

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

ARTHUR MIGUEL DE MEDEIROS BORTOLINI

**PROJETO E DESENVOLVIMENTO DE UM SISTEMA DE CONTROLE
PARA UM ATUADOR ELETRO-HIDRÁULICO**

Uma abordagem via linearização por realimentação

Natal – RN

2018

ARTHUR MIGUEL DE MEDEIROS BORTOLINI

**PROJETO E DESENVOLVIMENTO DE UM SISTEMA DE CONTROLE
PARA UM ATUADOR ELETRO-HIDRÁULICO**

Uma abordagem via linearização por realimentação

Relatório final apresentado ao Departamento de Engenharia Mecânica da Universidade Federal do Rio Grande do Norte como parte das exigências para a obtenção do título de Engenheiro Mecânico.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA MECÂNICA

Orientador: Prof. Dr. Wallace Moreira Bessa
Coorientador: Prof. Dr. João Deodato Batista dos Santos

Natal – RN
2018

ARTHUR MIGUEL DE MEDEIROS BORTOLINI

**PROJETO E DESENVOLVIMENTO DE UM SISTEMA DE CONTROLE
PARA UM ATUADOR ELETRO-HIDRÁULICO**

Uma abordagem via linearização por realimentação

Relatório final apresentado ao
Departamento de Engenharia Mecânica da
Universidade Federal do Rio Grande do
Norte como parte das exigências para a
obtenção do título de Engenheiro
Mecânico.

Natal, ____ de _____ de 2018.

BANCA EXAMINADORA

Prof. Dr. Wallace Moreira Bessa

Prof. Dr. João Deodato Batista dos Santos

Eng. Fernando Henrique Fernandes

Eng. Gabriel da Silva Lima

RESUMO

A evolução de novas abordagens de controle tem feito com que sistemas reais venham sendo controlados de forma satisfatória em aplicações que requerem certo grau de precisão. Além de melhorar o desempenho do controle, a utilização destas abordagens traz outras vantagens como a redução de custos, haja vista que controladores lineares exigem um aparato com certo grau de qualidade, e a eliminação da necessidade da linearização do sistema. Dentre as abordagens de controle não-linear, a que foi escolhida para o desenvolvimento deste trabalho foi a linearização por realimentação (*feedback linearization*). Esta é uma técnica de controle de implementação simples, mas seu desempenho depende muito da qualidade da modelagem do sistema. Neste trabalho, a abordagem de linearização por realimentação foi implementada no controle de um sistema eletro-hidráulico através de um micro controlador de baixo custo. Apesar da ampla utilização em diversos segmentos da indústria, sabe-se que os componentes destes sistemas são fontes de inúmeras não-linearidades as quais tornam difícil seu controle através de técnicas de controle linear, por isso, é importante que se estude a eficácia da implementação de técnicas de controle não-linear através da utilização de um aparato de baixo-custo. Ao controlador, foram submetidos diversos tipos de trajetória e o erro médio apresentado por cada trajetória foi medido.

Palavras-chave: linearização por realimentação; Arduino; controle não-linear; sistemas eletro-hidráulicos.

ABSTRACT

The evolution of new control approaches has made it possible for real systems to be satisfactorily controlled in applications that require a certain degree of precision. In addition to improving control performance, the use of these approaches has other advantages such as cost reduction, since linear controllers require an apparatus with a certain degree of quality, and eliminate the need for system linearization. Among the approaches of non-linear control, the one that was chosen for the development of this work was the feedback linearization. This is a simple implementation control technique, but its performance depends greatly on the quality of the system modeling. In this work, the feedback linearization approach was implemented in the control of an electro-hydraulic system through a low cost micro controller. In spite of the wide use in several segments of the industry, it is known that the components of these systems are sources of innumerable non-linearities which make their control difficult through linear control techniques, so it is important to study the effectiveness of the implementation of techniques of non-linear control through the use of a low-cost apparatus. To the controller, several types of trajectory were submitted and the average error presented by each trajectory was measured.

Keywords: feedback linearization; Arduino; nonlinear control; electro-hydraulic systems.

LISTA DE FIGURAS

Figura 1 – Representação do sistema eletro–hidráulico em estudo.	3
Figura 2 – Diagrama de corpo livre para o sistema.	4
Figura 3 – Corte da seção transversal da válvula e do cilindro.	5
Figura 4 – Não linearidade do tipo zona morta.	7
Figura 5 – Fluxograma representativo do algoritmo implementado.	14
Figura 6 – Esquema de montagem da bancada experimental.	15
Figura 7 – Atuador hidráulico de dupla ação.	17
Figura 8 – Potenciômetro linear.	17
Figura 9 – Fonte de tensão CC 24 V.	18
Figura 10 – Eletroválvula.	19
Figura 11 – Fonte de energia hidráulica.	19
Figura 12 – Arduino UNO R3 utilizado neste trabalho.	20
Figura 13 – L298N.	22
Figura 14 – Potenciômetro convencional.	22
Figura 15 – Esquema montado para calibração do potenciômetro linear.	23
Figura 16 – Esquema de circuito divisor de tensão.	24
Figura 17 – Curva de calibração do potenciômetro linear.	25
Figura 18 – Multímetro utilizado.	26
Figura 19 – Curvas de calibração das correntes nos solenoides da válvula.	26
Figura 20 – Comportamento da posição medida x_r para a Trajetória Senoidal.	27
Figura 21 – Comparação entre x_d , x_f e x_r para a Trajetória Senoidal (TS).	29
Figura 22 – Comparação entre \tilde{x}_f e \tilde{x}_r para a Trajetória Senoidal (TS).	29
Figura 23 – Comparação entre x e x_d para Trajetória Manual Lenta (TML).	32
Figura 24 – \tilde{x} para Trajetória Manual Lenta (TML).	32
Figura 25 – u_r e u_s para Trajetória Manual Lenta (TML).	33
Figura 26 – Comparação entre x e x_d para Trajetória Manual Rápida (TMR).	34
Figura 27 – \tilde{x} para Trajetória Manual Rápida (TMR).	34
Figura 28 – u_r e u_s para Trajetória Manual Rápida (TMR).	35
Figura 29 – Comparação entre x e x_d para Trajetória Quadrada (TQ).	36

Figura 30 – \tilde{x} para Trajetória Quadrada (TQ).	36
Figura 31 – u_r e u_s para Trajetória Quadrada (TQ).	37
Figura 32 – Comparação entre x e x_d para Trajetória Triangular (TT).	38
Figura 33 – \tilde{x} para Trajetória Triangular (TT).	38
Figura 34 – u_r e u_s para Trajetória Triangular (TT).	39
Figura 35 – Comparação entre x e x_d para Trajetória Senoidal (TS).	40
Figura 36 – \tilde{x} para Trajetória Senoidal (TS).	40
Figura 37 – u_r e u_s para Trajetória Senoidal (TS).	41
Figura 38 – Comparação do erro de todas as trajetórias.	43

LISTA DE TABELAS

Tabela 1 – Parâmetros do sistema.	12
Tabela 2 – Parâmetros do controlador.	13
Tabela 3 – Comparação entre <i>Arduino UNO R3</i> e <i>Advantech USB 4711A</i> .	21
Tabela 4 – Erro médio para Trajetória Manual Lenta (TML).	33
Tabela 5 – Erro médio para Trajetória Manual Rápida (TMR).	35
Tabela 6 – Erro médio para Trajetória Quadrada (TQ).	37
Tabela 7 – Erro médio para Trajetória Triangular (TT).	39
Tabela 8 – Erro médio para Trajetória Senoidal (TS).	41
Tabela 9 – Comparação entre erros médios de todas as trajetórias.	42

SUMÁRIO

1. INTRODUÇÃO	1
2. SISTEMA ELETRO–HIDRÁULICO	3
3. CONTROLE NÃO LINEAR	9
3.1. Linearização por realimentação	10
4. CONTROLADOR	12
4.1. Algoritmo	14
5. METODOLOGIA	15
5.1. Bancada	16
5.1.1. Atuador eletro–hidráulico	16
5.1.2. Potenciômetro linear	17
5.1.3. Fonte de tensão (24 V)	18
5.1.4. Válvula proporcional 4/3 vias	18
5.1.5. Unidade de energia hidráulica	19
5.1.6. Arduino <i>UNO R3</i>	20
5.1.7. Driver L298N	21
5.1.8. Potenciômetro convencional	22
5.2. Calibrações	23
5.2.1. Calibração do potenciômetro linear	23
5.2.2. Calibração dos terminais da eletroválvula	26
5.3. Tratamento de dados	27
6. ANÁLISE DOS RESULTADOS	31
6.1. TML – Trajetória manual lenta	32
6.2 TMR – Trajetória manual rápida	34
6.3. TQ – Trajetória quadrada	36
6.4. TT – Trajetória triangular	38

6.5. TS – Trajetória senoidal.....	40
6.6. Comparação geral.....	42
7. CONSIDERAÇÕES FINAIS	45
REFERÊNCIAS.....	47

1. INTRODUÇÃO

A necessidade de aprimorar, cada vez mais, a qualidade de processos industriais faz com que o estudo de técnicas de controle robustas se faça mais do que necessário. Sob este ponto de vista, alguns fatores que dificultam este aprimoramento são a robustez e o porte dos mecanismos empregados, as altas cargas e velocidades de trabalho de certas aplicações e os curtos períodos de tempo nos quais o controle deve ser executado. Desta forma, a utilização de abordagens robustas se mostra necessária.

De acordo com Ogata (2011), uma prática que se popularizou na indústria do século XX foi a utilização de teorias de controle linear em virtude da facilidade de sua implementação. Este tipo de abordagem foi utilizado por Saberi, Lin e Teel (1996). Entretanto, a utilização deste tipo de abordagem traz à tona a necessidade de linearizar sistemas, muitas vezes complexos, de modo que um modelo seja obtido a partir da escolha de certos parâmetros de controle.

Cumpra salientar que todos os sistemas reais possuem diversas não linearidades particulares e inerentes a estes e que estas não linearidades são outros fatores que dificultam um controle eficiente, visto que as mesmas implicam em imprecisões no modelo utilizado. Assim, a presença destas imprecisões, além de comprometer a performance do controlador, leva a uma redução do *range* de operação do mesmo.

Por estas razões, busca-se abordagens que ampliem o range de atuação e que considerem a modelagem das não linearidades. Assim, recorre-se às técnicas de controle não-linear. Dentre eles, está a linearização por realimentação (*feedback linearization*), a qual consiste em anular algebricamente as não-linearidades do sistema de modo que este tenha comportamento dinâmico em malha fechada. A utilização desta abordagem de controle implica, de certa forma, em contornar as imprecisões do modelo a partir de um monitoramento constante do controle, o qual é feito com uma instrumentação adequada. Cabe destacar que a qualidade do controle deste tipo de abordagem depende muito da qualidade do modelo utilizado no controlador.

Um tipo de sistema muito comum na automação de processo é o de sistemas eletro-hidráulicos. De acordo com Merrit (1967), a utilização de sistemas eletro-hidráulicos na automação industrial tornou-se popular pela possibilidade de

proporcionar cargas elevadas com uma boa precisão do movimento, boa dissipação de calor e lubrificação, boa rigidez e uma alta velocidade de resposta. Parte muito importante destes sistemas, os atuadores eletro–hidráulicos são fontes de inúmeras não linearidades (KIM et al., 2012). Além dos atuadores, outros itens desse tipo de sistema, como as próprias válvulas, também trazem imprecisões ao modelo de controle.

A implementação destas técnicas de controle requer certo aparato industrial: sensores e controladores, os quais possuem custo consideravelmente elevado. A implementação da técnica de linearização por realimentação foi realizada por Azevedo (2016) através do uso de uma placa de processamento de dados industrial.

Assim, alternativa é a utilização de aparato de baixo custo por parte da indústria. Embora também possua suas limitações, a utilização de controladores de baixo custo, como o Arduino, em detrimento de sistemas industriais diminui consideravelmente o custo de implementação destas técnicas de controle embora. Um Arduino foi utilizado por Juang e Lum (2013) no controle de um robô de duas rodas.

O objetivo deste trabalho é controlar um sistema eletro–hidráulico através da técnica de controle de linearização por realimentação, de modo que esta seja implementada através de um Arduino. O desempenho do controlador será avaliado mediante a imposição de diferentes tipos de trajetória ao atuador e da medição do erro médio para cada uma destas.

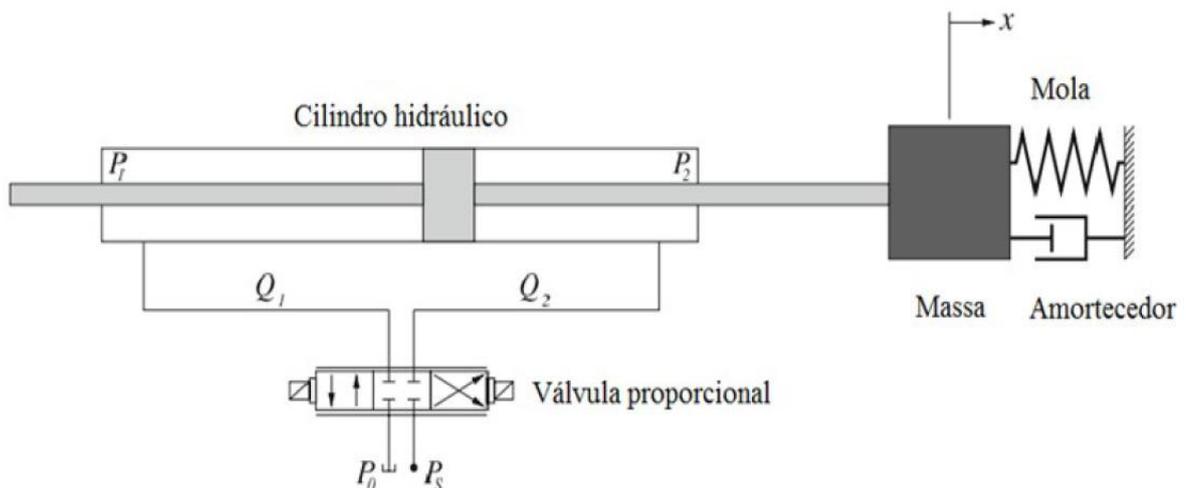
2. SISTEMA ELETRO-HIDRÁULICO

Diversas das atuais aplicações de engenharia fazem uso dos atuadores eletro-hidráulicos. Estes sistemas aliam uma eficiente transmissão de energia através de fluidos quase que incompressíveis com a velocidade e precisão dos sinais elétricos para o seu acionamento. Além disso, cargas elevadas podem ser suportadas por este tipo de componente mecânico. Desta forma, sistemas eletro-hidráulicos são utilizados nas mais diversas aplicações: aeronaves, embarcações de pesca (ALARÇIN, 2013), escavadores hidráulicos (YAN et al., 2013), suspensão ativa de automóveis (EKORU; PEDRO, 2013) e em manipuladores robóticos (YANG; HUANG; HAN, 2012).

Atrelado a isso, necessita-se cada vez mais de técnicas de controle que façam com que os sistemas reais reproduzam, com precisão, as tarefas desejadas. Deste modo, uma modelagem física de qualidade destes componentes torna-se imprescindível para a indústria.

O sistema é composto por uma válvula proporcional, ou servo-válvula, acionada por solenoides, e um cilindro hidráulico de dupla ação. O modelo utilizado neste trabalho baseia-se no modelo apresentado por Bessa, Dutra e Kreuzer (2016). A fig. (1) representa o sistema em estudo.

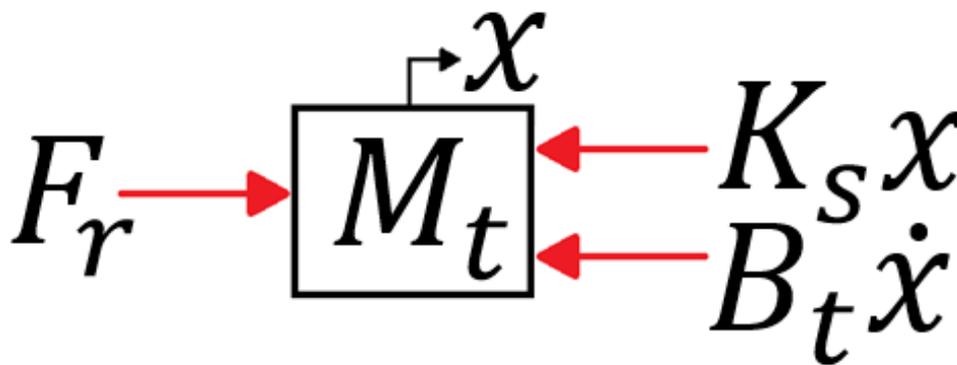
Figura 1 – Representação do sistema eletro-hidráulico em estudo.



Fonte: Bessa, Dutra e Kreuzer (2010, adaptado).

De início, aplica-se a Segunda Lei de Newton na extremidade do atuador afim de que se possa analisar as forças atuantes no local. Para o sistema em estudo neste trabalho, são considerados a força resultante F_r e as forças oriundas do amortecimento $B_t \dot{x}$, onde B_t é a constante de amortecimento do sistema e \dot{x} a velocidade do mesmo e da resistência da mola $K_s x$, onde K_s é a constante de rigidez da mola e x a posição. O diagrama de corpo livre da fig. (2) representa as forças atuantes no sistema, bem como suas direções e sentidos.

Figura 2 – Diagrama de corpo livre para o sistema.



Fonte: Autoria própria.

Onde M_t é a massa na extremidade do atuador e x é o grau de liberdade associado ao sistema. Admita-se que a diferença de pressão $\Delta P = P_1 - P_2$ entre as câmaras do cilindro, as quais, aplicadas às áreas A_1 e A_2 das câmaras do atuador, seja a responsável pelo surgimento da força resultante F_r na cabeça do pistão. Assim, após a aplicação da segunda lei de Newton, a equação que representa a dinâmica do sistema é dada por:

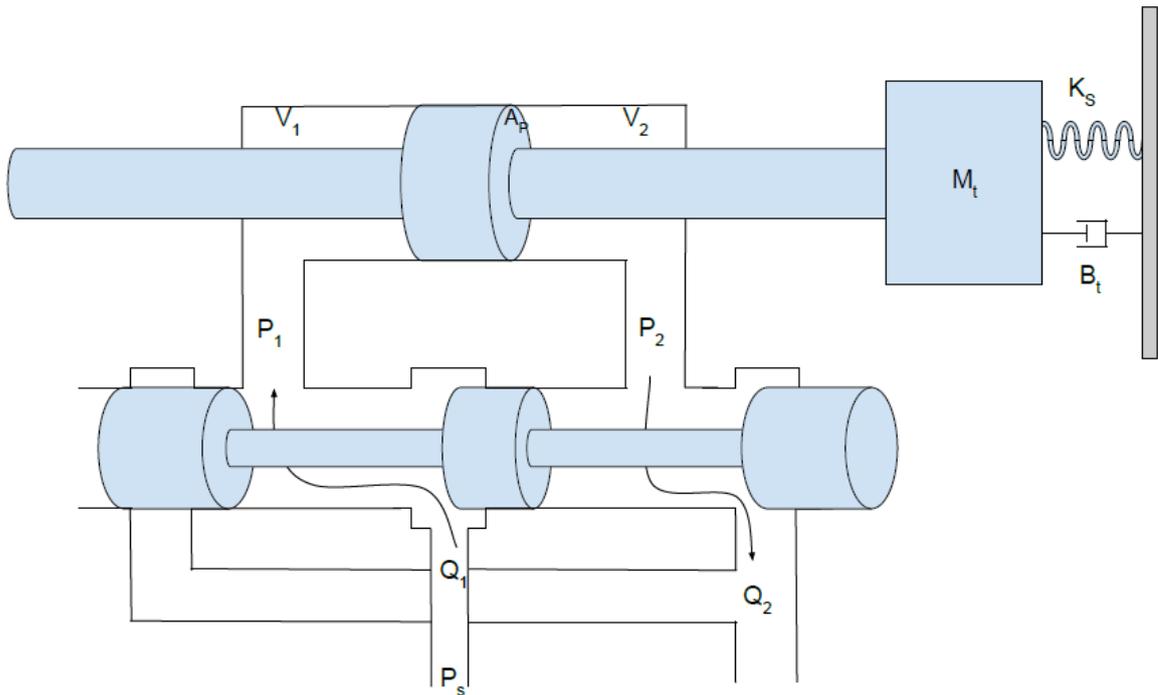
$$F_r = A_1 P_1 - A_2 P_2 = M_t \ddot{x} + B_t \dot{x} + K_s x \quad (2.1)$$

Por questões de simplificação, considera-se a haste do atuador como sendo passante. Isso implica que as áreas de atuação da força de pressão nas duas câmaras, sejam iguais. Assim, $A_p = A_1 = A_2$. Logo, a eq. (2.1), torna-se:

$$M_t \ddot{x} + B_t \dot{x} + K_s x = \Delta P A_p \quad (2.2)$$

A fig. (3) mostra uma vista em corte da seção transversal da válvula e do cilindro. Nela, é possível observar os orifícios da válvula por onde passa o fluido.

Figura 3 – Corte da seção transversal da válvula e do cilindro.



Fonte: Azevedo (2016).

Outro artifício utilizado para redução do número de variáveis é a utilização de uma vazão média $Q_l = (Q_1 + Q_2)/2$ do fluido nas linhas de transmissão do sistema. De acordo com Santos (2013), esta vazão é composta por três parcelas: a parcela do volume que ocupa o interior do atuador enquanto este se movimenta ($A_p \dot{x}$), a parcela do volume que vaza do cilindro ($C_{tp} \Delta P$) e a parcela da variação do volume devido à compressão ($(V_t/4\beta_e) \Delta \dot{P}$), aonde C_{tp} é o coeficiente de vazamento total do pistão, V_t é o volume total sob compressão e β_e é o módulo de rigidez volumétrica do óleo. Desta forma, tomando-se as câmaras do cilindro como um volume de controle e aplicando a este a equação da continuidade, o escoamento do fluido no sistema será dado por:

$$Q_l = A_p \dot{x} + C_{tp} \Delta P + \frac{V_t}{4\beta_e} \Delta \dot{P} \quad (2.3)$$

Agora, tomando a válvula como volume de controle, é possível realizar a mesma análise do fluxo a partir da equação da continuidade. Para isto, considera-se que a pressão de retorno do fluido P_0 como sendo nula ($P_0 \approx 0$), haja vista que esta é muito inferior às demais pressões que se desenvolvem no sistema. Desta forma, sendo C_d o coeficiente de descarga, ρ a densidade do fluido, w o gradiente de área do orifício da válvula, x_{sp} o deslocamento real do carretel da válvula, \bar{x}_{sp} o

deslocamento efetivo do mesmo carretel e P_s a pressão fornecida ao sistema, o escoamento do fluido pode também ser descrito como se segue:

$$Q_l = C_d w \bar{x}_{sp} \sqrt{\frac{1}{\rho} (P_s - \text{sgn}(\bar{x}_{sp}) \Delta P)} \quad (2.4)$$

A função $\text{sgn}(\cdot)$ pode ser definida da seguinte maneira:

$$\text{sgn}(z) = \begin{cases} -1, & z < 0 \\ 0, & z = 0 \\ 1, & z > 0 \end{cases} \quad (2.5)$$

A resposta da válvula é, por hipótese, rápida o suficiente para que o seu efeito na dinâmica do sistema possa ser desconsiderado. Desta forma, admite-se que o deslocamento do carretel é proporcional ao esforço de controle u aplicado. Assim, qualquer valor de u influenciará diretamente no fluxo de fluido hidráulico.

Uma das não linearidades presentes no sistema em estudo é o bloqueio do orifício de passagem do fluido durante a passagem do carretel. Esta não linearidade é do tipo zona morta. Fisicamente, a presença de uma zona morta implica que, para pequenos valores do esforço de controle u , o carretel da válvula não se movimenta efetivamente, isto é, o movimento do carretel não gera variação no fluxo do fluido hidráulico. Sendo k_v o ganho da válvula e δ_l e δ_r os valores mínimos de u a partir dos quais há movimento efetivo do carretel da válvula, a zona morta é representada matematicamente da seguinte forma:

$$\bar{x}_{sp} = \begin{cases} k_v(u(t) - \delta_l), & u(t) \leq 0 \\ 0, & \delta_l < u(t) < \delta_r \\ k_v(u(t) - \delta_r), & u(t) > \delta_r \end{cases} \quad (2.6)$$

O comportamento da zona morta também pode ser representado como em Bessa, Dutra e Kreuzer (2010):

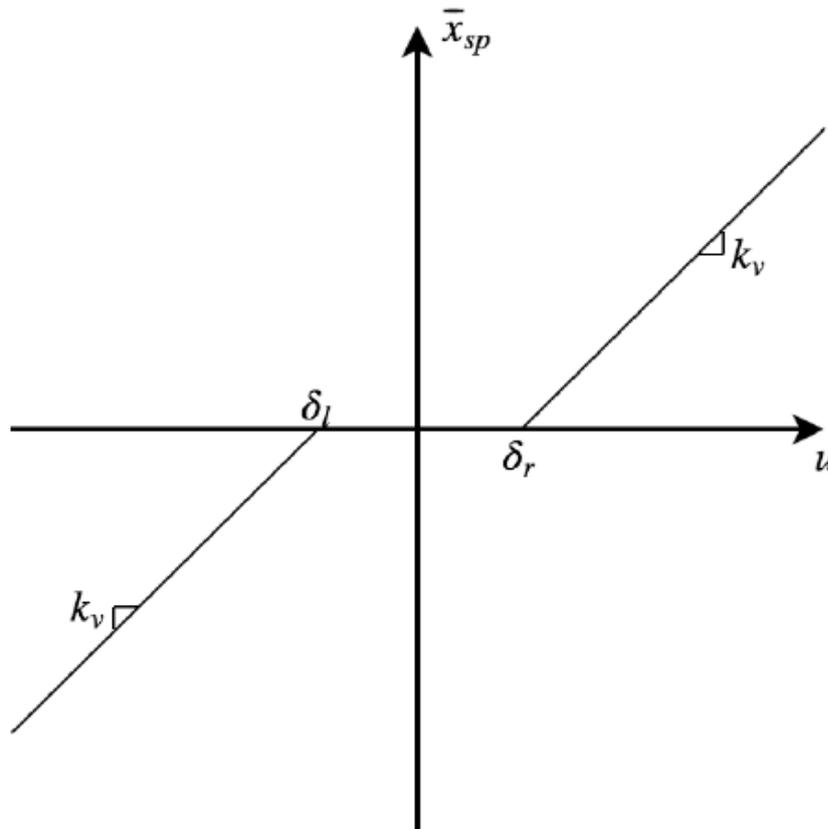
$$\bar{x}_{sp} = k_v[u(t) - d(u)] \quad (2.7)$$

Onde $d(u)$ é uma variável utilizada para auxiliar na representação matemática e pode ser descrita da seguinte forma:

$$d(u) = \begin{cases} \delta_l, & u(t) < \delta_l \\ u_t, & \delta_l < u(t) < \delta_r \\ \delta_r, & u(t) \geq \delta_r \end{cases} \quad (2.8)$$

A fig. (4) ilustra o comportamento do deslocamento efetivo do carretel, \bar{x}_{sp} , em função do esforço de controle u . Quando os valores de u estão entre δ_l e δ_r , não há deslocamento efetivo do carretel. Quando há movimento efetivo do carretel, este é proporcional ao esforço de controle u e ao ganho da válvula k_v .

Figura 4 – Não linearidade do tipo zona morta.



Fonte: Azevedo (2016).

A partir da combinação das equações (2.2), (2.3), (2.4), (2.7) e (2.8), é possível obter a representação da dinâmica do sistema, dada pela equação diferencial de terceira ordem abaixo:

$$\ddot{x} = -a^T X + bu - bd \quad (2.9)$$

Donde $X = [x \ \dot{x} \ \ddot{x}]^T$ é o vetor dos estados do sistema e $a = [a_0 \ a_1 \ a_2]^T$ é o vetor que contém as informações físicas do sistema eletro-hidráulico em estudo. Os parâmetros de a dão-se como a seguir:

$$a_0 = \frac{4\beta_e C_{tp} K_s}{V_t M_t} \quad (2.10)$$

$$a_1 = \frac{K_s}{M_t} + \frac{4\beta_e A_p^2}{V_t M_t} + \frac{4\beta_e C_{tp} B_t}{V_t M_t} \quad (2.11)$$

$$a_2 = \frac{B_t}{M_t} + \frac{4\beta_e C_{tp}}{V_t} \quad (2.12)$$

Além disso, o ganho b do controlador, presente na eq. (2.9), é calculado da seguinte maneira:

$$b = \frac{4\beta_e A_p C_d w k_v}{V_t M_t} \sqrt{\frac{1}{\rho} \left[P_s - \frac{\text{sgn}(u)(M_t \ddot{x} + B_t \dot{x} + K_s x)}{A_p} \right]} \quad (2.13)$$

Técnicas de controle serão apresentadas baseadas no modelo mostrado e seu desempenho, frente as não linearidades do sistema, será avaliado.

3. CONTROLE NÃO LINEAR

Nas atuais aplicações de engenharia, procura-se cada vez mais uma maior fidelidade dos modelos utilizados em relação aos sistemas reais. Tendo isso em vista, a utilização de modelos lineares simples torna-se obsoleta haja vista sua operabilidade limitada por diversas condições e a necessidade de uma precisão de operação cada vez maior. Assim, opta-se pela utilização de modelos não-lineares pois estes representam muito mais fielmente o comportamento dos sistemas reais. Entretanto, cumpre salientar que este tipo de abordagem requer um grande repertório de ferramentas para análise não linear.

Dentre os benefícios proporcionados pela utilização de modelos não lineares está o aumento da faixa de operação haja vista a redução nas restrições que seriam feitas para que o sistema não perdesse desempenho caso o modelo utilizado fosse linear. De acordo com Azevedo (2016), não linearidades do tipo histerese, atrito de Coulomb, saturação e zona morta são facilmente contornadas pela utilização deste tipo de abordagem.

Como mencionado por Bessa (2005), outra característica importante deste tipo de abordagem, é a possibilidade de utilização de um aparato de baixo custo, o que é um dos focos deste trabalho. Enquanto controladores lineares requerem uma aparelhagem que apresente um excelente comportamento linear em toda sua faixa operacional, uma estratégia de controle não linear, quando bem escolhida, pode ser aliada a um aparato de baixo custo sem que a performance do controlador seja comprometida. Neste trabalho, propõe-se a utilização do método de linearização por realimentação através de um Arduino o qual possui baixo custo em relação às placas de aquisição convencionais utilizadas industrialmente.

3.1. Linearização por realimentação

Para controle do sistema, foi escolhido um controlador que segue o método de linearização por realimentação. Este método é utilizado em controle de sistemas não lineares e consiste na aplicação de um esforço de controle o qual anulará, algebricamente, as não linearidades presentes no sistema. Considere-se um sistema não linear da seguinte forma:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \\ \vdots \\ \dot{x}_n = f(X, t) + b(X, t)u(t) \\ y = h(x) \end{cases} \quad (3.1)$$

Ou, de maneira equivalente:

$$\begin{cases} \dot{x}^{(n)} = f(X, t) + b(X, t)u(t) \\ y = x \end{cases} \quad (3.2)$$

Sendo $X = [x \ \dot{x} \ \ddot{x} \ \dots \ x^{(n-1)}]^\top = [x_1, x_2, \dots, x_n]^\top$ o vetor dos estados do dado sistema, t o tempo, f , b e h funções não lineares, u e y a entrada e saída dos sistema, respectivamente. O método propõe a aplicação de um esforço de controle u o qual, algebricamente, anule as não linearidades do sistema de modo que o sistema passe a se comportar como linear e em malha fechada.

Para isso, deve ser determinado um vetor de valores desejados X_d e um outro vetor para os erros do sistema \tilde{X} tais quais:

$$X_d = [x_d \ \dot{x}_d \ \ddot{x}_d \ \dots \ x_d^{(n-1)}]^\top \quad (3.3)$$

$$\tilde{X} = X - X_d \quad (3.4)$$

O esforço de controle u aplicado ao sistema deve fazer com que $X \rightarrow X_d$, ou seja $\tilde{X} \rightarrow 0$, à medida que $t \rightarrow \infty$. Assumindo que X é mensurável, que f e b são conhecidas e que b é não nulo, o esforço de controle u se apresenta na forma:

$$u = b^{-1}(-f + x_d^{(n)} - k_0\tilde{x} - k_1\dot{\tilde{x}} - \dots - k_{n-1}\tilde{x}^{(n-1)}) \quad (3.5)$$

Com a condição de que os coeficientes k_i ($i = 0, 2, \dots, n - 1$) façam com que o polinômio $p^n + k_{n-1}p^{n-1} + \dots + k_0$ seja um polinômio de Hurwitz, isto é, todas as raízes deste polinômio devem estar no semi-plano esquerdo do plano complexo,

admite-se que a dinâmica do sistema é exponencialmente estável e assim, assegura-se que $\tilde{X} \rightarrow 0$ à medida que $t \rightarrow \infty$.

Se a lei de controle u da eq. (3.5) for inserida na eq. (3.2) do sistema dinâmico não-linear, o comportamento em malha fechada do sistema, isto é, sua convergência é descrita segundo a eq. (3.6):

$$\tilde{x}^{(n)} + k_{n-1}\tilde{x}^{(n-1)} + \dots + k_1\dot{\tilde{x}} + k_0\tilde{x} = 0 \quad (3.6)$$

Assim, a lei que define o esforço de controle para o método de linearização por realimentação é redefinida da seguinte forma:

$$u = b^{-1}(-f + x_d^{(n)} - K^T \tilde{X}) \quad (3.7)$$

Onde $K = [c_0\lambda^n, c_1\lambda^{n-1}, \dots, c_{n-1}\lambda]$ é o vetor composto pelos coeficientes do polinômio de Hurwitz e λ é uma constante, sempre positiva, a qual determina a velocidade de convergência do controlador e junto com c_i ($i = 0, 1, \dots, n - 1$), é utilizada para a determinação dos coeficientes do polinômio. A obtenção de c_i é feita através do binômio de Newton:

$$c_i = \binom{n}{i} = \frac{n!}{i!(n-i)!} \quad (3.8)$$

Cumprе salientar que esta técnica de controle costuma ser utilizada quando tanto os parâmetros físicos quanto as não linearidades do sistema são bem conhecidos. Isso faz com que estas incertezas possam ser anuladas com facilidade. Na prática, o não conhecimento pleno das propriedades do sistema compromete o desempenho do controlador.

4. CONTROLADOR

Uma boa performance desta abordagem depende muito do quão bem o sistema está modelado. Logo, quanto maior o número de aproximações e simplificações, menos fiel o modelo será ao sistema real e menos eficiente será o controlador. A tabela (1) mostra os valores utilizados para cada parâmetro que compõe o modelo do sistema.

Tabela 1 – Parâmetros do sistema.

PARÂMETROS DO SISTEMA	
P_S	$7 \times 10^6 Pa$
ρ	$850 kg/m^3$
C_d	0,6
w	$2,5 \times 10^{-2} m$
A_p	$3 \times 10^{-4} m^2$
β_e	$7 \times 10^8 Pa$
V_t	$6 \times 10^{-5} m^3$
M_t	250 kg
B_t	100 Ns/m
K_v	$30 \times 10^{-6} m/V$
K_s	0 N/m
C_{tp}	0 m ³ /sPa

Os dados acima foram retirados de Azevedo (2016), à exceção do ganho da válvula K_v , o qual teve de ser adaptado para que o controle ocorresse de forma satisfatória, da constante de vazamento C_{tp} , a qual foi considerada nula por assumir-se que não há vazamento no sistema, e da constante da mola K_s , a qual não foi utilizada neste trabalho, e por isso, considerada nula.

É importante frisar que a massa de 250 kg foi escolhida visando dimensionar o controlador para que um controle satisfatório fosse efetuado quando uma carga desta ordem de grandeza fosse imposta ao mesmo

O sistema estudado possui a não linearidade do tipo zona morta, oriunda da sobreposição do carretel da válvula sobre o orifício de passagem do fluido quando o esforço de controle u é muito pequeno.

A partir da combinação das equações (2.9), (2.10), (2.11), (2.12) e (2.13), o modelo do esforço de controle u a ser utilizado torna-se:

$$u = b^{-1}(a^T X + \ddot{x}_d - 3\lambda\ddot{x} - 3\lambda^2\dot{x} - \lambda^3x) \quad (4.1)$$

As propriedades do sistema estão contidas no vetor $a = [a_0 \ a_1 \ a_2]^T$ enquanto que $\tilde{x} = x - x_d$ é o erro do sistema. A constante λ é sempre positiva e determina a velocidade com que o erro converge para 0.

Tomando como base os valores da tabela (1), onde $K_s = 0$ e $C_{tp} = 0$, as componentes do vetor a se apresentam de acordo com as equações (4.2), (4.3) e (4.4).

$$a_0 = 0 \quad (4.2)$$

$$a_1 = \frac{4\beta_e A_p^2}{V_t M_t} \quad (4.3)$$

$$a_2 = \frac{B_t}{M_t} \quad (4.4)$$

A tabela (2) mostra os valores de a_0 , a_1 , a_2 e λ para o controlador utilizado.

Tabela 2 – Parâmetros do controlador.

PARÂMETROS DO CONTROLADOR	
a_0	0
a_1	16800
a_2	0,4
λ	10

Ao compararmos as equações (3.7) e (4.1), nota-se que $-f = a^T X$, assim:

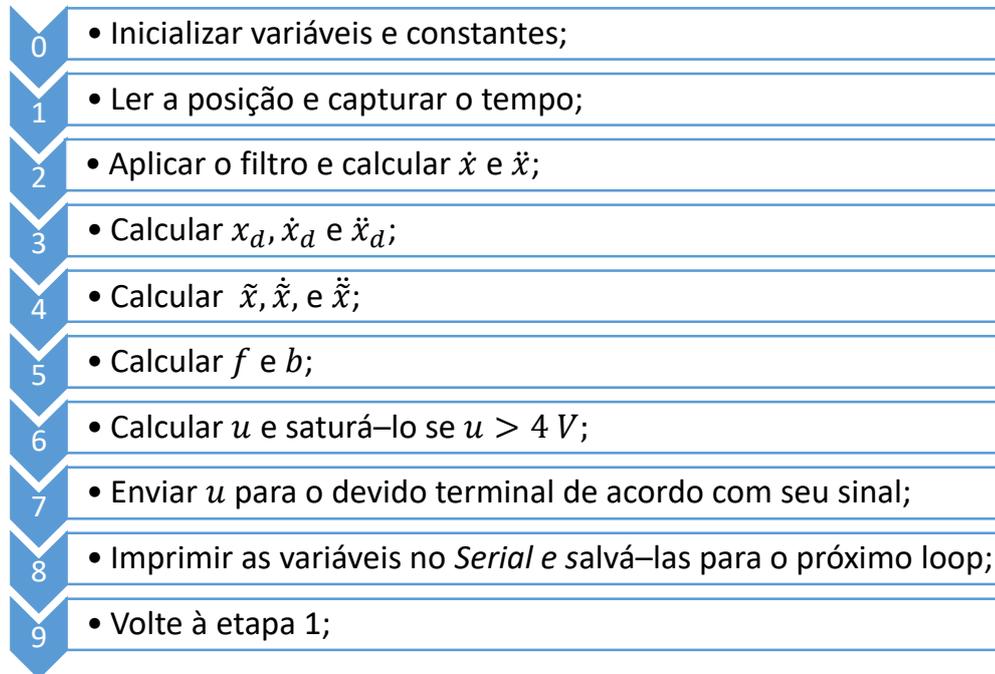
$$f = -(a_0 x + a_1 \dot{x} + a_2 \ddot{x}) \quad (4.5)$$

$$u = b^{-1}(-f + \ddot{x}_d - 3\lambda\ddot{x} - 3\lambda^2\dot{x} - \lambda^3x) \quad (4.6)$$

4.1. Algoritmo

A fig. (5) mostra um fluxograma representativo do algoritmo implementado.

Figura 5 – Fluxograma representativo do algoritmo implementado.



Fonte: Autoria própria.

As variáveis e constantes são iniciadas na etapa numerada como 0, esta etapa só é executada antes do primeiro *loop*. A seguir, a posição x é medida e o instante t em que a medição é feita é gravado. Feito isso, o filtro é aplicado para que o sinal não apresente oscilações bruscas e comprometa assim o desempenho do controlador. Aplicado o filtro, as derivadas da posição, filtrada, são calculadas.

A seguir, o controlador calcula x_d e suas respectivas derivadas. Em seguida, a partir dos valores obtido de x , x_d e suas respectivas derivadas, é calculado o erro \tilde{x} e suas derivadas.

A partir do erro, o controlador calcula f e b e, a partir destes, calcula u de acordo com a eq. (4.6). O controlador deve então limitar u a $\pm 4 V$ caso $|u| > 4 V$. Em seguida, a partir do sinal matemático de u , este é enviado ao respectivo solenoide da eletroválvula. Após isso, todas estas variáveis são impressas no *Serial* do Arduino e os valores de t , x , \dot{x} e \ddot{x} são salvos para os cálculos de velocidade e aceleração do próximo *loop*. Após isso, o controlador deve voltar à etapa 1 e reiniciar o processo a partir dali.

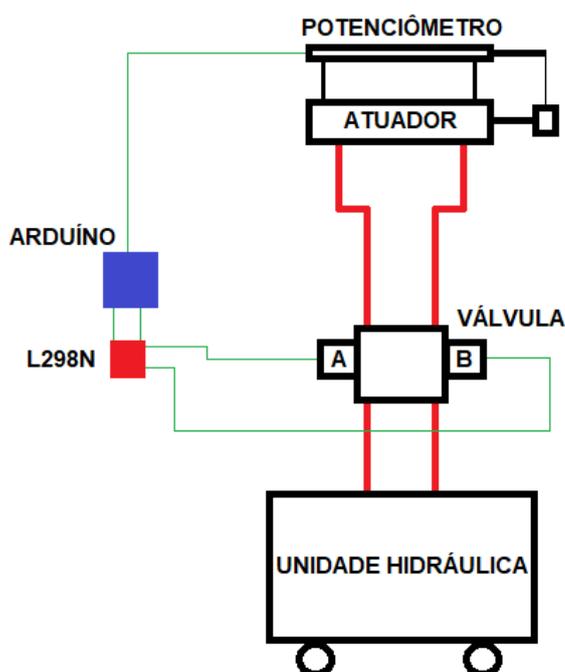
5. METODOLOGIA

Para a realização do experimento, a bancada eletro-hidráulica presente no Laboratório de Manufatura da Universidade Federal do Rio Grande do Norte foi utilizada junto com alguns itens pessoais. Todos os itens serão descritos detalhadamente mais a seguir.

O curso do atuador, de 200 *mm*, foi limitado aos 160 *mm* centrais para evitar imprecisões nas extremidades. Além disso, cada tipo de trajetória estudada neste trabalho foi executada por cerca de 120 *s*. Os componentes utilizados neste trabalho, e que serão detalhados mais a seguir foram 1 atuador eletro-hidráulico de dupla ação, 1 potenciômetro linear, 1 fonte de 24 V, 1 servo-válvula proporcional 4/3 vias, 1 unidade de energia hidráulica, 1 Arduino *UNO R3*, 1 driver L298N e 1 potenciômetro convencional.

Sendo a maioria destes componentes pertencentes à UFRN, à exceção dos três últimos, de propriedade pessoal. Os quatro primeiros itens da lista acima são parte da linha didática FESTO. Cumpre salientar ainda que todo um aparato de cabeamento foi utilizado nas conexões entre os itens acima listados. A fig. (6) ilustra o esquema de montagem da bancada.

Figura 6 – Esquema de montagem da bancada experimental.



Fonte: Autoria própria.

Como será visto a seguir, cinco diferentes trajetórias foram impostas ao controlador: duas trajetórias definidas manualmente, como auxílio de um potenciômetro convencional e as demais foram definidas de acordo com funções temporais através do micro controlador.

Após a definição trajetória, o micro controlador calcula o erro \tilde{x} de acordo com a posição x medida pelo potenciômetro linear. A partir do erro, o mesmo micro controlador calcula o esforço de controle u necessário para que o atuador se movimente até a posição desejada. O sinal é enviado para o driver L298N, a qual amplifica a corrente e a manda para o respectivo terminal da eletroválvula: A ou B na fig. (6). A válvula varia a passagem do fluido proporcionalmente ao sinal enviado e o atuador se movimenta. A partir deste ponto, o processo passa a se repetir.

Os dados foram obtidos através da porta serial do Arduino, a qual, para cada *loop*, salvava os dados de tempo, posição, erro e esforço de controle. Após a aquisição dos dados, foi calculado o erro médio absoluto de cada trajetória. Isto foi feito com o intuito de comparar o desempenho do controlador mediante os diferentes tipos de trajetória. O erro médio absoluto de cada trajetória foi calculado segundo a equação (5.1).

$$\bar{\tilde{x}}_{abs} = \frac{1}{N} \sum_{i=1}^N |\tilde{x}| \quad (5.1)$$

Onde N é quantidade de amostras obtidas em cada experimento.

5.1. Bancada

A seguir serão descritos os componentes da bancada experimental utilizada.

5.1.1. Atuador eletro–hidráulico

O atuador utilizado é um cilindro hidráulico de dupla ação. O pistão se movimenta a partir da pressurização das suas câmaras internas com o fluido de trabalho.

O item pertence à bancada didática FESTO e de acordo com o fabricante, o mesmo possui 40 *mm* de diâmetro e um curso de 200 *mm*. Além disso, a pressão máxima suportada pelo cilindro é de 10 *MPa* enquanto que a pressão de trabalho é 6 *MPa*. A fig. (7) mostra o componente.

Figura 7 – Atuador hidráulico de dupla ação.



Fonte: Azevedo (2016).

5.1.2. Potenciômetro linear

Trata-se do sensor de posição utilizado pelo controlador. Este componente também pertence à linha didática FESTO e possui uma faixa de medição de 200 *mm* sendo necessário que este seja acoplado ao cilindro de modo que sua extremidade, móvel, seja fixada próxima à cabeça do atuador e movimente-se junto com a haste do mesmo.

Para que o sensor funcione, o mesmo deve ser alimentado por uma fonte de tensão. Desta forma, o mesmo passa a enviar um sinal de saída que varia de 0 a 10 *V*. Sendo o sinal mínimo quando o mesmo medir 0 *mm* e o sinal máximo quando o mesmo medir 200 *mm*. A calibração deste item será detalhada mais à frente neste trabalho. A fig. (8) mostra o componente:

Figura 8 – Potenciômetro linear.

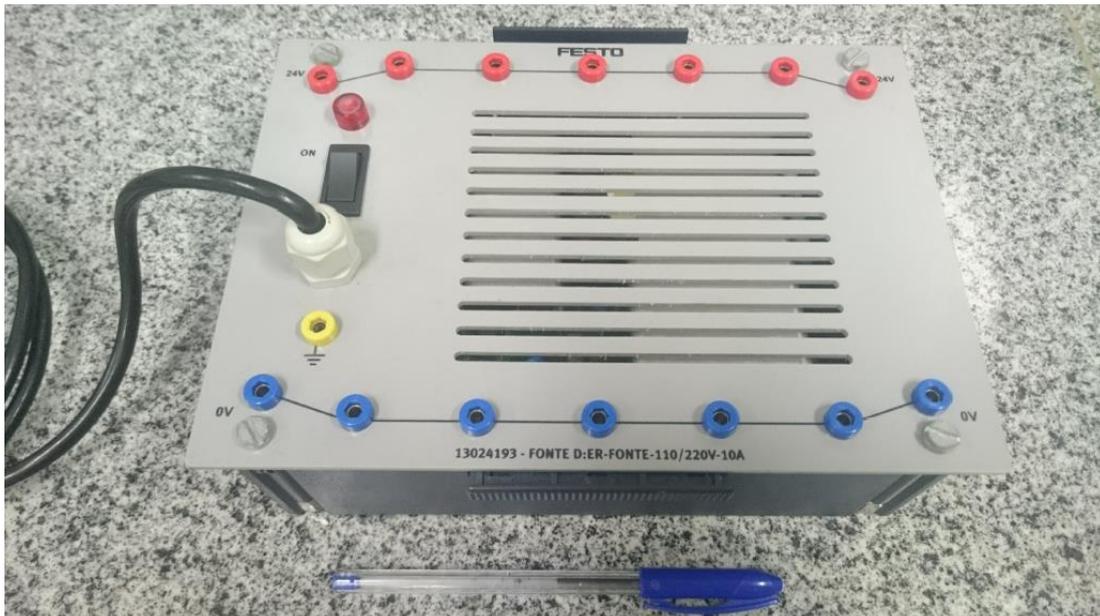


Fonte: www.festo-didactic.com

5.1.3. Fonte de tensão (24 V)

A fonte de tensão utilizada neste trabalho é do tipo CC e fornece uma tensão de alimentação de 24 V em um barramento e 0 V em outro. A mesma é utilizada para alimentar a válvula e o potenciômetro linear. Este componente também faz parte da linha didática FESTO e pode ser ligado em tensões de entrada de 110 a 220 V. A fig. (9) mostra o componente.

Figura 9 – Fonte de tensão CC 24 V.



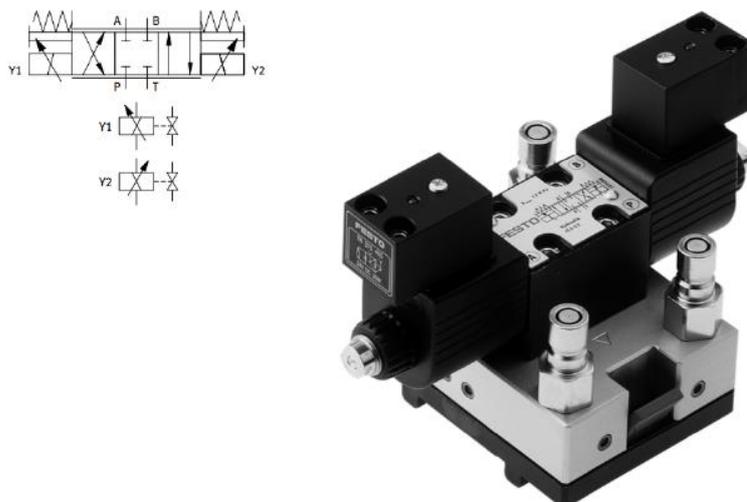
Fonte: Azevedo (2016).

5.1.4. Válvula proporcional 4/3 vias

A válvula utilizada possui 4 vias e 3 posições e possui 4 conectores de engate rápido e mais quatro conectores para seus solenoides. O controle do fluxo se dá a partir da variação da posição do carretel da válvula. Este carretel movimenta-se proporcionalmente à corrente imposta aos solenoides. O uso desse componente possibilita que haja variação na velocidade do movimento do atuador.

Este componente também faz parte da bancada didática da FESTO e de acordo com seu fabricante, a pressão de operação da mesma é de 6 MPa enquanto que a tensão de alimentação é de 24 V CC e a corrente nominal é de 800 mA. Além disso, também é informado que a perda por vazamento da mesma é de 0,01 l/min. A fig. (10) mostra o componente.

Figura 10 – Eletroválvula.



Fonte: www.festo-didactic.com

5.1.5. Unidade de energia hidráulica

A unidade de energia hidráulica utilizada neste trabalho possui um tanque de 40 l e trabalha com óleo ISO VG 46. Sua função é basicamente bombear o fluido a fim de que este atinja a pressão de trabalho.

A vazão fornecida pela bomba é de 10 l/min enquanto que suas pressões máxima e de trabalho são, respectivamente 10 e 6 MPa. Uma alimentação de 220 V e 15 A é necessária para seu funcionamento. A fig. (11) mostra o componente.

Figura 11 – Fonte de energia hidráulica.



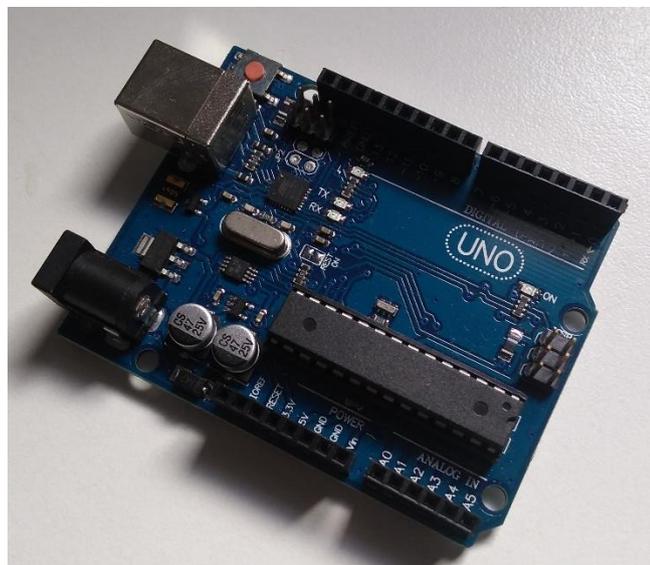
Fonte: Azevedo (2016).

5.1.6. Arduino *UNO R3*

O micro controlador utilizado para compilar o algoritmo do controlador foi uma placa do tipo Arduino *UNO R3*. De acordo com o site do fabricante, este modelo é baseado no micro controlador ATmega328P. Ainda de acordo com o fabricante, este modelo de Arduino possui 14 portas de comunicação (entrada ou saída), das quais 6 podem ser utilizadas como saídas do tipo PWM (*Pulse Width Modulation*), isto é, estas saídas podem emitir um sinal de saída de intensidade variável, 6 entradas analógicas (muito utilizadas em leituras), um cristal de quartzo de 16 MHz e conexão USB. O mesmo pode ser alimentado via USB ou via uma fonte independente de tensão.

Em questão de sinais, tanto seu envio quanto sua leitura possuem uma faixa que varia de 0 a 5 V. Entretanto, o Arduino interpreta esse *range* como sinais de unidades digitais. Os sinais de leitura variam de 0 a 1023 (10 *bits*) unidades enquanto que os sinais enviados variam de 0 a 255 unidades (8 *bits*). Assim, a placa possui uma resolução de leitura mais sensível que a de envio de sinais. A figura (12) mostra o item utilizado neste trabalho.

Figura 12 – Arduino *UNO R3* utilizado neste trabalho.



Fonte: Autoria própria.

A placa utilizada por Azevedo (2016) foi uma placa do modelo *USB-4711A* da fabricante *Advantech*. Em consulta feita em dezembro de 2018 site oficial da *Advantech*, uma placa do modelo *USB-4711A* estava custando cerca de R\$ 3.820,00 (três mil oitocentos e vinte reais) enquanto que, no mesmo período, um Arduino do

modelo *UNO R3* como o utilizado neste trabalho estava custando cerca de R\$ 40,00 (quarenta reais) no site *Baú da eletrônica*.

O custo de todo o aparato adquirido para a realização deste trabalho não ultrapassou R\$ 250,00 (duzentos e cinquenta reais). Cumpre salientar ainda que a placa *USB-4711A* requer para o seu funcionamento, um cartão de amplificação de sinal como o da linha *Festo* mencionado por Azevedo (2016) que custa em torno de R\$ 10.000,00 (dez mil reais) e um computador, o que em dezembro de 2018 custa cerca de R\$ 2.000,00 (dois mil reais).

A partir dos valores mencionados, percebe-se que o custo foi reduzido de um valor em torno de R\$ 15.820,00 (quinze mil oitocentos e vinte reais) para menos de R\$ 250,00 (duzentos e cinquenta reais), isto é, uma redução de 15,28 vezes.

A tabela (3) mostra uma comparação entre os dados técnicos do Arduino *UNO R3* e da *Advantech USB-4711A*. Os dados foram retirados dos respectivos sites oficiais dos equipamentos.

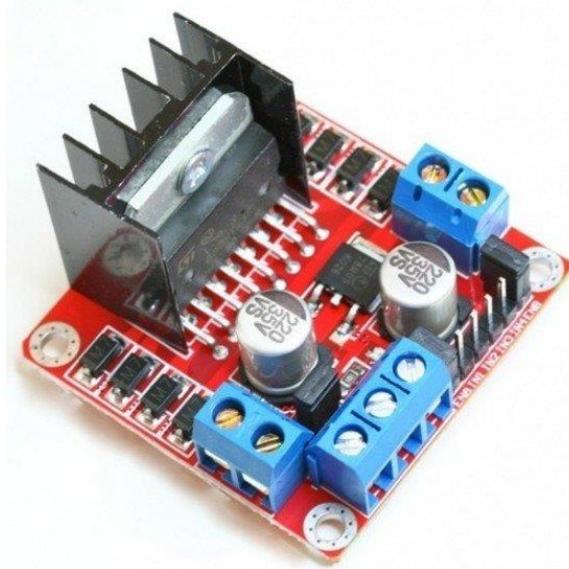
Tabela 3 – Comparação entre Arduino *UNO R3* e *Advantech USB-4711A*.

CARACTERÍSTICA	ARDINO UNO R3	ADVANTECH USB-4711A-AE
Resolução (IN)	4,89 mV (10 bits)	2,44 mV (12 bits)
Resolução (OUT)	19,61 mV (8 bits)	2,44 mV (12 bits)
Canais (IN)	6	8
Canais (OUT)	14 (6 PWM)	8
Baud rate	115,2 kHz	150 kHz
Tensão do sinal	$\pm 0 - 5 V$	$\pm 0 - 10 V$
Corrente	40 mA	5 mA

5.1.7. Driver L298N

Trata-se do circuito integrado (CI) utilizado com o intuito de amplificar a corrente do sinal enviado pelo Arduino. É popularmente conhecido como ponte H. De acordo com o fabricante (STMicroelectronics), sua temperatura de operação varia de 40 a 150 °C e o mesmo possui proteção contra superaquecimento, pode ser alimentado por uma fonte de até 46 V e fornece uma corrente de até 4 A. A fig. (13) mostra um driver L298N.

Figura 13 – L298N.



Fonte: <https://www.eletrogate.com>

5.1.8. Potenciômetro convencional

Trata-se do potenciômetro utilizado para ajuste das trajetórias manuais. Quando seu sinal fosse mínimo, o x_d deveria corresponder ao limite inferior, 20 *mm*, enquanto que, quando seu sinal fosse máximo, o x_d deveria corresponder ao limite superior 180 *mm*. Possui uma resistência de 10 *kΩ* e um diâmetro de base de 15 *mm*. A fig. (14) mostra um potenciômetro convencional como o utilizado.

Figura 14 – Potenciômetro convencional.



Fonte: <https://natalmakers.lojaintegrada.com.br/>

5.2. Calibrações

Para que o experimento fosse realizado corretamente, alguns componentes do sistema foram calibrados a fim de que se conhecesse o comportamento dos mesmos. Assim, tanto o potenciômetro linear quanto os terminais da eletroválvula foram calibrados.

O potenciômetro foi calibrado com o intuito de definir uma relação entre a posição medida e o sinal enviado pelo mesmo. Já os terminais foram calibrados a fim de que se soubesse o nível de intensidade da corrente que estava sendo enviada aos mesmos.

5.2.1. Calibração do potenciômetro linear

Para a calibração do potenciômetro foi usado um traçador de altura da marca Digimess presente no Laboratório de Metrologia da Universidade Federal do Rio Grande do Norte. O potenciômetro foi devidamente posicionado no mesmo e as medições foram feitas a cada 5 mm, de 0 a 200 mm (curso máximo do sensor). A fig. (15) mostra a montagem feita para a calibração.

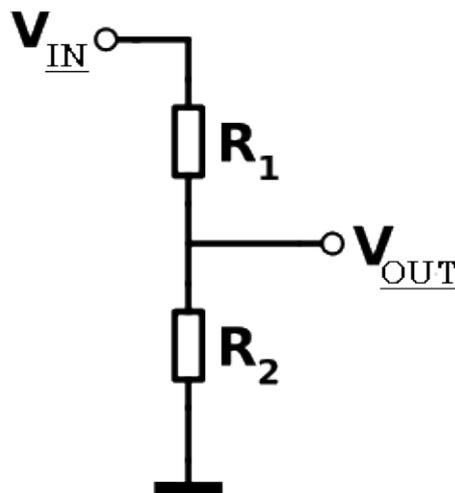
Figura 15 – Esquema montado para calibração do potenciômetro linear.



Fonte: Autoria própria.

O sinal máximo emitido pelo potenciômetro, quando o mesmo está na posição de 200 mm, é de 10 V, entretanto, de acordo com a literatura de referência do Arduino, o mesmo pode receber sinais de até 5 V. Para contornar isso, foi colocado um circuito divisor de tensão entre o sensor e o Arduino para que a tensão fosse reduzida pela metade. Assim, quando o sinal emitido pelo sensor for 10 V, a placa recebe um sinal de 5 V. A fig. (16) mostra como deve ser montado um circuito deste tipo.

Figura 16 – Esquema de circuito divisor de tensão.



Fonte: <http://nerdeletrico.blogspot.com/>

Onde V_{in} é a tensão vinda do sensor, V_{out} é a tensão que vai para o microcontrolador e R_1 e R_2 são as resistências utilizadas. Após a resistência R_2 , o circuito é ligado à tensão de referência (0 V). A eq. (5.2) mostra a relação entre V_{in} e V_{out} .

$$V_{out} = \frac{R_2}{R_1 + R_2} V_{in} \quad (5.2)$$

Ao observar-se a eq. (5.1), nota-se que a tensão de saída é diretamente proporcional à tensão de entrada e à razão entre R_2 e a soma entre R_1 e R_2 . Assim, como o sinal de entrada vindo do sensor é de até 10 V e deseja-se um sinal de até 5 V, $\frac{R_2}{R_1 + R_2}$ deve ser 0,5, logo $R_1 = R_2 = R$. A resistência R utilizada foi de 10 k Ω .

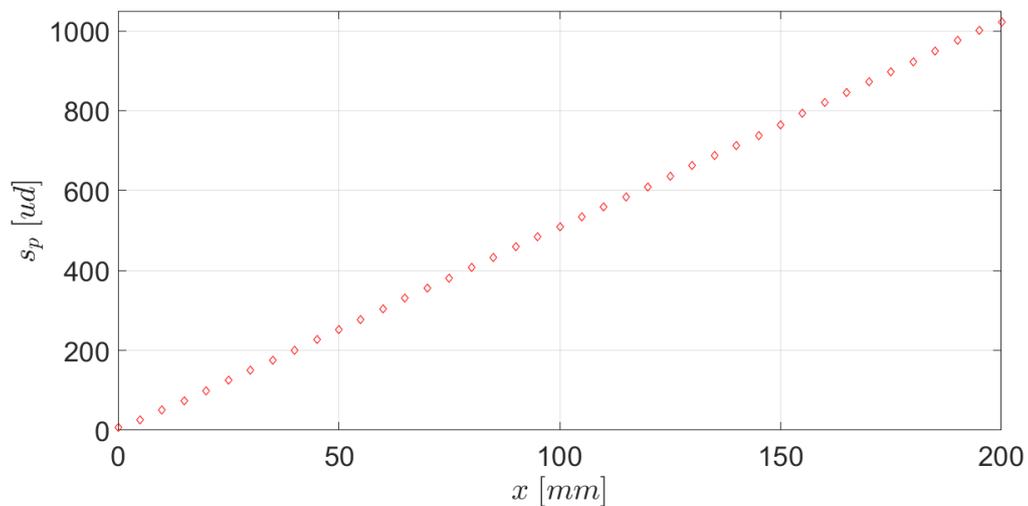
Outra característica que deve ser ressaltada é que o Arduino possui um conversor analógico-digital o qual converte, proporcionalmente, os valores de tensão recebidos (analógicos) de 0 a 5 V para unidades inteiras (digitais) entre 0 e 1023.

Sabendo isso, a resolução δ do micro controlador pode ser estimada da seguinte forma:

$$\delta = \frac{5000 \text{ mV}}{1024 \text{ bits}} \approx 4,89 \frac{\text{mV}}{\text{unidade}}$$

A partir dos valores obtidos, gerou-se o gráfico da fig. (17), o qual aproxima-se bastante de uma reta.

Figura 17 – Curva de calibração do potenciômetro linear.



Fonte: Autoria própria.

Os dados foram numericamente interpolados a fim de que se achasse a equação da reta que melhor se adequasse a estes. A função que melhor se adequou aos dados foi:

$$s_p(x) = -2,87456 + 5,13167x \quad (5.3)$$

Onde $s_p(x)$ é o sinal enviado pelo sensor e lido pelo Arduino, medido em unidades digitais e x é a posição medida em mm . Esta relação será a responsável pela medição da posição durante o experimento.

5.2.2. Calibração dos terminais da eletroválvula

De acordo com o fabricante, a corrente máxima que pode ser enviada aos solenoides da eletroválvula é de 800 mA . Para medição da corrente, foi utilizado um multímetro digital da marca Minipa presente no Laboratório de Manufatura da UFRN. A fig. (18) mostra o instrumento utilizado.

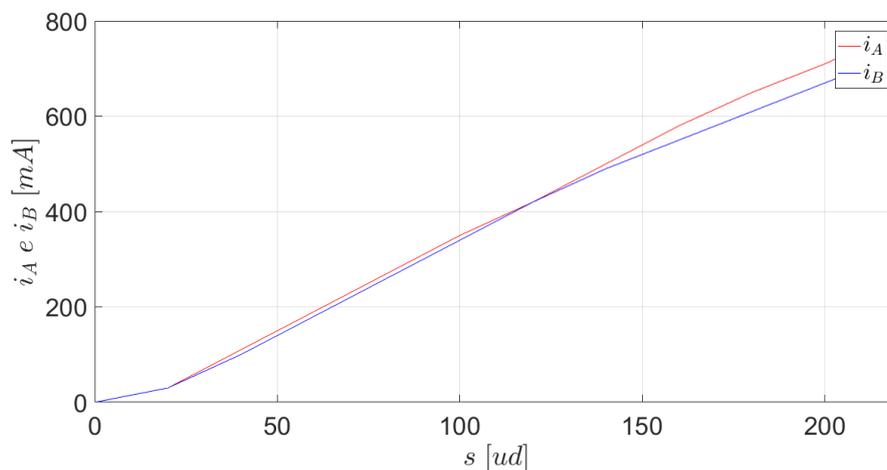
Figura 18 – Multímetro utilizado.



Fonte: Autoria própria.

Após os ensaios, os dados obtidos indicaram um comportamento linear da corrente que chegava à válvula em função do sinal enviado pelo Arduino. A fig. (19) mostra o comportamento da corrente em função do sinal enviado para os dois terminais da eletroválvula.

Figura 19 – Curvas de calibração das correntes nos solenoides da válvula.



Fonte: Autoria própria.

Além disso, durante os ensaios de calibração, constatou-se que o sinal máximo que o Arduino enviava (5 V ou 255 *unidades*) gerava uma corrente de cerca de 1 A. Assim, tomando como base o comportamento linear apresentado pela corrente nos dois terminais, para que a corrente fosse limitada à faixa recomendada pelo fabricante, o sinal emitido pelo micro controlador foi limitado a 80 % da sua capacidade total, isto é, 4 V ou 204 *unidades*. O controlador foi programado a saturar o esforço de controle calculado pelo mesmo sempre que este ultrapassasse este valor limite.

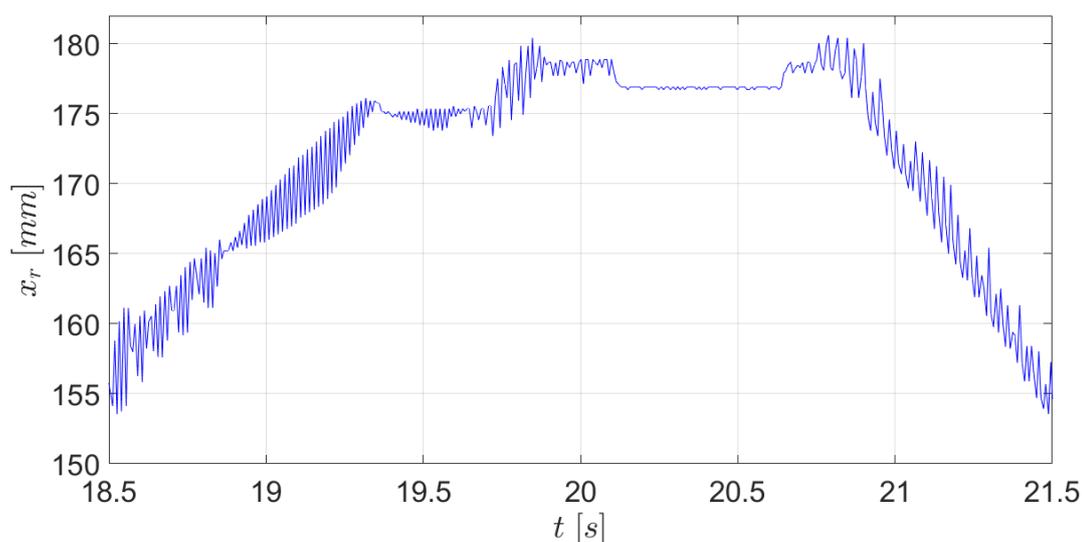
5.3. Tratamento de dados

A baixa qualidade de alguns componentes e conexões do sistema, como é o caso dos fios conectores (*jumpers*) e suas conexões e a utilização de componentes como as *protoboards*, implicam em um alto nível de ruídos durante a captação dos dados, comprometendo assim, o desempenho do controlador.

Outras fontes de ruídos são a baixa sensibilidade de leitura do Arduino, representada por sua resolução δ , a vibração à qual o sistema é submetido durante o seu funcionamento, a falta de conectores adequados, o que levou à utilização de conectores do tipo “jacaré” e etc.

A fig. (20) mostra o comportamento do sinal da posição medida pelo sensor e lida pelo Arduino, x_r , para a trajetória senoidal.

Figura 20 – Comportamento da posição medida x_r para a Trajetória Senoidal.



Fonte: Autoria própria.

A fig. (20) mostra que o sinal de leitura oscila até cerca de 5 mm entre duas leituras consecutivas. Parte desta oscilação pode ser atribuída à já mencionada baixa resolução de leitura do Arduino, o que faz com que pequenas variações no sinal de leitura não possam ser identificadas pelo micro controlador. Assim, apenas variações de tensão maiores que a resolução do Arduino são identificadas.

Tendo em vista que as demais variáveis utilizadas pelo controlador, como o erro $\tilde{x} = x - x_d$, a velocidade $\frac{\Delta x}{\Delta t}$ e outras, são obtidas através da posição, a utilização desses dados sem uma filtragem adequada faria com que a performance do controlador fosse muito aquém do que poderia ser.

Para contornar isso, optou-se pela utilização de um filtro digital a fim de que se suavizasse as oscilações do sinal da posição. O filtro escolhido foi o do tipo de médias móveis.

De acordo com Smith (1999), este filtro é muito utilizado em tratamento de dados em virtude da facilidade de entendimento e utilização, principalmente na filtragem de dados no domínio do tempo.

A ideia deste filtro é obter uma média de uma quantidade de leituras consecutivas N que estão dentro de uma “janela”, móvel, de amostragens de tamanho também N . A cada medição realizada, esta janela “movimenta-se”, abandonando a medição mais antiga e dando lugar à medição mais recente. A média aritmética de cada “janela”, é um valor que, de certa forma, acompanha os dados reais de medição.

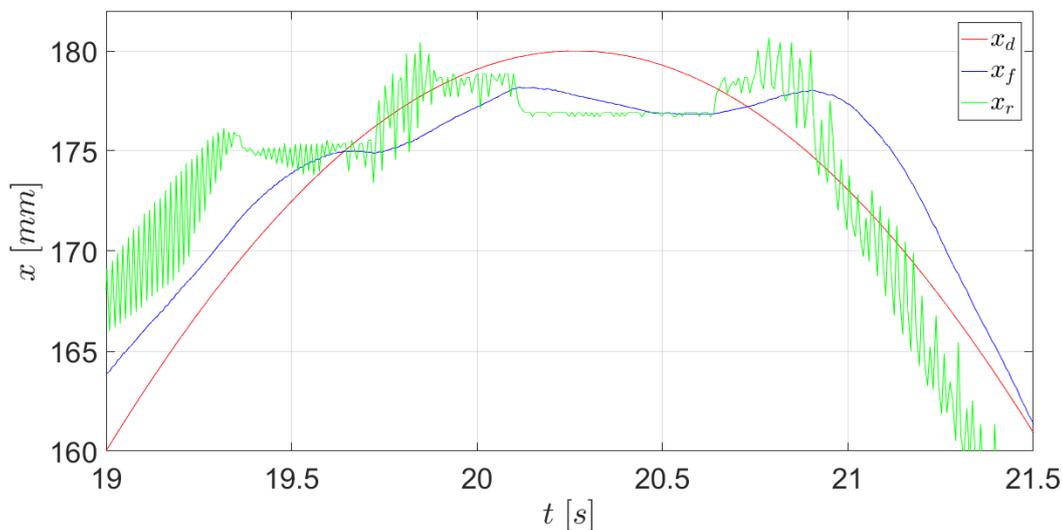
A saída do filtro, utilizada neste trabalho pelo controlador, nada mais é do que esta média. A equação que descreve este filtro é a seguinte:

$$x_f[i] = \frac{1}{N} \sum_{j=0}^{N-1} x_r[i - j] \quad (5.3)$$

Onde i é o índice da janela de filtragem. A quantidade N de medições utilizadas neste trabalho foi 50. Cumpre salientar que quanto maior N , mais lento será o controlador e menos fiel x_f será em relação à x_r .

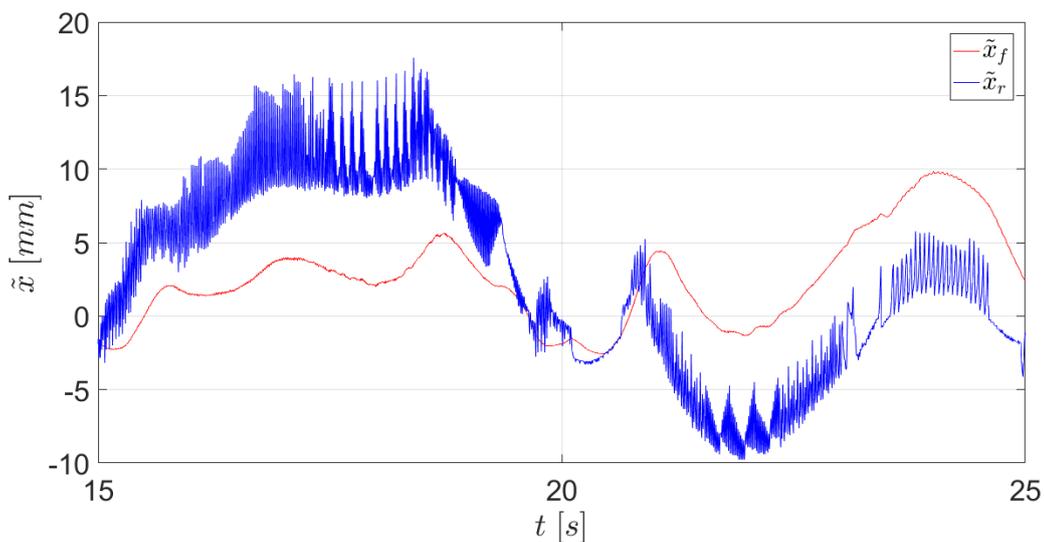
As figuras (21) e (22) comparam, para a trajetória senoidal, o comportamento da trajetória real lida x_r e da trajetória filtrada x_f em relação à trajetória desejada x_d e do erro filtrado \tilde{x}_f , utilizado no controlador, em relação ao erro real \tilde{x}_r calculado através de x_r .

Figura 21 – Comparação entre x_d , x_f e x_r para a Trajetória Senoidal (TS).



Fonte: Autoria própria.

Figura 22 – Comparação entre \tilde{x}_f e \tilde{x}_r para a Trajetória Senoidal (TS).



Fonte: Autoria própria.

A suavização do comportamento da leitura da posição é muito perceptível, bem como um leve *delay* entre x_f e x_r .

Ao analisar o gráfico do efeito do filtro no erro, é possível observar, mais uma vez esta leve diferença. A utilização de um filtro físico, eliminaria este *delay* haja vista que o sinal seria filtrado antes de chegar ao microcontrolador.

É importante frisar que a utilização de um filtro físico ou de outra abordagem de filtragem digital, como um filtro “*passa-baixa*”, poderia diminuir, ou até eliminar o *delay* gerado pelo filtro utilizado e melhorar o desempenho do controlador.

Cabe destacar também que os valores máximos de erro, foram diminuídos, o que também diminui o desempenho do controlador. Entretanto, tanto o comportamento da posição, quanto o do erro foram, de maneira geral, geometricamente mantidos nos gráficos.

6. ANÁLISE DOS RESULTADOS

Para que pudesse ser feita a avaliação do desempenho do controlador, foi desenvolvido um algoritmo em C++ o qual foi executado por um Arduino UNO R3. Os dados da trajetória foram coletados instante a instante a uma taxa de aproximadamente 150 amostras por segundo.

Nesta seção, serão avaliados, caso a caso, os dados coletados durante a implementação do controlador escolhido para este trabalho. Para a obtenção dos dados, ao controlador foram impostos diversos tipos de trajetórias, variadas manualmente (com utilização de um potenciômetro convencional conectado ao Arduino) e em função do tempo.

De antemão, adianta-se que o curso de 200 *mm* do atuador não foi explorado completamente para evitar que houvesse imprecisão dos dados coletados nas extremidades. Assim, para todas as trajetórias, o curso variou de 20 a 180 *mm*.

Cumprе salientar que, para um sinal de 5 *V*, máximo emitido pelo Arduino, a corrente de trabalho é cerca de 1000 *mA*. De acordo com o manual da eletroválvula, a corrente de trabalho deve ser limitada à 800 *mA* para evitar danos à mesma. Assim, limitou-se a intensidade do sinal a 80 % da capacidade total do micro controlador. Isto é, o esforço de controle máximo emitido pelo Arduino foi de 4 *V*. As trajetórias analisadas foram as seguintes:

- TML – Trajetória Manual Lenta (manual);
- TMR – Trajetória Manual Rápida (manual);
- TQ – Trajetória Quadrada (temporal);
- TT – Trajetória Triangular (temporal);
- TS – Trajetória Senoidal (temporal).

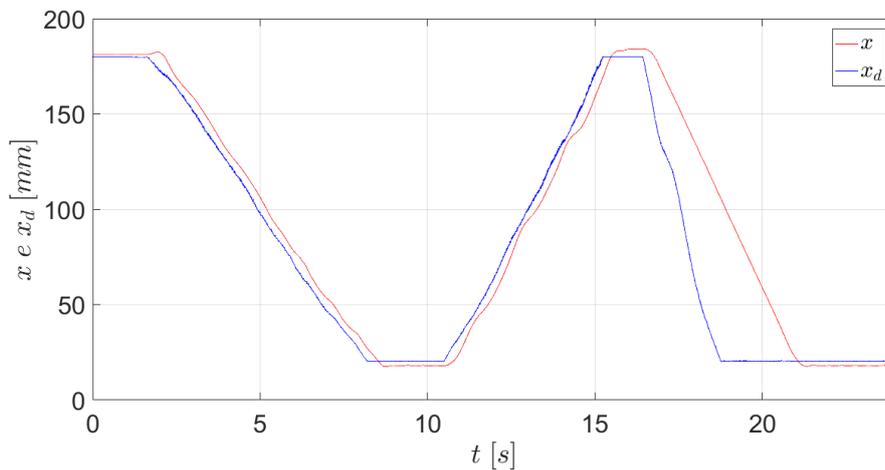
Sendo x_d a posição desejada, x a posição medida e filtrada, u_r o esforço de controle real, u_s o esforço de controle após a saturação e $\tilde{x} = x - x_d$ o erro do controlador, para cada caso, foram gerados gráficos que mostram o comportamento dessas variáveis. A seguir, será detalhado como foi obtida cada uma dessas trajetórias.

6.1. TML – Trajetória manual lenta

Para a obtenção dessa trajetória, o x_d foi variado, lentamente, de acordo com a posição de um potenciômetro convencional. Sabe-se que o sinal desse potenciômetro varia de 0 a 5 V, e o Arduino interpreta esse sinal analógico u em unidades que variam de 0 a 1023. Assim, quando o sinal emitido pelo potenciômetro for de zero unidades, espera-se que o x_d seja igual ao limite inferior (20 mm). Quando o sinal for de 1023 unidades, espera-se que o x_d seja igual ao limite superior (180 mm). Assim, a posição desejada para as trajetórias manuais será pela eq. (6.1):

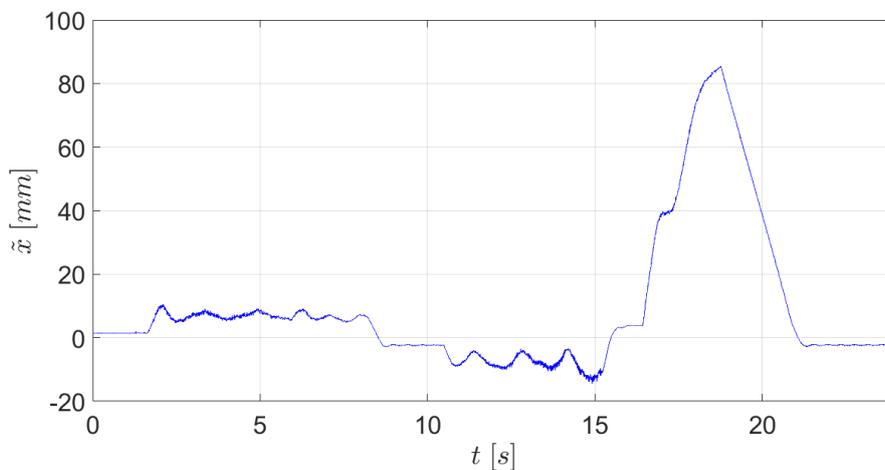
$$(x_d)_{TM} = 180 - 0,1564s \quad (6.1)$$

Figura 23 – Comparação entre x e x_d para Trajetória Manual Lenta (TML).

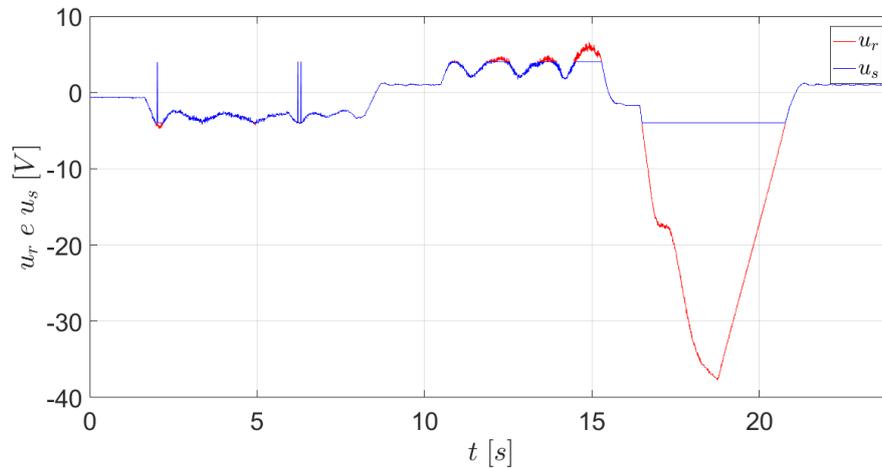


Fonte: Autoria própria.

Figura 24 – \tilde{x} para Trajetória Manual Lenta (TML).



Fonte: Autoria própria.

Figura 25 – u_r e u_s para Trajetória Manual Lenta (TML).

Fonte: Autoria própria.

A partir da observação das figuras (23), (24) e (25), percebe-se que, enquanto o x_d variava mais lentamente ($0 < t < 15\text{ s}$), a posição x o acompanhava facilmente, fazendo assim com que o erro \tilde{x} e os esforços de controle u_r e u_s se mantivessem bem próximos de zero.

Quando o x_d foi variado um pouco mais rapidamente (pouco após $t = 15\text{ s}$), o erro foi à pico, e, por conseguinte, assim foi o esforço de controle.

A tabela (4) mostra os valores de erro médio (absoluto e percentual em relação ao curso de 160 mm).

Tabela 4 – Erro médio para Trajetória Manual Lenta (TML).

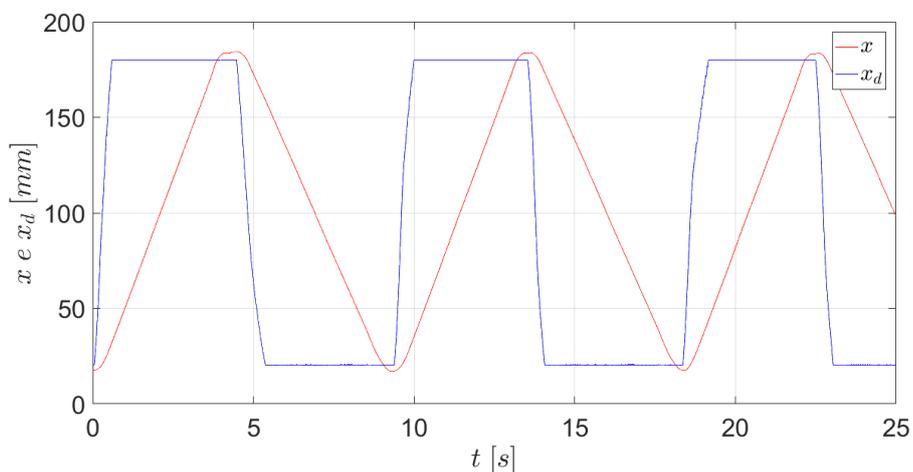
Média [mm]	Média percentual
11,59	7,25 %

6.2 TMR – Trajetória manual rápida

Neste caso, o x_d foi variado manual e rapidamente através do mesmo potenciômetro e de acordo com a mesma relação da eq. (6.1). Neste caso, o comportamento da posição desejada assemelhou-se bastante ao caso da trajetória quadrada (TQ) que será analisada a seguir.

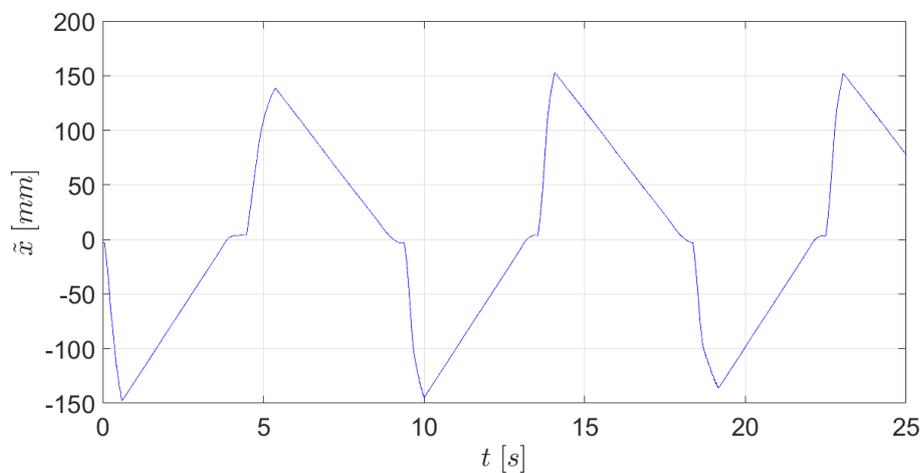
As figuras (26), (27) e (28) mostram que, sempre que o x_d é variado rapidamente, o erro e o esforço de controle vão a pico. Também é possível observar que, enquanto a posição desejada é mantida fixa, o erro leva cerca de 4 s para ir à zero, isto é, o atuador demora 4 s para completar um curso de 160 mm.

Figura 26 – Comparação entre x e x_d para Trajetória Manual Rápida (TMR).



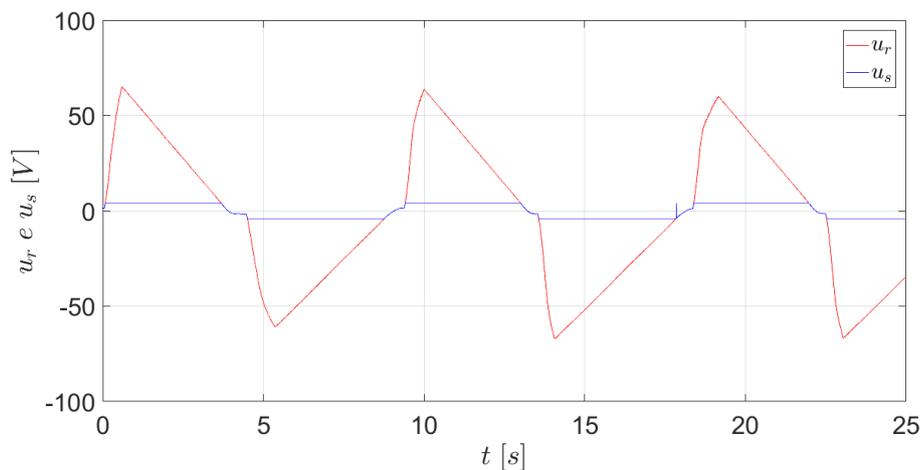
Fonte: Autoria própria.

Figura 27 – \tilde{x} para Trajetória Manual Rápida (TMR).



Fonte: Autoria própria.

Figura 28 – u_r e u_s para Trajetória Manual Rápida (TMR).



Fonte: Autoria própria.

A tabela (5) mostra os valores de erro médio (absoluto e percentual em relação ao curso de 160 mm).

Tabela 5 – Erro médio para Trajetória Manual Rápida (TMR).

Média [mm]	Média percentual
61,91	38,69 %

6.3. TQ – Trajetória quadrada

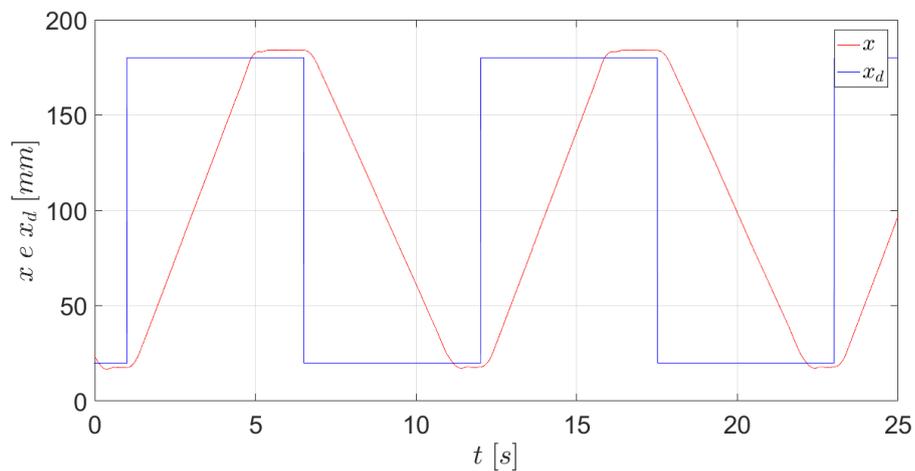
Neste caso, a onda senoidal w de período $T = 11 \text{ s}$ da eq. (6.2) foi utilizada como referência:

$$w = \sin\left(\frac{2\pi t}{T}\right) \quad (6.2)$$

Onde t é o tempo medido em segundos. Desse modo, o valor da posição desejada se comportaria de acordo com a eq. (6.3):

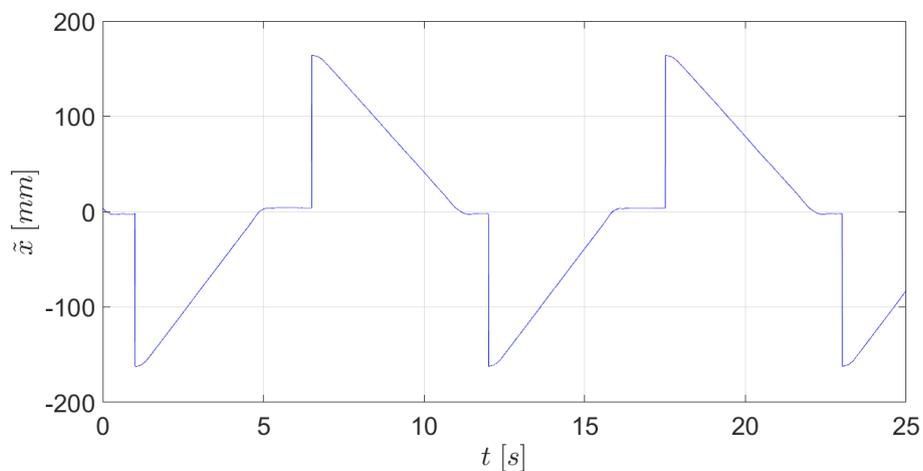
$$(x_d)_{TQ} = \begin{cases} 20 \text{ mm}, & w < 0 \\ 180 \text{ mm}, & w \geq 0 \end{cases} \quad (6.3)$$

Figura 29 – Comparação entre x e x_d para Trajetória Quadrada (TQ).

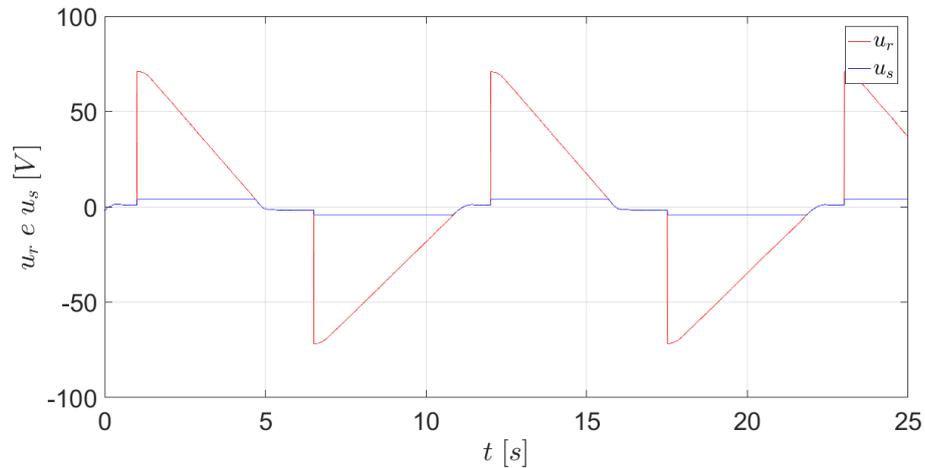


Fonte: Autoria própria.

Figura 30 – \tilde{x} para Trajetória Quadrada (TQ).



Fonte: Autoria própria.

Figura 31 – u_r e u_s para Trajetória Quadrada (TQ).

Fonte: Autoria própria.

As figuras (29), (30) e (31) mostram o comportamento da posição, do erro e do esforço de controle, respectivamente. Como mencionado, o comportamento desta modalidade de trajetória ficou muito similar ao visto no caso da TMR: Com picos para o erro e o esforço de controle sempre que o x_d muda de valor. Observa-se também o mesmo intervalo de 4 s para que o erro vá à zero.

A tabela (6) mostra os valores de erro médio (absoluto e percentual em relação ao curso de 160 mm).

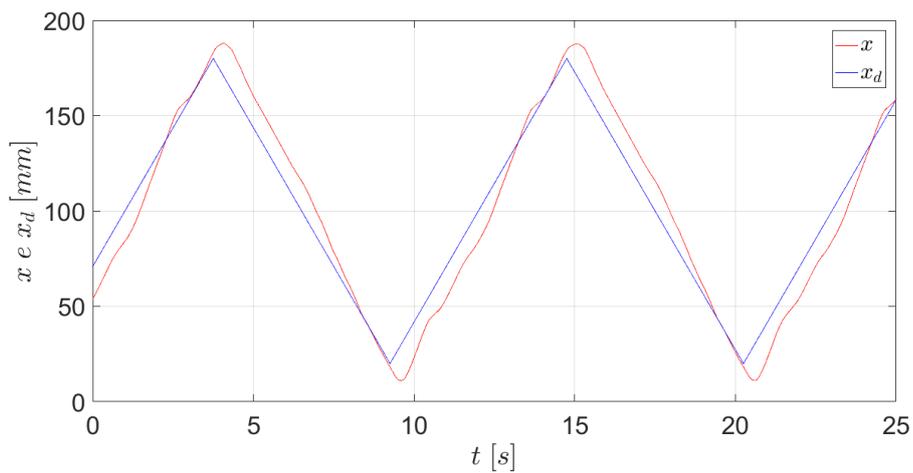
Tabela 6 – Erro médio para Trajetória Quadrada (TQ).

Média [mm]	Média percentual
65,35	40,84 %

6.4. TT – Trajetória triangular

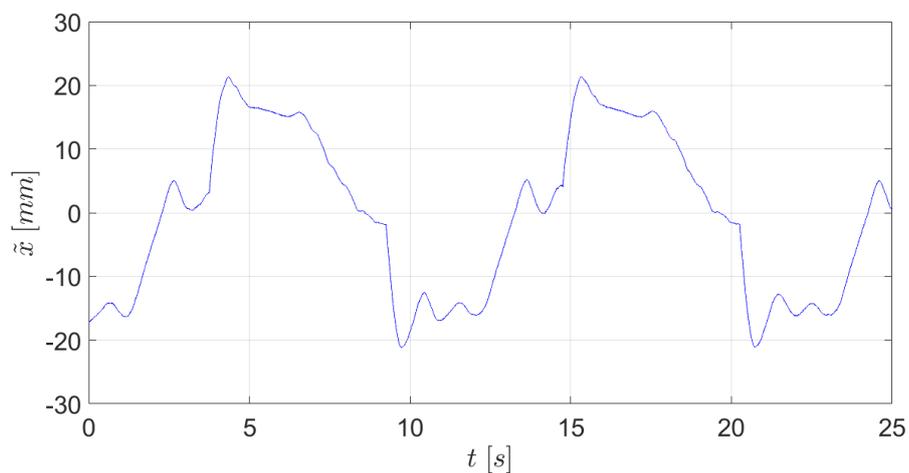
Esta trajetória foi gerada através da alternância entre retas. Para o curso, os mesmos limites inferior e superior de 20 mm e 180 mm , respectivamente, foram adotados. A trajetória foi projetada de tal forma que o movimento se repetisse a cada 11 s .

Figura 32 – Comparação entre x e x_d para Trajetória Triangular (TT).

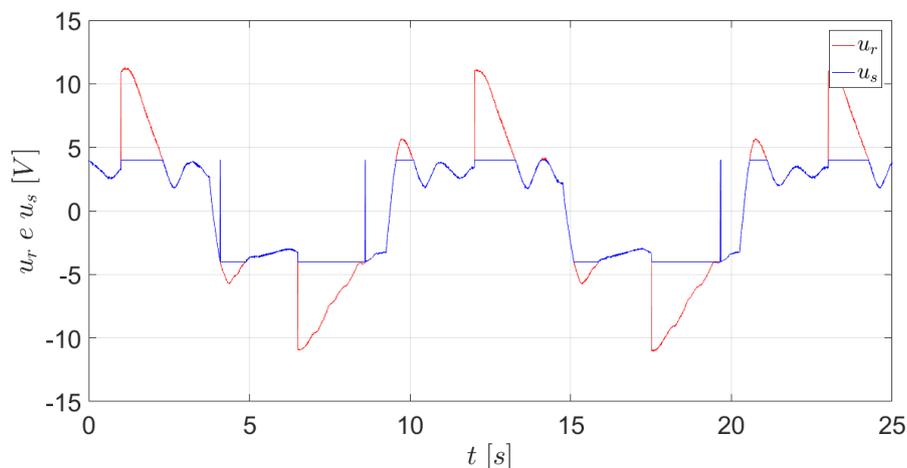


Fonte: Autoria própria.

Figura 33 – \tilde{x} para Trajetória Triangular (TT).



Fonte: Autoria própria.

Figura 34 – u_r e u_s para Trajetória Triangular (TT).

Fonte: Autoria própria.

As figuras (32), (33) e (34) mostram que, apesar da trajetória desejada ter sido, aparentemente, seguida de perto no gráfico da posição, o erro máximo chega a cerca de 20 mm. O mesmo erro aproxima-se de zero quando o atuador está nas regiões onde o atuador está finalizando um avanço, $125 < x_d < 180$ mm, e um recuo, $20 < x_d < 50$ mm, indicando que, nestas regiões, o controle é mais efetivo. Dentre os motivos que podem estar por trás disso estão a resposta do solenoide da eletroválvula à intensidade de corrente imposta pelo esforço de controle naquela região, balanceamento das pressões dentro do cilindro do atuador e etc.

Os picos de erro são, conseqüentemente acompanhados pelos picos no esforço de controle. Nas regiões onde o erro se aproxima de zero, assim acontece com o esforço de controle.

A tabela (7) mostra os valores de erro médio (absoluto e percentual em relação ao curso de 160 mm).

Tabela 7 – Erro médio para Trajetória Triangular (TT).

Média [mm]	Média percentual
10,96	6,85 %

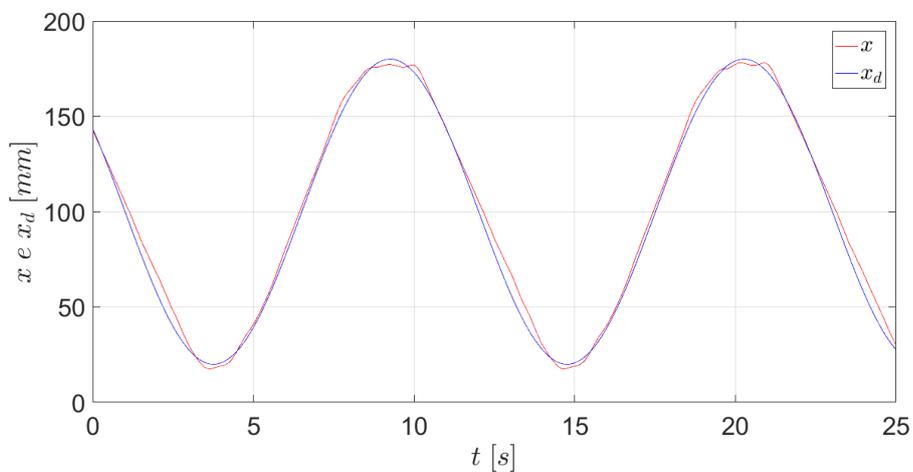
6.5. TS – Trajetória senoidal

A onda de referência para esta trajetória foi projetada para os mesmos limites inferior e superior utilizados nos casos anteriores e para um período $T = 11$ s. Desta forma, a posição desejada, em mm , comporta-se de acordo com a eq. (6.4):

$$(x_d)_{TS} = 100 - 80\sin\left(\frac{2\pi t}{T}\right) \quad (6.4)$$

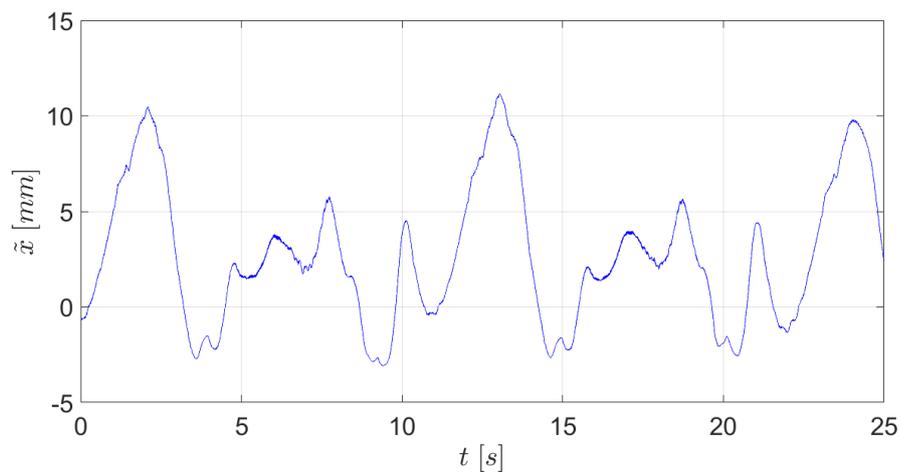
Onde t e T são medidos em segundos.

Figura 35 – Comparação entre x e x_d para Trajetória Senoidal (TS).

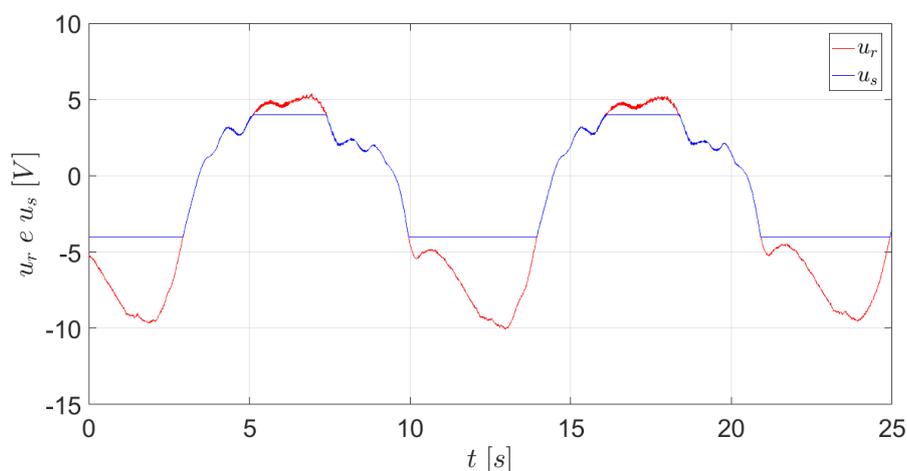


Fonte: Autoria própria.

Figura 36 – \tilde{x} para Trajetória Senoidal (TS).



Fonte: Autoria própria.

Figura 37 – u_r e u_s para Trajetória Senoidal (TS).

Fonte: Autoria própria.

As figuras (35), (36) e (37) mostram que, nas etapas de avanço, a posição desejada foi seguida de perto pelo atuador, com um erro máximo pontual de cerca de 5 mm. Entretanto, percebe-se que, nas etapas de recuo do atuador, o erro máximo chega a cerca de 10 mm. Esta diferença no desempenho do controle pode ser atribuída às diferenças de conservação de cada solenoide da eletroválvula. O erro vai à zero quando o atuador atinge as extremidades do curso.

O esforço de controle atinge seus valores máximos quando o atuador está nas regiões centrais do curso, o que se dá em função do erro, que, como mencionado, também atinge seus valores máximos nesta região. Como esperado, o esforço de controle atinge seu pico (cerca de 10 V) nas etapas de recuo do atuador, haja vista os maiores picos de erro. Assim como o erro, nas etapas de avanço, o esforço de controle máximo chega a metade do esforço de controle máximo das etapas de recuo (5 V).

A tabela (8) mostra os valores de erro médio (absoluto e percentual em relação ao curso de 160 mm).

Tabela 8 – Erro médio para Trajetória Senoidal (TS).

Média [mm]	Média percentual
4,91	3,07 %

6.6. Comparação geral

A tabela (9) reúne os dados de erro médio de todas as trajetórias:

Tabela 9 – Comparação entre erros médios de todas as trajetórias.

Trajectoria	Erro médio [mm]	Erro médio percentual
TML	11,59	7,25 %
TMR	61,91	38,69 %
TQ	65,35	40,84 %
TT	10,96	6,85 %
TS	4,91	3,07 %

A partir da análise dos dados acima, é possível observar que a trajetória sob a qual o controlador obteve um desempenho mais satisfatório foi a Trajetória Senoidal (TS).

A utilização de uma trajetória contínua e suave, é a razão pela qual o erro médio para esta modalidade de trajetória apresentou-se tão pequeno (3,07 % em relação ao curso do atuador). A trajetória que apresentou o segundo menor erro médio foi a Trajetória Triangular (TT).

Embora este tipo de trajetória apresente uma velocidade constante na maioria do tempo, há pontos não deriváveis na mesma, comprometendo assim, o desempenho do controlador.

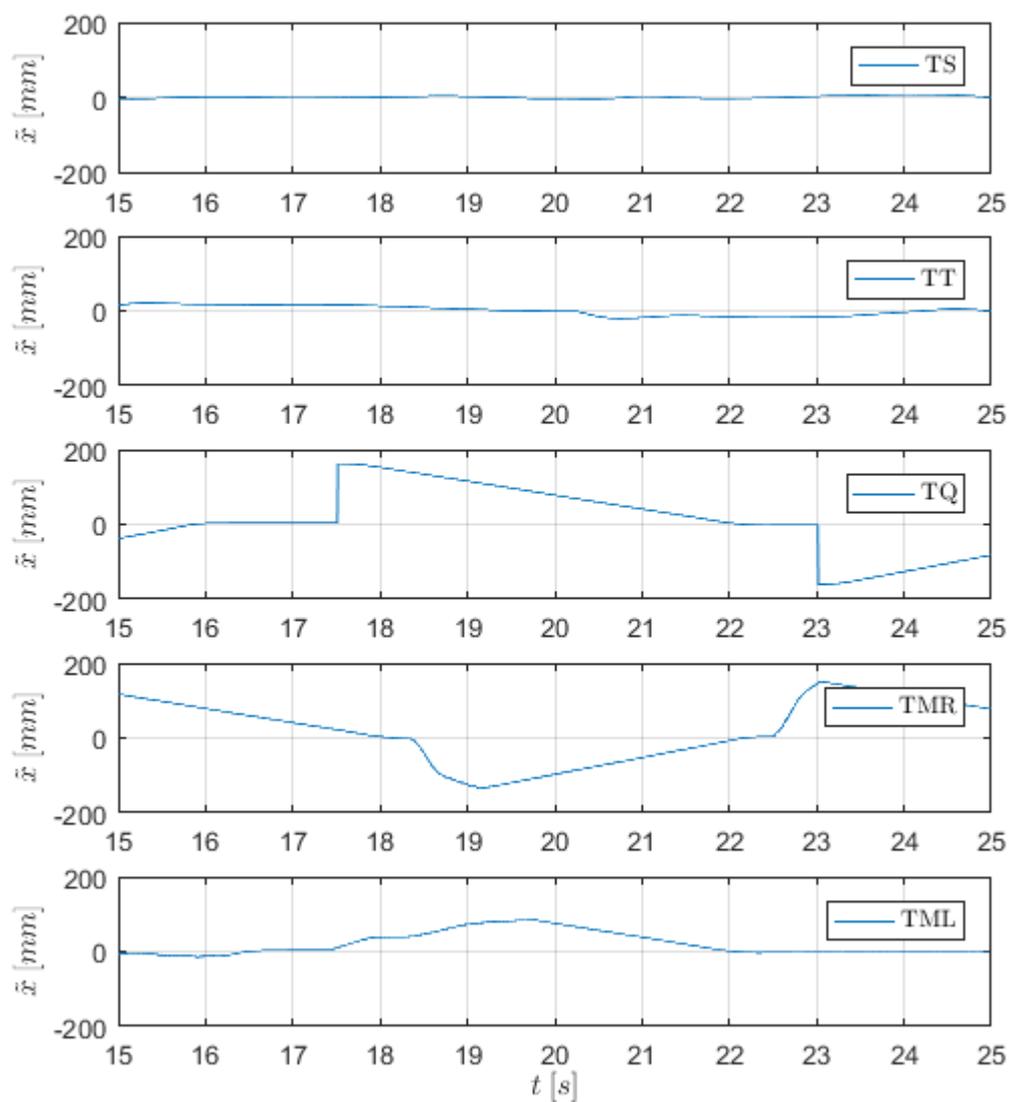
A Trajetória Manual Lenta (TML) também apresentou um erro médio relativamente baixo (7,25 %) e isso se dá em virtude da suavidade com que a mesma foi regulada.

O contrário acontece para a Trajetória Manual Rápida (TMR), onde o x_d era variado bruscamente, por isso, o erro médio desta trajetória foi relativamente alto (38,69 %). A Trajetória Quadrada (TQ) apresenta também um erro médio alto (40,84 %) em virtude do mesmo motivo da TMR.

A fig. (38) compara o comportamento do erro \tilde{x} para todas as trajetórias estudadas e em uma mesma escala.

Nela, é possível observar a discrepância entre a ordem de grandeza dos erros de cada trajetória.

Figura 38 – Comparação do erro de todas as trajetórias.

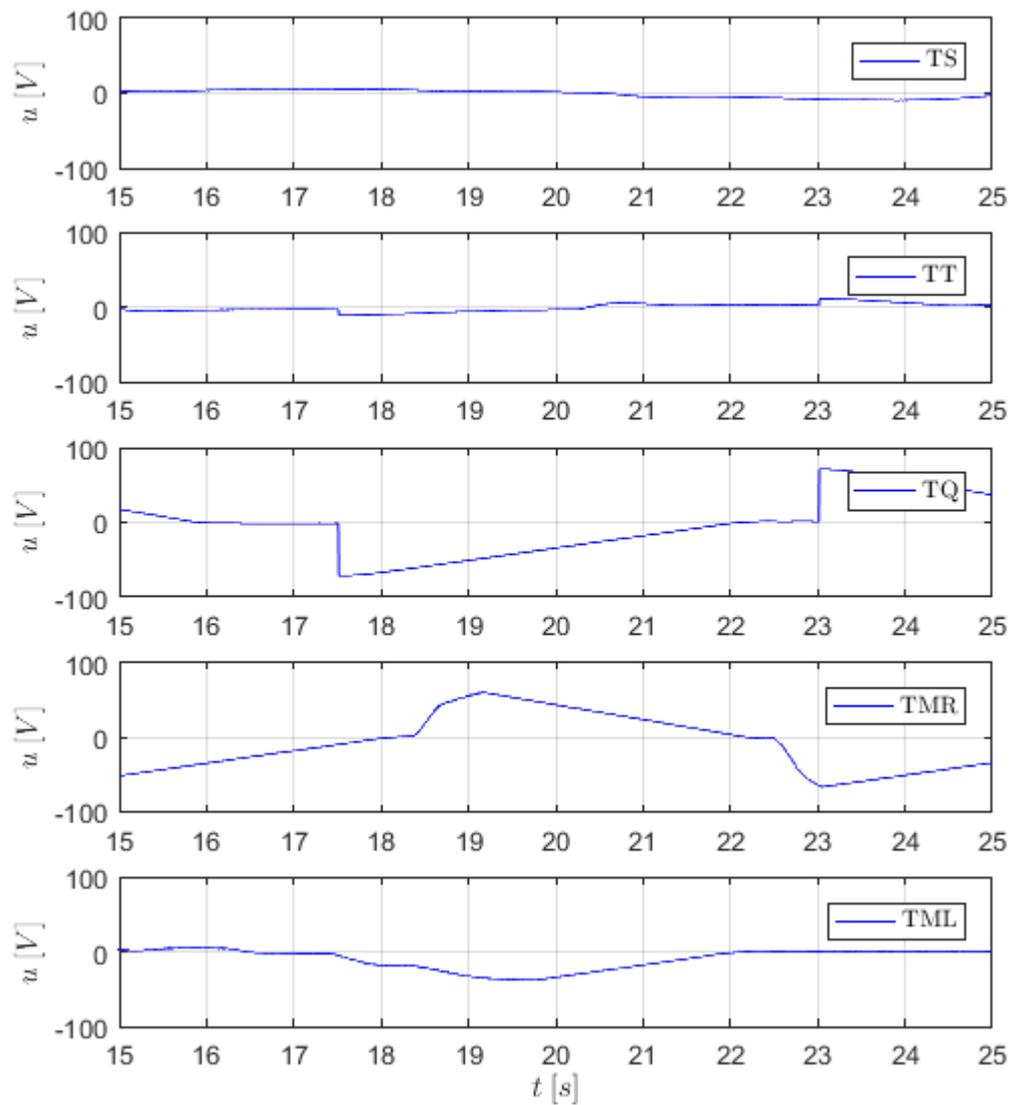


Fonte: Autoria própria.

A fig. (39) compara o comportamento do esforço de controle u para todas as trajetórias estudadas e em uma mesma escala.

Da mesma forma, é possível observar a diferença entre a ordem de grandeza dos esforços de controle desenvolvidos em cada trajetória.

Figura 39 – Comparação do esforço de controle não saturado de todas as trajetórias.



Fonte: Autoria própria.

7. CONSIDERAÇÕES FINAIS

Abordou-se neste trabalho a aplicação do método de linearização por realimentação a um sistema eletro-hidráulico cuja implementação foi feita via Arduino. A proposta desta solução é anular as não linearidades presentes no sistema através de um acompanhamento contínuo da trajetória.

A zona-morta modelada por Bessa, Dutra e Kreuzer (2010) é um dos principais geradores das incertezas presentes do sistema. Esta zona é oriunda do bloqueio do fluido durante a sobreposição do carretel na gaveta. Além disso, outras incertezas também interferem no comportamento da válvula e, por conseguinte, do atuador. Assim, o controlador deve contornar essas incertezas do sistema e realizar o controle como se o sistema fosse do tipo linear e em malha fechada.

A primeira etapa do trabalho consistiu no desenvolvimento do *hardware*, onde as relações entre o micro controlador e a eletroválvula e o mesmo micro controlador e o sensor de posição tiveram de ser modelados afim de que se conhecesse, de maneira precisa, como o sistema deveria responder.

Feito isso, a próxima etapa consistiu no desenvolvimento do algoritmo de controle, feito via Arduino. Durante esta etapa, o micro controlador teve de ser submetido a um aumento na sua taxa de processamento para que o controle se desse de maneira satisfatória. Uma das dificuldades durante esta etapa foi a escolha dos parâmetros que deveriam ser atribuídos ao sistema. Foi também nesta etapa onde foram implementadas as diferentes trajetórias as quais o atuador deveria seguir.

Estudou-se casos em que a trajetórias ajustadas manualmente através de um potenciômetro conectado ao Arduino (TML e TMR) e trajetórias as quais o comportamento dependia do tempo (TQ, TT e TS). Somente a posição era uma grandeza mensurável, assim, outros dados necessários como por exemplo a velocidade e o erro tiveram de ser medidos a partir dela. Tendo em vista a intensa geração de ruídos no sistema, um método de tratamento de dados foi implementado para que as flutuações da medição fossem amortizadas. O filtro empregado foi o filtro de médias móveis.

Gráficos foram gerados para demonstrar o comportamento do atuador em relação à trajetória desejada e o desempenho do controlador foi medido mediante os

erros médios apresentados para cada trajetória proposta. O desempenho mostrado foi condizente com a baixa capacidade de processamento da placa de controle empregada em relação a outras placas de aquisição empregadas comercialmente. Outro ponto importante, é que o controle de linearização por realimentação depende muito de um bom modelo do sistema do qual se deseja controlar, o que não é o caso deste trabalho. A redução do custo associado à montagem do sistema também em relação às placas comerciais é outro fator que deve ser avaliado e que pesa positivamente. Além disso, deve-se frisar que o Arduino possui uma grande comunidade *online* quando comparado com modelos de microcontroladores industriais, como a placa da *Advantech* mencionada neste trabalho, o que torna muito mais simples a obtenção de informações sobre a implementação de certas técnicas.

Para trabalhos futuros, sugere-se que seja desenvolvido um filtro físico para que os ruídos sejam amenizados sem que se comprometa a capacidade de processamento do micro controlador e sem que um *delay* seja gerado como ocorreu neste trabalho, implementação de outras técnicas de controle a fim de que os erros apresentados neste trabalho sejam reduzidos bem como a utilização de outros derivadores e ferramentas matemáticas que proporcionem desempenho mais satisfatório, modelagem das grandezas físicas do atuador hidráulico e a utilização de outros sistemas de processamento para implementação desta técnica de controle.

REFERÊNCIAS

ALARÇIN, Fuat. Nonlinear modelling of a fishing boat and Fuzzy Logic Control design for electro–hydraulic fin stabilizer system. **Nonlinear Dynamics**, [s.l.], v. 76, n. 1, p.581–590, 26 nov. 2013. Springer Nature. <http://dx.doi.org/10.1007/s11071-013-1152-9>.

AZEVEDO, George Oliveira de Araújo. **CONTROLE DE SISTEMAS ELETRO–HIDRÁULICOS VIA LINEARIZAÇÃO POR REALIMENTÇÃO COM COMPENSAÇÃO INTELIGENTE DE INCERTEZAS**. 2016. 141 f. Tese (Doutorado) – Engenharia Mecatrônica, Universidade Federal do Rio Grande do Norte, Natal, 2016.

BESSA, Wallace M. **CONTROLE POR MODOS DESLIZANTES DE SISTEMAS DINÂMICOS COM ZONA MORTA APLICADO AO POSICIONAMENTO DE ROVS**. 2005. 117 f. Tese (Doutorado) – Engenharia Mecânica, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2005.

BESSA, Wallace M.; DUTRA, Max S.; KREUZER, Edwin. An Adaptive Fuzzy Dead–Zone Compensation Scheme and its Application to Electro–Hydraulic Systems. **Journal Of The Brazilian Society Of Mechanical Sciences And Engineering**. p. 1–7. 03 2010.

EKORU, John E. D.; PEDRO, Jimoh O. Proportional–integral–derivative control of nonlinear half–car electro–hydraulic suspension systems. **Journal Of Zhejiang University Science A**, [s.l.], v. 14, n. 6, p.401–416, jun. 2013. Zhejiang University Press. <http://dx.doi.org/10.1631/jzus.a1200161>.

JUANG, Hau–shiu; LUM, Kai–yew. **Design and Control of a Two–Wheel Self–Balancing Robot using the Arduino Microcontroller Board**. In: IEEE INTERNATIONAL CONFERENCE ON CONTROL AND AUTOMATION (ICCA), 10., 2013, Hangzhou, China. Hangzhou, China: IEEE, 2013. p. 634 – 639.

KIM, Wonhee et al. Output feedback nonlinear control for electro–hydraulic systems. **Mechatronics**, [s.l.], v. 22, n. 6, p.766–777, set. 2012. Elsevier BV.

MERRITT, Herbert E. **Hydraulic Control Systems**. New York: J. Wiley & Sons, 1967.

OGATA, Katsuthiko. **Modern Control Engineering**. 5. ed. Englewood Cliffs: Prentice–hall, 2010.

SABERI, A.; LIN, Zongli; TEEL, A.r.. Control of linear systems with saturating actuators. **Ieee Transactions On Automatic Control**, [s.l.], v. 41, n. 3, p.368–378, mar. 1996. Institute of Electrical and Electronics Engineers (IEEE). <http://dx.doi.org/10.1109/9.486638>.

SANTOS, João Deodato Batista dos. **CONTROLE POR MODOS DESLIZANTES COM COMPENSAÇÃO DIFUSA APLICADO A SISTEMAS COM DESCONTINUIDADE**. 2013. 75 f. Dissertação (Mestrado) – Engenharia Mecânica, Universidade Federal do Rio Grande do Norte, Natal, 2013.

SMITH, Steven W. **The Scientist and Engineer's Guide to Digital Signal Processing**. 2. ed. San Diego: Ca: California Technical Publishing, 1999.

YAN, Jun et al. Nonlinear modeling and identification of the electro–hydraulic control system of an excavator arm using BONL model. **Chinese Journal Of Mechanical Engineering**, [s.l.], v. 26, n. 6, p.1212–1221, nov. 2013. Chinese Journal of Mechanical Engineering. <http://dx.doi.org/10.3901/cjme.2013.06.1212>.

YANG, Chifu; HUANG, Qitao; HAN, Junwei. Computed force and velocity control for spatial multi–DOF electro–hydraulic parallel manipulator. **Mechatronics**, [s.l.], v. 22, n. 6, p.715–722, set. 2012. Elsevier BV. <http://dx.doi.org/10.1016/j.mechatronics.2012.03.002>.