https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

A controlled protocol for synchronized advancement cost in GPU shallow-water models: scoped comparisons and implementation decisions with RapidFlow v0.1

Shuo Wang[1], Qinlu Zhang[1], and Yang Zhang[1]
[1]Independent researcher, Jinan, China
Correspondence: Shuo Wang (shuowang1998@163.com)

Abstract. We present a protocol for comparing synchronized advancement cost in GPU shallow-water models while holding local discrete evolution and absolute selected work fixed. Advancement includes host scheduling, selection, state maintenance and synchronization, and excludes construction and downloads. Its first gate checks sparse execution against the complete discrete operator. A non-flat Thacker candidate demonstrates why this matters: filtered and compacted paths agree with each other while both omit a nonzero momentum update. Changing only selection restores dense agreement through two periods. A default Thacker application passes its declared dense-stage, dry-extension and analytical-accuracy checks, with maximum strict 1 mm wet-set discrepancy of 5.15 % on a fixed common domain. Flat-domain experiments on RTX 4090 and RTX 5090 platforms then quantify implementation choices, with explicitly weaker dense-validation coverage. Across eight stored sizes, the conditional timing comparison finds filtering faster and its relative advantage generally narrows. Restoring omitted work confirms growing savings that the current compaction design does not convert into a net advantage. Preparation fusion helps small cases, but slows the largest flat case on the RTX 4090; its direction there is unresolved on the RTX 5090. The contribution is a reusable rejection and comparison procedure demonstrated by failure, repair and a successful application; flat timing matrices remain conditional implementation comparisons outside the audited representative endpoints. Numerical front failures and same-day timing sessions bound its use for model development and prevent general flood-accuracy or cross-date performance claims.

1 Introduction

Flood models resolve fronts that advance over topography, retreat and reactivate under rainfall or external inflow. On a graphics processing unit (GPU), the spatial distribution of wet regions determines both the work that can be omitted and the memory-access pattern of the remaining computation. Dense grid updates retain regular access but process dry regions. Cell lists restrict transport to a smaller domain at the cost of list construction and state maintenance. Block filtering offers an intermediate granularity: local stencil structure is retained, but selected blocks may contain many inactive cells. These choices are part of the broader accuracy and computational-cost trade-offs in flood modelling (Guo et al., 2021).

This study contributes an execution-comparison protocol with two demonstrated outputs: rejection of an invalid comparison and a measured implementation choice. The rejection is substantive: two sparse paths agree at every compared stage yet both

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

omit momentum updates made by dense execution; changing only their selection repairs the discrepancy (Sect. 4.2). The implementation choice cannot be read from launch counts alone: preparation fusion helps small cases but slows the largest flat case on the primary platform (Sect. 4.5). A default non-flat Thacker case supplies a complete successful application with analytical accuracy checks (Sect. 4.1). Protocol 1 combines dense omission validation, fixed-spacing dry extension, trajectory/support guards and restoration of omitted work. The extension separates stored extent from resolution, step count and absolute selected work. These controls, and the decisions they permit or reject, are the contribution; the cost of full-grid maintenance and the existence of sparse scheduling trade-offs are already known.

GPU shallow-water solvers have enabled rainfall–runoff and catchment-scale applications (Lacasta et al., 2015; Xia et al., 2019), including LISFLOOD-FP 8.0, TRITON and SERGHEI (Shaw et al., 2021; Morales-Hernández et al., 2021; Caviedes-Voullième et al., 2023). HiPIMS-CUDA and its Python interface connect rectangular-grid finite-volume calculations with input preparation and analysis (HEMLab, n.d.a, n.d.b); SynxFlow extends this lineage to multiple hazard solvers (Xia and Ming, 2025). Reconstruction, source discretization and stiff Manning friction also affect the operations required by each step (Xia et al., 2017; Xia and Liang, 2018; Zhao and Liang, 2022). Execution strategies must therefore be compared with their numerical operations held fixed.

Wet-region scheduling itself is well established. Sætra (2013) studies sparse domain updates and storage for shallow-water stencil computation. The available author draft compares full-domain execution, early exit, compacted sparse compute and sparse memory, and reports sparse memory becoming slower than early exit at approximately 35 % wet cells (its Sect. 3–4 and Fig. 5). Morales-Hernández et al. (2020) examine wet-domain versus full-domain computation alongside precision, data layout and output cost, emphasising elapsed time as well as scalability. These studies provide the direct context for assessing active-region management.

Related approaches reduce the number of spatial degrees of freedom. Aureli et al. (2020) combine block-uniform quadtree grids with GPU runoff modelling. LISFLOOD-FP 8.1 uses terrain-based multiresolution analysis, whereas version 8.2 implements dynamic multiwavelet adaptivity (Sharifian et al., 2023; Chowdhury and Kesserwani, 2025). Static and dynamic adaptivity have also been compared across flood regimes (Kesserwani and Sharifian, 2023). These methods change resolution and neighbour relationships as well as work counts. Sharifian et al. (2023) report a case in which fewer cells yield little runtime benefit because of the distribution of resolutions and neighbour-processing costs. The FIP solver similarly highlights the role of numerical implementation and memory and instruction utilisation (Rak et al., 2024).

Dynamic grid-adaptation studies also analyse evolving work and management costs. Chowdhury et al. (2023) combine Haar-wavelet adaptation, GPU tree traversal and CUB compaction with finite-volume updates; Chowdhury and Kesserwani (2025) report adaptation/update costs and step counts for multiwavelet DG2. Their changing spatial discretizations differ from the fixed-grid controls here. Neither reduced cell counts nor the need to include management time is a new observation claimed by this study.

Table 1 distinguishes stored-size controls from refinement and evolving wetness. Sætra's author draft already identifies full-grid block-launch latency and global memory reads (pp. 3–4). The present question is whether relative cost changes when dry storage grows while local discrete evolution and absolute selected work stay fixed. The 2020 wet/full comparison holds

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Table 1. Comparison of documented experimental designs. Fixed local evolution and absolute support refer to comparisons across stored sizes. NR means not explicitly reported in the examined full text, including the author draft for Sætra (2013); it does not imply absence. Timing boundaries differ and do not support a cross-solver speed ranking.

Study	Resolution across size tests	Fixed local discrete evolution	Fixed absolute selected support	Reported timing	Execution factors compared
Sætra (2013), author draft	Refinement of fixed 100 m square	NR	NR	Throughput and speedup; exact boundary NR	Full domain, early exit, sparse compute/memory
Morales-Hernández et al. (2020)	30 to 10 m	NR	NR	Wall clock, including CPU/GPU/MPI/I/O	Wet/all, precision, output
Chowdhury et al. (2023)	Adaptive level/tolerance	NR	NR	Solver runtime; exact host boundary NR	GPU tree traversal and CUB
Chowdhury and Kesserwani (2025)	Adaptive refinement level	NR	NR	Cumulative adaptation plus update cost	Adaptive/uniform comparison
This study	Fixed 0.125 m (flat); 0.15625 m (Thacker)	Saved states and steps; representative full-stage audit	Both-stage cumulative support	Synchronized advancement	Restore omitted work after compaction/preparation

inputs fixed at each resolution; its size comparison changes spacing and step counts. The 2025 study records changing cells, steps and adaptation costs (its Table 1). Our extension removes these particular confounders and restores omitted work in the compacted context. S16 identifies the examined versions and evidence locators. NR entries describe reporting coverage and cannot establish novelty on their own.

At fixed absolute support, enlarging the stored domain necessarily decreases relative support density. This experiment does not vary size independently of sparsity or refute a sparsity-based explanation. It distinguishes reduced relative density caused by added dry surroundings from a reduction in absolute transport support. Developers can then test whether compaction pays for its management at the same local workload, rather than infer that decision from wet fraction alone. The restored-work intervention tests omission in the compacted preparation context; it does not identify a pure bandwidth cost.

Sections 2–3 specify the operator, cost boundary and evidence requirements. Section 4 starts with the successful non-flat application and the rejected candidate with its selection-only repair, then examines flat-domain storage, omission and preparation decisions. Section 5 separates these execution results from numerical applicability. Auxiliary models remain in S32. Table 1 records reported controls, not a priority claim based on missing reports.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

2 Numerical formulation and GPU implementation

2.1 Conservation laws and spatial discretization

RapidFlow v0.1 is used here as a controlled execution experiment platform. The following equations and formal integration order define its operator; they do not establish general inundation accuracy. Section 5 separates the validated equilibrium/smooth-flow checks from the documented wet-front failures and the limited improved-front policy tests.

The conserved variables are water depth and discharge per unit width, $U = (h, q_x, q_y)^T = (h, hu, hv)^T$. On a Cartesian grid, the model solves

$$\partial_t U + \partial_x F(U) + \partial_y G(U) = S_b + S_f + S_c, \tag{1}$$

where F and G are the shallow-water fluxes, S_b represents bed-slope forcing, S_f Manning drag, and S_c rainfall, infiltration, drainage and specified mass or momentum inputs. The drag contribution to momentum in a wet cell is

$$S_{f,\boldsymbol{q}} = -g n_M^2 \frac{|\boldsymbol{q}|\boldsymbol{q}}{h^{7/3}}, \qquad \boldsymbol{q} = (q_x, q_y), \tag{2}$$

with Manning coefficient n_M and gravitational acceleration g.

The flux and forcing components are specified in Sect. S15. Hydrostatic reconstruction applies the bed contribution through face pressure corrections rather than as an uncoupled point source. Table 2 summarizes the depth thresholds.

Spatial discretization uses Harten–Lax–van Leer–Einfeldt (HLL/HLLE) fluxes with minmod-limited monotonic upstream-centred scheme for conservation laws (MUSCL) reconstruction of water surface, bed elevation and velocity. Hydrostatic reconstruction and matching pressure and bed-source corrections preserve the lake-at-rest balance (Audusse et al., 2004). Near-dry stencils revert to first order. A donor-cell positivity factor limits outgoing fluxes to the available water; preprocessing and transport use the same reconstructed face states. Dry-state and Froude clipping are applied after the update.

Depth thresholds affect both numerical evolution and transport selection. Local bed-difference reconstruction and explicit rounding prevent a near-dry face depth from exceeding its raw cell depth under fast math. The discrete rounding, positivity and clipping rules remain fixed across paths (Sect. S1). Boundary closure and its fallback are specified in Sect. S15; urban outflow conditions are given in Sect. S26.1.

These constants and numerical operations are fixed across execution paths. Source or boundary activity can select a region even when its depth is below the block-wetness threshold. Cell-index seeds retain every $h > 0$ cell.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Table 2. Principal numerical constants. Detailed discrete rules are given in Sect. S1 of the Supplement.

Quantity	Value	Role
Gravitational acceleration, g	9.81 m s^{-2}	Wave propagation and source terms
Near-dry depth, h_{near}	10^{-2} m	First-order reconstruction fallback
Transport depth, h_{tr}	10^{-3} m	Mass-transport threshold at reconstructed faces
Dry-state depth, h_{dry}	10^{-3} m	Inclusive block wet flags; momentum cleared below this depth
Positivity cutoff, ϵ_h	10^{-7} m	Smaller depths and associated momentum set to zero
Maximum Froude number	10	Post-transport velocity clipping

2.2 Time integration, sources and retained precision

Both time integrators use two stages of a strong-stability-preserving, second-order Runge–Kutta method (SSP RK2) for transport and bed-slope forcing. Let L denote this spatial operator, $\mathcal{S}_\tau$ the constant-rate constrained additive source flow, and $\mathcal{F}_\tau$ the fixed-depth Manning flow. The T1 method composes transport, sources and friction as follows (Strang, 1968):

$$\mathcal{F}_{\Delta t/2} \longrightarrow \mathrm{SSP\ RK2}(L) \longrightarrow \mathcal{S}_{\Delta t} \longrightarrow \mathcal{F}_{\Delta t/2}. \tag{3}$$

The rational friction flow keeps depth fixed and maps momentum to $\boldsymbol{q}/(1+\tau g n_M^2 |\boldsymbol{q}|/h^{7/3})$. T1 generally has first-order transport–source coupling; without additive sources it is a symmetric friction–transport composition. The smooth-solution splitting qualification is given in Sect. S15. Temporal convergence order is not measured in this campaign.

T2 uses the symmetric sequence

$$\mathcal{F}_{\Delta t/2} \longrightarrow \mathcal{S}_{\Delta t/2} \longrightarrow \mathrm{SSP\ RK2\ transport} \longrightarrow \mathcal{S}_{\Delta t/2} \longrightarrow \mathcal{F}_{\Delta t/2}, \tag{4}$$

For a source substep of length τ, let $a = r + \max(q_s, 0) + j$ be the supply rate and $b = \max(-q_s, 0) + i + d$ the requested removal rate. The constrained flow gives $h^+ = \max(0, h + \tau(a-b))$. If water is exhausted, the remaining supply is allocated to negative scalar source, infiltration and drainage in that order, and unmet demand is recorded. Injection adds its prescribed momentum; mass-only sinks retain momentum while wet. This closure removes water without removing horizontal momentum, so velocity can increase as depth falls before clipping. It differs from withdrawal carrying the local velocity, which would reduce momentum with depth; this distinction matters for the external-source comparisons (S26.3). Section S1 specifies applied integrals and clipping.

T2 uses a formally symmetric source composition. Formal second-order coupling under the smooth-flow assumptions in Sect. S15 is not a measured property of the complete implementation. It is not verified here by a spatially varying source temporal-convergence test. Clipping, exhaustion events and moving wet/dry interfaces fall outside that analysis.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Public state fields, reconstruction and fluxes use float32. Each depth and momentum state retains a float32 residual across stages and steps, while updates and convex averages use double intermediates. T1 evaluates its friction coefficient with double-precision square-root and power operations; T2 retains its float-coefficient friction formulation. The copy/fused controls keep this arithmetic fixed within each integrator; T1–T2 cost differences also include the friction formulation.

The use of retained residuals is motivated by rounding-error control in shallow-water calculations (Klöwer et al., 2020). A float32 main value and its float32 residual occupy eight bytes per scalar per stored stage, the same byte count as one float64 scalar. The design uses float32 reconstruction/flux arithmetic with double-intermediate state updates; its total storage is measured in Sect. S26.9. An all-float64 implementation would be needed to quantify a precision-related memory or runtime benefit.

Both methods advance one physical Δt per step. Limited boundary volumes receive half weight from each transport stage; source budgets record the actual applied source flows. A double host clock accumulates the float time steps used by the kernels. Public float32 snapshots reset hidden residuals on import, so snapshot import does not guarantee bitwise restart equivalence.

The adaptive time controller uses a global wet-cell wave-speed bound, prescribed inflow bounds and the Courant–Friedrichs–Lewy (CFL) condition, with growth, event and output limits (Sect. S2). The same controller is used across paths. Its minimum-step floor can override the CFL proposal and is monitored as a termination condition, not evidence of stability (Murillo et al., 2009).

2.3 Four execution paths

Dynamic block execution uses 16×16 spatial blocks. A 3×3 neighbourhood rule activates blocks around wet or source-active blocks. Fully valid blocks with valid halos use an interior kernel; clipped and mask-adjacent blocks use a boundary kernel. Transport loads a shared-memory tile with a two-cell halo. Kernels are launched over static valid-block coordinate lists and read the dynamic flags internally. Filtered blocks preserve their state and perform the required bookkeeping before skipping transport. Dense execution uses the same tiled numerical kernels over all valid blocks, providing the direct control for dynamic filtering.

Block-compacted execution stably selects interior and boundary coordinates from the same valid-block lists using the CUDA UnBound (CUB) library and the same activation flags. It launches the same numerical kernels over the compacted lists. Selection and host count readback occur at each transport stage. The present indexed implementation copies public and residual state and clears positivity storage over the full grid before updating selected blocks; this maintenance is included in complete advance time. The primary matrix uses copy preparation. The matched control in Sect. S26.9 replaces these copies/clears with one full-domain kernel preserving the same six fields and clearing positivity storage; list construction and numerical transport remain separate.

Cell-indexed execution selects valid cells with positive depth, nonzero momentum or active sources, expands their stencil neighbourhood, and uses stable CUB compaction to form flat cell indices. It rebuilds the list at each transport stage, reads the list count to the host and launches one thread per indexed cell with direct global stencil reads. Full-domain copies retain unlisted

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

public and residual state, and positivity buffers are reset before the stage. Both integrators construct two lists per physical step in this path.

The three depth residual buffers and six momentum residual buffers require $36N$ bytes, where N is the number of stored cells. This counts residual storage. The total device-memory deltas in Sect. S26.9 also include the main state and scheduler allocations.

Selection policies also differ: the indexed path retains all positive depths, whereas block wet seeds use $h \geq \mathrm{float32}(10^{-3})$ m, including equality where HLL can transport mass. Their neighbourhoods also differ in spatial granularity. Consequently, the cell/block comparison includes selection, state maintenance and memory access. The filtered-block/dense comparison most directly changes dynamic filtering within the same tiled implementation. Filtered and compacted blocks also share local numerical kernels, but change launch sets and preparation.

Table 3. Work at each transport stage. All paths perform full-valid-domain RK finalization, source/friction completion and wave-speed reduction; none may reuse a transport mask for finalization. The measurement platforms are described separately in Table 5.

Path	Seeds and halo	Positivity/transport traversal	State preservation / positivity reset	Count to host
Dense	All valid blocks	Full block lists; tiled stencil	In full block update	No
Filtered (F)	Wet/source/boundary blocks; 3 by 3 blocks	Full block lists; flag skips transport	Embedded before skip	No
Compacted (C)	Same flags as F	CUB-selected block lists; same tiled kernels	Full-domain copy/clear	Twice per step
Cell indexed	Positive depth/momentum or source seeds; radius-two stencil	CUB-selected cells; global stencil	Full-domain copy/clear	Twice per step

The cost structure defines the implementation family being compared (Table 4). Host count readback supplies the launch dimensions for compacted coordinate lists. Device-resident counts with persistent or overprovisioned launches, and alternative state-storage schemes, would change this structure and require new comparisons. Preserving omitted state is a numerical obligation; copying every stored cell is this implementation's way of meeting it.

For an omitted transport cell, a sufficient condition for exact preservation is that its complete local discrete stage map is the identity on public and residual state, its source/boundary contribution is zero, and it contributes no omitted speed or wet-flag maximum. Zero mass flux alone is insufficient: pressure/bed terms, clipping, residuals and RK2 averaging must also be accounted for. Neighbour expansion covers both reconstruction and donor-positivity stencils (radius two for spatial order 2). A raw subthreshold stencil uses first-order HR, whose depth cannot increase across its bed step; fully wet stencils already seed block activity. These conditions motivate the tested selection rule, with execution-consistency checks described in Sect. 3.1.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Table 4. Cost scaling for K physical steps, N stored cells and A cumulative selected block-stage updates across both stages. Scaling describes traversed work, not measured bytes or an additive timing model.

Operation	Work scale	Requirement and implementation choice
Wet flags and block selection	NK	Selection must cover every nonidentity update; full-grid flags and CUB scans are current choices
Count readback and launches	K	Host launch sizing is a current choice, not required by the numerical operator
Selected transport	256A	Required local stage map on selected blocks; blocks contain 256 cells
Omitted-state preservation / clear	NK in C/Cf	Correct state/residual preservation is required; full-grid copy or fused preparation is a choice
RK completion, sources and speed bound	NK in all tested paths	The complete valid domain must be accounted for; full-domain traversals are the current implementation

At each of the two transport stages, selection is rebuilt from the actual stage state. Positivity preprocessing and transport then update the selected region while preserving omitted public fields and residuals. RK2 finalisation covers every valid cell, including its stage-zero contribution; it cannot safely reuse a transport-stage mask. Source and friction completion are followed by updated wet flags and the wave-speed reduction for the next step.

Uniform rainfall bypasses block filtering. Nonzero source fields and open boundaries activate their required neighbourhoods even when water is absent. Hazard/plot wetness uses the separately stated strict diagnostic threshold; it does not determine transport selection. A self-contained face operator and retained-state stage description is in Sect. S1 of the Supplement.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

3 Controlled experimental design

3.1 Dry-domain extension and consistency controls

Protocol 1. Controlled dry-domain comparison. The checks and timing boundary are specified separately from the coverage achieved by each experiment (Table 6). Shared terminology is defined in Supplement Table S1. Inputs are a versioned operator and execution paths, local state/bed/mask and forcing, fixed spacing, output times, embeddings and declared numerical criteria. Output is a check-coverage record plus paired complete-time ratios and contrasts.

1. **Validate omission against complete execution.** On declared trajectories, compare each sparse path with dense or an independent complete execution of the same discrete operator: public states, hidden residuals and actual dt at each stage. A dilation/list oracle checks implementation of the selection rule, not whether that rule preserves the operator. Without a complete baseline, supply checkable local identity-map reasoning and boundary tests, explicitly labelling this weaker evidence. Revalidate after changes to numerical policy, thresholds, preservation or finalization; F/C agreement and mass conservation alone cannot pass this gate.
2. **Embed and guard.** For paths with declared omission evidence, add only dry surroundings. Check common states/residuals, actual dt, stage support and zero exterior across sizes, including inverse shifts where used. State the trajectory and capture coverage; historical saved-output or F/C-only checks retain their narrower labels. Failed guards reject fixed-workload attribution.
3. **Measure complete cost.** Randomize size/path order within paired repeats, warm up, synchronize advancement, and record both host clocks, platform metadata and process points. Keep numerical audits and profiles outside timing. Clock disagreement limits clock-specific interpretation; agreement does not calibrate historical times or establish cross-session robustness.
4. **Intervene on omission.** Retain compacted selection/preparation, restore full block launches and dry preservation, and recheck states/support. Report R-C within C's context and R-F with its interactions. The algebraic difference identity is an accounting check, not an independent mechanism test.
5. **Decide within scope.** Report endpoint preference only for passing cases, retain intervals spanning parity and no-reversal outcomes, and separate execution consistency from reference-based numerical accuracy. Auxiliary models use whole-geometry splits with explicit prospective/post-hoc status.

Eight local geometry groups contain rectangles with different aspect ratios and seeded patches with different dimensions and counts. Two development inputs serve as the core geometries for the stage audits, interventions and size scan: the continuous input is one 64×64-cell wet square (`dev_square_64`), and the fragmented input is 64 nonoverlapping 8×8-cell wet patches (`dev_patch_8x8`, seed 2026091402) distributed over aligned sites in a 192^2-cell region. Both contain 4096 initially wet cells, or 64 m^2, within a 256^2 local array on a flat bed with dry surroundings. Here fragmented describes the synthetic wet-patch distribution. Figure 4 uses four separate held-out inputs, whose dimensions and group assignments are listed in S8.

The same local input is embedded in 512^2, 1024^2 and 1536^2 dry domains at x offsets 0, 7 and 15 cells; the size scan of Sect. 4.6 adds 640^2, 768^2, 896^2, 1152^2 and 1280^2. Spacing is 0.125 m: these are 64, 128 and 192 m square stored domains, containing 262,144 to 2,359,296 cells. Initial wet depth is 1 m, Manning 0.03, CFL 0.4, cap 0.05 s, and the integrator is T2. Continuing windows end at 0.25, 1 and 2 s. Saved common-domain states, inverse translations, omitted dry strips, domain

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

boundary frames, steps and absolute selected update counts (not fixed instructions or DRAM traffic) are checked before cost interpretation. The grouped geometry campaign uses three processes with one warmup, three paired ordinary repeats and one separate profile per configuration; the endpoint and scan campaigns use three processes with five or seven paired repeats as stated in each section. Across the matrix these are saved-state and cumulative checks, not observations of every intermediate
195 stage. The S20 audit covers representative default trajectories; S27 adds 12 default preparation combinations. S31 adds 12 thin-momentum flat-endpoint trajectories. These audits do not cover all holdout or non-flat candidate configurations (Supplement Table S14). Full execution inventories are given in Sect. S8.

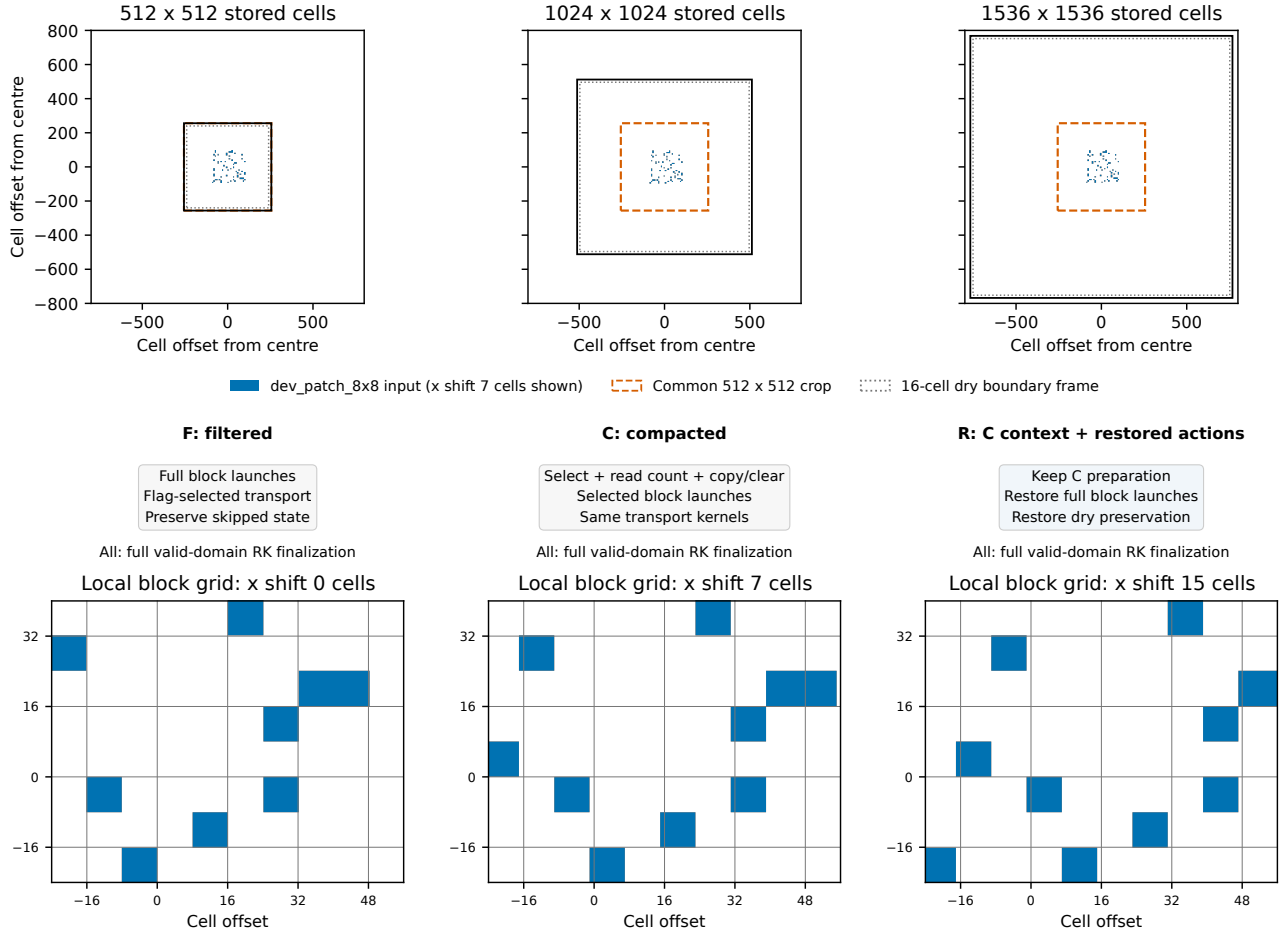


Figure 1. Controlled dry extension at fixed spacing. The same 256^2 local input (`dev_patch_8x8`) is centred in three stored domains; the seven-cell x translation is illustrated (the study also uses offsets 0 and 15). Dashed squares mark the common 512^2 crop; dotted frames delimit a 16-cell outer strip that must stay dry. Inverse translation also requires zero omitted strips. Local views show offsets 0/7/15 against the 16-cell block grid at true scale. The execution panels distinguish launch arrays, selection/preparation and preservation for F, C and R; all retain full-valid-domain RK finalization.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

3.2 Platform, timing and uncertainty

Measurements use two rented Linux GPU instances, described in Table 5. The primary platform is an NVIDIA RTX 4090 (Ada, sm_89), which carries the complete campaign: numerical validation, the grouped dry-extension study with its held-out geometries, the cost-model evaluations, the dual-clock endpoint measurements, the eight-size scan, the preparation endpoints and the bounded interventions, including the omission diagnosis and complete non-flat application (S33–S34). The second platform is an NVIDIA RTX 5090 (Blackwell, sm_120), which repeats a fixed subset under the same protocols: numerical validation, the eight-size scan, the dual-clock endpoints and the preparation endpoints. The RTX 5090 does not repeat the cost-model study or the full performance matrix, so no cross-device model transfer is claimed.

Both run Ubuntu 22.04 CUDA containers with nvcc 12.8.93, release mode, fast math, 16×16 blocks, HLL/HLLE and retained-residual updates, Python 3.10.21 and NumPy 2.2.6. The native numerical sources are identical; only the target architecture differs, and every interface is recompiled per device. The two instances differ in host processor, kernel build and driver version, so an absolute time difference between hosts cannot be attributed to the GPU architecture. Paired within-platform ratios are therefore the primary comparison throughout. Container GPU access is not verified exclusive use of the physical machine; our GPU workers run serially, but fixed clocks, controlled temperature and absence of other tenants are not guaranteed. Measured cost belongs to each complete execution platform.

Complete advancement includes host scheduling, list-count readback, numerical updates, state maintenance and synchronization. Construction, input preparation, downloads and metric reads are excluded; it is not whole-application elapsed time. Source-event measurements additionally include repeated source uploads; the urban constant-source upload occurs during excluded setup. Paired contrasts use the same interval boundary and physical output targets.

The grouped campaign uses synchronized `perf_counter` durations. Later controls also record `CLOCK_MONOTONIC_RAW` around the same advancement; their paired F/C ratios differ by at most 0.00011. We use raw durations for these controls and retain the original clock for historical samples. Separate CUDA-event profiles are not combined with host durations into a budget. Clock probes, boundaries and historical details are in S23–S24.

A ratio is always baseline ordinary time divided by variant ordinary time; values above one favour the variant. Reported 95 % intervals use 10,000 bootstrap resamples of processes and paired repetitions, with the median of process medians as the point estimate. For complete trajectories, the continuing-window durations are summed first within each process/repeat/path. The F/C ratio or F-C difference is then computed for each paired repeat, followed by the median over repeats within each process and the median over processes. Bootstrap draws resample processes and paired repeats within sampled processes; windows remain together as one trajectory, rather than being resampled independently. These intervals describe conditional variation within the recorded session on the stated device, not confidence intervals across sessions or devices. Three process clusters provide only coarse information about process-level variation; resampling repetitions or increasing the bootstrap draw count adds no independent sessions. Spatial realizations are fixed, and the intervals are not multiple-comparison-adjusted significance tests. The scan and preparation measurements were repeated in a second session on the primary platform, begun about one hour after the first on the same instance with the same binaries and no host reboot. That repetition tests session-to-

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Table 5. Measurement platforms and dataset coverage. L2 capacity is a CUDA device query, not a measured cache hit rate or DRAM traffic. Hosts differ in processor and driver, so cross-host absolute times are not comparable.

Property	Primary platform	Second platform
GPU	NVIDIA RTX 4090, 24,564 MiB	NVIDIA RTX 5090, 32,607 MiB
Architecture / build target	Ada, sm_89	Blackwell, sm_120
Reported L2 capacity	75,497,472 bytes (72 MiB)	100,663,296 bytes (96 MiB)
NVIDIA driver	570.124.04	580.76.05
Host processor	Intel Xeon Platinum 8481C	Intel Xeon Gold 6459C
Host kernel	Linux 5.15.0-94-generic	Linux 5.15.0-119-generic
Numerical and smoke validation	Yes	Yes
Eight-size F/C scan	Two separate sessions	One session
Dual-clock F/C/R endpoints	Yes	Yes
Preparation endpoints (F/C/Cf)	Two separate sessions	One session
Grouped geometry study and cost models	Yes	Not repeated
Full performance matrix and ancillary workloads	Yes	Not repeated
Omission diagnosis / non-flat full protocol (S33–S34)	Yes	Not repeated
Hardware counter access	Not available	Not available

session stability within one day only. Separated-date and post-reboot robustness has not been established, and results from two devices do not establish generality across GPUs. Instrumented profiles are separate runs; event ranges can include launch gaps and instrumentation costs. Embedded filtered-state preservation and explicit indexed preparation are not identical transport accounting categories. The reported event timings contain no hardware-counter measurements and cannot determine DRAM traffic, cache residency or a bandwidth bottleneck; these mechanisms remain possible.

3.3 Numerical diagnostics

Each comparison holds the numerical update, grid and input fixed while changing the execution path. Conserved fields and step counts are compared at saved times, so timing differences can be interpreted without attributing them to changed numerical trajectories. This constrains execution consistency within the tested problems, rather than establishing a general accuracy guarantee.

Four diagnostics are kept distinct: execution-path agreement, field error, wet-region discrepancy and volume balance. For conserved field a over valid cells with area A_i, the absolute area-weighted error is

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

$$E_1(a) = \frac{\sum_i A_i |a_i - \overline{a}_i^{\mathrm{ref}}|}{\sum_i A_i}. \tag{5}$$

For a strict reporting threshold η, define $W_\eta = \bigcup_{i:h_i>\eta} \Omega_i$ and

$$D_\eta = \frac{|W_\eta^{\mathrm{num}} \triangle W_\eta^{\mathrm{ref}}|}{|W_\eta^{\mathrm{ref}}|}. \tag{6}$$

Two empty wet sets give zero discrepancy. A nonempty numerical set against an empty reference has undefined discrepancy and fails the criterion. Both 1 and 10 mm are reported. These strict diagnostic cutoffs differ from the inclusive 1 mm numerical block-wetness flag; cell-index seeds retain all $h > 0$ cells. Triangular conserved fields are area-projected and conservatively restricted before thresholding; binary wet masks are not averaged to construct this diagnostic.

The acceptance thresholds and their dimensional scales are given in Supplement Table S2. They are retained diagnostic criteria, not application-calibrated flood prediction tolerances. Field agreement, front agreement and volume conservation are therefore interpreted separately.

3.4 Core experiment protocols

Supplement Table S30 consolidates core and auxiliary protocols. R reuses the audited restored-work module; F and C use the frozen baseline. Table 8 reports the dual-clock session's endpoint effect sizes and check coverage. Numerical validation, list/stage probes and ordinary timing are separate runs.

The single-target sparse runs take 183/195 steps for square/fragmented inputs; the continuing 0.25/1/2 s protocol takes 185/196 because output targets truncate steps. These are different specified trajectories. Static-model sensitivity reuses archived data without GPU integration (S25). Exact ancillary input protocols and execution inventories remain in the supplement and reproduction map; repetition counts do not create additional independent geometries.

Numerical inputs comprise Ritter and Thacker analytical solutions (Delestre et al., 2013), lake-at-rest and analytical channel checks (MacDonald et al., 1997), flat circular expansion, paired withdrawal/restart, and the Merewether terrain workload. Full source rectangles/rates, event times, terrain mapping, inlet coordinates and outflow closure are specified in S26. Numerical criteria are in S3. Front accuracy is assessed independently of execution agreement.

3.5 Acceptance and rejection before cost interpretation

Table 6 distinguishes complete-operator validation from sparse-path and cross-size agreement. The default non-flat application meets the dense stage gate. The added review audit closes that gate for the two core flat geometries, widths 512 and 1536, and the default, thin-momentum and stable-scan policy groups. Historical flat timing matrices and intermediate scan sizes retain weaker coverage. The sufficient condition in Sect. 2.3 motivates selection; it is not a trajectory-wide proof outside the audited configurations. We therefore report uncovered flat timings as conditional comparisons of the tested implementations, not as full passes of Protocol 1.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Table 6. Main evidence coverage. "Stage" means all six public/residual capture positions per step on the specified trajectories. A list oracle tests the implemented selection rule, not its numerical sufficiency. Full inventories and binary/release bindings are in Table S14.

Evidence	Dense comparison	Cross-size check	Residuals / actual dt	List oracle
Grouped flat extension (S8)	Saved public outputs	Saved fields, steps, cumulative support	Not full-stage	Not full-stage
Default flat endpoints (S20/S27)	No matched dense stage audit in historical runs	F/C or F/C/Cf stages and support	Stage / every dt	Both stages
Added representative flat audit (review_dense)	D/F/C/Cf stage equality at 512^2/1536^2	Both geometries and sizes	All six fields / every dt	Both stages
Policy / stable-scan endpoints (S31)	Historical run had no matched dense stage audit	Sparse variants at endpoints	Stage / every dt	Reported stage checks
Eight-size scan (S29)	No added dense audit for intermediate sizes	Saved fields, steps, support	Not full-stage	No added audit
Original / repaired candidate (S33)	Fail / pass dense stages	Single stored size	First failure / all stages	Failure localized; seed repair
Default non-flat extension (S34)	D/F/C/Cf/R stages	All three sizes and support	Stage / every dt	Both stages

The candidate policy provides a contrasting outcome. Removing the Froude cap and retaining thin-layer momentum improves the Ritter front, but at 256^2 in the non-flat Thacker test the two block paths agree with each other and disagree with dense, while cell indexing agrees with dense. This is a rejected execution comparison, regardless of its speed. Separately, default and candidate Ritter fronts lag by 57 and 13 cells, respectively, exceeding the two-cell criterion. Section 5 reports these accuracy and consistency limits in detail. Neither agreement between sparse paths nor mass conservation certifies valid omission or application accuracy.

4 Protocol outcomes and implementation decisions

4.1 Complete application on a non-flat moving shoreline

To test the protocol beyond flat wet patches, the default translating Thacker basin is held fixed on a 256^2, 40 m common domain while its parabolic bed and dry storage extend to 512^2 and 768^2. Resolution stays 0.15625 m. Eight outputs through two periods exercise wetting and retreat. Fifteen dense/F/C/Cf/R trajectories pass all 17,496 public/residual stage captures per trajectory, actual dt and zero-exterior checks; selected support totals 645,007 block-stage updates (counting a block again at each stage) across both stages and is identical across stored sizes for the sparse paths. Release states and steps match the audits (S34). For K=2916 steps and A=645,007 block-stage updates, the stage-averaged selected fraction 256A/(2KN) is 43.20 %, 10.80 % and 4.80 % at N=65,536, 262,144 and 589,824 cells, respectively. It is not the initial wet fraction.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Accuracy is evaluated on the fixed common crop. Over eight outputs, maximum depth L1 is 0.000783 m, either momentum L1 is 0.001847 m^2/s, and strict 1/10 mm wet discrepancy is 5.15/1.40 %. Relative volume error is below 4e-10. Doubling reference quadrature changes neither thresholded wet set. The chosen resolution and workload-specific accuracy bounds use the existing refinement results, so this is a bounded protocol application, not independent discovery of a broadly accurate numerical method.

Table 7. Default Thacker complete advancement through two periods on the RTX 4090. RAW-clock ratios and restoration effects have conditional 95 % process/repeat intervals (three processes, five paired repeats). All sizes preserve the same checked local trajectory. Absolute times and signed effects are in Table S43; numerical and spatial diagnostics are in Fig. 2.

Stored width	F/C [95 %]	C/Cf [95 %]	R-C ms [95 %]
256	0.614 [0.608, 0.674]	1.252 [1.243, 1.257]	25.8 [24.1, 30.1]
512	0.785 [0.775, 0.823]	1.209 [1.173, 1.219]	95.9 [91.2, 113.8]
768	0.850 [0.847, 0.875]	1.102 [1.097, 1.108]	118.3 [114.3, 123.1]

For this dense-audited representative Thacker application, filtering remains faster as F/C rises from 0.614 to 0.850. Fusion improves C at all three sizes, but F/Cf remains below one. Restoring omitted work costs 25.8–118.3 ms, yet this saving does not make C faster than F. The measured choice in this session is F; if C is used, fused preparation is preferable at the tested sizes. Thus fewer selected blocks alone still do not determine complete cost, while the preparation decision requires workload-specific measurement. These sizes differ from the flat 1536^2 test, so the contrast does not isolate terrain as the cause of different fusion outcomes.

The numerical bounds support a controlled moving-shoreline development benchmark: depth L1 <=0.005 m, each momentum L1 <=0.01 m^2/s and strict wet discrepancy <=10 % on the fixed 40 m crop. They were chosen using the prior refinement results and do not certify inundation extent or arrival time for an event model.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

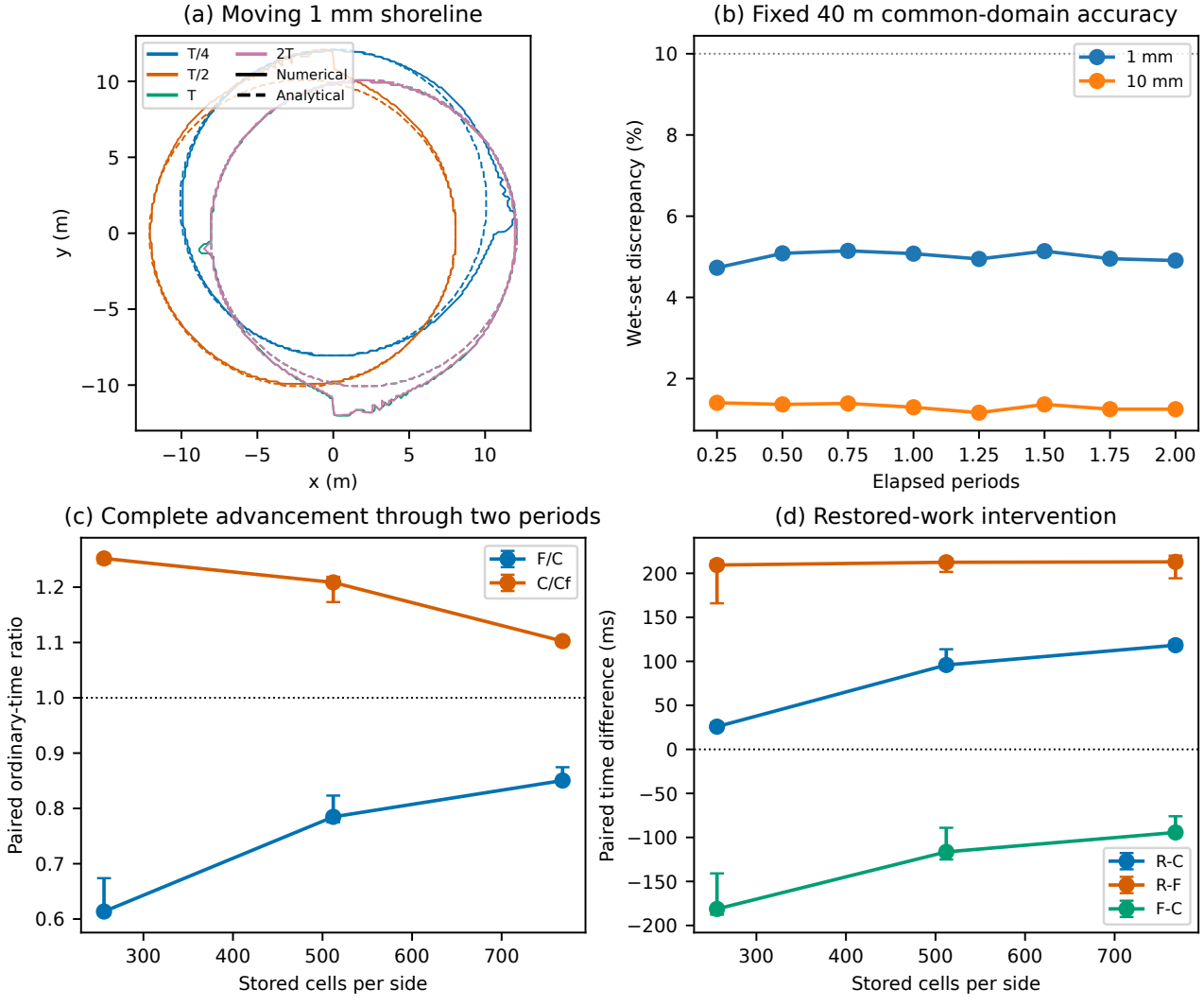


Figure 2. Default Thacker application: (a) numerical and analytical strict 1 mm shorelines at T/4, T/2, T and 2T, (b) wet discrepancy over eight quarter-period outputs on the fixed 40 m crop, (c) complete paired F/C and C/Cf ratios under dry extension, and (d) restored-work and net signed effects. Bars in (c–d) are conditional 95 % process/repeat intervals. Colours identify times; the analytical T and 2T contours coincide. Added dry surroundings change stored cost, not the checked common-domain trajectory or the error normalization.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

4.2 Rejection and selection-only repair

The contrasting follow-up tests the cap-off, thin-momentum policy on four Ritter grids and a translating Thacker basin through two full oscillation periods (Sect. S30). All trajectories retain volume and positivity guards. At 256^2, the candidate filtered and compacted Thacker paths agree with each other but depart from dense by up to 0.0008104 m in depth; the cell path agrees with dense. This case is rejected as a successful application of Protocol 1 and prevents candidate promotion: F/C agreement alone cannot establish valid omission. A stage-resolved replay locates the first difference at step 673, second Euler output, starting at t=6.534718 s (S33). Nineteen cells in an omitted block have depths below 1 mm. Their depth remains unchanged, but dense updates momentum and residuals while F/C preserve the input. The subthreshold flux retains pressure contributions, so zero mass transport does not imply an identity update when thin-layer momentum is retained. Changing only block seeds to all positive water restores agreement with the original dense trajectory at all 17,496 captured frames per path through two periods; release outputs also match. A 32^2 one-step counterexample reproduces the failure and its removal. This isolated selection repair is not a new numerical default and does not remove the Ritter accuracy failure.

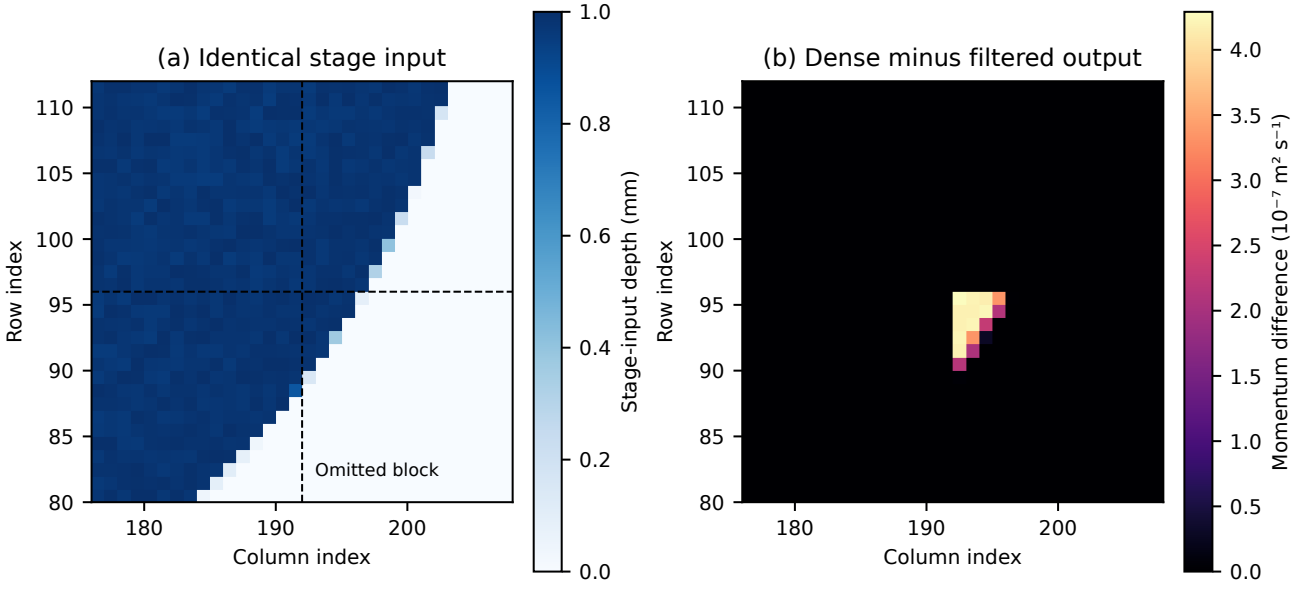


Figure 3. Candidate Thacker first divergence at step 673, second Euler output. The local stage-input depth is identical between paths; dashed block edges identify the omitted block. The right panel shows the magnitude of dense-minus-filtered momentum at the same output. Nineteen subthreshold cells change momentum under dense execution while filtering preserves their inputs. Depth is still identical. This is a stage diagnostic, not an accuracy comparison against the analytical solution.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

4.3 Stored extent changes relative cost without changing the ranking

The controlled extension preserves the saved common-domain fields, physical steps and absolute selected support while increasing stored cells. The timing matrix below is a conditional implementation comparison; the representative dense audit covers the two core geometries at 512^2 and 1536^2, not every point in this held-out matrix. All cross-size, translation, dry-exterior and boundary checks pass across 1728 paired ordinary samples and 9180 saved windows. Filtering is the faster path at every tested size. Across the twelve held-out geometry/offset combinations, filtered/compacted complete-time ratios rise from 0.757–0.811 at 512^2 through 0.856–0.876 at 1024^2 to 0.946–0.964 at 1536^2, and all 36 conditional 95 % intervals lie wholly below one. Adding dry storage therefore reduces the margin by which filtering wins, from roughly 19–24 % to roughly 4–5 % of compacted time, without reversing the ranking anywhere in this range.

The direction of that trend is consistent, and its endpoints are well separated, but the intermediate behaviour is not strictly ordered: the eight-size scan of Sect. 4.6 contains consecutive pairs whose point estimates move the other way. We therefore report a narrowing advantage over the sampled range rather than a monotonic law, and we make no claim about sizes outside 512^2–1536^2. A crossover beyond the tested range is neither demonstrated nor excluded.

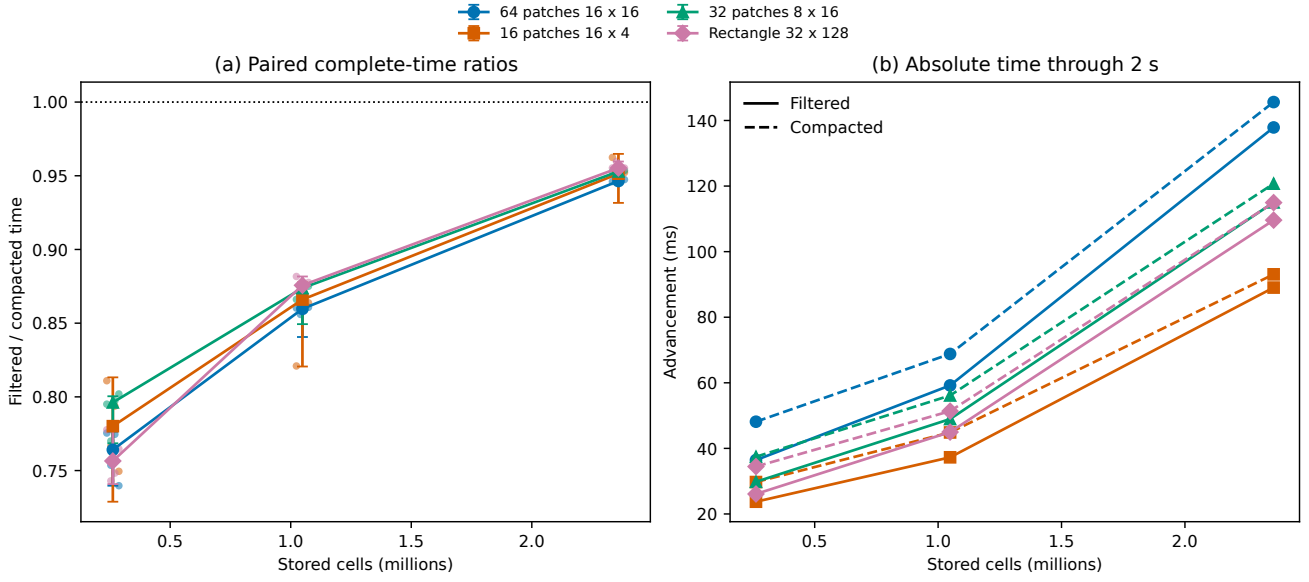


Figure 4. Conditional timing comparison for controlled dry-domain extension on the RTX 4090; the matched dense audit covers representative core endpoints, not these four held-out geometries. The figure shows four unshifted held-out geometries, distinct from the two core development inputs defined in Sect. 3.1: (a) paired filtered/compacted advancement-time ratios, small process-median points, and conditional 95 % hierarchical bootstrap intervals over three processes and three repeats; (b) absolute times through 2 s. Setup/downloads are excluded. Ratios below one favour filtering, which wins at every tested size; the dotted line marks parity. Lines connect tested sizes and do not extrapolate beyond them.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

The added review audit closes dense stage coverage for the two core flat inputs at 512^2 and 1536^2 under the default, thin-momentum and stable-scan policies: 48 D/F/C/Cf trajectories, 54,816 six-field captures, actual dt and release-window bindings all pass. The portable receipt replays stage hashes, order, dt and support; the original GPU runner performed the field comparisons, while full stage arrays are retained only where stated in the archive. It does not cover intermediate scan sizes, held-out geometries, shifted embeddings, Fr100, other cutoffs or the second GPU. The historical S20 audit still checks F/C stage fields, residuals, actual dt, selected support and zero exterior for its representative trajectories. Full-domain selection, preservation and finalization belong in cost assessment, but their measured costs are not assumed irreducible or identified with DRAM traffic.

4.4 Restoring omitted work with paired clocks

An ordinary-time intervention retains compacted selection, count readback and preparation but restores the original full block launch arrays for positivity and transport. It consequently restores the filtered branch's dry-block preservation and reductions while keeping all 27 numerical device-function occurrences instruction-identical. Call this construction R.

The default F/C/R construction is measured on each platform in three serial processes, with seven randomized ordinary repeats and both host clocks around every advance. All 432 trajectories per platform pass saved common-state, window-step, C/R support, mass and exterior checks. Table 8 reports a conditional timing comparison; representative dense stage coverage is supplied by the added audit, not by every timed endpoint. Filtering is faster at both sizes on both devices, and the compaction deficit narrows with stored extent in the same way as the grouped study: F/C rises from 0.805–0.811 at 512^2 to 0.954–0.961 at 1536^2 on the RTX 4090, and from 0.710–0.740 to 0.951–0.962 on the RTX 5090. Both clocks give the same values to within 0.00011 in the paired ratios.

Restoring omitted work costs measurably more as stored extent grows. On the RTX 4090, R-C is 6.5 [6.3, 7.4] ms at 512^2 and 36.9 [36.6, 37.2] ms at 1536^2 for the continuous input, and 5.0 [4.9, 5.2] to 38.7 [38.1, 38.8] ms for the fragmented input. R-F is larger still, 12.4 [12.2, 16.2] to 41.5 [40.8, 42.4] and 12.2 [11.9, 16.4] to 44.3 [43.5, 44.8] ms. Net F-C stays negative at both sizes, between -7.3 and -4.4 ms, because the measured compacted implementation's added selection and preparation context costs more than the dry-block work it omits at every size tested. The same pattern holds on the RTX 5090 with smaller magnitudes: R-C grows from 4.4 to 22.6 ms and from 4.8 to 23.3 ms across the two inputs.

For every paired sample, $F - C = (R - C) - (R - F)$, but separately summarized medians need not satisfy that identity exactly. Its validity is algebraic; causal information comes from the controlled restoration, not the identity. R-F is not pure compaction management overhead: it contains extra management, redundant preservation under the compacted preparation context, and cache and launch interactions. Because the restoration grows faster than the net difference, the measurements show that the omitted dry work is a real and growing saving that the current compacted implementation does not convert into a net gain within this range. They do not identify a per-kernel cost budget.

Separate timing-envelope checks are documented in S21 and S23. Instrumented partitions perturb execution and are not added to ordinary times as a budget.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Table 8. Conditional F/C endpoint timing comparison in the default dual-clock sessions on both platforms; dense stage coverage is representative rather than complete for every timed endpoint. Ratios and differences summarize paired complete trajectories on CLOCK_MONOTONIC_RAW; intervals are conditional 95 % process/repeat bootstrap intervals. Values below one favour filtering; F saving relative to C is $100(1-r)$ with bounds transformed from the ratio interval. All rows pass saved common fields, window steps, C/R support and dry-exterior guards. S20 separately audits the corresponding RTX 4090 default F/C endpoint configurations at every step and stage; it does not audit every timing repeat or improved numerical policy.

Platform / input / width	F/C [95 %]	F-C (ms) [95 %]	F saving (%) [95 %]
4090 / continuous / 512	0.811 [0.756, 0.821]	-6.1 [-9.0, -5.7]	18.9 [17.9, 24.4]
4090 / continuous / 1536	0.961 [0.956, 0.963]	-4.4 [-5.2, -4.2]	3.9 [3.7, 4.4]
4090 / fragmented / 512	0.805 [0.737, 0.815]	-7.3 [-11.4, -6.8]	19.5 [18.5, 26.3]
4090 / fragmented / 1536	0.954 [0.952, 0.960]	-5.6 [-6.0, -5.0]	4.6 [4.0, 4.8]
5090 / continuous / 512	0.710 [0.699, 0.715]	-9.4 [-10.0, -9.2]	29.0 [28.5, 30.1]
5090 / continuous / 1536	0.962 [0.958, 0.967]	-2.9 [-3.3, -2.6]	3.8 [3.3, 4.2]
5090 / fragmented / 512	0.740 [0.736, 0.743]	-9.4 [-9.5, -9.3]	26.0 [25.7, 26.4]
5090 / fragmented / 1536	0.951 [0.949, 0.953]	-4.2 [-4.4, -4.1]	4.9 [4.7, 5.1]

RTX 5090 signed differences are recomputed from the archived within-process, within-repeat paired durations using the same hierarchical estimator (S23). They describe that platform; differences between hosts cannot isolate GPU architecture effects.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

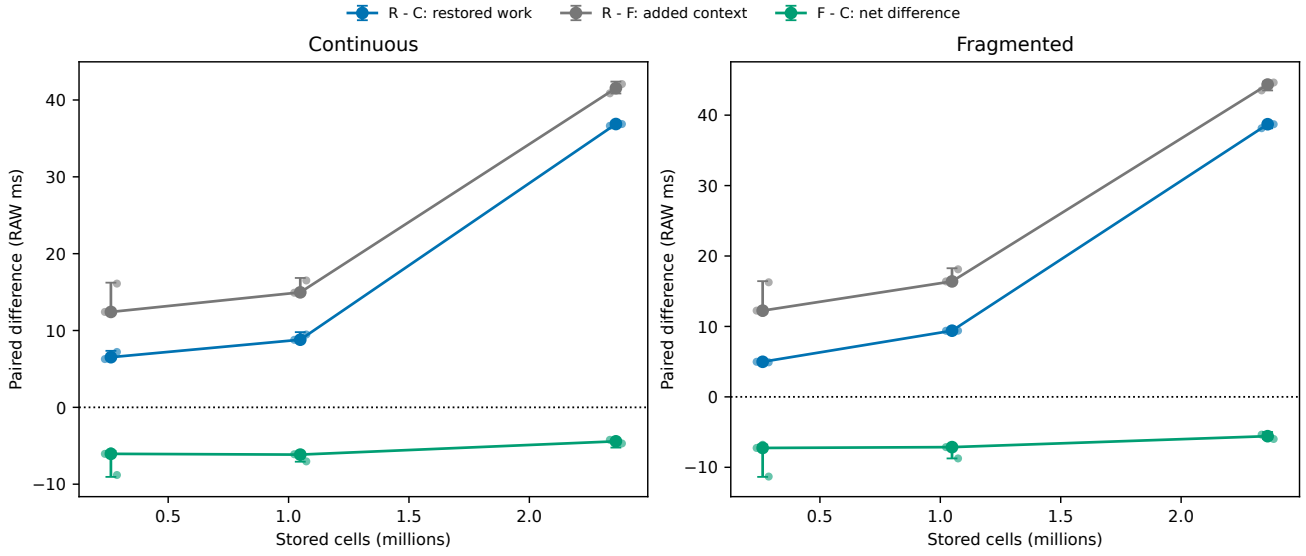


Figure 5. Dual-clock ordinary-time intervention on CLOCK_MONOTONIC_RAW for the RTX 4090. Table 8 and Table S22 report the RTX 5090. R restores full-domain positivity/transport launch arrays after C's selection and preparation. R-C is the effect of restoring that work in C's context; R-F includes extra management, redundant preservation and cache/launch interactions. Small points are process medians; bars are conditional 95 % intervals from three processes and seven paired repeats. Setup/downloads are excluded. F-C remains negative at every size, so the restored saving does not make compaction the faster path within this range.

4.5 Preparation fusion across sizes and platforms

The existing fused preparation combines the six-field copy and positivity clear in C without changing its selection or numerical update; call this path Cf. A matched preparation comparison with the coverage in Table 6 tests F, C and Cf on both endpoint sizes and both core geometries, on both platforms, and twice on the primary platform. Its timing matrix is conditional outside the representative dense-audited core endpoints. All 12 untimed trajectories per platform agree in actual dt, six public/residual capture positions per step and both stage supports; their saved public fields and window steps match release trajectories. The audit contains 13,716 capture positions and 4572 stage-support checks. All 216 warmup/ordinary release trajectories per session pass their saved-output guards (S27); the added review audit now supplies dense validation for its listed core endpoints, but not for every preparation or policy configuration. The second primary-platform session reuses that unchanged-binary stage audit rather than repeating it, and contributes fresh ordinary samples only.

Table 9 reports the C/Cf contrast, in which values above one mean fusion is faster. At 512^2 fusion helps on both devices and in both primary-platform sessions, by 12–16 % for the continuous input and 8–12 % for the fragmented input, with every interval above one. At 1536^2 the evidence resolves different within-platform conclusions. On the RTX 4090 fusion is slower, with C/Cf 0.958 and 0.966 for the continuous input in the two sessions and 0.957 and 0.964 for the fragmented input, all four intervals wholly below one. On the RTX 5090 the point estimates are also below one, 0.984 and 0.985, but both intervals

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

include one, so the direction at that size is unresolved on that device. We therefore do not describe the RTX 5090 large-size result as equivalent performance or as a confirmed slowdown.

This is the clearest implementation consequence in the study. Fusion removes launches and is beneficial at the small size everywhere tested, yet at the large size it is measurably harmful on one device and undetermined on the other. The controlled decision on the primary platform is to retain copy preparation for large inputs. That decision does not transfer by argument to the second device, which is precisely why the protocol measures it. This preparation contrast concerns two variants of the compacted path only. It is a different question from the filtered/compacted ranking of Sect. 4.3, which fusion does not change: F/Cf stays below one in all four groups on both platforms, so fused preparation never makes the compacted path faster than filtering. Per-process estimates, signed effects and conditional intervals remain in S27.

Table 9. Conditional copy-versus-fused preparation timing, C/Cf on CLOCK_MONOTONIC_RAW; dense stage coverage is limited to the listed representative core endpoints. Values above one favour fusion. Intervals are conditional 95 % process/repeat bootstrap intervals from three processes and five paired repeats. The two primary-platform sessions are reported separately and not pooled. An interval containing one is recorded as unresolved, not as equivalence.

Platform / session	Input	512^2 C/Cf [95 %]	1536^2 C/Cf [95 %]
4090 / session 1	Continuous	1.189 [1.179, 1.341]	0.958 [0.957, 0.971]
4090 / session 1	Fragmented	1.132 [1.131, 1.138]	0.957 [0.955, 0.958]
4090 / session 2	Continuous	1.197 [1.178, 1.226]	0.966 [0.962, 0.968]
4090 / session 2	Fragmented	1.137 [1.131, 1.142]	0.964 [0.963, 0.966]
5090 / single session	Continuous	1.171 [1.144, 1.227]	0.984 [0.982, 1.004]
5090 / single session	Fragmented	1.086 [1.065, 1.090]	0.985 [0.981, 1.007]

The two primary-platform sessions agree on direction. Their large-size C/Cf ratios shift by 0.008 and 0.007 for the two inputs, while the deficits from parity are 0.034–0.043. Thus the observed same-day shift is roughly one fifth of that deficit. Two sessions do not estimate cross-date variability; the copy preference is conditional on these platform, workload and session conditions.

A minimal record interface and runnable success/failure examples are provided in `docs/ENDPOINT_PROTOCOL.md`. The adapter supplies common-field hashes, window steps, support, guards and paired clock durations; the CPU validator rejects missing samples or mismatched records before reporting a ranking. Dense omission evidence is a separately bound prerequisite, not a property inferred by that record validator. The RapidFlow adapter additionally binds the actual stage audit and binaries; the synthetic failure example checks this interface rather than numerical accuracy.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

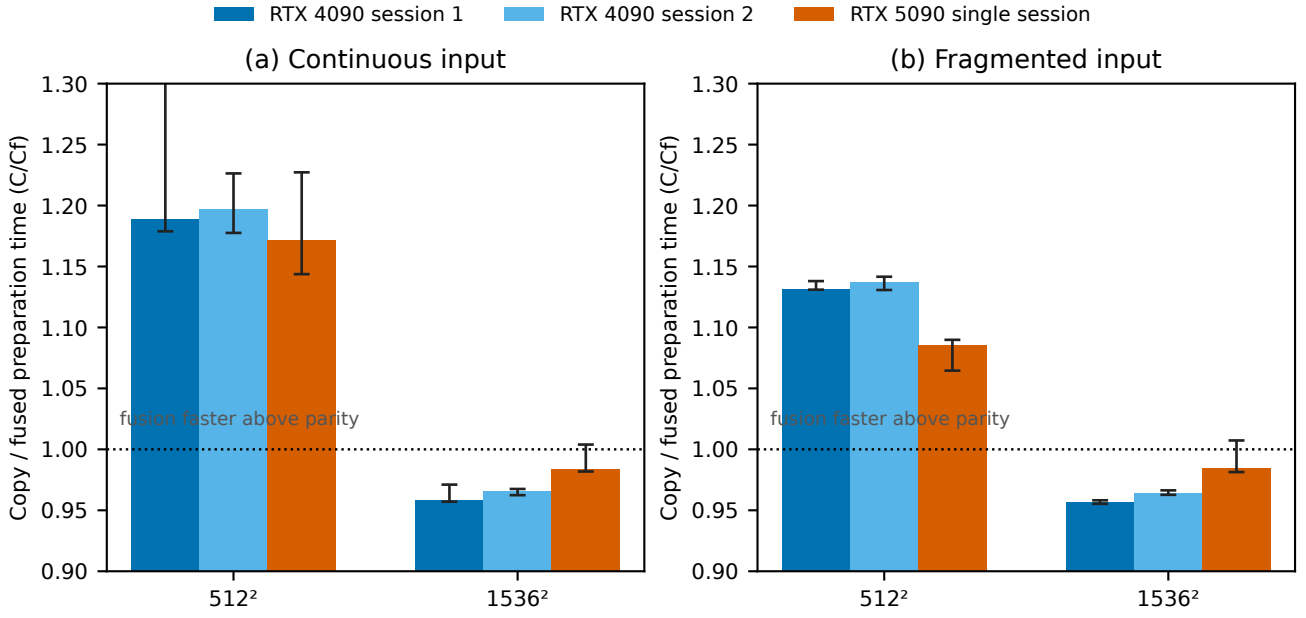


Figure 6. Copy versus fused preparation, C/Cf on CLOCK_MONOTONIC_RAW, for the continuous and fragmented inputs at both endpoint sizes. Values above the dotted parity line favour fusion. Bars are conditional 95 % intervals over three processes and five paired repeats; the two primary-platform sessions are plotted separately and not pooled. Fusion helps at 512^2 on both devices. At 1536^2 it is resolved as slower on the RTX 4090 and unresolved on the RTX 5090, where the interval crosses parity.

4.6 Eight-size scan: relative and absolute cost

The scan adds 640^2, 768^2, 896^2, 1152^2 and 1280^2 to the three endpoint sizes, with three serial processes and five paired repeats for F/C. These intermediate points have saved-output and support guards but no matched dense stage audit; their timings are therefore conditional implementation comparisons. It runs twice on the primary platform and once on the second, giving three separately recorded sessions of 576 warmup/ordinary trajectories each. All pass saved common-field, step and dry-strip guards, and C's selected support remains identical across sizes for a given input. Both clocks retain every direction.

Tables S35–S37 report all 48 sampled intervals, visualized in Fig. 7. Every one lies wholly below one: filtering is faster for both inputs, at all eight sizes, on both devices and in both primary-platform sessions. The ratio generally increases with stored size, from 0.703–0.852 at 512^2 to 0.949–0.964 at 1536^2, so the compaction deficit shrinks as dry storage grows. That increase is not monotonic at every step. On the RTX 5090 the continuous input gives 0.794 at 768^2 and 0.757 at 896^2, and the fragmented input 0.793 and 0.752; the primary platform shows smaller non-monotonic steps of the same kind. We therefore describe a narrowing trend with sampled exceptions, and we do not fit a slope law to these points.

The two primary-platform sessions agree closely, with differences of at most about 0.04 in the ratio and no direction change, which supports stability of the pattern across sessions within one day. The RTX 5090 shows systematically lower ratios at the

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

small and intermediate sizes, meaning filtering wins by a wider margin there, but converges to nearly the same ratio as the RTX
410 4090 at 1536^2. Because the two rented hosts differ in processor and driver, we report this as a difference between complete platforms, not as an effect of GPU architecture.

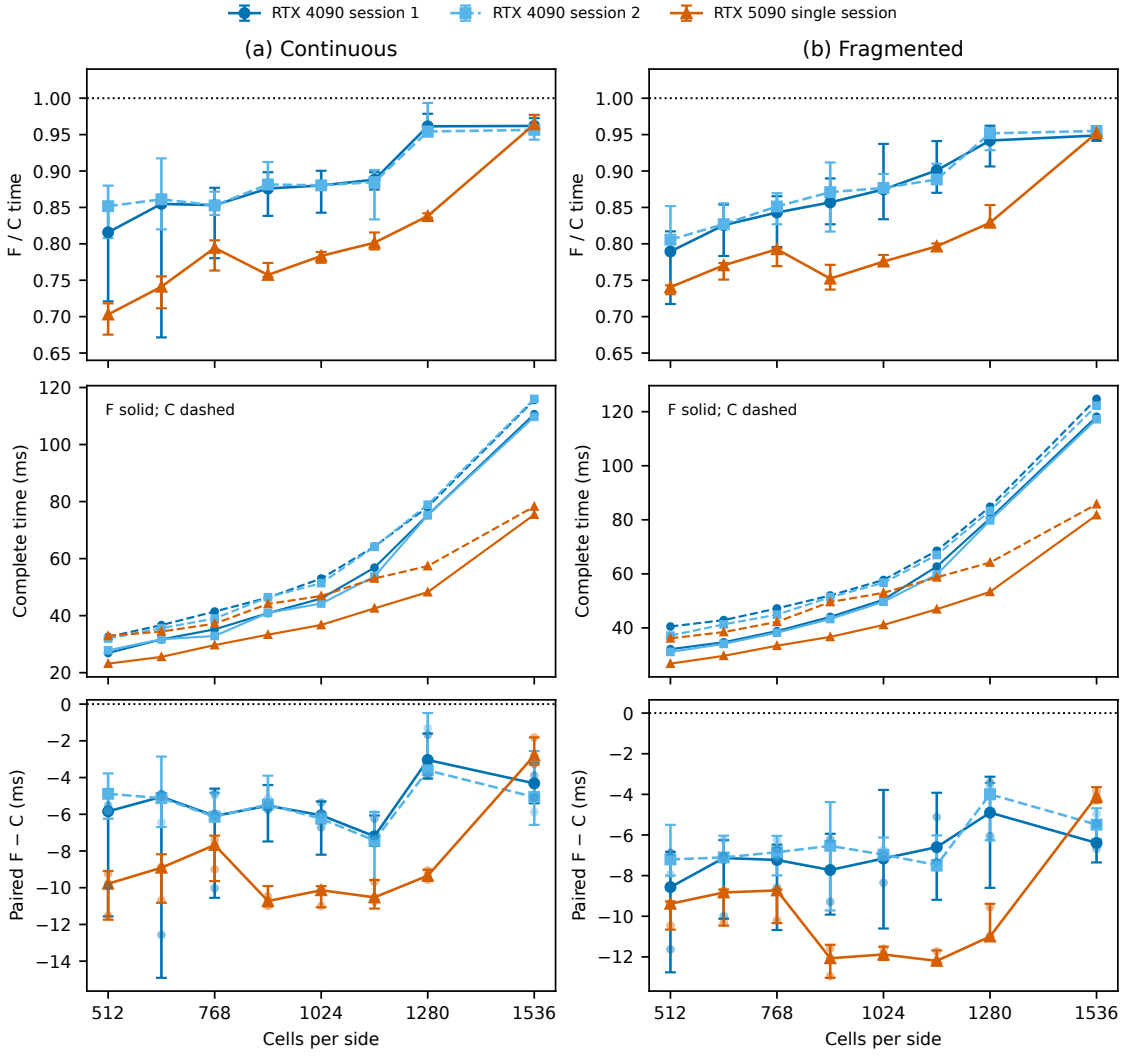


Figure 7. Conditional eight-size F/C scan for the continuous and fragmented inputs on both platforms; intermediate sizes and the second platform do not have matched dense stage audits. The figure shows the two primary-platform sessions separately. Bars are conditional 95 % intervals over three processes and five paired repeats. The top row shows paired ratios and parity; the middle row shows absolute F (solid) and C (dashed) median-of-process-medians times. The bottom row shows paired F-C intervals and all three process medians. Complete trajectories keep all continuing windows together; differences of absolute medians need not equal medians of paired differences. Table S44 reports the numerical values.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Absolute times distinguish a narrowing relative advantage from a vanishing cost saving. In primary session 1, the continuous input's F/C absolute times rise from 26.9/32.5 ms at 512^2 to 110.6/115.6 ms at 1536^2, while paired F-C changes from -5.8 to -4.3 ms. Fragmented times rise from 32.1/40.5 to 118.1/124.7 ms, with paired F-C changing from -8.6 to -6.4 ms. Thus total duration grows much more than the absolute gap shrinks. Shared full-domain work can dilute a relative difference; these measurements do not isolate that contribution from selection, preparation and execution interactions. For the flat scan K is 185/196 and cumulative A is 40,108/115,255 for continuous/fragmented inputs. With N increasing from 262,144 to 2,359,296, 256A/(2KN) falls from 10.59/28.71 % to 1.18/3.19 %, while A stays fixed. A capacity-equality heuristic and its limitations remain in S29.

4.7 A second compaction implementation

To test sensitivity to the selection implementation, a stable two-CTA block scan replaces the original two CUB selections while retaining count readback. The resulting timing comparison is conditional beyond the representative dense audit coverage, while its audited endpoint trajectories pass the listed stage checks. preparation and numerical kernels. Sparse-path stage audits pass at both endpoints and geometries; all 25 shared device functions retain their instruction words. In 360 matched trajectories on the primary platform, original/candidate copy ratios are 1.038 and 1.019 at 512^2, but 0.954 and 0.953 at 1536^2. Thus the alternative improves the small case and worsens the large case. Filtering remains faster than both copy and fused alternatives in the audited representative endpoints (Table S40). The result survives this particular selection change, but does not establish an optimal compacted implementation or a disadvantage of compaction generally.

5 Numerical validity and policy connection

5.1 Default front bias and its causes

Ritter depth error decreases with refinement, but the thresholded wet region moves farther from the analytical wet set (Table 10 and Fig. 8). At 1 mm, the discrepancy rises from 4.08 % to 6.70 % while depth E_1 falls by roughly a factor of three. A small integrated field error therefore does not ensure that cells cross a wet/dry reporting threshold at the correct locations. Quarter-period Thacker likewise has a submillimetre mean depth error together with a larger discrepancy at 1 mm than at 10 mm. These diagnostics constrain different features of the numerical solution.

Extending the grid check to 512 and 1024 cells increases strict 1 mm discrepancy to 7.22 % and 7.35 %. At 1024 cells the rightmost wet-cell edge is 5.57 m (57 cells) behind the exact cell-average wet-set edge. Reducing the timestep cap sufficiently to produce 1920 steps leaves these wet-set discrepancies unchanged. Thus the trend is not explained by a one-cell threshold rounding effect or removed by the tested smaller timesteps. The local profiles and position diagnostics are documented in Sect. S17.

An isolated Froude-cap intervention identifies one contributor (Sect. S19). The exact Ritter fan has $\mathrm{Fr} = 2(\sqrt{h_0/h} - 1)$, reaching 10 at $h = 0.05556$ m and $x = 69.93$ m at 3 s. Thus the default cap modifies more than an extremely thin wet layer. At

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

1024 cells, raising the cap to 100 or disabling it reduces depth L1 from 0.003059 to 0.001578 m and wet-edge lag from 5.57 to 3.81 m in the CFL-adaptive sequence. The two variants agree exactly and neither clips. A separate fixed-sequence comparison uses 1920 actual timesteps for both policies: its lag changes from 5.57 to 3.906 m (Table 11). The cap therefore explains part of the error, including under fixed timesteps, but a substantial residual front bias remains. Factor tests below separate momentum cleanup, transport gates and reconstruction fallback; flux diffusion and their remaining interactions are not fully isolated.

A factor check distinguishes three rules after cap removal while holding all 1920 timesteps and both reporting cutoffs fixed. At 1024 cells, retaining momentum below 1 mm reduces the edge lag from 3.91 to 1.27 m and depth L1 from 0.001648 to 0.000986 m. Lowering only the transport gates to 10^{-7} m leaves the lag unchanged, while lowering the near-dry reconstruction fallback threshold from 1 cm to 10^{-7} m increases it to 4.00 m. Combining momentum retention with the lower transport gate gives 1.37 m; halving the timestep again leaves the respective 1.27/1.37 m edges unchanged. Thus post-update dry-state momentum removal explains much of the residual observed after cap removal in this test. Remaining bias is still substantial, and these variants do not establish complete numerical convergence (Sect. S22).

The default policy is retained as the fixed, versioned operator for execution comparisons and regression checks. It is not recommended here for general inundation extent, front arrival or thin-layer momentum prediction. Where the Froude cap or dry-momentum reset remains active, refining the mesh alone need not recover the unmodified shallow-water solution. Lake/channel checks support their stated equilibrium and smooth-flow regimes, not a general accuracy envelope. The improved-front configuration is diagnostic: recommending it as a new default requires satisfactory front refinement and broader accuracy validation even after the bounded non-flat omission repair in S33. The policy-specific performance connection below supports the execution finding without making that numerical recommendation.

Table 10. Default Ritter with CFL-adaptive steps and a 0.05 s step cap (distinct from the fixed 1920-step sequence in Table 11): depth error, strict wet-set discrepancy and edge error for T2. The original T1 comparisons at 64–256 cells give the same states. The extended fine-grid diagnostics use T2.

Ritter configuration	Depth E_1 (m)	D at 1 mm (%)	Wet-edge error (m)
Default / n=64	0.014350	4.082	-3.125
Default / n=128	0.008204	6.186	-4.688
Default / n=256	0.005060	6.701	-5.078
Default / n=512	0.003698	7.22	-5.469
Default / n=1024	0.003059	7.35	-5.566

The lake and smooth-channel tests support the balance and boundary treatment within their stated conditions. They do not establish accuracy at hydraulic jumps, long-time shoreline behaviour or a universal stability bound. In particular, the front results below cannot be inferred from equilibrium preservation or from agreement between execution paths.

Candidate Ritter depth L1 decreases from 0.007747 m at n=128 to 0.000978 m at n=1024, but the remaining wet-edge lag is 13 cells, failing the two-cell development criterion recorded before this follow-up run. This 2 dx tolerance is a grid-scale

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Table 11. Isolated front factors at n=1024, t=3 s, using the same 1920-step sequence. Depth error and edge placement assess different features.

Numerical policy	Depth E_1 (m)	Strict 1 mm edge error (m)
Default Fr10	0.003071	-5.566
Cap off	0.001648	-3.906
Cap off + retain thin-layer momentum	0.000986	-1.270
Cap off + lower transport gate	0.001648	-3.906
Cap off + lower reconstruction fallback	0.001670	-4.004
Cap off + momentum + lower transport gate	0.000985	-1.367

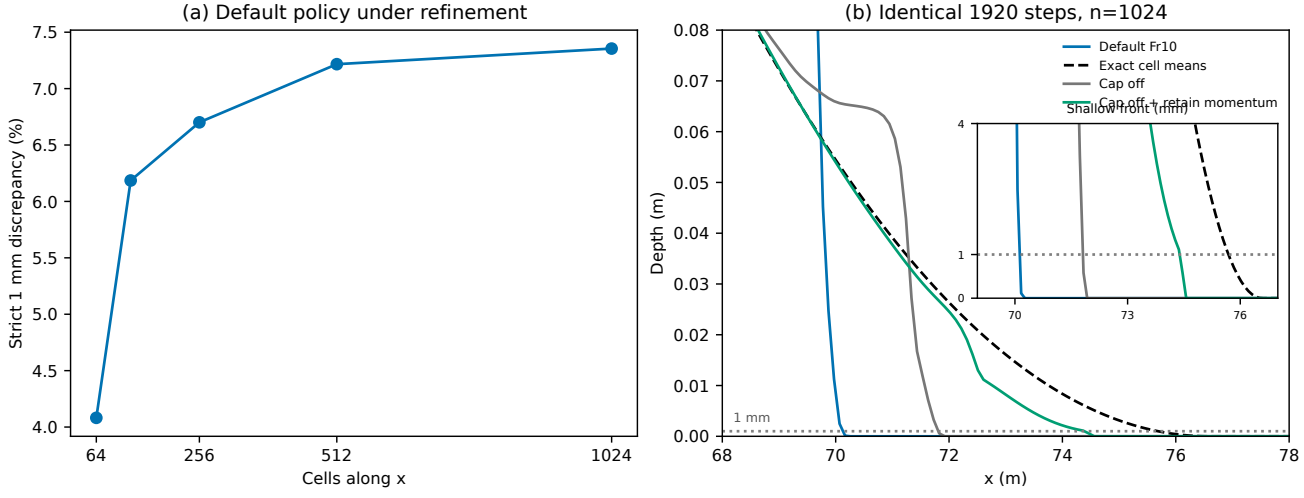


Figure 8. Ritter diagnostics at t=3 s: (a) default refinement from 64 to 1024 cells, with strict 1 mm wet-set discrepancy, and (b) near-front profiles at n=1024 with the same 1920-step sequence. Cap removal and thin-layer momentum retention reduce the lag but leave a bias. The inset resolves 0–4 mm depths and the dotted 1 mm threshold. Analytical references are cell averages; numerical policy and diagnostic threshold are separate.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

screen for front displacement, chosen to distinguish a few-cell location error from the observed multi-cell bias; it is not an application-calibrated or universal standard. The default remains unchanged. On the non-flat Thacker basin, both policies' errors decrease with refinement; after two periods at 256^2, default/candidate depth L1 is 0.000783/0.000800 m and strict 1 mm wet-set discrepancy is 4.91/4.75 %. The candidate improves front placement in these tests without uniformly improving all field errors.

5.2 Policy sensitivity on flat workloads

The Fr100 and cap-off/thin-momentum interventions preserve the observed filtering preference and its size trend on the two flat inputs (S22, Fig. 9). Each policy changes the operator and must be validated separately. Saved common fields, window steps and support agree within policy; S31 adds sparse stage checks for the thin-momentum endpoints. These checks cannot substitute for dense validation, as the non-flat failure in Sect. 4.2 demonstrates. The policy comparison is a bounded development sensitivity, not a numerical recommendation or a cross-platform performance guarantee.

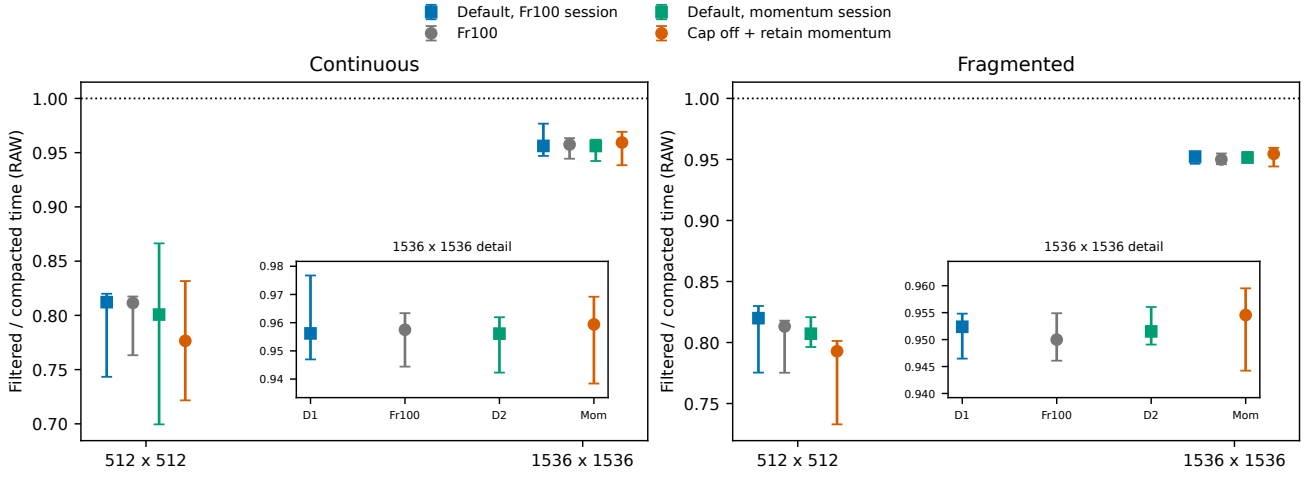


Figure 9. RAW-clock F/C ratios for default and candidate policies on two core geometries. Insets separate large-domain comparisons: D1/D2 are the Fr100/momentum-session defaults and Mom retains thin-layer momentum. Defaults are paired within each session; bars give conditional 95 % process/repeat intervals. Numerical policies are compared only within their stated audit scope.

5.3 Independent-reference limits

For circular expansion, the largest spatial-mean absolute errors E_1 across $64^2/256^2$ grids, both source-splitting configurations and saved times 0.25/0.5/0.75/1 s are 0.00829 m for depth and 0.02403 m^2 s^{-1} for either momentum component. These are maxima over comparisons, not spatial E_∞ norms. Wet-set discrepancy is at most 25 %. Paired withdrawal is more restrictive: at 256^2/T2 and 4.5 s, the default strict 1 mm discrepancy is 75 %, a symmetric difference of 1.0254 m^2 against 1.3672 m^2

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

reference wet area. At 10 mm the numerical set is empty while 0.4395 m^2 remains wet in the reference. A small full-domain depth error does not remove these local errors.

Figure 10 shows their spatial meaning during withdrawal and restart. At 4.5 s, the 0.5/1/2 mm numerical policies give 203.57/75/50 % discrepancy at the fixed 1 mm reporting cutoff. Missed area remains 0.683594 m^2, while excess area is 2.099609/0.341797/0 m^2. Choosing a threshold by the smallest observed discrepancy would change the numerical problem and is not an accuracy certification.

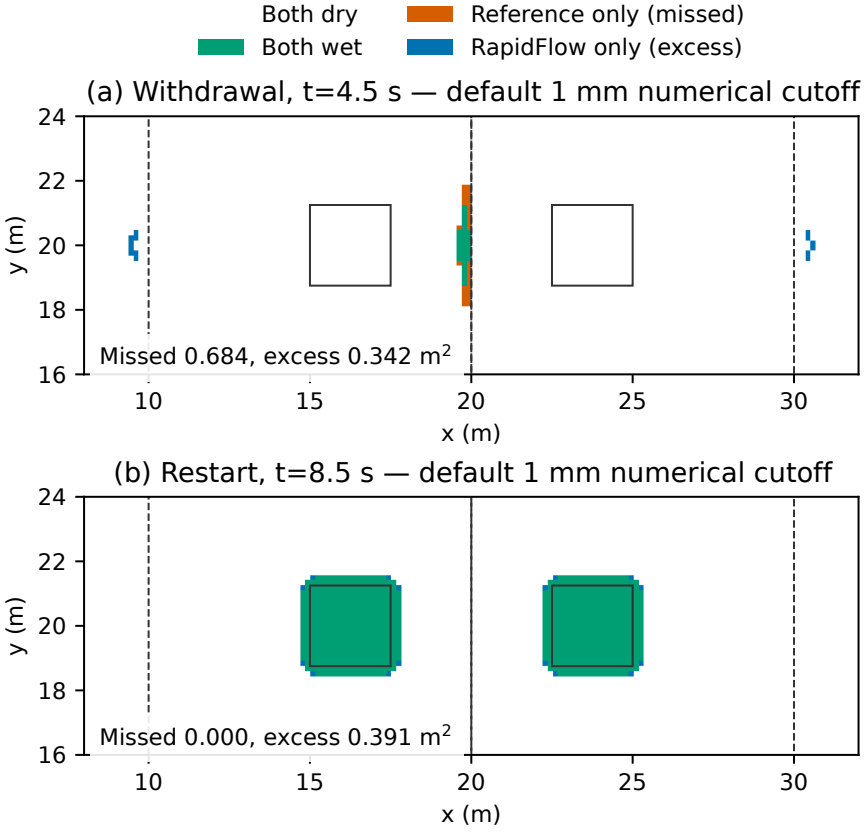


Figure 10. Default 1 mm numerical policy against the fixed ANUGA reference at the strict 1 mm reporting threshold: withdrawal at 4.5 s and restart at 8.5 s. Colours distinguish missed and excess water; area labels are full-domain values despite the dry-surrounding crop. Solid/dashed boundaries denote supply/withdrawal rectangles. The complete numerical-cutoff comparison and central-front enlargement are in Supplement Fig. 25.

After restriction to the comparison grids, the half-step and source momentum-clearance reference variants differ from the standard ANUGA reference by at most 4.55 % and 17.86 % in wet-set discrepancy, respectively. These are reference/reference comparisons, not changes in RapidFlow/reference error. At 4.5 s the 256-versus-512 reference-grid comparison gives 14.29 % wet-set discrepancy, but RapidFlow still differs by 70.83/75 % against those references. These checks expose reference

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

uncertainty without explaining away front bias. Detailed time series, numerical-cutoff failures and the diagnostic-only float64 source ledger control remain in S6, S10-S11, S17 and S26.

The real-terrain Merewether case is included only as an execution workload, with its fixed-site coverage and observational limits documented in S26.4; it does not validate event reconstruction or general numerical accuracy.

6 Discussion

6.1 Implications for sparse model execution

The protocol's strongest validation result is its rejection of the original candidate Thacker comparison despite F/C agreement. Its strongest successful application is the default non-flat case, which combines dense stage equality, unchanged local workload under extension and bounded analytical error. The flat campaign adds broader size/platform evidence with the weaker dense-validation coverage in Table 6. Its observed filtering preference is useful during implementation development, but does not complete the omission gate for every configuration. No observed ranking change is needed to make either a rejection or a conditional implementation comparison informative.

The restored-work intervention shows that omission saves time in C's context, but the retained implementation pays more elsewhere: R-F exceeds R-C at each endpoint. Figure 7 makes a complementary distinction. Relative F/C approaches one largely while both absolute times grow; the paired millisecond gap remains negative. Neither contrast decomposes the implementation into independent kernel costs, since redundant preservation and launch/memory interactions remain.

Preparation fusion offers a concrete implementation decision. It removes launches and helps at 512^2 on both platforms, yet is slower at 1536^2 in both recorded RTX 4090 sessions and unresolved on the RTX 5090. For the measured 4090 flat workload, retain copy preparation if C is required; independent-date confirmation is needed before treating this small effect as robust. The separate cell-indexed neighbourhood-expansion control (S32.2) gives about 2 % at 1536^2 and a slight penalty at 512^2. This is distinct from the block-scan variant of Sect. 4.7, whose large-size effect has the opposite sign.

What transfers is the experiment design and evidence record. The two-platform results show why launch counts and wet fractions alone cannot determine the choice. Numerical operators, host scheduling and device characteristics can all change the result. The tested variants retain count readback and full-domain maintenance (Table 4); they do not span device-resident scheduling or alternative state layouts. No hardware counter or controlled memory-system intervention establishes a cache cause for the preparation penalty or the size trend.

6.2 Accuracy, diagnostic precision and scope

Execution comparisons answer a development question: how much does it cost to advance this specified operator on this workload? A front bias shared by paths does not itself invalidate their cost contrast, but limits the trajectory's physical use. The successful Thacker application supplies an explicit analytical accuracy envelope for a moving-shoreline benchmark. The

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Fr100 and momentum interventions probe sensitivity to the default clipping rules; they do not supply a generally accurate replacement.

Execution consistency is not physical accuracy, and the numerical limits here are substantial. The default operator lags the analytical Ritter wet edge by 5.57 m, or 57 cells, at 1024 cells, and refining the grid increases the strict 1 mm wet-set discrepancy to 7.35 % rather than reducing it. Paired-source comparisons against the fixed external ANUGA references retain 28, 32 and 32 front-criterion failures against the standard, half-step and closure references, with 12 undefined wet sets in each case. The Merewether workload reaches only 4 of 5 observation sites on the coarse grid and 3 of 5 on the fine grid, with conditional stage errors of 0.174 and 0.152 m, and its vertical datum, fixed dry site and forcing uncertainties are unresolved. These limits are properties of the default operator and are unchanged by moving to new hardware. The ANUGA reference fields are reused fixed inputs with their original hashes, not newly generated references. A validated solver may change active support, timesteps, state storage and memory access, so its preference and savings must be measured anew. What transfers is Protocol 1: apply its guards and complete-cost measurements to an operator and benchmark with reference accuracy established for the intended use.

Tight volume balance does not certify front placement. Conversely, the accumulation-precision control (S11) shows that a failed diagnostic budget can arise without changed physical fields. Execution, diagnostic and reference accuracy checks therefore remain separate decisions.

The study uses two rented GPU platforms, source-free sparse groups for cost-model evaluation, and reused development inputs for the interventions. It does not validate an online selector, cross-device thresholds, an all-double solver or a calibrated flood event. The sessions were recorded on a single day per platform; the second primary-platform session began about an hour after the first on the same instance without a host reboot, so it tests within-day repeatability rather than cross-date or post-reboot robustness. We report the two sessions separately and do not pool them to claim a larger independent sample. No hardware counters were available on either platform, so every memory-system statement remains a hypothesis. The flat dry-extension experiments cover 512^2–1536^2 cells (0.26–2.36 million) and 64–192 m square domains. The non-flat Thacker application covers 256^2–768^2 cells (0.066–0.590 million) and 40–120 m square domains at fixed 0.15625 m spacing; kilometre-scale applications motivating sparse scheduling may occupy a different memory regime. Future claims in those directions require their own independent evidence rather than retuning this observed holdout or excluding unfavorable comparisons.

7 Conclusions

Protocol 1 combines fixed-spacing dry-domain extension, checks of local discrete evolution and absolute selected work, complete advancement timing and controlled restoration of omitted execution. It distinguishes a change in stored extent from a change in the local problem, and rejects comparisons that fail the execution guards.

For the tested RapidFlow implementations, the conditional timing matrix finds filtering faster across 512^2–1536^2 cells on both Linux platforms. The F/C ratio increases overall from about 0.70–0.85 to 0.95–0.96. Restoring omitted work in the compacted context reveals a growing saving, but that saving does not make C faster than F. Preparation fusion helps at 512^2 on

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

both platforms; at 1536^2 it is slower on the RTX 4090 and unresolved on the RTX 5090. Within the measured platform, workload and session conditions, these results support retaining filtering in this range and, when using C on the primary platform, retaining copy preparation at the large endpoint under the measured session conditions; independent-date confirmation is still required for that small effect. The stable-scan alternative does not alter the first decision. All tested compacted variants retain count readback and full-domain maintenance, so these are decisions about specific implementations.

Validation determines the scope of those decisions. Default flat representative trajectories and preparation controls pass sparse-path stage audits; the added review audit supplies matched dense coverage for the listed core endpoints and policies, while S31 preserves the historical stage records for its thin-momentum candidate. The original candidate's non-flat Thacker case fails dense/block agreement despite F/C agreement. A selection-only control repairs this omission failure without changing its dense trajectory; the original failure remains excluded from successful applications. Default and candidate Ritter front errors also prevent a general inundation-accuracy claim. The default non-flat Thacker application now passes its declared full protocol at three stored sizes with bounded analytical error, and supports filtering plus fusion when compaction is required (Sect. 4.1).

The reusable result is the comparison protocol and its acceptance logic. It is not a general wet-fraction scaling law: the principal flat experiment holds absolute selected support fixed while dry storage changes. Quantitative preferences require measurements for the intended operator, workload and platform; these same-day sessions establish neither cross-date robustness nor a cache mechanism.

Code and data availability

The RapidFlow v0.1 research package includes the solver, input-preparation and analysis scripts, physical inputs, fixed ANUGA reference fields, and RapidFlow states and timing samples. Baseline results use the same native numerical implementation; isolated cell-selection, numerical-cutoff and diagnostic-precision builds are archived separately with their exact changes. The ANUGA fields are fixed independent references. The Merewether inputs are distributed by Geoscience Australia (n.d.), with the original WRL terrain notice retained in the package.

Original RapidFlow software is licensed under the GNU General Public License version 3 only (GPL-3.0-only). Original experimental results, tables and figures are licensed under Creative Commons Attribution 4.0 International (CC BY 4.0). Third-party materials retain their own licences.

The exact-version inputs and experimental evidence are archived at https://doi.org/10.5281/zenodo.23477888 (Wang et al., 2026a), and the research software is archived at https://doi.org/10.5281/zenodo.23477886 (Wang et al., 2026b). Both records identify archive version 0.1.0-paper-20261019. The data record includes a restoration script, file inventories and checksums for the 57 transfer parts. Historical raw snapshots retain their original paths and are not additional independent experiments. Stage arrays are included where retained; hash-only receipts do not reconstruct unretained fields. The deposit selection report distinguishes current evidence from omitted historical outputs and compiled binary bundles. The included manuscript snapshot predates archive publication; this submission adds the archive citations without changing the numerical evidence.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Author contributions

SW led the study, designed the research, developed the code and performed the experiments. QZ and YZ contributed to the experiments, analysis and manuscript writing.

Competing interests

The authors declare that they have no conflict of interest.

Acknowledgements

The Merewether terrain was provided by the Water Research Laboratory, Manly, New South Wales, through the public ANUGA benchmark.

Financial support

This research received no external funding. The authors conducted the work independently in their own time.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

References

ANUGA developers: ANUGA 3.3.10, Python Package Index [code], https://pypi.org/project/anuga/3.3.10/ (last access: 12 September 2026), n.d.

Audusse, E., Bouchut, F., Bristeau, M.-O., Klein, R., and Perthame, B.: A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows, SIAM Journal on Scientific Computing, 25, 2050–2065, https://doi.org/10.1137/S1064827503431090, 2004.

Aureli, F., Prost, F., Vacondio, R., Dazzi, S., and Ferrari, A.: A GPU-Accelerated Shallow-Water Scheme for Surface Runoff Simulations, Water, 12, 637, https://doi.org/10.3390/w12030637, 2020.

Caviedes-Voullième, D., Morales-Hernández, M., Norman, M. R., and Özgen-Xian, I.: SERGHEI (SERGHEI-SWE) v1.0: a performance-portable high-performance parallel-computing shallow-water solver for hydrology and environmental hydraulics, Geoscientific Model Development, 16, 977–1008, https://doi.org/10.5194/gmd-16-977-2023, 2023.

Chowdhury, A. A., Kesserwani, G., Rougé, C., and Richmond, P.: GPU-parallelisation of Haar wavelet-based grid resolution adaptation for fast finite volume modelling: application to shallow water flows, Journal of Hydroinformatics, 25, 1210–1234, https://doi.org/10.2166/hydro.2023.154, 2023.

Chowdhury, A. A. and Kesserwani, G.: LISFLOOD-FP 8.2: GPU-accelerated multiwavelet discontinuous Galerkin solver with dynamic resolution adaptivity for rapid, multiscale flood simulation, Geoscientific Model Development, 18, 9827–9854, https://doi.org/10.5194/gmd-18-9827-2025, 2025.

Delestre, O., Lucas, C., Ksinant, P.-A., Darboux, F., Laguerre, C., Vo, T.-N.-T., James, F., and Cordier, S.: SWASHES: a compilation of shallow water analytic solutions for hydraulic and environmental studies, International Journal for Numerical Methods in Fluids, 72, 269–300, https://doi.org/10.1002/fld.3741, 2013.

Geoscience Australia: ANUGA Merewether case study: ARR Project 15 urban flood modelling, WRL terrain, model inputs and flood-mark data, Geoscience Australia [data set], https://github.com/GeoscienceAustralia/anuga_core/tree/c15aab56ed88ff3adbcd323e25d8ee0914d6c86b/validation_tests/case_studies/merewether (last access: 12 September 2026), n.d.

Guo, K., Guan, M., and Yu, D.: Urban surface water flood modelling – a comprehensive review of current models and future challenges, Hydrology and Earth System Sciences, 25, 2843–2860, https://doi.org/10.5194/hess-25-2843-2021, 2021.

HEMLab: HiPIMS-CUDA, GitHub [code], https://github.com/HEMLab/HiPIMS-CUDA (last access: 10 September 2026), n.d.a.

HEMLab: HiPIMS-API (pypims), GitHub [code], https://github.com/HEMLab/HiPIMS-API (last access: 10 September 2026), n.d.b.

Kesserwani, G. and Sharifian, M. K.: (Multi)wavelet-based Godunov-type simulators of flood inundation: Static versus dynamic adaptivity, Advances in Water Resources, 171, 104357, https://doi.org/10.1016/j.advwatres.2022.104357, 2023.

Klöwer, M., Düben, P. D., and Palmer, T. N.: Number Formats, Error Mitigation, and Scope for 16-Bit Arithmetics in Weather and Climate Modeling Analyzed With a Shallow Water Model, Journal of Advances in Modeling Earth Systems, 12, e2020MS002246, https://doi.org/10.1029/2020ms002246, 2020.

Lacasta, A., Morales-Hernández, M., Murillo, J., and García-Navarro, P.: GPU implementation of the 2D shallow water equations for the simulation of rainfall/runoff events, Environmental Earth Sciences, 74, 7295–7305, https://doi.org/10.1007/s12665-015-4215-z, 2015.

MacDonald, I., Baines, M. J., Nichols, N. K., and Samuels, P. G.: Analytic benchmark solutions for open-channel flows, Journal of Hydraulic Engineering, 123, 1041–1045, https://doi.org/10.1061/(ASCE)0733-9429(1997)123:11(1041), 1997.

https://doi.org/10.5194/egusphere-2026-6487
Preprint. Discussion started: 4 November 2026
© Author(s) 2026. CC BY 4.0 License.

Morales-Hernández, M., Sharif, M. B., Gangrade, S., Dullo, T. T., Kao, S.-C., Kalyanapu, A., Ghafoor, S. K., Evans, K. J., Madadi-Kandjani, E., and Hodges, B. R.: High-performance computing in water resources hydrodynamics, Journal of Hydroinformatics, 22, 1217–1235, https://doi.org/10.2166/hydro.2020.163, 2020.

Morales-Hernández, M., Sharif, M. B., Kalyanapu, A., Ghafoor, S. K., Dullo, T. T., Gangrade, S., Kao, S.-C., Norman, M. R., and Evans, K. J.: TRITON: A Multi-GPU open source 2D hydrodynamic flood model, Environmental Modelling & Software, 141, 105034, https://doi.org/10.1016/j.envsoft.2021.105034, 2021.

Murillo, J., García-Navarro, P., and Burguete, J.: Time step restrictions for well-balanced shallow water solutions in non-zero velocity steady states, International Journal for Numerical Methods in Fluids, 60, 1351–1377, https://doi.org/10.1002/fld.1939, 2009.

Rak, A., Mewis, P., and Guthe, S.: Accelerating flash flood simulations: An efficient GPU implementation for a slim shallow water solver, Environmental Modelling & Software, 177, 106030, https://doi.org/10.1016/j.envsoft.2024.106030, 2024.

Sætra, M. L.: Shallow Water Simulation on GPUs for Sparse Domains, in: Numerical Mathematics and Advanced Applications 2011, Springer, Berlin, Heidelberg, 673–680, https://doi.org/10.1007/978-3-642-33134-3_71, 2013.

Sharifian, M. K., Kesserwani, G., Chowdhury, A. A., Neal, J., and Bates, P.: LISFLOOD-FP 8.1: new GPU-accelerated solvers for faster fluvial/pluvial flood simulations, Geoscientific Model Development, 16, 2391–2413, https://doi.org/10.5194/gmd-16-2391-2023, 2023.

Shaw, J., Kesserwani, G., Neal, J., Bates, P., and Sharifian, M. K.: 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, 2021.

Strang, G.: On the construction and comparison of difference schemes, SIAM Journal on Numerical Analysis, 5, 506–517, https://doi.org/10.1137/0705041, 1968.

Thacker, W. C.: Some exact solutions to the nonlinear shallow-water wave equations, Journal of Fluid Mechanics, 107, 499–508, https://doi.org/10.1017/S0022112081001882, 1981.

Wang, S., Zhang, Q., and Zhang, Y.: RapidFlow v0.1: inputs and experimental evidence, 0.1.0-paper-20261019, Zenodo [data set], https://doi.org/10.5281/zenodo.23477888, 2026a.

Wang, S., Zhang, Q., and Zhang, Y.: RapidFlow v0.1: research software, 0.1.0-paper-20261019, Zenodo [code], https://doi.org/10.5281/zenodo.23477886, 2026b.

Xia, X., and Liang, Q.: A new efficient implicit scheme for discretising the stiff friction terms in the shallow water equations, Advances in Water Resources, 117, 87–97, https://doi.org/10.1016/j.advwatres.2018.05.004, 2018.

Xia, X. and Ming, X.: SynxFlow: A GPU-accelerated Python package for multi-hazard simulations, Journal of Open Source Software, 10, 7586, https://doi.org/10.21105/joss.07586, 2025.

Xia, X., Liang, Q., Ming, X., and Hou, J.: 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, 2017.

Xia, X., Liang, Q., and Ming, X.: A full-scale fluvial flood modelling framework based on a high-performance integrated hydrodynamic modelling system (HiPIMS), Advances in Water Resources, 132, 103392, https://doi.org/10.1016/j.advwatres.2019.103392, 2019.

Zhao, J., and Liang, Q.: Novel variable reconstruction and friction term discretisation schemes for hydrodynamic modelling of overland flow and surface water flooding, Advances in Water Resources, 163, 104187, https://doi.org/10.1016/j.advwatres.2022.104187, 2022.