

Universidades Lusíada

Oliveira, Rui Pedro Alves de

Matraquilhos inclusivos : controlo inteligente de eixos

<http://hdl.handle.net/11067/7039>

Metadata

Issue Date 2023

Abstract Nos dias de hoje, apesar de haver uma crescente consciencialização para uma melhor adaptação dos meios físicos de modo a permitir a interação de indivíduos com necessidades Específicas, existe ainda uma discrepância considerável na quantidade e qualidade de tarefas ou eventos em que podem participar. Uma atividade aparentemente simples como um jogo de matraquilhos fica vedada a indivíduos com Necessidades Específicas. Contudo, estas dificuldades podem ser atenuadas com a ajuda de meios eletrónic...

Nowadays, although there is a growing awareness for a better adaptation of physical environments to allow the interaction of individuals with Special Needs, there is still a considerable discrepancy in the quantity and quality of tasks or events in which they can participate. A seemingly simple activity like a game of foosball is closed to individuals with Special Needs. However, these difficulties can be alleviated with the help of electronic and computer aids. To help these people in a therap...

Keywords Automação, Base de dados

Type masterThesis

Peer Reviewed no

Collections [ULF-FET] Dissertações

This page was automatically generated in 2023-10-21T13:36:40Z with information provided by the Repository



UNIVERSIDADE LUSÍADA
VILA NOVA DE FAMALICÃO

Matraquilhos Inclusivos – Controlo Inteligente de Eixos

Rui Pedro Alves de Oliveira

Dissertação para obtenção do Grau de Mestre em Engenharia Electrónica e
Informática

Vila Nova de Famalicão
abril 2023



UNIVERSIDADE LUSÍADA
VILA NOVA DE FAMALICÃO

Matraquilhos Inclusivos – Controlo Inteligente de Eixos

Rui Pedro Alves de Oliveira

Orientador: Professor Doutor Pedro Reis
Orientador: Professor Doutor Rui Silva

Dissertação para obtenção do Grau de Mestre em Engenharia Electrónica e
Informática

Agradecimentos

O realizar deste projeto de investigação foi um processo longo, demorado e desgastante, cujas adversidades só foram ultrapassadas com o apoio de pessoas presentes no meu percurso académico. Estas que merecem todo e qualquer reconhecimento pelos apoios técnicos, psicológicos e financeiros.

Ao meu orientador, Prof. Doutor Pedro Reis Gomes, agradeço por todo o percurso académico, por todas as batalhas ultrapassadas no decorrer desta dissertação e pelo tempo dedicado.

Ao Prof. Doutor Rui Silva, pelos aconselhamentos que deu ao longo do projeto.

Ao Nuno Costa, cuja ajuda foi essencial.

Ao Prof. António Nicolau, pela contribuição e ensinamentos.

Às terapeutas da Associação Portuguesa de Pais e Amigos do Cidadão Deficiente Mental pelo tempo e dedicação disponibilizados.

Aos funcionários da instituição, em especial à D.Hermínia, D.Ana, Sr. Manuel, Sr. Aurélio, Doutora Paula Silva Sá e Dra. Cristina Sampaio por todo o carinho.

À psicóloga Dra Carla Oliveira, que me ajudou numa etapa difícil.

Um abraço especial aos amigos e colegas que tenho orgulho em dizer que tenho, especialmente ao José Costa e ao Filipe Santos, com os quais partilhei toda a minha vida universitária, ao Eduardo Santos, pelos recentes anos de mestrado, sem esquecer também o Ricardo Gomes e o Daniel Fernandes, que apesar de seguirem os seus rumos, sempre fizeram questão de apoiar em todos os momentos.

À minha família, cujo apoio e confiança foi absolutamente indispensável que me deixa emocionado e orgulhoso.

Por fim, à Universidade Lusíada, um grande obrigado!

Resumo

Nos dias de hoje, apesar de haver uma crescente consciencialização para uma melhor adaptação dos meios físicos de modo a permitir a interação de indivíduos com Necessidades Específicas, existe ainda uma discrepância considerável na quantidade e qualidade de tarefas ou eventos em que podem participar. Uma atividade aparentemente simples como um jogo de matraquilhos fica vedada a indivíduos com Necessidades Específicas. Contudo, estas dificuldades podem ser atenuadas com a ajuda de meios eletrónicos e informáticos.

Para ajudar estas pessoas em contexto terapêutico, o ponto central da dissertação é uma Mesa de Matraquilhos robotizada já presente na Universidade, na qual se definiu que o movimento autónomo dos eixos da mesa com a possibilidade de recolha dos dados para futura análise é o objetivo do projeto. Pretende-se que a base de dados seja boa o suficiente para que os seus dados sirvam para futuras análises de, por exemplo, terapias.

Foram estudados os componentes da Mesa de Matraquilhos e projetaram-se novos comandos do tipo ‘joystick’, para permitir um maior controlo dos eixos por parte dos jogadores. A Mesa de Matraquilhos sofreu uma atualização para permitir que os eixos fossem autónomos. Os *joysticks* criados mereceram a aprovação de terapeutas. Utilizou-se um sistema de aquisição de dados que permitiu recolher os dados dos sensores da mesa. A estes dados aplicou-se um método de Inteligência Artificial (*Naïve Bayes* gaussiano). O campo de jogo foi dividido em áreas por onde a bola se move e passou-se essa representação para as memórias dos autómatos. Depois, colocaram-se os eixos a mover autonomamente. Os dados recolhidos da mesa ficam associados a um jogador. Caso se pretenda, os dados podem ser acrescentados ao ficheiro para se aplicar novamente o método de Inteligência Artificial, tornando mais robustas as decisões tomadas.

O método de Inteligência Artificial acabou por prever 99,5% dos movimentos dos eixos e 96,7% dos estados de remate, culminando num acerto total de 96,2%, sendo considerado este um valor aceitável de acertos. O método de captação dos dados consegue exibir uma taxa de atualização de 50 ms sendo que o único fator negativo é o tamanho das áreas do campo para detetar a bola.

Os resultados alcançados foram positivos e mostram que, enquanto prova de conceito, é

possível utilizar uma Mesa de Matraquilhos robotizada para recolher informação sobre mobilidade e capacidade cognitiva, motivando a participação dos intervenientes utilizando aspetos lúdicos e inclusivos.

Palavras-chave: Automação, Base de Dados, Aprendizagem Automática, Matraquilhos Robotizados

Abstract

Nowadays, although there is a growing awareness for a better adaptation of physical environments to allow the interaction of individuals with Special Needs, there is still a considerable discrepancy in the quantity and quality of tasks or events in which they can participate. A seemingly simple activity like a game of foosball is closed to individuals with Special Needs. However, these difficulties can be alleviated with the help of electronic and computer aids.

To help these people in a therapeutic context, the central point of the dissertation is a robotic table soccer table already present at the University, in which it was defined that the autonomous movement of the axes of the table with the possibility of data collection for future analysis is the goal of the project. It is intended that the database be good enough for its data to be used for future analysis of, for example, therapies.

The components of the Foosball Table were studied and new joystick type controls were designed to allow for greater control of the axes by the players. The table soccer table was upgraded to allow the axes to be autonomous. The joysticks created met with the approval of therapists. A data acquisition system was used to collect data from the table's sensors. An Artificial Intelligence method (Gaussian Naïve Bayes) was applied to this data. The playing field was divided into areas where the ball moves and this representation was passed to the automata's memories. Then, the axes were placed to move autonomously. The data collected from the table is associated with a player. If desired, the data can be added to the file to apply the Artificial Intelligence method again, making the decisions made more robust.

The Artificial Intelligence method ended up predicting 99.5% of the axis movements and 96.7% of the shot states, culminating in a total accuracy of 96.2%, being considered an acceptable value of accuracy. The data capture method is able to display a 50 ms update rate, and the only negative factor is the size of the field areas to detect the ball.

The results achieved were positive and show that, as a proof of concept, it is possible to use a robotic foosball table to collect information on mobility and cognitive ability, motivating the participation of the participants using playful and inclusive aspects.

Índice

1.	Introdução	1
1.1	Enquadramento	2
1.2	Objetivos	4
1.3	Método experimental	4
1.4	Estrutura	5
2.	Revisão de literatura	7
2.1	Equipamentos lúdicos para apoio a pessoas com necessidades específicas	7
2.2	Técnicas de inteligência artificial	8
2.3	Tipos de mesas de matraquilhos	11
2.4	Controlo de eixos em mesas de matraquilhos	12
3.	Monitorização das ações de pessoas com necessidades específicas sobre uma mesa de matraquilhos	15
3.1	Adaptação da mesa de matraquilhos robotizada	15
3.2	Sistemas para controlo de processos em robótica	19
3.3	Desenvolvimento de um interface pessoa-máquina adaptado	21
3.4	Desenvolvimento de programa de teste aos tempos de reação	23
3.5	Técnica de inteligência artificial	27
3.6	Desenvolvimento de um <i>interface</i> de comunicação	30
3.7	Procedimento experimental para validação dos dados	34
3.8	Desenvolvimento da base de dados em contexto de prova de conceito	39
4.	Discussão dos resultados	43
5.	Conclusão e trabalho futuro	47
	Referências	51
	Anexos	59

Índice de Figuras

Figura 1 - Exemplos de configuração dos pés dos jogadores em MM clássicos [60], [61]	12
Figura 2 - Exemplo de MM robotizados [62]	12
Figura 3 - Exemplo de MM digitais [63]	12
Figura 4 - Exemplo de um fluxo de IA, aplicado a uma MM robotizada	13
Figura 5 - Exemplo de um modelo gerado por base em imagem [12].	13
Figura 6 - Fluxo da MM desenvolvida na Universidade (Adaptado de [68])	16
Figura 7 - MM desenvolvida na Universidade	16
Figura 8 - Emissores de laser	17
Figura 9 - Recetores de laser	17
Figura 10 - Atuador que permite a rotação do eixo para efetuar o remate	17
Figura 11 - Atuador que permite o movimento do eixo para efetuar o movimento dos jogadores	18
Figura 12 - Rampa de lançamento da bola	18
Figura 13 - Aquisição de dados do mundo real	19
Figura 14 - Sistema de aquisição de dados	20
Figura 15 - Recorte do joystick – vista lateral	22
Figura 16 - Recorte do topo do joystick – vista frontal	23
Figura 17 - Definição da estrutura de sensores do campo	24
Figura 18 - Sensores de posição da bola presentes na MM e respetiva seleção	26
Figura 19 - Nomenclatura de identificação de áreas	26
Figura 20 - Diagrama simplificado do código para escolha do método	27
Figura 21 - Pseudocódigo de IA (Naïve-Bayes)	28
Figura 22 - Memórias com informações das áreas e variável de controlo	30
Figura 23 - Área A1	31
Figura 24 - Área A2	32
Figura 25 - Área A12	33
Figura 26 - Área A16	34
Figura 27 - Apontadores de memória das áreas de memória do CJ2M	35
Figura 28 - Inserção do Project File para com o CX-Server	36
Figura 29 - Variáveis e seus respetivos valores no OPC Expert	36
Figura 30 - Escrita manual dos valores	36
Figura 31 - Equações com a função RTD fomecidas pelo programa	37
Figura 32 - Ponte entre D100 e CIO1	37
Figura 33 - MOV do D100 para a área onde se encontram saídas dos movimentos dos eixos	37
Figura 34 - Dados crus provenientes das memórias dos sensores	38
Figura 35 - Dados do ficheiro gerado pela IA correspondentes aos dados retirados da MM	38
Figura 36 - Resultado do atuador CIO1, consoante os dados dos sensores	39
Figura 37 - Tabelas da base de dados	40
Figura 38 - Registo de um utilizador, de momento apenas pelo nome	40
Figura 39 - Dados provenientes da MM	41
Figura 40 - Interface com dados de um utilizador para estudo futuro	42

Índice de Tabelas

Tabela 1 - Numerical Rating Scale for the International Classification of Functioning, Disability, and Health [21]	3
Tabela 2 - Possíveis valores para o CIO1	39
Tabela 3 - Acertos das previsões do método	44

Índice de Equações

Equação 1 - Regra de Bayes [55]	10
Equação 2 - Naïve Bayes gaussiano [57]	10

Lista de Abreviaturas

APPACDM	–	Associação Portuguesa de Pais e Amigos do Cidadão Deficiente Mental
CIO	–	Áreas de memória usadas para <i>input/output</i> de <i>bits</i>
CJ2M	–	Da família CJ2, um controlador modular de alto desempenho da OMRON
IA	–	Inteligência Artificial
KNN	–	<i>K-nearest neighbours</i>
ML	–	<i>Machine Learning</i>
MM	–	Mesa de matraquilhos
MySQL	–	<i>My Structured Query Language</i>
NB	–	<i>Naïve Bayes</i>
NE	–	Necessidades Específicas
OMS	–	Organização Mundial da Saúde
PLC	–	<i>Programmable Logic Controller</i>
RTD	–	<i>Real-Time Data</i>
SVM	–	<i>Support Vector Machine</i>

Lista de Símbolos

μs	–	microsegundo (10^{-6} segundos)
mm	–	milímetro (0,001 metros)
ms	–	Milissegundo (0,001 segundos)
N	–	<i>Newton</i>
ns	–	nanossegundo (10^{-9} segundos)
s	–	segundo
v	–	<i>Volt</i>

1. Introdução

O desenvolvimento tecnológico não revolucionou apenas o mercado da eletrônica de consumo, mas também vários equipamentos destinados a pessoas com Necessidades Específicas (NE). Estas pessoas podem apresentar diversas deficiências a nível físico, emocional, comportamental e de aprendizagem[1]. Existem no mercado diferentes produtos para permitir às crianças uma aprendizagem num ambiente lúdico, desde *software*, como o *ABC Sebran* [2], a artigos de robótica, como o caso dos exosqueletos [3], e estruturas estáticas de apoio postural como o *Standing Frame* [4].

O uso das tecnologias pelas pessoas com NE traz inúmeras vantagens [5]–[8], permitindo que estas cuidem de si e dos seus, trabalhem, consigam estar em ambientes de aprendizagem, saibam aceder a informação, fazendo que, no fim, desfrutem da vida e participem ativamente na vida comunitária [9], [10].

A tecnologia evolui para tornar a vida das pessoas mais fácil e agradável. Para as pessoas com necessidades específicas, estes avanços tecnológicos tornaram possíveis muitas das tarefas que outrora eram impossíveis, transformando a incapacidade em ação. Atualmente, existem diversas empresas especializadas em adaptação e fabrico de equipamentos inovadores. Como por exemplo, a *ReWalk Robotics* desenvolveu um exosqueleto cujo movimento é controlado por mudanças subtis no seu centro de gravidade. Uma inclinação para a frente da parte superior do corpo é detetada pelo sistema, que inicia a primeira etapa. O deslocamento repetido do corpo gera uma sequência de passos que imita uma marcha natural [11].

Devido ao crescente aumento do uso da tecnologia no nosso dia-a-dia, considera-se fundamental estar atento às novas propostas e como estas se relacionam ao nosso estilo de vida. Só desta forma se extrai o que de melhor a tecnologia oferece, visando o aperfeiçoamento da saúde física e mental. As inovações tecnológicas podem ser utilizadas com diferentes propósitos, alguns relacionados com o ambiente de trabalho, à melhoria da produtividade, outros relacionados com a vida social e aos cuidados com a saúde.

Vários sistemas de Mesas de Matraquilhos (MM) foram desenvolvidos até aos dias de hoje. Estes apresentam inovações baseadas no avanço tecnológico, aplicando formas de Inteligência Artificial (IA) para controlo dos eixos, mas que apenas se baseiam no uso

de câmaras e nunca em sistemas para melhorar na forma de interação entre a pessoa e a MM [12]–[17]. Assim, pretende-se desenvolver outro método não baseado em câmaras, evitando os custos do processamento de imagem. Pretende-se também que os dados da MM possam ser exportados, para permitir uma posterior análise, bem como melhorar a forma de interação com a MM, permitindo que pessoas com NE possam jogar.

Resumidamente, nas MM desenvolvidas até aos dias de hoje, o foco é colocado no desempenho da IA no sistema [15], apresentando muitas limitações nas vertentes relacionadas com os aspetos extra jogo, tais como a reabilitação [18], sendo esta um dos focos da dissertação. Os sistemas com o foco exclusivamente na qualidade da IA [13], [17], permitem jogar de forma pouco natural, algo que se quer evitar com esta dissertação.

A motivação desta dissertação, para se obter o movimento independente dos eixos, com ajuda de sensores, prende-se também ao facto de já terem sido realizados testes a pessoas com NE numa MM robotizada que deram resultados suficientes para um artigo científico [19]. A MM ainda necessitava de ser adaptada para melhorar a recolha dos dados e manipulação das memórias que foram realizadas manualmente.

1.1 Enquadramento

Um grupo de pessoas que está marginalizado é o das pessoas portadoras de alguma deficiência motora ou intelectual. Atualmente, assiste-se a uma sociedade preocupada em criar oportunidades e direitos iguais para todos os cidadãos, conforme estabelecido na Declaração de Salamanca da UNESCO de 1994, através da implementação de diferentes políticas em diferentes áreas da sociedade. No entanto, vários estudos continuam a evidenciar algumas lacunas nesta igualdade de oportunidades, sobretudo em pessoas com NE [20].

Existe um parâmetro de avaliação, chamado de *Numerical Rating Scale for the International Classification of Functioning, Disability, and Health* [21]. Este divide a dependência de uma pessoa em dez níveis (Tabela 1).

Tabela 1 - Numerical Rating Scale for the International Classification of Functioning, Disability, and Health [21]

Nível de Deficiência	Numerical Rating Scale
Sem problemas (independência completa)	0
Problema leve (supervisão)	1
	2
	3
Problema moderado (reduzida assistência)	4
	5
	6
Problema grave (elevada assistência)	7
	8
	9
Problema completo (dependência completa)	10

Nas pessoas com NE, o nível de envolvimento em atividades/ocupações nem sempre pressupõe uma participação ativa das mesmas em termos das escolhas das atividades que pretendem realizar ou do nível do seu próprio envolvimento nas mesmas. Verificam-se vários sinais de injustiça, com reflexos na qualidade de vida das pessoas, devendo esse fator ser mitigado através da implementação de melhores oportunidades de participação nas diferentes atividades e nos diferentes contextos da vida quotidiana [22].

Associações como a Associação Portuguesa de Pais e Amigos do Cidadão Deficiente Mental (APPACDM), têm o propósito de procurar soluções para desafios e dificuldades, visando a autonomia, a promoção da plena participação em todos os contextos de vida, no respeito pelos princípios da individualização, promoção do acesso a serviços, promoção da igualdade de oportunidades, educação, trabalho e formação ao longo da vida das pessoas com Deficiência Intelectual e Incapacidade [23].

1.2 Objetivos

A tese tem como objetivo principal o desenvolvimento de um sistema baseado numa MM robotizada, permitindo a sua utilização por pessoas com NE, recolhendo dados resultantes da interação entre as pessoas e a máquina. O sistema desenvolvido foi projetado para permitir pessoas cujo valor na escala da *Numerical Rating Scale* se situe entre os 0 e 6 (sem problemas a problemas moderados) ao promover a realização de exercício físico e mental.

O objetivo da dissertação é promover o desenvolvimento de tecnologias que potenciam uma interação entre dispositivos lúdicos e os seus utilizadores com recurso à IA. Pretende-se contribuir para o desenvolvimento de uma MM que permita a sua utilização por um maior número de pessoas.

O objetivo específico deste trabalho consiste na recolha de dados da posição da bola e do estado dos eixos e determinar através de um algoritmo com base em IA a posição futura dos eixos e estado de remate. Para recolher dados que alimentem o algoritmo, pretende-se desenvolver uma interface de comunicação entre a MM e a máquina que executa o algoritmo. Este mecanismo vai permitir fazer a aquisição e controlo dos eixos. Por fim, pretende-se desenvolver uma base de dados que permita armazenar dados resultantes de um jogo, para que posteriormente possam servir de ajuda no apoio ao diagnóstico para pessoas com NE, analisando por exemplo os tempos de reação.

1.3 Método experimental

Desenvolver um produto requer um processo cuidadoso e bem estruturado para garantir que as necessidades do cliente sejam atendidas. Em geral e, também, no caso de uma MM robotizada, o processo pode ser dividido em seis etapas [24]:

- Levantamento e análise de requisitos
- Idealização de protótipo
- Criação do protótipo
- Avaliação inicial pelos utilizadores

- Refinamento do protótipo
- Implementação do produto.

A etapa de levantamento e análise de requisitos é fundamental para entender as necessidades do utilizador e estabelecer metas claras para o projeto. Nesta fase, são realizadas pesquisas de para identificar o que já existe no mercado e, também, em ambiente universitário. Para complementar a pesquisa, procura-se saber uma opinião de possíveis futuros utilizadores para coletar informações sobre as suas necessidades e expectativas. Com base nas pesquisas e informações, são definidos os requisitos da MM. Na etapa de idealização de um protótipo, os requisitos são transformados num conceito de produto concreto. No caso da MM, foram definidos todos os componentes e alterações necessárias para satisfazer os requisitos dos utilizadores. É importante que a MM seja coerente com os requisitos para que seja viável do ponto de vista funcional. No terceiro ponto, criação do protótipo, são executadas as idealizações feitas anteriormente. O objetivo desta etapa é produzir um modelo funcional que possa ser usado em testes de funcionalidade em condições reais, sendo esta a quarta etapa. O utilizador faz uma avaliação inicial resultante dos testes à MM para ver se corresponde às suas necessidades e expectativas, para, em seguimento, fornecer *feedback*. Os *feedbacks* são recolhidos e analisados para que possam ser incorporados numa nova evolução da MM. Essa evolução corresponde à quinta etapa, que é o refinamento do protótipo. Finalmente, na sexta etapa, o produto final é implementado para o mundo real, fora de ambiente de teste. No caso da MM, o seu uso é disponibilizado a quem queira jogar ou queira realizar o apoio a diagnósticos de carácter terapêutico.

Resumidamente, o processo de desenvolvimento de produto para uma MM robotizada envolve uma série de etapas cuidadosamente planeadas e executadas. Cada etapa é crítica para o sucesso do projeto e requer um trabalho metódico para garantir que o produto final atenda às necessidades dos utilizadores finais.

1.4 Estrutura

Esta tese tem 5 capítulos organizados da seguinte forma. Após o presente, no segundo capítulo é feita a revisão de literatura, onde é realizado um estudo mais aprofundado sobre tecnologias existentes para ajuda a pessoas com Necessidades Específicas, MM e IA. O terceiro capítulo apresenta o desenvolvimento do trabalho realizado, onde o foco do

estudo é a MM, sistema de interação das pessoas e o sistema de aquisição de dados. No quarto capítulo são apresentados os resultados e é efetuada a análise dos mesmos. Finalmente, o quinto capítulo apresenta a conclusão e propostas de trabalho futuro.

2. Revisão de literatura

Os matraquilhos são um jogo de mesa, onde se simula um jogo de futebol. É jogado por duas equipas de dois jogadores, sendo muito comum em escolas, cafés com grande fluxo de clientes e até em eventos que tenham as típicas ‘zonas de lazer’, sendo por isso um elemento de convívio e de destreza física que apela ao desenvolvimento mental dos seus participantes. Apesar de jogos estarem conotados negativamente [25], está provado que o impacto destes pode ter muitos aspetos, tais como melhorar da saúde mental ao nível da reabilitação, psiquiatria, entretenimento, educação e comunicação [26], assim como reduzir os sintomas relacionados com ansiedade e depressão [27]. A realização de atividades com MM provaram influenciar positivamente pessoas com esquizofrenia [28].

Neste capítulo são apresentados alguns sistemas existentes de controlo adaptados para pessoas com NE, diferentes tipos de MM, e como é feito o controlo dos eixos e MM robotizados com base em técnicas de IA.

2.1 Equipamentos lúdicos para apoio a pessoas com necessidades específicas

Já têm sido desenvolvidos sistemas de controlo adaptados para pessoas com NE que exploram técnicas para tornar mais fácil o ato de pressionar os botões e os *joysticks*. Um exemplo é o *Microsoft's Xbox Adaptive Controller* [29], que apresenta dois botões de grandes dimensões, programáveis, de modo que o utilizador apenas apoie a mão sobre o mesmo para o pressionar. Este tipo de comandos potencia melhorias na capacidade das pessoas interagirem com o jogo, e, conseqüentemente, faz com que estas tenham uma melhor qualidade de vida [30]. Os comandos sem fios da *Wii* melhoraram a inclusão de pessoas com NE, nivelando a sua performance ao nível de indivíduos sem qualquer incapacidade [31].

Um dos mais usados e mais eficazes equipamentos lúdicos para apoio a pessoas com NE é o *joystick* adaptado [32]. Os *joysticks* adaptados para pessoas com NE são dispositivos tecnológicos que permitem a essas pessoas terem uma maior autonomia na realização de tarefas diárias. Estes dispositivos possuem diversas vantagens, que podem melhorar significativamente a qualidade de vida das pessoas com deficiência. Por exemplo, uma

pessoa com deficiência motora pode usar um *joystick* adaptado para controlar uma cadeira de rodas elétrica e se movimentar livremente [32], permitindo melhorar significativamente a independência e autoestima [33].

Uma vantagem dos *joysticks* adaptados é que eles podem ser personalizados para atender às necessidades individuais de cada pessoa com deficiência. Uma pessoa com deficiência motora pode precisar de um *joystick* adaptado que tenha botões maiores ou mais sensíveis, para que ela possa usá-lo com mais facilidade [34]. Também, o *joystick* pode ser programado para se adaptar às habilidades motoras específicas da pessoa [35], o que pode melhorar a precisão e eficácia no controle [36].

2.2 Técnicas de inteligência artificial

IA permite agrupar um grande volume de dados gerados por um sistema para que este possa aprender e fazer previsões para casos futuros, encontrar padrões ou classificar dados. O algoritmo a ser usado numa situação específica determina o tipo de IA, cujo processo varia. Existem quatro tipos de IA [37]: supervisionada, semi-supervisionada, não-supervisionada e aprendizagem por via de reforço.

Na IA supervisionada, o sistema está sempre a ser observado enquanto aprende, ou seja, são disponibilizadas ao algoritmo todas as informações necessárias para a aprendizagem, pares *input/output*, para que com novos *inputs* o método consiga adivinhar o *output* [37]. A IA supervisionada pode produzir previsões altamente precisas, especialmente quando treinada com grandes quantidades de dados [38]. Como o modelo requer uma etapa inicial de treino, pode processar grandes quantidades de dados rapidamente e de forma eficiente [39]. Este tipo de IA pode sofrer alterações para atender às necessidades de um utilizador, o que pode resultar em previsões mais precisas [40] e pode ser atualizado e melhorado continuamente à medida que novos dados são fornecidos [41].

Porém, também apresenta desvantagens. A IA supervisionada requer grandes quantidades de dados rotulados para ser treinada, cuja recolha dos mesmos pode ser muito demorada [42]. O modelo pode se tornar ineficaz se a amostra de dados para treino não for representativa da população de dados [43]. Também, treinar modelos de IA

supervisionada pode exigir muitos recursos computacionais, que pode fazer com que o tempo de processamento seja considerável [44].

Considerando as vantagens e desvantagens apresentadas, este tipo de IA é adequado para o uso na MM. Alguns dos algoritmos supervisionados mais conhecidos são algoritmos de classificação ou classificadores probabilísticos, como *Naïve Bayes* (NB), *K-nearest neighbors* (KNN), *Support vector machine* (SVM) e árvores de decisão [37].

O NB é baseado no teorema de *Bayes*, onde se supõe a independência entre cada ocorrência ou conjunto de dados [45]. A principal vantagem é que, em comparação com outros métodos, necessita apenas de uma pequena quantidade de dados para treino para identificar quais os parâmetros necessários para a obtenção da classificação final [46]. Porém, o desempenho pode ser afetado devido à própria suposição da independência entre cada ocorrência. Pode ser usado em situações como a classificação de texto previamente categorizado ou filtragem de e-mail considerado *spam*, ou até para identificar ruído num conjunto de dados para o poder descartar no modelo de previsão [47]. Em relação com diversas técnicas de ML, o classificador *Naïve Bayes* tem desempenhado um papel de destaque devido à sua simplicidade e eficiência [48].

O KNN é um algoritmo não-generalizante que não cria um modelo base, porém coloca as ocorrências observadas em treino em pontos num espaço de n dimensões, agrupando os dados com base na distância entre os pontos [46]. A precisão depende da qualidade dos dados. O maior problema com o KNN é escolher o número ideal de pontos a serem considerados para melhor categorizar a vizinhança dos mesmos [37]. Pode ser usado em aplicações como reconhecimento facial, sistemas de recomendação de produtos e vídeos, assim como reconhecimento de padrões em sistemas financeiros e médicos [49].

O SVM, num espaço de dados com muitas dimensões, agrupa os dados em divisões de espaços independentes. O objetivo do SVM é encontrar uma divisão de espaço independente boa [50], quanto a alguma medida que avalie o seu desempenho, num espaço de características de grandes dimensões. É eficaz em espaços de grandes dimensões e pode se comportar de maneira diferente com base em diferentes funções matemáticas [37]. Porém, quando o conjunto de dados contém mais ruído, o SVM não funciona tão bem. Entre muitas aplicações, está fortemente ligado à bioinformática [51].

As árvores de decisão são eficazes no domínio das aplicações, como na análise de

comportamento de utilizadores e análise de segurança cibernética [37]. Depois de organizar a árvore e identificar caminhos e nós desde a raiz até à folha, o método classifica as instâncias, verificando o atributo definido pelo nó, começando no nó raiz da árvore e, em seguida, subindo pela árvore no caminho correspondente ao que o valor do atributo possa indicar [37].

Uma comparação entre os classificadores [52], mostrou que o NB, além de ser o mais utilizado, revela-se o melhor classificador quando comparado a árvores de decisão, KNN e SVM, utilizando como métrica as percentagens de acerto.

Classificadores como o NB, são derivados de modelos generativos de probabilidade que utilizam um modelo estatístico de distribuição de probabilidade conjunta $P(X, Y=y)$ dada a variável observável X e a variável alvo Y [53]. Para estes casos, a regra de *Bayes* define a Equação 1, que serve para estimar a probabilidade condicional y , sendo feitas suposições ao modelo para decompor a probabilidade num produto de probabilidades condicionais. A regra de *Bayes* determina a probabilidade de um acontecimento, com base em condicionantes já conhecidas que estão relacionadas com o mesmo, servindo de evidência [54].

Equação 1 - Regra de Bayes [55]

$$Pr(y|x) = Pr(y|x^1, x^2, \dots, x^n) = \prod_{i=1}^n Pr(x^i|x^1, \dots, x^{i-1}, y) \frac{Pr(y)}{Pr(x)} = \prod_{j=1}^{n'} Pr(y^j|y) \frac{Pr(y)}{Pr(x)}$$

Sendo uma simplificação da regra de *Bayes*, o NB Gaussiano (Equação 2) é uma técnica de classificação probabilística baseada na suposição de que a distribuição de probabilidade condicional de cada recurso é gaussiana. Apesar de ser uma simplificação, muitas vezes apresenta-se suficiente para analisar dados do mundo real [56].

Equação 2 - Naïve Bayes gaussiano [57]

$$P(X|Y = c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{\frac{-(x-\mu_c)^2}{2\sigma_c^2}}$$

O algoritmo NB Gaussiano é utilizado para realizar classificação em problemas onde cada instância é descrita por um conjunto de atributos *features* contínuos. O objetivo é determinar a classe mais provável de cada instância com base nos valores desses atributos [58].

Para o NB Gaussiano, as probabilidades condicionais dos recursos são modeladas por distribuições normais, significando que, para cada classe, é estimada uma média e uma variância para cada atributo. Depois, a probabilidade de um valor observado do atributo ser produzido por essa distribuição é calculada usando a distribuição normal. A probabilidade da classe é então calculada, multiplicando as probabilidades condicionais dos recursos para essa classe [59].

2.3 Tipos de mesas de matraquilhos

Uma MM é composta por oito eixos ou varões, quatro para cada equipa, onde estão posicionados os vinte e dois jogadores. Como num jogo de futebol, o objetivo é introduzir a bola na baliza oposta.

No mercado existem vários tipos de MM, que podem ser distinguidos entre três classes, sendo estas a MM clássica, a MM robotizada e a MM digital.

A MM Clássica é a mais comum. Tal como mostra a Figura 1, a mesa apresenta eixos mecânicos que atuam os jogadores dispostos no campo. O movimento dos eixos fica sobre total controlo do utilizador, podendo realizar qualquer movimento. Dentro desta categoria existem diferentes configurações no que toca à posição dos pés dos jogadores. De resto são semelhantes ou até iguais em todos os outros aspetos. A figura 2 apresenta uma MM Robotizada com elementos eletrónicos que movimentam os eixos autonomamente ou por via de consolas, sendo sempre necessária alguma automação. Isto faz com que os movimentos sejam mais limitados por via das limitações que a máquina pode ter na rotação e velocidade dos eixos. Nos MM Digitais, da figura 3, apesar de poderem ter controlos tradicionais, o jogo é apresentado num ecrã não existindo ação mecânica entre o jogador e a bola. Este tipo de MM é inteiramente dependente da programação do jogo no que diz respeito à ação dos movimentos realizados pelos jogadores.

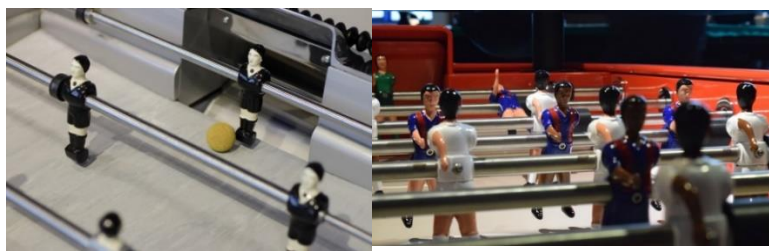


Figura 1 - Exemplos de configuração dos pés dos jogadores em MM clássicos [60], [61]



Figura 2 - Exemplo de MM robotizados [62]



Figura 3 - Exemplo de MM digitais [63]

2.4 Controlo de eixos em mesas de matraquilhos

Alguns dos MM robotizados foram desenvolvidos, no âmbito dos seus projetos, ainda que até ao momento não tenha havido produtos no mercado. Como exemplo, o ‘*Foosbot*’, projeto desenvolvido por estudantes da *Rice University*, que utiliza uma série de sensores infravermelhos e fototransístores para rastrear a bola pela MM e potenciómetros nas engrenagens para rastrear a posição dos eixos [12]. Foi o primeiro projeto de MM robotizadas a ser realizado, levando à criação de novas MM robotizadas um pouco por todo o mundo.

Genericamente, o movimento autónomo dos eixos de MM robotizados é conseguido através de técnicas de IA, como ilustrado na figura 4.

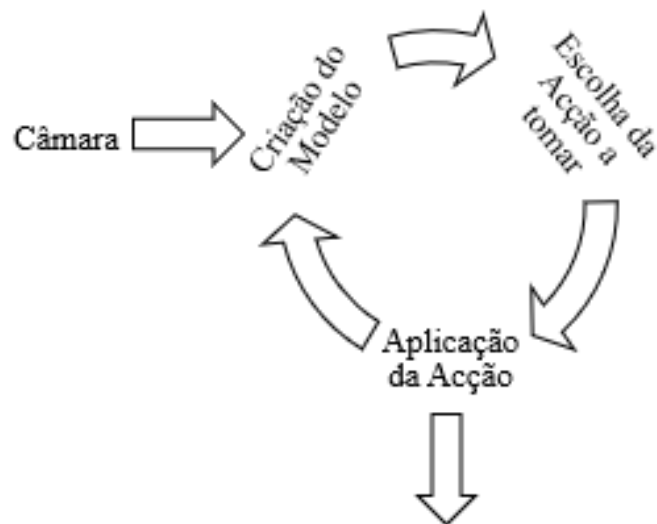


Figura 4 - Exemplo de um fluxo de IA, aplicado a uma MM robotizada

Como apresentado na figura 4, um sistema deste género é alimentado inicialmente por uma imagem proveniente de uma câmara posicionada em cima do campo, de modo a poder visualizá-lo por completo, distinguindo todos os seus componentes (jogadores, bola e balizas). Com base na imagem da câmara é criado um modelo conceptual do campo, como ilustrado na figura 5, sendo este interpretado pela máquina. Com esta técnica o algoritmo fica a saber a posição de cada elemento e também a inclinação dos jogadores, para depois definir a ação a realizar. O movimento resulta do envio de um sinal de saída que ativa os motores produzindo o movimento desejado. É de salientar o facto que cada 'frame' da câmara pode gerar um modelo ainda sem o movimento dos motores ter sido completado, podendo uma nova ação sobrepor-se à anterior. Pode ser uma continuação, não sofrendo quaisquer alterações, ou a IA pode mudar radicalmente de decisão [13].

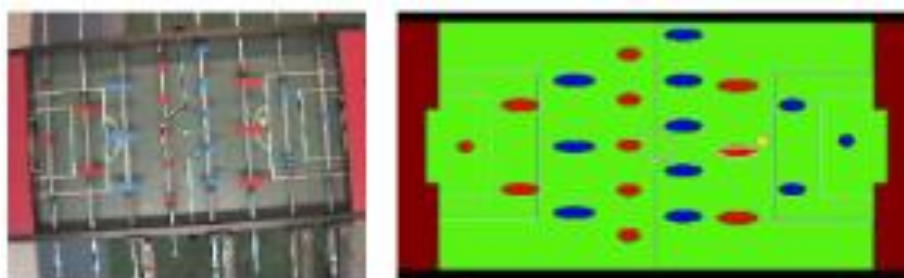


Figura 5 - Exemplo de um modelo gerado por base em imagem [12].

De momento, todas as técnicas apresentadas para mapear a bola num campo de MM baseiam-se neste processo de modelos gerados por imagem. Todas são de carácter académico, apesar de uma delas argumentar que o produto final tem capacidade para entrar no mercado [13]–[17], [62], [64].

Um sistema com sensores a *laser* e espelhos, que consegue calcular a velocidade e direção de um objeto, que neste exemplo, foi utilizada na baliza de uma MM [65]. Embora existam resultados promissores não existem ainda outras abordagens que permitam uma comparação e, portanto, a determinação de uma melhor técnica para o posicionamento dos eixos.

3. Monitorização das ações de pessoas com necessidades específicas sobre uma mesa de matraquilhos

Neste capítulo são detalhados todos os componentes principais do sistema proposto, descritos todos os desenvolvimentos realizados na MM desenvolvida na Universidade, descrevendo as melhorias que foram introduzidas. Faz-se a descrição do mecanismo de aquisição de dados que permite implementar o mecanismo de jogo autónomo e assistido da MM.

Seguidamente, foram realizados testes em ambiente real com os utentes da APPACDM, utilizando a MM e interagindo com ações sobre os novos comandos. Para o movimento autónomo dos eixos foi desenvolvido um algoritmo de IA, cujos dados serão captados através de autómatos e manipulados através de uma linguagem de baixo nível. Este sistema armazena uma base de dados os tempos nos quais houve alterações de posicionamento dos elementos existentes no campo de jogo e que posteriormente podem ser fornecidos ao utilizador, de modo a poder realizar análises ao seu desempenho.

3.1 Adaptação da mesa de matraquilhos robotizada

A Universidade Lusíada em Vila Nova de Famalicão desenvolveu uma MM robotizada (figuras 6 e 7), com patente registada [66]. A MM foi desenvolvida por antigos estudantes do curso de Engenharia Eletrónica e Informática. Mais tarde, sofreu adaptações para permitir a realização de testes aos tempos de reação de pessoas com NE, num ambiente de estudo e investigação [67].

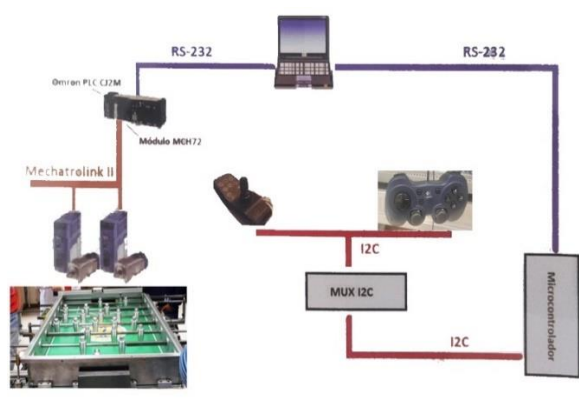


Figura 6 - Fluxo da MM desenvolvida na Universidade (Adaptado de [68])



Figura 7 - MM desenvolvida na Universidade

Na figura 6 é apresentado o sistema de controlo que regista toda a operação, recebendo *inputs* dos comandos com ajuda de um microcontrolador e informações da posição da bola e dos eixos com ajuda de autómatos e sensores da MM. Apesar de se poder monitorizar os dados e alterar parâmetros com o protocolo *Mechatrolink* [69] descrito na figura, a funcionalidade de alterar os parâmetros não está implementada.

Ao contrário do exemplo do sistema utilizado no capítulo 2.2, estes não apresentam qualquer IA ou outro método de aprendizagem. Contudo, a posição da bola é captada com uma matriz de pares de emissores (figura 8) e recetores ‘*laser*’ (figura 9) posicionados à volta do campo. A posição dos jogadores é captada por sensores magnéticos colocados debaixo do solo do campo, de modo a captar a posição de um jogador de cada eixo e extrapolar a posição dos restantes.

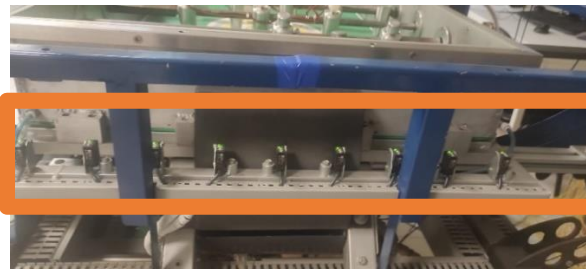


Figura 8 - Emissores de laser



Figura 9 - Recetores de laser

Autómatos são usados para recolher os dados dos sensores, enviando-os por *Ethernet/IP* para um computador local. Este computador permite visualizar a informação recolhida pelos sensores. Para controlar os eixos, são usados quatro comandos da marca *Logitech*, que normalmente são usados na consola de videojogos *XBox*. Os comandos estão conectados a um automático, sendo que este, com ajuda de ar comprimido, permite aos atuadores das figuras 10 e 11 mover os eixos nos dois sentidos e executam o movimento de remate.



Figura 10 - Atuador que permite a rotação do eixo para efetuar o remate

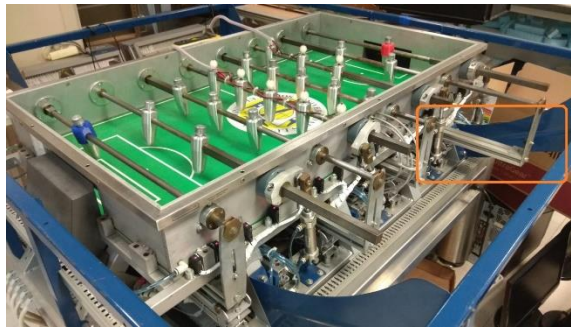


Figura 11 - Atuador que permite o movimento do eixo para efetuar o movimento dos jogadores

Como se trata de um sistema autónomo, o ar comprimido é também usado para lançar a bola para a MM. Como ilustrado na figura 12, após a bola entrar numa das balizas é guiada mecanicamente para uma posição na qual encontra um par emissor/recetor de sensores *laser* que, ao ser ativado, faz com que o ar comprimido mova a base, de forma comparável a um jogo de *pinball*, que consequentemente lança a bola sobre a MM.



Figura 12 - Rampa de lançamento da bola

Estudaram-se pormenorizadamente os componentes mais importantes presentes na MM que pertence à Universidade Lusíada.

- CJ2M-CPU31 da OMRON (*Anexo A*)
 - Estão presentes como autómatos que fazem a ligação I/O entre o computador e os restantes autómatos diretamente conectados à MM;
- CJ1W-ID211 da OMRON (*Anexo B*)
 - Este autómato recebe os dados dos sensores presentes na MM, mandando-os para o autómato antes referido.

- CJ1W-OC211 da OMRON (*Anexo C*)
 - Este autómato, apesar de permitir entradas e saídas como os outros, está destinado a fornecer as saídas para os mover os eixos presentes na MM com base no *input* dos comandos com a ajuda dos atuadores presentes nas figuras 10 e 11.
- Photoelectric Switch E3Z-T81-L/E3Z-T81-D (*Anexo D*)
 - Conjunto de emissores e recetores fotoelétricos que envolvem a mesa, como verificado nas figuras 8 e 9, de modo a captarem a posição da bola.
- Festo Reed Switch VN13 (*Anexo E*)
 - Sensores de proximidade que detetam o posicionamento dos eixos e, consequentemente dos jogadores fixos ao eixo.
- Festo DGPL-25 (*Anexo F*)
 - Atuadores para o deslocamento dos eixos, presentes na figura 11.
- SMC 85N25-50 (*Anexo G*)
 - Atuadores para o remate, visualizados na figura 10, presentes em simbiose com os atuadores anteriormente referidos.

3.2 Sistemas para controlo de processos em robótica

Atualmente, o conceito de robótica dilui-se com o conceito de automação industrial, podendo-se afirmar que se trata do uso de sistemas de controlo, como computadores ou robôs, e tecnologias da informação para lidar com diferentes processos e máquinas numa indústria para substituir um ser humano [70].

A aquisição de dados é a recolha de informação do mundo real, muitas vezes de natureza física, traduzida para valores analógicos, como ilustrado na figura 13. A recolha é feita por sensores e transdutores de modo a serem transformados em sinal elétrico.

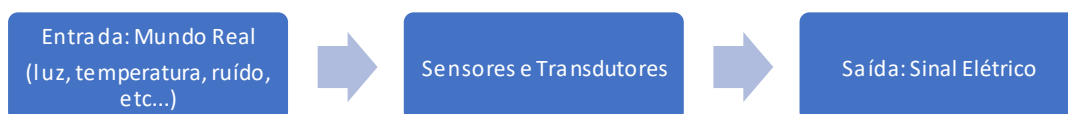


Figura 13 - Aquisição de dados do mundo real

A recolha de informação ilustrada na figura 13 é o equivalente a uma placa de aquisição de dados. Uma placa de aquisição de dados é um componente individual de hardware que é utilizado para capturar dados de sensores ou dispositivos de medição e processá-los em tempo real, transformando o sinal elétrico de analógico para digital.

Uma placa de aquisição de dados, por norma, está inserida num sistema de aquisição de dados. Um sistema de aquisição de dados é uma solução completa para a captura, processamento e armazenamento de dados de medição. Além de incluir placas de aquisição de dados, um sistema de aquisição de dados pode incluir outros componentes, como módulos de *input* e *output*, amplificadores de sinal, filtros e dispositivos para armazenamento de dados [71].

Resumidamente, uma placa de aquisição de dados é um componente individual utilizado para capturar e processar dados de medição, enquanto um sistema de aquisição de dados é uma solução completa para a captura, processamento, armazenamento, análise de dados, podendo por ventura também ter atuadores, como observado na figura 14 [71].

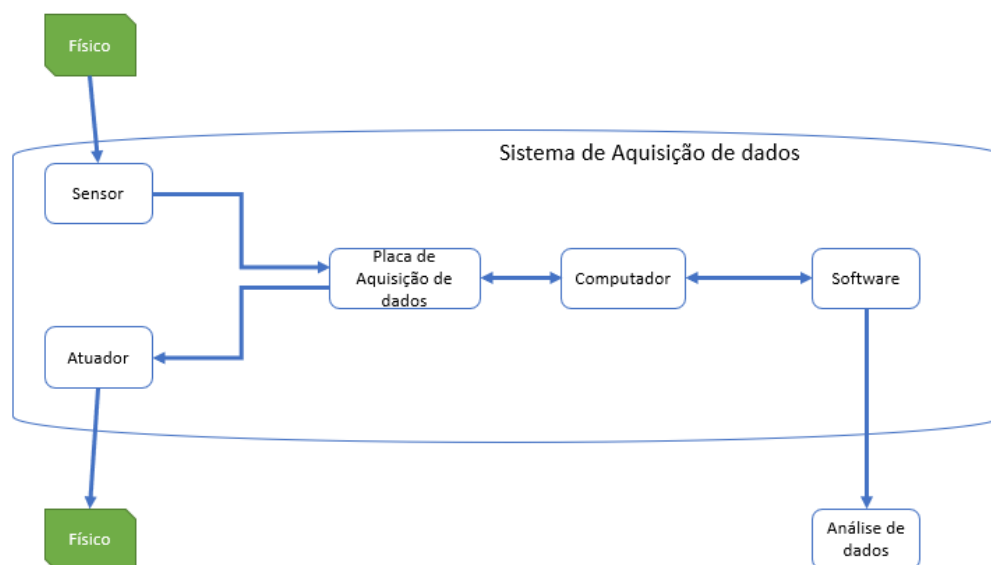


Figura 14 - Sistema de aquisição de dados

De acordo com a figura 14, num sistema exemplo de aquisição de dados, um sensor capta informação do meio físico. A informação é convertida em sinal elétrico e passada para a placa de aquisição de dados. Depois a placa envia a informação para o computador, onde é analisada por via de um software. É posteriormente feita uma análise dos dados e é

gerada uma resposta que chega ao atuador. Por último, o atuador faz alterações no meio físico.

3.3 Desenvolvimento de um interface pessoa-máquina adaptado

Para comprovar que a MM robotizada é um sistema apropriado para o uso por parte de pessoas com NE, foi desenvolvido, com o apoio da APPACDM de Braga, uma atividade com pessoas com NE que aceitassem a realização do mesmo, usando a MM presente na universidade como base. Essas pessoas apresentaram prolemas leves e moderados, de acordo com o *Numerical Rating Scale* [21]. Com a ajuda das terapeutas da APPACDM, definiu-se um teste aos tempos de reação que poderia ser executado com a MM. Também foi verificado pelas mesmas que novos comandos necessitariam de ser criados de modo a facilitar o jogo a quem não conseguisse jogar com os comandos presentes na MM.

Para idealizar um novo comando, teve de se abstrair do mundo em redor da MM, e pensar em mais exemplos práticos do mundo real. De acordo com as informações recolhidos no capítulo 2.1, tendo em conta o estudo ergonómico em colaboração com as terapeutas ocupacionais da APPACDM [19], um novo *joystick* foi pensado e desenhado para que qualquer pessoa o consiga manusear facilmente.

Um fator importante a realçar foi que um novo conjunto de *joysticks* necessitaria de ter uma maior área de ação para garantir que um ligeiro movimento da mão seja suficiente para fazer atuar a MM. A principal vantagem de uma maior área de ação dos *joysticks* adaptados é a possibilidade de movimentar os eixos de forma mais precisa, que pode ser especialmente útil para jogadores que têm dificuldade em mover rapidamente os *joysticks* convencionais. Com uma maior área de ação, os jogadores podem ter mais controlo sobre os eixos e, assim, melhorar a sua performance no jogo.

Outro ponto importante de ter maior área de ação em *joysticks* adaptados é a redução da fadiga muscular, pelo que jogar matraquilhos pode ser um jogo exigente fisicamente, especialmente para pessoas com deficiências físicas. Uma maior área de ação faz com que os jogadores possam reduzir o número de movimentos que precisam fazer para controlar os eixos, o que pode reduzir a fadiga muscular e, assim, prolongar o tempo de jogo. Assim, o *joystick* necessita de ser mais confortável para que se jogue por longos

períodos de tempo. Com um joystick convencional, os jogadores podem sentir desconforto ou até dor nas mãos e braços após um longo período de jogo [72]. Com um joystick adaptado com maior área de ação, objetivo é fazer com que o jogador possa encontrar uma posição mais confortável para as mãos e, assim, jogar por mais tempo sem desconforto.

Em relação à MM, um ponto a considerar é que o *joystick* tem de ser apropriado para este jogo específico, tendo de ser personalizado para atender às necessidades e dificuldades do mesmo. Considera-se que, com uma mão apenas, os jogadores que usarem o *joystick* terão de conseguir realizar as mesmas tarefas de um controlador normal, sendo estas mover o eixo e rematar, ficando assim em pé de igualdade, tornando o jogo mais acessível a todos.

Decidiu-se fazer uma fusão entre um *joystick* e um botão. O *joystick* proposto tem a capacidade de se mover numa só direção, mas nos dois sentidos, de modo a replicar um eixo de uma MM clássica. Para rematar basta pressionar o botão. A facilidade de manuseamento é o objetivo deste sistema recriado, assim a interação entre jogador e *joystick* é simples e intuitiva.

Na figura 15 ilustra-se que, com o simples movimento de empurrar para a frente ou para trás, consegue-se pressionar os interruptores *On/Off* que estão em posição horizontal, e assim dar o sinal para permitir aos eixos avançar ou recuar.

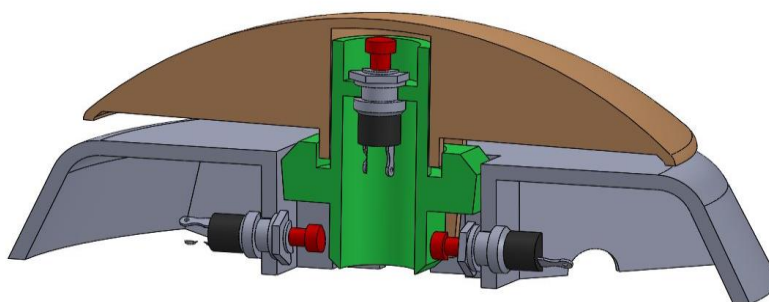


Figura 15 - Recorte do joystick – vista lateral

Para a ação de remate, ao pressionar a superfície do objeto (Figura 16), faz-se com que o interruptor vertical seja acionado e, conseqüentemente, seja enviado um sinal ao sistema de controlo.

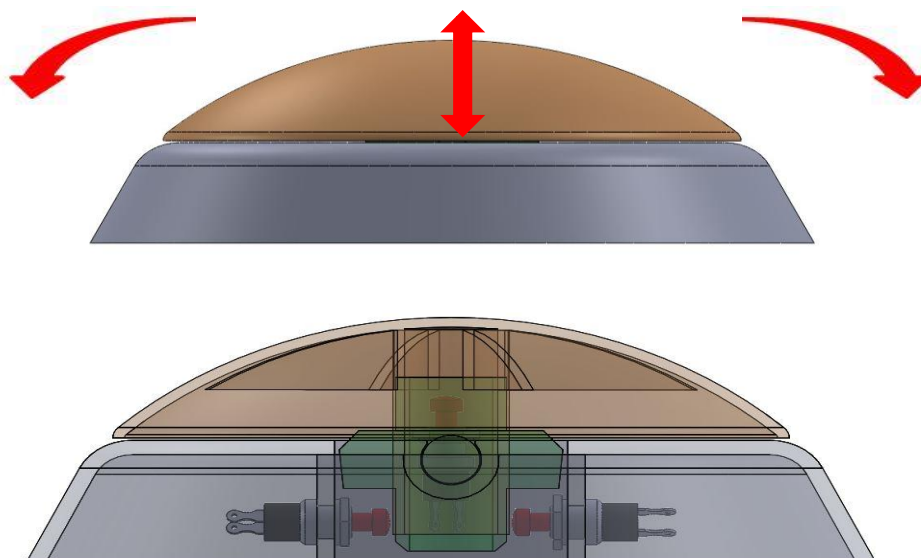


Figura 16 - Recorte do topo do joystick – vista frontal

Finalizou-se esta etapa com a ligação do *joystick* com o *Programmable Logic Controller* (PLC), como representado no *Appendix A*, de modo a substituir os comandos presentes no sistema.

É de realçar o tamanho da área de ação deste *joystick*. Apresenta um tamanho adequado à palma de uma mão, sendo equiparável a um rato de computador. De acordo com o protótipo criado por via de impressão 3D, os movimentos são simples mas bastante perceptíveis, dado que os cliques dos interruptores são notórios, garantindo que se adequa às NE dos utilizadores.

Após implementação, o *joystick* foi aprovado pelas terapeutas da APPACDM, que realçaram a importância deste equipamento lúdico para uma melhor interação homem-máquina.

3.4 Desenvolvimento de programa de teste aos tempos de reação

Com a ajuda das terapeutas da APPACDM, definiu-se um teste aos tempos de reação. Este teste consiste em empurrar a bola em direção ao eixo que está a ser controlado pelo

jogador e registrar o tempo que o jogador demorou a reagir, aplicando movimento ao eixo.

Para o desenvolvimento deste programa em linguagem *ladder*, definiu-se a malha de sensores no campo, como observável na figura 17, para selecionar quais os sensores a usar no teste, com o propósito determinar as sequências da bola.

Numa fase inicial, são definidas as linhas de sensores a serem usadas para o programa no PLC. Para a sequência da bola foi definindo as linhas 16, 12 e 11 porque o eixo que o jogador vai controlar situa-se entre os sensores das linhas 12 e 11. A linha 16 dá margem de tempo ao jogador para mover o eixo

O registo do tempo começa quando a bola ativa o sensor da linha 16. Se passar novamente pelo sensor da linha 16, significa que a bola bateu na parede oposta e regressou ao mesmo sítio e o tempo tem de ser novamente inicializado. O registo da passagem da bola pela linha 12 após passar pela linha 16 serve para validar a trajetória. Se houver movimento do eixo, o tempo é validado e registado. Se não houver movimento do eixo e a bola passar pela linha 11, o registo do tempo é cancelado, esperando-se uma nova passagem pela linha 16 para o tempo ser inicializado e recomeçar-se o teste com uma nova sequência.

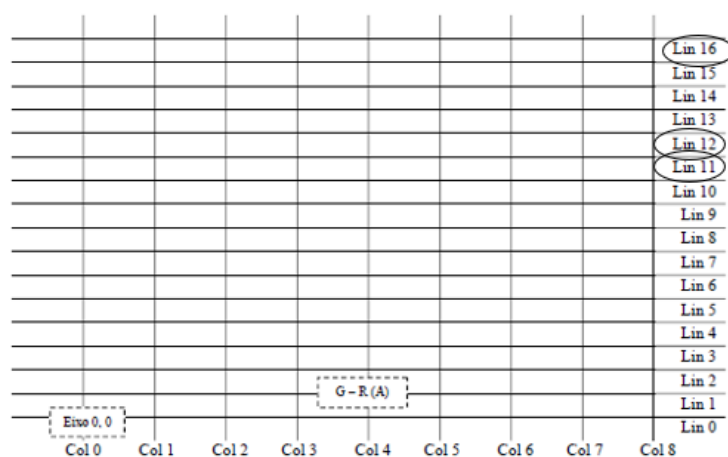


Figura 17 - Definição da estrutura de sensores do campo

Na definição do programa (*Appendix B(a)*), foram criados *bits* auxiliares para garantir que os sinais ficam ativados até alguma ação de fim. Por exemplo, a linha 16 quando ativada, origina a ativação do *bit* auxiliar 10.00, que se realimenta, ficando sempre ativado até que a ativação da linha 11 corte esse sinal.

Assim construído e garantindo o sinal sempre que a bola passa pelos sensores, os *bits* auxiliares garantem a sequência. Se outra trajetória for registrada sem ser a pretendida, os *bits* são anulados, abortando assim toda a sequência que se estava a formar, deixando tudo limpo para um novo registo de *bits*, até que a sequência seja atingida com sucesso.

O programa começa pelo registo de linhas por onde a bola passa, registando uma sequência. Sempre que a bola passa, a linha de sensores número 16 inicia a sequência no programa, ficando à espera que a linha número 12 seja ativada para verificar se a direção da bola é correta. Tendo a sequência certa o jogador terá de acertar na bola para a mesma voltar a passar a linha de sensores número 12. Para se registar nesse momento o seu tempo de remate. Se o remate não for eficaz, a bola passa pela linha de sensores número 11, que logo anula toda a sequência do programa fazendo com que a bola necessite de voltar ao início da jogada para iniciar o programa novamente. Com isso pretende-se que a bola consiga uma sequência limpa até chegar à zona de remate. A bola ao entrar na zona de remate, faz com que o temporizador comece a contabilizar o tempo até que o remate seja efetuado com sucesso, e nesse instante, através do comando MOV, que retira o valor do temporizador e envia essa informação para a memória (*Appendix B(b)*).

A instrução ‘MOV’ é uma instrução da lógica *ladder* que copia o valor de uma *Source* e coloca uma cópia na *Destination*. A *Source* permanece inalterada.

Depois de idealizados os novos comandos e realizado o teste aos tempos de reação, procedeu-se ao desenvolvimento do movimento autónomo dos eixos tendo por base técnicas de IA. Como a lista de I/Os persentes na MM é extensa e podendo o estudo ser realizado com apenas uma parte da MM, definiu-se apenas usar um eixo para demonstrar o efeito proposto neste projeto. Para tal escolheu-se o eixo de um guarda-redes, que contem sete sensores para informar o sistema da posição. Diminuíram-se as posições do campo para apenas posições próximas a esse mesmo eixo. Recorreu-se ao seguinte esquema (figura 18) para se saber quais linhas que devem ser acionadas, quando se estiver a trabalhar diretamente nos autómatos.

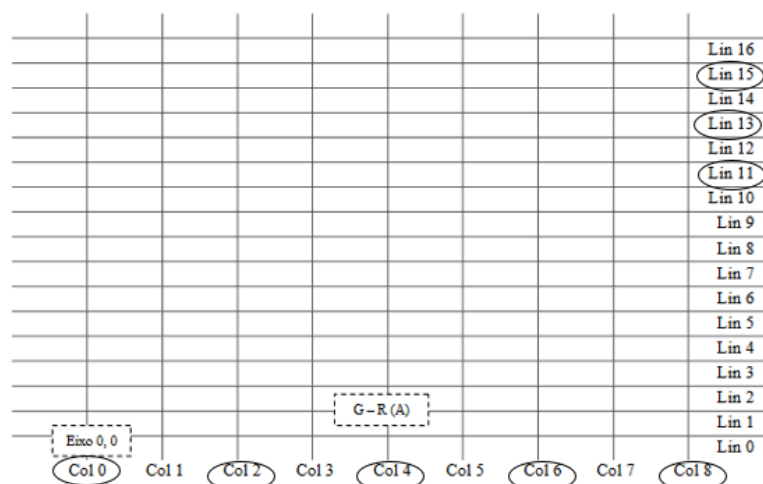


Figura 18 - Sensores de posi33o da bola presentes na MM e respetiva sele333o

Com a ajuda do esquema da figura 18, definiu-se que se vai usar as linhas 11,13 e 15, e as colunas 0,2,4,6 e 8, que, somadas com os sete sensores do eixo do guarda-redes, daria um conjunto de quinze dados, sendo necess33rio apenas um aut33mato para o efeito.

Foram atribuídos nomes (A1...A24), de acordo com a figura 19, para se realizar o mapeamento do movimento da bola. Esta nomenclatura 3e utilizada no c33digo fonte e nos diagramas em *ladder* que s3o necess33rios na programaç33o a baixo nív33l.

	A4	A8	A12	A16	A20	A24
linha 11	A3	A7	A11	A15	A19	A23
linha 13						
linha 15	A2	A6	A10	A14	A18	A22
	A1	A5	A9	A13	A17	A21
	coluna 8	coluna 6	coluna 4	coluna 2	coluna 0	

Figura 19- Nomenclatura de identifica333o de 3reas

3.5 Técnica de inteligência artificial

Por seguinte, procedeu-se na escolha do método geral de IA. Para tal, fez-se um diagrama simplificado (figura 20) do que se pretende com o código, de modo a enquadrar o código ao objetivo que se pretende, e não cair na tentação de fazer o inverso, que seria enquadrar o objetivo ao método utilizado.

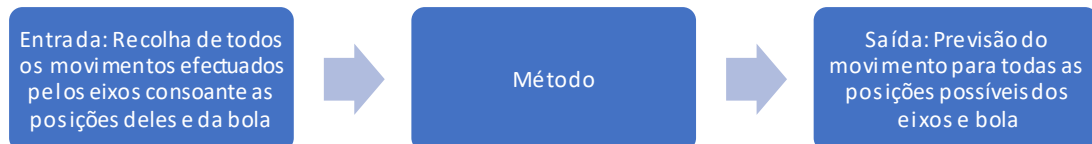


Figura 20 - Diagrama simplificado do código para escolha do método

Esta abordagem apresenta uma grande vantagem e uma desvantagem a ter em conta. Por um lado, todos os movimentos são aprendidos previamente pela MM, não necessitando de uma elevada capacidade de processamento durante o decorrer da atividade. Por outro, trata-se apenas de previsões, portanto, se os dados presentes na recolha forem muito diferentes dos dados reais num determinado momento, a previsão pode ser errada. Apresentando, a figura 20, características de um classificador probabilístico, verificou-se que esta seria a abordagem a usar.

Quanto à escolha do classificador probabilístico para este projeto, e devido a todos os fatores apresentados na revisão bibliográfica, escolheu-se usar o *Naïve Bayes* gaussiano.

De seguida, arrancou-se com o desenvolvimento do mesmo. Desenvolveu-se o seguinte pseudocódigo (figura 21), cujo código está disponível no *Appendix C*.

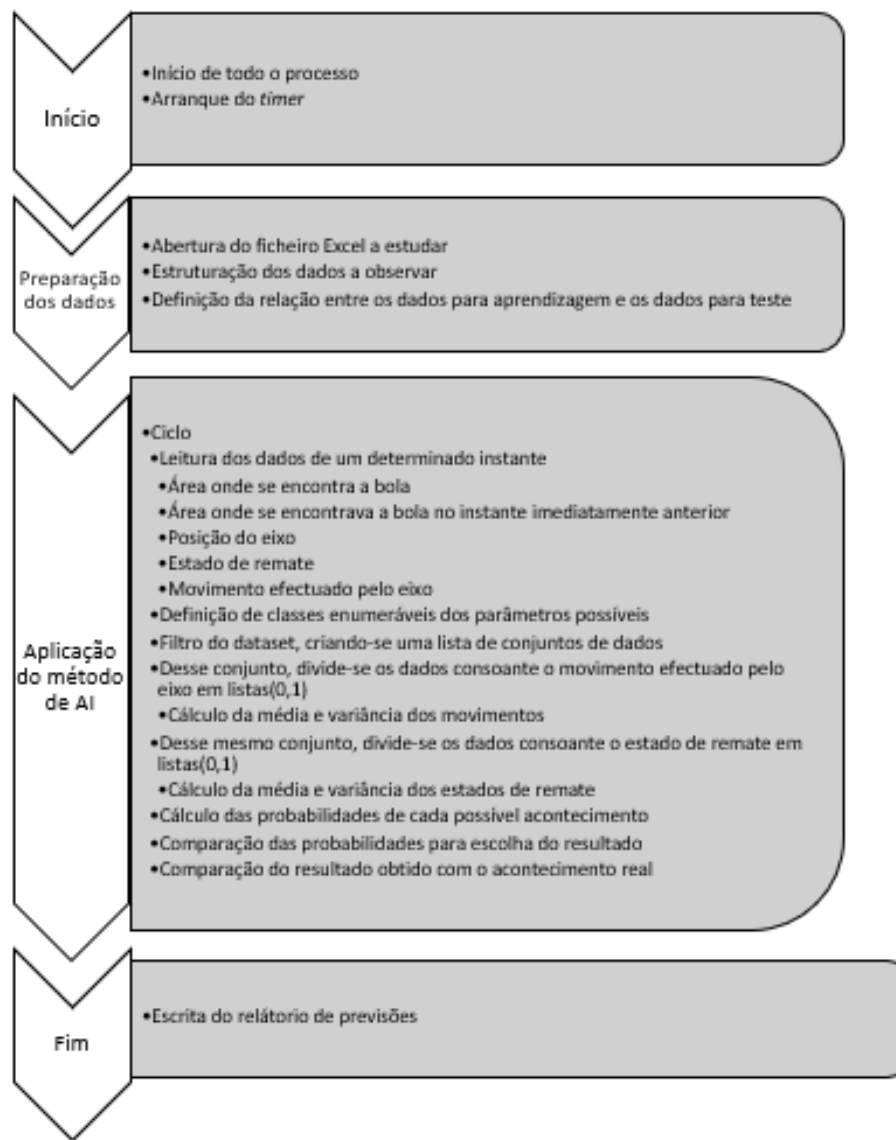


Figura 21 - Pseudocódigo de IA (Naïve-Bayes)

O código começa com um simples arranque de um *timer* que é útil para saber o tempo total de todo o processo. O ficheiro de dados em ‘.csv’ proveniente da MM que se pretende examinar e estruturam-se os dados para serem tratados, definindo também a percentagem de dados que vai servir para a criação do modelo.

No ponto chamado de ‘Aplicação do método de AI’ da figura 21, entra-se num ciclo, onde se leem os restantes dados relativos a cada determinado instante. É chamada uma classe com a IA desenvolvida, o *Naïve Bayes*, onde dentro da mesma era chamado um método. O objetivo deste método é retornar uma matriz de probabilidades com as seguintes informações:

- mover para esquerda;
- não mover;
- mover para a direita;
- rematar;
- não rematar,

para cada uma das vinte e quatro áreas definidas, para cada momento individual do jogo.

Para alcançar o objetivo, o método de cálculo necessita das médias e variâncias dos movimentos. Para tal, primeiro cria-se uma lista que possui todas as situações onde as posições atuais da bola e do eixo, posição anterior da bola e movimento são guardadas. Cria-se uma matriz, que para as vinte e quatro áreas regista um conjunto de ‘0’s’ e ‘1’s’ para cada situação.

Exemplo:						
O eixo moveu-se para a direita e não rematou, estando a bola na área A1						
Esquerda	Não Mover	Direita	Rematou	Não rematou	Posição Bola	Posição Eixo
0	0	1	0	1	A1	GR1
Imediatamente a seguir, o eixo ficou quieto e a bola deslocou-se para A2						
Esquerda	Não Mover	Direita	Rematou	Não rematou	Posição Bola	Posição Eixo
0	1	0	0	1	A2	GR2
Neste caso, o algoritmo fica a saber que quando estava na posição GR2 e a bola está na posição A2 vinda de A1, o eixo não se vai mover nem rematar						

Depois calcula-se a média e variância de cada movimento. O mesmo processo repete-se para os cálculos da média e variância das variáveis de remate.

Após estes calculados, calcula-se a probabilidade de cada possível acontecimento, retornando o resultado do *Naïve Bayes* para cada uma das vinte e quatro áreas e cinco possíveis movimentos.

Depois, os dados recebem um simples tratamento antes de um novo *datamodel* ser gerado e retornado. Por fim, os resultados gerados pelo método voltam para o ponto inicial para serem comparados pelos dados reais recolhidos. Os resultados gerados são guardados num ficheiro ‘.csv’ para futuramente poderem ser consultados pela MM. No último ponto é gerado o relatório de acertos e registado o tempo de execução, sendo concluído todo o processo.

3.6 Desenvolvimento de um *interface* de comunicação

Para este programa em *ladder*, acrescentou-se o conceito de área da figura 19 do subcapítulo 3.4. Neste contexto, área é um conceito abstrato, que tem de ser induzido no programa com ajuda das linhas e colunas.

Primeiro, definiu-se quais iriam ser as memórias que iriam ter as informações das áreas atuais e das áreas do momento imediatamente anterior (figura 22). Também, é inicializada uma variável de controlo (18.00), que vai depois ser utilizada. São necessárias duas áreas de memória para armazenar o valor de vinte e quatro variáveis, pois cada área de memória armazena dezasseis.

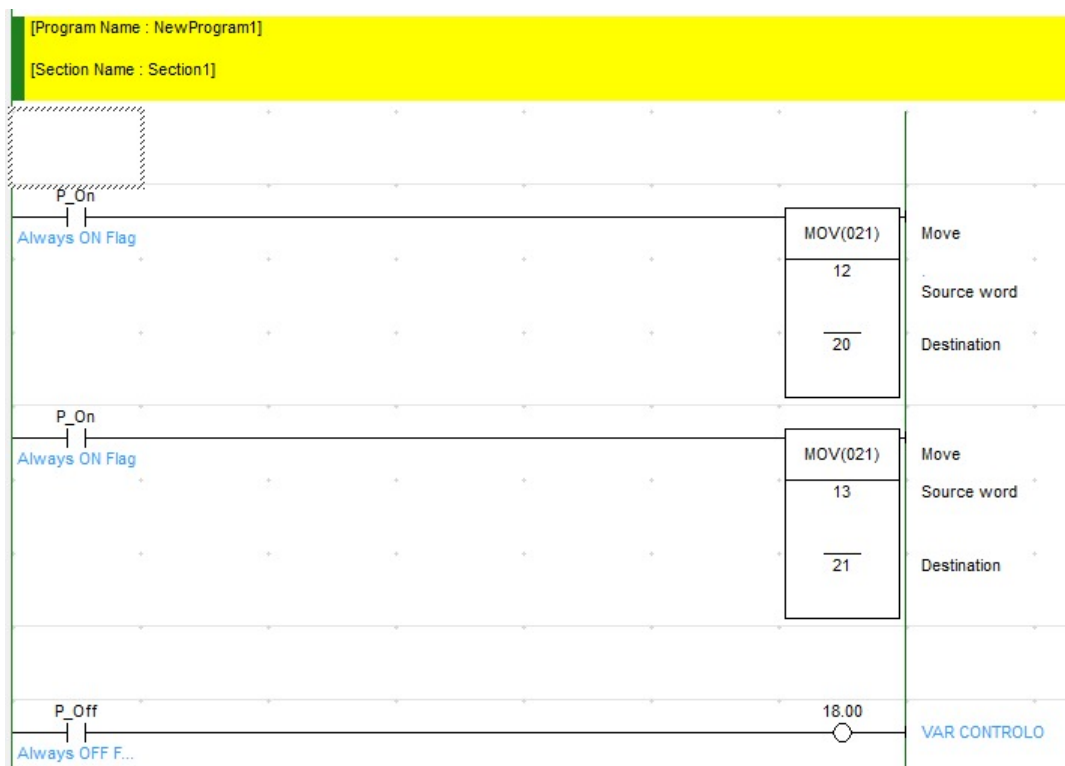


Figura 22 - Memórias com informações das áreas e variável de controlo

Foram necessárias quatro áreas de memória, em que a escolha das mesmas foi de acordo com o mapeamento de *I/O* do sistema de controlo:

- 12 e 13 para as áreas atuais
- 20 e 21 para as áreas imediatamente anteriores.

De seguida, começa-se com área ‘A1’ (figura 23). Esta, como mostra a figura 19, correspondente à nomenclatura das áreas, está limitada apenas pela linha 15 e coluna 8. A área ‘A1’ (12.01), se já estiver acionada, ativa uma primeira variável de controlo ‘JA ESTAVA’ (18.01).

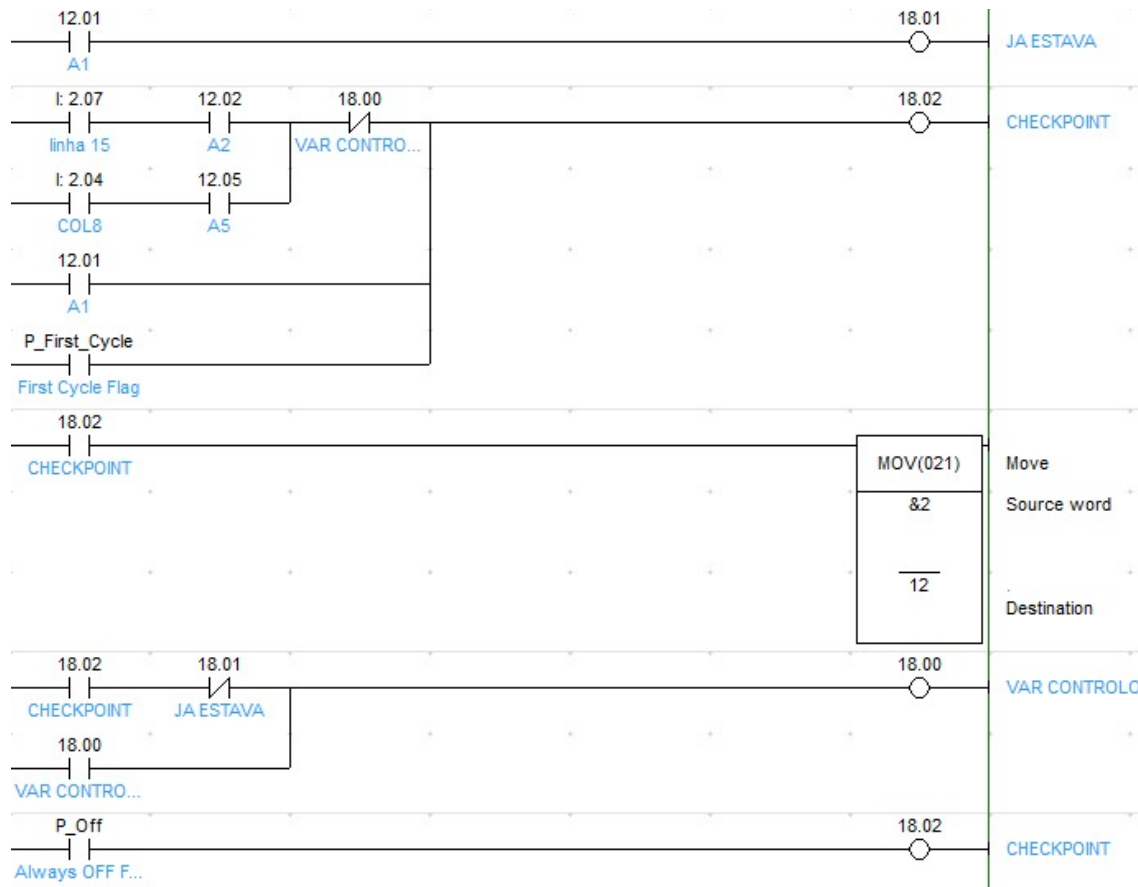


Figura 23 - Área A1

Depois, se a linha 15 (2.07) ou a coluna 8 (2.04), estiverem ativadas e, respetivamente, a área ‘A2’ ou ‘A5’ também estiverem ativadas, passa-se para a verificação da variável de controlo anteriormente mencionada (18.00). Esta apenas indica que se já houve uma alteração de áreas no presente ciclo, uma nova mudança é bloqueada. Ao passar essa verificação entra num ‘checkpoint’. Se já tiver o valor ‘A1’, não necessita de verificação para ir ao ‘checkpoint’. A última exceção especial encontra-se na ‘First Cycle Flag’, que é apenas ativada no primeiro ciclo e ativa única e exclusivamente o ‘A1’, simulando assim

que, quando começa o jogo, a bola passa sempre por aquele local.

O ‘checkpoint’ tem duas funções:

- Efetuar a mudança de área para ‘A1’
 - O MOV(021) apresentado, tem como destino a memória 12 com a *source word* ‘&2’. Isto significa que manda para lá o número 2 decimal, número esse que corresponde ao segundo *bit*, quando convertido a binário, ou seja, o *bit* ‘01’. Logo é ativado o 12.01.
- Se ‘JA ESTAVA’ não for ativado, confirmar a mudança de área na variável de controlo (18.00)

Por último, realimenta-se, caso necessário, a variável de controlo e desativa-se o ‘checkpoint’.

Depois das verificações correspondentes ao ‘A1’ iniciam-se as verificações ao ‘A2’ (figura 24). A única diferença a sublinhar em relação ao ‘A1’ é que, como ‘A2’ está envolvido por ‘A1’, ‘A3’ e ‘A6’, esses três elementos influenciam a ligação do ‘checkpoint’. O MOV(021) também tem valor diferente.

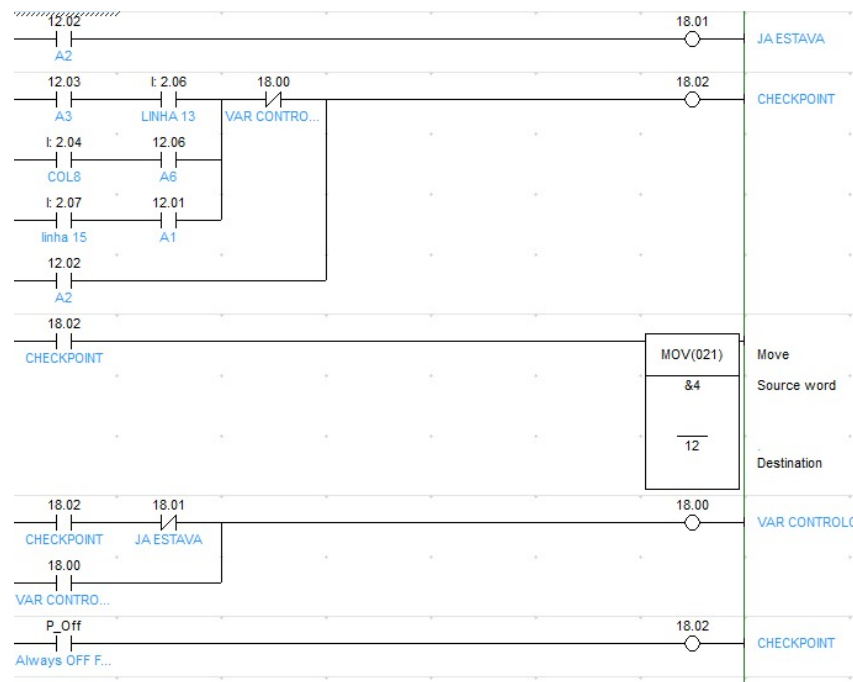


Figura 24 - Área A2

The diagram illustrates a control system with the following components and logic:

- Inputs:** 12.12, A12, 12.05, LINHA 11, 12.08, A8, 13.00, A16, 12.12, A12, 18.02, CHECKPOINT, 18.01, JA ESTAVA, 18.00, VAR CONTRO..., P_Off, Always OFF F...
- Outputs:** 18.01, 18.02, 18.00, 18.02, 18.00, 18.02
- Logic Blocks:**
 - VAR CONTRO...:** A complex logic block that receives inputs from 12.05, 12.11, 12.08, 12.03, 13.00, 12.02, and 12.12. It is used in rungs 1, 2, and 3.
 - MOV(021):** A Move instruction that moves data from a source word to a destination word. It is used in rungs 4 and 5.
 - Always OFF F...:** An instruction that forces an output to be always off. It is used in rung 6.

33

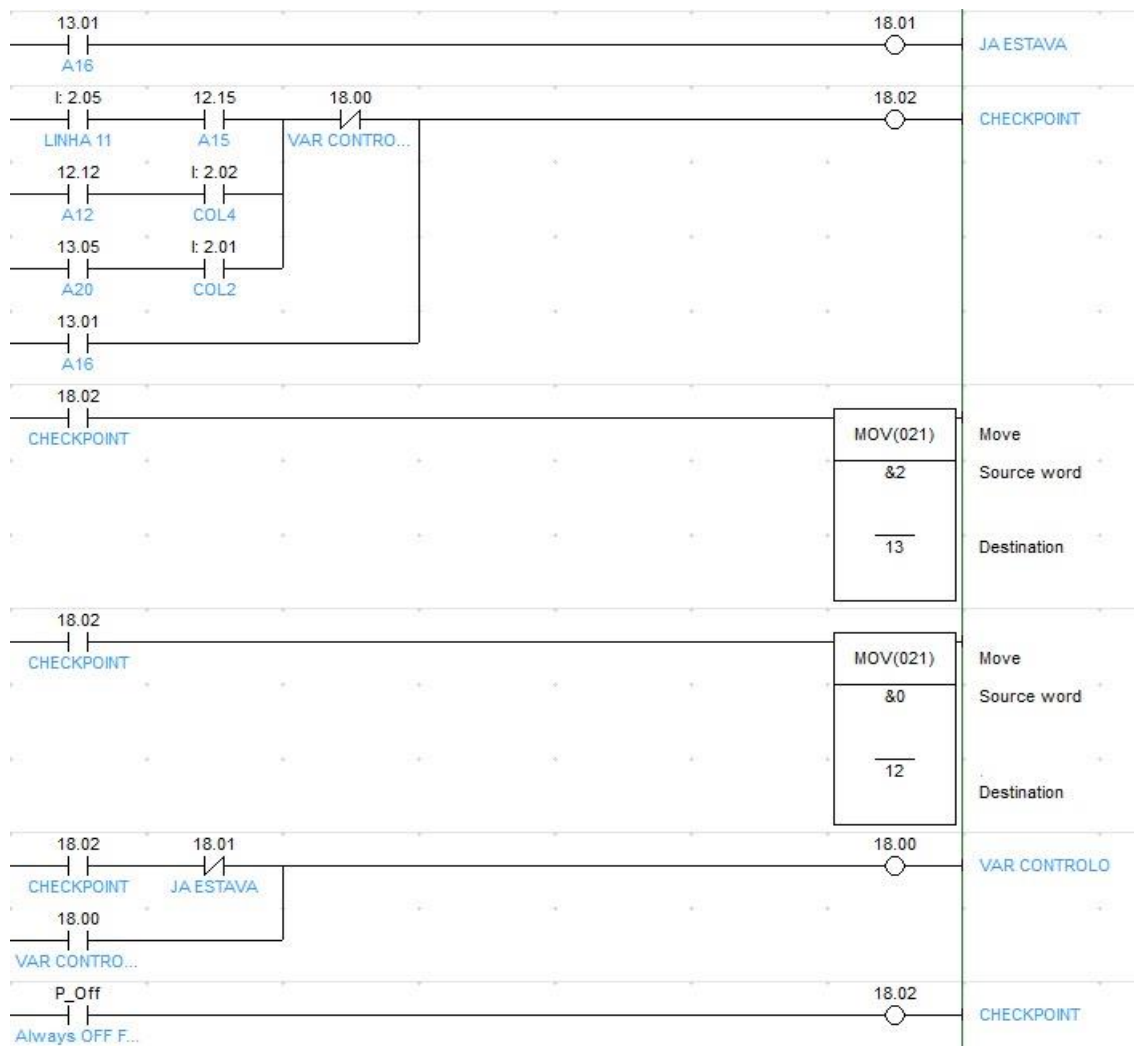


Figura 26 - Área A16

3.7 Procedimento experimental para validação dos dados

Para efeitos de validação do trabalho realizado foi efetuado um teste, cujo procedimento é descrito nos parágrafos seguintes.

Após a programação do algoritmo em *ladder*, procedeu-se ao envio dos dados para o *Excel*, correspondentes às áreas do campo no estado atual e imediatamente anterior, e posição do eixo (2.08 a 2.14), juntamente com a identificação do utilizador. De seguida, criou-se um *Project File* do *Cx-Server*. O *Device CJ2M* já estava criado, utilizado anteriormente, onde consequentemente se criaram os apontadores de memória (figura 27) que correspondem a áreas de memória dos *PLCs* que são usadas neste teste.

Name	Type	Address
\CIO1	Point	CJ2M/1/1/USH
\CIO2	Point	CJ2M/2/1/USH
\CIO12	Point	CJ2M/12/1/USH
\CIO13	Point	CJ2M/13/1/USH
\CIO20	Point	CJ2M/20/1/USH
\CIO21	Point	CJ2M/21/1/USH
\D100	Point	CJ2M/D100/1/USH
\D101	Point	CJ2M/D101/1/USH

Figura 27 - Apontadores de memória das áreas de memória do CJ2M

Na figura 27 estão representados os apontadores de memória que vão ser utilizados, isto é, são as áreas de memória que vão para o *Excel*. Umas com o propósito de comparar com os dados da AI, como o caso dos *CIO* correspondentes às áreas da MM e sensores de posição do eixo:

- CIO12 e CIO13 para os dados atuais;
- CIO20 e CIO21 para os dados imediatamente anteriores;
- CIO2 para os dados dos sensores de posição do eixo, que corresponde ao autómato da posição 2 ligado ao computador.

E outras funcionarão como atuadores:

- CIO1 correspondente aos movimentos do eixo, fazendo com que eixo se mova e remate, que corresponde ao autómato de entrada da posição 1.

Na posição de memória D101 guarda-se a informação sobre o utilizador. *Data Memories* também foram colocados para servirem de possíveis variáveis temporárias, caso não se pudesse alterar diretamente o *CIO*.

Com a ajuda do *OMRON OPC Open Data Server*, aponta-se o *Project File* criado como base (figura 28).

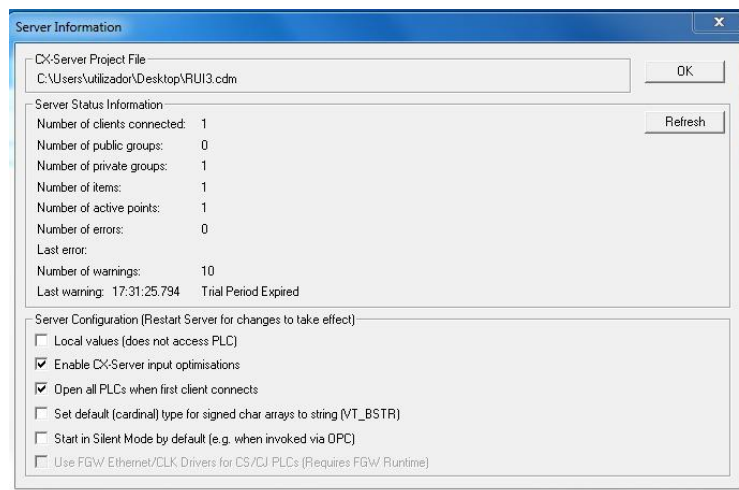


Figura 28 - Inserção do Project File para com o CX-Server

O último programa a ser aberto é o *OPC Expert*. Com este, temos acesso às variáveis anteriormente criadas (figura 29), podendo-se observar o valor em tempo real, conseguindo-se, ainda que manualmente, escrever nas memórias (figura 30)

Realtime Data									
Type	Status	Alarm	Computer	OPC Server	Item ID	Item Name	Value	Time Stamp	Quality
	✓		LEA8 (This Computer)	OPC Serve...	CIO1	CIO1	2080	5:47:14 PM	Good (192)
	✓		LEA8 (This Computer)	OPC Serve...	CIO2	CIO2	256	5:47:15 PM	Good (192)
	✓		LEA8 (This Computer)	OPC Serve...	CIO12	CIO12	2	5:47:18 PM	Good (192)
	✓		LEA8 (This Computer)	OPC Serve...	CIO13	CIO13	0	5:47:19 PM	Good (192)
	✓		LEA8 (This Computer)	OPC Serve...	CIO20	CIO20	2	5:47:22 PM	Good (192)
	✓		LEA8 (This Computer)	OPC Serve...	CIO21	CIO21	0	5:47:23 PM	Good (192)
	✓		LEA8 (This Computer)	OPC Serve...	D100	D100	2080	5:47:25 PM	Good (192)
	✓		LEA8 (This Computer)	OPC Serve...	D101	D101	0	5:47:28 PM	Good (192)

Figura 29 - Variáveis e seus respectivos valores no OPC Expert

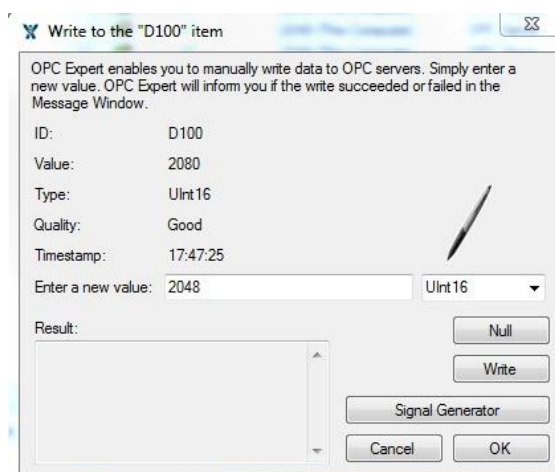
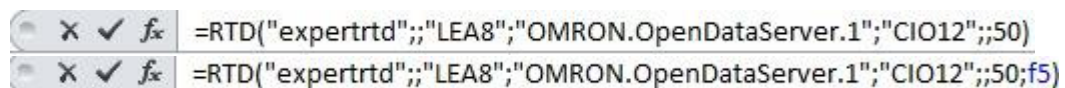


Figura 30 - Escrita manual dos valores

O padrão de atualização dos dados é de 1s, mas este valor pode ser alterado, fixando-se a atualização dos dados em 50 ms.

Por fim, ao ser arrastados os valores do *Realtime Data* para o *Excel*, são geradas equações (figura 31). Essas equações podem ser atualizadas para receber valores de outras células, e, assim atuar na MM como se fosse um comando.



The image shows two Excel formulas in a spreadsheet. Both formulas are: `=RTD("experttrtd";;"LEA8";"OMRON.OpenDataServer.1";"CIO12";;50)`. The first formula is in cell B2 and the second is in cell B3. Both cells show a green checkmark and the formula text.

Figura 31 - Equações com a função RTD fornecidas pelo programa

Nesta etapa verificou-se que os valores apenas alteravam no *Excel* e não alteravam no *OPC Expert*, ou seja, não alteravam no autômato. Foram ponderadas duas soluções. A primeira foi fazer uma ponte entre um *Data Memory* e um *CIO* (figura 32).

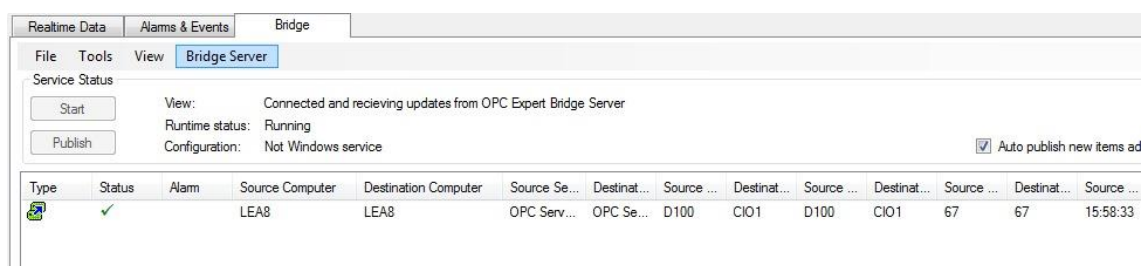


Figura 32 - Ponte entre D100 e CIO1

Após realizada a ponte, verificou-se que os *CIOs* alteravam os *Data Memories*, mas os *Data Memories* ou não alteravam os *CIOs* ou tal alteração durava frações de segundo, revertendo-se de imediato. Assim, procedeu-se a segunda abordagem. No código em *ladder* acrescentou-se um último comando, que move um valor do *Data Memory* (D100) ao CIO1 (figura 33).



Figura 33 - MOV do D100 para a área onde se encontram saídas dos movimentos dos eixos

Após uma alteração no valor do D100 no *Excel*, verificou-se no *OPC Expert* e também no autômato que os valores alteravam. Ou seja, a partir deste momento os eixos da MM movem-se e rematam com base numa alteração no valor do *Excel*, ou seja, já se movem de acordo com o resultado da IA.

Resolvido o problema anterior a unir os dados provenientes da IA com os dados retirados à MM. Os dados provenientes das memórias dos sensores são apresentados como na figura 34.

	A	B	C
1		SENSORES	
2		GR:	1024
3		CIO12:	8192
4		CIO13:	0
5		CIO20:	8192
6		CIO21:	0

Figura 34 - Dados crus provenientes das memórias dos sensores

De acordo com a figura 34, e sabendo que os dados dos sensores do eixo do guarda-redes começam no 2.08 e acabam no 2.14, como o valor é 1024 (2^{10}), sabe-se que o sensor correspondente ao 2.10 está ativado, sendo que este corresponde à posição GR3. Os CIO12 e CIO20 têm o valor 8192 (2^{13}), sabendo-se então que correspondem ao 2.13 (A13). Estes valores são comparados com os do ficheiro gerado pela IA através de uma *Macro*. Os dados captados correspondem aos seguintes dados da tabela (figura 35).

Coluna1	Coluna2	Coluna3	Coluna4	Coluna5
A13	cen	chut	GR3	A11
A13	cen	chut	GR3	A12
A13	cen	chut	GR3	A13
A13	cen	nchut	GR3	A14

Figura 35 - Dados do ficheiro gerado pela IA correspondentes aos dados retirados da MM

De acordo com a figura 35, após a posição atual e anterior serem ambas ‘A13’ e o guarda-redes estar posicionado em ‘GR3’, a informação desta linha é passada para o novo ficheiro (figura 36). A informação a passar é que o eixo não se move (cen), mas chuta (chut),

relativos às colunas 2 e 3 da figura 35. A figura 36 apresenta os valores dos sensores e o respetivo valor dos atuadores.

	A	B	C	D	E	F	G	H	I	J	K
1		SENSORES					ATUADORES				
2		GR:	1024				CIO1	2048		Utilizador	12
3		CIO12:	8192								
4		CIO13:	0								
5		CIO20:	8192								
6		CIO21:	0								

Figura 36 - Resultado do atuador CIO1, consoante os dados dos sensores

A figura 36 apresenta o valor 2048 (2^{11}) para o atuador CIO1. Tal acontece porque a ativação do remate encontra-se na memória 1.11 do CIO1. A ativação para a esquerda encontra-se no 1.06 (2^6) e para a direita no 1.07 (2^7). Pode então ter os seguintes resultados (tabela 2). A variável atuadora ‘Utilizador’ é inserida no manualmente no *Excel* para ser enviada para a MM.

Tabela 2 - Possíveis valores para o CIO1

Esquerda	Direita	Remate	Soma
0	0	0	0
0	0	2^{11}	2^{11} (2048)
0	2^7	0	2^7 (128)
0	2^7	2^{11}	2^7+2^{11} (2176)
2^6	0	0	2^6 (64)
2^6	0	2^{11}	2^6+2^{11} (2112)

3.8 Desenvolvimento da base de dados em contexto de prova de conceito

Num contexto de uma prova de conceito desenvolveu-se uma base de dados em *MySQL*. Esta é constituída por duas tabelas, ‘jogadores’ e ‘mesa’, cujo diagrama está na figura 37.

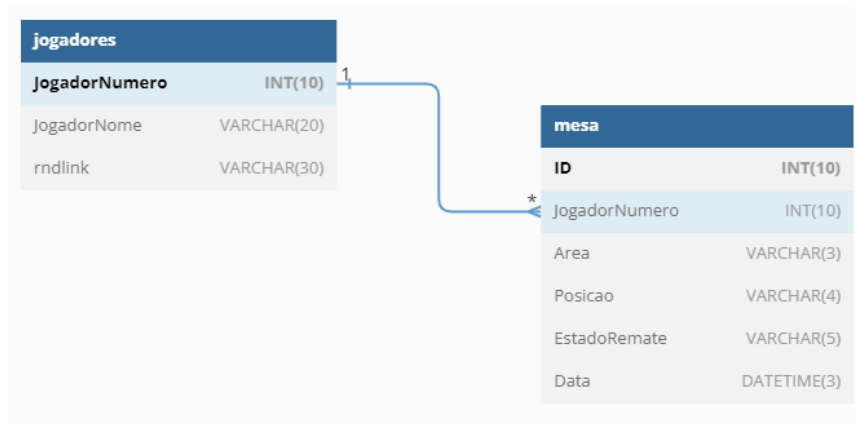


Figura 37 - Tabelas da base de dados

A tabela 'jogadores' tem as informações dos utilizadores. A variável *JogadorNumero*, apresenta o número atribuído ao jogador ao registar-se. Esta informação é inserida diretamente na MM para identificar o utilizador, guardando essa informação no PLC junto com os dados. Um detalhe importante é o facto de, quando o valor é 1, uma Macro faz com que os valores recolhidos aumentem a folha de *Excel* com os dados para análise. Assim, desativar-se-ia o CIO1, que é o autómato que envia informação para a MM, e controlar-se-ia o devido eixo manualmente. A variável *JogadorNome*, contém o nome da pessoa, que é pedido aquando do registo (figura 38). A variável *rndlink*, é apenas justificável enquanto prova de conceito. Neste ambiente, tem o propósito de ser uma *string* aleatória de trinta caracteres que ficaria associada ao utilizador. Esta *string* ficaria agregada ao *link* que o utilizador usaria para visualizar os dados da tabela mesa, criando-se assim um *link* único para cada utilizador. Com este efeito, o utilizador não necessitaria de aceder diretamente ao site para aceder à informação, podendo essa ser descarregada através de softwares próprios para o efeito, como por exemplo, o *Excel*. Esta variável serve também para demonstrar que a tabela 'jogadores' não pode ter apenas o nome e o número como dados, podendo ser reforçada do ponto de vista da segurança.

O formulário de registo de um utilizador apresenta um campo de texto rotulado 'Nome' com o valor 'MiniCaixotim' inserido. Abaixo do campo encontra-se um botão azul rotulado 'Registar'. Na base do formulário, há um link azul rotulado 'Index'.

Figura 38 - Registo de um utilizador, de momento apenas pelo nome

A Tabela ‘*mesa*’, tem as informações dos sensores da MM com os respectivos tempos. Contém a variável *JogadorNumero*, proveniente da tabela ‘*jogadores*’, *Área*, que mostra qual a posição atual da bola, *Posicao*, que mostra qual a posição do eixo, e *EstadoRemate*, que mostra se o eixo se encontra a rematar. Por fim, *Data*, fornecida pelo *Excel* com a função *Agora()*, até à milésima de segundo, para datar todos os movimentos e alterações na MM. A tabela ‘*mesa*’ é atualizada com ajuda de um ficheiro *Excel* (figura 39) com dados provenientes da MM incluindo o momento o qual o dado é captado, enviando de seguida para a base de dados. Ou seja, a tabela ‘*mesa*’ é constantemente atualizada.

	A	B	C	D	E	F
1						
2						
3						
4		Dados Limpos				
5		<i>Área</i>	<i>Posição Guarda-Redes</i>	<i>Estado de Remate</i>	<i>Data e Hora</i>	<i>Nº Utilizador</i>
6		A13	GR3	Não	07/02/2022 12:20:41,600	12

Figura 39 - Dados provenientes da MM

Como observável na figura 39, o ‘Nº Utilizador’ é o mesmo que a figura 36, mostrando que o valor que é enviado para a MM, é retornado para a base de dados, associando-se às restantes informações recolhidas.

Em termos práticos, a relação entre as tabelas pode ser utilizada para realizar consultas que combinem informações sobre os jogadores com informações sobre sua posição na mesa. Por exemplo, é possível consultar a tabela "mesa" para obter informações sobre o estado do remate de um determinado jogador em uma determinada data e hora, e em seguida utilizar a chave estrangeira para vincular essas informações à tabela "jogadores".

Para concluir todo o processo, foi idealizada uma interface *web* com o objetivo de fornecer todos os dados de um utilizador (figura 40). O objetivo é fazer com que uma pessoa, ou alguém responsável pela mesma, que queira ver os seus dados para futuras análises médicas/fisioterapêuticas ou até mesmo num contexto desportivo, possa analisar os tempos de reação.

Dados – MiniCaixotim

Olá MiniCaixotim, estes são os teus dados.

DATA e HORA	Posição Eixo 1	Estado do Remate	Área da Bola
27/9/2021 17:54:17,638	GR2	Não	A1
27/9/2021 17:54:17,727	GR2	Não	A5
27/9/2021 17:54:17,816	GR2	Não	A9
27/9/2021 17:54:17,891	GR2	Não	A13
27/9/2021 17:54:17,898	GR3	Não	A13
27/9/2021 17:54:18,077	GR3	Não	A17
27/9/2021 17:54:18,142	GR4	Não	A17
27/9/2021 17:54:18,150	GR5	Não	A17
27/9/2021 17:54:18,199	GR5	Não	A21
27/9/2021 17:54:18,308	GR5	Não	A17
27/9/2021 17:54:18,502	GR5	Não	A18
27/9/2021 17:54:18,584	GR5	Não	A19
27/9/2021 17:54:19,657	GR6	Não	A20
27/9/2021 17:54:19,910	GR7	Não	A20
27/9/2021 17:54:20,422	GR6	Não	A20

Figura 40 - Interface com dados de um utilizador para estudo futuro

4. Discussão dos resultados

Primeiramente, quanto à realização dos primeiros testes com a MM presente, apesar de não se retirarem conclusões com parâmetros científicos, verificou-se uma disponibilidade das pessoas que participaram e que estes também estavam focados no jogo.

O programa de teste aos tempos de reação que foi concebido mostrou que pode ser uma ferramenta útil para avaliar a eficácia da terapia e medir o progresso dos pacientes ao longo do tempo. Se os pacientes apresentarem melhorias nos seus tempos de reação e precisão no movimento do eixo da mesa de matraquilhos, pode indicar que a terapia a ser realizada está a ser eficaz quanto à coordenação motora e, também, concentração.

Os *joysticks* adaptados em MM são uma solução engenhosa que permite que pessoas com NE joguem matraquilhos. Estes *joysticks* foram projetados especificamente para pessoas com deficiência física e permitem que as pessoas joguem matraquilhos sem a necessidade de fazer movimentos mais complexos. Os *joysticks* foram aprovados pelas terapeutas, ficando assim demonstrado que a MM tem capacidade de se poder tornar bastante vantajosa para as pessoas com NE, ajudando-as nas vertentes terapêuticas e sociais.

Para o controlo da MM, optou-se pelo *ladder*, uma linguagem de baixo nível. O motivo que gerou esta decisão foi que as linguagens de baixo nível permitem um controlo mais preciso e direto sobre a MM. Como a MM é um sistema em tempo real, é necessário responder a constantes mudanças da posição da bola e dos eixos rapidamente e com alta precisão. Não menos importante, as linguagens de baixo nível também permitem um controlo mais detalhado da alocação de recursos do sistema, como a memória e a CPU, permitindo otimizar o desempenho. Especificamente, o *ladder* assemelha-se graficamente a circuitos elétricos, o que simplificou a compreensão do comportamento da MM, facilitando a introdução de alterações ao código do programa.

Os resultados correspondentes aos acertos provêm de dados simulados. Porém, ao reproduzirem fiavelmente a MM, podem-se considerar para este efeito dados provenientes diretamente da máquina. Consideraram-se 12800 dados, sendo que 80% são para aprendizagem e os restantes para teste. Cada dado consiste na posição do eixo a analisar, posição da bola e posição da bola no instante imediatamente anterior. Os acertos do movimento lateral, remate e movimento completo estão descritos na tabela 3 e

restantes detalhes no *Appendix D*.

Tabela 3 - Acertos das previsões do método

	Dados							Tempo de Execução
	Movimento Lateral			Remate			Acertos de Movimento Completo	583s
	Aprendizagem	Decisão	Acerto da Decisão	Aprendizagem	Decisão	Acerto da Decisão		
Valor	10240	2560	2547	10240	2560	2476	2463	
Porcentagem	80%	20%	99.5%	80%	20%	96.7%	96.2%	

A taxa de acerto obtida foi de 99.5% para os movimentos e 96.7% para os estados, que juntos alcançaram 96.2% de acertos, sendo estes valores bastante satisfatórios. Para o movimento dos eixos verificou-se uma taxa de 99.5%, o que significa que só falhou numa em cada duzentas previsões. Com o aumento dos dados a estudar, proporcional ao uso futuro da MM, estes valores podem melhorar. A vantagem do uso de um classificador probabilístico é o facto de ‘fazer a MM melhorar a sua capacidade’. O objetivo não passa por colocar a MM a jogar melhor, mas colocar a mesma a jogar com capacidade semelhante a um indivíduo. Assim sendo, os 583 segundos iniciais que o programa demora a fazer um novo mapa de previsões são meramente um detalhe que não afeta o resultado final.

A resolução do campo para a deteção da bola revelou-se um fator crítico que tem de ser abatido. Os presentes sensores dispõem de uma distância que não permite saber com exatidão a posição da bola, mas apenas uma área ampla onde a bola se situa num determinado momento. Colocando-se mais sensores no campo, conseguir-se-ia um melhor mapeamento e, por consequência, saber-se-ia com mais exatidão a posição da bola.

Um dos pontos que poderia afetar a concretização do projeto é a influência do atraso no envio das ações do algoritmo de IA. Um fator com peso no atraso que o algoritmo pode ter é o sistema de controlo onde está alojado todo o processo. Verificou-se que, quando o sistema de controlo se encontra a processar apenas os programas referentes à atividade, não se detetam atrasos significativos com a perceção de parecer instantâneo. Porém, quando há mais processos a decorrer na máquina que exijam um processamento mais complexo, a máquina apresenta atrasos consideráveis que tornam o algoritmo menos eficiente. Também, um incremento no número de sensores para um melhor mapeamento

da MM implica um sistema de controlo mais eficiente. Para evitar este tipo de problemas, uma máquina com mais poder de processamento, ou até uma máquina com o exclusivo propósito de ser apenas usada para a MM, podem ser soluções para este problema.

A base de dados apresentada consegue demonstrar, enquanto prova de conceito, a viabilidade do sistema, mostrando que os dados recolhidos permitem a análise dos tempos de reação de quem utiliza a MM. A tabela ‘mesa’ teria de aceitar um *array* de posições e de estados de remate, equivalentes aos eixos da MM, ao inverso de uma posição e um estado de remate, como se refere neste estudo. Assim, mesmo com um maior mapeamento da posição da bola e jogadores, apenas aumentavam os possíveis valores desses parâmetros da tabela. Por seu lado, a tabela ‘jogadores’, como não foi um ponto fulcral do projeto, apresenta debilidades relativas à segurança do sistema.

A idealização da interface *web* revela-se simples e prática. O utilizador fica com toda a informação da MM, relativamente a tempos, mudanças de posição da bola e do eixo, podendo ter uma compreensão completa do evento.

5. Conclusão e trabalho futuro

No trabalho desenvolvido mostra-se, enquanto prova de conceito, que é possível avançar para o desenvolvimento completo de uma MM robotizada que cumpra com o propósito de ajudar pessoas com e sem NE.

A solução encontrada pode ser usada em diversas áreas, como a saúde, onde indivíduos podem testar os seus tempos de reação para serem futuramente consultados e usados para análise. Poderá também existir um contexto competitivo, onde, por exemplo, com a adição dos sensores das balizas e uma adição de temporizadores ou contadores de golos, podem-se efetuar jogos, cujo resultado também poderia ser guardado na base de dados. Por fim, surge o contexto recreativo, onde se poderá escolher quantos eixos serão operados pela IA e jogar normalmente, sem nenhum objetivo além do lazer.

O principal benefício dos *joysticks* adaptados em mesas de matraquilhos é que tornam o jogo acessível a pessoas com NE. Isso significa que pessoas que de outra forma não poderiam jogar matraquilhos agora têm a oportunidade de o fazer. Isso aumenta a inclusão social e ajuda a melhorar a qualidade de vida dessas pessoas. Assim, a utilização de *joysticks* adaptados com maior área de ação pode ser vista como uma forma de promoção da igualdade de oportunidades. Pessoas com deficiências físicas muitas vezes enfrentam barreiras para participar em atividades físicas e desportos, pelo que com *joysticks* adaptados com maior área de ação, os jogadores com deficiências físicas podem ter as mesmas oportunidades de participar.

A terapia com o uso de uma MM robotizada pode ser uma forma inovadora e eficaz de tratar pacientes com diferentes tipos de necessidades terapêuticas. A coleta e análise dos dados recolhidos durante essas sessões de terapia são cruciais para avaliar a eficácia do tratamento e, por isso, a importância de mostrar esses dados aos pacientes é fundamental. A mostra dos dados recolhidos durante a terapia pode ajudar os pacientes a entenderem o progresso que estão a ter, sendo importante para os motivar a continuar a participar no tratamento e para se esforçarem mais para alcançar as suas metas terapêuticas. A mostra dos dados pode ajudar a moldar a terapia de acordo com as necessidades de cada paciente. Com dados precisos e detalhados, os profissionais de saúde podem personalizar o tratamento para atender às necessidades individuais dos pacientes e adaptá-lo para alcançar melhores resultados. Os dados podem ajudar a identificar as áreas que precisam

de mais trabalho e ajustes para melhorar o desempenho e a eficácia da terapia. Para os casos onde um grupo de indivíduos é observado por terapeutas, os dados também podem ser usados para ajudar as terapeutas a visualizar e compreender seu desempenho e progresso de uma pessoa em relação ao grupo, possibilitando a obtenção de novas referências para o progresso de uma terapia. Uma grande vantagem de uma apresentação crua dos dados é a transparência. Quando os pacientes têm acesso aos dados recolhidos durante a terapia, podem ter uma melhor compreensão do tratamento e do seu processo, podendo observar como os seus comportamentos e progressos são medidos e avaliados, o que pode ajudar a fortalecer sua confiança no tratamento e na equipa terapêutica.

Para a IA foi escolhido um classificador probabilístico com a base de se saber que assim os eixos iriam jogar de forma idêntica aos utilizadores cujos dados são transportados para o ficheiro *Excel* da IA. Assim, um dos passos que se teria a fazer para se tornar o mais idêntico possível à realidade seria aumentar o número de movimentos anteriores para análise, de modo a poder interpretar, e por fim, executar movimentos mais complexos como fintas e simulações. Com as percentagens de acerto geral em 96,2%, pode-se concluir que o método aplicado consegue representar e replicar os movimentos realizados pelos jogadores, sendo o *Naive Bayes* gaussiano uma escolha acertada.

Contudo, a MM necessita de melhorias para apresentar resultados mais fiáveis relativamente à posição da bola e para estar mais bem preparada para a ajuda em terapia. Observações podem ser feitas ao *hardware* e ao método de IA

Considera-se necessário para um trabalho futuro, um melhor mapeamento da MM. Para isso é aconselhável utilizar um tapete de sensores eletromagnéticos. Para um conjunto de novos e melhores comandos ser idealizado, pode ser feito um estudo ergonómico dos utilizadores.

É aconselhável a MM ter um *display* para que se possa introduzir os dados de utilizadores, que neste momento se insere como atuador num ficheiro *Excel*. Poderiam-se colocar no *display* dados em tempo real da terapia a realizar. Também se poderia colocar o resultado do jogo nesse *display* com a informação de sensores que se encontram dentro das balizas.

Ao serem adicionados todos os sensores da atual ou futura MM, existe a possibilidade de fazer atuar todos os atuadores que a MM apresenta, como com o exemplo do atuador que transporta a bola de volta para o campo de jogo após marcar um golo na MM atual,

necessitando-se apenas de atualizar o programa em *ladder*.

Quanto a uma possível alteração no método de IA, se o objetivo passar a ser fazer a MM jogar o melhor possível, recomendar-se-ia duas soluções. As duas envolvem aprendizagem por via de recompensas, como em redes neuronais, onde a MM era premiada se marcasse golo e penalizada se sofresse, até chegar a uma fórmula de sucesso. A diferença estaria no método de recolha dos dados. Primeiramente, pode ser com a própria MM, que tem a vantagem de ser o sítio onde futuramente o método de IA iria ser aplicado, não se descartando qualquer variante física da MM, mas com a desvantagem de ser extremamente moroso. A segunda solução, mais rápida, seria recriar virtualmente um campo com as dimensões da MM, e fazer rapidamente o estudo. Neste último caso, por mais atenção ao detalhe que se tenha, pode ser muito complicado simular o comportamento físico de todo o sistema. Mesmo assim, poderá ser a forma mais eficaz.

Quanto ao *hardware*, podem-se realizar algumas melhorias. No que toca às áreas do campo, verifica-se que são demasiado grandes para se saber ao detalhe a posição da bola e a posição do eixo é analisada por relativamente poucos sensores, ou seja, o aumento de sensores é considerado essencial para um bom funcionamento da MM. É necessária uma maior capacidade de memória dos controladores, sendo insuficiente para enviar todos os dados dos autómatos da MM para o computador local. Os tempos de resposta do sistema completo, apesar de ao início parecerem longos e problemáticos, revelaram-se rápidos e suficientes para a taxa de amostra de 50 ms do programa OPC Expert.

Relativamente à base de dados, futuramente necessitaria de ser acrescentada segurança, tanto numa estrutura de *login* como na encriptação dos dados da máquina. Quanto à visualização dos dados, poder-se-ia desenvolver um serviço para fazer uma análise estatística dos dados sem recorrer a programas externos, de modo a complementar a vertente terapêutica do projeto.

Com as variadas possíveis alterações ao *software* e *hardware* da MM, podendo a mesa necessitar de melhoramentos recorrentes, é aconselhável utilizar uma base de dados não relacional, como o caso do *MongoDB*, cujas colunas das tabelas podem ser acrescentadas ou modificadas sem que isso afete os dados anteriormente recolhidos.

Referências

- [1] Merriam Webster, “Special Needs.” https://www.merriam-webster.com/dictionary/special_needs
- [2] CANTIC, “ABC Sebran Main Page (PT),” *Sobre o Sebran*, 2015. <https://cantic.org.pt/wp-content/uploads/2015/05/sebran-300x189.png> (accessed Sep. 01, 2021).
- [3] The Mirror, “Robotic legs,” *Robotic legs for disabled shown off in Tokyo*, 2012. <https://i2-prod.mirror.co.uk/incoming/article410634.ece/ALTERNATES/s615b/hightech-robots-pic-getty-364353144.jpg> (accessed Sep. 02, 2021).
- [4] M. Laranjeira *et al.*, “Mechanical Design of a Standing Frame adapted for Children with mental deficiency,” *Procedia CIRP*, vol. 70, pp. 278–283, 2018, doi: 10.1016/j.procir.2018.04.086.
- [5] B. Allen, “Delivering the benefits of technology to people with disabilities,” *Comput. Control Eng. J.*, vol. 9, no. 2, pp. 89–94, Apr. 1998, doi: 10.1049/cce:19980206.
- [6] A. E. Blackhurst, “Perspectives on Applications of Technology in the Field of Learning Disabilities,” *Learn. Disabil. Q.*, vol. 28, no. 2, pp. 175–178, May 2005, doi: 10.2307/1593622.
- [7] A. Craig, Y. Tran, P. McIsaac, and P. Boord, “The efficacy and benefits of environmental control systems for the severely disabled.,” *Med. Sci. Monit.*, vol. 11, no. 1, pp. RA32-9, Jan. 2005, [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/15614204>
- [8] M. J. Scherer, “Assessing the benefits of using assistive technologies and other supports for thinking, remembering and learning,” *Disabil. Rehabil.*, vol. 27, no. 13, pp. 731–739, Jun. 2005, doi: 10.1080/09638280400014816.
- [9] Eunice Kennedy Shriver National Institute of Child Health & Human Development, “How does rehabilitative technology benefit people with disabilities?,” 2018.

- <https://www.nichd.nih.gov/health/topics/rehabtech/conditioninfo/help> (accessed Dec. 19, 2021).
- [10] K. Rubin, “The benefits of assistive technology and how innovation in the tech world can help people who have disabilities.”
<https://www.stratosphenetworks.com/blog/the-benefits-of-assistive-technology-and-how-innovation-in-the-tech-world-can-help-people-who-have-disabilities/>
 - [11] ReWalk, “ReWalk: More Than Walking,” 2015. <https://rewalk.com/rewalk-personal-3/>
 - [12] Wayback Machine - Wikipedia, “Foosbot,” 2005.
<https://web.archive.org/web/20060913000000/http://en.wikipedia.org/wiki/Foosbot>
 - [13] M. Tacke, T. Weigel, and B. Nebel, “Decision-theoretic planning for playing table soccer,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 3238 LNAI, pp. 213–225, 2004, doi: 10.1007/978-3-540-30221-6_17.
 - [14] et al. AEERHARD, Michael, “Autonomous Foosball Table,” 2007. [Online]. Available: http://www.eskibars.com/projects/foosball_robot/proposal.pdf
 - [15] R. Janssen, M. Verrijt, J. de Best, and R. van de Molengraft, “Ball localization and tracking in a highly dynamic table soccer environment,” *Mechatronics*, vol. 22, no. 4, pp. 503–514, Jun. 2012, doi: 10.1016/j.mechatronics.2012.02.009.
 - [16] T. Weigel, D. Zhang, K. Rechert, and B. Nebel, “Adaptive Vision for Playing Table Soccer,” 2004, pp. 424–438. doi: 10.1007/978-3-540-30221-6_32.
 - [17] M. Aeberhard, S. Connelly, E. Tarr, and N. Walker, “Single Player Foosball Table with an Autonomous Opponent,” 2007, [Online]. Available: http://www.eskibars.com/projects/foosball_robot/final_rpt.pdf
 - [18] E. Knox, “Rehabilitation Through Table Football.”
<https://www.theadventuremedic.com/features/rehabilitation-table-football/>
 - [19] P. R. Gomes *et al.*, “Task-Based Automatic Evaluation of People with

- Intellectual Disabilities Performed on a Robotic Table Soccer,” in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2019, pp. 6145–6148. doi: 10.1109/EMBC.2019.8856411.
- [20] UNESCO, “the Salamanca Statement Framework,” *Policy*, no. June, pp. 7–10, 1994, doi: E D -94/WS/ 1 8.
- [21] J. Li *et al.*, “Validation of a Disability Assessment Tool Based on the International Classification of Functioning, Disability, and Health in the Chinese Context,” *Front. Rehabil. Sci.*, vol. 3, Apr. 2022, doi: 10.3389/fresc.2022.855502.
- [22] United Nations, “Ending marginalization of persons with disabilities ‘a matter of justice’, UN-backed summit told,” 2018.
<https://news.un.org/en/story/2018/07/1015492>
- [23] APPACDM - Braga, “Historia,” 2022. <https://www.appacdm-braga.pt/história> (accessed May 09, 2022).
- [24] Mockitt, “Definition and Stages of Prototype development.”
<https://mockitt.wondershare.com/prototyping/prototype-development.html>
- [25] S. L. Calvert *et al.*, “The American Psychological Association Task Force assessment of violent video games: Science in the service of public interest.,” *Am. Psychol.*, vol. 72, no. 2, pp. 126–143, Feb. 2017, doi: 10.1037/a0040413.
- [26] S. Al-zahrani *et al.*, “Opinion of mental healthcare providers on the impact of electronic games on psychiatric patients in Saudi Arabia,” *Informatics Med. Unlocked*, vol. 25, p. 100647, 2021, doi: 10.1016/j.imu.2021.100647.
- [27] M. Kowal, E. Conroy, N. Ramsbottom, T. Smithies, A. Toth, and M. Campbell, “Gaming Your Mental Health: A Narrative Review on Mitigating Symptoms of Depression and Anxiety Using Commercial Video Games,” *JMIR Serious Games*, vol. 9, no. 2, p. e26575, Jun. 2021, doi: 10.2196/26575.
- [28] “AEROBIC EXERCISE IN SCHIZOPHRENIA: CLINICAL CONSEQUENCES AND NEUROBIOLOGICAL EFFECTS OF A NOVEL TREATMENT

- STRATEGY,” *Schizophr. Res.*, vol. 136, p. S56, Apr. 2012, doi: 10.1016/S0920-9964(12)70203-7.
- [29] H. Dingman, “Hands-on: How Microsoft’s Xbox Adaptive Controller lets gamers with disabilities play: The Xbox Adaptive Controller is a versatile device for people with limited mobility.,” *Good Gear Guide*. p. 1, 2018. [Online]. Available:
<https://search.ebscohost.com/login.aspx?direct=true&db=edb&AN=130188009&site=eds-live>
- [30] Windows Central, “How Microsoft’s Xbox Adaptive Controller changed the game for one man with disabilities,” 2019.
<https://www.windowscentral.com/how-xbox-adaptive-controller-champions-inclusion>
- [31] P. J. Standen, C. Camm, S. Battersby, D. J. Brown, and M. Harrison, “An evaluation of the Wii Nunchuk as an alternative assistive device for people with intellectual and physical disabilities using switch controlled software,” *Comput. Educ.*, vol. 56, no. 1, pp. 2–10, Jan. 2011, doi: 10.1016/j.compedu.2010.06.003.
- [32] M. Mrabet, Y. Rabhi, and F. Fnaiech, “Development of a New Intelligent Joystick for People with Reduced Mobility,” *Appl. Bionics Biomech.*, vol. 2018, pp. 1–14, 2018, doi: 10.1155/2018/2063628.
- [33] M. Lindström, A.-C. Bäckström, C. Henje, and G. Stenberg, ““When I use the electric wheelchair, I can be myself” – real-life stories about occupational identity construction,” *Scand. J. Occup. Ther.*, pp. 1–15, Jul. 2022, doi: 10.1080/11038128.2022.2093268.
- [34] Logitech, “Adaptive Gaming Kit”, [Online]. Available:
<https://www.logitechg.com/en-eu/products/gamepads/adaptive-gaming-kit-accessories.943-000339.html>
- [35] T. Gulrez *et al.*, “Controlling wheelchairs by body motions: A learning framework for the adaptive remapping of space,” Jul. 2011, [Online]. Available: <http://arxiv.org/abs/1107.5387>

- [36] L. Torok, M. Pelegriño, D. G. Trevisan, E. Clua, and A. Montenegro, “A Mobile Game Controller Adapted to the Gameplay and User’s Behavior Using Machine Learning,” 2015, pp. 3–16. doi: 10.1007/978-3-319-24589-8_1.
- [37] I. H. Sarker, “Machine Learning: Algorithms, Real-World Applications and Research Directions,” *SN Comput. Sci.*, vol. 2, no. 3, p. 160, May 2021, doi: 10.1007/s42979-021-00592-x.
- [38] C. C. Aggarwal, *Neural Networks and Deep Learning*. Cham: Springer International Publishing, 2018. doi: 10.1007/978-3-319-94463-0.
- [39] J. Heaton, “Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning,” *Genet. Program. Evolvable Mach.*, vol. 19, no. 1–2, pp. 305–307, Jun. 2018, doi: 10.1007/s10710-017-9314-z.
- [40] X. Yao and Y. Liu, “Machine Learning,” in *Search Methodologies*, Boston, MA: Springer US, pp. 341–373. doi: 10.1007/0-387-28356-0_12.
- [41] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.
- [42] T. Hastie, R. Tibshirani, and J. Friedman, “Overview of Supervised Learning,” 2009, pp. 9–41. doi: 10.1007/978-0-387-84858-7_2.
- [43] J. Zou and L. Schiebinger, “AI can be sexist and racist — it’s time to make it fair,” *Nature*, vol. 559, no. 7714, pp. 324–326, Jul. 2018, doi: 10.1038/d41586-018-05707-8.
- [44] M. I. Jordan and T. M. Mitchell, “Machine learning: Trends, perspectives, and prospects,” *Science (80-.)*, vol. 349, no. 6245, pp. 255–260, Jul. 2015, doi: 10.1126/science.aaa8415.
- [45] G. H. John and P. Langley, “Estimating Continuous Distributions in Bayesian Classifiers,” Feb. 2013, doi: 10.48550/1302.4964.
- [46] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” Jan. 2012, [Online]. Available: <http://arxiv.org/abs/1201.0490>
- [47] I. H. Sarker, “A Machine Learning based Robust Prediction Model for Real-life

- Mobile Phone Data,” Feb. 2019, [Online]. Available:
<http://arxiv.org/abs/1902.07588>
- [48] D. J. Hand and K. Yu, “Idiot’s Bayes: Not So Stupid After All?,” *Int. Stat. Rev.*, vol. 69, no. 3, pp. 385–398, Dec. 2001, doi: 10.1111/j.1751-5823.2001.tb00465.x.
 - [49] A. Hussain, “K-Nearest Neighbors (KNN) and its Applications,” 2020.
https://medium.com/@arman_hussain786/k-nearest-neighbors-knn-and-its-applications-7891a4a916c6
 - [50] A. Mammone, M. Turchi, and N. Cristianini, “Support vector machines,” *WIREs Comput. Stat.*, vol. 1, no. 3, pp. 283–289, Nov. 2009, doi: 10.1002/wics.49.
 - [51] J. M. Moguerza and A. Muñoz, “Support Vector Machines with Applications,” *Stat. Sci.*, vol. 21, no. 3, Aug. 2006, doi: 10.1214/088342306000000493.
 - [52] S. L. Ting, W. H. Ip, and A. H. C. Tsang, “Is Naïve bayes a good classifier for document classification?,” *Int. J. Softw. Eng. its Appl.*, vol. 5, no. 3, pp. 37–46, 2011.
 - [53] Stringfixer, “Modelo Gerativo.” https://stringfixer.com/pt/Generative_model (accessed Nov. 18, 2021).
 - [54] W. Bussab and P. Morettin, *Estatística básica*. 2010.
 - [55] A. Garg and D. Roth, “Understanding probabilistic classifiers,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2167, pp. 179–191, 2001, doi: 10.1007/3-540-44795-4_16.
 - [56] R. Vats, “Gaussian Naive Bayes: What You Need to Know?”
<https://www.upgrad.com/blog/gaussian-naive-bayes/>
 - [57] Machine Learning +, “Gaussian Naive Bayes,” *How Naive Bayes Algorithm Works?* <https://www.machinelearningplus.com/wp-content/uploads/2018/11/f4.png>
 - [58] “Pattern Recognition and Machine Learning,” *J. Electron. Imaging*, vol. 16, no. 4, p. 049901, Jan. 2007, doi: 10.1117/1.2819119.

- [59] P. Dönmez, “Introduction to Machine Learning,” *Nat. Lang. Eng.*, vol. 19, no. 2, pp. 285–288, Apr. 2013, doi: 10.1017/S1351324912000290.
- [60] The Man’s Cave, “Italian Configuration,” *Foosball A-Z – Different Types of Tables*, 2019. <https://i2.wp.com/www.themenscave.sg/wp-content/uploads/2017/11/Bonzini-002.jpg?w=800&ssl=1> (accessed Jul. 09, 2021).
- [61] The Man’s Cave, “Spanish Configuration,” *Foosball A-Z – Different Types of Tables*, 2019. <https://i1.wp.com/www.themenscave.sg/wp-content/uploads/2017/11/RS-Barcelona.jpg?w=800&ssl=1> (accessed Jul. 09, 2021).
- [62] B. Kelly and R. Perrine, “Automated Foosball Table Project,” *Princeton University Senior Thesis Project*, 2016. <https://i.ytimg.com/vi/lUqlOcFfsVY/maxresdefault.jpg> (accessed Jul. 10, 2021).
- [63] L. Dorn, “Digital Foosball Table,” *Laughing Squid*, 2019. <https://i0.wp.com/laughingsquid.com/wp-content/uploads/2019/08/Interactive-Digital-Foosball-Table.png?resize=350%2C200&ssl=1> (accessed Nov. 23, 2021).
- [64] R. Janssen, J. de Best, and R. van de Molengraft, “Real-Time Ball Tracking in a Semi-automated Foosball Table,” 2010, pp. 128–139. doi: 10.1007/978-3-642-11876-0_12.
- [65] S. Lindner, R. Weigel, and A. Koelpin, “A low-cost laser barrier based vectorial velocity measurement system,” in *2016 IEEE SENSORS*, Oct. 2016, pp. 1–3. doi: 10.1109/ICSENS.2016.7808494.
- [66] P. M. P. R. Gomes and V. E. de M. L. da S. Pereira, “Mesa Robotizada Para Jogo De Futebol Ou Matraquilhos,” PT 105490 B, 2012 [Online]. Available: http://servicosonline.inpi.pt/pesquisas/GetFasciculo?bole_ano=2012&bole_num=137&proc_moda=01&proc_num=105490
- [67] Perdana, *2005_Book_SearchMethodologies*, vol. 53, no. 9. 2018.
- [68] P. R. G. Lima *et al.*, “Task-Based Automatic Evaluation of People with

- Intellectual Disabilities Performed on a Robotic Table Soccer,” *IEEE Trans. Eng. Manag.*, 2019, doi: 10.1109/EMBC.2019.8856411.
- [69] Yaskawa, “MECHATROLINK-II.”
<https://www.yaskawa.com/products/drives/industrial-ac-drives/industrial-network-communication-protocols/mechatrolink-ii>
- [70] SUSE, “Industrial Automation.” <https://www.suse.com/suse-defines/definition/industrial-automation/>
- [71] M. Baptista, “Sistemas de Aquisição de dados,” 2005, [Online]. Available: http://www.telematica.utfsm.cl/telematica/site/artic/20121218/asocfile/20121218214713/ili_239.pdf
- [72] Fabio Tandioy, “Why do gamers hands hurt?” <https://fabiotandioy.com/en/why-do-gamers-hands-hurt/>

Anexos

Anexo A – CJ2M-CPU31 da OMRON



Este autômato possui as seguintes características:

- Processadores rápidos; O tempo de execução da instrução é de 40 ns,
- Módulos de *I/O* pulsados podem ser montados para habilitar funções de posicionamento para até quatro eixos. O módulo fornece contadores de alta velocidade, entradas de interrupção e saídas *Pulse Width Modulation*.
- Execução de blocos de funções mais rápida, tratamento de interrupções mais rápido, menos tempo de sobrecarga.
- A memória de execução adicionada para blocos de funções permite a programação estruturada e orientada a objetos.
- A porta *Ethernet* suporta *EtherNet/IP*, conexão com software de suporte, comunicações entre *PLCs* e transferências de dados com *File Transfer Protocol*
- A porta USB padrão em todos os modelos permite que o software de suporte se conecte diretamente a um grande número de sistemas.
- Compatível com todas as unidades existentes de fonte de alimentação, *I/O*, controle e comunicação *CJ1*.

Anexo B – CJ1W-ID211 da OMRON



As características deste autômato são:

- Modelos de entrada de alta velocidade disponíveis, atendendo a aplicações versáteis.
- Modelos de 24V em corrente contínua, 100V e 200V em corrente alternada para ligar dispositivos com diferentes tipos de saídas.
- Modelos de 24V em corrente contínua podem ser conectados a dispositivos com saídas NPN ou PNP, não havendo necessidade de selecionar a polaridade.
- Um filtro digital na unidade pode ser definido para reduzir a influência do ruído externo.

Anexo C – CJ1W-OC211 da OMRON



As principais características são:

- Modelos de saída de alta velocidade estão disponíveis, atendendo a aplicações versáteis.
 - Tempo de resposta ON: 15 μ s
 - Tempo de resposta OFF: 80 μ s
- As unidades de saída estão disponíveis com qualquer um dos três tipos de saída: relé, triac ou transístor.
- Possui unidades de saída com proteção contra curto-circuito de carga.

Anexo D – Photoelectric Switch E3Z-T81-L /E3Z-T81-D



Emissores e recetores fotoelétricos infravermelhos

- Método de deteção através de um feixe
- Distância de deteção de 15000 mm
- Método de configuração através de ajuste manual
- Método de conexão por cabo
- Comprimento do cabo de 5m
- Tipo de saída PNP
- Tempo de resposta de 1ms
- Tensão de alimentação é de 12V a 24V em corrente contínua

Anexo E – Festo *Reed Switch* VN13



É um sensor de proximidade magnético. É posicionado um conjunto destes sensores dentro do atuador do eixo (Festo DGPL-25) de modo a mapeá-lo. Um sensor é ativado quando a parte movel do atuador (sendo de material capaz de fazer atuar o sensor), passa rente ao mesmo.

Anexo F – Festo DGPL-25



É um eixo atuador preparado para ambientes industriais, acionado com o apoio de ar comprimido, cuja precisão do passo é de 10mm. Para evitar travagens bruscas que possam danificar o atuador, ou até mesmo o sistema no qual se encontra inserido, aquando de uma passagem de estado nas memórias, este atuador possui amortecimento eletrónico.

Anexo G – SMC 85N25-50



Atuador cilíndrico que permite o remate dos eixos da MM, acionado com ar comprimido. Como o código do seu nome indica, tem 25mm de diâmetro e 50mm de curso. A variação do curso aplicada sobre o eixo da MM faz com que o eixo rode, permitindo o remate.

Appendices

Appendix A - Ligação do joystick com o *Programmable Logic Controller*

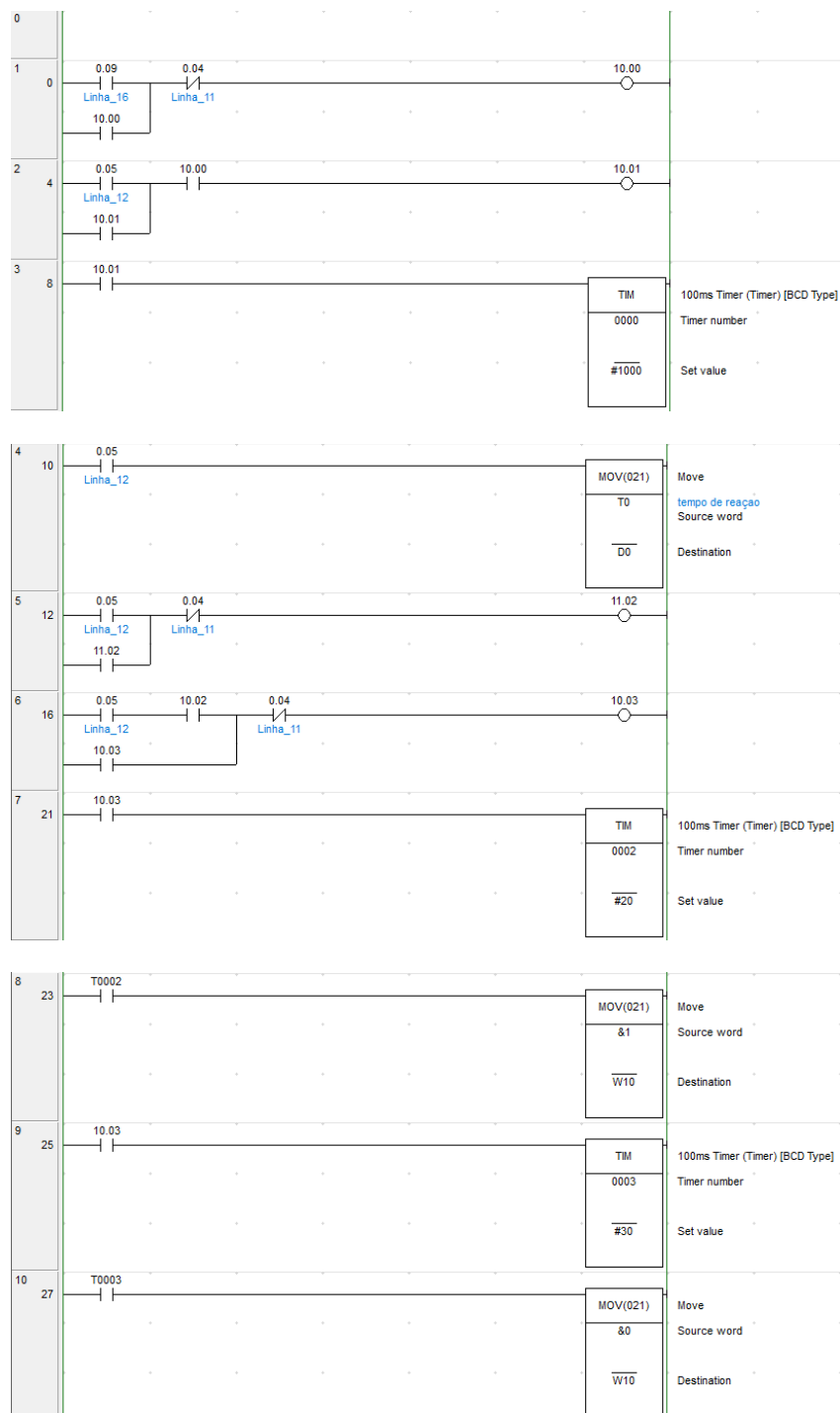
Esta parte do projeto usa um *PLC* CJ2M-CPU31 com dois módulos: um módulo de entradas e outro de saídas.



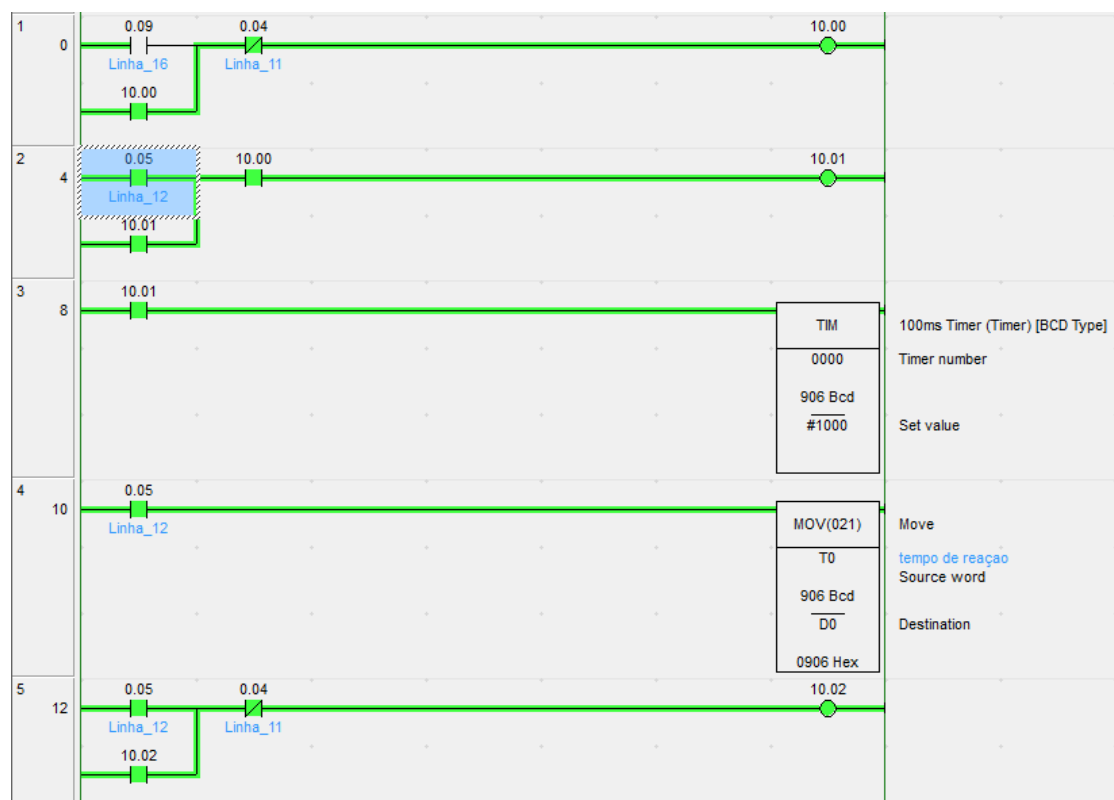
Tendo o *PLC* montado juntamente com os módulos, as ligações de entradas e saídas desse mesmo *PLC* foram feitas através das entradas e saídas da MM, assim, sempre que ativa uma entrada na MM, esse mesmo sinal seja mandado para o *PLC* utilizado. Sempre que o programa manda um sinal de saída é dado um sinal do *PLC* para a MM, de modo que tenham a informação necessária para atuar, tendo uma ação sobre o jogo.

Appendix B – Programação do autômato para os primeiros testes

(a)



(b)



Appendix C – Códigos correspondentes ao desenvolvimento da IA

Ficheiro 'Início.py'

```
from FicheiroNaiveBayes import *
import time
start_time = time.time()
per=0.8
cnt=0
cntchut=0
cnttudo=0
data=learn=search=resultL=[]
fich= open("ficheiro_dados_maquina.csv", "r", encoding='utf-8-sig')

for x in fich:
    l=x.replace("\n","")
    s=l.replace(",",".")
    d=s.split(";")
    dados=(d[30],d[26],d[27],d[28],d[29])
    element=datamodel(dados[0],dados[1],dados[2],dados[3],dados[4])
    data.append(element)

learn=data[0:int(len(data)*per)]
search=data[int(len(data)*per):]

for x in search:
    b=nbayes(learn,x)
    element=b.Result()
    if x.A29 == element.A29:
        cnt +=1
    if x.A27 == element.A27:
        cntchut +=1
    if x.A29 == element.A29 and x.A27 == element.A27:
        cnttudo+=1
    resultL.append(element)

fResult=open("resultado_final.csv","w")
d=""

for x in resultL:
    d+=str(x.A1)+";"+x.A29+";"+str(x.A27)+";"+x.GR+";"+x.ZANT+"\n"

fResult.write(d)
fResult.close()
```

```

print("Dados para aprendizagem:")
print(len(learn))
print("Dados a descobrir")
print(len(search))
print("Previsões Corretas Movimento")
print(cnt)
print("Percentagem Previsões Corretas Movimento")
print(cnt/len(search)*100)
print("Previsões Corretas Remate")
print(cntchut)
print("Percentagem Previsões Corretas Remate")
print(cntchut/len(search)*100)
print("Previsões Totalmente Corretas")
print(cnttudo)
print("Percentagem Previsões Totalmente Corretas")
print(cnttudo/len(search)*100)
print("percentagem centro correto")
print(cntcenele/cntcen)
print("Tempo de execução")
print((time.time() - start_time))

```

Ficheiro *'FicheiroNaiveBayes.py'*

```

import math
from itertools import chain
from enum import Enum
from collections import defaultdict
import numpy as np

class datamodel:
    def __init__(self, a1,a27,a28,a29,a30):
        #zonas do campo bola
        self.A1=str(a1)
        self.A27=A27(a27).name
        self.ZANT=str(a28)
        self.A29=A29(a29).name
        self.GR=str(a30)

class A29(Enum):
    esq="esq"
    dir="dir"
    cen="cen"
    nada="nada"

class A27(Enum):
    chut="chut"
    nchut="nchut"

```

```

class GR(Enum):
    GR1="GR1"
    GR2="GR2"
    GR3="GR3"
    GR4="GR4"
    GR5="GR5"
    GR6="GR6"
    GR7="GR7"

class nbayes(datamodel):

    def __init__(self,dataset,d):
        self.dataset=dataset
        self.d=d

    def Result(self):
        f1=self.CalculoA1()
        arraynova=0
        fEsquerda=1
        fDireita=1
        fNada=1

        fchut=1
        fnchut=1
        arraynova=np.array(f1)
        #print("arraynova;",arraynova)
        for j in arraynova:

            if(j[0]>0 or j[1]>0 or j[2]>0 and (j[3]>0 or j[4]>0)):
                fEsquerda*=j[0]
                fNada*=j[1]
                fDireita*=j[2]

                fchut*=j[3]
                fnchut*=j[4]
                #print(fEsquerda,fNada,fDireita,fchut,fnchut)
            if fEsquerda>max(fDireita,fNada):
                if fnchut>fchut:
                    new_value=
datamodel(self.d.A1,A27.nchut.value,self.d.ZANT,A29.esq.value,self.d.G
R)

                else:
                    new_value=
datamodel(self.d.A1,A27.chut.value,self.d.ZANT,A29.esq.value,self.d.GR
)

            elif fDireita>=max(fEsquerda,fNada):
                if fnchut>fchut:
                    new_value=
datamodel(self.d.A1,A27.nchut.value,self.d.ZANT,A29.dir.value,self.d.G

```

```

R)
        else:
            new_value=
datamodel(self.d.A1,A27.chut.value,self.d.ZANT,A29.dir.value,self.d.GR
)
        elif fNada>=max(fEsquerda,fDireita):
            if fnchut>fchut:
                new_value=
datamodel(self.d.A1,A27.nchut.value,self.d.ZANT,A29.cen.value,self.d.G
R)
            else:
                new_value=
datamodel(self.d.A1,A27.chut.value,self.d.ZANT,A29.cen.value,self.d.GR
)
        return new_value

def CalculoA1(self):#calculo de quando o jogador esta nas posicoes
p

        listach=[]
        listanch=[]
        vdict=[]
        mean=[ [ 0 for i in range(5) ] for j in range(24) ]
        variance=[ [ 0 for i in range(24) ] for j in range(5) ]
        rtp1=list(filter(lambda x: x.A29==self.d.A29
and x.ZANT==self.d.ZANT and x.GR==self.d.GR,self.dataset))
        vdict.append([(x.A1, x.A29, x.ZANT,x.GR) for x in rtp1])

        o=0
        ma=[[[[] for x in range(24)] for y in range(3)]]

        for key in vdict:
            if len(key)>0:
                for key2 in key:
                    o+=1
                    for b in range(24):
                        if key2[1]=='esq':

                            if key2[0]=="A%d" %(b+1):
                                ma[0][b].append(1)
                            else:
                                ma[0][b].append(0)

                        elif key2[1]=='cen':
                            if key2[0]=="A%d" %(b+1):
                                ma[1][b].append(1)
                            else:
                                ma[1][b].append(0)

```



```

        elif key2[1]=='dir':
            if key2[0]=="A%d" %(b+1):
                ma[2][b].append(1)
            else:
                ma[2][b].append(0)

    for key in vdict:
        if len(key)>0:
            for key2 in key:
                for b in range(24):
                    if key2[1]=='esq':
                        if key2[0]=="A%d" %(b+1):
                            mean[b][0]+= float(1/o)
                    elif key2[1]=='cen':
                        if key2[0]=="A%d" %(b+1):
                            mean[b][1]+= float(1/o)
                    elif key2[1]=='dir':
                        if key2[0]=="A%d" %(b+1):
                            mean[b][2]+= float(1/o)

    for a in range(3):
        for b in range(24):
            if(len(ma[a][b])!=0):
                variance[a][b]=
math.sqrt(1/len(ma[a][b]))*sum(map(lambda x: math.pow(x-
mean[b][a],2),(ma[a][b])))

    rtp1=list(filter(lambda x: x.A27 =="chut" and x.ZANT and x.GR,
self.dataset))
    listach.append([(x.A1, x.A27, x.ZANT,x.GR) for x in rtp1])
    rtp1=list(filter(lambda x: x.A27 =="nchut" and x.ZANT and
x.GR, self.dataset))
    listanch.append([(x.A1, x.A27, x.ZANT,x.GR) for x in rtp1])

    man=[[] for x in range(24)] for y in range(2)]

    for l in listach:
        if len(l)>0:
            for l2 in l:
                for b in range(24):
                    if l2[1]=="chut":
                        if o>0:
                            if l2[0]=="A%d" %(b+1):
                                man[0][b].append(1)
                                mean[b][3]+= float(1/o)
                            else:
                                man[0][b].append(0)

```

```

for l in listanch:
    if len(l)>0:
        for l2 in l:
            for b in range(24):
                if l2[1]=="nchut":
                    if o>0:
                        if l2[0]=="A%d" %(b+1):
                            man[1][b].append(1)
                            mean[b][4]+= float(1/o)
                        else:
                            man[1][b].append(0)

for a in range(2):
    for b in range(24):
        if(len(man[a][b])!=0):
            variance[a+3][b]=math.sqrt(1/len(man[a][b]))*sum(m
ap(lambda x: math.pow(x-mean[b][a+3],2),(man[a][b])))

Prob=[ [ 0 for i in range(5) ] for j in range(24) ]

for x in range(24):
    for y in range(5):
        if mean[x][y]==0:
            Prob[x][y] = 0
        elif mean[x][y]==1 and variance[y][x]==0:
            Prob[x][y] = 1
        elif variance[y][x]==0:
            Prob[x][y] = 0
        else:
            Prob[x][y] =
((1/(variance[y][x]*math.sqrt(2*math.pi)))*math.exp(-math.pow(0-
mean[x][y],2)/(2*math.pow(variance[y][x],2))))#Gaussian naïve Bayes
            for b in range(24):
                if self.d.A1=="A%d" %(b+1):
                    Prob[x][y]=((1/(variance[y][x]*math.sqrt(2
*math.pi)))*math.exp(-math.pow(1-
mean[x][y],2)/(2*math.pow(variance[y][x],2))))#Gaussian naïve Bayes
            return Prob

```

Appendix D – Terminal de resultados das previsões da IA

A informação que consta do registo de dados indica, primeiramente encontra-se o estado de remate, seguido da posição anterior da bola, movimento do eixo, posição do eixo e posição atual da bola. O eixo usado neste estudo foi um dos guarda-redes, que inclui sete sensores de posição. Os movimentos do eixo estão representados por:

- esq, para quando se move para a esquerda;
- cen, para quando não se move;
- dir, para quando se move para a direita.
- nchut, para quando não está a rematar;
- chut, para quando está a rematar.

Os valores ‘nada’ indicam, no primeiro caso, que não há estado anterior, e, no segundo, que não tomou nenhuma decisão, pois não tinha dados suficientes. Definiu-se uma relação de 80-20 entre os dados de aprendizagem e os dados de teste. Optou-se por não usar extensões com o algoritmo, pois seria de maior interesse implementá-lo de raiz, para que fosse mais fácil efetuar adaptações ao mesmo. Ao executar o programa, a figura seguinte é o resultado das percentagens obtidas no terminal.

```
Dados para aprendizagem:
10240
Dados a descobrir
2560
Previsões Corretas Movimento
2547
Percentagem Previsões Corretas Movimento
99.4921875
Previsões Corretas Remate
2476
Percentagem Previsões Corretas Remate
96.71875
Previsões Totalmente Corretas
2463
Percentagem Previsões Totalmente Corretas
96.2109375
Tempo de execução
583.8551898002625
```