Center for Infrastructure Modeling and Management

Rolling out SWMM5+ for public use and open-source development

BEN R. HODGES AND SAZZAD SHARIOR

INTERNATIONAL CONFERENCE ON WATER MANAGEMENT MODELING

MARCH 1-2, 2023

Acknowledgements

This presentation was developed under Cooperative Agreement No. 83595001 awarded by the U.S. Environmental Protection Agency to The University of Texas at Austin. It has not been formally reviewed by EPA. The views expressed in this document are solely those of the authors and do not necessarily reflect those of the Agency. EPA does not endorse any products or commercial services mentioned in this publication.

Many thanks to Prof. Jose Vasconcelos for collaboration on the new Preissmann Slot

Some quick history

- NCIMM established by a US EPA grant to Univ. of Texas, the Urban Watersheds Research Institute (UWRI), and other collaborators in November 2016.
- Goals for SWMM project :
    - a finite-volume hydraulic engine
    - Parallel code that scales well
    - SWMM5+ as an “add–on” to the EPA SWMM model
- Untimely death of our co-founder, Charles Rowney, in March 2020.
- Bob Brashear: outreach and develop the non-profit counterpart CIMM.
- Beta (soft) roll-out of the code in early 2022
- It’s been a long road, but we’re almost there...

CIMM

What’s coming up…

- SWMM5+ version 1.0 will be released this spring.
- Code is “Public Domain” --- using the *unlicense* license.
- Users will be able to download source code from GitHub.
- Developers will be able to contribute through GitHub.
- Documentation includes
  - Internal code comments
  - Programmer’s guide with details of algorithms
  - Users guide with installation and implementation

What to expect when downloading SWMM5+

- Download package is a “Docker” container that installs:
  - Source code
  - Compiler (Intel oneAPI)
  - Cmake utility
  - Json library
  - HDF5 library
  - Test cases
  - SWMM5+/EPASWMM comparison scripts
- Docker container can be installed on either Linux or Windows.
- We will not (at this time) distribute executables.
- SWMM5+ couples with ***EPA SWMM v 5.1.0.13***

CIMM

Why Fortran?

- Goals for code:
  - Easy to read and learn
  - Parallelization in a few subroutines.
  - Simple 2D data structures
- Fortran provides some advantages over C:
  - Array processing commands: A = B x C is one statement
  - Array indexes start at 1.
  - Coarray parallelization is in the Fortran standard
  - Users familiar with Matlab, Julia, and Python Numpy will find it easy to transition
- EPA SWMM C code is still used for input and hydrology

Why Intel oneAPI compiler?

- gCC and gFortran were tried
- gFortran does not include Fortran08 Coarrays
- The Intel compiler is free to download...

  ... as long as you do not distribute executables.
- Porting to other compilers is possible.

What’s new and old for the input and output?

- Input with standard EPA SWMM *.inp files.
- Additional controls in a *.json file.
- New output via HDF5 binary files
- Docker installation includes example Python scripts for HDF5 reading.

- SWMM5+ reads US or SI units, but does all computations in SI
- SWMM5+ output (for now) is SI units

- Output provided with averaging to original link/node system or with full finite-volume sub-discretization

- SWMM5+ does not (for now) average over time for output

CIMM

Sub-discretization

- Finite-Volume methods work best with consistent cell sizes
- User provides target sub-discretization length (e.g., 50 m)
- SWMM5+ provides automatic sub-discretization of links.

- SWMM5+ controls the model time step for stability with sub-discretization

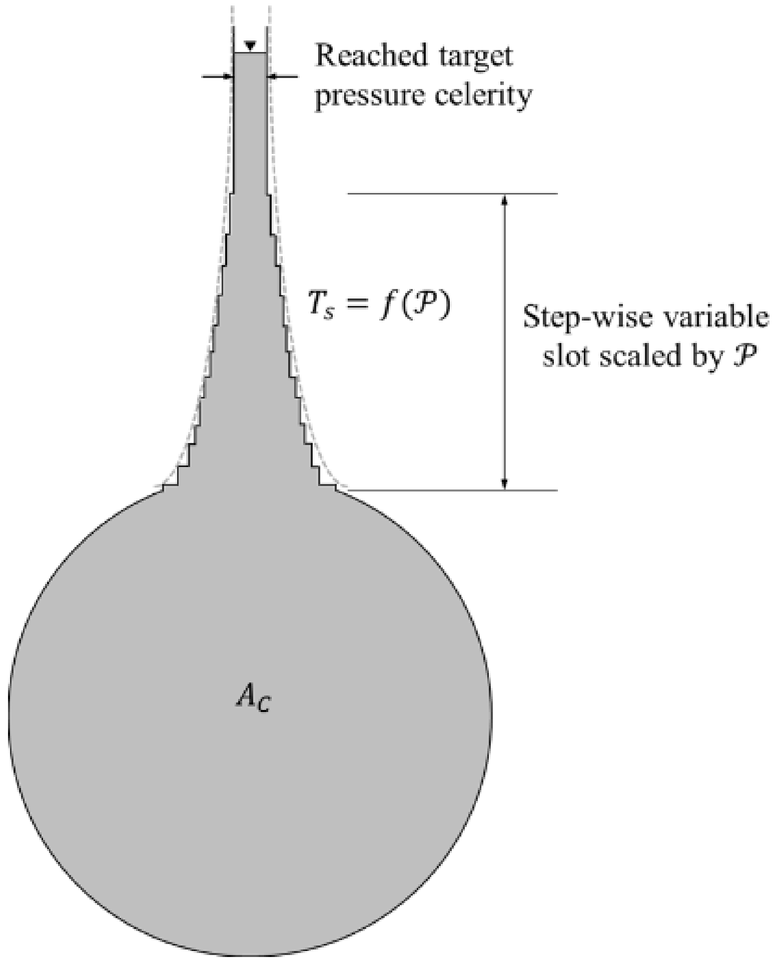
- SWMM5+ breaks-up of system into N pieces for solution on N processors.

What's the new algorithm?

- Full dynamic equations (Saint-Venant) without damping.
- Finite-volume formulation.
- Explicit Runge-Kutta 2nd-order time marching.
- Spatial discretization is combination 1st/2nd order controlled by Froude number.
- Surcharged pipe uses a new non-dimensional form of the Preissmann Slot.
- Mass conservation to machine accuracy.
- Hydraulic jumps occur at faces between cells.

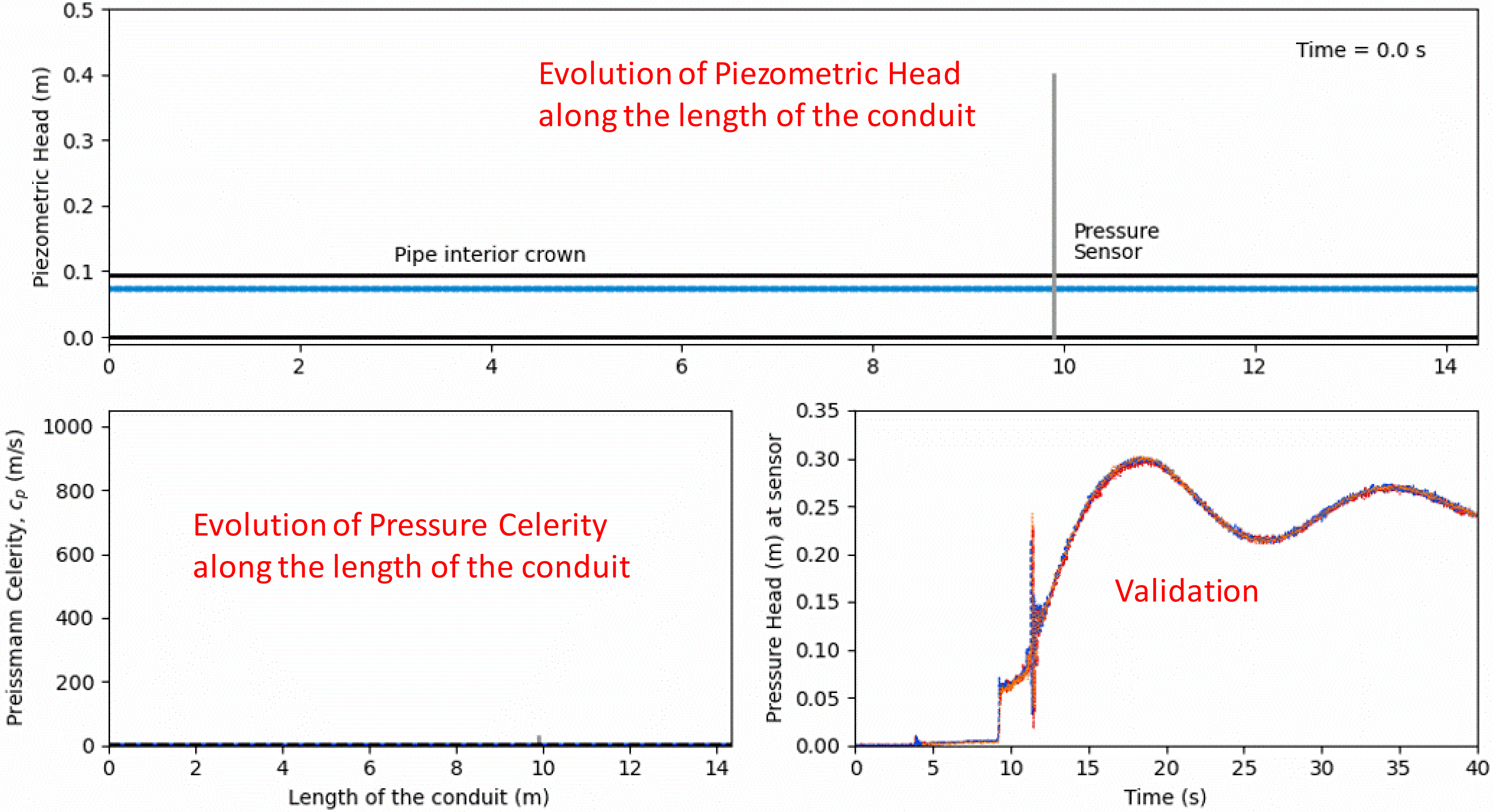
Stepwise dynamic Preissmann Slot

- User selects times-scale for pressure celerity (e.g., 25 m/s) instead of slot width.
- Slot is dynamically created to ensure smooth celerity transitions from free-surface flow.
- Result is reduction in oscillations and shocks at transitions.

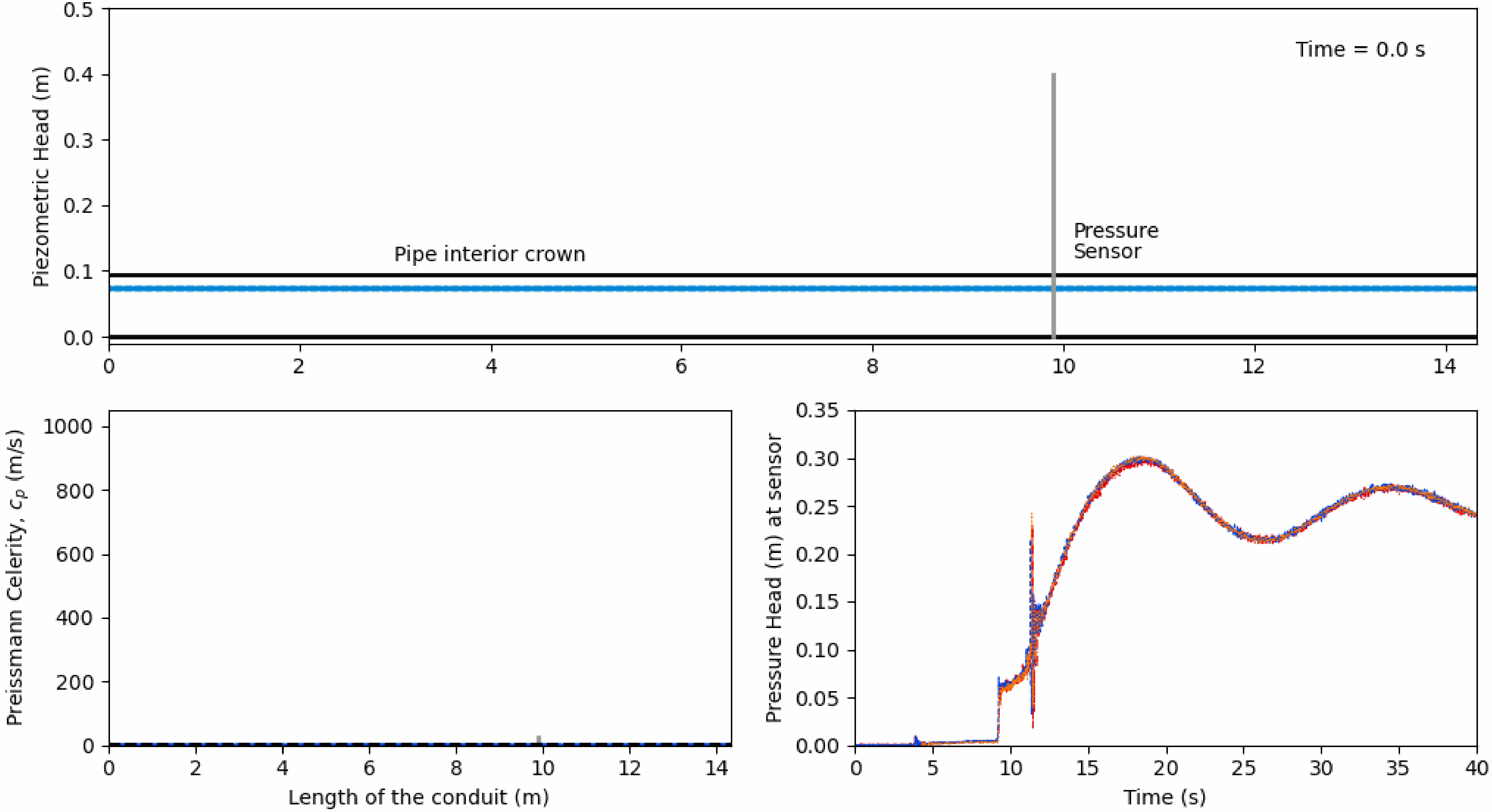


Stepwise dynamic Preissmann Slot

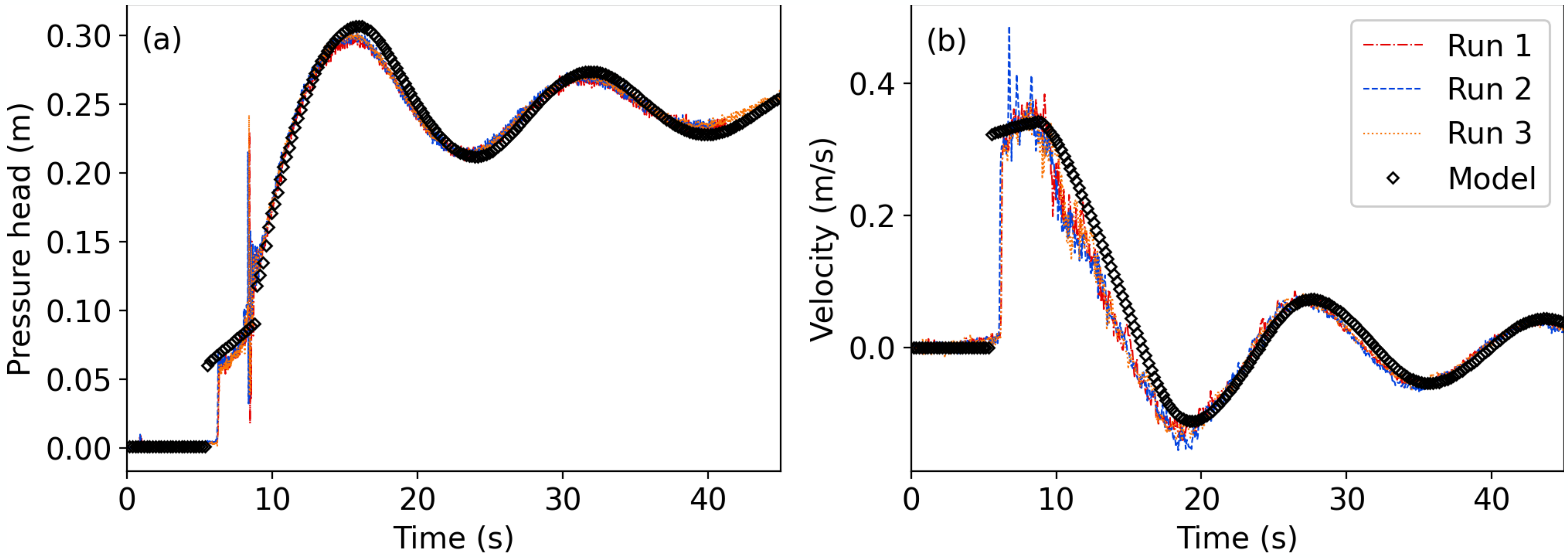
Test case of Vasconcelos et al, 2006



Test case of Vasconcelos et al, 2006

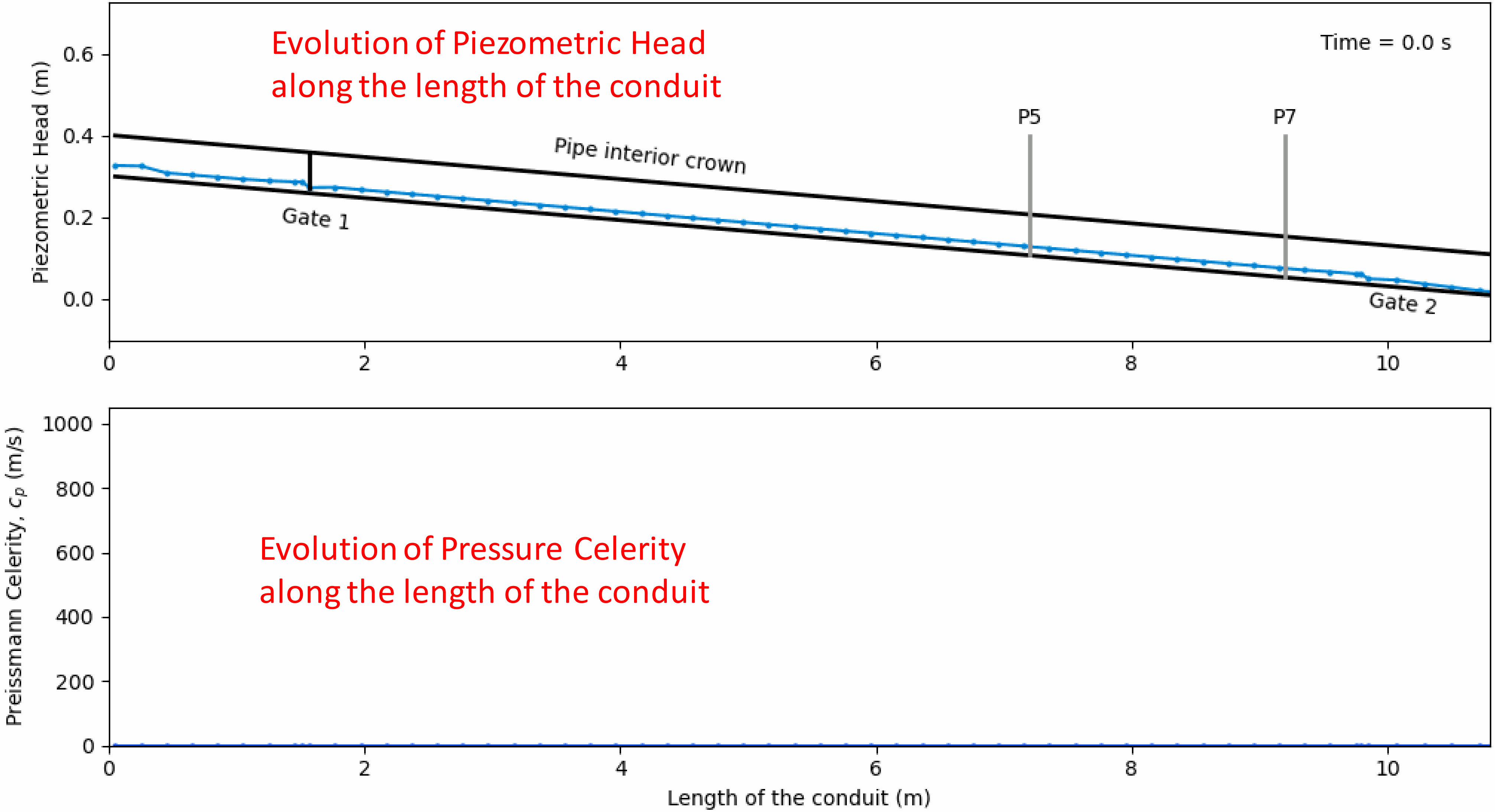


Test case of Vasconcelos et al, 2006

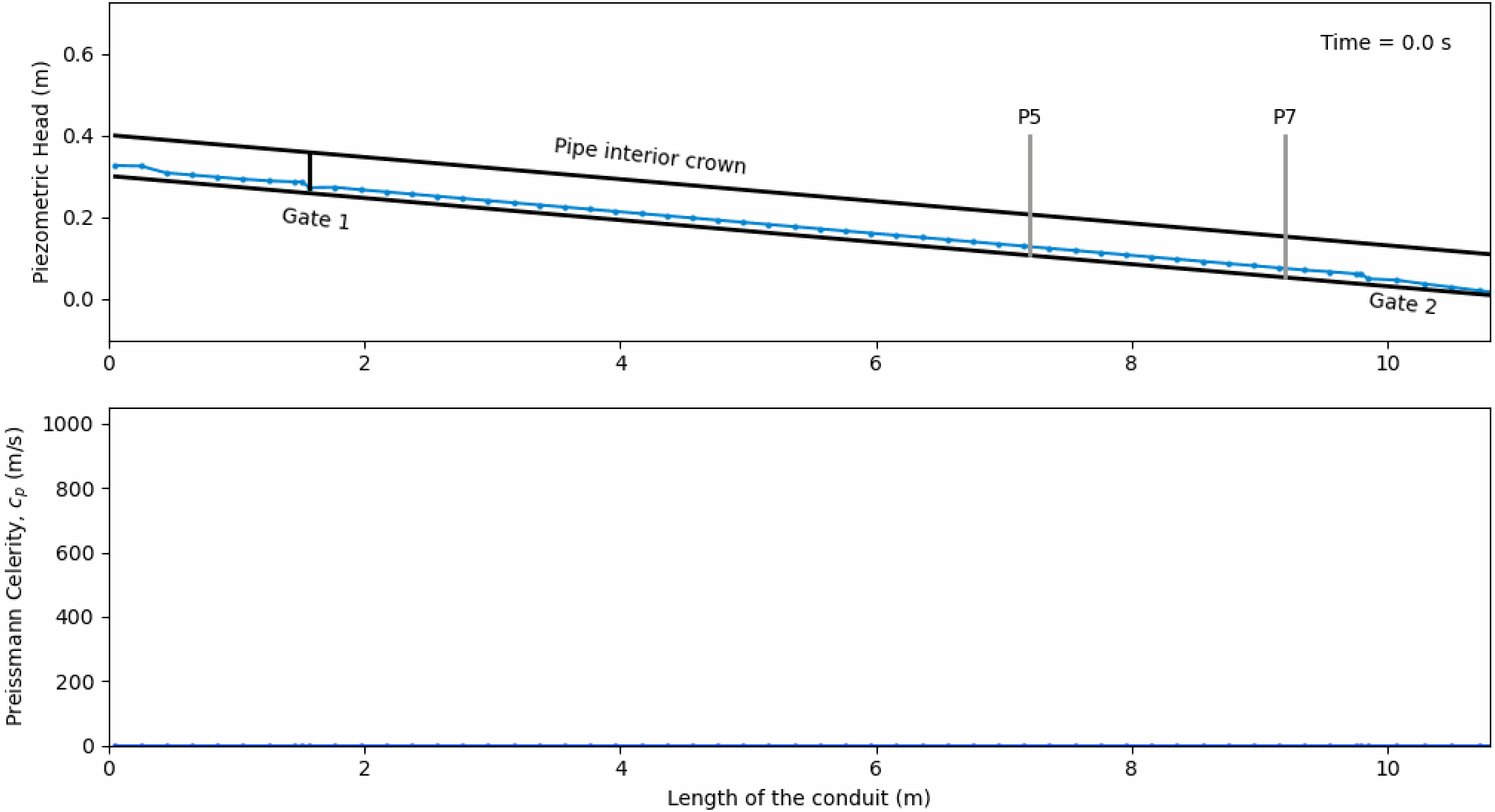


(a) Pressure head and (b) velocity comparing laboratory-observed Run 1, Run 2, Run 3 from Vasconcelos et al. (2006) with new dynamic Preissmann Slot (Model)

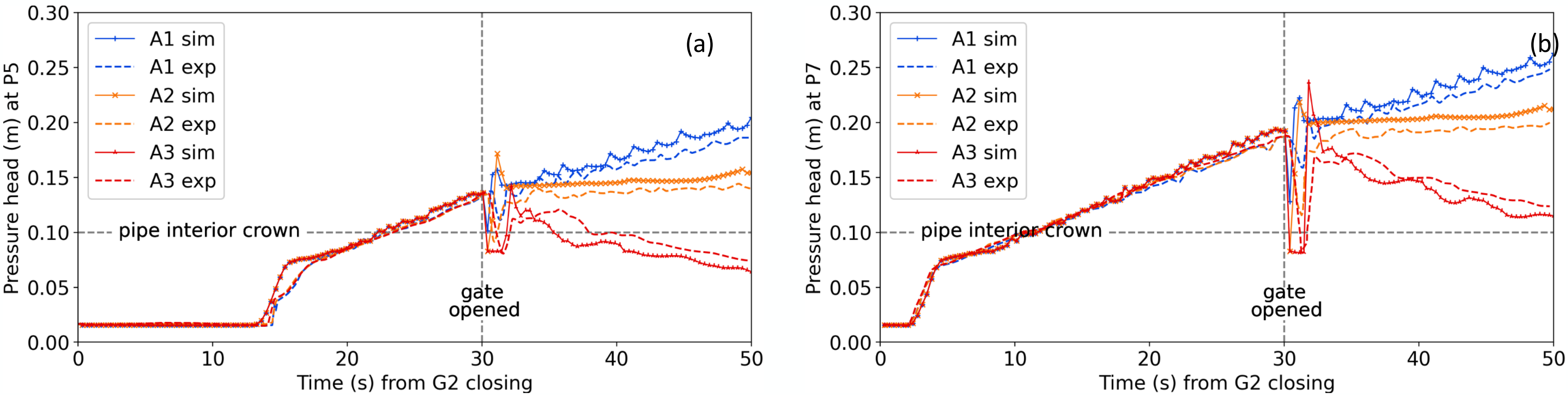
Test case of Trajkovic et al, 1999



Test case of Trajkovic et al, 1999

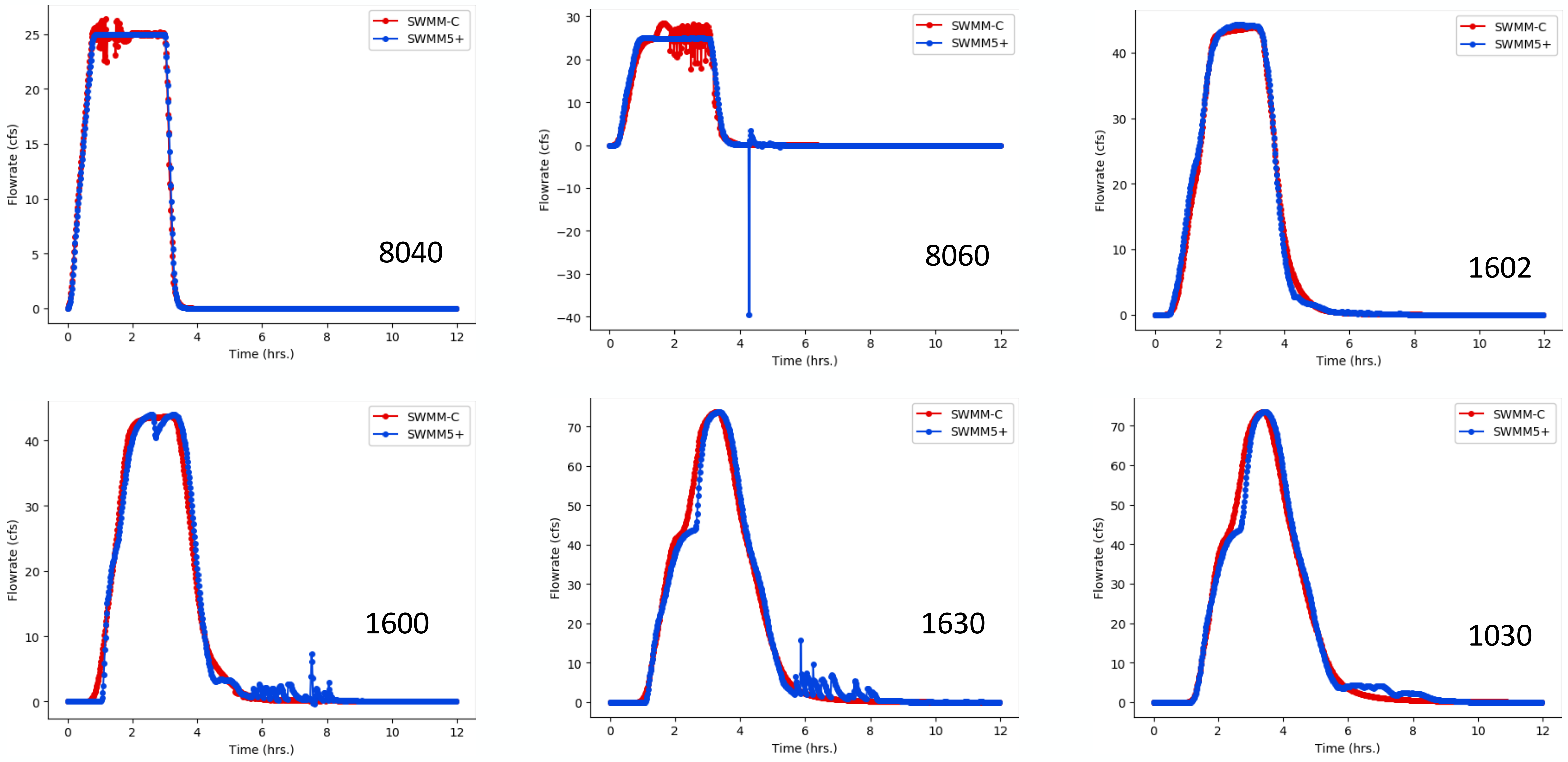


Test case of Trajkovic et al, 1999

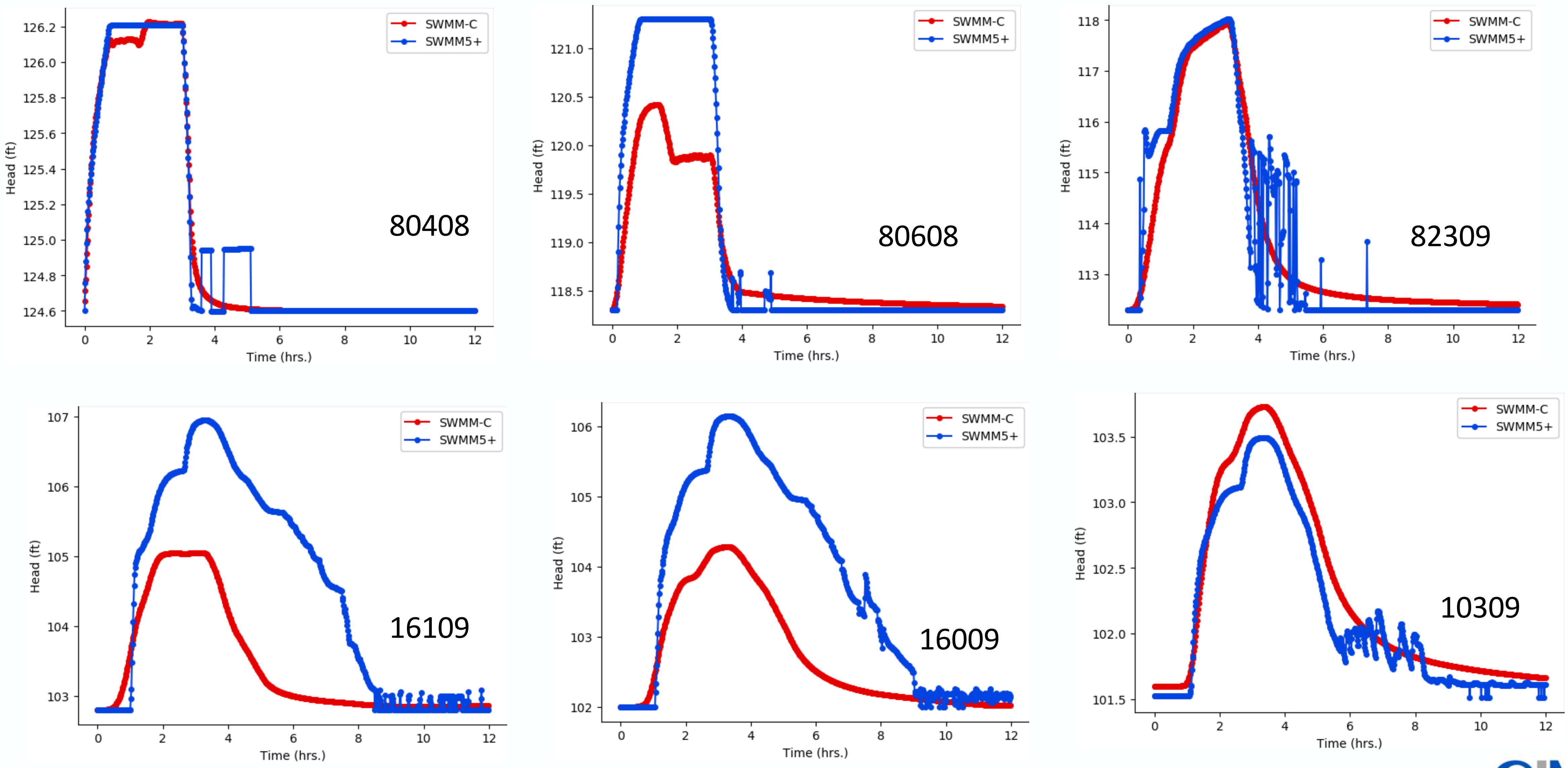


Comparison between DPS simulations and experimentally-observed pressure heads at sensor location (a) P5, and (b) P7 for the cases A1-A3 of Trajkovic et al. (1999)

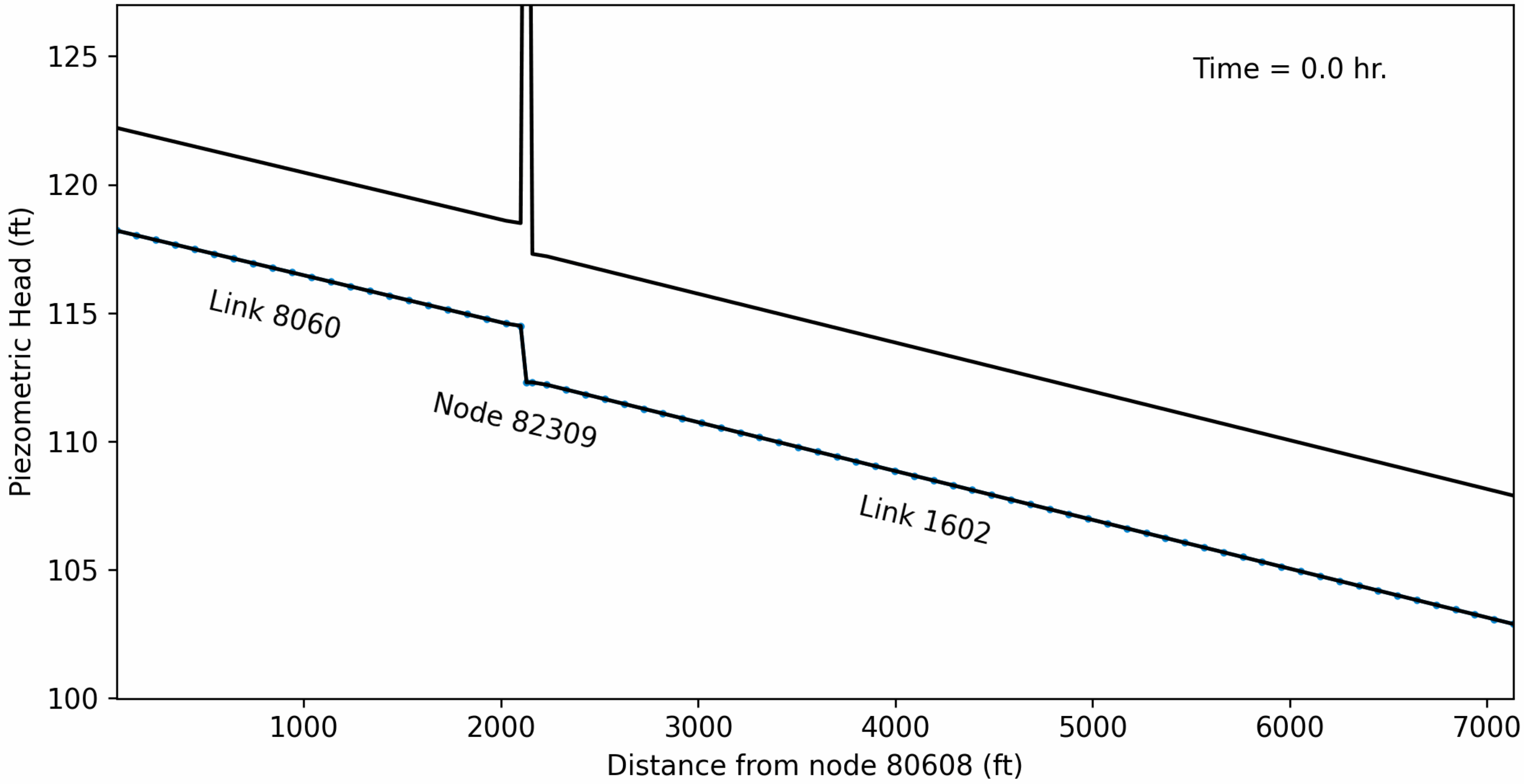
Flowrate comparison with EPA SWMM for extran1.inp



Head comparison with EPA SWMM for extran1.inp



SWMM5+ for extran 1 case at drop node



What will be missing in v 1.0 of SWMM5+

- Pollutant transport
- Restart/hot start capability
- Power function cross-sections
- Kinematic wave solution
- Extran solution method
- Normal flow limiter, inertial damping
- Artificial lengthening based on CFL
- Force mains (implemented, but buggy)
- Additions since EPA SWMM v 5.1.0.13

Summary

- SWMM5+ will be released soon
- Public domain code
- Installation using Docker container
- Requires compilation with Intel oneAPI compiler (installed with Docker)
- For updates and info, contact me at

hodges@utexas.edu