

ESCOLA POLITÉCNICA
UNIVERSIDADE DE SÃO PAULO



SISTEMA DE TRANSPORTE
AUTOESTABILIZADO SOBRE DUAS RODAS

Alunos:

Henrique Golhiardi Malachias
João Eduardo dos Santos
Vítor Cristiano Teixeira Fernandes

Orientador:

Prof. Dr. André
Riyuti Hirakawa

26 de novembro de 2015

Resumo

Este trabalho visa desenvolver um protótipo funcional de um sistema autoestabilizado de transporte sobre duas rodas. Tal sistema já existe no mercado e é comercializado pela empresa Segway. Desenvolvemos um modelo mecânico não-linear com alto nível de rigor físico e matemático e, utilizando técnicas de controle linear, um sistema de controle em malha fechada que visa permitir a pilotagem funcional do dispositivo por uma pessoa adulta. Em paralelo, documenta-se as etapas de construção física do protótipo, passando pela montagem mecânica e elétrica até a implementação do algoritmo de controle digital. As especificações desejadas para tal protótipo são a manutenção do equilíbrio em diversas situações adversas dentro da normalidade; o atingimento, com segurança, de uma velocidade de aproximadamente 18km/h a critério do motorista; o transporte de uma pessoa de até cerca de 90kg; uma autonomia de bateria de até aproximadamente meia hora e, não menos importante, o incentivo à pesquisa de novos paradigmas na tão falada mobilidade urbana e na criação de veículos alternativos para os cenários urbanos da atualidade, mais eficientes, compactos, seguros e ecológicos.

Sumário

1	Conceitos iniciais	3
1.1	O projeto	3
1.2	O método de Lagrange	4
1.3	Métodos de Controle	5
1.3.1	Controlador PID	5
1.3.2	Realimentação de estados	6
1.3.3	Controle ótimo	7
1.4	Comportamento de sensores reais	8
1.4.1	Acelerômetro	9
1.4.2	Giroscópio	9
1.4.3	Unindo os sensores	10
2	Modelagem matemática e simulações	12
2.1	Modelo não linear	12
2.2	Linearização do modelo	16
2.3	Teste em malha aberta dos modelos	17
3	Projeto do Controlador	19
3.1	Estratégia de Controle	19
3.2	Tipo de Controlador	19
3.2.1	Especificações	19
3.2.2	Discretização do modelo e matrizes de custos	20
3.3	Simulações do sistema em malha fechada	21
4	Software	24
4.1	Simuladores	25
5	Desenvolvimento e implementação prática	27
5.1	Protótipo 1	27
5.1.1	Plataforma de sustentação	27
5.2	Rodas de tração	30
5.2.1	Motores elétricos	30
5.2.2	Sistema de redução da velocidade por engrenagens	31
5.2.3	Baterias	32
5.2.4	Eixos do sistema	32
5.2.5	Placa controladora	33
5.2.6	Drivers de acionamento dos motores	36
5.2.7	Placa de comunicação	36
5.2.8	Sistemas auxiliares	41
5.2.9	Disposição dos sistemas elétricos	42
5.2.10	Configuração final do sistema	42
5.2.11	Testes realizados e problemas encontrados	42
5.3	Protótipo 2	46
5.3.1	Disposição dos novos motores	46

5.3.2	Configuração do sistema de tração	47
5.3.3	Confecção do guidão do sistema	48
5.3.4	Configuração mecânica final do projeto	48
5.3.5	Testes iniciais e elaboração de um template para o Arduino	50
5.4	Protótipo em escala	54
6	Resultados	58
6.1	Implementação do Controlador	58
6.2	Testes iniciais de funcionamento	58
6.3	Dados dos sensores	58
6.4	Adequando o controlador	59
6.5	Resultados na miniatura	60
7	Discussões e Conclusões	61
7.1	Cumprimento dos objetivos	61
7.2	Desempenho	61
7.3	Desenvolvimento futuro	61

1 Conceitos iniciais

Esta seção tem o objetivo de explicar brevemente o que se deseja obter com este projeto e a teoria por trás do desenvolvimento matemático deste trabalho. Não se pretende desenvolver, obviamente, um tratado completo e definitivo sobre o assunto, mas, sim, algumas breves disquisições sobre teoria mecânica, elétrica e de engenharia de controle para que o leitor que não tenha experiência na área possa entender de onde vem as técnicas aqui usadas e, se assim desejar, pesquisar mais a fundo em fontes mais completas e tradicionais.

Primeiramente, convém saber que existem diversas maneiras de se chegar às equações de movimento de um sistema mecânico. Pode-se, por exemplo, desenhar o diagrama de corpo livre com as forças e torques que ali atuam. Daí, pode-se chegar às equações usando-se das leis de Newton e resolvendo o sistema de equações para a aceleração, com o objetivo de obter-se equações diferenciais que descrevem aquele sistema. Embora bastante didática e interessante, essa metodologia requer um amplo conhecimento do sistema mecânico e costuma gerar um equacionamento bastante intenso e extenso, que costuma ser de resolução bastante cansativa e passível de erros. Seu uso é bastante difundido no ensino médio e cursos pré-vestibulares, por ser a base da mecânica clássica e por não exigir, em geral, o uso de ferramentas de matemática superior como o cálculo integral e diferencial para os casos ali expostos.

A alternativa que iremos aqui descrever é a análise das energias do sistema para chegar nas equações de movimento. Esse método, conhecido como método de Lagrange, posto que usa as famosas equações de Lagrange, dá a quem o usa a clara vantagem de trabalhar com grandezas escalares (energia) e não com grandezas vetoriais (forças), o que obviamente conduz a um equacionamento mais simples, embora produza um resultado equivalente. A desvantagem que pode existir, é que esse método faz uso de integrais que podem não ser fáceis de se resolver ou podem até mesmo não ter um resultado analítico. Isso, porém, não acontece na nossa aplicação e dificilmente acontece no geral dada a forma como são escritas as equações de movimento da maioria dos sistemas.

1.1 O projeto

Acredita-se que o leitor esteja familiarizado com a palavra Segway, embora seu uso seja relativamente moderno, conforme pode ser visto na figura 1. Trata-se de um neologismo baseado na palavra inglesa *segue*, que quer dizer "continuar sem pausar", como quando uma orquestra passa de um movimento a outro de uma sinfonia sem parar.

A empresa que leva o nome de Segway teve seu início em 1999 quando Dean Kamen quis desenvolver um veículo de alta eficiência, sem emissão de carbono com base na estabilização dinâmica. Posteriormente, em 2001, a empresa adotou e patenteou o nome Segway. Desde então, a empresa vem desenvolvendo melhorias e novos veículos, todos baseados na ideia de estabilização dinâmica.

Nosso escopo para este projeto é o de criar um dispositivo semelhante aos desenvolvidos por esta empresa para estudar como a teoria de controle e de

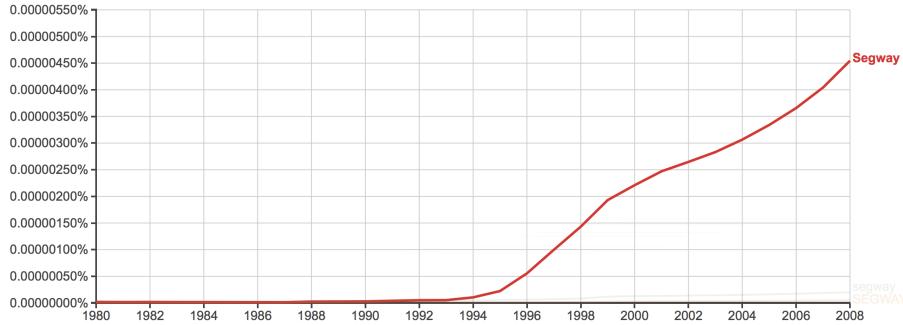


Figura 1: O uso da palavra Segway com o passar dos anos. Fonte: Google Books

algoritmos pode ser utilizada com esse fim. Com esse protótipo, será possível estudar o mercado, a fim de decidir quanto à possibilidade de produção em larga escala de um produto equivalente ou semelhante, porém mais barato que o Segway.

1.2 O método de Lagrange

Para usar-se o método de Lagrange, deve-se seguir algumas simples etapas:

1. Determina-se um conjunto de coordenadas independentes que descrevam o movimento do sistema de maneira conveniente. Estas são as chamadas coordenadas generalizadas e o número de coordenadas generalizadas é o número de graus de liberdade do sistema.
2. Equacionar as expressões das energias no sistema e aplicar a função de Lagrange:

$$L = T - U \quad (1)$$

onde T é a energia cinética e U, a energia potencial do sistema.

3. Resolver as equações de Lagrange:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = F_{q_i} \quad (2)$$

onde q_i é a i-ésima coordenada generalizada, e F_{q_i} é a i-ésima força generalizada. Para cada coordenada generalizada, a força generalizada é a soma das projeções das forças que atuam no sistema sobre o eixo de q_i .

Com isso, obtém-se as equações que regem o movimento do sistema. Convém novamente ressaltar que, tanto o método de Lagrange, quanto o método de diagrama de corpo livre levam a um mesmo resultado, pois são, felizmente, duas teorias consistentes entre si.

1.3 Métodos de Controle

Convém adiantar que o modelo que usaremos possui múltiplas entradas e saídas e é altamente não linear, acoplado e tudo aquilo que não se aprende em controle linear. Assim sendo, deve-se imaginar que as técnicas mais simples que estão no repertório de um aluno de engenharia não serão suficientes neste caso. Será necessária a utilização de um método mais avançado da teoria de controle multivariável. Há diversas técnicas possíveis, dentre as quais enumeraremos as mais conhecidas e usadas para esse fim.

1.3.1 Controlador PID

O famoso PID (Proporcional Integral Diferencial) é amplamente utilizado na indústria e nas mais diversas aplicações passadas e atuais. Como a maior parte dos controladores, ele busca minimizar o erro entre o sinal de referência e a saída do sistema e poderia ser usado para controlar o dispositivo em questão, porém com o detalhe de ser necessária sua aplicação em várias variáveis de controle.

Embora existam diversas metodologias de sintonia de PID, como o método de Ziegler-Nichols, o PID tem a vantagem de ter uma sintonia bastante intuitiva, dado que:

- Quando a parte proporcional (P) aumenta, o controlador fica mais rápido e agressivo, as margens de estabilidade diminuem e a habilidade de manejar perturbações de carga aumenta, porém, em contra partida, há um aumento no esforço de controle, o que produz um maior gasto de energia, aquecimento, eventual desgaste mecânico e o perigo de saturar os atuadores.
- A parte integral (I) fornece ganhos elevados para baixas frequências e, portanto, compensa eventuais erros de regime ou perturbações de baixa frequência em geral. Em geral, é importante para retirar o erro de regime.
- A parte derivativa (D) aumenta as margens de estabilidade, com a desvantagem de um aumento no ganho de alta frequência. Isso pode levar a problemas com relação ao ruído, que pode promover ações de controle indesejadas, como vibrações no eixo do motor que levam a um desgaste prematuro e causam desconforto ao operador. É, portanto, conveniente implementar a parte derivativa associada a um filtro passa-baixas.

Assim sendo, um operador experiente pode sintonizar o controlador de uma maneira heurística, buscando manualmente uma combinação de ganhos convenientes.

O controlador PID costuma ser implementado com algumas outras funcionalidades para torná-lo mais realável na prática. Um fenômeno que pode acontecer é o *windup*, uma dinâmica que se manifesta quando ocorre saturação dos atuadores. O integrador começa a somar um erro que não é corrigido e, quando o atuador sai do estado de saturação, o integrador volta a saturá-lo,

tentando corrigir o erro acumulado. Isso deve ser previsto e corrigido no projeto do controlador.

A equação do controlador PID no domínio do tempo é:

$$u(t) = K \left(e(t) + \frac{1}{T_i} \int_0^t e(s) \, ds + T_d \frac{de(t)}{dt} \right) \quad (3)$$

No domínio da frequência, temos:

$$U(s) = K \left(E(s) + \frac{1}{sT_i} E(s) + T_d s E(s) \right) \quad (4)$$

1.3.2 Realimentação de estados

A alocação de polos por realimentação de estados é uma maneira de controlar o comportamento de sistemas de malha fechada através da alocação de polos de malha fechada em posições onde se atinja o comportamento desejado do sistema.

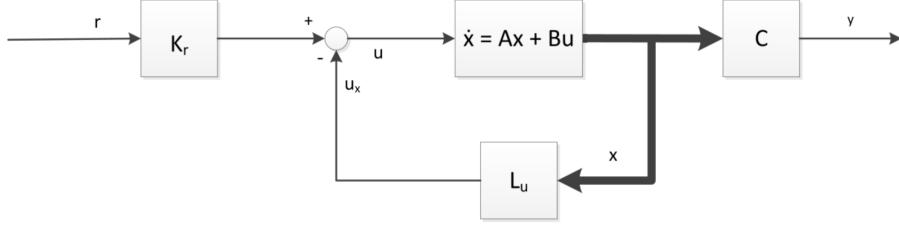


Figura 2: Controlador por realimentação de estados

Os estados e as saídas de um sistema linear invariante no tempo podem ser descritos pelo seguinte sistema de equações:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \quad (5)$$

Se assumirmos que todos os estados são conhecidos ou observáveis, obtemos a seguinte lei de controle:

$$u(t) = -L_u x(t) + K_r r(t) \quad (6)$$

onde L_u é o vetor usado para alocação de polos e o ganho K_r é calculado tal que o ganho para baixas frequências seja unitário do sinal de referência r para a saída y . A função de transferência do sinal de referência para a saída é dada por:

$$G_{ry}(s) = C(sI - A)^{-1} B K_r \quad (7)$$

Fazendo $G_{ry}(0) = 1$, obtém-se a seguinte expressão para K_r :

$$K_r = \frac{1}{C(-A)^{-1}B} \quad (8)$$

1.3.3 Controle ótimo

Uma desvantagem que há no controle por realimentação de estados é que sua sintonia não é elementar. Costuma ser difícil escolher adequadamente entre as distintas opções de posicionamento de polos, ainda mais quando temos sistemas de complexidade elevada, como este do qual estamos tratando. Fica difícil decidir uma alocação de polos convenientes quando o projetista se vê fazendo escolhas que melhoraram algo mas pioraram outros pontos. O controlador linear quadrático (LQ) é implementado como um controlador de realimentação de estados com um vetor de ganhos de realimentação L que é determinado através da minimização de J , que é dado por:

$$J = \int_0^{\infty} (x^T(t)Q_1x(t) + 2x^T(t)Q_{12}u(t) + u^T(t)Q_2u(t)) dt \quad (9)$$

O controlador dependerá das matrizes Q_1 e Q_2 , que determinam como serão penalizados o esforço de controle e o erro de controle. Visto que isso não costuma ser dado explicitamente, tais matrizes são consideradas como parâmetros de projeto. Elas devem ser simétricas e a matriz Q_1 precisa ser semi-definida ($Q_1 \geq 0$) e Q_2 precisa ser positiva definida ($Q_2 > 0$). A lei de realimentação que minimiza a função de custo J é dada por:

$$u(t) = -Lx(t) \quad (10)$$

onde L é dado por:

$$L = Q_2^{-1}B^T S(t) \quad (11)$$

e $S(t)$ é dado pela seguinte equação de Riccati em tempo contínuo:

$$A^T S(t) + S(t)A - S(t)BQ_2^{-1}B^T S(t) + Q_1 = 0 \quad (12)$$

Este controlador pode ser combinado com um filtro de Kalman para estimar estados desconhecidos e para reduzir o impacto do ruído no sinal de referência e nas variáveis medidas. Quando é feita esta combinação, pode-se chamar o controlador de LQG, onde o G refere-se a Gaussiano. A matriz R é a matriz de covariância e descreve o ruído que afeta os estados (R_1), o ruído afetando as medições (R_2) e como eles podem estar correlacionados (R_{12}).

$$R = \begin{bmatrix} R_1 & R_{12} \\ R_{12}^T & R_2 \end{bmatrix} \quad (13)$$

O problema de determinar o ganho de Kalman (K) é bastante similar ao cálculo do vetor L no caso anterior de controlador LQ e podem até ser considerados como problemas duais. O ganho K que minimiza a variância do erro para quando se tem ruído branco e Gaussiano é:

$$K = (PC^T + NR_{12}) R_2^{-1} \quad (14)$$

onde P é dado pela seguinte equação de Riccati:

$$AP + PA^T + NR_1N^T - (PC^T + NR_{12}) R_2^{-1} (PC + NR_{12})^T = 0 \quad (15)$$

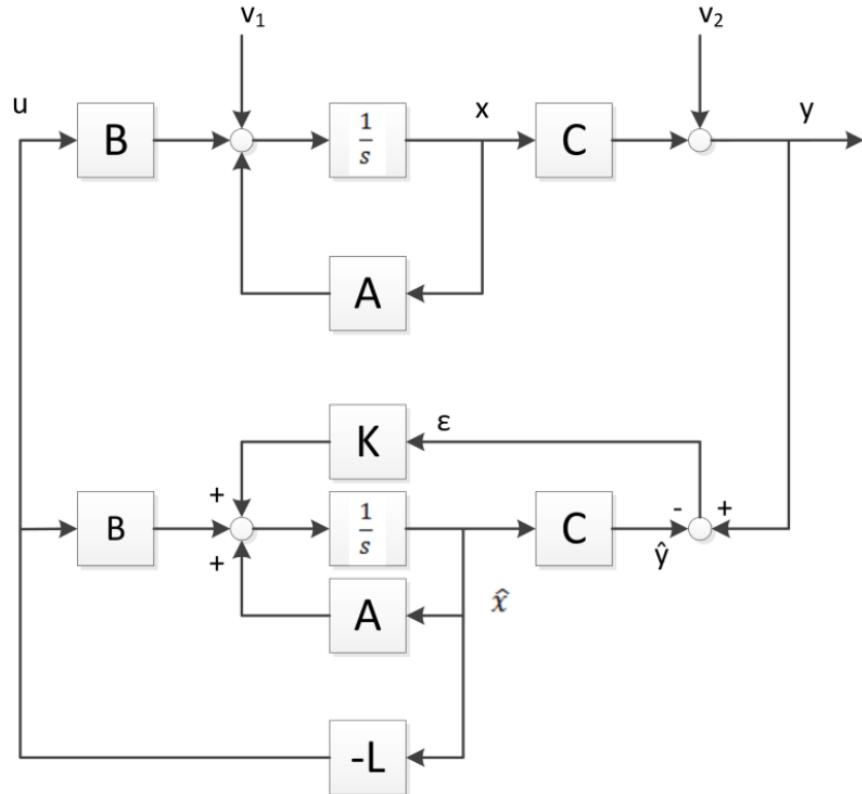


Figura 3: Controlador LQG

1.4 Comportamento de sensores reais

Em se tratando de implementação de sensores em sistemas reais, deve-se entender que não é comum ter-se o comportamento ideal apresentado na teoria. Obviamente, existem uma série de fatores que não são considerados nos livros básicos de engenharia. Os sinais precisam ser medidos pelos sensores e, nem

sempre podem ser medidos diretamente, mas sim, calculados usando outros sinal. Nisso, entram problemas de ruído, discretização, resolução e interferências. Faz-se, portanto, necessário o uso de filtros passa-baixa que produzem uma diminuição nas margens de estabilidade.

O objetivo desta seção é o de explorar os sinal relevantes à aplicação que aqui propomos, como estes podem ser medidos e quais distorções podem ocorrer nestas medições.

1.4.1 Acelerômetro

Um acelerômetro, como o próprio nome diz, é um dispositivo capaz de medir acelerações. Sua utilização cresceu bastante nas últimas décadas e hoje, com o desenvolvimento ocorrido, é um dispositivo bastante compacto, barato e de fácil aquisição.

Acelerômetros modernos baseiam-se em uma estrutura de mola, cuja deflexão causada por forças externas pode ser mensurada por um sensor capacitivo e convertida em sinal elétrico.

Um problema que se manifesta em aplicações como esta é que o acelerômetro, quando sujeito a acelerações que ocorrem naturalmente com o deslocamento do veículo, tanto linear quanto rotacional, quando o sensor está fora do eixo de rotação do veículo.

Pode-se resumir os problemas do acelerômetro como problemas de alta frequência, como ruído e perturbações de elevada frequência.

1.4.2 Giroscópio

Um giroscópio pode ser usado para medir a velocidade angular de um corpo com relação a um referencial inercial. Aqueles que visam um baixo custo e tamanho reduzido costumam ter um desenho mecânico bastante semelhante aos acelerômetros descritos acima. O sistema mecânico pode variar, mas o tipo mais comum utiliza duas massas vibrantes para medir a aceleração de Coriolis e, com isso, calcular a velocidade angular.

A saída do giroscópio pode também ser integrada para obter-se o ângulo do corpo com relação ao ponto de partida. Essa estimativa é menos afetada pela aceleração linear do que a fornecida pelo acelerômetro, porém sofrerá de um desvio decorrente da integração numérica. Além disso, tais sensores também possuem um certo viés, ou seja, a saída não será nula quando o dispositivo estiver completamente parado. Esse viés é dependente de temperatura, vibrações, tensão de alimentação, entre outros, e é bastante difícil de ser previsto ou removido.

Por outro lado, pode-se resumir os problemas do giroscópio como desvios de baixa frequência. Essa dualidade entre esses dois sensores já permite ao leitor imaginar que sua união inteligente pode ser interessante.

1.4.3 Unindo os sensores

Conhecendo os pontos fortes e fracos de cada um desses sensores acima apresentados, podemos uni-los sabiamente para obter uma estimativa do ângulo mais próxima da realidade e mais propícia para realizar um controle eficaz. Sabemos que o acelerômetro fornece uma estimativa absoluta do ângulo de inclinação quando este se encontra em um referencial inercial, porém este produz resultados errôneos quando submetido a acelerações. O giroscópio, por sua vez, pode ser usado para estimar o ângulo por integração numérica da sua saída, sem prejuízo ao estar submetido a acelerações. Sua desvantagem reside no fato de que essa estimativa incremental produz desvios crescentes graças a erros numéricos no tempo.

Um filtro que costuma ser usado em projetos simples é o chamado filtro complementar. Este filtro requer pouco poder de processamento e é relativamente fácil de implementar. Este filtro usa, basicamente, o giroscópio para estimar o ângulo no curto prazo e corrige essas estimativas baseado na referência absoluta do acelerômetro. Uma forma de representar este filtro é como segue:

$$\theta = 0.98 * (\theta + gyro * dt) + 0.02 * (acc) \quad (16)$$

onde θ é o ângulo, $gyro$ é a saída do giroscópio e acc a saída do acelerômetro.

Pode-se dizer que ele aplica um filtro passa-baixa no acelerômetro e um filtro passa-alta no giroscópio. Ajustando-se as constantes, zera-se, portanto, o erro de desvio do giroscópio. Porém este tipo de filtragem tem seus problemas. Um deles é que ele não permite obter o *offset* do giroscópio, sendo então impossível usar sistemas de controle que usem a velocidade angular como entrada, tal como o próprio PID ou variações deste.

Outra implementação usual é o filtro de Kalman, que requer mais processamento e esforço. O filtro de Kalman vem da teoria de otimização e é o estimador ótimo quando as perturbações do sistema podem ser aproximadas como ruído branco normalmente distribuído. Este filtro resolve o problema que tinha o último no que se refere ao bias.

Uma outra possibilidade seria a de tentar estimar e remover tanto quanto for possível do *offset* do giroscópio antes de aplicar os filtros acima descritos, através de um estudo empírico do giroscópio usado no projeto. Ressaltamos, todavia, a dificuldade deste procedimento, dado que esse *offset* é provocado por uma série de fatores nada lineares, variáveis no tempo e de difícil previsibilidade.

Talvez a melhor alternativa, porém, seja utilizar características internas de circuitos integrados com sensores conjugados, como o utilizado neste projeto, que já oferecem algoritmos digitais de processamento de sinais embutidos com alta precisão, rapidez e eficiência.

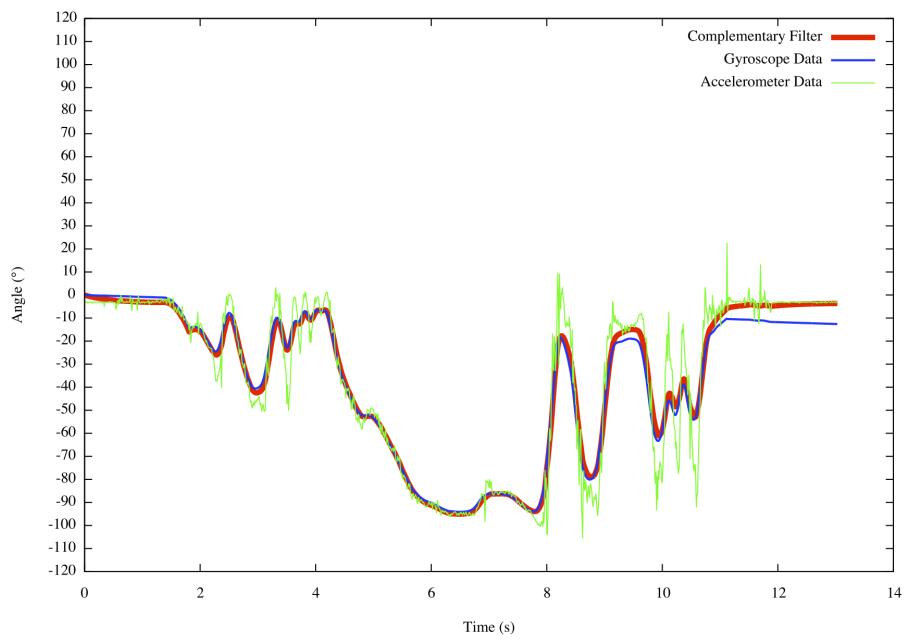


Figura 4: Efeito do filtro complementar

2 Modelagem matemática e simulações

Esta seção discorre sobre o modelo matemático da planta a ser utilizado no projeto do controlador. Escolhemos não desenvolver um modelo do zero, visto que já existe uma vasta literatura sobre o assunto. Realizamos uma pesquisa e encontramos diversos modelos, a maioria baseada no pêndulo invertido. Assim sendo, escolhemos como base o modelo mais adequado ao nosso projeto, que é o mostrado em [6], [8], [7], [10] e [13], com algumas modificações que realizamos, a fim de representar melhor o piloto do nosso dispositivo.

Primeiramente, desenvolveremos um modelo mecânico na forma de equações diferenciais utilizando o método de Lagrange. Posteriormente, este passará por uma linearização, afim de podermos utilizar as técnicas de controle linear que conhecemos, como as ensinadas em [9] e [12], e, por fim, faremos uma análise qualitativa do modelo, com o propósito de validar seu funcionamento.

A fim de se obter uma competente modelagem matemática, aqui faz-se necessário encontrar um meio termo entre complexidade e usabilidade. Para tanto, foram estudados diversos modelos utilizados em projetos semelhantes e, ao final, decidimos que era necessário realizar algumas modificações no modelo escolhido como base, visto que esse modelava uma carga bastante diferente de uma pessoa, como no caso deste projeto.

2.1 Modelo não linear

Conforme podemos ver na figura 5, vamos modelar o dispositivo como duas rodas sustentando um sólido formado por um cilindro fixo sobre um paralelepípedo.

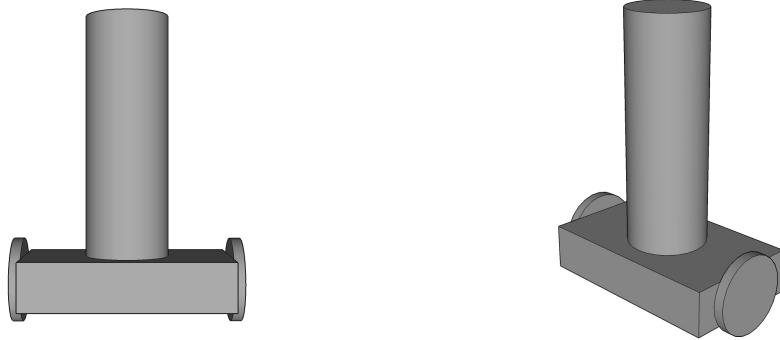


Figura 5: Vistas em 3D do modelo

Iremos utilizar o método de Lagrange, conforme descrito previamente, para chegar nas equações de movimento, então precisamos calcular alguns momentos de inércia.

Começando com as rodas, é imediato da teoria de mecânica clássica, que:

$$J_w = \frac{mR^2}{2} \quad (17)$$

Em, seguida, temos que calcular o momento de inércia do sólido anteriormente descrito com relação a dois eixos de rotação, aquele dado por ϕ (ângulo de trajetória do veículo) e o outro dado por ψ (inclinação do motorista com relação à vertical). Para ambos os casos, iremos dividir a distribuição de massa em dois conjuntos distintos, a fim de usar resultados já conhecidos de momentos de inércia de sólidos comuns na engenharia.

Primeiramente, em relação a ϕ , temos:

- Base:

$$J_{\phi,b} = \frac{m_b (w_b^2 + d_b^2)}{12} \quad (18)$$

- Cilindro:

$$J_{\phi,c} = \frac{m_c r_c^2}{2} \quad (19)$$

Somando as equações 18 e 19, obtemos o momento de inércia total em relação a ϕ :

$$J_\phi = J_{\phi,b} + J_{\phi,c} \quad (20)$$

Dando continuidade, repetimos o procedimento para ψ :

- Base:

$$J_{\psi,b} = \frac{m_b (w_b^2 + h_b^2)}{12} + m_b \left(\frac{h_b}{2} \right)^2 \quad (21)$$

- Cilindro:

$$J_{\psi,c} = \frac{m_c h_c^2}{3} + m_c h_b^2 \quad (22)$$

Somando as equações 21 e 22, obtemos o momento de inércia total em relação a ψ :

$$J_\psi = J_{\psi,b} + J_{\psi,c} \quad (23)$$

Feito isso, podemos continuar o processo, determinando as energias cinética e potencial do sistema, a fim de calcular posteriormente o Lagrangiano.

A energia cinética pode, para fins de bom entendimento, ser dividida em translacional e rotacional:

$$\begin{aligned} K_t = & \frac{1}{2}m (\dot{x}_l^2 + \dot{y}_l^2 + \dot{z}_l^2) + \\ & + \frac{1}{2}m (\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2) + \\ & + \frac{1}{2}M (\dot{x}_{cm}^2 + \dot{y}_{cm}^2 + \dot{z}_{cm}^2) \end{aligned} \quad (24)$$

$$\begin{aligned}
K_r = & \frac{1}{2} J_w \dot{\theta}_l^2 + \frac{1}{2} J_w \dot{\theta}_r^2 + \\
& + \frac{1}{2} J_w \dot{\psi}^2 + \frac{1}{2} J_\phi \dot{\phi}^2 + \\
& + \frac{1}{2} n^2 J_m (\dot{\theta}_l - \dot{\psi})^2 + \frac{1}{2} n^2 J_m (\dot{\theta}_r - \dot{\psi})^2
\end{aligned} \tag{25}$$

Na sequência, podemos calcular a energia potencial do sistema:

$$U = mgz_l + mgz_r + Mgz_{cm} \tag{26}$$

O Lagrangeano será, portanto, expresso em função do que foi determinado acima. Serão usadas as seguintes coordenadas generalizadas:

- θ : Média dos ângulos das rodas esquerda e direita
- ψ : Ângulo de inclinação do usuário
- ϕ : Ângulo de trajetória do veículo

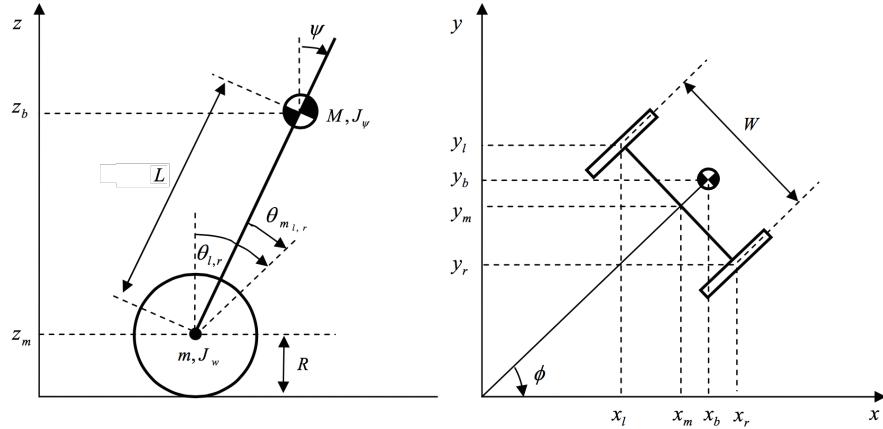


Figura 6: Vista esquemática do modelo

Para a presente modelagem, as coordenadas da figura 6 são melhor representadas por:

$$(x_m, y_m, z_m) = (R\theta \cos \phi, R\theta \sin \phi, R) \tag{27}$$

$$(\theta, \phi) = \left(\frac{1}{2} (\theta_l + \theta_r), \frac{R}{W} (\theta_r - \theta_l) \right) \tag{28}$$

$$(x_l, y_l, z_l) = \left(x_m - \frac{W}{2} \sin \phi, y_m + \frac{W}{2} \cos \phi, z_m \right) \tag{29}$$

$$(x_r, y_r, z_r) = \left(x_m + \frac{W}{2} \sin \phi, y_m - \frac{W}{2} \cos \phi, z_m \right) \tag{30}$$

$$(x_b, y_b, z_b) = (x_m + L \sin \psi \cos \phi, y_m + L \sin \psi \sin \phi, z_m + L \cos \psi) \tag{31}$$

As equações de Lagrange são:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = F_\theta \quad (32)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\psi}} \right) - \frac{\partial L}{\partial \psi} = F_\psi \quad (33)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} = F_\phi \quad (34)$$

Calculando o Lagrangiano e substituindo nas equações de Lagrange para as coordenadas generalizadas, obtemos:

$$\begin{aligned} & [(2m + M) R^2 + 2J_w + 2n^2 J_m] \ddot{\theta} + (MLR \cos \psi - 2n^2 J_m) \ddot{\psi} \\ & - MLR \dot{\psi}^2 \sin \psi - [(2m + M) R^2 \theta + MLR \sin \psi] \dot{\phi}^2 = F_\theta \end{aligned} \quad (35)$$

$$\begin{aligned} & (MLR \cos \psi - 2n^2 J_m) \ddot{\theta} + (ML^2 + J_\psi + 2n^2 J_m) \ddot{\psi} \\ & - MgL \sin \psi - (MLR \theta + ML^2 \sin \psi) \dot{\phi}^2 \cos \psi = F_\psi \end{aligned} \quad (36)$$

$$\begin{aligned} & \left[\frac{1}{2} mW^2 + J_\phi + \frac{W^2}{2R^2} (J_w + n^2 J_m) + (2m + M) R^2 \theta^2 \right. \\ & \left. + 2MLR \theta \sin \psi \right] \ddot{\phi} + 2 \left[(2m + M) R^2 \theta \dot{\theta} \right. \\ & \left. + ML^2 \dot{\psi} \sin \psi \cos \psi + MLR (\dot{\theta} \sin \psi + \theta \dot{\psi} \cos \psi) \right] \dot{\phi} = F_\phi \end{aligned} \quad (37)$$

Considerando o torque dos motores DC e o atrito viscoso destes, as forças generalizadas podem ser dadas por:

$$(F_\theta, F_\psi, F_\phi) = \left(\frac{1}{2} (F_l + F_r), F_\psi, \frac{R}{W} (F_r - F_l) \right) \quad (38)$$

$$F_l = nK_t i_l + f_m (\dot{\psi} - \dot{\theta}_l) \quad (39)$$

$$F_r = nK_t i_r + f_m (\dot{\psi} - \dot{\theta}_r) \quad (40)$$

$$F_\psi = -nK_t i_l - nK_t i_r - f_m (\dot{\psi} - \dot{\theta}_l) - f_m (\dot{\psi} - \dot{\theta}_r) \quad (41)$$

Como não é conveniente controlar a corrente, é mais interessante expressar as equações anteriores em função da tensão de armadura. O comportamento de um motor DC, se considerarmos a sua constante mecânica muito maior que a sua constante elétrica, é regido por:

$$i_{l,r} = \frac{v_{l,r} + K_b (\dot{\psi} - \dot{\theta}_{l,r})}{R_m} \quad (42)$$

Por fim, as forças generalizadas podem ser expressas, usando a tensão de armadura, como:

$$F_\theta = \frac{\alpha}{2} (v_l + v_r) - \beta (\dot{\theta} - \dot{\psi}) \quad (43)$$

$$F_\psi = -\alpha (v_l + v_r) + 2\beta (\dot{\theta} - \dot{\psi}) \quad (44)$$

$$F_\phi = \frac{R}{W} \alpha (v_r - v_l) - \beta \dot{\phi} \quad (45)$$

$$\alpha = \frac{nK_t}{R_m}, \beta = \frac{nK_t K_b}{R_m} + f_m \quad (46)$$

Com isso, conclui-se a derivação do modelo não linear. Temos em mãos um sistema MIMO (*multiple input multiple output*) com as entradas definidas por v_l e v_r e as saídas por θ , ϕ e ψ . Fica claro pelas suas equações de movimento que se trata de um sistema altamente não linear, então convém linearizá-lo para desenvolver o controlador. Vale também notar neste ponto que nem todas as saídas são fáceis de medir no sistema real, portanto, dependendo do controlador, é natural pensar no uso de observadores de estado, a fim de estimar estas saídas.

2.2 Linearização do modelo

Nosso objetivo com este projeto é conseguir estabilizar o sistema em torno do seu ponto de equilíbrio instável, i. e., $\psi = 0$. Para isso, iremos linearizar o sistema em torno deste ponto, fazendo $\psi \rightarrow 0$, o que é bastante razoável, já que o controlador deverá manter o ângulo suficientemente próximo de zero a todo momento. O resultado desta aproximação consiste em ignorar termos de segunda ordem e considerar:

$$\cos \psi \approx 1 \quad (47)$$

$$\sin \psi \approx \psi \quad (48)$$

Aplicando este resultado nas equações 35, 36 e 37 e usando o resultado das equações 43, 44 e 45, obtemos o modelo linearizado, que é dado por:

$$[(2m + M) R^2 + 2J_w + 2n^2 J_m] \ddot{\theta} + (MLR - 2n^2 J_m) \ddot{\psi} = \frac{\alpha}{2} (v_l + v_r) - \beta (\dot{\theta} - \dot{\psi}) \quad (49)$$

$$(MLR - 2n^2 J_m) \ddot{\theta} + (ML^2 + J_\psi + 2n^2 J_m) \ddot{\psi} - MgL\psi = -\alpha (v_l + v_r) + 2\beta (\dot{\theta} - \dot{\psi}) \quad (50)$$

$$\left[\frac{1}{2} mW^2 + J_\phi + \frac{W^2}{2R^2} (J_w + n^2 J_m) \right] \ddot{\phi} = \frac{R}{W} \alpha (v_r - v_l) - \beta \dot{\phi} \quad (51)$$

Como se trata de um sistema MIMO, conforme já discutido, é interessante colocá-lo na representação de espaço de estados, a fim de permitir o uso de

técnicas modernas de controle. Para tanto, vamos definir o vetor de estados e de entradas:

$$\mathbf{x} = \begin{pmatrix} \theta \\ \dot{\theta} \\ \phi \\ \dot{\phi} \\ \psi \\ \dot{\psi} \end{pmatrix} \quad (52)$$

$$\mathbf{u} = \begin{pmatrix} v_l \\ v_r \end{pmatrix} \quad (53)$$

Com isso, fica completamente definido o modelo linear em espaço de estados, obtido por manipulação das equações 49, 50 e 51:

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{(2k_2+k_3)\beta}{k_2^2-k_1k_3} & 0 & 0 & \frac{gk_2LM}{k_2^2-k_1k_3} & \frac{-(2k_2+k_3)\beta}{k_2^2-k_1k_3} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{-\beta}{k_4} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{-(2k_1+k_2)\beta}{k_2^2-k_1k_3} & 0 & 0 & \frac{-gk_1LM}{k_2^2-k_1k_3} & \frac{(2k_1+k_2)\beta}{k_2^2-k_1k_3} \end{pmatrix} \quad (54)$$

$$B = \begin{pmatrix} 0 & 0 \\ \frac{-(2k_2+k_3)\alpha}{2k_2^2-2k_1k_3} & \frac{-(2k_2+k_3)\alpha}{2k_2^2-2k_1k_3} \\ 0 & 0 \\ \frac{-R\alpha}{Wk_4} & \frac{R\alpha}{Wk_4} \\ 0 & 0 \\ \frac{(2k_1+k_2)\alpha}{2k_2^2-2k_1k_3} & \frac{(2k_1+k_2)\alpha}{2k_2^2-2k_1k_3} \end{pmatrix} \quad (55)$$

Onde:

$$k_1 = ((2m + M)R^2 + 2J_w + 2n^2J_m) \quad (56)$$

$$k_2 = (MLR - 2n^2J_m) \quad (57)$$

$$k_3 = (ML^2 + J_\phi + 2n^2J_m) \quad (58)$$

$$k_4 = \left(\frac{1}{2}mW^2 + J_\phi + \frac{W^2}{2R^2} (J_w + n^2J_m) \right) \quad (59)$$

2.3 Teste em malha aberta dos modelos

Nesta seção apresentamos algumas respostas dos modelos, a fim de mostrar que estes estão dando resultados satisfatórios para atuarmos no projeto de controladores.

Simulando a planta com uma condição inicial de $\psi = 0.1$, obtemos a resposta esperada, isto é, o pêndulo irá tender ao seu ponto de equilíbrio estável ($\psi = \pi$). Vale ressaltar que tal ponto não faz o menor sentido no mundo real, pois

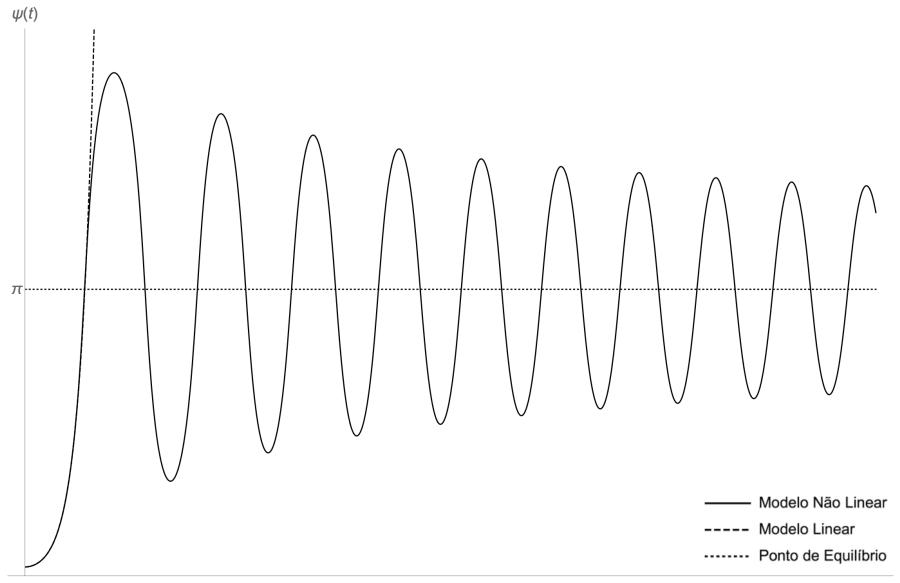


Figura 7: Respostas temporais com condição inicial $\psi = 0.1$

envolveria um piloto pendurado pelos pés enquanto balança sob o solo, abaixo do chão que sustenta o dispositivo.

Deve ficar claro, porém, que isso não invalida nosso modelo. Já deve ter ficado claro para o leitor, dado o senso comum e a linearização que realizamos na seção anterior, que nos interessa manter ψ o mais próximo de zero possível. Qualquer coisa fora disso é muito difícil de controlar e torna a pilotagem desnecessariamente perigosa, além de não ser o objetivo deste veículo.

Vemos que o modelo linearizado oferece uma resposta bastante semelhante ao não-linear em uma faixa angular bastante grande, sendo, portanto, satisfatório para o que se deseja neste projeto, conforme fica claro na figura 7. É totalmente esperado que o modelo linear desvie do não linear para ângulos grandes e isso não nos afeta negativamente de forma alguma.

3 Projeto do Controlador

Como já foi dito anteriormente, o objetivo deste projeto é o de usar um controlador linear. A base utilizada para o projeto foi, portanto, o modelo linearizado. Nesta seção trataremos sobre a estratégia de controle usada e mostraremos algumas respostas do sistemas.

3.1 Estratégia de Controle

Sabe-se que o ponto de equilíbrio do veículo quando este está de pé é instável, tendendo a cair se não for controlado. Esse é o primeiro objetivo do controlador e o segundo é o de fazer o usuário ser capaz de controlar a velocidade de translação do veículo.

A primeira técnica que vem à mente para atingir tais objetivos seria a de mudar o sinal de referência, usando o controlador para fazer a saída do sistema acompanhá-lo. Mas se queremos que o usuário controle a velocidade através de inclinações de seu corpo, torna-se complicado de implementar este comportamento. Quando o usuário inclina-se para frente ou para trás, o centro de gravidade do veículo muda de posição e o sistema começa a tombar. Seria impossível para o controlador saber se esse comportamento ocorreu por causa da vontade do usuário de acelerar ou por causa de perturbações externas que necessitariam ser corrigidas, como uma rajada de vento ou uma imperfeição no solo. É, portanto, bastante difícil projetar a estratégia de controle utilizando o ângulo de inclinação ou a velocidade rotacional das rodas como referência para fazer com que o veículo se movimente.

Para conseguir controlar a velocidade através da inclinação do corpo do usuário, usaremos uma estratégia diferente e um pouco heterodoxa, onde o objetivo é o de sempre manter o ângulo zerado. Projetando o controlador de forma a ele não ser capaz de remover erros estacionários, a inclinação do usuário pode ser vista como uma perturbação que gera um erro estacionário. O controlador continuará tentando obter um ângulo zero, girando as rodas e o erro estacionário provocará um aumento da velocidade do veículo.

É bastante interessante observar as diversas técnicas de controle sendo usadas por meio do MATLAB, analisando as respostas de cada uma, conforme é feito em [1], na Universidade de Michigan.

3.2 Tipo de Controlador

Um controlador que encaixa muito bem no que se pede é o LQR (Linear Quadratic Regulator). Este tipo de controlador será usado com realimentação negativa dos estados para levá-los a zero.

3.2.1 Especificações

Para projetar o controlador, iremos especificar o custo dos estados que desviam de zero e o custo do esforço de controle relacionado. Para isso, basta

buscar uma sintonia onde se tenha o cumprimento das especificações de projeto. Escolhemos a seguinte especificação:

O controlador deve retornar o veículo ao ângulo zero de um ponto de partida não superior a 6 graus, ponto este próximo à saturação dos atuadores.

Esta especificação foi escolhida dado que foi mais fácil encontrar as limitações do sinal de controle do que definir quanto rápido o controlador deve ser para que o manejo do veículo seja agradável ao usuário. Dessa forma, a resposta será basicamente tão rápida quanto possível, dadas as limitações impostas ao sinal de controle. As limitações do sinal de controle vêm das seguintes hipóteses: condições normais de uso incidem em ângulos não superiores a 3 graus e o sinal de controle precisa de uma folga para lidar com imperfeições no solo sem saturar. Para o escopo deste projeto, decidimos dobrar este valor de 3 graus para garantir uma operação segura do dispositivo.

3.2.2 Discretização do modelo e matrizes de custos

Discretizamos o modelo do sistema utilizando uma frequência de amostragem de 1kHz (dado que iremos trabalhar com um período de amostragem de 1ms no Arduino). Os polos do sistema discretizado confirmam, pela teoria de controle linear, que trata-se de um sistema instável:

$$P_d = \begin{pmatrix} 1.0000 \\ 0.7044 \\ 0.9977 \\ 1.0025 \\ 1.0000 \\ 0.9681 \end{pmatrix} \quad (60)$$

Tal conclusão pode ser tirada do fato de que existem polos fora do círculo unitário e pode ser encontrada em qualquer livro ou apostila de controle digital.

As matrizes de custos foram selecionadas sabendo que a bateria fornece 12V. Por tentativa e erro, sintonizamos as matrizes até obter as especificações desejadas. Com o vetor de estados x e desprezando a matriz Q_{12} , as matrizes Q_1 e Q_2 ficam:

$$Q_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 * 10^8 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (61)$$

$$Q_2 = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix} \quad (62)$$

Note que, com essa escolha, está sendo priorizada a derivada do ângulo de inclinação. Isto se mostrou bem mais eficiente no controle do ângulo, provocando movimentos mais suaves e menos sobressinal e respostas de alta frequência na saída. O vetor de ganho para um usuário de 80kg e altura 1,8m é, portanto:

$$L = \begin{bmatrix} -0.0011 & -15.6349 & -0.7060 & -0.2609 & -225.6177 & -77.1251 \\ -0.0011 & -15.6349 & 0.7060 & 0.2609 & -225.6177 & -77.1251 \end{bmatrix} \quad (63)$$

Os polos de malha fechada do sistema mostram agora que o sistema está estabilizado (estabilidade marginal, mas isso não é um problema para a aplicação proposta):

$$P_{dc} = \begin{pmatrix} 0.0000 + 0.0000i \\ 0.9852 + 0.0000i \\ 1.0000 + 0.0000i \\ 0.9971 + 0.0000i \\ 0.9971 - 0.0000i \\ 0.9994 + 0.0000i \end{pmatrix} \quad (64)$$

3.3 Simulações do sistema em malha fechada

Convém agora mostrar algumas simulações realizadas com o controlador que foi projetado. Vale salientar também que foi usado o modelo não linear para estes testes, a fim de verificar como o controlador se comportaria com um modelo mais próximo da realidade.

Conforme pode ser visto nas figuras 8 e 9, as rodas moveram-se para trás para compensar um ângulo negativo, o que faz sentido para o comportamento esperado, isto é, o dispositivo andar para trás quando o piloto inclina-se para trás. Claramente, o mesmo acontece quando o ângulo é positivo (usuário inclinado para frente).

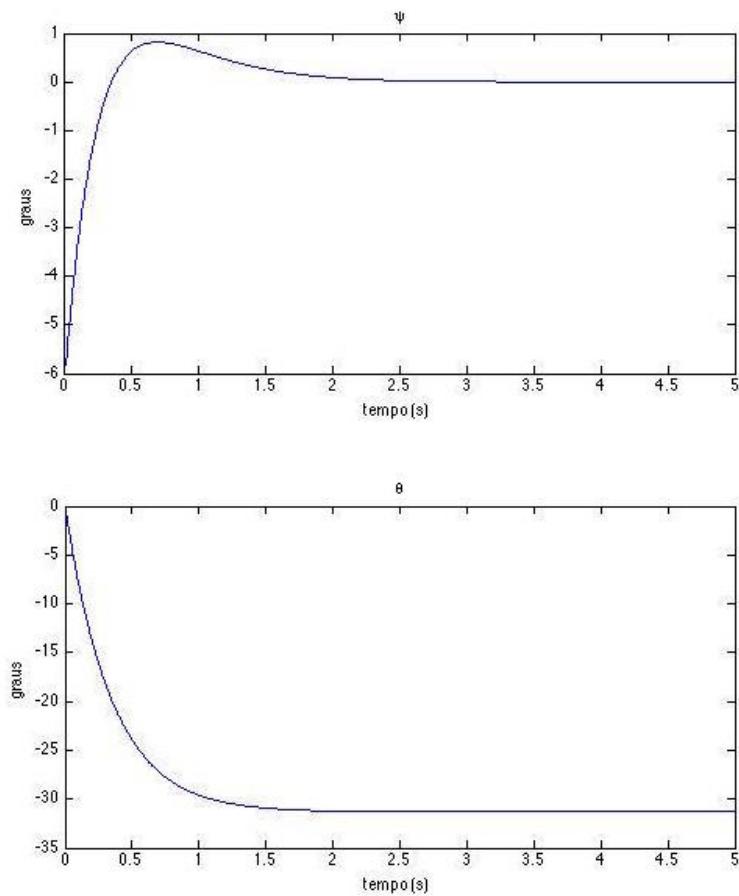


Figura 8: Respostas temporais do modelo não-linear para ângulo de inclinação de -6 graus

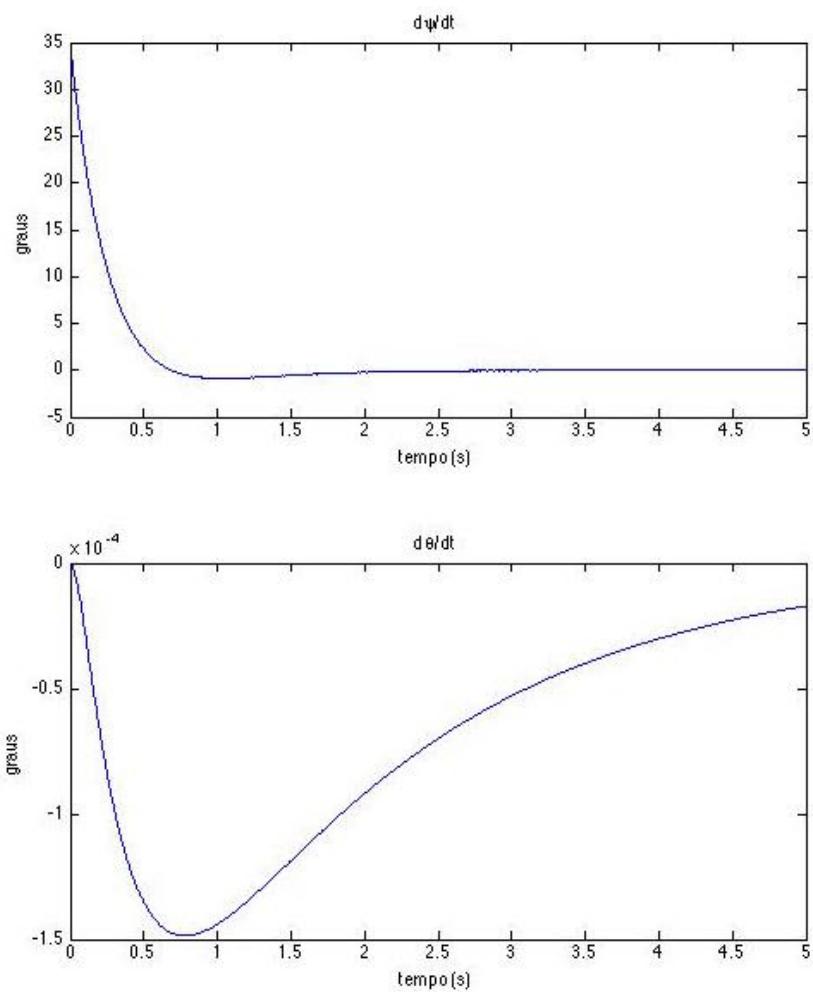


Figura 9: Derivadas das respostas temporais do modelo não-linear para ângulo de inclinação de -6 graus

4 Software

Para implementar a metodologia de controle discutida em outras seções, obviamente é necessário ter um software específico no Arduino. Esse software deve ser capaz de ler as medições angulares dos sensores e transformá-las em uma medida de ângulo confiável e, com essa medida, aplicar o algoritmo de controle, atuando corretamente nos motores.

A questão de leitura dos sensores envolve, como já foi previamente discutido neste trabalho, em [11] e em diversas outras literaturas, uma junção entre o sinal do acelerômetro e do giroscópio. Para isso, existem diversos algoritmos conhecidos e testados, cada um com suas vantagens e desvantagens.

Inicialmente, tentamos desenvolver nossos próprios algoritmos, mas as medições não estavam adequadas, principalmente em situações envolvendo grandes acelerações. Para o escopo deste trabalho, decidimos, portanto, utilizar a biblioteca i2cdev (www.i2cdevlib.com), que implementa um filtro de fusão de sensores que utiliza uma funcionalidade interna do MPU6050 chamada DMP (digital motion processing), com resultados bastante precisos, mesmo com acelerações bastante intensas. Este algoritmo usa como base os ângulos de Euler que estão descritos em [2] e [4].

Posto isso, partimos para a abstração do dispositivo, criando classes convenientes, conforme é mostrado na figura 10.

Esta abstração está em linha com as boas práticas de programação e facilita a criação de códigos fáceis de manter e alterar. Um exemplo disso, é que nosso dispositivo tem algumas limitações mecânicas, como a necessidade de haver uma pequena pausa quando da inversão dos motores, a fim de proteger os drivers de superaquecimento causado por dois comandos opostos sendo executados ao mesmo tempo. Assim sendo, a classe dos motores implementa isso automaticamente, sem que seja necessário se preocupar com essa pausa fora da classe.

Por isso, na versão final, foram criados blocos de abstração para diferentes componentes que compõem o carro. Isso foi feito para organizar o projeto e facilitar os testes e o desenvolvimento, uma vez que é possível testar os módulos separada e independentemente para verificar o seu funcionamento antes de serem conectados todos os blocos em um mais complexo.

O código desenvolvido pela equipe, para Arduino, utilizado no projeto encontra-se no apêndice deste trabalho e no CD que encontra-se anexo a ele.

Programa Principal						
Namespace AngleSensor	Class DualMotor		Class Joystick	Class PinRange	Class PinRange	Namespace Bluetooth
MPU6050	Motor 1	Motor 2	Joystick	Potenciômetro 1	Potenciômetro 2	HC-06
DMP						
Acelerômetro						
Giroscópio						

Figura 10: Níveis de abstração e classes utilizadas

4.1 Simuladores

Além do projeto físico, para facilitar o desenvolvimento e favorecer o paralelismo de tarefas, foram criados diferentes simuladores que modelam diferentes aspectos do projeto, para que fosse possível construir a parte de software sem que ainda estivesse concluída a parte de hardware ou mecânica.

O foco de todos eles sempre foi o de testar diferentes programas de equilíbrio e controle sem precisar do carro físico e, mesmo com ele construído, os simuladores agilizam testes rápidos, uma vez que não é necessário transferir o programa para o Arduino ou mesmo compilar o programa, já que foram desenvolvidos para serem interpretados por navegadores.

Esses simuladores cumprem o importante papel de validar candidatos a controlador contra um modelo da planta, com diferentes graus de precisão e rigor.

Deve ficar evidente que esses simuladores também possibilitam um melhor entendimento de como cada tipo de controlador e seus parâmetros afetam o comportamento do dispositivo, sem a necessidade de se estar submetendo o carro ao desgaste mecânico de constantes e repetitivos experimentos com diferentes parâmetros.

Nas figuras 11, 12 e 13, pode-se evidenciar alguns dos simuladores que realizamos. Os mesmos encontram-se disponíveis no CD que se encontra anexo a este trabalho e necessitam apenas de um navegador moderno para funcionar.

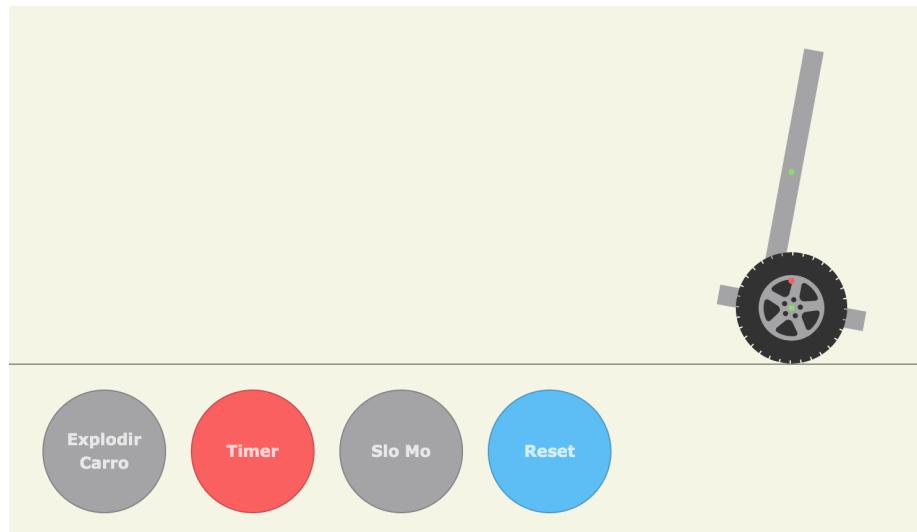


Figura 11: Simulador gráfico

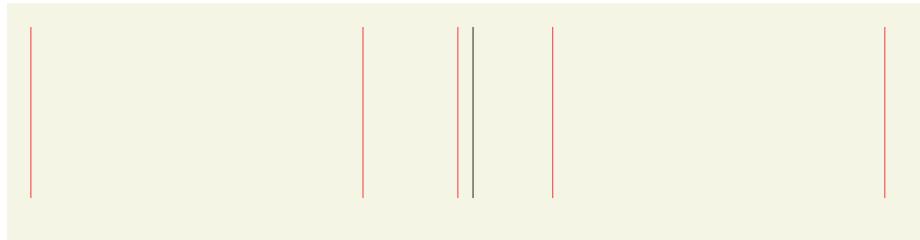


Figura 12: Simulador de ângulo de equilíbrio



Figura 13: Respostas variando de acordo com os parâmetros do controlador

5 Desenvolvimento e implementação prática

Vale destacar inicialmente que, além do estudo da modelagem do sistema proposto, onde se percebe maior aplicação dos conceitos desenvolvidos nas disciplinas da Engenharia Elétrica com ênfase em Controle e Automação, o desenvolvimento prático das partes elétricas e mecânicas do projeto é também uma grande fonte de conhecimento, além de ser decisiva para a verificação da parte teórica estudada.

Neste sentido, foram desenvolvidos alguns tópicos que versam sobre as dificuldades, as decisões e os resultados obtidos na elaboração física do sistema de duas rodas auto-estabilizável, que estão nas próximas páginas listados.

Os tópicos são divididos em dois blocos, que condizem com as fases de desenvolvimento do sistema, sendo chamados de Protótipo 1 e Protótipo 2.

5.1 Protótipo 1

Esta seção versa sobre os estudos desenvolvidos, as dificuldades encontradas, as escolhas realizadas e os resultados obtidos, no período de 11 de março de 2015 à 11 de agosto de 2015 no contexto de desenvolvimento das partes elétricas e mecânicas do sistema físico.

5.1.1 Plataforma de sustentação

Há a necessidade de suportar cerca de 130Kg, que equivale ao peso do projeto junto ao de um usuário, sem apresentar deformações. Estudou-se a possibilidade de utilização de uma plataforma de ferro de 5mm de espessura, entretanto, embora a mesma fosse robusta o suficiente para atender os requisitos de não deformação, o peso de aproximadamente 10kg se mostrou demasiadamente elevado.

A opção escolhida ficou a cargo de uma plataforma de aço, conforme a figura 14, de cerca de 4kg, permitindo uma grande economia no peso do projeto, mas tendo ainda sua rigidez reforçada pela presença de uma cantoneira transversal, a qual sustentará o eixo do sistema.

Com a escolha da plataforma de sustentação, passou-se, então, à definição das suas dimensões e dos orifícios que possuiria para a fixação dos demais componentes do sistema. Nas imagens 15, 16, 17 e 18, são ilustrados os estudos desenvolvidos neste âmbito, bem como a estrutura final apresentada pela plataforma. Vale destacar, que a plataforma foi modelada inicialmente para o suporte de quatro motores de tração, uma bateria estacionária de 12V, as placas controladoras do sistema e do guidão do projeto. Os cortes laterais, a soldagem das extremidades e a confecção dos orifícios ficou a cargo de uma manufatura particular devido à falta de intimidade dos alunos com serviços do tipo.



Figura 14: Plataforma de sustentação

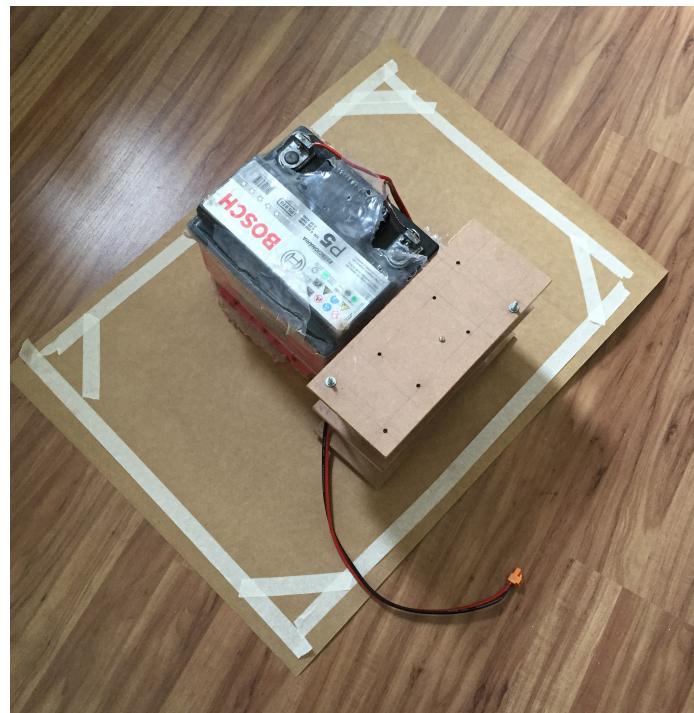


Figura 15: Estudo das dimensões da plataforma

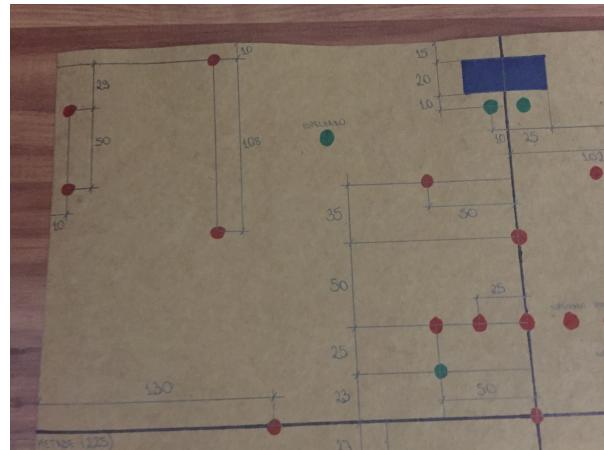


Figura 16: Ilustração dos furos enviados à manufatura



Figura 17: Plataforma obtida



Figura 18: Plataforma obtida com usuária em cima

5.2 Rodas de tração

O suporte de uma carga de 130Kg, muito semelhante ao que um carrinho de mão é destinado (instrumento tipicamente usado para movimentação de cargas de algumas dezenas de quilogramas), indicaram desde o início aos alunos a viabilidade de utilização das mesmas rodas que as utilizadas no respectivo veículo. O equipamento, exibido na figura 19, apresenta ainda grande facilidade para aquisição e custo moderado.



Figura 19: Roda de carrinho de mão

5.2.1 Motores elétricos

Um sucinto estudo de projetos de Segway, de cadeira de rodas automatizadas e de bicicletas elétricas, recomendou à equipe o emprego de motores com 12V ou 24V de alimentação, potência nominal de 1000W e elevado torque em baixas rotações. Foi, então, encontrado um motor de baixíssimo custo, em comparação à média de motores de especificações semelhantes no geral, cujas características são: 12V de alimentação, 180W de potência nominal, força de 5kgf.cm, rotação de 3500RPM e dimensões reduzidas.

Neste sentido, decidiu-se investir na compra de 4 motores (apresentados na figura 20), que juntos são capazes de promover uma elevada potência, tornando o projeto suficientemente robusto. Isto é, aparentemente, do ponto de vista dos motores, o limite de massa suportado pelo projeto é bastante elevado. Um problema na utilização desse motor é a elevada rotação oferecida. Será necessário fazer uma redução da velocidade com consequente aumento do torque dos motores através de engrenagens, correntes dentadas e de um controle PWM.



Figura 20: Motores DC

5.2.2 Sistema de redução da velocidade por engrenagens

Após a compra dos motores, a necessidade de se efetuar a redução da velocidade do motor, com consequente aumento do torque oferecido ao sistema, se mostrou ainda mais clara. Inicialmente, para efeitos de acoplamento da força gerada pelos motores no eixo do projeto, a utilização de algum sistema de redução já havia sido considerada. Assim, prevendo um suporte do controle PWM na estabilização da velocidade, foi definida como meta a elaboração de um sistema de redução do tipo 1:3 (basicamente pretende-se a diminuição de 3 vezes da velocidade de rotação e incremento de 3 vezes do torque).

A princípio, a equipe trabalhava com a ideia de desenvolvimento da sua própria caixa de redução. Entretanto, dado que esta é uma atividade complexa, que demandaria tempo excessivo, além de contar com baixa experiência dos alunos na confecção de peças em oficinas, a projeção de um sistema próprio de redução foi abandonada. A segunda ideia proposta era de utilização de engrenagens e correntes dentadas de motocicletas. Estes são equipamentos que suportam elevados valores de tração, além de serem facilmente adquiridos e terem preço acessível. Contudo, apresenta pequena relação de redução, traz mais massa ao projeto e prevê a ocorrência de mais ruídos assim que o sistema estiver em movimento. Foram adquiridos 4 conjuntos idênticos de engrenagens para ligação dos 4 motores em uso (vide figura 21). As coroas têm 48 dentes e os pinhões têm 14 dentes, promovendo uma redução de aproximadamente 3,5 vezes.



Figura 21: Componentes do sistema de redução

5.2.3 Baterias

O estudo inicial das baterias parte da ideia de que há um projeto robusto que precisa ser alimentado, ao mesmo tempo em que se pretende economizar ao máximo o peso de suas partes integrantes. A bateria tem que prover 12V ao projeto, com capacidade de abastecer com até 15A cada motor simultaneamente. Assim, a soma de todos os motores em funcionamento pode demandar até 60A da bateria em regime. Trabalhou-se com as alternativas de utilização de baterias de carro e baterias estacionárias.

Sabe-se que a bateria de carro é fácil de ser adquirida, mas não oferece elevada corrente em regime. A mesma é projetada para promover elevadas correntes em um curto espaço de tempo, o que é necessário para o acionamento do motor de partida. A bateria estacionária, utilizada tipicamente em sons potentes de carros, apresenta elevada corrente em regime, entretanto é mais difícil de ser encontrada. Costuma, além disso, ser levemente mais cara que uma bateria de carro. O peso das características levantadas para cada bateria fez o grupo optar por uma bateria estacionária, apresentada na figura 22. Vale destacar também que a aquisição da bateria acrescenta peso elevado ao projeto, além de possuir dimensões consideráveis.

5.2.4 Eixos do sistema

Como projetado inicialmente, o eixo do sistema deveria ter uma função extra de contribuir com a robustez da plataforma de sustentação. Assim, mirando um gasto baixo na compra de materiais, a equipe decidiu investir em materiais de custo baixo, encontrados nos chamados comércios de sucatas. Os materiais adquiridos estão apresentados na Figura 23.



Figura 22: Bateria estacionária

Os materiais foram cortados, torneados e lixados na oficina do Laboratório de Microeletrônica do prédio da Engenharia Elétrica, pelo Sr. Jair Barboza, e posteriormente foram soldados na cantoneira selecionada em uma manufatura privada. O eixo alcançou, assim, a configuração ilustrada nas Figuras 24 e 25.

5.2.5 Placa controladora

A equipe desde o início já trabalhava com a ideia de utilização de um Arduino para controle e regulação da estabilidade do sistema. A escolha de qual tipo de Arduino trabalhar, todavia, teve que ser mais profundamente analisada. Sabe-se que o controlador deverá suportar entrada de 6 potenciômetros de ajuste de parâmetros do programa, 6 sensores de temperatura (um pra cada motor e driver), 6 pinos de informação do Acelerômetro/Giroscópio e 2 pinos de informação do módulo Bluetooth. Ainda deverá ser capaz de controlar 2 drivers com 2 entradas PWM cada e a velocidade de rotação das ventoinhas de resfriamento dos mesmos. Assim, devido ao elevado número de entradas analógicas requisitadas no projeto, o Arduino Uno, que seria o pretendido a ser utilizado devido à facilidade de obtenção, foi substituído pelo Arduino Mega, mostrado na figura 26. Na escolha do controlador, vale ainda fazer uma pequena observação: as funções listadas acima podem ou não serem implantadas no projeto, no entanto, é necessário que o controlador do projeto aceite a possível inserção das mesmas.



Figura 23: Materiais de sucata adquiridos



Figura 24: Eixo sem roda do sistema



Figura 25: Eixo com roda do sistema



Figura 26: Arduino Mega

5.2.6 Drivers de açãoamento dos motores

Estes são dispositivos responsáveis por efetuar o intercâmbio dos sinais de baixa potência gerados na saída do Arduino (em formato PWM) em sinais de alta potência capazes de efetuar o açãoamento dos motores.

Foram utilizados como base para o estudo e desenvolvimento dos drivers alguns esquemas disponíveis sem custos na internet, com destaque ao material desenvolvido pelo site Robotizando, mantido por Daniel Omar Basconcello Filho [5].

O esquema elétrico e a placa de circuito impresso produzidos pela equipe estão apresentado nas figuras 27 e 28, respectivamente. Na figura 29 o circuito está subdividido em partes distintas para facilitar o entendimento do conjunto.

Nesta ilustração, pode-se destacar a existência de um controle de polaridade por meio de transistores simples (1), um chaveamento das correntes elevadas com transistores de efeito de campo (FETs) (2), um circuito dobrador de tensão (3), além das entradas em formato PWM (4), as saídas para o motor (5) e um circuito regulador e indicador de funcionamento (6). Na figura 30, é exibido um dos drivers desenvolvidos.

5.2.7 Placa de comunicação

Foi notado que, com a utilização de um grande número de entradas, saídas e controladores externos dos parâmetros do Arduino, as dimensões do aglomerado de fios que se formaria tornariam complexas as tentativas de localização de problemas no circuito.

Neste contexto, os alunos perceberam que seria interessante a confecção de um circuito de comunicação entre o controlador e todas as demais partes do projeto. Assim, desenvolveu-se uma placa de circuito impresso, apresentadas nas

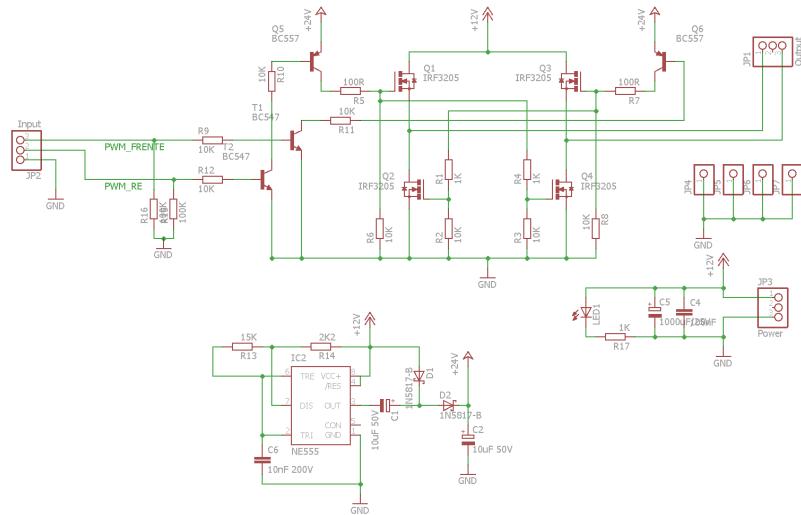


Figura 27: Esquema elétrico do driver

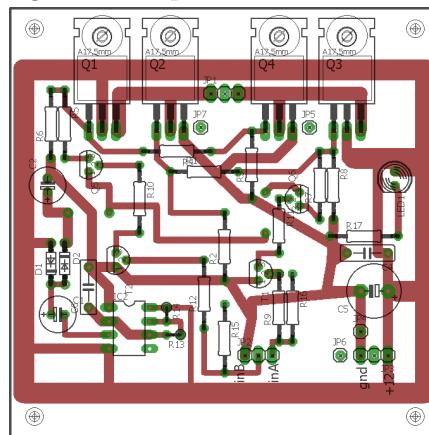


Figura 28: Placa de circuito impresso do driver

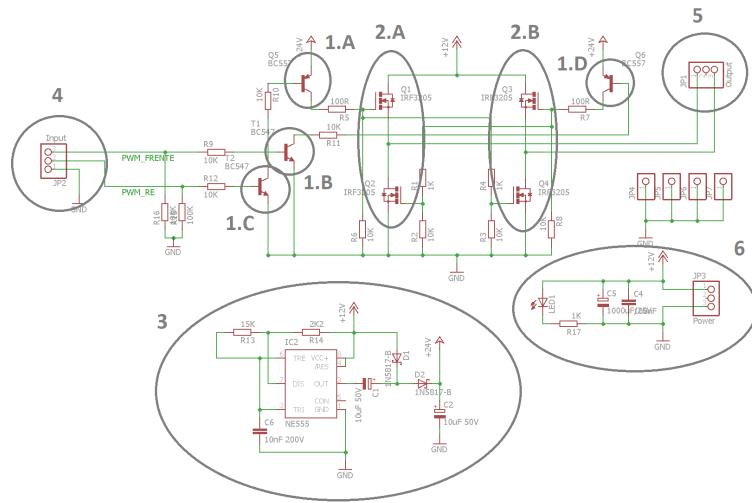


Figura 29: Esquema elétrico detalhado do driver

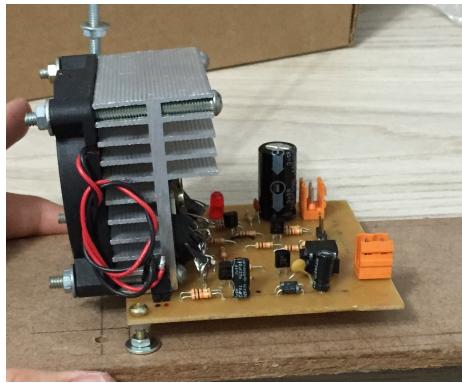


Figura 30: Driver montado pelo grupo

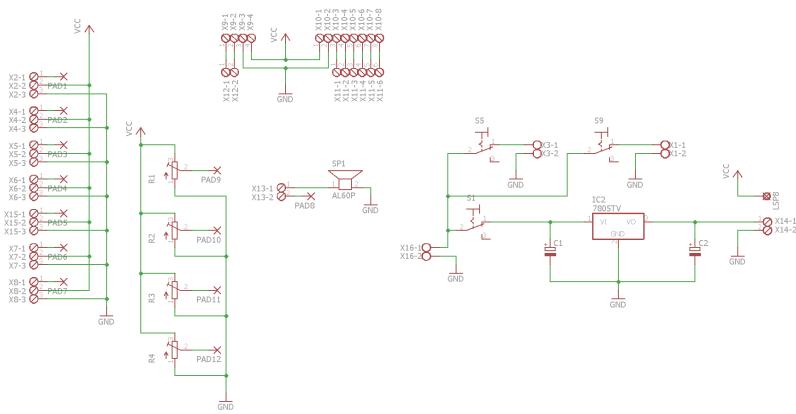


Figura 31: Esquema elétrico da placa de comunicação

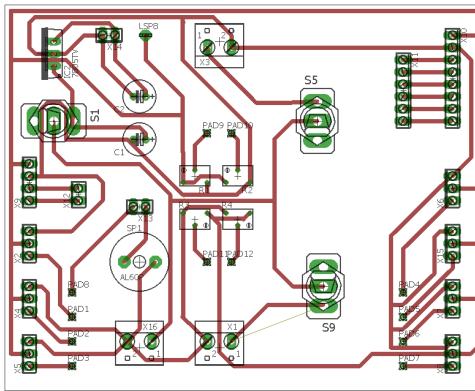


Figura 32: Placa de circuito impresso da placa de comunicação

Figuras 31 e 32, com as funções de receber os dados do projeto, como os sensores de temperatura e os potenciômetros de controle de estabilidade; gerenciar a alimentação do Arduino e dos drivers; e receber e enviar as informações dos módulos Bluetooth e do Acelerômetro/Giroscópio.

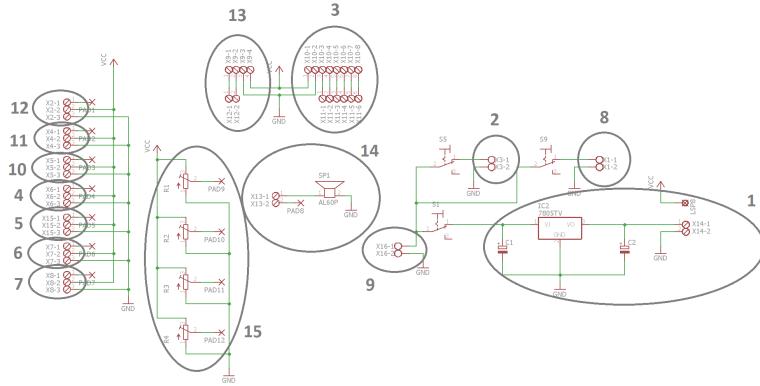


Figura 33: Esquema elétrico detalhado da placa de comunicação

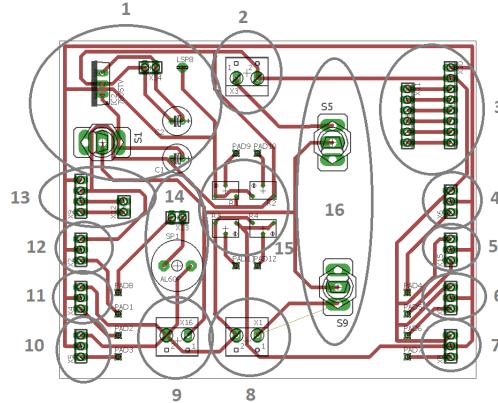


Figura 34: Placa de circuito impresso da placa de comunicação

Nas figuras 33 e 34, as partes estão devidamente selecionadas e numeradas para facilitar o entendimento. Destaca-se o circuito de gerenciamento de uma alimentação de 9V (1), a conexão da alimentação dos drivers (2 e 8), o gerenciamento do acionamento dos mesmos (16), o módulo de entrada do Acelerômetro/Giroscópio (3), o módulo de entrada do Bluetooth (13), os módulos de entrada dos sensores de temperatura (4 a 7, 10 a 12), os módulos de entrada de potenciômetros (15), o sistema de aviso sonoro (14) e a entrada de alimentação da bateria de 12V (9). Na figura 35, é exibida a configuração final da placa de comunicação.



Figura 35: Montagem da placa de comunicação

5.2.8 Sistemas auxiliares

O projeto conta inicialmente com os sensores de temperatura para monitoramento do aquecimento dos motores e dos drivers, os sensores de posição (Acelerômetro/Giroscópio) e ainda um módulo Bluetooth. Optou-se pela aquisição de sensores de temperatura do tipo LM35, vide figura 36, pela facilidade de aquisição e pela simplicidade do encapsulamento.

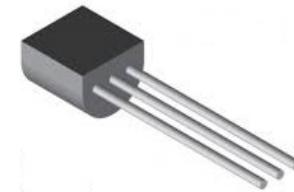


Figura 36: Encapsulamento LM35

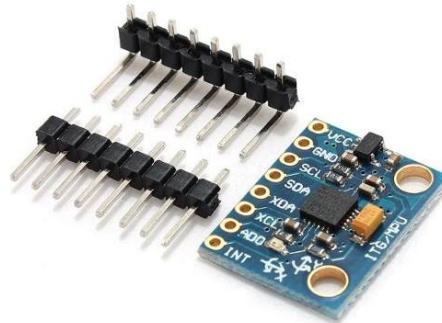


Figura 37: Sensor MPU-6050

Para controle de estabilidade, foi escolhida a compra de um instrumento que agregasse Acelerômetro e Giroscópio no mesmo dispositivo. Assim, foi obtido

um sensor Acelerômetro/Giroscópio 3 Eixos MPU-6050, apresentado na figura 37, que é largamente utilizado para estudo de posição junto a controladores Arduinos.

O módulo Bluetooth será utilizado para efeitos de apresentação. Foi comprado o modelo HC06, vide figura 38, pela facilidade de manuseamento. Através deste aparelho, será possível construir gráficos de posição e movimento da plataforma do projeto em um computador fixo e externo, em tempo real. Além disso, pode-se explorar a ideia de desenvolvimento de um aplicativo para celular Android (o módulo não tem suporte à comunicação com sistemas iOS) que exiba a temperatura dos 2 drivers e dos 4 motores do projeto em uma fase mais avançada do projeto.



Figura 38: Módulo Bluetooth

5.2.9 Disposição dos sistemas elétricos

O desenvolvimento da parte elétrica gerou 4 placas de circuito impresso que devem ficar localizadas em zona segura e de fácil acesso no projeto. Assim, foi confeccionado um sistema de placas com camadas, vide figura 39, que asseguram uma boa disposição do que foi esquematizado.

5.2.10 Configuração final do sistema

Após cinco meses de desenvolvimento, mais precisamente do período de 11 de março de 2015 a 11 de agosto de 2015, a equipe finalmente alcançou um modelo semelhante ao que havia sido projetado, o qual está ilustrado na Figura 40 e 41.

5.2.11 Testes realizados e problemas encontrados

Ao final da prototipagem do sistema, o mesmo pode ser exposto a testes mais fiéis a condições que o mesmo estaria vulnerável em funcionamento real. O modelo de teste realizado está ilustrado na Figura 42.

Neste contexto, pode-se perceber alguns defeitos de mecanismos específicos que afetavam e comprometiam o sistema como um todo. Percebeu-se que o maior problema se encontrava nos motores. Os motores comprados tem a característica de oferecer apenas um fio de ligação da alimentação, que consideramos ser o polo positivo. O outro fio esperado, que consideramos ser o negativo,

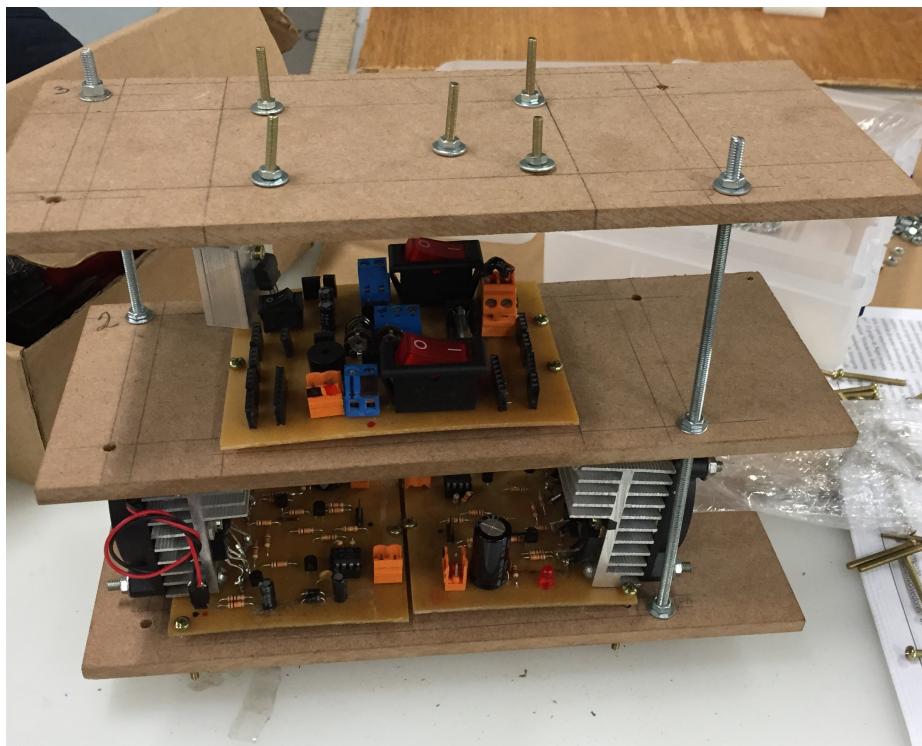


Figura 39: Parte eletrônica montada

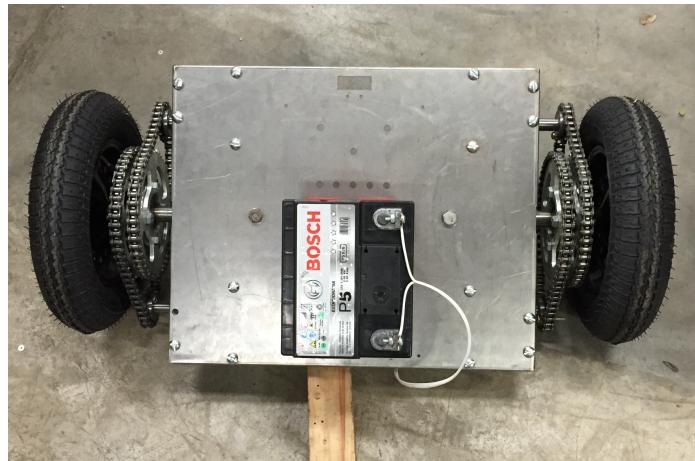


Figura 40: Protótipo 1 visto de cima

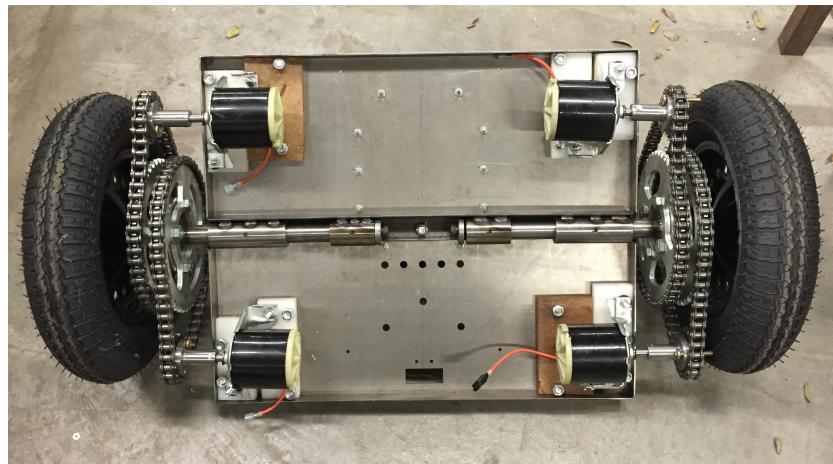


Figura 41: Protótipo 1 visto de baixo

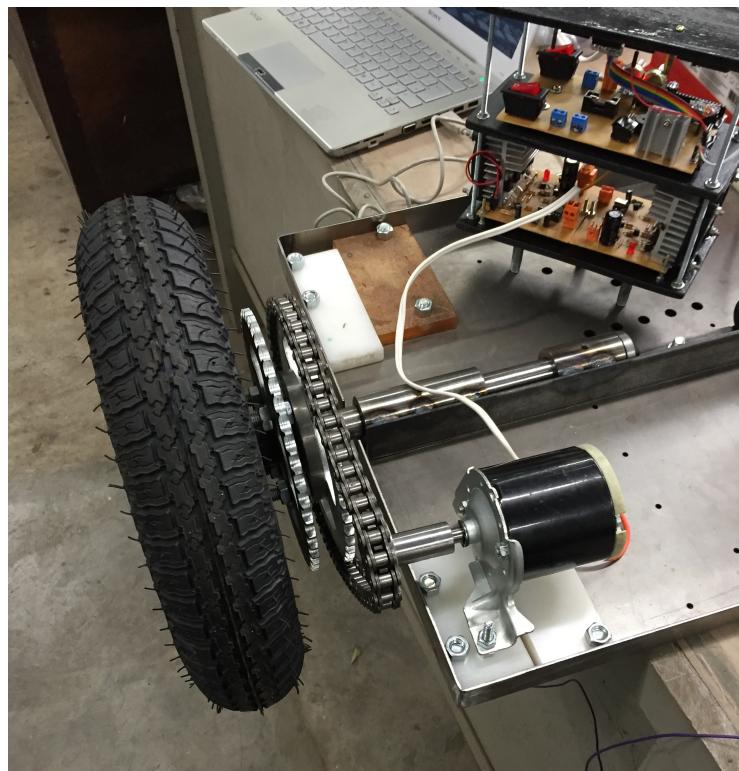


Figura 42: Ilustração de um dos testes de funcionamento do circuito realizados

encontra-se soldado internamente à carcaça do motor. Assim, para que o funcionamento conjunto dos motores fosse possível, a carcaça dos mesmos deveria ser isolada da plataforma e dos eixos, que são constituídos de materiais condutores. O sistema de isolamento projetado com placas de nylon não atendeu às necessidades, fazendo com que o acionamento conjunto dos drivers colocasse em curto a alimentação dos motores. Um segundo problema encontrado é a elevada corrente requisitada pelos motores, às quais não há capacidade de gerenciamento adequado oferecido pelos drivers. Os testes realizados perceberam danos contínuos na placa de comunicação e nos drivers do sistema, ilustrado na Figura 43. A corrente que circulava pelas placas excedia o máximo suportado, chegando a romper trilhas. Inclusive a fiação como um todo do sistema esquenta bruscamente com o funcionamento do sistema de tração. Devido a isso, uma nova placa de comunicação teve de ser feita. O driver danificado continuou operando conforme era previsto, portanto não necessitou de substituição.

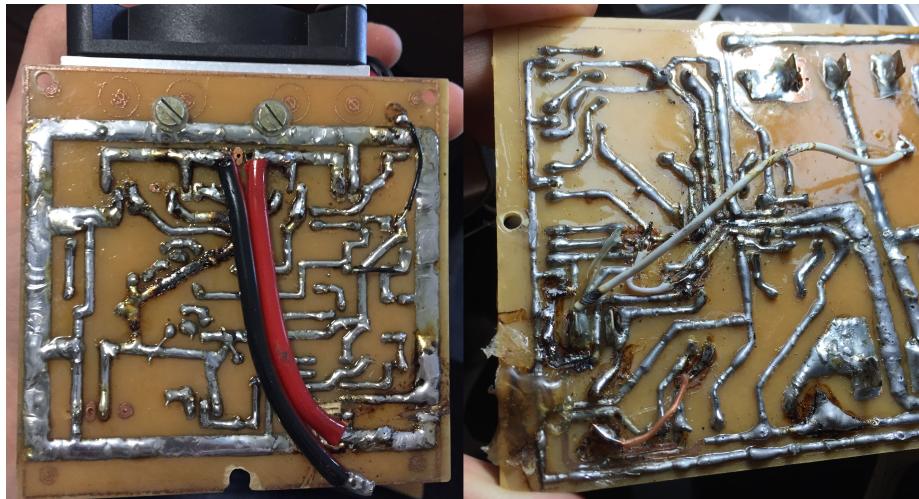


Figura 43: Protótipo 1 visto de cima

5.3 Protótipo 2

Esta segunda seção versa basicamente sobre as dificuldades encontradas na elaboração do Protótipo 1 e nas novas escolhas tomadas com finalidade de obter um modelo físico mais eficiente, capaz de atender as especificações iniciais.

5.3.1 Disposição dos novos motores

Devido ao mau acoplamento dos motores adquiridos durante a elaboração do protótipo 1, cerca da metade do sistema de tração e transmissão teve de ser abandonado. Iniciou-se, assim, um novo estudo de configuração de novos

motores, como é ilustrado na Figura 44. Neste contexto, algumas peças foram abandonadas, minimizando os efeitos de sobrepeso do projeto. O motor adquirido é do tipo largamente utilizado no funcionamento de limpadores de parabrisas de pickups. Alcançam amperagem aproximada de 10A, trabalhando com alimentação de 12V.

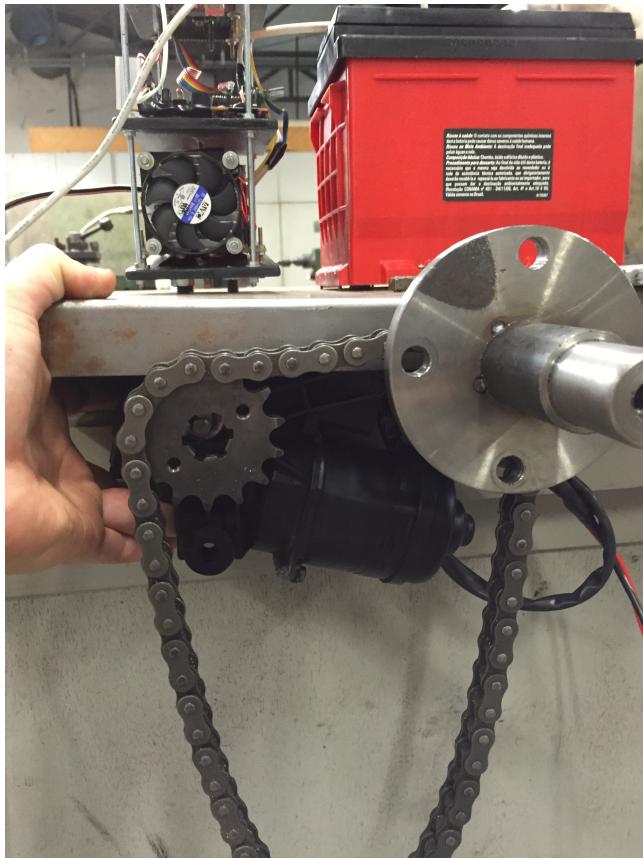


Figura 44: Estudo de um novo sistema de tração

5.3.2 Configuração do sistema de tração

Para se adequar a velocidade de rotação das rodas a um valor razoavelmente aceitável, a equipe decidiu não implementar um conjunto redutor da velocidade através de engrenagens, como havia sido feito anteriormente. Sabendo que os novos motores utilizados já possuem uma robusta caixa de redução interna, de relação 1:50 aproximadamente, o torque do motor acabou sendo disponibilizado em robustez suficiente sem necessitar de redução. Adaptando-se o motor às rodas, chegamos nas configurações disponíveis nas Figuras 45, 46 e 47.

Devido ao elevado grau de atrito entre as buchas (soldadas às cantoneiras) e os eixos giratórios (levados às rodas), a equipe decidiu também aplicar uma graxa branca, com finalidade de suavizar os movimentos, o que pode ser visto na Figura 48.



Figura 45: Eixos direito e esquerdo desenvolvidos

5.3.3 Confecção do guidão do sistema

Inicialmente, buscou-se confeccionar um guidão com funcionamento semelhante ao utilizado em um Segway. Entretanto, devido ao elevado grau de atrito proporcionado pelo acoplamento do guidão à plataforma do sistema, optou-se por abandonar a ideia e investir na implementação de um módulo joystick, que fosse capaz de coordenar através do funcionamento do seu potenciômetro embutido, a movimentação lateral do sistema, o que está evidenciado na Figura 49.

5.3.4 Configuração mecânica final do projeto

Por fim, ao longo de aproximadamente dois meses de trabalho sobre o protótipo 2, somando-se ainda os cinco meses de atividades desenvolvidas ao longo do protótipo 1, o projeto alcançou uma configuração mecânica satisfatória, sobre a qual será possível trabalhar na elaboração do programa que visa coordenar a movimentação do dispositivo. As imagens abaixo, 50 e 51, apresentam as peças finais, devidamente levantadas e pintadas, que compõe o sistema mecânico

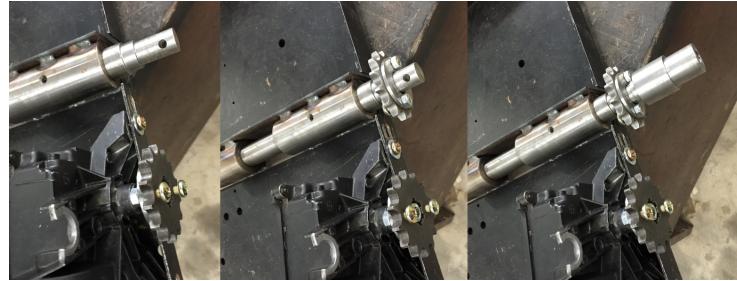


Figura 46: Mostra das peças desenvolvidas para cada eixo

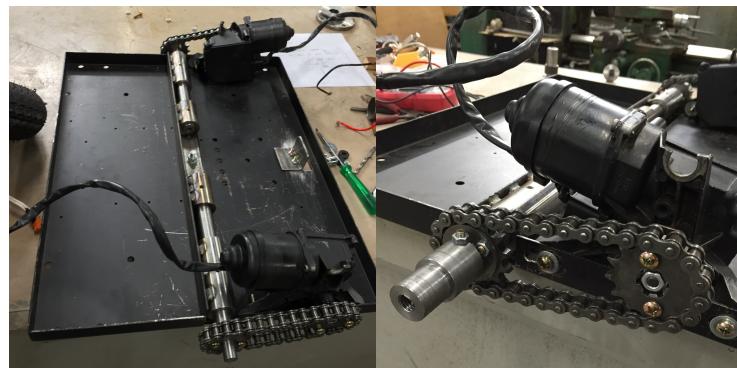


Figura 47: Nova transmissão obtida vista na parte inferior do sistema



Figura 48: Graxa aplicada sobre os eixos do sistema



Figura 49: Apresentação do formato do guidão e do joystick embutido

como um todo, e a disposição assumida pelo projeto montado, que alcança o peso de aproximadamente 33Kg.



Figura 50: Apresentação das peças empregadas na parte mecânica do sistema

5.3.5 Testes iniciais e elaboração de um template para o Arduino

Os primeiros testes realizados fazem referência à capacidade do sistema levar uma pessoa sobre sua estrutura. Percebeu-se que uma pessoa, de até 70Kg, é facilmente levada pelo protótipo 2. Isso deu motivação à equipe a continuar trabalhando na elaboração do programa que coordenará a movimentação do sistema. Na Figura 52 é ilustrado o teste executado.



Figura 51: Disposição mecânica final assumida pelo projeto



Figura 52: Teste de suporte de uma pessoa sobre o sistema

Em seguida, foram estudadas as respostas obtidas do módulo acelerômetro/giroscópio entregues às entradas do Arduino. Para isso o sistema foi disponibilizado em diversas configurações, as quais são ilustradas nas Figuras 53 e 54. As respostas obtidas estão apresentadas na Figura 55.



Figura 53: Inclinação máxima para frente e para trás



Figura 54: Diposição paralela ao chão e de equilíbrio natural

Através das respostas obtidas e analisando-se as entradas e saídas configuradas no Arduino, a equipe deu início à formatação de um programa no Arduino, que fosse um template básico, ou seja, uma plataforma para o desenvolvimento da lógica de controle do sistema. Nas Figuras 56 e 57 tem-se uma mostra do programa confeccionado.

```

AxX = -1604 | AxY = -6172 | AxZ = 15692 | Tmp = 27.87 | GyX = -137 | GyY = -165 | GyZ = -121
AxX = -1596 | AxY = -6148 | AxZ = 15689 | Tmp = 27.78 | GyX = -137 | GyY = 185 | GyZ = -114
AxX = -1620 | AxY = -6168 | AxZ = 15724 | Tmp = 27.78 | GyX = -137 | GyY = 188 | GyZ = -97
AxX = -1646 | AxY = -6104 | AxZ = 15760 | Tmp = 27.62 | GyX = -132 | GyY = 151 | GyZ = -120
AxX = -1628 | AxY = -6204 | AxZ = 15624 | Tmp = 27.73 | GyX = -111 | GyY = 168 | GyZ = -144
AxX = -1624 | AxY = -6320 | AxZ = 15756 | Tmp = 27.82 | GyX = 11 | GyY = 165 | GyZ = -91
AxX = 704 | AxY = -212 | AxZ = 16303 | Tmp = 27.78 | GyX = -36 | GyY = 282 | GyZ = -193
AxX = 244 | AxY = -652 | AxZ = 17120 | Tmp = 27.78 | GyX = 41 | GyY = 148 | GyZ = -93
AxX = 244 | AxY = -652 | AxZ = 17120 | Tmp = 27.78 | GyX = 41 | GyY = 148 | GyZ = -93
AxX = 216 | AxY = -454 | AxZ = 17090 | Tmp = 27.73 | GyX = -145 | GyY = 184 | GyZ = -53
AxX = 172 | AxY = -444 | AxZ = 17100 | Tmp = 27.73 | GyX = -25 | GyY = 145 | GyZ = -119
AxX = 156 | AxY = -460 | AxZ = 17220 | Tmp = 27.78 | GyX = -134 | GyY = 151 | GyZ = -138
AxX = 172 | AxY = -460 | AxZ = 17152 | Tmp = 27.87 | GyX = -138 | GyY = 169 | GyZ = -124
AxX = 196 | AxY = -668 | AxZ = 17112 | Tmp = 27.82 | GyX = -17 | GyY = 138 | GyZ = -89
AxX = 604 | AxY = -106 | AxZ = 17188 | Tmp = 27.82 | GyX = 100 | GyY = 138 | GyZ = -117
AxX = 409 | AxY = -128 | AxZ = 17196 | Tmp = 27.87 | GyX = 38 | GyY = 126 | GyZ = -99
AxX = 408 | AxY = -184 | AxZ = 17192 | Tmp = 27.73 | GyX = -54 | GyY = 147 | GyZ = -96
AxX = 352 | AxY = -196 | AxZ = 17040 | Tmp = 27.82 | GyX = -45 | GyY = 94 | GyZ = -142
AxX = 3420 | AxY = -849 | AxZ = 11688 | Tmp = 27.73 | GyX = 1493 | GyY = -444 | GyZ = -120
AxX = 400 | AxY = -100 | AxZ = 11688 | Tmp = 27.73 | GyX = 1493 | GyY = -444 | GyZ = -120
AxX = 4628 | AxY = -12072 | AxZ = 11690 | Tmp = 27.73 | GyX = 239 | GyY = 123 | GyZ = -128
AxX = 4628 | AxY = -12072 | AxZ = 11690 | Tmp = 27.73 | GyX = 239 | GyY = 123 | GyZ = -128
AxX = 4472 | AxY = -13798 | AxZ = 11692 | Tmp = 27.75 | GyX = -20 | GyY = -121 | GyZ = -69
AxX = 4580 | AxY = -13680 | AxZ = 11684 | Tmp = 27.75 | GyX = -127 | GyY = 155 | GyZ = -106
AxX = 4528 | AxY = 13760 | AxZ = 11492 | Tmp = 27.75 | GyX = -190 | GyY = 169 | GyZ = -103
AxX = 4528 | AxY = 13760 | AxZ = 11652 | Tmp = 27.75 | GyX = -157 | GyY = 165 | GyZ = -123
AxX = 4400 | AxY = 13890 | AxZ = 11448 | Tmp = 27.78 | GyX = -140 | GyY = 190 | GyZ = -99
AxX = 4432 | AxY = 13760 | AxZ = 11652 | Tmp = 27.82 | GyX = -129 | GyY = 163 | GyZ = -139
AxX = 4304 | AxY = 11640 | AxZ = 11584 | Tmp = 27.73 | GyX = -118 | GyY = 179 | GyZ = -116
AxX = 4412 | AxY = 11824 | AxZ = 11552 | Tmp = 27.76 | GyX = -132 | GyY = 162 | GyZ = -113
AxX = 4492 | AxY = 11872 | AxZ = 11884 | Tmp = 27.73 | GyX = -287 | GyY = 218 | GyZ = -90
AxX = 888 | AxY = 2344 | AxZ = 17256 | Tmp = 27.82 | GyX = -3038 | GyY = 1223 | GyZ = -113
AxX = 460 | AxY = 224 | AxZ = 17256 | Tmp = 27.82 | GyX = 40 | GyY = 91 | GyZ = -128
AxX = 460 | AxY = -36 | AxZ = 17132 | Tmp = 27.78 | GyX = -203 | GyY = 198 | GyZ = -162
AxX = 280 | AxY = -144 | AxZ = 17200 | Tmp = 27.78 | GyX = -358 | GyY = 188 | GyZ = -174
AxX = 256 | AxY = -146 | AxZ = 17176 | Tmp = 27.73 | GyX = -123 | GyY = 163 | GyZ = -106
AxX = 452 | AxY = -44 | AxZ = 17220 | Tmp = 27.82 | GyX = 63 | GyY = 80 | GyZ = -103
AxX = 532 | AxY = 208 | AxZ = 17320 | Tmp = 27.78 | GyX = 51 | GyY = 100 | GyZ = -95
AxX = -1440 | AxY = -5940 | AxZ = 15810 | Tmp = 27.82 | GyX = -3351 | GyY = 1246 | GyZ = -85
AxX = -1668 | AxY = -6116 | AxZ = 15868 | Tmp = 27.78 | GyX = -28 | GyY = 154 | GyZ = -96
AxX = -1700 | AxY = -6256 | AxZ = 15768 | Tmp = 27.87 | GyX = -194 | GyY = 160 | GyZ = -95
AxX = -1644 | AxY = -6240 | AxZ = 15656 | Tmp = 27.78 | GyX = -93 | GyY = 188 | GyZ = -126
AxX = -1600 | AxY = -6188 | AxZ = 15760 | Tmp = 27.73 | GyX = -93 | GyY = 174 | GyZ = -132

```

Inclinação máx pra frente

Paralelo ao chão

Inclinação máx pra trás

Paralelo ao chão

Inclinação máx pra frente

Figura 55: Resultados colhidos do módulo acelerômetro/giroscópio

5.4 Protótipo em escala

A versão miniatura do carro foi desenvolvida com a intenção de permitir o desenvolvimento do software sem precisar ser transportada a versão de tamanho real. Além disso, os comandos de movimentação são dados de forma diferente, uma vez que não há interferência física de um ser humano. Por isso, para receber comandos, a versão miniatura usa o módulo Bluetooth HC-06, que funciona no modo slave e recebe através de uma interface serial no computador pareado.

Essa miniatura foi baseada em um projeto anterior de um dos membros, sendo aprimorada com os conhecimentos adquiridos desde então. Em tese, ela se assemelha bastante ao veículo em tamanho real, tendo sido possível utilizar o mesmo modelo matemático para projetar o controlador.

Com o menor tamanho, vieram algumas vantagens, a saber:

- Motores mais lineares e muito mais baratos
- Bateria mais leve e barata
- Ausência de correntes
- Drivers mais simples

Essas vantagens se mostraram bastante significativas, pois foi possível estabelecer um controle eficaz em pouco tempo, sem a necessidade de grandes ajustes posteriores. Isso nos leva a crer que o dispositivo em tamanho real, por ter correntes associadas ao motor, acrescentou um atraso de transporte que não havia sido modelado e que é bastante difícil de se lidar, gerando assim, certas dificuldades na hora de controlar o carro.

```

// TCC_Segway_Template_Arduino | Arduino 1.6.3
Arquivo Editar Sketch Ferramentas Ajuda
TCC_Segway_Template_Arduino
//SISTEMA DE CONTROLE SOBRE DUAS RUEDAS AUTOESTABILIZADO
//Programa: Teste Acelerometro e Giroscópio (MPU6050) e Controle dos Motores
//Obs: Não alterar as entradas e saídas do Arduino. Elas já estão ligadas no projeto.

#include<Wire.h> //Carrega a biblioteca Wire

const int MPU=0x68; //Endereço I2C do MPU6050
int AccX,AccY,AccZ,Tap,gyX,gyY,gyZ; //Variáveis para armazenar os valores dos sensores

//Obs: Não alterar as entradas e saídas do Arduino. Elas já estão ligadas no projeto.

int motor_dir_frente = 11; // Driver 1
int motor_dir_tras = 10;
int motor_esq_frente = 6; //Driver 2
int motor_esq_tras = 5;

int pot1 = 0; //Potenciómetro 1 (para testes)
int pot2 = 1; //Potenciómetro 2 (para testes)

int joystick_x = 2; //Joystick
//Observação: o joystick tem o joystick direito para a pessoa mover lateralmente o Segway.
//O eixo de rotação do joystick é o eixo Y.
//Posição central: 514 (Considerar a posição central a faixa: 480-600)
//Máxima amplitude para a direita: 273
//Máxima amplitude para a esquerda: 1015

int istrução = 0; //Variável para guardar o valor lido do potenciómetro 1
int mov_lateral = 0; //Variável para guardar o valor lido do joystick

void setup()
{
  Serial.begin(9600); //Inicializa a comunicação serial a 9600bits por segundo

  Wire.begin();
  Wire.beginTransmission(MPU);
  Wire.write(0x6B);
}

```

Figura 56: Primeira página do template básico desenvolvido

```

// TCC_Segway_Template_Arduino | Arduino 1.6.3
Arquivo Editar Sketch Ferramentas Ajuda
TCC_Segway_Template_Arduino
Serial.print(" | AccX = "); Serial.print(AccX);
//Calcula a temperatura e grava Celsius
Serial.print(" | Tap = "); Serial.print(Tap/140.00+36.53);
//Envia valor X do giroscópio para a serial
Serial.print(" | gyX = "); Serial.print(gyX);
//Envia valor Y do giroscópio para a serial
Serial.print(" | gyY = "); Serial.print(gyY);
//Envia valor Z do giroscópio para a serial
Serial.print(" | gyZ = "); Serial.print(gyZ);
//Envia valor 2 do giroscópio para a serial e o LCD
Serial.print(" | gy2 = "); Serial.println(gy2);
//Observação: ACCELEROMETER/GYROSCOPE:
//O eixo de rotação do motor não é o eixo de giroscópio e o Y, devido a sua posição no projeto.
//Todas as variáveis ACCELEROMETER para o acelerômetro no eixo Y abaixo:
//Para máxima inclinação para frente, AccY=620
//Para máxima inclinação para trás, AccY=1700
//Posição paralela ao chão, AccY=0
//Proximo ao equilíbrio natural, AccY=1000

//O programa abaixo movimenta o Segway com velocidade controlada pelo potenciômetro
//fazendo-o andar por 2 segundos para frente, depois parar por 0,5 segundo,
//em seguida movimenta para trás por mais 2 segundos e finaliza parando por 0,5 segundo.
//Isso é só para testes e é só o utilizado para se gerar os vídeos enviados.
//Observação:
//Sempre efetuar a parada completa dos motores antes de ligá-los novamente
//em outro sentido de rotação, por pelo menos 30ms. Se mandar por um instante um motor
//para frente e para trás ao mesmo tempo, o driver do motor queima
//Observação:
//Valor do "analogWrite(motor_A_B, X)" para o carro não consegue pelo atrito das engrenagens e do chão
//Valor do "analogWrite(motor_A_B, X)" para o carro atingir velocidade máxima: X=555
//este valor atinge mais ou menos 50m/h na total
//estes valores não consideram carga na cima do Segway

leitura = analogRead(pot1); //leitura" varia de 0 a 1023. A saída "analogWrite" pode ser de 0 a 255
Serial.println(leitura);
}

```

Figura 57: Segunda página do template básico desenvolvido

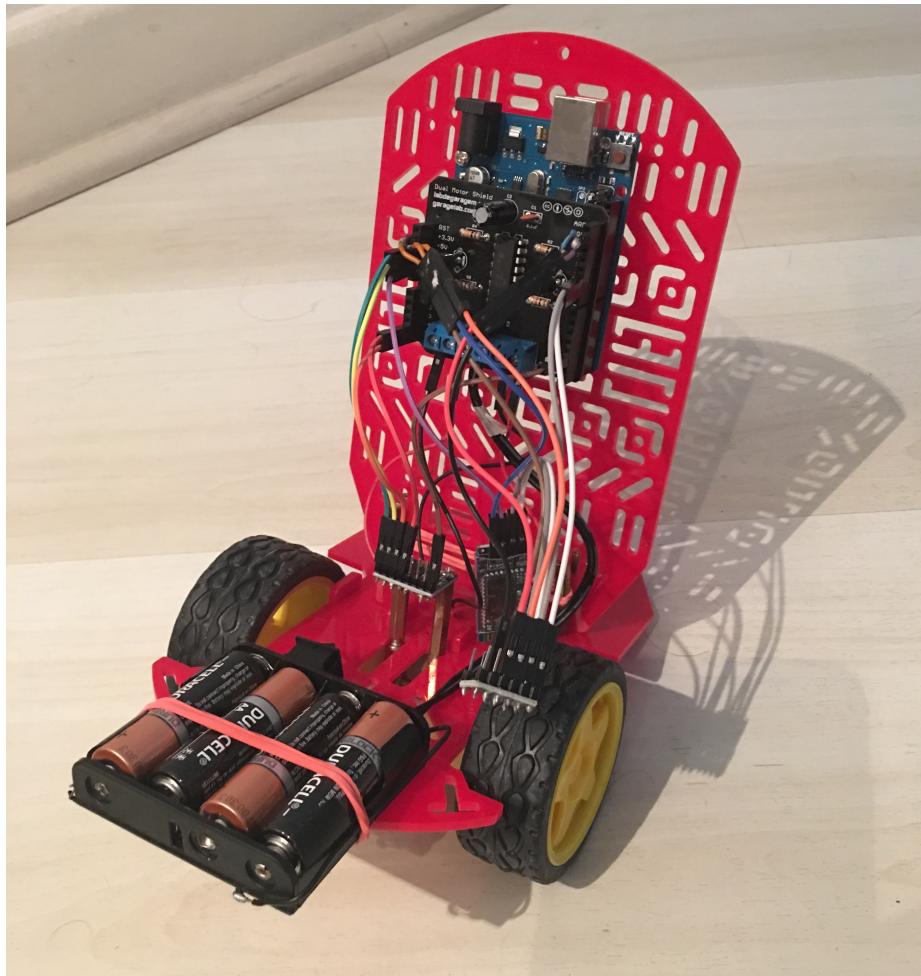


Figura 58: Protótipo em miniatura desligado

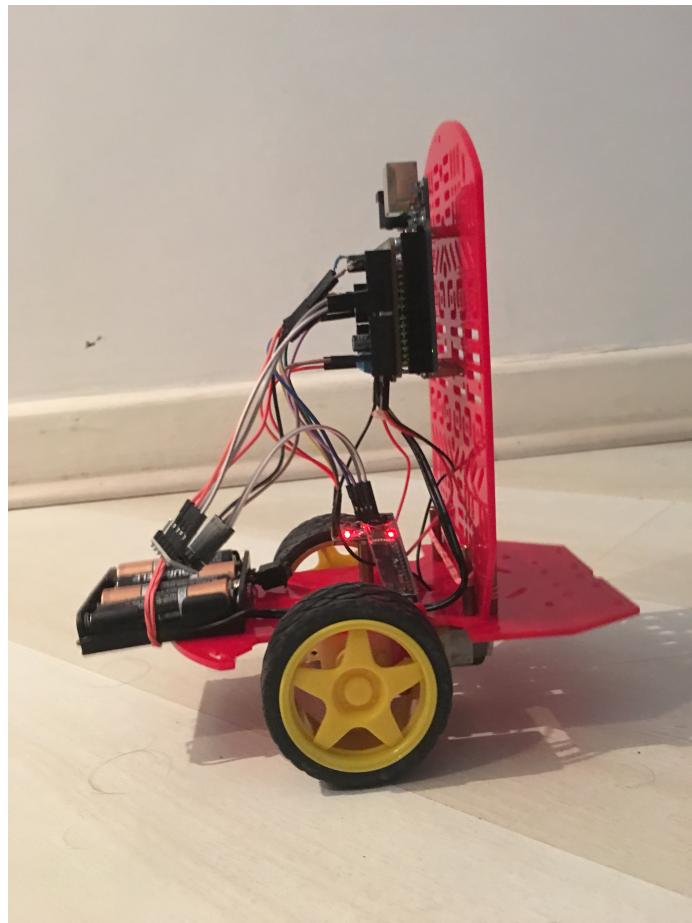


Figura 59: Protótipo em miniatura em funcionamento

6 Resultados

Nesta seção apresentamos os resultados obtidos neste projeto. Primeiramente, discutiremos brevemente a implementação real do controlador e suas consequências para a pilotagem do dispositivo, seguiremos com os resultados dos testes iniciais e as correções necessárias e melhorias tomadas e terminaremos com uma análise teórica de como as características físicas do piloto interferem na controlabilidade do dispositivo.

6.1 Implementação do Controlador

Ficou prontamente evidente que nosso projeto teórico não havia considerado o forte atrito estático do conjunto das rodas, correntes, engrenagens e motores. Foi necessário adicionar um sinal de compensação de atrito no PWM, para obtermos um comportamento adequado e mais próximo do calculado.

Para permitir a realização de curvas, o operador deve usar o joystick presente no cabo de apoio, virando-o proporcionalmente à velocidade de giro desejada. Além disso, implementamos um sistema de segurança que diminui a velocidade máxima de giro dependendo da velocidade do veículo, para evitar que esse perca o equilíbrio e caia.

6.2 Testes iniciais de funcionamento

Os primeiros testes foram feitos visando garantir algumas características elementares como:

- Rotação das rodas no sentido correto
- Velocidade máxima adequada
- Interação correta entre as partes mecânicas

Tendo isso como base, foi possível passar à análise de elementos um pouco mais sutis porém igualmente críticos, como a temperatura dos drivers, a manutenção do aperto dos parafusos, a lubrificação das partes móveis, entre outros. Com isso, chegamos a um ponto onde podíamos começar a testar a dirigibilidade do dispositivo.

Em um estágio inicial, começamos a testá-lo sem um piloto, para que pudéssemos trabalhar sobre a estabilidade sem maiores preocupações e riscos para o equipamento e/ou para o piloto. Pudemos constatar que os motores são capazes de oferecer uma velocidade bastante significativa em potência máxima, o que será suficiente para nosso escopo.

6.3 Dados dos sensores

A primeira dificuldade que se interpôs foi a de usar os dados dos sensores (acelerômetro e giroscópio), visto que esses precisam ser calibrados. Foi necessário desenvolver um programa que coletasse os dados dos sensores com o

dispositivo na vertical e em repouso, a fim de identificar os níveis de tolerância para que fossem considerados como o ”zero”. Vale ressaltar que, tendo poder de processamento para isso, é interessante que esse programa rode em conjunto com o uso normal, com o objetivo de compensar eventuais mudanças nos valores entregues pelos sensores, mantendo assim a sua calibração.

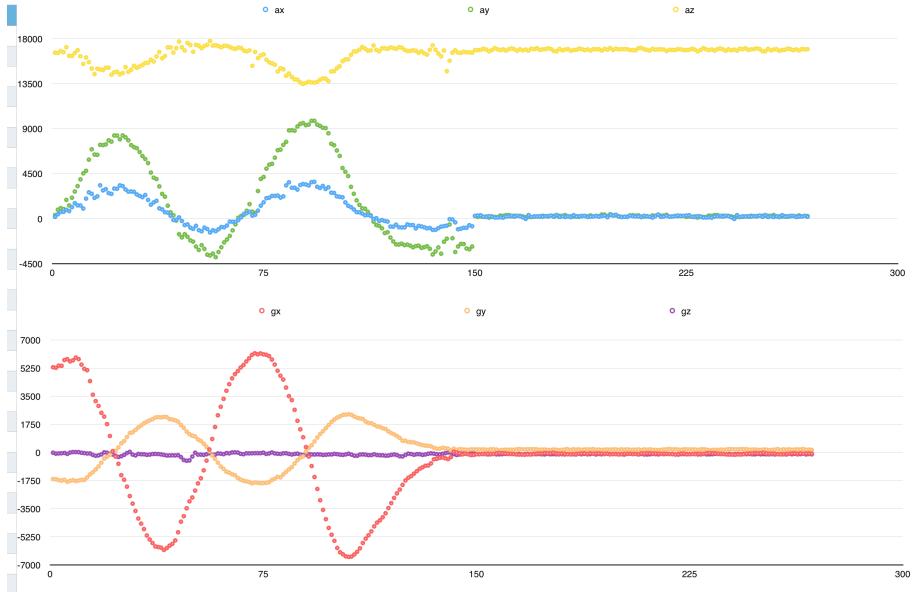


Figura 60: Primeiros dados coletados dos sensores (não tratados)

Além disso, foi necessário unir os sinais dos sensores, visando obter uma medida de ângulo de inclinação confiável. Para isso, usamos o filtro complementar citado em seções anteriores e, como mostra a figura 61, a medição foi bem mais aproveitável. Infelizmente, esse filtro ainda não foi suficiente para dar uma medição angular suficientemente precisa para o controlador. Foi necessário usarmos uma biblioteca chamada i2cDevLib, que faz a abstração do processo de medição de forma bastante eficiente e precisa. Pudemos rapidamente perceber como esta estava fornecendo dados adequados quando comparamos o ângulo de queda livre a uma senóide e estes eram bastante próximos. Isso bate com a teoria mecânica e com o modelo do dispositivo. Essa biblioteca está disponível para diversos tipos de sensores e pode ser encontrada em [3].

6.4 Adequando o controlador

Além de ser necessário considerar o atrito estático, foi necessário adequar os parâmetros do LQR de forma empírica. Ressaltamos que a necessidade disso não vem de uma falha do método ou de sua aplicação, mas sim de imperfeições do modelo, que não leva em conta diversas dinâmicas não-modeladas e atrasos

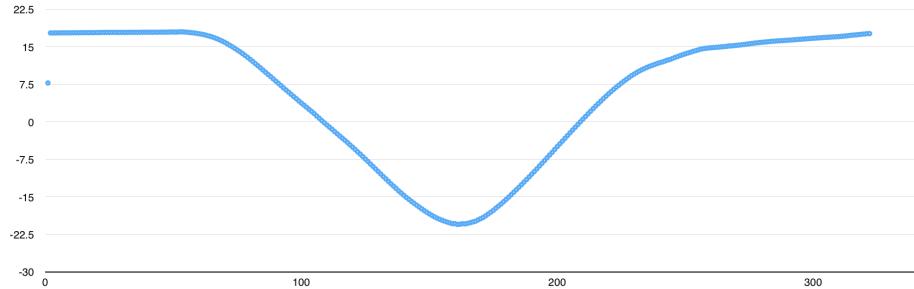


Figura 61: Medida do ângulo usando filtro complementar

de transporte/transmissão. Com isso, conseguimos adequar o comportamento do dispositivo para que fosse atingido o equilíbrio em torno da vertical.

6.5 Resultados na miniatura

No protótipo em miniatura, conseguimos controlar o ângulo e a posição com sucesso. Através do módulo bluetooth, conseguimos desenvolver a possibilidade de controlar a posição do carrinho por um controle remoto ou computador, simulando uma pessoa em cima do Segway e seus movimentos para frente e para trás.

Tal resultado nos mostra de forma expressiva que a estratégia de controle e sua implementação é suficiente para garantir um bom funcionamento do dispositivo, havendo condições mecânicas adequadas para isso. Ressaltamos que o maior problema do protótipo em tamanho real são as correntes e o atraso de transporte que estas inseriram no sistema.

7 Discussões e Conclusões

Aqui, tomaremos a liberdade de expor nossas opiniões sobre o projeto, os objetivos, as dificuldades em atingi-los e possíveis continuidades a serem dadas a este trabalho.

7.1 Cumprimento dos objetivos

No protótipo em miniatura, atingimos plenamente os objetivos de ter um dispositivo teleguiado, capaz de manter o equilíbrio enquanto anda por diversas superfícies. Tal protótipo serve como um validador físico do modelo matemático, do algoritmo de controle e da implementação programática deste.

Por outro lado, no protótipo em tamanho real, não foi possível adequar o seu comportamento para que um piloto pudesse dirigí-lo. Conseguimos, todavia, alcançar o equilíbrio em torno da vertical e acreditamos que só seria possível a pilotagem se fosse radicalmente melhorado o projeto mecânico, retirando folgas e minimizando ou removendo o atraso de transporte adicionado pelas correntes.

Não é exagero dizer que este projeto mostra que é plenamente possível o desenvolvimento de um veículo autoestabilizado com um orçamento relativamente limitado e uma adequada dose de conhecimento em mecânica, controle, eletrônica e programação.

7.2 Desempenho

O desempenho obtido pelo protótipo em miniatura foi totalmente satisfatório. Ele é capaz de andar com velocidade conveniente em diversos tipos de superfícies sem perder o equilíbrio. Além disso, as baterias oferecem tempo suficiente de autonomia para demonstrações longas.

No protótipo em tamanho real, após a troca de motores, obtivemos velocidade e potência adequados para um desempenho suficiente para suportar um piloto. O problema, porém, foi o atraso de transporte das correntes, que impedi um controle eficiente em situações nas quais o piloto estaria tentando mover o carro com o peso de seu corpo.

7.3 Desenvolvimento futuro

Com a experiência adquirida neste projeto, pode-se concluir que a fabricação de um dispositivo autoestabilizado é bastante difícil mas também altamente possível.

Pelo menos para engenheiros elétricos, a maior dificuldade, certamente, é a parte mecânica dada a pouca experiência que temos neste campo.

Com isso, queremos dizer que, para atingir os objetivos inicialmente propostos, acreditamos que seria necessário ter um projeto mecânico diferente em pelo menos um dos seguintes aspectos:

- Transmissão: uso de caixas de redução ou outro mecanismo que não adicione tanto atraso de transporte como as correntes.

- Drivers: ao invés do uso do PWM, uso de drivers lineares, que permitem um controle mais eficaz dos motores, sem os problemas inerentes ao PWM.
- Controle por Corrente: uso de motores controlados por corrente e não por voltagem para um controle melhor do torque corretivo. Eventualmente, seria interessante estudar também a possibilidade de se controlar a potência dos motores.

Com o que pudemos perceber nesse ano trabalhando sobre este projeto, acreditamos que pelo menos uma dessas melhorias seria suficiente para compensar as outras e permitir um funcionamento adequado para que um piloto possa controlar o veículo.

Quanto às possibilidades do uso de veículos semelhantes em larga escala, é necessário analisar diversos aspectos, dentre os quais, podemos citar:

- Custo do veículo: Mesmo com produção em quantidade e a decorrente eficiência de escala, é evidente que trata-se de um veículo relativamente caro para o consumidor brasileiro.
- Autonomia da bateria: Não é segredo algum que baterias são artefatos caros e de difícil descarte. Esse, inclusive, é um dos maiores problemas enfrentados pelos veículos elétricos. Há a necessidade de encontrar uma bateria leve, potente, durável e de recarga rápida para que o produto seja interessante para o consumidor. Infelizmente, cada uma dessas virtudes encarece ainda mais a bateria.
- Usabilidade no meio urbano: Hoje em dia já se vê diversos seguranças de shopping utilizando-se de Segways, mas é bem raro ver alguém utilizando-os fora desse contexto. É necessário entender se isso é um problema de marketing ou de produto, isto é, se as pessoas não usam porque acham estranho ou se não usam por não ser uma maneira eficiente de transporte urbano.
- Segurança: Por mais autoestabilizado que seja o veículo, sempre há o perigo da perda da controlabilidade, ocasionando quedas e ferimentos possivelmente graves.
-

Lista de Símbolos

d_b	Profundidade da plataforma de sustentação
f_m	Coeficiente de atrito viscoso dos motores
h_b	Altura da plataforma de sustentação
h_c	Altura do cilindro
$J_{\phi,b}$	Momento de inércia da base com relação a ϕ
$J_{\phi,c}$	Momento de inércia do cilindro com relação a ϕ
J_ϕ	Momento de inércia do corpo com relação a ϕ
$J_{\psi,b}$	Momento de inércia da base com relação a ψ
$J_{\psi,c}$	Momento de inércia do cilindro com relação a ψ
J_ψ	Momento de inércia do corpo com relação a ψ
J_w	Momento de inércia da roda
K_b	Constante contra eletromotriz dos motores
K_t	Constante de torque dos motores
L	Distância do eixo das rodas ao centro de massa
M	Massa do corpo, soma das massas do cilindro e da base
m	Massa da roda
m_b	Massa da plataforma de sustentação
m_c	Massa do cilindro
n	Relação de transmissão
R	Raio da roda
r_c	Raio do cilindro
R_m	Resistência elétrica dos motores DC
W	Distância entre as rodas
w_b	Largura da plataforma de sustentação

Referências

- [1] *Control tutorials for matlab and simulink - inverted pendulum: System modeling.* <http://ctms.engin.umich.edu/CTMS/index.php?example=InvertedPendulum§ion=SystemModeling>.
- [2] *Euler angles – from wolfram mathworld.* <http://mathworld.wolfram.com/EulerAngles.html>.
- [3] *I2Cdev Library.* <http://www.i2cdevlib.com>.
- [4] *Quaternion – from wolfram mathworld.* <http://mathworld.wolfram.com/Quaternion.html>.
- [5] *Robotizando – ponte-h 12v 40a.* http://www.robotizando.com.br/artigo_ponte_h_pg1.php.
- [6] C. BEALL, *Segway Remote Rider*, The Univeristy of Tennessee, 2012.
- [7] M. BURKERT, *Segway Design Project*, Grand Valley State University, 2004.
- [8] B. CARLSSON, *Mobile inverted pendulum - Control of an unstable process using open source real-time operating system*, Chalmers University of Technology, 2009.
- [9] J. J. DA CRUZ, *Controle Robusto Multivariável*, EDUSP, 1996.
- [10] D. JIN, *Development of a Stable Control System for a Segway*, Research Academy for Young Scientists, 2008.
- [11] H. G. MIN, *Complementary Filter Design for Angle Estimation using MEMS Accelerometer and Gyroscope*, Changwon National University, 2010.
- [12] K. OGATA, *Modern Control Engineering*, Prentice-Hall, 5 ed., 2009.
- [13] Y. YAMAMOTO, *NXTway-GS Model-Based Design - Control of self-balancing two-wheeled robot built with LEGO Mindstorms NXT*, CYBER-NET SYSTEMS CO., LTD., 2008.