# A Unifying view on implicit bias in training linear neural networks. - Chulhee Yun et al.

이종진

Seoul National University

*ga0408@snu.ac.kr*

May 20, 2021

# Table of Contents

# Contributions

- Implicit bias gradient flow of the linear tensor networks.
- Consider two cases(separable classification / undeterminded regression)
- Subsume existing results without removing standard convergence assumptions.

# Contributions

- Linear tensor networks / classfication

  $\rightarrow$ Singular vectors of a tensor defined by the network.

- Orthogonally decomposable linear network / classification

  $\rightarrow$ A solution of minimizing $\ell_{2/L}$ max-margin problem in a "transformed" input space defined by the network.

- Orthogonally decomposable linear network / regression

  $\rightarrow$ A solution of minimizing norm-like functions that interpolates between weighted $\ell_1$ and $\ell_2$ in a "transformed" input space.

## Contributions; in a separable classification

- (Thm 1), A linear tensor networks.
  - (Cor 1) A $L-$layer linear fully-connected network
- (Thm 2), A orthogonally decomposable linear network
  - (Cor 2), A $L-$layer linear diagonal network.
  - (Cor 3), A $L-$layer linear full-length convolution network.
- (Thm 3), A 2$-$layer linear network with a single data point (x,y)
  - (Cor 4), A $2-layer$ linear convolutional network with a single data point (x,y)

## Contributions; in a undertermined regression

- (Thm 5), A orthogonally decomposable linear network
  - (Cor 5), A $L-$layer linear diagonal network.
  - (Cor 6), A $L-$layer linear full-length convolution network.
- (Thm 6), A $2-$layer linear network with a single data point (x,y).

# Problem settings

- $\{(x_i, y_i)\}_{i=1}^n$, where $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$
- $\boldsymbol{X} \in \mathbb{R}^{n \times d}$, $y \in \mathbb{R}^n$
- For binary classification,
  - $y_i \in \{\pm 1\}$
  - Data is separable
  - Exponential loss, $\ell(\hat{y}, y) = \exp(-\hat{y}y)$
- For regression
  - Undetermined case $(n \geq d)$
  - Squared error loss, $\ell(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$

# Tensor networks

- A linear map $\mathrm{M}$ that maps $x$ to an order- $L$ tensor $\mathrm{M}(x) \in \mathbb{R}^{k_1 \times \cdots \times k_L}$, where $L \geq 2$.
- A tensor network with parameters $v_l \in \mathbb{R}^{k_l}$ and activation $\phi$,

$$\mathrm{H}_1(\boldsymbol{x}) = \phi\left(\mathrm{M}(\boldsymbol{x}) \circ (\boldsymbol{v}_1, \boldsymbol{I}_{k_2}, \ldots, \boldsymbol{I}_{k_L})\right) \in \mathbb{R}^{k_2 \times \cdots \times k_L}$$

$$\mathrm{H}_l(\boldsymbol{x}) = \phi\left(\mathrm{H}_{l-1}(\boldsymbol{x}) \circ \left(\boldsymbol{v}_l, \boldsymbol{I}_{k_{l+1}}, \ldots, \boldsymbol{I}_{k_L}\right)\right) \in \mathbb{R}^{k_{l+1} \times \ldots, k_L}, \text{ for } l = 2, \ldots, L-1$$

$$f(\boldsymbol{x}; \Theta) = \mathrm{H}_{L-1}(\boldsymbol{x}) \circ v_L \in \mathbb{R}$$

  where $\circ$ is a multilinear multiplication.
- Use $\Theta$ to denote the collection of all parameters $(v_1, \ldots, v_L)$ and name $\mathrm{M}(x)$ as a data tensor.

## A multilinear multiplication

- Given a tensor $\mathrm{A} \in \mathbb{R}^{k_1 \times \cdots \times k_L}$ and linear maps $B_l \in \mathbb{R}^{p_l \times k_l}$ for $l \in [L]$, the multilinear multiplication $\circ$ between them is defined as

$$\mathrm{A} \circ \left(\boldsymbol{B}_1^T, \boldsymbol{B}_2^T, \ldots, \boldsymbol{B}_L^T\right) = \sum_{j_1,\ldots,j_L} [\mathrm{A}]_{j_1,\ldots,j_L} \left(e_{j_1}^{k_1} \otimes \cdots \otimes e_{j_L}^{k_L}\right) \circ \left(\boldsymbol{B}_1^T, \ldots, \boldsymbol{B}_L^T\right)$$

$$:= \sum_{j_1,\ldots,j_L} [\mathrm{A}]_{j_1,\ldots,j_L} \left(\boldsymbol{B}_1 e_{j_1}^{k_1} \otimes \cdots \otimes \boldsymbol{B}_L e_{j_L}^{k_L}\right) \in \mathbb{R}^{p_1 \times \cdots \times p_L}$$

# Linear tensor networks

- The tensor formulation includes
  1. Diagonal networks
  2. Convolution networks
  3. Fully-connected networks.
- Consider linear tensor networks, which means $\phi(t) = t$.

$$f(x;\Theta) = \mathrm{M}(x) \circ (v_1, v_2, \ldots, v_L)$$

- The output of the network can also be written as $f(\boldsymbol{x};\Theta) = \boldsymbol{x}^T \boldsymbol{\beta}(\Theta)$, where $\boldsymbol{\beta}(\Theta) \in \mathbb{R}^d$

# Diagonal networks

- An $L$ -layer diagonal network can be written as

$$f_{\textbf{diag}}\ (x; \Theta_{\textbf{diag}}\ ) = \phi\left(\cdots\phi\left(\phi\left(x \odot w_1\right) \odot w_2\right)\cdots \odot w_{L-1}\right)^T w_L$$

  where $w_l \in \mathbb{R}^d$ for $l \in [L]$.
- $\mathrm{M}_{\textbf{diag}}\ (x) \in \mathbb{R}^{d\times\cdots\times d}$ and $[\mathrm{M}_{\textbf{diag}}\ (x)]_{j,j,\ldots,j} = [x]_j$, while other components are 0.
- $v_l = w_l$ for all $l$

# Convolutional networks

- The convolutional networks can be written as

$$f_{\text{conv}}(x;\Theta_{\text{conv}}) = \phi(\cdots\phi(\phi(x\star w_1)\star w_2)\cdots\star w_{L-1})^T w_L,$$

  where $w_l \in \mathbb{R}^{k_l}$ with $k_l \le d$ and $k_L = d$, and $\star$ defines the circular convolution.
- $a \star b \in \mathbb{R}^d$ defined as $[a\star b]_i = \sum_{j=1}^{k}[a]_{(i+j-1) \bmod d}[b]_j$, for $i \in [d]$. for any $a \in \mathbb{R}^d$ and $b \in \mathbb{R}^k (k \le d)$
- $\mathrm{M}_{\text{conv}}(x) \in \mathbb{R}^{k_1\times\cdots\times k_L}$ as $[\mathrm{M}_{\text{conv}}(x)]_{j_1,j_2,\ldots,j_L} = [x]_{\left(\sum_{l=1}^{L} j_l - L + 1\right) \bmod d}$ for $j_l \in [k_l]$, $l \in [L]$.
- $v_l = w_l$ and $\mathrm{M} = \mathrm{M}_{\text{conv}}$.

# Fully-connected networks

- An $L$ -layer fully-connected network is defined as

$$f_{\mathrm{fc}}\left(x;\Theta_{\mathrm{fc}}\right)=\phi\left(\cdots\phi\left(\phi\left(x^{T}W_{1}\right)W_{2}\right)\cdots W_{L-1}\right)w_{L}$$

  where $\boldsymbol{W}_{l}\in\mathbb{R}^{d_{l}\times d_{l+1}}$ for $l\in[L-1]$ (we use $d_{1}=d$ ) and $w_{L}\in\mathbb{R}^{d_{L}}$.
- One can represent $f_{\mathrm{fc}}$ as the tensor form by
  - Defining parameters $v_{l}=\operatorname{vec}\left(\boldsymbol{W}_{l}\right)$ for $l\in[L-1]$ and $v_{L}=w_{L}$
  - Constructing the tensor $\mathrm{M}_{\mathrm{fc}}(x)$ by a recursive "block diagonal" manner.

## Singular value decomposition of tensor

- Given an order- $L$ tensor $\mathrm{A} \in \mathbb{R}^{k_1 \times \cdots \times k_L}$, we define the singular vectors $u_1, u_2, \ldots, u_L$ and singular value $s$ to be the solution of the following system of equations:

$$su_l = \mathrm{A} \circ (u_1, \ldots, u_{l-1}, I_{k_l}, u_{l+1}, \ldots, u_L), \text{ for } l \in [L]$$

- We can characterize the limit direction of parameters after reaching 100% training accuracy.

# Theorem 1

## Theorem 1

*Assume that the gradient flow satisfies* $\mathcal{L}(\Theta(t_0)) < 1$ *for some* $t_0 \geq 0$ *and* $X^T r(t)$ *converges in direction, say* $u^\infty := \lim_{t\to\infty} \frac{X^T r(t)}{\left\|X^T r(t)\right\|_2}$*. Then,* $v_1, \ldots, v_L$ *converge to the singular vectors of* $\mathrm{M}(-\boldsymbol{u}^\infty)$*. where* $\boldsymbol{r}(\boldsymbol{t}) \in \mathbb{R}^n$ *is defined as*

$$[r(t)]_i = \ell'(f(x_i; \Theta(t)), y_i) = \begin{cases} -y_i \exp(-y_i f(x_i; \Theta(t))) & \textit{for classification,} \\ f(x_i; \Theta(t)) - y_i & \textit{for regression.} \end{cases}$$

- $\dot{\boldsymbol{v}}_l = -\nabla_{\boldsymbol{v}_l}\mathcal{L}(\Theta) = \mathrm{M}\left(-\boldsymbol{X}^T\boldsymbol{r}\right) \circ (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{l-1}, \boldsymbol{I}_{k_l}, \boldsymbol{v}_{l+1}, \ldots, \boldsymbol{v}_L), \quad \forall l \in [L]$

# Corollary 1

**Corollary 1. (cf. Ji & Telgarsky, 2020)**

*Consider an $L$ -layer linear fully-connected network. If the training loss satisfies $\mathcal{L}\left(\Theta_{\mathrm{fc}}\left(t_0\right)\right) < 1$ for some $t_0 \geq 0$, then $\beta_{\mathrm{fc}}\left(\Theta_{\mathrm{fc}}(t)\right)$ converges in a direction that aligns with the solution of the following optimization problem*

$$\textit{minimize}_{\boldsymbol{z} \in \mathbb{R}^d} \quad \|\boldsymbol{z}\|_2^2 \quad \textit{subject to} \quad y_i \boldsymbol{x}_i^T z \geq 1, \forall i \in [n]$$

# Theorem 2

- Theorem 1 is not a full characterization of the limit directions, because therare usually multiple solutions that satisfy a condition of singular value and vectors.
- Singular vectors of high order tensors are much less understood than the matrix conuterparts, let alone orthogonal decompositions.
- The following assumptions defines a class of orthogonally decomposable data tensors $\boldsymbol{M}(x)$

## Condition for orthogonally decomposable data tensor.

### Assumption 1

*For the data tensor* $\mathrm{M}(x) \in \mathbb{R}^{k_1 \times \cdots \times k_L}$ *of a linear tensor network (6), there exist a full column rank matrix* $S \in \mathbb{C}^{m \times d}$ $(d \leq m \leq \min_l k_l)$ *and matrices* $U_1 \in \mathbb{C}^{k_1 \times m}, \ldots, U_L \in \mathbb{C}^{k_L \times m}$ *such that* $U_l^H U_l = I_m$ *for all* $l \in [L]$*, and the data tensor* $\mathrm{M}(x)$ *can be written as*

$$\mathrm{M}(x) = \sum_{j=1}^{m} [\boldsymbol{S}\boldsymbol{x}]_j \left( [\boldsymbol{U}_1]_{\cdot,j} \otimes [\boldsymbol{U}_2]_{\cdot,j} \otimes \cdots \otimes [\boldsymbol{U}_L]_{*,j} \right)$$

# Theorem 2

## Theorem 2

*Suppose a linear tensor network satisfies Assumption 1. If there exists $\lambda > 0$ such that the initial directions $\bar{v}_1, \ldots, \bar{v}_L$ of the network parameters satisfy $\left|\left[U_\ell^T \bar{v}_l\right]_j\right|^2 - \left|\left[U_L^T \bar{v}_L\right]_j\right|^2 \geq \lambda$ for all $l \in [L-1]$ and $j \in [m]$, then $\beta(\Theta(t))$ converges in a direction that aligns with $S^T \rho^\infty$, where $\rho^\infty \in \mathbb{C}^m$ denotes a stationary point of the following optimization problem*

$$\mathit{minimize}_{\rho \in \mathbb{C}^m} \quad \|\rho\|_{2/L} \quad \text{subject to} \quad y_i \mathbf{x}_i^T \mathbf{S}^T \rho \geq 1, \quad \forall i \in [n]$$

# Theorem 2

- The gradient flow finds sparse $\rho^\infty$ that minimizes the $\ell_{2/L}$ norm in the "singular value space," where the data points $x_i$ are transformed into vectors $Sx_i$ consisting of singular values of $\mathrm{M}(x_i)$.
- Also, the proof of Theorem 2 reveals that in case of $L = 2$, the parameters $v_l(t)$ in fact converge to the top singular vectors of the data tensor $\mathrm{M}\left(-X^T r\right)$;
- Compared to Theorem 1. we have a more complete characterization of "which" singular vectors to converge to.

# Corollary 2

## Corollary 2

*Consider an $L$ -layer linear diagonal network. If there exists $\lambda > 0$ such that the initial directions $\bar{w}_1, \ldots, \bar{w}_L$ of the network parameters satisfy $[\bar{w}_l]_j^2 - [\bar{w}_L]_j^2 \geq \lambda$ for all $l \in [L-1]$ and $j \in [d]$, then $\beta_{diag}\left(\Theta_{diag}(t)\right)$ converges in a direction that aligns with a stationary point $z^\infty$ of*

$$\text{minimize}_{\boldsymbol{z} \in \mathbb{R}^d} \quad \|\boldsymbol{z}\|_{2/L} \quad \text{subject to} \quad y_i \boldsymbol{x}_i^T \boldsymbol{z} \geq 1, \forall i \in [n]$$

## Corollary 3 (cf. Gunasekar et al., 2018b)

### Corollary 3

*Consider an L-layer linear full-length convolutional network. If there exists* $\lambda > 0$ *such that the initial directions* $\bar{w}_1, \ldots, \bar{w}_L$ *of the network parameters satisfy* $\left|[\boldsymbol{F}\bar{\boldsymbol{w}}_\ell]_j\right|^2 - \left|[\boldsymbol{F}\bar{\boldsymbol{w}}_L]_j\right|^2 \geq \lambda$ *for all* $\ell \in [L-1]$ *and* $j \in [d]$*, then* $\boldsymbol{\beta}_{conv}\left(\Theta_{conv}(t)\right)$ *converges in a direction that aligns with a stationary point* $z^\infty$ *of*

$$\text{minimize}_{\boldsymbol{z}\in\mathbb{R}^d} \quad \|\boldsymbol{F}\boldsymbol{z}\|_{2/L} \quad \text{subject to} \quad y_i \boldsymbol{x}_i^T z \geq 1, \forall i \in [n].$$

*where* $\boldsymbol{F} \in \mathbb{C}^{d\times d}$ *to be the matrix of discrete Fourier transform basis* $[F]_{j,k} = \frac{1}{\sqrt{d}} \exp\left(-\frac{\sqrt{-1}\cdot 2\pi(j-1)(k-1)}{d}\right)$.

# Corollary 3

- For full-length convolution networks($k_1 = \cdots = k_L = d$) satisfy Assumption 1.
- $S = d^{\frac{L-1}{2}}$ and $U_1 = \cdots = U_L = F^*$

# Theorem 3

## Theorem 3

*Suppose we have a 2-layer linear tensor network and a single data point* $(x, y)$*. Consider the compact* SVD $\mathrm{M}(x) = U_1 \operatorname{diag}(s) U_2^T$*, where* $\boldsymbol{U}_1 \in \mathbb{R}^{k_1 \times m}, \boldsymbol{U}_2 \in \mathbb{R}^{k_2 \times m}$*, and* $s \in \mathbb{R}^m$ *for* $m \leq \min\{k_1, k_2\}$*. Let* $\rho^\infty \in \mathbb{R}^m$ *be a solution of the following optimization problem*

$$\operatorname{minimize}_{\rho \in \mathbb{R}^m} \quad \|\rho\|_1 \quad \textit{subject to} \quad y s^T \rho \geq 1$$

*Assume that there exists* $\lambda > 0$ *such that the initial directions* $\bar{v}_1, \bar{v}_2$ *of the network parameters satisfy* $\left[\boldsymbol{U}_1^T \overline{\boldsymbol{v}}_1\right]_j^2 - \left[\boldsymbol{U}_2^T \overline{\boldsymbol{v}}_2\right]_j^2 \geq \lambda$ *for all* $j \in [\mathrm{m}]$*. Then,* $v_1$ *and* $v_2$ *converge in direction to* $\boldsymbol{U}_1 \boldsymbol{\eta}_1^\infty$ *and* $U_2 \eta_2^\infty$*, where* $|\eta_1^\infty| = |\eta_2^\infty| = |\rho^\infty|^{\odot 1/2}$*, and* $\operatorname{sign}(\eta_1^\infty) = \operatorname{sign}(y) \odot \operatorname{sign}(\eta_2^\infty)$.

# Underdetermined regression

- Due to the fact that the parameters diverge to infinity in separable classification problems, so that the initialization becomes unimportant in the limit.
- This is not the case in regression setting.
- $w_\ell(0) = \alpha \bar{w}_\ell$ for $\ell \in [L-1]$ and $w_L(0) = 0$.

# Lemma 4

- To define norm like function, We use the following lemma on a relevant system of ODEs:

## Lemma 4

*Consider the system of ODEs, where* $p, q : \mathbb{R} \to \mathbb{R}$ *:*

$$\dot{p} = p^{L-2} q, \quad \dot{q} = p^{L-1}, \quad p(0) = 1, \quad q(0) = 0.$$

*Then, the solutions* $p_L(t)$ *and* $q_L(t)$ *are continuous on their maximal interval of existence of the form* $(-c, c) \subset \mathbb{R}$ *for some* $c \in (0, \infty]$*.*
*Define* $h_L(t) = p_L(t)^{L-1} q_L(t)$*; then,* $h_L(t)$ *is odd and strictly increasing, satisfying* $\lim_{t \uparrow c} h_L(t) = \infty$ *and* $\lim_{t \downarrow -c} h_L(t) = -\infty$*.*

# Theorem 5

## Theorem 5

*Suppose a linear tensor network satisfies Assumption 1. Assume further that the matrices $U_1, \ldots, U_L$ and $S$ from Assumption 1 are all real matrices. For some $\lambda > 0$, choose any vector $\bar{\eta} \in \mathbb{R}^m$ satisfying $[\bar{\eta}]_j^2 \geq \lambda$ for all $j \in [m]$, and choose initial directions $\bar{v}_\ell = U_\ell \bar{\eta}$ for $\ell \in [L-1]$ and $\bar{v}_L = 0$. Then, the linear coefficients $\beta(\Theta(t))$ converge to $S^T \rho^\infty$, where $\rho^\infty$ is the solution of*

$$\text{minimize}_{\rho \in \mathbb{R}^m} \quad Q_{L,\alpha,\bar{\eta}}(\rho) := \alpha^2 \sum_{j=1}^{m} [\bar{\eta}]_j^2 H_L\left(\frac{[\boldsymbol{\rho}]_j}{\alpha L \left|\bar{\eta}_j\right|^L}\right) \quad \text{subject to} \quad X\boldsymbol{S}^T\boldsymbol{\rho} = \boldsymbol{y}$$

*where $Q_{L,\alpha,\bar{\eta}} : \mathbb{R}^m \to \mathbb{R}$ is a norm-like function defined using $H_L(t) := \int_0^t h_L^{-1}(\tau) d\tau$.*

# Theorem 5

- $Q_{L,\alpha,\bar{\eta}}(\rho) := \alpha^2 \sum_{j=1}^{m} [\bar{\eta}]_j^2 H_L \left( \frac{[\rho]_j}{\alpha L \left| \bar{\eta}_j \right|^L} \right)$
- $Q_{L,\alpha,\bar{\eta}}(\rho)$ interpolates between the weighted $\ell_1$ and weighted $\ell_2$ norm of $\boldsymbol{\rho}$
- $H_L(t)$:
  - grows like the absolute value function if $t$ is large.
  - grows like a quadratic function if $t$ is close to zero.

## Corollary 5 (cf. Woodworth et al., 2018b)

### Corollary 5

*Consider an L -layer linear diagonal network. For some* $\lambda > 0$, *choose any vector* $\bar{w} \in \mathbb{R}^d$ *satisfying* $[\bar{w}]_j^2 \geq \lambda$ *for all* $j \in [d]$, *and choose initial directions* $\bar{w}_l = \bar{w}$ *for* $l \in [L-1]$ *and* $\bar{w}_L = 0$. *Then, the linear coefficients* $\beta_{diag}(\Theta_{diag}(t))$ *converge to the solution* $z^\infty$ *of*

$$\text{minimize}_{z \in \mathbb{R}^d} \quad Q_{L,\alpha,\overline{\boldsymbol{w}}}(\boldsymbol{z}) := \alpha^2 \sum_{j=1}^{d} [\overline{\boldsymbol{w}}]_j^2 H_L \left( \frac{[\boldsymbol{z}]_j}{\alpha^L \left|[\boldsymbol{w}]_j\right|^L} \right) \quad \textit{subject to} \quad \boldsymbol{X} z = y$$

# Corollary 6

## Corollary 6

*Consider an L-layer linear full-length convolutional network. Assume that the data points* $\{x_i\}_{i=1}^n$ *are all even. For some* $\lambda > 0$*, choose any even vector* $\bar{w}$ *satisfying* $[F\bar{w}]_j^2 \geq \lambda$ *for all* $j \in [d]$*, and choose initial directions* $\bar{w}_l = \bar{w}$ *for* $l \in [L-1]$ *and* $\bar{w}_L = 0$*. Then, the linear coefficients* $\boldsymbol{\beta}_{conv}\left(\Theta_{conv}(t)\right)$ *converge to the solution* $z^\infty$ *of*

$$\underset{z \in \mathbb{R}^d,\ even}{\text{minimize}}\ Q_{L,\alpha,\boldsymbol{F}\overline{\boldsymbol{w}}}(\boldsymbol{F}\boldsymbol{z}) := \alpha^2 \sum_{j=1}^{d} [\boldsymbol{F}\overline{\boldsymbol{w}}]_j^2 H_L\left(\frac{[\boldsymbol{F}\boldsymbol{z}]_j}{\alpha^L \left|[\boldsymbol{F}\overline{\boldsymbol{w}}]_j\right|^L}\right) \quad \text{subject to} \quad \boldsymbol{X}z = \boldsymbol{y}$$

# Theorem 6

## Theorem 6

*Suppose we have a 2-layer linear tensor network and a single data point* $(x, y)$*. Consider the compact* SVD $\mathrm{M}(x) = U_1 \operatorname{diag}(s) U_2^T$*, where* $\boldsymbol{U}_1 \in \mathbb{R}^{k_1 \times m}$, $\boldsymbol{U}_2 \in \mathbb{R}^{k_2 \times m}$*, and* $s \in \mathbb{R}^m$ *for* $m \leq \min\{k_1, k_2\}$*. Assume that there exists* $\lambda > 0$ *such that the initial directions* $\bar{v}_1, \bar{v}_2$ *of the network parameters satisfy* $\left[\boldsymbol{U}_1^T \overline{\boldsymbol{v}}_1\right]_j^2 - \left[\boldsymbol{U}_2^T \overline{\boldsymbol{v}}_2\right]_j^2 \geq \lambda$ *for all* $j \in [\mathrm{m}]$*. Then, gradient flow converges to a global minimizer of the loss* $\mathcal{L}$*, and* $v_1(t)$ *and* $v_2(t)$ *converge to the limit points:*

$$v_1^\infty = \alpha \boldsymbol{U}_1 \left( \boldsymbol{U}_1^T \overline{\boldsymbol{v}}_1 \odot \cosh\left( g^{-1}\left(\frac{y}{\alpha^2}\right) s \right) + \boldsymbol{U}_2^T \overline{\boldsymbol{v}}_2 \odot \sinh\left( g^{-1}\left(\frac{y}{\alpha^2}\right) s \right) \right) + \alpha \left( \boldsymbol{I}_{k_1} - \boldsymbol{U}_1 \boldsymbol{U}_1^T \right) \overline{\boldsymbol{v}}_1$$

$$v_1^\infty = \alpha \boldsymbol{U}_2 \left( \boldsymbol{U}_1^T \overline{\boldsymbol{v}}_1 \odot \cosh\left( g^{-1}\left(\frac{y}{\alpha^2}\right) s \right) + \boldsymbol{U}_2^T \overline{\boldsymbol{v}}_2 \odot \sinh\left( g^{-1}\left(\frac{y}{\alpha^2}\right) s \right) \right) + \alpha \left( \boldsymbol{I}_{k_2} - \boldsymbol{U}_2 \boldsymbol{U}_2^T \right) \overline{\boldsymbol{v}}_2$$

*where* $g^{-1}$ *is the inverse of the following strictly increasing function*

$$g(\nu) = \sum_{j=1}^m [s]_j \left( \frac{\left[U_1^T \bar{v}_1\right]_j^2 + \left[U_2^T \bar{v}_2\right]_j^2}{2} \sinh\left(2[s]_j \nu\right) + \left[U_1^T \bar{v}_1\right]_j \left[U_2^T \bar{v}_2\right]_j \cosh\left(2[s]_j \nu\right) \right)$$

# The End