# (ECE183DA/MAE162D)

Prof. Ankur Mehta, Prof. Jacob Rosen
mehtank@ucla.edu, jacobrosen@ucla.edu

**Joint Lab Assignment 2**
**Due 3pm Friday, Feb. 5, 2021**

## 1 Lab Overview

### 1.1 Objectives and Goals

In the previous lab, you derived the kinematics of a differential drive robot moving in a 2D plane, you created a CAD assembly of a Segway with an added castor wheel, and you created a simulation of said robot moving in a 2D plane. In this lab we will verify that the kinematics models derived matches between the CAD and computational models to give us confidence that they will work in reality. This is a four step process:

- Decide what experiments must be run in order to verify our models
- Configure our tools (CAD and code) to run these experiments
- Run the experiments and obtain the resulting data
- Do appropriate statistical analysis on the resulting data to verify our models.

For simplicity in Lab 2, we are not considering the sensor model. The sensors will be addressed in future labs.

You will be working in your project teams, and will be responsible as a team for dividing the various tasks of this project between all members. Your grade will be based both on team and individual performance.

### 1.2 General Aims

Your design and its mathematical formulation have to be verified before proceeding into prototyping or developing algorithms. Thus, we conduct quantitative analysis between computational simulations based on mathematical equations and SolidWorks numerical simulations of two differential drive robots to justify your results.

### 1.3 Specific Aims

Through this lab,

- Both MAE and ECE students are to demonstrate understanding of the underlining design procedures.
- Both MAE and ECE students are to demonstrate understanding of statistical analysis on the results to verify our models.
- ECE (and optionally ME) students are to demonstrate understanding of how to conduct coded motion simulations to collect meaningful data with appropriate setups.
- MAE (and optionally ECE) students are to demonstrate understanding of how to conduct motion analysis simulations in SolidWorks to collect meaningful data with appropriate setups.
- MAE (and optionally ECE) students are to learn the basics of several of the analyses available in SolidWorks that are useful in guiding the design of mechanical parts.
- Both MAE and ECE students will work together to explain and correct any discrepancies between the results of the coded and CAD simulations.

### 1.4 Deliverables/Method of Reporting

For this joint lab, your team is required to submit:

- A write-up describing your process of verification on computational and SolidWorks simulation, and their results and data with statistical analysis by 3pm Friday, Feb. 5, 2021.

ECE students will also create a well documented git repository for your simulator containing all your code and data. MAE students will add their simulation files and data in the same repo. Include in your write-up links to your code repository / documentation, as well as a complete list of references you've used and in what manner.

ME students will also individually complete the "Lab Exercise 2" SolidWorks simulations tutorials, and submit the required PDF detailed within. This exercise is anticipated to be finished in 2-3 hours during lab, and the submission is due 3pm Friday, Jan. 29, 2021.

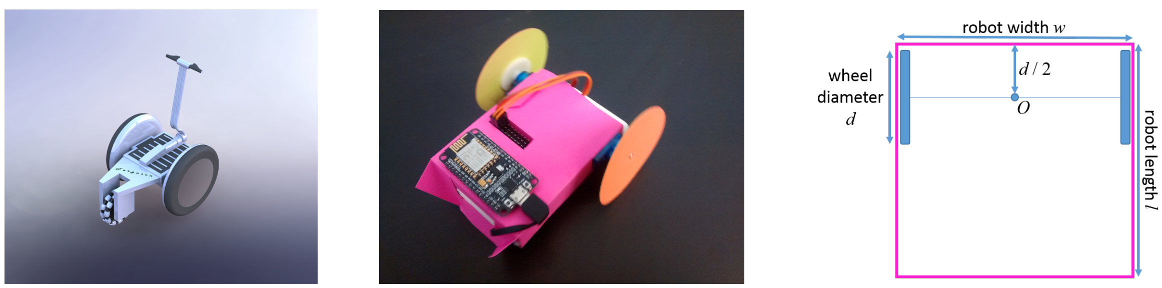


Figure 1: Two wheeled tank-drive robots with symbolic dimensions.

For all deliverables, you will be assessed on both the clarity and completeness of your content. Submit pdfs on CCLE. Submissions that are late will be accepted with a 50% grade penalty.

# 2 Joint Assignment Descriptions

## 2.1 System overview

Both MAE and ECE students modeled the same two-wheeled robots in Joint Lab 1, similar to the Segway and Paperbot shown in Fig. 1. These robots have two wheels of diameter $d \approx 502mm/50mm$ (Segway/Paperbot), separated by a distance $w \approx 530mm/90mm$. Each wheel is direct driven from a continuous rotation servo. They drag a tail (or castor wheel in the case of the Segway) for stability, that contact the ground at a distance $l \approx 682mm/75mm$ behind their front edge. The position of each of these robots in the environment is defined relative to their centerpoints $O$.

We do not consider sensors in this lab, but we verify our sensor models in the next lab. More details about Paperbot is available at the git repository: `https://git.uclalemur.com/mehtank/paperbot`

## 2.2 Actuation model

Each wheel is powered independently by a continuous rotation servo —part number FS90R for the Paperbot, an unspecified industrial continuous rotation servo for the Segway— with the angular velocity of the wheel controlled by a PWM signal from the microcontroller. The control input to the robot hardware will be the PWM values you send to each wheel, for a total of 2 input variables. This allows the robot to drive forwards or backwards at variable speed, or turn with any turning radius.

## 2.3 Robot Input Sets and Simulation Conditions

Your team decides sets of two inputs to verify your mathematical formulation and code with numerical simulation based on CAD models. What sets of inputs do you need to verify them? What initial conditions should we have/try? How long/how many time steps do you need to run simulations? How many times do you need to test the same set of inputs in order to analyze and compare your results of your analytical and numerical simulations?

# 3 Section Specific Tasks

## 3.1 ECE Students Only

### 3.1.1 Coded Simulation: Simulation Setups and Settings

Using your analytical simulation frameworks developed in Joint Lab 1, setup an open environment with the ability to initially place your robots at any specified initial states. Adjust your simulation settings to follow what your team decides to test in Section 2.3. Be sure to use the same time steps and the data you create match with those of SolidWorks. Do appropriate modifications to your codes or interfaces if necessary to run simulations with the same setups as ones in SolidWorks.

Consider what noises you should include in your system or environment, and apply them in your simulation if necessary. The simulation conditions must be identical or as close as possible between your code and SolidWorks to appropriately compare the results.

#### 3.1.2 Coded Simulation: Data Collection

Using your analytical simulator setup in Section 3.1.1, Run your defined sets of inputs to drive both Segway and Paperbot. Set up the initial conditions of your simulation appropriately and the conditions should match with SolidWorks simulation setups. Be sure to experiment with diverse sets of inputs so that we can obtain varieties of trajectories as you have defined in Section 2.3. We are only focusing on system dynamics and we will verify your sensor model in the next lab.

### 3.2 MAE Students Only

#### 3.2.1 Provided Files:

- Floor: This is a 1 x 1 x 0.1 meter rectangular prism rubber floor we will use for the PaperBot simulations. Note that the checkered side is the top surface.
- BigFloor: This is a 10 x 10 x 0.1 meter rectangular prism rubber floor we will use for Segway simulations. Note that the checkered side is the top surface.
- PaperBot: An assembly of a differential drive robot as described in Joint Lab Assignment 1. Composed of PaperBase and PaperWheels. Unfortunately, paper is so light it causes errors in the Motion Analysis, so the material used for the parts is wood (oak to be specific).
- FinalSegway: A slightly modified version of the Segway created in the previous lab. Mates were redone to reduce redundancies. Wheels are reassigned to be alloy steel (and their "Override Mass Properties" was disabled). A point aligned at the midpoint between the two wheels is added to serve as the robot origin point. Lastly, indicators have been added onto the wheels to let us observe them spinning more clearly.

#### 3.2.2 SolidWorks Simulation Tutorials:

MAE students will individually complete a number of Solidworks Simulations tutorials to familiarize them with some of the tools available in SolidWorks. Details are provided in the "Lab Exercise 2" for the MAE Lab.

#### 3.2.3 PaperBot Motion Analysis:

Each MAE student should have *at least one* trajectory of meaningful complexity for the PaperBot robot to compare against coded simulation results.

##### 3.2.3.1 Setting Up the Simulation:

- Create a new assembly. Browse for the "Floor" part, but do not click the graphics area to insert it. Instead, click the green OK check mark, which will automatically insert the part in the assembly such that the part and assembly origins are coincident. Save your assembly as PaperBotSimulation.
- Assembly → Insert Components. Select "PaperBot", and click in the graphics area to insert. Right click PaperBot in the FeatureManager Design Tree and select "Make Subassembly Flexible"
- Insert mates between the 3 surfaces shown in Fig. 2 and the top of the Floor. Each wheel should be mated with a Tangent mate, and the tail should use a Coincident mate. This ensures your robot is in contact with the Floor when the simulation starts, preventing bouncing and impacts.

Figure 2: Surfaces to mate to Floor.

- Insert a distance mate between the origin point (the only visible sketch point) of the PaperBot and the side wall of the Floor that runs in the Y-Z plane of the assembly. This will be your initial X position, and you can set it to any arbitrary value you like that keeps the robot within the Floor. Similarly, assign a starting

Y position with another distance mate but this one should be between the origin point and the side wall of the floor that runs in the X-Z plane. Lastly, assign an arbitrary angle mate between the right side wall of the PaperBase and the side wall of the floor that runs in the X-Z plane to get an initial $\theta$.

- In the FeatureManager Design Tree, open the drop-down arrow for Mates, select all the mates you placed in the previous step, right click, and select suppress. We do this because, while the mates let us specify are starting conditions, they would prevent us from moving in the actual simulation. If you need to move your parts at any point during the procedure, you can unsuppress and then resuppress your mates to restore part position/orientation. Sometimes these mates cause a "Not-a-number (NAN)" error to be thrown by the integrator when running the analysis. If this occurs, delete these mates.
- Go to Tools (at the top of your screen) → Add-Ins → Select "SOLIDWORKS Motion" → press OK.
- At the bottom of your window, click the tab labeled "Motion Study 1" If no such tab exists, right click the blank space to the right of the "Motion" and "3D Views" tab and select "Create New Motion Study". In the tab, select the drop-down labeled "Animation" and select "Motion Analysis". Save your assembly.
- There are a number of icons along the same bar to the right of the drop-down you just used. Most of the commands we will use are there. Hover your cursor over each just to familiarize which named icons are where.
- Add contacts between the Floor and each of the parts of the PaperBot. You can add the contacts by going to "Contact" and clicking the two parts to simulate contact between. You can see your selection if you drop-down the "Selections" option. SolidWorks has built in materials that it can simulate friction and impacts with using proper mechanical properties, which is an easy way to do things if your parts are those materials. Unfortunately our PaperBot is wood, which is not one of the options so uncheck the "Material" option. For each wheel-floor contact, set $\mu_k = 0.3$, $\mu_s = 0.5$, select "Restitution coefficient" for elastic properties and enter 0.05 for its value. For the base-floor contact, do the same for the Elastic Properties, but uncheck the Friction option. Save your progress.
- At this point, you may notice a few yellow bars pop up on the timeline showing the duration for which each of your effects are active. By default, the simulation goes for 5 seconds. You can change the duration by right clicking the black diamond on the right edge of the bars, selecting "Edit Key Point Time" and entering a new value. If your computer is less powerful, you may need to reduce this noticeably for your simulation to solve in a reasonable amount of time, but your analysis must be long enough to enable testing of a sufficiently complex trajectory. There is also a vertical bar across the timeline that indicates the start time of any new effects you have. All of our effects will start at $t = 0s$ so make sure that bar stays to the left. A way to help prevent moving it is to uncheck the "Autokey" option.
- In the toolbar you were asked to familiarize yourself with previously, click "Gravity". By default, it selects the Z direction of the assembly, which should be correct for you. If it is not, your "Floor" was inserted wrong, go back to the first step. More generally, if you wanted to simulate gravity in other directions (or with other magnitudes) you can select the direction and input magnitude as you like though. Click the green OK check to insert gravity.
- If desired, you can move your camera throughout your animation. To do so, click on the black diamond next to "Orientation and Camera Views", and drag it to a point in time where you want to specify camera position. Using your normal camera navigation options, go to the camera orientation you would like at that point in time. Then, click the newly created black diamond and select "Replace Key". You can similarly change the camera orientation of your existing key frames. Prebuilt orientations are available by right clicking the black diamond and choosing an option from "View Orientation".

#### 3.2.3.2 Specifying Motor Inputs

Here you will specify the motor commands to the wheels. Do note that if only one wheel has the command specified, the other wheel will roll freely as physics would require it to. For each of the two wheels do the following:

- Select the "Motor" command. Make sure the "Motor Type" is Rotary Motor. For both "Motor Location" and "Motor Direction" select the circular face of the wheel (the one with the red marks). For "Component to Move Relative to" select the PaperBase. Select the "Reverse Direction" toggle if needed to cause the direction of rotation (shown by the curved red arrow) to match what you define in your coded simulations.
- Under "Motion" click on the drop-down labeled "Constant Speed" and switch it to "Data Points". This will cause a window to pop up. For "Value(y)" select velocity, for "Independent variable (x)" select Time (s), and for "Interpolation type" choose Linear (the others are also valid if you prefer). Click where it says "Click to add row" and add as many values at time points as you feel is needed to simulate a sufficiently complex trajectory to verify your coded simulation against. You should simulate for 5 seconds, and your value at time=0s should be 0.00deg/s. While specifying values, keep in mind that both negative and positive velocities are valid and that there is a maximum angular velocity of each wheel specified in Joint Lab Assignment 1. Once you are

done, you can check the “Make function availble for use elsewhere in this document” and give the function a name if you like, and press OK.

#### 3.2.3.3 Running the Motion Analysis

The trickiest part of running a Motion Analysis is specifying the numerical simulation parameters. If accuracy is too low, your system will have large errors. If accuracy is too high, then your analysis will take a very long time. There are three different algorithms used as integrators, with pros and cons each. There is also the ability to set initial, minimum, and maximum integration step size. If the minimum step size is too large, the simulation won't converge and will exit in an error. If maximum step size is too large, your analysis may glitch out (e.g. launching vertically off the floor at escape velocity).

After approximately 15 hours of bug testing, the settings that work for me for the PaperBot are given below. These settings can be reached by going to the “Settings” icon on the toolbar you familiarized yourself with, and in the “Advanced Options” of the resulting menu. There is unfortunately little documentation explaining what each option does, so I can't promise they will work for you, but they are a starting point.

- Frames per second: 25
- Animate during simulation: Checked
- Replace redundant mates with bushings: unchecked
- 3D Contact Resolution: 100. This will slow the analysis at the start as it rebuilds, but drastically speeds up analysis after. You can use Precise Contact instead, which will increase accuracy but can be orders of magnitude slower.
- Accuracy: 0.0001. This can be increased or decreased. Improving accuracy will slow down simulation and can cause your integrator to not converge. Decreasing accuracy will noticeably negatively effect the results.
- Cycle Rate: 100
- Integrator Type: WSTIFF. The other two integrators can work as well, but may result in different accuracy or slower analysis. GSTIFF will be more inaccurate if there are discontinuities in the simulation. SI2_GSTIFF will be slower but can be more accurate in some cases.
- Maximum Iterations: 25. This is how many times the simulator will attempt to find a solution of the given accuracy before it gives up.
- Initial Integrator Step Size: 0.00001000000
- Minimum Step Size: 0.0000001000
- Maximum Step Size: 0.0000100000 If this value is too large your model may skyrocket/bounce off the floor randomly or fall through the floor. If either of these occur, decrease your Maximum Step Size. Decreasing Maximum Step Size limits max speed of solving the analysis though.
- Jacobian Re-evaluation: Always

Once you have changed the settings, save everything. Then, click “Calculate”. If you are not using Precise Contact, your program might freeze up for a minute while it is rebuilding, this is normal. Observe the analysis unfold for a few minutes to check to make sure unexpected behavior is not occurring (e.g. movement in the Z direction causing loss of contact of the wheels). If there is an issue, click the “Stop” icon and try to troubleshoot. If everything looks good, do something to occupy yourself for a while, and check back later. Once your simulation is done, make sure to save. If you want to run more analysis, I recommend right clicking the Motion Study tab, clicking “Copy Study”, and doing the additional analysis in the new study tab.

As a side note, do *NOT* click the “Save Animation” command unless you have already saved everything immediately prior. The command is supposed to export a video of your results, but it immediately crashes my SolidWorks every time I have pressed it.

#### 3.2.3.4 Obtaining Results

Fortunately, the easiest part of the entire motion analysis is obtaining and exporting results. SolidWorks has a number of built in results it can measure, and all of them are automatically calculated during the analysis even if you don't ask for them until afterwards. To add an output, click the “Results and Plots” icon on the toolbar you familiarized yourself with. When creating plotted results, I recommend using the “Add to existing plot” option for all results after the first in order to make exporting the results easier later. Some of the results you may find useful to add include:

- “Displacement/Velocity/Acceleration” → “Trace Path” → Select the origin point of the robot. This will visually create a line tracing out the path of your robot origin.
- “Displacement/Velocity/Acceleration” → “Linear Displacement” → Select the origin point of the robot and the Y-Z plane or X-Z plane of the Floor. This will allow you to plot the X and Y coordinates of your robot

origin over time.

- "Displacement/Velocity/Acceleration" → "Angular Displacement" → "Magnitude" → select the X-Z plane of the Floor and either plane of the PaperBase that run parallel to the wheels (the left or right face of the base). This will plot your $\theta$.
- "Displacement/Velocity/Acceleration" → "Angular Velocity" → "Z Component" → select the X-Z plane of the Floor and either plane of the PaperBase that run parallel to the wheels (the left or right face of the base). This will plot your $d\theta$.

In addition to the above results for your states, you should also record the actual inputs given to your system, namely by adding:

- 'Displacement/Velocity/Acceleration" → "Angular Velocity" → "X Component" → select the circular face of your left wheel and any plane of the PaperBase. Then, for "Component to define XYZ directions (optional)" select the PaperBase. This will plot the angular velocity of the motor for the left wheel. Do the same thing for the right wheel.

To export your data, right click the plot you want to export as shown in Fig. 3, and select "Export to Spreadsheet". This should automatically create and open an Excel Spreadsheet that contains a table of your values at each time step and a plot of the values against time.

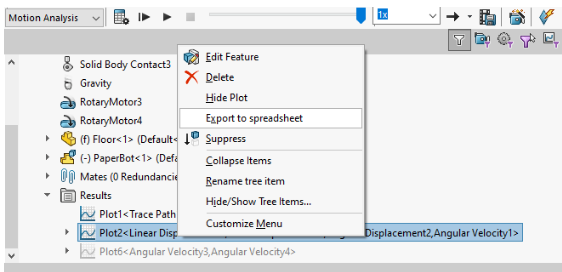


Figure 3: Right click your plot to export, and select "Export to Spreadsheet"

Note that the time steps are not constant. This will need to be accounted for in the coded simulation such that your SolidWorks simulation results can be compared to the coded simulation results for the same inputs.

### 3.2.4 Segway Motion Analysis

Using the provided FinalSegway assembly, along with the BigFloor, set up an complete a motion analysis on the Segway robot. Each MAE student should have *at least one* trajectory of meaningful complexity for the Segway robot to compare against coded simulation results.

The Segway motion analysis can be set up very similar to the PaperBot. One advantage, however, is that the materials of the Segway are already included in the Contact options of the motion analysis. Namely, you can treat the three contacts as rubber on rubber. In this way, you do not need to manually specify friction or restitution coefficients in the Contacts (although you should disable friction for the castor wheel). Also of note, to remove mate redundancies in the assembly (which slow down motion analysis), the castor wheel does not spin. Your contact between it and the BigFloor should be on the surface that touches the BigFloor.

Precise Contact can be used for this analysis, and it will run quickly. Maximum Step Size can also be set larger (e.g. 0.0100000000). And be aware of giving motor imputs that cause your robot to slip/drift.

# 4 Simulation Data Analysis and Comparison (Both MAE and ECE)

## 4.1 Statistical Analysis between Analytical and Numerical Simulation

Using the data obtained from coded and SolidWorks simulations, verify if your analytical simulations match the result of SolidWorks numerical simulations. How similar are the robot trajectories of the two simulations? What is the range of errors? Are there any notable deviations in errors among different trajectories or initial conditions? Why do you think the results match well or not based on your mathematical formulation from Joint Lab 1? Can you think of any changes in your simulation setups that may improve errors between analytical and numerical results (i.e. time steps, SolidWorks simulation settings)? Conduct additional simulations to verify your ideas.

## 4.2 Statistical Analysis between Segway and Paperbot Simulation

Based on Section 4.1 analysis, are there any significant differences in errors between Paperbot and Segway? Are there any specific situations/trajectories/initial conditions that makes one of our robot's analytical solutions diverge from SolidWorks numerical solutions, but not the other robot? Why do you think the results of the two robots match well or not based on your mathematical formulation from Joint Lab 1? If their errors are similar for all of your setups and inputs, can you think of any situations where the two robots' results would diverge? Conduct additional simulations to verify your ideas if necessary.

## 4.3 System Errors and Potential Simulation Scenario

Have you included any noises in your analytical simulations? Explain in a few sentences why you do or don't. Come up with situations that may significantly diverge the results between analytical and numerical simulations? Come up with noises/disturbances/situations that your mathematical formulation cannot describe or capture well/at all. What modifications do you need to do to capture those situations in your analytical simulation appropriately? Explain in one paragraph. How do you define and include noises/disturbances/situations in analytical and numerical simulations? (i.e. wind is an external force) For this section, you don't necessary to show additional simulations. You may do that in the future labs.

## 4.4 Design Process

Through this Joint Lab 2, how confident are you to start prototyping your own CAD models or developing algorithms based on your mathematical formulations. How do or do not simulations help deciding how to proceed with your designs? Briefly explain your thoughts.