Abstract: In this paper we study Hilbert space embeddings of dynamical systems and embeddings generated via dynamical systems. This is achieved by following the behavioural framework invented by Willems, namely by comparing trajectories of states. As important special cases we recover the diffusion kernels of Kondor and Lafferty, generalised versions of directed graph kernels of Gärtner, novel kernels on matrices and new similarity measures on Markov Models.

Keywords: Dynamical Systems, Reproducing Kernel Hilbert Space, Kernel Methods, Behavioural Framework, Graphs, Markov Models

1. INTRODUCTION

When dealing with dynamical systems, one may compare their similarities by checking whether they satisfy similar functional dependencies. With suitable parameterizations this is useful in determining when systems are similar. However, it is not difficult to find rather different functional dependencies, which, nonetheless, behave almost identically, e.g., as long as the domain of initial conditions is sufficiently restricted. For instance, consider the maps

\[ x \leftarrow a(x) = |x|^p \quad \text{and} \quad x \leftarrow b(x) = \min(|x|^p, |x|) \quad (1) \]

for \( p > 1 \). While \( a \) and \( b \) clearly differ, the two systems behave identically for all initial conditions satisfying \( |x| \leq 1 \). This example may seem contrived and it is quite obvious from (1) that identical behaviour will occur in this case, yet for more complex maps and higher dimensional spaces such statements are not quite as easily formulated.

This leads to the approach taken in the present paper, namely to compare trajectories of dynamical systems and derive measures of similarity from them. On graphs, for instance, which are described by their adjacency matrix (or the so called graph Laplacian), this corresponds to comparing paths followed by a diffusion process (Kondor and Lafferty, 2002). The advantage of our approach is that it is independent of the parameterization of the system, its downside being that one needs efficient mechanisms to compute the trajectories, which may or may not always be available. This approach is in spirit similar to the behavioural framework of (Willems, 1986a; Willems, 1986b; Willems, 1987), which identifies systems by identifying trajectories.

We begin with some definitions. For the remainder of the paper we assume that the state space \( \mathcal{X} \), with \( x \in \mathcal{X} \), is a Hilbert space. This is not a major restriction, since e.g., any countable set \( S \) can be made into a Hilbert space by mapping it into \( \ell^2(S) \). Similar choices can be made for non-countable sets.\(^1\)

Moreover we denote by \( \mathcal{A} \) the set of time evolution operators, such that

\[ x_A(t) := A(t)x \quad \text{for} \quad A \in \mathcal{A}. \quad (2) \]

Here \( x \) is subjected to the dynamics introduced by \( A \) and \( t \in \mathcal{T} \) is the time of the measurement. We will choose \( \mathcal{T} = \mathbb{N}_0 \) or \( \mathcal{T} = \mathbb{R}_+ \), depending on whether we wish to deal with discrete-time or continuous-time systems.

Note that \( A \) may be a nonlinear operator and that both \( x \) and \( A \) may be random variables rather than

\(^1\) Should further generality be required, one may use algebraic semi-ring methods proposed by (Cortes et al., 2002), albeit at a significant technical complication. This leads to rational series and functions on them.
deterministic variables. In those cases, we assume that both $x$ and $A$ are endowed with suitable probability measures. For instance, we may want to consider initial conditions with additive noise.

Finally, we assume that there exists a valid probability measure $\mu$ on $\mathcal{F}$, such that $\mu(\mathcal{F}) = 1$ and no $t \in \mathcal{F}$ has nonzero measure (or density). For the sake of analytical tractability we will assume that for some $\lambda > 0$

$$\mu(t) = \lambda^{-1} e^{-\lambda t} \text{ for } \mathcal{F} = \mathbb{R}_0^+ \quad (3)$$

$$\mu(t) = \frac{e^{-\lambda t}}{1 - e^{-\lambda}} \text{ for } \mathcal{F} = \mathbb{N}_0. \quad (4)$$

Such exponential discounting is a popular choice in reinforcement learning (Sutton and Barto, 1998; Baxter and Bartlett, 1999) and control theory. In our case it will ensure convergence of integrals.

2. DOT PRODUCTS ON TRAJECTORIES

We define a dot product on the trajectories by taking expectations over the dot product on the coordinates. This leads to a Hilbert space on $\mathcal{X}^\mathcal{F}$ via

$$\langle \theta, \theta' \rangle := E_\mu [\langle \theta(t), \theta'(t) \rangle] \text{ for } \theta, \theta' \in \mathcal{X}^\mathcal{F}. \quad (5)$$

If we now identify $(x, A)$ with the trajectory $x_A(\mathcal{F}) \in \mathcal{X}^\mathcal{F}$, we have a dot product between $(x, A)$ pairs:

$$k((x, A), (\tilde{x}, \tilde{A})) := E_\mu [\langle A(x), A(\tilde{x}) \rangle]. \quad (6)$$

Here $k$ was used as a shorthand for a dot product. It is in general referred to as a Mercer kernel and it can be used to solve numerous problems in machine learning (Wahba, 1990; Schölkopf and Smola, 2002). Note that (6) or any other Mercer kernel $k$ defines a dot product and hence allows us to define a metric on its arguments. This implies that, as per our model, two pairs of initial conditions and time propagation operators are identical if they produce the same trajectory.

One may derive further kernels from (6) simply by using any of the composition rules for kernels, such as taking integer powers of kernels, i.e., $k' = k^n$, exponentiation of kernels $k' = e^{x}$, convex combination, normalization in feature space, or any composition thereof. (Haussler, 1999) describes a large class of such transformations in great detail.

2.1 Kernels on Dynamical Systems

Rather than specifying a set of initial conditions explicitly we may wish to compare two dynamical systems, yielding a kernel $k(A, \tilde{A})$. This can be achieved by restricting $k((x, A), (\tilde{x}, \tilde{A}))$ to $x = \tilde{x}$, i.e., we compare only trajectories for identical initial conditions, and the expectation over $x$, should we be interested in the behaviour of $A, \tilde{A}$ over a range of $x$. Consequently we obtain

$$k(A, \tilde{A}) := E_x [k((x, A), (x, \tilde{A}))]. \quad (7)$$

This is a convex combination of kernels, hence (7) is a kernel. From the fact that $\mathcal{F}$ is a Hilbert space itself and the requirement that no $t \in \mathcal{F}$ have nonzero measure (or density) it follows that the canonical metric

$$d(A, \tilde{A})^2 := k(A, A) + k(\tilde{A}, \tilde{A}) - 2k(A, \tilde{A}) \quad (8)$$

is zero only if the trajectories of $x_A(t)$ and $x_{\tilde{A}}(t)$ are identical for all $x \in \mathcal{X}$ with nonzero probability measure $p(x)$.

We are therefore able to determine the proximity between various dynamical systems with respect to their initial conditions. If $A, \tilde{A}$ should happen to be random variables, we can generalise the setting by taking expectations over each of the random variables independently to obtain

$$k(\mathcal{A}, \mathcal{A}) := E_A E_{\tilde{A}} [k((x, A), (x, \tilde{A}))]. \quad (9)$$

Such situations may occur, e.g., when $\mathcal{A}, \mathcal{A}$ are so-called lattices. Nondeterministic finite state automata can also be described in this fashion. See also (Cortes et al., 2002).

2.2 Kernels via Dynamical Systems

In complete symmetry to the reasoning in the previous section we might also want to compare initial conditions with respect to a fixed class of dynamical systems and therefore define

$$k(x, \tilde{x}) := E_A [k((x, A), (\tilde{x}, A))]. \quad (10)$$

In this context, we will consider initial conditions as similar, if the trajectories they induce in a dynamical system are similar. For instance, source code can be considered similar, if the dynamical system (here the compiler or possibly a set of various compilers) treats the code similarly and produces similar parse trees and/or executables.

Again, if $(x, \tilde{x})$ themselves are random variables, we may extend (10) to

$$k(x, \tilde{x}) := E_x E_{\tilde{x}} E_A [k((x, A), (\tilde{x}, A))] \quad (11)$$

by taking expectations over the random variables themselves. Again the canonical metric induced by $k$ is zero only when two initial conditions lead to identical trajectories. This means that, e.g., for a dynamical system with various basins of attraction, rather than using the distance in the initial conditions, the overall distance of the trajectories is used.

3. LINEAR SYSTEMS

A special, yet important case to consider are linear, time-invariant systems, where time propagation oc-
curs as a linear function of the current state. We begin with two technical lemmas:

**Lemma 1.** Denote by $A, B$ linear operators on $\mathcal{X}$ with $\|A\|, \|B\| \leq \Lambda$ for some $\Lambda > 0$. Then for all $\lambda$ with $e^\lambda > \Lambda^2$ and for all linear operators $W : \mathcal{X} \to \mathcal{X}$ the series

$$ M := \sum_{i=0}^{\infty} e^{-\lambda t} A_i W B^i $$

converges and $M$ can be computed by solving the Sylvester equation $e^{-\lambda} AMB + W = M$.

Note that Sylvester equations of type $AXB + CXD = E$ can be readily solved at $O(n^3)$ time (Gardiner et al., 1992) with freely available code ($A, B, C, D \in \mathbb{R}^{n \times n}$).

**Proof.** To show that $M$ is well defined we use the triangle inequality, leading to

$$ \|M\| = \left\| \sum_{i=0}^{\infty} e^{-\lambda t} A_i W B^i \right\| \leq \sum_{i=0}^{\infty} \left\| e^{-\lambda t} A_i W B^i \right\| \\ \leq \sum_{i=0}^{\infty} \left( e^{-\lambda t} \Lambda^2 \right)^i \|W\| = \frac{\|W\|}{1 - e^{-\lambda t} \Lambda^2}. $$

Next we decompose the sum in $M$ to obtain

$$ M = A^0 WB^0 + \sum_{i=1}^{\infty} e^{-\lambda t} A_i W B^i $$

$$ = W + e^{-\lambda t} A \left( \sum_{i=0}^{\infty} e^{-\lambda t} A_i W B^i \right) B = W + e^{-\lambda t} AMB. $$

A similar result holds for continuous time systems:

**Lemma 2.** Denote by $A, B, W$ linear operators $\mathcal{X} \to \mathcal{X}$ such that $\|A\|, \|B\| \leq \Lambda$. Then, for all $\lambda > 2\Lambda$ the integral

$$ M := \int_0^{\infty} e^{-\lambda t} \exp(A t) W \exp(B t) dt $$

converges and $M$ is the solution of the Sylvester equation $(A^\top + \frac{1}{2} I) M + M (B + \frac{1}{2} I) = -W$.

**Proof.** (Sketch only) Convergence follows from the triangle inequality and $M$ can be rewritten to satisfy the self-consistency condition by partial integration. ■

3.1 Discrete-Time Systems

Here we assume that time propagation occurs as

$$ x_A(t + 1) = A x_A(t) + a_i + \xi_i $$

where $A : \mathcal{X} \to \mathcal{X}$ is a linear operator (typically a matrix), $a_i \in \mathcal{X}$ is a linear offset, and the noise variables $\xi_i$ are assumed independent with zero mean and covariance $C$. Repeated substitution of (14) allows us to write $x_A(t)$ as

$$ x_A(t) = A^t x_0 + \sum_{i=0}^{t} A^{t-i} \xi_i + A^{t-i} a_i. $$

(15)

$\langle x_A, \hat{x}_A \rangle$ can be computed in closed form by straightforward algebra (mainly rearranging sums and exploiting harmonic series expansions). To simplify our presentation we henceforth assume that $a_i = 0$ and obtain

$$ \sum_{t=0}^{\infty} e^{-\lambda t} \langle A^t x_0, \Lambda^t \hat{x}_0 \rangle $$

$$ = x_0^\top \left( \sum_{j=0}^{\infty} e^{-\lambda j} (A^j)^\top \Lambda^j \right) \hat{x}_0 = x_0^\top M \hat{x}_0 = \text{tr}[\langle \hat{x}_0, x_0 \rangle M] $$

$$ = \text{tr} \left( \sum_{t=0}^{\infty} \sum_{j=0}^{\infty} e^{-\lambda (t+j)} (A^{t+j})^\top \Lambda^{t+j} C \right) $$

$$ = \text{tr} \left( \sum_{j=0}^{\infty} e^{-\lambda j} (A^j)^\top M \Lambda^j C \right) = \text{tr}(CM). $$

Here $M, \Lambda$ satisfy the conditions of Lemma 1, that is $e^{-\lambda} A^\top M \Lambda + 1 = M$ and $e^{-\lambda} A^\top \Lambda M \Lambda + M = \Lambda$. This leads to

$$ k((A, \hat{x}), (A, \hat{x})) = \text{tr}(CM) + x_0^\top M \hat{x}_0. $$

(16)

Following the reasoning of Section 2.1 we obtain a kernel on the initial conditions $(x_0, \hat{x}_0)$ by setting $A = \hat{A}$. This simply results in a modified metric tensor $M$ plus the offset $\text{tr}(CM)$ with respect to the normal scalar product. In other words

$$ k(x_0, \hat{x}_0) = \text{tr}(CM) + x_0^\top M \hat{x}_0. $$

(17)

Furthermore, following the reasoning of Section 2.2 we get a kernel on dynamical systems by taking the expectation over the initial conditions. Assuming a covariance $\hat{C}$ for the initial conditions we have

$$ k(A, \hat{A}) = \text{tr}(\hat{C} M) + \text{tr}(CM). $$

(18)

A nice side-effect is that this allows us to measure similarities between objects such as matrices based on their spectral properties. See (Vishwanathan, 2002, Chapter 6) for more details.

Two special cases are worth-while considering (we assume $C = 0$ for simplicity):

- If $A = \hat{A}$ we can solve the Sylvester equation in closed form as $M = (1 - e^{-\lambda} A^\top A)^{-1}$. This shows that the RKHS norm of $A$, as given by $\sqrt{k(A,A)}$ depends on how close to $e^\lambda$ the eigenvalues of $A^\top A$ are: for large singular values the RKHS norm is large.
- If $A^\top \hat{A} = 0$ the dynamics induced by $A$ and $\hat{A}$ differ as much as possible. Here all terms in the infinite sum except for the one due to the initial conditions vanish and we obtain $k(A, \hat{A}) = \text{tr} C$. 

3.2 Continuous-Time Systems

Next we study continuous-time systems, where $x_A(t)$ satisfies the stochastic differential equation

$$\frac{d}{dt}x_A(t) = Ax_A(t) + a(t) + \xi(t). \quad (19)$$

Here $\xi(t)$ is a stochastic process with $E[\xi(t)] = 0$ (in many applications we set $\xi(t) = 0$). Standard algebra shows that (19) can be solved by

$$x_A(t) = \exp(At)x_0 + \int_0^t \exp(A(t-\tau))(a(\tau) + \xi(\tau))d\tau. \quad (20)$$

As before, we need to compute $k((x,A),(\tilde{x},\tilde{A}))$ and take expectations over $\xi$ to eliminate the dependency on the random variable $\xi(t)$. For certain types of $a(t)$ this can be done in closed form.

To keep equations simple and for the purpose of conveying the basic idea we will limit ourselves to the special case of $a(t) = \xi(t) = 0$. This leads to

$$k((x,A),(\tilde{x},\tilde{A})) = \int_0^t e^{-\lambda t}(\exp(At)x_0,\exp(\tilde{A}t)x_0^\top)dt. \quad (21)$$

The latter can be solved using Lemma 2 and we obtain

$$k((x,A),(\tilde{x},\tilde{A})) = \lambda^{-1}x^\top Mx. \quad (22)$$

Here $M$ satisfies $(A^\top + \frac{1}{2} I)M + M^\top(\tilde{A} + \frac{1}{2} I) = I$.

If we fix $A = \tilde{A}$ to obtain kernels on initial conditions this leads to

$$k(x,\tilde{x}) = x^\top Mx. \quad (23)$$

In this case we can solve the Sylvester equation to obtain $M = \frac{1}{2}(A + \frac{1}{2} I)^{-1}$. Likewise, if we wish to study kernels on dynamical systems we obtain

$$k(A,\tilde{A}) = \text{tr}(MC). \quad (24)$$

where $C$ is the covariance over the initial conditions. It is easy to see that both discrete and continuous-time dynamical systems lead to kernels with very similar structure. This is not surprising, given the similar differential equations (in fact, the discrete case is often used as an approximation for the continuum).

In the following we study two special cases of dynamical systems:

**Snapshots:** If $\mu(t) = \delta_T(t)$, that is, if we only care about a snapshot of the dynamical system at time $T$, we obtain

$$k((x,A),(\tilde{x},\tilde{A})) = x_0 \exp(At) \exp(\tilde{A}t)^\top x_0^\top. \quad (25)$$

For a suitable choice of $A$ this leads to kernels on graphs (Kondor and Lafferty, 2002; Vishwanathan, 2002).

**Differential Equations:** It is well known that linear differential equations can be transformed into first order linear differential equation by including the state space (Hirsch and Smale, 1974). This means that differential equations thereby impose a kernel on the initial conditions. Similarity here corresponds to correlation of the state space trajectories.

Likewise we can introduce a metric on the space of linear differential equations based on the similarity properties in their time evolution behaviour.

The kernels described above, appear somewhat simple minded at first sight. After all, with regard to initial conditions we are only replacing the standard Euclidean metric by the covariance of two trajectories under a dynamic system. The next section will show several useful and relevant cases.

4. MARKOV PROCESSES

Markov processes have the property that their time evolution behaviour depends only on their current state and the state transition properties of the model. Denote by $S$ the set of states, then for $x \in \mathbb{R}^3$ the dynamics are given by

$$x(t+1) = Ax(t) \quad (26)$$

for discrete-time and continuous time processes respectively. In this particular case $x$ is the vector of state probabilities and $A$ is the state transition matrix, that is for discrete processes $A_{ij} = p(i|j)$. Here $p(i|j)$ is the probability of reaching state $i$ from state $j$.

4.1 General Properties

In Discrete-Time Markov Processes $k(i,j)$ computes the average overlap between the states when originating from $x = e_i$, $\tilde{x} = e_j$. Here $e_i$ are “pure” states, that is the system is guaranteed to be in state $i$ rather than in a mixture of states.

Since $A$ is a stochastic matrix (positive entries with row-sum 1), its eigenvalues are bounded by 1 and therefore, any discounting factor $\lambda > 0$ will lead to a well-defined kernel.

Note that the average overlap between state vectors originating from different initial states are used in the context of graph segmentation and clustering (Weiss, 1999; Harel and Koren, 2001). This means that $\mu(t)$ is nonzero only for some $t \leq t_0$, which is similar to heavy discounting.

Recall, however, if $e^\lambda$ is much smaller than the mixing time, $k$ will almost exclusively measure the overlap between the initial states $x, \tilde{x}$ and the transient distribution on the Markov process. The quantity of interest here will be the ratio between $e^\lambda$ and the gap between 1 and the second largest eigenvalue of $A$ (Graham, 1999).

An extension to Continuous-Time Markov Chains (CMTC) is straightforward. Again $x(t)$ corresponds to the state at time $t$ and the matrix $A$ (called the rate matrix in this context) denotes the differential change in the concentration of states.
When the CTMC reaches a state, it stays there for an exponentially distributed random time (called the state holding time) with a mean that depends only on the state. For instance, diffusion processes can be modelled in this fashion. Clearly we can define kernels using diffusion on CTMCs by plugging the rate matrix $A$ into (20).

4.2 Graphs

An important special case of Markov processes are random walks on (directed) graphs. Here diffusion through each of the edges of the graph is constant (in the direction of the edge). This means that, given an adjacency matrix representation of a graph via $E$ (here $E_{ij}$ denotes the weight on an edge from vertex $j$ to vertex $i$), we define a diagonal matrix $D_{ii} := \sum_j E_{ij}$ to compute the Laplacian $L = D - E$ and the normalized Laplacian $\tilde{L} := D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$ of the graph, and use the former to define a diffusion process $\frac{\partial}{\partial t}x(t) = Lx(t)$.

- Using the snapshot weights where we measure the overlap at time $t$, we obtain
  \[ K = \exp(tL)^\top \exp(tL) \]  
  as covariance matrix between the distributions over various states. $K_{ij}$ therefore equals the probability that any other state $l$ could have been reached jointly from $i$ and $j$ (Kondor and Lafferty, 2002).

- The time $T$ chosen for the snapshot is typically user defined and consequently it tends to be debatable. On the other hand, we might as well average over a large range of $T$ (as typically done in Section 3.2), leading to a kernel matrix
  \[ K = \frac{1}{2} \left( \tilde{L} + \frac{\lambda}{2} \mathbf{1} \right)^{-1}. \]  
  This yields a kernel whose inverse differs by $\frac{1}{2} \mathbf{1}$ from the normalized graph Laplacian (another important quantity used for graph segmentation). The attraction of (26) is that its inverse is easy to come by and sparse, translating into significant computational savings for estimation.

- The kernel proposed by (Gärtner, 2002) can be recovered by setting $W$ (the normalizing term in Lemma 1) to have entries $\{0,1\}$ according to whether vertices $i$ and $j$ bear the same label and considering a discrete time random walk rather than a continuous time diffusion processes (various measures $\mu(t)$ take care of the exponential and harmonic weights).

- Finally, discrete-time Markov processes can be treated in complete analogy, yielding either $K = (\mathbf{I} - A)^{-1}$ or similar variants, should snapshots be required.

(Kondor and Lafferty, 2002) suggested to study diffusion on undirected graphs. Their derivations are a special case of (23). Note that for undirected graphs the matrices $E$ and $L$ are symmetric. This has the advantage that (23) can be further simplified to

\[ k(x, \tilde{x}) = x^\top \exp(LT)^\top \exp(LT) \tilde{x} = x^\top \exp(2LT) \tilde{x}. \]

5. DISCUSSION

5.1 Nonlinear Systems

In nonlinear systems, in general, a closed-form computation of (6) will not be possible. Instead, we will need to resort to approximate solutions, mainly via numerical quadrature. This means that we will need to simulate trajectories in order to compare them. The computational cost depends on the discounting factor $\lambda$, since we only need to compute summands of $k$ until their contribution vanishes, which happens exponentially fast (with $e^{-\lambda t}$ controlling the speed).

While this may not seem satisfactory at first, we are confident that further efficient methods for computing trajectories analytically will be found. Of particular interest in this context are techniques for dealing with piecewise linear systems, as described in (Johansson and Rantzer, 1998).

Moreover, the area of linear matrix inequalities and control is appealing for the design of kernels, given the similarities that our calculations exhibit with some standard techniques in this area.

5.2 Pair-HMM Kernels

(Watkins, 2000) introduced the concept of Pair-HMMs for the purpose of computing kernel functions on sequences. While the focus was mainly on biological sequence analysis, a kernel computing

\[ k(x, \tilde{x}) := p(x|\sigma)p(\tilde{x}|\sigma)p(\sigma) \]  

is clearly also useful for the comparison of temporal sequences. Here $\sigma$ denotes a path through the Pair-HMM (a Hidden Markov Model with a conditional independence property for the emissions) and the average is taken over all paths. Efficient computation is ensured via dynamic programming, thus bounding the (otherwise) exponential number of terms in (27).

It would be interesting to see whether, rather than using sequences as arguments, one may also use automata directly as inputs for such similarity measures, while being able to perform dynamic programming on the triple of states (for the two inputs plus the automaton). The advantage of (27) over our “naive” correlation approach is that it can deal better with temporally misaligned data, that is, if particular events occur in a qualitatively similar, yet slightly time-shifted version.
5.3 Summary and Outlook

The current paper sets the stage for kernels on dynamical systems as they occur frequently in linear and affine systems. By using correlations between trajectories we were able to compare various systems on a behavioural level rather than a mere functional description. This allowed us to define similarity measures in a natural way.

While the larger domain of kernels on dynamical systems is still untested, special instances of the theory have proven to be useful in areas as varied as classification with categorical data (Kondor and Lafferty, 2002; Gärtner, 2002) and speech processing (Cortes et al., 2002). This gives reason to believe that further useful applications will be found shortly.

For instance, we could use kernels in combination with novelty detection to determine unusual initial conditions, or likewise, to find unusual dynamics. In addition, we can use the kernels to find a metric between objects such as HMMs, e.g., to compare various estimation methods.

Acknowledgements

Parts of this work were supported by the Australian Research Council. SVNV was supported by grants from Netscaler Inc. and Trivium India Software as well as an Infosys fellowship. We thank Laurent El Ghaoui, Patrick Haffner, Daniela Pucci de Farias, and Bob Williamson for helpful discussions.

6. REFERENCES


