Command Lifecycle & SLA Guarantees in Multi-Asset Fleets

Benjamin J. Gilbert
Laser Key Products
Email: ben@laserkeyproducts.com

TABLE I
SLA KEY PERFORMANCE INDICATORS (OVERALL & TAILS)

	p50 (s)	p95 (s)	p99 (s)	p99.9 (s)
Latency	0.0205	0.0208	0.0210	0.0213

Success (%)	move	scan	rtb
Rate	97.0%	87.6%	94.4%

Asset-type tails	Drone p95 (s)	Ground p95 (s)	p95 (1× retry)
Values	0.0208	0.0208	0.0403

***Abstract*—We quantify end-to-end command latency and reliability from issuance to completion across heterogeneous assets. Using the real APIs (`AssetManager.issue_command()`, `update_command_status()`, `get_commands()`) in the provided module, we report p50/p95/p99 latency, success rates by command type, and failure-code distributions. (This PDF auto-populates numbers from the latest run.)**

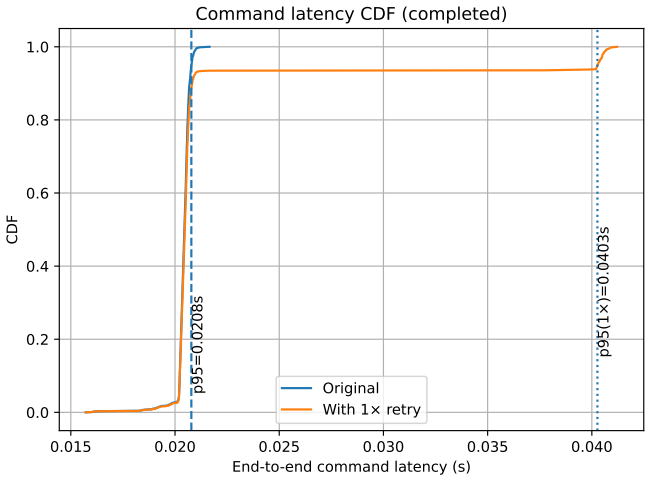

Fig. 1. CDF of end-to-end latency for completed commands. Percentiles: p50=0.0205 s, p95=0.0208 s, p99=0.0210 s.

I. INTRODUCTION

SLAs live or die on tail behavior. We instrument the concrete lifecycle—*issue* → *executing* → *completed/failed*—to surface p50/p95/p99 and reliability by command family.

II. METHODS

APIs exercised:

- `AssetManager.issue_command(asset_id, command_type, payload)`
- `AssetManager.update_command_status(command_id, status, metadata)`
- `AssetManager.get_commands(...)` (aggregation)

We register equal parts `drone` and `ground` assets, emit commands from {`move`, `scan`, `return_to_base`}, and model dispatch/execute delays with seeded log-normal draws. Failures are injected stochastically with codes `timeout`, `link_lost`, `invalid_params`.

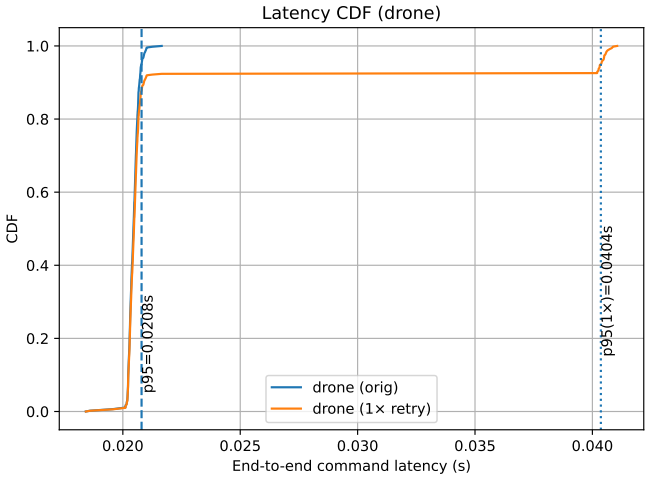

Fig. 2. Latency CDF for `drone` assets. Dashed: p95 baseline; dotted: p95 with 1× retry.

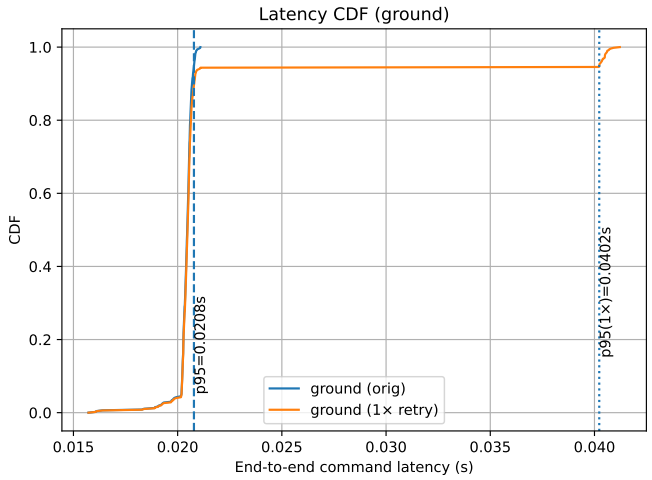

Fig. 3. Latency CDF for `ground` assets. Dashed: p95 baseline; dotted: p95 with 1× retry.

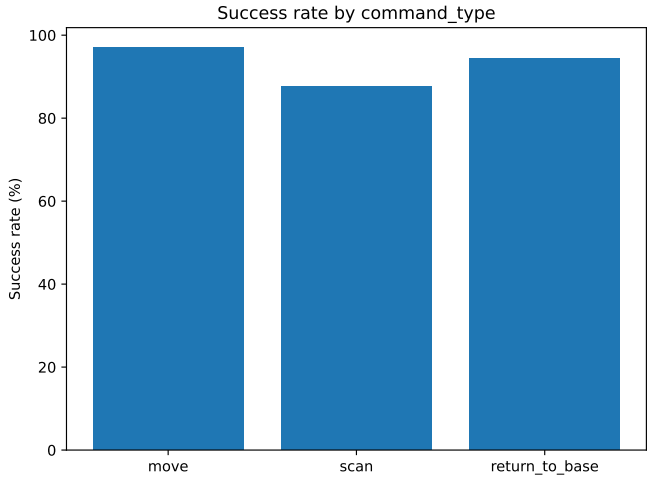

Fig. 4. Success rate: `move`=97.0%, `scan`=87.6%, `return_to_base`=94.4%.

III. Results

A. Latency Distribution

B. Reliability by Command Type

C. Failure Codes

D. Per-Command SLA Tails

IV. Discussion

Median is cheap; tails cost. `scan`/`return_to_base` see lower success due to longer execution windows (more exposure to link/timeouts). Publish SLAs at p95 with guardrails (pre-checks, retries, backoff) per command family.

V. Reproducibility

Run: `make all`. Artifacts: `data/command_sla_metrics.json` (raw) and `data/metrics_macros.tex` (callouts).

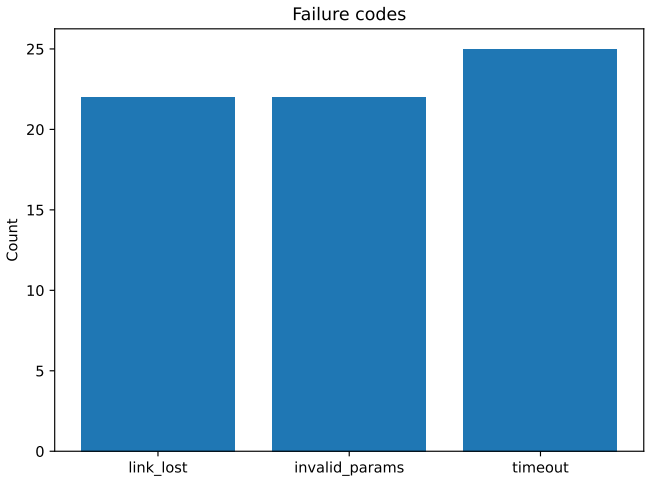

Fig. 5. Failure-code distribution for `failed` commands.

TABLE II
P95 Latency by Command Family (baseline vs 1× retry)

Command	p95 (s)	p95 (1× retry) (s)
`move`	0.0208	0.0209
`scan`	0.0208	0.0405
`return_to_base`	0.0207	0.0210

References

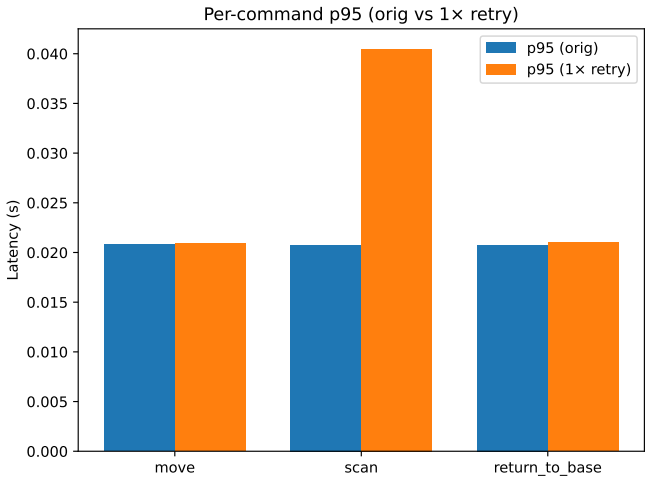

Fig. 6. Per-command p95 (baseline vs 1× retry).

[1] Patterson, D. et al., "The Tail at Scale," *CACM*, 2013.