

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

JONAS CRAUSS RODRIGUES DE FREITAS

Sensores Inerciais e Protocolo ZigBee no Desenvolvimento de
Tecnologia Assistiva para Reabilitação Motora dos Membros
Superiores Humanos

Trabalho de Graduação.

Prof. Dr. Alexandre Balbinot
Orientador

Porto Alegre, julho de 2013.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitor: Prof. Carlos Alexandre Netto
Vice-Reitor: Prof. Rui Vicente Oppermann
Pró-Reitora de Graduação: Profa. Valquiria Link Bassani
Diretor do Instituto de Informática: Prof. Flávio Rech Wagner
Chefe do Departamento de Engenharia Elétrica: Roberto Petry Homrich
Coordenador da COMGRAD ECP: Prof. Marcelo Götz
Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

AGRADECIMENTOS

A minha família por toda a estrutura e apoio que me deu durante todo o meu percurso, estando perto ou longe deles.

Aos meus amigos por se mostrarem presentes em todos os momentos e entenderem os momentos em que tive que me ausentar.

Aos meus professores, que são os principais responsáveis pelo meu sucesso durante a faculdade.

Ao meu orientador por me auxiliar no desenvolvimento deste projeto.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS.....	6
LISTA DE FIGURAS.....	7
LISTA DE TABELAS.....	9
RESUMO.....	10
ABSTRACT.....	11
1 INTRODUÇÃO.....	12
2 MONITORANDO O MOVIMENTO DE MEMBROS DO CORPO.....	14
2.1 Modelo Mecânico do Membro Superior Humano.....	15
2.2 Modelo de Denavit e Hartenberg.....	16
2.3 Ângulos Euler e de Tait Bryan.....	19
3 SENSORES INERCIAIS E MODELO VIRTUAL.....	21
3.1 Acelerômetros.....	21
3.2 Giroscópios.....	21
3.3 Magnetômetros.....	22
3.4 Sistema ArduIMU V3.....	22
3.5 Modelo Virtual.....	23
4 TRANSMISSÃO DE DADOS ZIGBEE.....	25
5 METODOLOGIA EXPERIMENTAL.....	29
5.1 Aparato Experimental.....	29

5.1.1	ArduIMU	31
5.1.2	XBee	33
5.1.3	Blender	33
5.2	Conexão Xbee - ArduIMU	36
5.3	Conversão dos Dados dos Sensores em Projeção de Eixos	36
5.4	Aplicação do Modelo de Denavit e Hartenberg	39
5.4.1	Solução do sistema corpo - braço.....	41
5.4.2	Solução do sistema braço – antebraço.....	42
5.4.3	Solução do sistema antebraço - mão.....	43
5.5	Aplicação dos Ângulos de Tait Bryan	43
5.6	Calibração dos Sensores.....	46
5.7	Programa Python no Computador.....	47
5.8	Verificação do movimento.....	49
6	RESULTADOS E DISCUSSÕES	51
6.1	Prototipagem	51
6.2	Calibração	53
6.3	Análise da Matriz de Projeções.....	56
6.4	Análise da Conversão da Matriz de Projeções em Ângulos.....	60
6.5	Funcionamento do Modelo Virtual	60
6.6	Verificação do Movimento	61
7	CONCLUSÕES	62
8	PROPOSTAS DE TRABALHOS FUTUROS.....	63
	REFERÊNCIAS.....	64
	APÊNDICE A <MATRIZES DE DENAVIT E HARTENBERG>.....	66
	ANEXO A.....	69
	ANEXO B.....	73

LISTA DE ABREVIATURAS E SIGLAS

AVC	Acidente Cardiovascular
FFD	<i>Full Function Device</i>
IEE	Instrumentação Eletro-Eletrônica
IEEE	Instituto de Engenheiros Eletricistas e Eletrônicos
MEMS	Microeletromecânicos
OpenGL	<i>Open Grapichs Library</i>
P2P	Par a par
PAN	<i>Personal Area Network</i>
RFD	<i>Reduced Function Device</i>
UART	<i>Universal Asynchronous Receiver/Transmitter</i>
UFRGS	Universidade Federal do Rio Grande do Sul
VANT	Veículo Aéreo Não Tripulado
WAN	<i>Wireless Area Network</i>
WPAN	<i>Wireless Personal Area Network</i>

LISTA DE FIGURAS

Figura 2.1. Análise de tipos de movimentos.	14
Figura 2.2. Posição de repouso dos membros superiores.....	15
Figura 2.3. Representação genérica do membro superior humano esquerdo.....	16
Figura 2.4. Parâmetros de Denavit e Hartenberg.....	17
Figura 2.5. Representação gráfica dos ângulos de Euler	19
Figura 2.6. Rotação de Tait Bryan intrínseca na ordem Z, Y', X"	20
Figura 2.7. Exemplo de rotação extrínseca a ordem Z,X,Z	20
Figura 3.1. Foto da placa ArduIMU V3.....	22
Figura 3.2. Exemplo de modelo criado no MakeHuman.....	24
Figura 4.1. Exemplos de arquiteturas estrela e P2P	26
Figura 4.2. Módulo XBee.....	27
Figura 4.3. Transmissão de dados via XBee em modo transparente	27
Figura 4.4. Transmissão de dados via XBee em modo API.....	28
Figura 4.5. Formato do pacote enviado pelo XBee.....	28
Figura 5.1. Fluxograma do caminho de dados	30
Figura 5.2. Módulos do sistema.....	31
Figura 5.3 Eixos de rotação do ArduIMU.....	32
Figura 5.4 Software de teste para ArduIMU.....	32
Figura 5.5. Sistema de transmissão de dados sem fio.....	33
Figura 5.6. Ambiente de programação Blender.....	34
Figura 5.7. Corpo humano representado por bones	34
Figura 5.8. <i>Bones</i> utilizados para representar o membro superior	35
Figura 5.9. Eixos do membro superior humano esquerdo no modelo Blender	35
Figura 5.10. Eixo dos acelerômetros em relação a aceleração da gravidade.....	37
Figura 5.11. Ângulo entre medida do magnetômetro e eixo Y do sistema.....	38
Figura 5.12. Braço posicionado na configuração vertical.....	38
Figura 5.13. Representação de ligação pivô.....	39
Figura 5.14. Modelo de articulações do braço humano em posição de repouso.....	39
Figura 5.15. Exemplo de goniômetro	47
Figura 5.16. Fluxograma do programa Python.....	48
Figura 6.1. Protótipo do sensor e transmissor sem fio.....	51
Figura 6.2. Módulo receptor a ser conectado no computador.....	52
Figura 6.3. Protótipos em funcionamento para teste.....	52
Figura 6.4. Dados recebidos dos protótipos em função de teste.....	53
Figura 6.5. Método de calibração do acelerômetro no eixo X e Y.....	54
Figura 6.6. Método de calibração do acelerômetro no eixo Z.....	55
Figura 6.7. Fixação para calibração no eixo Z.....	55
Figura 6.8. Valor do acelerômetro no eixo Y em relação ao ângulo α	56

Figura 6.9. Valor do acelerômetro no eixo Y em relação ao cosseno do ângulo α	56
Figura 6.10. Matriz de projeções com a placa na sua posição original.	57
Figura 6.11. Matriz de projeções com a placa rotacionada 45° sobre o eixo Z.	57
Figura 6.12. Matriz de projeções com a placa rotacionada 45° sobre o eixo X.	58
Figura 6.13. Montagem para medição de ângulos de rotação do sistema.	58
Figura 6.14. Ângulo principais de rotação.	59
Figura 6.15. Gráfico da rotação sobre o eixo Z medida pelo sistema na horizontal.	59
Figura 6.16. Gráfico da rotação sobre o eixo Z medida pelo sistema na vertical.	59
Figura 6.17. Teste experimental do sistema.	60
Figura 6.18. Movimento do usuário a esquerda e movimento padrão a direita.	61

LISTA DE TABELAS

Tabela 4.1. Especificações XBee.....	26
Tabela 5.1. Conversão de eixos do modelo de Denavit e Hartenberg para o Blender....	40
Tabela 5.2. Parâmetros de Denavit e Hartenberg extraídos	40
Tabela 6.1. Configuração da rede Xbee.	52
Tabela 6.2. Teste de eficiência da rede XBee	53
Tabela 6.3. Resultado da calibração dos acelerômetros.....	56

RESUMO

O presente trabalho tem como objetivo mostrar o desenvolvimento de uma ferramenta de baixo custo e fácil utilização que pode ser utilizada para assistir exercícios de reabilitação motora. Pacientes que sofreram alguma lesão ou mesmo um AVC, normalmente precisam realizar um longo tratamento para readquirirem por completo as suas habilidades motoras. Fazem parte desse tratamento, exercícios envolvendo movimentos bastante complexos e difíceis de serem reproduzidos sem o auxílio de um profissional, como por exemplo, o movimento chamado de diagonal funcional de Kabot, de interesse desse trabalho. Contudo, manter um profissional especializado auxiliando o paciente durante todo o tratamento pode se tornar muito caro, e muitas vezes incômodo para o paciente, devido à necessidade de se deslocar até o ambiente hospitalar.

A necessidade de uma ferramenta assistiva para esses exercícios então se faz presente. Portanto, foi desenvolvida uma ferramenta robusta, de baixo custo e de código aberto, baseada em sensores inerciais. Esta ferramenta é capaz de caracterizar o movimento realizado pelo paciente e compará-lo com um movimento pré-definido e, então, verificar a correta execução ou não do movimento.

Através dessa ferramenta, o movimento do usuário é capturado, armazenado em um banco de dados e comparado com um movimento padrão previamente definido. Isso permite que o movimento do usuário seja analisado posteriormente por um profissional. Além disso, um modelo virtual animado pelo software Blender representa o movimento realizado pelo usuário tornando a interface bastante amigável. O erro estático médio de medida do sistema é de $0,97^\circ$, o que representa um número baixo para um sistema de baixo custo e para a aplicação desejada.

Palavras-Chave: Sensores inerciais, movimento humano, acelerômetro, reabilitação motora, diagonal funcional de Kabot, Blender.

Inertial Sensors and ZigBee Protocol in the Development of an Assistive Technology for Motor Rehabilitation of the Human Upper Limb

ABSTRACT

The following paper has as goal to show the development of a low-cost and easy usage tool that can be used to assist motor rehabilitation exercises. Patients who have suffered an injury or even a stroke, usually need to perform a long treatment to completely regain their motor skills. Exercises involving quite complex and difficult movement to be reproduced without the assistance of a professional are part of this treatment, such as, for example, the movement called Kabot's functional diagonal. However, maintaining a specialist assisting the patient throughout the treatment can become very expensive, and often uncomfortable for the patient, due to the need to displacement to a hospital setting.

The need for an assistive tool for those exercises is present. Therefore, a robust, low cost and open source based on inertial sensors tool was developed. This tool is able to characterize the movement performed by the patient and compare it with a standard motion, and then verify the correct execution of the movement or not.

Through this tool, the user's movement is captured, stored in a database and compared with a standard motion. This allows the user's movement to be analyzed by a professional later. In addition, a virtual model is animated by the software Blender, representing the movement performed by the user, making the interface very user friendly. The average static error of measurement is $0,97^\circ$, which is quite satisfactory for a low-cost system and for the desired application.

Keywords: Inertial sensors, human motion, accelerometer, motor rehabilitation, Kabot's functional diagonal, Blender.

1 INTRODUÇÃO

A busca pela medição de movimentos dos membros humanos tem sido alvo de pesquisas científicas nas últimas décadas. Pessoas que sofreram traumas e que possuam necessidade de realizar exercícios constantes de reabilitação física, como pessoas que sofreram um AVC (Acidente Vascular Cerebral), acidentes de carro, entre outros, necessitam de auxílio médico especializado para readquirir as suas funções motoras normais. O caminho para a total reabilitação é longo e intenso, e requer exercícios de repetição por semanas ou mesmo meses [Zeng et al. 2005].

A realização desse tratamento em um ambiente hospitalar, ou mesmo simplesmente acompanhado de um profissional da área, é muito caro, pois requer muito tempo destes profissionais. Entretanto, se o paciente realizar o exercício de forma autônoma, é bastante provável que ele os execute de forma errônea, o que impossibilitará a sua correta reabilitação.

Uma solução para este problema é utilizar uma ferramenta assistiva que possibilitará a correta execução dos exercícios. Em função disso, pretende-se criar um dispositivo capaz de caracterizar o movimento realizado pelo paciente, compará-lo com o movimento correto do exercício que ele deve realizar, e informá-lo se está executando-o corretamente. Desta forma, o paciente será capaz de realizar o tratamento de forma autônoma e correta, sem a necessidade de deslocamento para um ambiente hospitalar e sem necessitar de um profissional o acompanhando. Além disso, o profissional poderá analisar o resultado do tratamento do paciente através dos dados armazenados no banco de dados e então fornecer um *feedback* para o paciente.

Portanto, o presente trabalho visa desenvolver uma ferramenta assistiva capaz de caracterizar o movimento de um braço humano em 3D. Algumas necessidades devem ser obrigatoriamente satisfeitas por essa ferramenta. São elas:

- o sistema deve perceber os movimentos realizados pelo braço do usuário em 3 dimensões;
- o movimento deve gerar dados em tempo real para comparação instantânea com um movimento pré-definido;
- os dados gerados devem ser armazenados em um banco de dados para que o movimento realizado possa ser recriado para análise posterior;
- o sistema deve ser compacto para que não se torne incômodo de ser utilizado pelo usuário e que não limite os seus movimentos;
- o sistema deve ser robusto o suficiente para ser utilizado em ambientes não controlados, como por exemplo, na residência do usuário;

- o sistema deve ser de baixo custo, permitindo que cada paciente possa ter uma unidade própria em sua residência.

2 MONITORANDO O MOVIMENTO DE MEMBROS DO CORPO

A análise do movimento do corpo humano pode ser dividida em duas categorias: estática, que estuda tudo relacionado aos sistemas sem movimento, e dinâmico, que estuda os sistemas em movimento. A categoria dinâmica pode ainda ser dividida em outras duas subcategorias: cinemática, estuda o deslocamento em tempo e espaço, e cinética, estuda as forças agindo no sistema que influenciam no movimento. A cinemática ainda pode ser dividida em linear, movimento retilíneo ou curvo, onde todos os pontos de um corpo se deslocam igualmente, e angular, que envolve rotação ao redor de um ponto fixo e velocidade diferentes ao longo do corpo. A Figura 2.1 apresenta de forma clara e sucinta as áreas supracitadas da análise do movimento humano (Hamill, 2003).

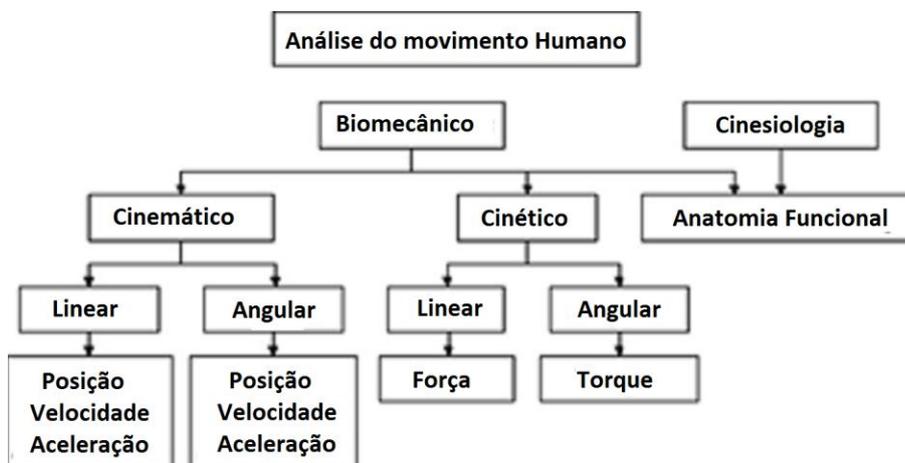


Figura 2.1. Análise de tipos de movimentos (adaptado de Hamill, 2003).

No caso da caracterização do movimento do membro superior, o objetivo desse trabalho é apenas a análise cinemática do movimento. Não será feita uma análise das forças envolvidas no movimento. As articulações a serem analisadas serão as seguintes: ombro, cotovelo e pulso. Devido a grande complexidade de movimentos possíveis com os dedos da mão, esses movimentos foram considerados fora do escopo desse trabalho. A simples ação de abrir e fechar a mão é relevante para os exercícios de reabilitação, mas como é necessário uma abordagem diferente para caracterizar esse movimento, ele foi considerado como fora do escopo desse trabalho.

Vale destacar também que em todas as articulações existem pequenos movimentos de translação e, em algumas, o movimento de translação é significativo, como no caso do ombro (Aggarwal, J.K, 1999). Contudo esses movimentos não serão levados em conta para os exercícios de reabilitação considerados nesse trabalho. O

grande interesse desse trabalho é analisar os movimentos de rotação dessas articulações. Dessa maneira, é necessário apenas caracterizar a inclinação de cada membro (braço, antebraço e mão) para obter a posição completa do membro superior no espaço.

2.1 Modelo Mecânico do Membro Superior Humano

Para caracterizar os movimentos possíveis de cada junta a ser analisada (ombro, cotovelo e pulso), é necessário modelar as rotações possíveis de cada uma dessas juntas através de ligações mecânicas. Tendo como base o estudo realizado por ARENDI (1996), foram identificados 7 graus de liberdade de rotação ao longo do braço humano. Esses graus de liberdade estão distribuídos da seguinte maneira:

- ombro: 3 graus de liberdade de rotação (esferóide);
- cotovelo: 2 graus de liberdade de rotação (elipsóide);
- pulso: 2 graus de liberdade de rotação (elipsóide).

As articulações foram representadas com o membro superior humano em sua posição de repouso, ou seja, estendido na vertical, paralelo ao corpo com a parte interna da mão voltada para o corpo, como mostrado na Figura 2.2.

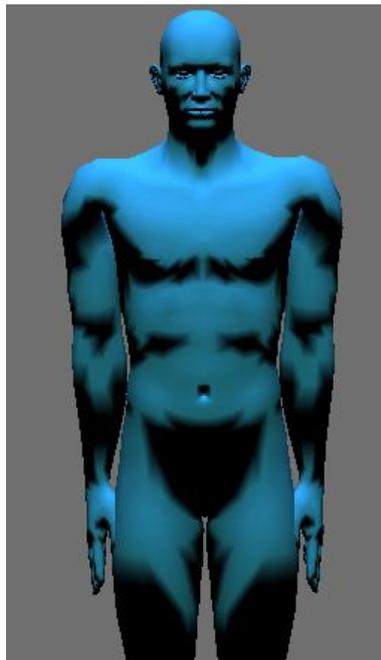


Figura 2.2. Posição de repouso dos membros superiores.

O membro superior esquerdo pode então ser modelado em sua posição de repouso como mostrado na Figura 2.3:

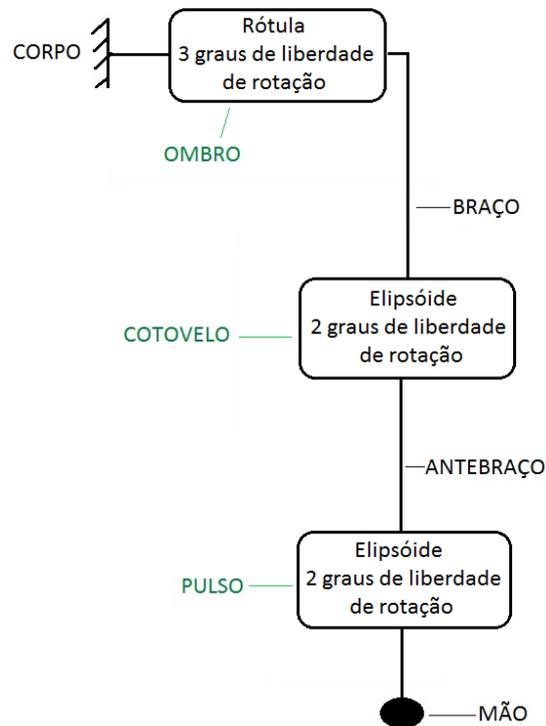


Figura 2.3. Representação genérica do membro superior humano esquerdo.

2.2 Modelo de Denavit e Hartenberg

Denavit e Hartenberg estabeleceram em 1955 uma convenção para a representação de juntas mecânicas e é, hoje em dia, amplamente utilizada na área de robótica. Essa convenção define um eixo para cada ligação e permite descobrir, de maneira inambígua, a posição e a configuração de cada ligação e braço mecânico representado. Existem, contudo, algumas regras a serem seguidas na representação de Denavit e Hartenberg. As ligações só podem ser de dois tipos, pivô ou prismática. Uma ligação pivô admite apenas um grau de liberdade de rotação e nenhum grau de liberdade de translação enquanto uma ligação prismática admite apenas um grau de liberdade de translação (RIGAUD, 2011).

Cada ligação possui os seus próprios eixos de referências, que precisam ser definidos seguindo as seguintes regras:

- sendo O_i a origem da ligação i , X_i o eixo X da ligação i e A_i o eixo de rotação da ligação;
- O_i está situado sobre A_i e sobre a perpendicular comum entre A_i e A_{i-1} ;
- X_i tem como suporte a perpendicular comum entre A_i e A_{i-1} e está orientado de A_{i-1} para A_i ;
- Z_i tem como suporte o eixo A_i e sua orientação é arbitrária;
- O vetor Y_i resulta do produto vetorial entre X_i e Z_i .

O próximo passo para completar o modelo de Denavit e Hartenberg é definir os parâmetros de Denavit e Hartenberg. Os parâmetros ' α ' e ' θ ' são responsáveis por descrever a inclinação dos eixos da próxima articulação e os parâmetros ' r ' e ' a ' são

responsáveis por definir a posição da origem da próxima articulação. No caso desse projeto, é desejado representar apenas as angulações dos eixos, não existindo interesse em calcular a posição da origem dos eixos. Portanto, apenas os parâmetros ‘ α ’ e ‘ θ ’ serão identificados. Para definir tal parâmetros, é preciso seguir as seguintes regras:

- α_i é o ângulo entre Z_i e Z_{i+1} medido sobre o eixo X_{i+1} ;
- θ_i é o ângulo entre X_i e X_{i+1} medido sobre o eixo Z_i .

A Figura 2.4 representa graficamente as definições dos parâmetros de Denavit e Hartenberg.

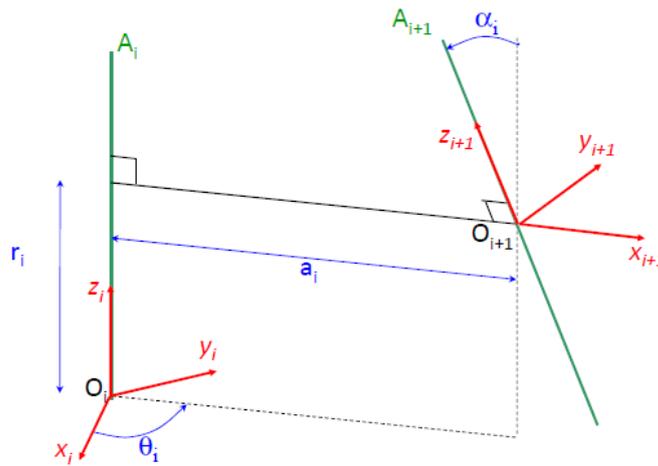


Figura 2.4. Parâmetros de Denavit e Hartenberg (RIGAUD, 2011).

Para se obter a descrição dos eixos da segunda articulação sobre os eixo de referência em função de θ_i , utiliza-se a matriz de passagem homogênea $T_{1,2}$. E assim é feito sucessivamente, para se obter a descrição dos eixos da articulação $i+1$ em função dos eixos da articulação i utiliza-se a matriz de passagem homogênea $T_{i,i+1}$, que pode ser vista na equação (2.1).

$$T_{i, i+1} = \begin{pmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & r_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.1)$$

Essa matriz é extremamente útil pois ela é capaz de descrever completamente os eixos da articulação $i+1$ em função dos eixos da articulação i . No caso, as 3 primeiras linhas da primeira coluna descrevem o eixo X_{i+1} em função de X_i , Y_i e Z_i respectivamente. Na segunda coluna, o mesmo ocorre para descrever o eixo Y_{i+1} e na terceira coluna, o eixo Z_{i+1} . Na última coluna, os valores das 3 primeiras linhas correspondem à posição do ponto O_{i+1} em função das coordenadas X, Y e Z relativas ao ponto O_i . Portanto, é possível descrever completamente os eixos da articulação $i+1$. Como no caso desse projeto o interesse é unicamente descrever a angulação dos eixos, a

quarta coluna pode ser desprezada dos cálculos. Isso implica que a quarta linha da matriz também pode ser desprezada, pois ela só existe para que a matriz seja quadrada. Dessa maneira, a matriz da equação (2.1) é simplificada para uma matriz 3x3 representada na equação (2.2).

$$\underline{T}_{i, i+1} = \begin{vmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i \\ 0 & S\alpha_i & C\alpha_i \end{vmatrix} \quad (2.2)$$

Essa matriz também pode ser utilizada para a descrição dos eixos de qualquer articulação em função dos eixos de uma articulação anterior e de todas as variáveis intermediárias. Isso pode ser feito de acordo com (2.3).

$$T_{i,i+2} = T_{i,i+1} \cdot T_{i+1,i+2} \quad (2.3)$$

A descrição dos eixos de articulações posteriores sobre o eixo de referência e em função da posição das articulações anteriores é chamada de Modelo Geométrico Direto.

Uma outra funcionalidade interessante do modelo de Denavit e Hartenberg é o Modelo Geométrico Inverso. Esse modelo nos permite calcular os valores dos ângulos θ_i partindo da descrição dos eixos posteriores em função dos eixos de referência. Para criar esse modelo é utilizada a matriz de passagem homogênea inversa, representada na equação (2.4).

$$\underline{T}_{i+1, i} = \begin{vmatrix} C\theta_i & S\theta_i & 0 & -a_i \\ -C\alpha_i S\theta_i & C\alpha_i C\theta_i & S\alpha_i & -r_i S\alpha_i \\ S\alpha_i S\theta_i & -S\alpha_i C\theta_i & C\alpha_i & -r_i C\alpha_i \\ \hline 0 & 0 & 0 & 1 \end{vmatrix} \quad (2.4)$$

Esse é o modelo necessário para controlar um movimento robótico em função da sua posição final. Para encontrar a solução analítica desse problema, é necessário seguir 3 passos:

- primeiro passo: utilizando o modelo geométrico direto, pode-se calcular $T_{1,n}$, $T_{2,n}$, $T_{3,n}$, e assim por diante;
- segundo passo: dada uma posição final expressa através da matriz $T^*_{1,n}$, calcula-se $T^*_{2,n}$, sabendo que $T^*_{2,n} = T_{2,1} \cdot T^*_{1,n}$. Segue-se a mesma lógica para calcular $T^*_{3,n}$ e assim sucessivamente;
- terceiro passo: igualar as matrizes $T_{x,n}$ e $T^*_{x,n}$ e tentar encontrar termos que sejam função de apenas uma variável para facilitar a sua resolução. Continuar com essa ideia até encontrar todas as soluções.

Contudo, esse método não é sempre explorável, pois pode ocorrer que o sistema não tenha solução devido a graus de liberdade insuficientes no sistema. Um outro problema que pode ser encontrado também é que o sistema tenha infinitas soluções devido a redundâncias nas articulações.

2.3 Ângulos Euler e de Tait Bryan

Os ângulos de Euler foram criados por Leonhard Euler para descrever a orientação espacial de corpos rígidos. Através dos ângulos de Euler, é possível descrever uma rotação tridimensional qualquer em relação a um sistema de coordenadas cartesianas absoluto em função de três parâmetros (BRITS, 2009).

Dado o sistema de coordenadas de referência, realiza-se uma rotação sobre o eixo z no valor de α graus. Em seguida, realiza-se uma rotação no valor de β graus sobre o eixo N, resultado da rotação do eixo x em torno de z. Por último, é realizada uma terceira rotação no valor de γ graus sobre o eixo Z, resultado das últimas duas rotações anteriores. Os ângulos α , β e γ são chamados de ângulos de Euler.

Esse sistema de caracterização da orientação de um corpo no espaço é amplamente utilizado na aeronáutica e na computação gráfica (FAUGERAS, 1993). Uma representação gráfica dos ângulos de Euler podem ser vistos na Figura 2.5.

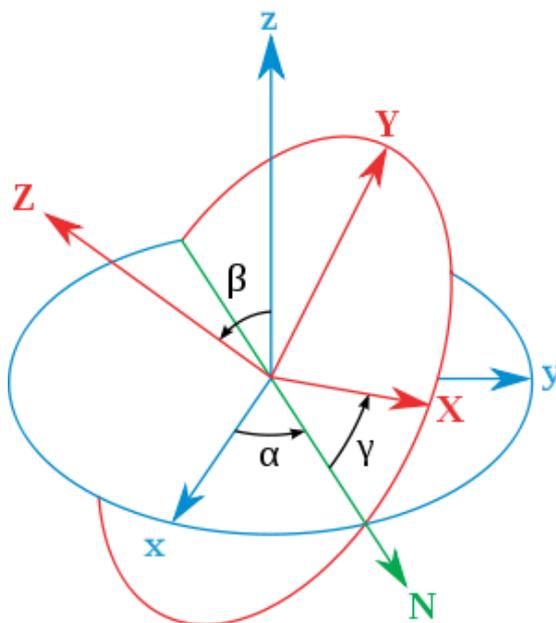


Figura 2.5. Representação gráfica dos ângulos de Euler (BRITS, 2009).

Os ângulos de Tait Bryan são muito semelhantes aos ângulos de Euler, eles também são capazes de descrever qualquer rotação tridimensional de um sistema de coordenadas. Contudo, ao invés de realizar as rotações sobre o eixo z, N, e Z, os ângulos de Tait Bryan são mais genéricos, permitindo que a ordem dos eixos de rotação seja arbitrária, mas as rotações devem ocorrer sobre os três eixos, Z, Y e X. Dessa maneira, existem 6 possibilidades de ordem de rotações para os ângulos de Tait Bryan.

Quanto aos eixos de rotação, eles podem se referir tanto aos eixos rotacionados quanto aos eixos absolutos. No caso em que eles se referem aos eixos rotacionados, a rotação é chamada de rotação intrínseca. No caso em que os eixos de rotação não se alteram, ou seja, a rotação é sempre realizada sobre os eixos absolutos, a rotação é chamada de rotação extrínseca.

Pode-se ver um exemplo gráfico de rotação intrínseca na ordem z, N(y') e X na Figura 2.6. Os ângulos de rotação estão representados por ψ , ϕ e θ respectivamente.

Um exemplo de rotação extrínseca pode ser visualizado na Figura 2.7, onde são realizadas rotações sobre o eixo Z, X e novamente sobre o eixo Z absolutos. Nesse caso os eixos de rotação não são alterados. Os ângulos de rotação do exemplo da Figura 2.7 não são ângulos de Tait Bryan pois ocorrem duas rotações sobre o eixo Z.

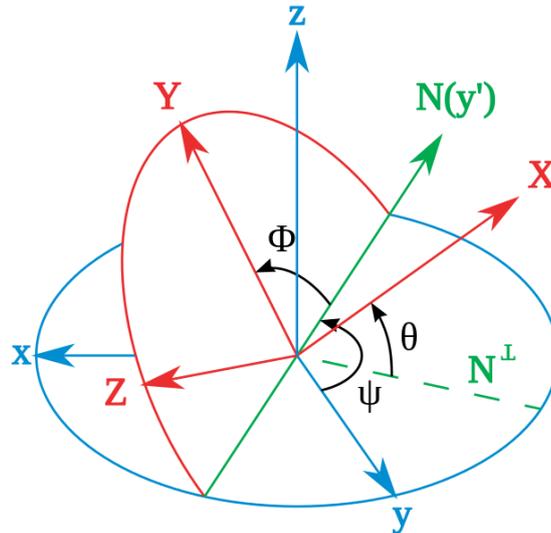


Figura 2.6. Rotação de Tait Bryan intrínseca na ordem z, N, x (BRITS, 2009).

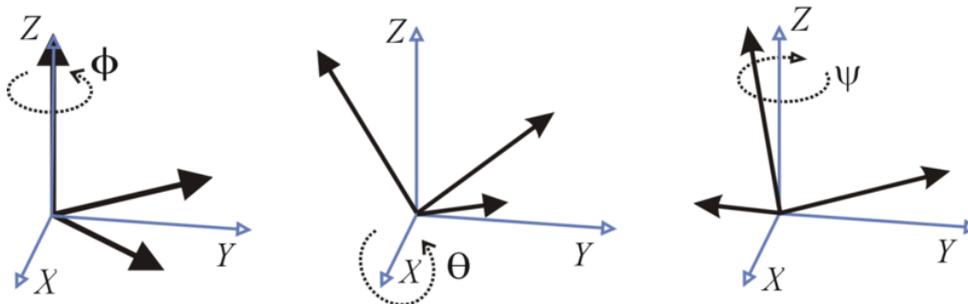


Figura 2.7. Exemplo de rotação extrínseca a ordem Z,X,Z (MALAN, 2004).

3 SENSORES INERCIAIS E MODELO VIRTUAL

É possível caracterizar o movimento humano através de marcadores em locais específicos do corpo monitorados por câmeras, demonstrado por Aggarwal e Cai (1999) e por Welch Foxlin (2002). Contudo, esse sistema sofre do problema de oclusão, ou seja, em certas posições, os marcadores ficarão escondidos e as câmeras simplesmente não irão detectá-lo [Bhatia et al. 2004]. Como os movimentos de reabilitação são bastante complexos, esse problema não é aceitável.

Uma forma eficiente de analisar o movimento realizado é através de sensores inerciais, que medem aceleração (acelerômetros) e velocidade angular (giroscópios). Esses sensores são pequenos, leves e adequados para caracterizar o movimento de membros humanos [Emil, 2005], [Zhou e Hu, 2007]. Acelerômetros já foram utilizados para monitorar o movimento do corpo humano em 1950, mas devido ao seu custo elevado, tornou-se inviável. Com o avanço da tecnologia e a baixa no preço desses sensores, voltaram a ser utilizados no início da década de 70 para medir o movimento do corpo humano e, a partir de então, o seu uso para essa finalidade tem aumentado [Godfrey et al. 2008].

3.1 Acelerômetros

Acelerômetros são sensores inerciais capazes de medir a aceleração resultante em um dado instante de tempo. O princípio de funcionamento da maioria dos acelerômetros é baseado no sistema massa-mola, ou seja, a deformação da mola é proporcional à aceleração resultante naquele eixo. São bastante utilizados para análises cinemáticas biomecânicas. O valor resultante da aceleração pode ser utilizado para que seja encontrada a inclinação do corpo medido (usando a gravidade como referência). Para obter a inclinação do corpo em 3D, é necessário medir a aceleração nos três eixos, seja através de três acelerômetros posicionados ortogonalmente ou um acelerômetro tri-axial, que realiza a mesma função.

Um dos modelos de acelerômetros existentes no mercado, e de baixo custo, é o acelerômetro capacitivo. Este tipo de acelerômetro funciona através da alteração da capacitância com a aceleração. Uma das placas paralelas do capacitor é colocada em uma superfície que se desloca proporcional a aceleração exercida e a outra é fixa. Quando a aceleração sobre a placa varia, a tensão sobre o capacitor também varia, criando a saída do sensor [Godfrey et al, 2008].

3.2 Giroscópios

Giroscópio é um diferente tipo de sensor inercial, capaz de medir a velocidade angular em certo instante. Giroscópios mecânicos, ou micromecânicos (MEMS), funcionam de acordo com a teoria de que quando uma velocidade angular é aplicada a

um corpo, uma força de Coriolis é gerada, proporcional à velocidade angular. Essa força é, então, transformada em sinal elétrico, gerando assim o sinal de saída do sensor. Na mesma lógica dos acelerômetros, podemos utilizar três giroscópios ortogonais para medir a velocidade angular nos três eixos. [Barbour, 2001].

3.3 Magnetômetros

Magnetômetros são sensores capazes de detectar a direção do campo magnético da Terra. Eles são normalmente utilizados em conjunto com giroscópios para corrigir possíveis erros de medida dos giroscópios. Esses sensores devem ser calibrados de acordo com a localidade da sua utilização, pois o norte magnético da Terra não coincide com o Norte geográfico. Além disso, o campo magnético da Terra se altera ano após ano, necessitando assim que a calibração desses sensores sejam ajustadas de tempos em tempos.

No caso particular de Porto Alegre, a inclinação magnética é de $16^{\circ} 8' 56''$ a oeste e está sendo alterada de $9.0'$ a oeste por ano (www.ngdc.noaa.gov). Comparando a direção do norte magnético da Terra com a direção da placa, é possível detectar movimentos de rotação desta.

3.4 Sistema ArduIMU V3

Como a busca de dados relativos ao posicionamento 3D de um objeto é um problema bastante comum, existem circuitos integrados dedicados a isso. O hardware ArduIMU V3 possui um acelerômetro tri-axial (capacitivo MEMS), um giroscópio tri-axial (MEMS) e um magnetômetro tri-axial capaz de medir os movimentos do corpo humano, além de um microcontrolador integrado (Atmega328) capaz de tratar esses dados. Esse hardware é bastante utilizado na aeronáutica para elaboração de pilotos automáticos, mas, por ser bastante compacto (placa de 3,8cm X 2,5 cm) é também utilizada em projetos de monitoramento de movimentos humanos (DIY Drones). Uma foto desse hardware pode ser vista na Figura 3.1.

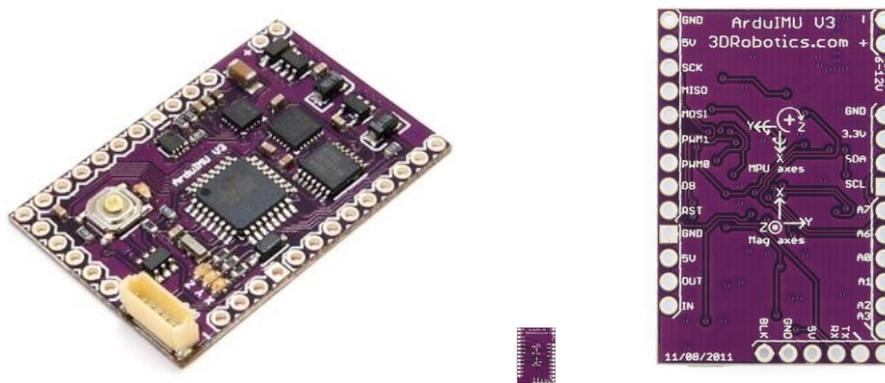


Figura 3.1. Foto da placa ArduIMU V3 (DIY Drones).

Os sensores inerciais possuem conversores AD de 16 bits integrados, e as suas escalas de alcance podem ser digitalmente programadas, tanto no caso do giroscópio (± 250 , ± 500 , ± 1000 , ou ± 2000 %/s) quanto no caso do acelerômetro ($\pm 2g$, $\pm 4g$, $\pm 8g$, ou $\pm 16g$). Além disso, os sensores possuem filtros passa-baixa configuráveis pelo usuário, o que é interessante para minimizar a captação ruídos.

O microcontrolador integrado recebe os dados dos sensores inerciais e é capaz de tratá-los, fornecendo na saída a inclinação da placa em cada eixo em relação a posição de repouso calibrada inicialmente.

O software que é executado nessa placa é de código aberto, o que facilita no desenvolvimento e adaptação para uma aplicação específica. Os dados de saída são enviados por uma porta serial e podem ser programadas no microprocessador em linguagem C.

3.5 Modelo Virtual

A fidelidade do modelo virtual é definida pelo número de polígonos que o compõe, e quanto maior o número de polígonos, maior a necessidade por processamento. Como não é necessário para esse projeto um modelo extremamente fiel do corpo humano, deseja-se que o processamento do modelo possa ser realizado em tempo real para fornecer um *feedback* para o usuário e é necessário que tudo isso possa ser processado e executado no computador pessoal do usuário. Um modelo amigável e que permita identificar com detalhes o movimento que está sendo realizado no seu braço é o suficiente.

A utilização do modelo virtual é extremamente importante por dois motivos: fornecer o *feedback* para o usuário sobre o seu movimento e para permitir uma análise posterior detalhada, por um profissional da área da saúde, dos movimentos realizados. Isso permitirá ao profissional da área da saúde diagnosticar o paciente, relevar as suas dificuldades e avanços e encaminhá-lo para o tratamento mais adequado.

Com a popularização de modelos virtuais, tanto de humanos, como de objetos e mesmo de criaturas surreais, foram criados softwares específicos para o seu desenvolvimento. Alguns desses softwares são de código aberto e baseados na biblioteca gráfica OpenGL. Dessa maneira, com poucos cliques e uma interface amigável, é possível criar um modelo virtual semelhante ao corpo humano.

Criar um modelo virtual pode parecer uma tarefa muito complexa, porém, felizmente, existem ferramentas específicas para esse tipo de tarefa que facilitaram o desenvolvimento deste trabalho. Como se buscam sempre ferramentas de código livre, foram selecionados dois softwares capazes de modelar e animar um modelo humano virtual.

O MakeHuman é um software livre específico para modelar um corpo humano através de malha de polígonos. Com uma interface bastante simples pode-se, além de criar figuras, personalizar humanóides já existentes. As personalizações são das mais diversas, como por exemplo, gênero, peso, altura e até posição das sobrancelhas. Além disso, este software permite exportar um arquivo que contém todas as informações sobre o modelo gerado, incluindo todas as articulações do corpo humano. Pelas políticas do software, este modelo pode ser explorado pelo usuário da maneira que ele desejar. A Figura 3.2 representa o modelo que foi criado no software MakeHuman e que será utilizado para as animações.

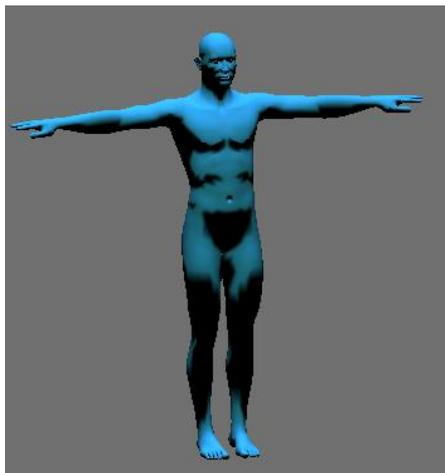


Figura 3.2. Exemplo de modelo criado no MakeHuman.

Para realizar animações sobre este modelo humano, existe o programa Blender. Um software livre capaz de importar o arquivo gerado pelo MakeHuman e modificar as posições das articulações do objeto em sequência para gerar animações. O Blender é capaz de gerar animações de duas maneiras diferentes. Uma delas é através da definição da posição das articulações em determinados quadros de uma sequência, realizando assim um movimento pré-definido pelo usuário.

Uma outra maneira de realizar animações no modelo, e mais interessante no caso desse projeto, é através do envio da posição das articulações através de uma comunicação via sockets. Dessa maneira, o Blender é capaz de realizar o movimento em tempo real e sem ser pré-determinado, gerando uma animação amigável para o usuário.

É importante ressaltar que o modelo virtual utilizado nesse trabalho foi desenvolvido pelo bolsista de iniciação científica do laboratório de Instrumentação Eletro-Eletrônica (IEE), Eduardo Fernandes de Conto (CONTO, 2012).

4 TRANSMISSÃO DE DADOS ZIGBEE

O sistema de transmissão de dados não deve ser incômodo para o utilizador do sistema e nem deve limitar os seus movimentos. A melhor opção é, sem dúvidas, um sistema de transmissão de dados sem fio, o que proporciona total liberdade de movimento ao usuário. Contudo, o hardware ArduIMU possui apenas as saídas analógicas dos sensores e uma saída serial com os dados já tratados. A melhor solução nesse caso seria um módulo com entrada serial, capaz de transmitir os dados de entrada via *wireless* para outro dispositivo conectado ao computador. Contudo, vale lembrar que o dispositivo deve ser compacto e de baixo consumo, pois será alimentado por bateria.

ZigBee é um padrão de comunicação sem fio entre dispositivos eletrônicos de baixa potência, baixa transmissão de dados e baixo custo. A comunicação é realizada através do protocolo IEEE 802.15.4. Este protocolo foi estabelecido especialmente para redes que utilizam dispositivos de curto alcance, baixa taxa de transmissão de dados e baixo consumo de potência (normalmente alimentados por bateria), portanto, ideal para a aplicação deste trabalho.

No protocolo IEEE 802.15.4, estão definidos dois tipos de dispositivos, o *Full Function Device* (FFD) e o *Reduced Function Device* (RFD). Na rede deve existir somente um coordenador. Este coordenador é responsável pela inserção de novos dispositivos na rede. O coordenador deve ser obrigatoriamente um dispositivo FFD. Dispositivos FFD podem gerar e enviar dados ou funcionar como roteadores. Dispositivos RFD podem apenas gerar e enviar dados, não sendo capazes de associar mais nenhum dispositivo ligado a eles, a sua grande vantagem é um consumo reduzido de energia por possuir menos funcionalidades.

Este protocolo prevê a criação de uma rede WPAN (*Wireless Personal Area Network*), que pode possuir duas arquiteturas básicas diferentes: Arquitetura estrela ou P2P, como pode ser visto na Figura 4.1.

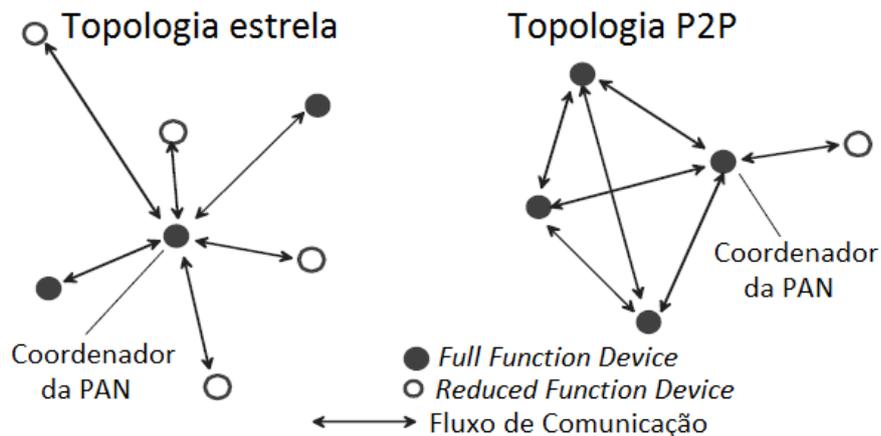


Figura 4.1. Exemplos de arquiteturas estrela e P2P (adaptado de IEEE 802.15.4).

Na arquitetura estrela, todo o fluxo de dados ocorre entre o coordenador da PAN e os dispositivos conectados a ele. Não existe comunicação entre os outros dispositivos. Os dispositivos periféricos podem ser do tipo RFD ou FFD, sendo o coordenador o único obrigado a ser do tipo FFD.

Na arquitetura P2P, os dispositivos são capazes de se comunicar entre si, e os dispositivos FFD podem utilizar a sua capacidade de roteamento. Pode-se, dessa maneira, formar redes mais complexas como Cluster Tree e Mesh. A única comunicação que não é possível é entre dois dispositivos RFD.

O módulo XBee 802.15.4 é um dispositivo para comunicação sem fio, desenvolvido pela Digi International, que utiliza o protocolo IEEE 802.15.4 (Digi International). Este dispositivo é capaz de criar uma WPAN tanto no formato estrela quanto no formato P2P. Suas principais características são: baixo consumo, comunicação serial UART, configuração de gerência de energia, comunicação sem fio. Maiores detalhes sobre as especificações desse dispositivo podem ser encontrados na Tabela 4.1.

Tabela 4.1. Especificações XBee (Digi International).

Desempenho	
Alcance Urbano	30 metros
Alcance em Área Aberta	100 metros
Potência de Transmissão	1mW
Velocidade de Transmissão RF	250 000 bps
Velocidade de Transmissão Serial	1200 – 115 200 bps
Sensibilidade de Recepção	-92 dBm (1% taxa de erro de pacote)
Requisitos de Alimentação	
Tensão de Alimentação	2.8 – 3.4 V
Consumi Típico	50 mA

Continuação da Tabela 4.1:

Gerais	
Frequência de Operação	ISM 2.4 Ghz
Dimensões	2.438 cm x 2.761 cm
Temperatura de Operação	-40 até 85° C
Rede e Segurança	
Topologia Suportada	P2P, estrela
Número de Canais	12 Canais
Opções de Endereçamento	PAN ID, Canal, Endereço
Criptografia	AES 128-bits

Desta maneira, o módulo XBee será capaz de receber dados enviados pela hardware ArduIMU através da sua porta de comunicação serial UART e retransmiti-los a um módulo conectado ao computador capaz de interpretar esses dados. Uma foto de um módulo XBee pode ser visto na Figura 4.2.



Figura 4.2. Módulo XBee (Digi International).

O módulo XBee permite 2 modos de transmissão de dados: transparente e API. No modo transparente, os dados recebidos pela porta UART do módulo conectado aos sensores serão recebidos pelo computador, sem nenhuma informação adicional.



Figura 4.3. Transmissão de dados via XBee em modo transparente.

Fica fácil perceber a limitação desse modo de transmissão de dados, pois, se houver mais de um módulo transmitindo dados, ficará impossível descobrir qual módulo enviou qual dado. Para que isso seja possível, é necessário colocar esse dado dentro de um pacote, onde haja um cabeçalho com dados referentes ao módulo de origem. Dessa maneira, analisando o cabeçalho, é possível retirar informações a respeito do módulo de origem.

O outro modo de transmissão de dados permitido pelo Xbee é o modo API (ver Figura 4.4). Nesse modo, os dados recebidos pelo computador encontram-se dentro de um pacote.



Figura 4.4. Transmissão de dados via XBee em modo API.

O pacote possui o formato representado pela Figura 4.5:

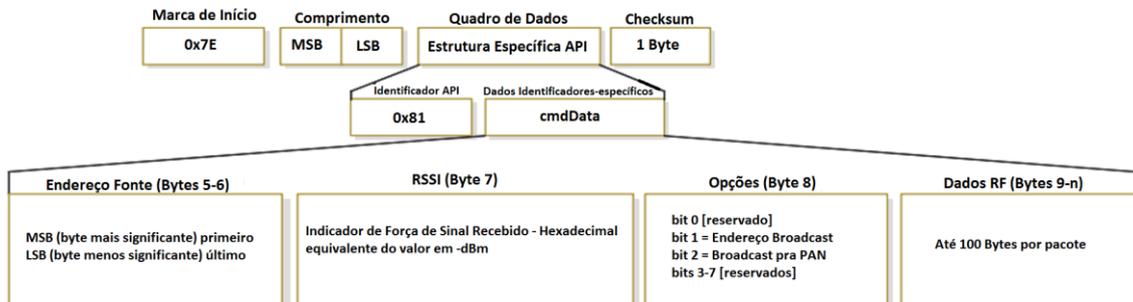


Figura 4.5. Formato do pacote enviado pelo XBee (adaptado de XBee datasheet).

Um símbolo de início de pacote, definido com 0x7E, seguido do tamanho do pacote em bytes (Frame Data + Checksum), em seguida é enviado o identificador do API, o endereço do módulo que enviou o pacote, a potência do sinal recebido em -dBm, um byte para identificar broadcast, os dados efetivamente enviados no pacote e um byte de checksum para validação de consistência do pacote.

Dessa maneira, é possível desmembrar o pacote recebido no computador e analisar os dados recebidos, bem como o endereço de origem daqueles dados. Então é possível identificar a qual membro do corpo corresponde à inclinação recebida.

5 METODOLOGIA EXPERIMENTAL

Tendo em vista as necessidades levantadas por esse projeto, foram procuradas diversas soluções no mercado que pudessem ser úteis para o seu desenvolvimento. Vale lembrar que é de grande importância que os softwares utilizados sejam de código livre para permitir maior flexibilidade ao projeto e manter o seu baixo custo.

Pode-se categorizar a metodologia experimental nas seguintes partes:

- aparato experimental: onde são listados e descritos as ferramentas e equipamentos necessários para a realização desse projeto;
- sensores inerciais: descrição detalhada dos sensores inerciais que serão utilizados, suas configuração e rotinas de medidas;
- transmissão de dados: modo para transmissão de dados com o mínimo de interferência no movimento do usuário, nesse caso utilizando o módulo XBee para transmissão de dados sem fio;
- tratamento de dados dos sensores e conversão em ângulos de rotação;
- modelo virtual: desenvolvimento de um modelo humano virtual através de softwares especializados;
- armazenamento de dados: banco de dados criado para armazenamento das informações para recriação do movimento.

Um fluxograma de todo o caminho dos dados pode ser verificado na Figura 5.1.

5.1 Aparato Experimental

O sistema foi elaborado para ser completamente modular, de modo que, se soluções mais baratas surgirem no mercado, poderão ser implementadas nesse projeto sem necessidade de grandes mudanças. Dessa maneira, esse projeto poderá sempre ser atualizado e se manterá competitivo. Os módulos existentes são divididos em “sensores inerciais”, “transmissão de dados”, “modelo virtual” e “banco de dados”, conforme esboçado na Figura 5.2.

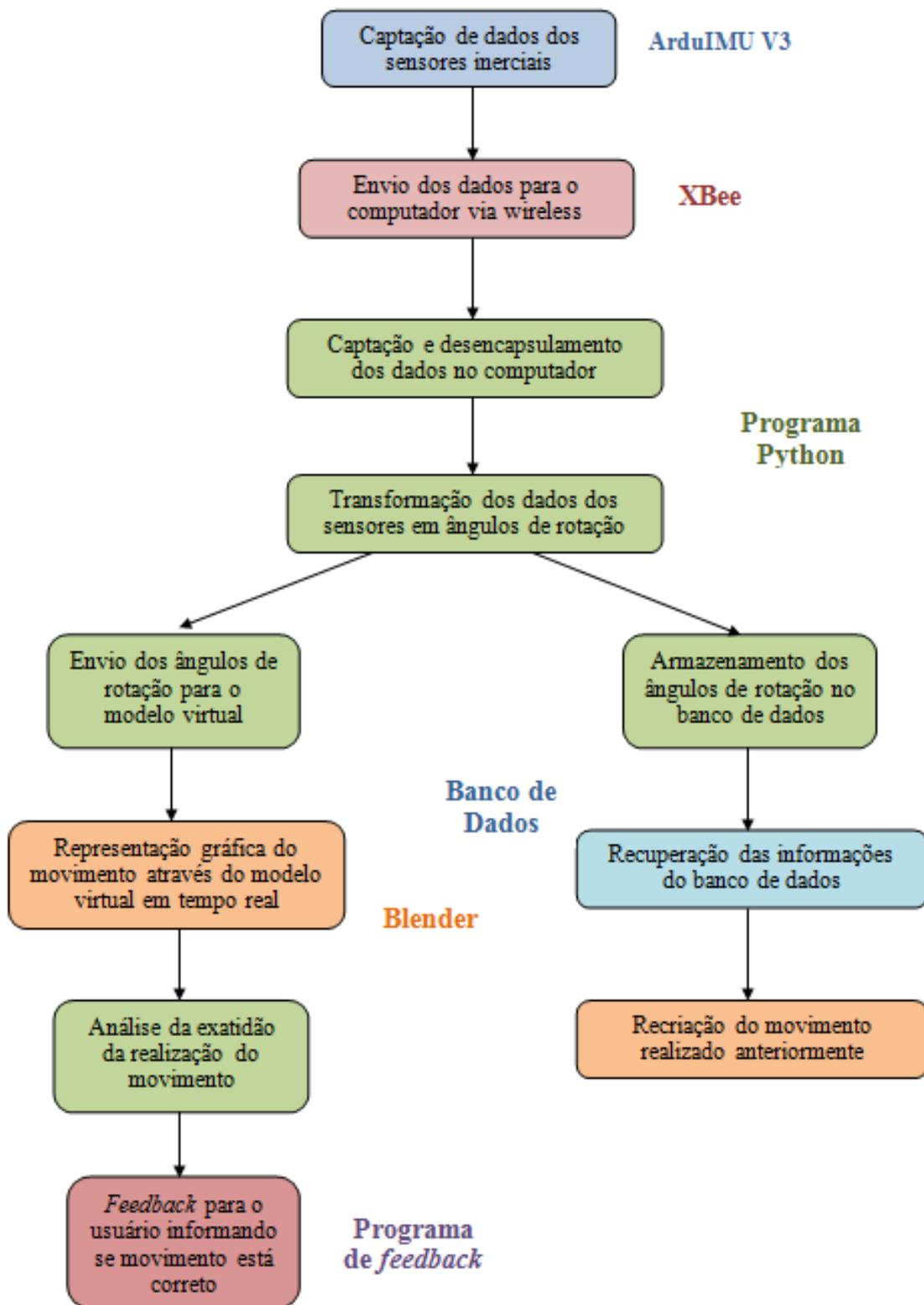


Figura 5.1. Fluxograma do caminho de dados.



Figura 5.2. Módulos do sistema.

O módulo dos sensores inerciais consiste em uma forma de medição da inclinação no espaço 3D dos sensores posicionados no corpo humano. Os sensores devem fornecer a inclinação nos 3 eixos (x, y e z) em relação a uma posição de repouso, a qual foi calibrada na inicialização do sistema. Para realizar essas medições será utilizado um hardware desenvolvido pela empresa DYI Drones, chamado ArduIMU. Esse hardware foi desenvolvido para ser utilizado em VANTs (veículos aéreos não tripulados), mas pode ser utilizado para qualquer aplicação de monitoramento de movimento que necessite de sensores inerciais.

Para a transmissão de dados é necessário algo que interfira o mínimo possível na movimentação do usuário, portanto, foi decidido que a transmissão de dados até o computador será feita através de uma rede sem fio. Existem aparelhos que utilizam protocolos sem fio desenvolvidos especialmente para aplicações que necessitam de baixo consumo e baixa transmissão de dados. O dispositivo utilizado nesse projeto foi o módulo XBee, pois ele atende a todas as especificações e encontra-se disponível no Laboratório IEE.

Para a elaboração de um modelo virtual no computador, serão utilizados programas específicos para modelagem de um corpo humano e suas articulações. Foram escolhidos os programas MakeHuman e Blender pela experiência e sucesso que o Laboratório IEE possui em desenvolver modelos humanos através desses softwares.

O programa a ser desenvolvido para manipulação dos dados e comunicação com os módulos Xbee e Blender será implementado em linguagem Python. Essa escolha foi feita pela facilidade que essa linguagem proporciona na comunicação serial e especialmente na comunicação com o Blender.

5.1.1 ArduIMU

O hardware ArduIMU é capaz de realizar medições de angulação sobre os 3 eixos, x, y e z. Para isso, basta programá-lo com softwares de código livre disponíveis na internet no endereço code.google.com/p/ardu-imu/.

Pode-se ver a representação dos eixos em relação a placa na Figura 5.3.

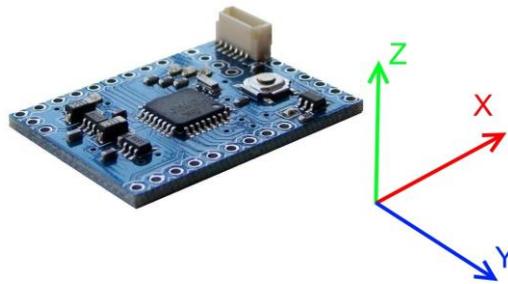


Figura 5.3 Eixos de rotação do ArduIMU (DIY Drones).

Existe também um programa de testes para o ArduIMU chamado “ArduIMU test” desenvolvido por Paul Mather e disponibilizado gratuitamente na internet (Manual ArduIMU). Pode-se ver a interface desse software na Figura 5.4.

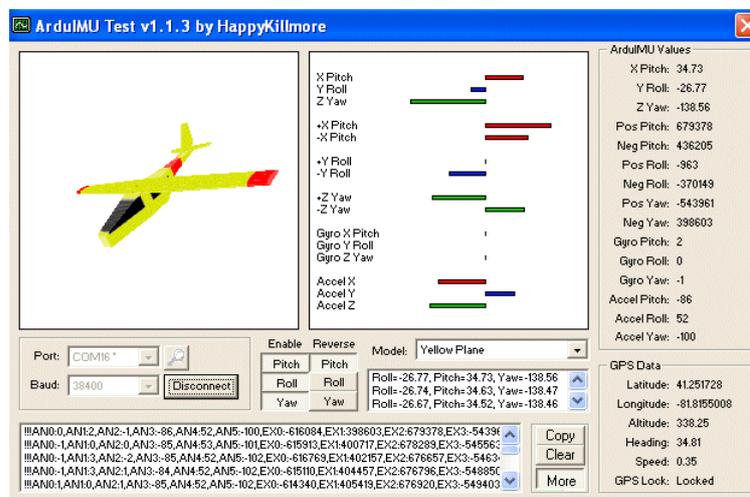


Figura 5.4 Software de teste para ArduIMU (Manual ArduIMU).

Através dessas ferramentas foi possível testar as funcionalidades do hardware. Os resultados obtidos não foram satisfatórios para a medição dos ângulos de um movimento delicado e preciso como são os movimentos de reabilitação motora. As oscilações nas medições foram excessivamente grandes, chegando na ordem de dezenas de graus.

Foi percebido que os valores brutos dos sensores eram praticamente estáticos, o que leva a conclusão de que o problema para oscilações dessa magnitude deveriam estar no código do programa que interpreta esses dados. A partir dessa conclusão, um código novo foi desenvolvido para interpretar os dados dos sensores do ArduIMU.

Como o poder de processamento do computador é muito maior do que o poder de processamento do hardware ArduIMU, os valores dos conversores ADs dos sensores serão enviados para o computador e este irá interpretá-los de modo a convertê-los em inclinações. O pacote de dados enviado para o Xbee será então composto pelos dados dos conversores AD dos 9 sensores inercias (acelerômetro, giroscópio e magnetômetro tri-axiais). Como cada conversor é de 16 bits (2 bytes), o tamanho de dados do pacote será de 18 bytes.

O ArduIMU é capaz de gerar novos dados a uma frequência de aproximadamente 100Hz, ou seja, um período de 10ms. Como o movimento relativo a diagonal funcional de Kabot, assim como a maioria dos exercícios de reabilitação, não é

um movimento muito rápido, essa taxa de amostragem se mostra suficiente. Contudo, se for desejado detectar um movimento mais veloz, seria necessário adquirir um hardware com uma frequência de amostragem mais rápida.

5.1.2 XBee

No caso desse projeto, deseja-se que todos os dados se concentrem no módulo conectado ao computador, os sensores não precisam se comunicar entre si e o consumo de energia desses sensores deve ser o menor possível. Dessa maneira, a arquitetura estrela se mostra a mais vantajosa. Como o coordenador estará conectado ao computador e, portanto, alimentado por uma fonte de energia, não é grave se o consumo de energia dele for elevado. Contudo, os sensores serão alimentados por bateria, o que torna vantajoso o fato de todos eles serem do tipo RFD (IEEE, 2006). A Figura 5.5 esboça como será feita a comunicação entre os diferentes dispositivos do sistema.

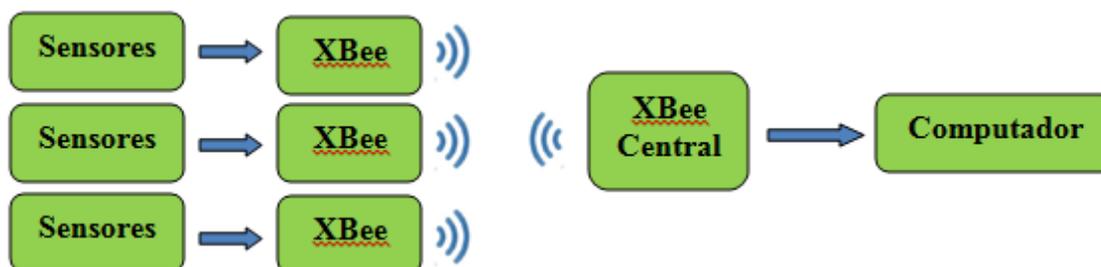


Figura 5.5. Sistema de transmissão de dados sem fio.

Para receber os dados seriais enviados pelo módulo XBee ao computador, foi desenvolvida uma aplicação na linguagem Python com o auxílio da biblioteca “serial”. Esta biblioteca possui funções específicas para comunicação serial, como o envio e o recebimento de dados, o que facilita o desenvolvimento da aplicação.

Foram realizados testes com a rede configurada para funcionar em modo estrela, com dois módulos RFD conectados aos hardwares ArduIMU enviando dados ao módulo conectado ao computador. A comunicação foi configurada para ocorrer no modo API de modo que seja possível identificar o módulo que enviou o dado. O programa no computador é responsável por receber esses dados, separar os dados de cada sensor e identificar o módulo responsável pelo envio dos dados.

5.1.3 Blender

O software Blender permite que o modelo virtual representado por ele seja controlado e animado através de um script que pode ser programado na interface do próprio Blender. Dessa maneira, é possível fazer com que esse script receba dados de um outro programa via socket com informações relativas às novas posições do modelo virtual (Ramos, 2011). Esse ambiente de programação pode ser visto na Figura 5.6.

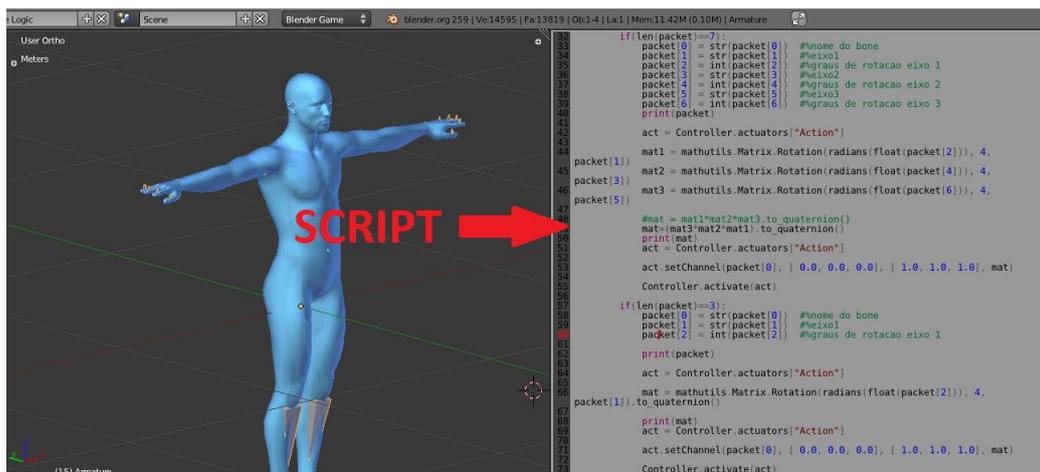


Figura 5.6. Ambiente de programação Blender.

Com essa possibilidade de controle do modelo, foi desenvolvido um script que abre uma conexão via socket e aguarda receber informações sobre as rotações a serem realizadas em um membro específico, também informado através do socket. Ao receber os dados, o Blender irá animar o modelo de acordo com os dados recebidos.

Para representar as articulações humanas, o Blender utiliza uma estrutura composta por diversos *bones*. Um *bone* corresponde a um corpo rígido, conectado a outros corpos rígidos através de ligações que permitem a realização de rotações. Através dessas rotações, é possível mover os membros do corpo humano. O modelo virtual do Blender representado através dos *bones* pode ser visualizado na Figura 5.7.

Os *bones* que serão monitorados pelos sensores são apenas aqueles que representam o braço, antebraço e mão. Esses *bones* foram nomeados de *UpArm*, *LoArm* e *Hand*, respectivamente. Uma imagem detalhada dos *bones* que compõem o membro superior pode vista na Figura 5.8.

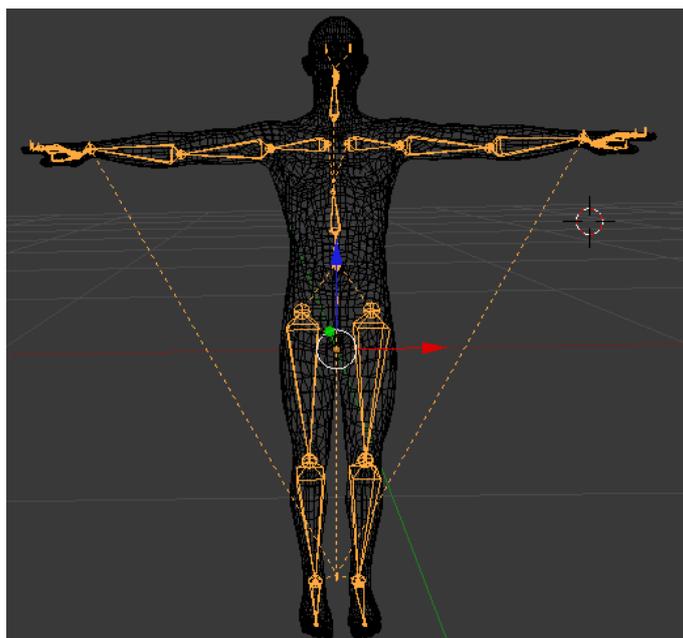


Figura 5.7. Corpo humano representado por bones.

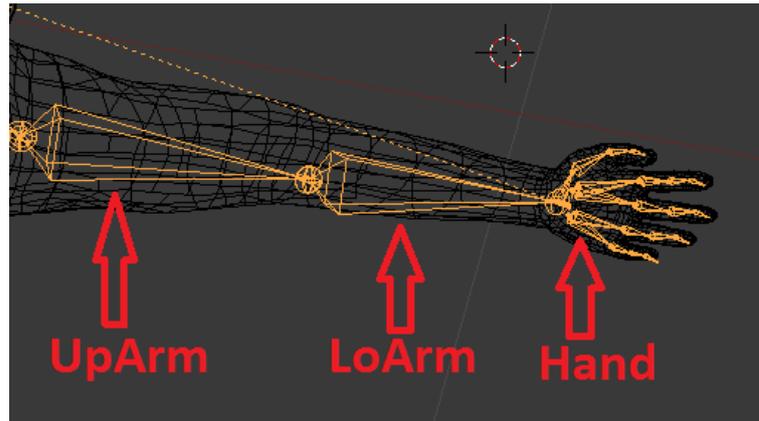


Figura 5.8. Bones utilizados para representar o membro superior.

Para a rotação dos *bones*, o Blender cria um sistema de coordenadas cartesianas para cada *bone*. Sempre que um *bone* é rotacionado, ele é rotacionado utilizando-se os ângulos de Tait Bryan extrínsecos, sempre em relação aos eixos do *bone* anterior, que é considerado como referência tanto para posição como para orientação. Dessa maneira, se o *bone UpArm* for rotacionado, os eixos de rotação do *bone LoArm* serão rotacionados, contudo, os eixos de rotação do *UpArm* estão localizados no corpo do modelo. Como o corpo é considerado estático, os eixos de rotação do *UpArm* serão sempre estáticos.

A Figura 5.9 mostra a orientação dos eixos de rotação para cada um dos três *bones* que compõem o membro superior humano. Para animar o modelo, será então necessário informar ao Blender os ângulos de rotação de Tait Bryan extrínsecos em relação aos seus eixos de rotação para cada um dos *bones* que compõem o membro superior humano.

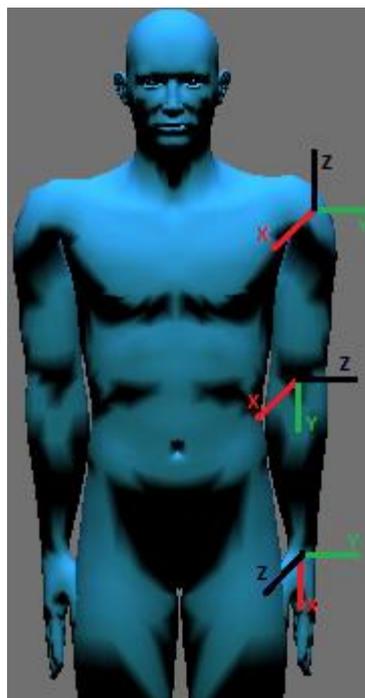


Figura 5.9. Eixos do membro superior humano esquerdo no modelo Blender.

5.2 Conexão Xbee - ArduIMU

Entre as diversas saídas do ArduIMU, ele possui uma saída em +3,3V e outra em GND que podem ser utilizadas para alimentar um pequeno dispositivo, como o XBee. Portanto, a alimentação do XBee vem diretamente do ArduIMU, fazendo com que a bateria seja responsável por alimentar todo o circuito.

Para a comunicação entre ArduIMU e o XBee foi utilizada a comunicação serial pelo padrão UART, presente nos dois dispositivos. A taxa de transmissão de dados padrão do ArduIMU é de 38400 bps. O XBee foi então configurado para funcionar nessa taxa. Para permitir a comunicação entre eles, bastou então conectar o pino Tx do ArduIMU ao pino Rx do XBee e o pino Rx do ArduIMU ao pino Tx do XBee. Desse modo, é feita a comunicação entre os dispositivos.

Para definir quando todos os dados do pacote foram recebidos pelo XBee, o XBee espera até que o ArduIMU deixe de enviar dados por um tempo de envio 3 bytes, nesse caso 625 μ s. O XBee então finaliza o pacote e envia para o módulo coordenador. Como a frequência de amostragem do ArduIMU é em torno de 100 Hz, ou seja, uma amostragem a cada 10ms, o ArduIMU não irá enviar novos dados em um intervalo tão pequeno.

5.3 Conversão dos Dados dos Sensores em Projeção de Eixos

Para conhecer a orientação espacial do hardware ArduIMU e conseqüentemente de cada *bone* do membro superior, é necessário descrever os eixos do hardware em relação aos eixos de referência inicialmente determinados. Isso resulta em uma matriz 3x3 no modelo da equação (5.1) onde ProjXY corresponde a projeção do eixo X da placa sobre o eixo Y de referência.

$$M_{Proj} = \begin{vmatrix} ProjXX & ProjYX & ProjZX \\ ProjXY & ProjYY & ProjZY \\ ProjXZ & ProjYZ & ProjZZ \end{vmatrix} \quad (5.1)$$

Dessa maneira, os eixos da placa estão representados pelas colunas da matriz em relação aos eixos originais.

Os acelerômetros fornecem o valor da aceleração na direção dos seus eixos. Os movimentos realizados no escopo desse projeto são movimentos bastante lentos, o que faz com que a única aceleração significativa presente seja a aceleração da gravidade. Neste tipo de configuração, considerou-se que o sensor mede aceleração estática. Desta maneira, considerando o eixo de um acelerômetro em uma orientação qualquer, existirá um ângulo α entre o eixo do acelerômetro e a aceleração da gravidade, como representado na Figura 5.10. A projeção da aceleração da gravidade sobre o eixo do acelerômetro é facilmente conhecida através de (5.2).

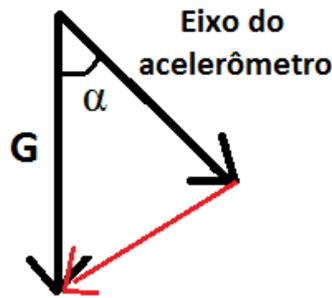


Figura 5.10. Eixo dos acelerômetros em relação a aceleração da gravidade.

$$\text{Projeção da gravidade sobre o eixo} = G \cdot \cos(\alpha) \quad (5.2)$$

Por definição, o eixo da gravidade corresponde ao eixo Z de referência no sentido oposto, logo, o valor dos acelerômetros correspondem diretamente a projeção do eixo do acelerômetro na direção Z com o sinal invertido, multiplicado pelo valor da gravidade. Mas o valor da gravidade, como é o único valor de aceleração presente no sistema, corresponde a soma vetorial de todos os valores de aceleração medidos pelos acelerômetros, o que é dado por (5.3), onde *acelX* corresponde ao valor medido pelo acelerômetro sobre o eixo X da placa.

$$G = \sqrt{\text{acelX}^2 + \text{acelY}^2 + \text{acelZ}^2} \quad (5.3)$$

Desta maneira, a projeção normalizada dos eixos da placa sobre os eixos de referência são dados por (5.4). Vale lembrar que se o movimento sofresse alguma aceleração significativa, essas equações não seriam mais válidas.

$$\begin{aligned} \text{ProjXZ} &= -\text{acelX}/G \\ \text{ProjYZ} &= -\text{acelY}/G \\ \text{ProjZZ} &= -\text{acelZ}/G \end{aligned} \quad (5.4)$$

Os dados dos magnetômetros informam a direção do campo magnético terrestre. Esse campo magnético possui componentes verticais e horizontais, contudo, o eixo vertical já foi determinado pelos acelerômetros, resta apenas definir os eixos horizontais (X e Y). Para determinar os eixos horizontais a partir dos magnetômetros é interessante analisar apenas as componentes horizontais dos dados desses sensores. Para que isso seja possível, é necessário subtrair a componente vertical desse vetor. Para realizar essa função, é realizada a projeção do vetor dos campos magnéticos sobre o eixo vertical (5.5) e o resultado dessa projeção é subtraída do vetor inicial medido pelos magnetômetros.

$$\text{proj}_{\vec{v}} \vec{u} = \left(\frac{\vec{u} \cdot \vec{v}}{\vec{v} \cdot \vec{v}} \right) \vec{v} \quad (5.5)$$

Em posse da componente horizontal, resta conhecer o ângulo existente entre o campo magnético e o eixo Y do corpo humano. Através da Figura 5.9, pode-se perceber que o eixo Y está na direção lateral, saindo do corpo, no caso do braço esquerdo. É preciso no mínimo uma referência horizontal para determinar o ângulo α representado na Figura 5.11, existente entre a medida do magnetômetro e o eixo Y. Essa referência pode ser obtida através de uma posição conhecida dos sensores do magnetômetro.

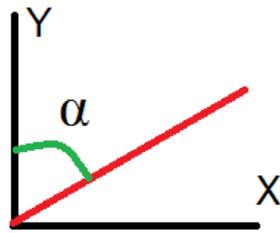


Figura 5.11. Ângulo entre medida do magnetômetro e eixo Y do sistema.

Posicionando o braço estendido na posição vertical, como demonstrado na Figura 5.12 e posicionando o ArduIMU com o seu eixo X (ver Figura 5.3) apontando para o sentido longitudinal do membro superior, no caso, eixo Y para o braço, será possível obter uma posição conhecida do magnetômetro em relação aos eixos do corpo.

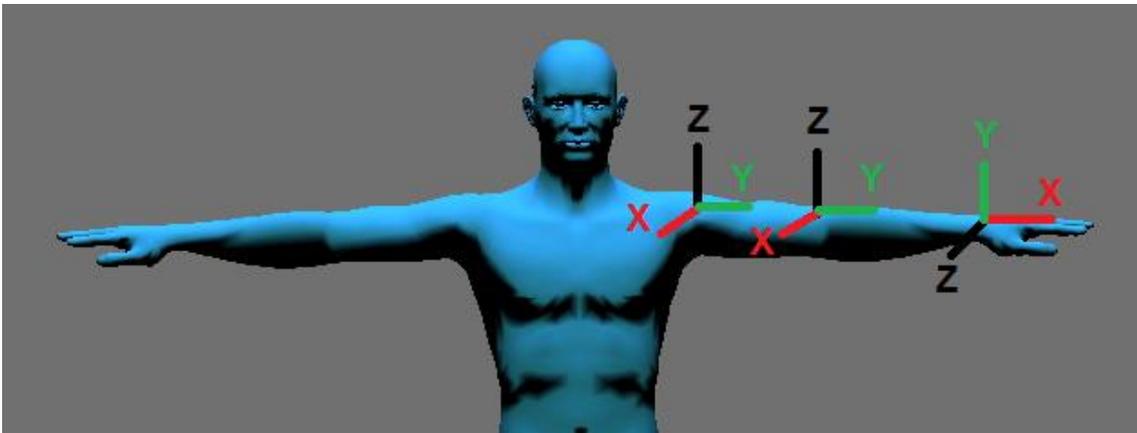


Figura 5.12. Braço posicionado na configuração vertical.

Assim, nessa posição, os eixos do magnetômetro são coincidentes com os eixos do sistema. Analisando o caso do braço, é preciso conhecer então o ângulo da medida horizontal do magnetômetro com o eixo Y do magnetômetro para se conhecer o ângulo entre a medida do magnetômetro e o eixo Y de referência. Esse ângulo é facilmente descoberto através do arcotangente do valor medido pelo eixo X do magnetômetro dividido pelo valor medido pelo eixo Y, o que resulta em (5.5). Portanto, para se obter a direção do eixo Y, basta rotacionar a medida do magnetômetro de α graus sobre o eixo Z. Por último, o eixo X pode ser calculado através do produto vetorial entre Y e Z.

$$\alpha = \text{atan}(\text{magX}/\text{magY}) \quad (5.6)$$

Dessa maneira, os eixos de referência podem ser adquiridos a partir da posição inicial seguindo os seguintes algoritmos:

- o eixo Z é resultado das medidas dos acelerômetros com o sinal invertido e normalizado;
- o eixo Y é a medida dos magnetômetros subtraída da sua projeção sobre o eixo Z, rotacionado de α sobre o eixo Z e normalizado;
- o eixo X é calculado a partir do produto vetorial entre Y e Z.

Em posse dos eixos de referência, basta realizar esses procedimentos repetidamente para redefini-los a cada novo dado adquirido e projetar os eixos do

ArduIMU sobre esses eixos de referência calculados. Como os eixos de referência já estão normalizados, basta realizar o produto escalar entre os eixos calculados pelos sensores e os eixos de referência para se obter os valores da segunda e terceira linha da matriz de projeções.

Realizando o produto matricial entre a matriz transposta dos eixos de referência e a matriz dos eixos calculados, o resultado é exatamente a matriz de projeções, pois esse cálculo realiza o produto escalar entre os vetores de referência e calculados.

5.4 Aplicação do Modelo de Denavit e Hartenberg

Como visto no Capítulo 2.1, o membro superior humano foi representado por 3 articulações principais: ombro, cotovelo e pulso. Estas articulações podem ser representadas por uma rótula, um elipsóide e outro elipsóide, respectivamente. Contudo, para realizar o modelo de acordo com a representação de Denavit e Hartenberg, é preciso representar todas as ligações de rotação como ligações pivô. No caso, uma rótula será representada por 3 ligações pivôs ortogonais e uma ligação elipsóide pode ser representada por 2 ligações pivô. Uma ligação pivô é representada pela imagem da Figura 5.13.

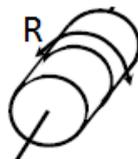


Figura 5.13. Representação de ligação pivô.

A Figura 5.14 representa as articulações da Figura 2.3 em forma de ligações pivô desenvolvidas para este trabalho. Todas as matrizes de passagem desse sistema podem ser conferidas no Apêndice A.

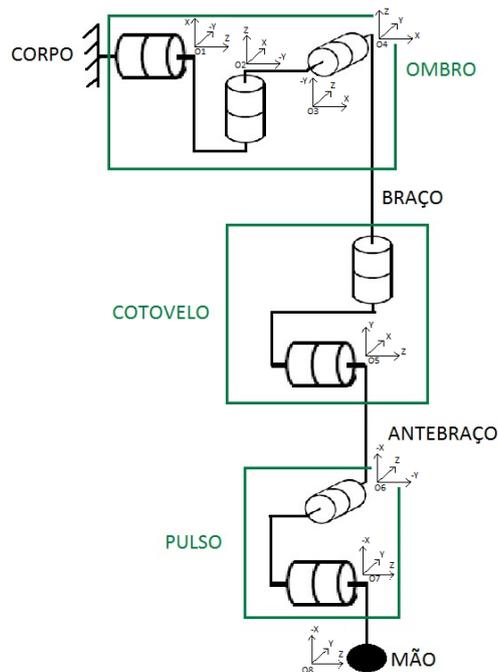


Figura 5.14. Modelo de articulações do braço humano em posição de repouso.

Os eixos da ligação O_1 (primeira ligação pivô) são invariáveis enquanto todos os outros são rotacionados de acordo com a rotação das ligações anteriores. Esses eixos invariáveis correspondem aos eixos do corpo humano, pois foi considerado que o corpo manterá uma posição estática. Os eixos da ligação O_4 representam os eixos do braço após as rotações do ombro. Os eixos da ligação O_6 , representam os eixos do antebraço e os eixos da ligação O_8 representam os eixos da mão.

O braço será sempre rotacionado em relação aos eixos do corpo, enquanto o antebraço será rotacionado sobre os eixos do braço e a mão será rotacionada sobre os eixos do antebraço. Como pode ser visto na Figura 5.14 e na Figura 5.9, os eixos do modelo de Denavit e Hartenberg não correspondem diretamente aos eixos do Blender. Isso ocorre devido a necessidade de seguir as regras de posicionamento de eixos do modelo proposto por Denavit e Hartenberg. Para resolver esse problema, foi criada a Tabela 5.1 que realiza a conversão de eixos do modelo para o Blender.

Tabela 5.1. Conversão de eixos do modelo de Denavit e Hartenberg para Blender.

	Eixos no Modelo de Denavit e Hartenberg								
	Ombro			Cotovelo			Pulso		
	X	Y	Z	X	Y	Z	X	Y	Z
Eixos no Blender	Z	X	Y	Z	-X	-Y	X	-Y	-Z

O próximo passo é o cálculo dos parâmetros de Denavit e Hartenberg. Calculando os parâmetros, representados pela Figura 2.4, para o modelo obtido para o membro superior humano, foi possível extrair dos dados da Tabela 5.2. Os dados de $\theta_{inicial}$ representam os valores de θ_i para que o modelo encontre-se na posição inicial estabelecida, no caso, na posição de repouso.

Tabela 5.2. Parâmetros de Denavit e Hartenberg extraídos.

	1	2	3	4	5	6	7
α	-90	-90	90	90	-90	90	0
θ	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6	θ_7
$\theta_{inicial}$	-90	-90	0	90	-90	0	0

Para definir os ângulos de rotação em função das inclinações dos sensores é necessário utilizar o modelo inverso de Denavit e Hartenberg. Contudo, no caso particular deste problema, é possível segmentá-lo em 3 partes. Ao invés de definir os ângulos de rotação de todos em membros em função dos eixos absolutos, é possível definir os ângulos de rotação em relação ao membro anterior. Nesse caso, as rotações do braço foram definidas em função do eixo absoluto, as rotações do antebraço foram definidas em relação aos eixos do braço após a sua rotação e finalmente, as rotações da mão foram definidas em relação aos eixos do antebraço.

Essa segmentação é interessante pois para definir as rotações da mão, por exemplo, será necessário analisar unicamente 2 variáveis: os graus de liberdade do pulso. Sendo assim, a sua rotação pode ser definida sem a necessidade de avaliar as rotações ocorridas nos membros anteriores.

Isso é possível graças a duas razões:

- existe um sensor em cada segmento do membro superior que fornece a sua inclinação em relação ao eixo absoluto;
- o blender recebe como entrada a rotação sobre os eixos do membro anterior e não em relação ao eixo absoluto.

Foram analisados de forma independente a definição dos ângulos de rotação do braço em relação ao corpo, do antebraço em relação ao braço e da mão em relação ao antebraço.

5.4.1 Solução do sistema corpo - braço

A medida do sensor localizado no braço do usuário fornece a inclinação do eixo O_4 , representado na Figura 5.14, em relação aos eixos absolutos, ou seja, O_1 . Considerando que a matriz fornecida pelo sensor é a matriz genérica 3x3 representada pela matriz abaixo.

$$M = \begin{vmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{vmatrix}$$

A inclinação do eixo O_4 depende unicamente das três rotações possíveis na articulação do ombro, representadas por θ_1 , θ_2 e θ_3 . Ou seja, a matriz dos eixos do braço descritos em função dos eixos absolutos é igual a matriz T_{14} , representada na equação (5.7).

$$T_{14} = \begin{vmatrix} \sin(t1)*\sin(t3) + \cos(t1)*\cos(t2)*\cos(t3) & -\cos(t1)*\sin(t2) & \cos(t1)*\cos(t2)*\sin(t3) - \cos(t3)*\sin(t1) \\ \cos(t2)*\cos(t3)*\sin(t1) - \cos(t1)*\sin(t3) & -\sin(t1)*\sin(t2) & \cos(t1)*\cos(t3) + \cos(t2)*\sin(t1)*\sin(t3) \\ -\cos(t3)*\sin(t2) & -\cos(t2) & -\sin(t2)*\sin(t3) \end{vmatrix} \quad (5.7)$$

Analisando a matriz T_{14} , percebe-se que o termo da terceira linha e segunda coluna depende unicamente da variável θ_2 . Dessa maneira, pode-se facilmente chegar no resultado de θ_2 igualando esses dois termos das matrizes. Contudo, o valor do arccosseno permite 2 respostas válidas, se α é um ângulo válido, $-\alpha$ também é válido como resposta.

Utilizando a matriz de transformação inversa, pode-se obter:

$$\begin{aligned} T^{*14} &= M \\ T^{*24} &= T_{21} * T^{*14} \\ T_{24} &= T_{23} * T_{34} \\ T^{*24} &= T_{24} \end{aligned}$$

Dessa maneira:

$$T_{24} = \begin{vmatrix} \cos(t2)*\cos(t3) & -\sin(t2) & \cos(t2)*\sin(t3) \\ \cos(t3)*\sin(t2) & \cos(t2) & \sin(t2)*\sin(t3) \end{vmatrix}$$

$$T^{*24} = \begin{array}{c} \left| \begin{array}{ccc} -\sin(t_3) & 0 & \cos(t_3) \end{array} \right| \\ \left| \begin{array}{ccc} a_{11}\cos(t_1) + a_{21}\sin(t_1) & a_{12}\cos(t_1) + a_{22}\sin(t_1) & a_{13}\cos(t_1) + a_{23}\sin(t_1) \\ -a_{31} & -a_{32} & -a_{33} \\ a_{21}\cos(t_1) - a_{11}\sin(t_1) & a_{22}\cos(t_1) - a_{12}\sin(t_1) & a_{23}\cos(t_1) - a_{13}\sin(t_1) \end{array} \right| \end{array}$$

Pode-se então igualar os termos da terceira linha e segunda coluna dessas matrizes e chegar a equação (5.8):

$$a_{22}\cos(t_1) - a_{12}\sin(t_1) = 0 \quad (5.8)$$

Essa é uma equação conhecida da trigonometria e admite como soluções os seguintes valores (Rigaud, 2011):

$$a.\cos\theta + b.\sin\theta = x$$

$$\text{seja } K_1 = \text{ArcSin}(b / \text{raiz}(a^2+b^2)) \text{ e } K_2 = \text{ArcCos}(x / \text{raiz}(a^2+b^2))$$

As duas soluções possíveis para θ são S1 e S2 tais que:

$$S_1 = K_1 + K_2$$

$$S_2 = K_1 - K_2$$

Igualando os termos da matriz T14 com a matriz M, pode-se obter:

$$-\cos(t_1)\sin(t_2) = a_{12}$$

$$-\sin(t_1)\sin(t_2) = a_{22}$$

Através dessas duas equações, pode-se definir quais pares de ângulos t_1 e t_2 correspondem entre si como respostas válidas. Em seguida, usam-se as equações:

$$-\cos(t_3)\sin(t_2) = a_{13}$$

$$-\sin(t_2)\sin(t_3) = a_{33}$$

E dessa maneira, é possível obter o valor de t_3 correspondente a cada valor válido de t_2 . Pode-se obter então 2 respostas válidas para t_1 , t_2 e t_3 , que resultam na mesma matriz de rotação obtida.

Matematicamente ambos os valores são válidos, contudo, se for levado em conta que o movimento medido é bastante lento e que a frequência de aquisição de medidas é bastante elevada em relação ao movimento (na ordem de dezenas de hertz), o ângulo considerado correto para a rotação será aquele que tiver a menor variação em relação a medida anterior. Essa relação pode ser usada pois é esperado que entre cada medida, seja rotacionado um ângulo muito pequeno de cada articulação.

5.4.2 Solução do sistema braço – antebraço

Para encontrar os ângulos de rotação do antebraço sobre o braço, é possível utilizar a mesma lógica que no caso anterior, realizando a rotação do eixo O_6 em relação ao eixo O_4 . Contudo, a matriz de inclinação dos eixos fornecida pelo sensor é medida em relação aos eixos absolutos e não aos eixos de O_4 . Será então necessário então reescrever a inclinação dos eixos de O_6 em função dos eixos de O_4 . Como é obtido, através dos sensores, os eixos de O_4 e de O_6 em relação aos eixos absolutos, basta fazer

uma projeção dos eixos de O_6 sobre os eixos de O_4 para obter essa relação. Isso pode ser feito realizando um produto escalar entre os eixos de O_6 e O_4 . Por exemplo, o produto escalar da descrição do eixo x de O_6 sobre o eixo x de O_4 , resultará no primeiro termo da matriz.

A matriz de inclinação dos eixos O_6 em função dos eixos de O_4 (I_{46}) será então:

$$I_{46} = I_4' \cdot I_6$$

Onde I_4 é a inclinação dos eixos de O_4 em relação aos eixos absolutos e o símbolo ' corresponde a transposta da matriz. Dessa maneira, pode-se igualar a matriz I_{46} a matriz T_{46} e encontrar os 2 ângulos de rotação correspondentes ao antebraço.

$$T_{46} = \begin{vmatrix} \cos(t4)*\cos(t5) & -\sin(t4) & -\cos(t4)*\sin(t5) \\ \cos(t5)*\sin(t4) & \cos(t4) & -\sin(t4)*\sin(t5) \\ \sin(t5) & 0 & \cos(t5) \end{vmatrix}$$

Nesse caso, somente existirá uma resposta correta.

5.4.3 Solução do sistema antebraço - mão

Da mesma maneira que para o sistema braço – antebraço, é necessário fazer a projeção dos eixos de O_8 para os eixos de O_6 , de tal modo que:

$$I_{68} = I_6' \cdot I_8$$

Igualando a matriz I_{68} a matriz T_{68} é possível obter os valores de rotação para a articulação do pulso.

$$T_{68} = \begin{vmatrix} \cos(t6)*\cos(t7) & -\cos(t6)*\sin(t7) & \sin(t6) \\ \cos(t7)*\sin(t6) & -\sin(t6)*\sin(t7) & -\cos(t6) \\ \sin(t7) & \cos(t7) & 0 \end{vmatrix}$$

Novamente, nesse caso haverá apenas uma resposta correta.

5.5 Aplicação dos Ângulos de Tait Bryan

Para escolher qual a melhor forma de representar os ângulos de rotação de cada segmento do membro superior humando, é interessante analisar o procedimento que o Blender utiliza para realizar as rotações sobre os *bones* do modelo virtual. Como o Blender utiliza os ângulos de Tait Bryan extrínsecos para orientar os seus *bones* no espaço, o mais interessante é então transformar os eixos projetados nos três ângulos de rotação de Tait Bryan extrínsecos.

A escolha da ordem dos eixos de rotação é arbitrária, precisando apenas ser repassada posteriormente para o Blender. Portanto, foi escolhida como ordem de rotação, a ordem Z,Y,X, ou seja, a primeira rotação ocorre sobre o eixo Z, a segunda sobre o eixo Y e a terceira sobre o eixo X. A rotação sobre o eixo Z será chamada de γ , sobre o eixo Y de β e sobre o eixo X de α .

A matriz resultante das três rotações corresponde a matriz com os eixos projetados. Seja a matriz resultante a matriz genérica apresentada abaixo, seguida das matrizes de rotação sobre o eixo x, y e z.

$$M = \begin{vmatrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{vmatrix}$$

$$R_x = \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{vmatrix}$$

$$R_y = \begin{vmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{vmatrix}$$

$$R_z = \begin{vmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

A matriz de rotação sobre os três eixos é dada pela multiplicação $R_{zyx} = R_x.R_y.R_z$, o seu resultado pode ser visualizado na equação (5.9).

$$R_{zyx} = \begin{vmatrix} \cos(\beta)*\cos(\gamma) & -\cos(\beta)*\sin(\gamma) & \sin(\beta) \\ \sin(\alpha)*\sin(\beta)*\cos(\gamma) + \cos(\alpha)*\sin(\gamma) & \cos(\alpha)*\cos(\gamma) - \sin(\alpha)*\sin(\beta)*\sin(\gamma) & -\cos(\beta)*\sin(\alpha) \\ \sin(\alpha)*\sin(\gamma) - \cos(\alpha)*\cos(\gamma)*\sin(\beta) & \cos(\gamma)*\sin(\alpha) + \cos(\alpha)*\sin(\beta)*\sin(\gamma) & \cos(\alpha)*\cos(\beta) \end{vmatrix} \quad (5.9)$$

Os termos da matriz são referenciados da seguinte maneira: $M(i,j)$ corresponde ao termo da linha i e coluna j da matriz M .

Percebe-se que o termo $R_{zyx}(1,3)$ depende unicamente de β , portanto, pode-se descobrir o valor de β através da operação inversa de $\sin(\beta)$ para esse termo. Logo $\beta = \text{asin}(R_{zyx}(1,3))$. Contudo, a função arcosseno possui dois valores válidos como resposta, os valores de β são então os descritos em (5.10).

$$\begin{aligned} \beta_1 &= \text{asin}(R_{zyx}(1,3)) \\ \beta_2 &= \pi - \text{asin}(R_{zyx}(1,3)) \end{aligned} \quad (5.10)$$

A divisão do termo $R_{zyx}(2,3)$ pelo $R_{zyx}(3,3)$ resulta na tangente de α com o sinal negativo, portanto:

$$\alpha = \text{atan}(-R_{zyx}(2,3) / R_{zyx}(3,3))$$

A função arcotangente possui dois valores possíveis como resultado, em quadrantes diferentes, mas através do sinal do seno e do cosseno de α , é possível descobrir exatamente em que quadrante está o ângulo α . Em Python, a função $\text{atan2}(\sin, \cos)$ retorna o valor do ângulo em seu quadrante correto levando em consideração o sinal de \sin e de \cos . Deve-se portanto eliminar o sinal de $\cos(\beta)$ nos termos $R_{zyx}(2,3)$ e $R_{zyx}(3,3)$ da matriz. Tem-se portanto que os valores possíveis de α são os valores de (5.11):

$$\begin{aligned} \alpha_1 &= \text{atan2}(-R_{zyx}(2,3)/ \cos(\beta_1)) / (R_{zyx}(3,3)/ \cos(\beta_1)) \\ \alpha_2 &= \text{atan2}(-R_{zyx}(2,3)/ \cos(\beta_2)) / (R_{zyx}(3,3)/ \cos(\beta_2)) \end{aligned} \quad (5.11)$$

De forma análoga, percebe-se que o valor de γ é encontrado através do arcotangente da divisão de $-R_{zyx}(1,2)$ por $R_{zyx}(1,1)$. Portanto, os valores possíveis de γ são os descritos em (5.12).

$$\begin{aligned} \gamma_1 &= \text{atan2}(-R_{zyx}(1,2)/ \cos(\beta_1)) / (R_{zyx}(1,1)/ \cos(\beta_1)) \\ \gamma_2 &= \text{atan2}(-R_{zyx}(1,2)/ \cos(\beta_2)) / (R_{zyx}(1,1)/ \cos(\beta_2)) \end{aligned} \quad (5.12)$$

Contudo, essa lógica não pode ser aplicada quando $\cos(\beta)$ valer 0, pois haverá uma divisão por zero. É preciso lidar com o caso de $\cos(\beta) = 0$ separadamente. Para que essa relação seja válida, é preciso que β seja igual a $\pi/2$ ou $-\pi/2$.

No caso onde $\beta = \pi/2$, o termo $R_{zyx}(2,2)$ resulta em $\cos(\alpha).\cos(\gamma) - \sin(\alpha).\sin(\gamma)$, pois $\sin(\pi/2) = 1$. Logo, $R_{zyx}(2,2) = \cos(\alpha + \gamma)$. Da mesma maneira, o termo $R_{zyx}(3,2)$ resulta em $\cos(\gamma).\sin(\alpha) + \cos(\alpha).\sin(\gamma)$. Logo, $R_{zyx}(3,2) = \sin(\alpha + \gamma)$. Portanto $\alpha + \gamma = \text{atan2}(R_{zyx}(3,2), R_{zyx}(2,2))$. O que resulta em (5.13).

$$\alpha = -\gamma + \text{atan2}(R_{zyx}(3,2), R_{zyx}(2,2)) \quad (5.13)$$

No caso onde $\beta = -\pi/2$, o termo $R_{zyx}(2,2)$ resulta em $\cos(\alpha).\cos(\gamma) + \sin(\alpha).\sin(\gamma)$, logo, $R_{zyx}(2,2) = \cos(\alpha - \gamma)$. Da mesma maneira, o termo $R_{zyx}(3,2)$ resulta em $\cos(\gamma).\sin(\alpha) - \cos(\alpha).\sin(\gamma)$. Logo, $R_{zyx}(3,2) = \sin(\alpha - \gamma)$. Portanto $\alpha - \gamma = \text{atan2}(R_{zyx}(3,2), R_{zyx}(2,2))$. O que resulta em (5.14).

$$\alpha = \gamma + \text{atan2}(R_{zyx}(3,2), R_{zyx}(2,2)) \quad (5.14)$$

E nesse caso, o valor de γ pode ser qualquer valor, é normal escolher como valor o zero.

No caso do antebraço e do pulso, existem apenas dois graus de liberdade de rotação, exatamente sobre os eixos Y e Z do modelo do Blender em ambos os casos. A ordem de rotação escolhida foi primeiramente sobre o eixo Z e posteriormente sobre o eixo Y. A matriz resultante ($R_{zy} = R_y \cdot R_z$) resultou na matriz da equação (5.15).

$$R_{zy} = \begin{vmatrix} \cos(\beta)*\cos(\gamma) & -\cos(\beta)*\sin(\gamma) & \sin(\beta) \\ \sin(\gamma), & \cos(\gamma) & 0 \\ -\cos(\gamma)*\sin(\beta), & \sin(\beta)*\sin(\gamma), & \cos(\beta) \end{vmatrix} \quad (5.15)$$

Analizando a matriz da equação (5.15), releva-se (5.16) e (5.17).

$$\beta = \text{atan2}(R_{zy}(1,3), R_{zy}(3,3)) \quad (5.16)$$

$$\gamma = \text{atan2}(\text{Rzy}(2,1), \text{Rzy}(2,2)) \quad (5.17)$$

5.6 Calibração dos Sensores

Os acelerômetros foram programados para terem uma sensibilidade de até $\pm 4g$, os giroscópios de até $\pm 2000^\circ/\text{seg}$. Esses dados são transmitidos após passarem por um conversor digital de 16 bits. No caso do magnetômetro, o conversor digital é de 12 bits. Dessa maneira, a resolução do conversor AD do acelerômetro é de $8/2^{16} = 0,0012g$ por bit, o que é, no caso desse projeto, uma resolução adequada. E no caso do giroscópio de $4000/2^{16} = 0,061^\circ$ por bit. Tendo em vista a imprecisão no posicionamento do sensor no braço e na imprecisão do movimento realizado por um membro superior humano, essa resolução é satisfatória.

Além disso, como o movimento realizado é bastante lento, e ocorrem variações nas medidas dos sensores, é vantajoso passar os dados dos sensores por um filtro passa baixa. Dessa maneira, o movimento realizado pelo modelo virtual será mais suave e mais parecido com o movimento real. Depois de alguns testes, pode-se perceber empiricamente que um filtro passa baixa de 10Hz (integrado no ArduIMU) é a melhor opção.

Para que a medida dos sensores sejam mais confiáveis, é necessário que cada sensor seja calibrado individualmente. Para isso, é necessário adquirir, para cada sensor, o valor de *offset* e de sensibilidade. O valor de *offset* pode ser calculado pela equação (5.18). Esse valor corresponde a um deslocamento no valor medido pelo sensor, e deve ser subtraído da medida para que ela represente um valor válido. Os símbolos Max e Min correspondem ao valor máximo de leitura do sensor e o valor mínimo, respectivamente.

$$\text{Offset} = (\text{Max} + \text{Min}) / 2 \quad (5.18)$$

Para se obter o valor de sensibilidade de cada sensor, utiliza-se a equação (5.19).

$$\text{Sensibilidade} = (\text{Max} - \text{Min}) / 2 \quad (5.19)$$

Cada leitura do sensor será então calculada de acordo com a equação (5.20).

$$\text{Valor_experimental} = (\text{leitura} - \text{offset}) / \text{sensibilidade} \quad (5.20)$$

Vale lembrar que isso deve ser feito para cada eixo de cada sensor (acelerômetro, giroscópio e magnetômetro), o que representa 9 valores de *offset* e de sensibilidade por ArduIMU.

Para se descobrir os valores Max e Min de cada eixo do acelerômetro, deve-se apontar os eixos X, Y e Z na direção da gravidade, ou seja, para baixo e, posteriormente, apontá-los na direção oposta. O mesmo deve ser feito com os magnetômetros, contudo, ao invés de apontá-los em direção a gravidade, deve-se apontá-los na direção do campo magnético terrestre. Como é muito difícil definir a direção do campo magnético sem auxílio de ferramentas, o valor medido por cada sensor é mostrado na tela do computador. Assim é possível descobrir a direção do campo magnético quando um sensor atinge o seu valor máximo. Percebe-se que ele atingiu o seu valor máximo pois para qual que seja a direção que seja movimentado, o seu valor será reduzido.

No caso dos giroscópios, a calibração é um pouco diferente. Pode-se descobrir o *offset* de cada sensor quando, ao não se impor nenhuma rotação aos sistema, ele retorna

algum valor diferente de zero. Esse valor retornado deverá ser subtraído do valor medido pelo sensor. Quanto a determinação de sensibilidade, é necessário um equipamento específico capaz de rotacioná-lo a uma velocidade angular conhecida e então, analisar qual a proporção dessa velocidade está sendo retornada pelo sensor. Novamente, a equação para calcular o valor correto do sensor é a apresentada na equação (5.20).

Como o laboratório não dispõe dessa ferramenta de calibração de giroscópios, esses sensores não puderam ser corretamente calibrados. O que incentivou a utilização predominante dos outros sensores. Outras medidas importantes a serem adquiridas dos sensores são a sua repetibilidade e a sua linearidade de medida. Para realizar tais medidas, foi utilizado um instrumento chamado goniômetro (Figura 5.15), capaz de medir um determinado ângulo de inclinação. Dessa maneira, o ArduIMU foi fixado sobre esse instrumento, a parte fixa do instrumento foi fixada na direção da aceleração da gravidade e os sensores foram rotacionado sobre ângulos conhecidos de modo a se obter dados da saída do sensor gerando uma relação entre a sua inclinação e a aceleração da gravidade. Como o movimento a ser realizado nos exercícios de reabilitação são bastante lentos, não serão levantados dados sobre a resposta do sensor a acelerações dinâmicas, apenas estáticas.

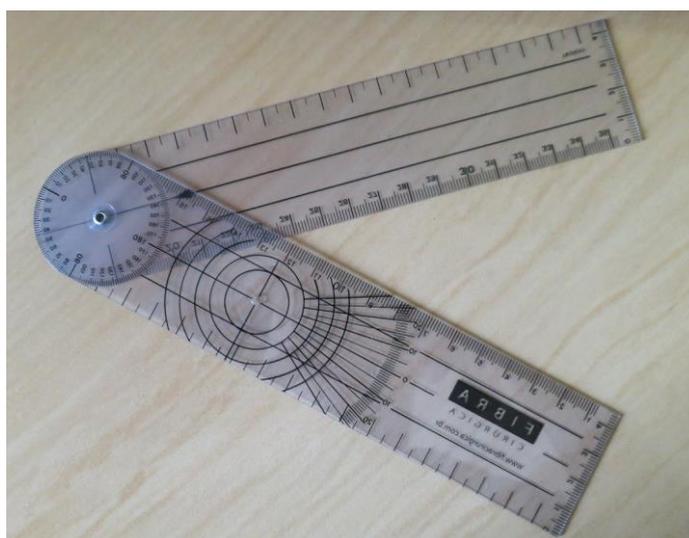


Figura 5.15. Exemplo de goniômetro.

Novamente, como é necessário saber a direção exata do sinal resultante para realizar esses procedimento, não é possível realizá-lo em relação ao magnetômetro devido a direção desconhecida do campo magnético. Nesse caso, deve-se ater aos dados fornecidos pelo fabricante.

5.7 Programa Python no Computador

No computador ao qual o módulo Xbee FFD estará conectado, um programa em python deverá estar em funcionamento. Esse programa tem quatro funções principais, são elas:

- transformar os dados dos conversores ADs dos sensores em inclinações relativas aos eixos absolutos;

- transformar as inclinações dos eixos em ângulos de rotação de Tait Bryan;
- enviar os ângulos de rotação e o membro correspondente ao Blender através de uma comunicação via socket;
- armazenar os dados de rotação em um banco de dados.

Para que o programa possa realizar essas funções principais, existem algumas funções secundárias que devem também ser implementadas. Podem ser destacadas:

- comunicação serial com módulo Xbee;
- desencapsulamento do pacote Xbee;
- análise de integridade de dados do pacote;
- calibração dos sensores;
- identificação do Xbee emissor;

Todas as funções podem ser vistas no fluxograma da Figura 5.16.

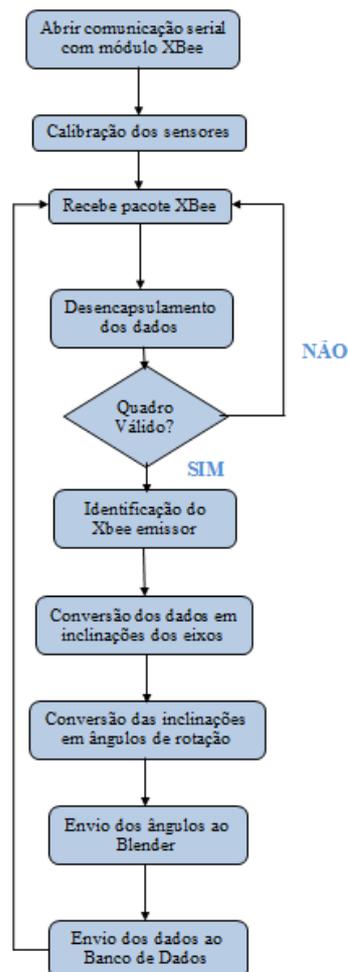


Figura 5.16. Fluxograma do programa Python.

5.8 Verificação do movimento

Para verificar se o movimento realizado pelo usuário está de acordo com o movimento padrão do exercício, é necessário possuir dados sobre o movimento padrão. Portanto, o programa foi adaptado para ser capaz de recolher dados de um movimento qualquer e armazená-lo em um banco de dados. Os dados relativos ao posicionamento do membro superior (rotação no braço, antebraço e mão) são armazenados no banco de dados a uma taxa de 10Hz.

Para verificar se o movimento realizado pelo usuário está de acordo com o movimento padrão, os mesmos dados serão recolhidos para a movimentação do usuário e então serão comparados com o dados relativos ao movimento padrão.

Para que o movimento do usuário esteja conforme, dois critérios devem ser satisfeitos:

- a diferença entre os ângulos de rotação do usuário e do movimento padrão não pode ser superior a um valor limite em nenhum momento do movimento;
- o erro médio do movimento não deve ultrapassar um determinado valor.

Uma diferença de ângulos de 5° no movimento é pouco significativo e pode ser até mesmo ocasionado devido a medidas imprecisas dos sensores. diferenças maiores, na ordem da dezena de graus, devem ter um peso bem maior no cálculo do erro médio do movimento. Por isso, para calcular o erro médio do movimento, é utilizado o método da soma do quadrado das diferenças. Através desse método, diferenças mais significativas entre os movimentos têm um peso muito maior no cálculo do erro médio.

Realizando a soma do quadrado das diferenças para os ângulos do braço, antebraço e mão, separadamente, é possível identificar qual *bone* teve o pior desempenho. Essa informação é útil para que o usuário saiba onde ele deverá corrigir o seu movimento. Contudo, é necessário ter cautela em dois pontos principais ao realizar a soma do quadrado das diferenças:

- a diferença entre ângulos positivos e negativos não são medidas reais da distância entre os ângulos (i.e. 175° e de -180° estão distantes de apenas 5°).
- a menor distância entre dois ângulos pode ser através de uma rotação no sentido anti-horário (i.e. 355° e 5° estão distantes de apenas 10°).

Para que o cálculo da distância entre ângulos seja confiável, todos os ângulos serão interpretados como positivos. Para isso, se um ângulo for negativo de módulo x , ele será substituído por $360^\circ - x$.

O módulo da diferença entre dois ângulos não pode ser maior do que 180°. Se ela for maior que 180°, a diferença real entre eles é dada por $360^\circ - \text{módulo da diferença}$.

Para definir qual o valor máximo da soma do quadrado das diferenças em que o movimento ainda é aceitável, foi desenvolvida uma equação, considerando um erro de ângulação aceitável durante todo o movimento. Considerando que um erro de g° em cada eixo de cada articulação, durante todo o movimento, seja aceitável. São capturadas medidas a cada intervalo de tempo de S segundos e o movimento tem uma duração total de T segundos. Como existem 7 eixos de rotação ao todo, seguindo o modelo

apresentado na Figura 2.3, o valor máximo admitido para a soma do quadrados da diferença é dado por (5.21).

$$\text{Valor_m\u00e1ximo} = 7 * T / S * g^2 \quad (5.21)$$

Para que o usu\u00e1rio tenha um *feedback* mais amig\u00e1vel, deve ser mostrado ao mesmo tempo na tela o movimento que ele est\u00e1 realizando e o movimento padr\u00e3o que deve ser realizado. Como o usu\u00e1rio se baseia no movimento padr\u00e3o mostrado na tela para realizar o exerc\u00edcio, \u00e9 esperado que o movimento do usu\u00e1rio seja realizado com um certo atraso em rela\u00e7\u00e3o ao movimento padr\u00e3o. Por isso, no momento de analisar se o movimento realizado pelo usu\u00e1rio foi v\u00e1lido ou n\u00e3o, deve-se tamb\u00e9m analisar o movimento do usu\u00e1rio com defasagem temporal em rela\u00e7\u00e3o ao movimento padr\u00e3o. Isso \u00e9 necess\u00e1rio pois, mesmo que o movimento seja realizado com defasagem temporal, ele pode ter sido executado corretamente.

6 RESULTADOS E DISCUSSÕES

Os procedimentos experimentais citados no capítulo 5 serão avaliados para validação, ou não, dos conceitos apresentados.

6.1 Prototipagem

Foram contruídos protótipos do ArduIMU conectado ao transmissor sem fio Xbee montados sobre uma placa perfurada. Com essa montagem, os sensores ficam posicionados sobre uma superfície plana e o eixo Y fica paralelo ao sentido longitudinal da placa perfurada, facilitando o posicionamento do circuito sobre o membro do usuário. A alimentação do ArduIMU se dá por uma bateria de 9V, e uma saída controlada a 3.3V do ArduIMU alimenta o transmissor sem fio. Uma foto desse protótipo pode ser vista na Figura 6.1. Conectado ao computador, está o módulo FFD da comunicação sem fio, responsável por receber todos os dados da comunicação enviada pelos módulos periféricos. Esse módulo pode ser visualizado na Figura 6.2.

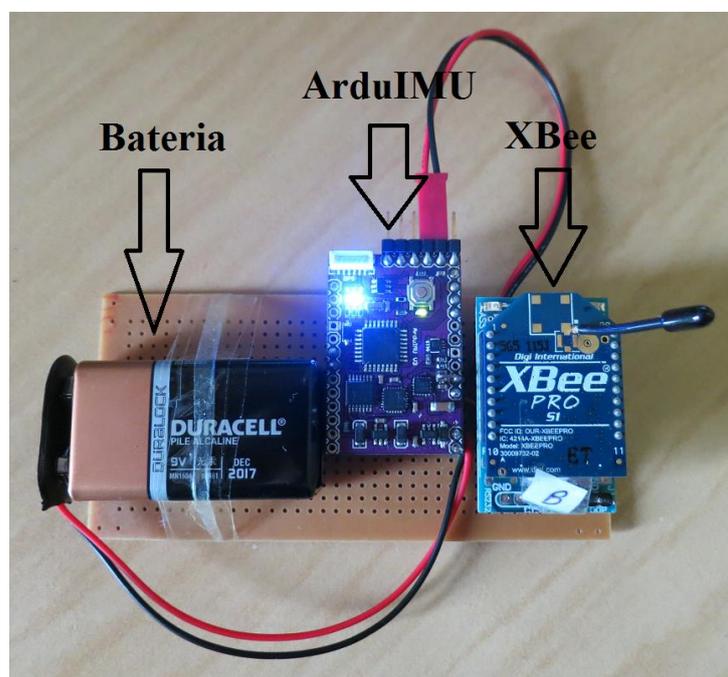


Figura 6.1. Protótipo do sensor e transmissor sem fio.



Figura 6.2. Módulo receptor a ser conectado no computador.

A rede Xbee foi configurada para que funcione no modo estrela, ou seja, todos os Xbees enviam dados para o dispositivo FFD, conectado ao computador. Para que a comunicação possa ocorrer, todos os Xbees devem estar configurados para operar na mesma rede e sobre o mesmo canal. Na Tabela 6.1 pode ser vista a configuração de cada Xbee.

Tabela 6.1. Configuração da rede Xbee.

	ID	Enviar para	FFD	Rede	Canal
Receptor	0x1234	0xFFFF	SIM	0X3332	C
Braço	0xEEEE	0X1234	NÃO	0X3332	C
Antebraço	0xCCCC	0X1234	NÃO	0X3332	C
Mão	0xDDDD	0X1234	NÃO	0X3332	C

Afim de verificar se a comunicação sem fio estava ocorrendo de forma correta, uma conexão teste foi estabelecida entre dois protótipos e o receptor Xbee (Figura 6.3). Os ArduIMUs foram programados para enviar uma mensagem padrão, para que possa ser verificado se ela chegou corretamente ao computador. Os Xbees utilizados foram os referentes ao braço (endereço 0xeee) e ao antebraço (endereço 0xcccc). A mensagem enviada pelo ArduIMU conectado ao Xbee com o endereço do braço foi “@Eu sou o braço!” e pelo antebraço “@Eu sou o antebraço!”. Foi impressa na tela a mensagem recebida pelo receptor seguida do endereço do Xbee emissor. Os resultados desse teste podem ser visualizados na Figura 6.4.

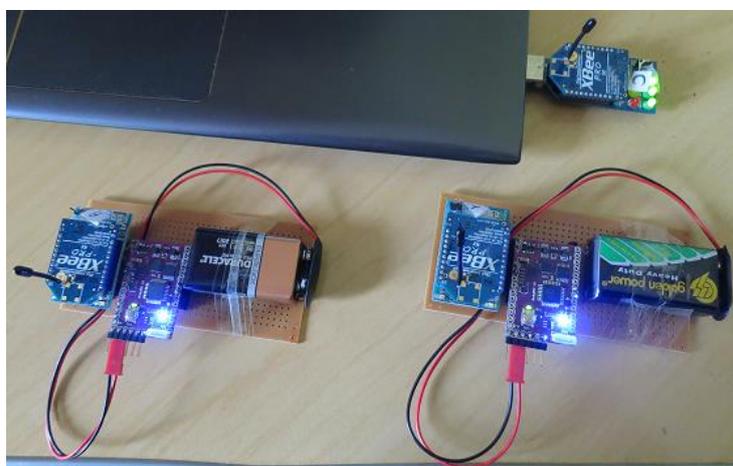


Figura 6.3. Protótipos em funcionamento para teste.

```

C:\Windows\system32\cmd.exe
Eu sou o braco! Recebido de: 0xeeee
Eu sou o antebraço! Recebido de: 0xcccc
Eu sou o braco! Recebido de: 0xeeee
Eu sou o antebraço! Recebido de: 0xcccc
Eu sou o braco! Recebido de: 0xeeee
Eu sou o antebraço! Recebido de: 0xcccc
Eu sou o braco! Recebido de: 0xeeee
Eu sou o braco! Recebido de: 0xeeee
Eu sou o braco! Recebido de: 0xeeee
Eu sou o antebraço! Recebido de: 0xcccc
Eu sou o braco! Recebido de: 0xeeee
Eu sou o antebraço! Recebido de: 0xcccc

```

Figura 6.4. Dados recebidos dos protótipos em função de teste.

A fim de testar a eficiência da rede, foram realizados testes para estimar a quantidade de quadros perdidos na transmissão. Variando a frequência de transmissão dos dados do ArduIMU, foram contados os números de quadros recebidos pelo computador e o número estimado de envios em um intervalo de tempo conhecido. A partir desses dados, pode-se estimar a porcentagem de quadros perdidos. Os resultados encontram-se na Tabela 6.2.

Tabela 6.2. Teste de eficiência da rede Xbee.

Frequência de transmissão ArduIMU	Tempo de transmissão	Quadros esperados por ArduIMU	Quadros recebidos do Xbee Braço	Quadros recebidos do Xbee Antebraço	Porcentagem de quadros recebidos
10Hz	30s	300	300 (100%)	300 (100%)	100%
15Hz	30s	450	411 (91,33%)	385 (85,6%)	88,4%
20Hz	30s	600	580 (96,7%)	593 (98,8%)	97,8%
25Hz	30s	750	735 (98%)	745 (99,3%)	98,7%
33,33Hz	30s	1000	854 (85,4%)	922 (92,2%)	88,8%
40Hz	30s	1200	1124 (93,7%)	1124 (93,7%)	93,7%
50Hz	30s	1500	1491 (99,4%)	1480 (98,7%)	99%

A partir desses dados percebe-se que a eficiência da rede não é um problema muito importante, pois ela se mantém acima de 90% na maior parte das experiências. Seria interessante realizar os mesmos testes com mais módulos Xbee+ArduIMU conectados a rede, contudo, o laboratório IEE não dispõe de outros ArduIMUs no momento. A frequência escolhida para funcionamento foi de 20Hz, pela alta eficiência e também por motivos que são discutidos no capítulo 6.6.

6.2 Calibração

Utilizando um programa desenvolvido para capturar os valores máximos e mínimos dos acelerômetros e dos magnetômetros em um intervalo de tempo de 50 segundos (tempo necessário, definido empiricamente, para rotacionar os sensores em

todas as direções relevantes), foi realizado uma calibração experimental que resultou nos seguintes dados:

Valores dos acelerômetros (eixos X, Y e Z, respectivamente):

- máximos: 8137, 8576, 9812;
- mínimos: -9215, -8496, -8399.

Valores dos magnetômetros (eixos X, Y e Z, respectivamente):

- máximos: 88, 298, 5;
- mínimos: -511, -402, -550.

A partir desses valores, podem ser definidos os valores de *offset* e de sensibilidade para cada sensor, o que resultou nesse caso em:

- *offset* dos acelerômetros (X,Y e Z): -539, 40, 706;
- sensibilidade dos acelerômetros: 8676, 8536, 9105;
- *offset* dos magnetômetros (X,Y e Z): -212, -52, -273;
- sensibilidade dos magnetômetros: 299, 350, 277;

Em seguida, para verificar a repetibilidade e a linearidade dos acelerômetros, foi montado um sistema de testes com o ArduIMU fixo ao goniômetro. Através da rotação do goniômetro foi possível relevar dados da inclinação ao mesmo tempo em que o ArduIMU enviava dados relativos aos seus sensores para o computador. As imprecisões de medida devido ao erro mecânico foram reduzidas ao máximo. O ArduIMU foi posicionado paralelamente ao braço de medidas do goniômetro e a parte fixa do goniômetro foi colocada na posição vertical.

Os valores dos acelerômetros devem ser linearmente proporcionais à aceleração sobre aquele eixo. Como pode ser visto na Figura 5.10, a aceleração sobre o eixo Y da Figura 6.5 deve ser então linearmente proporcional ao cosseno do ângulo α , enquanto o valor resultante sobre o eixo X deve ser linearmente proporcional ao seno desse mesmo ângulo. No caso da calibração do eixo Z (Figura 6.6), o valor do sensor deve ser diretamente proporcional ao cosseno do ângulo de rotação.

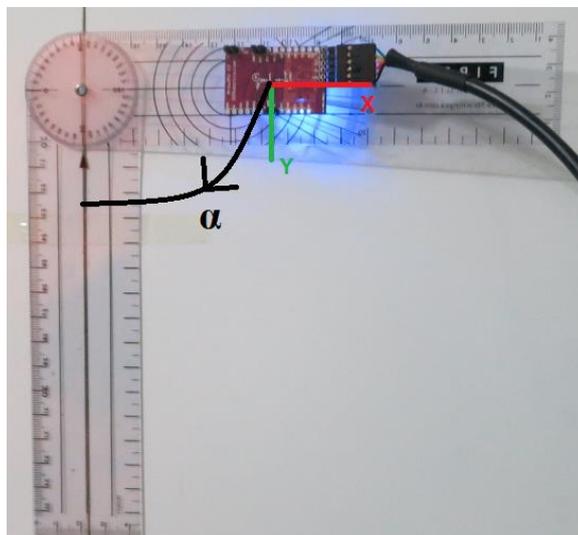


Figura 6.5. Método de calibração do acelerômetro no eixo X e Y.

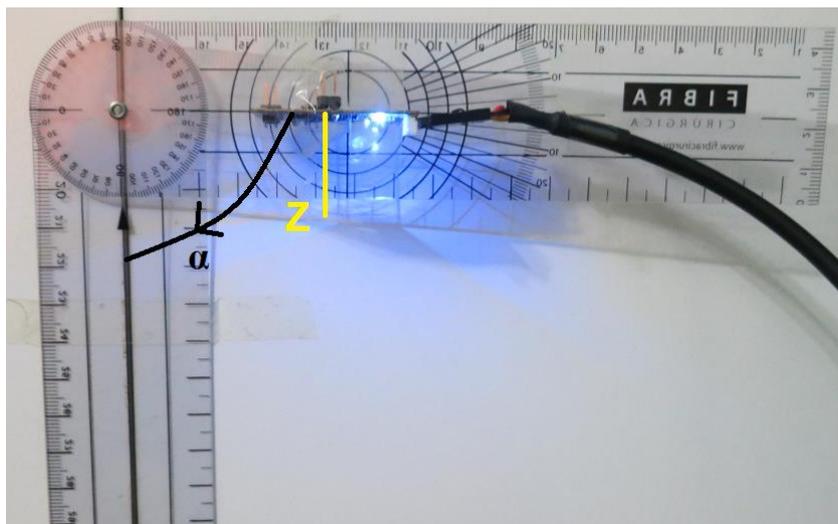


Figura 6.6. Método de calibração do acelerômetro no eixo Z.

No caso da calibração do eixo Z, o posicionamento correto do sensor (perpendicular ao braço do goniômetro) é um pouco mais difícil de ser realizado sem equipamento adequado. Portanto, alguns erros de medida um pouco mais significativos podem aparecer nesse caso. Pode-se ver a dificuldade na precisão do posicionamento do ArduIMU na Figura 6.7.

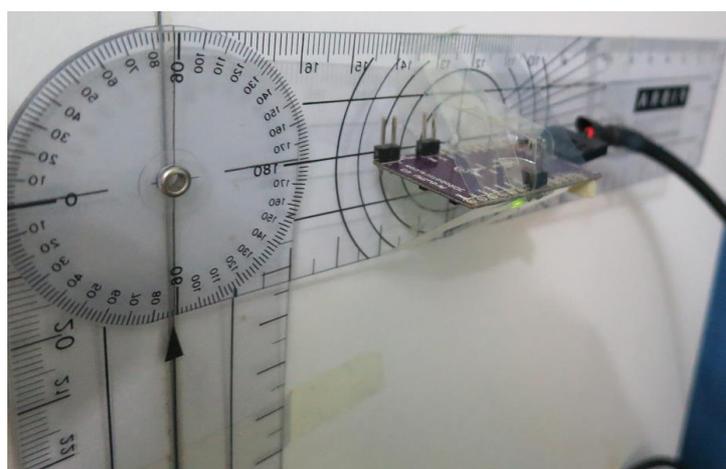


Figura 6.7. Fixação para calibração no eixo Z.

Analisando os resultados das medidas para o eixo Y, foram obtidos os gráficos da Figura 6.8 e Figura 6.9. Os dados detalhados destas medidas e os gráficos dos outros eixos podem ser vistos no Anexo A.

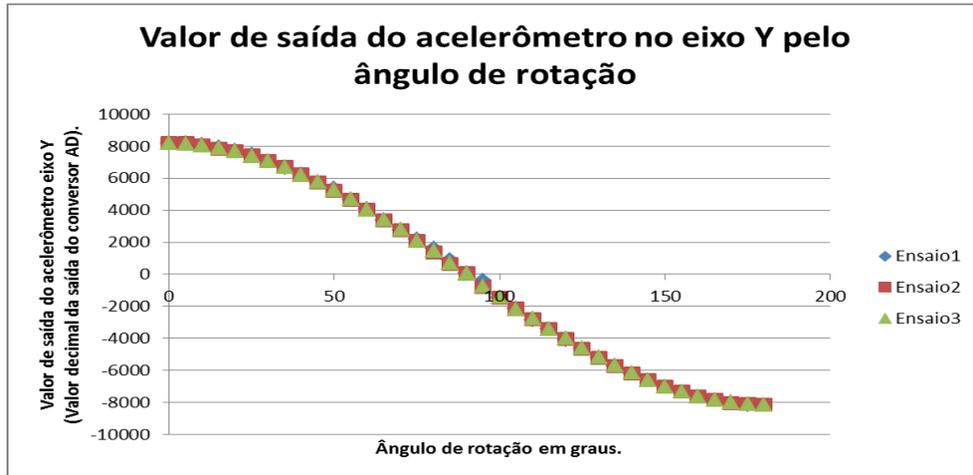


Figura 6.8. Valor do acelerômetro no eixo Y em relação ao ângulo α .

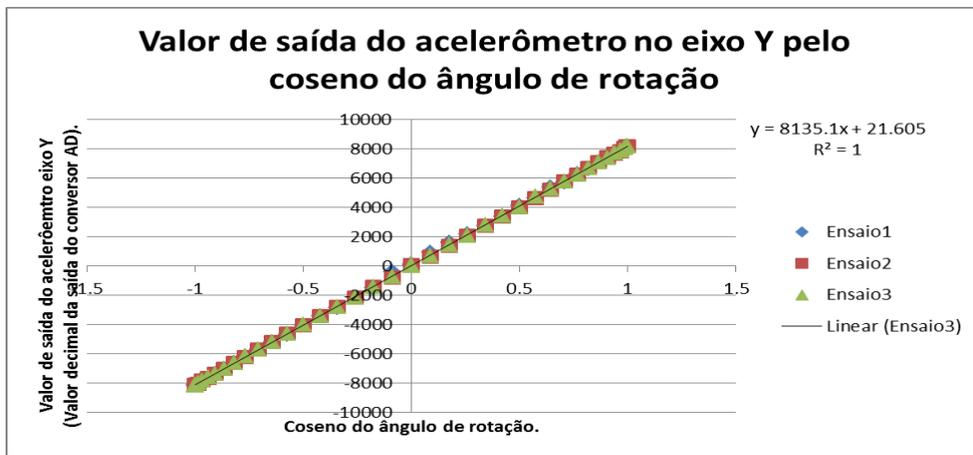


Figura 6.9. Valor do acelerômetro no eixo Y em relação ao cosseno do ângulo α .

Em função dos dados experimentais obtidos da Figura 6.9 e nos gráficos do Anexo A, a função de transferência e o coeficiente de determinação (R^2) para cada eixo podem ser vistos na Tabela 6.3.

Tabela 6.3. Resultado da calibração dos acelerômetros.

Eixo	Função de transferência	R^2
X	$y = 8081.6x - 123.7$	0.9994
Y	$y = 8135.1x + 21.605$	1
Z	$y = -8458.9x + 791.03$	0.9999

Como os valores de R^2 estão muito próximos de 1, os dados dos conversores AD podem ser considerados lineares, levando em conta a aplicação desejada.

6.3 Análise da Matriz de Projeções

Os eixos X,Y e Z são definidos logo no início do programa, quando o usuário posiciona o seu braço na posição horizontal e tecla <ENTER>. Neste momento os eixos

X e Z de referência são definidos em relação a rotação da placa sobre o eixo Y, e o eixo Y é considerado como sendo o próprio eixo Y da placa subtraído da sua componente Z. A partir disso, a rotação da placa fará com que os eixos rotacionados sejam descritos em função dos eixos de referência.

Pode ser visualizado na Figura 6.10 um exemplo onde o sistema foi inicializado e os eixos ainda encontram-se em sua posição original.

```

: 0.999968497952 - -0.00784308982113 - -0.00122015333375 :
: 0.00784366488333 - 0.999969132716 - 0.000458589033737 :
: 0.00121656270702 - -0.00046856963173 - 0.999999150461 :

```

Figura 6.10. Matriz de projeções com a placa na sua posição original.

Ao rotacionar a placa 45° sobre o eixo Z, os eixos da placa, é esperado encontrar a seguinte matriz:

$$M = \begin{vmatrix} 0.7071 & -0.7071 & 0 \\ 0.7071 & 0.7071 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

O resultado experimental é o apresentado na Figura 6.11.

```

: 0.711227709059 - -0.702798161433 - 0.0151675546668 :
: 0.702926881511 - 0.711241696303 - -0.00539517445956 :
: -0.00699607861939 - 0.0144911602924 - 0.999870410292 :

```

Figura 6.11. Matriz de projeções com a placa rotacionada 45° sobre o eixo Z.

A diferença entre os dados experimentais e os dados teóricos foram muito pequenos, com uma diferença máxima entre os valores das matrizes (considerando 4 dígitos significativos) de 0.0152. Esse erro é considerado pequeno dentro do escopo desse projeto. A matriz com as diferenças entre os termos experimentais e teóricos pode ser conferida abaixo.

$$M_{\text{diferença}} = \begin{vmatrix} 0.0041 & 0.0044 & 0.0152 \\ 0.0042 & 0.0041 & 0.0054 \\ 0.0069 & 0.0145 & 0.0001 \end{vmatrix}$$

E no caso de rotacionar a placa 45° sobre o eixo X, espera-se encontrar a matriz seguinte:

$$M = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 0.7071 & -0.7071 \\ 0 & 0.7071 & 0.7071 \end{vmatrix}$$

O resultado experimental encontra-se na Figura 6.12.

```

| 0.986750580417 - -0.107143124242 - 0.121835317248 |
| 0.162060773645 - 0.686623669293 - -0.708725444768 |
| -0.00771995419624 - 0.719074605096 - 0.694884450402 |

```

Figura 6.12. Matriz de projeções com a placa rotacionada 45° sobre o eixo X.

Esses resultados mostraram uma diferença máxima entre os dados teóricos e experimentais de 0.1621, o que ainda é pequeno considerando o escopo desse projeto.

$$M_{\text{diferença}} = \begin{vmatrix} 0.0132 & 0.1071 & 0.1218 \\ 0.1621 & 0.0205 & 0.0016 \\ 0.0077 & 0.0120 & 0.0122 \end{vmatrix}$$

Em posse desses dados, foram realizadas medidas experimentais para verificar o comportamento dos dados medidos relativos a matriz de projeções. O sistema com o ArduIMU e o transmissor sem fio foi colocado sobre a haste móvel do goniômetro, enquanto a haste fixa foi posicionada na posição de repouso calibrada inicialmente pelo programa. Foram realizados ensaios com o goniômetro no plano horizontal, sobre uma mesa, e no plano vertical, na lateral de uma bancada. Os ensaios no plano horizontal analisam principalmente a influência dos magnetômetros na matriz de rotação, enquanto no plano vertical, a principal variação é causada pelos acelerômetros. Na Figura 6.13 podemos ver a montagem dessas configurações. Na esquerda temos a medida no plano horizontal e na direita sobre o plano vertical.

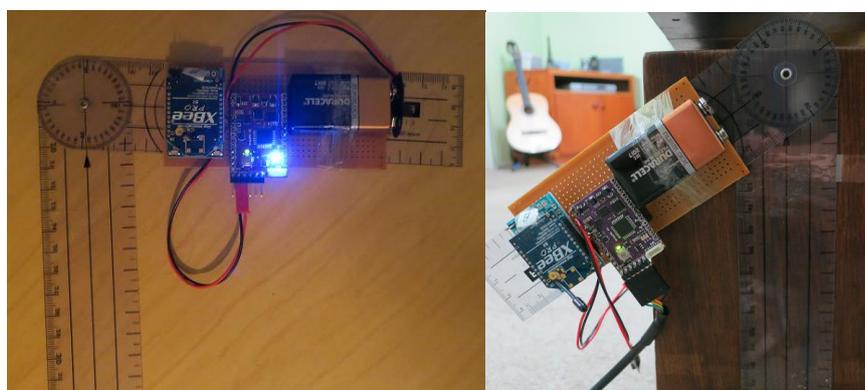


Figura 6.13. Montagem para medição de ângulos de rotação do sistema.

A partir dessas montagens, foram realizadas medições do ângulo rotacionado através do goniômetro e através da matriz de projeções. As rotações no caso desse experimento ocorreram sobre o eixo Z. O ângulo de rotação variou entre -90° e 90°, que são os ângulos principais a serem considerados em um exercício de reabilitação. Esses ângulos estão visualmente representados na Figura 6.14.

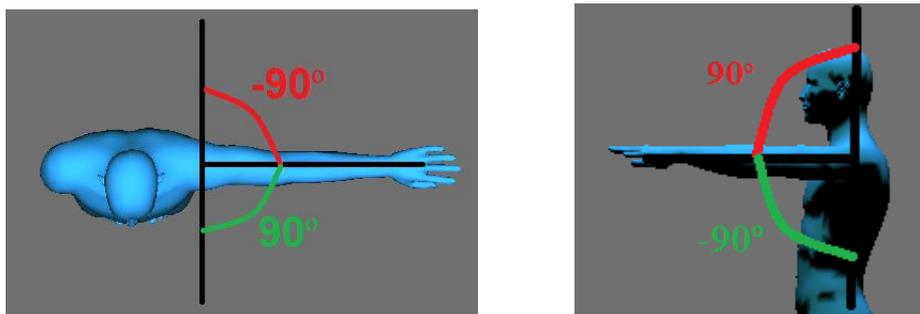


Figura 6.14. Ângulo principais de rotação.

Os resultados dessas medidas podem ser vistos na Figura 6.15 e na Figura 6.16. Os valores de r-quadrado ficaram muito próximos de um, o que define uma boa linearidade para o sistema. Devido a alta dificuldade em realizar o mesmo procedimento para os outros eixos causada pela fixação do sistema no goniômetro, só foram realizadas medidas para rotações sobre o eixo Z. O erro médio do sistema foi de $0,97^\circ$ em relação às medidas do goniômetro.

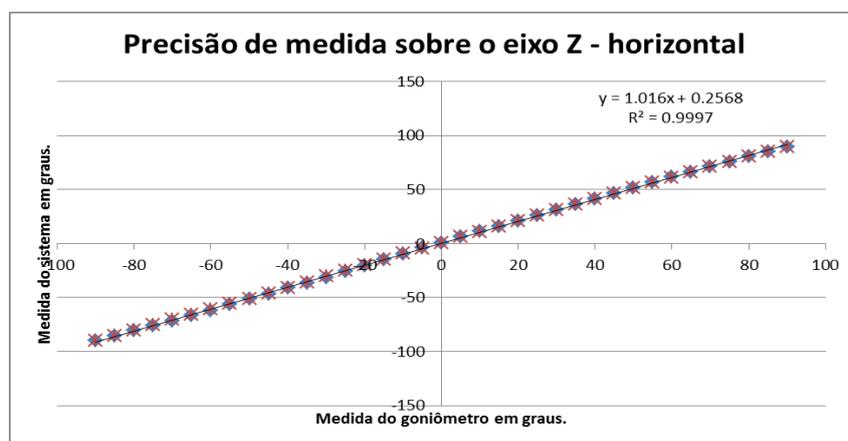


Figura 6.15. Gráfico da rotação sobre o eixo Z medida pelo sistema na horizontal.

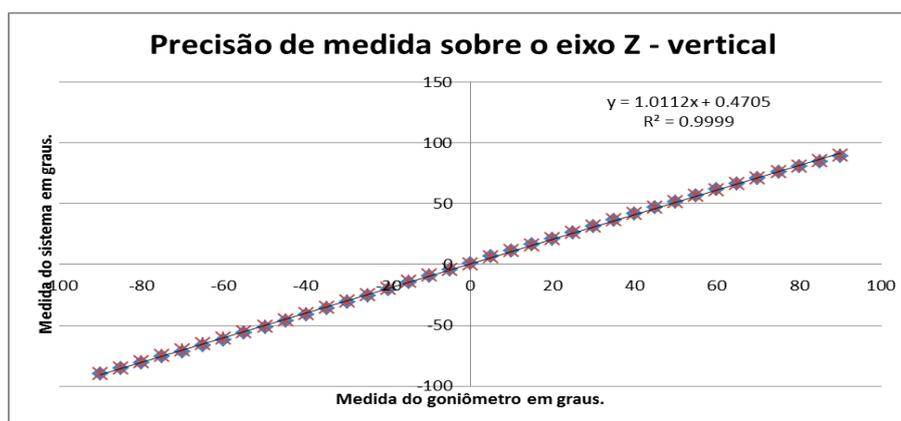


Figura 6.16. Gráfico da rotação sobre o eixo Z medida pelo sistema na vertical.

6.4 Análise da Conversão da Matriz de Projeções em Ângulos

É possível transformar a matriz de projeções em qualquer ângulo de rotação desejado, tanto ângulos de Euler como Tait Bryan. Através do modelo de Denavit e Hartenberg, a matriz de projeções é transformada em ângulos de rotação intrínsecos, o que não é vantajoso para esse projeto, tendo em vista que o Blender recebe os dados de rotação extrínsecos.

Portanto, fazendo uso do modelo de Tait Bryan extrínseco, a matriz de projeções é transformada em ângulos de rotação no mesmo modelo em que o Blender deve receber. Esse modelo foi implementado em Python e testado. Ele foi capaz de converter a matriz de projeções nos ângulos exatos correspondentes a ela, de maneira compatível com o Blender.

6.5 Funcionamento do Modelo Virtual

O modelo virtual gerado pelo software MakeHuman e animado pelo Blender mostrou-se muito amigável ao usuário. É possível identificar com clareza a posição do membro superior que está sendo medida pelos sensores. Contudo, o Blender não permite uma taxa de atualização muito alta da posição dos *bones* do modelo. Com uma taxa de envio de dados de 100Hz ao Blender, ele mostrou um atraso significativo na animação realizada. Como este projeto prevê o envio de dados a partir de 3 sensores distintos, a taxa de recebimento de dados do Blender é 3 vezes maior do que a taxa de envio de dados de cada sensor. Com uma taxa de envio de 20Hz por sensor, o Blender recebe dados a uma taxa de 60Hz, o que se mostrou satisfatório e sem atrasos nos ensaios realizados.

Na Figura 6.17 vemos o sistema completo em funcionamento. Os componentes do sistema presentes na foto estão numerados de 1 a 4, segue a descrição de cada um:

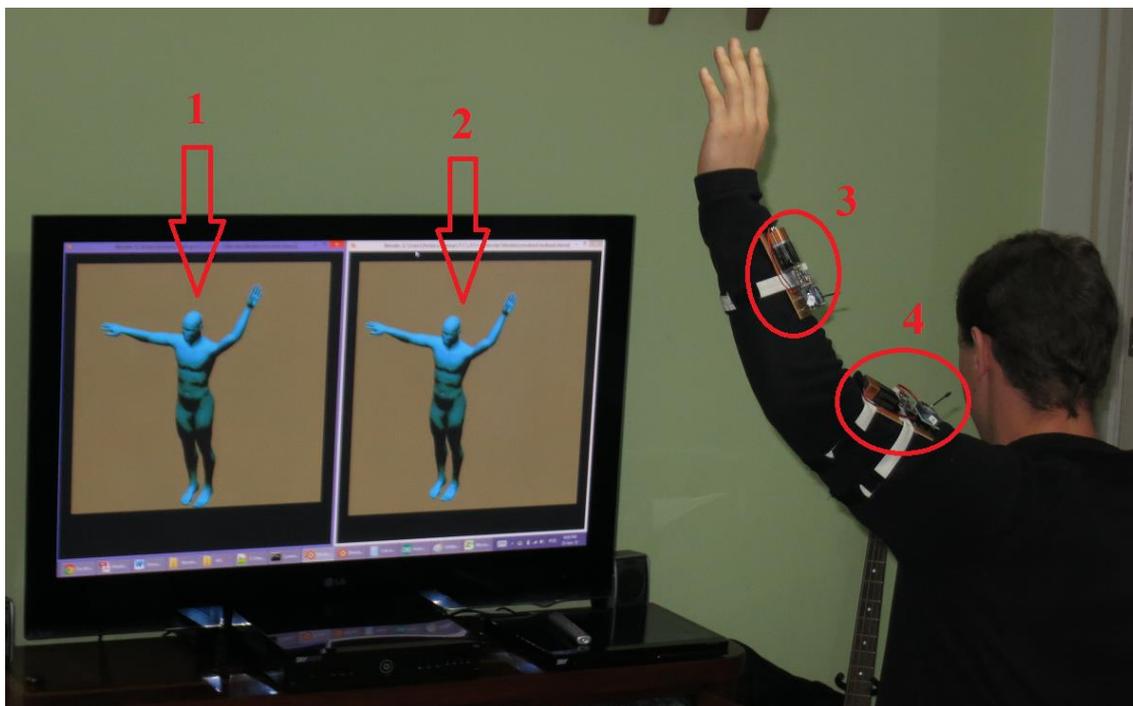


Figura 6.17. Teste experimental do sistema.

- 1: representação gráfica do movimento realizado pelo usuário;
- 2: representação gráfica do movimento padrão a ser executado pelo usuário;
- 3: sensor que monitora o movimento do antebraço do usuário;
- 4: sensor que monitora o movimento do braço do usuário.

6.6 Verificação do Movimento

Uma função é chamada para que os dados do movimento padrão sejam gerados. Esses dados são armazenados em um arquivo para que sirvam de comparação com o movimento realizado pelo usuário.

Quando o usuário começa a realizar o seu movimento, duas interfaces do programa Blender devem estar abertas, uma delas irá representar o movimento realizado pelo usuário e a outra será o movimento padrão do exercício que ele deve realizar. Isso ajuda o usuário a visualizar o movimento que ele deve estar realizando. Na Figura 6.18 temos um exemplo da tela do computador do usuário ao realizar o movimento. Ao final da execução do movimento do usuário, definido pelo tempo de execução do movimento padrão, os dois movimentos são comparados, através da soma do quadrado das diferenças e pelo erro máximo.

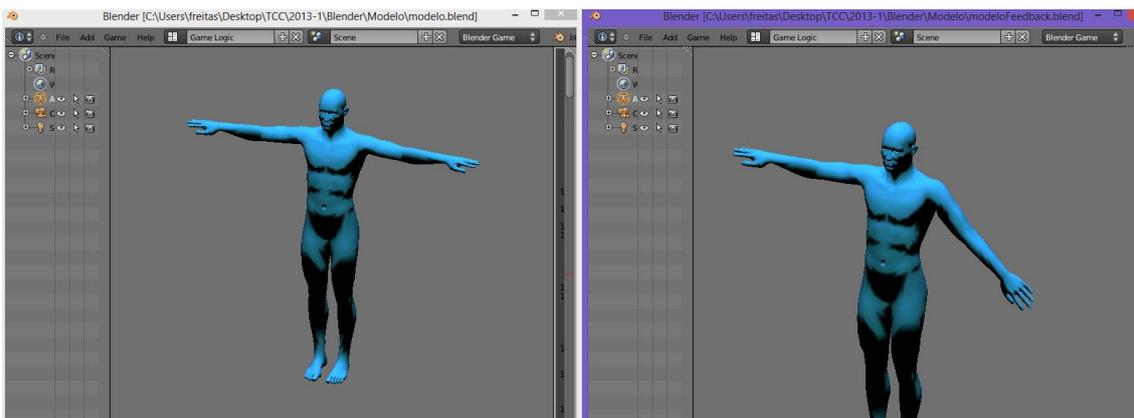


Figura 6.18. Movimento do usuário a esquerda e movimento padrão a direita.

Através dos testes realizados, a análise com defasagem temporal se mostrou muito importante. Em 100% dos testes realizados, quando o movimento foi realizado corretamente, ele foi realizado com uma defasagem temporal entre 0,2 e 2 segundos. A exatidão exigida pela verificação do movimento pode ser controlado pela variável g da equação (5.21), relativa ao erro angular médio tolerável. Quanto maior o valor de g , maior o erro aceitável no movimento. Pelos testes, o valor 5 para g se mostrou um bom valor, pois erros mínimos não comprometiam a correta execução do exercício e quando o usuário fugia do traçado correto, o movimento era considerado inválido.

7 CONCLUSÕES

O objetivo inicial do projeto foi alcançado com êxito. Uma ferramenta barata, portátil e de código livre foi desenvolvida para auxiliar pessoas que necessitem realizar exercícios de reabilitação motora, tais como a diagonal funcional de Kabot. O sistema faz uso exclusivamente de sensores inerciais e demonstra o movimento realizado para o usuário de forma gráfica e clara.

O resultado final já encontra-se funcional e pronto para ser utilizado em medidas de exercícios de reabilitação, contudo, diversas melhoras são almejadas para trabalhos futuros, sendo mais detalhadas no capítulo 8. É esperado que com essas melhoras, essa ferramenta possa auxiliar diversas pessoas a realizarem exercícios de recuperação motora de forma correta.

A reprodução do movimento pelo modelo virtual é um recurso muito poderoso e muito amigável ao usuário. Esse recurso se mostrou tanto útil na apresentação do resultado para o usuário quanto para o desenvolvimento do projeto. É muito importante um *feedback* gráfico para um entendimento fácil e rápido do que está acontecendo no programa.

Além da proposta inicial, esse sistema poderá ser utilizado para diferentes aplicações que necessitem de medidas estáticas de ângulos em 3 dimensões. O erro estático de $0,97^\circ$ torna essa ferramenta útil para diversas aplicações, como por exemplo, a medida de inclinações em articulações robóticas e próteses humanas, que aceitem um erro dessa magnitude..

8 PROPOSTAS DE TRABALHOS FUTUROS

Apesar do bom funcionamento do trabalho, existem diversos pontos que podem, e devem, ser melhorados nesse projeto. A maioria desses pontos fazem referência a uma melhor interface entre usuário e programa e alguns quesitos técnicos. Entre os pontos de usabilidade, acredito que devem ser destacados os seguintes:

- o programa deve ter uma interface gráfica para se comunicar com o usuário, fazendo assim com que ele seja mais amigável ao usuário;
- o modelo virtual pode ter uma renderização mais real, adicionando vestes ao modelo e mesmo deixando o usuário selecionar o sexo do modelo;
- o *feedback* ao usuário pode ser feito através de um modelo sobreposto ao movimento do usuário, com uma determinada transparência;
- reduzir tamanho e peso do hardware, deixando o sistema mais leve e portátil.

Entre os pontos técnicos a serem melhorados no futuro, destacam-se os seguintes:

- expandir o monitoramento dos membros para captar movimentos como abrir e fechar a mão.
- avaliar o desempenho do movimento do usuário independentemente por membro (braço, antebraço e mão) e por intervalos de tempo. Dessa maneira, será possível informá-lo precisamente onde e quando o exercício foi mal executado, no caso de movimento errado;
- usar banco de dados para armazenar os dados relativos aos movimentos, por enquanto ainda estão sendo armazenados em arquivos;
- monitorar o movimento do tronco do usuário, adicionando um sensor, pois, ao movimentar o corpo, o sistema interpreta incorretamente a inclinação do braço;
- adaptar o programa para considerar movimentos com aceleração, dessa maneira poderá ser utilizado para aplicações relacionadas ao esporte em alto nível.

Ao realizar essas melhoras, o desenvolvedor não deve esquecer o caráter modular e de baixo custo intrínseco a esse projeto. Os softwares envolvidos devem ser de código aberto e os sensores devem ser de baixo custo.

Por fim, esse projeto abre oportunidades para uma gama muito variada de aplicações tanto na área médica como na área esportiva.

REFERÊNCIAS

ARENDI, C. F; DUARTE, T, P. **Princípios da Anatomia Humana**. São Paulo: CI, 1996.

BOULIC, R. and RENAULT, O. **3D Hierarchies for Animation**. In: *New Trends in Animation and Visualization*, Editado por Nadia Magnenat-Thalmann and Daniel Thalmann, John Wiley & Sons ltd., Inglaterra, 1991

HAMILL J, KNUTZEN K. (2003). **Biomechanical basis of human movement**. 2nd ed. Baltimore: Lippincott Williams & Wilkins.

AGGARWAL, J.K. and CAI, Q., **Human motion analysis: a review**, 1999.

WELCH, G. and FOXLIN, E., **Motion tracking: no silver bullet, but a respectable arsenal**, IEEE Computer Graphics and Applications, Vol. 22 No. 6, pag. 24-38. 2002.

BHATIA S., et al, **3D human limb detection using space carving and multi-view eigen models, in: Proceedings of the IEEE Workshop on Articulated and Nonrigid Motion**. Washington, DC, 2004.

EMIL J., ALEKSANDAR M. et al., **A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation**. Journal of neuroengineering & rehabilitation [1743-0003] volume 2 página: 6, 2005.

GODFREY, A., CONWAY, R. et al., **Direct measurement of human movement by accelerometry**. Medical Engineering & Physics 30 páginas: 1364–1386. 2008.

ZHOU, H. e HU, H.. **Inertial sensors for motion detection of human upper limbs**. Sensor Review. Volume 27, número 2, páginas: 151–158. 2007.

DIY Drones, **Introduction to ArduIMU V3**, disponível em <<https://code.google.com/p/ardu-imu/wiki/IntroductionPage>> acessado em julho de 2013.

BARBOUR, N. and SCHMIDT, G. **Inertial Sensor Technology Trends**. IEEE SENSORS JOURNAL, Vol. 1, oO. 4, Dezembro, 2001.

IEEE Standard for Information technology— **Telecommunications and information exchange between systems**— Local and metropolitan area networks— Specific requirements— Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)

RIGAUD, Emmanuel. **Dynamique des Systèmes Polyarticulés**. Notas de aula na École Centrale de Lyon, Lyon, França. 2011.

RAMOS, Luiza. **Realidade Virtual e Acelerometria no Desenvolvimento de Tecnologia Assistiva: um sistema de apoio à reabilitação postural na caminhada de idosos**. Programa de Iniciação Científica, UFRGS, 2011.

GALO Maurício, Clésio L. Tozzi. **A representação de matrizes de rotação e o uso de quatérnios em ciências geodésicas**.

MALAN, DF, **Euler Angles x-convention**, disponível em: <<http://commons.wikimedia.org/wiki/File:EulerX.png>>, 2004.

BRITS, Lionel, **Tait-Brian angles**, disponível em : <<http://en.wikipedia.org/wiki/File:Taitbrianangles.svg>> 2009.

FAUGERAS, O.; **Three-Dimensional Computer Vision - A Geometric Viewpoint**, The MIT press, Cambridge, England, 1993.

KEITH R. Symon, **Mechanics**, terceira edição, 1971.

CONTO, Eduardo Fernandes. **Relatório de Iniciação Científica**, UFRGS, DELET, 2012.

APÊNDICE A <MATRIZES DE DENAVIT E HARTENBERG>

Neste apêndice são explicitadas todas as matrizes de passagem diretas e inversas do modelo de Denavit e Hartenberg explicitado na Figura 5.14.

Matizes de passagem direta:

$$T_{12} = \begin{vmatrix} \cos(t_1) & 0 & -\sin(t_1) \\ \sin(t_1) & 0 & \cos(t_1) \\ 0 & -1 & 0 \end{vmatrix}$$

$$T_{23} = \begin{vmatrix} \cos(t_2) & 0 & -\sin(t_2) \\ \sin(t_2) & 0 & \cos(t_2) \\ 0 & -1 & 0 \end{vmatrix}$$

$$T_{34} = \begin{vmatrix} \cos(t_3) & 0 & \sin(t_3) \\ \sin(t_3) & 0 & -\cos(t_3) \\ 0 & 1 & 0 \end{vmatrix}$$

$$T_{45} = \begin{vmatrix} \cos(t_4) & 0 & \sin(t_4) \\ \sin(t_4) & 0 & -\cos(t_4) \\ 0 & 1 & 0 \end{vmatrix}$$

$$T_{56} = \begin{vmatrix} \cos(t_5) & 0 & -\sin(t_5) \\ \sin(t_5) & 0 & \cos(t_5) \\ 0 & -1 & 0 \end{vmatrix}$$

$$\begin{vmatrix} \cos(t_6) & 0 & \sin(t_6) \end{vmatrix}$$

$$T67 = \begin{vmatrix} \sin(t6) & 0 & -\cos(t6) \\ 0 & 1 & 0 \end{vmatrix}$$

$$T78 = \begin{vmatrix} \cos(t7) & -\sin(t7) & 0 \\ \sin(t7) & \cos(t7) & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$T14 = \begin{vmatrix} \sin(t1)*\sin(t3) + \cos(t1)*\cos(t2)*\cos(t3) & -\cos(t1)*\sin(t2) & \cos(t1)*\cos(t2)*\sin(t3) - \cos(t3)*\sin(t1) \\ \cos(t2)*\cos(t3)*\sin(t1) - \cos(t1)*\sin(t3) & -\sin(t1)*\sin(t2) & \cos(t1)*\cos(t3) + \cos(t2)*\sin(t1)*\sin(t3) \\ -\cos(t3)*\sin(t2) & -\cos(t2) & -\sin(t2)*\sin(t3) \end{vmatrix}$$

$$T46 = \begin{vmatrix} \cos(t4)*\cos(t5) & -\sin(t4) & -\cos(t4)*\sin(t5) \\ \cos(t5)*\sin(t4) & \cos(t4) & -\sin(t4)*\sin(t5) \\ \sin(t5) & 0 & \cos(t5) \end{vmatrix}$$

$$T68 = \begin{vmatrix} \cos(t6)*\cos(t7) & -\cos(t6)*\sin(t7) & \sin(t6) \\ \cos(t7)*\sin(t6) & -\sin(t6)*\sin(t7) & -\cos(t6) \\ \sin(t7) & \cos(t7) & 0 \end{vmatrix}$$

Matrizes de passagem inversa:

$$T87 = \begin{vmatrix} \cos(t7) & \sin(t7) & 0 \\ -\sin(t7) & \cos(t7) & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$T76 = \begin{vmatrix} \cos(t6) & \sin(t6) & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

$$\begin{vmatrix} \sin(t_6) & -\cos(t_6) & 0 \end{vmatrix}$$

$$T_{65} = \begin{vmatrix} \cos(t_5) & \sin(t_5) & 0 \\ 0 & 0 & -1 \\ -\sin(t_5) & \cos(t_5) & 0 \end{vmatrix}$$

$$T_{54} = \begin{vmatrix} \cos(t_4) & \sin(t_4) & 0 \\ 0 & 0 & 1 \\ \sin(t_4) & -\cos(t_4) & 0 \end{vmatrix}$$

$$T_{43} = \begin{vmatrix} \cos(t_3) & \sin(t_3) & 0 \\ 0 & 0 & 1 \\ \sin(t_3) & -\cos(t_3) & 0 \end{vmatrix}$$

$$T_{32} = \begin{vmatrix} \cos(t_2) & \sin(t_2) & 0 \\ 0 & 0 & -1 \\ -\sin(t_2) & \cos(t_2) & 0 \end{vmatrix}$$

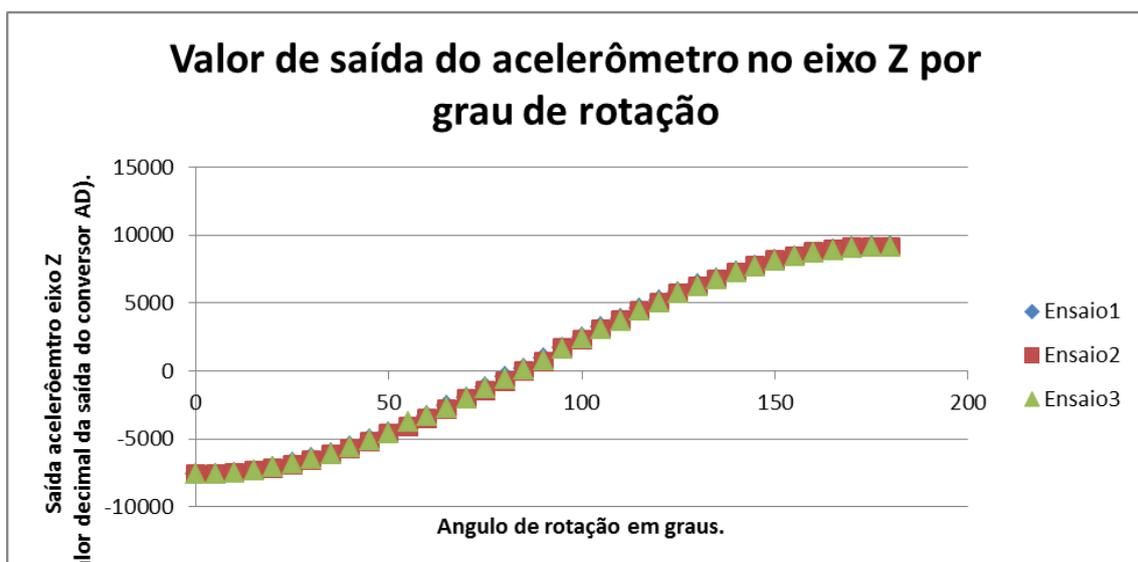
$$T_{21} = \begin{vmatrix} \cos(t_1) & \sin(t_1) & 0 \\ 0 & 0 & -1 \\ -\sin(t_1) & \cos(t_1) & 0 \end{vmatrix}$$

ANEXO A

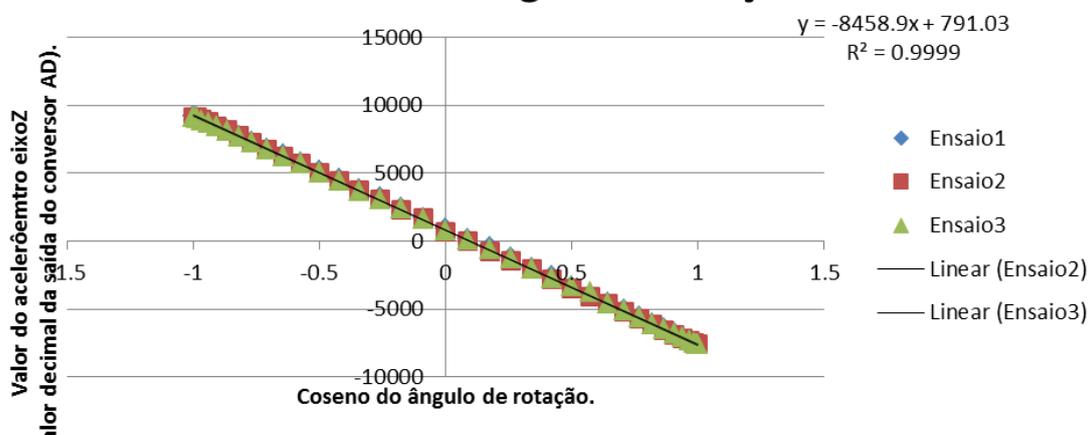
Neste anexo são mostrados todos as medidas relativas a calibração dos acelerômetros.

Rotacao (θ)	cos(θ)	sen(θ)	Ensaio 1			Ensaio 2			Ensaio 3		
			Eixo X	EixoY	Eixo Z	Eixo X	EixoY	Eixo Z	Eixo X	EixoY	Eixo Z
0	1	0	331	8197	-7593	-295	8189	-7631	-332	8192	-7619
5	0.996	0.087156	284	8180	-7559	240	8173	-7614	490	8149	-7574
10	0.985	0.173648	1066	8079	-7470	1177	8062	-7535	1198	8051	-7488
15	0.966	0.258819	1673	7952	-7295	2189	7820	-7368	1903	7900	-7357
20	0.94	0.34202	2456	7718	-7075	2633	7669	-7134	2541	7692	-7117
25	0.906	0.422618	3080	7474	-6767	3251	7403	-6869	3260	7392	-6803
30	0.866	0.5	3830	7092	-6435	3902	7063	-6560	3859	7086	-6507
35	0.819	0.573576	4517	6647	-6015	4470	6687	-6163	4462	6701	-6100
40	0.766	0.642788	5010	6260	-5573	5067	6220	-5734	5076	6215	-5673
45	0.707	0.707107	5601	5717	-5066	5578	5750	-5216	5564	5761	-5159
50	0.643	0.766044	5941	5351	-4520	6078	5201	-4595	6026	5262	-4575
55	0.574	0.819152	6482	5554	-3886	6509	4631	-4082	6437	4725	-3770
60	0.5	0.866025	6844	4103	-3318	6881	4018	-3470	6876	4022	-3350
65	0.423	0.906308	7221	4471	-2595	7218	3373	-2794	7200	3413	-2712
70	0.342	0.939693	7469	2736	-1924	7473	2747	-2089	7457	2791	-2039
75	0.259	0.965926	7654	2162	-1196	7672	2065	-1420	7673	2070	-1313
80	0.174	0.984808	7790	1612	-444	7820	1374	-757	7817	1458	-571
85	0.087	0.996195	7888	911	206	7918	646	-48	7918	692	59

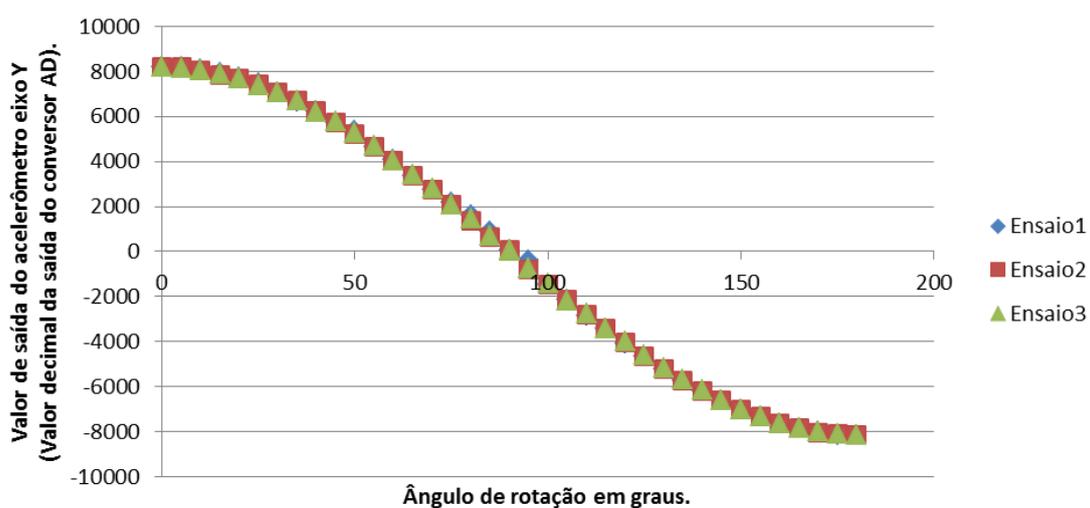
90	6E-17	1	7942	93	917	7950	83	703	7946	12	774
95	-0.087	0.996195	7941	-406	1703	7912	-814	1683	7923	-744	1575
100	-0.174	0.984808	7812	-1489	2501	7820	-1448	2308	7835	-1409	2344
105	-0.259	0.965926	7665	-2144	3272	7664	-2138	3081	7665	-2162	3058
110	-0.342	0.939693	7438	-2841	3867	7462	-2781	3704	7475	-2765	3679
115	-0.423	0.906308	7195	-3433	4557	7205	-3405	4384	7196	-3418	4387
120	-0.5	0.866025	6860	-4077	5157	6881	-4042	5017	6898	-4004	5053
125	-0.574	0.819152	6490	-4653	5768	6508	-4640	5706	6513	-4618	5682
130	-0.643	0.766044	6065	-5214	6359	6080	-5201	6254	6096	-5173	6221
135	-0.707	0.707107	5594	-5726	6853	5610	-5712	6756	5631	-5704	6728
140	-0.766	0.642788	5119	-6176	7366	5068	-6219	7254	5143	-6158	7233
145	-0.819	0.573576	4525	-5697	7768	4555	-6625	7787	4589	-6600	7666
150	-0.866	0.5	3933	-7028	8159	3967	-7004	8155	3997	-7005	8079
155	-0.906	0.422618	3317	-7350	8471	3338	-7349	8460	3370	-7335	8390
160	-0.94	0.34202	2696	-7623	8718	2696	-7621	8752	2735	-7611	8662
165	-0.966	0.258819	2033	-7836	8937	2058	-7838	8965	2059	-7841	8888
170	-0.985	0.173648	1329	-8002	9087	1343	-8022	9078	1408	-7998	8998
175	-0.996	0.087156	588	-8129	9157	716	-8113	9145	658	-8121	9104
180	-1	1.23E-16	-58	-8147	9177	-62	-8167	9155	-29	-8161	9150



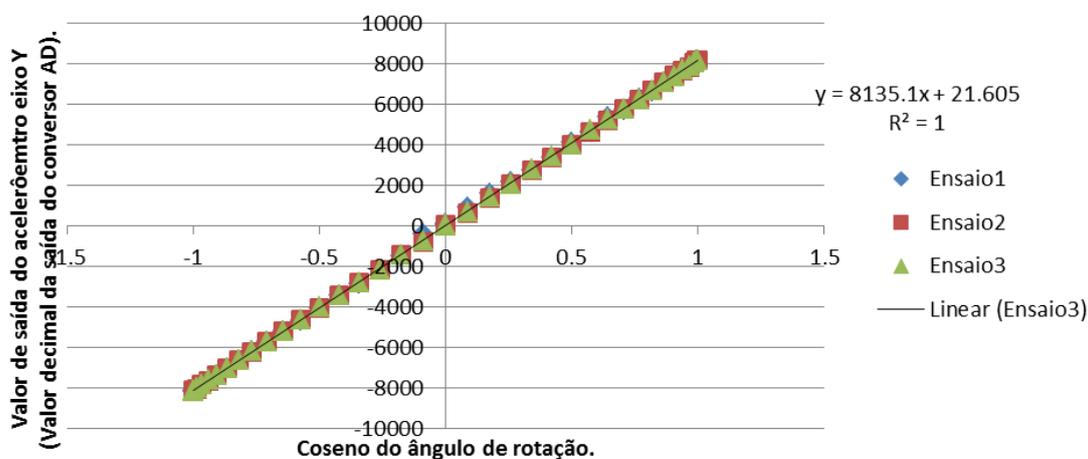
Valor de saída do acelerômetro no eixo Z pelo coseno do ângulo de rotação



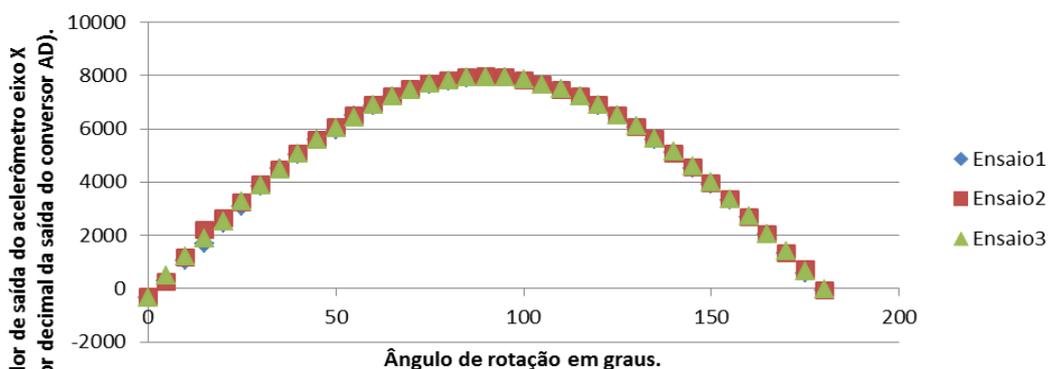
Valor de saída do acelerômetro no eixo Y pelo ângulo de rotação



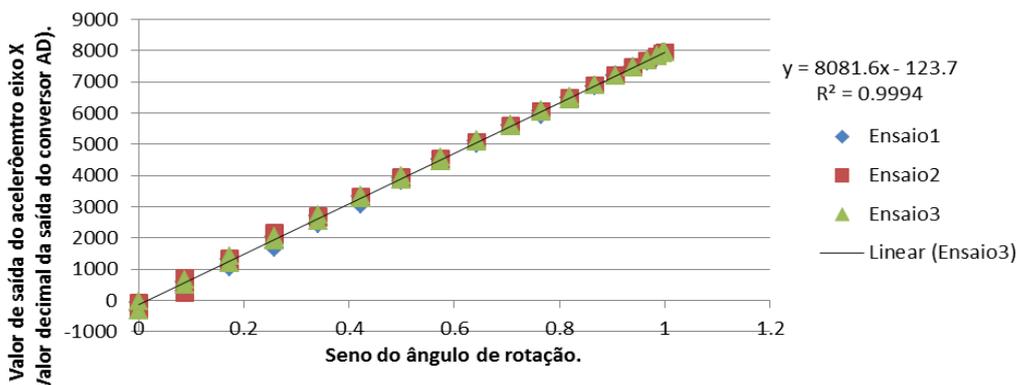
Valor de saída do acelerômetro no eixo Y pelo coseno do ângulo de rotação



Valor de saída do acelerômetro no eixo X pelo ângulo de rotação



Valor de saída do acelerômetro no eixo X pelo seno do ângulo de rotação

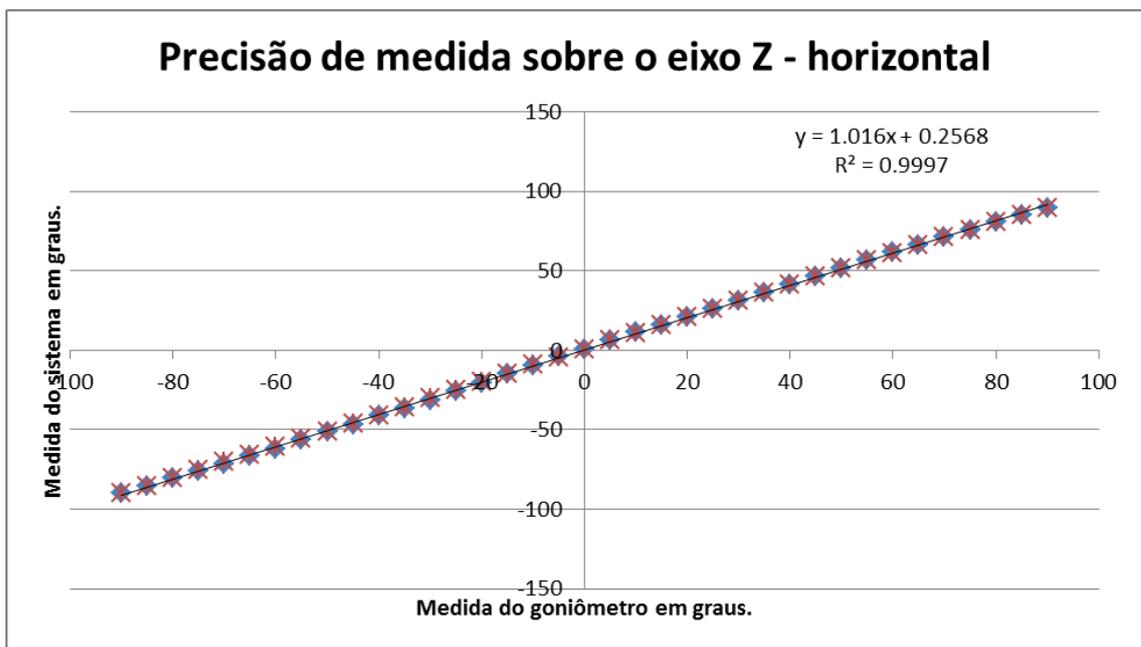


ANEXO B

Seguem as tabelas e gráficos com os dados referentes as medidas experimentais realizadas com o goniômetro na horizontal e vertical sobre a medida de graus de rotação sobre o eixo Z do sistema.

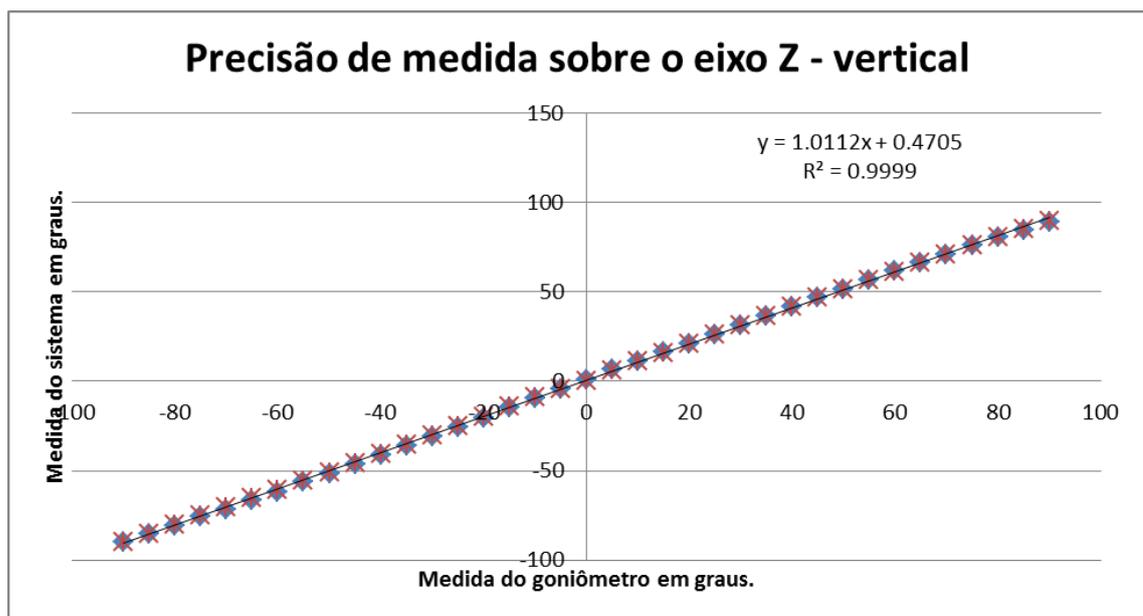
Rotacao eixo Z - Horizontal				
Goniometro	Ensaio 1	Erro1	Ensaio 2	Erro2
-90	-90	0	-90	0
-85	-85.6	0.6	-85.2	0.2
-80	-80.6	0.6	-80.3	0.3
-75	-75.7	0.7	-75.1	0.1
-70	-71.4	1.4	-70.4	0.4
-65	-66.7	1.7	-65.5	0.5
-60	-61.8	1.8	-60.6	0.6
-55	-56.4	1.4	-55.8	0.8
-50	-51.4	1.4	-50.8	0.8
-45	-46.7	1.7	-45.7	0.7
-40	-41	1	-40.6	0.6
-35	-36.2	1.2	-35.7	0.7
-30	-31.1	1.1	-30.3	0.3
-25	-25.5	0.5	-24.9	0.1
-20	-20.4	0.4	-19.8	0.2
-15	-14.7	0.3	-14.3	0.7
-10	-9.2	0.8	-8.8	1.2
-5	-4	1	-4.5	0.5
0	0.9	0.9	0.3	0.3
5	6.5	1.5	6.3	1.3
10	11.3	1.3	11	1
15	16.3	1.3	16	1
20	21.2	1.2	20.9	0.9
25	26.3	1.3	26	1
30	31.4	1.4	31.1	1.1
35	36.6	1.6	36.3	1.3
40	41.8	1.8	41.7	1.7
45	46.9	1.9	46.7	1.7

50	51.6	1.6	51.6	1.6
55	56.9	1.9	56.7	1.7
60	61.6	1.6	61.4	1.4
65	66.5	1.5	66.6	1.6
70	71.2	1.2	71.1	1.1
75	75.9	0.9	76	1
80	80.8	0.8	80.8	0.8
85	84.9	0.1	85.3	0.3
90	89.3	0.7	89.9	0.1
	Erro médio:	1.137838	Erro médio:	0.8



Rotacao eixo Z - Vertical				
Goniômetro	Ensaio 1	Erro1	Ensaio 2	Erro2
-90	-90	0	-89.99	0.01
-85	-85.6	0.6	-85.2	0.2
-80	-80.6	0.6	-80.3	0.3
-75	-75.7	0.7	-75.1	0.1
-70	-71.4	1.4	-70.4	0.4
-65	-66.7	1.7	-65.5	0.5
-60	-61.8	1.8	-60.6	0.6
-55	-56.4	1.4	-55.8	0.8
-50	-51.4	1.4	-50.8	0.8
-45	-46.7	1.7	-45.7	0.7
-40	-41	1	-40.6	0.6
-35	-36.2	1.2	-35.7	0.7

-30	-31.1	1.1	-30.3	0.3
-25	-25.5	0.5	-24.9	0.1
-20	-20.4	0.4	-19.8	0.2
-15	-14.7	0.3	-14.3	0.7
-10	-9.2	0.8	-8.8	1.2
-5	-4	1	-4.5	0.5
0	0.9	0.9	0.3	0.3
5	6.5	1.5	6.3	1.3
10	11.3	1.3	11	1
15	16.3	1.3	16	1
20	21.2	1.2	20.9	0.9
25	26.3	1.3	26	1
30	31.4	1.4	31.1	1.1
35	36.6	1.6	36.3	1.3
40	41.8	1.8	41.7	1.7
45	46.9	1.9	46.7	1.7
50	51.6	1.6	51.6	1.6
55	56.9	1.9	56.7	1.7
60	61.6	1.6	61.4	1.4
65	66.5	1.5	66.6	1.6
70	71.2	1.2	71.1	1.1
75	75.9	0.9	76	1
80	80.8	0.8	80.8	0.8
85	84.9	0.1	85.3	0.3
90	89.3	0.7	89.9	0.1
	Erro médio:	1.137838	Erro médio:	0.80027



ANEXO C

Este anexo corresponde ao artigo desenvolvido para o TG1 deste projeto de graduação.

Sensores Inerciais e Protocolo ZigBee no Desenvolvimento de Tecnologia Assistiva: Reabilitação Motora dos Membros Superiores Humanos.

Jonas Crauss Rodrigues de Freitas¹

¹Instituto de Informática / Departamento de Engenharia Elétrica – Universidade Federal do Rio Grande do Sul (UFRGS)
Porto Alegre – RS – Brazil
jcrfreitas@inf.ufrgs.br

***Abstract.** The following paper has the goal to describe the tools and the approach to be used in the implementation of the graduation work for the Computation Engineering course from UFRGS as well as show some preliminary results achieved. This work aims to create a small, robust and low-cost device able to characterize the 3D movement of a human arm and compare it with a standard movement. Systems like this one have several applications in physiotherapy and rehabilitation of motor skills for posttraumatic patients.*

***Resumo.** O presente artigo tem como objetivo descrever as ferramentas e a abordagem a serem utilizadas na realização do trabalho de graduação do curso de Engenharia de Computação da UFRGS assim como alguns resultados preliminares obtidos. Este trabalho objetiva em criar um dispositivo compacto, robusto e de baixo custo capaz de caracterizar o movimento de um braço humano em 3D e compará-lo com um movimento pré-definido. Sistemas como este possuem diversas aplicações na área da saúde, como fisioterapia e reabilitação de funções motoras para pacientes pós-traumas.*

1. Introdução

A busca pela medição de movimentos dos membros humanos tem sido alvo de pesquisas científicas nas últimas décadas. Pessoas que sofreram traumas e que possuam necessidade de realizar exercícios constantes de reabilitação física, como pessoas que sofreram um AVC (Acidente Vascular Cerebral), acidentes de carro, entre outros, necessitam de auxílio médico especializado para readquirir as suas funções motoras normais. O caminho para a total reabilitação é longo e intenso, e requer exercícios de repetição por semanas ou mesmo meses [Zeng et al. 2005].

A realização desse tratamento em um ambiente hospitalar, ou mesmo simplesmente acompanhado de um profissional da área, é muito caro, pois requer muito tempo destes profissionais. Entretanto, se o paciente realizar o exercício de forma autônoma, é bastante provável que ele os execute de forma errônea, o que impossibilitará a sua correta reabilitação.

Uma solução para este problema é utilizar uma ferramenta assistiva que possibilitará a correta execução dos exercícios. Em função disso, pretende-se criar um dispositivo capaz de caracterizar o movimento realizado pelo paciente, compará-lo com o movimento correto do exercício que ele deve realizar, e informá-lo se ele está executando-o corretamente. Desta forma, o paciente será capaz de realizar o tratamento de forma autônoma e correta, sem a necessidade de deslocamento para um ambiente hospitalar e sem necessitar de um profissional o acompanhando. Além disso, o profissional poderá analisar o resultado do tratamento do paciente através dos dados armazenados no banco de dados e então fornecer um *feedback* para o paciente.

Portanto o presente trabalho visa desenvolver uma ferramenta assistiva capaz de caracterizar o movimento de um braço humano em 3D. Algumas necessidades devem ser obrigatoriamente satisfeitas por essa ferramenta. São elas:

- o sistema deve perceber os movimentos realizados pelo braço do usuário em 3 dimensões.
- o movimento deve gerar dados em tempo real para comparação instantânea com um movimento pré-definido.
- os dados gerados devem ser armazenados em um banco de dados para que o movimento realizado possa ser recriado para análise posterior.
- o sistema deve ser compacto para que não se torne incômodo de ser utilizado pelo usuário e que não limite os seus movimentos.
- o sistema deve ser robusto o suficiente para ser utilizado em ambientes não controlados, como por exemplo, na residência do usuário.
- o sistema deve ser de baixo custo, permitindo que cada paciente possa ter uma unidade própria em sua residência.

2. Monitorando o Movimento de Membros do Corpo Humano

A análise do movimento do corpo humano pode ser dividida em duas categorias: estática, que estuda tudo relacionado aos sistemas sem movimento, e dinâmico, que estuda os sistemas em movimento. A categoria dinâmica pode ainda ser dividida em outras duas subcategorias: cinemática, estuda o deslocamento em tempo e espaço, e cinética, estuda as forças agindo no sistema que influenciam no movimento. A

cinemática ainda pode ser dividida em linear, movimento retilíneo ou curvo onde todos os pontos de um corpo se deslocam igualmente e angular, que envolve rotação ao redor de um ponto fixo e velocidade diferentes ao longo do corpo [Hamill, 2003]. A Figura 2.1 apresenta de forma clara e sucinta as áreas supracitadas da análise do movimento humano.

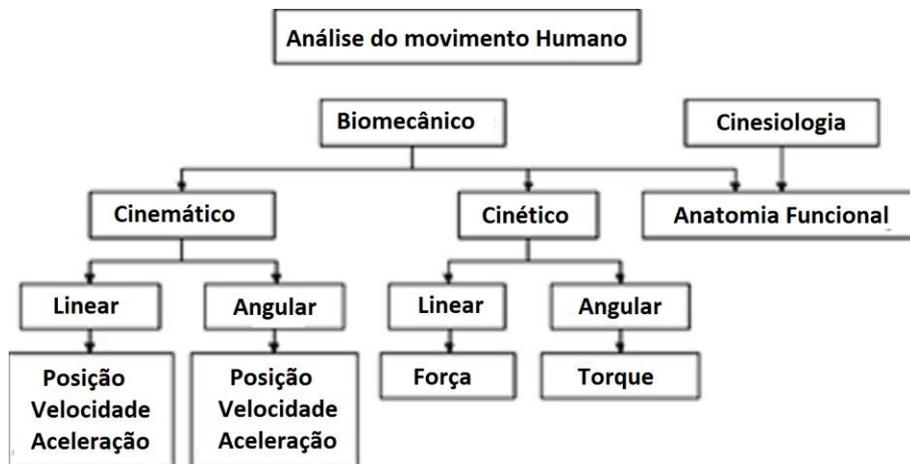


Figura 0.1. Análise de tipos de movimentos [adaptado de Hamill, 2003].

No caso da caracterização do movimento do membro superior, o objetivo desse trabalho é apenas a análise cinemática do movimento. Não será feita uma análise das forças envolvidas no movimento. As articulações a serem analisadas serão as seguintes: ombro, cotovelo e pulso. Devido a grande complexidade de movimentos possíveis com os dedos da mão e pela baixa importância em caracterizar precisamente esses movimentos para recuperação de pacientes com deficiências, não iremos nos interessar por caracterizar esses movimentos. A simples ação de abrir e fechar a mão é relevante para os exercícios de reabilitação, mas como é necessário uma abordagem diferente para caracterizar esse movimento, ele foi considerado como fora do escopo desse trabalho.

Vale destacar também que em todas as articulações existem pequenos movimentos de translação e, em algumas, o movimento de translação é significativo, como no caso do ombro [Aggarwal, J.K, 1999]. Contudo esses movimentos não serão levados em conta para os exercícios de reabilitação considerados nesse trabalho. O grande interesse desse trabalho é analisar os movimentos de rotação dessas articulações. Dessa maneira, é necessário apenas caracterizar a inclinação de cada membro (braço, antebraço e mão) para obter a posição completa do membro superior no espaço.

2.1. Acelerômetros, Giroscópios e Magnetômetros

É possível caracterizar o movimento humano através de marcadores em locais específicos do corpo monitorados por câmeras, demonstrado por Aggarwal e Cai (1999) e por Welch Foxlin (2002). Contudo, esse sistema sofre do problema de oclusão, ou seja, em certas posições, os marcadores ficarão escondidos e as câmeras simplesmente não irão detectá-lo [Bhatia et al. 2004]. Como os movimentos de reabilitação são bastante complexos, esse problema não é aceitável.

Uma forma eficiente de analisar o movimento realizado é através de sensores inerciais, que medem aceleração (acelerômetros) e velocidade angular (giroscópios). Esses sensores são pequenos, leves e adequados para caracterizar o movimento de membros humanos [Emil, 2005], [Zhou e Hu, 2007]. Acelerômetros já foram utilizados

para monitorar o movimento do corpo humano em 1950, mas devido ao seu custo elevado, tornou-se inviável. Com o avanço da tecnologia e a baixa no preço desses sensores, voltaram a ser utilizados no início da década de 70 para medir o movimento do corpo humano e, a partir de então, o seu uso para essa finalidade tem aumentado [Godfrey et al. 2008].

2.2. Acelerômetros

Acelerômetros são sensores inerciais capazes de medir a aceleração resultante em um dado instante de tempo. O princípio de funcionamento da maioria dos acelerômetros é baseado no sistema massa-mola, ou seja, a deformação da mola é proporcional à aceleração resultante naquele eixo. São bastante utilizados para análises cinemáticas biomecânicas. O valor resultante da aceleração pode ser utilizado para que seja encontrada a inclinação do corpo medido (usando a gravidade como referência). Para obter a inclinação exata do corpo em 3D, é necessário medir a aceleração nos três eixos, seja através de três acelerômetros posicionados ortogonalmente ou um acelerômetro triaxial, que realiza a mesma função.

Um dos modelos de acelerômetros existentes no mercado, e de baixo custo, é o acelerômetro capacitivo. Este tipo de acelerômetro funciona através da alteração da capacitância com a aceleração. Uma das placas paralelas do capacitor é colocada em uma superfície que se desloca proporcional a aceleração exercida e a outra é fixa. Quando a aceleração sobre a placa varia, a tensão sobre o capacitor também varia, criando a saída do sensor [Godfrey et al, 2008].

2.3. Giroscópios

Giroscópio é um diferente tipo de sensor inercial, capaz de medir a velocidade angular em certo instante. Giroscópios mecânicos, ou micromecânicos (MEMS), funcionam de acordo com a teoria de que quando uma velocidade angular é aplicada a um corpo, uma força de Coriolis é gerada, proporcional à velocidade angular. Essa força é, então, transformada em sinal elétrico, gerando assim o sinal de saída do sensor. Na mesma lógica dos acelerômetros, podemos utilizar três giroscópios ortogonais para medir a velocidade angular nos três eixos. [Barbour, 2001].

2.4. Magnetômetros

Magnetômetros são sensores capazes de detectar a direção do campo magnético da Terra. Eles são normalmente utilizados em conjunto com giroscópios para corrigir possíveis erros de medida dos giroscópios. Esses sensores devem ser calibrados de acordo com a localidade da sua utilização, pois o norte magnético da Terra não coincide com o Norte geográfico. Além disso, o campo magnético da Terra se altera ano após ano, necessitando assim de a calibração desses sensores sejam ajustadas de tempos em tempos.

No caso particular de Porto Alegre, a inclinação magnética é de $16^{\circ} 8' 56''$ a oeste e está sendo alterada de $9.0'$ a oeste por ano [www.ngdc.noaa.gov]. Comparando a direção do norte magnético da Terra com a direção da placa, é possível detectar movimentos de rotação desta.

redes que utilizam dispositivos de curto alcance, baixa taxa de transmissão de dados e baixo consumo de potência (normalmente alimentados por bateria), portanto, ideal para a aplicação deste trabalho.

No protocolo IEEE 802.15.4, estão definidos dois tipos de dispositivos, o *Full Function Device* (FFD) e o *Reduced Function Device* (RFD). Na rede deve existir somente um coordenador. Este coordenador é responsável pela inserção de novos dispositivos na rede. O coordenador deve ser obrigatoriamente um dispositivo FFD. Dispositivos FFD podem gerar e enviar dados ou funcionar como roteadores. Dispositivos RFD podem apenas gerar e enviar dados, não sendo capazes de associar mais nenhum dispositivo ligado a eles, a sua grande vantagem é um consumo reduzido de energia por possuir menos funcionalidades.

Este protocolo prevê a criação de uma WPAN (*Wireless Personal Area Network*), que pode possuir duas arquiteturas básicas diferentes: Arquitetura estrela ou P2P, como pode ser visto na Figura 4.1.

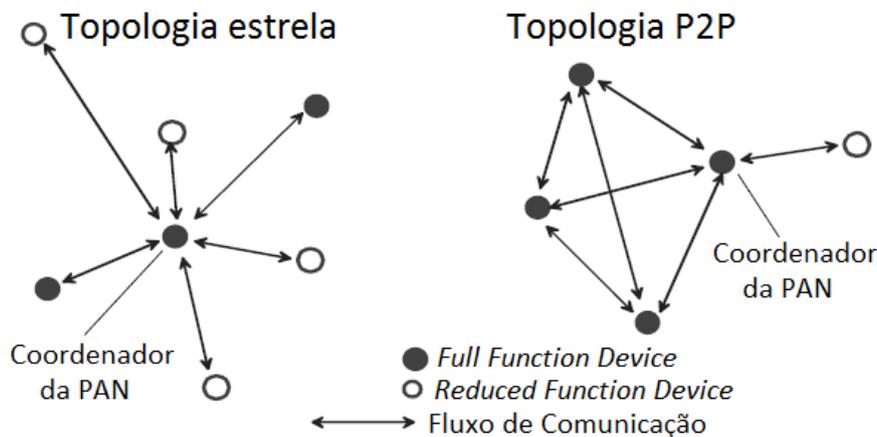


Figura 0.3. Exemplos de arquiteturas estrela e P2P [adaptado de IEEE 802.15.4]

Na arquitetura estrela, todo o fluxo de dados ocorre entre o coordenador da PAN e os dispositivos conectados a ele. Não existe comunicação entre os outros dispositivos. Os dispositivos periféricos podem ser do tipo RFD ou FFD, sendo o coordenador o único obrigado a ser do tipo FFD.

Na arquitetura P2P, os dispositivos são capazes de se comunicar entre si, e os dispositivos FFD podem utilizar a sua capacidade de roteamento. Pode-se, dessa maneira, formar redes mais complexas como Cluster Tree e Mesh. A única comunicação que não é possível é entre dois dispositivos RFD.

O módulo XBee 802.15.4 é um dispositivo para comunicação sem fio, desenvolvido pela Digi International, que utiliza o protocolo IEEE 802.15.4 [Digi International]. Este dispositivo é capaz de criar uma WPAN tanto no formato estrela quanto no formato P2P. Suas principais características são: baixo consumo, comunicação serial UART, configuração de gerência de energia, comunicação sem fio. Maiores detalhes sobre as especificações desse dispositivo podem ser encontrados na Tabela 4.1.

Tabela 0.1. Especificações XBee [Digi International].

Desempenho	
Alcance Urbano	30 metros
Alcance em Área Aberta	100 metros
Potência de Transmissão	1mW
Velocidade de Transmissão RF	250 000 bps
Velocidade de Transmissão Serial	1200 – 115 200 bps
Sensibilidade de Recepção	-92 dBm (1% taxa de erro de pacote)
Requisitos de Alimentação	
Tensão de Alimentação	2.8 – 3.4 V
Consumi Típico	50 mA
Gerais	
Frequência de Operação	ISM 2.4 Ghz
Dimensões	2.438 cm x 2.761 cm
Temperatura de Operação	-40 até 85° C
Rede e Segurança	
Topologia Suportada	P2P, estrela
Número de Canais	12 Canais
Opções de Endereçamento	PAN ID, Canal, Endereço
Criptografia	AES 128-bits

Desta maneira, o módulo XBee será capaz de receber dados enviados pela hardware ArduIMU através da sua porta de comunicação serial UART e retransmiti-los a um módulo conectado ao computador capaz de interpretar esses dados. Uma foto de um módulo XBee pode ser visto na Figura 4.2.



Figura 0.4. Módulo XBee [Digi International].

4. Modelo Virtual

A fidelidade do modelo virtual é definida pelo número de polígonos que o compõe, e quanto maior o número de polígonos, maior a necessidade por processamento. Como não é necessário para esse projeto um modelo extremamente fiel do corpo humano, deseja-se que o processamento do modelo possa ser realizado em tempo real para fornecer um *feedback* para o usuário e é necessário que tudo isso possa ser processado e executado no computador pessoal do usuário. Um modelo amigável e que permita identificar com detalhes o movimento que está sendo realizado no seu braço é o suficiente.

A utilização do modelo virtual é extremamente importante por dois motivos: fornecer o *feedback* para o usuário sobre o seu movimento e para permitir uma análise posterior detalhada, por um profissional da área da saúde, dos movimentos realizados.

Isso permitirá ao profissional da área da saúde diagnosticar o paciente, relevar as suas dificuldades e avanços e encaminhá-lo para o tratamento mais adequado.

Com a popularização de modelos virtuais, tanto de humanos, como de objetos e mesmo de criaturas surreais, foram criados softwares específicos para o seu desenvolvimento. Alguns desses softwares são de código aberto e baseados na biblioteca gráfica OpenGL. Dessa maneira, com poucos cliques e uma interface amigável, é possível criar um modelo virtual semelhante ao corpo humano.

4.1. Softwares MakeHuman e Blender

Criar um modelo virtual pode parecer uma tarefa muito complexa, porém, felizmente, existem ferramentas específicas para esse tipo de tarefa que facilitam este trabalho. Como se buscam sempre ferramentas de código livre, foram selecionados dois softwares capazes de modelar e animar um modelo humano virtual.

O MakeHuman é um software livre específico para modelar um corpo humano através de malha de polígonos. Com uma interface bastante simples pode-se, além de criar figuras, personalizar humanoides já existentes. As personalizações são das mais diversas, indo de gênero, peso, altura até posição das sobrancelhas. Além disso, este software permite exportar um arquivo que contém todas as informações sobre o modelo gerado, incluindo todas as articulações do corpo humano. Pelas políticas do software, este modelo pode ser explorado pelo usuário da maneira que ele desejar. Na Figura 3.2 vemos o modelo que foi criado no software MakeHuman e que será utilizado para as animações.

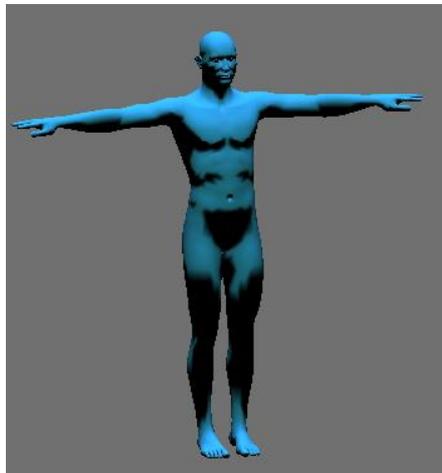


Figura 0.5. Exemplo de modelo criado no MakeHuman.

Para realizar animações sobre este modelo humano, existe o programa Blender. Um software livre capaz de importar o arquivo gerado pelo MakeHuman e modificar as posições das articulações do objeto em sequência para gerar animações. O Blender poderá então receber dados de um programa que analisará as variações nas leituras dos sensores e interpretará como alteração na posição das articulações. Dessa maneira, é possível gerar uma animação simples e de fácil leitura para o usuário.

5. Metodologia Experimental

Tendo em vista as necessidades levantadas por esse projeto, foram procuradas diversas soluções no mercado que pudessem ser úteis para o seu desenvolvimento. Vale lembrar

que é de grande importância que os softwares utilizados sejam de código livre para permitir maior flexibilidade ao projeto e manter o seu baixo custo.

Pode-se categorizar a metodologia experimental nas seguintes partes:

- aparato experimental: onde são listados e descritos as ferramentas e equipamentos necessários para a realização desse projeto;
- sensores inerciais: descrição detalhada dos sensores inerciais que serão utilizados, suas configuração e rotinas de medidas;
- transmissão de dados: modo para transmissão de dados com o mínimo de interferência no movimento do usuário, nesse caso utilizando o módulo XBee para transmissão de dados sem fio;
- modelo virtual: desenvolvimento de um modelo humano virtual através de softwares especializados;
- armazenagem de dados: banco de dados criado para armazenamento das informações para recriação do movimento.

5.1. Aparato Experimental

O sistema foi pensado para ser completamente modular, de modo que, se soluções mais baratas surgirem no mercado, poderão ser implementadas nesse projeto sem necessidade de grandes mudanças. Dessa maneira, esse projeto poderá sempre ser atualizado e se manterá competitivo. Os módulos existentes são divididos em “sensores inerciais”, “transmissão de dados”, “modelo virtual” e “banco de dados”, conforme esboçado na Figura 5.2.



Figura 0.6. Módulos do sistema.

O módulo dos sensores inerciais consiste em uma forma de medição da inclinação no espaço 3D dos sensores posicionados no corpo humano. Os sensores devem fornecer a inclinação nos 3 eixos (x, y e z) em relação a uma posição de repouso, a qual deve ser calibrada na inicialização do equipamento. Para realizar essas medições será utilizado um hardware desenvolvido pela empresa DIY Drones, chamado ArduIMU. Esse hardware foi desenvolvido para ser utilizado em VANTs (veículos aéreos não tripulados), mas pode ser utilizado para qualquer aplicação de monitoramento de movimento que necessite de sensores inerciais.

Para a transmissão de dados é necessário algo que interfira o mínimo possível na movimentação do usuário, portanto, foi decidido que a transmissão de dados até o computador será feita através de uma rede wireless. Existem aparelhos que utilizam protocolos wireless desenvolvido especialmente para aplicações que necessitam de baixo consumo e baixa transmissão de dados. O aparelho que será utilizado nesse

projeto será o módulo XBee, pois ele atende a todas as especificações e encontra-se disponível no laboratório.

Para a elaboração de um modelo virtual no computador, serão utilizados programas específicos para modelagem de um corpo humano e suas articulações. Foram escolhidos os programas MakeHuman e Blender pela experiência e sucesso que o laboratório possui em desenvolver modelos humanos através desses softwares.

5.1.1. ArduIMU

O hardware ArduIMU é capaz de realizar medições de angulação sobre os 3 eixos, x, y e z. Para isso, basta programá-lo com softwares de código livre disponíveis na internet no endereço code.google.com/p/ardu-imu/. Os dados de saída da porta serial são:

- ROLL: Medida em graus da rotação sobre o eixo X.
- PITCH: Medida em graus da rotação sobre o eixo Y.
- YAW: Medida em graus da rotação sobre o eixo Z negativo.

Podemos ver a representação dos eixos em relação a placa na Figura 5.3.

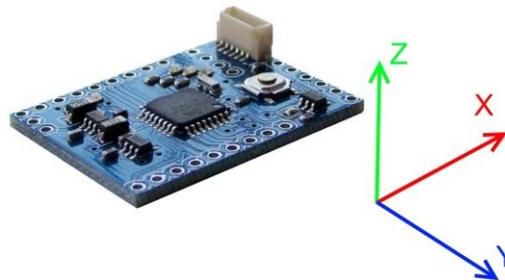


Figura 0.7 Eixos de rotação do ArduIMU [DIY Drones].

Existe também um programa de testes para o ArduIMU chamado “ArduIMU test” desenvolvido por Paul Mather e disponibilizado gratuitamente na internet [Manual ArduIMU]. Podemos ver a interface desse software na Figura 5.4.

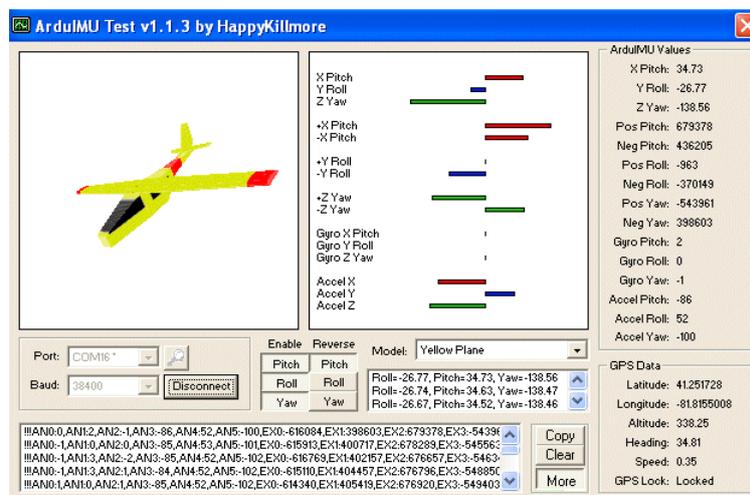


Figura 0.8 Software de teste para ArduIMU [Manual ArduIMU].

Através dessas ferramentas foi possível testar as funcionalidades do hardware. Os resultados obtidos não foram satisfatórios para a medição dos ângulos de um movimento delicado e preciso como são os movimentos de reabilitação motora. As oscilações nas medições foram excessivamente grandes, chegando na ordem de dezenas de graus.

Como as saídas analógicas não apresentavam uma oscilação muito significativa, o provável problema encontra-se no algoritmo do software para definir a angulação da placa. Esse problema deverá ser resolvido durante o desenvolvimento do TG2.

5.1.2. XBee

No caso desse projeto, deseja-se que todos os dados se concentrem no módulo conectado ao computador, os sensores não precisam se comunicar entre si e o consumo energético desses sensores deve ser o menor possível. Dessa maneira, a arquitetura estrela se mostra a mais vantajosa. Como o coordenador estará conectado ao computador e, portanto, alimentado por uma fonte de energia, não é grave se o consumo energético dele for elevado. Contudo, os sensores serão alimentados por bateria, o que torna vantajoso o fato de todos eles serem do tipo RFD [IEEE, 2006]. A Figura 5.5 esboça como será feita a comunicação entre os diferentes dispositivos do sistema.

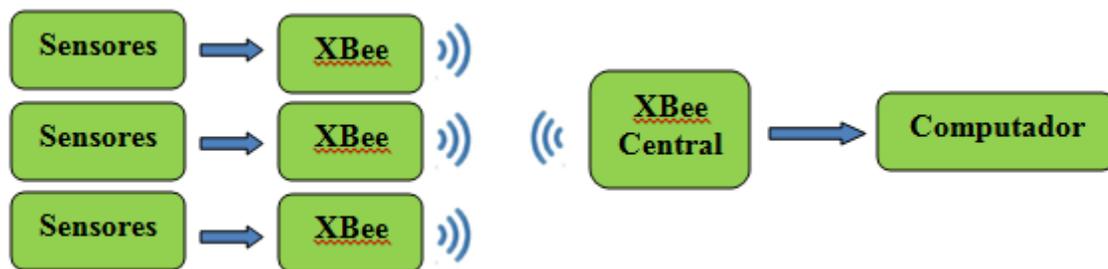


Figura 0.9. Sistema de transmissão de dados sem fio.

Para receber os dados seriais enviados pelo módulo XBee ao computador, foi desenvolvida uma aplicação na linguagem Python com o auxílio da biblioteca “xbee”. Esta biblioteca possui funções específicas para comunicação com aparelhos XBee como recebimento e envio de frames, o que facilita o desenvolvimento da aplicação.

Foram realizados testes com a rede configurada para funcionar em modo estrela, com dois módulos RFD conectados aos hardwares ArduIMU enviando dados ao módulo conectado ao computador. A comunicação foi configurada para ocorrer no modo API de modo que seja possível identificar o módulo que enviou o dado. O programa no computador é responsável por receber esses dados, separar a angulação em cada eixo e identificar o módulo responsável pelo envio dos dados. Nesse caso os endereços dos módulos conectados aos sensores são 0xDDD e 0xEEE, que correspondem ao sensor da mão e do braço, respectivamente. É possível ver na **Figura 0.10** os dois módulos se comunicando ao mesmo tempo com o XBee central e a identificação dos ângulos e do módulo responsável por enviar esses dados.

```

C:\Windows\system32\cmd.exe
RLL: -6.35 - YAW: -71.25 - PCH: 0.55 | Sender: dddd <MAO>
RLL: -3.73 - YAW: 79.27 - PCH: -8.19 | Sender: eeee <BRACO>
RLL: -6.21 - YAW: -71.22 - PCH: 0.88 | Sender: dddd <MAO>
RLL: -4.06 - YAW: 79.27 - PCH: -8.97 | Sender: eeee <BRACO>
RLL: -5.98 - YAW: -71.26 - PCH: 0.17 | Sender: dddd <MAO>
RLL: -4.08 - YAW: 79.28 - PCH: -8.12 | Sender: eeee <BRACO>
RLL: -5.68 - YAW: -71.18 - PCH: 0.68 | Sender: dddd <MAO>
RLL: -3.70 - YAW: 79.30 - PCH: -8.02 | Sender: eeee <BRACO>
RLL: -6.02 - YAW: -71.10 - PCH: 0.64 | Sender: dddd <MAO>
RLL: -4.24 - YAW: 79.32 - PCH: -8.52 | Sender: eeee <BRACO>
RLL: -5.61 - YAW: -71.07 - PCH: 0.63 | Sender: dddd <MAO>
RLL: -3.83 - YAW: 79.30 - PCH: -8.19 | Sender: eeee <BRACO>
RLL: -5.71 - YAW: -71.03 - PCH: 0.71 | Sender: dddd <MAO>
RLL: -4.39 - YAW: 79.35 - PCH: -8.80 | Sender: eeee <BRACO>

```

Figura 0.10. Leitura do envio de dados via rede XBee.

6. Etapas para Implementação Prática do Projeto

O projeto foi dividido em etapas para facilitar o monitoramento do seu avanço. Algumas etapas são independentes e podem ser realizadas em paralelo e outras necessitam da conclusão de determinada tarefa para serem iniciadas. Podemos destacar as seguintes atividades:

1. Revisão bibliográfica: Sem dependências
2. Funcionamento dos sensores: Sem dependências
3. Funcionamento do Módulo XBee: Sem dependências
4. Criação do Modelo Virtual: Sem dependências
5. Animação do Modelo Virtual: Dependente de 4
6. Animação do Modelo Virtual Através dos Dados dos Sensores: Dependente de 2, 3 e 5
7. Teste de Validação em Humanos: Dependente de 6
8. Análise de Resultados: Dependente de 7

7. Cronograma

A Tabela 0.2 mostra como foi organizado o cronograma para a realização do projeto.

Tabela 0.2. Cronograma de atividades do TG2

	Março	Abril	Maio	Junho	Julho
1					
2					
3					
4					
5					
6					
7					

8											
	TG1				TG2						

8. Conclusão

A medição de movimentos do corpo humano é um problema bastante abordado na área de tratamento de pacientes em recuperação. Felizmente, muitas tecnologias de outras áreas podem ser exportadas para esse intuito, como a medição da orientação de veículos aéreos. Utilizando o hardware ArduIMU, a medição da inclinação dos membros humanos se torna bastante simples. Além disso, com a utilização dos módulos XBee para transmissão de dados, rapidamente foi possível criar uma WPAN e enviar os dados do ArduIMU para o computador central.

Os dados enviados pelos sensores ainda se mostram imprecisos, o que precisará ser tratado para a segunda parte do projeto. Contudo, o avanço do projeto se mostra em acordo com o cronograma previsto para o seu desenvolvimento.

9. Referências

- Zheng, H. , Black, N., Harris, N. (2005). “Position-sensing technologies for movement analysis in stroke rehabilitation”.
- Hamill J, Knutzen K. (2003). Biomechanical basis of human movement. 2nd ed. Baltimore: Lippincott Williams & Wilkins.
- Aggarwal, J.K. and Cai, Q. (1999), “Human motion analysis: a review”,
- Welch, G. and Foxlin, E. (2002), “Motion tracking: no silver bullet, but a respectable arsenal”, IEEE Computer Graphics and Applications, Vol. 22 No. 6, pp. 24-38.
- S. Bhatia, L. Sigal, M. Isard, M. Black, 3D human limb detection using space carving and multi-view eigen models, in: Proceedings of the IEEE Workshop on Articulated and Nonrigid Motion. Washington, DC, 2004.
- Emil J., Aleksandar M. et al. (2005), “A wireless body area network of intelligent motion sensors for computer assisted physical rehabilitation”. Journal of neuroengineering & rehabilitation [1743-0003] volume 2 página: 6
- Godfrey, A., Conway, R. et al. (2008) ,“Direct measurement of human movement by accelerometry”. Medical Engineering & Physics 30 páginas: 1364–1386.
- Zhou, H. e Hu, H. (2007). “Inertial sensors for motion detection of human upper limbs”. Sensor Review. Volume 27, número 2, páginas: 151–158
- Anderson, Chris (2012), em <https://code.google.com/p/ardu-imu/wiki/IntroductionPage>, acessado em 25/04/2013.

Barbour, N. and Schmidt, G. (2001). “Inertial Sensor Technology Trends”. IEEE SENSORS JOURNAL, VOL. 1, NO. 4, DECEMBER 2001

Digi International. Endereço web <http://www.digi.com> acessado em 07/05/2013.

DIY Drones. Endereço web <http://www.diydrones.com> acessado em 07/05/2013.

Manual ArduIMU. Endereço web <https://code.google.com/p/ardu-imu/> acessado em 07/05/2013.

IEEE Standard for Information technology— Telecommunications and information exchange between systems— Local and metropolitan area networks— Specific requirements— Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)