Certified Efficient Algorithms for Tensor Completion: A Formally Verified PCG Approach with Explicit Complexity Bounds

Amadeu Zou
amadeuzou@gmail.com

February 13, 2026

Abstract

We revisit the mode-k linear subproblem in CP tensor completion with missing entries under RKHS constraints. The unknown factor is parameterized by $A_k = KW$, where $K \in \mathbb{R}^{n\times n}$ is a kernel Gram matrix and $W \in \mathbb{R}^{n\times r}$ is the optimization variable. Instead of forming an $nr \times nr$ normal matrix explicitly, we use an observation-driven operator formulation and solve by preconditioned conjugate gradients (PCG). The first contribution is a complete Lean 4 formalization of solver admissibility (SPD structure) and arithmetic complexity identities for matvec, iteration, and total cost. The second contribution is a theoretical v2 extension: we derive condition-number bounds that expose dependence on regularization λ and observation density $\rho = q/N$, and we convert them into explicit iteration bounds for target accuracy ε. The third contribution is methodological: we connect the verified operator framework to mainstream tensor completion methods (ALS/SGD), explain what is mathematically different, and discuss transfer to Tucker and tensor-train updates. We also give a reproducible numerical protocol (non-formal) to validate runtime scaling predictions. The paper separates machine-checked theorems from mathematical extensions not yet formalized, providing a rigorous and transparent path toward a fully certified algorithmic theory. Lean4 code and formal artifacts: https://github.com/amadeuzou/1stProof-lean4.

1 Introduction

Let $\mathcal{T} \in \mathbb{R}^{n_1\times\cdots\times n_d}$ be a partially observed tensor. In alternating CP optimization, fixing all factors except mode k yields a linear subproblem. Under RKHS constraints one writes

$$A_k = KW,$$

where $K \in \mathbb{R}^{n\times n}$ $(n = n_k)$ is a kernel matrix and $W \in \mathbb{R}^{n\times r}$ is unknown.

Define

$$N = \prod_{i=1}^{d} n_i, \qquad M = \prod_{i\neq k} n_i, \qquad q = |\Omega| \ll N,$$

where Ω is the observed set. The classical normal equation has size $nr \times nr$:

$$\Big[(Z\otimes K)^T SS^T (Z\otimes K) + \lambda(I_r \otimes K)\Big]\operatorname{vec}(W) = (I_r\otimes K)\operatorname{vec}(B), \tag{1}$$

with $Z \in \mathbb{R}^{M\times r}$ (Khatri–Rao side factor), selection matrix S, and $B = TZ$.

Direct dense solution is expensive in both memory and arithmetic. The central strategy here is to work only with operator actions over observed entries.

Contributions of this v2 manuscript.

1. **Certified operator framework.** We encode the mode-k system operator and PCG admissibility in Lean 4/mathlib with no placeholders.

2. **Explicit complexity formulas.** We formally verify arithmetic identities for one system matvec, one PCG step, and total cost.

3. **Condition-number theory.** We derive upper/lower spectral bounds and an explicit condition-number estimate depending on (λ, ρ).

4. **Convergence-to-accuracy complexity.** We derive a closed-form iteration bound k_ε and a corresponding total arithmetic complexity bound.

5. **Algorithmic context and transferability.** We compare with ALS/SGD and discuss operator-level generalization to Tucker/TT settings.

2 Observation-Driven Operator Formulation

Let the observed index list be

$$\Omega = \{(i_p, m_p)\}_{p=1}^{q} \subseteq [n] \times [M].$$

For $X \in \mathbb{R}^{n\times r}$ define

$$s_p(X) := \sum_{t=1}^{r} (KX)_{i_p t} Z_{m_p t}, \tag{2}$$

and

$$[D(X)]_{j\ell} := \sum_{p=1}^{q} K_{j i_p}\, Z_{m_p \ell}\, s_p(X). \tag{3}$$

Set

$$A(X) := D(X) + \lambda K X. \tag{4}$$

The right-hand side is RHS $= KB$.

Proposition 1 (Operator realization)**.** *For all $X \in \mathbb{R}^{n\times r}$,*

$$\mathrm{vec}(A(X)) = \Big[(Z \otimes K)^T S S^T (Z \otimes K) + \lambda (I_r \otimes K)\Big]\, \mathrm{vec}(X).$$

Hence solving (1) *is equivalent to solving $A(W) = KB$.*

Proof. Use $\mathrm{vec}(KX) = (I_r \otimes K)\,\mathrm{vec}(X)$. Expanding the observed part $(Z \otimes K)^T S S^T (Z \otimes K)\,\mathrm{vec}(X)$ entrywise yields exactly (2)–(3). Therefore the first term equals $\mathrm{vec}(D(X))$, and adding regularization gives the claim. □

3 Verified Arithmetic Complexity

The key implementation applies $A(\cdot)$ without forming an $nr \times nr$ matrix.

One operator application

For input X:

1. Compute $Y = KX$ (cost n^2r).
2. Compute all $s_p(X)$ (cost $q \cdot 2r$ in the adopted operation model).
3. Accumulate (3) (cost $q \cdot nr$).
4. Add λY.

Theorem 1 (Per-matvec cost, formalized)**.**

$$C_{\text{mv}} = q(2r + nr) + n^2r. \tag{5}$$

Moreover, with $N = nM$*,*

$$C_{\text{mv}} \leq nM(2r + nr) + n^2r, \tag{6}$$

and if $q < nM$ *then strict improvement holds:*

$$C_{\text{mv}} < nM(2r + nr) + n^2r. \tag{7}$$

Proof. The identity follows by summing the operation counts above. The bound and strict bound are immediate from $q \leq nM$ and $q < nM$. □

One PCG iteration

The verified operation model gives

$$\begin{aligned} C_{\text{prec}} &= n^2r, &(8)\\ C_{\text{vec}} &= 6nr, &(9)\\ C_{\text{iter}} &= C_{\text{mv}} + C_{\text{prec}} + C_{\text{vec}} \\ &= q(2r + nr) + 2n^2r + 6nr. &(10) \end{aligned}$$

For k iterations,

$$C_{\text{total}}(k) = k\, C_{\text{iter}}. \tag{11}$$

4 Condition Number Analysis and Dependence on (λ, ρ)

This section strengthens mathematical depth beyond basic SPD admissibility.

Gram decomposition

Let $k_i \in \mathbb{R}^n$ denote row i of K, $z_m \in \mathbb{R}^r$ row m of Z, and define

$$\phi_p := z_{m_p} \otimes k_{i_p} \in \mathbb{R}^{nr}.$$

Build $F \in \mathbb{R}^{q \times nr}$ with row p equal to ϕ_p^T. Then

$$\widehat{A} := F^TF + \lambda(I_r \otimes K), \tag{12}$$

which is the matrix form of A under vectorization.

Assumption 1 (Spectral and feature bounds)*.(B1) K is SPD, with eigenvalues $0 < \mu_{\min} \leq \mu_{\max}$.*

(B2) There exist constants $L_K, L_Z > 0$ such that

$$\|k_i\|_2 \le L_K \quad \forall i \in [n], \qquad \|z_m\|_2 \le L_Z \quad \forall m \in [M].$$

Theorem 2 (Eigenvalue sandwich)**.** *Under the spectral and feature bounds above,*

$$\lambda_{\min}(\widehat{A}) \ge \lambda\mu_{\min}, \tag{13}$$

and

$$\lambda_{\max}(\widehat{A}) \le qL_K^2 L_Z^2 + \lambda\mu_{\max}. \tag{14}$$

Hence

$$\kappa(\widehat{A}) \le \frac{qL_K^2 L_Z^2 + \lambda\mu_{\max}}{\lambda\mu_{\min}}. \tag{15}$$

Proof. Since $F^T F \succeq 0$, we have $\lambda_{\min}(\widehat{A}) \ge \lambda_{\min}(\lambda(I_r \otimes K)) = \lambda\mu_{\min}$, proving (13).

For the upper bound,

$$\lambda_{\max}(\widehat{A}) \le \lambda_{\max}(F^T F) + \lambda\lambda_{\max}(I_r \otimes K) = \lambda_{\max}(F^T F) + \lambda\mu_{\max}.$$

Also,

$$\lambda_{\max}(F^T F) \le \operatorname{tr}(F^T F) = \sum_{p=1}^{q} \|\phi_p\|_2^2 \le qL_K^2 L_Z^2,$$

because $\|\phi_p\|_2 = \left\|z_{m_p}\right\|_2 \left\|k_{i_p}\right\|_2 \le L_Z L_K$. Substituting gives (14), and dividing by (13) yields (15). $\square$

Corollary 1 (Observation density form)**.** *Let $\rho := q/N$ with $N = nM$. Then*

$$\kappa(\widehat{A}) \le \frac{\rho n M L_K^2 L_Z^2 + \lambda\mu_{\max}}{\lambda\mu_{\min}}. \tag{16}$$

Remark 1 (Regularization tradeoff)**.** Larger λ improves the lower spectral bound $\lambda\mu_{\min}$ and typically decreases condition number. Excessively large λ may, however, oversmooth the statistical objective. Equation (16) makes this computational tradeoff explicit.

5 PCG Convergence Rate and ε-Complexity

In the Frobenius inner product space, define the preconditioner

$$P_\mu = I_r \otimes (K + \mu I_n), \qquad \mu > 0.$$

The Lean development proves SPD admissibility for A and for the default choice $\mu = \lambda$ under structured assumptions (symmetry/positivity and $\lambda > 0$).

Let

$$\widetilde{A} := P_\mu^{-1/2} \widehat{A} P_\mu^{-1/2}, \qquad \kappa_P := \kappa(\widetilde{A}).$$

Theorem 3 (Standard PCG error contraction)**.** *For exact-arithmetic PCG iterates x_k solving $\widehat{A}x = b$,*

$$\|x_k - x_*\|_{\widehat{A}} \le 2\left(\frac{\sqrt{\kappa_P} - 1}{\sqrt{\kappa_P} + 1}\right)^k \|x_0 - x_*\|_{\widehat{A}}. \tag{17}$$

Therefore, to ensure $\|x_k - x_*\|_{\widehat{A}} \le \varepsilon \|x_0 - x_*\|_{\widehat{A}}$, it suffices that

$$k \ge k_\varepsilon := \left\lceil \frac{\log(2/\varepsilon)}{\log\left(\frac{\sqrt{\kappa_P}+1}{\sqrt{\kappa_P}-1}\right)} \right\rceil. \tag{18}$$

Corollary 2 (Arithmetic complexity to target accuracy)**.** *Using* (10) *and* (18)*,*

$$C_{\text{total}}(\varepsilon) \le k_\varepsilon\big(q(2r+nr) + 2n^2r + 6nr\big). \tag{19}$$

Hence

$$C_{\text{total}}(\varepsilon) = \mathcal{O}\left((qnr + n^2r)\sqrt{\kappa_P}\log\frac{1}{\varepsilon}\right),$$

up to lower-order linear terms in nr*.*

6 Positioning Against ALS/SGD and Generalizability

Comparison to common tensor-completion updates

Method	Computational profile	Theoretical/computational characteristic
ALS (exact sub-solve)	Typically solves normal equations directly (or via dense linear algebra) per block	Strong per-block decrease but may require expensive matrix formation/factorization in RKHS-coupled settings
SGD / stochastic updates	Low per-sample step cost	Requires stepsize schedules; convergence sensitivity and variance can dominate runtime
Operator-PCG (this work)	Observation-driven matvec + SPD preconditioner, no explicit $nr \times nr$ matrix	Deterministic linear-system perspective, explicit iteration complexity via condition number

Transfer to Tucker and Tensor Train

The same blueprint extends whenever a mode/local subproblem has normal-equation form

$$\widehat{A}_{\text{local}}x = b, \qquad \widehat{A}_{\text{local}} = F_{\text{local}}^T F_{\text{local}} + \lambda G.$$

For Tucker, F_{local} arises from Kroneckerized fixed factors. For TT, it comes from left/right environment contractions. In both cases, key tasks are: (i) observation-driven evaluation of $F_{\text{local}}^T(F_{\text{local}}x)$, (ii) proving SPD of G, and (iii) controlling local condition numbers.

7 Formalization Methodology in Lean 4

This project treats formal verification as part of mathematical methodology.

(i) Dimension-safe modeling. Unknowns are represented as

```
Matrix (Fin n) (Fin r) Real,
```

and observation maps as

```
obs : Fin q -> Fin n * Fin M.
```

This makes index domains explicit and prevents out-of-range errors by construction.

(ii) Operator-first encoding. Instead of materializing a giant normal matrix, Lean definitions use `sampleScore`, `dataTerm`, and `systemOp`. This mirrors efficient implementation and keeps proofs aligned with runtime reality.

(iii) Structured proof interfaces. Predicates `Symmetric`, `PosDef`, `PosSemidef`, and `SPD` isolate algebraic assumptions from algorithmic conclusions. The theorem `pcg_ready_fully_structured` composes these assumptions into PCG admissibility.

(iv) Certified operation counting. Costs are formalized as natural-number expressions (e.g., `costSystemMatVec`, `costPCGIter`, `totalCost`), enabling machine-checked arithmetic identities and monotonicity arguments.

8 Numerical Experiments (Non-formal but Reproducible)

The following protocol is recommended to empirically validate the theory.

Goals

1. Verify near-linear runtime growth in q at fixed (n, r).
2. Measure iteration counts versus λ and compare with the condition-number trend.
3. Compare wall-clock behavior against a dense normal-equation baseline on small/medium instances.

Protocol

1. **Synthetic dimensions:** $n \in \{200, 500, 1000\}$, $r \in \{10, 20\}$, and fixed M per suite.
2. **Observation density:** $\rho = q/(nM) \in \{0.1\%, 0.5\%, 1\%, 5\%\}$.
3. **Regularization sweep:** $\lambda \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$.
4. **Stopping rule:** relative preconditioned residual $\leq 10^{-6}$.
5. **Metrics:** matvec time, per-iteration time, total runtime, iteration count, final residual.

Falsifiable hypotheses

(H1) Runtime per iteration scales approximately linearly with q (from (10)).

(H2) Increasing λ reduces iterations in regimes where regularization dominates the smallest eigenvalue.

(H3) The operator-PCG method outperforms dense normal-equation solves once $q \ll N$.

9 Lean Theorem Map and Formal Status

Statement	Lean artifact	Status
Observation-driven operator definitions	`Question10/Defs.lean`	Formalized
Matvec cost identity (5)	`costSystemMatVec_eq`	Formalized
Ambient bound/strict improvement	`costSystemMatVec_le_ambient_scaled`, `..._lt_...`	Formalized
System SPD from structured assumptions	`spd_systemOp_of_assumptions`	Formalized
Preconditioner SPD	`spd_preconditioner_of_kernel_assumptions`	Formalized
PCG admissibility	`pcg_ready_fully_structured`	Formalized
Per-iteration and total cost formulas	`costPCGIter_eq`, `totalCost_eq`	Formalized
Condition-number bridge from extremal assumptions	`kappaFromExtremes_le_kappaUpperFromCount` (`SpectralConvergence.lean`)	Formalized
Structured spectral-assumption package and bundled bounds	`SpectralAssumptions.eigen_sandwich`, `...kappa_bound_count`	Formalized
Matrix-level positivity chain for regularized normal matrix	`gramMatrix_posSemidef`, `regularizedNormalMatrix_posDef`	Formalized
Quadratic-form lower/upper envelopes for regularized normal matrix	`regularizedNormalMatrix_quad_lower`, `...quad_upper`	Formalized
Feature-bound to explicit $qL_K^2L_Z^2$ Gram-envelope bridge	`gramMatrix_quad_upper_of_feature_bounds_q`	Formalized
Kronecker row-factorization norm bound	`rowNormSq_kron_le_product_bounds`	Formalized
Factorized-feature to explicit $qL_K^2L_Z^2$ bridge	`gramMatrix_quad_upper_of_factorized_feature_bounds_q`	Formalized
Observed-map plus factor-matrix to explicit $qL_K^2L_Z^2$ bridge	`gramMatrix_quad_upper_of_observed_factor_matrices_q`	Formalized
Quadratic-form envelopes to eigenvalue sandwich	`regularizedNormalMatrix_eigenvalue_lower`, `...eigenvalue_upper`	Formalized
Eigenvalue-ratio (condition-number style) upper bound in count form	`regularizedNormalMatrix_eigen_ratio_le_kappaUpperFromCoun`	Formalized
k_ε ceiling lower-bound identity	`kEps_ge_log_ratio`	Formalized
Logarithmic-to-geometric ε step at k_ε	`geometric_term_le_eps_of_kEps`	Formalized
Assumption-driven ε guarantee at k_ε	`error_bound_at_kEps_of_assumptions`	Formalized
Direct ε guarantee at k_ε from the log step	`error_bound_at_kEps`	Formalized
Condition-number bounds (Section 4)	This manuscript, Theorem 2	Mathematical (v2)
ε-iteration and total complexity bounds	Theorem 3, Cor. 2	Mathematical (v2)

10 Conclusion

This v2 manuscript upgrades the original verified algorithm analysis into a more standard research-paper form. It preserves machine-checked results for operator design, admissibility, and arithmetic complexity, and adds explicit condition-number and convergence-rate analysis tied to regularization and observation density. The resulting picture is both rigorous and actionable: one gets verified per-iteration complexity plus a mathematically explicit iteration budget for target accuracy.

The analytic logarithmic step in Section 5 (turning k_ε into the explicit geometric ε guarantee) is now formalized in Lean. A remaining direction is to formalize the full PCG contraction theorem itself under the same abstract interface, so the entire convergence chain is derived within one machine-checked framework.