Python-Based Modernization and PWR Benchmarking of the GALE Methodology for Future SMR Effluent Assessment

*Joo Hyung Kim, Seung Beom Goh, Yong Kyun Kim**
Department of Nuclear Engineering, Hanyang University, Seoul, Korea
**Corresponding author: ykkim4@hanyang.ac.kr*

Keywords: PWR-GALE, radioactive effluent, code modernization, small modular reactor

1. Introduction

The Gaseous and Liquid Effluent (GALE) methodology for pressurized water reactors (PWRs) estimates annual liquid and gaseous radioactive releases using coolant source concentrations, system flows, hold-up times, decontamination factors (DFs), decay, and pathway-specific release assumptions [1]. It remains a useful design-basis method, but its legacy implementation is difficult to inspect, modify, and connect with contemporary data-processing and verification workflows. These limitations become more important when the method is used as a computational basis for future reactor concepts, including small modular reactors (SMRs), for which GALE applicability has been examined [2]. This study therefore reconstructed the conventional PWR-GALE calculation procedure as a modular Python code, expanded its input-variable framework, and evaluated the implementation against existing PWR design cases. The objective was code modernization and verification of GALE-compatible calculations, rather than direct validation of SMR-specific operating behavior.

2. Methods and Results

2.1 PWR-GALE calculation flow

As shown in Fig. 1, PWR-GALE combines input-card parameters, primary- and secondary-coolant concentrations, and nuclide constants. The liquid calculation applies stream-specific collection, storage, treatment, DFs, decay, and release fractions. The gaseous calculation evaluates noble-gas, iodine, particulate, H-3, and C-14 pathways using hold-up, filtration, purge, and discharge assumptions. The pathway results are then integrated into nuclide-specific annual release estimates.

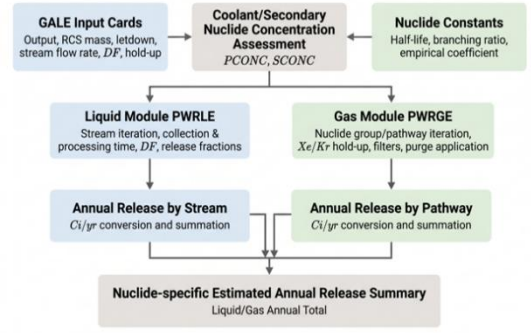

Fig. 1. Calculation flow of the conventional PWR-GALE

This calculation sequence and its principal engineering assumptions were retained as the reference logic for the Python implementation.

2.2. Python-based modernization

As shown in Fig. 2(b), the conventional PWR-GALE procedure was reorganized into modular liquid and gaseous kernels sharing a common nuclide-data interface. Input processing, source-term calculation, pathway treatment, radioactive decay, and annual summation were separated into traceable computational stages while preserving the conventional concentration-flow-DF-decay relationships.

Intermediate concentrations and pathway contributions were retained to support inspection of the calculation process.

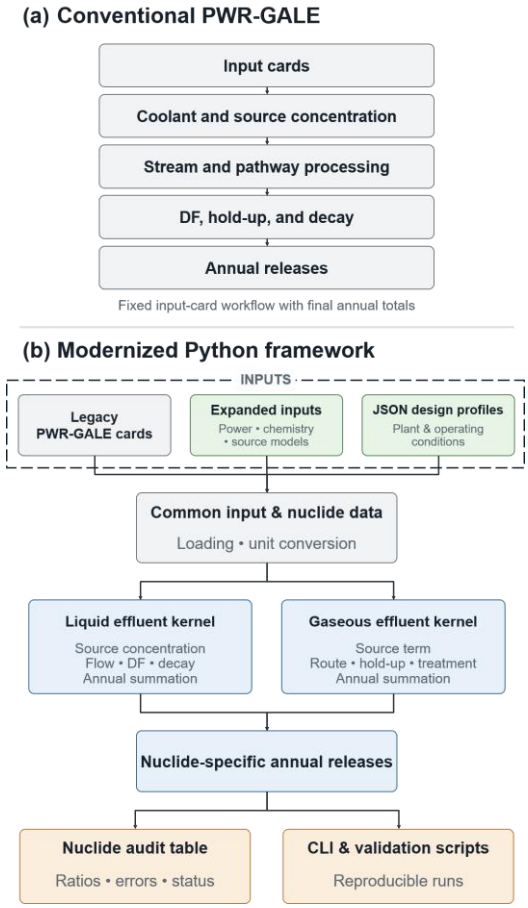

Fig. 2. Comparison of the conventional PWR-GALE workflow and the modernized modular Python framework

The legacy input-card system was extended to represent power and capacity-factor scaling, water chemistry, failed-fuel and CRUD source terms, H-3/C-14 source models and pathway partitioning, and liquid and gaseous treatment conditions. JSON-based design profiles allow plant and operating conditions to be changed without modifying the calculation kernels. This structure provides a computational basis for incorporating SMR-specific design characteristics in future applications.

2.3 PWR-based verification

The Python code was benchmarked using the design inputs and predicted releases reported for Shin-Kori Units 1 and 2 and Hanbit Units 5 and 6. Before comparison, PWR compatibility was checked against the original pgalegs.for treatment of primary-coolant source scaling, containment purge flow, iodine routing, and twice-yearly shutdown degassing. The comparison included only matched nuclides having valid positive annual-release values in both the code output and the FSAR reference. Figure 3 presents two complementary views of agreement. The blue circle is the median of the individual code-to-FSAR ratios and indicates how well a typical nuclide is reproduced without being dominated by its activity magnitude. The red diamond compares the sums of the matched releases and indicates how well the overall release magnitude is reproduced.

For both metrics, a value of 1 indicates agreement with the FSAR, a value below 1 indicates underprediction, and a value above 1 indicates overprediction. The green band denotes the factor-of-two (FAC2) model-performance range, $0.5 \leq \text{Code/FSAR} \leq 2.0$ [3], used here only as a screening criterion rather than a regulatory acceptance limit. The median and summed ratios were considered together because a few high-release nuclides, such as H-3, can dominate the total without similarly controlling the median. Because FSAR estimates include conservative and plant-specific assumptions, these measures were used as implementation benchmarks rather than exact reproduction requirements.

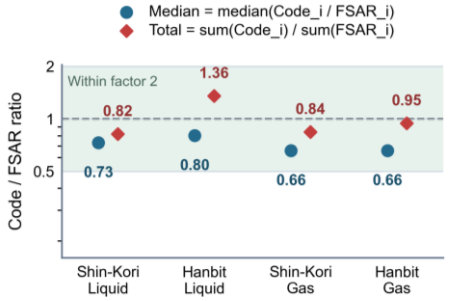

Fig. 3. PWR benchmark metrics for matched nuclides

As shown in Fig. 3, for liquid effluents, the median ratios were 0.730 for Shin-Kori and 0.802 for Hanbit, and the summed-release ratios were 0.820 and 1.357, respectively. Both markers therefore remained within the factor-of-two band. Moreover, 34 of 36 and 35 of 36 matched liquid nuclides were within a factor of 10. These results indicate that the liquid calculation reproduced both the typical nuclide-level pattern and the overall release magnitude reasonably well.

For gaseous effluents, the median ratio was 0.659 for both plants, while the summed-release ratios were 0.842 for Shin-Kori and 0.945 for Hanbit; 32 of 33 and 33 of 33 matched nuclides, respectively, were within a factor of 10. Correcting the original GALE mapping of primary-coolant source scaling, containment purge flow, iodine routing, and twice-yearly shutdown degassing removed the previous systematic underprediction of major Kr, Xe, and iodine releases. The sole factor-of-10 deviation was Shin-Kori H-3, reflecting a mismatch between the model's power-based H-3 source with 8% gaseous allocation and the plant-specific partition implicit in the FSAR. The revised benchmark therefore supports the PWR-GALE compatibility of the Python kernel while identifying H-3 source magnitude and phase partitioning as plant-specific inputs.

3. Conclusions

The conventional PWR-GALE methodology was reconstructed as a modular Python code for liquid and gaseous radioactive effluent assessment. The modernization preserved the principal GALE calculation logic while expanding the input-variable framework and improving intermediate-result traceability, nuclide-level auditing, and reproducible execution. PWR benchmarks reproduced the typical nuclide pattern and matched-release magnitude for both liquid and gaseous effluents. The remaining principal discrepancy was H-3, whose total source and liquid-gas allocation require plant-specific treatment. These results support the computational validity of the implementation as a GALE-compatible assessment tool. Because verification was limited to existing large PWRs, no direct claim is made regarding SMR-specific operating characteristics. Future work will develop input datasets reflecting SMR-specific design and operating characteristics and use the present framework to evaluate expected radioactive effluent releases from SMRs.

ACKNOWLEDGEMENTS

This work was supported by the Innovative Small Modular Reactor Development Agency grant funded by the Korea Government (MOTIE) (No. RS-2024-00400615).

REFERENCES

[1] U.S. NRC, Calculation of Release of Radioactive Materials in Gaseous and Liquid Effluents from Pressurized-Water Reactors: GALE-PWR 3.2 Code, NUREG-0017 Rev. 2, 2020.
[2] K.J. Geelhood and J.P. Rishel, Applicability of GALE-86 Codes to Integral Pressurized Water Reactor Designs, PNNL-21386, Pacific Northwest National Laboratory, 2012.
[3] J.C. Chang and S.R. Hanna, Air Quality Model Performance Evaluation, Meteorology and Atmospheric Physics, Vol. 87, pp. 167–196, 2004.