Tensor trains for high-dimensional problems

Mi-Song Dupuy

March 30, 2023

Contents

Introduction

These notes are a short introduction to the tensor train decomposition, with a particular focus on solving linear equations within this format. The tensor train decomposition [OT09] is presented as a generalisation of the singular value decomposition for matrices, which is central in the characterisation of the low-rank approximation problem. Approximation results for the tensor train format as well as the tensor train manifold are discussed.

The second part deals with the numerical resolution of linear systems or eigenvalue problems. The historical algorithm is an alternating scheme, known as the density matrix renormalisation group (DMRG) [Whi92, HRS12a], using the variational formulation of symmetric linear problems. Another way to solve linear problems is to adapt the classical iterative methods to the tensor train format [KU16]. Both approaches are presented and discussed in the present notes.

These notes are inspired by the following texts on the tensor train decomposition [Hac12, Hac14, Sch11, BSU16, UV20].

Chapter 1

Tensor trains

1.1 Singular value decomposition and generalisations for tensors

This chapter is devoted to the tensor train decomposition, as a generalisation of the singular value decomposition (SVD) for high-dimensional tensors. The SVD arises in the low-rank approximation of matrices, as such, it is natural to look for generalisation of the SVD for high-dimensional tensors. As it will be mentioned, the historical tensor formats, i.e. the CP decomposition and the Tucker decomposition suffer from drawbacks that the tensor train format does not have.

1.1.1 Tensors and reshapes

A tensor C of order $L \in \mathbb{N}$ is a multidimensional array $C_{i_1 \dots i_L} \in \mathbb{R}^{n_1 \times \dots \times n_L}$.

A convenient way to represent tensor and product of tensors is the graphical representation. Let $C \in \mathbb{R}^{n_1 \times \dots \times n_L}$ be a tensor. The graphical representation of C is given by

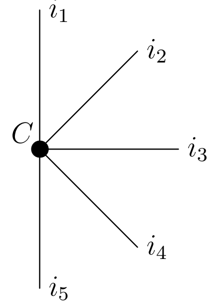
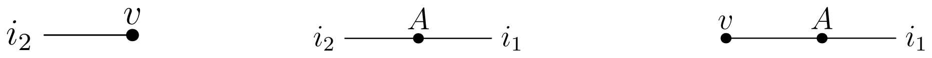


Figure 1.1: Graphical representation of C.

Each vertex represents a tensor and each edge an index of the tensor.



(a) Vector v_{i_2}. (b) Matrix $A_{i_1}^{i_2}$. (c) Matrix-vector product $(Av)_{i_1} = \sum_{i_2} A_{i_1}^{i_2} v_{i_2}$.

Figure 1.2: Contraction of tensors. Every pair of connected edges is a summation over the shared index.

Definition 1.1.1 (Reshape of a tensor)**.** *Let $C \in \mathbb{R}^{n_1 \times \cdots \times n_L}$ be a tensor. Let $(j_1, \dots, j_\ell, k_1, \dots, k_n)$ be a permutation of $\{1, \dots, L\}$. We say that the matrix $C_{i_{j_1} \cdots i_{j_\ell}}^{i_{k_1} \cdots i_{k_n}} \in \mathbb{R}^{n_{j_1} \cdots n_{j_\ell} \times n_{k_1} \cdots n_{k_n}}$ is a reshape of C.*

The reshapes $C_{i_1 \dots i_\ell}^{i_{\ell+1} \dots i_L}$ will be of particular interest for tensor trains.

1.1.2 The low-rank approximation for matrices

Theorem 1.1.2. *Let $A \in \mathbb{R}^{m \times n}$ be a matrix. There exist orthogonal matrices $U \in \mathbb{R}^{m \times r_A}$ and $V \in \mathbb{R}^{n \times r_A}$, and a diagonal matrix $\Sigma = \mathrm{Diag}(s_1, \dots, s_{r_A})$ with $s_1 \geq \cdots \geq s_{r_A} > 0$ such that $A = U \Sigma V^T$. The triplet of matrices (U, Σ, V^T) satisfying these properties is called a singular value decomposition (SVD) of A.*

An important property of the singular value decomposition is the following.

Theorem 1.1.3 (Best rank r approximation of a matrix [Sch08])**.** *Let $A \in \mathbb{R}^{m \times n}$ be a matrix and (U, Σ, V^T) an SVD of A. The best rank-r of A in the Frobenius norm is given by*

$$A_r = U_r \Sigma_r V_r^T = \sum_{k=1}^{r} s_k u_k v_k^T,$$

where $U_r \in \mathbb{R}^{m \times r}$, $\Sigma_r \in \mathbb{R}^{r \times r}$ and $V_r \in \mathbb{R}^{n \times r}$ are the respective truncations of U, Σ and V. The error is given by

$$\|A - A_r\|_F = \Big(\sum_{k \geq r+1} s_k^2 \Big)^{1/2}. \tag{1.1.1}$$

The best approximation is unique if $s_r > s_{r+1}$.

Another way to phrase the best rank r approximation of a matrix is to take the subspace point of view. A matrix $A \in \mathbb{R}^{m \times n}$ can be viewed as a vector of the product space $\mathbb{R}^m \otimes \mathbb{R}^n$ which is isometrically isomorphic to $\mathbb{R}^{mn}$. The subspace problem is phrased as follows: find subspaces $\mathcal{U} \subset \mathbb{R}^m$ and $\mathcal{V} \subset \mathbb{R}^n$ both of dimension r such that it minimises the distance

$$\operatorname{dist}(A, \mathcal{U} \otimes \mathcal{V}) = \|A - \Pi_{\mathcal{U}\otimes\mathcal{V}} A\| = \min_{\substack{\widetilde{\mathcal{U}} \subset \mathbb{R}^m, \dim \widetilde{\mathcal{U}} = r \\ \widetilde{\mathcal{V}} \subset \mathbb{R}^n, \dim \widetilde{\mathcal{V}} = r}} \|A - \Pi_{\widetilde{\mathcal{U}}\otimes\widetilde{\mathcal{V}}} A\|, \tag{1.1.2}$$

where $\Pi_{\mathcal{W}}$ is the orthogonal projection onto the subspace $\mathcal{W} \subset \mathbb{R}^{mn}$. The SVD of the matrix (A_i^j) is also a representation of the vector $(A_{ij})_{1\leq i\leq m, 1\leq j\leq n}$ in the orthonormal basis $(u_i \otimes v_j)_{1\leq i\leq m, 1\leq j\leq n}$:

$$A = \sum_{k=1}^{r_A} s_k u_k \otimes v_k. \tag{1.1.3}$$

Proposition 1.1.4. *Let $A \in \mathbb{R}^{m\times n}$, (U, Σ, V^T) its SVD and $r \in \mathbb{N}$. Denote $(u_1, \dots, u_{r_A})$ and $(v_1, \dots, v_{r_A})$ the respective columns of U and V. A solution to the subspace minimisation problem (1.1.2) is given by*

$$\mathcal{U} = \operatorname{Span}(u_1, \dots, u_r), \quad \mathcal{V} = \operatorname{Span}(v_1, \dots, v_r). \tag{1.1.4}$$

The solution is unique if $s_r > s_{r+1}$.

Proof. Let $\widetilde{\mathcal{U}}$ and $\widetilde{\mathcal{V}}$ be respectively subspaces of $\mathbb{R}^m$ and $\mathbb{R}^n$ of dimension r. Let $(\widetilde{u}_i)_{1\leq i\leq r}$ and $(\widetilde{v}_i)_{1\leq i\leq r}$ be ONB of respectively $\widetilde{\mathcal{U}}$ and $\widetilde{\mathcal{V}}$. The minimisation problem (1.1.2) can be rewritten as

$$\min_{\substack{\widetilde{\mathcal{U}} \subset \mathbb{R}^m, \dim \widetilde{\mathcal{U}} = r \\ \widetilde{\mathcal{V}} \subset \mathbb{R}^n, \dim \widetilde{\mathcal{V}} = r}} \|A - \Pi_{\widetilde{\mathcal{U}}\otimes\widetilde{\mathcal{V}}} A\| = \min_{\substack{\widetilde{\mathcal{U}} \subset \mathbb{R}^m, \dim \widetilde{\mathcal{U}} = r \\ \widetilde{\mathcal{V}} \subset \mathbb{R}^n, \dim \widetilde{\mathcal{V}} = r}} \|A - P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}}\|_F^2,$$

where $P_{\widetilde{\mathcal{U}}}$ (resp. $P_{\widetilde{\mathcal{V}}}$) is the orthogonal projection onto $\widetilde{\mathcal{U}}$ (resp. $\widetilde{\mathcal{V}}$).

Let $\widetilde{\mathcal{U}}$ and $\widetilde{\mathcal{V}}$ be respectively subspaces of $\mathbb{R}^m$ and $\mathbb{R}^n$ of dimension r. Let $(\widetilde{u}_i)_{1\leq i\leq r}$ and $(\widetilde{v}_i)_{1\leq i\leq r}$ be ONB of respectively $\widetilde{\mathcal{U}}$ and $\widetilde{\mathcal{V}}$. Then we have

$$\begin{aligned}
\|A - P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}}\|_F^2 &= \operatorname{Tr}\big((A - P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}})^T (A - P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}})\big) \\
&= \operatorname{Tr}\big(A^T A - P_{\widetilde{\mathcal{V}}} A^T P_{\widetilde{\mathcal{U}}} A - A^T P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}} + P_{\widetilde{\mathcal{V}}} A^T P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}}\big) \\
&= \operatorname{Tr}\big(A^T A\big) - \operatorname{Tr}\big(P_{\widetilde{\mathcal{V}}} A^T P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}}\big),
\end{aligned}$$

where we have used that since $P_{\widetilde{\mathcal{V}}}$ is an orthogonal projection, we have $\operatorname{Tr}\big(P_{\widetilde{\mathcal{V}}} A^T P_{\widetilde{\mathcal{U}}} A\big) = \operatorname{Tr}\big(A^T P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}}\big) = \operatorname{Tr}\big(P_{\widetilde{\mathcal{V}}} A^T P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}}\big)$. We realise that

$$\operatorname{Tr}\big(P_{\widetilde{\mathcal{V}}} A^T P_{\widetilde{\mathcal{U}}} A P_{\widetilde{\mathcal{V}}}\big) = \sum_{1\leq i,j\leq r} \langle \widetilde{u}_i, A\widetilde{v}_j\rangle^2.$$

Solving the minimisation problem (1.1.2) is equivalent to maximising $\sum_{1\leq i,j\leq r} \big(\langle \widetilde{u}_i, A\widetilde{v}_j\rangle\big)^2$ where $(\widetilde{u}_i)_{1\leq i\leq r}$ and $(\widetilde{v}_i)_{1\leq i\leq r}$ are orthonormal families. Using the SVD of A, the previous quantity is maximised for $\widetilde{\mathcal{U}} = \operatorname{Span}(u_1, \dots, u_r)$ and $\widetilde{\mathcal{V}} = \operatorname{Span}(v_1, \dots, v_r)$. □

1.1.3 Generalisations of the SVD for tensors

For higher-order tensors, different schematic generalisations of the SVD are possible. With the previous discussion, there are two natural options that emerge:

- write the tensor as a sum of rank-1 tensors:

$$C = \sum_{k=1}^{r} u_k^{(1)} \otimes \cdots \otimes u_k^{(L)},$$

  where $u_k^{(j)} \in \mathbb{R}^{n_j}$. This is the *canonical polyadic decomposition* (CP decomposition);

- consider the subspace minimisation problem:

$$\operatorname{dist}(C, \mathcal{U}_1 \otimes \mathcal{U}_2 \otimes \cdots \otimes \mathcal{U}_L) = \min_{\widetilde{\mathcal{U}}_1 \subset \mathbb{R}^{n_1}, \dim \widetilde{\mathcal{U}}_1 = r_1, \dots, \widetilde{\mathcal{U}}_L \subset \mathbb{R}^{n_L}, \dim \widetilde{\mathcal{U}}_L = r_L} \|C - \Pi_{\widetilde{\mathcal{U}}_1 \otimes \cdots \otimes \widetilde{\mathcal{U}}_L} C\|,$$

  where $\dim \mathcal{U}_k = r_k$ for all $1 \leq k \leq L$. This yields the Tucker decomposition.

The canonical decomposition looks the most appealing as it is the most sparse way to represent a tensor. It has however one major drawback, being that the best rank r approximation (in the sense of the CP decomposition) is *ill-posed!* [DSL08] Consider noncolinear vectors $a \in \mathbb{R}^n$, $b \in \mathbb{R}^n$ and the tensor

$$C = b \otimes a \otimes a + a \otimes b \otimes a + a \otimes a \otimes b.$$

which is a tensor of canonical rank 3. It can however be approximated as well as we wish by a tensor of canonical rank 2: let $\varepsilon > 0$, then we see that

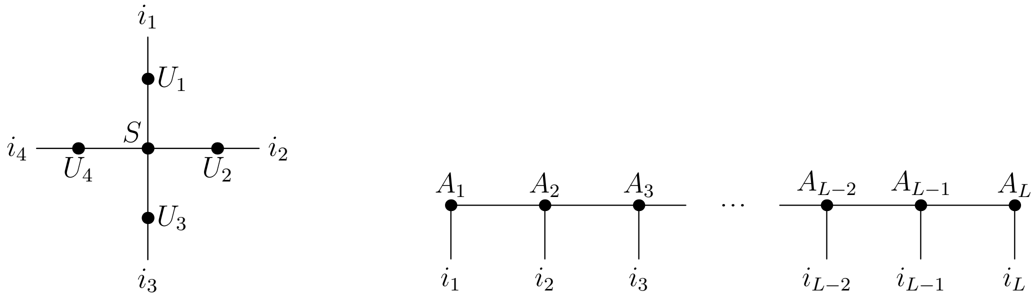
$$C - \left(\frac{1}{\varepsilon}(a + \varepsilon b) \otimes (a + \varepsilon b) \otimes (a + \varepsilon b) - \frac{1}{\varepsilon} a \otimes a \otimes a \right) = \mathcal{O}(\varepsilon). \tag{1.1.5}$$

Contrary to matrices, the set of tensors of canonical rank less than r is not closed.

Regarding the Tucker decomposition, let $C \in \mathcal{U}_1 \otimes \cdots \otimes \mathcal{U}_L$. Then there is a core tensor $S \in \mathbb{R}^{r_1 \times \cdots \times r_L}$ and matrices $(U_k)_{1 \leq k \leq L} \in \bigotimes_{k=1}^{L} \mathbb{R}^{n_k \times r_k}$ such that

$$\forall\, 1 \leq i_k \leq n_k,\ C_{i_1 \dots i_L} = \sum_{j_1=1}^{r_1} \cdots \sum_{j_L=1}^{r_L} S_{j_1 \dots j_L} (U_1)_{i_1}^{j_1} \cdots (U_L)_{i_L}^{j_L}.$$

The storage cost of the tensor C is still exponential in the order L of the tensor (except if some r_k are equal to 1). As such it is a useful decomposition only for low order tensors. In the following, we will focus on the efficient representation of tensors of order up to a hundred, for which the Tucker decomposition is not suited.



(a) Tucker decomposition (b) Tensor train decomposition

Figure 1.3: Tucker and tensor train decompositions

1.2 Tensor train decomposition

1.2.1 Definition

The tensor train (TT) decomposition [OT09], also called *matrix product state* [KSZ91] in the physics litterature is the simplest instance of a tensor network. The TT decomposition is related to the density-matrix renormalisation group (DMRG) [Whi92] pioneered by White for the computation of properties of one-dimensional statistical physics systems. The connection between DMRG and TT has been realised later [OR95, DMNS98].

Definition 1.2.1 ([KSZ91, OT09])**.** *Let $C \in \mathbb{R}^{n_1 \times \cdots \times n_L}$ be a tensor. We say that $(A_1, \dots, A_L)$ is a tensor train decomposition of C if we have for all $1 \le i_k \le n_k$*

$$C_{i_1 \dots i_L} = A_1[i_1] A_2[i_2] \cdots A_L[i_L] \tag{1.2.1}$$

$$= \sum_{\alpha_1=1}^{r_1} \sum_{\alpha_2=1}^{r_2} \cdots \sum_{\alpha_{L-1}=1}^{r_{L-1}} A_1[i_1]_{\alpha_1} A_2[i_2]_{\alpha_2}^{\alpha_1} \cdots A_L[i_L]^{\alpha_{L-1}}, \tag{1.2.2}$$

where for each $1 \le i_k \le n_k$, $A_k[i_k] \in \mathbb{R}^{r_{k-1} \times r_k}$ ($r_0 = r_L = 1$). The tensor A_k are called the TT cores *and the sizes of the TT cores are the* TT ranks *of C.*

Such a representation has a storage cost of $\sum_{k=1}^{L} n_k r_{k-1} r_k$. Provided that the TT ranks do not increase exponentially with the order L of the tensor, the TT decomposition is a sparse representation of the tensor C. As it will be highlighted later, an exact TT representation of any tensor C can be derived using the hierarchical SVD. Generically, the TT ranks of the tensor will be exponential in L, however, good approximations for problems can be achieved for problems with some notion of sparsity [Has07, DDGS16].

Example 1.2.2.

- *a tensor product $C_{i_1 \dots i_L} = u_{i_1}^{(1)} \cdots u_{i_L}^{(L)}$ is a TT of TT rank 1, as the cores are $(u_{i_k}^{(k)})_{1 \le k \le L, 1 \le i_k \le n_k}$.*

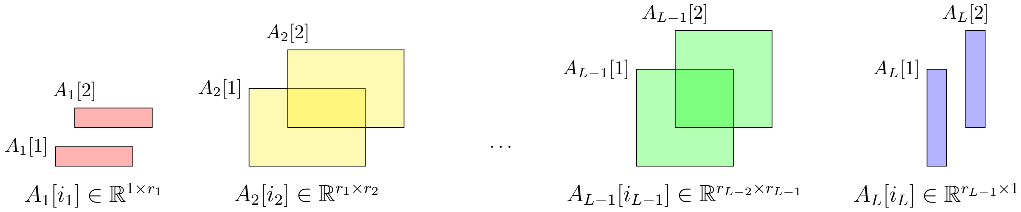


Figure 1.4: Schematic representation of the TT decomposition

- *the unnormalised Bell state* $B \in \bigotimes_1^{2L} \mathbb{R}^2$

$$B_{i_1\dots i_{2L}} = \big(\delta_{1,i_1}\delta_{2,i_2} + \delta_{2,i_1}\delta_{1,i_2}\big)\big(\delta_{1,i_3}\delta_{2,i_4} + \delta_{2,i_3}\delta_{1,i_4}\big)\cdots \big(\delta_{1,i_{2L-1}}\delta_{2,i_{2L}} + \delta_{2,i_{2L-1}}\delta_{1,i_{2L}}\big),$$

is a TT of rank 2: let $(B_k)_{1\le k\le 2L}$ *be defined by*

$$B_{2k-1}[i_{2k-1}] = \begin{bmatrix}\delta_{1i_{2k-1}} & \delta_{2i_{2k-1}}\end{bmatrix}, \quad B_{2k}[i_{2k}] = \begin{bmatrix}\delta_{2i_{2k}} \\ \delta_{1i_{2k}}\end{bmatrix}, \quad k = 1,\dots,L.$$

By a direct calculation, we can check that $B_{i_1\dots i_{2L}} = B_1[i_1]\cdots B_{2L}[i_L]$.

- *for* $L = 2$, *the following reordering of the indices of the Bell state* $\widetilde{B} \in \bigotimes_1^4 \mathbb{R}^2$

$$\widetilde{B}_{i_1\dots i_4} = \big(\delta_{1,i_1}\delta_{2,i_3} + \delta_{2,i_1}\delta_{1,i_3}\big)\big(\delta_{1,i_2}\delta_{2,i_4} + \delta_{2,i_3}\delta_{1,i_4}\big)$$

has a TT decomposition of rank 4*:*

i_k	$\widetilde{B}_1$	$\widetilde{B}_2$	$\widetilde{B}_3$	$\widetilde{B}_4$
1	$\begin{bmatrix}1 & 0\end{bmatrix}$	$\begin{bmatrix}0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0\end{bmatrix}$	$\begin{bmatrix}0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0\end{bmatrix}$	$\begin{bmatrix}1 \\ 0\end{bmatrix}$
2	$\begin{bmatrix}0 & 1\end{bmatrix}$	$\begin{bmatrix}1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1\end{bmatrix}$	$\begin{bmatrix}1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0\end{bmatrix}$	$\begin{bmatrix}0 \\ 1\end{bmatrix}$

This elementary example highlights the importance of the ordering of the indices of the tensor for an efficient TT representation.

Remark 1.2.3. *The reordered Bell state example* $\widetilde{B} \in \bigotimes_1^{2L} \mathbb{R}^2$

$$\widetilde{B}_{i_1 \dots i_{2L}} = \prod_{k=1}^{L} \left(\delta_{1,i_k} \delta_{2,i_{k+L}} + \delta_{2,i_k} \delta_{1,i_{k+L}} \right)$$

has a TT decomposition of rank 2^L. *The optimality of the ranks is proved by the characterisation of the TT ranks stated in Theorem 1.2.7.*

It is clear that there is no uniqueness of the TT decomposition. Indeed for a tensor $C \in \mathbb{R}^{n_1 \times \dots \times n_L}$ if $(A_1, \dots, A_L)$ is a tensor train decomposition, then for any invertible matrices $(G_k)_{1 \le k \le L-1} \in \bigotimes_{k=1}^{L-1} \mathrm{GL}_{r_k}(\mathbb{R})$, the TT cores $(\widetilde{A}_1, \dots, \widetilde{A}_L)$ defined by

$$\begin{cases} \widetilde{A}_1[i_1] = A_1[i_1] G_1, \ i_1 = 1, \dots, n_1, \quad \widetilde{A}_L[i_L] = G_{L-1}^{-1} A_L[i_L], \ i_L = 1, \dots, n_L \\ \widetilde{A}_k[i_k] = G_{k-1}^{-1} A_k[i_k] G_k, \ i_k = 1, \dots, n_k, \ k = 2, \dots, L-1, \end{cases}$$

is an equivalent TT representation.

As we are going to see later on, it is possible to partially lift this gauge freedom by imposing additional properties on the TT cores (A_k).

Proposition 1.2.4 (Algebraic properties of TT)**.** *Let* $(A_1, \dots, A_L)$*and* $(\widetilde{A}_1, \dots, \widetilde{A}_L)$ *be the respective TT decompositions of the tensors* $C, \widetilde{C} \in \mathbb{R}^{n_1 \times \dots \times n_L}$. *Then*

$$\begin{aligned} B_1[i_1] &= \begin{pmatrix} A_1[i_1] & \widetilde{A}_1[i_1] \end{pmatrix}, \quad B_L[i_L] = \begin{bmatrix} A_L[i_L] \\ \widetilde{A}_L[i_L] \end{bmatrix} \\ B_k[i_k] &= \begin{bmatrix} A_k[i_k] & 0 \\ 0 & \widetilde{A}_k[i_k] \end{bmatrix}, k = 2, \dots, L-1 \end{aligned} \tag{1.2.3}$$

is a TT decomposition of the sum $C + \widetilde{C}$.

The proof consists in expanding the TT decomposition $(B_1, \dots, B_L)$. The TT decomposition (1.2.3) is in general not minimal and can be compressed as explained in Section 1.3.

Remark 1.2.5. *Since a tensor product* $u^{(1)} \otimes \dots \otimes u^{(L)}$ *is a TT of rank 1, we deduce that a CP decomposition of rank* r *has at most a TT representation of rank* r. *The TT decomposition is a generalisation of the CP format, with advantageous algebraic and topologic properties.*

1.2.2 The hierarchical SVD

The hierarchical SVD (HSVD) is an algorithm [Vid03, OT09] to obtain a tensor train representation of any tensor. In the HSVD, we apply successive SVD to $C \in \mathbb{R}^{n_1 \times \cdots \times n_L}$:

$$\begin{aligned}
C_{i_1 \dots i_L} &= (C_{i_1}^{i_2 \dots i_L}) && \text{(reshape of } C \text{ to } n_1 \times n_2 \cdots n_L) \\
&= (U_1)_{i_1}^{\alpha_1} (\Sigma_1 V_1)_{\alpha_1}^{i_2 \dots i_L} && \text{(SVD)} \\
&= (U_1)_{i_1}^{\alpha_1} (\Sigma_1 V_1)_{\alpha_1 i_2}^{i_3 \dots i_L} && \text{(reshape of } \Sigma_1 V_1) \\
&= (U_1)_{i_1}^{\alpha_1} (U_2)_{\alpha_1 i_2}^{\alpha_2} (\Sigma_2 V_2)_{\alpha_2}^{i_3 \dots i_L} && \text{(SVD of } \Sigma_1 V_1) \\
&= (U_1)_{i_1}^{\alpha_1} (U_2)_{\alpha_1 i_2}^{\alpha_2} (\Sigma_2 V_2)_{\alpha_2 i_3}^{i_4 \dots i_L} && \text{(reshape of } \Sigma_2 V_2),
\end{aligned}$$

we repeat the process until we get

$$C_{i_1 \dots i_L} = (U_1)_{i_1}^{\alpha_1} (U_2)_{\alpha_1 i_2}^{\alpha_2} \cdots (U_{L-1})_{\alpha_{L-2} i_{L-1}}^{\alpha_{L-1}} (\Sigma_{L-1} V_{L-1})_{\alpha_{L-1}}^{i_L}.$$

The identification with the TT decomposition is clear, one simply needs to be careful with the switch in the role played by the virtual indices:

$$\begin{aligned}
C_{i_1 \dots i_L} &= (U_1)_{i_1}^{\alpha_1} \quad (U_2)_{\alpha_1 i_2}^{\alpha_2} \quad \cdots \quad (U_{L-1})_{\alpha_{L-2} i_{L-1}}^{\alpha_{L-1}} (\Sigma_{L-1} V_{L-1})_{\alpha_{L-1}}^{i_L} \\
&= A_1[i_1]_{\alpha_1} \quad A_2[i_2]_{\alpha_2}^{\alpha_1} \quad \cdots \quad A_{L-1}[i_{L-1}]_{\alpha_{L-1}}^{\alpha_{L-2}} \quad A_L[i_L]^{\alpha_{L-1}}.
\end{aligned}$$

There are a few immediate remarks:

(i). it is possible to start at the end, *i.e.* by first reshaping C into the matrix $C_{i_1 \dots i_{L-1}}^{i_L} \in \mathbb{R}^{n_1 \cdots n_{L-1} \times n_L}$, perform its SVD and carry on. Another TT representation is obtained this way;

(ii). from the HSVD algorithm, we guess that the singular values Σ_k are related to the singular values of the reshapes $C_{i_1 \dots i_k}^{i_{k+1} \dots i_L} \in \mathbb{R}^{n_1 \cdots n_k \times n_{k+1} \cdots n_L}$ and that they play a key role in the best approximation by a TT at fixed TT ranks. This is indeed the case and it will be treated in Section 1.3.

This algorithm is central in the theory of TT and more generally in the approximation theory by tensor networks. It is somewhat clear that such an algorithm extends to the decomposition into a tree tensor network. Indeed, in the HSVD algorithm, we simply partition $\{1, \dots, L\}$ into the sets $(\{1\}, \{2, \dots, L\})$, then $(\{1\}, \{2\}, \{3, \dots, L\})$, and so on so forth. For trees, we choose different partition choices that does not have to reduce to a singleton right away. For tensor networks with loops, there is no equivalent of the HSVD for the construction of a tensor network directly from the tensor. This makes the analysis of such networks much more difficult.

1.2.3 Normalisation and gauge freedom

Definition 1.2.6. *We say that a TT decomposition* $(A_1, \dots, A_L)$ *is*

- left-orthogonal *if for all* $1 \leq k \leq L-1$ *we have*

$$\sum_{i_k=1}^{n_k} A_k[i_k]^* A_k[i_k] = \mathrm{id}_{r_k}; \tag{1.2.4}$$

- right-orthogonal *if for all* $2 \leq k \leq L$ *we have*

$$\sum_{i_k=1}^{n_k} A_k[i_k] A_k[i_k]^* = \mathrm{id}_{r_{k-1}}. \tag{1.2.5}$$

From the HSVD algorithm, we see that we obtain a left-orthogonal TT decomposition of the tensor C. By starting from the end, we would get a right-orthogonal TT representation of C.

Such a normalisation turns out to be convenient for the computation of the norm a tensor. Suppose that $(A_1, \dots, A_L)$ is a left-orthogonal TT decomposition. The norm of the corresponding tensor C remarkably simplifies

$$\begin{aligned}
\|C\|_F^2 &= \sum_{i_1=1}^{n_1} \cdots \sum_{i_L=1}^{n_L} \big(A_1[i_1] \cdots A_L[i_L]\big)^2 \\
&= \sum_{i_1=1}^{n_1} \cdots \sum_{i_L=1}^{n_L} A_L[i_L]^T \cdots A_1[i_1]^T A_1[i_1] \cdots A_L[i_L] \\
&= \sum_{i_1=1}^{n_1} \cdots \sum_{i_L=1}^{n_L} A_L[i_L]^T \cdots A_1[i_1]^T A_1[i_1] \cdots A_L[i_L] \\
&= \sum_{i_2=1}^{n_2} \cdots \sum_{i_L=1}^{n_L} A_L[i_L]^T \cdots \Big(\sum_{i_1=1}^{n_1} A_1[i_1]^T A_1[i_1]\Big) \cdots A_L[i_L] \\
&= \sum_{i_2=1}^{n_2} \cdots \sum_{i_L=1}^{n_L} A_L[i_L]^T \cdots A_2[i_2]^T A_2[i_2] \cdots A_L[i_L],
\end{aligned}$$

where the left-orthogonality of A_1 has been used. Hence by iterating this argument, the norm of C is simply the norm of the last TT core A_L.

Another instance where the choice of the normalisation is crucial is in solving eigenvalue problems in DMRG (see Chapter 2).

It is also possible to mix both normalisations, in the sense that for some $2 \leq n \leq L-1$, we have

- the first $n-1$ TT cores are left-orthogonal: for $1 \leq k \leq n-1$
$$\sum_{i_k=1}^{n_k} A_k[i_k]^T A_k[i_k] = \mathrm{id}_{r_k};$$
- the last $L-n+1$ TT cores are right-orthogonal: for $n+1 \leq k \leq L$
$$\sum_{i_k=1}^{n_k} A_k[i_k] A_k[i_k]^T = \mathrm{id}_{r_{k-1}}. \tag{1.2.6}$$

In that case, the norm of the tensor is carried by the TT core that is not normalised, using the following trick:

$$\begin{aligned}
\|C\|_F^2 &= \sum_{i_1=1}^{n_1} \cdots \sum_{i_L=1}^{n_L} A_L[i_L]^T \cdots A_1[i_1]^T A_1[i_1] \cdots A_L[i_L] \\
&= \sum_{i_1=1}^{n_1} \cdots \sum_{i_L=1}^{n_L} \mathrm{Tr}\left(A_L[i_L]^T \cdots A_1[i_1]^T A_1[i_1] \cdots A_L[i_L]\right) \\
&= \sum_{i_1=1}^{n_1} \cdots \sum_{i_L=1}^{n_L} \mathrm{Tr}\left(A_{k+1}[i_{k+1}] \cdots A_L[i_L] A_L[i_L]^T \cdots A_1[i_1]^T A_1[i_1] \cdots A_k[i_k]\right) \\
&= \sum_{i_k=1}^{n_k} \mathrm{Tr}\left(A_k[i_k]^T A_k[i_k]\right).
\end{aligned}$$

Conversion between left and right orthogonal TT representations

By successive LQ decompositions, it is possible to transform a left-orthogonal to a right orthogonal TT decomposition. Let $(A_1, \dots, A_L)$ be a left-orthogonal TT decomposition of $C \in \mathbb{R}^{n_1 \times \cdots \times n_L}$. Then we have

$$\begin{aligned}
C_{i_1 \dots i_L} &= A_1[i_1] \cdots A_L[i_L] \\
&= A_1[i_1]^{\alpha_1} A_2[i_2]_{\alpha_1}^{\alpha_2} \cdots A_{L-1}[i_{L-1}]_{\alpha_{L-2}}^{\alpha_{L-1}} \left(A_L\right)_{\alpha_{L-1}}^{i_L} \\
&= A_1[i_1]^{\alpha_1} A_2[i_2]_{\alpha_1}^{\alpha_2} \cdots A_{L-1}[i_{L-1}]_{\alpha_{L-2}}^{\alpha_{L-1}} \left(L_L\right)_{\alpha_{L-1}}^{\beta_{L-1}} \left(Q_L\right)_{\beta_{L-1}}^{i_L} \\
&= A_1[i_1]^{\alpha_1} A_2[i_2]_{\alpha_1}^{\alpha_2} \cdots A_{L-2}[i_{L-2}]_{\alpha_{L-3}}^{\alpha_{L-2}} \left(A_{L-1} L_L\right)_{\alpha_{L-2}}^{i_{L-1}\beta_{L-1}} \left(Q_L\right)_{\beta_{L-1}}^{i_L} \\
&= A_1[i_1]^{\alpha_1} A_2[i_2]_{\alpha_1}^{\alpha_2} \cdots A_{L-2}[i_{L-2}]_{\alpha_{L-3}}^{\alpha_{L-2}} \left(L_{L-1}\right)_{\alpha_{L-2}}^{\beta_{L-2}} \left(Q_{L-1}\right)_{\beta_{L-2}}^{i_{L-1}\beta_{L-1}} \left(Q_L\right)_{\beta_{L-1}}^{i_L},
\end{aligned}$$

we repeat this process until we reach

$$\begin{aligned}
C_{i_1 \dots i_L} &= (A_1 L_2)^{i_1 \beta_1} \quad \left(Q_2\right)_{\beta_1}^{i_2 \beta_2} \quad \cdots \quad \left(Q_{L-1}\right)_{\beta_{L-2}}^{i_{L-1}\beta_{L-1}} \quad \left(Q_L\right)_{\beta_{L-1}}^{i_L} \\
&= \quad B_1[i_1]_{\beta_1} \quad B_2[i_2]_{\beta_2}^{\beta_1} \quad \cdots \quad B_{L-1}[i_{L-1}]_{\beta_{L-1}}^{\beta_{L-2}} \quad B_L[i_L]^{\beta_{L-1}}.
\end{aligned}$$

We simply need to check that the TT cores $B_2, \dots, B_L$ are right-orthogonal:

$$\sum_{i_k=1}^{n_k} B_k[i_k] B_k[i_k]^* = \mathrm{id}_{r_{k-1}} .$$

Theorem 1.2.7 (Characterisation of the TT ranks [HRS12b]). *Let $C \in \mathbb{R}^{n_1 \times \cdots \times n_L}$ be a tensor. Then the following assertions are true:*

(i). the HSVD algorithm given in Section 1.2.2 gives a TT decomposition of minimal TT rank;

(ii). the minimal TT rank $(r_1, \dots, r_{L-1})$ is equal to the rank of the reshapes of C, i.e.

$$r_k = \mathrm{Rank}\, C_{i_1 \dots i_k}^{i_{k+1} \dots i_L}. \tag{1.2.7}$$

Proof. Let $(A_1, \dots, A_L)$ be the TT cores given by the HSVD algorithm. The proof of item (ii) follows from the following identity

$$C_{i_1 \dots i_k}^{i_{k+1} \dots i_L} = \big(A_1[i_1] A_2[i_2] \cdots A_k[i_k]\big)\big(A_{k+1}[i_{k+1}] \ \cdots \ A_L[i_L]\big),$$

where $\big(A_1[i_1] A_2[i_2] \cdots A_k[i_k]\big) \in \mathbb{R}^{n_1 \cdots n_k \times r_k}$ and $\big(A_{k+1}[i_{k+1}] \ \cdots \ A_L[i_L]\big) \in \mathbb{R}^{r_k \times n_{k+1} \cdots n_L}$. By construction and by the property of the SVD, both matrices are full rank, hence $r_k = \mathrm{Rank}\, C_{i_1 \dots i_k}^{i_{k+1} \dots i_L}$. □

These normalisations have the advantage of reducing the gauge freedom in the TT representation.

Proposition 1.2.8 (Gauge freedom of left-orthogonal TT decompositions [HRS12b]). *A left-orthogonal TT representation of minimal TT rank $(r_1, \dots, r_{L-1})$ is unique up to the insertion of orthogonal matrices,* i.e. *if $(A_1, \dots, A_L)$ and $(B_1, \dots, B_L)$ are left-orthogonal TT representations of the same tensor C, then there are orthogonal matrices $(Q_k)_{1 \le k \le L-1}$, $Q_k \in \mathbb{R}^{r_k \times r_k}$ such that for all $1 \le i_k \le n_k$ we have*

$$\begin{aligned} A_1[i_1] Q_1 = B_1[i_1], \quad Q_{L-1}^* A_L[i_L] = B_L[i_L] \\ Q_{k-1}^* A_k[i_k] Q_k = B_k[i_k], \ \text{for } k = 2, \dots, L-1. \end{aligned} \tag{1.2.8}$$

Proof. The proof relies on the following observation: let $M_1, N_1 \in \mathbb{R}^{p \times r}$ and $M_2, N_2 \in \mathbb{R}^{r \times q}$ be matrices of rank r such that

$$M_1 M_2 = N_1 N_2 \quad \text{and} \quad M_1^* M_1 = N_1^* N_1 = \mathrm{id}_r,$$

there is an orthogonal matrix $Q \in \mathbb{R}^{r \times r}$ such that

$$M_1 = N_1 Q \quad \text{and} \quad M_2 = Q^* N_2.$$

The proof of this lemma is straightforward:

$$N_2 = N_1^* M_1 M_2 = N_1^* M_1 M_1^* N_1 N_2,$$

which shows that $N_1^* M_1$ is an orthogonal matrix. Denote this matrix Q. Hence $N_2 = QM_2$ and $M_1 N_1^* N_1 = M_1$ thus, $N_1 = M_1 Q^*$.

The proof then goes by iteration. We have

$$\begin{aligned} \big(A_1[i_1]\big)\big(A_2[i_2]\cdots A_L[i_L]\big) &= \big(B_1[i_1]\big)\big(B_2[i_2]\cdots B_L[i_L]\big) \\ \sum_{i_1=1}^{n_1} A_1[i_1]^* A_1[i_1] &= \sum_{i_1=1}^{n_1} B_1[i_1]^* B_1[i_1] = \mathrm{id}_{r_1}. \end{aligned}$$

Since $\big(A_1[i_1]\big), \big(A_2[i_2]\cdots A_L[i_L]\big), \big(B_1[i_1]\big)$ and $\big(B_2[i_2]\cdots B_L[i_L]\big)$ have rank r_1, by the lemma there is an orthogonal matrix $Q_1 \in \mathbb{R}^{r_1 \times r_1}$ such that

$$\begin{aligned} A_1[i_1]Q_1 &= B_1[i_1] \\ Q_1^*\big(A_2[i_2]\cdots A_L[i_L]\big) &= \big(B_2[i_2]\cdots B_L[i_L]\big). \end{aligned}$$

For the next iteration, we have

$$\begin{aligned} \big(Q_1^* A_2[i_2]\big)\big(A_3[i_3]\cdots A_L[i_L]\big) &= \big(B_2[i_2]\big)\big(B_3[i_3]\cdots B_L[i_L]\big) \\ \sum_{i_2=1}^{n_2} A_2[i_2]^* Q_1 Q_1^* A_2[i_2] &= \sum_{i_2=1}^{n_2} B_2[i_2]^* B_2[i_2] = \mathrm{id}_{r_1}. \end{aligned}$$

Applying again the lemma, we have

$$\begin{aligned} Q_1^* A_2[i_2] Q_2 &= B_2[i_2] \\ Q_2^*\big(A_3[i_3]\cdots A_L[i_L]\big) &= \big(B_3[i_3]\cdots B_L[i_L]\big). \end{aligned}$$

By iteration, we prove the proposition. □

The Vidal representation

A convenient - albeit numerically unstable - way to convert easily between left-orthogonal and right-orthogonal TT representations is to use the Vidal representation [Vid03].

Definition 1.2.9 (Vidal representation [Vid03])**.** *Let $C \in \mathbb{R}^{n_1 \times \cdots \times n_L}$ be a tensor. We say that $(\Gamma_k)_{1 \le k \le L}$, $(\Sigma_k)_{1 \le k \le L-1}$ is a Vidal representation if Σ_k are diagonal matrices with positive entries,*

$$C_{i_1,\dots,i_L} = \Gamma_1[i_1]\Sigma_1\Gamma_2[i_2]\Sigma_2\cdots\Sigma_{L-1}\Gamma_L[i_L], \tag{1.2.9}$$

and the matrices $\Gamma_k[i_k] \in \mathbb{R}^{r_{k-1}\times r_k}$ *satisfy*

$$\sum_{i_1=1}^{n_1} \Gamma_1[i_1]^*\Gamma_1[i_1] = \mathrm{id}_{r_1}, \quad \sum_{i_L=1}^{n_L} \Gamma_L[i_L]\Gamma_L[i_L]^* = \mathrm{id}_{r_{L-1}} \tag{1.2.10}$$

$$\forall\, k = 2,\dots,L-1,\ \sum_{i_k=1}^{n_k} \Gamma_k[i_k]^*\Sigma_{k-1}^2\Gamma_k[i_k] = \mathrm{id}_{r_k}, \quad \sum_{i_k=1}^{n_k} \Gamma_k[i_k]\Sigma_k^2\Gamma_k[i_k]^* = \mathrm{id}_{r_{k-1}}. \tag{1.2.11}$$

The Vidal representation directly gives left and right orthogonal TT decompositions:

(i). $(A_1,\dots,A_L)$ left-orthogonal TT representation

$$\begin{aligned} A_1[i_1] &= \Gamma_1[i_1], \quad A_L[i_L] = \Sigma_{L-1}\Gamma_L[i_L] \\ A_k[i_k] &= \Sigma_{k-1}\Gamma_k[i_k], \quad k = 2,\dots,L-1; \end{aligned}$$

(ii). $(B_1,\dots,B_L)$ right-orthogonal TT representation

$$\begin{aligned} B_1[i_1] &= \Gamma_1[i_1]\Sigma_1, \quad B_L[i_L] = \Gamma_L[i_L] \\ B_k[i_k] &= \Gamma_k[i_k]\Sigma_k, \quad k = 2,\dots,L-1. \end{aligned}$$

The conversion from left (or right) orthogonal decomposition to a Vidal representation is more involved [Sch11, Section 4.6]. Let A_k be the TT components of a left-orthogonal TT representation. Notice that for all k, let Σ_k be the singular values of the tensor reshape $C_{i_1\dots i_k}^{i_{k+1}\dots i_d}$. Then we have

$$C_{i_1\dots i_k}^{i_{k+1}\dots i_L} = \underbrace{\begin{bmatrix} A_1[1]A_2[1]\cdots A_k[1] \\ \vdots \\ A_1[n_1]A_2[n_2]\cdots A_k[n_k] \end{bmatrix}}_{=:M_k\in\mathbb{R}^{n_1\cdots n_k\times r_k}} \underbrace{\begin{bmatrix} A_{k+1}[i_{k+1}] & \cdots & A_L[i_L] \end{bmatrix}}_{\in\mathbb{R}^{r_k\times n_{k+1}\cdots n_L}}$$

Because A_k are left-orthogonal, then $M_k^T M_k = \mathrm{id}_{r_k}$, hence the singular values of the reshaped tensor is exactly the singular values of the right matrix.

With this remark, we can now write the iterative algorithm to get the Vidal representation of the tensor.

Algorithm 1 Left-orthogonal to Vidal representation

Input: $(A_1, \dots, A_L)$ left-orthogonal TT representation
Output: $(\Gamma_1, \dots, \Gamma_L), (\Sigma_1, \dots, \Sigma_{L-1})$ Vidal representation

```
function LeftToVidal((A_1, ..., A_L))
    U_{L-1}, Σ_{L-1}, V_L^T = svd( [A_L[1] A_L[2] ··· A_L[n_L]] )
    [Γ_L[1] ··· Γ_L[n_L]] = V_L^T
    for k = L-1, ..., 1 do
        U_{k-1}, Σ_{k-1}, V_k^T = svd( [A_k[1]U_kΣ_k ··· A_k[n_k]U_kΣ_k] ).
        Γ_k solution to V_k^T = [Γ_k[1]Σ_k ··· Γ_k[n_k]Σ_k]
    end for
    return (Γ_1, ..., Γ_L), (Σ_1, ..., Σ_{L-1}).
end function
```

By induction, one can show that the singular values of the successive SVD in the previous algorithm are indeed the singular values of the tensor reshape.

Proposition 1.2.10. *Let $(\Gamma_k)_{1 \leq k \leq L}$, $(\Sigma_k)_{1 \leq k \leq L-1}$ be a Vidal representation of $C \in \mathbb{R}^{n_1 \times \cdots \times n_L}$. Then Σ_k is the matrix of the singular values of the reshape $C_{i_1 \dots i_k}^{i_{k+1} \dots i_L} \in \mathbb{R}^{n_1 \cdots n_k \times n_{k+1} \cdots n_L}$.*

Proof. By definition of the SVD, the Vidal TT components Γ_k satisfy

$$\sum_{i_k=1}^{n_k} \Gamma_k[i_k] \Sigma_k^2 \Gamma_k[i_k]^T = \mathrm{id}_{r_{k-1}} .$$

We also have

$$\begin{bmatrix} A_k[1]U_k & \cdots A_k[n_k]U_k \end{bmatrix} = \begin{bmatrix} U_{k-1}\Sigma_{k-1}\Gamma_k[1] & \cdots & U_{k-1}\Sigma_{k-1}\Gamma_k[n_k] \end{bmatrix} .$$

Thus

$$\begin{aligned} \sum_{i_k}^{n_k} \Gamma_k[i_k]^T \Sigma_{k-1}^2 \Gamma_k[i_k] &= \sum_{i_k}^{n_k} \Gamma_k[i_k]^T \Sigma_{k-1} U_{k-1}^T U_{k-1} \Sigma_{k-1} \Gamma_k[i_k] \\ &= \sum_{i_k}^{n_k} U_k^T A_k[i_k]^T A_k[i_k] U_k \\ &= \mathrm{id}_{r_k} . \end{aligned}$$

□

1.3 Approximation by tensor trains

A natural way to reduce the TT ranks of the TT representation of a tensor is to truncate the SVD at each step of the HSVD algorithm to a tolerance ε:

$$\begin{aligned}
C_{i_1\dots i_L} &= C_{i_1}^{i_2\dots i_L} && \text{(reshape of } C \text{ to } n_1 \times n_2 \cdots n_L) \\
&\simeq (U_1)_{i_1}^{\alpha_1} (\Sigma_1^\varepsilon V_1^T)_{\alpha_1}^{i_2\dots i_L} && \text{(truncated SVD)} \\
&\simeq (U_1)_{i_1}^{\alpha_1} (\Sigma_1^\varepsilon V_1^T)_{\alpha_1 i_2}^{i_3\dots i_L} && \text{(reshape of } \Sigma_1^\varepsilon V_1^T) \\
&\simeq (U_1)_{i_1}^{\alpha_1} (U_2)_{\alpha_1 i_2}^{\alpha_2} (\Sigma_2^\varepsilon V_2^T)_{\alpha_2}^{i_3\dots i_L} && \text{(truncated SVD of } \Sigma_1^\varepsilon V_1^T) \\
&\simeq (U_1)_{i_1}^{\alpha_1} (U_2)_{\alpha_1 i_2}^{\alpha_2} (\Sigma_2^\varepsilon V_2^T)_{\alpha_2 i_3}^{i_4\dots i_L} && \text{(reshape of } \Sigma_2^\varepsilon V_2^T),
\end{aligned}$$

we repeat the process until we get

$$C_{i_1\dots i_L} \simeq (U_1)_{i_1}^{\alpha_1} (U_2)_{\alpha_1 i_2}^{\alpha_2} \cdots (U_{L-1})_{\alpha_{L-2} i_{L-1}}^{\alpha_{L-1}} (\Sigma_{L-1}^\varepsilon V_{L-1})_{\alpha_{L-1}}^{i_L}.$$

This algorithm is often called a *TT rounding* [Ose11] or *TT compression*. Truncating the successive SVDs gives an estimate on the best approximation by a tensor train of fixed TT ranks.

Theorem 1.3.1 ([Gra10, Ose11, Hac12, Hac14])**.** *Let $C \in \mathbb{R}^{n_1\times\cdots\times n_L}$, $(\tilde r_1, \dots, \tilde r_{L-1}) \in \mathbb{N}^{L-1}$ and $\mathcal{M}_{\tilde{\mathbf{r}}}$ be the space of tensor trains of ranks bounded by $(\tilde r_1, \dots, \tilde r_{L-1})$. Then we have*

$$\min_{V\in\mathcal{M}_{\tilde{\mathbf{r}}}} \|C - V\| \le \sqrt{\sum_{k=1}^{L-1} \sum_{j>\tilde r_k} {\sigma_j^{(k)}}^2} \le \sqrt{L-1} \min_{V\in\mathcal{M}_{\tilde{\mathbf{r}}}} \|C - V\|,$$

where for $1 \le k \le L-1$, $(\sigma_j^{(k)})_{1\le j\le r_k}$ are the singular values of the reshape $(\Psi^{\mu_1\cdots\mu_k}_{\mu_{k+1}\cdots\mu_L})$.

Proof. The proof of the left-hand side inequality follows from the HSVD algorithm. Let $\Pi_k : \mathbb{R}^{n_1\cdots n_k\times n_{k+1}\cdots n_L} \to \mathbb{R}^{n_1\cdots n_k\times n_{k+1}\cdots n_L}$ be the SVD truncation of rank $\tilde r_k$. This operator is an orthogonal projection in the Hilbert space $\mathbb{R}^{n_1\cdots n_k\times n_{k+1}\cdots n_L}$ equipped with the Frobenius norm. The HSVD algorithm with truncation at each step is the tensor $\Pi_{L-1}\cdots\Pi_1 C$. We thus have using the property of the SVD truncation:

$$\begin{aligned}
\|C - \Pi_{L-1}\cdots\Pi_1 C\|_F^2 &\le \|\Pi_{L-1}^\perp C\|^2 + \|\Pi_{L-1} C - \Pi_{L-1}\cdots\Pi_1 C\|_F^2 \\
&\le \sum_{j>\tilde r_k} {\sigma_j^{(k)}}^2 + \|C - \Pi_{L-2}\cdots\Pi_1 C\|_F^2,
\end{aligned}$$

hence by iteration

$$\|C - \Pi_{L-1}\cdots\Pi_1 C\|_F^2 \le \sum_{k=1}^{L-1} \sum_{j>\tilde r_k} {\sigma_j^{(k)}}^2.$$

This provides a bound on the best approximation by a tensor train in $\mathcal{M}_{\tilde{\mathbf{r}}}$.

For the lower bound on the best approximation C_{best}, we have for each k by definition of the SVD truncation

$$\|C - \Pi_k C\|_F^2 = \sum_{j > \tilde{r}_k} {\sigma_j^{(k)}}^2 \leq \|C - C_{\text{best}}\|_F^2,$$

hence by summing over k we get the lower bound. □

A drawback of the HSVD algorithm or its truncated version is that it requires to handle the full tensor. If the tensor is already in a TT format, it is possible to reduce the cost of this truncation. Let $(A_1, \dots, A_L)$ be a right-orthogonal TT representation of the tensor $C \in \mathbb{R}^{n_1 \times \dots \times n_L}$. The first reshape is

$$C_{i_1}^{i_2 \dots i_L} = \begin{bmatrix} A_1[1] \\ \vdots \\ A_1[n_1] \end{bmatrix} \begin{bmatrix} A_2[1] \cdots A_L[1] & \cdots & A_2[n_2] \cdots A_L[n_L] \end{bmatrix},$$

and since the TT cores $(A_2, \dots, A_L)$ are right-orthogonal, the matrix $V_2 = \begin{bmatrix} A_2[1] \cdots A_L[1] & \cdots & A_2[n_2] \cdots A_L[n_k] \end{bmatrix}$ satisfies $V_2 V_2^* = \text{id}_{r_1}$. Hence the first step of the HSVD truncation can be reduced to the SVD of the reshape of A_1. The same would hold for the next step of the HSVD truncation, hence the total cost of the TT compression of C in a TT format is reduced to $\mathcal{O}(Lr^3)$ where $r = \max(r_k)$.

The algorithm is summarised in Algorithm 2.

1.4 Manifold of tensor trains

Even in finite-dimensions, the example exhibited in eq. (1.1.5) shows that the set

$$\mathcal{M}_{\text{CP}_{\leq r}} = \left\{ C = \sum_{i=1}^r v_1^{(i)} \otimes \cdots \otimes v_L^{(i)}, \forall 1 \leq i \leq r, 1 \leq j \leq L, v_j^{(i)} \in \mathbb{R}^{n_j} \right\},$$

is not closed if $L \geq 3$.

For tensor trains, the question of the closedness has a clear answer, as the characterisation of the TT rank relies on the matricisation of the tensor.

Proposition 1.4.1. *The set of tensor trains with TT rank less that r*

$$\mathcal{M}_{\text{TT}_{\leq r}} = \left\{ C \mid \forall 1 \leq i_k \leq n_k,\ C_{i_1 \dots i_L} = A_1[i_1] \cdots A_L[i_L],\ A_k[i_k] \in \mathbb{R}^{r_{k-1} \times r_k}, r_k \leq r \right\},$$

is a closed set.

Algorithm 2 TT rounding algorithm

Input: $(A_1, \dots, A_L)$ right-orthogonal TT representation, $\varepsilon > 0$ tolerance
Output: $(A_1^\varepsilon, \dots, A_L^\varepsilon)$ TT representation such that $\|\mathrm{TT}(A_i^\varepsilon) - \mathrm{TT}(A_i)\|_F \leq \sqrt{L-1}\,\varepsilon$

function HSVD($(A_1, \dots, A_L), \varepsilon$)
  for $k = 1, \dots, L-1$ **do**
    $U_k, \Sigma_k, V_k^T = \mathtt{svd}\Big(\begin{bmatrix} A_k[1] \\ \vdots \\ A_k[n_k] \end{bmatrix} \Big)$
    $r_k = \underset{r}{\arg\max} \|\Sigma_k[1:r] - \Sigma_k\| \leq \varepsilon$
    $(A_k^\varepsilon)_{i_k \alpha_{k-1}}^{\alpha_k} = (U_k)_{i_k \alpha_{k-1}}^{\alpha_k}, \quad i_k = 1, \dots, n_k, \alpha_{k-1} = 1, \dots, r_{k-1}, \alpha_k = 1, \dots, r_k$
    $A_{k+1}[i_{k+1}] = \Sigma_k[1:r] V_k^T[1:r,:] A_{k+1}[i_{k+1}], \quad i_{k+1} = 1, \dots, n_{k+1}$
  end for
  $A_L^\varepsilon = A_L$
  return $(A_1^\varepsilon, \dots, A_L^\varepsilon)$
end function

Proof. The proof follows from the characterisation of the TT ranks given by Theorem 1.2.7: given a tensor C, for $1 \leq k \leq L-1$, the minimal TT rank r_k is equal to the rank of the matrix $C_{i_1 \dots i_k}^{i_{k+1} \dots i_L}$. We conclude by recalling that the set of matrices with rank less than r is a closed set. □

Proposition 1.4.2. *The set of tensor trains with TT rank* $\mathbf{r} = (r_1, \dots, r_{L-1})$

$$\mathcal{M}_{\mathrm{TT}_\mathbf{r}} = \left\{ C \mid \forall\, 1 \leq i_k \leq n_k,\ C_{i_1 \dots i_L} = A_1[i_1] \cdots A_L[i_L],\ A_k[i_k] \in \mathbb{R}^{r_{k-1} \times r_k} \right\},$$

is of dimension

$$\dim \mathcal{M}_{\mathrm{TT}_\mathbf{r}} = \sum_{i=1}^{L} r_{i-1} n_i r_i - \sum_{i=1}^{L-1} r_i^2. \tag{1.4.1}$$

Proof. Two TT representations $(A_1, \dots, A_L)$ and $(\tilde{A}_1, \dots, \tilde{A}_L)$ of a same tensor are related by a gauge $(G_1, \dots, G_{L-1}) \in \mathrm{GL}_{r_1}(\mathbb{R}) \times \cdots \mathrm{GL}_{r_{L-1}}(\mathbb{R})$

$$\forall\, 1 \leq i_k \leq n_k, A_k[i_k] = G_{k-1} \tilde{A}_k[i_k] G_k, \quad k = 1, \dots, L, \quad (G_0 = G_L = 1).$$

The dimension of $\mathrm{GL}_{r_k}(\mathbb{R})$ is r_k^2, hence the dimension of $\mathcal{M}_{\mathrm{TT}_\mathbf{r}}$ is

$$\dim \mathcal{M}_{\mathrm{TT}_\mathbf{r}} = \sum_{i=1}^{L} r_{i-1} n_i r_i - \sum_{i=1}^{L-1} r_i^2.$$

□

Proposition 1.4.3 (Tangent space of $\mathcal{M}_{\mathrm{TT}_\mathbf{r}}$ [HRS12b]). *Let $A \in \mathcal{M}_{\mathrm{TT}_\mathbf{r}}$ and $(A_1, \dots, A_L)$ be a left-orthogonal TT representation of A. Let $\delta A \in \mathcal{T}_A \mathcal{M}_{\mathrm{TT}_\mathbf{r}}$.*

There are unique components $(W_k)_{1 \le k \le L} \in \bigotimes_{k=1}^L \mathbb{R}^{r_{k-1} \times n_k \times r_k}$ such that

$$\delta A = \sum_{k=1}^{L} \delta A^{(k)}, \tag{1.4.2}$$

with

$$\delta A^{(k)}_{i_1 \dots i_L} = A_1[i_1] \cdots A_{k-1}[i_{k-1}] W_k[i_k] A_{k+1}[i_{k+1}] \cdots A_L[i_L], \tag{1.4.3}$$

and where for $k = 1, \dots, L-1$ we have

$$\sum_{i_k=1}^{n_k} A_k[i_k]^T W_k[i_k] = \mathbf{0}_{r_k \times r_k}. \tag{1.4.4}$$

Proof. By definition of the tangent space $\mathcal{T}_A \mathcal{M}_{\mathrm{TT}_\mathbf{r}}$, the tangent vectors are given by the derivatives $\dot{\Gamma}$ of the differentiable curves $\Gamma : \mathbb{R} \to \mathscr{M}_{\mathrm{TT}_\mathbf{r}}$ such that $\Gamma(0) = A$.

For all $t \in \mathbb{R}$, since $\Gamma(t) \in \mathcal{M}_{\mathrm{TT}_\mathbf{r}}$, we can choose a left-orthogonal TT representation of $\Gamma(t)$ such that

$$\Gamma(t)_{i_1 \dots i_L} = \Gamma_1^{(t)}[i_1] \cdots \Gamma_L^{(t)}[i_L],$$

where for all $1 \le k \le L$, $t \mapsto \Gamma_k^{(t)} \in \mathbb{R}^{n_k \times r_{k-1} \times r_k}$ is differentiable and $\Gamma_k^{(0)} = A_k$.

Since for $1 \le k \le L-1$, $\sum_{i_k=1}^{n_k} \Gamma_k^{(t)}[i_k]^T \Gamma_k^{(t)}[i_k] = \mathrm{id}_{r_k}$, there is a differentiable function $t \mapsto U_k(t) \in \mathcal{O}_{n_k r_{k-1}}(\mathbb{R})$ such that

$$\begin{bmatrix} \Gamma_k^{(t)}[1] \\ \vdots \\ \Gamma_k^{(t)}[n_k] \end{bmatrix} = U_k(t) \begin{bmatrix} A_k[1] \\ \vdots \\ A_k[n_k] \end{bmatrix}.$$

This implies that $\begin{bmatrix} \dot{\Gamma}_k^{(0)}[1] \\ \vdots \\ \dot{\Gamma}_k^{(0)}[n_k] \end{bmatrix} = S_k \begin{bmatrix} A_k[1] \\ \vdots \\ A_k[n_k] \end{bmatrix}$ for some antisymmetric matrix $S_k \in \mathbb{R}^{n_k r_{k-1} \times n_k r_{k-1}}$.

Let

$$\begin{bmatrix} W_k[1] \\ \vdots \\ W_k[n_k] \end{bmatrix} = S_k \begin{bmatrix} A_k[1] \\ \vdots \\ A_k[n_k] \end{bmatrix}.$$

Then

$$\sum_{i_k=1}^{n_k} A_k[i_k]^T W_k[i_k] = \begin{bmatrix} A_k[1]^T & \dots & A_k[n_k]^T \end{bmatrix} S_k \begin{bmatrix} A_k[1] \\ \vdots \\ A_k[n_k] \end{bmatrix},$$

which is a symmetric and an antisymmetric matrix, hence it is zero.

The tangent vectors are hence necessarily of the form given by eq. (1.4.2)-(1.4.4). By dimension counting and invoking Proposition 1.4.2 shows the uniqueness of the representation. □

Chapter 2

DMRG

Density matrix renormalisation group [Whi92] (DMRG) is an alternating scheme to solve linear problems or eigenvalue problems in the tensor train format. In the mathematical community, it is also referred to the *alternating linear scheme* (ALS) in its simplest version or to the *modified ALS (MALS)* [HRS12a], which is the equivalent to the two-site DMRG. In DMRG, given a symmetric matrix $H \in \mathbb{R}^{n_1 \cdots n_L \times n_1 \cdots n_L}$, we want to solve for $x \in \mathbb{R}^{n_1 \cdots n_L}$ the linear problem

$$Hx = b, \tag{2.0.1}$$

for a given $b \in \mathbb{R}^{n_1 \cdots n_L}$, or for $(\lambda, x) \in \mathbb{R} \times \mathbb{R}^{n_1 \cdots n_L}$ the lowest eigenvalue problem

$$Hx = \lambda x. \tag{2.0.2}$$

For both problems, a tensor train representation of the operator H is needed in order to efficiently implement the DMRG algorithm.

2.1 Tensor train operators

2.1.1 Definition and graphical representation

Definition 2.1.1 (Tensor train operator)**.** *Let $H \in \mathbb{R}^{n_1 \cdots n_L \times n_1 \cdots n_L}$ be a matrix. A tensor train operator (TTO) representation of the matrix is any tuple of order 4 tensors $(H_1, \ldots, H_L)$, $H_k \in \mathbb{R}^{n_k \times n_k \times R_{k-1} \times R_k}$ ($R_0 = R_L = 1$) such that*

$$H_{i_1 \ldots i_L}^{j_1 \ldots j_L} = H_1[i_1, j_1] \cdots H_L[i_L, j_L], \forall\, i_k, j_k = 1, \ldots, n_k.$$

The diagrammatic representation of a TTO is similar to the diagrammatic of a TT as illustrated in Figure 2.1.

A TTO representation of a matrix can be obtained by reordering the indices of the matrix H and performing a TT-SVD of the resulting tensor. More precisely, by defining

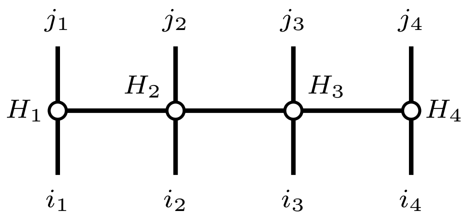


Figure 2.1: Diagrammatic representation of a TTO

the tensor $\widetilde{H} \in \mathbb{R}^{n_1^2 \times \cdots \times n_L^2}$

$$\widetilde{H}_{i_1 j_1; \dots ; i_L j_L} = H^{j_1 \dots j_L}_{i_1 \dots i_L},$$

we realise that a TTO representation is simply a TT representation of $\widetilde{H}$.

Proposition 2.1.2. *Let $H \in \mathbb{R}^{n_1 \cdots n_L \times n_1 \cdots n_L}$ be a symmetric matrix. Then there is a TTO representation of H such that*

$$\forall\, 1 \leq i_k, j_k \leq n_k, \;\; H_k[i_k, j_k] = H_k[j_k, i_k], \quad k = 1, \dots, L. \tag{2.1.1}$$

Proof. □

Example 2.1.3. *Let us consider the following matrix $H \in \mathbb{R}^{n^L \times n^L}$*

$$H = h \otimes \mathrm{id} \otimes \cdots \otimes \mathrm{id} + \cdots + \mathrm{id} \otimes \mathrm{id} \otimes \cdots \otimes h, \tag{2.1.2}$$

where $h \in \mathbb{R}^{n \times n}$ is a symmetric matrix and id is the identity in $\mathbb{R}^{n \times n}$. The matrix $h \otimes \mathrm{id} \otimes \cdots \otimes \mathrm{id}$ is in fact a TTO of rank 1. A naïve application of Proposition 2.1.4 yields a TTO representation of H of rank L. However it is possible to achieve a rank 2 representation with the following construction

$$\begin{aligned} H_1[i_1, j_1] &= \begin{pmatrix} h_{i_1 j_1} & \delta_{i_1 j_1} \end{pmatrix}, \quad H_L[i_L, j_L] = \begin{pmatrix} \delta_{i_L j_L} \\ h_{i_L j_L} \end{pmatrix} \\ H_k[i_k, j_k] &= \begin{pmatrix} \delta_{i_k j_k} & 0 \\ h_{i_k j_k} & \delta_{i_k j_k} \end{pmatrix}, \quad k = 2, \dots, L-1. \end{aligned} \tag{2.1.3}$$

Note that this representation also satisfies the property stated in Proposition 2.1.2.

2.1.2 Algebraic properties

Like the TT representation of vectors, the TTO format has some algebraic stability property.

Proposition 2.1.4. *Let $G, H \in \mathbb{R}^{n_1 \cdots n_L \times n_1 \cdots n_L}$ be matrices and $(G_1, \dots, G_L)$, $G_k \in \mathbb{R}^{n_k \times n_k \times R^G_{k-1} \times R^G_k}$ and $(H_1, \dots, H_L)$, $H_k \in \mathbb{R}^{n_k \times n_k \times R^H_{k-1} \times R^H_k}$ be respectively TTO representations of G and H. Let $A, B \in \mathbb{R}^{n_1 \cdots n_L}$ be vectors with respcetive TT representations $(A_1, \dots, A_L)$, $A_k \in \mathbb{R}^{n_k \times r^A_{k-1} \times r^A_k}$, $(B_1, \dots, B_L)$, $B_k \in \mathbb{R}^{n_k \times r^B_{k-1} \times r^B_k}$. Then* *(i). the sum $G+H$ has a TTO representation $(S_1,\dots,S_L)$ given by*

$$S_1[i_1,j_1] = \begin{pmatrix} G_1[i_1,j_1] & H_1[i_1,j_1] \end{pmatrix}, \quad S_L[i_L,j_L] = \begin{pmatrix} G_L[i_L,j_L] \\ H_L[i_L,j_L] \end{pmatrix}$$
$$S_k[i_k,j_k] = \begin{pmatrix} G_k[i_k,j_k] & 0 \\ 0 & H_k[i_k,j_k] \end{pmatrix}, k=2,\dots,L-1 \tag{2.1.4}$$

(ii). the matrix-vector product $C = HA$ has a TT representation $(C_1,\dots,C_L)$ with $C_k[j_k] \in \mathbb{R}^{R^H_{k-1}r^A_{k-1}\times R^H_k r^A_k}$

$$C_k[i_k] = \sum_{j_k=1}^{n_k} H_k[i_k,j_k] \otimes A_k[j_k], \quad k=1,\dots,L. \tag{2.1.5}$$

(iii). the product GH has a TTO representation $(P_1,\dots,P_L)$ given by

$$P_k[i_k,j_k] = \sum_{\ell_k=1}^{n_k} G_k[i_k,\ell_k] \otimes H_k[\ell_k,j_k], \quad k=1,\dots,L. \tag{2.1.6}$$

Proof. This is a direct computation. □

Remark 2.1.5. *The TTO representations of the sum and the product of the operators are not optimal. This is clear in the case of the sum $G+H$ when we consider $G=H$. A TT rounding step is required in order to reduce the TTO ranks of the representation. This is not innocuous as essential properties of the matrix can be lost in the rounding procedure (symmetry for instance).*

A diagrammatic proof of the formula for the product of two TTO representations is given in Figure 2.2, avoiding cumbersome computations.

2.2 The DMRG algorithm

The DMRG algorithm is an algorithm to solve linear systems $Hx_* = b$ or the lowest eigenvalue problem $Hx_* = \lambda x_*$ using the variational characterisation of the solution to both problems. As such it is limited in the resolution of linear problems with *symmetric* matrices. In the following, we assume that H is a symmetric, positive-definite matrix.

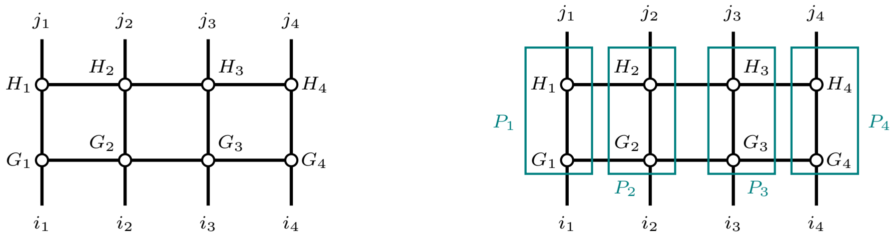
Assumption 2.2.1. *The matrix $H \in \mathbb{R}^{n_1\cdots n_L \times n_1\cdots n_L}$ is symmetric and positive-definite.*

The solution to the linear system $Hx = b$ is also the minimiser of the functional

$$x_* = \operatorname*{arg\,min}_{x\in\mathbb{R}^{n_1\cdots n_L}} \frac{1}{2}\langle x, Hx\rangle - \langle b, x\rangle. \tag{2.2.1}$$

Using the Rayleigh-Ritz principle, the lowest eigenvalue of H is given by

$$x_* = \operatorname*{arg\,min}_{x\in\mathbb{R}^{n_1\cdots n_L}} \frac{\langle x, Hx\rangle}{\langle x, x\rangle}. \tag{2.2.2}$$



(a) Diagrammatic representation of the product of two TTO

(b) Diagrammatic representation of the product of two TTO

Figure 2.2: Diagrammatic proof of the product of two TTO. The left panel is the diagrammatic representation of the product of two TTO. On the right panel, the boxed tensors P_k are the TTO cores of a TTO representation of the product GH, provided that the double edges shared between neighbouring P_k are gathered into one edge.

2.2.1 General algorithm

The DMRG algorithm, also known as *alternating linear scheme* (ALS) [HRS12a], is an alternating optimisation over the TT manifold. The general idea is to perform a descent step for each TT core separately. More precisely, the solution to the linear system $Hx_* = b$ is approximated on the TT manifold

$$\mathscr{M}_{\mathrm{TT}_{\leq r}} = \left\{ C \mid \forall 1 \leq i_k \leq n_k, C_{i_1 \dots i_L} = A_1[i_1] \cdots A_L[i_L], A_k[i_k] \in \mathbb{R}^{r_{k-1} \times r_k}, r_k \leq r \right\}. \tag{2.2.3}$$

Denoting by j the map $J \circ \mathrm{TT}$ where

$$\mathrm{TT} \; : \begin{cases} \mathbb{R}^{n_1 \times r_0 \times r_1} \times \cdots \times \mathbb{R}^{n_L \times r_{L-1} \times r_L} \to \mathbb{R}^{n_1 \cdots n_L} \\ \qquad\qquad (A_1, \dots, A_L) \mapsto (A_1[i_1] \cdots A_L[i_L]), \end{cases}$$

and $J(x) = \frac{1}{2}\langle x, Hx \rangle - \langle b, x \rangle$.

Minimising J over the manifold $\mathscr{M}_{\mathrm{TT}_{\leq r}}$ is the same as minimising the functional j.

The optimisation steps (2.2.4) and (2.2.5) are called *microsteps*. An iteration over the loop n is called a sweep. Notice that at each microstep (2.2.4) or (2.2.5) the left TT cores are left-orthogonal and the right-TT cores are right-orthogonal.

The microsteps of the DMRG algorithm applied to the linear problem $Hx_* = b$ are linear problems involving an operator $P_k : \mathbb{R}^{r_{k-1} \times n_k \times r_k} \to \mathbb{R}^{n_1 \times \cdots \times n_L}$ defined by

$$(P_k V)_{i_1 \dots i_L} = A_1[i_1] \cdots A_{k-1}[i_{k-1}] V[i_k] A_{k+1}[i_{k+1}] \cdots A_L[i_L], \tag{2.2.6}$$

where $(A_1, \dots, A_L)$ are TT cores that are left-orthogonal for $j \leq k-1$ and right-orthogonal for $j \geq k+1$. The tensor $B_k^{(n+1)}$ of the microstep problem (2.2.4) is the solution to the linear system

$$P_k^T A P_k B_k^{(n+1)} = P_k^T b. \tag{2.2.7}$$

Algorithm 3 DMRG with sweeps

Input: $(A_1^{(0)}, \dots, A_L^{(0)})$ in right-orthogonal TT representation
Output: $(A_1^{(n)}, \dots, A_L^{(n)}) \in \mathscr{M}_{\mathrm{TT}_{\leq r}}$ approximation of the minimiser in of J

function DMRG($(A_1^{(0)}, \dots, A_L^{(0)})$)
  $n = 0$
  while `not converged` **do**
    for $k = 1, 2, \dots, L-1$ **do** ▷ Forward half-sweep

$$B_k^{(n+\frac{1}{2})} = \underset{V_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}}{\arg\min} j(A_1^{(n+\frac{1}{2})}, \dots, A_{k-1}^{(n+\frac{1}{2})}, V_k, A_{k+1}^{(n)}, \dots, A_L^{(n)}) \tag{2.2.4}$$

      $Q, R = \mathtt{qr}(\big(B_k^{(n+\frac{1}{2})}\big)_{\alpha_{k-1} i_k}^{\beta_k})$ ▷ QR decomposition
      $\big(A_k^{(n+\frac{1}{2})}[i_k]\big)_{\alpha_{k-1}}^{\alpha_k} = Q_{\alpha_{k-1} i_k}^{\alpha_k}$ ▷ Keep Q
      $\big(A_{k+1}^{(n)}[i_{k+1}]\big)_{\alpha_k}^{\alpha_{k+1}} \leftarrow \big(R A_{k+1}^{(n)}[i_{k+1}]\big)_{\alpha_k}^{\alpha_{k+1}}.$ ▷ Shift R to the right
    end for
    for $k = d, d-1, \dots, 2$ **do** ▷ Backward half-sweep

$$B_k^{(n+1)} = \underset{V_k \in \mathbb{R}^{r_{k-1} \times n_k \times r_k}}{\arg\min} j(A_1^{(n+\frac{1}{2})}, \dots, A_{k-1}^{(n+\frac{1}{2})}, V_k, A_{k+1}^{(n+1)}, \dots, A_L^{(n+1)}) \tag{2.2.5}$$

      $L, Q = \mathtt{lq}\Big(\big(B_k^{(n+1)}\big)_{\alpha_{k-1}}^{\beta_k i_k}\Big)$ ▷ LQ decomposition
      $\big(A_k^{(n+1)}[i_k]\big)_{\alpha_{k-1}}^{\alpha_k} = \big(Q\big)_{\alpha_{k-1}}^{\alpha_k i_k}$ ▷ Keep Q
      $\big(A_{k-1}^{(n+\frac{1}{2})}[i_{k-1}]\big)_{\alpha_{k-2}}^{\alpha_{k-1}} \leftarrow \big(A_{k-1}^{(n+\frac{1}{2})}[i_{k-1}] L\big)_{\alpha_{k-2}}^{\alpha_{k-1}}$ ▷ Shift L to the left
    end for
    $n = n + 1$
  end while
  return $(A_1^{(n)}, \dots, A_L^{(n)})$
end function

Proposition 2.2.2. *Assume that* $(A_i^{(n+\frac{1}{2})})_{1\le i\le k-1}$ *are left-orthogonal and* $(A_i^{(n)})_{k+1\le i\le L}$ *are right-orthogonal. Then the microstep* (2.2.4) *has a unique solution.*

Proof. It is equivalent to check that eq. (2.2.7) has a unique solution, *i.e.* that the matrix $P_k^T H P_k$ is invertible. As H is symmetric and positive-definite, it is sufficient to prove that P_k is an injective operator. Let $V \in \mathbb{R}^{r_{k-1}\times n_k \times r_k}$ such that $\|P_k V\| = 0$. Then we have

$$\begin{aligned}
\|P_k V\|^2 &= \sum_{i_1=1}^{n_1}\cdots\sum_{i_L=1}^{n_L} \operatorname{Tr}\big(A_L[i_L]^T\cdots A_{k+1}[i_{k+1}]^T V[i_k]^T A_{k-1}[i_{k-1}]^T\cdots A_1[i_1]^T \\
&\qquad\qquad A_1[i_1]\cdots A_{k-1}[i_{k-1}]V[i_k]A_{k+1}[i_{k+1}]\cdots A_L[i_L]\big) \\
&= \sum_{i_1=1}^{n_1}\cdots\sum_{i_L=1}^{n_L} \operatorname{Tr}\big(V[i_k]^T A_{k-1}[i_{k-1}]^T\cdots A_1[i_1]^T A_1[i_1]\cdots A_{k-1}[i_{k-1}]V[i_k] \\
&\qquad\qquad A_{k+1}[i_{k+1}]\cdots A_L[i_L]A_L[i_L]^T\cdots A_{k+1}[i_{k+1}]^T\big) \\
&= \sum_{i_k=1}^{n_k} \operatorname{Tr}\big(V[i_k]^T V[i_k]\big),
\end{aligned}$$

where we have used the cyclicity of the trace and the orthogonality of the TT cores. Hence $P_k V = 0$ if and only if $V = 0$. □

2.2.2 Implementation details

2.3 Convergence of DMRG

The global convergence of DMRG is a difficult problem, as the TT manifold is not a convex set. The convergence results on DMRG are local and assume that the Hessian of the functional j is of full-rank.

Assumption 2.3.1. *At the local minimiser* A_**, the Hessian* j'' *is of full rank*

$$\operatorname{rank} j''(A_*) = \sum_{i=1}^{L} r_{i-1}n_i r_i - \sum_{i=1}^{L-1} r_i^2, \quad \textit{i.e.}\ \ker j''(A_*) = T_{A_*}\mathcal{M}_{\mathrm{TT}_{\le r}}. \tag{2.3.1}$$

2.3.1 Local convergence of DMRG

2.3.2 Half-sweep convergence

2.4 Optimisation on the TT-manifold

Bibliography

[BSU16] Markus Bachmayr, Reinhold Schneider, and André Uschmajew. Tensor networks and hierarchical tensors for the solution of high-dimensional partial differential equations. *Found. Comput. Math.*, 16(6):1423–1472, 2016.

[DDGS16] Wolfgang Dahmen, Ronald DeVore, Lars Grasedyck, and Endre Süli. Tensor-sparsity of solutions to high-dimensional elliptic partial differential equations. *Found. Comput. Math.*, 16(4):813–874, 2016.

[DMNS98] Dukelsky, J., Martín-Delgado, M. A., Nishino, T., and Sierra, G. Equivalence of the variational matrix product method and the density matrix renormalization group applied to spin chains. *Europhys. Lett.*, 43(4):457–462, 1998.

[DSL08] Vin De Silva and Lek-Heng Lim. Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM Journal on Matrix Analysis and Applications*, 30(3):1084–1127, 2008.

[Gra10] Lars Grasedyck. Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.*, 31(4):2029–2054, 2009/10.

[Hac12] Wolfgang Hackbusch. *Tensor spaces and numerical tensor calculus*, volume 42 of *Springer Series in Computational Mathematics*. Springer, Heidelberg, 2012.

[Hac14] Wolfgang Hackbusch. Numerical tensor calculus. *Acta Numer.*, 23:651–742, 2014.

[Has07] Matthew B Hastings. An area law for one-dimensional quantum systems. *Journal of statistical mechanics: theory and experiment*, 2007(08):P08024, 2007.

[HRS12a] Sebastian Holtz, Thorsten Rohwedder, and Reinhold Schneider. The Alternating Linear Scheme for Tensor Optimization in the Tensor Train Format. *SIAM Journal on Scientific Computing*, 34(2):A683–A713, January 2012.

[HRS12b] Sebastian Holtz, Thorsten Rohwedder, and Reinhold Schneider. On manifolds of tensors of fixed TT-rank. *Numer. Math.*, 120(4):701–731, 2012.

[KSZ91] A Klumper, A Schadschneider, and J Zittartz. Equivalence and solution of anisotropic spin-1 models and generalized t-j fermion models in one dimension. *Journal of Physics A: Mathematical and General*, 24(16):L955–L959, aug 1991.

[KU16] Daniel Kressner and André Uschmajew. On low-rank approximability of solutions to high-dimensional operator equations and eigenvalue problems. *Linear Algebra Appl.*, 493:556–572, 2016.

[OR95] Stellan Östlund and Stefan Rommer. Thermodynamic limit of density matrix renormalization. *Phys. Rev. Lett.*, 75:3537–3540, Nov 1995.

[Ose11] Ivan V Oseledets. Tensor-train decomposition. *SIAM Journal on Scientific Computing*, 33(5):2295–2317, 2011.

[OT09] I. V. Oseledets and E. E. Tyrtyshnikov. Breaking the curse of dimensionality, or how to use SVD in many dimensions. *SIAM J. Sci. Comput.*, 31(5):3744–3759, 2009.

[Sch08] Erhard Schmidt. On the theory of linear and nonlinear integral equations. III: On the solution of nonlinear integral equations and their bifurcations. *Math. Ann.*, 65:370–399, 1908.

[Sch11] Ulrich Schollwöck. The density-matrix renormalization group in the age of matrix product states. *Ann. Physics*, 326(1):96–192, 2011.

[UV20] André Uschmajew and Bart Vandereycken. Geometric methods on low-rank matrix and tensor manifolds. In *Handbook of variational methods for nonlinear geometric data*, pages 261–313. Cham: Springer, 2020.

[Vid03] Guifré Vidal. Efficient classical simulation of slightly entangled quantum computations. *Phys. Rev. Lett.*, 91:147902, Oct 2003.

[Whi92] Steven R White. Density matrix formulation for quantum renormalization groups. *Physical review letters*, 69(19):2863, 1992.