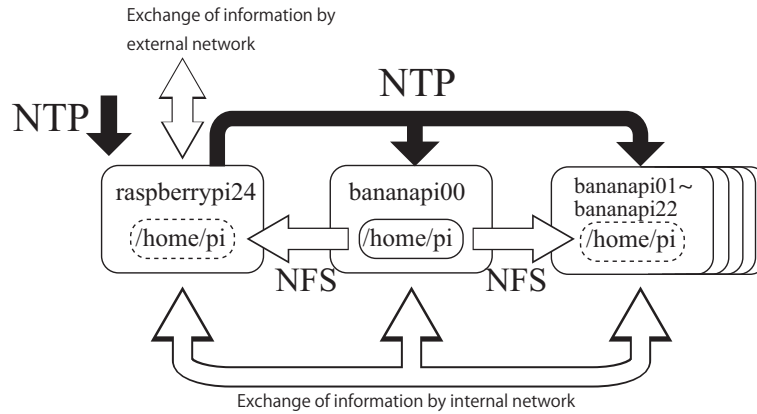
Figure 1: The system diagram of PIs. The "/home/pi" directory of the host node bananapi00 is made common to other nodes by NTP. Only the raspberrypi24 node is connected to the external network, and this node is accessed to upload and download files. The raspberrypi24 node synchronizes the time with the outside by NTP, but the other nodes synchronize with the raspberrypi24 via the internal network.

by spacers. PIs is configured with 2×12 boards, 4 fans (2 per 12 boards), a 24 port switching hub, 5 V USB chargers, power strips, power cables and LAN cables shown in Fig. 4. A program for the
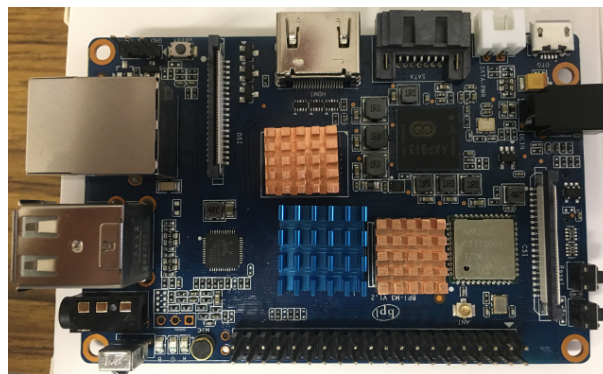
Figure 2: The single board computer of the Banana Pi BPI-M3. Put heat sinks on the main chips for heat protection.

EGS5 code was run without heat sinks and fans, it took 7.45 hours. However, running the same program with seat heat sinks and fans installed, it took only 1.61 hours. In this way, considering the cooling of the single board computer, the computing capacity is greatly increased.

2.4 Network settings

For each node from bananapi00 to bananapi22, internal network addresses 192.168.240.100 to 192.168.240.122 are allocated. The raspberrypi24 node is assigned 192.168.240.124. Specifically, it is set as follows (in the case of bananapi00).

IP address setting(`/etc/NetworkManager/system-connection/Wired connection 1`).

```
...............
[ipv4]
```

Figure 3: 12 single board computers are connected in one tower. The configuration of PIs is prepared 1 tower of 12 Banana Pi nodes and the other tower of 11 Banana Pi + 1 Raspberry Pi nodes.

Figure 4: Overall picture of PIs. PIs is configured with 2 towers in Fig. 3, 4 fans (2 per 12 boards), a 24 port switching hub, 5 V USB chargers, power strips, power cables and LAN cables. Fans are necessary for cooling the entire device.

```
address1=192.168.240.100/24,192.168.240.254
dns=8.8.8.8;
dns-search=
methods=
...............
```

Host name setting(`/etc/hostname`).

```
bananapi00
```

Setting the host name in the internal network(`/etc/hosts`).

```
...............
192.168.240.100    bananapi00
192.168.240.101    bananapi01
...............
192.168.240.121    bananapi21
192.168.240.122    bananapi22
192.168.240.124    raspberrypi24
```

The setting of raspberrypi24 is shown below.

IP address setting(`/etc/dhcpcd.conf`).

```
.................
interface enxb***************
static ip_address=192.168.240.124
static routers=192.168.240.254
static domain_name_servers=8.8.8.8
static domain_search=

interface wlan0
static ip_address=???.???.???.???
static routers=???.???.???.???
static domain_name_servers=???.???.???.???
static domain_search=
.................
```

Host name setting(`/etc/hostname`).

```
raspberrypi24
```

`/etc/hosts` is the same as that of the Banana Pi node.

A summary of the internal network is shown in Table 1.

Table 1: The internal network address of PIs.

IP address	bananapi00:	192.169.240.100
	bananapi01:	192.168.240.101
		⟨
	bananapi22:	192.168.240.122
	raspberrypi24:	192.168.240.124
Net mask		255.255.255.0
Gateway address		192.168.240.254
Domain name server		8.8.8.8

2.5 Installing and configuring NSF

When using MPI with multiple nodes, NFS and ssh are needed to install and configure. Describe the installation and configuration of NFS (Network File System) [4]. Let bananapi00 be the master node and the other node be the slave node. Set the user name of all nodes to "pi".

The master node setting

Execute the following command in superuser mode.

```
# apt-get install nfs-kernel-server
# apt-get install nfs-common portmap rpcbind
```

After installation, these daemons start up on Linux on each node by executing the following command.

```
# service rpcbind start
# service nfs-common start
# service nfs-kernel-server start
```

To start up automatically at OS startup, execute the following command.

```
# update-rc.d nfs-kernel-server enable
# update-rc.d nfs-common enable
# update-rc.d rpcbind enable
```

Add /etc/exportfs as follows.

/etc/exportfs setting.

```
...............
/home    192.268.240.0/255.255.255.0(rw,sync,no_root_squash,no_subtree_check)
...............
```

If the following command is executed on the master node, it is successful if "pi" is visible under /mnt directory.

```
# exportfs -av
# mount bananapi00:/home    /mnt
```

The slave node setting

Install softwares related NFS and run as a daemon.

```
# apt-get install nfs-common nfs-common rpcbind
# service rpcbind start
# service nfs-common start
# update-rc.d rpcbind enable
# update-rc.d nfs-common enable
```

Add /etc/fstab as follows.

/etc/fstab setting.

```
...............
bananapi00:/home    /mnt/pi_home nfs noatime 0  0
...............
```

If the slave node has a user name "pi", it is necessary to change. Execute the following command in superuser mode.

```
# mv /home/pi /home/pi.tmp
# ln -s /mnt/pi_home/pi   /home/pi
```

After rebooting, if there are the same files or directories as the master node in /home/pi of the slave node, it is successful.

2.6 Configuring ssh

When installing the distribution Ubuntu 16.04 mate or RASPBIAN STRETCH WITH DESK TOP, ssh (Secure Shell) has already been installed. When MPI is used with multiple nodes, it is necessary to log in without a password. After setting NTP, set as follows at the master node (bananapi00).

```
$ ssh-keygen -t rsa
$ cat /home/pi/.ssh/id_rsa.pub >> /home/pi/.ssh/authorized_keys
$ chmod 600 /home/pi/.ssh/authorized_keys
```

As a result, the public key and secret key are distributed to all the nodes. Also edit `/etc/ssh/sshd_config` in the all nodes as follows.

`/etc/ssh/sshd_config` setting.

```
...............
AuthorizedkeysFile %h/.ssh/authorized_keys
...............
PasswordAuthentication yes
...............
```

After rebooting, if you can execute the following command and login without password, it is successful.

```
$ ssh pi@bananapi?? (Enter 01 to 22 in ??.)
```
or
```
$ ssh pi@raspberrypi24
```

2.7 Setting NTP

Set the time server in bananapi nodes to raspberrypi24. Edit `/etc/ntp.conf` of bananapi nodes as follows.
```
server 192.168.240.124  iburst
```

2.8 Setting of the host file for MPI

When using MPICH with multiple nodes, prepare the host file for MPI. Write the following contents in `/home/pi/mpihosts/hosts`.

```
bananapi00
bananapi01
bananapi02
...............
bananapi22
```

It is also possible to limit the number of cores of each node by MPI calculation.

`/home/pi/mpihosts/hosts` setting.

```
bananapi00:6
bananapi01:8
bananapi02:3
...............
bananapi22:5
```

Add the following line to `.bashrc`.

```
export HYDRA_HOST_FILE=~/mpihosts/hosts
```
After setting all, compile from the host bananapi00 by `mpicc`, `mpifort` etc. Execute the following command.

```
mpirun -np 184 execute_file
```
The number 184 means 23 nodes × 8 cores from bananapi00 to bananapi22 nodes.

2.9   Installation and configuration of EGS5 and EGS5-MPI

Installing EGS5 [5] and EGS5-MPI [6] in the home directory of bananapi00 install them on all nodes. Rewrite `egs5run` as follows.

```
#!/bin/bash
..............

BASKET=directory of egs5
MY_MACHINE=gfort
..............
```

`egs5mpirun` is rewritten as follows.

```
#!/bin/bash
..............

BASKET=directory of egs5
MPI_BASKET=directory of egs5mpi
MY_MACHINE=MPICH
OPT_LEVEL=O2
MPI_SIZE=184
..............

elif test "$MY_MACHINE" = "MPICH" <-- change where "Cygwin MPICH2" is written
  ............
CFLAGS="-fno-automatic -finit-local-zero -fno-range-check"
```

With the above setting, EGS5-MPI (`egs5mpirun`) can be executed.

3   Performance of PIs

3.1   Remote operation from the network

The raspberrypi24 node in PIs has an IP address set for Wi-Fi exchanges with the outside so that commands can be executed and files are downloaded and uploaded through the network. The Fig. 5 shows that Tera Term program (ssh) in Windows can manipulate PIs commands.

3.2   Computational power

The computing power of PIs was examined compared with the PC (1 core in CPU). Specification of the PC has Core i7 6700K (4.5 GHz), 32 GB memory. The PC was started VMware on Windows
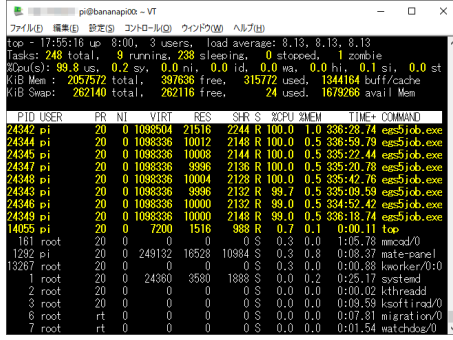
Figure 5: Tera Term (ssh) in Windows accesses PIs. The execution of parallel calculation is checked.

10 pro, and then run Scientific Linux on it. The program for PC and PIs uses the dose calculation for brachytherapy with mHDR-v2 source. Conditions for calculation of absorbed dose is 2.5 to 20 cm (bin width: 0.1 mm (distance<1.0 cm), 1 mm (distance $\leq$1.0 cm)) in the distance from the source and from 0 to 180 degrees (bin width: 0.2 degree) in the angle in the 40 cm radius water sphere phantom. The PC took 74.6 hours for 5.6×10^8 event calculation processing. From these
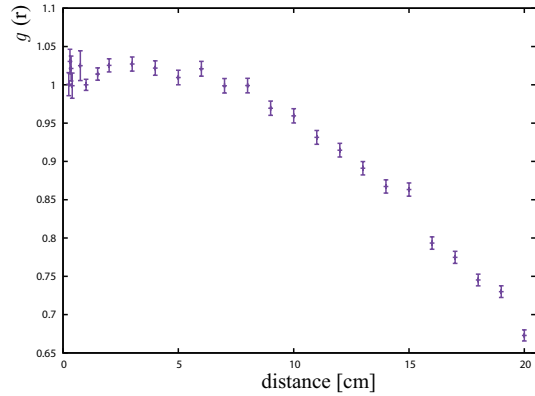

Figure 6: Radial dose function $g(r)$ obtained on PC (Core i7 6700 K (4.5 GHz), 32 GB memory). It takes 74.6 hours to calculate 5.6×10^8 events.
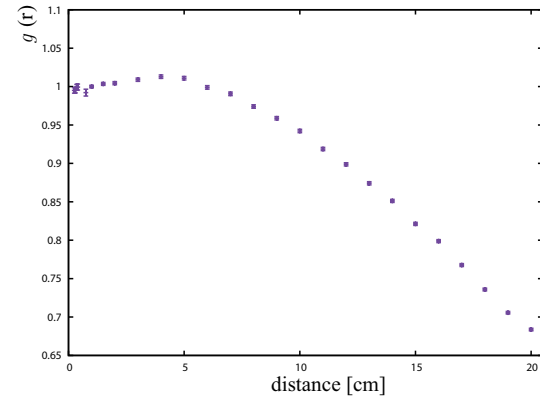

Figure 7: Radial dose function $g(r)$ obtained on PIs with 20 Banana Pi nodes. It takes 78.8 hours to calculate 1.0×10^{10} events.

results, the result of converting to radial dose function $g(r)$ [7, 8] in TG-43U1 [8] protocol is shown in the Fig. 6. On the other hand, in PIs using Banana Pi 20 nodes it took 78.8 hours to process 1.0×10^{10} events. The results of converting these results into $g(r)$ are shown in the Fig. 7. The computing capacity of PIs with 20 nodes is 16.9 times as compared with PC (1 core). If you let PC calculate the same 1.0×10^{10} events as PIs, it takes 55.5 days (1332 hours). From this result, when 23 Banana Pi nodes in PIs are used, the computing capacity can be estimated to be 19.4 times as compared with PC (1 core).

3.3 Power consumption

The total power consumption of PIs using 20 nodes was 85 W in the idling state and 153 W during the MPI calculation. The power consumption of PIs when 23 nodes were used was 93 W in the idling state and 173 W in the MPI calculation.

4 Problem of PIs

The problem with PIs is that the communication on the internal network expires. If you are doing calculations for several days, you may get the following error.

```
$ packet_write_wait: connection to IP_ADDRESS port 22:Broken pipe
```

In this case, calculation of the `IP_ADDRESS` node is stopped. This is because ssh is disconnected during program execution. To prevent disconnection of ssh, the following description can be done.

`/home/pi/.ssh/config` setting.

```
ServerAliveInterval    100
ServerAliveCountMax    8640
```

This responds to the server 8640 times every 100 seconds. That is, keep connecting the network for 10 days. However even with the above settings, when computing using the Banana Pi node, if more than 20 nodes are used, the internal network is disconnected per about 2 or 3 days. In assembling and running the cluster computer in Banana Pi,

- 20 or less Banana Pi nodes should be used,

- calculation duration should be around 3 days,

are desirable. Now PIs with 20 Banana Pi nodes for calculations are running stable.

5 Conclusion

In this research, I developed a computer cluster PIs by combining a single board computer, and actually ran EGS5-MPI on it. In development of PIs, concrete settings for network setting, NTP, ssh are indicated. Compared to the PC of core i7 (4.5 GHz), the calculation capability of PIs is 16.9 times at 20 nodes. In addition, the power consumption with 20 nodes was 153 W at the time of MPI calculation. The problem is that the network is disconnected during calculation, but it can be operated stably by using a maximum of 20 nodes and setting the calculation duration to about 3 days. Considering from the power consumption on the computing capacity, it is very economical.

References

[1] S.Tsuji, "Running EGS5 with Raspberry Pi," Proseedings of the Twenty-Fourth EGS Users' Meeting in Japan 4:1-6 (2017)

[2] S.Tsuji, "Development of computer cluster "Parallel integrated system"," Kawasaki Ikaishi Arts & Sci (43) : 37-48 (2017) [in Japanese]

[3] https://www.mpich.org/

[4] https://ameblo.jp/takeoka/entry-11543214113.html [in Japanese]

[5] H. Hirayama, Y. Namito, A.F. Bielajew, S. J. Wilderman and W. R. Nelson, "The EGS5 Code System," SLAC-R-730 (2005) and KEK Report 2005-8 (2005), http://rcwww.kek.jp/research/egs/egs5.html

[6] M.Shimizu, EGS5-MPI, https://unit.aist.go.jp/rima/ioniz-rad/egs5mpi/index_eng.html, https://unit.aist.go.jp/rima/ioniz-rad/egs5mpi/doc/egs5mpi_manual_eng.pdf.

[7] S.Tsuji, N.Narihiro, "ABOUT $g(r)$ FUNCTION IN AAPM TG-43," Proseedings of the Eighteenth EGS Users' Meeting in Japan 6:36-42 (2011)

[8] M. J. Rivard, B. M. Coursey, L. A. DeWerd, W. F. Hanson, M. S. Huq, G. S. Ibbott, M. G. Mitch, R. Nath, and J. F. Williamson, "Update of AAPM Task Group No.43 Report: A revised AAPM protocol for brachytherapy dose calculations," Med. Phys. **31**,633-673 (2004)

The method of obtaining point kernels for use in the convolution dose calculation in radiation therapy.

K. Masaki[1], S. Terashima[2], H. Endo[1], S. Komori[1], T. Kato[1]

[1]*Department of Radiation Physics and Technology,*
Southern Tohoku Proton Therapy Center, Koriyama 963-8052, Japan.
[2] *Department of Radiation Science, Hirosaki University Graduate School of*
Health Sciences, Hirosaki 036-8564, Japan.
e-mail: wurstp38@yahoo.co.jp

Abstract

Monte Carlo (MC) method is a useful dose calculation algorithm that can calculate dose distributions most accurately even in the complex geometry. However, due to the long computation time of MC method, it is not used clinically in most cases. In the present study, we developed a method of obtaining point kernels by using EGS5. The obtained kernels can be used in convolution dose calculation algorithms, and it is possible to shorten the computation time of dose calculations compared with that of using MC method. The point kernels were obtained from MC simulations with a forced interaction technique. The dose distribution of one monoenergetic primary photon interaction was calculated and recorded as a point kernel. The monoenergetic kernels were simulated with energies from 0.5 MeV to 6.5 MeV at 0.5 MeV intervals. The obtained kernels were used in the convolution dose calculation. The dose distribution in water for basic situation was calculated by the fast Fourier transform convolution algorithm and was compared with the measurement data of clinical machine commissioning. The calculated percentage depth dose agreed with measured values within 1 mm distance-to-agreement (DTA) value in the build-up region and within 1% error in the depth beyond the build-up region. Also, the calculated off-center ratio agreed with measured values within 2% errors in the radiation fields, within 1 mm DTA value in the penumbra region, and within 3% errors outside the radiation fields. The convolution dose calculation was performed with high calculation accuracy, and it was confirmed that the obtained point kernels can be used in dose calculations.

1. Introduction

Medical linear accelerator (linac), one of the most widely used radiation therapy devices, can perform highly flexible cancer treatment. Before the treatment, dose distributions in the patient are calculated to develop optimal treatment plans for each patient. Monte Carlo (MC) method, the most accurate dose calculation algorithm in particular in the inhomogeneous medium such as a human body, is recognized as a gold standard dose calculation algorithm[1]. However, due to the long computation time of MC method, full-MC dose calculation is not used clinically in most cases. A calculation algorithm with high calculation accuracy and short computation time has been required in clinical situations.

At present, convolution algorithm, which aims to shorten the computation time by diverting the MC simulation result called "dose kernel", is widely used. Convolution algorithm requires two components. One is total energy released per unit mass (TERMA), which represents the intensity of the energy imparted to the medium by interactions of primary photons. The other is a dose kernel, which represents the energy deposited about a primary photon interaction site. Therefore, convolution dose calculation is performed according to the following equation,

$$D = T \otimes K \quad \cdots \quad (1)$$

where D is the dose, T is the TERMA, and K is the dose kernel. Also, TERMA is calculated by the following equation,

$$T = \sum_{n=1}^{N} \Psi_n \times \left(\frac{\mu}{\rho}\right)_n \quad \cdots \quad (2)$$

where T is the TERMA, Ψ is the energy fluence, μ is the linear attenuation coefficient, ρ is the mass density, and n is the number of energy bins.

There are several variations of the convolution algorithms depending on the types of dose kernels. For example, point kernel convolution algorithm uses a dose kernel that consists of dose depositions of one primary photon interaction, and pencil beam convolution algorithm uses a dose kernel that consists of all interactions caused by one pencil beam. Generally, the amount of information contained in the dose kernels affects calculation accuracy and computation time of convolution algorithms.

As previously mentioned, dose kernels can be obtained from MC simulations[2]. The simulation to obtain pencil beam kernels is simple. Primary photons are generated on

the surface of a semi-infinite water phantom, and the energy deposited within each voxel in the array is calculated. However, in the simulation to obtain the point kernels, it is necessary that the interactions of primary photons occurred at the appropriate point within the array. A forced interaction technique[3] is needed; however, the specific method varies depending on the MC codes.

The aim of this study is to develop a method of obtaining point kernels by using the EGS5 MC code. The validity of obtained point kernels was confirmed by using them in the convolution dose calculation. The dose distribution in water for basic situation was calculated and was compared with the measurement data.

2. Materials and methods

2.1 MC simulations to obtain point kernels.

The point kernel contains the information of deposited energy when the primary photon interactions occurred at an appropriate point. Since the forced interaction technique must be used only for primary photons, primary photons and the others were discriminated using the latch variable, which is a user-definable particle parameter. When any interaction of primary photons occurred, the value of latch variable was added 1 in the subroutine "ausgab". Therefore, it is possible to discriminate between primary photons and others depending on whether the latch variable value of each photon is 0 or not.

The dpmfp variable, which is the distance traveled by photons until the interaction occurs, is determined using random numbers in the subroutine "photon". We edited the subroutine "photon" so that this distance for primary photons always becomes 1×10^{-6}, which is the minimum value that can be set in EGS5 code, in order to make them interact simultaneously with incidence. Since the latch variable value of primary photons was 0, the following code was inserted after the dpmfp variable was determined by random numbers.

```
if(latch(np).eq.0) then
    dpmfp = 1.E-6
endif
```

The MC simulations to obtain point kernels were performed in the water phantom. The phantom size must be large enough not to disturb the charged particle equilibrium. Therefore, the phantom size was set to 25 cm to the front, 10 cm to the back, and 20 cm to the other directions from the incident point of primary photons. The incident photon

energy was changed from 0.5 MeV to 6.5 MeV at 0.5 MeV intervals. The deposited energy was recorded as a three-dimensional (3D) distribution. The scoring array was a Cartesian coordinate system, and its grid size was $2.5 \times 2.5 \times 2.5$ mm³. The history number was 5×10^7 for all simulations. The cutoff energy value of AP was 0.010 MeV, and that of AE was 0.521 MeV.

2.2 Calculating dose distribution in water by the convolution algorithm.

The dose distribution in water for basic situation was calculated by the convolution algorithm. The 6 MV X-ray beam from the TrueBeam (Varian Medical Systems, Palo Alto, CA) linac was used. The size of the water phantom was $30 \times 30 \times 30$ cm³, and its surface was positioned at the source to surface distance (SSD) of 100 cm. The radiation field was shaped by the jaw collimators, and its size was set to 10×10 cm² at the isocenter plane. The calculation grid size was $2.5 \times 2.5 \times 2.5$ mm³.

The phase space data (PSD)[4], which is the information of the radiation source, was obtained from the MC simulation in the linac head[5]. The details of the simulation were reported in the previous study[6]. The fluence map of the beam was analyzed from the PSD, and it is shown in the Fig. 1. TERMA and the weight of energy components of kernels were calculated according to this data. Also, the relationship between the photon energy and the linear attenuation coefficients of water was derived from the National Institute of Standards and Technology (NIST) database[7].

The computation time of convolution algorithms depends on the matrix size of convolution components. However, where the kernels are spatially invariant, the convolution process can be performed in the Fourier space[8]. In the Fourier space, the convolution process becomes a multiplication of the Fourier-transformed TERMA array and the Fourier-transformed kernel array. Therefore, the equation (1) is transformed the following equation in the Fourier space,

$$\mathcal{F}(D) = \mathcal{F}(T) \times \mathcal{F}(K) \quad \cdots \quad (3)$$

where $\mathcal{F}(D)$ is the Fourier-transformed dose, $\mathcal{F}(T)$ is the Fourier-transformed TERMA, and $\mathcal{F}(K)$ is the Fourier-transformed kernel.

We performed the convolution process in the Fourier space. The convolution components were Fourier-transformed by using the fast Fourier transform (FFT) algorithm, and the dose was obtained from the Fourier space by taking the inverse FFT to reduce the computation time. The dose calculation was performed using the MATLAB ver. R2018a (MathWorks, Natick, MA) software. The calculated dose distribution was

compared with the measurement data of clinical machine commissioning. The percentage depth dose (PDD) and the off-center ratio (OCR) in the depth of 1.5 cm, 5.0 cm, 10.0 cm, and 20.0 cm were evaluated.

3. Results

Some of the monoenergetic kernels are shown in the Fig. 2. The point kernels were in form of teardrops, and its shape expanded for the forward direction as the energy increased. The comparison of PDDs between the calculated and measured values is shown in the Fig. 3. Also, the comparison of OCRs on the x-axis is shown in the Fig. 4. In both cases, dose was normalized at the 10 cm depth on the beam axis. The calculated dose distribution was evaluated with reference to the tolerance provided in the International Atomic Energy Agency (IAEA) technical report[9]. The calculated PDD agreed with measured values within 1 mm distance-to-agreement (DTA) value, which is the nearest distance from a point of a reference dose to the point of the same amount of dose, in the build-up region and within 1% error in the depth beyond the build-up region. Also, the calculated OCRs agreed with measured values within 2% errors in the radiation fields, within 1 mm DTA value in the penumbra region, and within 3% errors outside the radiation fields. These calculation errors were within the tolerance.

4. Discussion

The calculated dose distribution was in good agreement with the measurement data and was satisfied with the tolerance provided in the IAEA technical report[9]. Therefore, it was confirmed that the obtained point kernels can be used in the convolution dose calculation. Total computation time of convolution dose calculation was 10 seconds. The convolution process with FFT took less than 1 second, and the other processes (e.g., reading kernels, calculating TERMA, et al.) were run at the rest of time. Just for reference, the convolution process in the real space running on CPU took about 10 minutes and running on GPU took about 2 minutes. CPU was Core i-7 8700K (Intel Corporation, Santa Clara, CA) processer, and GPU was GeForce GTX 1080 (NVIDIA Corporation, Santa Clara, CA) processer. When the dose calculation of the same situation was performed using MC method, it took about 100 hours until the value of variances became sufficiently small. It is widely known that convolution algorithms are not as accurate as MC method in case of complex geometry[1,10,11]. However, taking into account much shorter computation time compared with the MC method, the convolution algorithm is assumed to be a useful dose calculation method in clinical situations.

5. Conclusion

We developed a method of obtaining point kernels by using EGS5. The validity of obtained point kernels was confirmed by using them in convolution dose calculation. It was verified that the calculated dose distribution was in good agreement with the measurement data. The computation time of the convolution algorithm was much shorter than the MC method; therefore, the convolution algorithm is assumed to be a useful dose calculation method in cases where short computation time is required.

References

1) American Association of Physicists in Medicine, "Tissue inhomogeneity corrections for megavoltage photon beams.", AAPM report No. 85 Task Group No. 65 (2004).

2) T. Mackie, A. Bielajew, D. Rogers, et al., "Generation of photon energy deposition kernels using the EGS Monte Carlo code.", Phys. Med. Biol. 33, 1-20 (1988).

3) A. Bielajew, D. Rogers, "Lecture notes: Variance Reduction Techniques.", National Research Council of Canada report PIRS-0396 (1994).

4) International Nuclear Data Committee (INDC), "Phase-Space Database for External Beam Radiotherapy. Summary Report of a Consultants' Meeting.", IAEA Technical Report INDC (NDS) - 0484 (2006).

5) R. Mohan, C. Chui, and L. Lidofsky, "Energy and angular distributions of photons from medical linear accelerators.", Med. Phys. 12, 592-597 (1985).

6) K. Masaki, S. Dobashi, Y. Ishizawa, et al., "Investigating the characteristics of beams from a clinical linear accelerator for clinical dose calculations.", Proceedings of the Twenty-Third EGS Users' Meeting in Japan (2017).

7) NIST, "X-Ray Mass Attenuation Coefficients NIST Standard Reference Database 126." (2004), doi: https://dx.doi.org/10.18434/T4D01F.

8) A. Boyer, E. Mok, "A photon dose distribution model employing convolution calculations.", Med. Phys 12, 169-177 (1985).

9) IAEA, "Commissioning and Quality Assurance of Computerized Planning Systems for Radiation Treatment of Cancer.", Technical Reports Series No. 430 (2004).

10) L. Aarup, A. Nahum, C. Zacharatou, et al., "The effect of different lung densities on the accuracy of various radiotherapy dose calculation methods: Implications for tumour coverage.", Radiother Oncol 91, 405-414 (2009).

11) J. Ojala, M. Kapanen, S. Hyödynmaa, et al., "Performance of dose calculation algorithms from three generations in lung SBRT: comparison with full Monte Carlo-based dose distributions.", J Appl Clin Med Phys 15, 4-18 (2014).
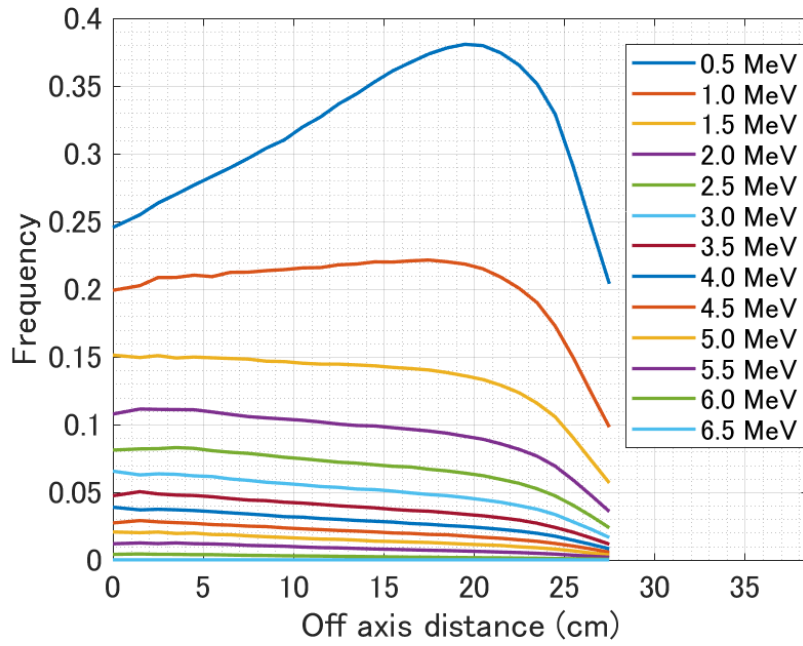
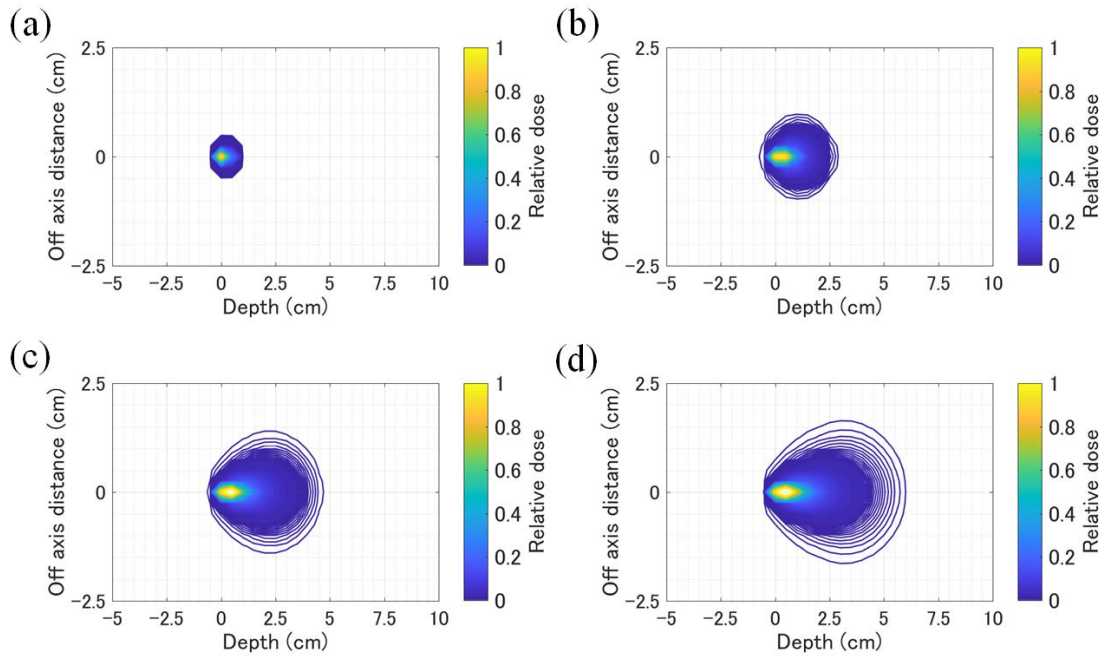Fig. 1. The photon fluence map derived from PSD at the SSD = 100 cm plane.

Fig. 2. 2D-distribution of obtained monoenergetic point kernels.

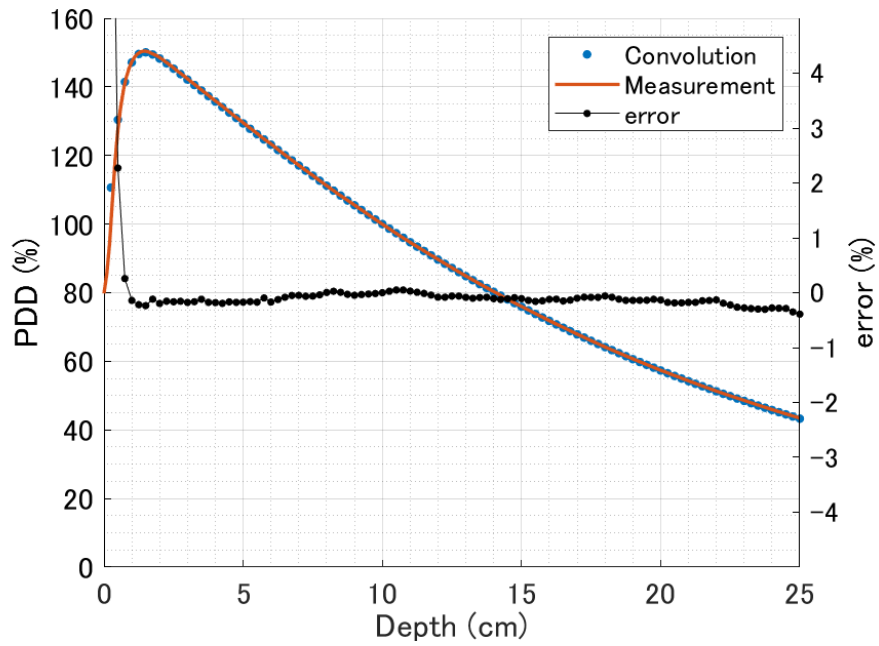(a) 1.0 MeV, (b) 3.0 MeV, (c) 5.0 MeV, (d) 6.5 MeV

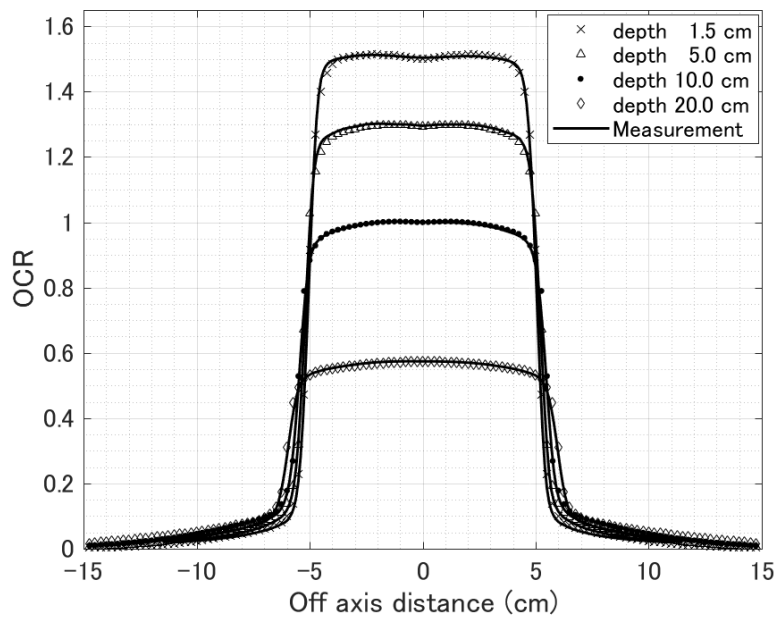Fig. 3. The comparison between calculated and measured PDDs.

Fig. 4. The comparison between calculated and measured OCRs on the x-axis.

PREDICTION OF INTERNAL EXPOSURE BY CS-137 OF RANBURY COLLECTED IN DIFFICULT-TO-RETURN AREA OF FUKUSHIMA DAI-ICHI NUCLEAR POWER PLANT 2
- EGS5 USER CODE AND INTERNAL EXPOSURE CALCULATION FROM 2012 TO 2016-

D Endoh*[1], H Hirayama[2], H Ishiniwa[3] and M Onuma[4]

[1] Department of Radiation Biology, School of Veterinary Medicine, Rakuno Gakuen University, Ebetsu 069-8501, Japan

[2] Radiation Science Center, KEK, High Energy Accelerator Research Organization, Tsukuba 305-0801, Japan

3 Institute of Environmental Radioactivity at Fukushima University, Kanayagawa, Fukushima City, Fukushima Prefecture 960-1296, Japan

4 Ecological Risk Assessment and Control Section, Center for Environmental Biology and Ecosystem, National Institute for Environmental Studies, 16-2 Onogawa, Tsukuba 305–8506, Japan

E-mail of corresponding author: dendoh@rakuno.ac.jp

Abstract

A user code was made to compute beta and gamma-rays of Cs-137, Cs-134 and K-40 based on ICRP Publication 107. Internal radiation doses of the field mice (*Apodemus speciosus*) sampled in difficult-to-return area of Fukushima Daiichi Nuclear Power Plant from 2012 to 2016 was calculated by EGS5. After measurement of Cs-137 specific radioactivity in the mice, internal doses from Cs-137 were calculated to 0.1 - 12 µGy/h. The calculated doses of beta ray was about 10 times larger than those of gamma ray. The internal radiation dose has been reduced to about 1/10 in 2013 compared with 2012. After 2013, those doses were maintained in similar level. The user code which was made in thin study can calculate beta ray and gamma ray internal radiation doses and is expected to be used through combination with script languages such as Python and Ruby.

1. Introduction

As a research project of National Institute for Environmental Studies, Onuma *et al.* have been focusing on the wild field mice, as an indicator for estimating the effects of

diffuse radioactive isotope on wild animals in the difficult-to-return area of Fukushima Daiichi Nuclear Power Plant (NPP) [1]. In 2013, the number of 8OH-dG positive cells was 10 times larger in the testis of Fukushima mice than those in control area (Aomori). Whereas in the following year, the number of 8OH-dG positive cells in Fukushima were almost the same as Aomori. As for the external dose rate in the mice living area, 18.64 µGy/h was recorded in 2012, and there was no major change in the following year. In this study, the internal radiation doses from intra-body Cs-137 beta and gamma ray were recalculated on the samples in 2012 to 2016.

Upon recalculation, we prepared a User Code of EGS 5 that calculates internal doses of ingested Cs-137, Cs-134 and K-40. Since the program can be controlled with arguments, the user code can be adopted for automatic calculation of internal doses on many samples using script languages such as Python and Ruby.

2. Materials and Methods

2.1 Radiation measurements of Mice

Mice were captured by traps in the difficult-to-return area of Fukushima Daiichi NPP, and Cs-137 gamma rays were measured by a germanium measuring instrument for part of the body tissue after euthanasia (Table 1). For Cs-137 beta ray and gamma ray, it was calculated by EGS 5 based on the amount of Cs-137 in body tissue calculated based on the frequency of measured gamma rays. With respect to Cs-134 and K-40, radiation dose was not calculated, since there was no measurement result.

Table 1 The number of the large Japanese field mouse (*Apodemus speciosus*) used to measure Cs-137 concentrations in the present study.

Trapping time		Number		
Year	Month	male	female	Total in year
2012	Aug. - Nov.	15	12	27
2013	Jul. - Oct.	13	6	19
2014	Jun. - Aug.	10	6	16
2015	Aug.	8	3	11
2016	Aug. - Sep.	10	8	18
	Total	**56**	**35**	**91**

Onuma *et al.* [1]

2.2 User Code

As for the geometry for calculation of internal dose, assuming a cylindrical shape as a body of a mice. Radio nucleotides assumed to be dispersed throughout the body, and
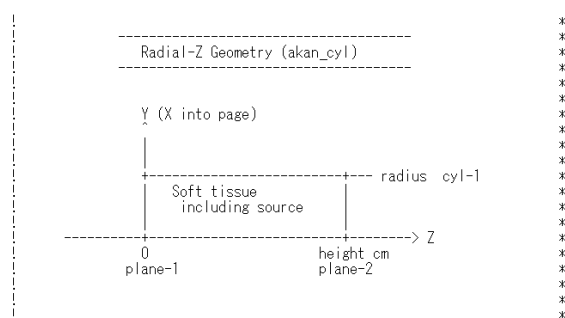
the cylindrical shape correspond to the entire body was set as a measuring device (Fig. 1). Distribution of radioactive isotope for calculation of internal radiation dose, radiation direction of radiation and position where radiation and absorbing material interact is described in User Code. Regarding the energy settings of beta and gamma-rays, the energy distributions for Cs-137, Cs-134 and K-40 were recorded in the "Initialization section" of the User Code (Figures 2 to 4). Variables for beta and gamma-ray-energies were declared as array variables (Fig. 2A). Energies and their distributions were written directly in User Code referring to ICRP Publication 107 (Fig. 2BC).

The elemental composition of the murine body tissue is assumed to be soft tissue and described in a PEGS file (*.inp) file (FIG. 3). The User Code loaded the PEGS file in the "pegs 5 – call" section. The radiation generation position and radiation direction from the nuclide were written in the "Shower-call" section assuming that the radioactive isotope was randomly present in the murine body and released in a random direction from the nucleotide (Fig. 4). For the energy absorbed in the material, interactions of radio-particles were set to occur in the entire cylinder as a measuring instrument in the "Output-of-results" section and the code to sum the absorbed energy were written (Fig. 5). As a result, for beta and gamma-rays, values in Gray per hour (Gy/h) per radioactivity per weight (Bq/g) were calculated.

In executing the User Code, after setting the subject radiation (beta-ray or gamma-ray) and nuclide (Cs-137, Cs-134 or K-40) on a command line, user code is executed and the calculated value was output as a result file.

Fig. 1


```
                ------------------------------------            *
                  Radial-Z Geometry (akan_cyl)                  *
                ------------------------------------            *
                                                                *
                                                                *
                  Y (X into page)                               *
                                                                *
                  |                                             *
                  |                                             *
      +-----------------------------+--- radius  cyl-1          *
                  | Soft tissue                                 *
                  |   including source                          *
                  |                                             *
      -----------+-----------------------+--------> Z           *
                  0                  height_cm                   *
              plane-1                plane-2                     *
```

Geometry-comment which defined in the user code.

Fig. 2

A

```
   real*8↓
 * ebeta(74),pbeta(74),cbeta(74),egamma(16),pgamma(16),cgamma(16),↓
 * ebcs137(60),pbcs137(60),egcs137(3),pgcs137(3),ebcs134(74),↓
 * pbcs134(74),egcs134(9),pgcs134(9),ebk40(67),pbk40(67),↓
 * egk40(16),pgk40(16) ↓
```

B

```
  ICRP 107 Cs-137 beta data↓
 data ebcs137/0.00000,0.02000,0.04000,0.06000,0.08000,0.10000,↓
 *    0.12000,0.14000,0.16000,0.18000,0.20000,0.22000,0.24000,↓
 *    0.26000,0.28000,0.30000,0.32000,0.34000,0.36000,0.38000,↓
 *    0.40000,0.42000,0.44000,0.46000,0.48000,0.50000,0.52000,↓
 *    0.54000,0.56000,0.58000,0.60000,0.62000,0.64000,0.66000,↓
 *    0.68000,0.70000,0.72000,0.74000,0.76000,0.78000,0.80000,↓
 *    0.82000,0.84000,0.86000,0.88000,0.90000,0.92000,0.94000,↓
 *    0.96000,0.98000,1.00000,1.02000,1.04000,1.06000,1.08000,↓
 *    1.10000,1.12000,1.14000,1.16000,1.17563/↓

 data pbcs137/0.0,0.67579E-01,0.64985E-01,0.63006E-01,0.61128E-01,↓
 *    0.59349E-01,0.57611E-01,0.55874E-01,0.54112E-01,0.52270E-01,↓
 *    0.50313E-01,0.48206E-01,0.45909E-01,0.43403E-01,0.40647E-01,↓
 *    0.37622E-01,0.34327E-01,0.30723E-01,0.26949E-01,0.23041E-01,↓
 *    0.18999E-01,0.15080E-01,0.11283E-01,0.76665E-02,0.51303E-02,↓
 *    0.27741E-02,0.15107E-02,0.13401E-02,0.11878E-02,0.11448E-02,↓
 *    0.11199E-02,0.10934E-02,0.10651E-02,0.10363E-02,0.10046E-02,↓
 *    0.97242E-03,0.93808E-03,0.90157E-03,0.86451E-03,0.82409E-03,↓
 *    0.78312E-03,0.74002E-03,0.69481E-03,0.64918E-03,0.60104E-03,↓
 *    0.55247E-03,0.50299E-03,0.45258E-03,0.40218E-03,0.35178E-03,↓
 *    0.30138E-03,0.25514E-03,0.21306E-03,0.17098E-03,0.12890E-03,↓
 *    0.86818E-04,0.57081E-04,0.39686E-04,0.22292E-04,0.53119E-05/↓
```

C.

```
  ICRP 107 Cs-134 gamma data↓
 data egcs137/0.0332,0.0364,0.662/↓
 data pgcs137/0.056,0.013,0.851/↓

  ICRP 107 Cs-134 gamma data↓
 data egcs134/4.754E-01,5.632E-01,5.693E-01,6.047E-01,↓
 *   7.959E-01,8.020E-01,1.039E+00,1.168E+00,1.365E+00/↓
 data pgcs134/1.49E-02,8.35E-02,1.54E-01,9.76E-01,↓
 *   8.55E-01,8.69E-02,9.88E-03,1.79E-02,3.01E-02/↓

  ICRP 107 K-40 gamma data↓
 data egk40/2.928E-03,2.931E-03,3.163E-03,3.163E-03,6.400E-05,↓
 *    6.634E-05,2.988E-04,2.990E-04,2.340E-06,2.205E-04,↓
 *    2.350E-04,2.182E-04,2.325E-04,2.327E-04,5.110E-01,↓
 *    1.461E+00/↓
 data pgk40/2.96E-03,5.85E-03,2.42E-04,4.79E-04,5.02E-06,1.35E-05,↓
 *    3.92E-06,6.19E-06,9.61E-14,2.97E-05,1.28E-08,6.10E-05,↓
 *    1.30E-08,1.28E-08,1.80E-03,1.07E-01/↓
```

Settings in User Code. A: Declarations of Array-variables for each nucleotides. B: Data for distribution and energies for beta-ray of Cs-134 written directly in User Code. C: Data for gamma-rays from Cs-137, Cs-134 or K-40 directly written in User Code.

Fig. 3

```
MIXT↓
 &INP NE=11, RHO=1.0, RHOZ=10.44,23.22,2.49,63.02,0.11,0.01,0.13,0.20,0.13,0.20,
0.02 /END↓
SOFT TISSUE                ↓
H  C  N  O  NA MG P  S  CL K  CA  ↓
ENER↓
 &INP AE=0.521,AP=0.010,UE=2.511,UP=2.0 /END↓
PWLF↓
 &INP  /END↓
DECK↓
 &INP  /END↓
```

Settings for elements including in mice body in a PEGS file.

Fig. 4

```
!       ---------------------------↓
!       Determine source   direction ↓
!       ---------------------------↓
1300    call randomset(rnnow)↓
        zi0=0.2*rnnow-0.1↓
        call randomset(rnnow)↓
        xi0=0.2*rnnow-0.1↓
        call randomset(rnnow)↓
        yi0=0.2*rnnow-0.1↓
        rr0=dsqrt(xi0*xi0+yi0*yi0+zi0*zi0)↓
        if(rr0.gt.1.0) go to 1300↓
        win = zi0/rr0↓
        uin = xi0/rr0↓
        vin = yi0/rr0↓
↓
!       ----------------------------------------↓
!       Determine source position inside cylinder↓
!       r=radius and z=heght↓
!       ----------------------------------------↓
!       modified by HH 2018/3/6↓
1000    call randomset(rnnow)↓
        xin=rnnow*2.0-1.0↓
        call randomset(rnnow)↓
        yin=rnnow*2.0-1.0↓
        call randomset(rnnow)↓
        rr0=sqrt(xin*xin+yin*yin)↓
        if (rr0.gt.1.0) go to 1000↓
!       Scale with radius↓
        xin=xin*radius↓
        yin=yin*radius↓
        zin=rnnow*height↓
```

Settings for radioisotope distribution and direction of release in User Code.

Fig. 5

```
if (totke .le. 0.D0) then↓
   write(6,'(/A,G15.5/A,G15.5,A,I10)') ↓
*      ' Stopped in MAIN with TotKE=',totke,↓
*      ' AvailKE=',availke,' Ncount=',ncount↓
   stop↓
end if↓

if(istype.eq.0) then↓
   if(nuclide.eq.0) then↓
     write(6,'(/A)') ' Results for Cs-137 beta-ray'↓
   elseif(nuclide.eq.1) then↓
     write(6,'(/A)') ' Results for Cs-134 beta-ray'↓
   else↓
     write(6,'(/A)') ' Results for K-40 beta-ray'↓
   end if↓
else↓
   if(nuclide.eq.0) then↓
     write(6,'(/A)') ' Results for Cs-137 gamma-ray'↓
   elseif(nuclide.eq.1) then↓
     write(6,'(/A)') ' Results for Cs-134 gamma-ray'↓
   else↓
     write(6,'(/A)') ' Results for K-40 gamma-ray'↓
   end if↓
end if↓

write(6,'(/A,F10.3,A,F10.3,A)') ' Organ radius =',↓
*    radius,' cm radius and ',height,' cm height cylinder'↓
```

Settings for calculation of energy absorption of beta or gamma-rays from Cs-137, Cs-134 or K-40, directly written in User Code.

3.   Results

Regarding the specific radioactivity in the mice, it was about 100,000 Bq/kg for female and 50,000 Bq/kg for male in 2012, but it decreased drastically in 2013 and maintained

to similar values in 2014 to 2016 (Fig. 6). Regarding the dose rate calculated by EGS5, the beta-ray was 5-12 µGy/h in 2012, whereas it remained constant at around 1-5 µGy/h after 2013 (Fig. 7A). Internal exposure dose of gamma-rays was about 10 times lower than that of beta-ray, which is 0.5-1 µGy/h in 2012. The dose rate from gamma-rays also showed relatively lower value similar to those for beta-rays (0.1-0.25 µGy/h after 2013, Fig. 7B). For both periods, dose rate from internal gamma-ray was about 1 / 10th of the dose rates of internal beta-ray. The total dose of beta- and gamma-rays was close to the dose of beta-ray (Fig. 7C).
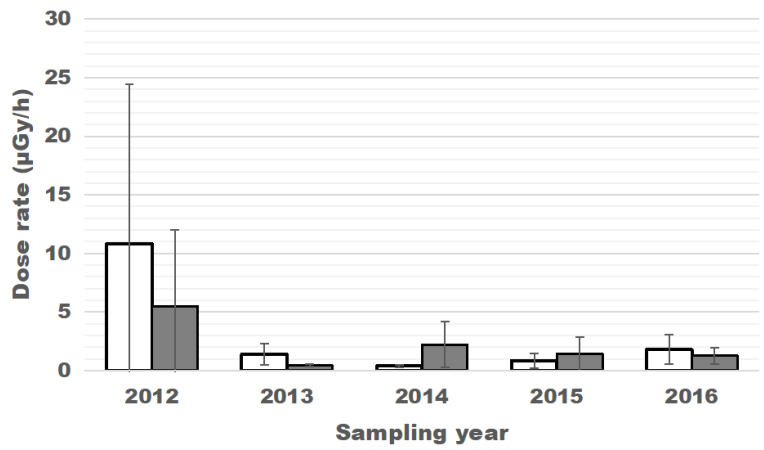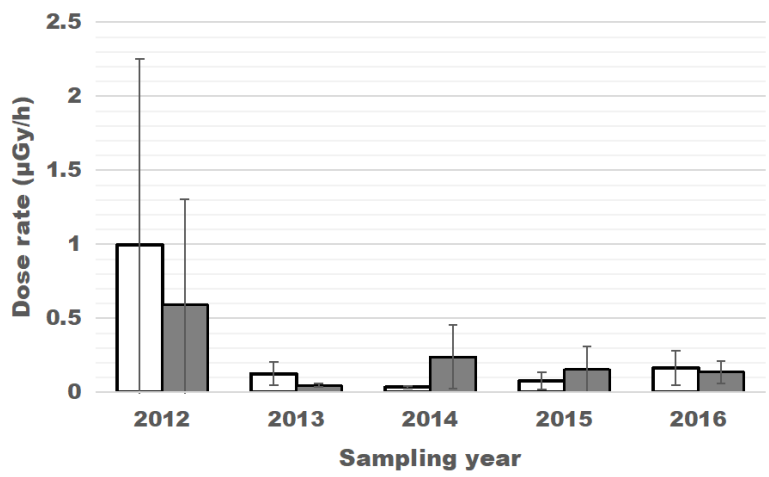
Fig. 1
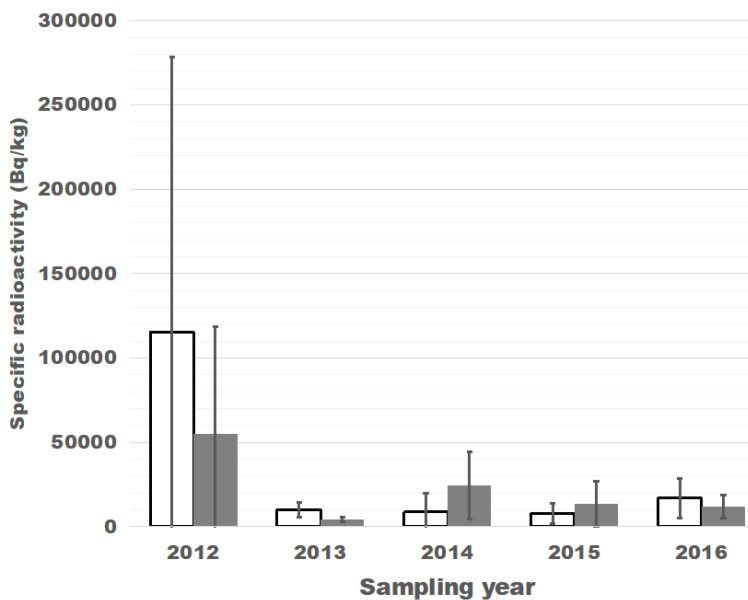
Specific radio activities in field mice which was measured with Ge-radiation detector. White bar indicate Female mice and Gray bar indicate Male mice. Error bars indicate standard deviations. (Reconstructed from Reconstructed from Onuma *et al.* [1]).
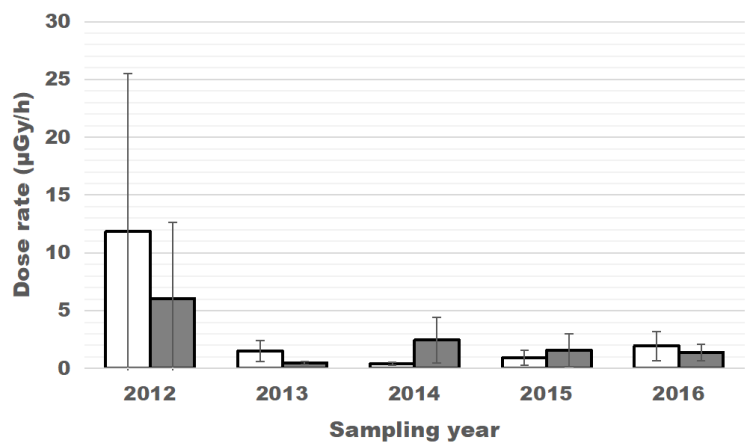
Fig. 2

A

Calculated internal radiation doses from beta-ray (A), gamma-ray (B) or beta- and gamma-rays (C) in field mice. White bar indicate Female mice and Gray bar indicate Male mice. Error bars indicate standard deviations. (Reconstructed from Onuma *et al.* [1]).

4. Discussions

With the radioactivity predicted from the gamma-ray counts of Cs-137 with Ge-radiation detector, there is a sharp decline of specific radio activities from 2012 to 2013 (Fig. 6). On the other hand, such a rapid decline is not shown in the external exposure in the mice-sampling region (Onuma *et al.* 2018). For Cs-137, since strong or irreversible adsorption to the soil is reported [2], during this period a decrease of radioactive elements may decrease in the feeding materials. In the future, when the difference between the external exposure dose and the internal radiation dose increases, it is desirable to analyze the localization of Cs-137 in feeding materials or soil in the sampling field. From the data on Cs-137- influence for a long time suggested to be obtained.

On the use of EGS5, we modified User Code to calculate not only Cs-137 but also the internal exposure dose of Cs-134 and K-40. Calculation of internal radiation doses is required for various subjects to estimate influences of radio nucleotides. Since the contribution of Cs-137, Cs-134 and K-40 is practically large in Japan, computer programs simulating internal radiation dose from Cs-137, Cs-134 and K-40 suggested to be required. Since these calculations are originally carried out on a large amount of samples, it is desirable that a large number of data can be automatically executed. Because EGS5 is described in FORTRAN and the calculation result is created in a file, calculation automation has been done in FORTRAN or Shell Script [3]. However, in order to calculate numerous data by taking in data such as tables of spreadsheet software such as Microsoft Excel, the computer program was expected to use spreadsheet automatically. Scripting computer languages such as Python or Ruby can use spreadsheet file with many additive analysis options. By setting energy, emitting position and directions, one can use the User Code with some command-line arguments without any modification argument of the script for the user code created in this study. Scripting program languages also treat the result file generated by EGS5. It is expected to be used for automatic calculation of internal radiation doses in various species of wild animals and ecosystems in the future.

6. References

[1] Onuma, M , Endoh, D., Ishiniwa H., Tamaoki M., (2018) "Estimation of Dose rate for the large Japanese Field Mice (Apodemus speciosus) Distributed in the "Difficult-to-Return Zones" in Fukushima Prefecture" in Manabu Fukumoto (Ed), *Impacts on the ecosystem affected by the Fukushima Daiichi Nuclear Power Plant (FNPP) accident: What we are learning from animals and plants,* Springer (in preparation).

[2] Niimura, N., Kikuchi, K., Tuyen, N. D., Komatsuzaki, M., & Motohashi, Y. Physical properties, structure, and shape of radioactive Cs from the Fukushima Daiichi Nuclear Power Plant accident derived from soil, bamboo and shiitake mushroom measurements. *Journal of environmental radioactivity*, 139, 234-239, 2015.

[3] Wilderman, S. J. "Automated electron step size optimization in EGS5." Proc. 13th EGS Users' Meeting in Japan (KEK, Tsukuba, Japan). 2006.