# CMP9764M Learning time-invariant dynamical systems

line 1: 2nd Stephen Rerri-Bekibele
line 2: *School of Computer Science. University of Lincoln (of Affiliation)*
line 3: Lincoln, United Kingdom
line 4: 16663359@students.lincoln.ac.uk

## Table of Contents

***Abstract*—Time Invariant Dynamics systems are a set of Dynamic Movement Primitives that guarantee the convergence of a system when control theory is used. This paper compares the advantages and disadvantages of A Time Invariant Dynamic System in comparison to a Time Dependent System.**

***Keywords*—*Time Dependent System, Time Invariant System, Convergence, control theory* (*key words*)**

## INTRODUCTION

A linear, time-invariant (LTI) systems is a Physical system that can be modelled with reasonable engineering fidelity [1]. In advanced Robotics a Time Invariant Dynamical system is a set of Dynamic Movement Primitives

$$\dot{x} = f(x)$$

Where x is the state and s, a time or phase variable.

Time-invariant dynamical systems (TIDS) are an alternative to a Time-dependent dynamical systems (TDDS).

TIDS systems are robust to perturbations but are a challenging machine learning problem. A Time-invariant dynamic system ensures stability and convergence when a control theory such as Lyapunov theorem is used.

TDDS are a set of Dynamic Movement Primitives (DMPs) $\dot{x} = f(x, s)$

Where x is the state and s is a time or phase variable for states

$$s \in [1, 0]$$

TDDS are easy machine learning problems to solve however they are not robust to perturbations.

If the convergence of a system is defined as its ability to reach its steady state. Then Convergence is guaranteed when using Dynamical Systems

A disadvantage of using TIDS is that the state x as defined above is usually multi-dimensional, and this is a challenging machine learning problem. i.e if the state represented a manipulator that can have 67 dimensions of freedom, then the machine learning model would struggle to learn and lead to an unstable system.
A method that guarantees the stability of a system is a control theory method called the Lyapunov theorem, this maps our state to scalar values that belong to real numbers. In other for the function to be a valid Lyapunov function there are certain conditions:

1. $V(x_*) = 0$, stability at the steady-state

The function should be equal to 0 at a steady state where x* is the point in space that we want our trajectory to reach, the goal of the trajectory.

2. $V(x) > 0 \forall x \in \Re$,

Second the function must be positive for each x in the space whilst also being more than 0 except when it is at the steady state as defined in the previous equation.

3. $\frac{dV}{dt} < 0$, convergence

Lastly, the time derivative must be less than 0 (negative) to make the motion converge at the desired point.
As the time elapses, the system should release energy until it reaches steady state.

## PART 1 LITERATURE REVIEW

In "Discrete Time Adaptive Control of Linear Dynamic Systems with a Two-Segment Piecewise-Linear Asymmetric Non Linearity", a method is presented for constructing the discrete time adaptive control of a model, a nonlinear system [3]. The system is given a cascade connection of a "two segment piecewise-linear asymmetric non-linearity" followed a linear system. To make the parameter identification time invariant, a switching gain sequence is also introduced and this algorithm ensures that the closed-loop system is globally convergent and stable for both the deterministic and stochastic cases.

Stability is achieved in the non-linear system by designing an adaptive control law to achieve closes-loop stability. The proof of global convergence for deterministic and stochastic case is always based on the elementary properties of a parameter estimation algorithm.

Results showed that under suitable conditions, the discrete time adaptive algorithm will be globally convergent in both stochastic and deterministic cases.

In this second paper on "Distributed state estimation for linear time-invariant dynamical systems: A review of theories and algorithms", a section on Distributed Kalman filtering focuses on the research of the distributed state estimation problem for a LTI system with Gaussian noise[4].

The paper states that the "consensus-based distributed Kalman filters feature consensus iterations implemented at each agent to agree on some global information. In this sense, the stabilization of the Kalman filter under joint observability becomes possible". Therefore, in order for the Kalman filter to be stabilized there needs to be a way for the agents to agree on some global information.

In works by (R.Olfati-Saber, 2006) however a method to combine the consensus algorithm with the distributed Kalman filtering to solve the distributed estimation problem was provided and analysed in detail. The

A last paper by "extends the rational Krylov subspace algorithm from the computation of the action of the matrix exponential to the solution of stable dynamic systems"

$$\tilde{A}\left(\frac{d}{dt}\right)u(t) = b(t), \quad u|_{t<0} = 0, \quad \tilde{A}\left(\frac{d}{dt}\right) = \sum_{i=0}^{m} A_i\left(\frac{d}{dt}+sI\right)^i,$$

(Druskin .V et al). In which an equation: $m \in \mathbb{N} \cup \{\infty\}, A_i = A_i^* \in \mathbb{R}^{N\times N}, s \le 0$, and $u(t), b(t) \in \mathbb{R}^N$, $b|_{t<0} = 0$ , not assuming that evolution of b(t) is described by a low dimensional subspace. And the reduction of said equation reaches convergence and is stable.

The paper is highly mathematically based and the results showed that error estimates allow the incorporate arbitrary time-variable casual right hand sides. Possibly showing a new approach to constructing exponential integrals.

Moreover, the non-linear numerical range of a dynamical system defines behaviour of its structure-preserving projection.

All in all, there are some unresolved algorithmic issues that should be addressed in the future.

## PART 2 LYAPUNOV STABILITY FOR TIME-INVARIANT DYNAMICAL SYSTEM AND STABLE ESTIMATOR OF DYNAMICAL SYSTEMS

Linear dynamical systems are dynamical systems whose equation functions are linear. Dynamic systems in general do not have closed-form solutions, however a linear dynamical system can be solved exactly and they have a rich set of mathematical properties.

In a Linear dynamical system, the variation of a state vector equals a constant matrix multiplied by x where x is an N-dimensional vector and A is a matrix. This variation can take two forms, either as a flow in which x varies continuously with time:

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t)$$

Or as a mapping in which x varies in discrete steps:

$$\mathbf{x}_{m+1} = \mathbf{A}\mathbf{x}_m$$

These equations are linear in the sense that if x(t) and y(t) are two valid solutions then so is any linear combination of the two solutions, e.g., $\mathbf{z}(t) \stackrel{\text{def}}{=} \alpha\mathbf{x}(t) + \beta\mathbf{y}(t)$ where alpha and beta are two scalars and it is irrelevant if Matrix A is symmetrical.

If the initial vector $\mathbf{x}_0 \stackrel{\text{def}}{=} \mathbf{x}(t=0)$ is aligned with a right eigenvector $r_k$ of the matrix A, the dynamics are simple

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{A}\mathbf{r}_k = \lambda_k\mathbf{r}_k$$

Where $\lambda_k$ is the correspoinding eigenvalue, the solution of the equation is: $\mathbf{x}(t) = \mathbf{r}_k e^{\lambda_k t}$

This can be confirmed with substitution.

If A is non-defective then any vector in an N-dimensional space can be represented by a linear combination of right and left eigenvectors (denoted $1_k$) of the matrix A.

$$\mathbf{x}_0 = \sum_{k=1}^{N} (\mathbf{l}_k \cdot \mathbf{x}_0)\,\mathbf{r}_k$$

Therefore, the general solution for x(t) is a linear combination of the individual solutions for the right eigenvectors.

$$\mathbf{x}(t) = \sum_{k=1}^{n} (\mathbf{l}_k \cdot \mathbf{x}_0)\,\mathbf{r}_k e^{\lambda_k t}$$

SEDS which stands for Stable Estimator of Dynamical Systems is a powerful method to tackle Dynamic systems. SEDS learns that parameters of the system to ensure that all motion follows the demonstrations closely while ultimately reaching in and stopping at the target.

Given a training set from a demonstration:

$\mathbb{D}\{x_1, \dot{x}_1, x_2, \dot{x}_2, \ldots, x_n, \dot{x}_n\}$, D contains a set of positions and velocities $x_1$-$x_n$ and if we want to learn a mapping a that maps f() such that the output is a time invariant dynamical system

$\dot{x} = f(x)$ then the transition to the next state only depends on the current state.

When using a Gaussian Mixture Regression this will require a Joint model over both positions and velocities

$$p(\dot{x}|x) = \textstyle\sum_n^N o_n(x)\mathcal{N}(\dot{x}|x; m^n_{\dot{x}|x}\Sigma^n_{\dot{x}|x})$$

$$\dot{x} = f(x) = \textstyle\sum_{n=1}^N o_n(x) m^n_{\dot{x}|x}$$

and condition on that. SEDS will guarantee stability when using a control theory such as Lyapunov theorem Because the output of the mapping would be a time invariant dynamical system. If a candidate lyapunov function is defined:

$V = \frac{1}{2}x^T x$ Then the constraints are that the system must be stable and it must converge.

1. $V(x_*) = 0$, stability at the steady-state
2. $V(x) > 0 \forall x \in \Re$,
3. $\frac{dV}{dt} < 0$, convergence

Using the chain rule, the time derivative of the Lyapunov function is decompose into known terms. The time derivative (dv/dt) is then written as a product of the time derivative of the lyapunov with respect to x and the derivative of x with respect to time.

For SEDS the constraints are that:

$$
\begin{aligned}
b_k &= -A_k x^* \\
A_k + (A_k)^T &\prec 0 \\
\Sigma_k &\succ 0 \\
0 < o_k &\leq 1 \\
\textstyle\sum_{k=1}^{K} o_k &= 1
\end{aligned}
$$

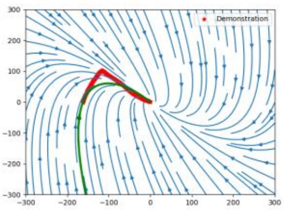
- $k$ Number of Gaussian Components
- $x^*$ Desired point of stability
- $b_k$ Bias term of each Gaussian Regression line
- $A_k = \Sigma^k_{\dot{\xi}\xi}(\Sigma^k_{\xi})^{-1}$ Slope term of each Gaussian Regression line
- $o_k$ mixture coefficient

The Bias term must pass through the point of stability. The slope must be below 0, meaning matrix A must be negative definite, if both of these constraints are met then the stability of the system at the desired point and convergence is guaranteed. The last three constrains are derived from the theory of Gaussian Distributions

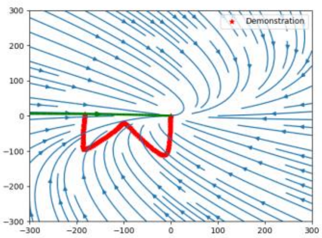
The next constraint is that the covariance matrix of the Gaussian mixture has to be positive definite and the mixture co-efficient have to be between 0 and 1.

Lastly the sum must be equal to 1.

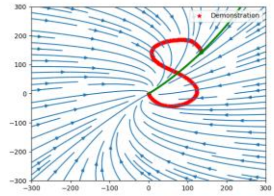
*A. Examples of demonstrations where the linear time-invariant dynamic system performs well and when it does not*



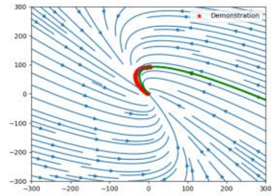

The Time invariant system almost performed well here, the arrows follow the line almost close enough to form the angle but even the angle might be too complex of a demonstration for the time invariant system



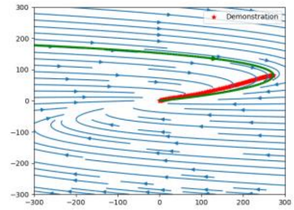

The time invariant system does not perform as well on the W shape due to the complexity of the demonstration




The time invariant system performs poorly on the S demonstration because it is too complex to follow



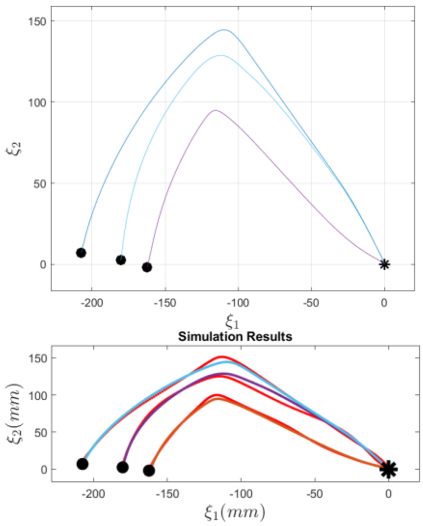

The time invariant system was able to model the demonstration accurately here as the c demonstration

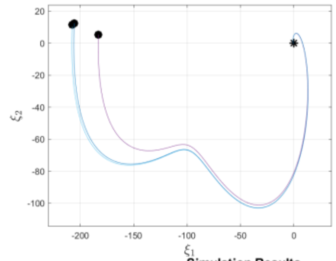


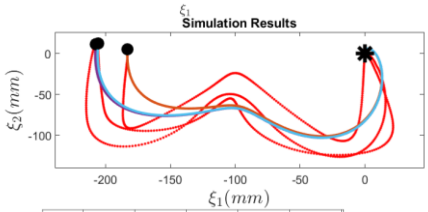

was not too complicated.

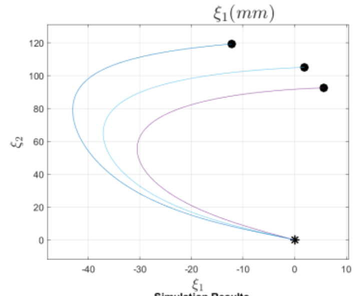
The time invariant system was able to follow the demonstration of the line because it was a simple demonstration

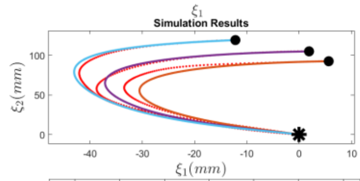
*B. Examples of demonstrations where SEDS performs well and when it does not*

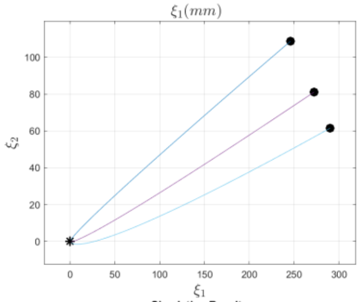


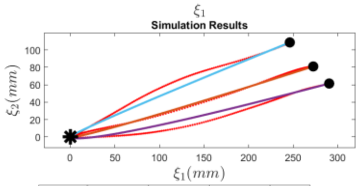


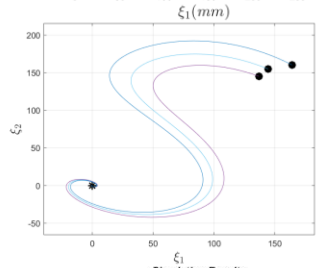


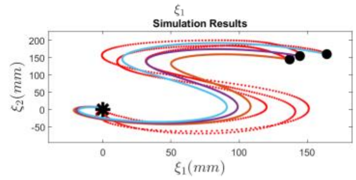










The demonstrations are not optimally modelled because it would be impossible to model all of them accurately with the same number of Gaussians for all of them due to the complexities and differences in each shape.

The C shape is not largely affected by a change in different numbers of Gaussian components however the angle is, 0 seems to be the optimal value for the Angle while any value works for the C shape.

Increasing the number of Gaussian models too greatly will cause the code to exit before it reaches convergence as the max number of iterations will be exceeded.

For the Line and C-Shape demonstrations I would use a linear model for the Line and SEDS for the C-Shape because the line is simple enough to work efficiently with 1 Gaussian whereas the C-Shape models are more complex and therefore SEDS more accurately illustrates the shapes.

SEDS performs better on more complex shapes such as W and S and Angle in contrast to the time –invariant method which could not follow the demonstrations. The C and Line however were passable on both SEDS and the Time invariant system.

## *Acknowledgment*

I would like to acknowledge The Lecturers for their explanation of HRI principles and how they could be applied.

## References

[1] Gavin, H.P., 2018. Linear Time-Invariant Dynamical Systems. *Duke University: Durham, NC, USA*.

[2] Hallauer Jr, W.L., 2016. Introduction to Linear, Time-Invariant, Dynamic Systems for Students of Engineering. Virginia Tech.

[3] Kung, M.C. and Womack, B., 1984. Discrete time adaptive control of linear dynamic systems with a two-segment piecewise-linear asymmetric nonlinearity. *IEEE transactions on automatic control*, *29*(2), pp.170-172.

[4] Huang, S., Yuzhe, L.I. and Junfeng, W.U., 2021. Distributed state estimation for linear time-invariant dynamical systems: A review of theories and algorithms. *Chinese Journal of Aeronautics*.

[5] Spanos, D.P., Olfati-Saber, R. and Murray, R.M., 2005, April. Approximate distributed Kalman filtering in sensor networks with quantifiable performance. In *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005.* (pp. 133-139). Ieee.

[6] Druskin, V. and Zaslavsky, M., 2012. On convergence of Krylov subspace approximations of time-invariant self-adjoint dynamical systems. *Linear algebra and its applications*, *436*(10), pp.3883-3903.

[7]