Implementation notes for G-PCCA

Fabian Paul

16-JAN–2019

Contents

1 Handling absorbing states in G-PCCA

The G-PCCA algorithm consists of two stages:

1. computing the generalized Schur vectors
2. transforming the generalized Schur vectors (simplex algorithms)

For step (2), degeneracy of the eigenvalue $\lambda_0 = 1$ can be problematic. The standard algorithm assumes that the eigenspace of the eigenvalues is spanned by the constant vector $\mathbf{1} = (1, \ldots, 1)^\top$ and possibly other vectors, depending on the multiplicity of the eigenvalue 1. In the degenerate case which occurs for transition matrices with absorbing states, there is not guarantee that eigensolvers or related methods will return $\mathbf{1} = (1, \ldots, 1)^\top$ as an eigenvector. Other bases for the eigenspace are possible and will be selected by the numerical routines. This is at odds with the requirement of the second stage of the (G-)PCCA algorithm. Here I propose to reformulate the first stage of the (G-)PCCA algorithm such that the eigenvector $\mathbf{1}$ is treated explicitly and appears as a basis vector for the eigenspace.

1.1 Removing the mean in the computation of the spectral components

Let $\mathbf{C}_\tau$ be the count matrix and let $\mathbf{C}_0 = \mathrm{diag}(\mathbf{C}_\tau \mathbf{1})$ be the matrix with the number of state exits on its diagonal. Formally $\mathbf{C}_\tau$ and $\mathbf{C}_0$ are covariance matrices, that can be computed form the sequence of microstates as explained in the following. Let $\mathbf{D} \in \mathbb{R}^{(N+\ell)\times n}$ be the one-hot encoded microstate trajectory, where $N + \ell$ is the total number of time steps, ℓ is the lag-time expressed in units of time steps, and n is the number of different microstates. Let $\mathbf{X} \in \mathbb{R}^{N\times n}$ be the N first rows (time steps) of $\mathbf{D}$ and let $\mathbf{Y} \in \mathbb{R}^{N\times n}$ be the N last rows of $\mathbf{D}$. Using these definitions, the count matrices can be computed as follows:

$$\mathbf{C}_0 = \mathbf{X}^\top \mathbf{X} \tag{1}$$

$$\mathbf{C}_\tau = \mathbf{X}^\top \mathbf{Y} \tag{2}$$

We will now perform an affine coordinate transform, where the empirical mean $\mathbf{x} = \frac{1}{N}\mathbf{C}_\tau \mathbf{1}$ will be removed from $\mathbf{X}$ and the empirical mean $\mathbf{y} = \frac{1}{N}\mathbf{C}_\tau^\top \mathbf{1}$ removed from $\mathbf{Y}$. The transformed trajectories are

$$\bar{\mathbf{X}} = \mathbf{X} - \mathbf{1}\mathbf{x}^\top \tag{3}$$

$$\bar{\mathbf{Y}} = \mathbf{Y} - \mathbf{1}\mathbf{y}^\top \tag{4}$$

The covariance matrices computed form the mean-free data are

$$\bar{\mathbf{C}}_0 = (\mathbf{X} - \mathbf{1}\mathbf{x}^\top)^\top (\mathbf{X} - \mathbf{1}\mathbf{x}^\top) = \mathbf{C}_0 - \mathbf{X}^\top \mathbf{1}\mathbf{x}^\top - \mathbf{x}\mathbf{1}^\top \mathbf{X} + \mathbf{x}\mathbf{1}^\top \mathbf{1}\mathbf{x}^\top \tag{5}$$

$$= \mathbf{C}_0 - N\mathbf{x}\mathbf{x}^\top - N\mathbf{x}\mathbf{x}^\top + N\mathbf{x}\mathbf{x}^\top \tag{6}$$

$$= \mathbf{C}_0 - N\mathbf{x}\mathbf{x}^\top = \mathrm{diag}(\mathbf{x}) - N\mathbf{x}\mathbf{x}^\top \tag{7}$$

and

$$\bar{\mathbf{C}}_\tau = (\mathbf{X} - \mathbf{1}\mathbf{x}^\top)^\top (\mathbf{Y} - \mathbf{1}\mathbf{y}^\top) = \mathbf{C}_\tau - \mathbf{X}^\top \mathbf{1}\mathbf{y}^\top - \mathbf{x}\mathbf{1}^\top \mathbf{Y} + \mathbf{x}\mathbf{1}^\top \mathbf{1}\mathbf{y}^\top \tag{8}$$

$$= \mathbf{C}_\tau - N\mathbf{x}\mathbf{y}^\top - N\mathbf{x}\mathbf{y}^\top + N\mathbf{x}\mathbf{y}^\top \tag{9}$$

$$= \mathbf{C}_\tau - N\mathbf{x}\mathbf{y}^\top \tag{10}$$

One sees that the generalized eigenproblem,

$$\mathbf{C}_0 \mathbf{v} = \lambda \mathbf{C}_\tau \mathbf{v} \tag{11}$$

is unchanged by this transformation except for the eigenpair $\lambda_0 = 0$, $\mathbf{v}^{(0)} = \mathbf{1}$. To see this, compute

$$\bar{\mathbf{C}}_0 \mathbf{1} = \mathrm{diag}(N\mathbf{x})\mathbf{1} - N\mathbf{x}\mathbf{x}^\top \mathbf{1} = N\mathbf{x} - N\mathbf{x} = 0 \tag{12}$$

$$\bar{\mathbf{C}}_\tau \mathbf{1} = \mathbf{C}_\tau \mathbf{1} - N\mathbf{x}\mathbf{y}^\top \mathbf{1} = N\mathbf{x} - N\mathbf{x} = 0 \tag{13}$$

from which follows

$$\bar{\mathbf{C}}_0 \mathbf{1} = 0\bar{\mathbf{C}}_\tau \mathbf{1} \tag{14}$$

that is 0 is the new eigenvalue to the eigenvector $\mathbf{1}$. By truncating the spectrum, i.e. by removing the spectral components with small magnitude of the eigenvalue, the eigenvector $\mathbf{1}$ will be removed too. Since presence of $\mathbf{1}$ is essential for the PCCA algorithm, it will be to be added again explicitly.

Note 1: The generalized eigenvalue problem remains unchanged if $\mathbf{C}_0$ and $\mathbf{C}_\tau$ are scaled by the same factor. Therefore one can effectively always set $N = 1$.

Note 2:

$$\sum_i x_i = \sum_{i,j} C_{i,j} = N$$

Note 3: Our notation can be mapped to the one of Weber /Reuter via

$$\begin{aligned}
\mathbf{C}_\tau &= N\mathbf{D}_\eta \mathbf{P} \\
\boldsymbol{\eta} &= \frac{1}{N}\mathbf{C}_\tau \mathbf{1} = \mathbf{x} \\
\mathbf{y} &= \frac{1}{N}\mathbf{C}_\tau^\top \mathbf{1} = (\boldsymbol{\eta}^\top \mathbf{P})^\top
\end{aligned}$$

1.2 Solving the generalized Schur decomposition (with mean removed)

Let the transition matrix in mean-free and whitened basis be

$$\tilde{\mathbf{P}} = \bar{\mathbf{C}}_0^{-\frac{1}{2}} \bar{\mathbf{C}}_\tau \bar{\mathbf{C}}_0^{-\frac{1}{2}} \tag{15}$$

Since $\bar{\mathbf{C}}_0$ is no longer diagonal here, we will use principal component analysis to compute the square root of the pseudo-inverse $\bar{\mathbf{C}}_0^{-\frac{1}{2}}$.

$$\bar{\mathbf{C}}_0^{-\frac{1}{2}} = \mathbf{Q}\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{Q}^\top \tag{16}$$

where $\mathbf{Q}$ and $\boldsymbol{\Lambda}$ are the eigenvectors and eigenvalues of $\bar{\mathbf{C}}_0$ i.e. $\bar{\mathbf{C}}_0 = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^\top$. [Alternative would be Cholesky decomposition?] Spectral components with very small eigenvalues Λ_{ii} are discarded (like when computing the pseudo-inverse). The shifted singular vector $\mathbf{1}$ (with eigenvalue 0) is removed too.

Inserting yields

$$\tilde{\mathbf{P}} = \mathbf{Q}\underbrace{\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{Q}^\top \bar{\mathbf{C}}_\tau \mathbf{Q}\boldsymbol{\Lambda}^{-\frac{1}{2}}}_{=:\mathbf{W}}\mathbf{Q}^\top \tag{17}$$

Now, perform the Schur decomposition of the inner part $\mathbf{W} = \mathbf{U}\mathbf{T}\mathbf{U}^\top$. We find for $\tilde{\mathbf{P}}$

$$\tilde{\mathbf{P}} = \underbrace{\mathbf{Q}\mathbf{U}}_{\tilde{\mathbf{V}}}\bar{\mathbf{T}}\underbrace{\mathbf{U}^\top\mathbf{Q}^\top}_{\tilde{\mathbf{V}}^\top} \tag{18}$$

which is still a Schur decomposition of $\tilde{\mathbf{P}}$ since $\mathbf{QU}$ is orthonormal (by the group property of orthonormal matrices).

In the last step, we will go back to the untilded basis. Call $\tilde{\mathbf{V}} = \mathbf{QU}$. The matrix of generalized Schur vectors $\bar{\mathbf{V}}$ (coefficients) in the untilded (but mean-free) basis can then be computed as (see equation XX in [Weber])

$$\bar{\mathbf{V}} = \bar{\mathbf{C}}_0^{-\frac{1}{2}}\tilde{\mathbf{V}} = \mathbf{Q}\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{Q}^\top\mathbf{Q}\mathbf{U} = \mathbf{Q}\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U} \tag{19}$$

For the later PCCA stages, we need one Representative per Markov state in the original basis. The latter PCCA stage does not require any time ordering of the states. Algorithmically, we can therefore prepare an artificial trajectory $X_{ij}^{\text{sweep}} = \delta_{ij}$ that visits all the microstates and map this trajectory to the space of generalized Schur vectors

$$\mathbf{V}_{:,2:n} = \bar{\mathbf{X}}^{\text{sweep}}\bar{\mathbf{V}} = (\mathbf{X}^{\text{sweep}} - \mathbf{1}\mathbf{x}^\top)\bar{\mathbf{V}} = \bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top\bar{\mathbf{V}} \tag{20}$$

where $\mathbf{x}$ is the mean of the original data that was used to estimate $\bar{\mathbf{V}}$ (and not the mean of $\mathbf{X}^{\text{sweep}}$). Since the constant Schur vector was removed together with the mean, we need to add it back explicitly. We set $\mathbf{V}_{:,1} = (1,\ldots,1)^\top/\sqrt{N}$. Together we have

$$\boxed{\mathbf{V} = (\mathbf{1}/\sqrt{N} \mid \bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top\bar{\mathbf{V}})} \tag{21}$$

It can be shown that the columns of $\mathbf{V}$ fulfill the definition of generalized Schur vectors for the original count matrices that include the contributions from the mean, namely

$$\mathbf{V}^\top\mathbf{C}_0\mathbf{V} = \mathrm{Id}_{n\times n} \tag{22}$$

and

$$\mathbf{V}^\top\mathbf{C}_\tau\mathbf{V} = \mathbf{T} \tag{23}$$

where the matrix $\bar{\mathbf{T}}$ must be extended as follows (see below for proof) and which is still triangular.

$$\boxed{\mathbf{T} = \begin{pmatrix} 1 & \sqrt{N}(\mathbf{y}^\top - \mathbf{x}^\top)\bar{\mathbf{V}} \\ \mathbf{0} & \bar{\mathbf{T}} \end{pmatrix}} \tag{24}$$

Note 4: Since $\bar{\mathbf{V}} \propto \frac{1}{\sqrt{N}}$, all scalings with $\sqrt{N}$ are consistent.

2 Proofs

2.1 Proof of $\mathbf{V}^\top \mathbf{C}_0 \mathbf{V} = \mathrm{Id}$:

- Non-constant Schur vectors

$$
\begin{aligned}
&(\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}})^\top \mathbf{C}_0 (\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}}) = \\
&\quad \bar{\mathbf{V}}^\top \mathbf{C}_0 \bar{\mathbf{V}} - \bar{\mathbf{V}}^\top \mathbf{C}_0 \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}} - \bar{\mathbf{V}}^\top \mathbf{x}\mathbf{1}^\top \mathbf{C}_0 \bar{\mathbf{V}} + \bar{\mathbf{V}}^\top \mathbf{x}\mathbf{1}^\top \mathbf{C}_0 \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}} = \\
&\quad \bar{\mathbf{V}}^\top \mathbf{C}_0 \bar{\mathbf{V}} - N\bar{\mathbf{V}}^\top \mathbf{x}\mathbf{x}^\top \bar{\mathbf{V}} - N\bar{\mathbf{V}}^\top \mathbf{x}\mathbf{x}^\top \bar{\mathbf{V}} + N\bar{\mathbf{V}}^\top \mathbf{x}\mathbf{x}^\top \bar{\mathbf{V}} = \\
&\quad \bar{\mathbf{V}}^\top \mathbf{C}_0 \bar{\mathbf{V}} - N\bar{\mathbf{V}}^\top \mathbf{x}\mathbf{x}^\top \bar{\mathbf{V}} = \\
&\quad\quad \bar{\mathbf{V}}^\top (\mathbf{C}_0 - N\mathbf{x}\mathbf{x}^\top) \bar{\mathbf{V}} = \\
&\quad\quad\quad \bar{\mathbf{V}}^\top \bar{\mathbf{C}}_0 \bar{\mathbf{V}} \overset{2.3}{=} \mathrm{Id}
\end{aligned}
$$

- Constant Schur vector

$$(\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}})^\top \mathbf{C}_0 \mathbf{1} = N(\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}})^\top \mathrm{x} = N\bar{\mathbf{V}}^\top \mathrm{x} - N\bar{\mathbf{V}}^\top \mathrm{x}\mathbf{1}^\top \mathrm{x} = 0$$

where I have used $\sum_i x_i = 1$.

- Finally

$$\frac{1}{\sqrt{N}}\mathbf{1}^\top \mathbf{C}_0 \frac{1}{\sqrt{N}}\mathbf{1} = \frac{N}{N} = 1$$

2.2 Proof of $\mathbf{V}^\top \mathbf{C}_\tau \mathbf{V} = \mathbf{T}$:

- Non-constant Schur vectors

$$
\begin{aligned}
&(\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}})^\top \mathbf{C}_\tau (\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}}) = \\
&\quad \bar{\mathbf{V}}^\top \mathbf{C}_\tau \bar{\mathbf{V}} - \bar{\mathbf{V}}^\top \mathbf{C}_\tau \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}} - \bar{\mathbf{V}}^\top \mathbf{x}\mathbf{1}^\top \mathbf{C}_\tau \bar{\mathbf{V}} + \bar{\mathbf{V}}^\top \mathbf{x}\mathbf{1}^\top \mathbf{C}_\tau \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}} = \\
&\quad \bar{\mathbf{V}}^\top \mathbf{C}_\tau \bar{\mathbf{V}} - N\bar{\mathbf{V}}^\top \mathbf{x}\mathbf{x}^\top \bar{\mathbf{V}} - N\bar{\mathbf{V}}^\top \mathbf{x}\mathbf{y}^\top \bar{\mathbf{V}} + N\bar{\mathbf{V}}^\top \mathbf{x}\mathbf{x}^\top \bar{\mathbf{V}} = \\
&\quad \bar{\mathbf{V}}^\top \mathbf{C}_\tau \bar{\mathbf{V}} - N\bar{\mathbf{V}}^\top \mathbf{x}\mathbf{y}^\top \bar{\mathbf{V}} = \\
&\quad\quad \bar{\mathbf{V}}^\top (\mathbf{C}_\tau - N\mathbf{x}\mathbf{y}^\top) \bar{\mathbf{V}} = \\
&\quad\quad\quad \bar{\mathbf{V}}^\top \bar{\mathbf{C}}_\tau \bar{\mathbf{V}} \overset{2.4}{=} \mathbf{T}
\end{aligned}
$$

- Constant Schur vectors (compute the missing entries of $\mathbf{T}$):

$$\mathbf{T}_{2:n,1} = (\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}})^\top \mathbf{C}_\tau \mathbf{1} = N(\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}})^\top \mathbf{x} = 0$$

(see above)

$$
\begin{aligned}
\mathbf{T}_{1,2:n} &= \frac{1}{\sqrt{N}}\mathbf{1}^\top \mathbf{C}_\tau (\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}}) \\
&= \sqrt{N}\mathbf{y}(\bar{\mathbf{V}} - \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}}) \\
&= \sqrt{N}\mathbf{y}^\top \bar{\mathbf{V}} - N\mathbf{y}^\top \mathbf{1}\mathbf{x}^\top \bar{\mathbf{V}} \\
&= \sqrt{N}\mathbf{y}^\top \bar{\mathbf{V}} - N\mathbf{x}^\top \bar{\mathbf{V}} \\
&= \sqrt{N}(\mathbf{y}^\top - \mathbf{x}^\top)\bar{\mathbf{V}}
\end{aligned}
$$

$$\mathbf{T}_{1,1} = \frac{1}{\sqrt{N}}\mathbf{1}^\top \mathbf{C}_\tau \frac{1}{\sqrt{N}}\mathbf{1} = \frac{N}{N} = 1$$

2.3 Verification of $\bar{\mathbf{V}}^\top \bar{\mathbf{C}}_0 \bar{\mathbf{V}} = \mathrm{Id}$:

$$\begin{aligned}\bar{\mathbf{V}}^\top \bar{\mathbf{C}}_0 \bar{\mathbf{V}} &= \mathbf{U}^\top \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{Q}^\top \bar{\mathbf{C}}_0 \mathbf{Q} \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{U} \\ &= \mathbf{U}^\top \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{Q}^\top \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^\top \mathbf{Q} \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{U} = \mathrm{Id}\end{aligned}$$

2.4 Verification of $\bar{\mathbf{V}}^\top \bar{\mathbf{C}}_\tau \bar{\mathbf{V}} = \bar{\mathbf{T}}$:

First step:

$$\begin{aligned}\bar{\mathbf{C}}_\tau \bar{\mathbf{V}} &= \bar{\mathbf{C}}_0 \bar{\mathbf{V}} \bar{\mathbf{T}} \Leftrightarrow \\ \bar{\mathbf{C}}_\tau \mathbf{Q} \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{U} &= \bar{\mathbf{C}}_0 \mathbf{Q} \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{U} \bar{\mathbf{T}} \Leftrightarrow \\ \bar{\mathbf{C}}_\tau &= \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^\top \mathbf{Q} \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{U} \bar{\mathbf{T}} \mathbf{U}^\top \mathbf{\Lambda}^{\frac{1}{2}} \mathbf{Q}^\top \Leftrightarrow \\ \bar{\mathbf{C}}_\tau &= \mathbf{Q} \mathbf{\Lambda} \mathbf{Q}^\top \mathbf{Q} \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{Q}^\top \bar{\mathbf{C}}_\tau \mathbf{Q} \mathbf{\Lambda}^{-\frac{1}{2}} \mathbf{\Lambda}^{\frac{1}{2}} \mathbf{Q}^\top = \bar{\mathbf{C}}_\tau\end{aligned}$$

Second step:

$$\begin{aligned}\bar{\mathbf{C}}_\tau \bar{\mathbf{V}} &= \bar{\mathbf{C}}_0 \bar{\mathbf{V}} \bar{\mathbf{T}} \Leftrightarrow \\ \bar{\mathbf{V}}^\top \bar{\mathbf{C}}_\tau \bar{\mathbf{V}} &= \bar{\mathbf{V}}^\top \bar{\mathbf{C}}_0 \bar{\mathbf{V}} \bar{\mathbf{T}} \overset{2.3}{=} \bar{\mathbf{T}}\end{aligned}$$

3 Kinetic map

The Schur vectors, multiplied by the triangular matrix induce a kinetic map $\boldsymbol{\gamma}(i) := \mathbf{V}_{i,:}\mathbf{T}$. To shows this, we write the transition matrix as

$$\mathbf{P} = \mathbf{V}\mathbf{T}\mathbf{V}^{-1} = \mathbf{V}\mathbf{T}\mathbf{V}^\top \mathbf{C}_0$$

since

$$\mathbf{V}^{-1} = (\mathbf{C}_0^{-\frac{1}{2}} \tilde{\mathbf{V}})^{-1} = \tilde{\mathbf{V}}^\top \mathbf{C}_0^{\frac{1}{2}} = \mathbf{V}^\top \mathbf{C}_0$$

When written out in components, this becomes

$$p_{ij} = \sum_{l,h} v_{il} T_{lh} v_{hj} c_j$$

The squared kinetic distance between Markov states i and j is defined as (see for instance [Noé and Clementi, JCTC,11 5002 (2015)])

$$D^2(i,j) := \sum_k \frac{(p_{ik} - p_{jk})^2}{c_k}$$

Inserting the Schur decomposition yields

$$D^2(i,j) = \sum_k \left(\sum_{l,h} v_{il} T_{lh} v_{hk} c_k - \sum_{l,h} v_{jl} T_{lh} v_{hk} c_k \right)^2 c_k^{-1}$$
$$= \sum_k \left(\sum_{l,h} (v_{il} - v_{jl}) T_{lh} v_{hk} \sqrt{c_k} \right)^2$$

With the definition $\delta_l^{(i,j)} := v_{il} - v_{jl}$, we can write (suppressing the fixed indices i and j)

$$D^2 = \sum_k \left(\sum_{l,h} \delta_l T_{lh} v_{hk} \sqrt{c_k} \right)^2$$

$$= \sum_k \sum_{l,h,l',h'} \delta_l T_{lh} v_{hk} \sqrt{c_k} \delta_{l'} T_{l'h'} v_{h'k} \sqrt{c_k}$$

$$= \sum_{l,h,l',h'} \delta_l T_{lh} \delta_{l'} T_{l'h'} \underbrace{\sum_k v_{h'k} c_k v_{hk}}_{\delta_{h'h} \text{ (Kronecker)}}$$

$$= \sum_h \left(\sum_l \delta_l T_{lh} \cdot \sum_{l'} \delta_{l'} T_{l'h} \right)$$
$$= \boldsymbol{\delta}^\top \mathbf{T} \mathbf{T}^\top \boldsymbol{\delta} = \left\| \mathbf{T}^\top \boldsymbol{\delta} \right\|^2 \geq 0$$

Note that this object is symmetric, since exchanging i, j is equivalent to $\delta \to -\delta$, which leaves $D^2(i,j)$ unchanged. Importantly, this also holds if $\mathbf{T}$ only contains the dominant Schur block.

$$D^2(i,j) = \left\| \boldsymbol{\delta}^\top \mathbf{T} \right\|^2 = \left\| (\mathbf{V}_{i,:} - \mathbf{V}_{j,:}) \mathbf{T} \right\|^2$$

The Schur decomposition therefore induces a kinetic map $\{\boldsymbol{\gamma}(i)\}_i$ with

$$\boldsymbol{\gamma}(i) := \mathbf{V}_{i,:} \mathbf{T}$$

such that

$$D^2(i,j) = \left\| \boldsymbol{\gamma}(i) - \boldsymbol{\gamma}(j) \right\|^2$$

3.1 Ambiguity of the kinetic map

[TODO] The kinetic map can be transformed with an additional orthogonal transform $\mathbf{O}$

$$\boldsymbol{\gamma}'(i) = \mathbf{V}_{i,:}\mathbf{TO} = \mathbf{V}_{i,:}\mathbf{T}'$$

such that D_{ij}^2 is conserved for all i, j. For instance $\mathbf{O}$ could be a permutation matrix that reorders the columns of $\mathbf{T}$ (permutation cannon change T form upper triangular to lower triangular, can a general orthonormal transform do this?). From the QR-decomposition, we know that any matrix $\mathbf{A}$ can be decomposed as the product of a orthogonal matrix and triangular matrix. That is $\mathbf{TO} = \mathbf{T}'$. From this follows that for the definition of the kinetic map, the matrix $\mathbf{T}$ can be a general matrix and does not have to be triangular.

3.2 Choice of upper or lower triangular form

[TODO] $\mathbf{T}$ can be chosen to be upper or lower triangular. The type of the Schur determined the interpretation of the kinetic map. For the upper triangular form: increasingly more components $\mathbf{V}_{:,j}$ are used to build $\boldsymbol{\gamma}_j(i)$ as j is increased. Lower form: increasingly less components $\mathbf{V}_{:,j}$ are used to build $\boldsymbol{\gamma}_j(i)$ as j is increased. The first choice is more interpretable: the first Schur vector is the best 1-D approximation to the distance. A superposition of the first two Schur vectors is the best 2-D approximation etc. [TODO: check upper/lower]