EECS 367 & ROB 320 Lab KinEval Pose and Audio

Michigan EECS 367 Introduction to Autonomous Robotics | ROB 320 Robot Operating Systems

Administrative

- Assignment #3: Forward Kinematics
 - Due Friday, February 18, 11:59pm
- New quiz policy
 - Late quizzes will be accepted (including quiz 1-3)
 - Late policy follows that of projects
 - Up to 80% within 2 weeks
 - 60% within 4 weeks
 - 50% by end of grading period

New Robot Definition

- Feature for assignment 3
 - Everyone will submit as a pair of students
 - You can choose pairs or come to interactive session
- Next Wednesday, February 16th
 - Dedicated course time for paired programming
- Following Wednesday, February 23rd
 - Every team will showcase their robot during interactive session (with pizza!)
- 1 point for working forward kinematics
- 1 point for showcasing their robot

Let's introduce ourselves

- Go around the room and introduce yourself
 - Name, program
 - An aspect or topic within robotics that excites you
 - An interesting fact about yourself

Lab Takeaways

- 1. KinEval overview
- 2. KinEval walkthrough
- 3. Adding music for your dance
- \rightarrow How to start Assignment 4

Dance Controller Overview



- Assignment 4: Dance Controller
- Quaternion joint rotation
- Interactive base control
- Pose setpoint controller
- Dance FSM

KinEval Overview

📮 auto	orob / kineval-stencil	
<> Cod	de (!) Issues 1 1 13 Pull requests	➢ Actions ^{□□} Projects ^{□□}
	양 master → 양 1 branch ⓒ 0 tags	
-	ohseejay Merge pull request #3 from c	xt98/master ····
	js	initial commit Fall 2018
	kineval	initial commit Fall 2018
	project_pathplan	Adds refactored stencil files for project 1.
	project_pendularm	add refactor of assignment2, tested with
	robots	initial commit Fall 2018
	tutorial_heapsort	initial commit Fall 2018
	tutorial_js	initial commit Fran 2018
	worlds	initial commit Fall 2018
	LICENSE	add refactor of assignment2, tested with
	README.md	initial commit Fall 2018
	home.html	initial commit Fall 2018



KinEval Overview

Ē

📮 autorob	/ kineval-stencil		⊙ Watch ▾ 5 ☆ Star 9
<> Code	(!) Issues 1 1; Pull requests (.) Actions	III Projects 🖸 Wiki 🕐 Security 🗠 Insights	
	ᢞ master - kineval-stencil / kineval /		Go to file Add file -
	zhezhou1993 Factorize kineval stencil for FK	problems, fix bugs in previous version	70d8e4b 9 days ago 🕚 History
	🗅 kineval.js	initial commit Fall 2018	2 years ago
	kineval_collision.js	initial commit Fall 2018	2 years ago
	kineval_controls.js	initial commit Fall 2018	2 years ago
	kineval_forward_kinematics.js	initial commit Fall 2018	2 years ago
	kineval_inverse_kinematics.js	initial commit Fall 2018	2 years ago
	kineval_matrix.js	Factorize Kineval Stensil for FK problems, fix bugs in previous version	9 days ago
	kineval_quaternion.js	Factorize kineval stencil for Frequences, fix bugs in previou	
	kineval_robot_init.js	Factorize kineval stench for FN gradining	All code tor
	kineval_robot_init_joints.js	Factorize kineval stencil for FK gradining	assignment 4
	kineval_rosbridge.js	initial commit Eau 2018	z yearo ago
	kineval_rrt_connect.js	initial commit Fall 2018	2 years ago
	kineval_servo_control.js	initial commit Fall 2018	2 years ago
	kineval_startingpoint.js	initial commit Fall 2018	2 years ago
	kineval_threejs.js	initial commit Fall 2018	2 years ago
	kineval_userinput.js	initial commit Fall 2018	2 years ago

kineval forward kinematics.js Revisited

kine	val_forward_kinematics.js
18	
19	<pre>kineval.robotForwardKinematics = fu</pre>
20	
21	<pre>if (typeof kineval.buildFKTrans</pre>
22	textbar.innerHTML = "forwar
23	return;
24	}
25	
26	<pre>// STENCIL: implement kineval.b</pre>
27	
28	}
29	
30	<pre>// STENCIL: reference code alte</pre>
31	<pre>// links and joints starting</pre>
32	// traverseFKBase
33	// traverseFKLink
34	// traverseFKJoint
35	//

unction robotForwardKinematics () {

sforms === 'undefined') { rd kinematics not implemented";

buildFKTransforms();

For each joint, incorporate .axis and .angle within forward kinematics. You will then be able to control joints!

ernates recursive traversal over from base, using following functions:



kineval_quaternion.js

kineval_quaternion.js

QUATERNION TRANSFORM ROUTINES ///// STENCIL: reference quaternion code has the following functions: quaternion_from_axisangle quaternion_normalize quaternion_to_rotation_matrix quaternion_multiply 10 // **** Function stencils are provided below, please uncomment and implement them ****// 11 12 // kineval.quaternionFromAxisAngle = function quaternion_from_axisangle(axis,angle) {

Joint frame without control



joint.angle (dynamic w/ control)

joint.axis / (constant)



Define quaternion helper functions \rightarrow Create a joint's rotation matrix from any axis-angle pair







kineval controls.js

kineval controls.js kineval.applyControls = function robot_apply_controls(curRobot) { // includes update of camera position based on base movement // update robot configuration from controls for (x in curRobot.joints) { // update joint angles if ((typeof curRobot.joints[x].type !== 'undefined') || (typeof curRobot.joints[x].type !== 'fixed')) { if (isNaN(curRobot.joints[x].control)) curRobot.joints[x].angle += curRobot.joints[x].control; } // STENCIL: enforce joint limits for prismatic and revolute joints // clear controls back to zero for next timestep curRobot.joints[x].control = 0;





kineval_servo_control.js

kineval_servo_control.js

19	kineval.setpointDanceSequence = function exect
20	
21	<pre>// if update not requested, exit routine</pre>
22	<pre>if (!kineval.params.update_pd_dance) retuin </pre>
23	
24	<pre>// STENCIL: implement FSM to cycle through</pre>
25	}
26	
27	<pre>kineval.setpointClockMovement = function exect</pre>
28	
29	<pre>// if update not requested, exit routine</pre>
30	<pre>if (!kineval.params.update_pd_clock) retuin </pre>
31	
32	<pre>var curdate = new Date();</pre>
33	<pre>for (x in robot.joints) {</pre>
34	<pre>kineval.params.setpoint_target[x] = c</pre>
35	}
36	}
37	
38	
39	<pre>kineval.robotArmControllerSetpoint = function</pre>
40	
41	<pre>// if update not requested, exit routine</pre>
42	<pre>if ((!kineval.params.update_pd)&&(!kineval</pre>
43	
44	<pre>kineval.params.update_pd = false; // if up</pre>
45	
46	<pre>// STENCIL: implement P servo controller of</pre>
47	}



kineval_servo_control.js

kineval_servo_control.js

19	<pre>kineval.setpointDanceSequence = function exec</pre>
20	
21	<pre>// if update not requested, exit routine</pre>
22	<pre>if (!kineval.params.update_pd_dance) retu</pre>
23	
24	<pre>// STENCIL: implement FSM to cycle throug</pre>
25	}
26	
27	<pre>kineval.setpointClockMovement = function exec</pre>
28	

Thought experiment: 1. Why are we only asking for a P controller? What would control look like with a PID controller? 3. What about a PD controller?

30	
39	<pre>kineval.robotArmControllerSetpoint = function</pre>
40	
41	<pre>// if update not requested, exit routine</pre>
42	<pre>if ((!kineval.params.update_pd)&&(!kineva</pre>
43	
44	<pre>kineval.params.update_pd = false; // if u</pre>
45	
46	<pre>// STENCIL: implement P servo controller</pre>
47	}







home.html

home.html

131	///////////////////////////////////////
132	///// MAIN FUNCTION CALLS
133	///////////////////////////////////////
134	
135	// start KinEval execution once the pa
136	//window.onload = kineval.start;
137	<pre>document.body.onload = kineval.start;</pre>
138	
139	// STUDENT: my_animate is where your
140	<pre>function my_init() {</pre>
141	
142	<pre>kineval.startingPlaceholderInit()</pre>
143	Initialize kine
144	<pre>} kineval.params.da</pre>

Poses for servo can be set and stored **interactively** in KinEval using [0-9] keys and Shift+[0-9]



Create a cool dance routine by defining a sequence of joint angle setpoints to be used by the FSM implementation

age and its resources are loaded

robot's controls and movement are updated over time

; // a quick and dirty JavaScript tutorial val.setpoints and ance_sequence_index here

JSON.stringify(kineval.setpoints) will output the currently available servo setpoints to the console as a string





HTML5 Audio

dance routine!

Uses the audio element offered by HTML5 We can load a song in home.html Then our FSM can play/pause the song along with the dance

home.html

// STENCIL: my_animate is where your robot's controls and movement are updated over time 123 124 function my_init() { 125 126 // Adding music for the dance FSM 127 // The song I have chosen is 'Wave' by Antonio Carlos Jobim 128 // My dance waves, but does not necessarily coincide with the beat of the song 129 song = document.createElement("audio"); 130 song.src = "music/Wave.mp3" 131 132 startingPlaceholderInit(); // a quick and dirty JavaScript tutorial 133 134

With two small additions to the stencil code, you can add music for your







