Title:

More Data + Heterogeneous Compute = Fractured Workflows: Why mathematical and computational structure matter more than ever

Abstract:

The growth of data and computational power can easily be taken to imply that numerical algorithms and software architecture matter less in the quest to solve big problems. In practice, however, the opposite is often true: modern computing is increasingly heterogeneous, and scientific workflows are increasingly fractured across models, languages, hardware, schedulers, and data products. Expectations—for speed, fidelity, and reliability—seem to be the only things growing faster than data and compute.

Meeting these expectations calls for structure at two levels: in the mathematics of the algorithms and in the orchestration of the computations. This presentation is organized around two common procedures that are often adopted by default: operator splitting for simulating multiphysics systems and bulk-synchronous MPI for parallel execution. Both are powerful, and both can be made more so when algorithms and execution are co-designed for adaptivity, heterogeneous hardware, and irregular workloads.

We revisit operator splitting as an opportunity rather than a necessity. When designed with stability and error in mind, splitting becomes a structured way to compose physics and numerics. In this way, mathematics provides the way to build reliable composite integrators rather than a collection of ad hoc fractional steps.

We also revisit bulk-synchronous parallelism as a starting point rather than a destination. When designed with orchestration and irregularity in mind, execution becomes a structured way to compose kernels and data products. In this way, actors provide the means to build automated and robust computational workflows rather than a collection of ad hoc scripts and manual intervention.

Applications to cardiac electrophysiology and hydrology are used to illustrate how these ideas turn fractured workflows into routine simulation capability.