**Slide Notes**:

Slide 1:

* My project is on a Nonlinear Dynamical System used to imitate human movements in an autonomous Humanoid robot

Slide 2:

* The paper was written by Jan Ijspeert, Jun Nakanishi and Stefan Schaal
* They created a system to imitate human movements in an autonomous humanoid robot
* What that means is this system doesn’t just imitate a human movement but it learns the movements and therefore can adjust to avoid perturbations and changes in the environment as well as be able to reuse this motion later for different tasks
* They only deal with movements of the arm in their paper
* The system works as a control policy which sends messages to the controller of the robot telling it where to move
* To accomplish their goals they decided to represent movement in kinematic coordinates. This means movement is described as the changing of joint angles
* Therefore, their dynamical system represents kinematic movement plans, which means moving as a function of state. So the path the robot takes is
* The system doesn’t tell the robot how to move but instead it gives the robot’s controller the correct state to be in to reach a desired position, then the controller converts that into motor commands.

Slide 3:

* This system is our control policy
* It is essentially nonlinear except for the velocity term v, which will depend on a second dynamical system
* In z-dot we have 2 constants alpha-z and beta-z which are both greater than 0.
* G is our goal state, the state we want to get to while y is our desired state which will be outputted to the robots controller
* y-dot is our desired velocity
* I’m only going to dive deeper into this dynamic equation but first I want to give you an overview of all the math and systems in the paper

Slide 4:

* This is our second system. This system describes the internal state of the robot
* It is essential to determining the velocity used in our first system
* Constants and
* X is used to localize the Gaussian Kernel in the first system
* V is a scaling term used to ensure the second term in y-dot remains transient. This means v starts and returns to zero at the beginning and end of the discrete movement. So
* This guarantees we will have a unique attractor

Slide 5

* The Gaussian kernel is used to predict trajectories in order for machine learning
* found using locally weighted regression

Slide 6

* For stability I will only focus on the main system
* Since v is transient, as t goes to infinity v goes to 0, so our second term in y-dot will go to 0, leaving us with y-dot = z.
* This gives us one fixed point at (0,g)
* Next we find the Jacobian which is
* Then we evaluate it at (0,g), which obviously will change nothing
* Thus our eigenvalues are
* So this gives us three separate cases to consider

Slide 7

* When we get imaginary eigenvalues
* Since the constants are always positive, the alpha for our complex eigenvalue is < 0
* Thus our Phase Portrait will spiral in as we can see here.
* This phase portrait is with alpha-z and beta-z equaling 2
* and we have stability at (0,g)

Slide 8

* When we get real eigenvalues
* Lambda 1 will always be negative but we have 2 different cases for lambda 2
* The first case is when then we get lambda 1 < 0 and lambda 2 > 0
* Therefore we get this type of saddle point phase portrait
* This phase portrait was made using alpha-z = 10 and beta-z = 1
* Since lambda 1 < 0 the trajectories move into the fixed point
* that is stable

Slide 9

* The second case is when
* It’s not possible for to be less than but we can get really close to them being equal
* If they’re equal then lambda-2 will equal 0.
* This gives us an infinite line of fixed points at 0, as you can see here
* This phase portrait was created with alpha-z = 5 and beta-z = .001
* Since lambda-1 < 0 the trajectories will move towards the fixed point giving us a stable point at (0,g)

Slide 10

* In all three of our phase portraits we see that all our trajectories converge to our fixed point (0,g), so it is globally attracting
* Also since all the trajectories will stay close to the fixed point we also have Liapunov stability, so we call the fixed point asymptotically stable.
* Since (0,g) is the goal state the robot wants to get to, it being an attractor like this means on paper, that no matter what state the robot starts in it will end in the goal state
* So the dynamic system they set up works very well
* They created this dynamic system because they are doing work with stroke patients
  + This can help them in rehabilitation exercises by demonstrating the movements and a monitoring system to track their movements