Introducing SWMM5+

Ben R. Hodges

Professor, Maseeh Dept. of Civil, Architectural, and Environmental Engineering, Univ. of Texas at Austin, 301 E. Dean Keeton, Austin, TX 78712 (corresponding author). ORCID: https://orcid.org/0000-0002-2007-1717. Email: hodges@utexas.edu

Sazzad Sharior, S.M.ASCE

Ph.D. Candidate, Maseeh Dept. of Civil, Architectural, and Environmental Engineering, Univ. of Texas at Austin, 301 E. Dean Keeton, Austin, TX 78712. ORCID: https://orcid.org/0000-0001-8544-1902

Edward D. Tiernan, A.M.ASCE

Engineer, Southern California Coastal Water Research Project, 3535 Harbor Blvd., Costa Mesa, CA 92626. ORCID: https://orcid.org/0000-0001-7981-1495

Eric Jenkins

Master's Student, Maseeh Dept. of Civil, Architectural, and Environmental Engineering, Univ. of Texas at Austin, 301 E. Dean Keeton, Austin, TX 78712. ORCID: https://orcid.org/0000-0002-7250-8239

Gerardo Riaño-Briceño, S.M.ASCE

Ph.D. Candidate, Maseeh Dept. of Civil, Architectural, and Environmental Engineering, Univ. of Texas at Austin, 301 E. Dean Keeton, Austin, TX 78712. ORCID: https://orcid.org/0000-0002-2995-1139

Cesar Davila-Hernandez, S.M.ASCE

Ph.D. Candidate, Maseeh Dept. of Civil, Architectural, and Environmental Engineering, Univ. of Texas at Austin, 301 E. Dean Keeton, Austin, TX 78712. ORCID: https://orcid.org/0000-0002-4832-9038

Ehsan Madadi-Kandjani

Assistant Professor, Dept. of Mechanical and Aerospace Engineering, California State Univ., Long Beach, CA 90840. ORCID: https://orcid.org/0000-0003-3216-3245

Cheng-Wei Yu

Assistant Professor, Dept. of Bioenvironmental Systems Engineering, National Taiwan Univ., No. 1, Sec. 4, Roosevelt Rd., Taipei 106319, Taiwan. ORCID: https://orcid.org/0000-0003-2474-1738

Forum papers are thought-provoking opinion pieces or essays founded in fact, sometimes containing speculation, on a civil engineering topic of general interest and relevance to the readership of the journal. The views expressed in this Forum article do not necessarily reflect the views of ASCE or the Editorial Board of the journal.

https://doi.org/10.1061/JOEEDU.EEENG-7680

Introduction

The US Environmental Protection Agency Storm Water Management Model (EPA SWMM) has been the preeminent stormwater model for more than 50 years (CDM Smith 2023; Dickinson 2015; Huber et al. 2010; Rossman et al. 2004). However, the EPA SWMM hydraulic solver sometimes has poor mass conservation and requires a modeler's art to find a fix Furthermore, future faster computers will require efficient parallel computing (Flamm 2017), and the best parallel speed-up for EPA SWMM is about 10× (Burger et al. 2014). Finally, the EPA SWMM water quality module uses an approximate solution to a first-order kinetics transport equation rather than simulation of general advection-diffusion-reaction (ADR) transport (Medina et al. 1981; Rossman and Huber 2016), which limits the model's use for water chemistry with more complex source/sink processes.

The SWMM5+ project began with the observation that the EPA SWMM hydraulic solver (1) is the primary consumer of computational time, (2) is the source of model instability/mass nonconservation, (3) limits parallelization, and (4) cannot support mass-conservative ADR transport modeling. The new SWMM5+ solver provides the foundations for addressing these issues with a mass-conservative hydraulic model that is coded for parallel efficiency. The SWMM5+ code acts as the top-level controlling program with the EPA SWMM code providing a library of functions for hydrology and input data parsing. The overall structure of SWMM5+ and its coupling to EPA SWMM are shown in Fig. 1. The goals of the SWMM5+ project are to

- provide scalable parallelization of SWMM hydraulics,
- create a new hydraulic solver that could be linked to EPA SWMM with minimal impact on the existing code,
- ensure well-controlled mass conservation for both reliable hydraulic simulation and as a basis for ADR water quality transport,
- maintain, as much as practical, consistent input/output data with EPA SWMM, and
- provide a flexible code for wrapping by Python applications or other calling programs.

SWMM5+ is written in Fortran 2008 (instead of C/C++ as used in EPA SWMM v5) and couples with EPA SWMM hydrology through a Fortran/C application programming interface (API). This choice was made after discussion with the broader community—not all of whom agreed. The main reason for selecting Fortran 2008 is that it simplifies writing parallel code for high-performance computing (HPC) and desktop multiprocessor machines. A secondary benefit is that programmers can write array-processing code, i.e., $a = b + c$ operates over all elements of the arrays a, b, and c without requiring do loops over the array indexes, which allows cleaner code that is easier to follow and debug. Array processing is also available in Matlab, Python, Julia, and R, but not in C/C++. An important change in SWMM5+ is that the Fortran data arrays and procedures are not posed as object-oriented types. This choice makes the SWMM5+ code more accessible to engineering programmers and for code wrapping, albeit being less amenable to flexible modularization of component algorithms.

This article provides an overview of the SWMM5+ project, including (1) parallelization, (2) explicit finite-volume method

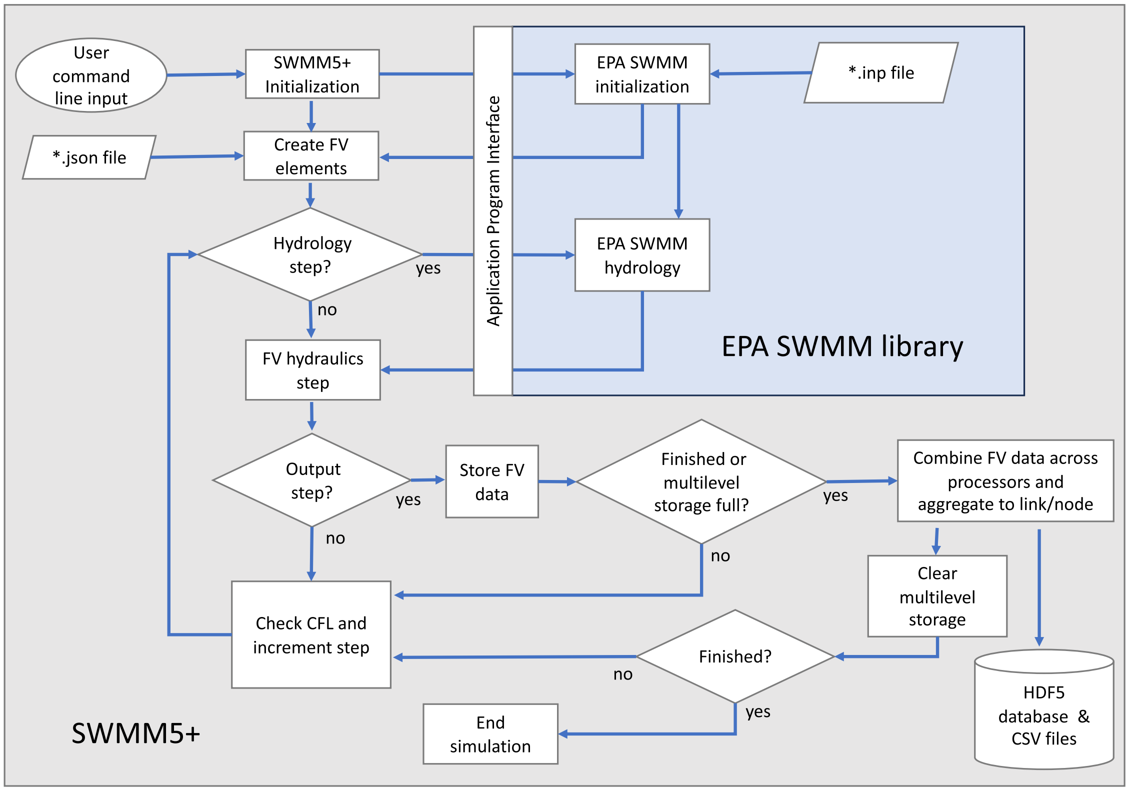

Fig. 1. Outline of the code coupling between SWMM5+ and EPA SWMM.

(3) mass conservation and stability, (4) input/output data, (5) advantages/disadvantages, (6) accessing SWMM5+, and (7) our thoughts for the future.

Parallelization in SWMM5+

SWMM5+ uses the single program, multiple data (SPMD) parallelization method. Each processor receives exactly the same program but executes it over only a portion of the overall data; i.e., the full stormwater network is broken up into smaller subnetworks assigned to individual processors. Efficient subdivision of the network is done automatically as part of initialization of the a SWMM5+ simulation (Tiernan and Hodges 2022). Although SPMD parallelism can be implemented in almost any modern program language—typically using message passing interface (MPI) or open multiprocessing (Open-MP) libraries—Fortran has a distinct advantage in that SPMD is built into the Fortran 2008 coding standard for coarrays, which provide a simple approach for parallelization of subnetworks. Our experience is that Coarray Fortran is easier to understand, write, and debug than either MPI or Open-MP codes, in part because coarrays are integral to the language and in part because the coarrays are (intentionally) less flexible and force the programmer into using data structures that are easy to understand.

SWMM5+ parallelization was designed such that the total number of connections between processors is roughly proportional to the number of processors applied, which is possible because stormwater systems typically only have three or four connections per junction. Using SPMD, a SWMM5+ network simulation should be parallel-efficient for a large number of processors. This large processor limit depends on (1) sufficient computational elements on each processor, and (2) Amdahl's law (Amdahl 1967). Firstly, a large number of processors requires a large system; i.e., there must be enough computational elements on each processor for compute time to dominate communication time between processors. Secondly, Amdahl's law limits achievable parallel speed-up when a code contains a mixture of serial and parallel algorithms. The parallel SWMM5+ hydraulics couples with serial EPA SWMM hydrology, which therefore limits the parallel efficiency. We are presently investigating the achievable parallel speed-up of SWMM5+ for large systems.

Explicit Finite-Volume Hydraulic Solver

The SWMM5+ hydraulic solver uses an explicit finite-volume (FV) algorithm with a Runge-Kutta second-order (RK2) time-advance for the Saint-Venant equations along a subdivided link. Manholes and storage nodes are computed using a continuity equation coupled to an energy equation for downstream fluxes. Each link of a standard SWMM input file is automatically subdivided into multiple FV elements using a global scale set by the user (e.g., 10 m) as part of the SWMM5+ initialization. A minimum of three elements per link is set by the model. The consistent spatial discretization of the network allows the model to represent spatial gradients and progressive motion of trans-critical hydraulic features (e.g., hydraulic jumps, bores, and critical flow controls).

The new FV scheme uses a time-interpolation method to transfer cell-centered data to the faces between cells; the advantage of this approach is that it correctly handles the transition from subcritical to supercritical flow without requiring special treatment (Hodges and Liu 2019, 2020). The low-order (RK2) time-marching approach is used instead of a Runge-Kutta fourth-order (RK4) because spatial rather than temporal discretization typically controls overall error when only a few elements represent each link. Furthermore, the four-step RK4 requires four parallel communication events per time step compared with the two events per time step required by a two-step RK2, so the RK2 has a smaller parallel communications burden. Surcharged pipe flows in SWMM5+ are computed using a new Preissmann slot algorithm that is more robust than prior approaches (Sharior et al. 2023a). All the hydraulic features of EPA SWMM are supported in SWMM5+, including pumps, orifices, weirs, culverts, and controls.

Mass Conservation and Stability

The FV method of SWMM5+ provides mass conservation to numerical precision. The 64-bit CPUs on desktop workstations and most HPC machines have 16-digit precision, so FV error scales on 10^{-16} per time step, which allows simulation of tens of millions of time steps while maintaining cumulative error less than 10^{-8}. However, the coupling between the FV time-marching scheme and equations for various hydraulic features (e.g., storage, pumps, weirs, and culverts) creates locations where mass nonconservation can be accidentally introduced in the code. Thus, in SWMM5+, a mass conservation failure is indicative of a code error that needs to be fixed by the development team rather than a user configuration error that requires art to overcome.

SWMM5+ is linearly stable as long as the local Courant-Friedrichs-Lewy (CFL) number is everywhere constrained by $\mathrm{CFL} < \sqrt{2}/2$. The time step in SWMM5+ is automatically controlled to ensure stability throughout the network. Junctions in SWMM5+ are often the principal limitation of the time step as a small manhole diameter is effectively a small length scale in the CFL computation. As a result, the SWMM5+ time step is often only 10% of the EPA SWMM time step for the same system. We are investigating a locally implicit junction algorithm that will allow larger time steps while retaining stability.

Input/Output Data

SWMM5+ maintains backward compatibility by using the same *.*inp* file as EPA SWMM for basic controls, defining the hydraulic network, and hydrological forcing. SWMM5+ also has a range of advanced controls associated with the new hydraulics solver that are set in a text file using the JavaScript Object Notation (JSON) format (Bray 2017). Output from SWMM5+ is not identical with that from EPA SWMM due to issues associated with the SWMM5+ subdivision of links into FV elements and parallel partitioning into subnetworks. SWMM5+ provides new output functions using the Hierarchical Data Format version 5 (HDF5) library (e.g., Folk et al. 2011). The HDF5 approach provides flexible data storage and retrieval that is well documented with metadata. The SWMM5+ HDF5 file allows users to access either the full FV data or link-node data equivalent to EPA SWMM.

Advantages/Disadvantages of SWMM5+

The key advantages of SWMM5+ are as follows: (1) robust mass-conservative hydraulics simulation, and (2) direct modeling of hydraulic jumps, bores, and rapid surcharge transitions. Our observations are that SWMM5+ has fewer spurious oscillations than EPA SWMM for similar conditions. Furthermore, the new mass-conservative hydraulics are a necessary precursor to true ADR transport simulation (which has not yet been developed for SWMM5+). Some of the capabilities of SWMM5+ and comparisons with EPA SWMM have been demonstrated in a technical report by Hodges and Sharior (2023).

However, SWMM5+ incurs increased costs in the number of computations, computer memory, and disk space; all these increases are driven by the need to subdivide links for the FV algorithm. Although our investigations are not yet complete, it appears that SWMM5+ will always be slower than EPA SWMM unless the problem is large enough to use 32+ parallel processors. We estimate efficient parallelization will use at least 5,000 computational elements per processor, so the minimum network size for SWMM5+ to be faster than EPA SWMM may be as large as 160,000 FV elements. Depending on the relationships between link lengths and the users's target (nominal) element length, 160,000 FV elements might be a stormwater system of 20,000 to 50,000 links (i.e., full city scale).

For smaller problems—necessarily using a smaller number of parallel processors—SWMM5+ will be noticeably slower than EPA SWMM. This slowness is due to the extra computations required for smaller FV elements. Of course, the slower speed of SWMM5+ is only a disadvantage compared with EPA SWMM if the latter can reliably produce a nonoscillatory, mass-conservative solution at the faster speed. Complex EPA SWMM models often have spatiotemporally localized problems that are difficult to debug and require multiple simulations and modeler art to resolve. In such cases, a single simulation of SWMM5+ may be more computationally efficient despite its longer run time.

Accessing SWMM5+

The SWMM5+ code has been released as public-domain software that is free for use without any restrictions. The source code is available on GitHub (CIMM 2023) and includes an installation guide (Davila-Hernandez et al. 2023), a usermanual (Sharior et al. 2023b), and a draft programmer's guide (Hodges et al. 2023). The present code can be installed directly on any Linux system or through a Docker container for Linux/Windows (direct installation on Linux is recommended). To date, we have not ported the installation to any MacOS computers. Compiling has been tested with the Intel oneAPI compiler, which is a free cross-platform, multilanguage compiler (Intel 2023). A Windows OS executable download is in development for users who do not want to install and compile the source code. SWMM5+ has not been integrated with the EPA SWMM GUI and can only be run at the command line. The installation package includes Python scripts that access and plot the HDF5 output data. However, users are cautioned that the output processing from SWMM5+ is rudimentary at present.

The Future of SWMM5+

The SWMM5+ project has built the foundations of a comprehensive, mass-conservative, highly-parallel stormwater network simulation that integrates with EPA SWMM hydrology. Users should keep in mind that SWMM5+ is a research code using a completely new set of hydraulic algorithms. There are likely hidden bugs at the interface between FV hydraulics and hydraulic components (e.g., culverts, weirs, pumps, and controls) that might only appear under complex flow conditions. Users are encouraged to communicate bugs and code errors through the Center for Infrastructure Modeling and Management (CIMM) GitHub site (CIMM 2023). There remains much work to do in testing, debugging, and improving the resiliency of the code. The new parallel FV method brings new possibilities for running large SWMM models on HPC cloud computers with hundreds of processors.

The low-hanging fruit for improving SWMM5+ is in developing a new FV algorithm for ADR water quality transport.

The mass-conservative hydraulic solution of SWMM5+ is a necessary precondition for ADR transport, but the CIMM team did not have sufficient time to write/test a new set of ADR transport algorithms for water quality modeling. We are presently improving the programmer's guide that will help new developers get started with SWMM5+; anyone interested in working on the SWMM5+ project should contact this paper's lead author. Although research using SWMM5+ is continuing under a US National Science Foundation project, CIMM does not presently have general code development funds and is looking to contributions from the broader community for testing and debugging support. It is the hope of the CIMM and the SWMM5+ team that the broader SWMM community will find this model useful and contribute to its long-term success.

Data Availability Statement

All data, models, and code generated or used during the study are available in a repository online in accordance with funder data retention policies. See (Hodges et al. 2023; CIMM 2023; Hodges and Sharior 2023).

Acknowledgments

This material is based upon work partially supported by the National Science Foundation under Grant No. 2049025 and was developed using data/models created under Cooperative Agreement No. 83595001 awarded by the US Environmental Protection Agency to the University of Texas at Austin. It has not been formally reviewed by EPA or NSF. The views expressed in this document are solely those of the authors and do not necessarily reflect those of the Agency or the Foundation. EPA and NSF do not endorse any products or commercial services mentioned in this publication.

References

Amdahl, G. M. 1967. "Validity of the single processor approach to achieve large scale computing capabilities." In *AFIPS'67 (Spring): Proc., AFIPS Spring Joint Computer Conf.*, 483–485. New York: Association for Computing Machinery. https://doi.org/10.1145/1465482.1465560.

Bray, T. 2017. "The JavaScript Object Notation (JSON) data interchange format." Accessed June 21, 2024. https://www.rfc-editor.org/info/std90.

Burger, G., R. Sitzenfrei, M. Kleidorfer, and W. Rauch. 2014. "Parallel flow routing in SWMM 5." *Environ. Model. Software* 53 (Mar): 27–34. https://doi.org/10.1016/j.envsoft.2013.11.002.

CDM Smith. 2023. "How SWMM changed the way cities view water." Accessed June 21, 2024. https://www.cdmsmith.com/en/Client-Solutions/Insights/EPA-SWMM-Storm-Water-Management-Model.

CIMM. 2023. "SWMM5plus." Accessed June 21, 2024. https://github.com/CIMM-ORG/SWMM5plus.

Davila-Hernandez, C., B. R. Hodges, S. Sharior, and E. Jenkins. 2023. *SWMM5+ installation manual version 1.0*. Technical Report, Texas Data Repository V1. Austin, TX: Texas Digital Library. https://doi.org/10.18738/T8/U8IYZZ.

Dickinson, R. E. 2015. "History of #SWMM or the EPA Stormwater Management Model Versions 1,2,3,4,5,6 to the Year 2017." Accessed June 21, 2024. https://medium.com/@rdickinson/history-of-swmm-1-2-3-4-5-to-the-year-2015-bab9d21c8a09.

Flamm, K. 2017. "Has Moore's Law been repealed? An economist's perspective." *Comput. Sci. Eng.* 19 (2): 29–40. https://doi.org/10.1109/MCSE.2017.30.

Folk, M., G. Heber, Q. Koziol, E. Pourmal, and D. Robinson. 2011. "An overview of the HDF5 technology suite and its applications." In *AD'11: Proc., EDBT/IDCT 2011 Workshop on Array Databases*, edited by P. Baumann, B. Howe, K. Orsborn, and S. Stefanova, 36–47. New York: Association for Computing Machinery. https://doi.org/10.1145/1966895.1966900.

Hodges, B. R., and F. Liu. 2019. *Algorithms in the SvePy solver for the unsteady Saint-Venant equations*. Technical Report, Texas Data Repository V1. Austin, TX: Texas Digital Library. https://doi.org/10.18738/T8/ETSJDJ.

Hodges, B. R., and F. Liu. 2020. "Timescale interpolation and no-neighbor discretization for a 1D finite-volume Saint-Venant solver." *J. Hydraul. Res.* 58 (5): 738–754. https://doi.org/10.1080/00221686.2019.1671510.

Hodges, B. R., and S. Sharior. 2023. "*SWMM5+ example simulations and comparisons*. Technical Report, Texas Data Repository V1. Austin, TX: Texas Digital Library. https://doi.org/10.18738/T8/VHLNTR.

Hodges, B. R., S. Sharior, E. Jenkins, E. D. Tiernan, and G. Riaño-Briceño. 2023. *SWMM5+ programmer manual version 1.0*. Technical Report, Texas Data Repository V1. Austin, TX: Texas Digital Library. https://doi.org/10.18738/T8/ZA3VR8.

Huber, W. C., L. A. Rossman, and R. E. Dickinson. 2010. "EPA storm water management model, SWMM5." Chap. 14 in *Watershed models*, edited by V. P. Singh and D. K. Frevert, 339–360. Boca Raton, FL: CRC Press.

Intel. 2023. "oneAPI: A new era of heterogeneous computing." Accessed June 21, 2024. https://www.intel.com/content/www/us/en/developer/tools/oneapi/overview.html.

Medina, M. A., Jr., W. C. Huber, and J. P. Heaney. 1981. "Modeling stormwater storage/treatment transients: Theory." *J. Environ. Eng. Div.* 107 (4): 781–797. https://doi.org/10.1061/JEEGAV.0001199.

Rossman, L. A., R. Dickinson, T. Schade, C. Chan, E. H. Burgess, D. Sullivan, and F.-H. Lai. 2004. "SWMM 5—The next generation of EPA's storm water management model." *J. Water Manage. Model.* 12 (Feb): 339–358. https://doi.org/10.14796/JWMM.R220-16.

Rossman, L. A., and W. C. Huber. 2016. *Storm water management model reference manual, volume iii—Water quality*. Rep. No. EPA/600/R-16/093. Cincinnati: USEPA, National Risk Management Laboratory, Office of Research and Development.

Sharior, S., B. R. Hodges, and J. G. Vasconcelos. 2023a. "The generalized, dynamic, transient-storage form of the Preissmann slot." *J. Hydraul. Eng.* 149 (11): 04023046. https://doi.org/10.1061/JHEND8.HYENG-13609.

Sharior, S., B. R. Hodges, C.-W. Yu, E. D. Tiernan, E. Jenkins, and G. Riaño-Briceño. 2023b. *SWMM5+ User manual version 1.0*. Austin, TX: Texas Digital Library. https://doi.org/10.18738/T8/OWQOZ3.

Tiernan, E. D., and B. R. Hodges. 2022. "A topological approach to partitioning flow networks for parallel simulation." *J. Comput. Civ. Eng.* 36 (4): 1–13. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001020.