

Programação de robôs e consumo energético

Rui Manuel Soares Carvalho

Dissertação de Mestrado

Orientador: Professor Paulo Abreu

Coorientador: Professor Manuel Romano Barbosa



Mestrado Integrado em Engenharia Mecânica

Especialização em Automação

Setembro 2021

Resumo

Atualmente, a problemática do consumo energético é um tema cada vez mais recorrente nos processos de fabrico da indústria. Devido aos elevados preços da energia, alguns processos industriais têm sido otimizados quer em termos do tempo do próprio processo, quer em termos do consumo de energia que lhe está associado. No campo dos robôs industriais, a preocupação pelo consumo energético ainda está numa fase inicial, não existindo muitas soluções para otimizar o trabalho destes equipamentos.

Nesta dissertação são analisados alguns parâmetros de programação das trajetórias de robôs que podem estar associados ao consumo energético do seu movimento, sendo estes, a velocidade de execução, a aceleração do movimento, o tipo de interpolação do movimento, a zona de aproximação ao ponto (*zone*) e o local do espaço de trabalho. Assim, procura-se avaliar as implicações desses parâmetros no tempo de execução e no consumo energético de determinados movimentos. O robô IRB2400, existente no laboratório de robótica do DEMec foi utilizado no estudo ao nível experimental. O *software* de programação *offline* RobotStudio foi utilizado na análise ao nível da simulação. Os resultados de consumo energético, ao nível experimental, implicaram a aquisição de um equipamento de medição de energia elétrica e o desenvolvimento e implementação de um sistema de recolha e monitorização de dados via web. Na simulação, o *software* RobotStudio dispõe de uma funcionalidade, o *Signal Analyzer*, onde se avalia, entre outros parâmetros, o consumo energético. Os tempos de execução da trajetória são registados através da consola virtual, na simulação, e da consola do robô, ao nível experimental.

O estudo envolveu a definição de dez percursos de teste, para avaliar o impacto dos diferentes parâmetros de programação no tempo de ciclo e no consumo energético do movimento. Os resultados foram analisados, verificando-se que, os valores obtidos não podem ser dissociados dos percursos executados, embora seja evidente o impacto dos diferentes parâmetros de programação de trajetória estudados, no tempo de ciclo e no consumo energético robô.

O consumo energético em robôs industriais é um campo de estudo em crescimento, verificando-se que a parametrização da trajetória é um ponto fulcral para a aumentar a eficiência energética.

Abstract

Currently, the issue of energy consumption is an increasingly theme in the industry's manufacturing processes. Due to high energy prices, some industrial processes have been optimized both in terms of the time of the process itself and in terms of the energy consumption associated with it. In the field of industrial robots, the concern for energy consumption is still at an early stage, and there are not many solutions to optimize these equipments.

In this study, some programming parameters of robot trajectories that may be associated with the energy consumption of their movement are analyzed, namely, speed, acceleration, the type of movement, the point approach zone and the location on the workspace. Therefore, we seek to assess the implications of these parameters on the execution time and energy consumption of certain movements. The robot IRB2400, existing in the robotics laboratory of DEMec, was used in the study at the experimental level. The offline programming software, RobotStudio, was used to obtain the results at the simulation level, of both time and energy consumption. The results of energy consumption, at the experimental level, involved the acquisition of an electrical energy measurement equipment and the development and implementation of a web-based data collection and monitoring system.

The study involved the definition of ten test paths to assess the impact of different programming parameters on the cycle time and energy consumption of the movement. The results were analyzed, verifying that the obtained values cannot be dissociated from the executed paths, although the impact of the different trajectory programming parameters studied is evident, on the cycle time and on the robot energy consumption.

Energy consumption in industrial robots is a growing field of study and the parameterization of the trajectory is a key point to increase their energy efficiency.

Agradecimentos

Começo por expressar o meu profundo agradecimento aos meus orientadores, Professor Paulo Abreu e Professor Manuel Romano Barbosa, por todo o apoio, disponibilidade e dedicação ao longo deste trabalho, sem os quais não teria sido possível realizar.

À minha família, namorada e amigos, deixo também o meu obrigado, por toda a ajuda, carinho e preocupação que sempre demonstraram durante este longo percurso.

Índice de Conteúdos

1	Introdução	1
1.1	Enquadramento do projeto e motivação	1
1.2	Objetivos	2
1.3	Método seguido no projeto	3
1.4	Estrutura da dissertação	4
2	Otimização energética em robôs industriais	5
2.1	Evolução da procura de robôs industriais e suas principais aplicações	5
2.2	Contextualização da eficiência energética em robôs industriais	7
2.3	Programação <i>offline</i> de robôs industriais	10
2.4	Medição de energia elétrica	11
3	Análise energética do robô IRB2400	15
3.1	Programação e estudo energético em RobotStudio	16
3.1.1	Análise de instruções a utilizar no estudo energético	16
3.1.2	Descrição das capacidades do <i>Signal Analyzer</i>	19
3.2	Equipamento de medição de energia elétrica e sistema de recolha automática de dados	20
3.2.1	Instalação e configuração do PM5320	20
3.2.2	Validação do medidor de potência	22
3.2.3	Sistema de recolha automática de dados	22
4	Apresentação dos percursos de teste e exposição dos resultados	27
4.1	Testes de validação das funcionalidades do PM5320 e do <i>Signal Analyzer</i>	28
4.1.1	Medição do consumo energético com o PM5320	28
4.1.2	Impacto da massa nos resultados do <i>Signal Analyzer</i>	30
4.2	Percursos de teste e resultados obtidos	33
4.2.1	Consumo em <i>standby</i>	34
4.2.2	Movimento junta a junta	35
4.2.3	Percursos de teste segundo o ISO-Cube	42
4.2.4	Testes de tarefas típicas do IRB2400 na indústria	50

5	Discussão dos resultados obtidos	59
5.1	Movimento angular da junta 1	59
5.2	Movimento angular da junta 3	62
5.3	Movimento da junta 1 e 3	65
5.4	Localização no espaço de trabalho	67
5.5	Diagonais interiores do cubo ISO	68
5.6	Tarefas de <i>pick and place</i>	72
5.7	Percursos de soldadura	75
6	Conclusões e Trabalhos Futuros	79
6.1	Conclusões	79
6.2	Trabalhos Futuros	81
	Referências	83
	Anexos	87
A	Programa RAPID da Interface da consola	87
B	Programa desenvolvido em Node-Red	89
C	Equipamentos de medição de energia elétrica estudados	90

Siglas

AC- *Alternating current*

CSV- *Comma-separated values*

CT- *Current transformer*

DC- *Direct current*

FP- Fator de potência

IoT- *Internet of things*

PC- *Personal computer*

TCP- *Tool center point*

Lista de Figuras

1.1	Diagrama de Gantt deste projeto.	3
2.1	Evolução de robôs industriais em operação na última década.	6
2.2	Robôs em operação por indústria de 2017 até 2019.	6
2.3	Robôs em laboração tendo em conta a sua aplicação, desde 2017 até 2019.	7
2.4	Fluxo energético nos diferentes componentes de um robô industrial .	8
3.1	Interface desenvolvida para a consola do robô.	18
3.2	Interface desenvolvida para a consola do robô.	19
3.3	Interface de utilização do <i>Signal Analyzer</i>	20
3.4	PM5320.	20
3.5	Circuito elétrico de implementação do PM5320.	21
3.6	Arquitetura física do sistema de aquisição de dados.	23
3.7	Arquitetura de desenvolvimento da recolha de dados no PC.	23
3.8	<i>Flow</i> desenvolvido para a aquisição de dados.	24
3.9	Interface de monitorização em tempo real do consumo do robô IRB2400.	25
3.10	Interface de monitorização do histórico de consumo.	26
4.1	Perfil de velocidades para a trajetória de aceleração e desaceleração da junta 1 do robô IRB2400.	29
4.2	Perfil de velocidades para a trajetória a velocidade constante da junta 1 do robô IRB2400.	29
4.3	Gráfico da potência ativa durante a realização dos testes do PM5320.	30
4.4	Percurso realizado nos testes de verificação do <i>Signal Analyzer</i>	31
4.5	Resultados do consumo energético no movimento da junta 1 com duas cargas.	32
4.6	Resultados do consumo energético no movimento da junta 1 com diferentes modelos de robô.	33
4.7	Gráfico da potência ativa do robô quando imobilizado na posição "home".	34

4.8	Fluxo de interfaces no início de cada teste.	36
4.9	Código RAPID para a realização do percurso de movimento da junta 1.	37
4.10	Percurso (linha vermelha) programado para o movimento da junta 1.	37
4.11	Gráfico em Matlab, com as medições de Potência e Energia para o movimento angular da junta 1.	38
4.12	Resultados obtidos na ferramenta <i>Signal Analyzer</i> para o movimento da junta 1.	39
4.13	Percurso (linha vermelha) programado para o movimento da junta 3.	40
4.14	Percurso (linha vermelha) executado para o movimento conjugado da junta 1 e 3.	41
4.15	Teste do cubo ISO	43
4.16	IRB2400 na primeira orientação do <i>target</i> E ₄	44
4.17	IRB2400 na segunda orientação do <i>target</i> E ₄	44
4.18	Percurso (linha vermelha) realizado para a primeira diagonal através do movimento de juntas do robô.	46
4.19	Percurso realizado (linha vermelha) para a primeira diagonal através do movimento linear do robô.	47
4.20	Percurso (linha vermelha) do robô ao realizar a segunda diagonal através do movimento de juntas do robô.	48
4.21	Percurso (linha vermelha) do robô ao realizar a segunda diagonal através do movimento de juntas do robô.	49
4.22	Trajeto standard de <i>pick and place</i> definido pela Epson para os seus robôs.	50
4.23	Trajetória (linha vermelha) realizada pelo robô ao executar o percurso <i>standard</i> da Epson por movimento de juntas.	51
4.24	Trajetória (linha vermelha) realizada pelo robô ao executar o percurso <i>standard</i> da Epson por movimento linear.	51
4.25	Segundo percurso para tarefas de <i>Pick and Place</i>	53
4.26	Trajetória (linha vermelha) realizada pelo robô para o segundo percurso de tarefas <i>Pick and Place</i> por movimento de juntas.	53
4.27	Trajetória (linha vermelha) realizada pelo robô para o segundo percurso de tarefas <i>Pick and Place</i> por movimento de linear.	53
4.28	Trajetória de simulação de uma soldadura na vertical descendente.	55
4.29	Trajetória de simulação de uma soldadura no plano horizontal.	56
5.1	Tempo de ciclo médio em função da velocidade de execução para o movimento da junta 1.	60
5.2	Consumo total de energia em função da velocidade de execução para o movimento da junta 1.	60

5.3	Resultados do tempo de ciclo no movimento da junta 1 com diferentes velocidades e acelerações.	61
5.4	Resultados do consumo energético do percurso horizontal com diferentes velocidades e acelerações.	61
5.5	Tempo de ciclo em função da velocidade de execução para o movimento angular da junta 3.	63
5.6	Consumo total de energia em função da velocidade de execução para o movimento da junta 3.	63
5.7	Comparação dos diferentes parâmetros de aceleração no tempo de ciclo na execução percurso vertical	64
5.8	Comparação dos diferentes parâmetros de aceleração no consumo energético na execução percurso vertical.	65
5.9	Comparação dos diferentes parâmetros de aceleração no tempo de ciclo na execução percurso da junta 1 e 3.	66
5.10	Comparação dos diferentes parâmetros de aceleração no consumo energético na execução do percurso da junta 1 e 3.	66
5.11	Comparação do consumo energético nos diferentes vértices do cubo ISO.	68
5.12	Análise custo/benefício dos resultados obtidos para o percurso da diagonal 1 na célula do laboratório.	69
5.13	Tempo de ciclo e consumo energético das trajetórias da diagonal 1 realizadas no simulador.	70
5.14	Análise dos resultados das diferentes trajetórias da diagonal 2 na célula robótica.	70
5.15	Análise dos resultados das diferentes trajetórias da diagonal 2 no simulador do RobotStudio.	71
5.16	Análise dos resultados das diferentes trajetórias do primeiro percurso de <i>pick and place</i>	72
5.17	Perfil de velocidades (MoveJ) com dois parâmetros <i>zone</i> diferentes. .	73
5.18	Posição do TCP para as duas trajetórias (MoveJ e MoveL) com diferentes parâmetros <i>zone</i> (<i>fine</i> e <i>z10</i>).	74
5.19	Resultados de tempo de ciclo e consumo de energia para segundo percurso de testes de <i>pick and place</i>	74
5.20	Tempo de ciclo das diferentes trajetórias no percurso de soldadura na vertical.	75
5.21	Tempo de ciclo das diferentes trajetórias no percurso de soldadura na horizontal.	75

5.22 Consumo energético das diferentes trajetórias no percurso de solda- dura na vertical.	76
5.23 Consumo energético das diferentes trajetórias no percurso de solda- dura na horizontal.	77

Lista de Tabelas

2.1	<i>Software</i> de programação <i>offline</i> de diferentes fabricantes.	11
2.2	Especificações dos equipamentos de medição de energia elétrica dos diferentes fabricantes analisados	12
3.1	Características do IRB2400/16kg da ABB.	15
3.2	Instruções utilizadas na programação dos percursos de teste.	17
4.1	Resultados dos testes de validação do PM5320.	30
4.2	Resultados obtidos para o consumo de energia e tempo de execução no movimento angular da junta 1 nos 3 ciclos.	39
4.3	Resultados obtidos para o consumo energético e tempo de execução no movimento angular da junta 3 nos 3 ciclos.	40
4.4	Resultados obtidos para o consumo de energia e tempo de execução no movimento simultâneo da junta 1 e 3 nos 3 ciclos.	42
4.5	Consumo com o robô parado em locais distintos no espaço de trabalho.	45
4.6	Resultados obtidos para o consumo de energia e tempo de execução na execução da diagonal 1 por movimento de juntas nos 3 ciclos.	46
4.7	Resultados obtidos para o consumo de energia e tempo de execução na execução da diagonal 1 por movimento linear nos 3 ciclos.	47
4.8	Resultados obtidos para o consumo energético e tempo de execução na execução da diagonal 2 por movimento de juntas nos 3 ciclos.	48
4.9	Resultados obtidos para o consumo de energia e tempo de execução na execução da diagonal 2 por movimento linear nos 3 ciclos.	49
4.10	Resultados obtidos para o percurso 300x25cm de <i>pick and place</i> com um movimento de juntas nos 6 ciclos.	52
4.11	Resultados obtidos para o percurso 300x25cm de <i>pick and place</i> através de um movimento linear nos 6 ciclos.	52
4.12	Resultados obtidos para o percurso 600x50cm de <i>pick and place</i> com um movimento de juntas nos 6 ciclos.	54

4.13	Resultados obtidos para o percurso 600x50cm de <i>pick and place</i> com um movimento linear nos 6 ciclos.	54
4.14	Resultados obtidos para o percurso de soldadura num trajeto vertical descendente nos 3 ciclos.	56
4.15	Resultados obtidos para o percurso de soldadura num plano horizontal nos 3 ciclos.	57
5.1	Análise da variação do tempo de ciclo e do consumo de energia das trajetórias com parâmetros de aceleração mais baixos no movimento da junta 1, em relação à aceleração máxima, no sistema real.	62
5.2	Análise da variação do tempo de ciclo e do consumo de energia das trajetórias com parâmetros de aceleração mais baixos no movimento da junta 3, em relação à aceleração máxima, no sistema real.	64
5.3	Análise da variação do tempo de ciclo e do consumo de energia das trajetórias com parâmetros de aceleração mais baixos no movimento da junta 1 e 3, em relação à aceleração máxima, no sistema real.	67

Capítulo 1

Introdução

Neste primeiro capítulo é apresentado o tema a desenvolver, incluindo os principais objetivos a alcançar nesta dissertação. Inicialmente é feito um enquadramento do projeto, por forma a caracterizar o problema em causa, e a sua relevância no contexto das aplicações industriais onde se insere. É apresentada, ainda, uma descrição da estrutura da dissertação e uma síntese do que cada capítulo aborda.

1.1 Enquadramento do projeto e motivação

O robô industrial é o componente principal de muitos sistemas automáticos implementados na indústria. O seu campo de aplicações é vasto, podendo ser utilizado em tarefas de soldadura, *pick and place*, pintura, paletização, montagem, entre outras [1]. A flexibilidade da programação e a sua performance traduzida pela repetibilidade, precisão, velocidade e eficiência são as principais razões para esta escolha, permitindo diminuir o custo de operação e aumentar a produtividade [2]. Por este motivo, observa-se que, nas últimas décadas, a procura por este tipo de equipamentos de automação flexível tem crescido de uma forma considerável. Segundo dados da *International Federation of Robotics*, estima-se que no ano de 2019 estiveram em operação, cerca de 2.7 milhões de robôs pelo mundo [1]. Este desenvolvimento deve-se tanto à presença de novas indústrias a utilizar estes equipamentos, por exemplo o setor alimentar, como ao contínuo crescimento das células robóticas nas indústrias onde a sua representação já era considerável. O setor com maior utilização de robôs industriais é o automóvel, com cerca de 923 mil em operação em 2019 [1], estando envolvido em múltiplos processos como montagem, pintura, soldadura ou deposição de colas ou vedantes.

Na última década, o consumo de energia elétrica tem sido um tema cada vez mais abordado, principalmente devido ao impacto da produção desta no ambiente, mas também pelo contínuo aumento do preço da mesma. Desta forma, é necessário que os equipamentos sejam energeticamente otimizados, o que na indústria induz diretamente um menor custo de produção. Como tal, nos robôs industriais esta pro-

blemática também se tem levantado, principalmente porque em muitas ocasiões, a otimização da célula apenas tem em conta o tempo de ciclo, marginalizando-se a análise do consumo de energia. Por exemplo, para a produção de um automóvel de médio porte, estima-se que a energia elétrica despendida em todo esse processo seja de aproximadamente 3.5 MWh, o que equivale à média de energia consumida por ano numa habitação na Alemanha [3]. Como referido anteriormente, os robôs industriais são um dos principais componentes nas células deste setor, para o qual o estudo do seu consumo é vital, visto que representam cerca de 8% do consumo energético despendido na fase de produção de um automóvel [4].

Nesta perspetiva, existem algumas abordagens que apresentam alterações ao nível do *hardware* da célula, quer na escolha do robô, quer na incorporação de componentes energeticamente mais eficientes [5]. Também ao nível do *software*, algumas soluções são apresentadas, como por exemplo, a otimização da trajetória por modelação matemática da mesma, incluindo o parâmetro energético [6]. Apesar disto, muitos dos métodos de poupança energética em células robotizadas recaem, ou em mudanças na planta existente, ou numa redefinição dos percursos, o que, realisticamente, só é possível de ser utilizado no momento de projeto da célula [7].

Assim, na presente dissertação propõe-se uma estratégia alternativa, onde se pretende identificar o impacto da programação de trajetórias no consumo energético e no tempo de execução das mesmas. Como tal, a parametrização das diferentes trajetórias em estudo assenta nos seguintes aspetos:

- Velocidade de execução;
- Aceleração dos movimentos;
- Definição de diferentes zonas de aproximação ao ponto;
- Local no espaço de trabalho onde se executa a trajetória;
- Tipo de interpolação da trajetória, por movimento de juntas ou linear.

1.2 Objetivos

O objetivo principal deste trabalho consiste em avaliar o impacto da parametrização da trajetória no consumo energético de um robô industrial. Para tal, tendo em conta o robô existente (IRB2400) e as ferramentas disponíveis no *software* de programação do mesmo, RobotStudio, foram definidas as seguintes metas:

- Implementação de um sistema de medição de energia elétrica do controlador do robô e um sistema automático de aquisição e registo de dados;

- Validação da ferramenta de simulação de análise energética, *Signal Analyzer*, disponível no *software* RobotStudio;
- Identificação do impacto das instruções de programação no consumo energético do robô e no tempo de execução das trajetórias.

1.3 Método seguido no projeto

Para a execução deste projeto foram definidas as seguintes etapas:

- Revisão do estado da arte sobre o consumo energético em robôs industriais;
- Seleção do instrumento de medição de energia elétrica;
- Familiarização com o *software* de programação *offline*;
- Definição do plano de testes e trajetórias a realizar;
- Implementação do sistema de medição e recolha automática de dados;
- Realização de testes de simulação em programação offline;
- Implementação dos testes no manipulador IRB2400;
- Escrita da dissertação.

A calendarização e duração de cada etapa está espelhada no diagrama de Gantt da figura 1.1.

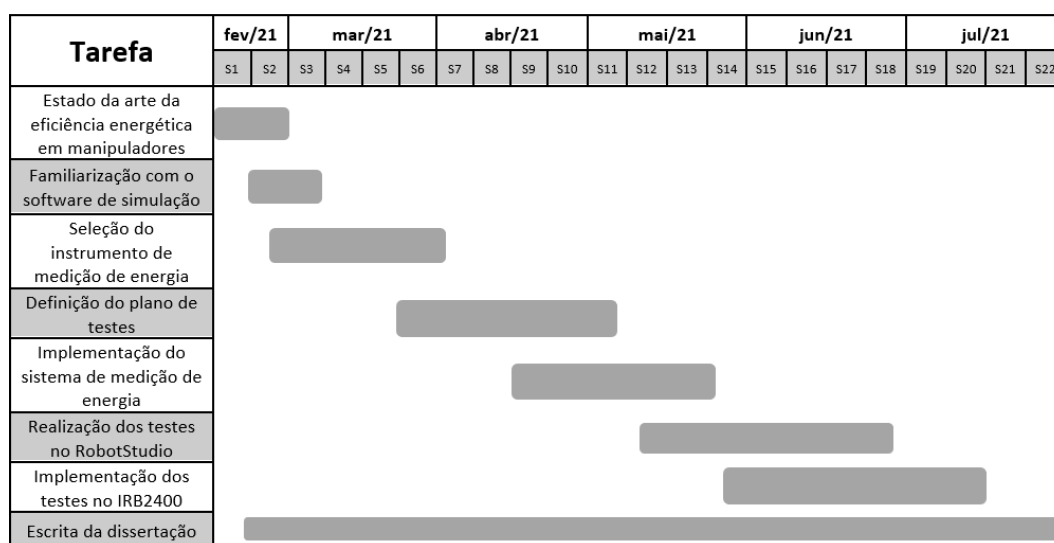


Figura 1.1: Diagrama de Gantt deste projeto.

1.4 Estrutura da dissertação

A dissertação encontra-se organizada, para além do presente capítulo, da seguinte forma:

- Capítulo 2 – Introdução ao problema da otimização energética em robôs industriais e apresentação das principais aplicações onde este é um fator preponderante. Exposição do estado da arte dos métodos de redução do consumo energético. Breve descrição dos diferentes *software* de programação *offline* de células robóticas e identificação das funcionalidades de otimização energética. Caracterização dos requisitos necessários para o sistema de medição a implementar e avaliação de várias soluções;
- Capítulo 3 – Análise do *software* utilizado para avaliação energética em regime *offline* e descrição dos parâmetros de programação de robôs a serem analisados. Implementação do sistema de medição e recolha automática de dados;
- Capítulo 4 – Definição e implementação da estratégia de testes a realizar, tendo em conta aplicações específicas dos robôs industriais. Apresentação dos resultados obtidos para cada uma das trajetórias realizadas;
- Capítulo 5 – Análise dos resultados do consumo energético e do tempo de execução obtidos para os diferentes percursos e parametrizações das trajetórias. Discussão do impacto de cada um dos parâmetros de programação na eficiência energética e no tempo de execução da trajetória.
- Capítulo 6 – Conclusões do trabalho realizado e identificação de trabalhos futuros.

Capítulo 2

Otimização energética em robôs industriais

Nas últimas décadas, devido ao gradual aumento dos preços da energia e à crescente preocupação com o impacto ambiental na sua produção, a eficiência energética é uma necessidade a ter em consideração. Por exemplo, a União Europeia estabeleceu objetivos de poupança de energia [8], entre outras iniciativas, e a pressão de grupos ecológicos, levou a que os diferentes setores industriais procurassem uma maior eficiência energética nos seus processos. Na realidade, apesar de alguns processos de alto consumo energético (químicos, aquecimento, etc) serem de elevada eficiência, isto não se verifica para muitos dos processos de fabrico realizados na indústria. Desta forma, ao analisar-se uma parte destes processos, verifica-se que a utilização de robôs industriais está presente em elevada percentagem e por isso, a análise energética destes equipamentos é um aspeto primordial a ter em conta [1].

Neste capítulo, é apresentado um estudo mais aprofundado sobre a evolução da utilização de robôs industriais no mundo.

2.1 Evolução da procura de robôs industriais e suas principais aplicações

Como referido anteriormente, os robôs industriais são parte integrante dos processos de muitas indústrias, tendo a sua utilização crescido de forma gradual na última década (figura 2.1).

A informação da figura 2.1 reporta um uso crescente dos robôs industriais, representando, assim, uma área com potencial para a adoção de estratégias de melhoria da eficiência energética. Desta forma, é importante perceber que tipos de indústrias se destacam na utilização destes equipamentos, para que a análise seja centralizada nos principais requisitos das aplicações mais utilizadas nestes setores.

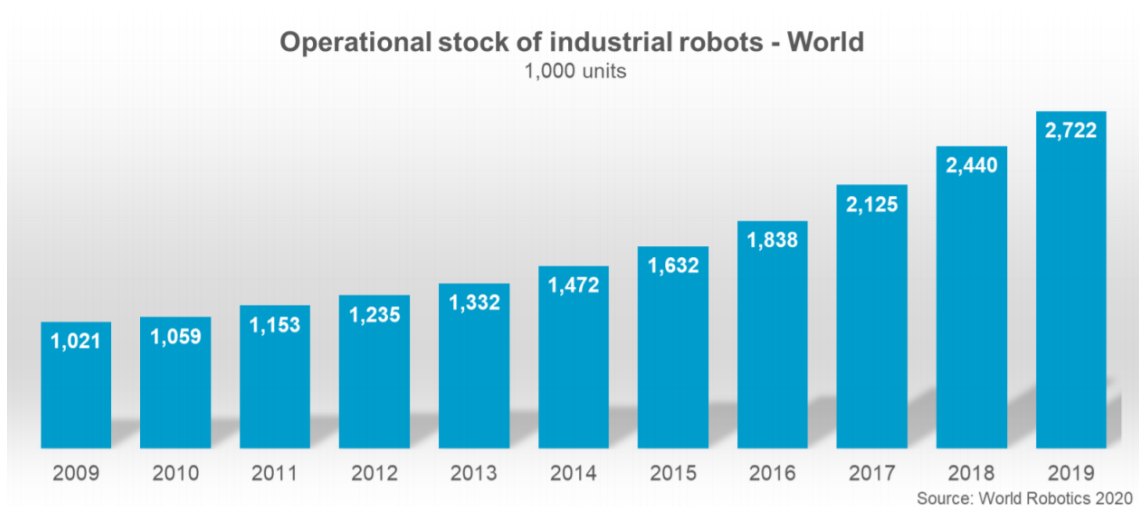


Figura 2.1: Evolução de robôs industriais em operação na última década [1].

Para tal, utilizaram-se novamente os dados divulgados pela International Federation of Robotics em 2020, que podem ser visualizados na figura 2.2.

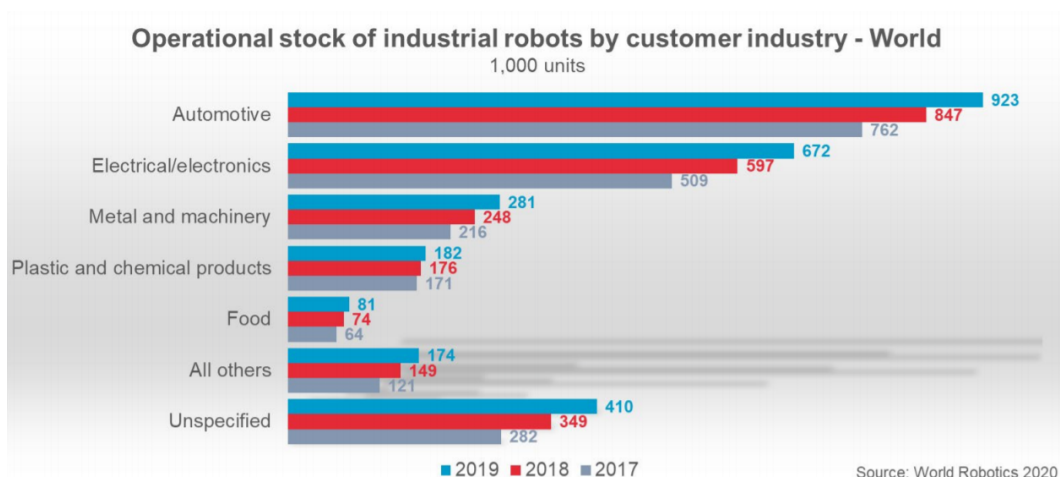


Figura 2.2: Robôs em operação por indústria de 2017 até 2019 [1].

Da figura 2.2, verifica-se que a indústria automóvel é a que apresenta mais peso na utilização de robôs industriais, com cerca de 34% dos robôs operacionais no mundo a serem utilizados neste setor em 2019. Em segundo lugar surge a indústria eletrónica, com uma representatividade de 25% de todos os robôs em operação.

Como forma de perceber as aplicações nas quais as funcionalidades dos robôs industriais são empregues nas diferentes indústrias, na figura 2.3 é apresentado o número de robôs em operação por aplicação.

Ao interpretar a figura 2.3, depreende-se que, como esperado, as aplicações

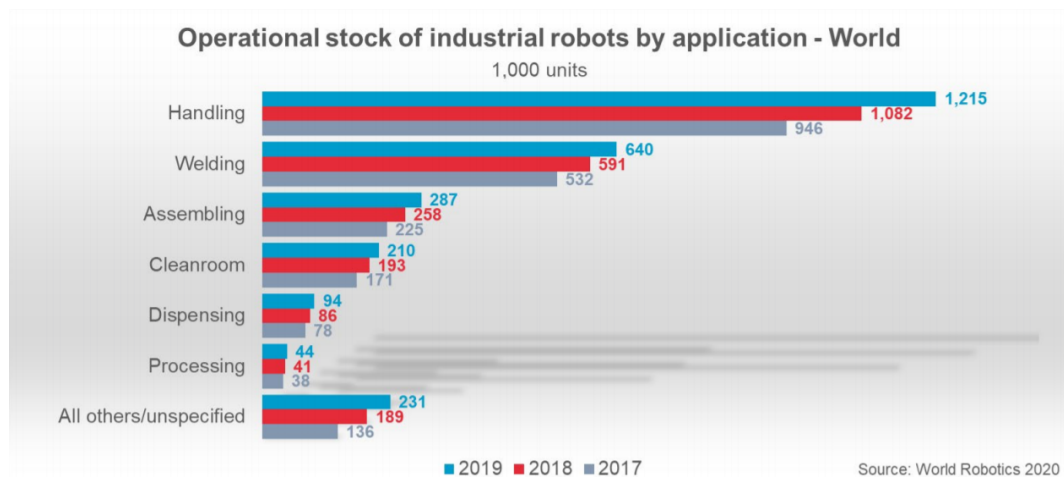


Figura 2.3: Robôs em laboração tendo em conta a sua aplicação, desde 2017 até 2019 [1].

como manuseamento, soldadura e montagem encontram-se no topo das mais utilizadas, visto serem as mais executadas na indústria automóvel e na de produtos eletrónicos. Dentro das aplicações de manuseamento incluem-se processos de paletização, embalagem, *pick and place*, medição, inspeção e teste, bem como, estampagem, forjamento e conformação plástica, entre outros processos, o que explica o facto de, por larga margem, ser o manuseamento a aplicação com maior relevo.

Com estes dados, e analisando a sua utilidade na indústria automóvel, é possível identificar quais as aplicações mais presentes nesta área e o porquê de esta ser a mais representativa na utilização de robôs. Soldadura por pontos e por arco elétrico, transferência de elementos do automóvel por ações de *pick and place*, pintura, selagem e revestimento, maquinagem e a montagem do conjunto são ações complexas e vitais na produção de um automóvel e a utilização das potencialidades dos robôs industriais favorecem, em muito, a otimização do processo. Assim, devido à elevada fiabilidade e repetibilidade, compreende-se que as linhas de produção de automóveis sejam constituídas por células robóticas com um elevado número de robôs. Desta forma, o consumo energético destas células é um ponto de preocupação de muitos fabricantes de automóveis.

2.2 Contextualização da eficiência energética em robôs industriais

Como referido na secção anterior, a problemática da eficiência energética em robôs industriais é um tema cada vez mais recorrente no setor industrial. Nesta perspetiva, é relevante analisar a estrutura física do robô industrial, com o intuito de identificar os seus componentes e perceber de que forma estes se relacionam com

o consumo energético do mesmo. Deste ponto de vista, para uma aplicação genérica, um robô industrial pode-se dividir da seguinte forma:[9].

- **Sistema de controlo:** componentes que se encontram integrados no armário do controlador, como por exemplo, drives dos motores, conversores AC/DC, inversores, entre outros.
- **Sistemas auxiliares:** componentes de apoio ao sistema de controlo, que inclui a consola (*teach pendant*), os sistemas de refrigeração do controlador e o PC para comunicação com o sistema de controlo;
- **Atuadores:** incorpora todos os motores de cada junta do robô e é subsequente ao sistema de controlo do mesmo.

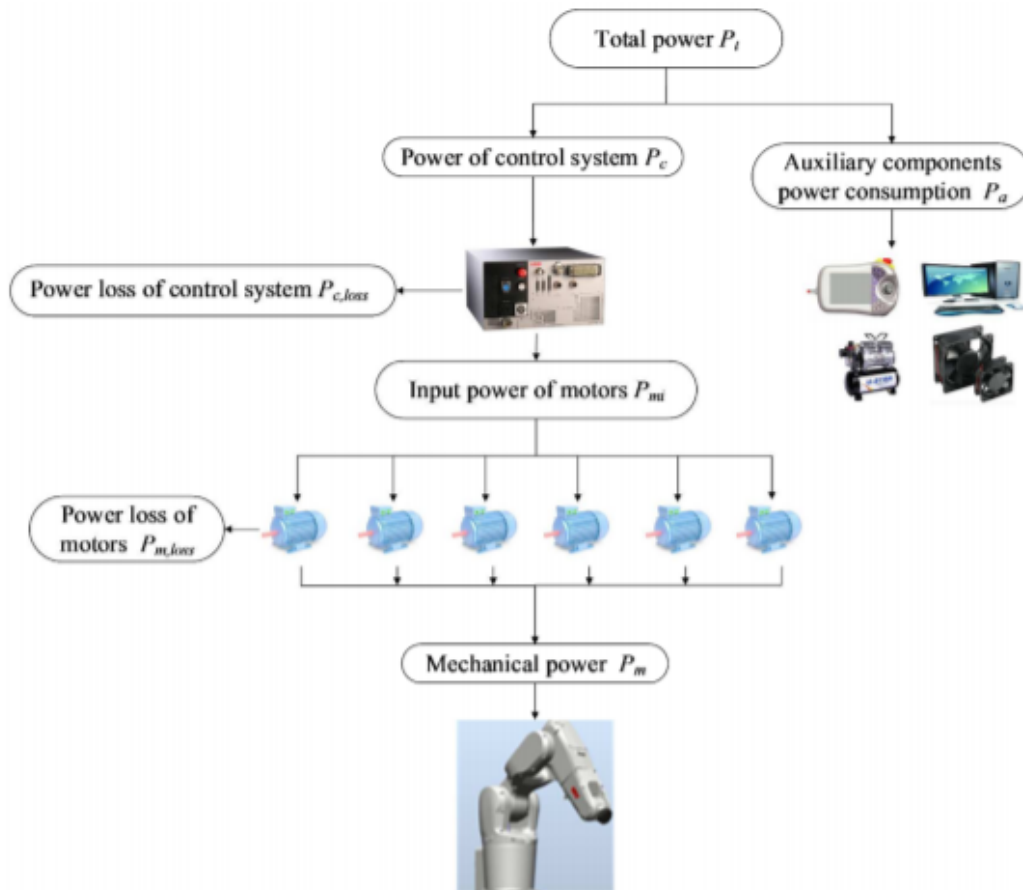


Figura 2.4: Fluxo energético nos diferentes componentes de um robô industrial [9].

Com base na figura 2.4, e na informação presente em [9], verifica-se que o consumo energético por parte dos sistemas auxiliares é, essencialmente constante, e que as principais perdas se dão tanto no sistema de controlo como na motorização dos eixos do robô. Em relação à motorização, nos robôs industriais são maioritariamente

utilizados servomotores. Estes tipos de motores apresentam alto rendimento (entre 83% e 92% [9]), embora uma eficiência superior possa ser atingida com alterações no projeto dos mesmos. A adoção de um núcleo magnético de baixas perdas ou o aumento da secção transversal da bobina, são algumas das estratégias apresentadas para esse efeito [10].

Para além disso, nos momentos de travagem, os motores elétricos apresentam a capacidade de funcionar como geradores de energia. A adoção de sistemas que permitam a aquisição e armazenamento dessa energia, apresenta-se como uma excelente opção para aumentar a eficiência energética em equipamentos como os robôs industriais. A implementação de, por exemplo, condensadores ligados a um barramento DC, permite que, um eixo do robô em desaceleração, gere e armazene energia para, posteriormente ser utilizada por outros eixos em movimento [5].

Por outro lado, as massas a movimentar representam um aspeto a considerar. Nesta perspetiva, as soluções podem passar por alterações nos materiais que integram os manipuladores, com a adoção de componentes mais leves. Esta opção permite uma diminuição na massa e inércia global do robô, induzindo um controlo do manipulador mais facilitado, tal como, uma redução na energia consumida pelo equipamento [5].

Apesar disto, alterações na estrutura do robô nem sempre são possíveis, e cabem sempre aos seus fabricantes a sua execução. Como tal, a análise das aplicações e trajetórias nas quais são empregues os robôs industriais, constituem uma perspetiva que permite ao utilizador aumentar a eficiência energética do equipamento. Neste sentido, no momento do projeto de uma célula robótica, conhecendo as tarefas e trajetórias que se pretendem realizar, uma correta seleção da configuração física do robô pode aumentar a eficiência energética da célula projetada. Por exemplo, em determinadas trajetórias horizontais, robôs de configuração paralela apresentam consumos de energia significativamente inferiores a um homólogo com configuração em série, como apresentado em [11]. Outra possibilidade passaria pela adoção de robôs redundantes para a trajetória pretendida. Como apresentam mais sistemas de atuação do que o necessário, para o mesmo movimento existem várias combinações de eixos a utilizar, possibilitando a escolha do que induz menor binário, e por conseguinte menor energia consumida [12].

Para além das soluções em que se aborda a seleção do robô com a aplicação em que é utilizado, alguns estudos indicam que é possível relacionar a trajetória executada com a energia consumida. Através de ferramentas computacionais pode-se desenvolver um modelo matemático do espaço de trabalho do robô e das trajetórias a executar, introduzindo uma componente energética. Desta forma, pela análise matemática desse modelo, obtém-se as equações mecânicas e eletro-mecânicas do

movimento do robô, em função do parâmetro energético. Esta estratégia permite a simulação, para qualquer trajetória, do consumo energético do robô, permitindo selecionar a trajetória que apresenta maior eficiência energética [13].

Na maioria das células robóticas utilizadas na indústria, o robô está dependente do trabalho de outros equipamentos. Desta forma, a sincronização dos movimentos dos diferentes equipamentos é essencial para o aumento da eficiência energética. No trabalho de robôs industriais, denota-se que estes são programados para trabalhar à maior velocidade possível, o que por vezes pode provocar tempos de inatividade elevados até que a próxima operação seja executada. Neste sentido, a adoção de um planeamento de operações em que se introduz uma redução, ao máximo admissível pelo processo, da velocidade de execução do robô, conduz a menor consumo de energia (a aceleração diminui, induzindo menores picos de corrente). Apesar do tempo de operação do robô aumentar, caso esteja sincronizado com os restantes equipamentos, o tempo de ciclo mantém-se, pois apenas se reduz o tempo de inatividade do robô [14]. Além desta solução, a atuação dos travões do robô nos momentos de inatividade é outra das possibilidades para diminuir o consumo de energia nestas situações [15].

Ao longo da revisão da literatura existente, muitos dos casos apresentados foram realizados recorrendo a simulações em modelos virtuais do robô e da célula em estudo. Para este efeito, alguns ambientes de programação *offline* disponibilizados pelos fabricantes de robôs industriais, permitem o desenvolvimento de uma célula virtual aproximada à existente na realidade. Como tal, o estudo das diferenças entre as aplicações de cada um dos fabricantes, é importante para o presente estudo, como forma de identificar se existe a funcionalidade de análise energética do robô.

2.3 Programação *offline* de robôs industriais

Alguns fabricantes de robôs industriais desenvolvem ambientes virtuais para a programação *offline* dos seus robôs. Estes simuladores permitem, o desenvolvimento de um modelo 3D idêntico à célula robótica existente, definir e simular os percursos pretendidos e, posteriormente, transferir o respetivo programa para o controlador do robô para serem executados. Em [16], expõe-se que no caso do fabricante de automóveis Chrysler, o tempo de programação dos seus robôs industriais reduz-se de 12 a 18 horas para 6.5 horas ao utilizar estas aplicações. Outros benefícios destes programas, é um aumento da segurança do operador, pela preditibilidade da trajetória, e uma mais rápida instalação de células robóticas, por análise de um modelo previamente desenvolvido [16].

Ao contrário de outros equipamentos, como por exemplo, as máquinas CNC

e respetiva linguagem de programação normalizada (ex: código G) , no âmbito da robótica industrial, não existe uma linguagem *standard*, isto é, cada fabricante tem uma linguagem própria para os seus robôs, colocando assim algumas dificuldades aos utilizadores que pretendam manter robôs de diferentes fabricantes. Na tabela 2.1, são apresentados quatro exemplos que se analisaram de fabricantes de robôs e as suas diferentes aplicações de programação *offline*.

Tabela 2.1: *Software* de programação *offline* de diferentes fabricantes.

Fabricante	<i>Software</i>	Linguagem
ABB	RobotStudio	RAPID
Fanuc	Robotguide	Karel
Kuka	KukaSim	KRL
Kawasaki	PC-Roset	AS

Na presente dissertação, os dois principais focos de utilização do *software* de programação *offline* são a programação da trajetória e a análise energética do robô ao executá-la. A linguagem de programação é o ponto fulcral para a definição e execução da trajetória do robô, mas não permite, em nenhum dos simuladores apresentados, obter informações sobre o consumo energético do robô. Como tal, o único *software* com uma ferramenta de análise do consumo energético é RobotStudio, da ABB, designada *Signal Analyzer*, que, entre outras capacidades, fornece o consumo energético total ao longo da execução de uma tarefa.

Na célula do laboratório de robótica, o robô existente é um IRB2400/16, da ABB, o que permite utilizar, neste trabalho, algumas informações adquiridas pelo simulador. Apesar disto, o controlador ao qual o robô está associado é de uma versão anterior à do lançamento do *Signal Analyzer*, o que não possibilita que esta ferramenta possa ser utilizada no estudo energético das tarefas executadas na célula, sendo apenas relevante nas simulações executados em *software*. Assim, é essencial perceber de que forma se pode medir a energia elétrica consumida na célula do laboratório.

2.4 Medição de energia elétrica

A medição do consumo de energia elétrica na célula robótica identificou-se como necessária para avaliar e comparar as diferentes estratégias de poupança energética em estudo. Para tal é necessário um equipamento que permita medir os consumos instantâneos e acumulados do robô durante as diferentes trajetórias programadas.

Uma das alternativas é o recurso a dispositivos que meçam a potência elé-

trica, a partir da medição da tensão e corrente em utilização, sendo designados por medidores de potência ("*power meter*"). Dado o elevado interesse na medição de energia elétrica nas mais variadas áreas, existem, atualmente, várias gamas destes dispositivos, desde os mais simples que medem a potência elétrica instantânea e a energia acumulada, até aos mais complexos que disponibilizam, por exemplo, uma análise à qualidade do sinal (medição de harmónicos), permitem a gerir alarmes e, por vezes, aceder à medição remotamente através de um WebServer.

Assim, devido à vasta variedade deste tipo de aparelhos, na sua seleção, deve-se definir as características essenciais para o caso em estudo. Na presente dissertação, os requisitos considerados são:

- Monitorização da tensão L-L, tensão L-N, corrente em cada linha e total, potência ativa, potência reativa, fator de potência e energia ativa fornecida;
- Frequência da rede compatível com a existente ($f=50$ Hz);
- Interface de ligação entre o equipamento e um computador para monitorização remota;
- Protocolo de comunicação que possibilite troca de dados equipamento-computador e que permita o desenvolvimento de um sistema de recolha de dados automático;
- Montagem em calha DIN;

Com estes requisitos como ponto de partida, seguiu-se a pesquisa por um equipamento que os cumprisse e que monetariamente fosse o mais viável. Desta forma, foram identificados vários equipamentos, apresentados na tabela 2.2 (versão ampliada no anexo C), juntamente com as suas características.

Fabricante		Schneider		Panasonic		Delta			Siemens		Carlo Gavazzi	
Produto		PM5320	PM5111	AKW91110	AKW264100A	DPM-C530	DPM-C530E	DPM-C520W	7KM3133-RTU	7KM3133-TCP/IP	WM40 96	WM30 96
Parâmetros medidos ⁽¹⁾		Todos	Todos	Todos	Todos	Todos	Todos	Todos	Todos	Todos	Todos	Todos
Tensão de alimentação		100...415 V AC 45...65 Hz	100...415 V AC 45...65 Hz	85 to 264V AC	100-240V AC	80 ~ 265 V AC	80 ~ 265 V AC	80 ~ 265 V AC	100-240 V AC	95-240 V AC	100-240 V AC	100-240 V AC
Frequência da rede		50 Hz 60 Hz	50 Hz 60 Hz	50 Hz 60 Hz	50 Hz 60 Hz	45 ~ 70 Hz	45 ~ 70 Hz	45 ~ 70 Hz	45-65 Hz	Deteção automática da frequência da rede	50/60 Hz	50/60 Hz
Protocolo de comunicação	ModBus RTU		✓	✓	✓	✓		✓	✓		✓	✓
	ModBus ASCII		✓			✓						
	ModBus TCP/IP	✓			✓		✓	✓		✓	Request	
	MEWTOTCOL			✓	✓							
	BACnet IP	✓									Request	
Porta de comunicação suportada	BACnet MS/TP					✓						
	RS232										✓	✓
	RS485		✓	✓	✓	✓		✓	✓		✓	
	Interface Wireless							✓				
	Ethernet	✓			✓		✓			✓	Request	Request
Tipo de montagem		Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN
Memória interna					✓							

(1)- Aparelho capaz de medir: Corrente, Tensão, Frequência, Potência Ativa, Potência Reativa, Potência Aparente, Fator de Potência, Energia Ativa e Energia Reativa

Tabela 2.2: Especificações dos equipamentos de medição de energia elétrica dos diferentes fabricantes analisados

Pela análise da Tabela 2.2, e tendo em vista o local onde seria implementado, verificou-se que seria de uma maior flexibilidade utilizar um equipamento com porta de comunicação Ethernet e protocolo de comunicação ModBus TCP/IP. Esta solução permite ligar o aparelho a um *switch* e, assim, ser acessível por parte de um qualquer dispositivo conectado à rede.

Após uma análise das características dos equipamentos estudados, concluiu-se que a melhor opção seria o medidor de potência da Schneider PM5320, que apresenta boas características quanto à resolução da medição e verifica todos os requisitos anteriormente definidos.

Capítulo 3

Análise energética do robô IRB2400

A análise do consumo de energia elétrica do robô da ABB (IRB2400/16), disponível no laboratório de Robótica do DEMec é o principal objetivo desta dissertação. Como tal, inicialmente é necessário conhecer este robô, e por isso na Tabela 3.1 estão apresentadas algumas das suas características mais relevantes para este caso de estudo.

Tabela 3.1: Características do IRB2400/16kg da ABB [17].

Modelo	Carga	Alcance	Nº de eixos	Fonte de alimentação	Consumo energético	Principais Aplicações
IRB 2400	16 kg	1,55 m	6	200-600 V 50/60 Hz	670W no teste ISO-Cube à velocidade máxima	Soldadura por arco elétrico; Corte; Polimento;
Juntas	Tipo	Área de Trabalho	Velocidade máxima		Potência nominal do Servomotor	
1	Rotativa	+180° até -180°	150°/s		933 W	
2	Rotativa	+110° até -100°	150°/s		1660W	
3	Rotativa	+65° até -60°	150°/s		933 W	
4	Rotativa	+200° até -200°	360°/s		380W	
5	Rotativa	+120° até -120°	360°/s		484W	
6	Rotativa	+400° até -400°	450°/s		380W	

Como referido anteriormente, para o desenvolvimento deste trabalho, a análise do consumo de energia elétrica é realizada a dois níveis, ao nível da simulação e ao nível experimental. Para a simulação utiliza-se o *software* de programação *offline* da ABB, RobotStudio, juntamente com a ferramenta *Signal Analyzer*, nele incluída. Do ponto vista experimental, faz-se uso da célula robótica do laboratório, onde está integrado o robô, e do equipamento de medição de energia, mencionado no Capítulo

2, PM5320, que permite monitorizar a energia elétrica fornecida ao controlador do robô.

3.1 Programação e estudo energético em RobotStudio

O RobotStudio é uma ferramenta de simulação e programação *offline* desenvolvida pela ABB para os seus robôs. Este *software* permite simular, em ambiente virtual, aquilo que acontece na realidade, devido à variada biblioteca de equipamentos da ABB disponível e à possibilidade de modelar ou importar outros dispositivos presentes na célula robótica. A elevada utilidade deste *software* é, ainda, demonstrada pela possibilidade de descarregar, diretamente para o controlador, os programas aí desenvolvidos, não sendo necessária uma fase de conversão. Assim, umas das grandes vantagens da utilização de robôs industriais, a sua capacidade de ajuste a tarefas diferentes quando as necessidades mudam, pode mais facilmente ser mantida, pois não implica uma longa paragem na produção para a reprogramação do robô [18].

Nesta secção, contextualiza-se a utilização do *software* RobotStudio, no âmbito do estudo do consumo energético em robôs industriais. Neste programa utilizaram-se as seguintes funcionalidades: programação em RAPID, com integração dos parâmetros com maior impacto no consumo energético, desenvolvimento de uma interface através da consola virtual FlexPendant e as funcionalidades de análise energética da ferramenta de simulação *Signal Analyzer*.

3.1.1 Análise de instruções a utilizar no estudo energético

No *software* de programação de robôs RobotStudio e nos controladores dos mesmos, a linguagem de programação utilizada é denominada de RAPID. É uma linguagem de alto nível onde as suas instruções, funções e dados permitem uma simples forma de conceção de movimentos de um robô.

Tendo em conta os parâmetros que podem afetar o consumo energético do IRB2400, como a velocidade de execução, aceleração ou a zona de aproximação, é necessário definir que tipo de instruções ou funções do RobotStudio, relacionadas com estes parâmetros, podem ter impacto no consumo energético e no tempo de ciclo da tarefa. Assim, na Tabela 3.2, apresentam-se as instruções de programação utilizadas para a execução dos diferentes percursos.

Tabela 3.2: Instruções utilizadas na programação dos percursos de teste.

Parâmetros	Instrução/ Função/ Dados	Especificação	Exemplo
Tipo de movimento	<i>MoveJ</i>	O robô e os seus eixos movem-se até uma posição, por um percurso não linear, sendo que todos os eixos chegam ao destino ao mesmo tempo	MoveJ Target50, v1000,z100, caneta\WObj:= wobjfolha
	<i>MoveL</i>	Movimento do TCP segundo uma trajetória linear	MoveL Target50, v1000,z100, caneta \WObj:=wobjfolha
Velocidade	<i>speeddata</i>	Especificação da velocidade do TCP, através da biblioteca do simulador.	v100
Zona de aproximação	<i>zonedata</i>	Define a aproximação a uma determinada posição, ou seja, a que distância da posição programada o robô deve começar instrução seguinte	<i>fine</i> - Paragem obrigatória no ponto;
			<i>z50</i> - Início da instrução seguinte a 50mm do ponto.
Aceleração	<i>AccSet</i>	Permite limitar o valor da aceleração/desaceleração utilizada em relação ao valor normal do modelo do robô em causa, bem como limitar a rampa de aceleração/desaceleração, isto é, aumentar o tempo que demora a atingir a aceleração especificada	<i>AccSet</i> 50,100
			<i>AccSet</i> 100,50

Para além das instruções associadas à definição da trajetória, utilizaram-se outras funcionalidades do RobotStudio com o objetivo de facilitar o processo de teste. Visto que a célula do laboratório de robótica tem, associada ao robô, uma consola de programação, fabricada pela ABB e designada por *FlexPendant*, o utilizador pode desenvolver interfaces de programação e execução de trajetórias, o que simplifica a automatização do processo de teste. No simulador do RobotStudio, existe também uma consola virtual, com as mesmas funcionalidades que a da célula real, garantindo que a execução dos testes seja idêntica tanto em simulador como no estudo experimental.

Assim, a interface é criada através de algumas instruções em código RAPID. Na presente dissertação, utilizaram-se as seguintes instruções para as interfaces de teste:

- ***TPReadFK***: utilizada para escrever texto nas zonas de teclado para identificar qual das opções foi selecionada;
- ***TPReadNum***: permite ler variáveis numéricas escritas pelo utilizador, como por exemplo, o número de vezes que o robô deve realizar um percurso;
- ***TPErase***: usada para limpar os campos de texto existentes na interface da consola;
- ***TPWrite***: escreve texto na interface da consola. Permite registar variáveis numéricas como, tempo de execução, velocidade ou aceleração;

Com estas instruções foram, assim, desenvolvidas as interfaces para os testes do estudo do consumo energético em robôs industriais, estando uma dessas interfaces apresentada na figura 3.1. No Anexo A pode-se consultar o programa desenvolvido para a conceção da mesma.

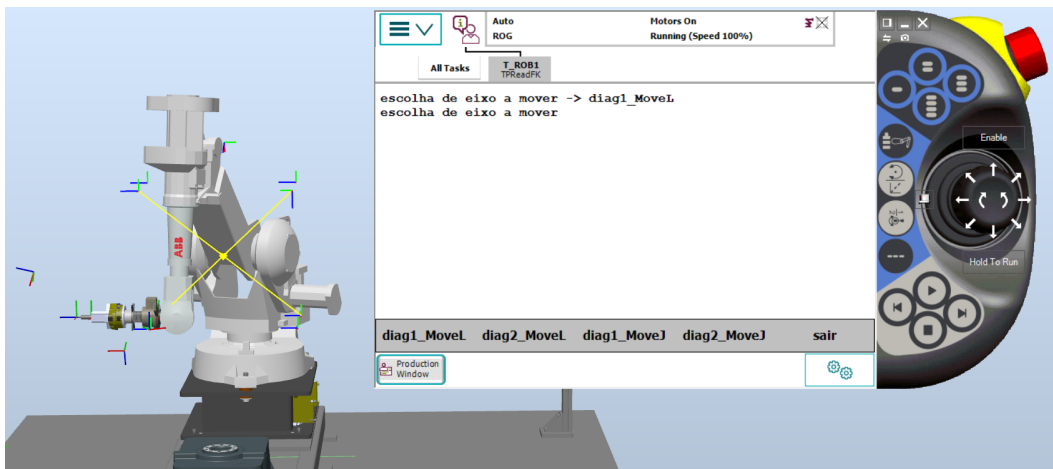


Figura 3.1: Interface desenvolvida para a consola do robô.

Para a contabilização do tempo de execução utilizaram-se as funções disponíveis para criação de contadores (*timers*) e as instruções de início, paragem e reinicialização ***ClkStart***, ***ClkStop***, ***ClkReset***, respetivamente. No final do teste, pela instrução ***TPWrite***, o tempo é apresentado na interface da consola, como demonstrado na figura 3.2.

O RobotStudio para além das instruções de programação e da ferramenta de consola virtual, tem, ainda, disponível a ferramenta do *Signal Analyzer*, que é essencial para o presente estudo, pois permite avaliar, de forma expedita, o consumo energético estimado pelo simulador para um determinado movimento do robô.

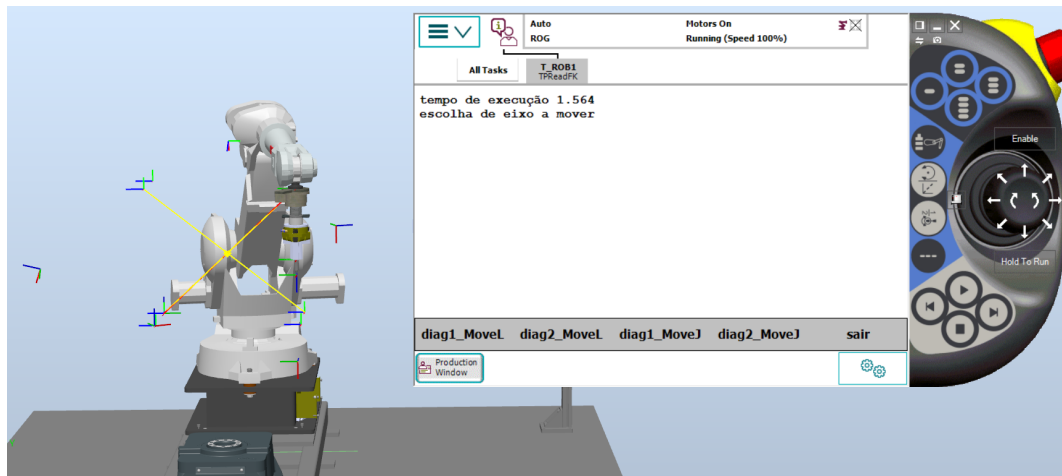


Figura 3.2: Interface desenvolvida para a consola do robô.

3.1.2 Descrição das capacidades do *Signal Analyzer*

O *Signal Analyzer* é uma ferramenta do RobotStudio que permite monitorizar e registar um conjunto alargado de sinais disponíveis no controlador do robô, nomeadamente, a posição do elemento terminal do robô, os valores de posição e velocidade de cada junta, a potência fornecida pelos motores e a energia total consumida. Os parâmetros do sinal que se pretendem analisar são seleccionados no Signal Setup e visualizados no *Signal Analyzer*. Assim, estas características do *software* permitem analisar a performance do robô.

Quanto à análise energética que é possível fazer pela utilização desta ferramenta, esta restringe-se a dois parâmetros, sendo estes:

- **Total Motor Power** – Refere-se à potência instantânea consumida pelos atuadores das juntas do braço robótico e não à que é fornecida ao controlador. Pode ser positiva ou negativa, dependendo se a junta se encontra em aceleração ou desaceleração, induzindo, desta forma, a disponibilidade de energia ao longo da execução do movimento. No caso do robô virtual, analisa-se o sinal em condições típicas de funcionamento, sendo a potência dependente do binário de cada junta.
- **Total Motor Energy** – Refere-se à energia consumida acumulada ao longo do movimento.

Todos estes parâmetros podem ser analisados em gráficos na interface do *Signal Analyzer* e podem ainda ser gravados e exportados para ficheiros Excel para posterior análise.

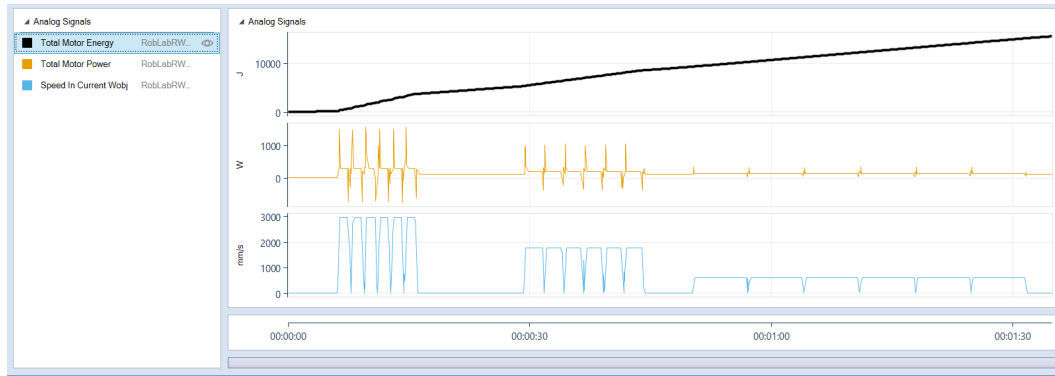


Figura 3.3: Interface de utilização do *Signal Analyzer*.

3.2 Equipamento de medição de energia elétrica e sistema de recolha automática de dados

Depois de adquirido o dispositivo de medição de energia elétrica (Schneider PM5320, apresentado na figura 3.4), foi necessário integrá-lo na célula robótica. Para tal, este processo divide-se em duas fases: configuração e instalação do PM5320 na célula robótica e desenvolvimento do sistema de aquisição de dados.



Figura 3.4: PM5320 [19].

3.2.1 Instalação e configuração do PM5320

A integração do PM5320 na célula robótica inicia-se pela instalação elétrica do mesmo, entre o controlador do robô e a rede elétrica. Para tal, seguiram-se as instruções propostas pelo fabricante no manual de instalação do equipamento [20], para um PM5320 com 3 fases e 1 neutro (3PH4W) e 3 transformadores de corrente (3CT), que foram adquiridos juntamente com o dispositivo. Os transformadores de corrente são equipamentos que reproduzem num circuito secundário, a corrente que circula no circuito primário, numa proporção conhecida e definida. No caso da medição com o PM5320, a sua utilização é essencial, visto que a corrente nominal do equipamento é de 5A (especificação do PM5320), que pode não ser suficiente em

alguns movimentos do robô. Por este motivo, é necessário transformar a corrente do circuito primário para que a medição seja possível. Como tal, adquiriram-se 3 CT de 60/5A, isto é, se no circuito primário circular 60A, no circuito secundário essa corrente é transformada em 5A, passível de ser medida pelo equipamento. Assim, as ligações elétricas do equipamento são apresentadas na figura 3.5.

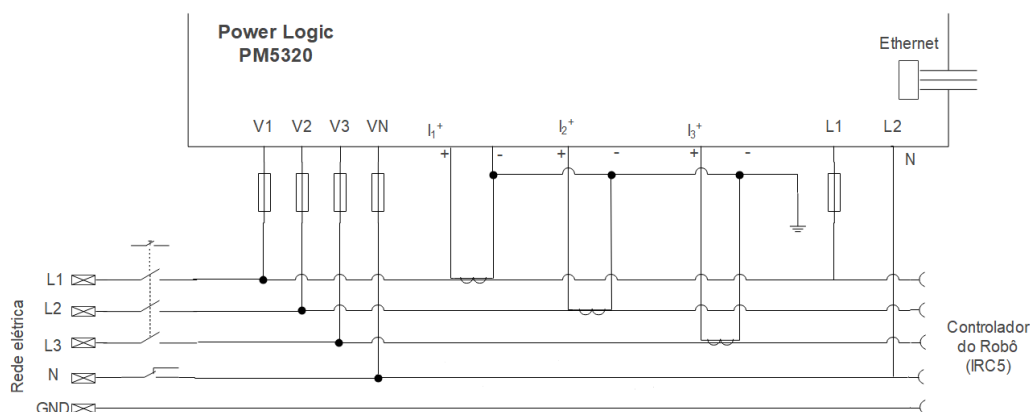


Figura 3.5: Circuito elétrico de implementação do PM5320.

De forma a aumentar a flexibilidade de utilização do equipamento, decidiu-se fazer um armário elétrico, para facilitar a sua transportabilidade e aumentar a segurança do utilizador.

Depois de feita a instalação elétrica do PM5320, foi necessário configurar algumas opções ao nível do *software* do medidor. Para isso, o equipamento dispõe de um *display* e um sistema operativo com uma interface para configuração do sistema. Inicialmente, foi configurado o protocolo de comunicação pretendido, ModBus TCP/IP, seguindo-se a atribuição de um endereço IP, uma *gateway* e uma *Subnet*, segundo a rede do laboratório de robótica. Assim, a partir do endereço IP, a comunicação fica disponível para qualquer computador que esteja ligado à mesma rede.

Após a ligação do equipamento à rede, as restantes configurações do sistema podem ser feitas no *display* ou por um *software* disponibilizado pela Schneider, designado ION Setup. De modo a garantir o correto funcionamento do PM5320, os parâmetros do sistema foram definidos da seguinte forma:

- **Tipo de sistema de potência:** 3Ph4W Grounded. Indica que o sistema a ser medido tem 3 fases com um neutro e ligação à terra.
- **CT Primary:** 30A. Indica a corrente nominal no circuito primário do transformador de corrente. É metade do valor da corrente do transformador que foi adquirido porque foi utilizado com enrolamentos com duas voltas;

- **CT Secondary:** 5A. Corrente nominal no circuito secundário do transformador de corrente;
- **Tensão nominal:** 230V;
- **Corrente nominal** 5A;
- **Frequência nominal da rede** 50Hz;
- **Rotação de fases:** ABC. Seleciona a sequência de fases do sistema trifásico.

Depois de executada a instalação e configuração do PM5320, executaram-se alguns testes para verificar se tudo estava corretamente instalado

3.2.2 Validação do medidor de potência

Como forma de verificar a instalação e configuração do medidor de potência foram realizados alguns testes com condições conhecidas. Inicialmente, optou-se por verificar a medição do fator de potência. Como $FP = \text{Potência Ativa} / \text{Potência Aparente}$, leu-se os dois valores de potência e verificou-se que o valor calculado era aproximadamente igual ao apresentado no *display* do medidor.

Para além deste teste, para verificar se a leitura da corrente nas fases estaria correta, decidiu-se ligar o equipamento a um motor existente no laboratório, com características conhecidas.

Num outro motor trifásico, tinha-se acesso às três fases, pelo que decidiu-se medir a corrente que circulava numa das fases e calculou-se a potência nessa fase e comparou-se com o valor obtido na medição, tendo sido aproximadamente igual.

Depois da validação das medições do PM5320, foi necessário desenvolver um sistema de aquisição de dados, de forma, a armazenar os dados obtidos a partir do equipamento de medição para posterior análise.

3.2.3 Sistema de recolha automática de dados

Visto que o PM5320 é um modelo que não tem disponível a opção de WebService, que possibilitaria a aquisição e tratamento de dados, foi necessário estabelecer um sistema de recolha de dados, de acordo com as necessidades estabelecidas para este trabalho, nomeadamente:

- Interface gráfica para a monitorização das grandezas físicas (P,E,V,I) consideradas para o estudo;
- Armazenamento numa base de dados das medições recolhidas para cada teste;

- Definição do início e fim de gravação para cada teste realizado;
- Interface com capacidade de disponibilizar os gráficos das variáveis pretendidas, com atualização em tempo real para a sua monitorização;

Tendo em vista estas prioridades, a solução desenvolvida requer uma comunicação entre o medidor de potência e o computador onde os dados serão tratados. Como referido, o medidor dispõe de uma porta Ethernet que o permite conectá-lo a um *switch*, e assim, torná-lo acessível aos computadores da rede. A arquitetura física do sistema desenvolvido é a apresentada na figura 3.6.

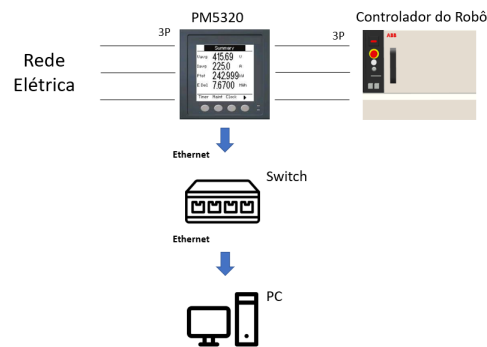


Figura 3.6: Arquitetura física do sistema de aquisição de dados.

O armazenamento e tratamento de dados é feito no computador (PC) a partir dos dados enviados pelo medidor por ModBus TCP/IP. Para a ligação e processamento de dados foram utilizadas três soluções de *software open-source*, como o Node-Red para a aquisição e processamento dos dados e desenvolvimento da interface gráfica de monitorização, o MariaDB [21] como base de dados para o armazenamento e o Grafana [22] para o desenvolvimento da interface gráfica da monitorização do histórico do consumo. Assim, a arquitetura do sistema de aquisição de dados encontra-se esquematizada na figura 3.7.

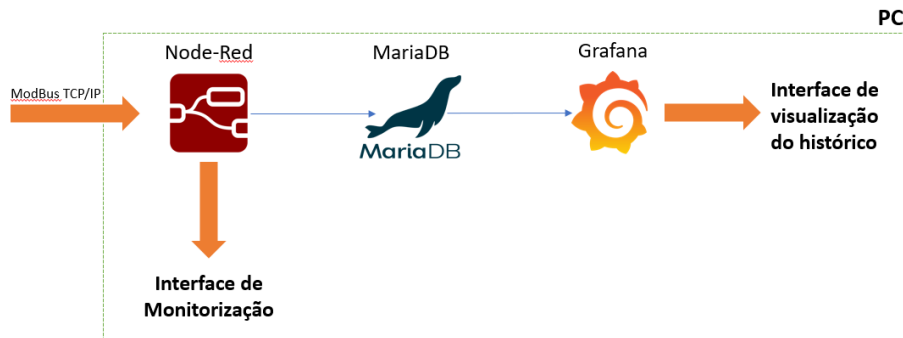


Figura 3.7: Arquitetura de desenvolvimento da recolha de dados no PC.

Node-Red

O Node-Red é um *software open-source* que utiliza uma programação gráfica através da especificação de um conjunto de interligações entre blocos (*flows*). Esta é uma ferramenta muito flexível, que facilita o rápido desenvolvimento de aplicações, muito associado com a corrente digitalização dos processos e da introdução do conceito da Internet das coisas (IoT), pois permite aos seus utilizadores de uma forma simples criar e configurar interfaces de aplicações *real-time* em dispositivos terminais. Este *software* é construído sobre a plataforma Node.js, sendo grátis e programável em JavaScript. Dentro dos *flows*, existem blocos que podem ser movidos e ligados a outros, para além daqueles que estão disponíveis por definição, sendo possível importar outros através de código JavaScript.

No caso deste projeto, o *flow* desenvolvido tem como funções principais: converter os dados recebidos por ModBus TCP/IP, enviar dados selecionados (Potência Ativa, Reativa e Aparente, Energia Ativa Fornecida, Recebida e Fornecida+Recebida) para uma base de dados e, ainda, desenvolvimento da interface de monitorização. Na figura 3.8, está representado o *flow* desenvolvido neste projeto (imagem ampliada no anexo B).

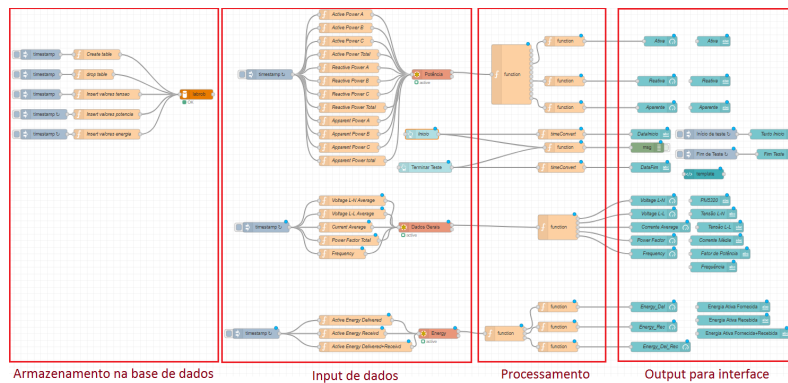


Figura 3.8: *Flow* desenvolvido para a aquisição de dados.

Na figura 3.8, o *flow* apresentado está dividido em 4 zonas de interesse, divididas pelas funções seguintes:

- **Input de dados:** é feita a seleção dos dados a receber do equipamento, através de blocos disponíveis da biblioteca do ModBus TCP/IP, onde se define o IP do equipamento, o tipo de registo a que se pretende aceder e o n.º deste registo;
- **Processamento:** os dados são enviados em diferentes formatos (*FLOAT32*, *INT64*), sendo necessário na receção o seu processamento de forma a torná-lo legível e no formato necessário para o estudo;
- **Armazenamento na base de dados:** depois de convertidos os dados para

o formato desejado, procedeu-se ao armazenamento dos mesmos. Para tal, utiliza-se a biblioteca *mysql*, de forma a conectar o Node-Red à base de dados. Através dos nós função, são definidas *queries* para criar e apagar tabelas e inserir dados nas mesmas.

- **Saída para a interface:** desenvolvimento da interface gráfica com aplicação dos nós da biblioteca da *dashboard*. A interface de monitorização desenvolvida está apresentada na figura 3.9.

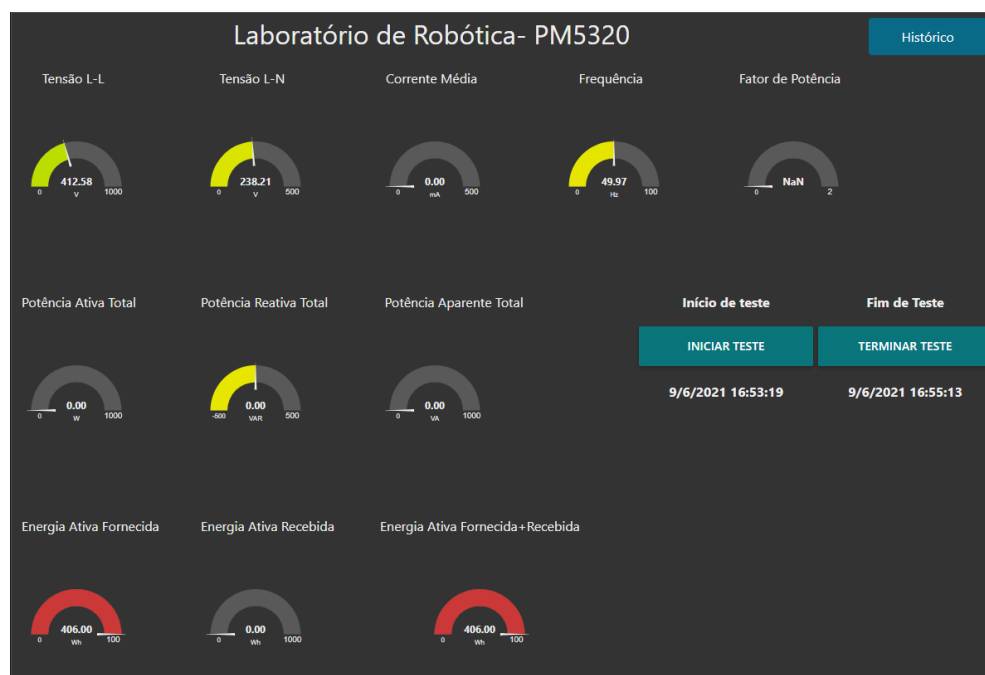


Figura 3.9: Interface de monitorização em tempo real do consumo do robô IRB2400.

De ressaltar que tanto a interface de programação como a de monitorização estão protegidas por um nome de utilizador e *password*, sendo que também só podem ser acessíveis a dispositivos ligados à rede.

MariaDB

O MariaDB é uma base de dados *open-source* que permite aos seus utilizadores, criar bases de dados e tabelas para envio e armazenamento de dados. Para tal, utiliza a linguagem padrão SQL, o que facilita a sua utilização. Para o efeito deste estudo, criaram-se duas tabelas: Potência e Energia Ativa, onde na primeira se guarda a Potência Ativa, Potência Reativa e Potência Aparente, e na segunda Energia Ativa Fornecida, Energia Ativa Recebida e Energia Ativa Fornecida+Recebida.

Grafana

O Grafana é, também, uma aplicação *open-source* que permite a análise e visualização de dados. Este *software* possibilita a visualização a partir da web de tabelas e gráficos quando conectado a uma base de dados.

Na presente dissertação, esta aplicação foi utilizada com o intuito de gerar gráficos de potência e energia ativa em função do tempo, apresentando-os numa interface e atualizando-os em tempo real. Desta forma, conectou-se às tabelas existentes na base de dados e geraram-se dois gráficos, um de potência outro de energia, estando estes acessíveis ao clicar no botão “Histórico” na interface de monitorização. É ainda possível definir qual a escala temporal que se pretende visualizar, como por exemplo, últimos cinco minutos ou últimos dois dias. Por fim, tem uma função onde é possível descarregar os dados contidos nos gráficos na escala de tempo selecionada, para ficheiros CSV. A interface desenvolvida está apresentada na figura 3.10 tendo sido selecionada uma escala de tempo de uma semana.

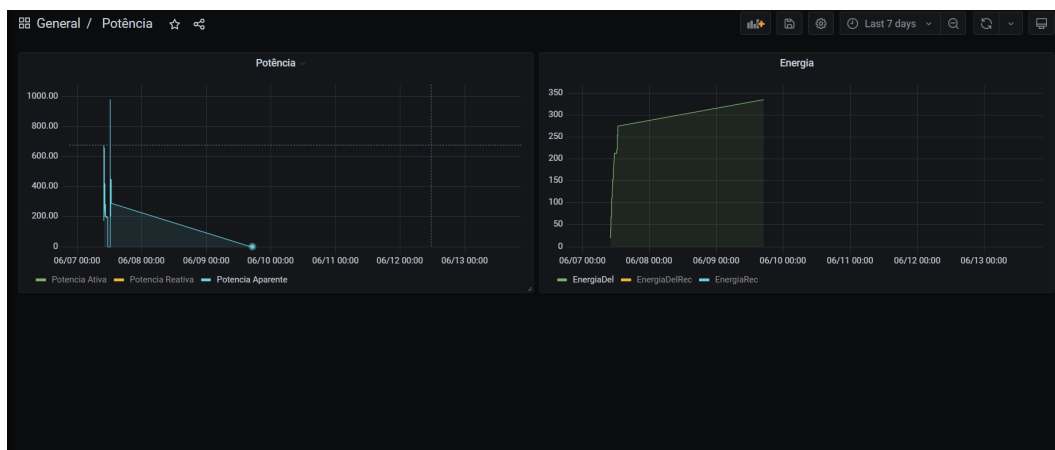


Figura 3.10: Interface de monitorização do histórico de consumo.

Neste capítulo, apresentou-se a caracterização do robô IRB2400 que integra a célula do laboratório de robótica, com um estudo sobre as instruções e ferramentas disponibilizadas no *software* de programação *offline* RobotStudio, da ABB, aprofundando as consideradas mais relevantes para o estudo do consumo energético em robôs industriais. Posteriormente, caracteriza-se a instalação do PM5320 na célula robótica, verificando-se a fiabilidade da medição através de testes em condições conhecidas. Este equipamento comunica com uma rede de computadores, por ModBus TCP/IP, para onde os dados são enviados, tendo sido desenvolvido um sistema de recolha automática de dados, para a monitorização e análise da medição.

Desta forma, estão todas as condições reunidas para a definição das trajetórias de teste e repetitiva implementação na célula robótica, já dotada de um sistema de monitorização do consumo energético.

Capítulo 4

Apresentação dos percursos de teste e exposição dos resultados

A presente dissertação, tem por base o estudo do impacto da programação da trajetória no consumo energético de um robô industrial. Desta forma, é necessário definir e implementar com o robô IRB2400 uma série de ensaios, que permitam avaliar os parâmetros programáveis que se demonstram relevantes para este objetivo. Nesta medida, numa trajetória, os parâmetros mais significativos para o consumo energético são:

- **Velocidade:** programada pela instrução *speeddata*;
- **Aceleração:** cada trajetória é realizada com um parâmetro de aceleração e variação de aceleração (*jerk*), através da instrução em código RAPID, *AccSet*;
- **Tipo de movimento:** interpolação da trajetória por movimento de juntas (*MoveJ*) ou movimento linear (*MoveL*);
- **Posição no espaço de trabalho:** utilização de diferentes localizações do volume de trabalho em momentos de paragem do robô;
- **Carga:** utilização de uma garra de 3.5kg, na célula real, para execução dos diferentes percursos. No simulador, executam-se algumas trajetórias com cargas diferentes;
- **Aproximação ao ponto utilizando o parâmetro "zone":** configuração através da instrução *zonedata*. Foi utilizada a definição *fine* em percursos com paragem obrigatória pelo ponto programado.

O impacto da utilização destes parâmetros é avaliado através da análise do custo/benefício que resulta em termos do tempo de execução e do consumo energético durante a realização da trajetória.

Do ponto de vista da implementação dos vários percursos, cada trajetória é executada na célula robótica e no simulador do RobotStudio. Na célula robótica,

os resultados para o tempo de execução são obtidos na consola do robô, através do programa disponível no Anexo A, e o consumo total fornecido ao controlador do robô, medido pelo PM5320, que foi instalado na célula robótica entre o controlador do robô e a rede elétrica. Na simulação, o tempo de execução obtém-se na interface da consola virtual, de forma igual à apresentada na figura 3.2, e o consumo energético é medido pela utilização da ferramenta *Signal Analyzer*. Apenas os testes que apresentam variações significativas na carga é que foram realizados exclusivamente no simulador.

Assim, durante a fase de testes foi decidido, inicialmente, realizar dois testes de validação. O primeiro, apenas na célula real, para validar a leitura dos picos energéticos por parte do dispositivo de medição de energia elétrica e um segundo, apenas em simulador, para verificar a influência da carga e do modelo do robô nos resultados obtidos no *Signal Analyzer*.

4.1 Testes de validação das funcionalidades do PM5320 e do *Signal Analyzer*

4.1.1 Medição do consumo energético com o PM5320

Quando se avalia o consumo energético de um motor elétrico, é sabido que na fase de arranque ou aceleração, são esperados maiores consumos devido aos picos de corrente exigidos para vencer as forças de aceleração e à diminuição das forças contraeletromotrizes a baixas velocidades. Visto que os dados medidos pelo PM5320, à taxa de 64 amostras/ciclo, são registados pelo sistema de recolha automática de dados, à frequência de 1Hz, definiram-se alguns testes para verificar se estes picos de consumo eram identificados e registados.

Considerando um mesmo percurso, definiram-se duas trajetórias com perfis de velocidade muito diferentes:

- uma de aceleração/desaceleração (perfil triangular, 2 segundos) ao longo de quatro pontos intermédios do percurso;
- outra de perfil trapezoidal com a mesma aceleração e velocidade máxima, mas sem paragem em pontos intermédios.

Como se pretendia realizar um movimento simples, para que fosse facilmente programada uma trajetória com apenas aceleração e desaceleração, o percurso definido tem, apenas, a junta 1 em movimento. Neste caso, o *Signal Analyzer* foi utilizado para obter a trajetória ideal com apenas acelerações e desacelerações. Assim, obteve-se um percurso, com a junta 1 a movimentar-se entre -90° e 60°, aceleração

máxima a ser de 4.5 m/s^2 , velocidade máxima de 2000 mm/s e 4 pontos de paragem (A, B, C e D, na figura 4.1). O perfil de velocidades para esta trajetória está apresentado na figura 4.1.

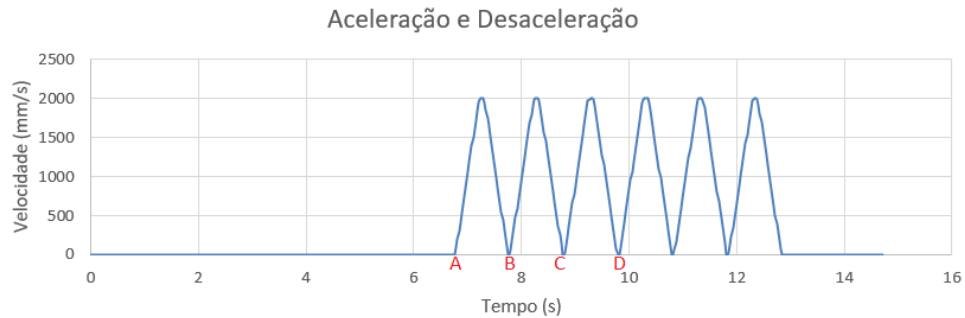


Figura 4.1: Perfil de velocidades para a trajetória de aceleração e desaceleração da junta 1 do robô IRB2400.

Como objeto de comparação, executou-se o mesmo percurso, mas uma trajetória onde o robô atinge a velocidade máxima, mantendo-a constante até atingir o ponto final programado no percurso. Neste caso, não existem zonas de paragens durante a trajetória, mas a amplitude do movimento, a aceleração máxima e velocidade máxima são mantidas. Teoricamente, é expectável que esta trajetória tenha uma potência média inferior à trajetória com constantes acelerações, o que, caso se verifique no PM5320, valida a leitura dos picos de corrente. O perfil de velocidades da trajetória com movimento a velocidade constante, está apresentado na figura 4.2.

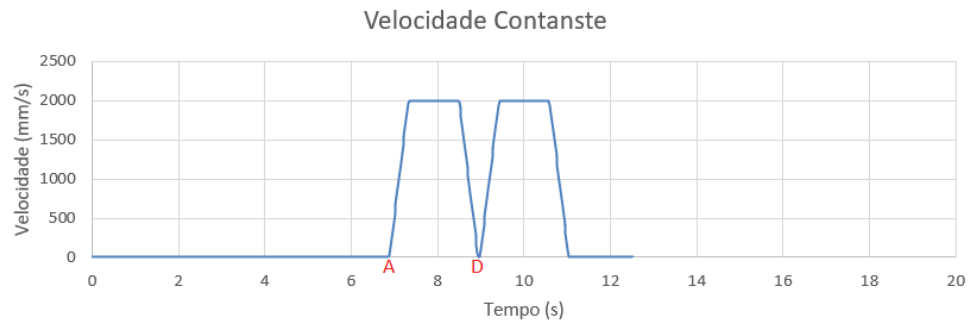


Figura 4.2: Perfil de velocidades para a trajetória a velocidade constante da junta 1 do robô IRB2400.

Nestes ensaios, a trajetória executada é o percurso até ao ponto final, e posterior retorno à posição inicial. Em cada ensaio, a trajetória foi executada 3 vezes, sendo que cada um dos testes foi também realizado 3 vezes. Assim, obtém-se 3 testes para a trajetória de aceleração e desaceleração e 3 testes para a trajetória a velocidade constante. Os resultados de consumo estão apresentados na figura 4.3.

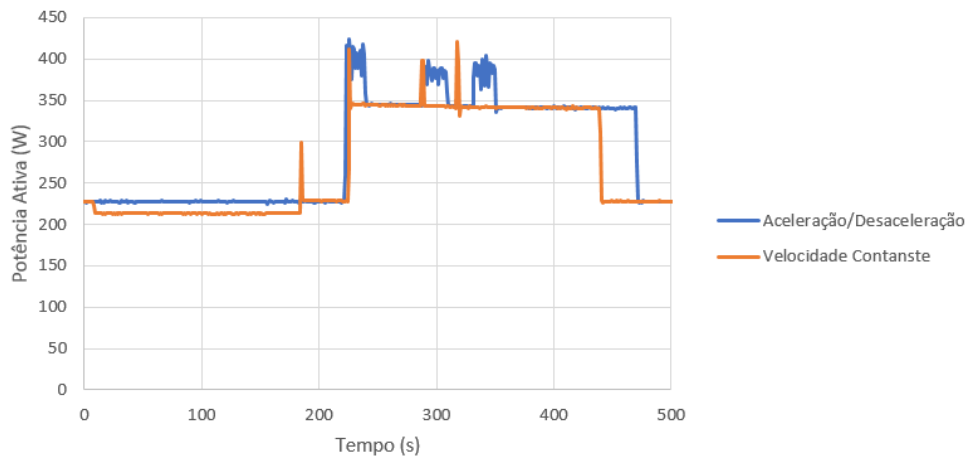


Figura 4.3: Gráfico da potência ativa durante a realização dos testes do PM5320.

Ao analisar a figura 4.3, e tendo em conta os resultados do tempo de execução, obtidos na consola do IRB2400, calculou-se a potência média durante cada um dos ensaios realizados. Os resultados obtidos estão apresentados na tabela 4.1.

Tabela 4.1: Resultados dos testes de validação do PM5320.

Trajectoria	Tempo de execução (s)	Potência Média (W)
Aceleração e Desaceleração	19,93	399,5
	19,96	382,3
	19,96	387,1
Velocidade Constante	13,28	349,7
	12,83	351,5
	12,83	350,6

Dos resultados apresentados na tabela 4.1, pode-se concluir que em termos de potência média os 3 testes com trajetória de aceleração e desaceleração, apresentam potências bastante superiores em relação à execução do mesmo percurso a velocidade constante, o que indica a contabilização dos picos de corrente por parte do PM5320.

Depois de validada a medição do PM5320, foi essencial perceber se parâmetros como a massa da ferramenta ou o modelo do robô tinham influência nos resultados de consumo energético obtidos na ferramenta de análise energética *Signal Analyzer*, do simulador do RobotStudio.

4.1.2 Impacto da massa nos resultados do *Signal Analyzer*

Durante o estudo das funcionalidades do *Signal Analyzer*, surgiram algumas dúvidas sobre os parâmetros que eram considerados na medição do consumo da simulação. O RobotStudio é um *software* que contempla o modelo cinemático e dinâmico dos robôs da ABB, mas não existe nenhuma referência sobre o impacto dos parâmetros da carga e do modelo do robô nos resultados do *Signal Analyzer*.

Desta forma, realizaram-se dois ensaios, para um percurso simples, movimento de 180° da junta 1 (-90° até 90°, figura 4.4), para verificar a influência dos seguintes dois parâmetros:

- **Carga:** em cada movimento é utilizada uma ferramenta (neste caso, *tooldata* caneta), à qual é atribuído, entre outros parâmetros, uma dada massa. Assim, para verificar o impacto de diferentes cargas nos resultados do *Signal Analyzer*, definiu-se para a mesma ferramenta duas massas diferentes: 0,5kg e 16kg;
- **Modelo do robô:** na literatura sobre o *Signal Analyzer*, não foi encontrada informação que esclarecesse se para o mesmo percurso, robôs com capacidades de carga diferentes, tenham resultados de consumo energético distintos. Como tal, para o percurso referido anteriormente, simulou-se o movimento com o IRB2400/16 e IRB6700-200/2.6;

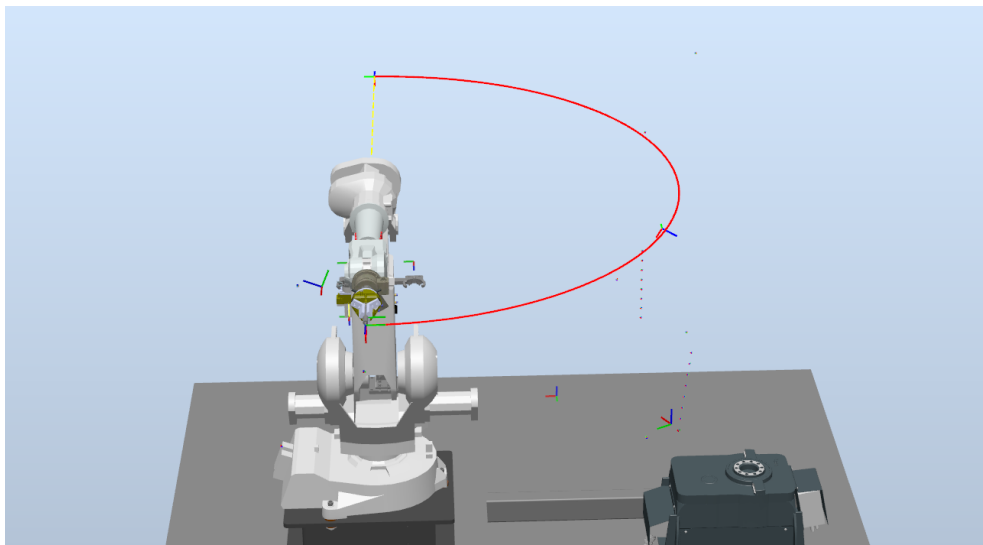


Figura 4.4: Percurso realizado nos testes de verificação do *Signal Analyzer*.

Carga

Nos ensaios de alteração da massa da ferramenta, o principal objetivo foi perceber o impacto da carga no consumo energético, logo, a trajetória foi executada com uma só velocidade (2000mm/s) e à aceleração máxima (*AccSet* 100/100). Como em todos os testes realizados, a trajetória é executada 3 vezes, para uma melhor aproximação do consumo energético. Na figura 4.5, apresentam-se os resultados obtidos para o teste de verificação do impacto da carga no simulador do RobotStudio.

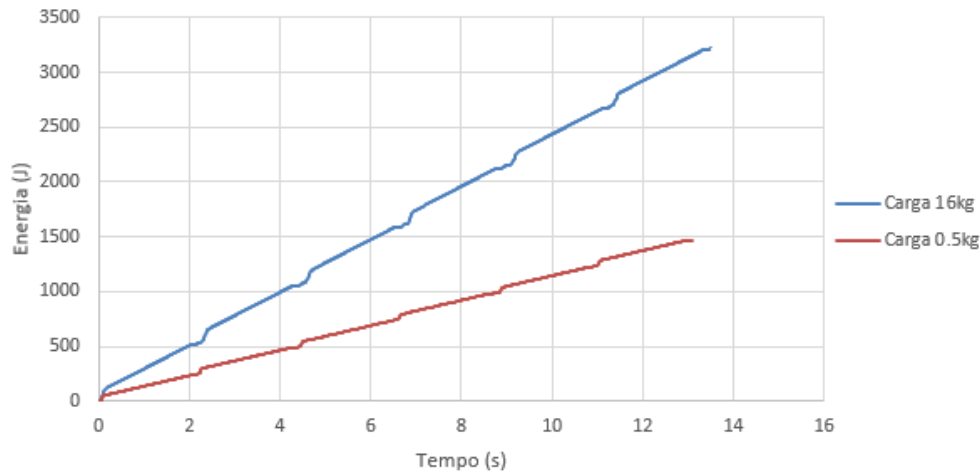


Figura 4.5: Resultados do consumo energético no movimento da junta 1 com duas cargas.

Da análise do gráfico apresentado na figura 4.5, verifica-se que os resultados obtidos através do *Signal Analyzer*, levam em consideração a carga que está aplicada ao robô, pois a curva de consumo da trajetória realizada com 16kg de carga é bastante superior à da carga de 0.5kg. Em termos de consumo médio, a trajetória com uma massa da ferramenta de 16kg apresenta um consumo total de energia de 3218J, muito acima dos 1470J de consumo de energia da trajetória realizada com 0.5kg de carga.

Modelo do robô

Como já referido, para além da carga, um dos parâmetros que se decidiu estudar foi a influência do modelo do robô utilizado nos resultados obtidos com o *Signal Analyzer*. Para tal, utilizou-se o percurso apresentado na figura 4.4, com uma velocidade de 2000mm/s, aceleração máxima (*AccSet* 100/100) e uma carga de 16kg. A trajetória é executada 3 vezes, de forma a obter um valor médio do consumo energético do percurso. Na figura 4.6, são apresentados os resultados do consumo energético do IRB2400 e do IRB6700-200/2.6 (modelo de robô disponível na biblioteca do RobotStudio).

Visto que o IRB6700 é um modelo de robô com uma massa (1205kg) bastante superior ao IRB2400 (380kg), é expectável que o motor implementado no eixo 1 do IRB6700 seja de uma potência nominal bastante superior ao do mesmo eixo do IRB2400. Assim, os resultados da figura 4.6, verificam o que seria esperado, com o IRB6700 a ter um consumo energético total muito superior ao verificado para o IRB2400, validando que o *Signal Analyzer* também tem em conta o modelo do robô utilizado no estudo do consumo energético.

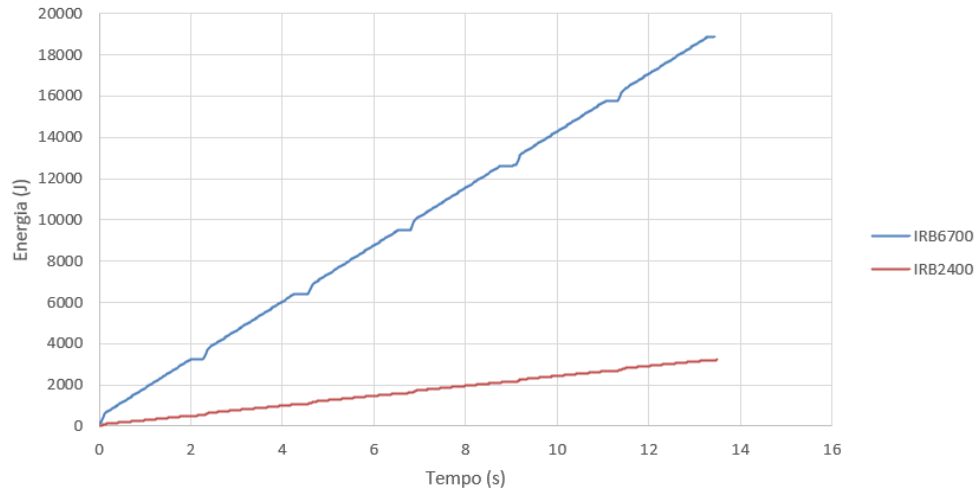


Figura 4.6: Resultados do consumo energético no movimento da junta 1 com diferentes modelos de robô.

4.2 Percursos de teste e resultados obtidos

Depois de validada a medição do PM5320 e algumas funcionalidades do *Signal Analyzer*, definiu-se os percursos para a análise do impacto dos diferentes parâmetros no consumo energético. A estratégia de testes inicia-se com um ensaio para determinar o consumo do robô em "standby", seguindo-se de alguns percursos simples, com movimentação de apenas uma junta. Posteriormente, executa-se um teste com as diagonais interiores do ISO-Cube (definido nas especificações do IRB2400) e realizam-se vários testes de emulação de tarefas industriais do robô, como por exemplo, *Pick And Place* e soldadura ponto a ponto.

Desta forma, durante a fase de testes foram executadas 116 trajetórias na célula do laboratório de robótica, e 120 trajetórias no simulador do RobotStudio, que incluem:

- 4 localizações no espaço de trabalho (4 vértices do ISO-Cube);
- 10 percursos distintos: Standby, Movimento angular da junta 1, Movimento angular da junta 3, Movimento da junta 1 e 3, duas diagonais interiores do ISO-Cube, dois percursos de soldadura, dois percursos de *Pick and Place*;
- 6 velocidades diferentes, em mm/s: 2000, 1800, 1000, 500, 300, 100;
- 3 condições de aceleração/jerk: 100/100, 20/100, 20/20;
- 1 carga a movimentar, 3,5kg correspondente à massa da garra montada no robô;
- 2 tipos de movimento: *MoveJ* e *MoveL*;

Nas seguintes secções, são apresentados cada um dos percursos de teste realizados, seguidos dos resultados obtidos, em termos de tempo de execução e consumo energético durante a realização de cada trajetória, na célula robótica e no simulador do RobotStudio.

4.2.1 Consumo em *standby*

Na análise do consumo energético de um robô industrial, um dos pontos fulcrais é o conhecimento do consumo energético do robô quando está parado. Como tal, a energia total consumida por um robô industrial, quando parado numa determinada posição, pode ser dividida nos seguintes níveis [23]:

1. **Consumo base do controlador:** energia necessária para ligar o controlador do robô;
2. **Acionamento dos motores:** corresponde ao momento em que os motores são ligados, e potência necessária para esse acionamento, mantendo os travões em funcionamento;
3. **Desativação dos travões:** potência necessária para manter o robô na posição, quando os travões mecânicos são desativados e respetiva energia de desativação.

Assim, ao ligar o IRB2400, na célula robótica, procurou-se definir estes níveis nos resultados medidos pelo PM5320. Na figura 4.7, apresenta-se a variação da potência ativa medida no dispositivo, durante as várias fases de acionamento do IRB2400, quando este está parado na posição "home".

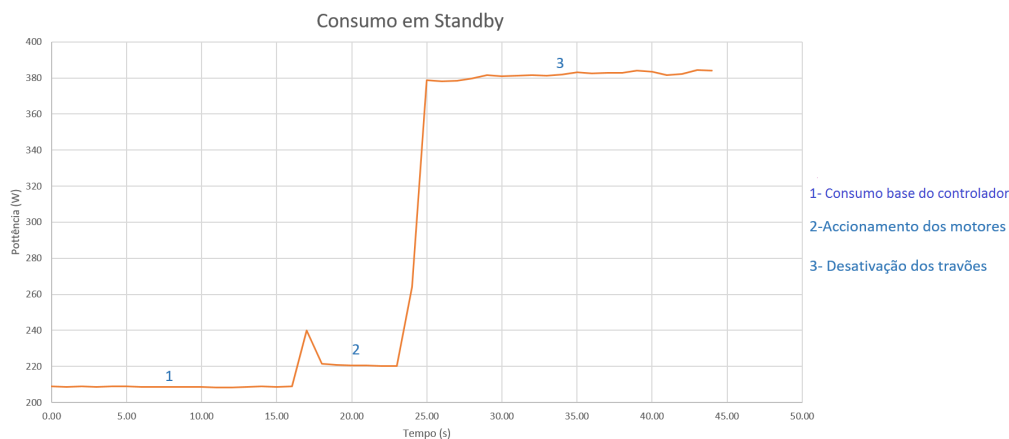


Figura 4.7: Gráfico da potência ativa do robô quando imobilizado na posição "home".

Pela figura 4.7, é facilmente deduzível o momento em que cada fase de acionamento do IRB2400 é realizada. De ressaltar que, no momento da desativação

dos travões, o consumo energético do robô tem uma subida abrupta, derivado à necessidade de realizar o controlo da posição. Os resultados obtidos vão de encontro à informação existente em [23], que para o mesmo modelo de robô apresenta um consumo em *standby* bastante idêntico.

Este teste não foi realizado no simulador, devido ao facto de não ser possível ligar o robô segundo estas fases de acionamento, e também, visto que a potência medida no *Signal Analyzer* diz respeito à energia consumida pelos motores e não à energia consumida na entrada do controlador.

Depois da análise do consumo de um robô industrial quando está imobilizado, foram realizados testes só com a movimentação de uma junta, como forma de avaliar o impacto dos parâmetros de velocidade e aceleração em movimentos simples.

4.2.2 Movimento junta a junta

No *software* de programação *offline* RobotStudio, uma das formas mais simples de movimentar o robô, é alterar o posicionamento de cada junta através da ferramenta "Mechanism Joint Jog". Aí é possível definir a amplitude pretendida para cada junta, sendo que os valores definidos, levarão o robô para uma dada posição do espaço de trabalho.

Desta forma, é possível definir um movimento de apenas uma junta, o que permite em diferentes percursos, avaliar o que cada um dos eixos representa no consumo global do robô, sendo a avaliação dos resultados simplificada, uma vez que a trajetória realizada não é afetada por outros eixos. Assim, decidiu-se realizar os seguintes 3 percursos de teste:

- **Movimento angular da junta 1:** percurso realizado em um plano horizontal, e definido através do movimento do eixo 1 do IRB2400, com a amplitude da junta a variar de -90° a 90° e os restantes 5 eixos permanecem fixos na posição 0;
- **Movimento angular da junta 3:** subida e descida do braço robótico, através do movimento da junta 3 do IRB2400, com uma amplitude de -36° até $48,40^\circ$ e os restantes eixos fixos em 0;
- **Movimento junta 1 e 3:** combinação dos dois percursos enumerados anteriormente, movimentando simultaneamente as juntas 1 e 3, com as amplitudes referidas para o seu movimento individual, e os restantes eixos fixos em 0.

Neste conjunto de ensaios pretendeu-se avaliar o impacto dos parâmetros de programação da velocidade e da aceleração no consumo energético de uma junta de movimento. Como tal, para cada um dos percursos, realizaram-se 9 trajetórias, com os seguintes parâmetros:

- **3 Velocidades:** 500, 1000, 2000 mm/s;
- **3 parâmetros Aceleração/Jerk:** 20/20, 20/100, 100/100.

Cada uma das trajetórias é executada tanto na célula robótica como no simulador do RobotStudio, como forma de verificar a validade dos resultados obtidos no simulador, em termos do tempo de execução de um trajeto e do consumo energético despendido para esse movimento.

Como referido anteriormente, cada trajetória é executada 3 vezes, como forma de obter um valor médio do consumo energético estatisticamente mais correto, pois tem-se três amostras da mesma trajetória. Neste caso, o impacto do parâmetro *zone* está a ser estudado nestas trajetórias, como tal, especificou-se este parâmetro como *fine*, com o objetivo de forçar a paragem do robô em cada ponto programado da trajetória para dispor de zonas de aceleração e desaceleração de maior duração.

Na execução de cada uma das trajetórias programadas, foi desenvolvido um programa que permite implementar a execução das trajetórias a partir da consola de programação do robô (no caso da célula do laboratório) ou da consola virtual do RobotStudio (no caso da execução e simulador). Na figura 4.8, são apresentados os vários ecrãs disponíveis para a realização de cada teste.

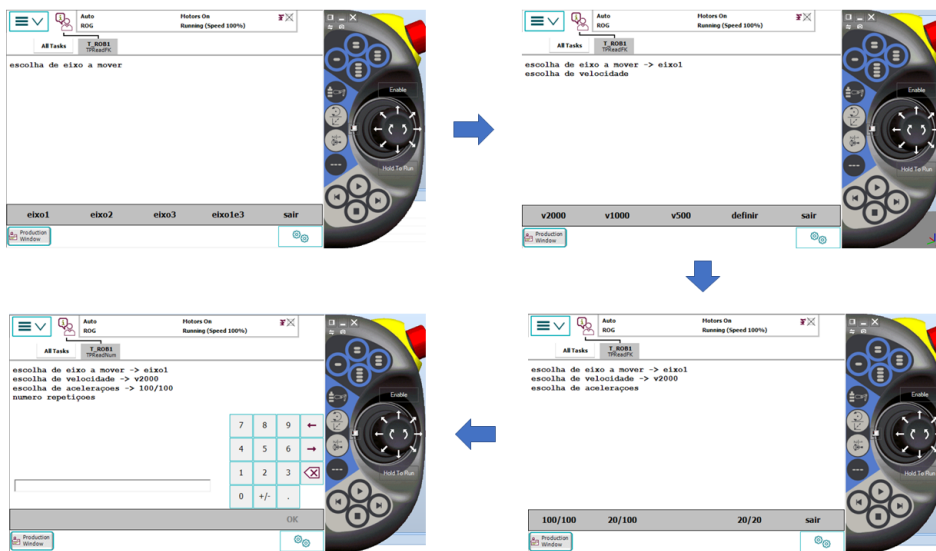


Figura 4.8: Fluxo de interfaces no início de cada teste.

O programa apresentado na figura 4.8, permitiu que o processo de teste fosse bastante mais flexível, sem que fosse necessária a alteração do código da trajetória para mudar a velocidade ou aceleração do teste. Assim, nas subsecções seguintes são apresentados os diferentes percursos de testes e resultados obtidos para cada um.

4.2.2.1 Movimento angular da junta 1

Como referido anteriormente, para a realização do movimento angular da junta 1 do robô, varia-se a junta 1 de -90° até 90° . Assim, para a implementação de qualquer movimento, é necessário desenvolver um *procedure* em código RAPID. No caso, do movimento da junta 1, o *procedure* tem o nome de **Path_10**, e o código desenvolvido para a sua execução está apresentado na figura 4.9.

```
PROC Path_10()
  MoveJ Target_50,myvel,fine,caneta\WObj:=wobj0;
  MoveJ Target_60,myvel,fine,caneta\WObj:=wobj0;
  MoveJ Target_50,myvel,fine,caneta\WObj:=wobj0;
ENDPROC
```

Figura 4.9: Código RAPID para a realização do percurso de movimento da junta 1.

A execução deste código, dá então origem ao percurso pretendido, que está apresentado na figura 4.10.

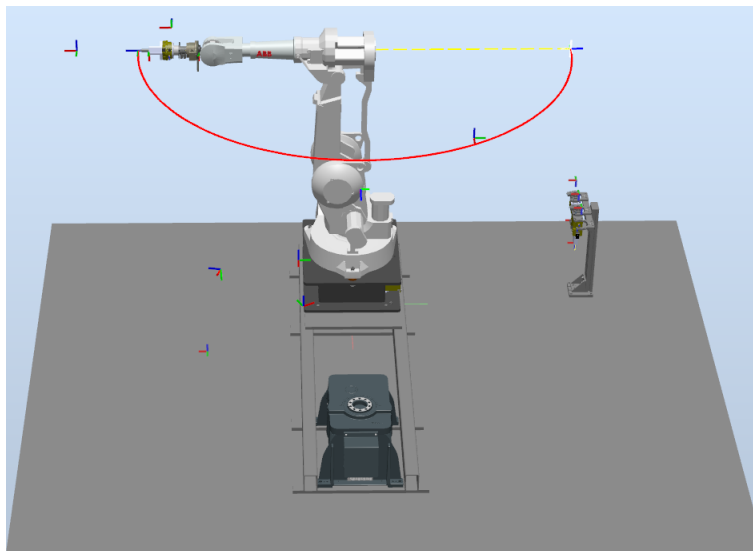


Figura 4.10: Percurso (linha vermelha) programado para o movimento da junta 1.

Depois da realização das 9 trajetórias no robô e no simulador, é necessário fazer o tratamento dos dados de cada uma das medições. No que diz respeito aos dados da célula robótica, inicialmente, é necessário descarregar em ficheiro Excel, as medições referentes aos gráficos desenvolvidos na interface do Grafana (figura 3.10), tanto para a Potência Ativa como para a energia acumulada fornecida ao controlador. Posteriormente, esses dados são importados para um ficheiro Matlab, e apresentados sob a forma de gráfico, como na figura 4.11.

Na figura 4.11, são apresentadas três curvas: a laranja, a Potência Ativa (em

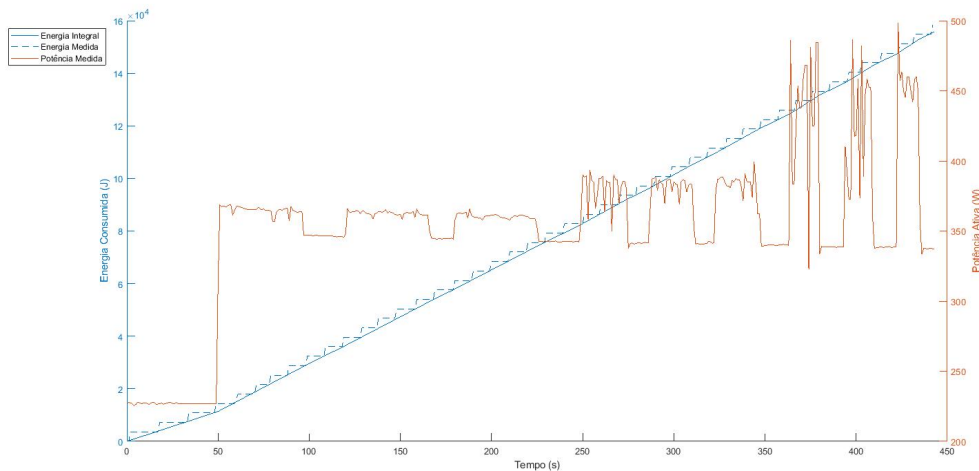


Figura 4.11: Gráfico em Matlab, com as medições de Potência e Energia para o movimento angular da junta 1.

W), medida durante o movimento angular da junta 1, a azul contínuo, a Energia acumulada integral (em J), obtida em Matlab pela integração a partir do gráfico da Potência Ativa, e a traço interrompido azul (em J), a energia acumulada medida pelo PM5320 durante o movimento. O facto da energia acumulada, medida no PM5320, ter uma resolução de 1KWh (3600J), apresentou-se como um entrave no tratamento dos resultados obtidos, pois em alguns movimentos, o consumo era inferior a 1KWh, o que não seria visível pela medição do PM5320. Assim, decidiu-se fazer a integração da Potência Ativa, de forma a ter acesso a todas essas variações inferiores a 1KWh, mas mantendo a mesma informação, pois as curvas são basicamente coincidentes.

Para obter o consumo total de uma trajetória, foi necessário procurar o valor inicial e final do tempo da sua realização, que é obtido através das abruptas transições ascendentes e descendentes na curva da Potência Ativa. Posteriormente, visto que as medições da energia estão, em Matlab, sobre a forma de vetor, bastou fazer a diferença entre a energia acumulada correspondente ao tempo final obtido e a energia acumulada no momento inicial.

No que diz respeito ao simulador, o tratamento de dados é um pouco mais facilitado. Na janela da ferramenta do *Signal Analyzer*, apresentada na figura 4.12, obtêm-se 3 gráficos. O primeiro diz respeito à energia acumulada consumida pelo motor (em J), a segunda, é a potência instantânea durante o movimento (em W), e a terceira curva, é a velocidade do elemento terminal do robô (mm/s) durante o teste.

O consumo total de cada trajetória realizada é calculado, através da diferença entre a energia acumulada no fim da trajetória e a inicial. Estes valores são adquiridos tendo em conta o valor do tempo em que a velocidade deixa de ser nula, no caso

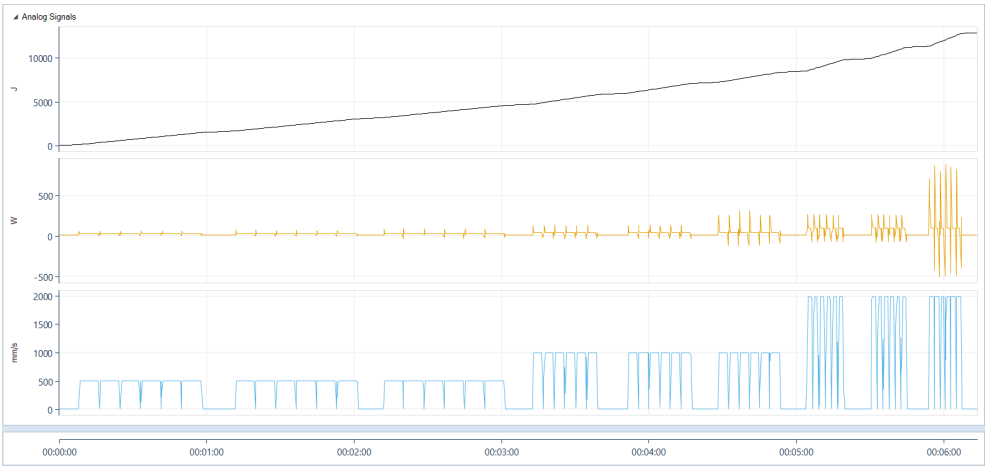


Figura 4.12: Resultados obtidos na ferramenta *Signal Analyzer* para o movimento da junta 1.

do momento inicial, e quando a velocidade retorna a 0 no caso do momento final.

Quanto ao resultado do tempo de execução da trajetória, este é lido diretamente da consola do robô e na consola virtual do simulador. Assim, para o movimento da junta 1, os resultados obtidos estão apresentados na tabela 4.2, apresentando-se o tempo de ciclo médio (ida e volta ao ponto inicial), a energia total consumida durante a execução da trajetória e a potência média da trajetória (consumo total/tempo de execução).

Tabela 4.2: Resultados obtidos para o consumo de energia e tempo de execução no movimento angular da junta 1 nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)	Potência Média ¹ (W)
Real	500	20/20	16,0	17394	362,4
		20/100	15,3	16670	362,4
		100/100	15,2	16559	363,9
	1000	20/20	8,6	9869	381,1
		20/100	8,1	9111	376,7
		100/100	7,8	9198	392,2
	2000	20/20	5,6	7293	431,1
		20/100	4,8	6666	461,3
		100/100	4,2	5840	464,5
Simulador	500	20/20	16,8	1988	39,46
		20/100	16,6	2023	40,6
		100/100	16,5	1993	40,3
	1000	20/20	8,9	1518	56,8
		20/100	8,7	1475	56,7
		100/100	8,4	1510	59,9
	2000	20/20	5,3	1582	100,1
		20/100	5,0	1544	103,4
		100/100	4,4	1790	135,0

¹Ao longo de 3 ciclos

4.2.2.2 Movimento angular da junta 3

O percurso programado para o IRB2400 realizar um movimento no plano vertical, neste caso, implica apenas a movimentação da junta 3 do robô. Neste percurso, o robô realiza uma trajetória onde inicialmente desce até ao ponto final do percurso, voltando a subir para o ponto inicial, como apresentado na figura 4.13. Os resultados do consumo energético e do tempo de ciclo foram obtidos da mesma forma que para o movimento da junta 1, e estão apresentados na tabela 4.3.

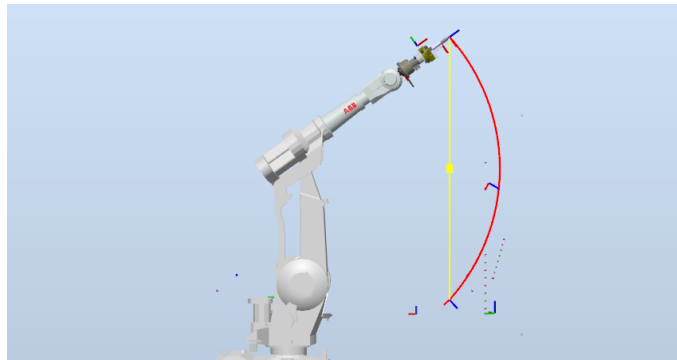


Figura 4.13: Percurso (linha vermelha) programado para o movimento da junta 3.

Tabela 4.3: Resultados obtidos para o consumo energético e tempo de execução no movimento angular da junta 3 nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)	Potência Média ¹ (W)
Real	500	20/20	7,9	8743	370,5
		20/100	7,4	8289	373,9
		100/100	7,3	8282	376,4
	1000	20/20	4,5	5341	395,3
		20/100	4,0	5055	416,2
		100/100	3,9	4707	402,9
	2000	20/20	3,2	4532	465,4
		20/100	2,6	3420	440,3
		100/100	2,2	3191	482,1
Simulador	500	20/20	7,6	1228	53,7
		20/100	7,4	1218	55,2
		100/100	7,3	1228	56,3
	1000	20/20	4,2	1064	84,4
		20/100	4,0	1061	88,9
		100/100	3,8	1088	95,6
	2000	20/20	2,8	1037	125,5
		20/100	2,5	1064	141,8
		100/100	2,1	1199	191,0

¹ Ao longo de 3 ciclos

4.2.2.3 Movimento simultâneo da junta 1 e 3

Por fim, nesta subsecção, apresenta-se um percurso que surge da conjugação dos dois movimentos apresentados anteriormente. A adoção deste teste, inicialmente, deveu-se a um estudo preliminar feito à ferramenta do *Signal Analyzer*, onde, em alguns movimentos, no gráfico da potência se verificavam zonas onde o valor era negativo, o que poderia indiciar a existência de regeneração ao nível dos drivers dos motores. Posteriormente, num contacto com o fabricante elucidou-se que tal não acontecia, pelo que a energia produzida em fase de desaceleração será dissipada.

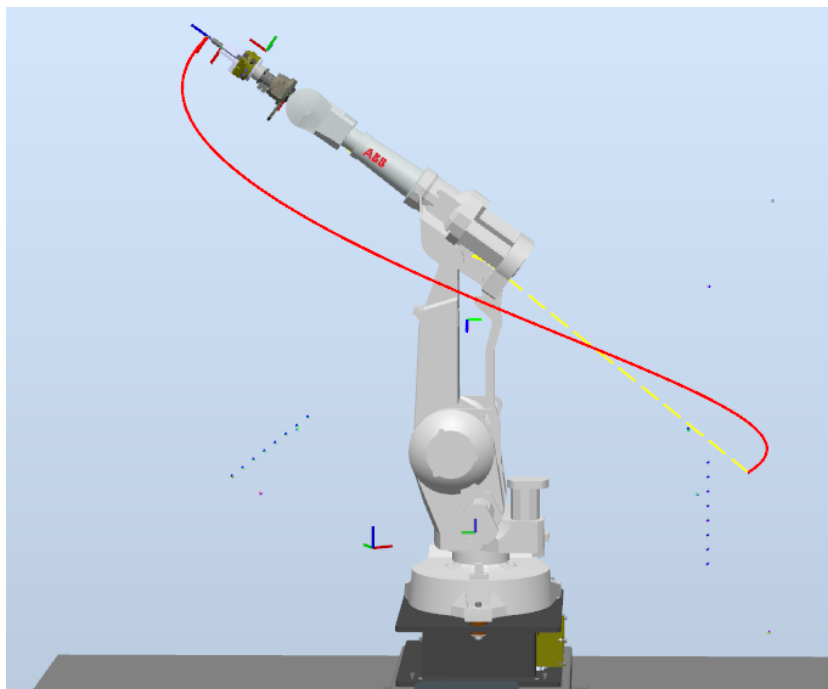


Figura 4.14: Percurso (linha vermelha) executado para o movimento conjugado da junta 1 e 3.

Desta forma, o percurso programado tem como ponto inicial, a posição do TCP onde a junta 1 toma o valor de -90° e a junta 3 de -36° , e ponto final, a posição com a junta a 90° e a junta 3 a $48,40^\circ$. A trajetória volta a ser realizada nos dois sentidos e repetida três vezes em cada ensaio, apresentando-se o percurso executado na figura 4.14.

Pelo mesmo procedimento que todos os testes realizados, os resultados do consumo energético e do tempo de ciclo foram obtidos tanto para o simulador como para a célula robótica, como apresentado na tabela 4.4.

Tabela 4.4: Resultados obtidos para o consumo de energia e tempo de execução no movimento simultâneo da junta 1 e 3 nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)	Potência Média ¹ (W)
Real	500	20/20	15,7	17018	362,1
		20/100	15,2	16834	369,9
		100/100	15,1	16737	369,0
	1000	20/20	8,4	10164	401,3
		20/100	8,0	9781	409,4
		100/100	7,8	9888	424,3
	2000	20/20	5,5	6989	420,1
		20/100	4,6	6773	488,4
		100/100	4,1	6593	530,2
Simulador	500	20/20	17,2	2405	46,51
		20/100	16,9	2404	47,31
		100/100	16,8	2423	48,0
	1000	20/20	9,0	1845	68,0
		20/100	8,8	1843	69,7
		100/100	8,6	1904	73,9
	2000	20/20	5,3	1912	121,4
		20/100	5,0	1907	127,3
		100/100	4,5	2127	157,5

4.2.3 Percursos de teste segundo o ISO-Cube

Atendendo a que no *datasheet* do IRB2400 [17] é referido o consumo energético, à velocidade máxima, para fazer o teste do cubo ISO [24] (apresentado na figura 4.15) decidiu-se utilizar este modelo como base para alguns ensaios.

Assim, foram relacionados dois parâmetros de estudo que se enquadravam com as características das trajetórias definidas com este cubo, sendo estes: utilização de diferentes localizações no espaço de trabalho e a interpolação do movimento utilizado. Desta forma, definiram-se os seguintes dois testes:

- **Diferentes localizações no espaço de trabalho:** utilizaram-se quatro vértices do cubo, para verificar se o consumo energético quando o robô está parado depende da sua zona do espaço de trabalho. Os vértices utilizados são E_1 , E_2 , E_3 , E_4 . Para além disto, atribuíram-se duas orientações diferentes ao *target* definido para cada um desses pontos, também para analisar o impacto deste parâmetro na energia consumida pelo robô;
- **Diagonais interiores do ISO-Cube:** um dos parâmetros mais relevantes na programação de uma trajetória num robô industrial, prende-se com a interpo-

¹ Ao longo de 3 ciclos

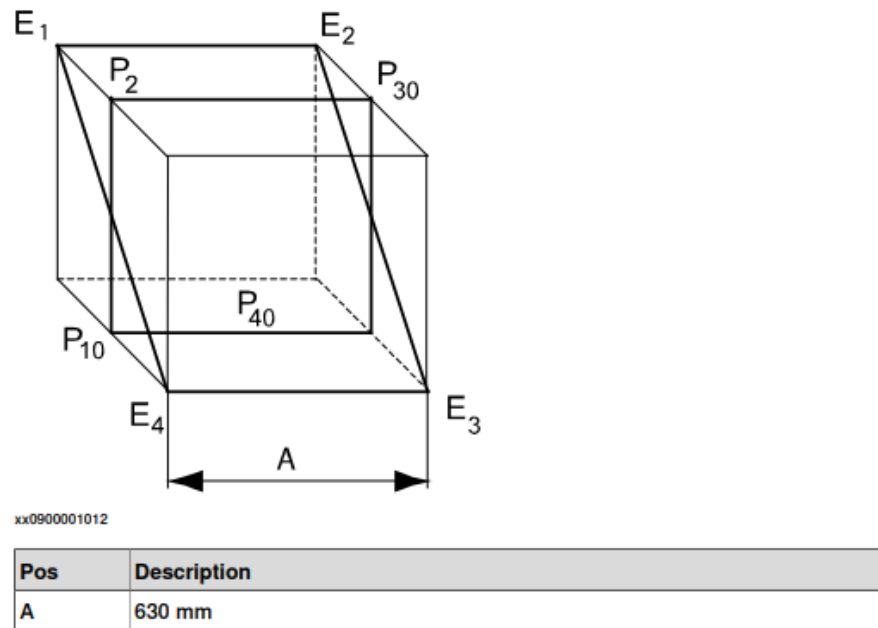


Figura 4.15: Teste do cubo ISO [24].

lação do movimento (movimento de juntas ou movimento linear). Como tal, definiram-se as duas diagonais interiores do cubo ISO (E_1-E_3 e E_2-E_4), como percursos para o estudo do impacto deste parâmetro no consumo energético e no tempo de execução de cada trajetória.

Nas subsecções seguintes, apresenta-se cada um destes ensaios e os respetivos resultados obtidos tanto na célula robótica como no simulador do RobotStudio.

4.2.3.1 Diferentes localizações no espaço de trabalho

Como referido, este teste tem por base, o estudo do consumo energético, tendo em conta o local do espaço de trabalho onde o robô se encontra imobilizado. Na realidade, este ensaio pode sugerir diferenças não muito elevadas, no que diz respeito à potência média, mas pequenas reduções nesse parâmetro enquanto o robô está parado, podem significar elevadas poupanças energéticas na indústria, em que o robô está, por exemplo, em trabalho 24h por dia. Assim, definiram-se 4 pontos que coincidem com 4 vértices do cubo ISO e estudou-se a potência média em cada um deles, para duas orientações do *target* diferentes. Nas figuras 4.16 e 4.17, apresenta-se o robô no vértice E_4 , com duas orientações distintas.

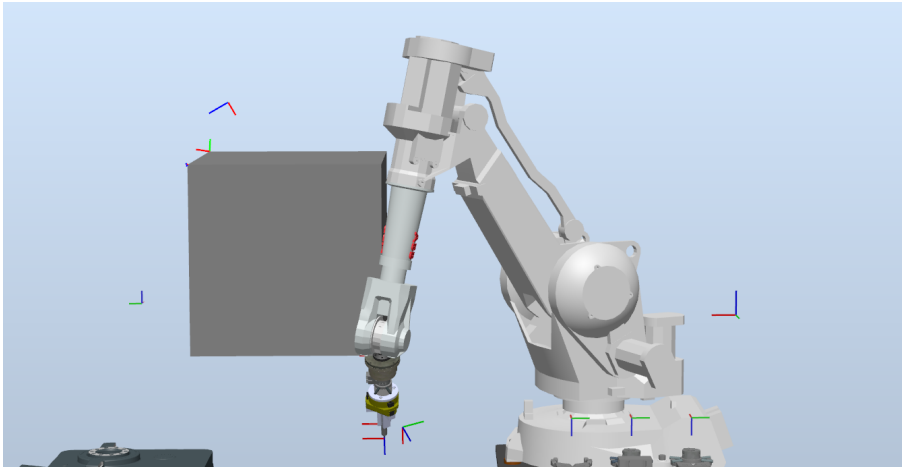


Figura 4.16: IRB2400 na primeira orientação do *target* E₄.

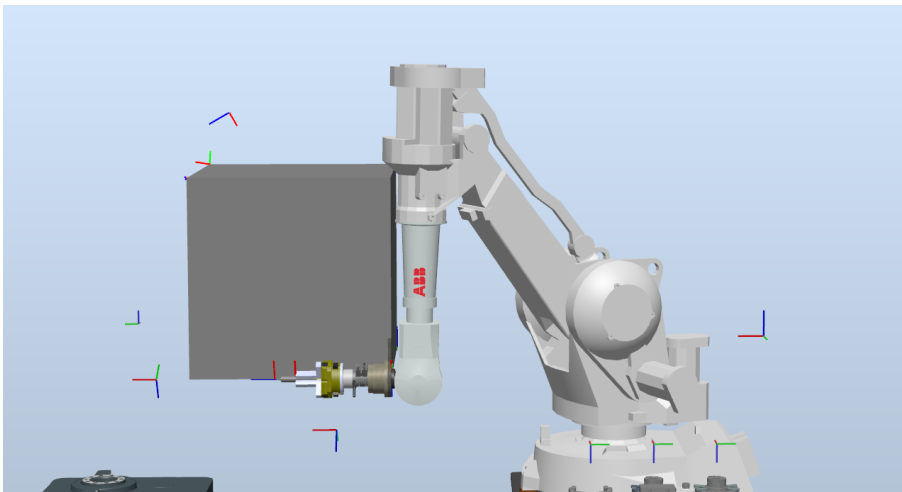


Figura 4.17: IRB2400 na segunda orientação do *target* E₄.

As orientações do *target* nos 4 pontos, foram obtidas da mesma forma, através do *jogging* do robô até cada um dos pontos, sendo assim definida a primeira orientação. A segunda orientação, surge da rotação de cada uma das orientações iniciais, 90°segundo o eixo dos y_y . Desta forma, os resultados obtidos para este ensaio estão apresentados na tabela 4.5.

Tabela 4.5: Consumo com o robô parado em locais distintos no espaço de trabalho.

Sistema	Vértice do cubo	Orientação do <i>target</i>	Potência Média (W)
Real	E ₄	1	373
		2	372
	E ₂	1	388
		2	386
	E ₁	1	381
		2	384
	E ₃	1	369
		2	367
Simulador	E ₄	1	39
		2	43
	E ₂	1	59
		2	57
	E ₁	1	56
		2	57
	E ₃	1	47
		2	44

4.2.3.2 Diagonais interiores do ISO-Cube

Na definição de uma *trajetória* no software de programação *offline* RobotStudio, um dos parâmetros de seleção obrigatória é o tipo de interpolação do movimento pretendida. Como tal, é importante perceber qual é o impacto da seleção da interpolação do movimento no consumo energético de um robô industrial. Desta forma, as duas diagonais interiores do cubo apresentado na figura 4.15 foram utilizadas neste estudo.

Na realização destes testes, definiu-se que a primeira diagonal a executar é entre os vértices E₁-E₃ e a segunda entre E₂-E₄, tal como estão denominados na figura 4.15. Para cada diagonal foram utilizados os seguintes parâmetros para a definição das trajetórias:

- **Interpolação do movimento:** Movimento de juntas (*MoveJ*) e movimento linear (*MoveL*);
- **Velocidade de execução:** 500, 1000, 2000 mm/s;
- **Aceleração (*AccSet*):** 20/20, 20/100, 100/100;

Assim, pode-se depreender que para cada uma das diagonais foram realizadas 18 trajetórias, correspondendo a duas interpolações de movimento diferentes, em que para cada uma dessas se executaram 9 trajetórias.

Como tal, iniciou-se este processo de testes pela realização do percurso da diagonal entre E_1 - E_3 , sendo este executado através de um movimento de juntas (*MoveJ*). A trajetória que o robô realizou está apresentada na figura 4.18 e os resultados obtidos em termos de tempo e consumo energético estão apresentados na tabela 4.6.

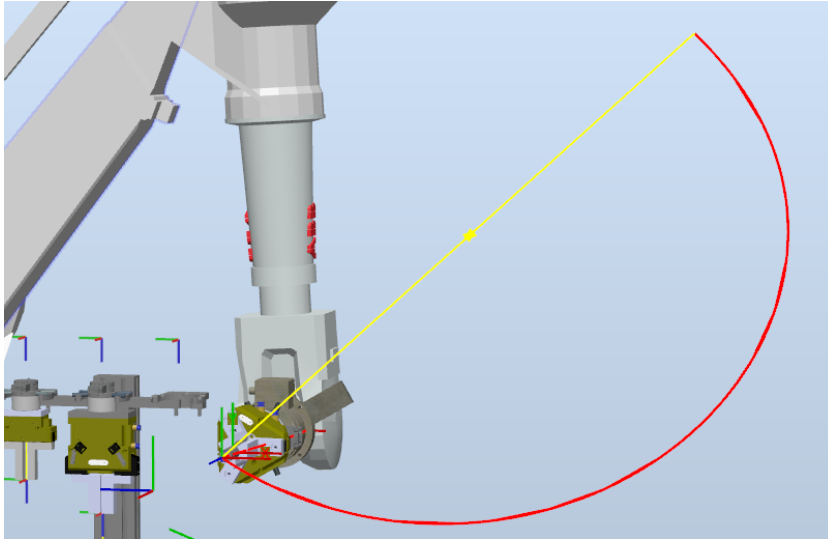


Figura 4.18: Percurso (linha vermelha) realizado para a primeira diagonal através do movimento de juntas do robô.

Tabela 4.6: Resultados obtidos para o consumo de energia e tempo de execução na execução da diagonal 1 por movimento de juntas nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)	Potência Média ¹ (W)
Real	500	20/20	6,4	9002	468,0
		20/100	6,2	8613	465,0
		100/100	6,1	8609	469,8
	1000	20/20	3,9	6131	528,8
		20/100	3,4	5775	563,6
		100/100	3,3	5438	554,4
	2000	20/20	2,9	5601	642,5
		20/100	2,3	4886	708,8
		100/100	1,9	4206	732,4
Simulador	500	20/20	6,7	2355	117,0
		20/100	6,5	2353	119,9
		100/100	6,5	2320	119,4
	1000	20/20	3,8	2004	175,8
		20/100	3,6	2008	187,6
		100/100	3,4	2020	198,0
	2000	20/20	2,5	2059	270,1
		20/100	2,3	2044	299,0
		100/100	1,9	2267	393,2

De seguida, realizou-se o movimento da mesma diagonal (E_1 - E_3 , na figura

¹ Ao longo de 3 ciclos

4.15), mas com a definição do movimento do robô como linear (*MoveL*), onde o percurso executado está apresentado na figura 4.19 e os resultados obtidos para estes percurso estão apresentados na tabela 4.7.

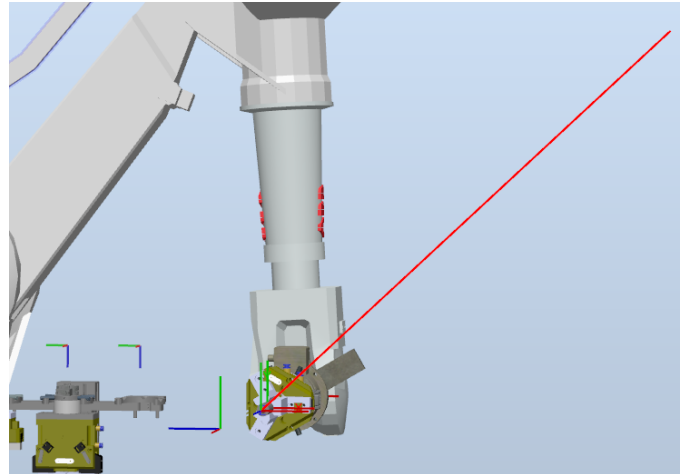


Figura 4.19: Percurso realizado (linha vermelha) para a primeira diagonal através do movimento linear do robô.

Tabela 4.7: Resultados obtidos para o consumo de energia e tempo de execução na execução da diagonal 1 por movimento linear nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)	Potência Média ¹ (W)
Real	500	20/20	6,0	9121	504,8
		20/100	5,4	8056	500,2
		100/100	4,8	7797	536,1
	1000	20/20	4,7	7706	542,4
		20/100	3,9	6663	576,6
		100/100	2,8	6344	757,8
	2000	20/20	4,7	7729	552,5
		20/100	4,5	6691	499,3
		100/100	2,0	5931	967,0
Simulador	500	20/20	5,5	2815	170,1
		20/100	5,2	2740	176,9
		100/100	4,7	2795	199,1
	1000	20/20	3,9	2961	254,5
		20/100	3,6	2963	278,0
		100/100	2,7	3572	448,2
	2000	20/20	3,8	2952	256,3
		20/100	3,5	3010	287,8
		100/100	1,9	4667	829,0

Por outro lado, foi ainda executada a outra diagonal interior do cubo ISO

¹Ao longo de 3 ciclos

(E_2 - E_4 , na figura 4.15), se o impacto do tipo de interpolação era idêntico para outro percurso. Para esta diagonal realizou-se por movimento de juntas (*MoveJ*), tal como apresentado na figura 4.20, e os resultados de consumo energético e tempo de execução demonstrados na tabela 4.8.

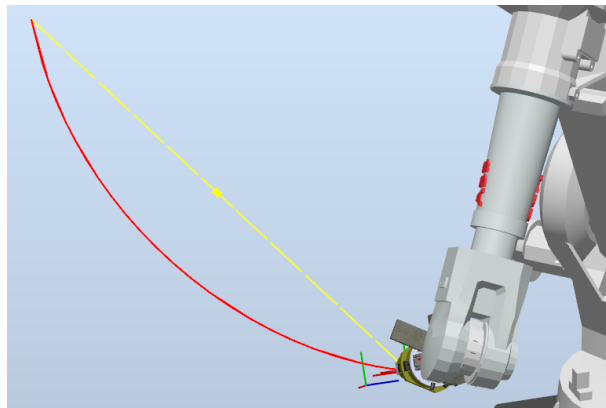


Figura 4.20: Percurso (linha vermelha) do robô ao realizar a segunda diagonal através do movimento de juntas do robô.

Tabela 4.8: Resultados obtidos para o consumo energético e tempo de execução na execução da diagonal 2 por movimento de juntas nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)	Potência Média ¹ (W)
Real	500	20/20	5,7	7359	428,3
		20/100	5,2	6637	427,5
		100/100	5,0	6318	420,1
	1000	20/20	3,9	5235	451,1
		20/100	3,2	4486	471,2
		100/100	2,7	4189	508,9
	2000	20/20	3,3	4612	459,3
		20/100	2,6	3773	484,4
		100/100	1,7	3387	654,5
Simulador	500	20/20	5,4	1668	103,4
		20/100	5,1	1620	105,9
		100/100	4,9	1647	112,0
	1000	20/20	3,4	1367	136,0
		20/100	3,0	1319	144,2
		100/100	2,7	1413	176,8
	2000	20/20	2,8	1426	171,0
		20/100	2,5	1391	189,1
		100/100	1,6	1666	347,2

Por fim, realizou-se um conjunto de testes para o mesmo percurso da segunda

¹ Ao longo de 3 ciclos

diagonal interior do cubo ISO, mas através de um movimento linear do robô (*Mov**veL*). A trajetória executada está apresentada na figura 4.21, e os resultados obtidos para o consumo energético do robô durante a realização destes testes, na tabela 4.9.

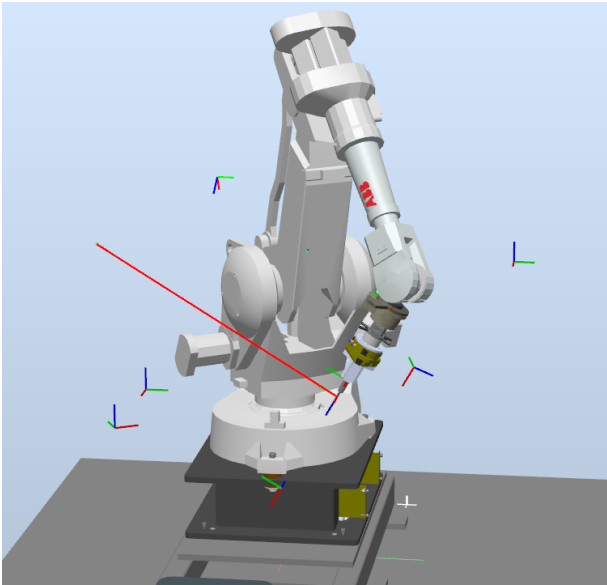


Figura 4.21: Percurso (linha vermelha) do robô ao realizar a segunda diagonal através do movimento de juntas do robô.

Tabela 4.9: Resultados obtidos para o consumo de energia e tempo de execução na execução da diagonal 2 por movimento linear nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)	Potência Média ¹ (W)
Real	500	20/20	5,4	7043	431,3
		20/100	5,0	6653	447,0
		100/100	4,8	6342	442,7
	1000	20/20	3,8	5348	473,0
		20/100	3,1	4630	502,0
		100/100	2,6	3960	499,6
	2000	20/20	3,4	5081	496,6
		20/100	2,6	4043	524,3
		100/100	1,7	3426	684,3
Simulador	500	20/20	5,0	1675	111,5
		20/100	4,8	1672	117,2
		100/100	4,6	1662	121,2
	1000	20/20	3,3	1491	149,8
		20/100	2,9	1463	169,7
		100/100	2,5	1526	206,1
	2000	20/20	2,6	1583	199,2
		20/100	2,4	1556	219,4
		100/100	1,5	1924	425,7

Depois de adquiridos os resultados do consumo energético e do tempo de execução dos diferentes percursos realizados no IRB2400 a partir do cubo ISO, decidiu-se

¹Ao longo de 3 ciclos 49

terminar a fase de testes do consumo energético do IRB2400 com a simulação de alguns movimentos em que o robô é tipicamente utilizado na indústria.

4.2.4 Testes de tarefas típicas do IRB2400 na indústria

Como referido no capítulo 2, os robôs industriais são maioritariamente utilizados em aplicações como o manuseamento ou soldadura. Como tal, verificou-se que existia algum interesse em estudar o consumo energético do robô IRB2400, em alguns percursos que simulassem esse tipo de tarefas, com o objetivo de aproximar este estudo das aplicações reais do robô. Assim sendo, a fase de testes do robô foi concluída com 2 testes de simulação destas tarefas típicas, a de manuseamento através de percursos de *pick and place* e a de soldadura por um percurso de soldadura ponto a ponto.

Nas subsecções seguintes são apresentadas as diferentes trajetórias executadas para esses percursos, bem como expostos os resultados obtidos tanto no robô IRB2400, como no simulador, para o consumo energético e o tempo de execução dessas trajetórias.

4.2.4.1 Simulação de percursos de *pick and place*

No momento da definição do percurso a realizar no teste de *pick and place*, procurou-se encontrar um trajeto *standard*, com o intuito de comparar com os resultados obtidos no IRB2400 com outros robôs no mesmo percurso. Para esse efeito, na especificação dos robôs do tipo SCARA da Epson [25], é fornecido um tempo de ciclo para um percurso *standard* à máxima velocidade. Assim, o percurso definido pela especificação é o apresentado na figura 4.22.

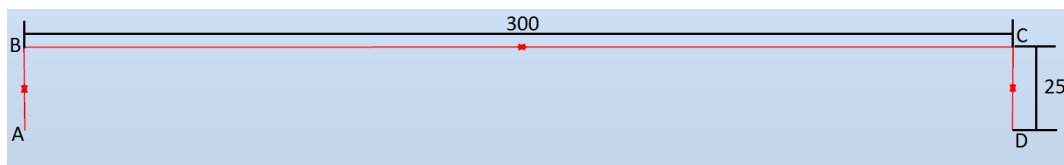


Figura 4.22: Trajeto standard de *pick and place* definido pela Epson para os seus robôs.

Ao analisar este percurso, e as características de uma tarefa de *pick and place*, verificou-se que este teste seria o adequado para avaliar as implicações do parâmetro programável *zone*. Visto que neste tipo de tarefas, os únicos pontos onde é obrigatória a passagem e paragem do robô são o de início do percurso (ponto A, na figura 4.22) e o final (ponto D, na figura 4.22), os outros dois pontos (B e C, na figura 4.22) permitem que exista uma zona de aproximação aos mesmos, revelando que este é o teste indicado para o estudo deste parâmetro. Para além do parâmetro *zone*, o segmento BC do percurso apresentado na figura 4.22, permite ainda avaliar

o impacto do tipo de interpolação do movimento em termos de consumo energético e tempo de execução.

Assim sendo, nestes testes, os parâmetros de programação da trajetória que foram considerados são:

- **Velocidade de execução:** 1800, 1000 mm/s;
- **Aceleração do movimento (*AccSet*):** 100/100, 20/100;
- **Zona de aproximação (*zone*):** *fine*, *z10*;
- **Tipo de interpolação do movimento:** *MoveJ*, *MoveL*.

Desta forma, realizaram-se 16 trajetórias diferentes no IRB2400 e as mesmas 16 no simulador, ou seja, executaram-se 8 trajetórias para cada tipo de interpolação utilizada. Os resultados destes testes estão apresentados nas tabelas 4.10 e 4.11, relativos à interpolação do movimento por juntas e linear, respetivamente. As trajetórias realizadas pelo IRB2400 para essas duas interpolações estão apresentadas nas figuras 4.23 e 4.24.

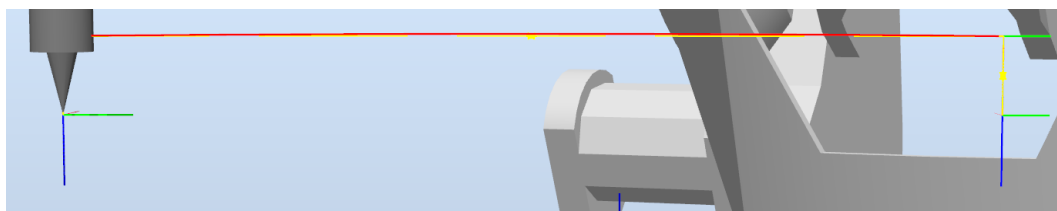


Figura 4.23: Trajetória (linha vermelha) realizada pelo robô ao executar o percurso *standard* da Epson por movimento de juntas.

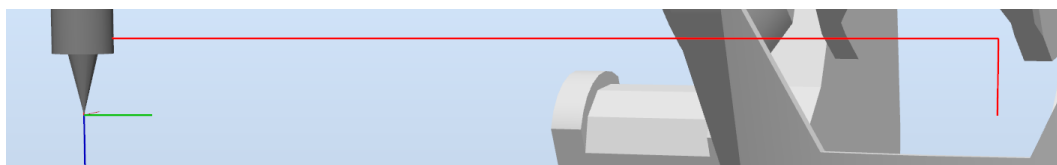


Figura 4.24: Trajetória (linha vermelha) realizada pelo robô ao executar o percurso *standard* da Epson por movimento linear.

Tabela 4.10: Resultados obtidos para o percurso 300x25cm de *pick and place* com um movimento de juntas nos 6 ciclos.

Sistema	Zone	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo Total ¹ (J)
Real	Fine	1800	100/100	0,87	2904
			20/100	1,31	3369
		1000	100/100	0,93	2686
			20/100	1,31	3325
	z10	1800	100/100	0,41	2027
			20/100	0,91	2522
		1000	100/100	0,50	1852
			20/100	0,92	2535
Simulador	Fine	1800	100/100	0,77	1199
			20/100	1,19	831
		1000	100/100	0,78	943
			20/100	1,19	815
	z10	1800	100/100	0,40	991
			20/100	0,90	781
		1000	100/100	0,47	817
			20/100	0,90	760

Tabela 4.11: Resultados obtidos para o percurso 300x25cm de *pick and place* através de um movimento linear nos 6 ciclos.

Sistema	Zone	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo Total ¹ (J)
Real	Fine	1800	100/100	0,95	2712
			20/100	1,31	3744
		1000	100/100	0,99	2660
			20/100	1,31	3743
	z10	1800	100/100	0,41	1996
			20/100	0,95	2904
		1000	100/100	0,48	1856
			20/100	0,94	2915
Simulador	Fine	1800	100/100	0,72	1170
			20/100	1,19	871
		1000	100/100	0,78	905
			20/100	1,19	824
	z10	1800	100/100	0,41	1029
			20/100	0,90	764
		1000	100/100	0,47	811
			20/100	0,90	746

¹ Ao longo de 6 ciclos

Apesar deste ser um teste *standard*, tendo em conta o tamanho e o espaço de trabalho do IRB 2400, pareceu redutor utilizar um percurso tão diminuto, visto que pode induzir em erros devido às pequenas zonas de aceleração e a zona de aproximação pode não ser realizada. Desta forma, decidiu-se testar outro percurso, mas com o dobro das dimensões do anterior, ou seja, 600 mm na horizontal e 50 mm na vertical, como apresentado na figura 4.25.

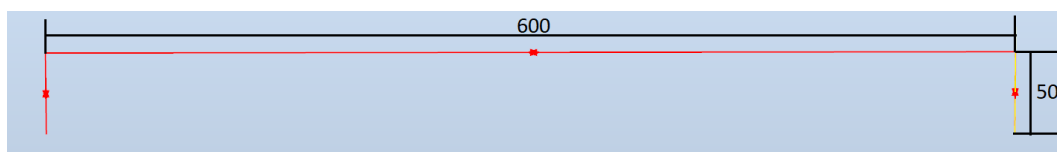


Figura 4.25: Segundo percurso para tarefas de *Pick and Place*.

Os parâmetros de definição da trajetória foram mantidos os mesmo que foram utilizados no primeiro trajeto de teste para tarefas de *pick and place*. Assim, os resultados para o percurso com uma interpolação por movimento de juntas estão apresentados na tabela 4.12, e para o movimento linear na tabela 4.13. As trajetórias executadas pelo robô para cada um dos tipos de interpolação de movimento estão apresentadas nas figuras 4.26 e 4.27.

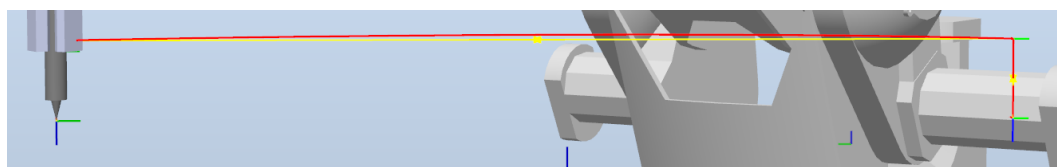


Figura 4.26: Trajetória (linha vermelha) realizada pelo robô para o segundo percurso de tarefas *Pick and Place* por movimento de juntas.

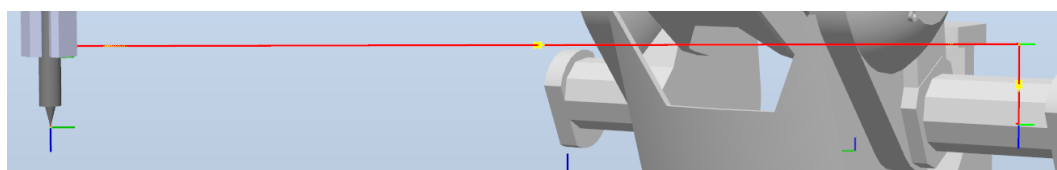


Figura 4.27: Trajetória (linha vermelha) realizada pelo robô para o segundo percurso de tarefas *Pick and Place* por movimento de linear.

Tabela 4.12: Resultados obtidos para o percurso 600x50cm de *pick and place* com um movimento de juntas nos 6 ciclos.

Sistema	Zone	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo Total ¹ (J)
Real	Fine	1800	100/100	1,27	4487
			20/100	2,11	6167
		1000	100/100	1,63	5073
			20/100	2,17	6416
	z10	1800	100/100	0,90	3434
			20/100	1,76	5219
		1000	100/100	1,22	4162
			20/100	1,85	5580
Simulador	Fine	1800	100/100	1,16	2414
			20/100	1,98	1960
		1000	100/100	1,51	2038
			20/100	2,05	1950
	z10	1800	100/100	0,86	2206
			20/100	1,66	1908
		1000	100/100	1,20	1868
			20/100	1,76	1877

Tabela 4.13: Resultados obtidos para o percurso 600x50cm de *pick and place* com um movimento linear nos 6 ciclos.

Sistema	Zone	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo Total ¹ (J)
Real	Fine	1800	100/100	1,28	4568
			20/100	2,11	6029
		1000	100/100	1,61	5051
			20/100	2,19	6363
	z10	1800	100/100	0,89	3437
			20/100	1,73	5195
		1000	100/100	1,21	4142
			20/100	1,80	5173
Simulador	Fine	1800	100/100	1,19	2487
			20/100	1,98	1952
		1000	100/100	1,49	2117
			20/100	2,04	1919
	z10	1800	100/100	0,86	2291
			20/100	1,66	1888
		1000	100/100	1,20	2017
			20/100	1,73	1905

¹ Ao longo de 6 ciclos

4.2.4.2 Simulação de percursos de soldadura ponto a ponto

Tal como os processos de *pick and place*, a soldadura é uma das principais aplicações dos robôs industriais. Assim, procurou-se encontrar uma trajetória normalizada destes processos em robôs industriais, mas neste caso, não existia nenhuma indicação nos catálogos dos diferentes fabricantes.

Como tal, projetou-se um percurso de uma soldadura ponto a ponto feita na vertical no trajeto descendente, e uma outra idêntica, mas executada na horizontal, de forma a perceber as possíveis diferenças. O espaçamento entre os pontos, foi sempre o mesmo, de 50 mm no eixo dos z no referencial *wobj0*, ao longo de 8 pontos de soldadura. Para além disto, programou-se uma paragem de 1s em cada um dos pontos, como se estivesse a realizar a soldadura, através da instrução, em código RAPID, **WaitTime\InPos,1**.

Neste caso, os parâmetros de programação que foram alterados foram apenas a velocidade de execução (100, 300 e 500 mm/s) e a aceleração do movimento (*AccSet* 20/20 e 100/100). O parâmetro *zone* é obrigatoriamente *fine* porque é necessária a paragem em cada um dos pontos de soldadura. Como tal, para cada um dos percursos foram realizadas 6 trajetórias, executadas tanto no IRB2400 como no simulador do RobotStudio.

Para o percurso de soldadura vertical descendente, o percurso executado está apresentado na figura 4.28, seguida dos respetivos resultados em termos de consumo energético e tempo de execução, encontram-se na tabela 4.14.

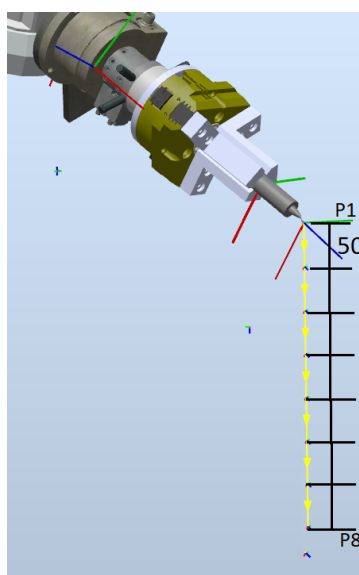


Figura 4.28: Trajetória de simulação de uma soldadura na vertical descendente.

Tabela 4.14: Resultados obtidos para o percurso de soldadura num trajeto vertical descendente nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)
Real	500	100/100	11,0	13167
		20/20	12,8	14797
	300	100/100	11,5	13431
		20/20	12,7	14410
	100	100/100	12,8	15508
		20/20	13,5	15089
Simulador	500	100/100	10,5	2148
		20/20	11,7	2097
	300	100/100	11,8	2203
		20/20	12,5	2249
	100	100/100	14,4	2554
		20/20	14,9	2596

Como já referido, o mesmo percurso foi executado, alterando apenas a disposição dos pontos, que em vez de se encontrarem posicionados na vertical, no segundo teste de uma tarefa de soldadura estão posicionados na direção horizontal. Desta forma, o percurso realizado pelo robô está apresentado na figura 4.29, e os respetivos resultados de consumo energético e tempo de execução estão apresentados na tabela 4.15.

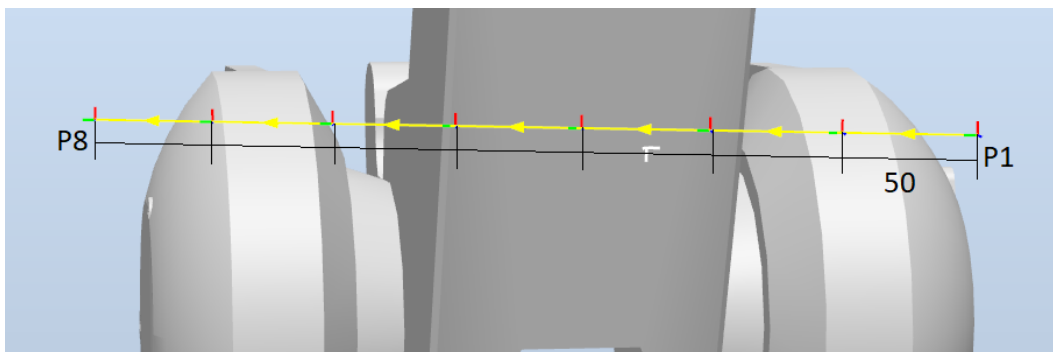


Figura 4.29: Trajetória de simulação de uma soldadura no plano horizontal.

¹ Ao longo de 3 ciclos

Tabela 4.15: Resultados obtidos para o percurso de soldadura num plano horizontal nos 3 ciclos.

Sistema	Velocidade (mm/s)	Aceleração	Tempo de ciclo (s)	Consumo total ¹ (J)
Real	500	100/100	11,0	11707
		20/20	13,1	13971
	300	100/100	11,6	12277
		20/20	12,9	13626
	100	100/100	12,8	13613
		20/20	13,6	14283
Simulador	500	100/100	10,4	1434
		20/20	12,2	1317
	300	100/100	11,7	1317
		20/20	12,9	1376
	100	100/100	14,3	1467
		20/20	15,1	1528

Neste capítulo, foram apresentados os testes realizados para o estudo do impacto dos diferentes parâmetros de programação de robôs industriais no consumo energético e no tempo de execução dos diferentes percursos. Inicialmente, o estudo focou-se no momento em que o robô se encontra parado, para de seguida a realização de alguns movimentos simples do robô, com o movimento de apenas uma junta de funcionamento, para gradualmente, aumentar a complexidade do movimento do robô, até às simulações de algumas tarefas em que normalmente o IRB2400 é utilizado.

Em cada um dos testes apresentados, foram expostos os resultados obtidos para o consumo energético durante o percurso, bem como o tempo de execução do robô na realização das diferentes trajetórias. No capítulo seguinte, procede-se ao estudo e interpretação desses resultados, com o objetivo de perceber o impacto de cada um dos parâmetros de programação no consumo energético do IRB2400.

¹ Ao longo de 3 ciclos

Capítulo 5

Discussão dos resultados obtidos

No capítulo 4 foram apresentados os percursos de teste elaborados para o estudo do impacto dos parâmetros de programação no consumo energético. Ao longo da exposição desses percursos foram exibidos os resultados do consumo energético do IRB2400, medido pelo PM5320, quando executado na célula robótica, e pelo *Signal Analyzer* quando o percurso foi realizado no simulador do RobotStudio. Para além do consumo energético, apresentou-se para cada um desses percursos o tempo de execução da trajetória.

No presente capítulo, esses resultados são analisados, com o objetivo de perceber qual o impacto dos parâmetros de programação no consumo energético e no tempo de execução de cada um dos percursos especificados. Na interpretação destes resultados, não se pode dissociar os parâmetros de programação do percurso, pois cada um deles tem particularidades diferentes, levando a que o custo/benefício do mesmo parâmetro possa ser diferenciado em cada um dos trajetos. Assim, a análise dos resultados está organizada segundo o percurso que foi executado.

5.1 Movimento angular da junta 1

Na definição dos primeiros percursos de teste, realizaram-se movimentos simples, de forma, a perceber mais facilmente o impacto dos parâmetros de programação implementados, bem como, para relacionar os resultados obtidos no simulador com os da célula robótica.

No caso das trajetórias de movimento da junta 1, os parâmetros analisados foram a velocidade de execução e a aceleração programada, não se podendo dissociar os dois, pois influenciam-se um ao outro. A análise dos resultados da tabela 4.2 inicia-se pela interpretação do impacto da velocidade no tempo de ciclo e no consumo total de energia, para a trajetória à aceleração máxima (*AccSet* 100/100), tanto para a célula robótica, como para o simulador, com a elaboração dos gráficos das figuras 5.1 e 5.2.

Da figura 5.1, verifica-se que tanto no caso da célula robótica como nos re-

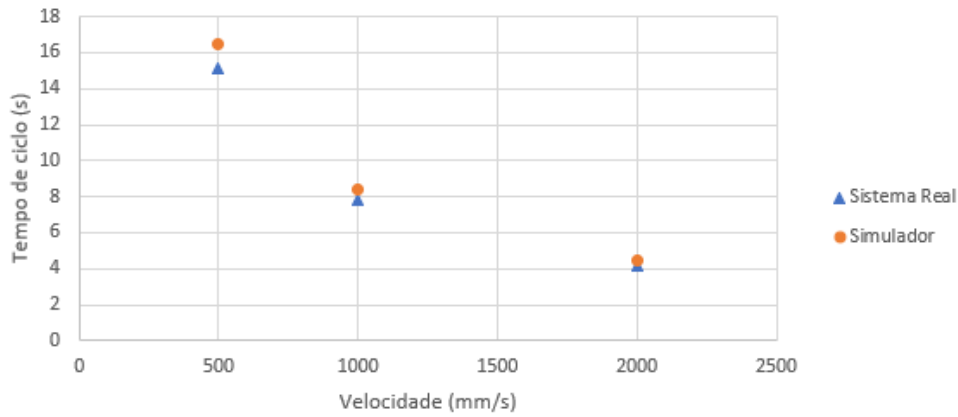


Figura 5.1: Tempo de ciclo médio em função da velocidade de execução para o movimento da junta 1.

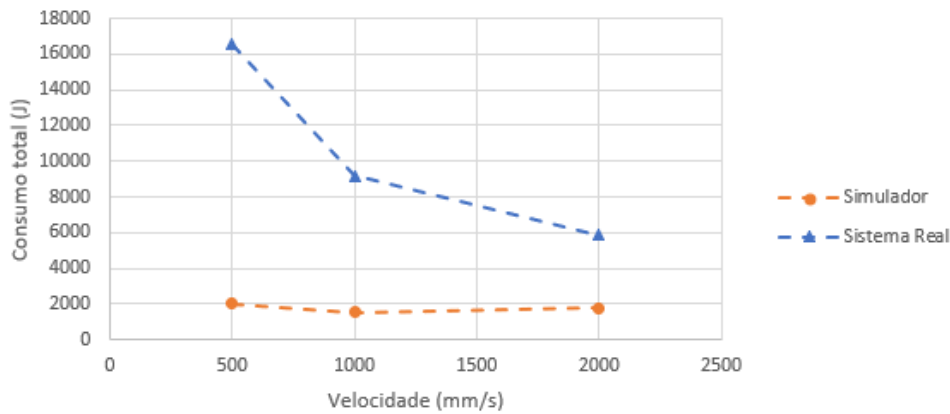


Figura 5.2: Consumo total de energia em função da velocidade de execução para o movimento da junta 1.

sultados obtidos no simulador, o tempo de ciclo tem tendência a diminuir de igual forma, à medida que a velocidade de execução aumenta. Para além disso, denota-se que o tempo medido na simulação se vai aproximando do tempo medido na célula robótica, com o aumento da velocidade. Desta forma, para a velocidade de 500 mm/s, a diferença entre os dois é de 1,3 segundos, e para 2000 mm/s é de apenas 0,2 segundos.

Quanto ao consumo de energia, apresentado na figura 5.2, denota-se uma elevada diferença entre as duas curvas. Este resultado era expectável, visto que a medição do consumo não ocorre no mesmo local. Para célula robótica o consumo é referente à energia fornecida pela rede ao controlador, e no simulador refere-se à energia consumida pelos motores em funcionamento, desprezando o consumo dos componentes elétricos do controlador. Ainda assim, os resultados obtidos pelo simulador não são condizentes com aquilo que se verificou na célula, visto que no simulador a trajetória com menor consumo energético é para a velocidade de 1000

mm/s e na célula é para 2000 mm/s, e a redução do consumo na célula robótica, entre cada uma das velocidades, é muito significativa, o que não se verifica no simulador.

Na análise do custo/benefício entre o tempo de ciclo e o consumo total de energia, verifica-se que a melhor solução é para a velocidade de 2000 mm/s, visto induz menor consumo e menor tempo de ciclo. De facto, pode-se afirmar que, para os casos dos resultados experimentais obtidos, o tempo de ciclo e o consumo energético diminuem à medida que a velocidade aumenta, pois, a velocidade de 2000 mm/s apresenta uma redução de consumo de 44% e 65%, e de tempo de ciclo de, 48% e 72%, em relação à velocidade de 1000 mm/s e 500 mm/s, respetivamente.

Quanto ao parâmetro de aceleração, os resultados obtidos foram dispostos em dois gráficos de barras, na figura 5.3 para o tempo de ciclo e na figura 5.4 para o consumo energético.

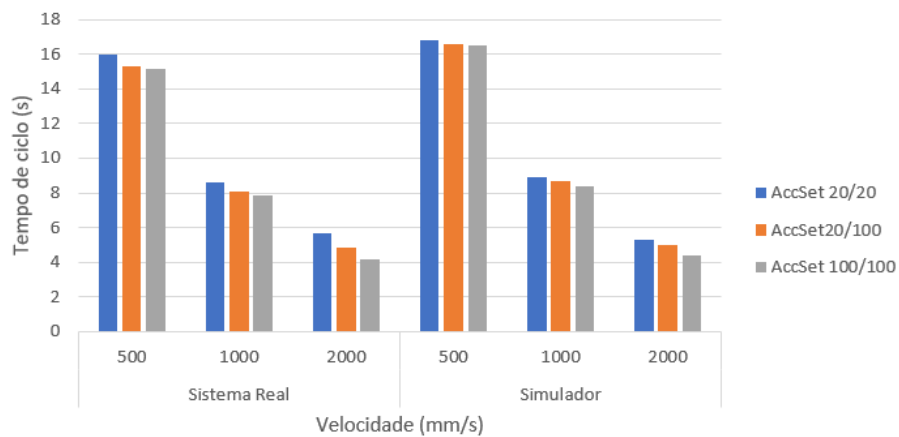


Figura 5.3: Resultados do tempo de ciclo no movimento da junta 1 com diferentes velocidades e acelerações.

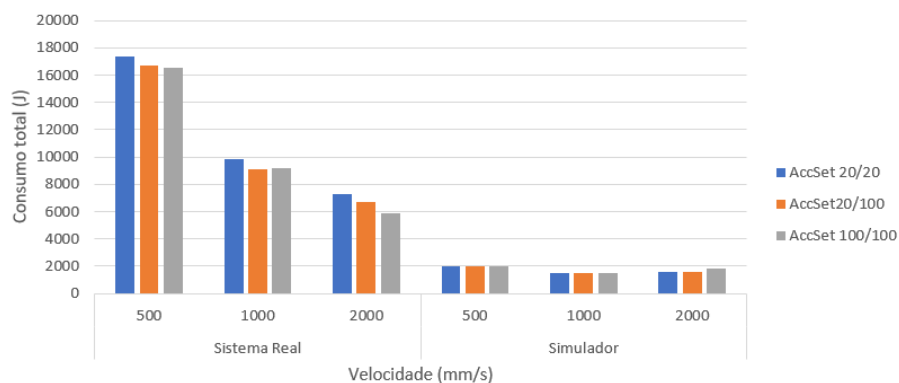


Figura 5.4: Resultados do consumo energético do percurso horizontal com diferentes velocidades e acelerações.

Mais uma vez denota-se que o tempo de ciclo medido no simulador é muito aproximado aos resultados obtidos na célula robótica, enquanto que para consumo

de energia volta-se a verificar que os resultados entre o simulador e a célula do laboratório são bastante diferentes. No caso da célula robótica, o consumo energético diminui, na maioria das trajetórias, com aumento da velocidade e da aceleração, o que não se pode concluir dos resultados do simulador, onde, por exemplo, para a velocidade de 2000 mm/s, à aceleração mais baixa (*AccSet* 20/20) o consumo de energia é superior em relação à aceleração intermédia (*AccSet* 20/100), mas inferior em relação à aceleração máxima (*AccSet* 100/100).

Para a análise do custo/benefício do parâmetro de aceleração, no tempo de ciclo e no consumo energético, elaborou-se a tabela 5.1, em que se relaciona as acelerações 20/20 e 20/100 com a aceleração máxima (*AccSet* 100/100), para cada uma das velocidades.

Tabela 5.1: Análise da variação do tempo de ciclo e do consumo de energia das trajetórias com parâmetros de aceleração mais baixos no movimento da junta 1, em relação à aceleração máxima, no sistema real.

Velocidade (mm/s)	Aceleração (<i>AccSet</i>)	Variação do tempo de ciclo (%)	Variação do consumo (%)
500	20/20	6%	5%
	20/100	1%	1%
1000	20/20	10%	7%
	20/100	3%	-1%
2000	20/20	35%	25%
	20/100	15%	14%

Da tabela 5.1, é facilmente perceptível que quanto menor a aceleração, maior o consumo de energia e o tempo de ciclo. Esta conclusão é suportada para a quase totalidade dos resultados da tabela, onde apenas a variação de consumo entre a aceleração 20/100 e a aceleração máxima, à velocidade de 1000 mm/s, é negativa. Apesar disso, a diferença é pouco significativa (1%), e provoca um aumento do tempo de ciclo (3%), revelando que na análise custo/benefício seria ainda preferível utilizar a trajetória de aceleração máxima.

5.2 Movimento angular da junta 3

No caso do movimento da junta 3, tal como para a junta 1, o parâmetros analisados foram a velocidade e a aceleração, com os valores que estes parâmetros tomam também a serem iguais. Desta forma, a análise para este movimento é em tudo idêntica à que foi feita anteriormente. Assim, dos resultados da tabela 4.3, relativos à aceleração máxima, elaborou-se os dois gráficos das figuras 5.5 e 5.6, para estudar o impacto da velocidade no consumo energético e no tempo de ciclo

para este percurso.

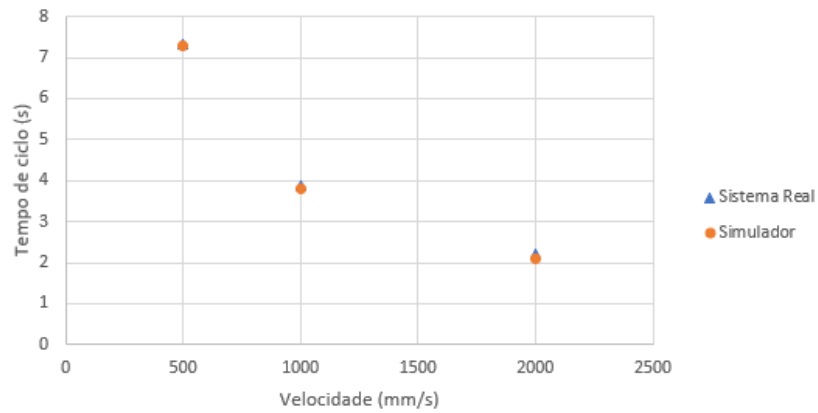


Figura 5.5: Tempo de ciclo em função da velocidade de execução para o movimento angular da junta 3.

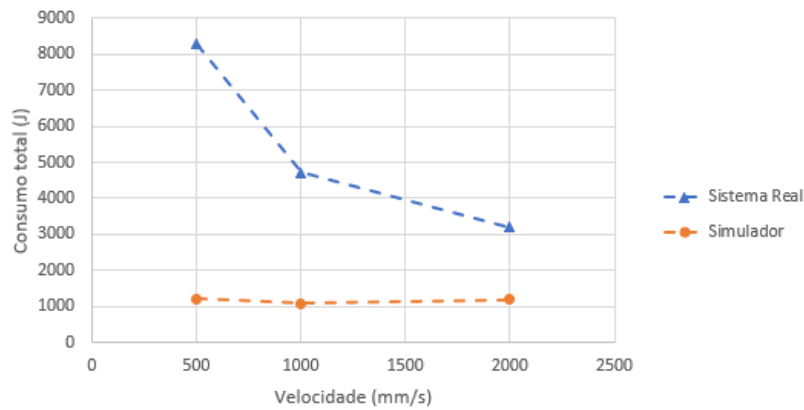


Figura 5.6: Consumo total de energia em função da velocidade de execução para o movimento da junta 3.

Ao analisar o gráfico da figura 5.5, verifica-se que para qualquer uma das velocidades de execução, neste percurso os tempos de ciclo tanto na célula robótica como no simulador são praticamente coincidentes. Isto indica que a simulação, para este percurso, em termos do tempo de ciclo é bastante próxima da realidade.

Do gráfico da figura 5.6, verifica-se mais uma vez que os resultados do simulador não estão de acordo com o que se mediu na célula do laboratório. Na célula do laboratório, verifica-se que à medida que a velocidade aumenta, existe uma redução significativa no consumo de energia, o que não acontece nas medições do simulador, porque, como para o movimento da junta 1, o consumo de energia é superior para a velocidade de 2000 mm/s em relação à velocidade de 1000 mm/s.

Quanto ao custo/benefício do parâmetro de velocidade, verifica-se, novamente, que com o aumento da velocidade, reduz-se o consumo de energia e o tempo de

execução da trajetória.

Para a análise do custo/benefício do parâmetro de aceleração, no tempo de ciclo e no consumo de energia, elaborou-se a tabela 5.2, em que se relaciona as trajetórias de aceleração inferior (20/20 e 20/100) com as de aceleração máxima (100/100), para cada velocidade.

Tabela 5.2: Análise da variação do tempo de ciclo e do consumo de energia das trajetórias com parâmetros de aceleração mais baixos no movimento da junta 3, em relação à aceleração máxima, no sistema real.

Velocidade (mm/s)	Aceleração (AccSet)	Variação do tempo de ciclo (%)	Variação do consumo (%)
500	20/20	7%	6%
	20/100	1%	0%
1000	20/20	16%	13%
	20/100	4%	7%
2000	20/20	47%	42%
	20/100	17%	7%

Da tabela 5.2, conclui-se que quanto maior for a aceleração, menor o tempo de ciclo e menor o consumo de energia. Para além disso, é ainda possível verificar que as diferenças de consumo de energia e de tempo de ciclo, entre os parâmetros de aceleração, aumentam, com o aumento da velocidade.

Para verificar se estes resultados, para o parâmetro de aceleração, são compatíveis com os resultados do simulador, geraram-se os dois gráficos apresentados nas figuras 5.7 e 5.8, para o tempo de ciclo e consumo total, respetivamente.

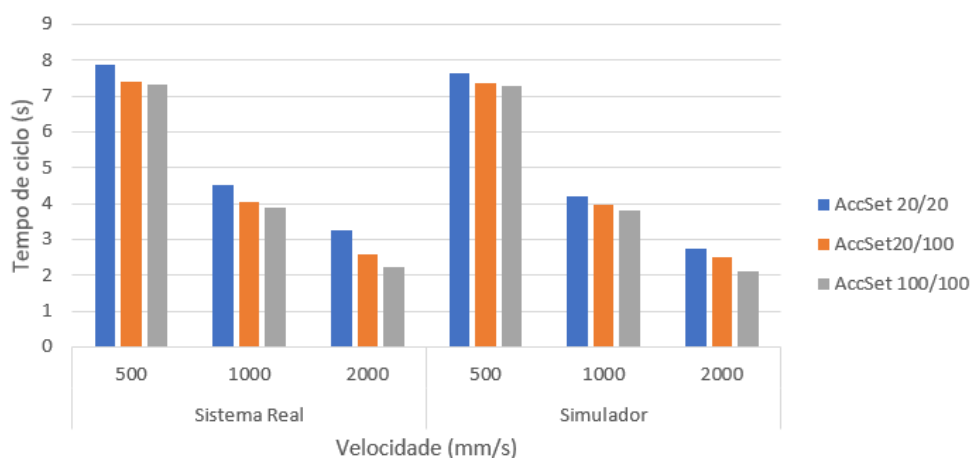


Figura 5.7: Comparação dos diferentes parâmetros de aceleração no tempo de ciclo na execução percurso vertical

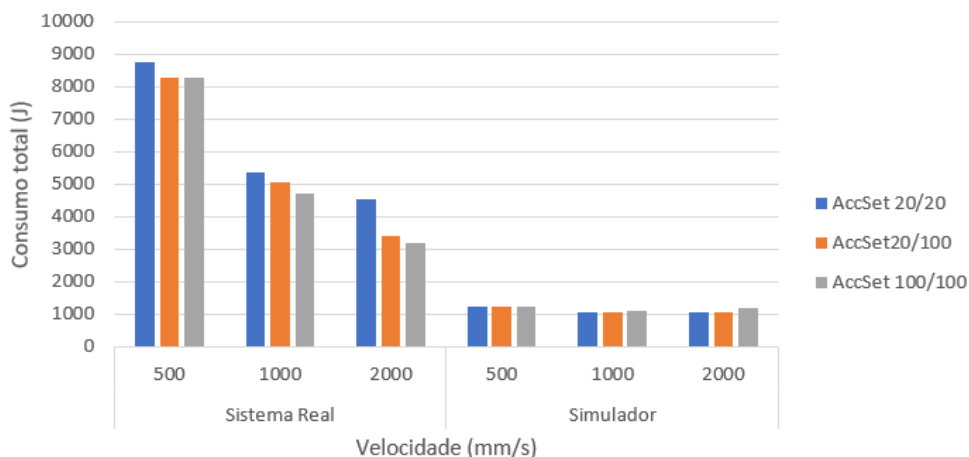


Figura 5.8: Comparação dos diferentes parâmetros de aceleração no consumo energético na execução percurso vertical.

Neste percurso, volta-se a verificar que os resultados obtidos no simulador são muito idênticos àqueles que se obtiveram na célula real, e principalmente, indicam exatamente a mesma tendência com o aumento da velocidade e da aceleração.

No que diz respeito ao gráfico do consumo energético, volta-se a concluir que os resultados do simulador não são compatíveis com os da célula do laboratório. Por exemplo, nos resultados experimentais, o menor consumo de energia dá-se para a trajetória executada a 2000 mm/s e aceleração máxima (*AccSet* 100/100), trajetória essa, que nos resultados do simulador é uma das que apresenta consumo mais elevado.

5.3 Movimento da junta 1 e 3

Depois de executar o movimento da junta 1 e da junta 3 em separado, decidiu-se realizar um percurso com a combinação desses dois movimentos. Visto que este teste surgiu da adição dos dois anteriores, os parâmetros de programação que foram avaliados foram os mesmo.

Como tal, para a análise dos resultados da tabela 4.4, apresentam-se os gráfico da figura 5.9, que se referem ao tempo de ciclo obtido para este percurso.

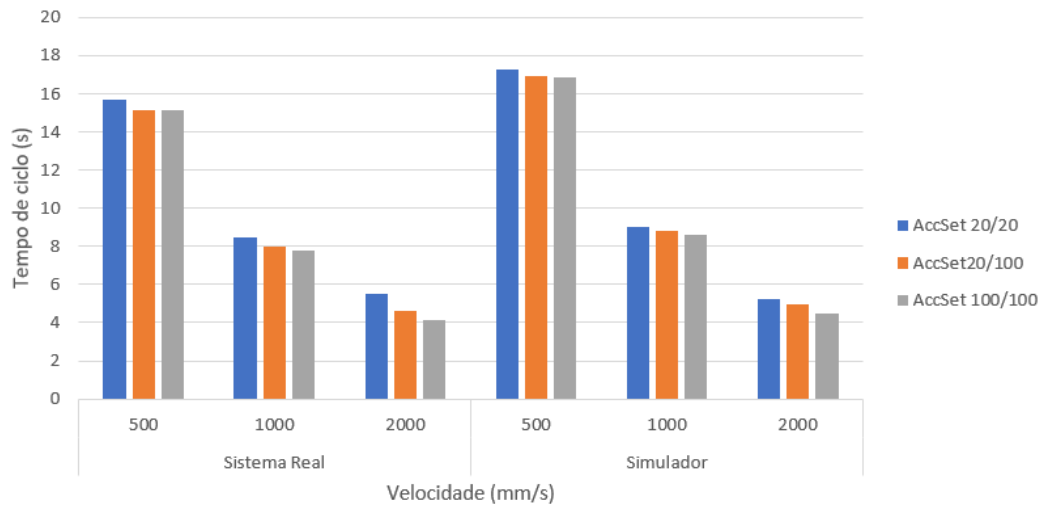


Figura 5.9: Comparação dos diferentes parâmetros de aceleração no tempo de ciclo na execução percurso da junta 1 e 3.

Ao interpretar a figura 5.9, verifica-se uma diferença entre os tempos de ciclo obtidos no simulador e na célula robótica. Analisando diretamente o gráfico, conclui-se que no simulador os tempos de ciclo são ligeiramente superiores aos medidos na consola do IRB2400 do laboratório, embora essas diferenças correspondam, no máximo, a um erro de 12%. Apesar disso, a tendência dos resultados obtidos, tanto no simulador como os resultados do sistema real é idêntica, sendo esse o principal objetivo das medições do RobotStudio, que representem uma aproximação do que pode acontecer na realidade.

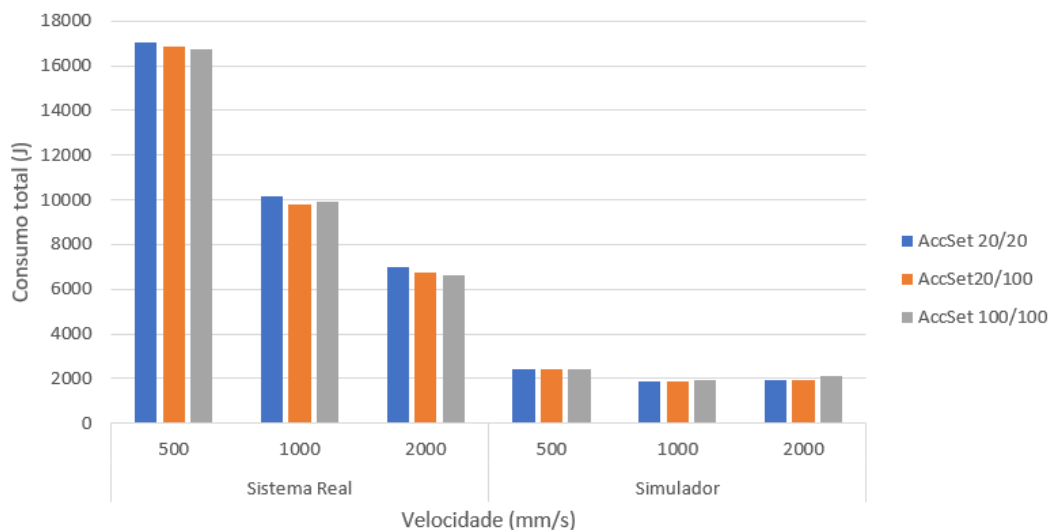


Figura 5.10: Comparação dos diferentes parâmetros de aceleração no consumo energético na execução do percurso da junta 1 e 3.

Para o consumo energético, elaborou-se a figura 5.10, onde se pode concluir

que os resultados do simulador não corresponde com aquilo que se mediu na célula do laboratório. Isso verifica-se pela análise dos resultados do simulador, em que todos os consumos de energia, para a velocidade de 1000 mm/s, são inferiores que aos da velocidade de 2000 mm/s, e na célula robótica, os consumos para a velocidade de 2000 mm/s são muito inferiores em relação às velocidades inferiores. Uma conclusão que é verificável nos dois resultados é que os maiores consumos de energia se dão para a velocidade de 500 mm/s.

Para além destes dois gráficos, foi construída a tabela 5.3, onde se apresenta a relação custo/benefício do tempo de execução e do consumo energético, para os valores medidos em todas as trajetórias executadas no IRB2400 da célula do laboratório, em relação à trajetória realizada à aceleração máxima (*AccSet* 100/100), para cada velocidade.

Tabela 5.3: Análise da variação do tempo de ciclo e do consumo de energia das trajetórias com parâmetros de aceleração mais baixos no movimento da junta 1 e 3, em relação à aceleração máxima, no sistema real.

Velocidade (mm/s)	Aceleração (AccSet)	Variação do tempo de ciclo (%)	Variação do consumo (%)
500	20/20	4%	2%
	20/100	0%	1%
1000	20/20	9%	3%
	20/100	3%	-1%
2000	20/20	34%	6%
	20/100	12%	3%

Neste caso, também para o percurso da junta 1 e 3, se retira como conclusão que com o aumentar da velocidade e da aceleração, o tempo de ciclo e o consumo de energia tendem a diminuir.

5.4 Localização no espaço de trabalho

Um dos momentos em que o consumo energético de um robô industrial é mais preocupante, verifica-se quando este se encontra parado à espera de alguma ação na célula robótica. Por esse motivo, realizou-se um teste onde se procurou perceber se o local do espaço de trabalho onde ocorre a paragem tem impacto no consumo energético. Para além disso, para o mesmo local avaliou-se o consumo de energia com duas orientações do *target* do ponto.

Como já referido no capítulo 4, para esse efeito utilizaram-se 4 vértices do cubo ISO, com o robô em duas configurações diferentes em cada um desses vértices. Assim, dos dados da tabela 4.5, gerou-se o gráfico de barras apresentado na figura

5.11, onde se pode analisar os consumos em cada uma das posições. Neste caso, avalia-se a potência média, porque não existe uma trajetória realizada, ou seja, não há um tempo de execução associado, avaliando-se o consumo em cada segundo.

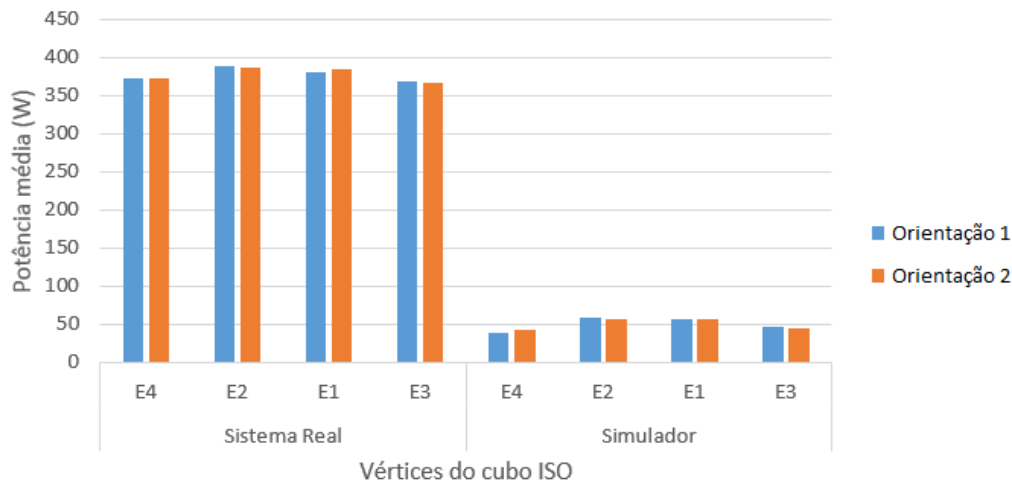


Figura 5.11: Comparação do consumo energético nos diferentes vértices do cubo ISO.

Ao analisar a figura 5.11, conclui-se que é no vértice E_3 que o robô apresenta menor consumo energético e no vértice E_2 o maior consumo. Outra das conclusões retirada da figura 5.11, é que, embora o simulador não tenha em conta o consumo do controlador, as variações no consumo energético medido com *Signal Analyzer* são similares com as obtidas na célula do laboratório, apesar de valores absolutos mais baixos.

5.5 Diagonais interiores do cubo ISO

Na apresentação, no capítulo 4, do teste das diagonais interiores do cubo ISO, referiu-se que um dos principais objetivos da realização deste ensaio, seria entender o impacto de diferentes tipos de interpolações de movimento no consumo energético e no tempo de execução do percurso. Para além disto, outro dos parâmetros que se pode avaliar é a execução do mesmo percurso (as duas diagonais são iguais em comprimento), mas em zonas do espaço de trabalho e direção da trajetória diferentes, o que pode revelar-se bastante impactante no consumo energético. Por fim, como em todas as trajetórias, também as velocidades de execução e aceleração do movimento são avaliadas, logo todos os restantes parâmetros estão dependentes destes.

Desta forma, decidiu-se designar como diagonal 1, a diagonal entre E_1 - E_3 , e diagonal 2, entre E_2 - E_4 , para serem mais facilmente identificáveis. Assim sendo, nesta secção, a análise dos resultados apresentados no capítulo 4, inicia-se pela elaboração de um gráfico de barras, apresentado na figura 5.12, onde se coloca,

do lado esquerdo, o tempo de ciclo medido na célula do laboratório em função da interpolação do movimento, da velocidade de execução e da aceleração do movimento . O mesmo acontece, do lado direito, só que neste caso apresenta-se o consumo energético medido no robô, em função dos três parâmetros programados.

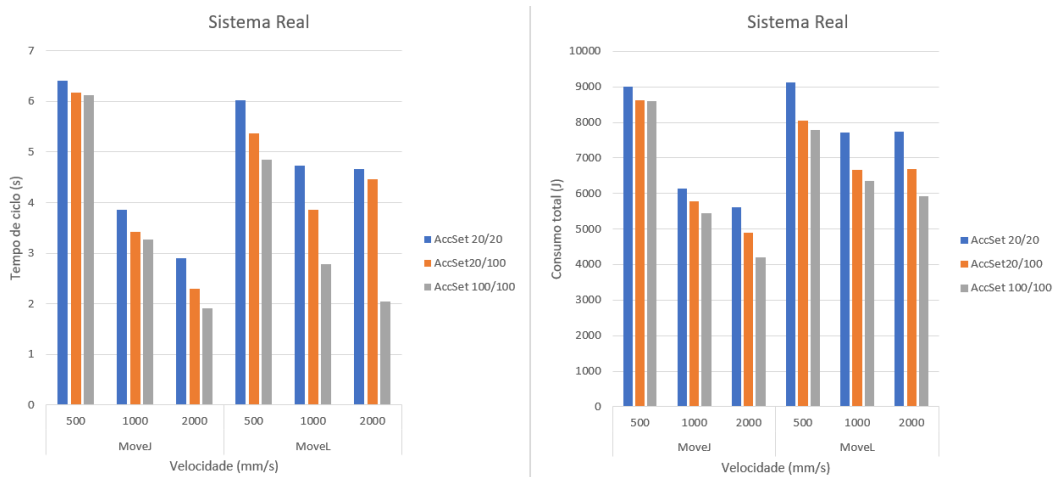


Figura 5.12: Análise custo/benefício dos resultados obtidos para o percurso da diagonal 1 na célula do laboratório.

Da interpretação da figura 5.12, verifica-se que a trajetória que apresentou menor consumo e menor tempo de ciclo foi quando se executou o percurso com o movimento de juntas, velocidade de 2000 mm/s e aceleração máxima (100/100). Neste caso, é de relevar o tipo de interpolação para o qual se obtém menor consumo, visto que os resultados da velocidade e aceleração são os esperados, tendo em conta os resultados anteriores. Para esse parâmetro verifica-se que, para a maioria das trajetórias estudadas, o movimento de juntas induz menores tempos de ciclo e menores consumos de energia. Para o consumo de energia, verifica-se que a trajetória, realizada em movimento linear, com menor consumo é para a velocidade máxima (2000 mm/s) e aceleração máxima (100/100), embora tenha um consumo superior a 5 trajetórias realizadas através de movimento de juntas (as três de velocidade 2000 mm/s e duas de velocidade de 1000 mm/s). Na relação com a trajetória homóloga, realizada por movimento de juntas, constata-se uma redução de 6% no tempo de ciclo e de 29% no consumo de energia. O movimento linear apresenta-se como uma solução com melhores resultados para a velocidade mais baixa (500 mm/s), visto que tem menores tempo de ciclo e, para as duas trajetórias de maior aceleração, menores consumos.

De seguida, procedeu-se à análise dos resultados da tabela 4.6 e 4.7, quando as trajetórias foram executadas no simulador. Para tal, elaborou-se um gráfico idêntico ao anterior, apresentado na figura 5.13, com o objetivo de se comparar os dois e perceber se é possível retirar as mesmas conclusões.

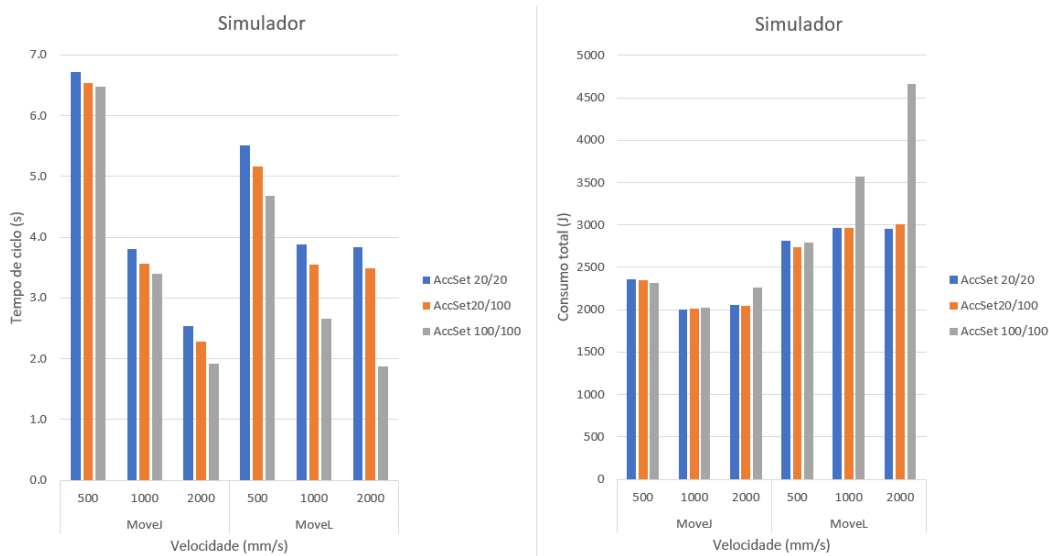


Figura 5.13: Tempo de ciclo e consumo energético das trajetórias da diagonal 1 realizadas no simulador.

Dos resultados obtidos no simulador, verifica-se que, para o tempo de ciclo, a aproximação apresenta a mesma tendência verificada no gráfico dos resultados da célula do laboratório. Quanto ao consumo de energia, embora se possa concluir que o movimento linear tem um consumo de energia superior ao movimento de juntas, os resultados do simulador não são condizentes com os apresentados para a célula robótica, visto que, por exemplo, todas as trajetórias executadas por movimento linear têm consumo superior às realizadas por movimento de juntas.

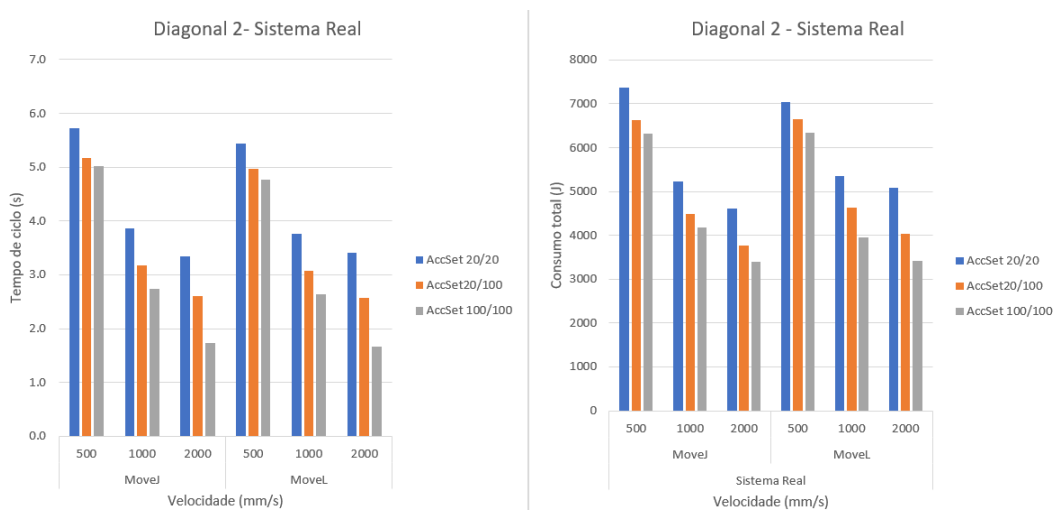


Figura 5.14: Análise dos resultados das diferentes trajetórias da diagonal 2 na célula robótica.

A mesma análise foi realizada para os resultados da diagonal 2, que para a célula robótica estão apresentados na figura 5.14. Ao comparar os resultados desta diagonal, com os da diagonal 1, conclui-se que o consumo energético reduz para todas

as trajetórias efetuadas, independentemente da velocidade, aceleração, ou tipo de interpolação utilizado. Também para o tempo de ciclo, a diagonal 2, na maioria dos resultados, apresenta uma diminuição neste parâmetro, indicando que o local do espaço de trabalho onde se executa a trajetória e a direção desse movimento apresentam uma preponderância no custo/benefício estudado.

Para além disso, verifica-se que o impacto da alteração do tipo de interpolação de movimento utilizado é atenuado para esta segunda diagonal, embora o movimento de juntas continue a apresentar a trajetória com menor tempo de ciclo e menor consumo de energia (velocidade 2000 mm/s e *AccSet* 100/100). Neste percurso, observa-se que, na maioria das situações, o tempo de ciclo é inferior para as trajetórias realizadas por movimento de juntas, ainda que as diferenças sejam pouco significativas para as mesmas velocidades e acelerações (0,3 segundos, no máximo). Também no consumo energético, as diferenças entre os dois tipos de interpolação de movimento são mais reduzidas que para a diagonal 1, visto que, comparando trajetórias com a mesma velocidade e aceleração, a maior diferença é de aproximadamente 10%, apresentando o movimento de juntas menor consumo, como na maioria dos casos.

Neste caso é também relevante avaliar os resultados obtidos no simulador do RobotStudio para o percurso da diagonal 2 do cubo ISO, como apresentado na figura 5.15. Para este percurso, volta-se a verificar que o simulador é capaz de aproximar o tempo de ciclo medido na célula robótica, mas para o consumo energético os resultados são muito diferentes dos observados ao nível experimental.

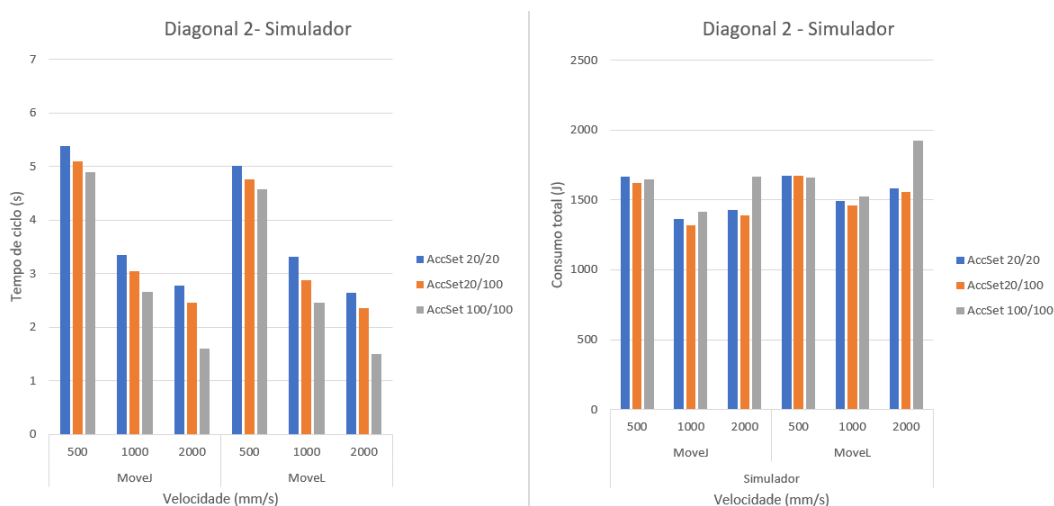


Figura 5.15: Análise dos resultados das diferentes trajetórias da diagonal 2 no simulador do RobotStudio.

5.6 Tarefas de *pick and place*

Nas diferentes trajetórias de tarefas de *pick and place*, procurou-se determinar o impacto de 4 parâmetros de programação (velocidade de execução, aceleração, *zone* e interpolação do movimento) no tempo de execução e no consumo energético destes percursos. De ressaltar, que o parâmetro *zone* é apenas avaliado nestes testes de *pick and place*, o que implica que a análise nestes ensaios recaia mais sobre esse parâmetro.

Assim sendo, avaliou-se os dados das tabelas 4.10 e 4.11 e elaborou-se a figura 5.16, onde se apresenta o consumo energético e o tempo de ciclo em função da velocidade, da aceleração, do parâmetro *zone* e do tipo de interpolação na execução do percurso da figura 4.22.

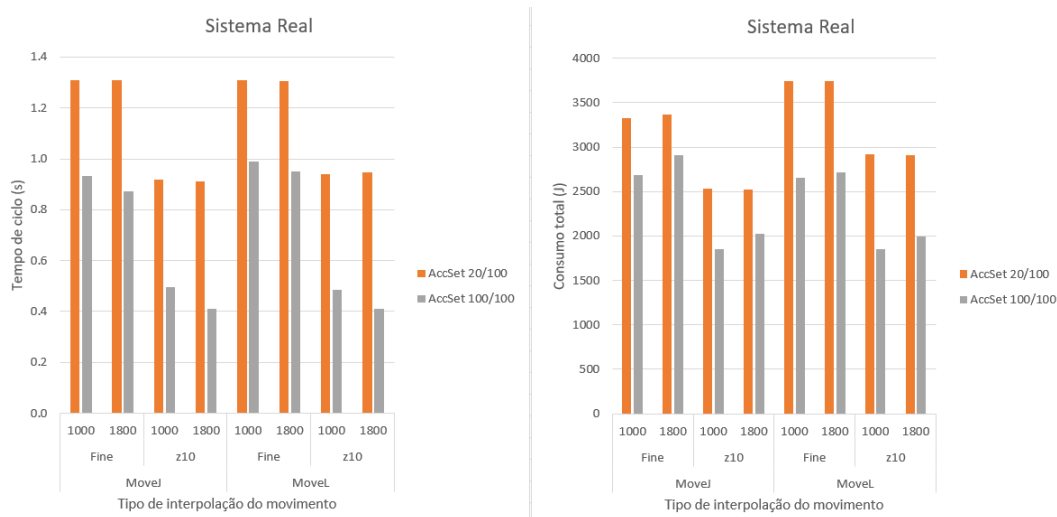


Figura 5.16: Análise dos resultados das diferentes trajetórias do primeiro percurso de *pick and place*.

Do gráfico da figura 5.16, continua a verificar-se a tendência da diminuição do tempo de ciclo com o aumento da velocidade e da aceleração. Apesar disso, para este percurso, deixa de se verificar que o menor consumo se dê para as velocidades mais elevadas, porque, quer para interpolação de movimento de juntas, quer para o movimento linear, a trajetória de menor consumo dá-se para a velocidade de 1000 mm/s. Nessas trajetórias, a condição de aceleração é a mesma (*AccSet* 100/100), mas para atingir a velocidade de 1800 mm/s, o robô despende mais tempo em zona de aceleração, em relação à velocidade de 1000 mm/s, o que dadas as dimensões reduzidas do trajeto, pode aumentar a preponderância desse tempo de aceleração, provocando um maior consumo.

O tipo de interpolação de movimento utilizado neste percurso tem um impacto é pouco significativo no tempo de ciclo e no consumo de energia. Em relação ao tempo de ciclo, os resultados são muito semelhantes entre os dois, para as mesmas

condições de velocidade, aceleração e *zone*. As diferenças mais significativas dão-se para o consumo energético, principalmente nas trajetórias de aceleração mais baixa (*AccSet* 20/100), onde o movimento de juntas apresenta menor consumo de energia: em média 13%, para a mesma velocidade e *zone*.

Quanto ao parâmetro *zone*, fica evidente da análise do gráfico, que ao executar a trajetória com uma zona de aproximação de 10 mm (*z10*), resulta um menor consumo de energia e menor tempo de ciclo, do que quando se utiliza o *fine*. Este resultado seria expectável, pois o parâmetro *fine* obriga o robô a parar no ponto programado, implicando maiores acelerações e desacelerações em relação ao *z10*. Para verificar esta conclusão, realizou-se, no simulador, duas trajetórias deste percurso, em que o único parâmetro variável entre elas foi o *zone*, recorrendo-se à funcionalidade do *Signal Analyzer* para traçar o perfil de velocidades apresentado na figura 5.17.

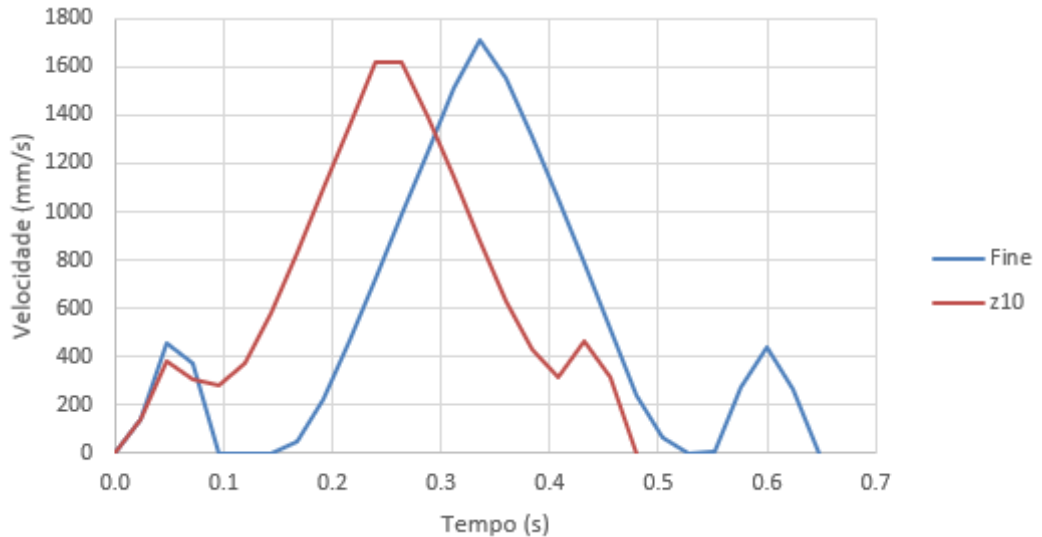


Figura 5.17: Perfil de velocidades (MoveJ) com dois parâmetros *zone* diferentes.

Como se verifica nos perfis de velocidades, a trajetória programada com o parâmetro *fine* está sempre em aceleração e desaceleração, o que não acontece quando usado *z10*, onde se verifica que o robô se movimentou a velocidade constante, embora durante um curto espaço de tempo. A figura 5.18, apresenta a trajetória do elemento terminal do robô (TCP) para diferentes parâmetros *zone* e tipo de interpolação do movimento.

Como se verifica pela análise da figura 5.18(gráfico da direita), as trajetórias executadas com o parâmetro *z10*, não implicam a passagem nas posições intermédias do percurso (B e C na figura 4.22).

Dadas as reduzidas dimensões do primeiro percurso de *pick and place*, relativamente ao espaço de trabalho e à dimensão do robô, realizou-se um segundo percurso

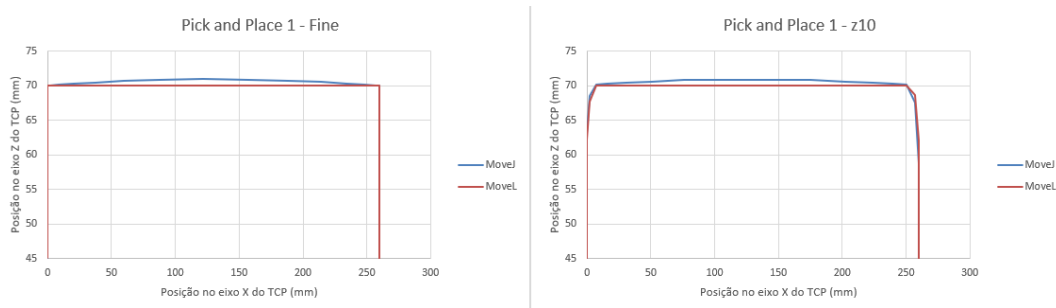


Figura 5.18: Posição do TCP para as duas trajetórias (MoveJ e MoveL) com diferentes parâmetros *zone* (*fine* e *z10*).

de *pick and place*, com o dobro das dimensões do primeiro (figura 4.25).

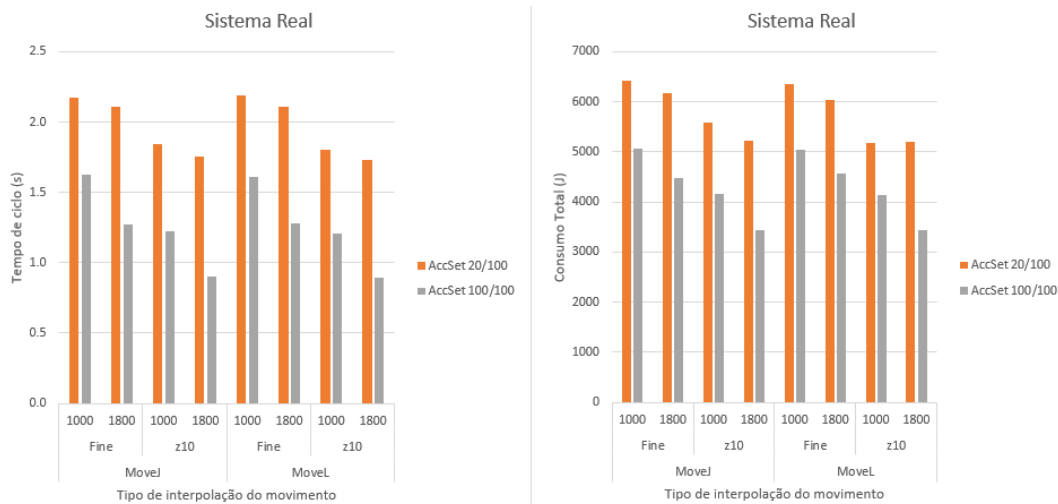


Figura 5.19: Resultados de tempo de ciclo e consumo de energia para segundo percurso de testes de *pick and place*.

Os resultados obtidos para segundo percurso de *pick and place* são apresentados nos gráficos da figura 5.19. Da análise desta figura, verifica-se que o tipo de interpolação de movimento utilizado tem pouco impacto no consumo de energia. A utilização do parâmetro *fine* implica sempre maior consumo de energia, independentemente da velocidade considerada ou da aceleração. O menor consumo, neste percurso, ocorre para a velocidade máxima (1800 mm/s) e aceleração máxima (*AccSet* 100/100), contrariamente ao que acontecia no percurso menor (figura 5.16), em que o menor consumo ocorre para a velocidade mais baixa (1000 mm/s). Uma possível justificação deste facto pode decorrer de, no percurso de maior dimensão, o tempo de ciclo ser superior.

5.7 Percursos de soldadura

O percurso de soldadura, apresentado no capítulo 4, foi executado em duas direções, uma na vertical e outra na horizontal, em dois locais distintos do espaço de trabalho. Para cada um dos percursos, avaliou-se o impacto da velocidade de execução e da aceleração do movimento, no tempo de ciclo e no consumo de energia da trajetória. Para análise do tempo de ciclo, elaboraram-se os gráficos das 5.20 e 5.21, para a soldadura vertical e horizontal, respetivamente, tendo sido considerados da célula real e de simulação.

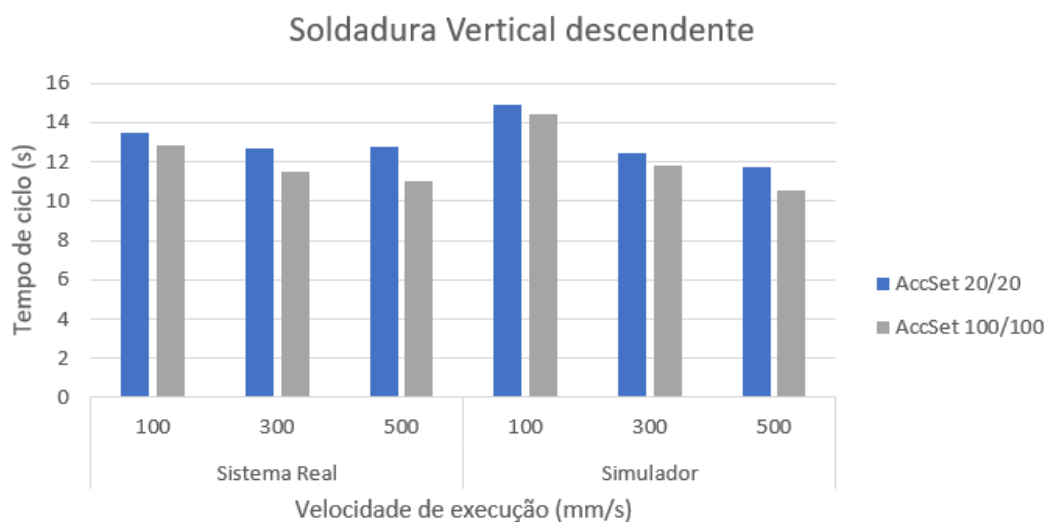


Figura 5.20: Tempo de ciclo das diferentes trajetórias no percurso de soldadura na vertical.

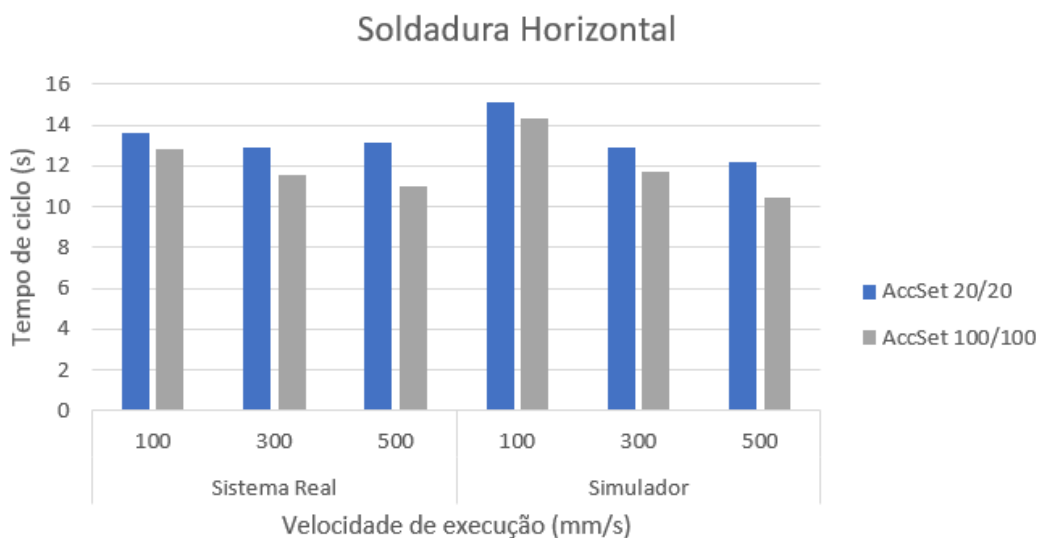


Figura 5.21: Tempo de ciclo das diferentes trajetórias no percurso de soldadura na horizontal.

Da análise das figuras 5.20 e 5.21, verifica-se que o tempo de ciclo é praticamente independente da direção da soldadura. Este resultado era previsível, dado que, neste percurso, o robô fica parado 1 segundo em cada ponto de soldadura, representando assim 8 segundos do tempo total de ciclo. Assim, o robô só está em movimento 3 a 6 segundos do tempo total do ciclo.

Os resultados do simulador, para ambos os percursos de soldadura, voltam a mostrar uma boa aproximação da tendência que se verifica nos resultados medidos na célula real. Neste caso, para as velocidades de 500 mm/s e 300 mm/s a aproximação do tempo de ciclo é sempre inferior a $\pm 10\%$ do valor medido na célula. No caso da velocidade de 100 mm/s, o tempo de ciclo medido pelo simulador apresenta um erro mais elevado, cerca de 12% superior ao medido na célula, para qualquer das acelerações especificadas.

O consumo de energia é apresentado nas figuras 5.22 e 5.23 para a soldadura vertical e horizontal, respetivamente, tendo sido considerados da célula real e de simulação.

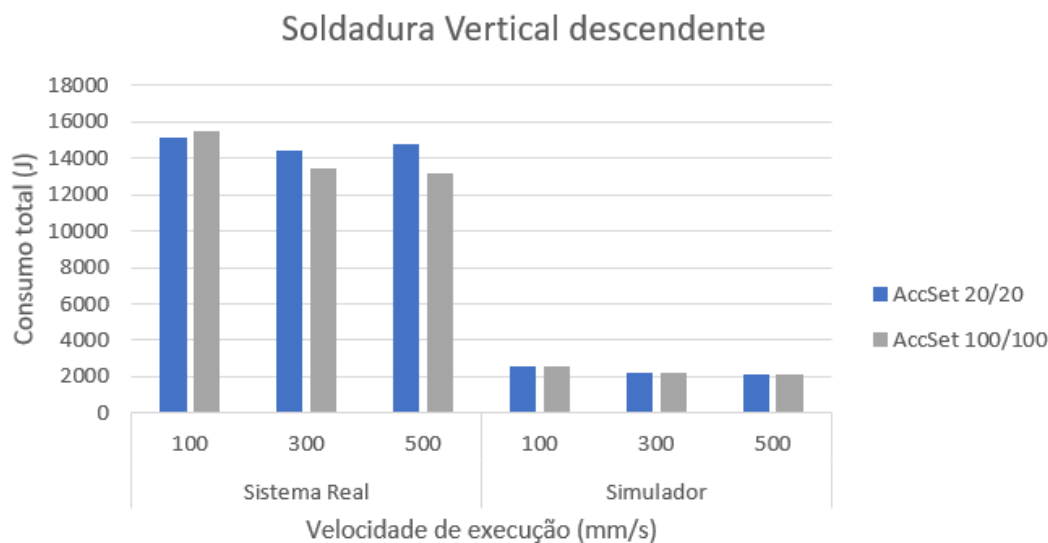


Figura 5.22: Consumo energético das diferentes trajetórias no percurso de soldadura na vertical.

Ao analisar os gráficos das figuras 5.22 e 5.23 verifica-se que para as mesmas condições de velocidade e aceleração, a soldadura na direção vertical tem um consumo superior ao da direção horizontal, identificado quer na célula real, quer no simulador.

Quanto ao impacto da velocidade de execução, verifica-se que, tendencialmente, o consumo de energia diminui com o aumento da velocidade. Os resultados do simulador, indicam, tal como nos resultados da célula robótica, uma diminuição no consumo de energia à medida que a velocidade de execução aumenta. Para a célula do laboratório, nas duas soldaduras, a trajetória executada à velocidade má-

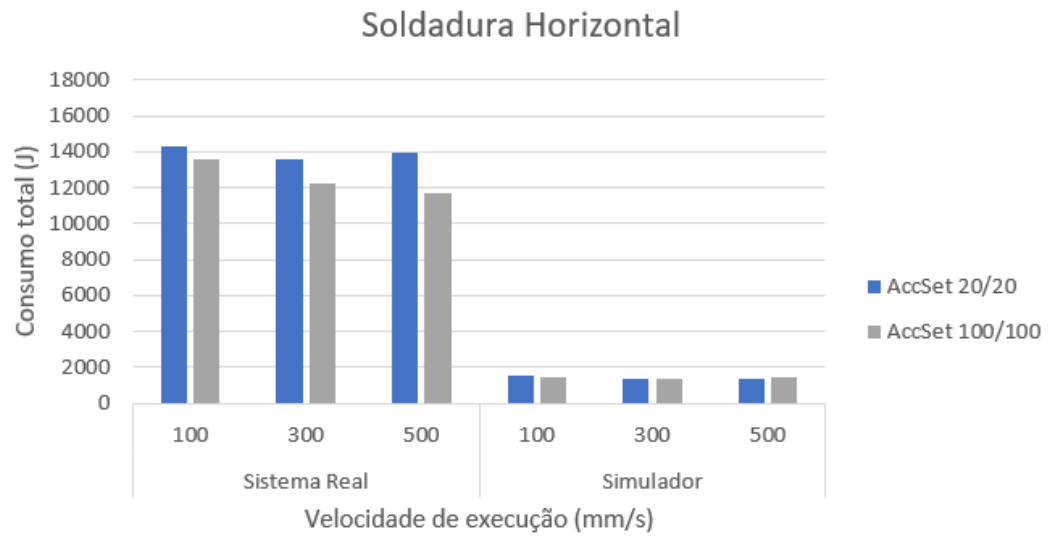


Figura 5.23: Consumo energético das diferentes trajetórias no percurso de soldadura na horizontal.

xima (500 mm/s) e à aceleração máxima (*AccSet* 100/100), resulta em menor tempo de ciclo e em menor consumo de energia. No entanto no simulador isso ocorre para a trajetória à velocidade máxima (500 mm/s) e aceleração mínima (*AccSet* 20/20), mas é preciso ter em conta que o simulador não inclui a energia gasta no controlador, o que pode condicionar os resultados obtidos.

Capítulo 6

Conclusões e Trabalhos Futuros

6.1 Conclusões

O consumo de energia elétrica de processos e equipamentos é um tema cada vez mais recorrente na indústria. No caso da robótica, a programação e parametrização dos percursos, por eles executados, é uma das principais fontes a analisar para aumentar a sua eficiência energética.

Na presente dissertação, estudou-se o impacto de diferentes parametrizações de trajetória no consumo energético de um robô industrial. Para tal, definiram-se uma série de percursos de teste que se realizaram a nível experimental, na célula robótica do laboratório, e ao nível da simulação, através do RobotStudio e da ferramenta *Signal Analyzer*.

A célula robótica está equipada com um robô industrial, o IRB2400, da ABB, e o respetivo controlador. Para a medição da energia elétrica foi necessário adquirir um medidor de potência, o PM5320 da Schneider, que foi implementado em série entre a rede elétrica e o controlador, com o objetivo de medir a energia fornecida.

Como o medidor de potência não disponibiliza um *software* que permita o registo direto das medições no computador, implementou-se um sistema de aquisição e monitorização de dados. Com esse intuito, a aquisição das medições utiliza a comunicação ModBUS TCP/IP, entre o PM5320 e o computador. Para este computador foi desenvolvida uma aplicação em Node-Red, que permite o processamento e registo dos dados. Para a monitorização das medições efetuadas conceberam-se duas interfaces gráficas que podem ser acedidas remotamente por qualquer dispositivo ligado à rede da FEUP. A primeira, desenvolvida em Node-Red, monitoriza as medições do PM5320 em tempo-real, enquanto a segunda, gerada no Grafana, permite a visualização e aquisição do histórico de medições.

Para a análise do impacto da parametrização da trajetória no consumo energético e no tempo de ciclo, foram analisados 5 parâmetros de programação: velocidade da trajetória, aceleração do movimento, tipo de interpolação do movimento, loca-

lização no espaço de trabalho e zona de aproximação ao ponto (*zone*). Ao longo dos diferentes percursos de teste, foi efetuada uma análise da influência de cada parâmetro no tempo de ciclo e no consumo de energia. Daí conclui-se que:

- apesar das versões do *software* RobotWare no controlador do robô (versão 5.12) e no simulador (versão 6.11) serem diferentes, na generalidade dos ensaios efetuados, os tempos de ciclo podem considerar-se praticamente iguais. Em relação ao consumo energético, identificou-se uma diferença substancial entre os valores simulados e os valores lidos com o medidor de energia. O que se pode considerar natural, uma vez que o consumo energético fornecido pelo simulador diz respeito à energia consumida nos motores;
- o aumento da velocidade da trajetória, conduz a reduções significativas no tempo de ciclo e no consumo de energia, na maioria das trajetórias executadas na célula real;
- parâmetros de aceleração, e variação de aceleração (*jerk*) mais elevados, diminuem o tempo de ciclo e o consumo de energia. O impacto do uso da instrução *AccSet* verificou-se ser mais significativo para trajetória executadas a velocidade mais elevadas, como visto na tabela 5.1;
- o impacto do tipo de interpolação, varia com a trajetória em execução, mas em geral, o movimento de juntas (*MoveJ*) apresenta menores tempos de ciclo e menores consumos de energia quando comparado com o movimento linear (*MoveL*). Este facto foi verificado quer a nível da célula real, quer a nível da simulação;
- a utilização de zonas de aproximação ao ponto (*z10*), nos percursos de *pick and place* realizados na célula robótica, resultaram em reduções no consumo energético (aprox. 22%) e no tempo de ciclo (aprox. 30%) quando comparado com a paragem obrigatória no ponto programado (*fine*);
- manter o robô imóvel em diferentes localizações do espaço de trabalho resulta em diferentes consumos de energia. Nos testes efetuados verificaram-se valores de, aproximadamente, 370 a 380W.

As conclusões que se obtiveram neste estudo não podem ser dissociadas dos percursos que foram realizados. O tema da eficiência energética em robôs industriais ainda tem muito por explorar, embora seja indiscutível que a parametrização da trajetória tem impacto na energia despendida. A utilização de um medidor de energia revelou-se importante para a medição e análise do consumo energético da célula robótica. Pode ainda ser útil para complementar a informação obtida com o

simulador que, na vertente energética apresenta uma limitação significativa ao não considerar o consumo do controlador do robô.

6.2 Trabalhos Futuros

Como trabalhos futuros sugere-se a instalação de um novo sistema no controlador do robô que permita aceder às novas funcionalidade de monitorização *online* no *Signal Analyzer*, do RobotStudio.

Outra sugestão é a realização de um conjunto de testes para identificar o consumo de base dos componentes elétricos do controlador do robô em diferentes situações de trabalho e com uma gama alargada de cargas.

Por outro lado, seria interessante explorar a possibilidade de ativar o travão eletromecânico em momentos de paragens longas, visto que o consumo energético quando o travão está ativo é bastante menor em relação aos momentos em que o robô está em controlo de posição.

Referências

- [1] “International federation of robotics.” <https://ifr.org/>. Acedido em Abril de 2021.
- [2] M. Brossog, M. Bornschlegl, J. Franke, et al., “Reducing the energy consumption of industrial robots in manufacturing systems,” The International Journal of Advanced Manufacturing Technology, vol. 78, no. 5-8, pp. 1315–1328, 2015.
- [3] M. Todtermuschke, M. Findeisen, and A. Bauer, “Methodology for creation a reference trajectory for energetic comparability of industrial robots in body shop,” Procedia CIRP, vol. 23, pp. 122–126, 2014.
- [4] D. Meike and L. Ribickis, “Energy efficient use of robotics in the automobile industry,” in 2011 15th international conference on advanced robotics (ICAR), pp. 507–511, IEEE, 2011.
- [5] G. Carabin, E. Wehrle, and R. Vidoni, “A review on energy-saving optimization methods for robotic and automatic systems,” Robotics, vol. 6, no. 4, p. 39, 2017.
- [6] K. Paes, W. Dewulf, K. Vander Elst, K. Kellens, and P. Slaets, “Energy efficient trajectories for an industrial abb robot,” Procedia Cirp, vol. 15, pp. 105–110, 2014.
- [7] M. Pellicciari, G. Berselli, F. Leali, and A. Vergnano, “A method for reducing the energy consumption of pick-and-place industrial robots,” Mechatronics, vol. 23, no. 3, pp. 326–334, 2013.
- [8] “Energy efficiency – european commision.” <http://ec.europa.eu/energy/en/topics/energy-efficiency>. Acedido em Abril de 2021.
- [9] A. Liu, H. Liu, B. Yao, W. Xu, and M. Yang, “Energy consumption modeling of industrial robot based on simulated power data and parameter identification,” Advances in Mechanical Engineering, vol. 10, no. 5, p. 1687814018773852, 2018.
- [10] R. Saidur, “A review on electrical motors energy use and energy savings,” Renewable and sustainable energy reviews, vol. 14, no. 3, pp. 877–898, 2010.

- [11] Y. Li and G. M. Bone, “Are parallel manipulators more energy efficient?,” in Proceedings 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation (Cat. No. 01EX515), pp. 41–46, IEEE, 2001.
- [12] A. G. Ruiz, J. V. Fontes, and M. M. da Silva, “The influence of kinematic redundancies in the energy efficiency of planar parallel manipulators,” in ASME 2015 International Mechanical Engineering Congress and Exposition, American Society of Mechanical Engineers Digital Collection, 2015.
- [13] A. Othman, K. Belda, and P. Burget, “Physical modelling of energy consumption of industrial articulated robots,” in 2015 15th International Conference on Control, Automation and Systems (ICCAS), pp. 784–789, IEEE, 2015.
- [14] D. Meike, M. Pellicciari, G. Berselli, A. Vergnano, and L. Ribickis, “Increasing the energy efficiency of multi-robot production lines in the automotive industry,” in 2012 IEEE International Conference on Automation Science and Engineering (CASE), pp. 700–705, IEEE, 2012.
- [15] D. Meike, M. Pellicciari, and G. Berselli, “Energy efficient use of multirobot production lines in the automotive industry: Detailed system modeling and optimization,” IEEE Transactions on Automation Science and Engineering, vol. 11, no. 3, pp. 798–809, 2013.
- [16] G. Wittenberg, “Developments in offline programming: an overview,” Industrial Robot: An International Journal, 1995.
- [17] ABB, Product specification IRB 2400. ABB, 2019.
- [18] C. Connolly, “Technology and applications of abb robotstudio,” Industrial Robot: An International Journal, 2009.
- [19] “Pm5320.” <https://www.se.com/il/en/>. Acedido em Maio de 2021.
- [20] S. Electric, PowerLogic™ PM5300 series User manual. Schneider Electric, 2021.
- [21] “Mariadb.” <https://mariadb.com/>. Acedido em Maio de 2021.
- [22] “Grafana.” <https://grafana.com/>. Acedido em Maio de 2021.
- [23] P. Ystgaard, T. B. Gjerstad, T. K. Lien, and P. A. Nyen, “Mapping energy consumption for industrial robots,” in Leveraging Technology for a Sustainable World, pp. 251–256, Springer, 2012.
- [24] A. Robotics, “Product specification irb 2400,” 2021.

REFERÊNCIAS

- [25] E. Robots, ROBOT SPECIFICATIONS 2020 CATALOG. Epson Robots, 2020.
An optional note.

Anexos

A Programa RAPID da Interface da consola

```
WHILE (TRUE) DO
    TPReadFK eixo,"escolha de eixo a mover","eixo1","eixo2","eixo3","sold","sair";
    IF eixo=5 THEN
        quit;
    ENDIF

    TPReadFK vel,"escolha de velocidade","v1800","V300","V500","definir","sair";

    IF vel=1 THEN
        myvel:=speed2000;
    ENDIF
    IF vel=2 THEN
        myvel:=speed300;
    ENDIF
    IF vel=3 THEN
        myvel:=speed500;
    ENDIF

    IF vel=4 THEN
        vel_definida:=UINumEntry(\header:="inserção e velocidade"
        \message:="qual a velocidade em mm/s ?"
        \icon:=iconInfo
        \minvalue:=1
        \maxvalue:=3500
        \asinteger);
        IF vel_definida<>3500 THEN
            per:=Abs(Round(1-(vel_definida/3500)*100));
            myvel:=speed600;
            myvel.v_tcp:=vel_definida;
            !Velset per,vel_definida;
```

```
ELSE
    per:=100;
    myvel:=speed600;
    myvel.v_tcp:=vel_definida;
ENDIF
ENDIF
IF vel=5 THEN
    quit;
ENDIF

TPReadFK ac,"escolha de acelerações","100/100","20/100","100/20","20/20","sair";

IF ac=1 THEN
    AccSet 100,100;
ENDIF
IF ac=2 THEN
    AccSet 20,100;
ENDIF
IF ac=3 THEN
    AccSet 100,20;
ENDIF

IF ac=4 THEN
    AccSet 20,20;
ENDIF
IF ac=5 THEN
    quit;
ENDIF

TPReadNum no,"numero repetições";

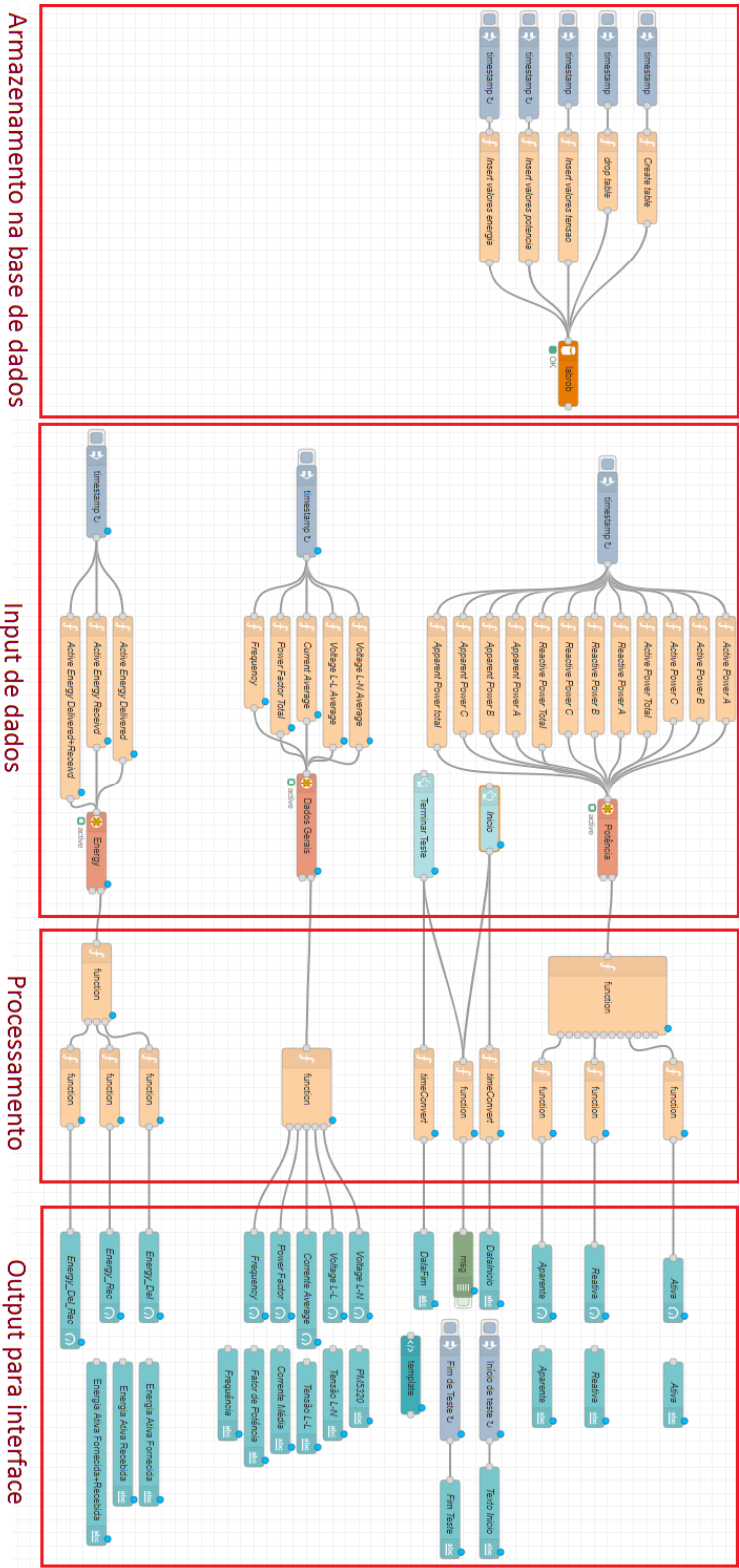
WHILE no<>0 DO
    IF eixo=1 THEN

        TPWrite " mov eixo1 ";
        TPWrite "velocidade de execução "\num:=vel;
        TPWrite "ac de execução "\num:=ac;

        ClkReset clock22;
        ClkStart clock22;
        FOR i FROM 1 TO no DO
            Path_100;
        ENDFOR

        ClkStop clock22;
        tempo_percurso:=ClkRead(clock22);
        TPErase;
        TPWrite "tempo de execução "\num:=tempo_percurso;
    ENDIF
```

B Programa desenvolvido em Node-Red



C Equipamentos de medição de energia elétrica estudados

Fabricante		Schneider		Panasonic		Delta			Siemens		Carlo Gavazzi	
Produto	PM5320	PM5111	AKW91110	AKW264100A	DPM-C530	DPM-C530E	DPM-C520W	7KM3133-RTU	7KM3133-TCP/IP	WM40 96	WM30 96	
Parâmetros medidos ⁽¹⁾	Todos	Todos	Todos	Todos	Todos	Todos	Todos	Todos	Todos	Todos	Todos	
Tensão de alimentação	100...415 V AC 45...65 Hz	100...415 V AC 45...65 Hz	85 to 264V AC	100-240V AC	80 ~ 265 V AC	80 ~ 265 V AC	80 ~ 265 V AC	100-240 V AC	95-240 V AC	100-240 V AC	100-240 V AC	
Frequência da rede	50 Hz 60 Hz	50 Hz 60 Hz	50 Hz 60 Hz	50 Hz 60 Hz	45 ~ 70 Hz	45 ~ 70 Hz	45 ~ 70 Hz	45-65 Hz	Deteção automática da frequência da rede	50/60 Hz	50/60 Hz	
Protocolo de comunicação	Modbus RTU	✓	✓	✓	✓		✓	✓	✓	✓	✓	
	Modbus ASCII		✓			✓						
	Modbus TCP/IP	✓			✓		✓		✓	Request		
	MEWTOTCOL			✓	✓							
	BACnet IP	✓								Request		
	BACnet MS/TP				✓							
	RS232									✓	✓	
	RS485		✓	✓	✓		✓	✓		✓		
	Interface Wireless						✓			Request		
Porta de comunicação suportada	Ethernet	✓		✓	✓	✓			✓	Request	Request	
	USB			✓								
Tipo de montagem	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	Calha DIN	
Memória interna				✓								