TwinAI-Demo: Real-Time Digital Twin and Graph Reinforcement Learning for Interactive Water Distribution Network Management

Pierluigi Locatelli*, Pietro Spadaccino*, Marco Casalbore*, Tiziana Cattai*, Stefano Servillo*, Francesca Cuomo*.
*Department of Information Engineering, Electronics, and Telecommunications, Sapienza University of Rome.
Email: {name.surname}@uniroma1.it

***Abstract*—This demo presents the functioning of TwinAI, an interactive framework for real-time management of Water Distribution Networks (WDNs) that integrates a dynamic digital twin with a graph-based reinforcement learning agent. Unlike traditional offline simulators, TwinAI enables live interaction with an ongoing hydraulic simulation, allowing operators to visualize network states, inject anomalies, and execute control actions directly from a Graphical User Interface (GUI). The demo showcases real-time operations, leak isolation, simulation branching for what-if analysis, and autonomous decision-making driven by a Graph-RL agent. The system demonstrates how physically consistent digital twins and learning-based control can be combined for adaptive WDN operation and for interactive what-if analyses.**

***Index Terms*—Water Distribution Networks, Digital Twin, Graph Reinforcement Learning**

I. Introduction

Efficient management of Water Distribution Network (WDN) is critical due to water scarcity, aging infrastructure, and limited monitoring capabilities [1]. These large-scale cyber-physical systems exhibit complex hydraulic dynamics governed by topology, demand, and operational constraints. However, obsolete components and sparse sensing result in water losses exceeding one-third of treated supply. While IoT deployments provide pressure and flow measurements at select locations, cost and accessibility constraints restrict coverage, forcing operators to manage under partial observability. Consequently, widely used hydraulic simulators like EPANET [2] and Python-based tools such as Water Network Tool for Resilience (WNTR) [3] remain confined to offline, scenario-based analyses lacking real-time interaction with evolving systems. This static paradigm hinders operational responsiveness to unpredictable anomalies like leaks. Although numerous model-based and data-driven leak mitigation strategies exist, they are rarely integrated into closed-loop, physically consistent control frameworks, leaving a gap in real-time adaptive WDN management solutions.

TwinAI addresses this gap by coupling a dynamic digital twin of the WDN with an autonomous graph-based reinforcement learning agent. The digital twin, implemented through Dyn-WNTR, a dynamic extension of the EPANET-based WNTR simulator [4], enables event-driven hydraulic simulation and online interaction, while the learning agent computes control actions directly on the network graph. Conventional hydraulic simulators and passive digital twin implementations lack continuous physical synchronization and closed-loop operational control. TwinAI addresses this limitation by implementing a bidirectional cyber-physical architecture that continuously aligns streaming sensor observations with a dynamic virtual model, enabling real-time anomaly injection, deterministic what-if branching, and immediate hydraulic response evaluation within an interactive management framework This demo focuses on the interactive dimension of TwinAI, showcasing a graphical user interface that allows real-time visualization, anomaly injection, manual intervention, and autonomous control, enabling human-in-the-loop operation within a physically consistent framework.

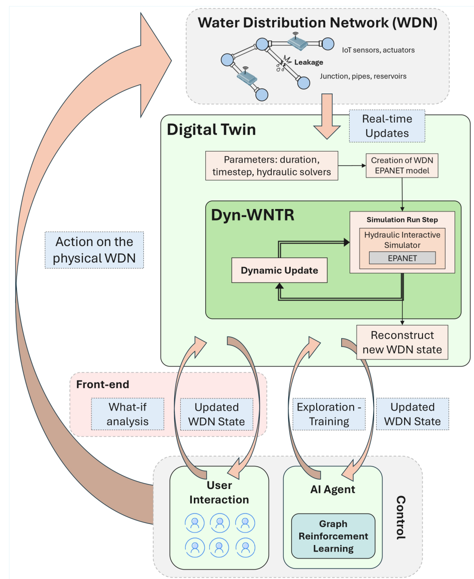

Fig. 1. Architecture of the proposed system containing the digital twin and control components.

II. TwinAI Architecture

This section describes the architecture of the TwinAI framework, depicted in Fig. 1. The framework consists of two

This work was supported by the European Union- Next Generation EU under the Italian National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.3, CUP B53C22004050001, partnership on "Telecommunications of the Future"(PE00000001 - program "RESTART")
ISBN 978-3-903176-82-9

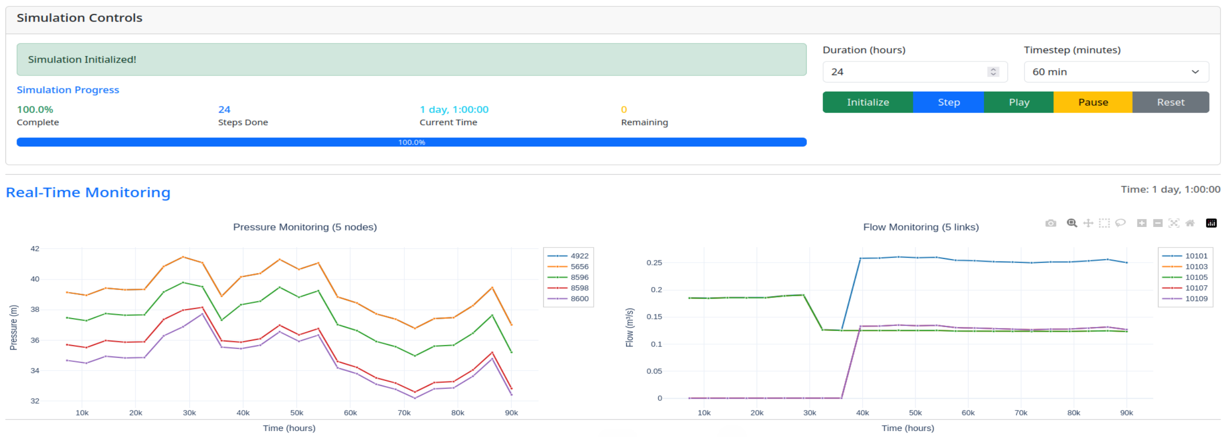

Fig. 2. Partial GUI of the TWIN-AI demo operating as a real-time hydraulic simulator on a pre-loaded water distribution network. The interface supports interactive configuration of the simulation horizon and time resolution, execution control (initialization, step-wise advance, play/pause/reset), and real-time visualization of network statistics through dedicated charts in both single-component and aggregated modes.

Fig. 3. GUI of a newly loaded WDN into the system.

interdependent modules: a Digital Twin representing the WDN and an autonomous AI agent. The system is deployed over a physical WDN instrumented with IoT and LPWAN sensors that continuously measure hydraulic variables, including pressure and flow. The physical infrastructure may experience operational irregularities such as leakages or component malfunctions. The Digital Twin component maintains a dynamic virtual model of the WDN, capturing both its hydraulic and topological states via streaming sensor data. Its internal representation includes network topology, pipe and junction properties, reservoir conditions, nodal demands, and hydraulic variables such as pressures, flows, and water levels. When a control intervention is considered, such as valve adjustment or flow reconfiguration, the Digital Twin incorporates the action into its model and evaluates the resulting hydraulic behavior using the Dyn-WNTR simulator. The second component is the AI agent, which autonomously supervises the WDN by detecting and addressing anomalies, including leakages, through actions such as pipe isolation and flow redistribution. The agent employs Double DQN with NoisyNet exploration, initially guided by an annealed flow-threshold mask. Training utilizes a stratified prioritized replay buffer and Huber loss optimized via RMSprop, while the WDN is encoded as a bidirectional graph where pipe states gate neighbor aggregation in a message-passing GNN. Graph reinforcement learning is selected over conventional approaches as it natively encodes topological dependencies, whereas standard architectures lack the structural awareness required to model complex network interactions. The agent operates in closed-loop interaction with the Digital Twin. For each candidate action, the Digital Twin simulates the corresponding hydraulic response, updates its internal state, and returns the resulting network configuration to the agent, which then computes performance metrics and derives a reward signal. This realizes the action–state–reward structure underlying Reinforcement Learning (RL) methods. During training, the agent learns a function $f(s_t, a, s_{t+1}) \rightarrow r_t$, where s_t denotes the current state, a the selected action, s_{t+1} the subsequent state generated by the Digital Twin, and r_t the associated reward. TwinAI maintains continuous interaction with the physical WDN. Actions determined by the AI agent are implemented via IoT-enabled actuators, including valves and pumps. The system state evolves in response to both control actions and external influences such as demand variability or newly arising faults. To preserve alignment between virtual and physical domains, the Digital Twin is continuously synchronized with real-time sensor observations. The control component receives updates from the Digital Twin and acts on the WDN in one of two modes. In user-based control, a front-end interface allows operators to issue commands and perform what-if analyses, observing outcomes simulated by the Digital Twin. In AI-based control, a Graph-Reinforcement Learning (GRL)-driven agent autonomously minimizes water losses by exploring the action space while the Digital Twin provides the resulting hydraulic states. All the experiments and trainings were performed using Python with PyTorch (CUDA-accelerated), Dyn-WNTR for hydraulic simulation, NetworkX for graph analysis, and NumPy for numerical computation, running on an Intel 12th Gen Core i7-12650H (10 cores / 16 threads, up to 4.7 GHz), NVIDIA RTX 4070 Laptop GPU (8 GB GDDR6), and 15 GB DDR5 RAM.

III. Demo Description

The demonstration platform enables operators to interact with TwinAI as an integrated environment for real-time analy-

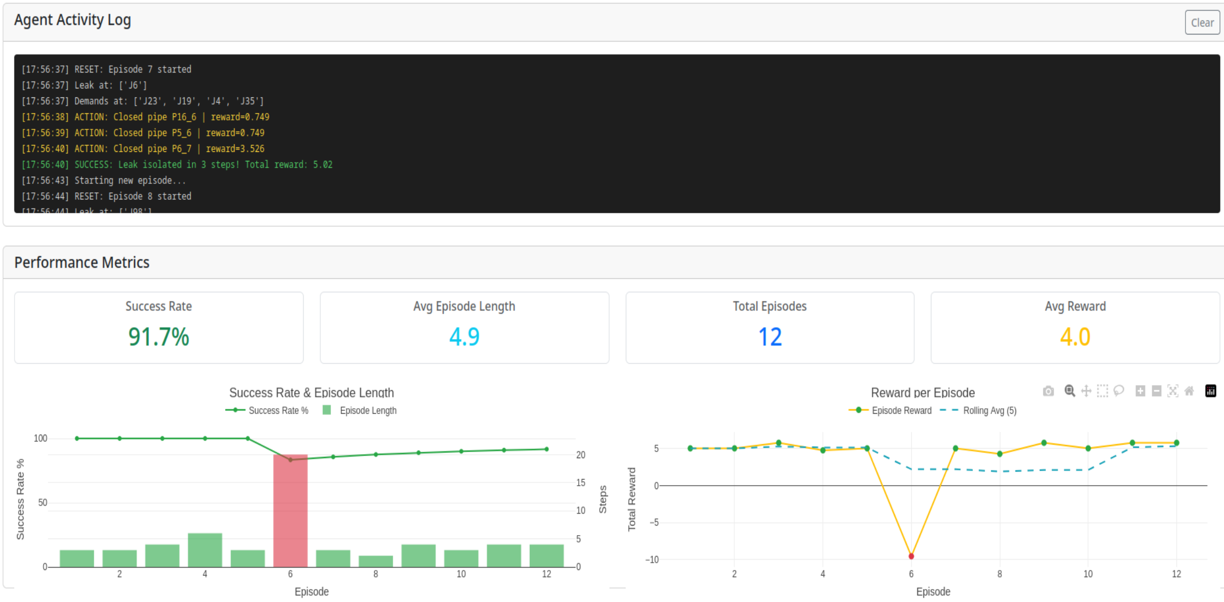

Fig. 4. Partial GUI of the TWIN-AI demo in GDRL mode, where an agent initialized from a pre-trained policy interacts with the hydraulic network to autonomously isolate leaks through control actions. Panels display real-time action logs and agent performance indicators across simulation episodes.

sis, simulation, and control of water distribution networks. The framework provides a set of pre-loaded benchmark networks with progressively increasing size, structural complexity, and hydraulic heterogeneity, allowing users to explore system behavior across different operating conditions. In addition, users can import custom EPANET `.inp` files, enabling direct application of the platform to real-world WDN models without requiring model redefinition or conversion.

Once a network is selected, the user is provided with an interactive visualization of its topology and component attributes, including nodes, pipes, pumps, valves, and storage elements (Fig. 3). At this stage, the static configuration of the system can be inspected, and simulation parameters can be specified. These include the simulation time horizon, hydraulic time step, and optional control intervals. After configuration, the user can initialize and manage the execution of a live hydraulic simulation driven by the digital twin.

During runtime, the platform continuously updates and displays the evolution of key hydraulic variables, such as nodal pressures, link flows, tank levels, and component operational states. These variables are accessible through dedicated visual panels that support both fine-grained inspection at the level of individual elements and aggregated summaries at the network scale (Fig. 2). This dual-level monitoring facilitates both detailed diagnostics and high-level situational awareness.

The platform further provides an interaction interface through which users can inject events dynamically during simulation execution. Supported actions include the introduction of one or multiple leaks at selected junctions, as well as direct manipulation of network components such as opening or closing valves and isolating pipes. Each action is processed in real time by the digital twin and incorporated into the ongoing simulation without interruption. The resulting system response is immediately observable, allowing users to assess the impact of interventions under physically consistent hydraulic dynamics.

Beyond manual operation, the demonstration includes an autonomous control mode based on a pre-trained graph reinforcement learning (GRL) agent. When activated (Fig. 4), the agent observes the current network state and executes sequential control actions on the graph-structured system. Its objective is to mitigate water losses by identifying and isolating leaks, while preserving service continuity. The digital twin provides the agent with a consistent and continuously updated environment, ensuring that control decisions are evaluated against realistic system dynamics.

Finally, the platform supports simulation branching, a feature that enables advanced what-if analysis. At any point during execution, the current simulation state can be paused and cloned into multiple independent instances. Each branch can then be evolved under different control strategies or disturbance scenarios, while sharing identical initial conditions. This capability allows operators to compare alternative intervention policies efficiently, without the need to restart simulations from the initial state, thereby reducing computational overhead and enabling rapid decision support.

References

[1] J. Liu, H. Yang, S. N. Gosling, M. Kummu, M. Flörke, S. Pfister, N. Hanasaki, Y. Wada, X. Zhang, C. Zheng, *et al.*, "Water scarcity assessments in the past, present, and future," *Earth's future*, vol. 5, no. 6, pp. 545–559, 2017.

[2] L. A. Rossman *et al.*, "Epanet users manual," 1994.

[3] K. A. Klise, D. Hart, D. M. Moriarty, M. L. Bynum, R. Murray, J. Burkhardt, and T. Haxton, "Water network tool for resilience (wntr) user manual," tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2017.

[4] P. Locatelli, T. Cattai, S. Palumbo, and F. Cuomo, "Dyn-wntr: Dynamic network adaptive extension for hydraulic simulations with wntr," in *2025 IFIP Networking Conference (IFIP Networking)*, IEEE, 2025.