Full Lecture Script: Koopman Control via Nystr"om Kernel Approximation

Muthana Alsaadi

Part I. Introduction (~10 minutes)

Slide 1 – Title

Good morning, everyone. My name is Muthana Alsaadi, and today I'm presenting on a fascinating topic: **Koopman Control via Nystr"om Kernel Approximation.**

Now, I know the title sounds intimidating. But don't worry — my goal is to make this talk understandable for everyone, whether or not you've taken advanced control theory.

We'll use real-life examples — like folding T-shirts, driving cars, or running surveys — to make sense of the math. By the end, you should have a clear picture of why this work matters.

Slide 2 – Why Can't a Robot Fold a T-Shirt?

Let's start with a simple question: why can robots build cars with perfect precision, but struggle to fold laundry?

- Robots can assemble engines, weld, and paint cars.
- But ask the same robot to fold a T-shirt, and it's hopeless.

Why? Because car parts are **rigid** and predictable. T-shirts are **floppy, nonlinear, chaotic**.

In engineering language:

- Predictable things are **linear systems**.
- Chaotic things are **nonlinear systems**.

Example:

- Linear: roll a ball down a ramp $\rightarrow$ always predictable.

- Nonlinear: drop a piece of cloth in the air → a tiny change makes it flutter unpredictably.

That's the problem we're trying to solve: how can we control things that behave like cloth?

(Pause and maybe ask the audience: "What's another everyday nonlinear system you can think of?" — e.g., weather, human body, traffic.)

Slide 3 – The Breakthrough Solution

To solve this, researchers combined three ideas:

1. **Koopman Operator** – a trick that makes nonlinear systems look linear.
2. **Kernel Methods** – flexible tools to represent complex patterns.
3. **Nystr"om Approximation** – a shortcut to make the whole process computable.

And the key result: this method is not only efficient but comes with **theoretical guarantees** — it gets better predictably as you add more data.

Part II. Foundations ($\tilde{1}$5 minutes)

Slide 4 – Systems Notation

In control theory, we describe systems mathematically:

$$x_{t+1} = f(x_t, u_t)$$

This just means: the next state depends on the current state plus the input.

Example: A car

- Current speed = 40 km/h.
- Input = press the gas pedal.
- Next speed = maybe 42 km/h.

Sometimes we write it as:

$$f(x, u) = g(x) + Bu$$

which means:

- $g(x)$: messy, natural behavior of the system (like friction, slopes, wind).

- Bu: neat effect of our control input (gas pedal, brake).

We often bundle them into one variable $w = [x, u]$.

Slide 5 – Koopman Operator (The Magic Trick)

The Koopman operator is like putting on magic glasses.

- Normally: system looks nonlinear and messy.
- With Koopman: we transform it into a new view where it looks linear.

Think of it like turning a messy squiggle into a straight line when you look at it from the right angle.

Mathematically, we use **observables** — special functions of the state. In that lifted space, the system behaves linearly.

Slide 6 – RKHS (The Infinite Stage)

These observables live in a Reproducing Kernel Hilbert Space (RKHS).

You can think of RKHS as:

- An infinite cheat sheet with every possible feature of the system.
- It's universal — it can approximate anything.

Analogy: imagine a dictionary that contains every possible sentence. That's RKHS.

Problem: it's infinite. And computers don't do infinity. So we need a shortcut.

Slide 7 – LQR (The Engineer's GPS)

Once we have a linear system, we can use **LQR — Linear Quadratic Regulator**.

LQR balances two things:

1. Keep the system close to the goal.
2. Don't waste too much control effort.

Analogy: Driving. You want to reach your destination quickly, but you also want to save fuel. LQR finds the balance.

The tool behind LQR is the Riccati equation, which gives us the optimal control rule.

Slide 8 – Quiz 1 (Audience Check)

Ask the audience:

1. What does the Koopman operator do?
   → It turns nonlinear systems into linear ones.

2. What's special about the control input?
   → It usually enters the system in a neat, linear way.

(Pause for answers.)

Part III. The Shortcut ($\tilde{1}$5 minutes)

Slide 9 – RKHS Challenge

RKHS is infinite; we need a finite approximation.

Slide 10 – Nystr"om Approximation

Nystr"om picks m representative landmarks. Analogy: survey 1000 people instead of every citizen.

Slide 11 – Summary Diagram

Pipeline: nonlinear data → kernel lift → Nystr"om approximation → linear model.

Slide 12 – Two LQR Problems

- Ideal LQR: infinite, perfect, intractable.
- Approximated LQR: finite, feasible.

Analogy: god-like driver vs. GPS.

Slide 13 – Final Control Input

Control law from simplified model applied to real system. Analogy: practice in a simulator before driving real car.

Slide 14 – Full Pipeline

1. Sample landmarks.
2. Build compressed model.
3. Solve LQR.
4. Apply to real system.

Slide 15 – Quiz 2

- What does Nystr"om solve? $\rightarrow$ Infinite to finite.
- What's the matrix size? $m \times m$.

Part IV. The Guarantees ($\tilde{1}$5 minutes)

Slide 16 – Assumptions

Need bounded kernel, stabilizability, detectability, Riccati condition.

Slide 17 – Transition Operator Accuracy

Error $\|G_\gamma - \tilde{G}_\gamma\|$ converges at rate $1/\sqrt{m}$.

Slide 18 – Riccati Operator Convergence

Error $\|P - \tilde{P}\|$ also converges at $1/\sqrt{m}$.

Slide 19 – LQR Objective Convergence

Cost error $J - \hat{J}$ converges at faster rate $1/m$. Analogy: practice improves skill slowly, performance improves quickly.

Slide 20 – Quiz 3

- Quadruple m: cost error drops by 4.
- Adjustment: rescale Q and R.

Part V. Applications and Future ($\tilde{1}$0 minutes)

Slides 21–25 – Experiments

Tested on Duffing oscillator and cloth manipulation. Nystr"om was robust vs. splines.

Slide 26 – Recap

- Koopman + kernels + Nystr"om = powerful.
- Computationally efficient.
- Theoretical guarantees.
- Real-world promise.

Slide 27 – Future Frontiers

Applications: manufacturing, surgical robotics, creative fields.

Slide 28 – Open Questions

Issues: regularization, extending theory, spline instability.

Slide 29 – Final Quiz

- Convergence rates: operator $1/\sqrt{m}$, cost $1/m$.
- Why splines unstable? Centers outside training data.

Slide 30 – Closing

Thank you for joining. Key message: smart math and shortcuts make controlling chaos possible — even folding a T-shirt.