A MATRIX-FREE PRECONDITIONED CONJUGATE-GRADIENT SOLVER FOR THE MODE-K RKHS SUBPROBLEM IN KERNELIZED CP TENSOR COMPLETION WITH MISSING ENTRIES

Abstract. We study the mode-wise alternating least-squares (ALS) update for a kernelized CP tensor completion model with missing (unaligned) data, focusing on an infinite-dimensional mode constrained to a reproducing kernel Hilbert space (RKHS). When all CP factors except the mode-k factor are fixed, the RKHS parameterization $A_k = KW$ leads to normal equations of dimension $nr \times nr$, where $n = n_k$ and r is the CP rank. Forming and solving this system directly costs $\Theta(n^3r^3)$ and is impractical. We derive a self-adjoint matrix-free operator enabling (preconditioned) conjugate gradients (PCG) using only kernel multiplications and sparse accumulations over the $q \ll N$ observed tensor entries, with no computation of order $N = \prod_i n_i$ and without forming any Kronecker or Khatri–Rao products. We rigorously address the positive-(semi)definiteness subtleties arising from a positive semidefinite kernel Gram matrix, and we propose provably SPD preconditioners, including a robust kernel-block preconditioner and a stronger Kronecker-spectral preconditioner calibrated to the sampling rate. Per PCG iteration, the matrix-vector product costs $O(qr + n^2r)$ (or $O(nr^2 + n^2r)$ with optional preprocessing), and the preconditioner applications cost $O(n^2r)$ or $O(n^2r + nr^2)$ depending on the choice.

1. Introduction

Let $\mathcal{T} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$ be a d-way tensor with missing (unaligned) entries. Let $N := \prod_{i=1}^{d} n_i$ be the total number of tensor entries and let $q \ll N$ be the number of observed entries. Kernelized tensor completion methods combine low-rank multilinear structure with smoothness constraints on one or more modes via RKHS penalties. A standard computational strategy is alternating minimization over the CP factors. When a mode is constrained to lie in an RKHS, the mode-wise subproblem is typically a large linear system whose naive solution scales cubically in the total number of unknowns (nr), and whose explicit formation is prohibitively expensive for missing data due to the ambient dimension N.

This paper gives a self-contained, rigorous derivation of a matrix-free preconditioned conjugate-gradient (PCG) solver for the mode-k subproblem when mode k is kernelized and data are missing. Our emphasis is the computation of matrix-vector products and preconditioner applications without forming any object of size N or $M = \prod_{i \neq k} n_i$, under the regime $n, r < q \ll N$ and $n \ll M$.

2. Problem setup and the mode-k normal equations

Fix an index $k \in \{1, \ldots, d\}$. Set

$$n := n_k, \qquad M := \prod_{i \neq k} n_i, \qquad N = nM.$$

Let the observed index set be

$$\Omega \subseteq [n_1] \times \cdots \times [n_d], \qquad |\Omega| = q.$$

We store observations as pairs $(\boldsymbol{i}^{(\ell)}, t_\ell)_{\ell=1}^{q}$ where $\boldsymbol{i}^{(\ell)} = (i_1^{(\ell)}, \ldots, i_d^{(\ell)}) \in \Omega$ and $t_\ell = \mathcal{T}_{\boldsymbol{i}^{(\ell)}} \in \mathbb{R}$.

2.1. **Masking operators.** Let $\mathcal{P}_\Omega$ denote the orthogonal projection that keeps entries in Ω and zeros out all other entries. On the mode-k unfolding, we write P_Ω for the corresponding projection acting on $n \times M$ matrices:

$$(P_\Omega(F))_{i,m} = \begin{cases} F_{i,m}, & \text{if } (i,m) \text{ corresponds to an observed tensor entry in } \Omega, \\ 0, & \text{otherwise.} \end{cases}$$

Equivalently, in vectorized form, P_Ω is an $N \times N$ diagonal projector with 1's on observed locations and 0's elsewhere. A common formal device is a selection matrix $S \in \mathbb{R}^{N \times q}$ such that $S^\top \operatorname{vec}(F) \in \mathbb{R}^q$ extracts the observed entries of F; then

$$P_\Omega = SS^\top, \qquad P_\Omega^2 = P_\Omega = P_\Omega^\top \succeq 0.$$

In computations we *never* form S or P_Ω explicitly; we work directly with the index list Ω.

2.2. **CP factors and Khatri–Rao structure.** Let $r \in \mathbb{N}$ be the CP rank. For each mode $j \neq k$ we have a factor matrix

$$A_j \in \mathbb{R}^{n_j \times r},$$

which is held fixed during the mode-k update. Let the Khatri–Rao product over all modes except k be

$$Z := A_d \odot \cdots \odot A_{k+1} \odot A_{k-1} \odot \cdots \odot A_1 \in \mathbb{R}^{M \times r},$$

where $\odot$ denotes the columnwise Kronecker (Khatri–Rao) product.

Let $T \in \mathbb{R}^{n \times M}$ be the mode-k unfolding of $\mathcal{T}$ in which missing entries are set to 0. Define the (masked) MTTKRP

$$B := TZ \in \mathbb{R}^{n \times r}.$$

Because missing entries of T are 0, B depends only on observed entries and can be assembled from Ω without forming T or Z (Section 7).

2.3. **Kernelized mode-k factor.** Let $K \in \mathbb{R}^{n \times n}$ be a symmetric positive semidefinite (psd) kernel Gram matrix, arising from an RKHS on the mode-k index set. We parameterize the mode-k factor as

$$A_k = KW, \qquad W \in \mathbb{R}^{n \times r} \text{ unknown.}$$

Let $w = \operatorname{vec}(W) \in \mathbb{R}^{nr}$.

2.4. **Least-squares subproblem and normal equations.** A standard mode-wise subproblem for tensor completion minimizes the squared error on observed entries plus an RKHS ridge penalty. In the present notation, this becomes

$$\min_{W \in \mathbb{R}^{n \times r}} \frac{1}{2} \left\| S^\top \operatorname{vec}(T) - S^\top (Z \otimes K) \operatorname{vec}(W) \right\|_2^2 + \frac{\lambda}{2} \operatorname{trace}(W^\top K W), \qquad \lambda > 0. \tag{1}$$

The Euler–Lagrange equations yield the normal equations

$$\Big[(Z \otimes K)^\top P_\Omega (Z \otimes K) + \lambda (I_r \otimes K) \Big] w = (Z \otimes K)^\top P_\Omega \operatorname{vec}(T). \tag{2}$$

Since T has zeros on missing entries, $P_\Omega \operatorname{vec}(T) = \operatorname{vec}(T)$. Using the vec–Kronecker identity $(B^\top \otimes A)\operatorname{vec}(X) = \operatorname{vec}(AXB)$ we have

$$(Z \otimes K)^\top \operatorname{vec}(T) = (Z^\top \otimes K)\operatorname{vec}(T) = \operatorname{vec}(KTZ) = \operatorname{vec}(KB).$$

Thus the system (2) can be written in the form used in the problem statement:

$$\Big[(Z \otimes K)^\top P_\Omega (Z \otimes K) + \lambda (I_r \otimes K)\Big] w = (I_r \otimes K)\operatorname{vec}(B). \tag{3}$$

3. Symmetry and positive-(semi)definiteness

Conjugate gradients (CG) and preconditioned CG (PCG) apply to linear systems with symmetric positive definite (SPD) matrices. The matrix in (3) is always symmetric and positive semidefinite, but it may be singular if K is singular. We treat this issue rigorously.

Proposition 1 (Self-adjointness and psd property)**.** *Let $K \succeq 0$ and $\lambda \geq 0$. Then the system matrix*

$$A := (Z \otimes K)^\top P_\Omega (Z \otimes K) + \lambda (I_r \otimes K)$$

is symmetric and positive semidefinite. Moreover, for any $W \in \mathbb{R}^{n \times r}$,

$$\operatorname{vec}(W)^\top A \operatorname{vec}(W) = \left\| P_\Omega(KWZ^\top) \right\|_{\mathrm{F}}^2 + \lambda \operatorname{trace}(W^\top K W). \tag{4}$$

Proof. Symmetry is immediate since $P_\Omega = P_\Omega^\top$ and $K = K^\top$. For the quadratic form, write $w = \operatorname{vec}(W)$ and note that

$$(Z \otimes K) w = \operatorname{vec}(KWZ^\top).$$

Then, since P_Ω is an orthogonal projector,

$$w^\top (Z \otimes K)^\top P_\Omega (Z \otimes K) w = \|P_\Omega (Z \otimes K) w\|_2^2 = \left\| P_\Omega(KWZ^\top) \right\|_{\mathrm{F}}^2.$$

Also $w^\top (I_r \otimes K) w = \operatorname{trace}(W^\top K W)$. Summing yields (4). □

3.1. When is the system SPD?.

Corollary 1 (SPD condition)**.** *If $K \succ 0$ and $\lambda > 0$, then $A \succ 0$ and CG/PCG applies to* (3)*.*

Proof. If $K \succ 0$ and $\lambda > 0$, then $\operatorname{trace}(W^\top K W) = \left\| K^{1/2} W \right\|_{\mathrm{F}}^2 > 0$ for any $W \neq 0$. Hence (4) implies $w^\top A w > 0$ for all $w \neq 0$. □

3.2. The psd kernel subtlety and two rigorous resolutions.

If $K \succeq 0$ is singular, then $\operatorname{trace}(W^\top K W) = 0$ for any W whose columns lie in $\ker(K)$. In that case, also $KWZ^\top = 0$, hence the data term vanishes, and the full matrix A is singular. This is not a pathology: it reflects non-identifiability of W under the parameterization $A_k = KW$ when K is singular.

We give two standard, rigorous resolutions.

Assumption 1 (Either strict PD or a well-posed regularized surrogate)**.** Throughout the remainder we assume one of the following:

(a) **Strict PD:** $K \succ 0$ (e.g. for strictly positive definite kernels on distinct points), so Corollary 1 applies; or
(b) **Nugget regularization:** we replace K by $\widetilde{K} := K + \varepsilon I_n$ with $\varepsilon > 0$, so $\widetilde{K} \succ 0$.

Remark 1 (What does adding εI change?)**.** Replacing K by $\widetilde{K} = K + \varepsilon I$ changes the objective (1) by replacing the RKHS seminorm term $\operatorname{trace}(W^\top K W)$ with $\operatorname{trace}(W^\top K W) + \varepsilon \|W\|_{\mathrm{F}}^2$, i.e. adding a small Euclidean ridge on W. This is a standard "nugget" or "jitter" that makes the normal equations strictly SPD and improves numerical stability. As $\varepsilon \downarrow 0$, solutions of the regularized problem converge (under mild conditions) to minimum-$\|\cdot\|_{\mathrm{F}}$ solutions among minimizers of the original psd problem; we do not pursue this limit analysis here. If one insists on *not* modifying the objective, one may instead reparameterize using a factorization of K and solve an SPD system in the range space; see Remark 2.

Remark 2 (Range-space formulation without modifying the objective)**.** Let $K = Q\Lambda Q^\top$ be an eigen-decomposition with $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, $\lambda_i \geq 0$. Let $Q_+ \in \mathbb{R}^{n \times m}$ and $\Lambda_+ \in \mathbb{R}^{m \times m}$ collect the positive eigenpairs, where $m = \operatorname{rank}(K)$, and define $L := Q_+ \Lambda_+^{1/2} \in \mathbb{R}^{n \times m}$ so that $K = LL^\top$. Write $A_k = KW = LL^\top W = LU$ with $U := L^\top W \in \mathbb{R}^{m \times r}$. Then $\operatorname{trace}(W^\top K W) = \|U\|_{\mathrm{F}}^2$, and $(Z \otimes K)\operatorname{vec}(W) = (Z \otimes L)\operatorname{vec}(U)$. The subproblem (1) is equivalent to

$$\min_{U \in \mathbb{R}^{m \times r}} \ \frac{1}{2} \left\| S^\top \operatorname{vec}(T) - S^\top (Z \otimes L) \operatorname{vec}(U) \right\|_2^2 + \frac{\lambda}{2} \|U\|_{\mathrm{F}}^2 ,$$

whose normal equations are SPD due to the λI term (dimension $mr \times mr$). All matrix-free ideas below carry over verbatim with K replaced by L and with one fewer kernel multiplication per matvec. This formulation avoids the nugget modification but requires access to (possibly low-rank) factors of K.

For concreteness, we proceed with $\widetilde{K} \succ 0$ (either because $K \succ 0$ or because a nugget was added) and drop the tilde from the notation when no confusion arises.

4. Kronecker/vec identities and a matrix-free operator

We derive a matrix-free operator that applies the $nr \times nr$ system matrix to a vector without forming any Kronecker products.

Lemma 1 (Vec–Kronecker identities)**.** *Let $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times m}$, and $X \in \mathbb{R}^{n \times m}$. Then*

$$(B^\top \otimes A)\ \operatorname{vec}(X) = \operatorname{vec}(AXB).$$

In particular, if $Z \in \mathbb{R}^{M \times r}$ and $K \in \mathbb{R}^{n \times n}$,

$$(Z \otimes K)\operatorname{vec}(X) = \operatorname{vec}(KXZ^\top), \qquad (Z \otimes K)^\top \operatorname{vec}(Y) = \operatorname{vec}(KYZ)$$

for all conforming matrices $X \in \mathbb{R}^{n \times r}$ and $Y \in \mathbb{R}^{n \times M}$.

Proof. Standard; see, e.g., any linear algebra text covering Kronecker products. It follows from bilinearity and checking on basis matrices. □

Proposition 2 (Matrix-form linear operator)**.** *Assume $K \succ 0$ (or K replaced by $\widetilde{K} \succ 0$). Define the linear map $\mathcal{A} : \mathbb{R}^{n \times r} \to \mathbb{R}^{n \times r}$ by*

$$\boxed{\mathcal{A}(X) = K\Big(P_\Omega(KXZ^\top)\, Z\Big) + \lambda KX.} \tag{5}$$

Then for all $X \in \mathbb{R}^{n \times r}$,

$$\operatorname{vec}(\mathcal{A}(X)) = \Big[(Z \otimes K)^\top P_\Omega (Z \otimes K) + \lambda (I_r \otimes K)\Big] \operatorname{vec}(X).$$

Moreover, $\mathcal{A}$ is self-adjoint and SPD with respect to the Frobenius inner product:

$$\langle X, \mathcal{A}(X)\rangle_{\mathrm{F}} = \left\|P_\Omega(KXZ^\top)\right\|_{\mathrm{F}}^2 + \lambda \left\|K^{1/2}X\right\|_{\mathrm{F}}^2 > 0 \quad \textit{for } X \neq 0.$$

Proof. Let $x = \mathrm{vec}(X)$. By Lemma 1, $(Z \otimes K)x = \mathrm{vec}(KXZ^\top)$. Apply P_Ω in vectorized form, then apply $(Z \otimes K)^\top$ using Lemma 1 again:

$$(Z \otimes K)^\top P_\Omega(Z \otimes K)x = (Z^\top \otimes K)\ \mathrm{vec}(P_\Omega(KXZ^\top)) = \mathrm{vec}(K(P_\Omega(KXZ^\top))Z),$$

which is the first term in (5). The ridge term satisfies $(I_r \otimes K)\,\mathrm{vec}(X) = \mathrm{vec}(KX)$, giving the second term.

Self-adjointness and SPD follow from Proposition 1 and Corollary 1, transported from vector to matrix form via vec. □

5. Observed-index representation and no N/M objects

The operator (5) appears to involve $KXZ^\top \in \mathbb{R}^{n\times M}$ and the projector P_Ω acting on an $n \times M$ matrix. We now show that both can be applied using only the q observed entries without forming any length-N vector, any $n \times M$ matrix, or Z itself.

5.1. **Evaluating rows of the Khatri–Rao product on demand.**

For each observation $\ell \in \{1, \dots, q\}$ with tensor index $\boldsymbol{i}^{(\ell)} = (i_1^{(\ell)}, \dots, i_d^{(\ell)})$, define the mode-$k$ row index

$$i^{(\ell)} := i_k^{(\ell)} \in \{1, \dots, n\}.$$

The unfolding column index $m^{(\ell)} \in \{1, \dots, M\}$ corresponding to $(i_1^{(\ell)}, \dots, i_{k-1}^{(\ell)}, i_{k+1}^{(\ell)}, \dots, i_d^{(\ell)})$ need not be formed. Instead we compute the corresponding *row of Z* directly from the fixed factor matrices:

$$z^{(\ell)} := A_d[i_d^{(\ell)}, :] \odot \cdots \odot A_{k+1}[i_{k+1}^{(\ell)}, :] \odot A_{k-1}[i_{k-1}^{(\ell)}, :] \odot \cdots \odot A_1[i_1^{(\ell)}, :] \in \mathbb{R}^r. \tag{6}$$

Computing $z^{(\ell)}$ costs $O((d-1)r)$ flops.

Remark 3 (Precompute vs. on-the-fly)**.** Since the A_j ($j \neq k$) are fixed during the mode-k solve, one can precompute and store all $z^{(\ell)}$ once per outer ALS step: time $O(q(d-1)r)$ and memory $O(qr)$. If memory is constrained, one may recompute $z^{(\ell)}$ on-the-fly inside each PCG iteration, increasing the per-iteration cost of the masked step from $O(qr)$ to $O(q(d-1)r)$.

6. Matrix-free matrix-vector products

We now derive an explicit algorithm for applying $\mathcal{A}$ in (5) to a matrix $X \in \mathbb{R}^{n\times r}$ using only kernel multiplications and a sparse accumulation over Ω.

6.1. **Derivation.**

Let

$$G := KX \in \mathbb{R}^{n\times r}.$$

The matrix $KXZ^\top$ is $GZ^\top \in \mathbb{R}^{n\times M}$. For observation ℓ, the entry of $GZ^\top$ at the observed unfolding coordinate $(i^{(\ell)}, m^{(\ell)})$ equals

$$(GZ^\top)_{i^{(\ell)}, m^{(\ell)}} = G[i^{(\ell)}, :] \cdot z^{(\ell)},$$

where $G[i, :] \in \mathbb{R}^{1\times r}$ denotes row i and $\cdot$ is the Euclidean inner product. Thus, defining

$$u_\ell := G[i^{(\ell)}, :] \cdot z^{(\ell)} \in \mathbb{R},$$

Algorithm 1 Matrix-free matvec $Y = \mathcal{A}(X)$

Require: $X \in \mathbb{R}^{n\times r}$, kernel matrix $K \in \mathbb{R}^{n\times n}$, ridge $\lambda > 0$, observations $\{i^{(\ell)}, z^{(\ell)}\}_{\ell=1}^{q}$.
Ensure: $Y = \mathcal{A}(X) \in \mathbb{R}^{n\times r}$.
1: $G \leftarrow KX$ ▷ $O(n^2 r)$ for dense K
2: $H \leftarrow 0 \in \mathbb{R}^{n\times r}$
3: **for** $\ell = 1, \dots, q$ **do**
4: $\quad u \leftarrow G[i^{(\ell)}, :] \cdot z^{(\ell)}$ ▷ $O(r)$
5: $\quad H[i^{(\ell)}, :] \leftarrow H[i^{(\ell)}, :] + u\, z^{(\ell)\top}$ ▷ $O(r)$
6: **end for**
7: $Y \leftarrow KH + \lambda G$ ▷ $O(n^2 r)$ for dense K
8: **return** Y

the masked matrix $P_\Omega(GZ^\top)$ is zero everywhere except at the q observed positions, where it equals u_ℓ.

We need to form

$$H := P_\Omega(GZ^\top)\, Z \in \mathbb{R}^{n\times r}.$$

Since only observed entries contribute, H can be computed by *scatter-add* over observations:

$$H[i,:] = \sum_{\ell:\, i^{(\ell)}=i} u_\ell\, z^{(\ell)\top}, \qquad i = 1, \dots, n. \tag{7}$$

Finally,

$$\mathcal{A}(X) = KH + \lambda G.$$

6.2. **Algorithm and complexity.**

Proposition 3 (Matvec cost; no N or M)**.** *Assume dense K. Given precomputed $\{z^{(\ell)}\}_{\ell=1}^{q}$, Algorithm 1 computes $Y = \mathcal{A}(X)$ with cost*

$$O(n^2 r) + O(qr) + O(n^2 r) = O(qr + n^2 r).$$

No object of size $N = nM$ is formed, and neither Z nor any Kronecker product is formed. If $z^{(\ell)}$ are computed on-the-fly, the masked step costs $O(q(d-1)r)$ instead of $O(qr)$.

Proof. Immediate from the loop counts and two dense kernel multiplications. The absence of N- or M-sized objects follows by inspection. □

6.3. **Optional preprocessing: row-wise Gram compression.**

When q is extremely large and r is small, it can be beneficial to preprocess the observation-dependent Gram matrices

$$C_i := \sum_{\ell:\, i^{(\ell)}=i} z^{(\ell)} z^{(\ell)\top} \in \mathbb{R}^{r\times r}, \qquad i = 1, \dots, n.$$

This costs $O(qr^2)$ per outer ALS step and stores $O(nr^2)$ numbers.

Proposition 4 (Row-wise Gram acceleration)**.** *With C_i precomputed, the masked accumulation in* (7) *satisfies*

$$H[i,:] = G[i,:]\, C_i \qquad (i = 1, \dots, n),$$

so the masked step can be computed in $O(nr^2)$ per matvec (instead of $O(qr)$). Thus the matvec cost becomes $O(n^2 r + nr^2)$ per iteration.

Proof. For fixed i, substitute $u_\ell = G[i,:] \cdot z^{(\ell)}$ into (7) and factor out $G[i,:]$:

$$H[i,:] = \sum_{\ell : i^{(\ell)} = i} (G[i,:] z^{(\ell)})\, z^{(\ell)\top} = G[i,:] \sum_{\ell : i^{(\ell)} = i} z^{(\ell)} z^{(\ell)\top} = G[i,:] C_i.$$

□

7. Right-hand side assembly

The right-hand side of (3) is $\mathrm{vec}(KB)$ where $B = TZ$. Since T has zeros at missing entries, B can be assembled from the observations without forming T or Z.

Proposition 5 (MTTKRP from observed entries)**.** *Let $B = TZ$ with T the zero-filled mode-k unfolding. Then*

$$B[i,:] = \sum_{\ell :\, i^{(\ell)} = i} t_\ell \, z^{(\ell)\top}, \qquad i = 1, \dots, n. \tag{8}$$

Thus B can be computed in $O(qr)$ flops given $\{z^{(\ell)}\}$, with no N- or M-sized objects.

Proof. By definition, $(TZ)[i,:] = \sum_{m=1}^{M} T_{i,m} Z[m,:]$. Only observed entries contribute because $T_{i,m} = 0$ if $(i,m) \notin \Omega$ in unfolding coordinates. Each observed entry in row i contributes $t_\ell z^{(\ell)\top}$, yielding (8). □

After assembling B, compute KB in $O(n^2 r)$ for dense K, and set $b = \mathrm{vec}(KB)$.

8. PCG in matrix form

We solve the SPD linear system

$$\mathcal{A}(W) = KB, \qquad W \in \mathbb{R}^{n \times r}, \tag{9}$$

where $\mathcal{A}$ is given by (5). Because vec is an isometry between $(\mathbb{R}^{n\times r}, \langle\cdot,\cdot\rangle_{\mathrm{F}})$ and $(\mathbb{R}^{nr}, \langle\cdot,\cdot\rangle)$, running CG/PCG on (3) is equivalent to running CG/PCG on (9) using Frobenius inner products.

Let $M : \mathbb{R}^{n\times r} \to \mathbb{R}^{n\times r}$ be an SPD preconditioner (linear, self-adjoint and SPD w.r.t. $\langle\cdot,\cdot\rangle_{\mathrm{F}}$). PCG requires:

(i) one application of $\mathcal{A}$ per iteration (Algorithm 1), and
(ii) one application of M^{-1} per iteration (Section 9).

Remark 4 (Stopping criteria)**.** Since $\mathcal{A}$ is SPD, the residual norm $\|R_j\|_{\mathrm{F}}$ is a standard and reliable stopping criterion. One may also monitor the (preconditioned) residual $\|U_j\|_{\mathrm{F}}$ or the energy norm.

9. Preconditioner design and application costs

A good preconditioner M should (i) be SPD, (ii) be cheap to apply, and (iii) reduce the condition number of $M^{-1}\mathcal{A}$ to decrease PCG iterations.

We present two principled choices.

Algorithm 2 PCG for $\mathcal{A}(W) = KB$ in matrix form

Require: Initial guess $W_0 \in \mathbb{R}^{n\times r}$, tolerance $\tau > 0$, operator $\mathcal{A}$, RHS KB, SPD preconditioner M.
Ensure: Approximate solution W.

$R_0 \leftarrow KB - \mathcal{A}(W_0)$
$U_0 \leftarrow M^{-1}(R_0)$
$D_0 \leftarrow U_0$
for $j = 0, 1, 2, \dots$ **do**
  $Q_j \leftarrow \mathcal{A}(D_j)$
  $\alpha_j \leftarrow \langle R_j, U_j\rangle_{\mathrm{F}} \,/\, \langle D_j, Q_j\rangle_{\mathrm{F}}$
  $W_{j+1} \leftarrow W_j + \alpha_j D_j$
  $R_{j+1} \leftarrow R_j - \alpha_j Q_j$
  if $\|R_{j+1}\|_{\mathrm{F}} \,/\, \|KB\|_{\mathrm{F}} \le \tau$ **then**
    return W_{j+1}
  end if
  $U_{j+1} \leftarrow M^{-1}(R_{j+1})$
  $\beta_j \leftarrow \langle R_{j+1}, U_{j+1}\rangle_{\mathrm{F}} \,/\, \langle R_j, U_j\rangle_{\mathrm{F}}$
  $D_{j+1} \leftarrow U_{j+1} + \beta_j D_j$
end for

9.1. **Kernel-block (ridge-term) preconditioner.** The simplest robust choice is to precondition by the ridge term:

$$M_A := \lambda(I_r \otimes K) \qquad \Longleftrightarrow \qquad M_A(R) = \lambda KR. \tag{10}$$

It is SPD since $K \succ 0$ and $\lambda > 0$.

Applying M_A^{-1} amounts to solving $\lambda KU = R$ column-wise. With a Cholesky factorization $K = LL^\top$, one application costs $O(n^2 r)$ via r triangular solves; the setup costs $O(n^3)$.

Remark 5 (Approximate solves)**.** If n is large, one may replace exact Cholesky solves by (incomplete) Cholesky, Nyström approximations, or iterative solves for $KU = R$. Such approximations can still yield effective preconditioning; the present paper focuses on exact SPD preconditioners for clarity.

9.2. **Kronecker-spectral preconditioner.** The masked data term $(Z \otimes K)^\top P_\Omega (Z \otimes K)$ destroys exact Kronecker structure, but a strong preconditioner can be obtained by approximating it by a Kronecker product motivated by full observation.

9.2.1. *Motivation by full observation and sampling rate.* If all entries were observed, $P_\Omega = I_N$, then

$$(Z \otimes K)^\top (Z \otimes K) = (Z^\top Z) \otimes (K^2).$$

Under approximately uniform missingness, it is natural to scale this by the sampling rate

$$\rho := \frac{q}{N} \in (0, 1],$$

which is computable from the tensor dimensions (no $O(N)$ operations are required; N is used only as a scalar). This leads to the SPD preconditioner

$$M_B := \rho\,(G_Z \otimes K^2) + \lambda(I_r \otimes K), \qquad G_Z \approx Z^\top Z. \tag{11}$$

Since $K \succ 0$ and $\lambda > 0$, M_B is SPD for any symmetric $G_Z \succeq 0$ and any $\rho \ge 0$.

Remark 6 (Choice of G_Z)**.** Two practical options avoid forming Z:

(i) **Exact Khatri–Rao Gram:** $G_Z = Z^\top Z$ can be computed using the Hadamard identity
$$Z^\top Z = *_{j\neq k}(A_j^\top A_j),$$
which costs $O(\sum_{j\neq k} n_j r^2)$; see Lemma 2.

(ii) **Observed Gram:** $G_{Z,\Omega} := \sum_{\ell=1}^{q} z^{(\ell)} z^{(\ell)\top}$, costing $O(qr^2)$, which can better reflect nonuniform sampling.

Lemma 2 (Khatri–Rao Gram identity)**.** *Let $Z = A_s \odot \cdots \odot A_1$ be a Khatri–Rao product of conforming matrices. Then*
$$Z^\top Z = (A_s^\top A_s) * \cdots * (A_1^\top A_1).$$

Proof. The (p, q) entry of $Z^\top Z$ is the inner product between the p-th and q-th columns of Z. Each column of Z is the Kronecker product of the corresponding columns of the A_j. Inner products of Kronecker products factor as products of inner products, yielding the Hadamard product identity. □

9.2.2. *Fast application of M_B^{-1}.* Let $G_Z = U\Sigma U^\top$ with $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_r) \succeq 0$, and $K = Q\Lambda Q^\top$ with $\Lambda = \mathrm{diag}(\kappa_1, \ldots, \kappa_n) \succ 0$. Then
$$M_B = (U \otimes Q)\ \mathrm{diag}\left(\rho\,\sigma_j\,\kappa_p^2 + \lambda\,\kappa_p\right)_{p=1,\ldots,n;\ j=1,\ldots,r}\ (U \otimes Q)^\top.$$

Thus, for $R \in \mathbb{R}^{n\times r}$, $U = M_B^{-1}(R)$ can be computed as follows:
$$\widehat{R} := Q^\top R\, U \in \mathbb{R}^{n\times r}, \qquad \widehat{U}_{p,j} := \frac{\widehat{R}_{p,j}}{\rho\,\sigma_j\,\kappa_p^2 + \lambda\,\kappa_p}, \qquad U := Q\,\widehat{U}\,U^\top.$$

The per-application cost is $O(n^2 r + nr^2)$ for dense orthogonal transforms, after one-time eigendecompositions: $O(n^3 + r^3)$.

Remark 7 (When is M_B worth it?)**.** M_B can significantly reduce PCG iterations when the masked term dominates and when K and G_Z have favorable spectra. It is most attractive when r is modest and an eigendecomposition of K is acceptable (or already available from kernel methods). If one prefers only Cholesky factors, M_A is simpler and robust.

10. Overall complexity and memory (no $O(N)$ terms)

We summarize costs for one mode-k update, assuming K is dense. Let t be the PCG iteration count.

10.1. **Precomputations per outer ALS step.**

- Compute $z^{(\ell)}$ for $\ell = 1, \ldots, q$ (optional but typical): $O(q(d-1)r)$ time, $O(qr)$ memory.
- Assemble B via (8): $O(qr)$ time, $O(nr)$ memory.
- Form KB: $O(n^2 r)$ time.
- Optional: precompute C_i (row-wise Grams): $O(qr^2)$ time, $O(nr^2)$ memory.
- Preconditioner setup:
  - M_A: Cholesky of K: $O(n^3)$ time, $O(n^2)$ memory.
  - M_B: eigendecompositions of K and G_Z: $O(n^3 + r^3)$ time, $O(n^2 + r^2)$ memory (plus computing G_Z or $G_{Z,\Omega}$ as discussed).

10.2. **Per PCG iteration.** Each iteration performs one matvec and one preconditioner application, plus $O(nr)$ inner products and saxpy operations.

- Matvec $\mathcal{A}(X)$:
  - without C_i: $O(qr + n^2 r)$ time;
  - with C_i: $O(nr^2 + n^2 r)$ time.
- Preconditioner apply:
  - M_A^{-1}: $O(n^2 r)$ time;
  - M_B^{-1}: $O(n^2 r + nr^2)$ time.

Thus the dominant total complexity is

$$O\Big(q(d-1)r \;+\; qr \;+\; n^2 r \;+\; \text{setup}(M) \;+\; t \cdot (\text{matvec} + \text{apply}(M^{-1}))\Big),$$

with *no* computations of order $N = nM$ and no explicit construction of Z or any Kronecker product.

11. Conclusion

The mode-k RKHS subproblem in kernelized CP tensor completion with missing entries leads to a large $nr \times nr$ SPD (or psd) linear system. We derived a rigorously justified matrix-free operator and an observed-index implementation of its matrix-vector products that avoids any $O(N)$ operations. This enables (preconditioned) conjugate gradients with per-iteration cost $O(qr + n^2 r)$, substantially improving over dense $O(n^3 r^3)$ solvers. We addressed the kernel psd subtlety by giving two mathematically sound routes to SPD: either a nugget-regularized kernel (common in practice) or an exact range-space reformulation. Finally, we proposed practical SPD preconditioners and detailed their application costs and trade-offs.

References

[1] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., Johns Hopkins University Press, 2013.
[2] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, 2003.
[3] T. G. Kolda and B. W. Bader, Tensor decompositions and applications, *SIAM Review* **51** (2009), no. 3, 455–500.
[4] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.