PCCA+ and its application to spatial timeseries clustering

March 4, 2015

Contents

1 Introduction

In this thesis I will try to give an overview over the PCCA+ algorithm, as well as it's application.

I therefore will cover the basic PCCA+ setting, as primarily investigated by Weber in his dissertation[7].

After an introduction to the theoretical setting I will explain the PCCA+ algorithm and discuss some of its developments. Finally I will showcase an application to spatial timeseries human eye tracking data used for object recognition.

The PCCA+ algorithm leads to a fuzzy clustering of the state space of a Markov process, or in other words it reduces the dimension of the state space, such that the resulting process on the reduced state space still represents the main dynamics on arbitrary large time scales exactly.

2 Theoretical background

2.1 Introduction to Markov chains

Let S be any finite set, i.e. $S =: \{1, ..., N\}$. Then a Markov chain on S is a stochastic process, consisting of a sequence of random variables $X_i : \Omega \to S$, $i \in \mathbb{N}$ satisfying the Markov property:

$$P(X_{t+1} = x | X_1 = x_1, X_2 = X_2, ..., X_t = x_t) = P(X_{t+1} = x | X_t = x_t) \, \forall t \in \mathbb{N}.$$

It is common to interpret S as the state space of possible outcomes of measurements at the time t represented by X_t. The Markov property assures, that the transitions to the next timestep $t + 1$ only depend on the current state x_t. This means that the process at time t has no memory of its previous history $(x_1, ..., x_{t-1})$, thus this also sometimes called the memoryless property.

We will furthermore assume that the process is autonomous, i.e. not explicitly depending on the time:

$$P(X_{t+1} = x | X_t = y) = P(X_t = x | X_{t-1} = y) \forall t \in \mathbb{N}$$

This does not realy impose a restriction as any non-autonomous process can be turned into an autonomous one: By adding all possible times to the state space S taking the cartesian product $S' := \mathbb{N} \times S$ the explicit time-dependence of the process on S can be implicitly subsumed by an autonomous process on S'.

As S is finite we can, enumerating all states in S, encode the whole process in the transition matrix

$$P_{ij} := P\left(X_{t+1} = j | X_t = i\right)$$

A *stationary distribution* is a row vector π satisfying

$$\pi P = \pi$$

A markov chain is called *reversible* if there exists a *stationary distribution* π satisfying the detailed balance equation

$$\pi_i P_{ij} = \pi_j P_{ji} \tag{2.1}$$

which assures equal back and forth transitions between any two states i, j, weighted by their stationary distribution π_i, π_j.

Introducing the diagonal matrix

$$D_\pi := \text{diag}\left(\pi\right)$$

this can also be written in matrix notation as

$$D_\pi P = P^T D_\pi.$$

2.2 Discretization of the state space

Although we only consider a discrete state space in this thesis, the results are extensible to continous state spaces as well.

The easiest way is using a set-based discretization, dividing the state space into a finite mesh of subsets.

For high dimensional state spaces, as for example met in molecular dynamics, this approach exhibits the curse of dimensionality, as the size of the mesh grows exponentially with the dimensions. As solution to this problem Weber developed a meshless version of PCCA+ using a global Galerkin discretization[7].

2.3 Clustering of the state space

As the goal of PCCA+ is to reduce the complexity of analysis of the markov chain by a dimension reduction we will now introduce the concept of clustering, that is subsuming different states of the state space to a smaller set of $n \in \mathbb{N}$ clusters $C := \{1, ..., n\}$.

The simplest possibility is assigning each state $k \in S$ to a cluster $i \in C$, which can be encoded by means of the *characteristic vector* $\chi_i \in \{0, 1\}^N$:

$$\chi_{i,k} = \begin{cases} 1, & \text{if state } k \text{ belongs to cluster } i \\ 0, & \text{else} \end{cases}.$$

Due to its discrete nature, this *crisp clustering* approach, used by *Perron Cluster Analysis* (PCCA) of Deuflhard et al. [1], has the disadvantage of not beeing robust against small pertubations, as continious changes in P finally result in discontinous changes in the clustering.

Weber and Galliat, Deuflhard and Weber therefore developed a robust version, *Robust Perron Cluster Analyis* (PCCA+), by making use of a *fuzzy clustering* representing each cluster by an *almost characteristic vector*

$$\chi_i \in [0,1]^n . \tag{2.2}$$

A*lmost characteristic vectors* $\{\chi_i\}_{i=1}^n$ satisfying the partition of unity property

$$\sum_{i=1}^{n} \chi_i = 1 \tag{2.3}$$

are called *membership vectors* as they describe the relative membership of each state to each cluster.

We will refer to the matrix collection $\chi := (\chi_i)_{i=1}^n \in \mathbb{R}^{N\times n}$ of the *membership vectors* as a *clustering*, whereas in the field of computational chemistry it is also referred to as *conformations*.

2.4 Galerkin projection of the transition matrix

2.4.1 The coupling matrix

To represent the dynamics on the reduced/clustered state space in the case of a *crisp clustering* χ, i.e. $\chi_i \in \{0,1\}$, Deuflhard et al. [1] introduced the *coupling matrix*

$$W_{ij} := \frac{\langle \chi_j, P\chi_i \rangle_\pi}{\langle \chi_i, 1 \rangle_\pi} = \frac{\chi_j^T D_\pi P \chi_i}{\pi^T \chi_i},$$

or in matrix notation

$$W := \text{diag}\left(\chi^T \pi\right)^{-1} \chi^T D_\pi P \chi.$$

The entries W_{ij} can thus be interpreted as conditional transition probability from cluster i to cluster j, given the starting distribution π.

In the *fuzzy clustering* setting the problem arises, that it is no more clear which state belongs to which cluster. I therefore propose to interpret the membershipership of state j to cluster χ_i, χ_{ij}, as probability of measuring/counting state j as cluster χ_i.

Then W_{ij} denotes the expectation value for measuring cluster χ_j after propagating the density given by χ_i weighted by π.

Note however that even if no real transitions are actually happening in the state space, we still may count transitions between clusters, as shown in example 3.

2.4.2 Metastability

One of the main motivations for developing PCCA+ was the wish to identify so called *metastable conformations* of molecular systems, e.g. to analyse the effectivity of active pharmaceutical ingredients in Computional Molecular Design.

These *conformations* are *almost invariant aggregates* of states, i.e. *membership vectors* with high self-transition probabilities, guaranteeing that the system resides in these states on longer timescales.

This can be formalized, as proposed by Huisinga [3], by the definition of the *metastability* of *membership vectors* as the trace of the corresponding *coupling matrix*:

$$\operatorname{tr}(W).$$

Note that this does not need to correspond with a high probability of the cluster.

2.4.3 The propagator matrix

Unfortunately the projection via the *coupling matrix* does not commute with time propagation, necessary for time-scale preservation, which is desired for long term analysis of the markov chain (as in bio/med...).

Therefore Kube and Weber [4] proposed the *coarse propagator matrix*

$$P_C := \left(\chi^T D_\pi \chi\right)^{-1} \chi^T D_\pi P \chi, \tag{2.4}$$

which coincides with the *coupling matrix* W in the *crisp clustering* setting.

Assuming that χ is a linear combination of vectors spanning an P-invariant subspace satisfying an orthonormality condition, as will be satisfied in the PCCA+ approach, it has the advantage that discretization via χ and time propagation commute, i.e.

$$P\chi = \chi P_C. \tag{2.5}$$

This property ensures of that the *coupling matrix* represents the right dynamics of the underlying markov chain on S on the reduced clustered state space, even for iterative application, i.e. $P^n\chi = \chi P_C^n$,

Theorem 1. *Let $\chi = XA$, $X \in \mathbb{R}^{N\times n}$, $A \in \mathbb{R}^{n\times n}$ satisfying the subspace condition*

$$PX = X\Lambda \tag{2.6}$$

for some $\Lambda \in \mathbb{R}^{n\times n}$ and the orthonormality condition

$$X^T D_\pi X = I. \tag{2.7}$$

Then the P_C is conjugate to Λ and discretization-propagation commutativity 2.5 holds.

Proof. We calculate

$$\begin{aligned} P_C &= \left(\chi^T D_\pi \chi\right)^{-1} \chi^T D_\pi P \chi \\ &= \left(A^T X^T D_\pi X A\right)^{-1} A^T X^T D_\pi X \Lambda A \\ &= \left(A^T A\right)^{-1} \left(A^T \Lambda A\right) \\ &= A^{-1} \Lambda A, \end{aligned} \tag{2.8}$$

which implies

$$P\chi = PXA = X\Lambda A = XAA^{-1}\Lambda A = \chi P_C.$$

□

2.4.4 Stochastic interpretation

Due to the matrix inversion the *propagator matrix* can have negative entries, as shown in example 3 below, thus prohibiting a natural stochastic interpretation.

But I think the following considerations, based on the work of Kube and Weber [4], might help in understanding. Consider the inverted part of the *propagator matrix*, $\chi^T D_\pi \chi$. It computes the overlap of all clusters with each other, weighted with the distribution π. If we have overlap between different clusters, we also count "transitions" in the denominator $\chi^T D_\pi P \chi$ just due to the effect of the cluster mixing without any underlying dynamics. $\chi^T D_\pi \chi$ computes the amount of transitions which are just due to the overlap (corresponding to $P = \mathrm{Id}$), and therfore applying $\left(\chi^T D_\pi \chi\right)^{-1}$ reverts this amount, removing the cluster induced transitions, leaving just the underlying dynamics.

We can furthermore consider $\chi : [0,1]^n \to A := \chi\,[0,1]^n \subset [0,1]^N$ as a change of basis between the state space and the cluster space. From 2.5 follows $P_C = \chi^{-1} P \chi$, i.e. P_C is just P restricted to the smaller set A, given to the basis of clusters.

To interpret this stochastically we need to interpolate back to the basis of all states, locally approximated by the stationary distribution π, measure the overlap with the other clusters and normalize with the stationary distribution on the clusters to obtain the conditional probability. Denoting this operation as the ***Sikorski-Transformation*** $T = diag(\pi^T \chi)^{-1} \chi^T D_\pi \chi$, we now see that this is exactly the relation between the *propagator-* and the *coupling matrix* $W = TP_C$.

Furthermore the same argument holds for iterates of the *propagator matrix*, P_C^n, thus giving us a way of a stochastic interpreting the evolution on the clusters even for longer times by TP_C^n.

2.5 Example Processes

To demonstrate the connections between the different projections we now will show some example systems.

Example 1 - the decoupled system Consider $P = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$. In this ideal decoupled system we have two invariant subspaces spanned by the so called *Perron eigenvectors* with eigenvalue 1, the vectors $\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$ and $\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$. These can be interpreted as a discrete a *crisp clustering*, and assuming an equidistributed starting distribution we can compute the, in this case coinciding, matrices $W = P_C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

Example 2 - the 3-pot Next we will consider the simpler process given by $P := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$, i.e. the stationary markov chain on three states, but with the *fuzzy clustering* $\chi = \begin{pmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \\ 0 & 1 \end{pmatrix}$. According to our definitions we now compute $P_C := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and $W = \frac{1}{6}\begin{pmatrix} 5 & 1 \\ 1 & 5 \end{pmatrix}$. We thus observe that P_C contains the expected (identity) dynamics on the reduced state space, while W accounts for the possible transitions of observing the second state once in cluster 1 and once in cluster 2, due to the overlap in the *clustering*.

Example 3 - negative entries The following exampe will illustrate the occurence of negative entries in P_C:

$$P = \frac{1}{4}\begin{pmatrix} 1 & 3 \\ 3 & 1 \end{pmatrix}, \chi = \begin{pmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \pi = \frac{1}{2}\begin{pmatrix} 1 & 1 \end{pmatrix}$$

$$P_C = \frac{1}{8}\begin{pmatrix} 5 & 3 \\ 9 & -1 \end{pmatrix}$$

Now $(0,1)\cdot P_C = \frac{1}{8}(9,-1)$ gives us a linear combination which represents the distribution after applying P to the distribution represented by $(0,1)$. Projecting these back to the full state space we

So indeed even negative entries finally represent positive quantities in the measurement, one might think of P_C as acting with the basis of the clusters.

Example 4 - pert. of ex4

$$P = \frac{1}{4}\begin{pmatrix} 1 & 3 & 0 \\ 2 & 1 & 1 \\ 0 & 3 & 1 \end{pmatrix}, \chi = \begin{pmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \\ 0 & 1 \end{pmatrix}, \pi = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$$

3 PCCA+

We now first present the PCCA+ algorithm in the case of reversible processes.

We first will construct the matrix X spanning the required invariant subspace, then examine the possible linear transformations A mapping these to a set of membership vectors and finally propose an optimization problem to specify a "good" solution, representing the goal of metastability in the form of a objective function.

We will first develop the theory for reversible processes and then show how it is extendable to non-reversible processes, as encountered in the application.

3.1 Motivation for the spectral decomposition

PCCA+ will construct the clusters, described by the *membership vectors*, as a linear combination of eigenvectors. This guarantees that χ spans an invariant subspace, whose dynamics is governed by the corresponding eigenvalues, thus leading to preservation of the slow time-scales. By choosing the $n < N$ eigenvectors with the largest eigenvalues one hopes to preserve the principal dynamics of P. The eigenvectors are good data for this goal, as an eigenvector with a high eigenvalue represents a high degree of self-mapping and thus expresses similar behaviour of the corresponding states.

Deuflhard et al. [1] have furthermore shown that the desired metastability is bounded from above by the sum of the chosen eigenvalues, and for ϵ-pertubations of the coupling of uncoupled markov chains also from below by $\sum \lambda_i - O\left(\epsilon^2\right)$, justifying the choice of high eigenvalues.

3.2 Reversible processes

3.2.1 Construction of X and Λ

To satisfy the orthonormality condition 2.7 we define the matrix $\tilde{P} := D_\pi^{\frac{1}{2}} P D_\pi^{-\frac{1}{2}}$, where D_π denotes the diagonal matrix with the stationary distribution π.

As we assume a reversible process the detailed balance condition 2.1 holds, assuring that $\tilde{P}$ is symmetric:

$$\tilde{P}^T = D_\pi^{-\frac{1}{2}} P^T D_\pi^{\frac{1}{2}} = D_\pi^{-\frac{1}{2}} D_\pi P D_\pi^{-1} D_\eta^{\frac{1}{2}} = \tilde{P}$$

We therefore can diagonalize $\tilde{P}$ such that $\tilde{P}\tilde{X}' = \tilde{X}'\Lambda'$ with $\tilde{X}' \in \mathbb{R}^{N\times N}$ beeing orthonormal and $\Lambda' \in \mathbb{R}^{N\times N}$ beeing diagonal.

We then select the n largest eigenvalues $\Lambda \in \mathbb{R}^{n\times n}$ and the corresponding eigenvectors $\tilde{X} \in \mathbb{R}^{N\times n}$, and define $X := D_\pi^{-\frac{1}{2}} \tilde{X}$.

We then check that the conditions for 1 are satisfied:

$$\begin{aligned} \tilde{P}\tilde{X} &= \tilde{X}\Lambda \\ \Leftrightarrow D_\pi^{\frac{1}{2}} P D_\pi^{-\frac{1}{2}} D_\pi^{\frac{1}{2}} X &= D_\pi^{\frac{1}{2}} X \Lambda \\ \Leftrightarrow PX &= X\Lambda \end{aligned}$$

$$\begin{aligned} \tilde{X}^T \tilde{X} &= I \\ \Leftrightarrow X^T D_\pi^{\frac{1}{2}} D_\pi^{\frac{1}{2}} X &= I \\ \Leftrightarrow X^T D_\pi X &= I \end{aligned}$$

3.2.2 Feasible Set

Given a fixed eigenvector matrix X, we will now examine the set of feasible solutions $F_A \subset \mathbb{R}^{n \times n}$ for the transformation matrix A leading to actual *membership vectors* $\chi := XA$.

As P is stochastic the constant one vector is mapped to itself, $P \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ and thus forms an eigenvector to eigenvalue 1, i.e. $X_{i,1} = 1, i = 1, ..., N$. Thus one can reformulate the positivity 2.2 and partition of unity 2.3 conditions in terms of the matrices X and A, leading to the following constraints for A:

$$A_{1,j} \geq -\sum_{k=2}^{n} X_{ik} A_{kj},\ i = 1, ..., N,\ j = 1, ..., n \text{ (positivity)}, \tag{3.1}$$

$$A_{i,1} = \delta_{i,1} - \sum_{j=2}^{n} A_{ij},\ i = 1, ..., n \text{ (partition of unity)} \tag{3.2}$$

Since these constraints are linear in A the set F_A is a convex polytope, and it is not empty as the matrix $A^*_{ij} := \frac{\delta_{i,1}}{n}$ satisfies these conditions.

As Deuflhard and Weber [2] have shown, the set F_A is indeed uncountable. We therefore look for some criterion to choose a specific solution by means of choosing an objective function for an optimization problem. To motivate the specific choices we first try to gain some insight into the geometry of the clustering problem.

3.2.3 Geometric interpretation

If one considers the N rows of the matrix χ as points in the space $\mathbb{R}^n$, the positivity and 2.2 and partition of unity 2.3 conditions force these points to lie on the standard $(n-1)$-simplex Δ. Now, if $\chi = XA$ this means the that the matrix A maps the the N rows of the eigenvector matrix X to that simplex.

As we have seen the first component of each row is 1, thus all rows lie on the hyperplane with first component 1. They furthermore are contained in a bounded region, and thus we can map them linearly onto Δ via A.

Assuming (*maximality assumption)* that the convex hull co (X) of the rows of X already has the form of an $(n-1)$-simplex, we can now choose A uniquely

(up to permutation) to map this exactly onto Δ, which among all the ways of mapping X into Δ gives us the highest distinguishability between the resulting clusters. This assumption is equivalent to the situation that for each corner there exists a row getting mapped into that corner, i.e.

$$\max_{i=1..N} \chi_{ij} = 1,\ j = 1, ..., n,$$

justifying its name.

3.2.4 Maximal scaling condition

As in the general the *maximality assumption* is not met, it seems natural to turn it into an optimization problem. This has been done in [2, 7] by imposing maximization of the *maximal scaling condition*

$$I_1(A) := \sum_{j=1}^{n} \max_{i=1..N} \chi_{ij} \leq n_C.$$

Assuming that the *maximality assumption* is almost met, i.e. $\max_{i=1..N} \chi_{ij} \approx 1,\ j = 1, ..., n$, Weber [7] argues that the maximizing indices can be determined by the *index mapping algortithm*3.2.8, turning this convex optimization problem into a linear one:

$$I_1(A) = \sum_{i,j=1}^{n} X_{ind(X)_j, i} A_{ij}$$

In [7] Weber furthermore shows that $W_{jj} \leq \max_{i=1..N} \chi_{ij}$, which implies that I_1 is an upper bound for the metastability, which thus should be large.

3.2.5 Maximal metastability condition

Another choice might be optimizing towards a maximal metastability, as done by Deuflhard and Weber [2, 7]

$$I_2(A) := \text{trace}(W) = \sum_{i=1}^{n} \lambda_i \sum_{j=1}^{n} \frac{A(i,j)^2}{A(1,j)},$$

where they establish the latter equation making use of $\pi_i = A_{1,i}$ ([7], Lemma 3.6).

\# interpretation in fuzzy setting, what happens with complex eigenvalues?

3.2.6 Crispness objective

Röblitz [6] argues that the use of W has no stochastic interpretation in the fuzzy setting. Optimization of the trace of P_C makes no sense as it is similar to Λ2.8 and therefore independent of A. Defining

$$\mathcal{S} = ..\chi^T D_\pi \chi$$

she therefore suggests maximization of

$$I_3 := \text{trace}\,(\mathcal{S}) = \sum_{i,j=1}^{n} \frac{(A_{ij})^2}{A_{1,j}}.$$

which is similar to I_2 ....

Maximizing the trace minimizes the off-diagonal entries of $\mathcal{S}$ which means that the corresponding clustering χ is as crisp as possible.

3.2.7 Unconstrained Optimization

Due to the high number of inequality constraints 3.1 solving these linear or convex problems may still be very time consuming. Following Deuflhard and Weber [2, 7] we will now show how to turn this constrained into an unconstrained optimization problem, basically by enforcing the constraints after each iteration.

Define the set $F_A^{'}$ by the equality constraints

$$\begin{aligned} A_{i,1} &= \delta_{i,1} - \sum_{j=2}^{n} A_{ij},\ i = 1, ..., n \\ A_{1,j} &= -\min_{l=1,...,N} \sum_{i=2}^{n} X_{li} A_{ij},\ j = 1, ..., n. \end{aligned} \tag{3.3}$$

Comparing these equalities to 3.1 one easily checks that $F_A^{'} \subset F_A$. Furthermore, as Weber shows in [7] (Lemma 3.5), the vertex set $v\,(F_A) \subset F_A^{'}$, #containg the optimal solutions, is still contained in this restriced set.

Now consider the *feasabilization algorithm* $F : \mathbb{R}^{(n-1)\times(n-1)} \twoheadrightarrow F_A^{'}$, mapping any arbitrary matrix $\left(\tilde{A}_{ij}\right)_{i,j=2,...,n}$ to a feasible transformation matrix A.

Feasabilization algorithm

1. For $i = 2, ..., n$ define $\tilde{A}_{i,1} := -\sum_{j=2}^{n} \tilde{A}_{ij}$
2. For $j = 1, ..., n$ define $\tilde{A}_{1,j} := -\min_{l=1,...,N} \sum_{i=2}^{n} X_{li} \tilde{A}_{ij}$
3. For $i, j = 1, ..., n$ define $A_{ij} := \frac{\tilde{A}_{ij}}{\sum_{j=1}^{k} \tilde{A}_{1,j}}$

Steps 1 and 2 guarantee feasability of $\tilde{A}$ with respect to 3.3 for $i = 2, ..., n$ respectively $j = 1, ..., n$. As these equalities are linear in A they are invariant under scalar multiplication and step 3 now furthermore assures the equality 3.3 for $i = 1$. Thus F indeed maps to $F_A^{'}$.

Furthermore, taking any matrix $A \in F_A^{'}$, dropping the first row and column to get $\tilde{A}$ and computing $F\left(\tilde{A}\right) = A$ we see that F is surjective.

As any objective function I_i, $i = 1, 2, 3$ is convex over F_A it attains its maximum at one of the vertices $v(F_A)$. We thus can also optimize the function $F \circ I_i$ over $\mathbb{R}^{(n-1)\times(n-1)}$ and so have transformed the constrained optimization problem in n^2 unknows to an unconstrained in $(n-1)^2$ unknows.

Next we will develop an initial guess to this global optimization problem, turning it into a local one.

3.2.8 Inner simplex algorithm

Based on Weber and Galliat [8] we outline the *inner simplex algorithm*, determining an initial guess for the matrix A.

The first step, the *index mapping algorithm,* looks for the indices i_j of the succesively farthest linear independent rows. It starts by choosing the largest row vector as starting point, and then iteratively adds the points with the largest distance to the hyperplane spanned by the chosen points so far:

Index mapping algorithm

1. Find starting point: $i_1 := \text{argmax}_{j \in C} \|X_{\cdot,j}\|_2$
   Translate to origin: For $i \in S$ set $X_{i,\cdot} \leftarrow\!\shortmid X_{i,\cdot} - X_{i_1,\cdot}$
2. For $j = 2, ..., n$
   Find next index: $i_j := \text{argmax}_{j \in C} \|X_{\cdot,j}\|_2$
   Projection to hyperplane by Gram-Schmidt process: $X \leftarrow\!\shortmid X - \frac{\mathrm{X} X_{i_j,\cdot}^T \otimes X_{i_j,\cdot}}{\|X_{i_j,\cdot}\|_2}$

Once having determined the indices of the n extremal points, we now construct the matrix A mapping these to the vertices of Δ.

$$A(X) := (X_{ij})^{-1}_{i=i_1,...,i_n,\, j=1,...,n}$$

In the case of the *maximality assumption*, X spans a $(n-1)$-simplex, and the *index mapping algorithm* determines its vertices, thus $X \cdot \text{co}(x) = \Delta$ and $X \in v(F_A)$ maximizes I_1.

For the general case though Weber [7] (Lemma 3.13, Theorem 3.14) has shown that the following statements are equivalent:

1. The convex hull $\text{co}(X)$ of X is a simplex.
2. The result of the *inner simplex algorithm* is feasible, i.e. $A \in F_A$.
3. $A \in v(F_A)$ and therefore maximizes I_1.

Therefore the result is not feasible in the generic case.

If however the *maximality assumption* almost holds, i.e. the convex hull of X is a small pertubation of a simplex, which according to Weber [7] (3.4.4) is satisfied in most of the applications, the algorithm still gives a solution near the unperturbed solution.

Therefore that A is near a vertex of the set F_A and thus a good initial guess for a local optimization of the unconstraint optimization.

3.2.9 The PCCA+ Algorithm

1. Compute X, Λ as in 3.2.1
2. Determine the, in general infeasible, initial guess $A_0 := A(X)$ using the *inner simplex algorithm*.
3. Perform an iterative local optimization A_0, A_1, ... of the objective function I_1, I_2 or I_3. In each step $A_k \to A_{k+1}$ only update the elements $A_{k,ij}$, $i, j \neq 1$ without constraints. Then use algorithm *3.2.7* to get a feasible matrix A_k before evaluating the corresponding objective function.

As the *feasibalization algorithm* is not differentiable, Deuflhard and Weber [2] propose the use of the nonlinear simplex method of Nelder and Mead [5] as local optimization routine.

3.2.10 Determination of the number of clusters n

So far we have imposed a desired number of clusters n.

3.3 Nonreversible processes

The whole algorithm is also applicable to nonreversible processes, with some smaller adjustments concerning the construction of the invariant subspace and the stochastic interpretation, as well as a new optimization goal.

3.3.1 Stochastic interpretation of the coupling matrix

3.3.2 Construction of X and Λ

When the underlying stochastic process is not reversible the matrix P is no more real diagonalizable. We therefore make use of the *real Schur decomposition*, decomposing a matrix $A = QTQ^{-1}$ into a orthonormal matrix Q, called the *Schur vectors,* and an upper quasi-triangular (1-by-1 and 2-by-2 blocks on its diagonal) matrix T, called the *Schur form*. The columns of Q are called the Schur vectors of P. The eigenvalues of P appear on the diagonal of T, where complex conjugate eigenvalues correspond to the 2-by-2 blocks.

To compute an orthonormal basis for an invariant subspace belonging to n eigenvalues one can reorder the diagonal blocks of T such that the upper left $n \times n$ block contains these n eigenvalues. Then the first n columns of the updated transformation matrix Q form a basis for the desired subspace [**?**].

So we define, analogously to 3.2.1 $\tilde{P} := D_\eta^{\frac{1}{2}} P D_\eta^{-\frac{1}{2}}$ and compute the *real Schur decomposition* of $\tilde{P}$. We then select the $n \times n$ blocks corresponding to the n eigenvalues with the highest absolute value by the reordering procedure. Let us denote the resulting *Schur vectors* by $\tilde{X}$ and the *Schur form* by Λ.

Now $X := D_\pi^{-\frac{1}{2}} \tilde{X}$ and Λ satisfy the conditions for 1 (same calculations as in3.2.1).

3.3.3 Optimization

Not only metastable, but also cyclic dynamics of interest.

4 Application to eyetracking data

This algorithm was applied to experimental eye-tracking data obtained by the department of psychology of the Universität Potsdam, with the goal to detect objects as metastable clusters using just the dynamics of the human eye, i.e. without any data of the image itself.

4.1 The experiment

A group of test persons was presented an image for the duration of ...

The eye-tracker measures the fixations $f_i \in \mathbb{R}^2$ of the test persons eyes and their respective time $t_i \in \mathbb{R}$.

This experiment was repeated with the same images mirrored horizontally, to test in how far our perception of images is influenced by its horizontal orientation.

4.2 Implementation

As state space we define the coordinates of each fixation $S := \{s_i\}\,,\ s_i = f_i$, though if one has too many fixations one can precluster the fixations for example with k-means.

We then compute the membership of each fixation f_i to each state s_j based on the row-sum normalized gaussian of the distance:

$$M_{ij} := \frac{e^{\frac{|f_i - s_j|^2}{2\sigma^2}}}{\sum_j e^{\frac{|f_i - s_j|^2}{2\sigma^2}}}$$

// check the 2

This assures that nearby fixations "overlap", introducing the metric distance information contained in the fixation data to the markov process.

We then fix a step time $\tau = 50$ms for the markov chain to compute the transitions along this time-grid.

Then summing the state correspondances over all fixation transitions $f_a \to f_b$, and normalizing to row sum 1 we define the *transition matrix* P:

$$P_{ij} \quad = \quad \frac{\sum_{f_a \to f_b} M_{ai} M_{bj}}{\sum_k \sum_{f_a \to f_b} M_{ai} M_{bk}}$$

We define the stationary distrubution at the i-th state as the normalized amount of counted transitions from that state:

$$\pi_i = \frac{\sum_{f_a \to f_b} M_{ai}}{\sum_i \sum_{f_a \to f_b} M_{ai}}$$

Once we have constructed P this way we now compute the invariant eigenspace using the weighted Schur decomposition as in 3.3.2 and pass is to PCCA+, which in return gives us the clustering χ, with which we also can compute the *coupling matrix* on our reduced state space.

To allow for comparison of two different experiments we reorder the clusters such that their respective correlation is maximized:

We define the correlation of two clusters as sum over all products of the gaussians of the states weighted with their cluster share. We now use the Hungarian algorithm to reorder the matrix to maximize the diagonal.

For the comparison of two experiments, A and B, different methods were implemented:

- Direct: One can either compare the different *coupling matrices* directly which each other, using their respective clustering.
- Coupling average: Apply each resulting clustering to each experiment and compute the average of the difference of the coupling matrices.
- Cluster average: Compute the average clustering and compare the corresponding *coupling matrices* directly.

4.3 Results

5 Junk

5.0.1 Stochastic interpretation

By multiplying the conditional transitions of P with the actual starting distribution we get the unconditional transitions matrix $D_\eta P$, which we can interpret as amount of actual transitions between the states. Now multiplying with a membership vector χ_i^T from the left gives the numbers of transitions starting in χ_i and finally multiplying with χ_j from the right measures the amount of these transitions landing in χ_j, i.e. $\chi_i^T D_\eta P \chi_j$.

gives the unconditional number of transitions from cluster i to cluster j. Note that this interpretation is associative in the sense that we come to the same results by for example first interpreting $\chi_i^T D_\eta$ as actual amount of states in the cluster χ_i and measure the overlap with where the transitions to χ_j, come from: $\left(\chi_i^T D_\eta\right)(P\chi_j)$.

We can vectorize this equation to compute the unconditional transitions by means of the *membership matrix* $\chi = (\chi_1, ..., \chi_n)$*:*

$$\chi^T D_\eta P \chi$$

We then turn this unconditional transitions into conditinal by dividing by the number of states belonging to χ_i and have to make up the fact that we do not count all transitions... Baustelle

References

[1] P. Deuflhard, W. Huisinga, A. Fischer, and Ch. Schütte. Identification of almost invariant aggregates in reversible nearly uncoupled markov chains. *Linear Algebra and its Applications*, 315:39–59, 2000.

[2] P. Deuflhard and M. Weber. Robust perron cluster analysis in conformation dynamics. *Linear Algebra and its Applications*, 398:161–184, 2005.

[3] Wilhelm Huisinga. *Metastability of Markovian systems: A transfer operator based approach in application to molecular dynamics*. PhD thesis, Free University Berlin, 2001.

[4] S. Kube and M. Weber. A coarse graining method for the identifiation of transition rates between molecular conformations. *Journal of Chemical Physics*, 126(2), 2007.

[5] J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965.

[6] S. Röblitz and M. Weber. Fuzzy spectral clustering by pcca+: application to markov state models and data classification. *Advances in Data Analysis and Classification*, 7:147–179, 2013.

[7] M. Weber. *Meshless methods in Conformation Dynamics*. PhD thesis, Free University Berlin, 2006.

[8] M. Weber and T. Galliat. Characterization of transition states in conformational dynamics using fuzzy sets. *ZIB-Report*, 02-12, 2002.