# VARSKIN+ v1.1
# Software Release Note

February 2022

## 1.0 Introduction

This document outlines in detail the modifications, additions, and/or removal of features of VARSKIN+ v1.0 to VARSKIN+ v1.1.

All modifications are based on items entered into RCD software issue tracker as a source for the basis of a change.

## Table of Contents

## 2.0 Abbreviations and Definitions

Table 2-1. Abbreviations and definitions.

| Term | Definition |
|---|---|
| DLL | dynamic-link library |
| GUI | graphical user interface |

## 3.0 Base Line Code

The changes identified here have been incorporated into the baseline of VARSKIN+ v1.1.

Table 3-1. GIT attributes.

| Git Location | Revision ID (Commit SHA) |
|---|---|
| https://gitlab.com/RCD-1/varskin/-/tags/varskin+v1.1 | d2f7c94e0b144f9830f151307078c27f68d63a6e |

Table 3-2. Executable attributes.

| | |
|---|---|
| **Executable Name** | VARSKIN+1.1.exe |
| **MD5sum** | 3b281c55526d2d364a36e099bd7df5dc |
| **SHA256** | 66cd7a712eb7ed41a9fd932941b238479e3173674a40aed5fd27c3aadfd7f7d4 |
| **SHA512** | 79771ba14c6c18a2791135875bfef31449597a4cb39421d620102986a0c988d9f7d14702a2bb40a9cab72decdc56a30b51742dde59c91d8c4251d74fdcde9ea2 |

VARSKIN+ has the following project dependency as noted in Table 3-3

Table 3-3. Dependency attributes.

| Software | Version | Git Location | Revision ID (Commit SHA) |
|---|---|---|---|
| Graphing | 0.0 | https://gitlab.com/RCD-1/graphing/-/tags/Graphingv0.0 | b1c564dcaad1e5bcb9ee78f38f39e2109d824535 |
| Intel Fortran | 19.1.3.311 | n/a | n/a |
| JDK | 16.0.2 | n/a | n/a |

## 4.0 Changes

The following sub-sections outline the changes made from VARSKIN+ version 1.0 to VARSKIN+ version 1.1. A summary of changes is provided below:

- Correction made to wound dose line source dose calculation (Issue 1).
- Correction and removal of incorrect yield check threshold for electrons (Issue 2).
- Correction made to regional settings impacting decimal separator (Issue 3 and 15).
- Correction made to EyeDose shielded dose when photon dose is zero (Issue 8).
- Correction to WoundDose backscatter coefficient with zero injury depth (Issue 10).
- Correction to integrated dose for large exposure times (Issue 11).
- Disabled off-axis scatter correction in photon dose calculation (Issue 12).
- Correction to incorrect parsing of minute half-life units (Issue 13).
- Correction to feet unit conversion and NeutronDose initialization (Issue 14).

### 4.1. Wound Dose Line Source Dose (Issue 1)

A correction was made to wound dose line source dose calculation. The intent of the line source was to calculate the dose from a series of point sources along a puncture depth (line). Points are at fixed increments of 10 microns. The user enters an absolute strength that is internally evenly distributed along the points that make up the puncture/line.

The previous version of the code (v1.0) calculates the dose at each of the points that make up the line. What is incorrect is how the dose from each point is added together. The doses should be directly added since the points are uniformly spaced and representing equal interval sizes. However, the dose from each point is weighted where the weighting factor is (1/#points). This is effectively an average dose from all the points. Thus, the answer obtained by v1.0 will be smaller than reality as it is reporting the average dose from the points not the total dose of all points.

The code was corrected, and a few tests were performed comparing the v1.0 with the error and a corrected beta version.

The first test consists of determining the electron dose at 70 microns of a 1 cm$^2$ area from a puncture of 1 MBq of Tc-99m. Numerous cases are performed at varying puncture depths to observe the response. The results are presented in Figure 4-1. As can be seen the response v1.0 shows a dip in the total dose starting from the skin surface (zero puncture depth) and approaching the dose depth plane at 0.07 mm. However, as the puncture depth approaches this plane the dose should increase. This correct response is captured by the corrected version of the code. As the puncture depth moves beyond the dose the plane the dose will begin to decrease as the source strength per unit length decreases with a fixed source strength.

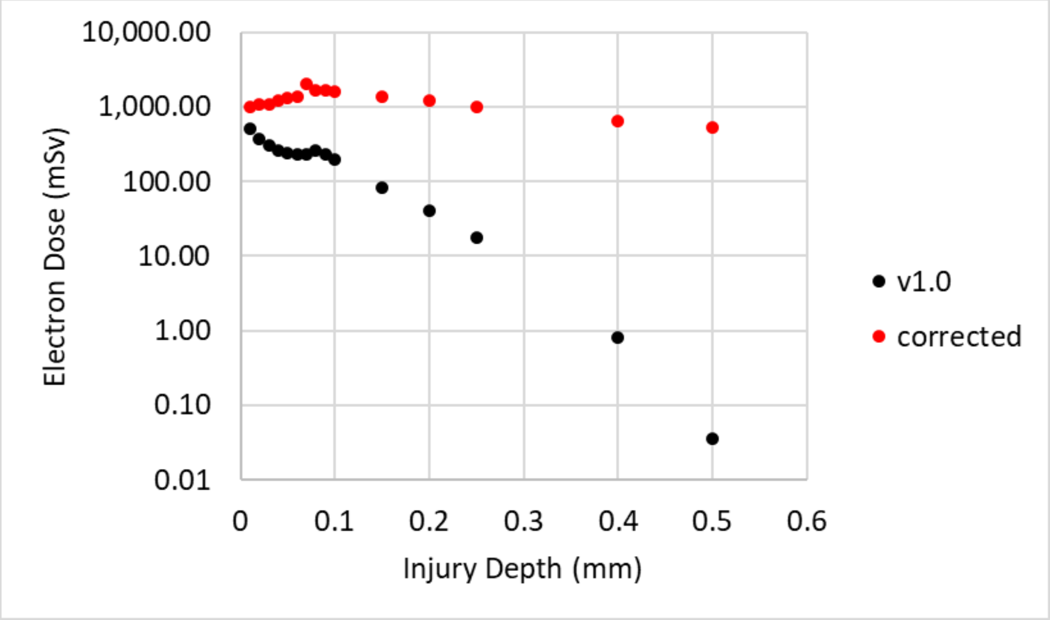


Figure 4-1. Electron dose as a function of depth.

A second test was performed that repeated the first test except at the various puncture depths the source strength was modified to maintain constant linear source strength. What is expected is that the total dose should increase as the puncture depth increases to some point where the puncture depth is too deep such that the electrons are attenuated before making it to the dose depth plane. The X90 distance for Tc-99m is 0.16 mm thus this distance should be at about 0.16mm + 0.07mm= 0.23 mm. Figure 4-2 shows that v1.0 does not show these characteristics while the corrected beta version in Figure 4-3 captures the expected behavior exactly.

This issue is resolved.

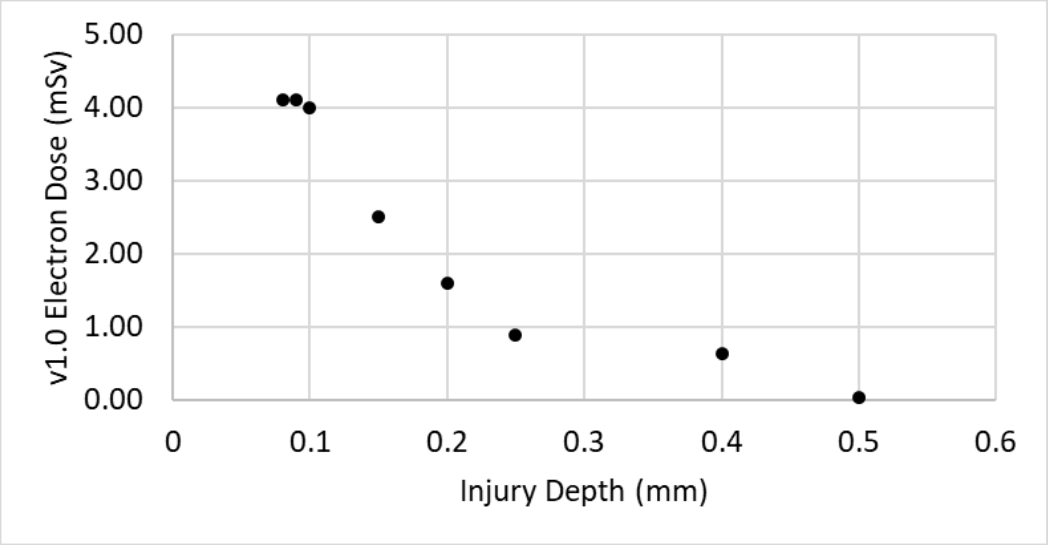


Figure 4-2. Electron dose as a function of depth for constant linear source strength with code in error.

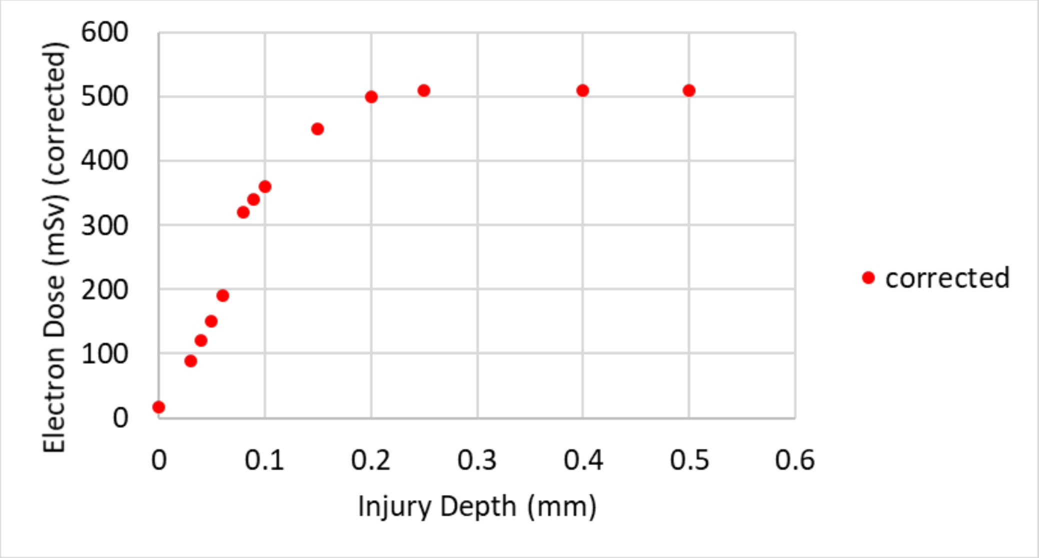


Figure 4-3. Electron dose as a function of depth for constant linear source strength with corrected code.

## 4.2. Correction to Electron Yield Threshold (Issue 2)

It was reported that for Tc-99m using the ICRP-38 database that there was no resulting electron dose being calculated. For version 1.0 the yield and energy threshold checks were consolidated into a single database read routine. However, one of the checks within sadcalc was overlooked and performing another erroneous check on the electrons yields. This yield check was in error and was resulting in sadcalc logic resulting in no valid electrons. Removing this erroneous check resolves this issue.

This issue is resolved.

### 4.3. Correction to Regional Settings (treatment of decimal delimitator) (Issue 3 and 15)

When launching the executable file by double-clicking the VARSKIN+ icon the VARSKIN+ main screen appears, but when clicking on one of the modules (SkinDose/WoundDose//NeutronDose/EyeDose) nothing occurs, and the radioactive sign continues to rotate. This was an issue reported by a user in Italy.

In regions where "," is used a decimal separator versus "." VARSKIN +v1.0 will not be able to open a module. Methods in Java (e.g., Double.parseDouble()) are expecting the use of a point as the decimal separator. Logic was added to handle these cases prior to parsing inputs.

This issue is resolved.

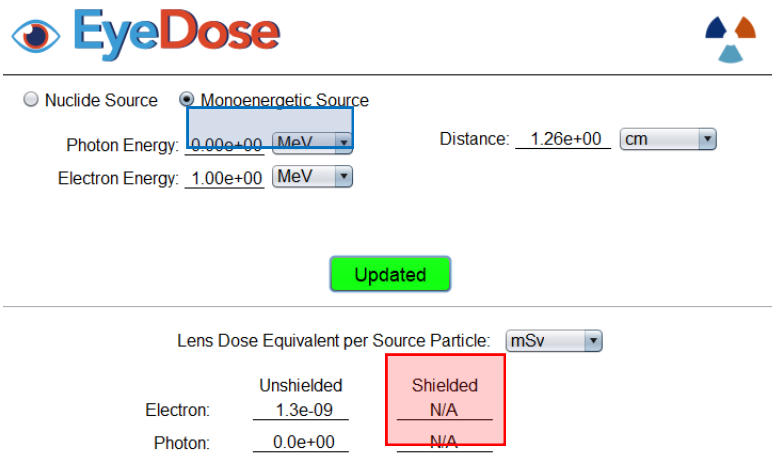
### 4.4. Eye Dose "N/A" Shielded Dose with Zero Photon Dose (Issue 8)

The shielded eye dose model is valid only for source distances of 1.25 centimeters or less. The Fortran module returns -0.0 for shielded dose if the provided distance is less than the required value. Java checks for the presence of -0.0 in the shielded photon dose; if present, "N/A" is displayed for all shielded doses to inform the user that the provided inputs are invalid.

The issue arises because Java does not distinguish between -0.0 and 0.0 in comparisons, so an actual shielded photon dose of zero (possible with e.g., monoenergetic sources that emit no photons) would be construed as an invalid response, setting all shielded photon dose displays to "N/A" even though there may be a valid shielded electron dose.

This issue has been rectified by preceding Java's simple "equals" comparison check with a "copy sign" operation, to extract the possible negative sign from the returned shielded dose. If a negative sign is present, assumed to be the case only when -0.0 has been returned (no negative doses possible), shielded dose is displayed as "N/A."

This issue is resolved.



○ Nuclide Source ● Monoenergetic Source

Photon Energy: 0.00e+00 MeV

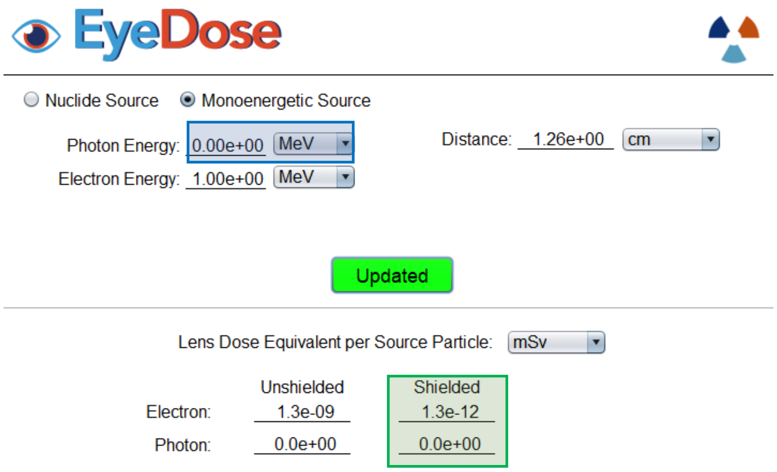
Electron Energy: 1.00e+00 MeV

Distance: 1.26e+00 cm

Lens Dose Equivalent per Source Particle: mSv

| | Unshielded | Shielded |
|---|---|---|
| Electron: | 1.3e-09 | N/A |
| Photon: | 0.0e+00 | N/A |

Figure 4-4. Eye dose zero dose results (code in error)



○ Nuclide Source ● Monoenergetic Source

Photon Energy: 0.00e+00 MeV

Electron Energy: 1.00e+00 MeV

Distance: 1.26e+00 cm

Lens Dose Equivalent per Source Particle: mSv

| | Unshielded | Shielded |
|---|---|---|
| Electron: | 1.3e-09 | 1.3e-12 |
| Photon: | 0.0e+00 | 0.0e+00 |

Figure 4-5. Eye dose zero dose results (corrected code)

## 4.5. Wound Dose BSCF Disabled with Zero Injury Depth (Issue 10)

Air backscatter correction factors are disabled by the wound model when injury depth equals zero. This is accomplished by saving the actual BSCFs (determined by sadcalc) to a temporary variable, resetting the BSCF array in the appropriate nuclide instance to all ones, and proceeding with the calculation. When the calculation is complete, the nuclide's BSCF array is reset to the values in the temporary variable.

The issue arises because of an iterative loop required for line source geometries, which involves summing the dose from a number of point sources. Since the temporary variable was set during this loop, and thus it was possible (if the previous iteration had used an injury depth of zero) to reset the temporary variable to contain all ones, overwriting the actual BSCFs. This was guaranteed to occur with line source geometries, which involve integration from zero depth (skin surface) to the injury depth. Note that this error can only occur during calculation of a line source geometry, as the point source geometry does not use the referenced iterative loop.

This issue has been rectified by setting the temporary variable once, before the iterative loop is entered, to ensure that it cannot be set after the nuclide's BSCF array has been set to 1.0. The "if" statement within the loop then selects the proper set of values to use for the current calculation, either resetting the nuclide's array to the temporary (actual) values or setting the nuclide's array to 1.0 to disable backscatter correction. The nuclide's BSCF array is then reset to the temporary values after the loop exits.

This effect can easily be seen by comparing the dose for a point source geometry calculation before and after calculating the dose for a line source geometry, with an injury depth of zero. Table 4-1 shows this phenomenon occurring in v1.0 and corrected in v1.1; doses are calculated using $^{16}$N from the ICRP107 database, and all other parameters set to defaults.

Table 4-1: Disabled BSCF effect on dose

| V+ Version | Dose<br>(Pt. Src., before Line Src.) | Dose<br>(Pt. Src., after Line Src) |
|---|---|---|
| v1.0 | 3.2E-07 mSv | 2.1E-09 mSv |
| v1.1 | 3.2E-07 mSv | 3.2E-07 mSv |

### 4.6. Integrated Dose Correction for Large Exposure Times (Issue 11)

Dose decay correction relies on a computed "decay factor;" the routine to calculate same performs a numeric integration from zero to the specified exposure time. The intended result is a value between zero and one, which defines the fraction of the specified nuclide that has decayed in the given time, and thus contributed to dose.

The numeric integration is performed using the trapezoidal method, and so relies on subdividing the region of integration into a fixed number of sections. However, the calculation was inadvertently set up such that if the quotient of the provided exposure time and the nuclide's half-life exceeded the number of subdivisions, the result of integration could exceed one.

This results in the evaluated dose leveling off after several tens of half-lives, as expected, and then starting to increase again after the integration point (previously set at 1,000) had been exceeded. This is clearly seen in dose response from v1.0 in Figure 4-6.

This issue has been rectified by first checking to ensure the provided exposure time, divided by the nuclide's half-life, does not exceed the integration point. If it does, the integration point is adjusted up appropriately.

This issue is resolved.

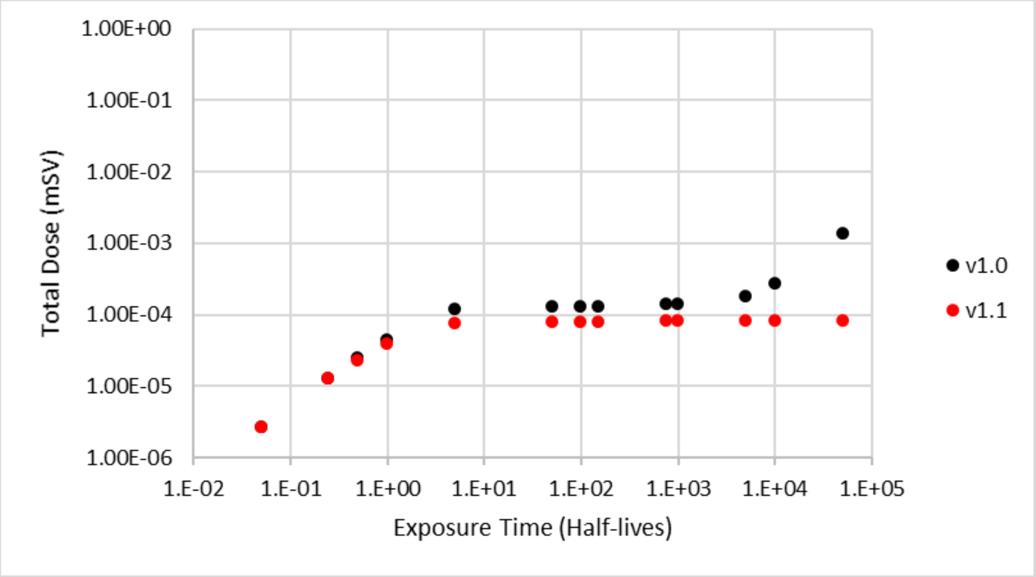


Figure 4-6. Integrated dose behavior.

## 4.7. Off-Axis Correction Factor Step Function (Issue 12)

An error was located in the determination of off-axis scatter correction factor in the gamcalc Fortran module. This factor was calculated based on the total depth (source, cover, air gap, and tissue) with air gaps less than or equal to 0.01 mm, and based on only tissue depth for air gaps greater than 0.01 mm. There was found to be a large discrepancy between these two correction factor values, especially for thick covers. This caused a notable difference in photon dose when the module switched from one calculation method to the other.

Photon dose results from v1.0 were compared to an interim version with off-axis scatter correction disabled. MCNP runs corresponding to the same set of parameters were also completed. The results of these runs, for increasing air gaps with Ba-137m, are summarized in Table 4-2 and Figure 4-7.

Table 4-2. Off-axis scatter dose comparison

| Air Gap (mm) | 7 mg/cm$^2$ Dose, mSv | | |
|---|---|---|---|
| | MCNP | V+ (w/ OA) | V+ (w/o OA) |
| 0 | 4.42E-13 | 4.09E-10 | 4.32E-10 |
| 0.0001 | 4.42E-13 | 4.09E-10 | 4.32E-10 |
| 0.001 | 4.42E-13 | 4.09E-10 | 4.32E-10 |
| 0.01 | 4.41E-13 | 4.08E-10 | 4.31E-10 |
| 0.011 | 4.41E-13 | 2.34E-10 | 4.30E-10 |
| 0.015 | 4.40E-13 | 2.34E-10 | 4.30E-10 |
| 0.1 | 4.29E-13 | 2.27E-10 | 4.21E-10 |

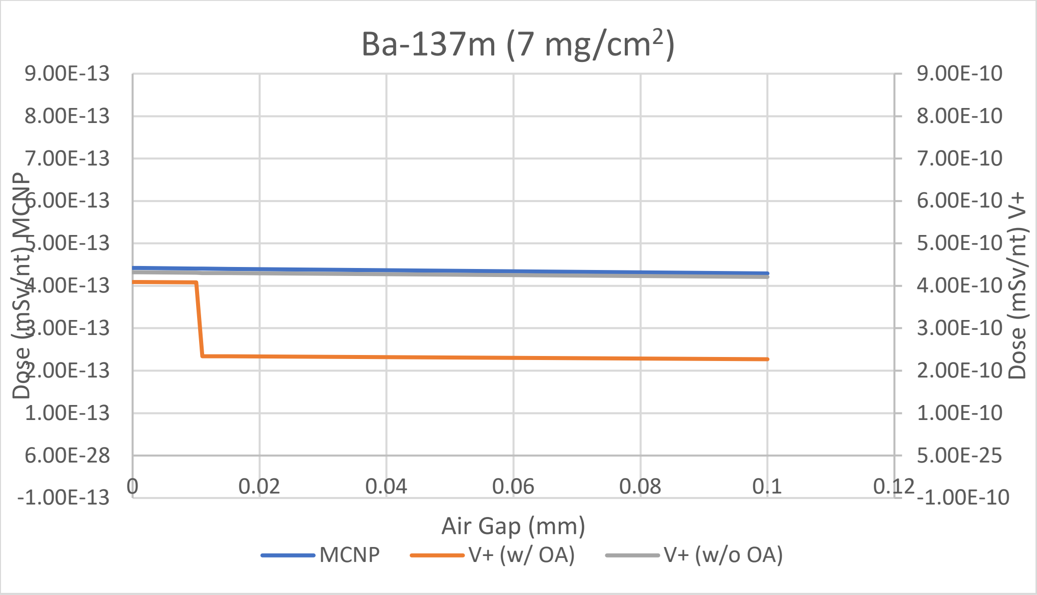


Figure 4-7. Off-axis scatter dose comparison

Because the basis for switching between two different depths in calculating the off-axis scatter correction factor is not known, this calculation has been temporarily disabled pending further research. v1.1 dose response follows the “V+ (w/o OA)” column in Table 4-2.

This issue is resolved.

### 4.8. Incorrect Half-Life Units (Issue 13)

Half-life, like other variables, is stored as a number and a string where the string holds the units of the associated value. Because any variable may have any type of unit, all the possible unit strings must be unique. Thus, “m” is used to denote meters, and “min” is used to denote minutes.

When nuclide information is first read from the ICRP38/107 database in the “Add Nuclides” screen, the units of the half-life variable are set directly from the string read from the ICRP data file. Because the ICRP databases use “m” to denote minutes, half-lives with units of minutes were erroneously imported as having units of meters. This resulted in an incorrect unit conversion.

This issue has been rectified by specifically replacing half-life units of “m” with “min” while reading the ICRP databases.

This issue is resolved.

### 4.9. Improper “Feet” Unit Conversion (Issue 14)

Unit conversions are handled internally by first converting the given variable from the unit it’s already in to a “standard” unit, then from the “standard” unit to the desired output unit. For length variables, the “standard” unit is centimeters; however, input or output units of feet used the inch-to-centimeter conversion factor.

This was only apparent in the neutron or eye dose modules; since those modules’ Source Distance variables are the only ones that can use units of feet. This can be seen using default values in the eye dose module (1 Bq of Ac-223 for 60 seconds) and comparing equivalent source distances. Only the distance with units of feet has a clearly incorrect dose, which is rectified in the new version.

This issue is resolved.

Table 4-3. EyeDose incorrect units

| Source Distance | Total Unshielded Dose mSv, v1.0 | Total Unshielded Dose mSv, v1.1 |
|---|---|---|
| 1 m | 7.00E-14 | 7.00E-14 |

| Source Distance | Total Unshielded Dose mSv, v1.0 | Total Unshielded Dose mSv, v1.1 |
|---|---|---|
| 100 cm | 7.00E-14 | 7.00E-14 |
| 1000 mm | 7.00E-14 | 7.00E-14 |
| 39.37 in | 7.00E-14 | 7.00E-14 |
| 3.28 ft | 1.00E-19 | 7.00E-14 |

### 4.10. Uninitialized Neutron Dose Time-Distance-Activity Variables (Issue 14)

The neutron dose module was observed to output zero dose for any combination of input variables, from the second time the "Calculate" button was clicked after opening the module, with a source term of spontaneous fission. Furthermore, selecting a source term other than spontaneous fission, then returning to spontaneous fission, resulted in a calculation being performed with an incorrect source distance. This was evidenced by the trend of dose to not decrease in accordance with the inverse square law, in this case.

Both manifestations of this error were determined to be caused by uninitialized time-distance-activity variables. Because the "set" routines had not been run on these variables when the module window opened, they retained their default values; more importantly, the fluence variable (not visible with a source term of spontaneous fission) retained its default value.

Clicking the "Calculate" button caused these variables to be set, and fluence recalculated; since the default value for distance is zero, this resulted in a zero fluence after the first calculation, unless the distance variable had been force-set by the user before clicking "Calculate." This occurred if the source term was changed, since the variables are updated whenever the source term (and therefore user input boxes) changes.

Both parts of this error have been rectified by running the "set" routines on each of the time-distance-activity variables, ensuring that the displayed value matches the actual value of the variable, and that the fluence variable matches the displayed time-distance-activity values.

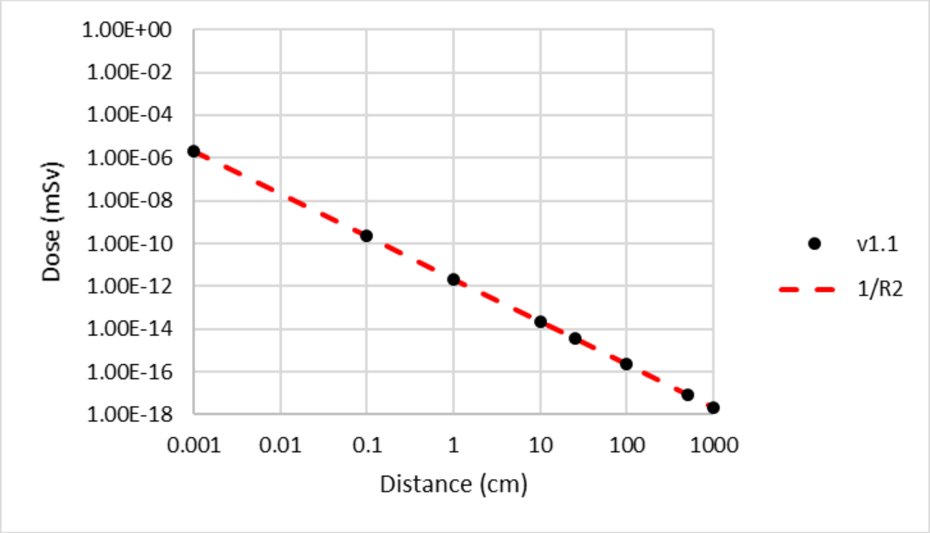


Figure 4-8. NeutronDose dose results as a function of distance.