Virtual Dosimeter for Interventional Fluoroscopy

Hawon Lee[1,2] and Andreu Badal[2]

[1] Montgomery Blair High School, Silver Spring, MD
[2] Division of Imaging Diagnostics and Software Reliability / OSEL / CDRH / FDA, Silver Spring, MD

Introduction

- Interventional fluoroscopy is an imaging modality that provides a real-time x-ray image of a patient's anatomy.
- The radiation dose delivered to patients and operators during interventional fluoroscopy procedures can be substantial, and cause radiation injuries [1].
- Dosimeter badges, the standard method used for dose estimation, measure radiation at discrete points, and the dose is only known after the procedure.
- With increasing use of fluoroscopy, there is a need for new tools that can reduce the dose and minimize side effects.
- At CDRH/OSEL/DIDSR we are developing *a Virtual Dosimeter* system [2] to estimate, computationally, the radiation delivered to operators and patients in real-time.

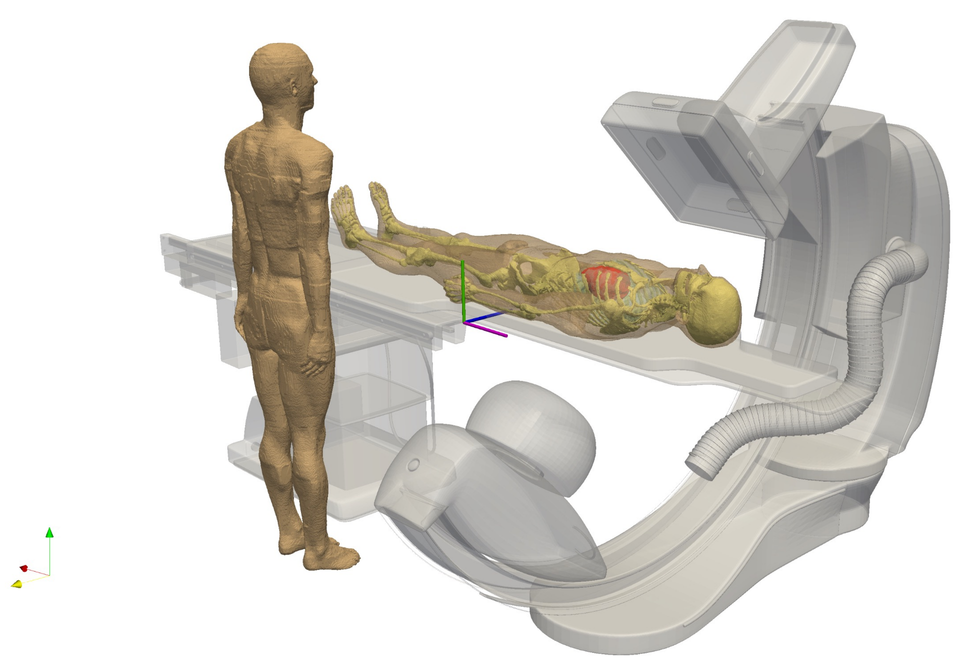

Figure 1. Computational model of an operator standing next to a patient during an interventional fluoroscopy procedure. [1]

Aim

- To develop a new *Conductor* software tool that coordinates the execution of the 3 existing modules of the *Virtual Dosimeter* and create a new *Display module*.

New Software Tool Requirements

- Written in Python for the Linux operating system.
- Runs in a NVIDIA Jetson TX2 embedded computer.
- Sends and receives messages between the independent processes using the Robot Operating System (ROS), a popular, open-source library used for robot applications.

New Software Tool Operation

- Start the independent programs (modules) in separate shells.
- Exchange information between the modules in 4 steps:
  - A. Receive x-ray source data from *Source module*.
  - B. Communicate with *Tracking module* to receive the operator position at the time of the irradiation.
  - C. Send source and operator data to *Monte Carlo module* for fast x-ray transport simulation using MC-GPU and receive the estimated organ and skin doses.
  - D. Send dosimetry results to *Display module* to inform the operator of the received radiation exposure.
- Repeat the process for each acquisition in the procedure.

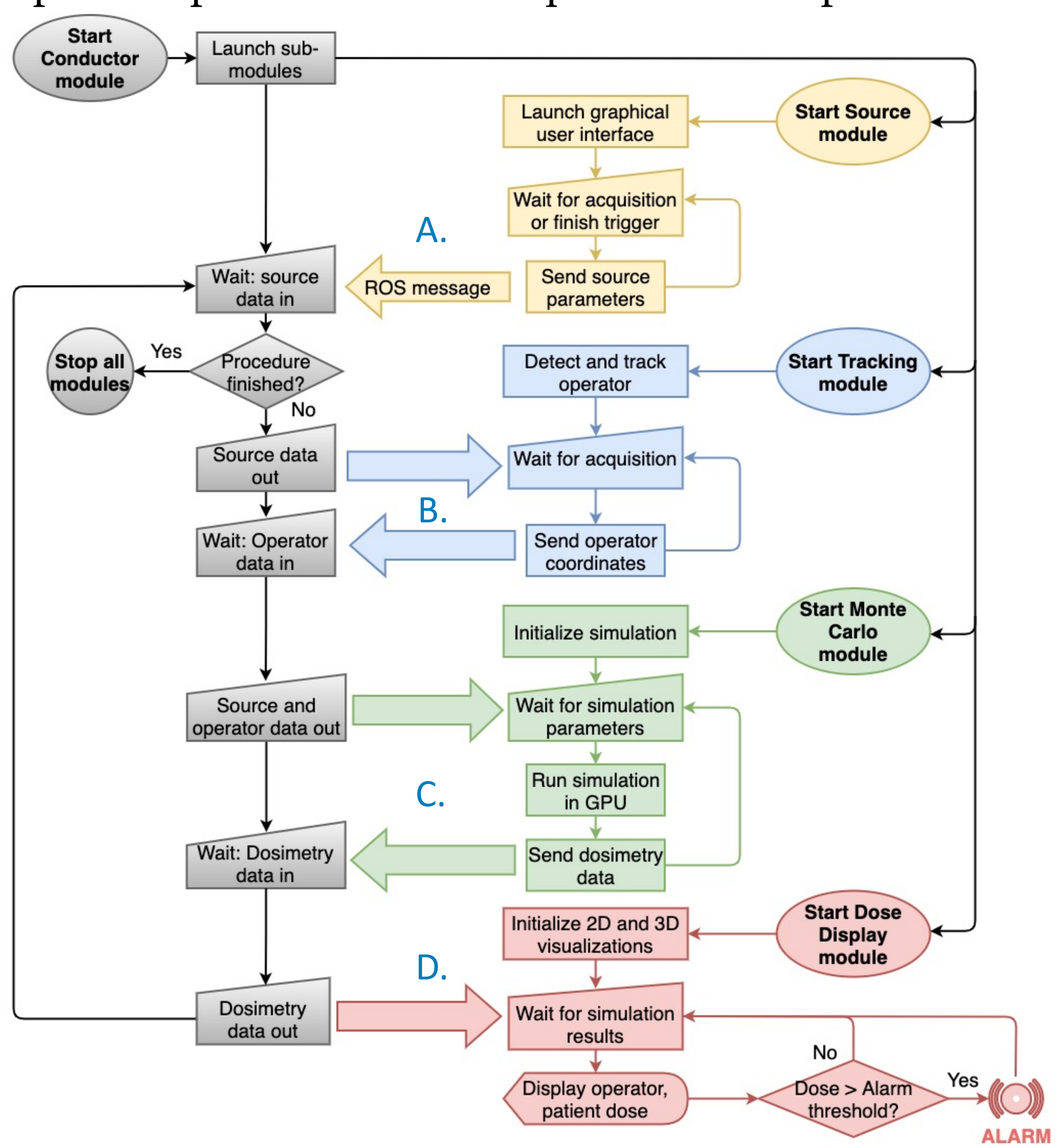

Figure 2. Flow chart of the *Virtual Dosimeter* software and the ROS messages exchanged between modules (steps A, B, C, D) using the new *Conductor* tool.

Discussion

- The new command-line *Conductor* application successfully communicates with the python and C++ modules with ROS messages.
- The modular design of the system allows testing different versions of the modules without affecting the execution of the complete system (for example, different technologies for operator location tracking).
- The initial *Display module* shows a 2D plot of the organ doses received by the operator in each exposure.

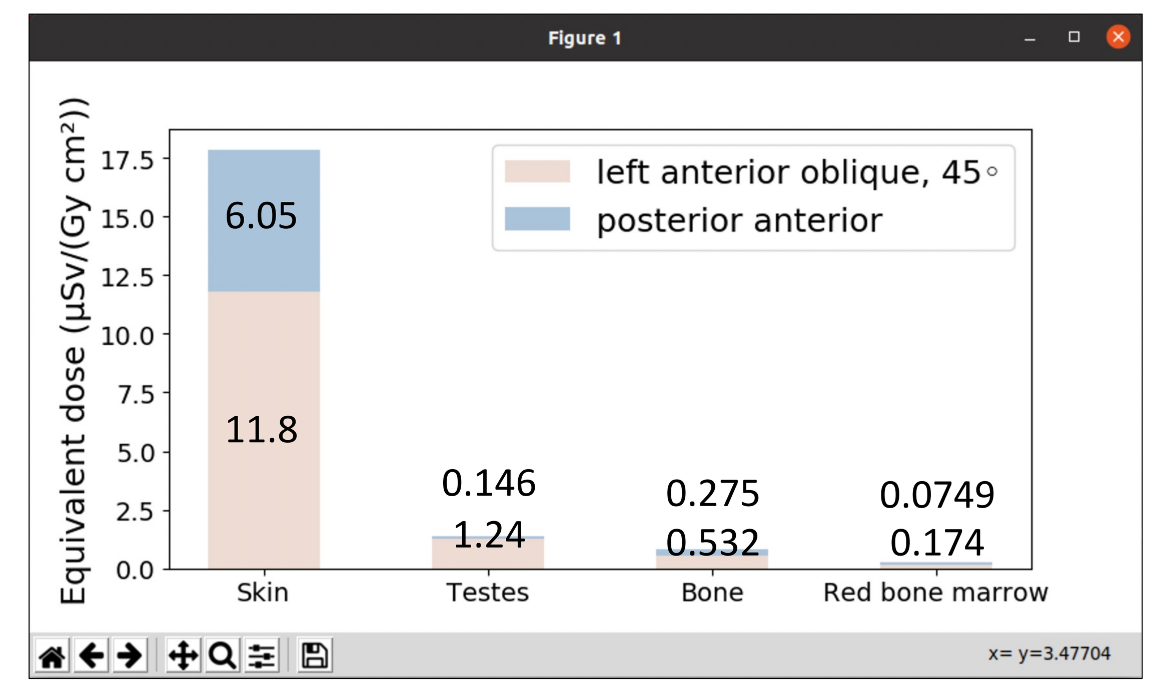

Figure 3. Bar graph of computed operator organ doses in two fluoroscopy views generated by the *Display module*.

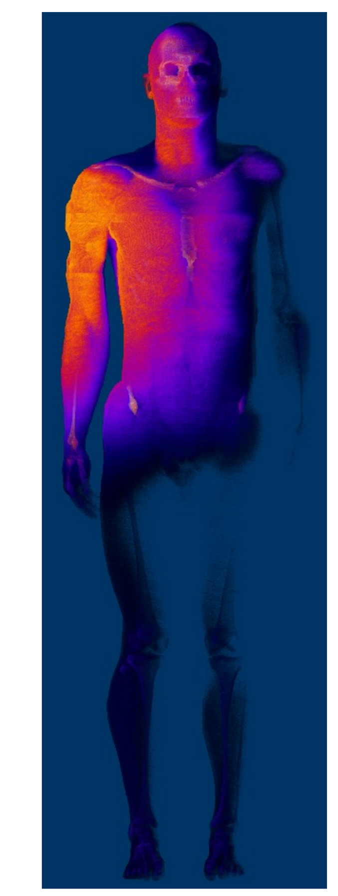

Figure 4. Example volume rendering of an operator dose distribution.

Conclusion and Future Work

- We have updated the *Virtual Dosimeter* software package with a new software module that uses the popular, open-source Robot Operating System library to coordinate the execution of the system.
- We are also developing a volume renderer for the *Display module* and implementing an alarm to warn operators of potential overexposures.

References

[1] Balter, S., Hopewell, J. W., Miller, D. L., Wagner, L. K., & Zelefsky, M. J. (2010). Radiology, 254(2), 326-341.

[2] Badal, A., Zafar, F., Dong, H., and Badano, A. In SPIE Medical Imaging 2013: Physics of Medical Imaging, 8668-28 (2013)

This project was supported by the 2021 Oak Ridge Institute for Science and Education (ORISE) Research Fellowship Program at the US Food and Drug Administration.