

ANÁLISE EXPERIMENTAL DO DESEMPENHO DOS SONARES LV-EZ, US100 E

HC-SR04

Antonio Eduardo Assis Amorim
Doutor, FATEC-JAHU – CEETEPS, aea.amorim@fatec.sp.gov.br

Sara Fernanda Prado
FATEC-JAHU – CEETEPS, sara.prado01@fatec.sp.gov.br

Robson Luiz Felipe Dias de Oliveira
FATEC-JAHU – CEETEPS, robson.oliveira33@fatec.sp.gov.br

Flavio Augusto Ribeiro da Silva
FATEC-JAHU – CEETEPS, flavio.silva131@fatec.sp.gov.br

João Pedro Brunello
FATEC-JAHU – CEETEPS, joao.brunello@fatec.sp.gov.br

RESUMO

Com a popularização de microcontroladores de baixo custo, uma gama destes dispositivos e sensores se encontram disponíveis no mercado, permitindo o desenvolvimento de projetos simples até aos mais sofisticados na área de robótica. Um dos sistemas de relevância nesta área, são os chamados sistemas de navegação, que permite ao veículo navegar de forma segura em um ambiente não estruturado, identificando obstáculos e traçando rotas até atingir o objetivo. Neste sistema são empregados sensores de distância. Um deles, o sonar, está disponível no mercado uma boa variedade, sendo os mais empregados o LV Max Sonar, US100 e o HC SR-04. Existem diversos projetos disponíveis na literatura utilizando a plataforma Arduino, alguns deles com bibliotecas específicas. A maioria dos sites ou livros que tratam destes projetos apresentam detalhes de montagem ou a sua codificação, mas não mostram detalhes da calibração dos sensores. Tais informações são relevantes quando se projeta um sistema de robótica, uma vez que é importante verificar a sua robustez, tanto de software como de hardware e se atende as exigências do projeto. Neste trabalho analisamos o comportamento destes sensores, assim como na codificação por meio de um procedimento de calibração do sonar, analisando a exatidão assim como a sua incerteza expandida. Mostramos que o sensor LV Max Sonar possui uma limitação de leitura para baixas distâncias se prestando para atividades de mapeamento de ambiente enquanto os sonares US100 e HC SR-04 possuem comportamentos similares e apresentam uma boa faixa de leitura, tanto pequena para médias distâncias, servindo como sensor para mapeamento e detecção de obstáculos. O uso da biblioteca produz pequenas variações de resultados, sendo que a escolha apenas irá depender do tamanho da codificação.

Palavras-chave: Robótica. Sonar. Arduino. Sistema de Navegação. Sistema de Mapeamento.

ABSTRACT

With the popularization of low-cost microcontrollers, a range of these devices and sensors are available on the market, allowing the development of simple projects even the most sophisticated in the area of robotics. One of the relevant systems in this area are the so-called navigation systems, which allow the vehicle to navigate safely in an unstructured environment, identifying obstacles and tracing routes until reaching the objective. Distance sensors are used in this system. One of them, sonar, is available on the market with a good variety, the most used being the LV Max Sonar, US100 and HC SR-04. There are several projects available in the literature using the Arduino platform, some

of them with specific libraries. Most of the websites or books dealing with these projects present details of assembly or their coding, but do not show details of the calibration of the sensors. Such information is relevant when designing a robotics system, since it is important to check its robustness, both in software and hardware, and if it meets the requirements of the project. In this work, we analyze the behavior of these sensors, as well as in the coding by means of a sonar calibration procedure, analyzing the accuracy as well as their expanded uncertainty. We show that the LV Max Sonar sensor has a reading limitation for low distances lending itself to environment mapping activities while the US100 and HC SR-04 sonar have similar behaviors and present a good reading range, both small for medium distances, serving as a sensor for mapping and detecting obstacles. The use of the library produces small variations in results, the choice of which will only depend on the size of the encoding.

Keywords: Robotics. Sonar. Arduino. Navigation System. Mapping System.

1. INTRODUÇÃO

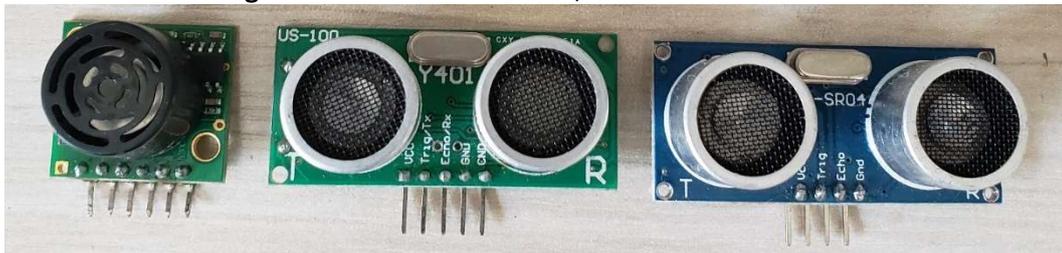
Com a recente popularização no uso de microcontroladores e a consequente redução de custos, há uma considerável bibliografia que trata sobre o uso da plataforma Arduino e sensores em projetos de Robótica e Sistemas Embarcados. Dentre os diversos sensores disponíveis a baixos custos, destacam-se os sonares que são utilizados para detecção de objetos e mapeamento de ambientes (SCHMIDT, 2011; SCHWARTZ, 2016).

Um dos sistemas de relevância na área de robótica são os chamados sistemas de navegação, que permite ao veículo navegar em um ambiente identificando obstáculos e traçando rotas até atingir o objetivo (ZIEMANN, 2018). Em geral, podem ser utilizados como sensores o sonar, sensor de distância infravermelho ou qualquer dispositivo que meça a distância entre o veículo e o obstáculo. Outra aplicação de relevância é o mapeamento do ambiente com o auxílio do sonar. Em geral, são utilizados vários sonares acoplados em uma cadeia ou instalando um sonar no servo motor (CHANG e colab., 2016; CONTE e colab., 2006; FALLON e colab., 2011).

Existem diversos projetos disponíveis na literatura, que tratam de projetos que envolvem o sonar como um dispositivo para a medida da distância, utilizando o Arduino como plataforma (EVANS, 2011; MONK, 2002; WARREN e colab., 2008). No mercado são comumente encontrados os dispositivos US100 (OEM), HC SR-04 (OEM) e a linha de sonares da MaxBotix, mostrados na Figura 1. Tais sensores emitem pulsos de onda sonora com a frequência de 40 kHz. Tais ondas são um tipo de onda mecânica que requer um meio material para se propagar e ela se dá longitudinalmente e em todas as direções (NUSSENZVEIG, 2014). Da mesma forma, existem codificações diferentes, algumas delas utilizando bibliotecas Ultrasonic ou NewPing.

A maioria dos sites ou livros que tratam destes sensores apresentam detalhes de montagem ou o código para a sua operação, mas não mostram detalhes da calibração dos sensores, do desempenho deles ou mesmo das diferenças das suas limitações operacionais. Tais informações são relevantes quando se projeta um sistema de robótica, uma vez que é importante verificar a sua robustez, tanto de software como de hardware, se atende as exigências do projeto e as limitações de hardware, das respostas dos sinais, as possíveis falhas de leitura etc.

Figura 1. Sonar LV Max Sonar, US100 e HC SR-04.



Fonte: Autores (2021).

Uma etapa essencial quando se utiliza sensores é calibração do dispositivo permitindo que sejam conhecidas a exatidão da medida da distância, a incerteza desta medida, que incorpora incertezas aleatórias e as não aleatórias (GUM e colab., 2003).

Tendo em vista que há diversos modelos de sonar e de codificações, a proposta deste artigo é analisar o comportamento dos sensores LV Max Sonar¹, US100² e HC SR-04³. Neste estudo, será usado um procedimento de calibração do sonar, extraíndo a curva de calibração de cada um deles, de forma a verificar os seus desempenhos. De forma similar, no tocante a codificação, também será analisado o desempenho em relação às bibliotecas. Para o estudo da incerteza expandida será utilizado um modelo matemático simples, envolvendo as incertezas da repetibilidade, da estatística da leitura e da escala do aparelho. Na seção 0 apresentamos um panorama sobre a codificação envolvendo sonar e a modelagem matemática para a calibração do sonar. Na seção 0 descrevemos os procedimentos experimentais e os equipamentos utilizados. A seção 0 apresenta os resultados envolvendo os três sonares e as suas respectivas codificações sem biblioteca, e com as bibliotecas Ping e UltraSonic. Por fim, a seção 0 apresenta as conclusões sobre o desempenho dos três sonares.

¹ www.maxbotix.com/documents/LV-MaxSonar-EZ_Datasheet.pdf

² www.adafruit.com/product/4019

³ <https://cdn.sparkfun.com/datasheet/Sensors/proximity/HCSR04.pdf>

2. SONARES PARA ARDUINO E OS CÓDIGOS

SCHMIDT, 2011 apresenta um projeto para detectar a distância na qual se emprega o comando `pulseIn` do Arduino⁴ acoplado a um sonar que possui apenas um pino para a transmissão do sinal, de modo que a sua codificação é feita por meio do chaveamento de uma entrada digital do Arduino. Este comando espera que o estado do pino seja modificado para que a contagem do tempo se inicie, interrompendo quando houver uma nova mudança de estado, retornando o comprimento do pulso em μs ou 0 se o pulso não foi recebido dentro do tempo limite.

Em sua codificação, o autor utiliza 2 μs para o sinal ficar no estado baixo, 5 μs no estado alto. Outro ponto de destaque é que a velocidade do som no ar depende da temperatura e da umidade do ambiente (MONK, 2002; SCHMIDT, 2011) .

MCROBERTS, 2011 emprega o sonar LV Max Sonar (LVMax EZ1) que permite trabalhar em três modos de operação, um sinal de saída de largura de pulso (*pulse width output*) – PWO, saída analógica e saída serial RS232. A codificação utiliza o mesmo comando `pulseIn`. Contudo, o autor emprega o comando *delay* de 100 ms entre as leituras. De acordo com o fabricante, ao ligar o sonar, ele necessita de um ciclo de leitura para se calibrar, sendo que requer um obstáculo situado a uma distância superior a 18 cm. De acordo com o autor, o sistema apresenta boa sensibilidade, requerendo para isto 0,5 ms. A qualidade da leitura piora para distâncias menores que 6 polegadas.

Para a mensuração do intervalo de tempo com o sonar usando a plataforma Arduino existem os comandos `PulseIn` e `PulseInLong`⁵. De acordo com o manual, eles medem o comprimento da largura do pulso em microssegundos, de forma empírica. Para pulsos longos, o erro será maior.

Se observa, em geral, nos sites e mesmo na literatura que trata destes sensores, a inexistência do procedimento de calibração (MARGOLIS, 2013; WARREN e colab., 2008). Nesta etapa são importantes não só a medida do sinal, mas também o valor da sua incerteza. De acordo com a literatura, existem dois tipos de erros, um deles aleatória, cuja incerteza é de repetibilidade enquanto a outra que envolve outros tipos de erros. A modelagem utilizada neste estudo envolve as seguintes incertezas:

⁴ <https://www.arduino.cc/reference/en/language/functions/advanced-io/pulsein/>

⁵ www.arduino.cc/reference/en/language/functions/advanced-io/pulseinlong/

- Repetibilidade i_r , calculada pelo desvio padrão da média;
- Estatística de leitura i_e , calculada por meio da amplitude das variações das leituras (Δm). $i_l = \frac{\Delta m}{\sqrt{3}}$;
- Escala da trena utilizada para a medida da distância, cuja limite de escala é 1mm $i_t = 0,05cm$.

A incerteza expandida, que contempla um nível de confiança de 95,45%, é

$$i_e = \sqrt{i_r^2 + i_l^2 + i_t^2}. \quad (1)$$

3. MATERIAIS E MÉTODOS

Para a análise foram utilizados o sonar da MaxBotix, modelo LVmaxS-EZ e os sonares US100 e o HC-SR04.

A leitura da distância do sonar ao obstáculo para a calibração é feita usando a trena de marca Baustroffe, cuja resolução é 1 mm. O sonar é conectado à placa protoboard e, desta placa, são feitas as conexões ao Arduino Mega 2560. Para o teste usando o sinal de saída PWO, o pino de saída é conectado à porta digital 13(PWM) da placa, na qual também foi conectado um osciloscópio UNI-T, modelo UT81B, na qual a resolução do osciloscópio é 0,001 Hz e uma acurácia de $\pm 0,1\% + 3$. O registro da forma da onda é feito pelo software que acompanha o osciloscópio.

A codificação registra o sinal e o cálculo da distância é obtida usando a média móvel com uma amostra de 20 leituras. Inicialmente é feito o teste de exatidão do sonar, na qual o obstáculo é afastado do sonar de 5 em 5 cm, sem conter termo de correção. Os resultados são plotados no Excel e a expressão de correção dos dados é obtida por meio do ajuste de curva linear.

Em seguida, um novo teste de calibração é feito com o termo de correção, na qual se extraem o valor médio da medida e as incertezas. Posteriormente a codificação é alterada para se estudar a saída analógica. Para o teste, o pino de saída é conectado à porta analógica A0 da placa e a tensão de entrada neste pino é medida com o mesmo osciloscópio, selecionando a faixa de tensão da voltagem.

No caso em questão, o Quadro 1 mostra a resolução e a acurácia do aparelho.

Quadro 1. Características operacionais do osciloscópio UNI-T, modelo UT81B.

Faixa	Resolução	Acurácia
400mV	100 μ V	$\pm (0,8\%+8)$

Fonte: Autores (2021).

Desta forma, o limite de calibração do equipamento L_c , é

$$L_c = \left(\frac{0,8}{100} \times \text{leitura} + 8 \times 10^{-4} \right) V. \quad (2)$$

A incerteza de calibração, é

$$u_c = \frac{L_c}{2}. \quad (3)$$

O ensaio de calibração segue o mesmo procedimento já citado. Por fim, testa-se nos sonares US100 e HC-SR04 o seu desempenho, que operam usando o trigger e o echo, sendo conectados nas portas 7 e 8 do Arduino, sem a biblioteca, com as bibliotecas Ultrasonic e NewPing.

4. RESULTADOS E DISCUSSÃO

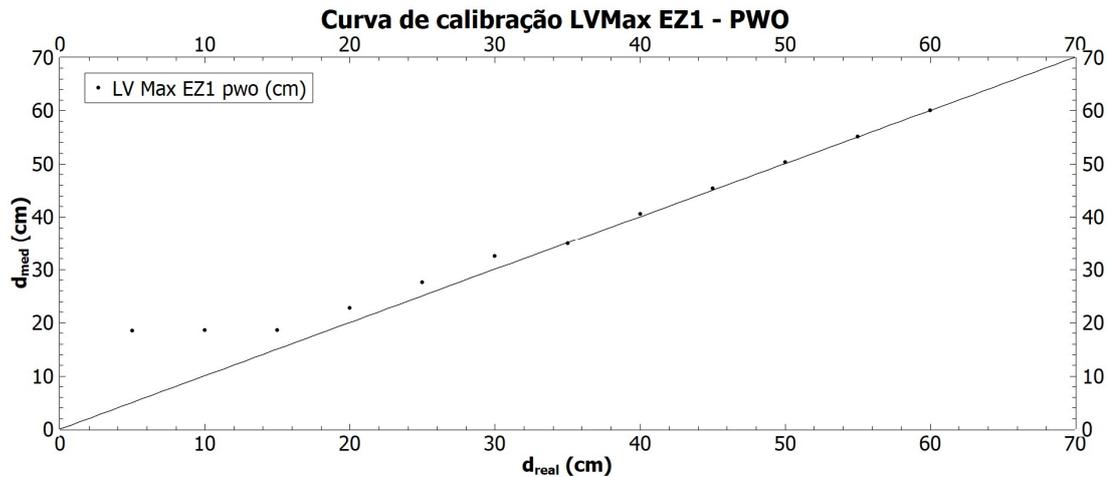
Inicialmente apresentamos as calibrações feitas para cada sensor. Para o modelo LVmaxS-EZ, que usa o modo PWO, usamos a seguinte expressão de calibração

$$\text{senal1} = 0,0167940213 \text{ sensor} + 4,83751, \quad (4)$$

na qual sensor é a variável de leitura extraído do comando pulseIn do Arduino e senal1 é o valor corrigido que será usado no cálculo da média móvel. A Figura 2 mostra a curva de calibração do sonar LVMax EZ1 operando no modo PWO.

Os pontos representam as medidas e a linha descreve a leitura esperada. Observamos, como mencionado pelo fabricante, o sonar apresenta uma boa exatidão de leitura a partir de 35 cm. A partir desta distância, a curva de calibração é aparentemente linear e a incerteza é pequena e constante ao longo das leituras.

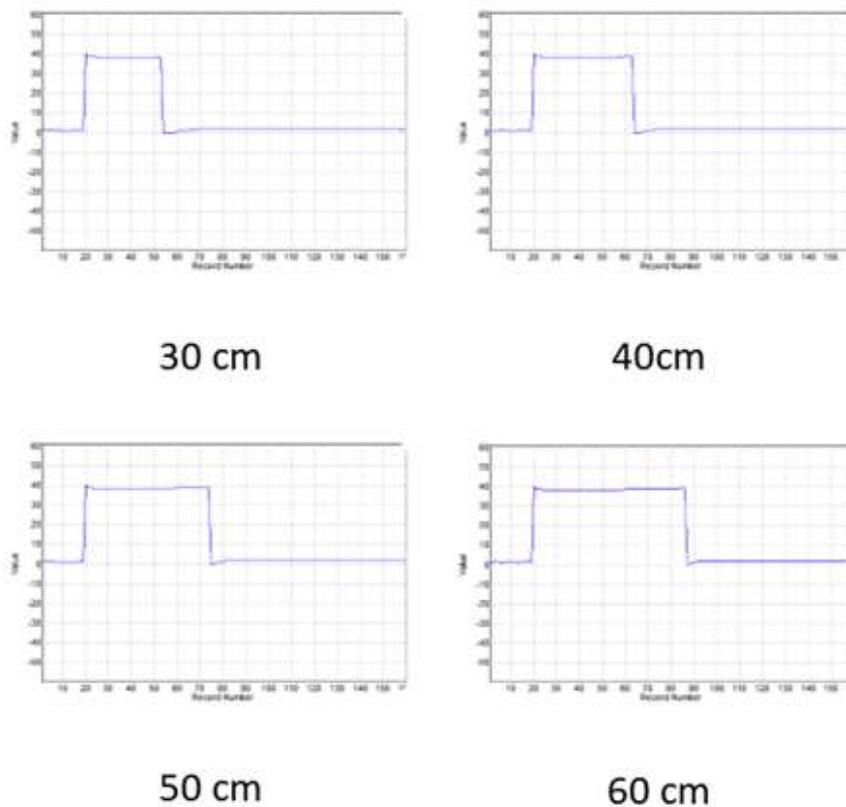
Figura 2. Curva de calibração do sonar no modo PWO. A linha representa o valor esperado da leitura.



Fonte: Autores (2021).

A Figura 3 mostra o resultado obtido no osciloscópio para um obstáculo que está em diversas posições no modo PWO. Observamos que à medida que o objeto se afasta do sonar, a largura do pulso aumenta, como era esperado. Contudo, o espaçamento não se altera se a distância entre o sonar e o obstáculo for menor que 15 cm.

Figura 3. Resultado da emissão de sinal do sonar da Maxbotix para um obstáculo no modo PWO.



Fonte: Autores (2021).

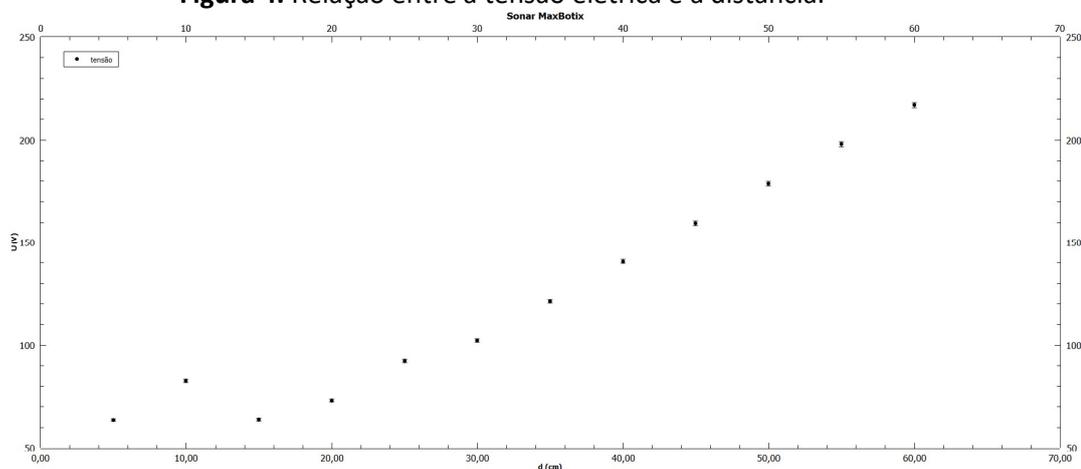
Para o modelo LVmax-EZ1, usando a saída analógica, empregamos a seguinte expressão para a calibração

$$mm = \frac{5}{3} \text{anvolt}, \quad (5)$$

na qual anVolt é a variável de leitura do sinal analógico por meio do comando analogRead e mm é a variável corrigida.

A Figura 4 mostra a relação entre a distância e a tensão gerada pelo sensor. Observamos a linearidade entre a tensão e a distância a partir de 20 cm. Abaixo deste valor, o sonar não responde adequadamente, o que era esperado.

Figura 4. Relação entre a tensão elétrica e a distância.



Fonte: Autores (2021).

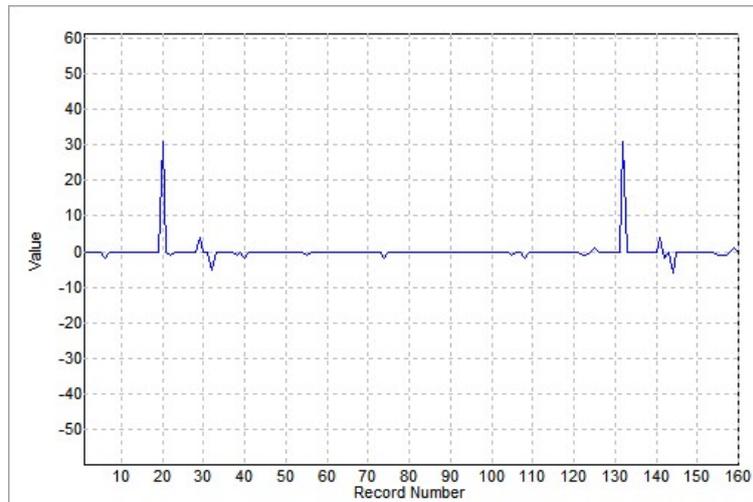
A Figura 5 mostra o comportamento do sinal envolvendo a emissão pela porta trigger e a recepção, pela porta echo para os modelos US100 e HC-SR04. O espaçamento entre os picos está relacionado com a distância. Os picos menores são ruídos. Embora presentes, eles não interferem no funcionamento do sonar.

Para o modelo US100, sem o uso de biblioteca, usamos a seguinte expressão para a calibração

$$\text{senal1} = 0,0172 \text{ sensor} - 0,4412, \quad (6)$$

na qual sensor é a variável de leitura do sinal analógico por meio do comando pulseIn e senal1 é o valor corrigido que será usado no cálculo da distância mediante a média móvel.

Figura 5. Formato dos pulsos de emissão (trigger) e de retorno (echo) para os modelos US100 e HC-SR04.



Fonte: Autores (2021).

Para o modelo US100, com a biblioteca Ultrasonic, empregamos a seguinte expressão de calibração

$$\text{senal1} = 0,9537 \text{ sensor} + 0,1589. \quad (7)$$

Para o mesmo modelo de sonar, utilizando a biblioteca NewPing, usamos a seguinte expressão para a curva de calibração

$$\text{senal1} = 0,9766 \text{ sensor} - 0,2625. \quad (8)$$

Para o modelo HC-SR04, sem biblioteca, a expressão de calibração utilizada foi

$$\text{senal1} = 0,0176 \text{ sensor} - 0,548. \quad (9)$$

Para o mesmo modelo de sonar, com a biblioteca Ultrasonic, a expressão de calibração do sensor empregada foi

$$\text{senal1} = 0,9811 \text{ sensor} - 0,7008. \quad (10)$$

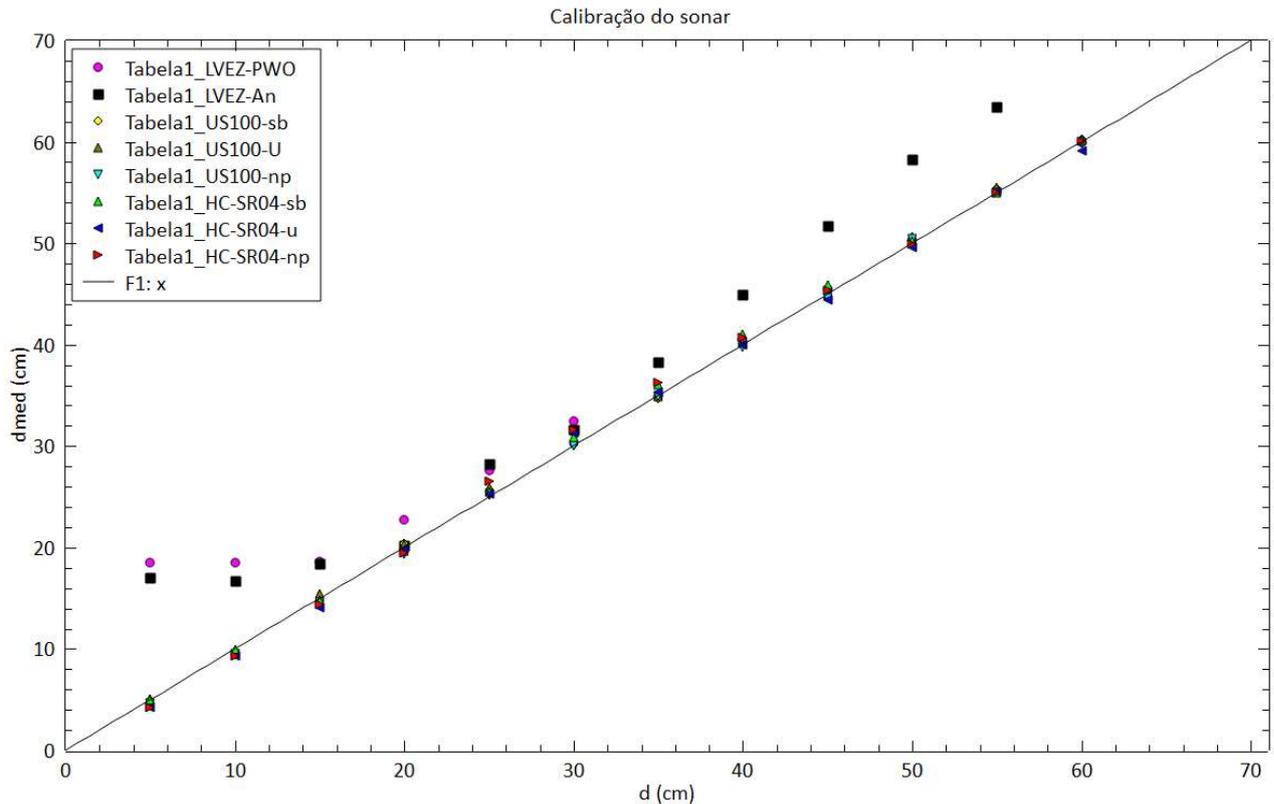
Por fim, o mesmo modelo de sonar com a biblioteca ping, empregamos a seguinte expressão de calibração

$$\text{senal1} = 1,0159 \text{ sensor} - 0,8874. \quad (11)$$

A Figura 6 mostra o resultado da calibração dos três sonares com as diferentes codificações. A linha azul representa o valor esperado da leitura. Observamos que o sonar

LVEZ apresenta um baixo desempenho para distâncias menores que 30 cm. Para distâncias maiores que 30 cm, o sonar apresenta um desempenho adequado, se prestando como um sensor de mapeamento, mas não para a detecção de obstáculos à curta distância. Os demais sonares apresentam um bom desempenho na faixa mensurada de 0 a 60 cm.

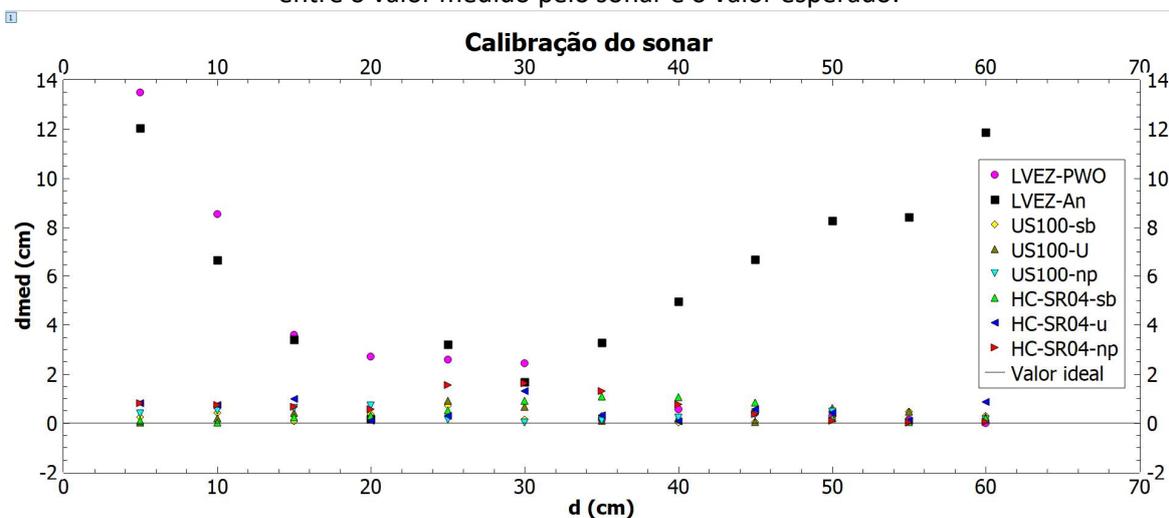
Figura 6. Curva de calibração do sonar. A reta representa a leitura esperada do sonar.



Fonte: Autores (2021).

Uma outra forma de comparar o desempenho de cada sensor é por meio da diferença absoluta entre a leitura do sensor e a distância esperada. No eixo x estão as distâncias padronizadas de comparação e no eixo vertical a diferença absoluta entre o valor medido pelo sonar e a leitura esperada. Os melhores resultados devem se aproximar do zero. O resultado é mostrado na Figura 7.

Figura 7. Curva de calibração do sonar LVEZ, US100 e HC-SR04. No eixo horizontal estão localizadas as distâncias padronizadas na qual o obstáculo foi posicionado e no eixo vertical a diferença absoluta entre o valor medido pelo sonar e o valor esperado.



Fonte: Autores (2021).

Inicialmente percebemos que o sonar LVEZ, tanto no modo analógico como no modo PWO é inadequado para se trabalhar em projetos que envolvam distâncias menores que 20 cm, como já destacado pelo fabricante. Porém, se observa que a exatidão, deste modelo usando o modo analógico, piora à medida que o obstáculo está mais distante. Já o seu uso no modo PWO melhora se trabalhando com distâncias maiores que 40 cm.

Tanto o modelo US100 como o HC-SR04 apresentaram bom comportamento para as distâncias testadas, e o uso ou não da biblioteca, aparentemente não mostrou vantagens significativas na calibração e o seu impacto aparece apenas no tamanho do arquivo, como é mostrado na Tabela 1. Estes modelos podem ser utilizados como sensores de obstáculos como para atividades de mapeamento de ambientes.

Tabela 1. Tamanho médio do arquivo em função do uso ou não das bibliotecas para o sonar.

Estrutura do código	Tamanho médio do arquivo (bytes)
Sem biblioteca	4784
Ultrasonic	5020
NewPing	4678

Fonte: Autores (2021).

5. CONSIDERAÇÕES FINAIS

Observamos que nos livros textos que tratam sobre projetos de Arduino a ênfase se concentra mais na codificação ou ao hardware. Contudo, quando se procura desenvolver projetos robóticos em que tais componentes fazem parte de sistemas de mapeamento ou de detecção de obstáculos, uma atenção especial precisa ser dada ao processo de calibração do sistema e que define a qualidade do sensor, servindo como um instrumento de decisão para a escolha do código ou do tipo de sonar.

Em geral a literatura trata do uso de três tipos de sonares, o LV-EZ da MaxBotix, US100 e o HC-SR04, que não destacam o fabricante (OEM). De forma similar são encontradas codificações sem uso de biblioteca, ou com as bibliotecas Ultrasonic e NewPing.

Este trabalho teve como propósito comparar a performance destes sensores e de suas respectivas codificações uma vez que a literatura analisada não apresenta algum processo de calibração ou mesmo um trabalho comparativo do desempenho destes sensores ou de suas codificações.

Neste estudo, a estrutura do código envolveu o uso da média móvel com uma amostra de 20 valores coletados pelo sensor, reduzindo a interferência de ruídos espúrios. Na análise da dispersão dos dados, empregamos a incerteza da repetibilidade, da estatística da leitura e da escala do instrumento de medida padrão da distância entre o sensor e o obstáculo.

Os dados mostram que o sonar LVEZ tem uma restrição de leitura para distâncias menores que 20 cm. Quando opera no modo analógico o seu desempenho não é bom ao contrário da operação no modo PWO. Portanto, recomendamos o seu uso para sistemas de mapeamento de ambientes que requer alcances elevados. As incertezas do sensor se mostraram constantes e pequenas, o que denota baixa dispersão de dados. Os sensores US100 e o HC-SR04 mostraram comportamentos similares, apresentando uma exatidão razoável ao longo da faixa estudada, de 5cm a 60 cm, e uma baixa dispersão de dados. O uso das bibliotecas não interferiu no desempenho do sensor e a sua escolha depende apenas do tamanho do arquivo no microcontrolador.

Esperamos que este estudo sirva como guia para a escolha de sensores de sonar para a robótica, mostrando a importância do processo de calibração para um desempenho significativo do sensor dentro da proposta do projeto a que está inserido, e na escolha do sensor. Tais informações são relevantes para o desenvolvimento do sistema, tornando-o

robusto, seja no aspecto de software como de hardware e que devem atender às exigências do projeto.

REFERÊNCIAS

CHANG, Ruijie e colab. Underwater object detection with efficient shadow-removal for side scan sonar images. 2016, Shangai, China: [s.n.], 2016. p. 7–11.

CONTE, G e ZANOLI, S M e GAMBELLA, L. Acoustic Mapping and Localization of an ROV. Jun 2006, Ancona - Italy: IEEE, Jun 2006. p. 1–6. Disponível em: <http://ieeexplore.ieee.org/document/4125012/>.

EVANS, Brian. **Beginning Arduino Programming**. Berkeley, CA: Apress, 2011. Disponível em: <http://link.springer.com/10.1007/978-1-4302-3778-5>.

FALLON, Maurice F e colab. Efficient AUV navigation fusing acoustic ranging and side-scan sonar. Maio 2011, Shangai, China: IEEE, Maio 2011. p. 2398–2405. Disponível em: <http://ieeexplore.ieee.org/document/5980302/>.

GUM, I S O e ABNT, I e SBM., P.R.H.M. **Guia para a expressão da incerteza de medição**. . Duque de Caxias, RJ: [s.n.], 2003. Disponível em: https://www.gov.br/inmetro/pt-br/centrais-de-conteudo/publicacoes/documentos-tecnicos-em-metrologia/gum_final.pdf/view.

MARGOLIS, Michael. **Make an Arduino Controlled Robot**. 1ª ed. Gravenstein Highway North, Sebastopol, CA: Make: Community, 2013.

MCROBERTS, Michael. **Arduino Básico**. São Paulo: Novatec Editora Ltda., 2011.

MONK, Simon. **Arduino+ Android Projects for the Evil Genius**. [S.l.]: McGraw-Hill Companies, Inc., 2002.

NUSENZVEIG, H. Moyses. **Curso de Física Básica: fluidos, oscilações e ondas e calor**. 5ª ed. [S.l.]: Editora Blucher, 2014.

SCHMIDT, Maik. **A Quick-Start Guide**. [S.l.: s.n.], 2011. Disponível em: <http://www.haroonbaig.com>.

SCHWARTZ, Marco. **Internet of Things with Arduino Cookbook**. Birmingham, UK: Packt Publishing Ltd., 2016.

WARREN, John-David e ADAMS, Josh e MOLLE, Harald. **Arduino Robotics**. [S.l.]: Apress, 2008.

ZIEMANN, Volker. **A hands-on course in sensors using the Arduino and Raspberry Pi**. Boca Raton - Florida, US: CRC Press, 2018.

“Os autores declaram estar cientes quanto a responsabilidade pelo conteúdo do artigo.”