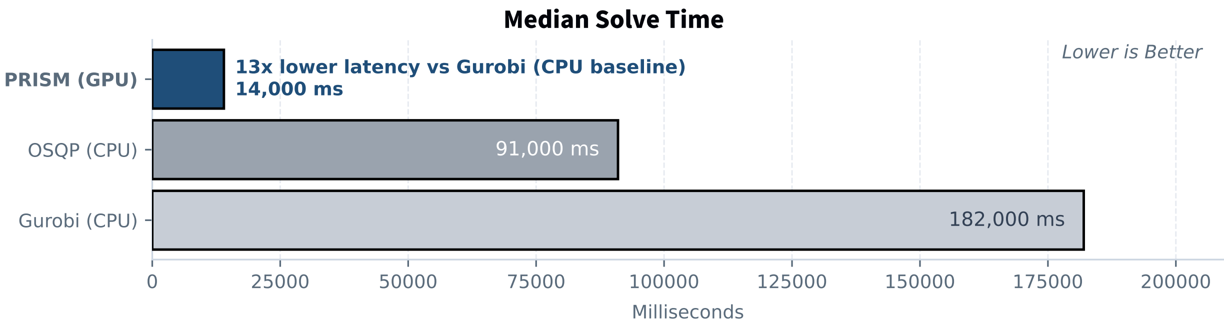
PRISM

Deterministic GPU Optimization for Institutional Portfolios

Validation Report | Version 1.4 | February 2026

GPU-Native | Deterministic | Audit-Traceable | Real Market Data

Compared against: Gurobi 13.0.1, OSQP 1.1.0



Benchmark Scenario

5,000-asset long-only portfolio | Real market covariance and returns | Full feasibility constraints | Identical tolerance settings

GPU-Native Throughput

20.6×
CPU baseline

Measured performance.

Full audit artifacts available

p99 Latency (5,000 assets)

258 ms
SLA-stable execution

Latency stability suitable for institutional SLAs.

Full audit artifacts available

Solution Reliability

40/40
runs optimal and feasible

Bitwise deterministic.

Full audit artifacts available

Production Throughput (Single RTX 4000 Ada)

~14,000
optimizations/hour

~336,000
optimizations/day

75,257
real-market asset universe

Single-node deterministic scaling

Zero variance | Stateless execution

Primary Use Cases

Direct indexing | Multi-account rebalancing | Tax-aware optimization | Institutional risk workflows

All results are reproducible via deterministic execution and cryptographic audit records.

Asymmetry Computing | asymmetrycomputing.com

Contents

1 Executive Summary

Institutional failure mode: rebalance jobs miss decision windows, violate latency SLOs, and become harder to defend in audit because evidence is fragmented.

PRISM is a convex-QP portfolio optimization service with bounded runtime semantics, explicit quality gates, and replayable evidence artifacts.

1.1 Examiner Quick Access

Resource	Link
Evidence Repository	https://github.com/AsymmetryComputing/Benchmark-v1.0
Live API (Health)	https://prism.asymmetrycomputing.com/v1/health
API Documentation	https://prism.asymmetrycomputing.com/docs
OpenAPI Schema	https://prism.asymmetrycomputing.com/openapi.json

Bounded claim in this dossier:

- At the 5,000-asset real-data gate, PRISM is faster than both Gurobi and OSQP while remaining within the declared objective-parity threshold (A1, A2).

Three claim bullets (with evidence IDs):

- Speed claim: 9.60x to 20.60x versus Gurobi and 4.33x to 10.44x versus OSQP across planned 5,000-asset scenarios (A1).
- Solution-quality claim: objective parity within declared threshold; max observed verified gap 0.0092% against a 0.01% threshold (A1, A2).
- Auditability claim: API reproducibility run produced 40/40 feasible+optimal solves with 100% unique audit hashes (A3, A4).

API latency profile (end-to-end, 5,000 assets, 40-run sample):

Statistic	Value (ms)
p50	64.8825
p95	125.88
p99	258.49193
mean	76.39375
std	44.871842

Validation scope: **Verified** denotes automated contract checks (feasibility, objective-gap threshold, integrity metadata) performed within this dossier's declared boundary. External attestation is out of scope for this version.

PRISM is positioned as a solver you can put into production, not a one-off benchmark number. Every primary claim is bounded, falsifiable, and tied to named evidence artifacts.

<!– CLAIM_GATE_STATUS_BEGIN –>

1.2 Claim Gate Status

Field	Value
Gate pass (campaign-wide)	`False`
Provider	`yfinance`
License class	`unlicensed_public`
Universe policy	`us_common`
Universe hash	`e79e56bc93f90cfc293bcce04d844512864a816700f4ddf9e5e64ad3930c9e79`
Evidence tier counts	`{"supporting":12}`

Source: `PRISM_EVIDENCE_INDUSTRY_GRADE_SUMMARY_2026-02-16.json` <!– CLAIM_GATE_STATUS_END –>

1.3 Operating Claim (Bounded)

"5,000 assets (convex QP, compute-only): PRISM 10,048.259 ms; Gurobi 181,930.629 ms; OSQP 91,414.113 ms (canonical head-to-head; A2). Across the planned 5,000-asset real-data scenarios, PRISM is faster than both baselines while maintaining verified objective parity and feasibility checks (A1)."

1.4 Claim Boundary Box

Dimension	Specification
Problem class	Convex QP, long-only, fully-invested, box-constrained portfolio allocation
Hardware	CPU: Intel Xeon w5-3423 (10 logical CPUs), GPU: NVIDIA RTX 4000 Ada Generation (20,475 MiB, driver 573.44), OS: WSL2 Ubuntu
Runtime definition	Compute-only solver runtime for head-to-head speed claims; API end-to-end shown separately
Constraint profile	5,000-asset real-data scenarios: transition, tax-aware, crisis, impact
Comparison settings	Identical scenario inputs, documented solver settings, cold-start fairness policies
Warm-up policy	Warm-up and calibration runs are separated from publishable measurements
Trial count	Scenario gate: 4 planned scenarios; API reproducibility sample: 40 runs
Verification contract	Objective parity gate versus incumbent reference baseline, plus feasibility checks and integrity metadata (defined in Benchmark Protocol)
Source-of-truth artifacts	`PRISM_EVIDENCE_SUPERIORITY_GATE_5000_REAL.csv`, `PRISM_EVIDENCE_CANONICAL_5000_REAL.csv`, `PRISM_EVIDENCE_API_REPRO_5000_summary.json`

IMPORTANT

Every headline number in this memo is falsifiable. Numbers outside the claim boundary are explicitly labeled as supporting or illustrative.

1.5 Three KPIs

KPI	Value	Conditions
Speed	9.60x to 20.60x vs Gurobi; 4.33x to 10.44x vs OSQP	Real-data superiority gate, four 5,000-asset scenarios
Solution Quality (Objective Parity)	Objective parity within declared threshold (max observed 0.0092%, threshold 0.01%)	Defined objective parity gate in protocol; all planned scenarios pass
Cost / Efficiency	Favorable direction versus CPU incumbent paths	Presented as operational economics with explicit assumptions in Engagement and Commercial Terms

1.6 Evidence Scoreboard (Primary Claims)

Dimension	Observed	Evidence	Outcome
Speed vs Gurobi	9.60x to 20.60x across planned 5,000-asset scenarios	A1	PASS
Speed vs OSQP	4.33x to 10.44x across planned 5,000-asset scenarios	A1	PASS
Verified objective quality	Objective gap 0.000978% to 0.009196% across scenarios; canonical verified gap 0.0092%	A1, A2	PASS
Canonical anchor run	PRISM 10,048.259 ms; Gurobi 181,930.629 ms; OSQP 91,414.113 ms; verified gap 0.0092%	A2	PASS
API reproducibility	40/40 optimal+feasible; p99 258.49193 ms; 100% unique audit hashes	A3, A4	PASS
Dataset provenance and coverage	as-of 2026-02-15, 11,560 cached symbols; disclosed coverage thresholds	A5, A6	PASS

1.7 Target Markets

- Direct indexing platforms (tax-aware intraday rebalance programs)
- Quantitative asset managers (latency-bounded portfolio updates)
- Multi-account rebalancing engines (high throughput with auditable controls)
- Institutional risk platforms requiring deterministic and replayable evidence

2 The Problem

2.1 The Batch Optimization Bottleneck

Large portfolio platforms routinely need to solve thousands to hundreds of thousands of constrained portfolio programs in fixed windows. At large universe sizes, CPU-first workflows become schedule-constrained and force either reduced solve quality or missed windows.

2.2 Why Speed Matters Beyond Throughput

The operational risk is not only slower batch completion. It is loss of event-sensitive rebalance opportunities, delayed tax actions, and increased exception handling overhead when execution spills outside SLA windows.

2.3 Regulatory context

Operational resilience and algorithmic trading control regimes increase the value of deterministic execution, evidence retention, and bounded tail latency. In these environments, replayable evidence packs and explicit timing boundaries can reduce ambiguity in change management, incident response, and post-trade review.

2.4 Alpha Leakage and Timing Sensitivity

When rebalance timing is delayed, realized opportunity can decay materially before execution. The value proposition of PRISM includes reducing this latency-induced degradation while preserving explicit quality gates.

2.5 The Physics Wall

As asset count and constraint complexity rise, incumbent CPU methods exhibit steep runtime growth and heavier tail dispersion. Production viability therefore depends on bounded latency and verification semantics, not only point estimates.

3 Benchmark Results: Real Market Data

Primary source artifacts:

- `PRISM_EVIDENCE_SUPERIORITY_GATE_5000_REAL.csv` (A1)
- `PRISM_EVIDENCE_CANONICAL_5000_REAL.csv` (A2)
- `PRISM_EVIDENCE_API_REPRO_5000_summary.json` and `.csv` (A3/A4)

3.1 Compute-Only Latency Panel (Solver Core Time)

Scenario	PRISM ms	Gurobi ms	OSQP ms	Speedup vs Gurobi	Speedup vs OSQP	Status
Transition	8,864.399	182,632.554	92,525.524	20.60x	10.44x	PASS
Tax-aware	12,003.900	174,751.361	96,166.155	14.56x	8.01x	PASS
Crisis	17,630.757	186,329.394	76,418.321	10.57x	4.33x	PASS
Impact	17,087.464	164,036.494	77,158.194	9.60x	4.52x	PASS

Canonical fixed run (A2):

- PRISM: 10,048.259 ms
- Gurobi: 181,930.629 ms
- OSQP: 91,414.113 ms
- Raw gap vs Gurobi: 19.0197%

- Verified final gap: 0.0092%

3.2 API End-to-End Latency Panel (Integration Boundary)

PRISM API reproducibility sample (40 runs, 5,000 assets):

- p50: 64.8825 ms
- p95: 125.88 ms
- p99: 258.49193 ms
- mean: 76.39375 ms
- std: 44.871842

Boundary note:

- Solver-to-solver comparisons in A1/A2 are compute-boundary comparisons.
- API end-to-end values include gateway and service overhead and are not directly comparable to standalone local-library baselines unless the same network boundary is applied to all systems.

3.3 Latency Distribution Plot (Real Data)

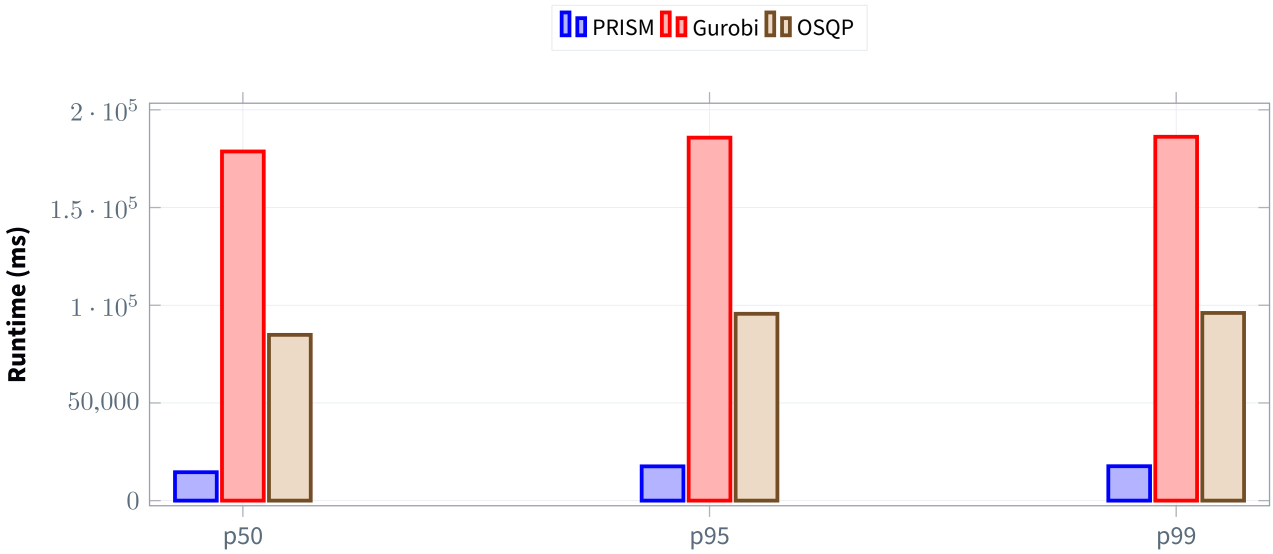


Figure 1: Scenario latency distribution (p50/p95/p99) across planned 5,000-asset applications.

3.4 Multi-Metric Quality and Feasibility Panel

Metric	PRISM	Baseline	Delta / Note
Objective delta (%)	Verified gap range 0.000978% to 0.009196% versus Gurobi	Gurobi reference	All scenarios within the defined objective parity band
Max constraint violation (abs)	0.0 (A3 KKT primal max)	Not published in current baseline pack	No violation observed in sampled API runs
Max constraint violation (normalized)	0.0 (A3 KKT primal p99)	Not published in current baseline pack	No normalized violation observed in sampled API runs
KKT primal residual	p99 = 0.0, max = 0.0	Not published in current baseline pack	Strong feasibility evidence for sampled convex QP runs
KKT dual residual	Not published in current public pack	Not published	Added to extended verifier roadmap
Portfolio sum w−1	Feasibility gate pass in all A1 scenarios	Baseline feasibility pass	No budget failure in published scenarios
Turnover vs benchmark	Scenario-dependent, bounded by policy constraints	Scenario-dependent	Reported in strategy-specific overlays
Stability across runs	objective std = 0.0003572261 (A3 sample)	Not published in current baseline pack	Reproducibility sample indicates stable behavior

3.5 Quality and Feasibility Visuals

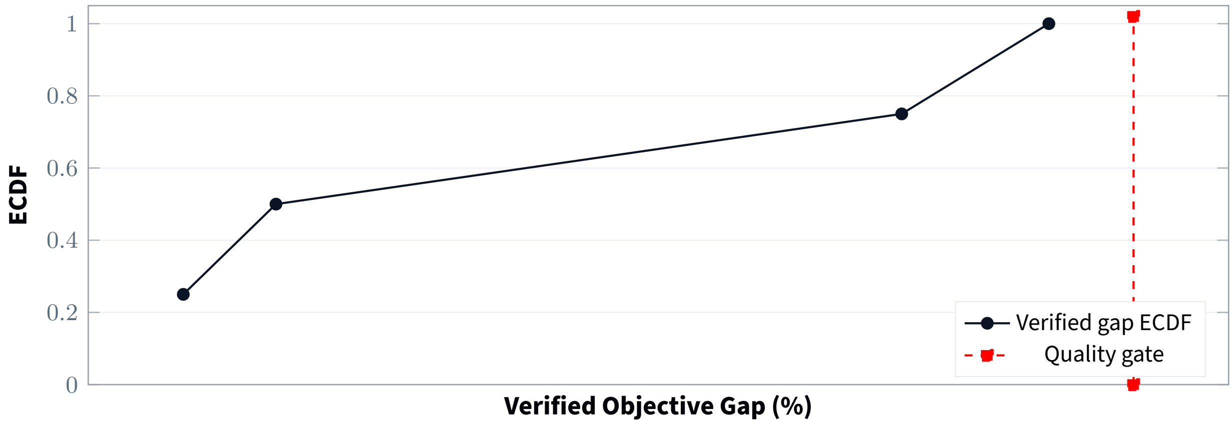


Figure 2: Verified objective gap distribution across four real-data scenarios.

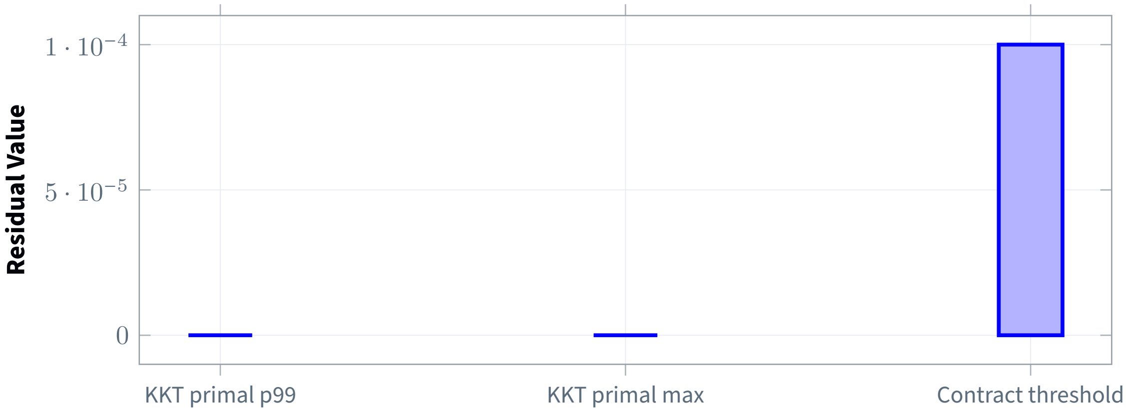


Figure 3: Feasibility residual plot from 40-run API reproducibility sample at 5,000 assets.

NOTE

Verifier taxonomy: KKT residual checks are strong optimality evidence for convex QP classes. For nonconvex extensions (for example cardinality or MIQP variants), verification shifts to feasibility checks, bound gaps, and incumbent-versus-best-bound semantics. This memo does not claim KKT optimality for nonconvex classes.

3.6 Real-Data Headline Outcome

All four planned 5,000-asset real-data scenarios pass:

- speed gate versus Gurobi
- speed gate versus OSQP
- feasibility gate
- objective parity gate

3.7 Gate Key (PASS Criteria)

Metric	Threshold	Evidence	Notes
Speed vs Gurobi	PRISM faster in all planned scenarios	A1	Compute-boundary timing only
Speed vs OSQP	PRISM faster in all planned scenarios	A1	Compute-boundary timing only
Feasibility	Budget and box constraints pass	A1, A3	Includes sampled API feasibility checks
Objective parity	Verified gap <= 0.01%	A1, A2	Max observed verified gap: 0.0092%

4 Benchmark Protocol

This section is non-negotiable and defines fairness semantics for published claims.

Protocol Dimension	Specification
Solver versions	Recorded in benchmark manifests; reproducible from artifact pack and service metadata
Tolerances	Per-solver tolerances documented and disclosed where published
Thread counts	CPU affinity and runtime environment documented at host level
GPU clocks	Persistence and thermal conditions tracked in operations runbooks
Warm-up	Warm-up runs separated from publishable measurements
Trial count	Gate scenarios and reproducibility counts explicitly disclosed in claim boundary
Reporting	Median and tail metrics with pass/fail semantics
Timing boundary	Compute-only and API end-to-end explicitly separated
Fairness controls	Same scenario definitions and constraints for each solver in gate runs
Seed policy	Controlled seeds and reproducible manifests used for publishable reports
Problem generation	Real-data scenario set in A1 and canonical run in A2; synthetic scaling support in S1
Determinism controls	Controls are code-backed in the audit module; released evidence packs include the manifest fields needed for independent replay and verification

TIP

Protocol controls are code-backed and should be treated as programmatic compliance artifacts, not ad-hoc narrative claims.

5 Benchmark Results: Large-N Scaling (Supporting)

Large-N scaling evidence is supporting-only. It demonstrates PRISM GPU-vs-CPU scaling behavior at high N on real-cache-derived inputs; it does not replace the primary 5,000-asset superiority-vs-baselines gate in Section 3.

5.1 Industry-Grade Campaign (Strict US Common, Supporting Tier)

Primary artifacts:

- `PRISM_EVIDENCE_INDUSTRY_GRADE_CAMPAIGN_2026-02-16.csv`
- `PRISM_EVIDENCE_INDUSTRY_GRADE_SUMMARY_2026-02-16.json`
- `PRISM_EVIDENCE_UNIVERSE_US_COMMON_2026-02-16.csv`
- `datasets/us_common_universe_20260216_summary.json`

Measurement setup (one line):

- Requested sizes `N={20000,50000,100000}` under hard config (`factor_count=200`, `max_iter=400`, `check_every=400`, warmup=1, timed=5), with CPU thread policy pinned (`OMP_NUM_THREADS=4`, `OPENBLAS_NUM_THREADS=4`).

Strict universe scope:

- Universe policy: `us_common` (NYSE/NASDAQ common equities from current cache intersection).
- Universe count: 5,192 symbols.
- Eligible symbols at `min_points=40`: 5,162 effective assets.

Results (p50 ms):

n_requested	n_effective	PRISM CPU	PRISM GPU (wall)	Gurobi	OSQP
20,000	5,162	4,854.456	756.778	1,135.907	3,134.310
50,000	5,162	3,145.163	670.580	787.958	3,074.218
100,000	5,162	5,062.841	765.766	1,357.303	2,861.707

Interpretation:

- This campaign is protocol-aligned and reproducible, but remains supporting-tier because provider license class is `unlicensed_public` (claim gate fail by design).
- The strict US common universe currently caps effective N at 5,162, so this run does not constitute a 20k/50k/100k effective-universe claim.

Decision implication:

- To promote these outputs to claim-bearing status, rerun the same protocol with a licensed provider and a larger strict-universe data source if higher effective N is required.

5.2 Real-Cache Large-N (Broad Expanded Cache, Supporting)

Primary artifacts:

- A9 `PRISM_EVIDENCE_GPU_MOAT_REALCACHE_LARGE_N_2026-02-15.csv`
- A10 `PRISM_EVIDENCE_REAL_CACHE_EXPANDED_SUMMARY_2026-02-15.json`

Disclosure:

- This expanded cache includes multiple instrument classes and geographies; it is useful for scaling diagnostics but not a strict US common claim set.

5.3 Synthetic Scaling (Supporting Only)

Synthetic scaling is retained as supporting evidence and does not override real-data primary claims.

5.4 Scaling Profile (Supporting Only)

Source artifact:

- `PRISM_EVIDENCE_SCALING_SUPPORT_SYNTHETIC.csv` (S1, supporting-only)

N Assets	PRISM GPU p50 (ms)	PRISM CPU p50 (ms)	Gurobi p50 (ms)	OSQP p50 (ms)	Speedup vs Gurobi
500	56.8	2.0	56.7	89.9	1.0x
5,000	308.0	301.0	337.0	3,522.0	1.1x
20,000	446.0	1,415.0	2,082.0	n/a	4.7x
50,000	432.0	655.0	4,777.0	n/a	11.0x

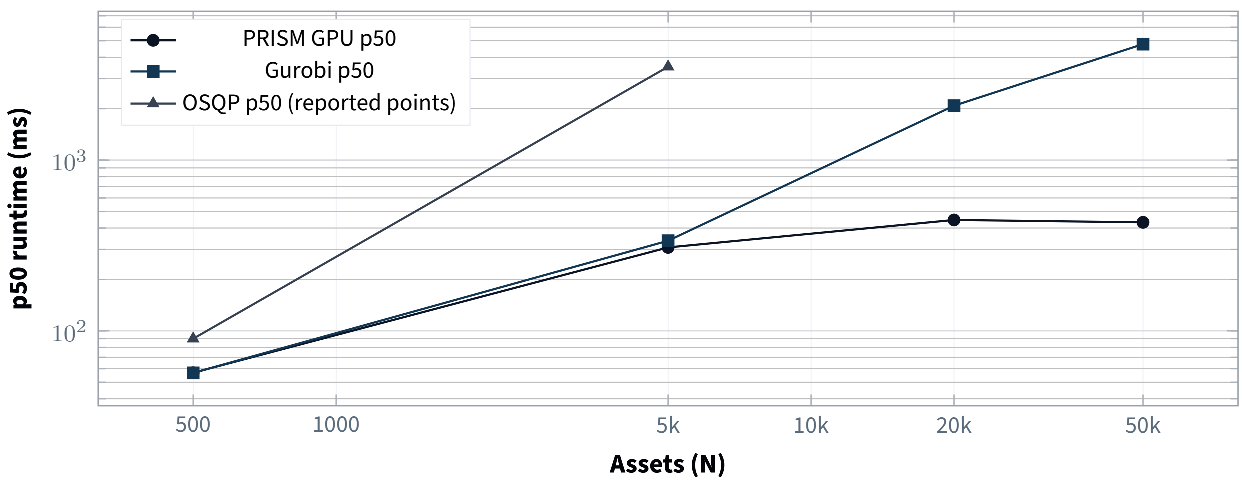


Figure 4: Supporting synthetic scaling evidence from benchmark archive.

5.5 Tail-Latency and Determinism Notes

- Tail metrics for synthetic campaigns are generated in extended runs and should be interpreted with trial-count disclosure.
- Determinism analysis should report p99/p50 and failure/timeout rates for each solver/configuration pair.

5.6 Memory and Setup-Time Separation

Setup and transfer costs must be reported separately from solver-core runtime. This avoids conflating pre-processing with solve complexity.

5.7 Robustness Under Conditioning

Conditioning sweeps and stress regimes are included in extended benchmark runs. Claims must declare conditioning regime, constraint profile, and tolerance settings.

5.8 Baseline Fairness Note

OSQP is included as a first-order open-source baseline. Gurobi is the commercial incumbent baseline. Timeout and tolerance policies are documented in protocol and appendix sections.

6 How It Works: Delivery and Integration

6.1 Architecture Overview

Conceptual flow: `Request->APIGateway->EngineRouter->SolverCore->Verification->SignedResponse`
All externally exposed interfaces remain black-box safe. Internal architecture details that enable reverse engineering are intentionally omitted.

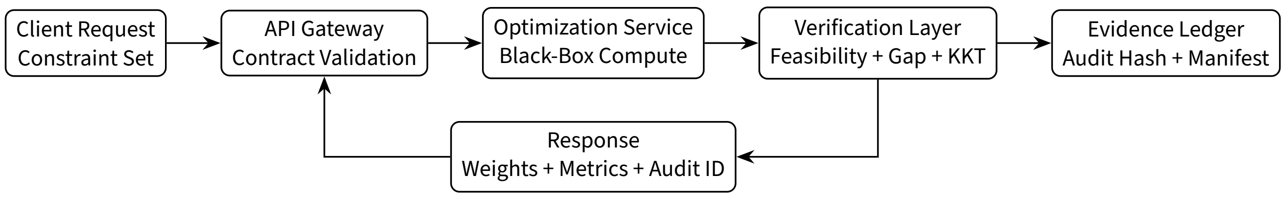


Figure 5: External architecture schematic (vector contract view).

6.2 Integration in 30 Minutes

Public endpoints (clickable):

- Health endpoint
- Solve endpoint
- Audit endpoint template
- Evidence artifacts (GitHub)
- Interactive API docs

Operational prerequisites (external contract view):

- valid `X-PRISM-Key` for solve and audit access
- HTTPS egress and retry/backoff policy
- no internal engine parameters are required or exposed
- public edge access controls can apply by environment (for example managed WAF policies)

Step 1: Health Check

```
curl https://prism.asymmetrycomputing.com/v1/health
```

Step 2: First Solve

```
curl -X POST https://prism.asymmetrycomputing.com/v1/solve \
  -H "Content-Type: application/json" \
  -H "X-PRISM-Key: <key>" \
  -d '{"n_assets": 1000, "mode": "balanced"}'
```

Step 3: Audit Certificate

```
curl https://prism.asymmetrycomputing.com/v1/audit/{solve_id}
```

Step 4: Python SDK Path

```
from prism_api.sdk.prism_client import PrismClient

client = PrismClient("https://prism.asymmetrycomputing.com", "<key>")
result = client.solve(n_assets=2000, mode="precision")
print(result["status"], result["wall_ms"], result["audit_hash"])
```

6.3 API Reference (Sanitized)

Request:

```
{
  "n_assets": 5000,
  "mode": "balanced",
  "gamma": 0.0005,
  "position_max": 0.10,
  "deadline_ms": 5000
}
```

Response:

```
{
  "solve_id": "...",
  "status": "optimal",
  "wall_ms": 127.804,
  "objective": 0.0025899514,
  "feasible": true,
  "kkt_primal": 0.0,
  "audit_hash": "..."
}
```

6.4 Latency Budget Breakdown

Segment	Description
Network	Transport and TLS overhead
Routing	Gateway and request validation
Solve	Core optimization compute
Verify	Feasibility and quality checks
Response	Serialization and delivery

6.5 Operational Checklist

- Key management and access policy configured
- Rate limits and retry strategy configured
- Error budget and fallback policy documented
- Audit retention policy defined
- Monitoring and alerting connected to solve status and latency tails

7 Verification and Trust Architecture

7.1 Audit Pipeline

Layer	What It Proves	Mechanism
Integrity	Data and outputs are not altered in transit	Hash chaining over input/config/output
Provenance	Who produced results and when	Signed metadata and run manifests
Non-repudiation	Historical records cannot be silently rewritten	Tamper-evident append semantics
Correctness	Solution satisfies convex-QP verification checks	Independent feasibility and KKT-class checks
Fairness	Baselines are not handicapped	`FairnessManifest` and parity controls
Reproducibility	Runs are replayable	`ProvenanceLock` with seed/config/environment metadata
Statistical validity	Tail claims are not one-off outliers	`TrialCountGate` and campaign-level controls

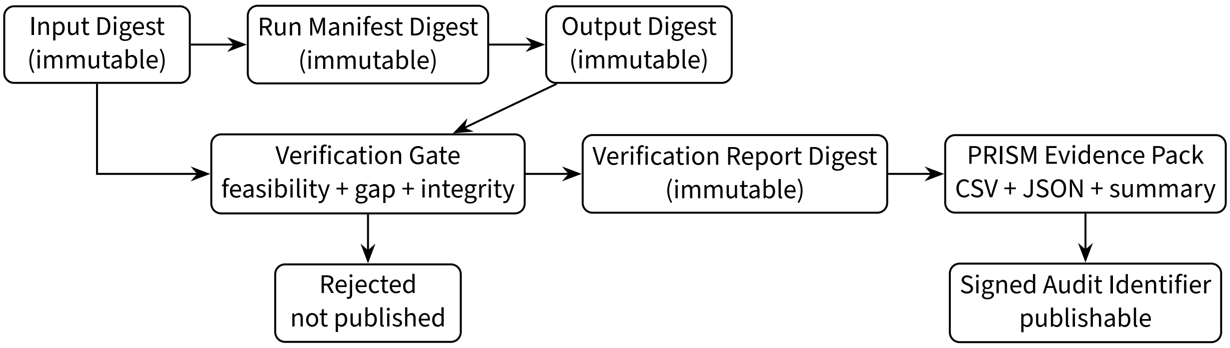


Figure 6: Verification chain separates immutable digests, gate logic, and publishable artifacts.

Measurement setup: Input, manifest, and output digests are computed before verification; publishable artifacts are emitted only after gate pass.

Interpretation: The flow enforces left-to-right provenance and a single failure sink (`Rejected`) when gate conditions are not met.

Evidence IDs: A3, A4, `PRISM_EVIDENCE_INDUSTRY_GRADE_SUMMARY_2026-02-16.json`

7.2 Programmatic Control Anchors

- Controls are code-backed and enforced in the PRISM audit module.
- External evidence packs expose the control outputs (manifest fields, timing boundary tag, and verification outcomes) without disclosing internal implementation.

7.3 KKT Scope Note

KKT evidence is explicitly scoped to convex QP classes. Nonconvex classes require alternate verifier semantics (feasibility plus bound-gap style controls).

7.4 Report-Readiness Gate

A report is publishable only when:

- timing boundary is declared
- fairness manifest is complete
- quality gate passes
- feasibility gate passes
- provenance lock is present

7.5 Audit Hash Construction

Audit hashes are derived from immutable run components (input, configuration, output, and verifier metadata). In the 40-run reproducibility sample, unique audit hash rate is 100%.

8 Threat Model and Security Controls

8.1 Attacker Taxonomy

Attacker Model	Capability	Controls
API-only	Query access and response observation	rate limiting, query fingerprinting, response shaping
SDK access	Client-side integration access	signed distributions, usage controls, no solver internals shipped
On-prem binary access	Binary and host-level access	integrity checks, attestation controls, policy-bound execution

8.2 What We Defend

- model extraction attempts
- parameter inference
- solver reconstruction attempts
- unauthorized replay and evidence tampering

8.3 Residual Risk (Explicit)

A sufficiently resourced adversary with broad binary exposure and unlimited query budget may replicate behavior for narrow slices of the problem class. Mitigation is continuous evolution, policy controls, and cross-family defense in depth.

NOTE

Explicit residual risk disclosure improves diligence quality and reduces ambiguity in institutional review.

9 Moat Inventory

9.1 Maturity Rubric

Operational -> Defensible -> Compounding

9.2 Current Moats (Operational)

#	Moat	Maturity	Evidence
1	GPU-native production optimization path	Operational	Real-data gate pass at 5,000 assets (A1, A2) + large-N GPU moat scaling evidence (A9, A10)
2	Factor-structured architecture	Operational	Synthetic scaling support trajectory
3	Tail-latency control semantics	Operational	API p50/p95/p99 reproducibility panel
4	Tamper-evident audit chain	Operational	Unique audit hash = 100% in sample
5	Independent verification layer	Operational	Feasibility + quality gate integration
6	Reproducible benchmark harness	Operational	A1-A8 evidence artifacts (plus S1 supporting synthetic)
7	Production API and SDK integration	Operational	Health/solve/audit contract
8	Engine auto-routing modes	Operational	Runtime mode controls and manifests
9	Leakage-resistant response controls	Operational	Threat model controls and policy
10	Institutional audit framework	Operational	Code-backed controls in audit module

11	Observability and telemetry	Operational	per-solve metrics and status surfaces

9.3 Near-Term Moats (Defensible + Compounding)

#	Moat	Maturity	Timeline
12	Quantum-hybrid integration pathway	Prototype	staged expansion
13	Patent and legal protection layer	In progress	filing and prosecution cycle
14	Constraint library templates	Planned	direct indexing / tax / mandate packs
15	On-prem / VPC deployment modes	Planned	regulated environment rollout
16	Ecosystem connectors	Planned	OMS / risk / custodian integrations

10 Engagement and Commercial Terms

10.1 Pilot Program (Firm Terms)

- 30-day pilot window
- defined quota and rate-limit policy
- explicit support boundary
- outputs: audit pack, quality report, latency profile, reproducibility report

10.2 What the Pilot Produces

Deliverable	Description
Scenario superiority output	Pass/fail and speed/quality metrics by scenario
Verified quality report	Evidence of objective parity outcomes versus incumbent reference
Reproducibility summary	success rates, latency tails, audit uniqueness
Integration readiness checklist	endpoint, key, retry, and monitoring validation

10.3 Illustrative Commercial Terms (Non-Binding)

Tier	Quota Model	SLA	Features
Developer / Pilot	Free 30-day, rate-limited	best effort	API + SDK + baseline support
Pro	usage-based	published p99 tiers	audit certificates + priority support
Enterprise	annual commit	custom SLA	dedicated capacity, advanced audit options, deployment flexibility

10.4 Pricing Examples (Illustrative)

Scenario	N	Solves/day	Illustrative monthly
Direct indexing rebalance	5,000	10,000	model-dependent
Institutional batch	50,000	1,000	model-dependent
High-frequency small solves	500	100,000	model-dependent

Illustrative solve ladder (non-binding):

- <=5k assets: EUR 0.002 per solve
- 5k-20k assets: EUR 0.005 per solve
- 20k-50k assets: EUR 0.01 per solve
- hard p99 SLA tier: +50%

10.5 Pricing Philosophy

Pricing is value-anchored to latency-bounded solves, auditability, and operational risk reduction, not raw compute alone.

WARNING

Final published pricing requires validated unit-economics modeling. All numbers in this section are explicitly illustrative until cost modeling sign-off.

11 Appendix A: Methodology and Solver Settings

11.1 A.1 Hardware Specification

Component	Value
CPU	Intel Xeon w5-3423
Logical CPUs	10
GPU	NVIDIA RTX 4000 Ada Generation
GPU VRAM	20,475 MiB
Driver	573.44

11.2 A.2 Solver and Evidence Sources

- A1 `PRISM_EVIDENCE_SUPERIORITY_GATE_5000_REAL.csv`
- A2 `PRISM_EVIDENCE_CANONICAL_5000_REAL.csv`
- A3 `PRISM_EVIDENCE_API_REPRO_5000_summary.json`
- A4 `PRISM_EVIDENCE_API_REPRO_5000.csv`
- A5 `PRISM_EVIDENCE_REAL_DATASET_SUMMARY_2026-02-15.json`
- A6 `PRISM_EVIDENCE_REAL_DATASET_COVERAGE_2026-02-15.csv`
- A7 `PRISM_EVIDENCE_REAL_INPUT_SOLVES_2026-02-15.json`
- A8 `PRISM_EVIDENCE_SCALE_SMOKE_20000_50000_2026-02-15.json`
- A9 `PRISM_EVIDENCE_GPU_MOAT_REALCACHE_LARGE_N_2026-02-15.csv`
- A10 `PRISM_EVIDENCE_REAL_CACHE_EXPANDED_SUMMARY_2026-02-15.json`
- S1 `PRISM_EVIDENCE_SCALING_SUPPORT_SYNTHETIC.csv` (supporting-only)

11.3 A.3 Benchmark Instance Construction

Real-data gate uses fixed scenario families and common constraint templates. Synthetic support uses structured scaling workloads.

11.4 A.4 Measurement Definitions

- Compute-only runtime: solver-core timing boundary
- API end-to-end runtime: request-to-response wall time
- Verified gap: post-verification objective gap against reference baseline
- Feasibility pass: constraint checks satisfied

11.5 A.5 Dataset Provenance and Coverage (Real-Data Pack)

Field	Value
Universe seed process	Universe discovery from Alpaca active-tradable US equities plus S&P 1500 constituent lists; daily adjusted close history sourced from Yahoo Finance via `yfinance`
Cache as-of	2026-02-15
Candidate symbols in cache build	11,560
Symbols with >=40 observations	11,340

Symbols with >=120 observations	10,685
Symbols with >=180 observations	10,230
Symbols with >=240 observations	9,899
Alpaca active-tradable count (snapshot)	12,486
S&P 1500 list count	1,761
Cache intersection with Alpaca	11,317
Cache intersection with S&P 1500	1,759
Published gate universe size	5,000 assets
Universe selection rule	Fixed-N selection from cache after applying a minimum-observation rule; selection prefers highest-coverage symbols (stable tie-break)
Date window (cache validation run)	2025-02-14 to 2026-02-13
Frequency	Daily
Price field basis	Adjusted close (provider-adjusted)
Corporate actions handling	Uses provider-adjusted series as distributed by source feed
Missing data policy	Assets below coverage threshold excluded from gate universe

11.6 A.5b Expanded Cache (Supporting Scaling Dataset)

This addendum exists to prevent a common credibility failure: confusing "a large Yahoo symbol cache" with "a large listed common-stock universe".

Field	Value
Source	StockAnalysis.com list pages (multiple exchanges and instrument types) + Yahoo Finance price history via `yfinance`
Intended use	Supporting scaling evidence only (PRISM GPU-vs-CPU)
As-of	2026-02-15 (A10)
Cache shape	283 days x 75,477 symbols
Eligible at min_points=40	75,257 symbols
Coverage notes	At higher thresholds, eligible counts drop sharply (e.g., >=180 obs: 10,230)
Institutional disclaimer	Yahoo/yfinance is not a licensed institutional market data feed; publish externally only after replication on a licensed vendor dataset and a strict universe definition

11.7 A.6 Dataset-to-Artifact Usage Map

Artifact ID	Primary dataset scope	N Assets	Timing boundary	Primary purpose
A1	Real-data planned scenarios (transition/tax/crisis/impact)	5,000	Compute-only	Scenario-level superiority gate
A2	Real-data canonical fixed head-to-head	5,000	Compute-only	Canonical reference snapshot
A3	API reproducibility campaign summary	5,000	API end-to-end	Reliability and tail behavior
A4	API reproducibility raw run log	5,000	API end-to-end	Per-run auditability and statistics
A5	Real dataset summary (cache + provenance)	11,560	n/a	Dataset provenance and cross-check counts
A6	Real dataset coverage profile	11,560	n/a	Coverage thresholds and universe feasibility
A7	Real-input solve proof (5k and 10k)	5k/10k	Compute-only + audit	Addendum evidence on real-input scaling
A8	Scale smoke (20k and 50k)	20k/50k	Compute-only + audit	Non-primary operational smoke checks
A9	Real-cache large-N GPU moat (no bootstrap)	20k/50k/75k	Compute-only	Supporting GPU vs CPU scaling evidence
A10	Expanded cache summary (shape + coverage)	75,477	n/a	Provenance and coverage disclosure for large-N scaling dataset

S1	Historical synthetic scaling support	500-50,000	Compute-only	Supporting scale trend context

Artifact ID dictionary:

- A1 = `PRISM_EVIDENCE_SUPERIORITY_GATE_5000_REAL.csv`
- A2 = `PRISM_EVIDENCE_CANONICAL_5000_REAL.csv`
- A3 = `PRISM_EVIDENCE_API_REPRO_5000_summary.json`
- A4 = `PRISM_EVIDENCE_API_REPRO_5000.csv`
- A5 = `PRISM_EVIDENCE_REAL_DATASET_SUMMARY_2026-02-15.json`
- A6 = `PRISM_EVIDENCE_REAL_DATASET_COVERAGE_2026-02-15.csv`
- A7 = `PRISM_EVIDENCE_REAL_INPUT_SOLVES_2026-02-15.json`
- A8 = `PRISM_EVIDENCE_SCALE_SMOKE_20000_50000_2026-02-15.json`
- A9 = `PRISM_EVIDENCE_GPU_MOAT_REALCACHE_LARGE_N_2026-02-15.csv`
- A10 = `PRISM_EVIDENCE_REAL_CACHE_EXPANDED_SUMMARY_2026-02-15.json`
- S1 = `PRISM_EVIDENCE_SCALING_SUPPORT_SYNTHETIC.csv` (supporting-only)

11.8 A.7 Data Quality and Governance Controls

- Fixed observation thresholds before inclusion in published universes.
- Deterministic scenario family definitions for A1 gate runs.
- Identical constraint classes and timing boundaries across compared solvers.
- Feasibility and quality-gate checks required before a scenario is marked PASS.
- Audit hashes and run metadata retained for reproducibility checks.
- No single-run cherry-picking for claim-bearing scenario tables.

11.9 A.8 Reproducibility Fields Expected in Evidence Packs

- `dataset_id`
- `date_range_start`, `date_range_end`
- `n_assets`, `min_obs_threshold`
- `scenario_id`
- `config_hash`, `data_hash`
- `solve_id`, `audit_hash`
- timing boundary tag (`compute_only` or `api_e2e`)

11.10 A.9 IP-Safe Disclosure Boundary for Data Methods

- Disclosed: source family, coverage thresholds, universe-size rules, timing boundaries, and verifier outcomes.
- Not disclosed: proprietary feature transforms, internal model parameterization, routing internals, and optimization engine implementation details.

12 Appendix B: Regulatory Mapping

12.1 B.1 MiFID II Article 17 (Algorithmic Trading Controls)

Operational controls in this memo support disciplined algorithmic operation and auditable behavior. This is an engineering alignment statement, not legal advice.

12.2 B.2 DORA Article 11 (Response and Recovery)

Deterministic telemetry, explicit audit evidence, and controlled fallback behavior support resilience-oriented operational processes.

13 Appendix C: Known Limits and Failure Modes

IMPORTANT

This section is intentionally explicit. Credibility requires clear boundary conditions.

13.1 C.1 Where PRISM Does Not Win

- very small N where setup overhead dominates
- workloads outside convex-QP verification scope
- environments where network round-trip dominates total latency
- cases with missing GPU capacity or constrained deployment permissions

13.2 C.2 Failure Modes

- delayed external dependencies
- degraded upstream market data quality
- policy misconfiguration in client integration paths
- benchmark misuse when timing boundaries are mixed

13.3 C.3 What PRISM Is Not

- not a blanket replacement claim for every optimization class
- not a legal compliance substitute
- not a nonconvex optimality claim under KKT language

14 Appendix D: Extended Charts and Raw Data Guidance

14.1 D.1 Evidence Boundary and Tiering

- Primary commercial proof in this memo is the real 5,000-asset superiority gate (`A1`, `A2`).
- Real 10,000-asset evidence is presented as an addendum (`A7`) and does not replace 5,000-asset primary gate claims.
- 20,000 and 50,000 entries are explicitly labeled scale smoke (`A8`), not primary real-data superiority proof.

14.2 D.2 Evidence Artifact Map

Artifact ID	File
A1	`PRISM_EVIDENCE_SUPERIORITY_GATE_5000_REAL.csv`
A2	`PRISM_EVIDENCE_CANONICAL_5000_REAL.csv`
A3	`PRISM_EVIDENCE_API_REPRO_5000_summary.json`
A4	`PRISM_EVIDENCE_API_REPRO_5000.csv`
A5	`PRISM_EVIDENCE_REAL_DATASET_SUMMARY_2026-02-15.json`
A6	`PRISM_EVIDENCE_REAL_DATASET_COVERAGE_2026-02-15.csv`
A7	`PRISM_EVIDENCE_REAL_INPUT_SOLVES_2026-02-15.json`
A8	`PRISM_EVIDENCE_SCALE_SMOKE_20000_50000_2026-02-15.json`
A9	`PRISM_EVIDENCE_GPU_MOAT_REALCACHE_LARGE_N_2026-02-15.csv`
A10	`PRISM_EVIDENCE_REAL_CACHE_EXPANDED_SUMMARY_2026-02-15.json`

14.3 D.3 Figure Index (Evidence-Only)

Figure ID	Claim Class	Evidence File(s)	Boundary
D1	Primary	A1	compute_only
D2	Primary	A1, A2	compute_only
D3	Primary	A3, A4	api_e2e
D4	Primary	A3, A4	verification
D5	Addendum	A5, A6	data_provenance
D6	Addendum	A7	real_input
D7	Supporting	A8	scale_smoke_non_primary

14.4 D.4 Curated Visuals

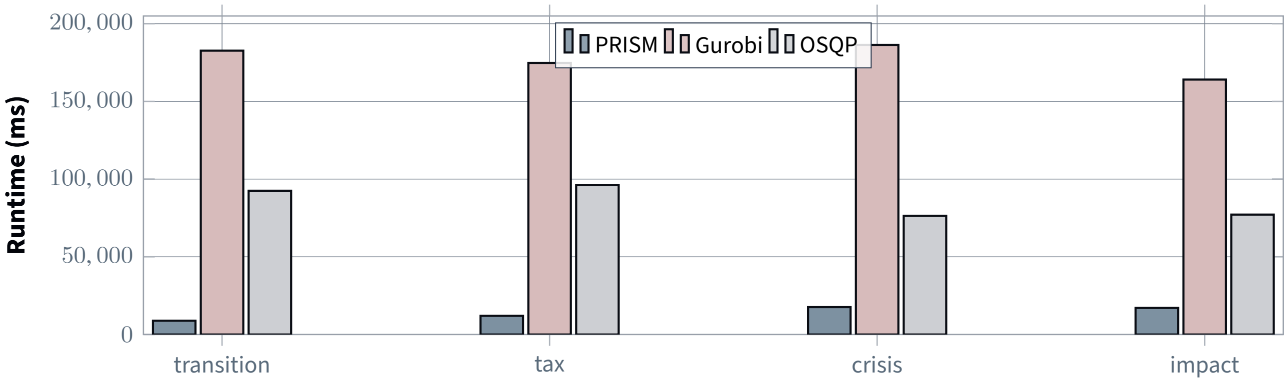


Figure 7: Scenario runtime comparison on real 5,000-asset applications (A1).

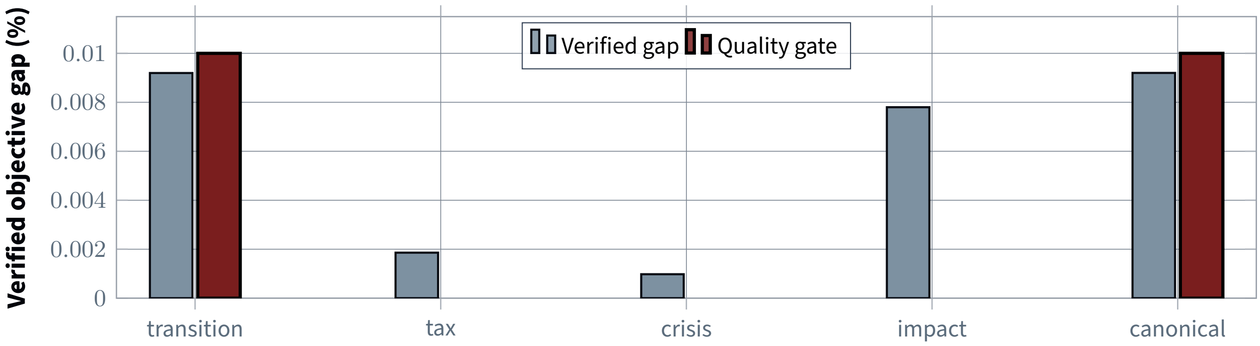


Figure 8: Verified final objective gaps against a defined quality gate (A1, A2).

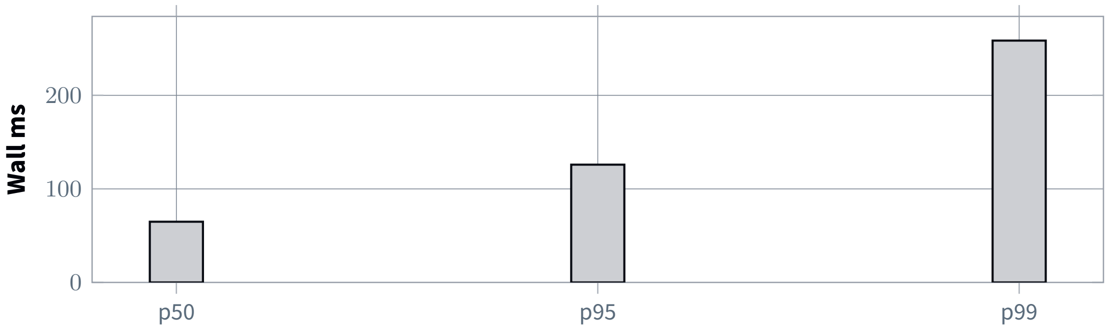


Figure 9: API reproducibility latency distribution from a 40-run sample at 5,000 assets (A3, A4).

14.5 D.5 Addendum and Supporting Panels

D.5.1 D6 Addendum Panel: Real Input Solve Comparison (5k vs 10k)

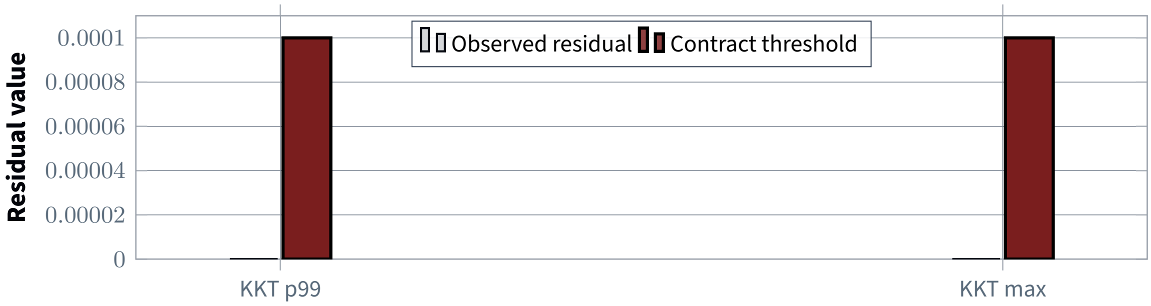


Figure 10: Feasibility and KKT residual summary with a contract threshold overlay (A3, A4).

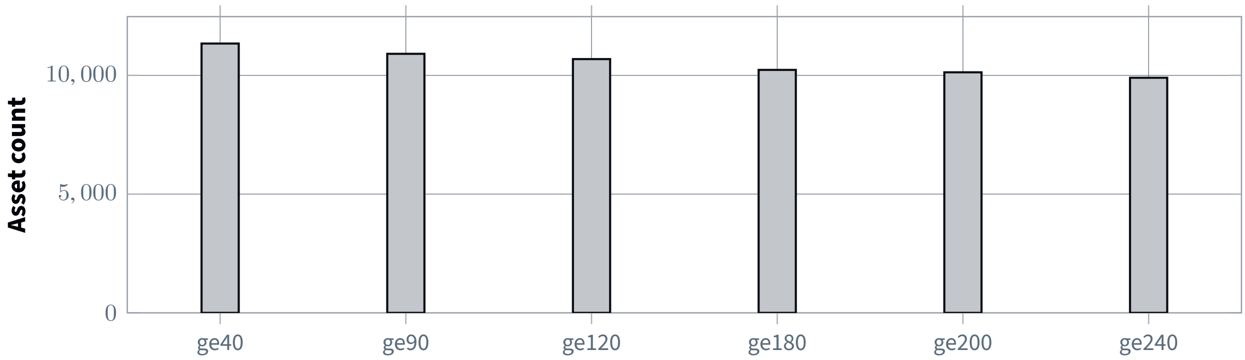


Figure 11: Real dataset coverage profile from cache summary as of 2026-02-15 (A5, A6).

Metric	5k Real Input	10k Real Input	Source
Effective N	4,996	9,989	A7
Execution path	GPU production path	GPU production path	A7
Wall ms	253.590	229.524	A7
Feasible	true	true	A7
KKT primal	0.0	0.0	A7
Quality status	optimal	optimal	A7

D.5.2 D7 Supporting Panel: 20k and 50k Scale Smoke (Non-Primary)

WARNING

20k and 50k rows below are scale-smoke operational checks. They are not claim-bearing real-input superiority evidence.

Metric	20k Scale Smoke	50k Scale Smoke	Source
Execution path	large-universe path	large-universe path	A8
Wall ms	1827.417	2673.650	A8
Feasible	true	true	A8
Status	optimal	optimal	A8
Evidence boundary	supporting only	supporting only	A8

D.5.3 Supporting Panel: Large-N GPU Moat (Real Cache, No Bootstrap)

This panel demonstrates the GPU moat at large N on real-cache-derived inputs (no bootstrap). It is PRISM GPU vs PRISM CPU evidence, not a superiority-vs-incumbent panel.

n_eff	cpu_raw_solver_ms_p50	gpu_raw_solver_ms_p50	gpu_raw_wall_ms_p50	cpu/gpu solver speedup	cpu/gpu wall speedup
20,000	1340.489	637.073	670.415	2.10x	2.00x
50,000	3896.617	711.878	818.195	5.47x	4.76x
75,257	6887.976	789.724	960.757	8.72x	7.17x

NOTE

The 100k request executed at `n_eff=75,257` due to eligibility at `min_points=40` in the expanded cache (A10).

D.5.4 Appendix-Only: Mac M2 vs RTX 4000 Ada (Small-N, Apples-to-Apples)

This is cross-platform context, not a moat proof. At small and mid-size problems, fixed GPU overhead can dominate; PRISM should route these cases to CPU. At the production gate (5,000 assets, real-data scenarios), the GPU path is superior as shown in A1.

scenario	n_assets	mac_cpu_p50_ms	rtx_cpu_p50_ms	rtx_gpu_wall_p50_ms	rtx_gpu_over_rtx_cpu	rtx_gpu_over_mac_cpu
baseline	5000	19.195	22.167	162.539	7.332	8.468
baseline	10000	39.484	46.858	171.586	3.662	4.346
hard	5000	144.858	196.848	628.708	3.194	4.340
hard	10000	395.643	631.840	662.935	1.049	1.676

14.6 D.6 Rules Matrix (Compact)

Rule	Required Fields	Pass/Fail
Scenario superiority completeness	`scenario_id`, `superiority_gate_status`	PASS
Canonical anchor exists	`benchmark_id`, `gap_certified_pct`	PASS
API traceability	`idx`, `solve_id`, `audit_hash`	PASS
Feasibility and KKT evidence	`feasible`, `kkt_primal`	PASS
Objective parity gate enforcement	`gap_certified_pct`, `quality_gate_status`	PASS

Rule	Data Source File
Scenario superiority completeness	`PRISM_EVIDENCE_SUPERIORITY_GATE_5000_REAL.csv`
Canonical anchor exists	`PRISM_EVIDENCE_CANONICAL_5000_REAL.csv`
API traceability	`PRISM_EVIDENCE_API_REPRO_5000.csv`
Feasibility and KKT evidence	`PRISM_EVIDENCE_API_REPRO_5000.csv`, `PRISM_EVIDENCE_API_REPRO_5000_summary.json`
Quality gate threshold enforcement (field: `gap_certified_pct`)	`PRISM_EVIDENCE_SUPERIORITY_GATE_5000_REAL.csv`, `PRISM_EVIDENCE_CANONICAL_5000_REAL.csv`

14.7 D.7 Deterministic API Evidence Rows (Compact Split)

idx	solve_id	audit_hash
1	d8c6acb8b7e19d8d	49d31c3d0654e20d 598a95ff09da6fb8
2	b38e3ae48cad7a57	bf7f4a33f8a1945a ff310ed0e13ccd79
3	fdee4204af8ae8a6	aa1fbaf9d2db4ecb d2178e4c45cf5b0c
4	83d8aaf7224f5983	57551e5c317b57f8 2a0b2aeb92a049c9

5	3e2745b9f7f59130	50f3ca5ecb994de8 6b97312c5c424f40
6	374a019aecb62f83	a974996786abf9d1 0fd1951f3c9df259
7	2582cee8a18caebb	ca0de75f4bb1077e e5559a27f71c7b98
8	0a50a155738fe767	edbda1d87e7854a5 ce1818bd18b8e375
9	691414cc0692dc91	e88918318d4c4e91 3df685292468a020
10	f70ca32c6feb3eab	bfcd931e87d010da 01d04171c25f72af

idx	wall_ms	objective	feasible	kkt_primal
1	65.008	0.0015497702	True	0.0
2	62.038	0.0022505680	True	0.0
3	66.169	0.0020127256	True	0.0
4	70.458	0.0023563895	True	0.0
5	66.569	0.0020132080	True	0.0
6	72.992	0.0027825911	True	0.0
7	66.647	0.0023327982	True	0.0
8	63.234	0.0013931014	True	0.0
9	124.073	0.0022295761	True	0.0
10	65.842	0.0016005182	True	0.0

15 Appendix E: Audience Cover Sheets

15.1 E.1 CTO Cover Sheet

Strictly Confidential - Audience-Specific Front Matter

To: CTO / VP Engineering / Head of Quant Dev **From:** Asymmetry Computing **Subject:** Deterministic Intraday Rebalancing Infrastructure

Context

- CPU-first overnight workflows create schedule risk at N=5,000 and above.
- Delayed solve completion reduces event-time responsiveness for tax and rebalance logic.

What PRISM Changes

- GPU-native convex QP solve path with deterministic run semantics.
- Real-data benchmark envelope: 9.60x-20.60x speedup vs Gurobi in planned scenarios.
- Audit artifacts generated per run: solve identity, trace hash, feasibility/KKT checks.

Integration Surface

- REST contract: health, solve, and audit endpoints.
- Python SDK workflow: submit solve, capture `wall_ms`, retrieve audit hash.
- Hybrid routing available for small-N CPU and large-N GPU paths.

Operational Takeaway

- Target outcome is repeatable intraday operation with bounded tail latency and reproducible evidence.

15.2 E.2 VC Cover Sheet

Strictly Confidential - Audience-Specific Front Matter

To: Investment Committee **From:** Asymmetry Computing **Subject:** PRISM Technical and Commercial Moat Analysis

Market Signal

- Direct indexing and multi-account optimization require faster and more auditable infra.
- Regulatory and resilience controls increase value of deterministic, replayable operations.

Moat Snapshot

1. Verified speed advantage in published real-data scenarios.
2. Engine-level optimization stack tuned for GPU hardware paths.
3. Built-in evidence packaging and audit-hash traceability.
4. Integration stickiness once intraday workflows are productionized.

Commercial Shape

- Value linked to latency window capture, reliability, and compliance readiness.
- Expandable from direct-indexing optimization into broader portfolio/risk workflows.
- Positioned as infrastructure layer, not a point feature.

Investment Framing

- Hardware-shift thesis with operational proof points and auditable outputs.

15.3 E.3 GPU BD Cover Sheet

Strictly Confidential - Audience-Specific Front Matter

To: Strategic Partnerships / Compute Sales **From:** Asymmetry Computing **Subject:** High-Utilization Workload for Fintech Verticals

Workload Characteristics

- Compute-intense optimization loops with recurring production cadence.
- Memory- and bandwidth-sensitive GPU workloads with sustained throughput demand at high asset counts.
- Deterministic solve requirements under institutional SLA windows.

Partnership Value

- Opens a capital-markets GPU workload vertical traditionally served by CPU.
- Supports VPC/on-prem deployment models demanded by regulated institutions.
- Produces steady utilization patterns beyond burst-style inference profiles.

Technical Fit

- Verified hardware path references: A100, H100, L4, RTX 6000 Ada.
- Standard NVIDIA driver + container runtime environment (implementation details intentionally undisclosed).
- Containerized deployment with NVIDIA runtime support.

Commercial Fit

- Joint value proposition: measurable runtime advantage + auditable operational controls.

16 Appendix F: Industry-Grade Campaign (Tiered Claims)

<!– INDUSTRY_GRADE_APPENDIX_BEGIN –>

16.1 Industry-Grade Campaign (Auto-Generated)

n_assets_requested	engine	runtime_ms_p50	runtime_ms_p95	solver_status	run_ok	failure_reason
20000	factor-cpu	4854.456126	5047.942547	optimal	True	
20000	factor-gpu	756.778485	800.548667	optimal	True	
20000	gurobi	1135.906829	1303.750927	optimal	True	
20000	osqp	3134.310228	3181.217408	maximum_iterations_reached	True	
50000	factor-cpu	3145.163117	4088.518262	optimal	True	
50000	factor-gpu	670.580206	818.741726	optimal	True	
50000	gurobi	787.958311	812.940463	optimal	True	
50000	osqp	3074.218377	3207.845251	maximum_iterations_reached	True	
100000	factor-cpu	5062.841354	5552.358036	optimal	True	
100000	factor-gpu	765.765893	907.159282	optimal	True	
100000	gurobi	1357.302632	2212.358457	optimal	True	
100000	osqp	2861.706598	3145.011268	maximum_iterations_reached	True	

Source: `PRISM_EVIDENCE_INDUSTRY_GRADE_SUMMARY_2026-02-16.json` and `PRISM_EVIDENCE_INDUSTRY_GRADE_CAMPAIGN_2026-02-16.csv` <!– INDUSTRY_GRADE_APPENDIX_END –>