**1. Deployed Optimization Algorithms with Source Code Repositories**

**Title:** *Version-Controlled, Documented ML & Optimization Algorithm Repositories*

**A. Repository Structure**

/smart-5g-optimization

│

├── /algorithms

│ ├── power\_allocation\_rl/

│ │ ├── README.md

│ │ ├── model.py

│ │ ├── train.py

│ │ ├── inference.py

│ │ └── requirements.txt

│ ├── traffic\_prediction\_dnn/

│ │ ├── README.md

│ │ ├── model.py

│ │ ├── train.py

│ │ ├── preprocess.py

│ │ └── requirements.txt

│ └── fault\_detection\_rf/

│ ├── README.md

│ ├── model.py

│ ├── train.py

│ └── requirements.txt

│

├── /docs

│ ├── algorithm\_design\_specifications.pdf

│ ├── data\_preprocessing\_guidelines.md

│ └── deployment\_instructions.md

│

├── /tests

│ ├── unit\_tests/

│ ├── integration\_tests/

│ └── performance\_tests/

│

└── LICENSE

**B. Version Control & Documentation**

* **Git platform:** GitHub / GitLab with branch protection, pull requests, and CI checks.
* **Documentation:**
  + README files per algorithm explaining purpose, architecture, usage.
  + Docstrings in code following Google or NumPy style.
  + Deployment and environment setup guides.
* **CI/CD Integration:** Automated testing triggered on push to branches.

**C. Code Quality & Security**

* Linting with tools like pylint or flake8.
* Dependency scanning for vulnerabilities (e.g., Dependabot).
* Unit and integration tests covering >80% of code.