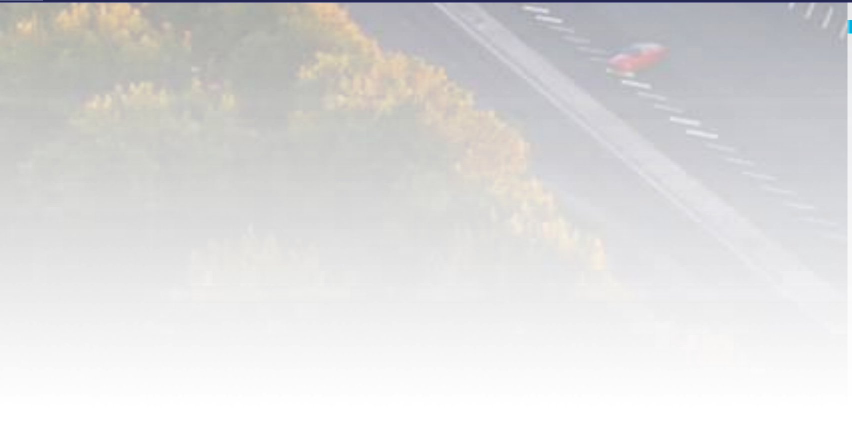
Recent updates to the Water Network Tool for Resilience software



Kate Klise, Dave Hart, Joe Hogge, Daniel Villa, Terra Haxton

World Environmental & Water Resources Congress

June 7-11, 2021

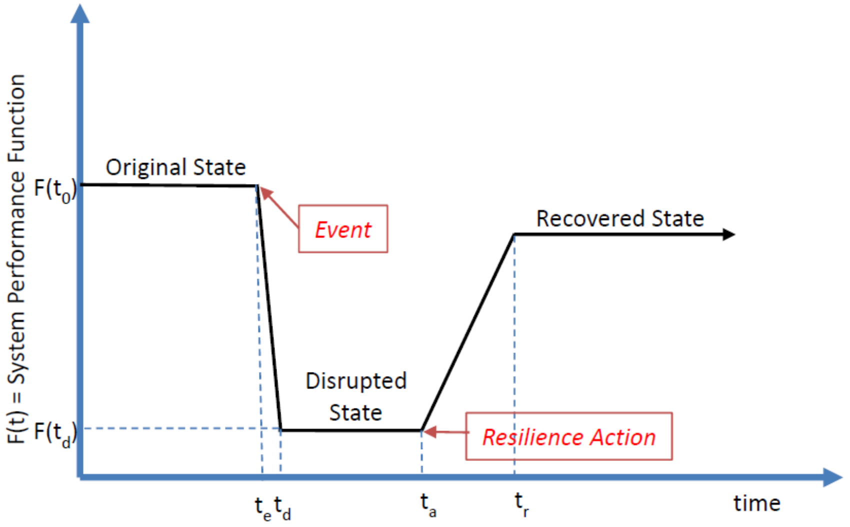
Outline

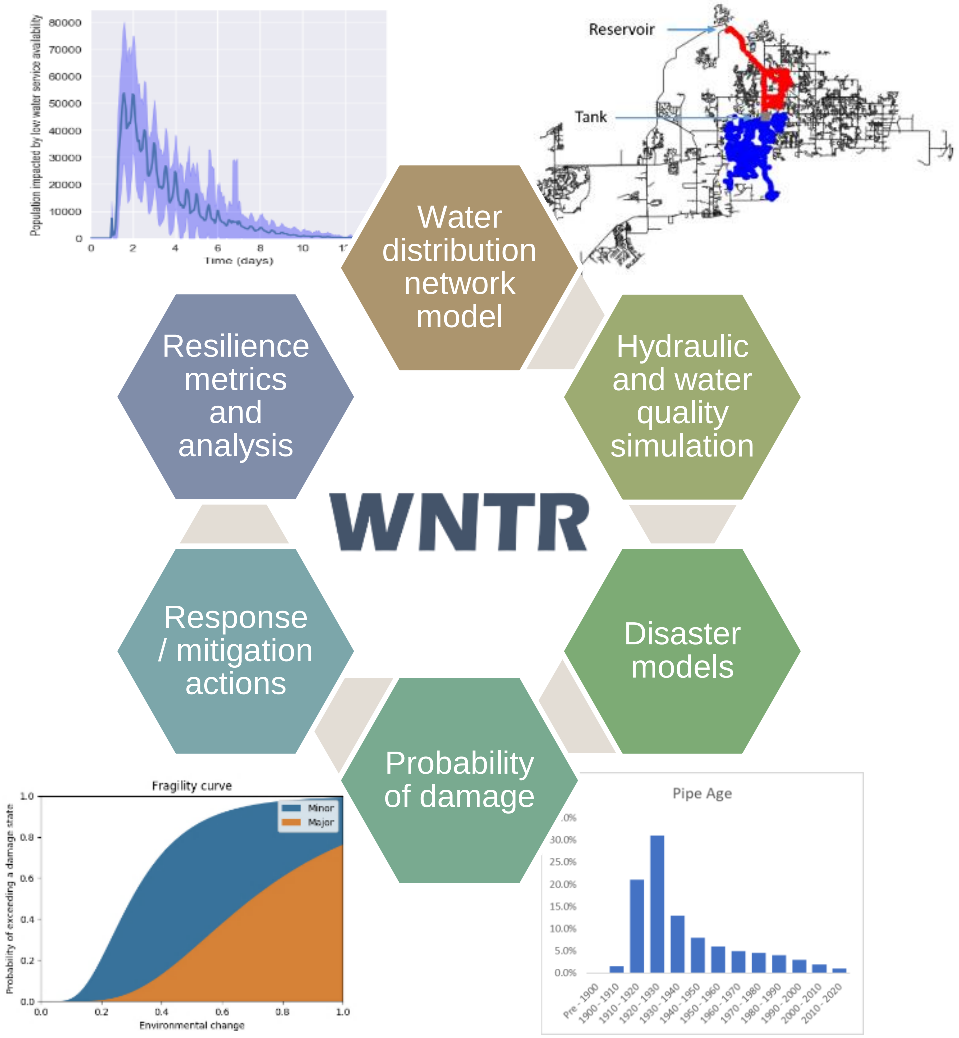
- WNTR overview
- User and developer community
- Recent updates
- Future development and ways to contribute

Water Network Tool for Resilience (WNTR)

WNTR is a Python package designed to analyze water distribution system failure and recovery

- First released in 2016
- Builds on EPANET
- Evaluate disruptions based on a wide range of hazards
- Evaluate and prioritize resilience-enhancing actions
- Quantify resilience using topographic, hydraulic, and water quality metrics
- Available through the EPA GitHub organization





https://github.com/USEPA/WNTR
http://wntr.readthedocs.io

Utility Specific Questions

What type of infrastructure damage could be caused by:

- A magnitude 7 earthquake (e.g., Napa Valley, CA 2014)?
- A hurricane (e.g., Harvey in TX 2017, Irma/Maria in PR 2017)?
- A regional power outage (e.g., Northeast Blackout 2003, Winter Storm 2021)?
- A contamination incident (e.g., Flint MI 2014, Elk River Spill in WV 2014)?
- A tornado (e.g., Joplin, MO 2011)?
- A cyber attack (e.g., Oldsmar, Florida 2021)?

How long can the system continue to provide water to customers?

How many people will be impacted?

Which critical facilities/functions will be impacted?

What is the best response in the immediate aftermath?

Which components should be hardened to minimize future disruptions?

Simulation and analysis tools can help water utilities explore how their system will respond to expected, and unexpected, events

Who is using WNTR? And what are they using it for?

Arcadis - Earthquake resilience analysis for Seattle

Alan Turing Institute – Coupled human and natural critical ecosystems analysis, chance-water-distribution

Architecture and Building Systems Research Group Switzerland - Urban energy simulation platform, CityEnergyAnalyst

Boise State University - Cyber resilience

Case Western Reserve University – Vulnerability analysis across water/transportation infrastructure using hydraulic and topographic metrics

Xylem (CitiLogics) – Ensemble simulation framework, CloudBurst

Delft University of Technology, Singapore University of Technology and Design – Cyber resilience, epanetCPA

Environmental Protection Agency – Military 14-day Water Security Initiative, CriticalityMaps, and premise plumbing analysis, PPMTool

Global Quality Corp – Sensor placement optimization

KIOS Research and Innovation Center of Excellence, University of Cyprus, benchmark dataset for leakage diagnosis, LeakDB

Los Alamos National Laboratory, National Renewable Energy Laboratory – Critical infrastructure resilience, WaterModels.jl

National Institute of Standards and Technology, Colorado State University,

University of Illinois – Community resilience, pyincore

National Technical Univ. of Athens, KWR Water Research Institute Netherlands – Cyber resilience, RISKNOUGHT

Naval Postgraduate School, Virgin Island Water and Power Authority - Critical infrastructure resilience and operations research for island communities and military installations

North Carolina University – Leak detection using neural networks

Rice University – Distributed direct potable water reuse and resilience analysis for Houston

Sandia National Laboratories – Microgrid design for energy-water systems and mission assurance

University of California, Irvine – Graph theory approach to fault identification under disruption scenarios, AquaEIS

University of California, Los Angeles – Earthquake resilience

University of South Florida – Prioritizing component repair across water and transportation infrastructure in Tampa

University of Texas, Austin – Hydraulic analysis for Austin and transient pressure analysis, TSNet

Plus…WNTR has been used in numerous Masters and PhD thesis

* Related software packaged in blue

WNTR User and Developer Community: By the Numbers

Over 95,000 downloads through PyPI and conda-forge

600 active users on the documentation website each month

- Top 50% of users from US (25%), China (12%), Japan (5%), India (4%), and Germany (4%)

Over 100 posts from users with issues, feedback and suggestions

- Support to users trying to debug issues related to EPANET and WNTR

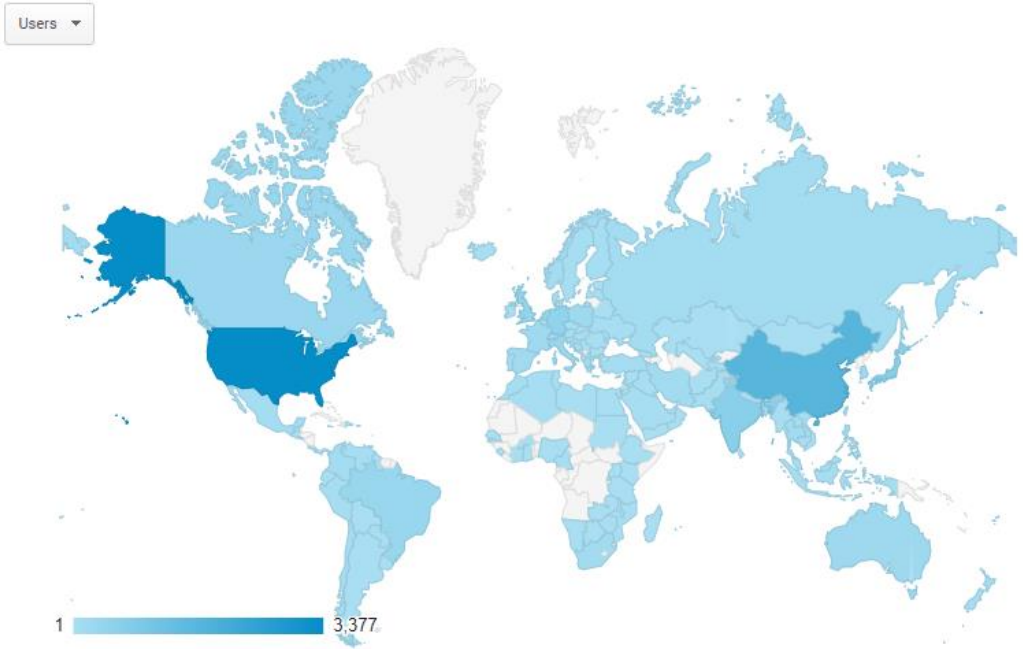
15 contributors from Sandia, EPA, and university/industry

110 forks (potential contributors)

12 software releases since 2016

Over 70 publications referencing WNTR

12 software packages building on components in WNTR



Recent Updates

EPANET 2.2 integration

Valve segmentation and related metrics

Multipoint pump and tank curves

Updated graphics capabilities

Code cleanup, helper functions, and additional testing

EPANET 2.2 Integration

WNTR includes two simulation engines

- EpanetSimulator – Calls EPANET
- WNTRSimulator – Custom WNTR simulator

The EpanetSimulator can now call **EPANET 2.00.12** and **EPANET 2.2**

- EPANET 2.2 includes PDD hydraulic simulation and tank overflow

```
>>> import wntr
>>> wn = wntr.network.WaterNetworkModel('Net3.inp')
>>> wn.options.hydraulic.demand_model = 'PDD'
>>> sim = wntr.sim.EpanetSimulator(wn)
>>> results = sim.run_sim(version=2.2)
```

Considerations when selecting a simulation engine

- **Speed**: EpanetSimulator is faster
- **Features**: headloss coefficient, leak model, water quality analysis, custom controls, tank overflow, spatially variable minimum pressure/required pressure/pressure exponent.
- **Advanced use cases**: WNTRSimulator's AML can be used to test out new equations and covert the model to Pyomo for optimization

	EpanetSimulator	WNTRSimulator
Code	EPANET 2.00.12 and 2.2 DLLs + Python wrappers	Python/C++ AML
Hydraulic simulation	DD/PDD Global minimum and required pressure, and pressure exponent	DD/PDD Global or **spatially variable** minimum and required pressure, and pressure exponent
Headloss equation	H-W, **D-W, C-M**	H-W
Valves	PRV, PSV, **PBV**, FCV, TCV, **GPV**	PRV, PSV, FCV, TCV,
Leak option	**Emitters**	**Leak model**
Pump curves	Multi-point curves are fit using **straight line segments**	Multi-point curves are fit using the **3-point equation**
Water quality simulation	**Concentration, Water age, Trace**	**None**
Controls/Rules	Trigger on time/clocktime or using **select** element attributes	Trigger on time/clocktime or using **any** element attribute
Tank overflow	**Option included**	**None**
Start/Stop capabilities	**Simulation always restarts from time 0**	**Simulation can resume from last timestep**

Simulator Testing

Integration of PDD into the EpanetSimulator facilitates more testing between the EpanetSimulator and WNTRSimulator

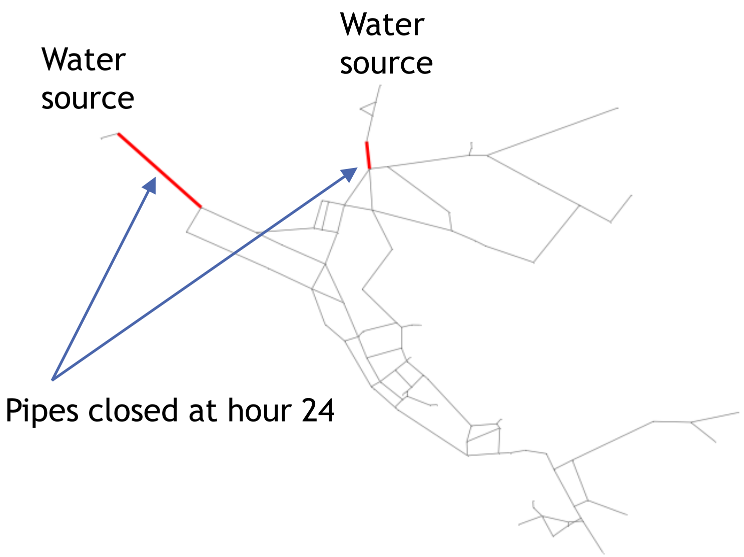
Some discrepancies have been noted when modeling extreme failure scenarios:

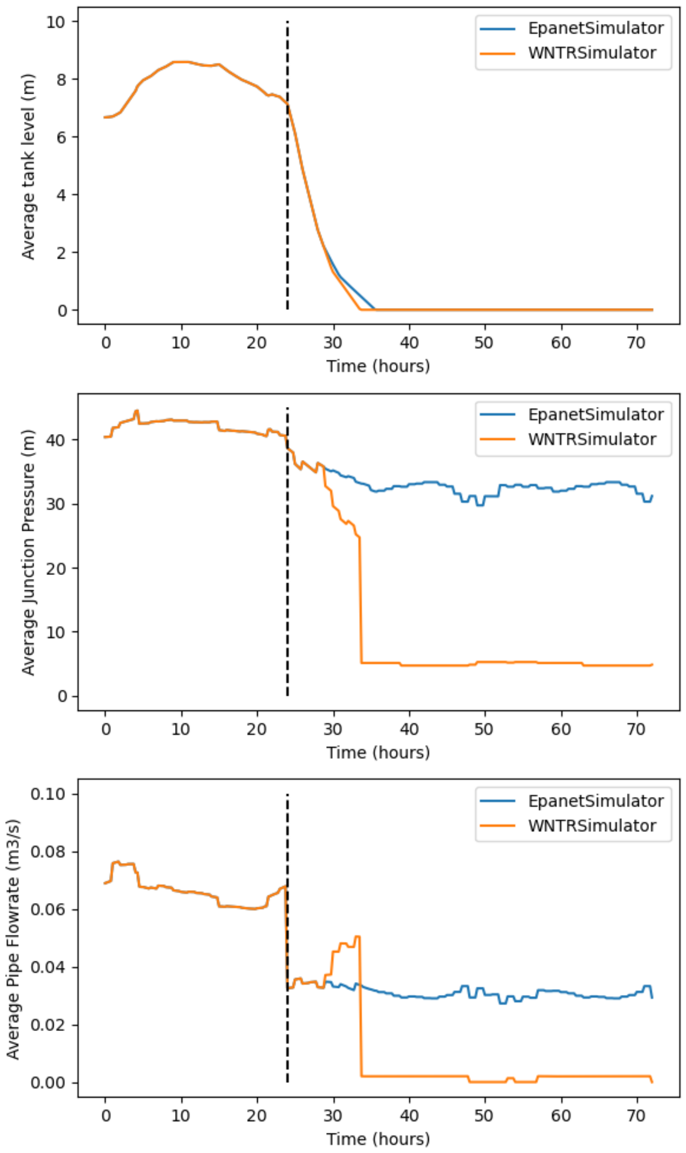
- EpanetSimulator: Cases where tanks continue to supply water to the network even when the system is depressurized, cases where pumps have flow when they are closed, cases where isolated junctions have flow
- WNTRSimulator: Cases where the simulation fails to converge when junctions become isolated

Example discrepancy

In this scenario, water supply is cut off by closing 2 pipes at hour 24. Tanks continue to drain when empty with the EpanetSimulator.

Solution proposed on OpenWaterAnalytics/EPANET





Valve Segmentation

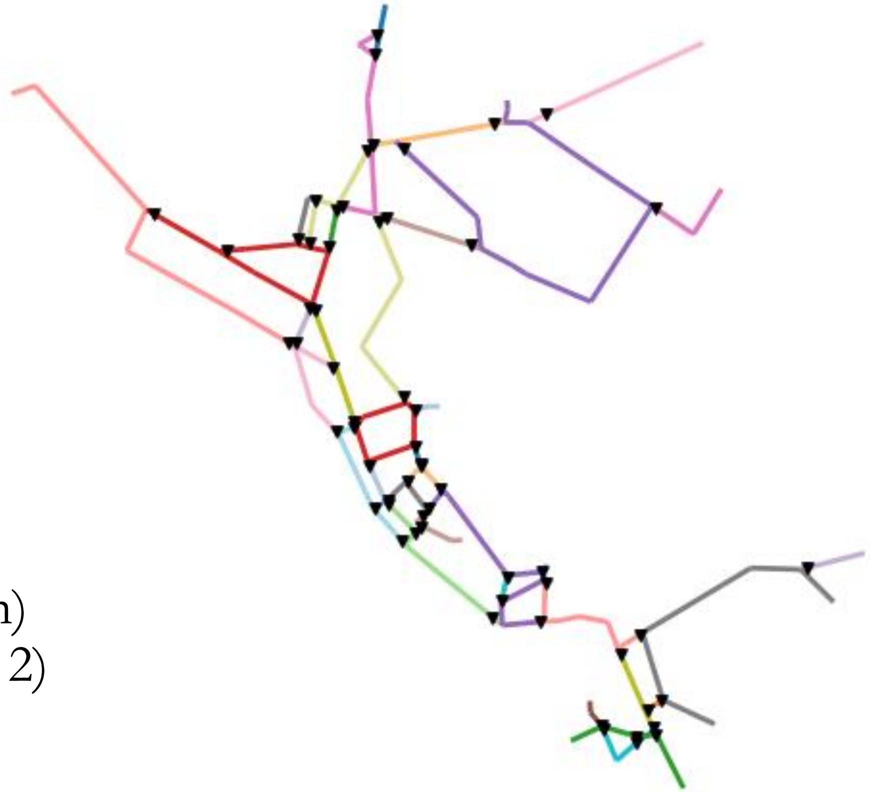
Valve segments groups links and nodes into clusters based on the ability to isolate a region of the network

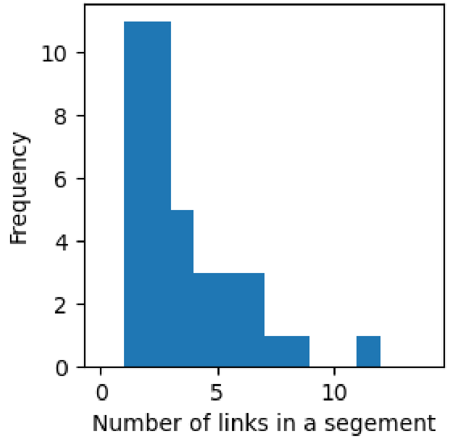
WNTR now includes the ability to create segments from the location of isolation valves

- Actual isolation valve locations
- Generate random placement (i.e., 10 valves)
- Generate strategic placement based on the number of pipes (n) from each node that do NOT include a valve (i.e., n = 0, 1, or 2)

Valve segments can be used in criticality analysis

Graphics options have been updated to visualize segments and valves



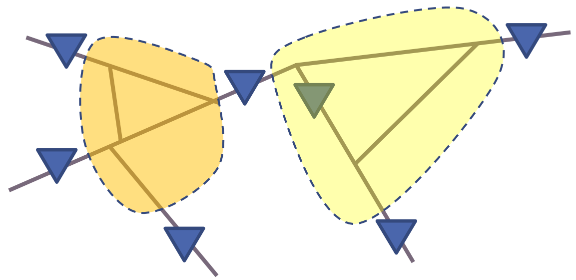


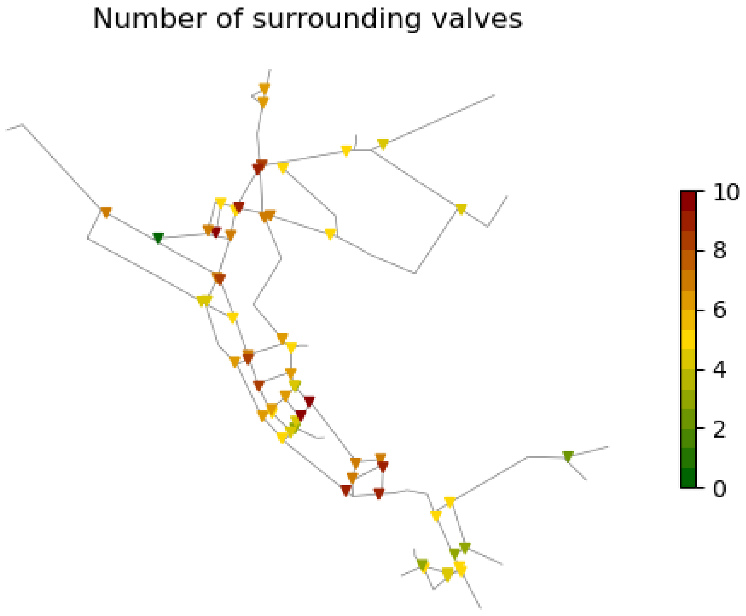

```
>>> import wntr
>>> wn = wntr.network.WaterNetworkModel('Net3.inp')
>>> G = wn.get_graph()
>>> valve_layer = wntr.network.generate_valve_layer(wn, 'strategic', 2)
>>> node_segments, link_segments, segment_size = wntr.metrics.valve_segments(G, valve_layer)
```

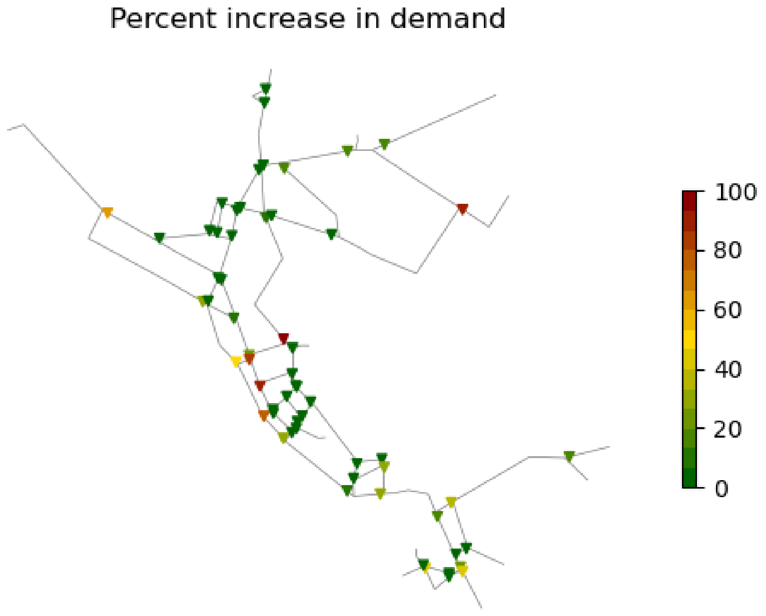
Valve Segmentation Metrics

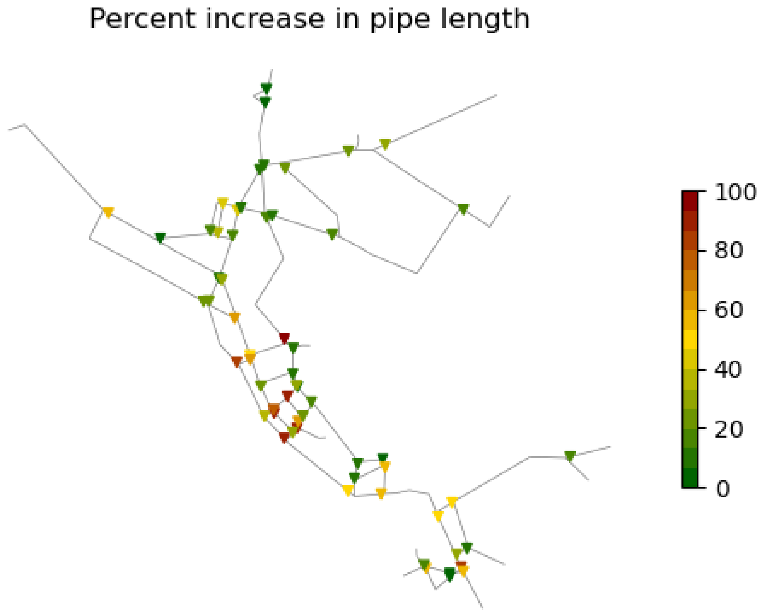
Valve segments can also be used to compute metrics related to valve failure:

- Number of valves necessary to isolate a segment if a given valve is removed or inoperable
- Percent increase in segment demand if a given valve is removed or inoperable
- Percent increase in segment length if a given valve is removed or inoperable









Multipoint Pump and Tank Curves

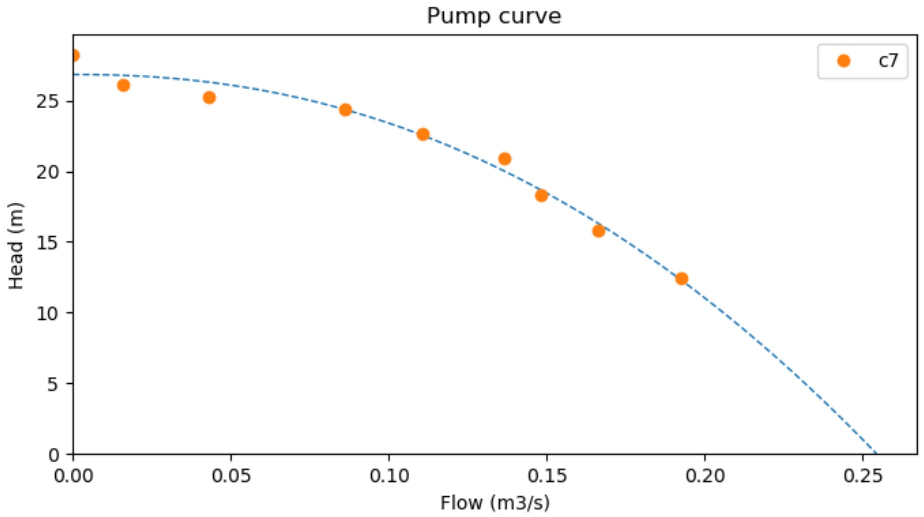
WNTRSimulator now supports multipoint pump curves and multipoint tank curves

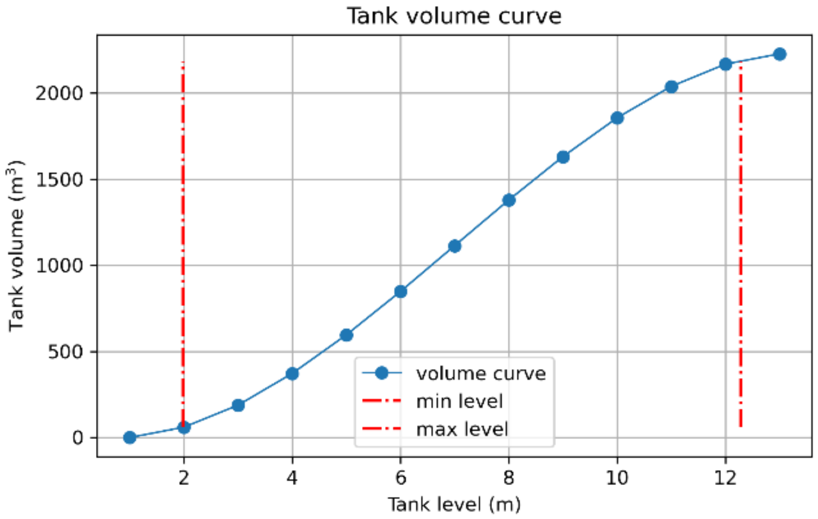
Multipoint pump curves are fit using the same equation used for the 3-point curve

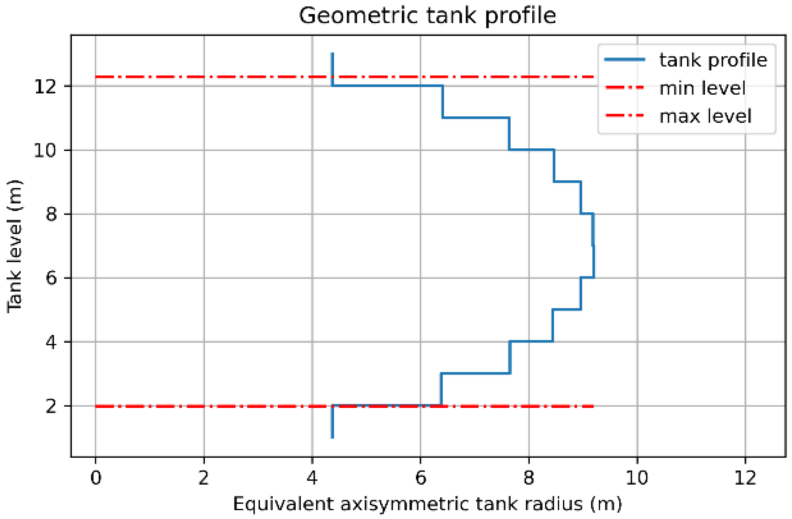
$$H = A - BQ^C$$

Statistically poor fits are rejected

Graphics options have been updated to visualize pump curves and tank curves



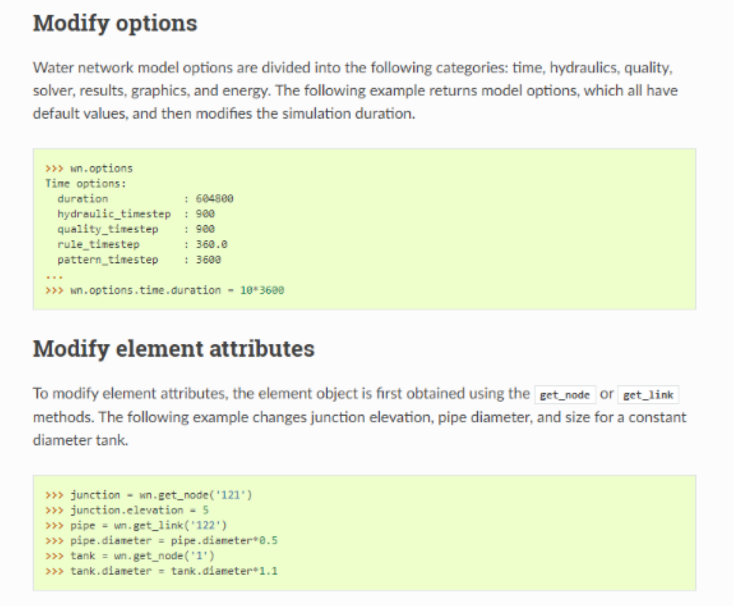




Additional Updates

Code cleanup and documentation

- Model and simulation options revised to better align with EPANET notation
- Better API documentation to clarify element attributes
- More code examples throughout the documentation

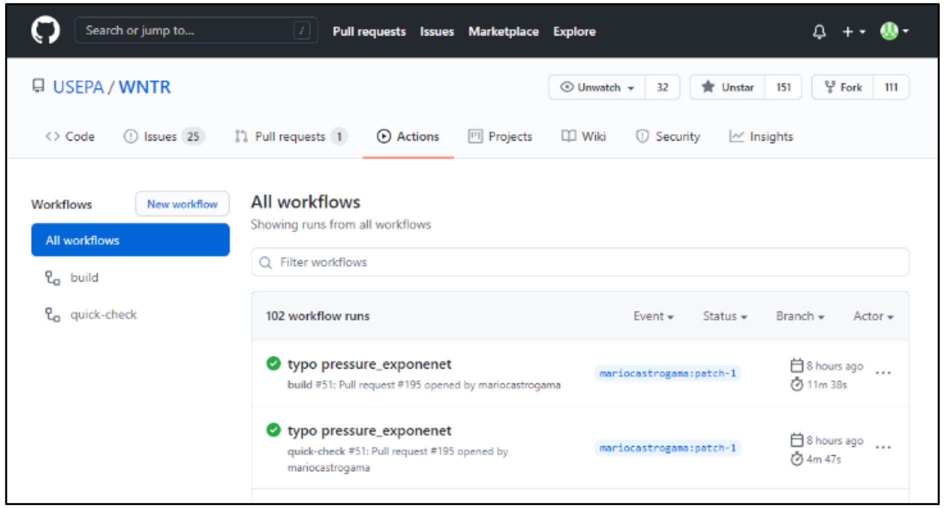


Helper functions

- Convert controls to rules
- Return tank volume at a given level
- Return expected demand for individual demand categories
- Updated method to add power outage controls to pumps

Software tests now use GitHub Actions

- GitHub Action tests are initiated after each push and pull requests to the master and dev branches
- USEPA GitHub site is now used as the host and development repository



Future Development and Ways to Contribute

Features in development

- Initialize the water network model using previous simulation results
- Stop criteria for both simulators
  - For example, stop the simulation if Pump A turns on, or stop the simulation if pressure at Junction X is < 20 psi
- Convert results to EPANET units
- Read/write data from GeoJSON data format
- Export Pyomo model for optimization

Contributing

The development team is always interested in hearing from users to help prioritize features that best support the user community

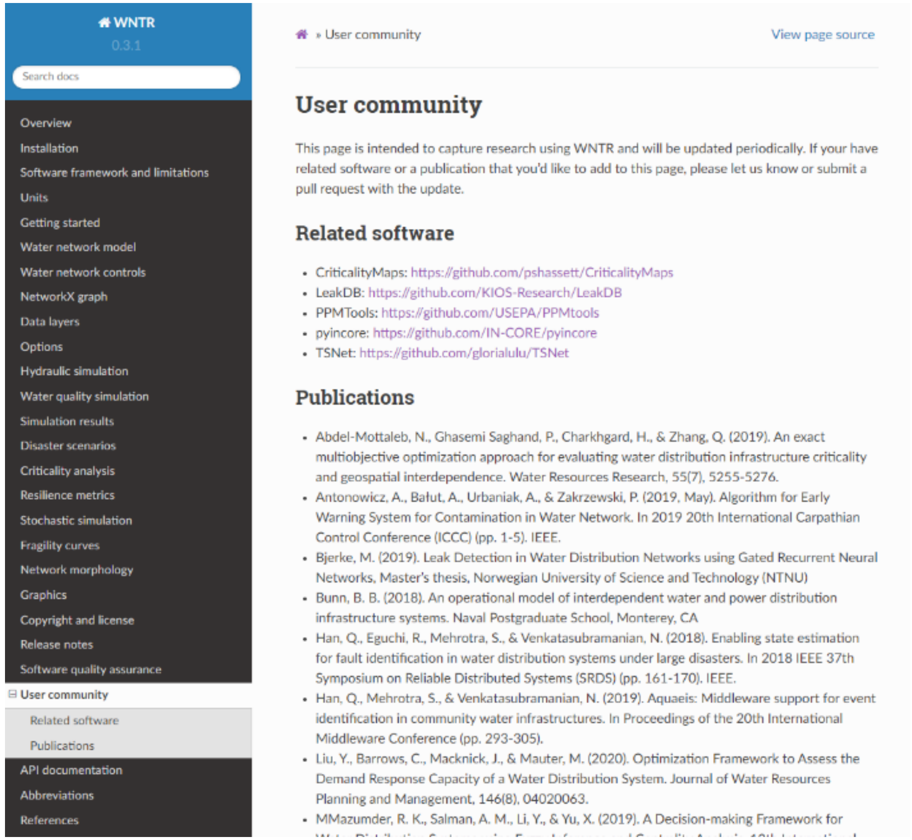
Numerous ways to contribute

- Post issues, feedback, suggestions
- Contribute code, documentation, or tests
- Add related work to the user community page

WNTR

https://github.com/USEPA/WNTR

http://wntr.readthedocs.io



Thank You

Thank you!

Questions and comments to kaklise@sandia.gov

WNTR

https://github.com/USEPA/WNTR
http://wntr.readthedocs.io

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

The U.S. Environmental Protection Agency (EPA) through its Office of Research and Development funded and managed the research described herein under Interagency Agreement (IA \#DW8992403601) with the Department of Energy's Sandia National Laboratories. It has been reviewed by the Agency but does not necessarily reflect the Agency's views. No official endorsement should be inferred. EPA does not endorse the purchase or sale of any commercial products or services.