Montreal, August 26th, 2024

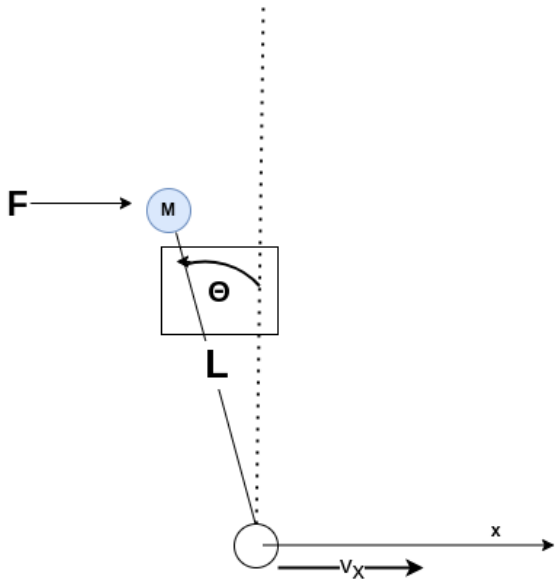
Vention Take Home Exercise: Firmware

Introduction

The following design challenge is meant to be an open-ended exercise. The candidate is free to determine the appropriate level of depth and completeness necessary in the final submission. If some aspects of the questions or content of the document are unclear, feel free to contact Vention for clarifications (francois.giguere@vention.cc)

The Design Challenge

Vention has signed a promising yet demanding client that will require your expertise. This client specializes in small-scale wind turbines that are deployed in remote locations. They have developed a curious (somewhat unrealistic) design that uses a Vention linear rail to keep it upright (see below). The client also insists on having an HMI that includes visualization of the status of the turbine that resides onsite. Your task is to develop them a system that can satisfy these constraints:



The turbine can be modeled as an inverted pendulum, defined entirely by its mass and height above the rail. We can control the rail in velocity (vx), measure the angle of the pendulum (Θ) and position of the rail (x) at 100Hz. The client requires a feedback controller to keep their unit vertical, as well as having it eventually settle in position to 0mm after being subject to a disturbance force (F) at the point mass (M). The mass of the turbine (M) is in Kg, and the height of the turbine (L) is in m.

The client requires:

- a headless simulation for this system to show its behavior as a response to random disturbance F and input vx
- A feedback controller that will keep the turbine upright
- A simple visualization tool (HMI) to show the state of the turbine.

Once you've done exactly what the client has asked for and deployed this solution, of course, they have requested a change of scope. It turns out that the rail we provided them has a finite length and they would like to be alerted in the event that x is within 100 mm of either end on the HMI. They also request a small change to the parameters of the controller you developed (i.e.gains). Since you anticipated that something like this would happen, you put in place some infrastructure to update the controller and the HMI without a failure of your controller.

Expected Deliverables

Please develop the original software solution proposed, with infrastructure in place to update the system without downtime of the controller. Then develop the requested update (for some sort of alert at boundaries of the rail and updated controller gains) and demonstrate that you can upgrade the unit while keeping the system stable. Feel free to assume the programming/hardware environment of your choice, and use what you feel appropriate for controller/HMI programming languages and frameworks.