PyCNO: An Open-Source Python Framework for Computational Nuclear Oncology and Virtual Theranostic Trials

James Fowler[1,2], Maziar Sabouri[1,2], Taylor J. McColl[1,3], Tahir I. Yusufaly[4], Christian Anderson[5], Mark K. Transtrum[5], Arman Rahmim[1,2,3,6], Carlos Uribe[1,3,6]

1 Department of Basic and Translational Sciences, BC Cancer Research Institute, Vancouver, BC, Canada | 2 Department of Physics & Astronomy, University of British Columbia, Vancouver, BC, Canada
3 Department of Radiology, University of British Columbia, Vancouver, BC, Canada | 4 Department of Radiology and Radiological Sciences, Johns Hopkins, Baltimore, MD, USA
5 Cross Stream Bioanalytics, Salt Lake City, UT, USA | 6 Department of Molecular Imaging and Therapy, BC Cancer, Vancouver, BC, Canada

2026 JULY 19–22 VANCOUVER, BC
JOINT AAPM | COMP MEETING

INTRODUCTION

Computational nuclear oncology (CNO): the field concerned with the application of mathematical and computational models to describe, predict, and optimize the use of radiopharmaceuticals in imaging and therapy of cancer [1].

Virtual Theranostic Trial (VTT): *in silico* trial framework within CNO for the evaluation and optimization of radiopharmaceuticals for theranostic applications [2].

Theranostic Digital Twin (TDT): a virtual representation of an individual patient for the purpose of personalization of imaging and therapy [3].

PyCNO

Current Challenges in CNO

- **Methodological Fragmentation:** Rapid advancements in CNO are hindered by non-standardized workflows that impede reproducibility, cross-study comparison, and collaborative dissemination.
- **Translational Bottlenecks:** This lack of integration restricts the translation of physiologically based pharmacokinetic / pharmacodynamic (PBPK/PD) models into clinically optimized radiopharmaceutical therapy (RPT) and nuclear imaging protocols.

The PyCNO Framework

- **Unified Open-Source Architecture:** PyCNO provides a standardized, open-source platform designed to unify and benchmark diverse CNO methodologies.

METHODS AND MATERIALS

Technical Architecture & Implementation

- **Open-Source Python Framework:** PyCNO is a modular Python library designed for reproducible, standardized CNO workflows.
- **SBML-Based Model Specification:** Uses Systems Biology Markup Language (SBML) to encode mechanistic PBPK/PD models in a standardized format.
- **Fast Simulation:** Uses libRoadRunner [4] for efficient ordinary differential equation solving.
- **JAX Integration for Optimization:** Translates SBML models into JAX (using SBMLtoODEJAX [5]), enabling automatic differentiation for gradient-based parameter estimation and inverse-problem formulations.
- **Dual-Representation Paradigm:** Integrates high-performance forward simulations (via SBML) with robust gradient-driven optimization pipelines (via JAX).

Validation & Benchmarking

- **Performance Verification:** Evaluated for numerical accuracy and computational efficiency across both its SBML and JAX backend implementations.
- **Cross-Platform Benchmarking:** Verified against established, MATLAB SimBiology models to ensure computational fidelity.

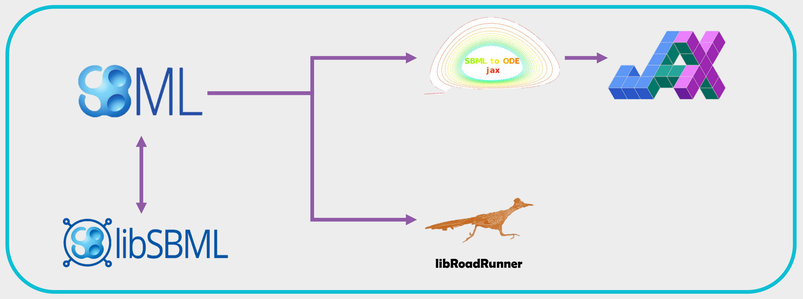

Figure 1. Library Orchestration.

```python
model_name = 'PSMA'
parameters ={
      'lambdaPhys': 0.000715,
      'bodyWeight': 70, #kg
      }
model = pycno.Model(model_name=model_name, parameters=parameters)
dose = pycno.Dose(
    times = [0],
    targets = {
        'Blood.Hot':[10],
        'Blood.Cold': [100]
    }
)
results = model.simulate(dose, stop=150, steps=1000)
times, tacs = results.times, results.tacs
```

Figure 2. Simple Python API.

RESULTS

- **Numerical Equivalence:** PyCNO demonstrated high computational fidelity, yielding results consistent with MATLAB SimBiology across both its SBML and JAX-based implementations.
- **Large-Scale Workload Performance:** In high-volume VTT workloads, PyCNO achieved computational throughput comparable to MATLAB SimBiology, showing that there is minimal orchestration overhead.
- **Single-Patient Simulation Efficiency:** For individual, single-patient simulations, PyCNO demonstrated superior execution speed compared to SimBiology due to a significant reduction in model initialization latency. The speed-up is compounded for iterative optimization pipelines.

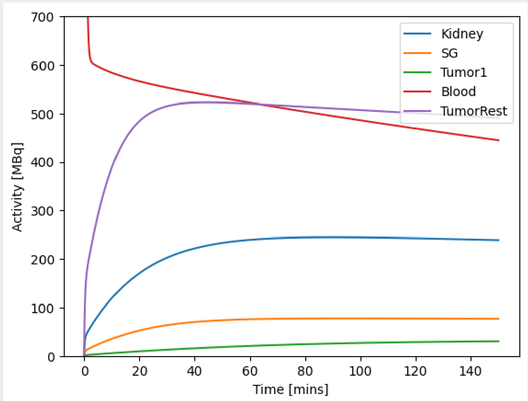

Figure 3. Example simulated time activity curves.

Number of Simulations	PyCNO Simulation Time (s)	Simbiology Simulation Time (s)
n = 1	0.008	0.278
n = 100	0.338	0.575
n = 1000	3.40	4.45
n = 10000	35	49

Table 1. Single-core simulation time performance.

CONCLUSIONS

Core Architecture & Capabilities

- **Extensible Open-Source Framework:** PyCNO is a modular platform for sharing, simulating, and fitting mechanistic CNO models.
- **Technical Integration:** Combines standardized model specifications with Python-based optimization for reproducible workflows.

Key Applications

- **In Silico Trial Support:** Enables the design and execution of VTTs.
- **Clinical Protocol Refinement:** Supports optimization of RPT dosing and scheduling.
- **Precision Medicine Infrastructure:** Provides a foundation for predictive, personalized TDTs.

REFERENCES

[1] T. Yusufaly *et al.*, "Computational Nuclear Oncology Toward Precision Radiopharmaceutical Therapies: Current Tools, Techniques, and Uncharted Territories," *J Nucl Med*, vol. 66, no. 4, pp. 509–515, Apr. 2025, doi: 10.2967/jnumed.124.267927.
[2] C. F. Uribe *et al.*, "Virtual Theranostic Trials: New Approach Methodologies (NAMs) and Precision Radiopharmaceutical Therapies," 2026, *arXiv*. doi: 10.48550/ARXIV.2602.10570.
[3] H. Abdollahi *et al.*, "Theranostic digital twins: Concept, framework and roadmap towards personalized radiopharmaceutical therapies," *Theranostics*, vol. 14, no. 9, pp. 3404–3422, 2024, doi: 10.7150/thno.93973.
[4] C. Welsh, J. Xu, L. Smith, M. König, K. Choi, and H. M. Sauro, "libRoadRunner 2.0: a high performance SBML simulation and analysis library," *Bioinformatics*, vol. 39, no. 1, p. btac770, Jan. 2023, doi: 10.1093/bioinformatics/btac770.
[5] M. Etcheverry, M. Levin, C. Moulin-Frier, and P.-Y. Oudeyer, "SBMLtoODEjax: Efficient Simulation and Optimization of Biological Network Models in JAX," 2023, *arXiv*. doi: 10.48550/ARXIV.2307.08452.000

BC CANCER | UBC | QURIT Quantitative Radiomolecular Imaging & Therapy | NSERC CRSNG

https://github.com/qurit/PyCNO