


SynxFlow: A GPU-accelerated Python package for multi-hazard simulations

Xilin Xia ¹¶ and Xiaodong Ming ²

1 School of Engineering, University of Birmingham, Birmingham, UK 2 Ping An P&C Insurance Company of China, Shenzhen, China ¶ Corresponding author

DOI: [10.21105/joss.07586](https://doi.org/10.21105/joss.07586)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Anjali Sandip](#) 

Reviewers:

- [@christopheredsall](#)
- [@PabloIbanez](#)
- [@thivinanandh](#)

Submitted: 28 October 2024

Published: 25 June 2025

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

SynxFlow can dynamically simulate flood inundation, landslides runout and debris flows using multiple CUDA-enabled GPUs. It also offers a Python interface for users to set up simulations and visualise outputs. The software is interoperable with popular packages such as numpy, pandas and rasterio. Therefore, SynxFlow can be integrated into data science workflows to streamline and accelerate hazard risk assessment tasks.

Statement of need

Natural hazards like floods, landslides, and debris flows cause significant damage each year, and due to climate change, the frequency and severity of these events are increasing. This rising threat highlights an urgent need to develop tools that can enhance our understanding of these risks, supporting crucial applications like early warning systems and comprehensive risk assessments.

Effectively assessing these risks requires integrating data from multiple sources—such as climate, geological, hydrological, and environmental data, while also accounting for complex, interacting hazards. For comprehensive risk assessment over a large region, processing vast amounts of data swiftly is essential, often necessitating cloud-based computing or the use of high-performance computing (HPC) systems. These HPC systems, especially when enhanced with Graphics Processing Units (GPUs), allow faster processing of complex simulations.

However, current modeling tools show significant limitations in meeting the growing demands of multi-hazard risk assessment and in leveraging modern computing technologies. For example, tools like LISFLOOD-FP 8.0 ([Shaw et al., 2021](#)) and TRITON ([Morales-Hernández et al., 2021](#)) offer GPU acceleration, but their scope is restricted to flood simulations and they lack native Python interfaces — making integration with data science workflows cumbersome. GeoClaw ([Berger & LeVeque, 1998](#); [Clawpack Development Team, 2024](#)) supports multiple hazards and is Python-compatible, yet its single-GPU acceleration makes it inefficient for large-scale, high-resolution applications.

HEC-RAS ([US Army Corps of Engineers, 2025](#)) is widely used due to its user-friendly graphical interface and support for third-party Python automation. However, it does not support multi-GPU acceleration, cannot simulate landslide runout, and remains proprietary software — limiting both customisation and open development. These constraints highlight a critical gap: existing tools are either too specialised, insufficiently scalable, or poorly integrated with modern data and compute ecosystems.

To this end, we believe modern multi-hazard simulation software should be

1. Performant – The software should utilise modern GPU-based HPC systems to accelerate

complex simulations, enabling faster data processing and real-time hazard prediction. This is crucial for rapid risk assessment over large regions with multiple hazards.

2. Accessible – The software should be open-source and easy-to-deploy, allowing transparency and collaboration. Accessible deployment options also lower infrastructure barriers, enabling use in diverse settings.
3. Interoperable – The software should integrate seamlessly with popular data science tools to leverage advanced analytics and machine learning. It should also enable easy coupling of various hazard simulations. This is crucial for comprehensive and effective risk analytics.

The SynxFlow project is our attempt to achieve this vision.

Design and Implementation

The software is a Python package with the following two parts:

Solvers

There are three modules, flood, landslide and debris, solving the shallow water type equations using CUDA-enabled multiple GPUs, to simulate flood inundation, landslides runout and debris flows respectively. The solvers can predict the depth and velocities of these hazards. The flood solver, inherited from Pypims (Xia & Ming, 2023), is a Python-bound version of the open-source code hipims-cuda (HEMLab, 2020). Since the start of the SynxFlow project, the landslide solver of hipims-cuda has also been Python-bound and added to SynxFlow, and a debris flow solver has been implemented. These solvers use Godunov-type shock-capturing finite volume methods, which are documented in Xia et al. (2017), Xia & Liang (2018) and Xia et al. (2023). The numerical methods have been extensively tested in both theoretical and real-world scenarios, demonstrating robustness and efficiency for practical applications.

IO (Input and Output)

The IO module comprises of several classes and functions that generates the inputs to the solvers and process their outputs. The IO functionality for the flood solver is based on the open-source hipims-io (Ming & Xia, 2023) package, which has been extended to support the landslide and debris solvers. To set up a simulation, the users need to define an object using the 'InputModel' class, using common data formats including numpy arrays and pandas dataframes among others. An 'OutputModel' type object can be used to perform a variety of visualisation tasks including plotting maps, hydrographs and creating animations. To assist users, the IO module also provide functionalities to process geospatial data such as rasters and shapefiles. The IO module effectively integrates the solvers into data science workflows, streamlining hazard risk assessment tasks, and also enables coupling between solvers to facilitate multi-hazard simulations. The Python-based IO module also enables controlling scripts to be generated by Large Language Models (LLMs) such as ChatGPT and Gemini, which offers huge potential for automation.

Compared with existing software, SynxFlow is unique for its combination of open-source accessibility, multi-GPU acceleration, multi-hazard simulation capabilities, and a Python-based user interface.

Acknowledgements

The development of the code was enabled by the Baskerville Tier-2 HPC, which is funded by the EPSRC and UKRI through the World Class Labs scheme (EP/T022221/1) and the Digital Research Infrastructure programme (EP/W032244/1).

References

- Berger, M. J., & LeVeque, R. J. (1998). Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems. *SIAM J. Numer. Anal.*, 35, 2298–2316. <https://doi.org/10.1137/S0036142997315974>
- Clawpack Development Team. (2024). *Clawpack software*. <https://doi.org/10.5281/zenodo.13376470>
- HEMLab. (2020). hipims-cuda, commit 4aa7516. *GitHub Repository*. <https://github.com/HEMLab/HiPIMS-CUDA>
- Ming, X., & Xia, X. (2023). hipims-io, commit bafc98d. *GitHub Repository*. https://github.com/mingxiaodong/hipims_io_python
- Morales-Hernández, M., Sharif, M. B., Kalyanapu, A., Ghafoor, S. K., Dullo, T. T., Gangrade, S., Kao, S.-C., Norman, M. R., & Evans, K. J. (2021). TRITON: A Multi-GPU Open Source 2D Hydrodynamic Flood Model. *Environmental Modelling & Software*, 141, 105034. <https://doi.org/10.1016/j.envsoft.2021.105034>
- Shaw, J., Kesserwani, G., Neal, J., Bates, P., & Sharifian, M. K. (2021). LISFLOOD-FP 8.0: the new discontinuous Galerkin shallow-water solver for multi-core CPUs and GPUs. *Geoscientific Model Development*, 14, 3577–3602. <https://doi.org/10.5194/gmd-14-3577-2021>
- US Army Corps of Engineers. (2025). *HEC-RAS River Analysis System*. <https://www.hec.usace.army.mil/software/hec-ras/>
- Xia, X., Jarsve, K. T., Dijkstra, T., Liang, Q., Meng, X., & Chen, G. (2023). An integrated hydrodynamic model for runoff-generated debris flows with novel formulation of bed erosion and deposition. *Engineering Geology*, 326(October), 107310. <https://doi.org/10.1016/j.enggeo.2023.107310>
- Xia, X., & Liang, Q. (2018). A new depth-averaged model for flow-like landslides over complex terrains with curvatures and steep slopes. *Engineering Geology*, 234(January), 174–191. <https://doi.org/10.1016/j.enggeo.2018.01.011>
- Xia, X., Liang, Q., Ming, X., & Hou, J. (2017). An efficient and stable hydrodynamic model with novel source term discretization schemes for overland flow and flood simulations. *Water Resources Research*, 53, 3730–3759. <https://doi.org/10.1002/2016WR020055>
- Xia, X., & Ming, X. (2023). pypims, commit 27376f9. *GitHub Repository*. <https://github.com/pypims/pypims>