

실시간 대규모 데이터 처리 및 확률 기반 시뮬레이션에 특화된 RAVEN의 서버 환경에서의 활용 방안 연구

류시명¹, 김동완², 전찬혁³

¹ 경희대학교 응용물리학과 학부생

² 경희대학교 산업경영공학과 학부생

³ 한국정보산업연합회 연구원

simy1025@khu.ac.kr, tomo2398@khu.ac.kr, jeonch@fkii.org

A Study on the Application of RAVEN in Server Environment Specialized in Real-Time Large-Scale Data Processing and Probability- Based Simulation

요 약

RAVEN (Risk Analysis Virtual ENvironment)은 아이다호 국립 연구소(INL)에서 개발한 다변수 불확실성 정량화 및 실시간 위험 분석을 위한 시뮬레이션을 제공하는 프레임워크이다. RAVEN은 원자력 발전소의 위험 관리를 위한 소프트웨어로 개발되었지만 범용적인 위험 관리 소프트웨어로 기능한다. 기존의 계산 소프트웨어인 MATLAB이나 UQpy 등은 오픈소스가 아니거나 통합된 환경을 제공하지 않는 점 등의 약점이 있는데 반해 RAVEN은 그것에 강점이 있다. 본 연구에서는 RAVEN의 장점을 포함한 특징과 설치 및 구동에 대해 간단히 논하고, PBS 등을 이용한 대규모 실시간 병렬 처리 시스템 운영 구조를 논한다. 또한 RAVEN이 제공하는 기계학습 라이브러리를 응용할 방법을 살펴본다. 마지막으로 해당 프레임워크가 실시간 연료 효율 분석, 실시간 경로 탐색 등의 문제 상황에서 어떤 우위가 있는지 논한다.

1. 서론 및 배경

RAVEN (Risk Analysis Virtual ENvironment)은 확률적 모델링을 위해 다양한 프레임워크를 제공한다. 샘플링 기법과 통계 처리 기법을 제공하며 근래에는 기계학습을 위한 라이브러리도 추가되었다. 이 뿐만 아니라 필요한 경우에 직접 프레임워크를 생성하여 사용할 수 있다. 과학계산을 제공하는 이른바 MATLAB 등과 같은 프로그램과 비교하면 오픈 소스이며 커스터마이징이 가능하다는 측면에서 장점이 있다.

RAVEN은 에너지 시스템, 복잡계 시스템을 다루는 것에 특화되어 있다. 원자력 발전소 관리에는 다양한 지점에서 얻은 데이터의 확률 분석과 불확실성 정량화 수행이 필요한데, 이에 대응하여 개발된 RAVEN은 실시간 데이터 처리에 장점이 있다.

한편 시스템이 데이터를 얻는 물리적 공간이 원자력 발전소뿐만 아니라 더 넓어지게 되면, 네트워킹 기술이 중요해질 것이다. 가령 예를 들어 자율주행 시스템에 RAVEN을 적용한다면, 개별 차량에서 얻은 데이터를 중앙 서버실에서 처리하거나 개별 객체(차량)에서 처리하는 것 모두가 가능한 구조일 것이다.

따라서 본 연구에서는 Raspberry Pi 및 Windows 상에서 RAVEN을 구동하는 예시를 보일 것이다. 한편 RAVEN에서 기계학습을 시행할 때 PyTorch나 TensorFlow가 제공하는 GPU 연산 핸들링을 이용할 수도 있을 것이다. 그러나 이는 얻어지는 데이터의 종류와 양에 따라 필요성이 달라질 것이므로 실제 적용 시에 고려해야 한다.

2. RAVEN의 기능

먼저 입출력 구조를 보면, XML 파일을 받아서 처리하며 각 프레임워크들은 Python으로 구성되어 있다. 출력 시에는 시간 종속성이 없는 단일 샘플 또는 시계열 데이터가 제공된다. 파일 유형은 CSV, HDF5, Plot이 있다.

주된 기능은 시뮬레이션 기반 가상 환경 제공이다. 다양한 샘플링 기법을 지원하며 물리 모델 및 외부 코드와 연동이 가능하다. 또한 Surrogate Model 생성을 지원한다.

PostProcessor를 통해 RAVEN 내부에서 통계, 민감도, 신뢰도 등의 분석을 수행하고 해당 데이터를 기계학

습용으로 사용할 수 있다. 기계학습을 위해 Scikit-Learn, PyTorch, TensorFlow 를 각각 제공한다. PyTorch 와 TensorFlow 는 Lazy-Loading 을 통해 연동되어, 프로그램 시작 시 불필요한 라이브러리 로딩으로 인한 오버헤드를 줄여준다.

3. RAVEN 환경설정 및 설치

RAVEN 을 Windows 환경에서 실행할 경우 WSL2 설치 후 Conda 상에서 구동이 가능하다. Miniconda 환경에서 실행되므로 Anaconda 가 아닌 Miniconda 를 설치한다. 하지만 bash 스크립트, 서버 기능 등을 고려해 Linux 실행이 권장된다. WSL2 는 Windows 커널 위에서 Linux 커널을 가상화하는 것이므로 기능을 모두 사용할 수 없다. RAVEN 기능을 모두 사용하려면 듀얼 부팅을 통한 Linux 부팅이 필요하다.

제대로 설치가 되었다면 테스트 구동이 가능하다. 기본적으로 제공되는 몬테카를로 샘플링 예시를 구동하면 다음과 같은 출력물을 볼 수 있다.

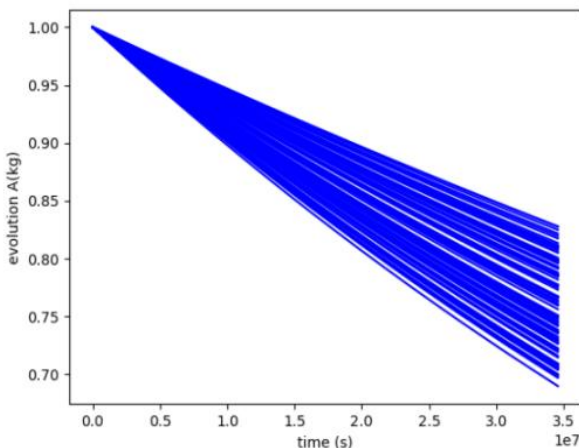


그림 1. 몬테카를로 샘플링 결과물. 시간 경과에 따른 물질 A의 변화량을 나타낸다.

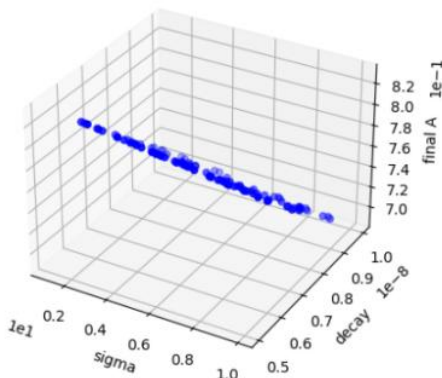


그림 2. 몬테카를로 샘플링 결과물. 입력 파라미터인 sigma-A 와 decay-A 값을 정의된 분포에서 무작위로 추출하여 최종 물질 A 양 간의 관계를 보여주는 산점도. 각 파라미터는 커스터마이징이 가능하다.

4. 병렬 컴퓨팅을 통한 RAVEN 구동

PBS, SLURM 등과 연계하여 다중 노드, 다중 쓰레드 등을 통한 병렬 컴퓨팅 및 HPC 환경에서의 연산을 지원한다. 이를테면 각 컴퓨터에서 NFS 프로토콜을 이용하여 동일한 구조의 디렉토리 공유를 하게 되면, PBS 클러스터 스케줄러가 여러 노드의 Job 을 자동 분산할 수 있다. 이 또한 Linux 환경에서 실행해야 PBS 기능을 온전히 사용할 수 있다.

5. RAVEN 의 응용 방안

실시간 위험분석 및 시뮬레이션을 응용하면 원자력 발전소뿐만 아니라 다양한 곳에 RAVEN 을 활용할 수 있다. 이를테면 GPS 신호 차단 현상이 우려되는 북극해 지역에서 선박의 위치추적이 중단되었을 때 선박의 위치를 추적하는 시뮬레이션 시스템을 구상할 수 있다. 또한 위치추적에 따른 실시간 항로 탐색도 가능하다. 시뮬레이션 결과는 기계학습의 입력 데이터로 재활용되어, 로컬 및 중앙 서버에서의 시뮬레이션과 기계학습의 연동 운영을 가능하게 한다. 이를 통해 선박의 연료 효율을 고려하여 운항 속도 및 항로 수정 등이 가능할 것이다. RAVEN 에서는 시스템의 한계를 탐색하는 기능도 있으므로 시스템의 멈춤을 방지하는 조건을 실시간으로 탐색할 수 있다.

6. 향후 연구 방안

RAVEN 은 Linux 듀얼 부팅 시에 온전한 기능을 발휘하므로 Linux 환경에서 실험할 것이다. PBS 내의 qsub 명령어가 클러스터 환경에서 Job-Handling 역할을 하는데, 이는 이미 구축된 클러스터 환경에서 기능하므로 AWS 등을 이용한 환경 구축 후에 병렬 실행을 하는 것이 목표다. 병렬 실행을 한다면 응용 방안에서 언급한 것들을 수행할 수 있다.

한편 변수가 추가되는 상황에서 기계학습은 한계를 드러내는데, 시뮬레이션을 통해 변수가 추가되거나 제거되는 상황에서도 지속적인 학습과 평가가 가능하다. RAVEN 의 외부 물리계 통합 기능을 사용하면 미분방정식을 사용할 수 있는데, LLM 다음 세대 연구 주제로 제시되는 PINN(Physics-Informed Neural Network)의 활용 또한 RAVEN 환경에서 가능할 것이다.

참고문헌

- [1] Andrea Alfonsi 외, RAVEN and Dynamic Probabilistic Risk Assessment: Software Overview, INL
- [2] Tao Liu 외, A best estimate plus uncertainty safety analysis framework based on RELAP5-3D and RAVEN platform for research reactor transient analyses, Progress in Nuclear Energy 132 (2021)
- [3] RAVEN_THEORY_MANUAL, INL
- [4] RAVEN_USER_MANUAL, INL