
Deterministic Decision Engines

Anderson Defense — Technical Whitepaper

Abstract

Estimation, optimization and control are frequently prototyped in research environments whose timing behavior cannot be certified. This paper describes our practice of implementing decision engines as auditable production software with bounded latency and provable behavior.

1. Bounded Latency

Every solver path carries a worst-case execution bound. Iteration counts are capped, allocations are static, and degraded solutions are defined in advance so the engine always returns within its deadline.

2. Auditability

Inputs, intermediate residuals and selected actions are logged with sufficient fidelity to reconstruct any decision after the fact. Post-mission reconstruction is a first-class requirement, not instrumentation added later.

3. Verification Evidence

Requirements are traced to tests; tests produce artifacts under configuration control. Revalidation after change is measured in hours rather than program phases.

4. Conclusion

Determinism is what makes advanced mathematics deployable in contested environments. The mathematics is not the deliverable — the assured behavior is.