

Adaptive Computing - A Brief

Why execution should adapt to the workload

Traditional silicon runs a fixed pipeline regardless of the work. Most real workloads are memory-bandwidth-bound, sparse, and varied - so a fixed path wastes cycles, bandwidth and power.

The XSYDA thesis

- Adaptive: reshape routing, precision and sparsity per workload, deterministically.
- Memory-centric: predict, prefetch and minimize memory movement - not just compute.
- Runtime-optimized: a closed loop tunes behavior while the workload runs.
- Explainable: every decision carries its reason and confidence.
- Verifiable: pair fast simulation with gate-level RTL validation.

Traditional vs XMCP

Traditional: static execution paths, fixed memory pathways, generic optimization, opaque behavior, simulation only. XMCP: adaptive execution, memory-centric intelligence, runtime optimization, explainable execution, hybrid simulation + RTL validation.