

31

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

**CONTROLO DA FORÇA DE CORTE NA MAQUINAÇÃO,
USANDO REDES NEURONAIS**

José Carlos Lobinho Gomes

Licenciado em Engenharia Electrotécnica e de Computadores
pela Faculdade de Engenharia da Universidade do Porto

Dissertação submetida para satisfação parcial dos
requisitos do grau de mestre
em

Engenharia Electrotécnica e de Computadores
(Área de especialização de Automação Industrial)

681.5 (043)
GOMJ/CON

043 7
GOMJ/CON

UNIVERSIDADE DO PORTO
Faculdade de Engenharia
BIBLIOTECA
N.º 7444
CDU
DATA 24 04 97

Dissertação realizada sob a supervisão de
Professor Doutor Francisco José de Oliveira Restivo,
do Departamento de Engenharia Electrotécnica e de Computadores
da Faculdade de Engenharia da Universidade do Porto

Porto, Novembro de 1996

AGRADECIMENTOS

Em primeiro lugar gostaria de agradecer aos meus pais o facto de me terem possibilitado a realização da Licenciatura e todo o incentivo e apoio que sempre me deram na prossecução dos meus estudos.

Agradeço ao meu orientador, o Prof. Dr. Francisco José de Oliveira Restivo a sua orientação e coordenação durante a realização dos trabalhos conducentes à elaboração desta Tese de Dissertação. Agradeço especialmente o esforço desenvolvido na leitura e nas sugestões de revisão que permitiram o enriquecimento do texto desta Tese.

Um Agradecimento muito especial à Lídia pela sua companhia e compreensão durante todo o tempo que durou o meu a elaboração desta Dissertação. Também pela sua preciosa colaboração e ajuda na edição e revisão da Tese.

Desejo também agradecer ao CCP - Centro de CIM do Porto, as facilidades que me concedeu e que permitiram a realização do trabalho conducente à elaboração desta Tese de Dissertação, assim como o facto de ter disponibilizado todos os meios materiais ao seu alcance que permitiram a realização deste trabalho.

Estou especialmente grato ao Sr. Guilhermino Coelho pela ajuda prestada na revisão dos textos pertencentes a esta Tese.

Agradeço também ao Sr. Manuel Alheiro pela ajuda que me prestou com a sua grande experiência do processo de maquinação.

Gostava também de agradecer aos colegas do Instituto Espanhol IAI - Instituto de Automatica Industrial, de Madrid, em especial ao colega Clodeinir Ronei Peres, pela disponibilidade para a troca de ideias.

Por último, desejo agradecer a todos os meus colegas do CCP, todas as trocas de ideias que tivemos e o óptimo ambiente de trabalho que têm criado. Um agradecimento muito especial ao Sr. Victor Monteiro, por todo o apoio dispensado durante o estágio.

A procura constante da indústria por incrementos de produtividade e cotas de mercado impelem o desenvolvimento de novos produtos capazes de dar resposta a estas preocupações. Concretamente os fabricantes de máquinas industriais têm tentado ultrapassar estes problemas incrementando as capacidades das máquinas que produzem essencialmente em velocidade e precisão

O recente estudo de alguns problemas associados ao processo de maquinação, revelou a possibilidade de incrementar a produtividade de algumas máquinas de fresagem vertical, unicamente através do controlo da força de corte, mantendo esta constante e igual ao valor óptimo definido para a ferramenta. O controlo da força de corte, devido às características do sistema, só pode ser implementado recorrendo ao controlo adaptativo.

Para implementar controladores adaptativos temos à disposição duas tecnologias que recentemente têm mostrado bons resultados. As tecnologias a que nos referimos são as Redes Neurais e a Logica *Fuzzy*.

Estando o IAI - Instituto de Automatica Industrial, de Madrid, envolvido num projecto de I&D cujo objectivo é controlar a força de corte, usando Lógica *Fuzzy*, entendeu-se que seria de investigar a utilização de Redes Neurais Artificiais na implementação de um controlador com o mesmo objectivo, de forma a comparar os resultados.

Foi este mesmo o objectivo da realização do trabalho descrito na presente Tese de Dissertação.

Os resultados obtidos foram animadores, demonstrando ser possível implementar tais controladores em sistemas reais.

The constant search of industry for productivity raises and market shares, pushes the development of new products capable of giving an answer to these concerns. Specifically the machine tools makers have tried to solve these problems incrementing the capability of the machines they produce, essentially in speed and precision.

The recent study of some problems associated to the machining process, has revealed the possibility of incrementing the productivity of some of vertical milling machine, only through the force control, keeping it constant and equal to the optimum value defined for the tool. The cutting force control, due to the system characteristics, can only be implemented by making use of adaptive control.

In order to implement adaptive controllers we have at our disposal two technologies that have recently been showing good results. These technologies are Neural Networks and Fuzzy Logic.

Since IAI - Instituto de Automatica Industrial, from Madrid, is involved in an R&D project, whose objective is the control of the cutting force, using Fuzzy Logic, we thought that it would be of interest to research the use of Artificial Neural Networks in implementation of a controller with the same objective, in order to compare the results.

This has been the objective of the development of the work described in this Dissertation Thesis.

The results obtained have been encouraging, showing the possibility of implementing those controllers in real systems.

Capítulo 1	17
1. Introdução	17
1.1 A máquina	18
1.2 Processo de corte	18
1.3 Processo de controlo	20
1.3.1 Dificuldades inerentes ao processo de corte .	20
1.4 Controlo da Força de Corte	24
1.5 Objectivos	25
1.6 Estrutura da tese	25
Capítulo 2	27
2. O problema da maquinação	27
2.1 Sumário	28
2.2 Tecnologia de corte	28
2.2.1 Força de corte	30
2.2.1.1 Força especifica de corte	30
2.2.1.2 Secção do corte	33
2.2.1.3 Calculo práctico da secção e respectiva força de corte	39
2.2.2 Força de avanço	44
2.2.3 Influencia da geometria da peça na força de corte	44
2.2.3.1 Variação da força de corte	46
2.2.3.2 Uso da ferramenta de geometria variável .	47
2.2.4 Influência do estado da ferramenta no corte .	48
2.3 Manter a força de corte constante	52
2.4 Conclusões	55
Capítulo 3	57
3. Sistema de Controlo	57
3.1 Sumário	58
3.2 O controlo	58
3.3 O controlo da força de corte	61

3.3.1 O Decisor	61
3.3.2 O Actuador	63
3.3.3 Controlo não linear	66
3.3.4 Variáveis de Controlo	70
3.4 Arquitectura do controlador	73
3.4.1 Principais problemas de controlo	74
3.4.2 O Actuador	74
3.4.3 Conclusões	76

Capítulo 4..... 77

4. Redes Neurais	77
4.1 Sumário	78
4.2 Redes Neurais	78
4.3 Evolução das Redes Neurais Artificiais	79
4.4 Biologicamente	81
4.4.1 Evolução do Conhecimento do Cérebro	81
4.4.1.1 O Neurónio	82
4.4.1.2 Estrutura do neurónio	82
4.4.1.3 Funcionamento do Neurónio	84
4.4.1.4 Modelo do Neurónio	84
4.5 Sistema Artificial	85
4.5.1 Algumas propriedades	86
4.5.1.1 Paralelismo	86
4.5.1.2 Capacidade de adaptação	86
4.5.1.3 Memória distribuída	87
4.5.1.4 Capacidade de generalização	87
4.5.1.5 Facilidade de desenvolvimento	87
4.5.2 Limitações das Redes Neurais	88
4.6 Redes Neurais. Teoria/Prática	88
4.6.1 Redes Neurais Artificiais - Fundamentos ...	88
4.6.1.1 Neurónios - unidades de processamento ...	89
4.6.1.2 Conexões entre neurónios	91
4.6.1.3 Regras e funções de activação da saída ..	92
4.6.2 Topologia das Redes Neurais Artificiais ...	93
4.6.3 Treino de Redes Neurais Artificiais	94
4.6.3.1 Paradigmas da aprendizagem	97
4.6.3.2 Regra para alteração dos pesos das conexões	97
4.7 Conclusões	98

Capítulo 5..... 99

5. Implementação	99
5.1 Sumário	100
5.2 Descrição	100

5.2.1 O Bloco de nível superior - Decisor	101
5.2.1.1 O Controlador do bloco Decisor	102
5.2.1.2 O Identificador do Bloco Decisor	105
5.2.2 O Bloco de nível inferior - Actuador	106
5.2.2.1 O Adaptador	108
5.2.2.2 O Identificador	114
5.2.2.3 O Controlador	121
5.2.2.4 O Sensor	125
5.2.2.5 Testes	126
5.3 Conclusões	129
Capítulo 6	131
6. Conclusões	131
6.1 Sumário	132
6.2 O problema	132
6.3 A tecnologia	133
6.4 A solução	135
6.5 Vantagens e desvantagens	137
6.6 O futuro	138
Referências	139
Apêndice A	141
7. <i>Script's</i> usados no MatLab	141
7.1 Valores usados no treino do Controlador do bloco Decisor	141
7.2 <i>Script</i> usado para treinar o Controlador do bloco Decisor	142
7.3 <i>Script</i> usado para testar o Controlador do bloco Decisor	143
7.4 <i>Script</i> usado para gerar a maior parte dos dados usados no treino das Redes Neurais Artificiais do bloco de nível inferior	143
7.5 <i>Script</i> usado para treinar o Identificador do bloco de nível inferior	144
7.6 <i>Script</i> usado para testar o Identificador	145
7.7 <i>Script</i> usado para treinar o Controlador do nível inferior	147
7.8 <i>Script</i> usado para testar o Controlador	148
7.9 <i>Scripts</i> usado para treinar/testar o Adaptador ..	149

7.9.1	estimar a velocidade de avanço	149
7.9.1.1	treinar	149
7.9.1.2	Testar	150
7.9.2	Estimar a velocidade de Rotação	150
7.9.2.1	Treinar	150
7.9.2.2	Testar	151
7.10	Script usado para treinar o conjunto (Controlador, Identificador e adaptador)	151
7.11	Seguem-se vários Scripts usados para criarem os gráficos que aparecem nesta tese	157
7.12	Script que permitiu armazenar os dados para treino da rede	162

Apêndice B..... 191

8.	Exemplo de algumas tabelas usadas	191
8.1	Fresa de 4 navalhas	191
8.2	Fresa de 2/3 navalhas	192
8.3	Roca de Esquadrar	193

Apêndice C..... 195

9.	Bibliografia	195
----	--------------------	-----

Figura 1.1 - Peça de teste onde se reproduz a maquinação clássica; e uma maquinação com força de corte controlada.....	19
Figura 1.2 - Máquina disponível no CCP, usada para realizar medidas de força de corte. .	21
Figura 1.3 - Exemplo de como pode aumentar a força de corte real ao longo de uma operação de maquinação	22
Figura 1.4 - Exemplo de como pode diminuir a força de corte ao longo de uma operação de maquinação	23
Figura 1.5 - Exemplo de como a força de corte pode variar mas sempre abaixo do valor optimo	23
Figura 2.6 - Trajectória da Ferramenta de Corte.....	29
Figura 2.7 - Secção da apara produzida no corte	34
Figura 2.8 - Secção da apara produzida no corte por uma ferramenta de várias arestas de corte	40
Figura 2.9 - Esquema de forças na ferramenta.....	43
Figura 2.10 - Exemplo da variação da geometria da peça ao longo da trajectória.....	45
Figura 2.11 - Exemplo do cálculo de custos para o corte da secção a tracejado.....	53
Figura 3.1 - Controlo de velocidade de uma máquina a vapor	59
Figura 3.2 - Controlador adaptativo genérico	60
Figura 3.3 - Diagrama de um controlador genérico	68
Figura 3.4 - Diagrama de blocos de um sistema de controlo onde se intruduz perturbações na malha de realimentação.....	69
Figura 3.5 - Controlador com com o bloco de identificação	70
Figura 3.6 - Velocidades de avanço e corte para materiais diferentes e espessuras de corte diferentes.....	72
Figura 3.7 - Diagrama de blocos do Controlador.....	75
Figura 4.1 - Esquema do Neuronio Biológico	82

Figura 4.2 - Esquema de comunicação entre neurónio - sinapses.....	83
Figura 4.3 - Representação do modelo de um neurónio	85
Figura 4.4 - Componentes básicos de uma rede neuronal (neurónios e respectivas ligações).....	89
Figura 4.5 - Rede neuronal genérica de dois níveis	90
Figura 4.6 - Funções normalmente usadas como activação nos neurónios.....	92
Figura 4.7 - Topologias de Redes Neurais Artificiais, a) " <i>Feed-forward</i> "; b) $1,2,3$ " <i>Recurrent</i> "	94
Figura 5.1 - Diagrama do Bloco Superior.....	102
Figura 5.2 - Rede neuronal representativa do controlador associado ao sistema Decisor ..	103
Figura 5.3 - uma possível estrutura para a Rede Neuronal Artificial que poderá implementar o Identificador	105
Figura 5.4 - Diagrama do bloco do nível inferior	107
Figura 5.5 - Estrutura da Rede Neuronal Artificial que implementa a parte do Adaptador que estima a velocidade de avanço	110
Figura 5.6 - Estrutura da Rede Neuronal Artificial que implementa a parte do Adaptador que estima a velocidade de rotação.....	112
Figura 5.7 - A constituição do bloco designado de Adaptador	113
Figura 5.8 - Diagrama genérico do Identificador.....	117
Figura 5.9 - Rede neuronal que implementa o Identificador	118
Figura 5.10 - Diagrama genérico do controlador	122
Figura 5.11 - Rede neuronal que implementa o Controlador.....	123
Figura 1.1 -Sistema de controlo baseado em redes neuronais	135
Figura 1.2 - Diagrama genérico do Controlador	136

Gráfico 2.1 - Variação da espessura da apara função do angulo de rotação para diferentes velocidades de avanço.....	36
Gráfico 2.2 - Variação da Força de Corte Função do angulo de inicio e fim	38
Gráfico 2.3 - Variação da espessura da apara função do angulo de rotação para diferentes velocidades de avanço, agora considerando o numero de dentes da fresa.....	40
Gráfico 2.4 - Variação da força de corte função do numero de dentes da ferramenta	42
Gráfico 2.5 - Variação da força de corte ao longo da trajetoria	46
Gráfico 2.6 - A força exercida pela ferramenta ao longo do seu tempo de maquinação	49
Gráfico 2.7 - Curva de Desgaste/Velocidade de corte para um tempo de corte constante de 60 minutos [3]	51
Gráfico 2.8 - Variação da Velocidade de avanço função da profundidade de corte, mantendo a força de corte constante	54
Gráfico 3.1- Exemplo da variação dos parâmetros que influenciam a força de corte	65
Gráfico 4.1- Exemplo de uma rede que apresenta um numero elevado de neurónios no nível intermédio. a) Resposta da rede para os valores que serviram de treino, b) resposta da rede para um conjunto de amostras maior. - Overfitting	95
Gráfico 4.2 - Exemplo de uma rede neuronal com um numero reduzido de neurónios no nível intermédio - <i>Underfitting</i>	96
Gráfico 5.1 - Resultado do treino da Rede Neuronal Artificial que implementa o Controlador	104
Gráfico 5.2- Teste realizado a Rede Neuronal Artificial que implementa a parte do Adaptador que estima a velocidade de avanço	110
Gráfico 5.3- Erro resultante da simulação da Rede Neuronal Artificial que implementa a parte do Adaptador que estima o valor da velocidade de avanço	111
Gráfico 5.4- Teste realizado à Rede Neuronal Artificial que implementa a parte do Adaptador que estima a velocidade de rotação	112
Gráfico 5.5- Erro resultante da simulação da Rede Neuronal Artificial que implementa a parte que estima o valor da velocidade de rotação.....	113

Gráfico 5.6 - Curva do erro em relação ao valor de entrada $mrr(t)$ e o valor de saída do Adaptador na forma F/S	114
Gráfico 5.7 - Resultado do treino da rede neuronal a) Evolução do quadrado do erro b) Comparação do valor mrr estimado pela rede e o verdadeiro valor de mrr (usado para treinar a rede).....	119
Gráfico 5.8 - Força de corte medida á saída da planta F_c	120
Gráfico 5.9- Perturbações apresentadas ao sistema	120
Gráfico 5.10- Comparação entre o sinal verdadeiro (representado por cruces) e o sinal estimado pelo Identificador (representado a solido)	121
Gráfico 5.11 - a) - Curva de aprendizagem do controlador. b) - Simulação usando os valores de treino	124
Gráfico 5.12- Teste á rede neuronal, usando valores diferentes dos usados no treino.....	125
Gráfico 5.13- Os valores de F e S á saída do Adaptador	127
Gráfico 5.14 - Evolução do quadrado do erro ao longo das várias iterações.....	128
Gráfico 5.15 - Evolução do erro da primeira para a vigésima iteração.....	128

1. Introdução

1. Introdução	17
1.1 A máquina	18
1.2 Processo de corte	18
1.3 Processo de controlo	20
1.3.1 Dificuldades inerentes ao processo de corte	20
1.4 Controlo da Força de Corte	24
1.5 Objectivos	25
1.6 Estrutura da tese	25

1.1 A máquina

Desde os tempos mais remotos que o homem produz ferramentas para o auxiliar no seu progresso. A partir da revolução industrial, em meados do século XIX, a máquina têm vindo a adquirir um papel cada vez mais preponderante na sociedade em que vivemos. Este progresso tem implicado substituir a mão de obra humana por máquinas, quer por razões de produtividade, quer por limitações físicas do homem.

Nos tempos que correm, as empresas estão totalmente dependentes das máquinas que utilizam. A sua posição no mercado depende da sua capacidade produtiva, tanto em quantidade como em qualidade e rapidez.

1.2 Processo de corte

As máquinas são normalmente produzidas com a finalidade de otimizar as operações realizadas pelo homem.

Quando se utilizam máquinas para substituir a mão de obra humana, procura-se um aumento de produtividade e uma melhoria da qualidade do produto final. A fiabilidade e repetitibilidade destas máquinas, está acima da capacidade humana.

A industria ligada a processos de corte, está debate-se presentemente com duas limitações que diminuem a sua taxa de produção

- a velocidade de avanço, que é sempre programada de um modo conservador pelos programadores “*off line*”,
- e a possibilidade de quebra da ferramenta, que normalmente provoca a destruição da peça.

Um sistema que permita a adaptação da velocidade de corte às condições reais de funcionamento da máquina, irá, de concerteza, permitir um melhoramento significativo do processo de corte.

Na Figura 1.1 temos duas imagens de uma peça maquinada (imagens *a* e *b*), sendo apresentadas duas imagens apenas para dar a noção geométrica da peça.

A imagem *a* é uma vista de topo da peça, e permite uma visualização da influência dos diversos factores no corte, enquanto a imagem *b* é uma vista de um ponto situado acima e à esquerda da peça, e permite ver melhor a geometria da peça em causa.

Esta peça foi maquinada segundo o processo tradicional e segundo um novo processo em estudo, em que se controlou a força de corte da ferramenta. Podemos observar, na mesma peça, as duas situações de maquinação.

A olho nu ou simplesmente passando a mão pela zona maquinada, é possível observar a diferença de qualidade apresentada pelas superfícies.

Se observarmos a imagem *a* podemos notar a diferença dos reflexos apresentados por um e outro tipo de maquinação. Nota-se perfeitamente nesta imagem, as marcas deixadas pela flexão da ferramenta, quando sujeita a esforços maiores.

Também foi possível observar durante o processo de maquinação, os tempos gastos pelo processo tradicional e pelo processo em estudo (controlando a força de corte).

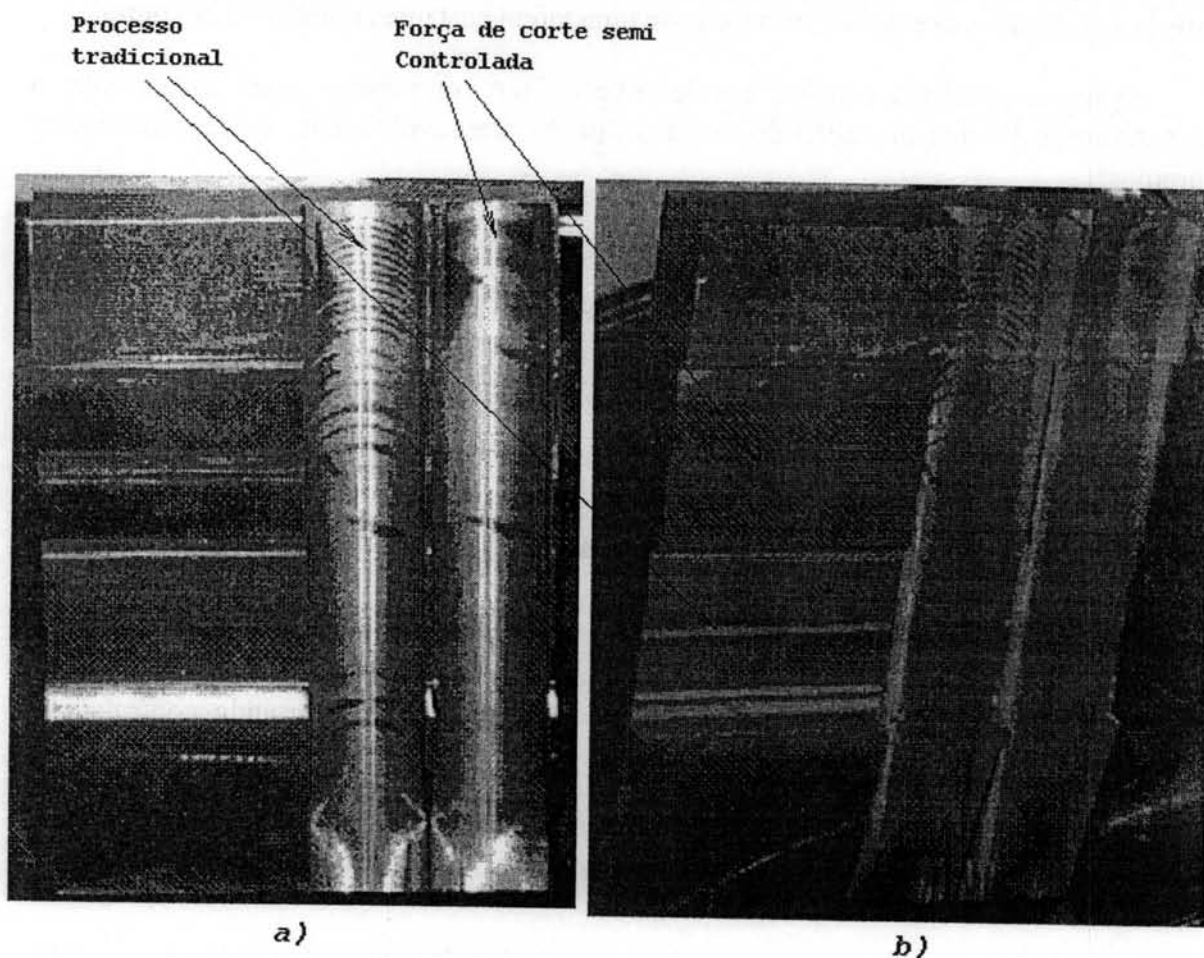


Figura 1.1 - Peça de teste onde se reproduz a maquinação clássica; e uma maquinação com força de corte controlada

No processo em estudo, calcula-se que o desgaste da ferramenta também será reduzido, quando comparado com o tradicional. No entanto, como a mesma ferramenta foi usada durante os dois processos, tornou-se impossível distinguir o desgaste provocado por um ou pelo outro.

A peça apresentada na Figura 1.1 e a informação referida atrás, provêm do Instituto Espanhol IAI - Instituto de Automatica Industrial, de Madrid, que tem realizado trabalho inovador nesta área, e com o qual o autor pode manter contactos durante a elaboração desta tese.

1.3 Processo de controlo

Substituir um sistema de controlo humano por sistemas electrónicos é bastante fácil e eficaz em algumas operações, mas torna-se uma tarefa bastante complicada em outras.

O nosso problema consiste em desenvolver um controlador capaz de controlar a força de corte de uma operação de maquinação. No desenvolvimento deste controlador, usaremos:

- uma fresadora vertical equipada com um comando CNC (Figura 1.2)
- como material, uma liga de alumínio
- ferramentas de corte HSS (aço rápido), com diâmetros compreendidos entre 10 e 30 mm e de 2 dentes.

1.3.1 Dificuldades inerentes ao processo de corte

As máquinas construídas hoje em dia, trazem já sistemas de controlo bastante precisos e elaborados. No entanto, o controlo da força num processo de corte é uma tarefa delicada e difícil, senão mesmo impossível, de ser implementada usando controladores clássicos.

Na realidade, o processo de corte de uma fresadora vertical, tem características muito diversas, dependendo do material, da ferramenta, e principalmente da geometria da peça a maquinar.



Figura 1.2 - Máquina disponível no CCP, usada para realizar medidas de força de corte.

A geometria da peça a maquinar é a principal responsável pelas dificuldades de controlo, pois quando se produz uma máquina deste tipo, esta deve ser capaz de trabalhar uma gama muito variada de peças, com geometrias totalmente imprevisíveis (Figura 1.1).

Podemos descrever as dificuldades dividindo-as em três classes importantes, a saber.

- a tecnologia de corte usada no processo de maquinação
- o problema do controlo
- a tecnologia usada para implementar o controlo.

O problema reside no facto de os parâmetros das operações serem especificados no programa de maquinação. Para cada trajectória definida no programa, são especificados os seguintes parâmetros:

- profundidade de corte,
- velocidade de avanço,
- velocidade de rotação da árvore.

Estes parâmetros definem um terceiro - A Força de Corte. Esta força deve ser escolhida tendo em conta o material a maquinar e a ferramenta. No entanto, esta força depende da secção de material a maquinar - secção da apara - e da aresta de corte da ferramenta.

Durante o percurso de uma das trajectórias definidas no programa, a secção poderá variar, implicando alterações da força de corte. As alterações da força de corte sofridas durante o processo de maquinação, retiram a ferramenta do ponto óptimo de funcionamento, implicando aumento de custos de produção.

Os custos de produção tem três componentes principais, que poderão ser definidos do seguinte modo:

1. Caso em que a força de corte aumenta com o percurso da trajectória, em valores superiores aos normais para a ferramenta

- Desgaste rápido da ferramenta de corte - a aresta de corte desgasta-se rapidamente, implicando a sua substituição ou tratamento em mais intervalos reduzidos,
- Mau acabamento das superfícies, com zonas marcadas pela ferramenta, por esta entrar em deflexão devido a um desgaste rápido do fio de corte.

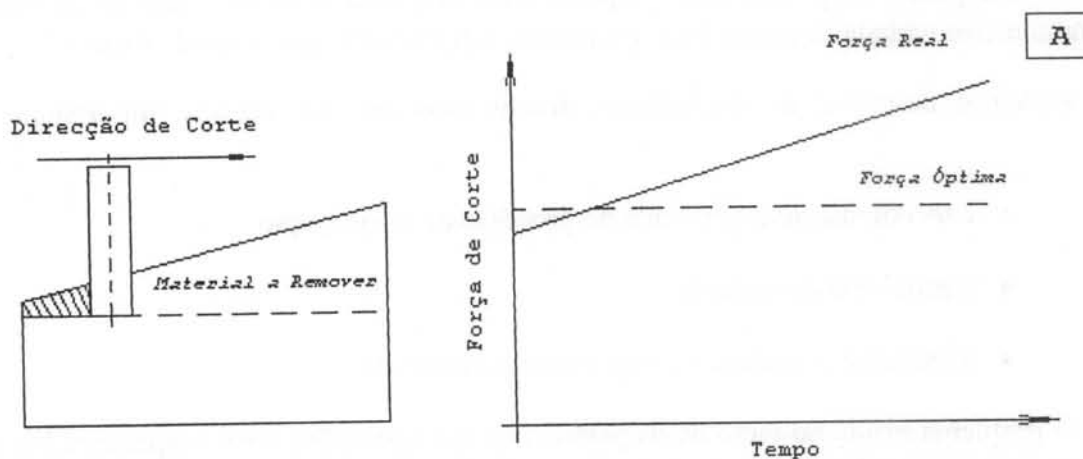


Figura 1.3 - Exemplo de como pode aumentar a força de corte real ao longo de uma operação de maquinação

A Figura 1.3 é um exemplo de como a força de corte pode aumentar ao longo da trajectória de maquinação, sendo certo que muitas outras situações poderão originar efeitos idênticos.

2. Caso em que a força de corte diminui com o percurso da trajectória

- Desaproveitamento da ferramenta, pois esta estará a trabalhar abaixo das suas possibilidades,

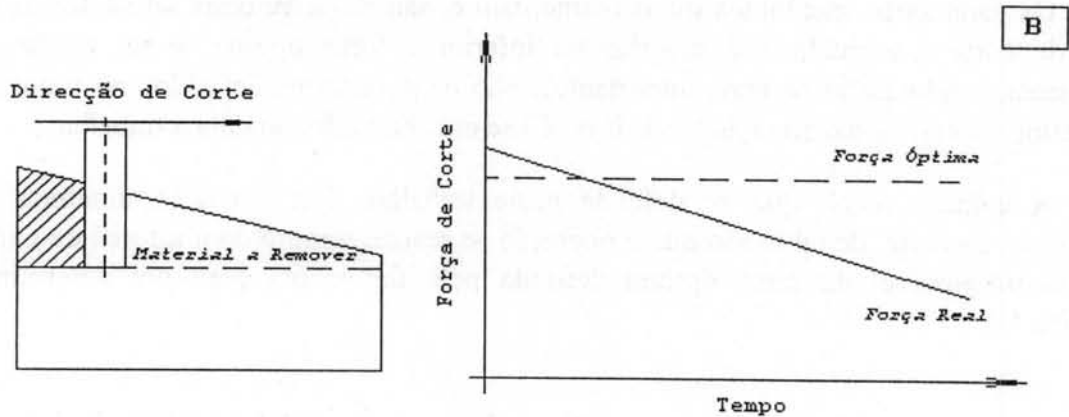


Figura 1.4 - Exemplo de como pode diminuir a força de corte ao longo de uma operação de maquinação

A Figura 1.4 é um exemplo de como a força de corte pode diminuir ao longo da trajectória de maquinação. Esta variação poderá no entanto mudar de sentido, voltando a força a crescer.

3. Caso em que a força de corte é sempre inferior á que poderá ser usada pela ferramenta, para um determinado material, temos:

- Tempo de operação elevado, sem que isso se reflecta numa melhor qualidade de acabamento ou num aumento da durabilidade da ferramenta.

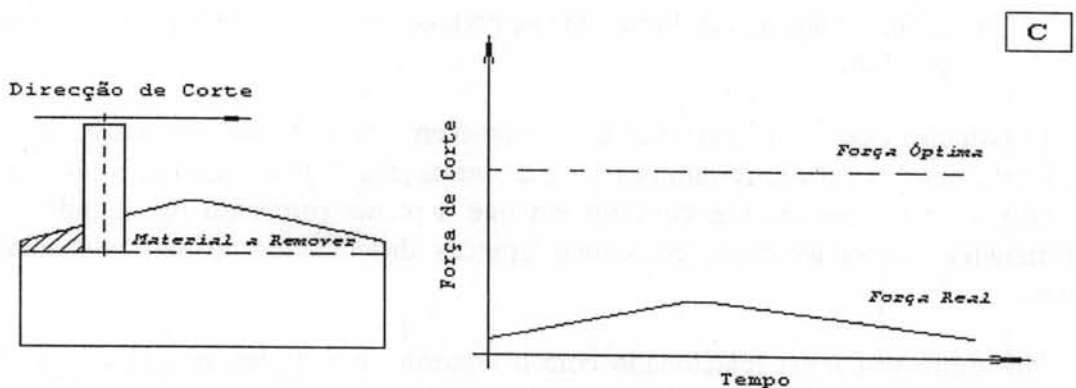


Figura 1.5 - Exemplo de como a força de corte pode variar mas sempre abaixo do valor ótimo

Neste exemplo, (Figura 1.5) mostra-se o caso em que a força de corte é muito inferior ao valor óptimo. Apesar de ser sempre inferior, esta força de corte poderia ser ajustada de modo a que o seu máximo fosse igual à força óptima para a ferramenta. Isto no entanto, isso não invalidaria o que se referiu atrás, pois continua a existir um desaproveitamento da ferramenta.

Os parâmetros escolhidos inicialmente, isto é, são os parâmetros que definem se a força de corte é normalmente superior ou inferior à força óptima, a ser usada pela ferramenta, sendo assim os mais importantes. São os parâmetros definidos no programa, que definem em qual das situações (A, B ou C) se encontra a ferramenta a trabalhar.

A solução ideal, que se defende neste trabalho, é a alteração dinâmica dos parâmetros de corte, de tal modo que a operação se realize sempre com a força de corte o mais perto possível da força óptima definida pela ferramenta para um determinado material.

1.4 Controlo da Força de Corte

Dadas as considerações anteriores, podemos dizer que se torna importante o controlo da força de corte numa operação de maquinação, de modo a otimizar essa mesma operação, minimizando os custos da peça a maquinar e melhorando a sua qualidade.

Para se poder realizar este controlo, é necessário:

- estudar e compreender as tecnologias de corte usadas normalmente numa operação de maquinação,
- verificar como se comporta a máquina no seu conjunto, para se perceber correctamente o problema que se pretende resolver,
- definir quais as condições que se pretende usar na tentativa da resolução do problema.

O primeiro desafio foi perceber qual o problema de controlo que estava em jogo, que tipo ou tipos de controlo seriam passíveis de ser usados, e qual a melhor solução a usar neste caso, pois trata-se de um controlo em que a planta (processo de maquinação) é completamente desconhecida e, colocando grandes dificuldades ao seu conhecimento completo.

Um outro problema relacionado com o controlo, é o da tecnologia que deve ser usada para se implementar o controlo escolhido. As Redes Neurais Artificiais e a Lógica *Fuzzy* são das possibilidades mais promissoras actualmente, para o controlo adaptativo da força de corte, e foram ambas consideradas.

Estando o IAI envolvido num projecto de I&D cujo objectivo é controlar a força de corte, usando Lógica *Fuzzy*, entendeu-se que seria de investigar a utilização de Redes Neurais Artificiais na implementação de um controlador de força de corte.

A utilização de uma tecnologia alternativa, vai permitir realizar importantes comparações no que respeita ao comportamento das Redes Neurais Artificiais e da Lógica *Fuzzy* na implementação de controladores adaptativos, principalmente quando aplicados ao controlo da força de corte .

Note-se ainda que qualquer que fosse o tipo de controlo a ser implementado, teria que ser um controlo adaptativo para sistemas não lineares.

1.5 Objectivos

Os objectivos principais deste trabalho, são

- demonstrar a importância do controlo da força de corte
- lançar algumas ideias base que poderão ser usadas futuramente no desenvolvimento de controladores da força de corte.

Para além destes objectivos gerais, podemos definir objectivos mais precisos para a elaboração desta tese.

Assim, diremos que temos também como objectivo definir a estrutura geral do controlador e a sua subestrutura, criando modelos e simulando o seu funcionamento em programas próprios, como por exemplo o MATLAB.

1.6 Estrutura da tese

Esta trabalho será descrito ao longo de seis capítulos, dos quais o primeiro é este mesmo - *Introdução*.

O segundo capítulo, designado *O problema da maquinação*, será dedicado ao estudo e apresentação dos conceitos ligados ao processo de corte. Serão descritas as dificuldades de controlo encontradas, e sua importância para o desenvolvimento do controlador.

O terceiro capítulo, intitulado *O sistema de controlo*, será dedicado a uma breve apresentação do controlo e do controlador, definindo de um modo lato a estrutura que será usada pelo controlador da força de corte a implementar.

No quarto capítulo, com o título *Redes Neurais Artificiais*, fazer-se-à uma descrição das redes neuronais e do seu principio de funcionamento. Serão descritas as semelhanças entre os princípios biológicos e computacionais, associados às redes neuronais.

A *Implementação* será o tema do quinto capítulo. Descrever-se-ão quais as soluções adoptadas, quais os resultados obtidos nas simulações e como se poderão interpretar alguns destes resultados.

As *Conclusões* serão apresentadas no sexto e ultimo capítulo.

Seguir-se-ão, em anexo, algumas listagens, e informação relevante usada ao longo do decorrer desta tese.

2. O problema da maquinação

2. O problema da maquinação.....	27
2.1 Sumário	28
2.2 Tecnologia de corte	28
2.2.1 Força de corte	30
2.2.1.1 Força especifica de corte	30
2.2.1.2 Secção do corte	33
2.2.1.2.1 Espessura da apara	34
2.2.1.2.2 Avanço/Velocidade de avanço	35
2.2.1.2.3 Influencia do angulo ϕ na espessura da apara	37
2.2.1.3 Calculo prático da secção e respectiva força de corte	39
2.2.1.3.1 Influência dos dentes de corte na força de corte	39
2.2.1.3.2 Avanço por dente	41
2.2.1.3.3 Secção por dente	41
2.2.1.3.4 Força de corte por dente	41
2.2.1.3.4.1 Força total de corte	42
2.2.2 Força de avanço	44
2.2.3 Influencia da geometria da peça na força de corte	44
2.2.3.1 Variação da força de corte	46
2.2.3.2 Uso da ferramenta de geometria variável ...	47
2.2.4 Influência do estado da ferramenta no corte ...	48
2.3 Manter a força de corte constante	52
2.4 Conclusões	55

2.1 Sumário

Neste capítulo, serão expostos os pressupostos nos quais se baseará todo o trabalho relacionado com o problema tratado nesta tese.

Começaremos por abordar os principais conceitos ligados com a tecnologia de corte, e nomeadamente com a força de corte. Tentaremos em seguida mostrar quais as implicações da geometria das peças e do estado da ferramenta, na maquinação e como estas podem afectar a força de corte.

Posteriormente, poderemos ver como as variações da força de corte afectam as ferramentas e as superfícies trabalhadas da peça.

No final do capítulo, será abordado o controlo da força de corte.

2.2 Tecnologia de corte

Para podermos definir o problema que nos propomos resolver de uma forma passível de ser optimizada, é necessário conhecer as tecnologias relacionadas com o corte e muito particularmente com as máquinas de fresagem vertical e ferramentas de vários dentes.

Quando observamos, a olho nu, uma operação de maquinação numa fresadora vertical, não nos saltam logo à vista quais são realmente os parâmetros envolvidos no processo.

Analizando o programa, ficamos a saber qual é a trajectória a ser percorrida pela ferramenta, desde o ponto inicial até ao seu termo, podendo este, tornar-se o ponto de início de outra trajectória.

A importância de se falar em trajectória, ponto de início e ponto de fim, prende-se com o facto de ser nestes pontos que se definem os parâmetros de corte, que se manterão constantes ao longo de cada trajectória. Os parâmetros definidos nestes pontos são a Profundidade de Corte, a Velocidade de Avanço e a Velocidade de Rotação da Ferramenta ou considerando que é proporcional, a Velocidade Linear de Corte ($v = \omega r$).

Como estes pontos e parâmetros são definidos no programa de maquinação, a máquina executa a trajectória mantendo constante os parâmetros definidos inicialmente. Podemos ver na Figura 2.6 um exemplo simples de um programa de maquinação para desbastar uma determinada espessura de material, usando uma fresa de 10 mm, também podemos ver como a máquina executará este programa, seguindo a trajectória assinalada a traço interrompido.

As instruções usadas no programa, são as normalmente usadas nos CNC actuais.

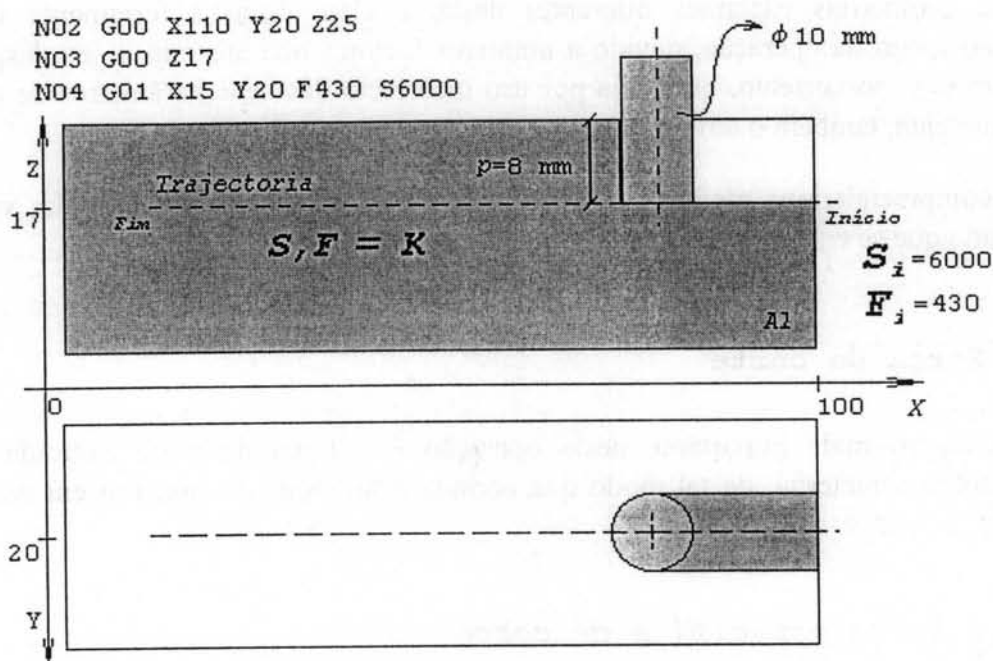


Figura 2.6 - Trajectória da Ferramenta de Corte

S = Velocidade de rotação da ferramenta - *Spindle*

F = Velocidade de avanço da ferramenta - *Feed Rate*

S_i, F_i = Valores no início da trajectória

O programa começa com um movimento em avanço rápido da ferramenta, para o posicionamento desta (linhas N02 e N03), seguindo-se um movimento de corte com uma velocidade de avanço de 430 mm/min e uma velocidade de rotação de 6000 RPM. Durante este movimento, a máquina mantém estes valores até encontrar outra linha de programa que os altere.

Este exemplo mostra uma trajectória simples, com uma espessura de corte constante, em que nada se diz em relação ao estado da ferramenta. Perante tal facto será de supor nada se poderá otimizar neste caso.

Partindo do principio de que todos os parâmetros foram escolhidos correctamente, o pressuposto estria correcto se a ferramenta mantivesse as sua características de corte ao longo de toda a trajectória. No entanto, a maquinação de uma peça contém um conjunto de trajectórias e geometrias bastantes diferentes desta, e além disso a ferramenta vai-se degradando ao longo da operação, devido a inúmeros factores que afectam o seu desgaste, quer seja com um uso correcto, quer seja por uso incorrecto dos seus parâmetros de corte. Além da ferramenta, também o tempo de maquinação poderá ser afectado.

Para compreendermos melhor o que se passa no momento da maquinação, vamos recorrer á teoria que se encontra por detrás da tecnologia de corte.

2.2.1 Força de corte

O parâmetro mais importante nesta operação é a força de corte exercida pela ferramenta sobre o material, de tal modo que permita o arranque do material em excesso (apara).

2.2.1.1 Força especifica de corte

Para cada material, existe uma força especifica de corte [1], também chamada de pressão de corte por certos autores [2]. A força especifica de corte para cada material é definida como *a força necessária para o corte de uma apara de 1 mm^2 de secção desse material* [1]. Para sabermos qual a força despendida no corte de material, qualquer que seja a secção basta multiplicar essa força especifica pela respectiva secção da apara

$$Fc = fc \times Sec \quad (\text{Kgf}) \quad (2.1)$$

em que

Fc = Força de corte exercida pela aresta de corte em (Kgf)

fc = força especifica de corte em ($\text{Kgf} \cdot \text{mm}^{-2}$)

Sec = secção da apara no instante em que se exerce a força em (mm^2)

ou, quando se define a pressão de corte,

$$Ks = \frac{Fc}{Sec} \quad (\text{Kgf} \cdot \text{mm}^{-2}) \quad (2.2)$$

em que:

Ks = pressão especifica de corte

Como se verifica pelas equações (2.1) e (2.2) f_c e K_s definem a mesma grandeza.

Os valores de f_c , que são característicos de cada material, determinam-se experimentalmente.

Os técnicos do estudo experimental do corte indicam que para cálculos correntes, se podem tomar para f_c , valores iguais a 3 ou 4 vezes os valores de σ_r , que é a carga de rotura do respectivo material, quando ensaiado à tracção. Ficamos assim com a força específica de corte:

$$f_c = 3 \cdot 4 \times \sigma_r \quad (\text{Kgf} \cdot \text{mm}^{-2}) \quad (2.3)$$

Experimentalmente também se verifica que a pressão específica de corte depende de vários factores [2], nomeadamente

- Do material a maquinar. Materiais com resistência mecânica diferente apresentam pressão específica de corte também diferente. Contudo, a relação entre a resistência mecânica de material a maquinar e o valor da pressão específica de corte não é proporcional, ou seja, para iguais parâmetros de maquinação o valor da pressão específica de corte de um aço com uma tensão de rotura de 1000 MPa não é o dobro do valor da pressão específica de corte para uma aço com uma tensão de rotura de 500 MPa.
- Da secção da apara. Experimentalmente, verificou-se que a pressão específica de corte diminui com o aumento da espessura da apara, mantendo-se sensivelmente constante para um aumento da largura da apara.
- Da velocidade linear de corte. Verificou-se na maquinação de vários metais com ferramentas de carbonetos sinterizados ($V_c > 60$ m/min) que a pressão específica de corte diminui lentamente com o aumento da velocidade de corte. Para velocidades inferiores, ou no caso de ferramentas de aço rápido, os valores da pressão específica de corte podem ser 5 a 20% superiores.
- Do fluido de corte. Apenas nas velocidades de corte baixas, os fluidos de corte contribuem para o abaixamento da pressão específica de corte. Esta diminuição é tanto mais significativa quanto mais fácil for a penetração do fluido na zona de contacto apara/ferramenta.

- Da geometria da ferramenta. Quanto maior é o ângulo de ataque (γ) menor é o valor da pressão específica de corte. O ângulo de saída (α) quando muito pequeno provoca o aumento do valor da pressão específica de corte, dado o aumento do atrito entre a peça e a face de saída da ferramenta. A influência do ângulo de inclinação de aresta (λ) verifica-se apenas para valores negativos elevados; nesse caso, a força de penetração aumenta consideravelmente podendo deflectir a peça, ou mesmo deslocar transversalmente a ferramenta. O valor da pressão específica de corte diminui com o aumento do ângulo de posição da aresta do corte principal (χ) desde que não se verifique a influência da aresta lateral de corte, ou seja, para $\chi > 5^\circ$.
- Do estado de afiação da ferramenta. Na gama de desgaste admissível da ferramenta a pressão específica de corte pode chegar a atingir valores 30% superiores.
- Da rigidez da ferramenta. Experimentalmente verificou-se que a rigidez, quando pequena, aumenta a pressão específica de corte.

O valor da pressão específica de corte (K_s), pode ser obtido experimentalmente recorrendo a ensaios de maquinação, simulando as condições de corte pretendidas, medindo a força de corte (F_c), e conhecendo a secção da apara (Sec), para o processo de maquinação em causa.

Existem no entanto algumas formulas analíticas (baseadas na experimentação) que relacionam a pressão específica de corte, com as diversas grandezas que a influenciam. Porém, o número de variáveis que intervêm no processo e o desconhecimento de outras, fazem com que essas fórmulas forneçam somente valores aproximados.

Taylor, um dos primeiros investigadores propôs as seguinte formulas[2]

$$K_s = \frac{88}{a^{0.25} p^{0.07}} \quad (\text{Ferro fundido cinzento}) \quad (2.4)$$

$$K_s = \frac{138}{a^{0.25} p^{0.07}} \quad (\text{Ferro fundido branco}) \quad (2.5)$$

$$K_s = \frac{200}{a^{0.07}} \quad (\text{Aço de construção}) \quad (2.6)$$

sendo a o avanço e p o penetramento.

A AWF (Ausschuss für Wirtschaftliche Fertigung - Associação de Produção Económica) da Alemanha sugere a formula seguinte:

$$Ks = \frac{Cw}{a^{0.477}} \quad (2.7)$$

sendo a o avanço e Cw uma constante do material e da ferramenta.

A ASME (American Society of Mechanical Engineers) apresenta a seguinte formula

$$Ks = \frac{Ca}{a^n} \quad (2.8)$$

sendo a o avanço, n o expoente do material a maquinar e Ca a constante do material da peça e da ferramenta.

Kronenberg resume a pressão específica da corte à seguinte formula:

$$Ks = Ks_0 \frac{\left(\frac{G}{G_0}\right)^g}{\left(\frac{S}{S_0}\right)^f} \quad (2.9)$$

sendo Ks a pressão específica de corte, Ks_0 a pressão específica de corte em condições padrão, G o índice de esbeltez, G_0 o índice de esbeltez padrão (= 5), S a secção da apara, S_0 a secção da apara padrão (1 mm²), e os expoentes g e f dependentes do material da peça e da ferramenta.

Do referido atrás podemos verificar a elevada componente experimental ligada ao corte de material e também a elevada diversidade de equações, que relacionam a força específica de corte com características do material e da ferramenta.

2.2.1.2 Secção do corte

Durante o corte e considerando uma rotação completa da ferramenta de corte, podemos afirmar, que no caso do corte vertical, a secção do material varia ao longo do corte. Esta variação da secção está representada a tracejado na Figura 2.7.

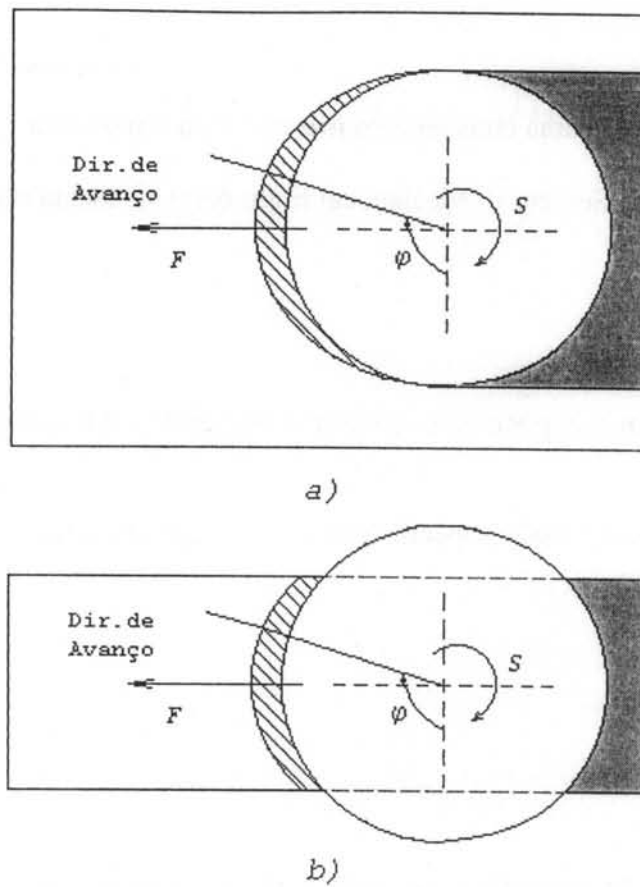


Figura 2.7 - Secção da apara produzida no corte

No caso representado na Figura 2.7, considerou-se que existe apenas uma aresta de corte por cada rotação, e que a profundidade do corte se mantém constante. Poderemos definir a força exercida pela aresta de corte ao longo de cada revolução (equação 2.1) mas para isso necessitamos de calcular a secção da apara.

2.2.1.2.1 Espessura da apara

A secção da apara é função da profundidade de corte e da espessura da apara.

$$Sec = p \times e \quad (mm^2) \quad (2.10)$$

em que:

p = Profundidade de corte segundo o eixo da ferramenta em (mm)

e = Espessura de corte

Para podermos calcular a secção da apara necessitamos primeiro de definir a sua espessura

$$e(\varphi) = a(\varphi) \times \sin(\varphi) \quad (mm) \quad (2.11)$$

em que:

e = Espessura da apara em (mm)

a = Avanço por rotação em (mm.rot⁻¹)

φ = Angulo entre a recta perpendicular á direcção de corte, e a recta que une o centro da ferramenta com o ponto de contacto na aresta de corte, ver Figura 2.7.

2.2.1.2.2 Avanço/Velocidade de avanço

O avanço não depende directamente do valor do angulo φ , e também não é função deste. No entanto, poderá ser relacionado com φ , através do tempo. O valor do avanço está relacionado com o tempo, segundo a equação

$$a(t) = F \times t \quad (mm \cdot min^{-1}) \quad (2.12)$$

sendo

F = Velocidade de avanço em (mm.min⁻¹).

O angulo φ , está relacionado com o tempo, segundo a equação

$$\varphi(t) = 2\pi \times S \times t \quad (rad) \quad (2.13)$$

sendo

S = Velocidade de rotação em (rot. min⁻¹)

Notar que o tempo está expresso em minutos, e que caso se pretenda o tempo expresso em segundos, será necessário dividir as equações por 60. O valor de 2π na equação 2.13, é o factor de conversão de rotações para radianos.

Se tudo estiver a trabalhar correctamente na máquina, então podemos supor que as equações traduzem a realidade, e logo podemos igualar o tempo nas duas equações, resultando a seguinte equação para o avanço função de φ ,

$$a(\varphi) = \frac{F}{S} \times \frac{\varphi}{2\pi} \quad (\text{mm} \cdot \text{rad}^{-1}) \quad (2.14)$$

De notar que a espessura da apara é função do avanço, representado por a , e função da velocidade de rotação da ferramenta, representada por φ , - mais propriamente pela derivada de φ .

O avanço por rotação ($\varphi = 2\pi$), poderá ser calculado a partir da velocidade de avanço, e da velocidade de rotação da ferramenta, como mostra a equação 2.15.

$$a = \frac{F}{S} \quad (\text{mm} \cdot \text{rot}^{-1}) \quad (2.15)$$

Este valor de a , representa o avanço por cada rotação da ferramenta, sendo constante ao longo da trajetória.

Podemos ver o resultado da variação da espessura de corte no Gráfico 2.1

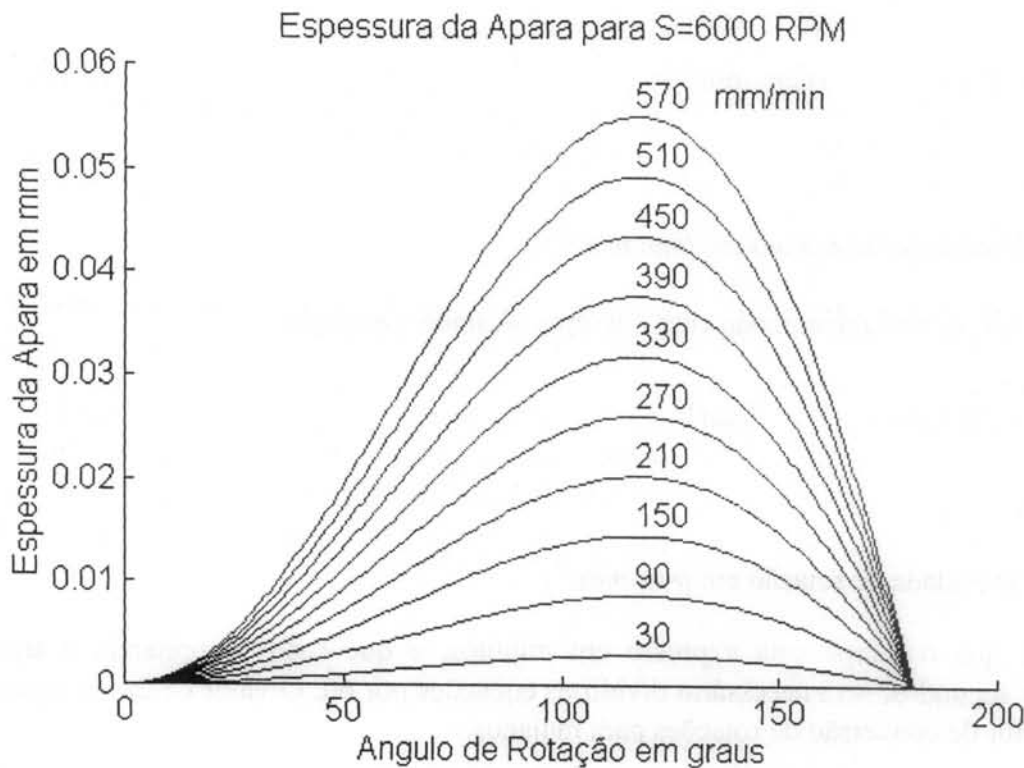


Gráfico 2.1 - Variação da espessura da apara função do ângulo de rotação para diferentes velocidades de avanço

Como podemos analisar no Gráfico 2.1, para uma velocidade de rotação constante, a espessura de corte varia em função do ângulo φ . É devido à velocidade de avanço que o máximo da espessura não aparece aos 90° mas sim mais próximo dos 120°.

De salientar que, com o aumento da velocidade de avanço, a espessura torna-se mais pronunciada, verificando-se a necessidade de se aumentar a velocidade de rotação, para que o corte se mantenha com a mesma espessura, ou seja, cujo derivada seja menor ao longo de toda a superfície de corte.

Note-se que, mesmo para a velocidade de avanço mais elevada, a espessura da apara é de pequena dimensão. Adiante, voltaremos a falar neste assunto.

Como ao longo de uma revolução, a profundidade de corte se mantém praticamente constante, a equação da secção seria então a seguinte

$$Sec = p \times \frac{F}{S} \times \frac{\varphi}{2\pi} \times \sin(\varphi) \quad (mm^2) \quad (2.16)$$

2.2.1.2.3 Influencia do ângulo φ na espessura da apara

O ângulo φ , poderá variar entre 0 e 180 graus, ou entre outros valores definidos pelo início e fim do corte. O facto do ângulo φ variar entre 0° e 180°, significa que ao longo de cada rotação, a força exercida pela ferramenta partirá de zero, atingirá um máximo e voltará a zero, como se vê na linha exterior do Gráfico 2.2, também representada no exemplo da Figura 2.7, alínea *a*.

Mas mais importante que a variação de φ , entre 0° e 180°, é o caso dos valores de início e fim serem maior que 0° e menor que 180°, respectivamente, como é o caso do exemplo da Figura 2.7 alínea *b*.

Neste caso podemos verificar que o material a maquinar é de secção inferior ao diâmetro da fresa, fazendo com que os ângulos de início (φ_i) e fim (φ_f) de corte, sejam cerca de 20° e 160° respectivamente, o que altera a função da espessura para uma equação não linear como se pode ver a seguir.

$$e(\varphi) = \begin{cases} 0 & \Leftarrow \varphi \leq \varphi_i \\ a(\varphi) \times \sin(\varphi) & \Leftarrow \varphi_i < \varphi < \varphi_f \\ 0 & \Leftarrow \varphi \geq \varphi_f \end{cases} \quad (2.17)$$

Os valores da função espessura, traduzem-se numa força aplicada à ferramenta com mudanças bruscas, como se mostra no Gráfico 2.2, linha interior.

Analisando o gráfico, vemos que não se trata mesmo assim de um caso muito problemático, pois, como podemos ver, a variação brusca e importante é a variação de 0 para um valor da força diferente de zero, e neste caso, essa variação não é significativa. A variação mais significativa verifica-se na passagem da força de corte de um determinado valor para zero, mas como é uma passagem para zero não tem grande importância.

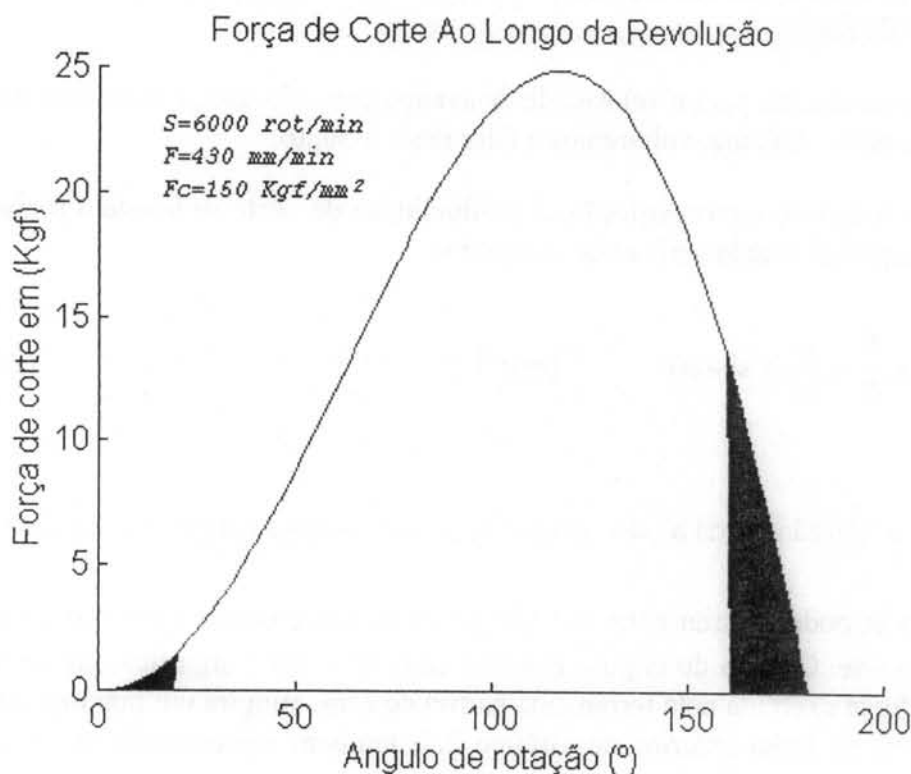


Gráfico 2.2 - Variação da Força de Corte Função do ângulo de início e fim

As ferramentas são projectadas para suportar este género de solicitações, independentemente do tipo de força a ser exercida, contando apenas o valor máximo da força exercida ao longo da maquinação.

2.2.1.3 Calculo prático da secção e respectiva força de corte

Na prática, como a velocidade de rotação da ferramenta é bastante elevada quando comparada com a velocidade de avanço, mesmo para velocidades de avanço mais elevadas, a espessura da apara é reduzida (60 μm para um avanço de 570 mm/min do exemplo), assim sendo, e desde que se encontre dentro dos limites considerados normais, poderemos considerar o valor da espessura da apara em cada rotação, igual ao avanço por rotação, para o calculo da secção e respectiva força de corte. Assim sendo, temos que a secção por cada rotação é

$$Sec = p \times \frac{F}{S} \quad (\text{mm}^2) \quad (2.18)$$

A partir deste momento, vamos considerar esta simplificação, sempre que nos referirmos a calculo de forças de corte. Como mais á frente demonstraremos, neste trabalho não se pretende controlar a força de corte ao longo da rotação da ferramenta, isto porque, o seu interesse é reduzido, comparado com as variações que a força pode sofrer, relacionadas com outros parâmetros da maquinação.

Se substituirmos na equação 2.1, o valor da *Sec* obteremos finalmente

$$Fc = fc \times p \times \frac{F}{S} \quad (2.19)$$

2.2.1.3.1 Influência dos dentes de corte na força de corte

Se considerarmos que a ferramenta de corte poderá ter mais que uma aresta de corte (dente), temos o caso representado na Figura 2.8

Observando a Figura 2.8, podemos verificar que durante o avanço, cada dente vai cortando uma apara com uma espessura menor - menor que no caso anterior, de apenas um dente. O facto de a espessura diminuir, significa que a secção, e por conseguinte a força de corte por aresta de corte, vai diminuir também.

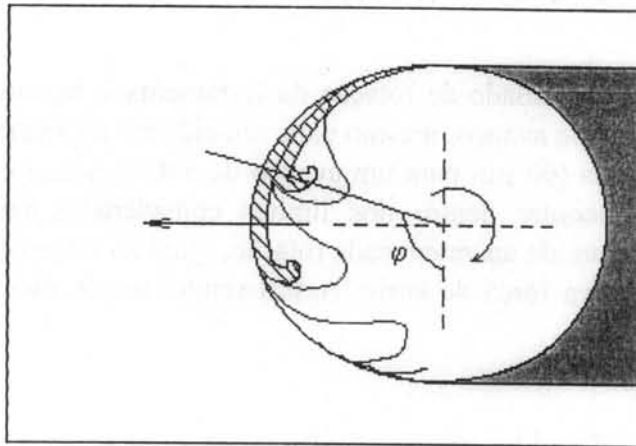


Figura 2.8 - Secção da apara produzida no corte por uma ferramenta de várias arestas de corte

No caso do uso de uma ferramenta de várias arestas de corte, podemos observar na Gráfico 2.3, o factor mais importante - as curvas têm os seus máximos muito mais achatados. Neste gráfico, foi mantida a escala do eixo y - espessura da apara, para mostrar exactamente a diferença para o Gráfico 2.1. Se observarmos o Gráfico 2.3, verificamos que o valor máximo de cada curva é muito menor, significando que a secção também o é.

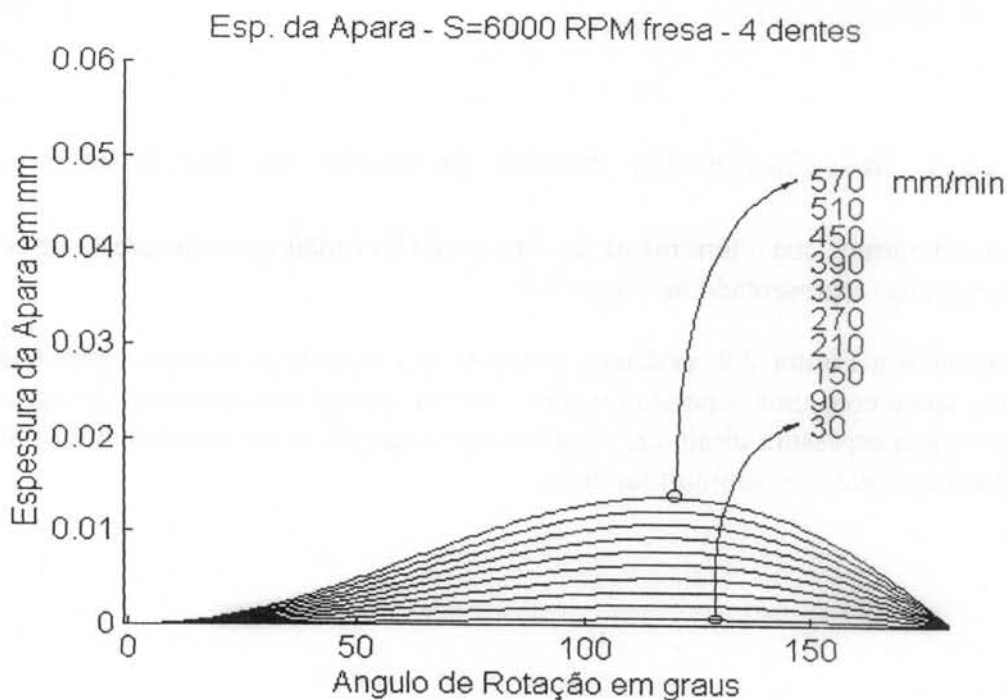


Gráfico 2.3 - Variação da espessura da apara função do ângulo de rotação para diferentes velocidades de avanço, agora considerando o numero de dentes da fresa

O facto de termos secções menores, e, como consequência, forças de corte menores, significa que para o mesmo tipo de corte, podemos usar parâmetros mais elevados, desde que a máquina assim o permita - potência e limites físicos disponíveis na máquina.

2.2.1.3.2 Avanço por dente

No caso que estamos a estudar, podemos definir um outro tipo de avanço - avanço por dente ad , que será substituído nas formulas anteriores, definindo-se assim a força de corte por dente Fcd . A força de corte total, exercida pela ferramenta, é o somatório das forças de corte dos vários dentes, que num determinado instante se encontram a cortar material.

$$ad = \frac{\frac{F}{S}}{N^{\circ} \text{ dentes}} \quad (2.20)$$

O facto de se dividir o avanço total por rotação, pelo numero de dentes, considera que por cada rotação, cada dente cortará uma secção da apara, relacionada com o avanço durante os instantes em que o dente se encontra em contacto com o material.

2.2.1.3.3 Secção por dente

A secção por dente define qual a secção que cada dente da fresa corta por rotação, entrando-se aqui já com as simplificações anteriores. A equação da secção representa-se a seguir

$$Secd = \frac{p \times \frac{F}{S}}{N^{\circ} \text{ dentes}} \quad (2.21)$$

2.2.1.3.4 Força de corte por dente

A força de corte exercida por cada dente da fresa, é proporcional á secção de cada apara arrancada por cada dente. Como esta é inferior, no caso de termos uma ferramenta com múltiplos dentes, também a respectiva força o é. É de notar que cada dente só exerce força quando em contacto com o material. A força exercida por cada dente é representada pela seguinte equação.

$$Fcd = fc \times p \times \frac{\frac{F}{S}}{N^{\circ} \text{ dentes}} \quad (2.22)$$

2.2.1.3.4.1 Força total de corte

Se considerarmos N_c , o numero de dentes que num determinado instante se encontram em contacto com o material, temos que a força total de corte exercida na ferramenta é

$$F_{tc} = \sum_1^{N_c} F_{cd} \quad (2.23)$$

Note-se, a força de corte total é sensivelmente igual à força de corte considerando que existe apenas uma aresta de corte, no entanto a força é mais uniforme ao longo do tempo.

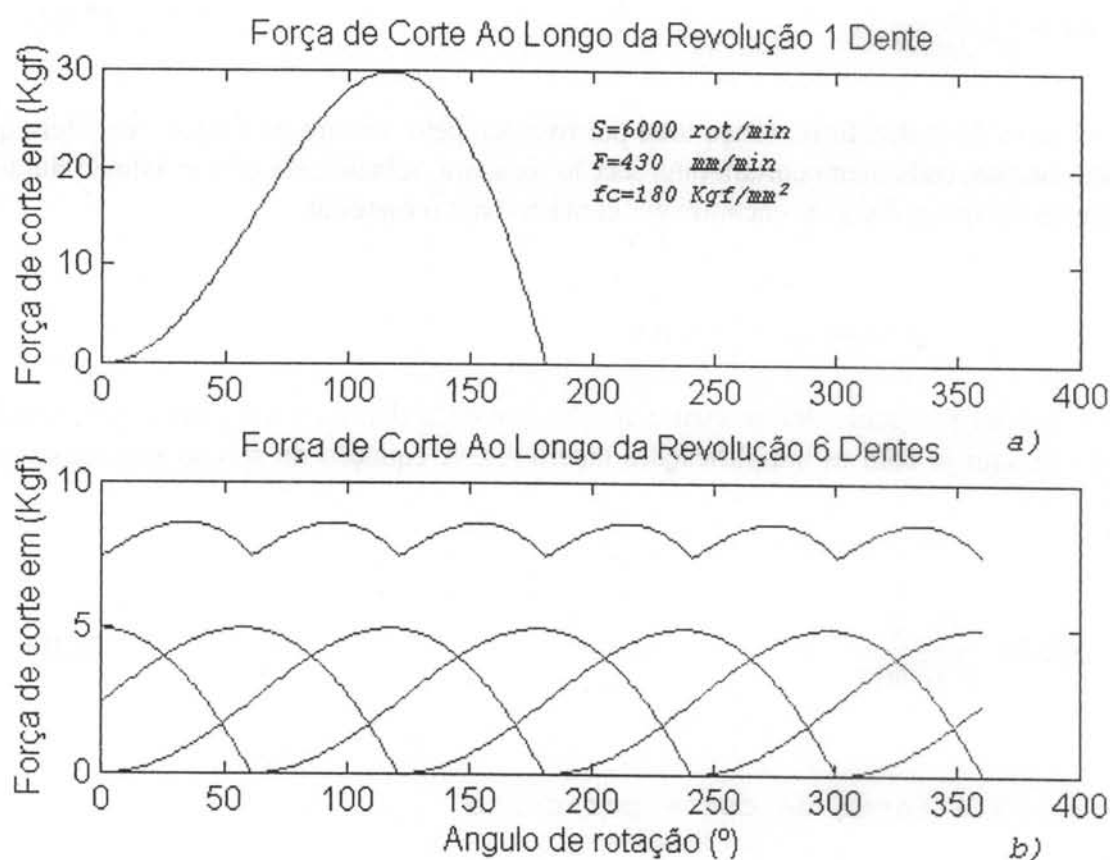


Gráfico 2.4 - Variação da força de corte função do numero de dentes da ferramenta

No Gráfico 2.4, é possível comparar como se distribuem as forças de corte ao longo de uma revolução.

Na alínea *a*), temos o caso da ferramenta ter apenas uma aresta de corte, a força está toda concentrada nessa aresta, e apenas actua entre os 0° e os 180° , com um valor máximo de 30 Kgf , e nulo ao longo do resto da revolução, enquanto que na alínea *b*), em que se usa uma ferramenta com 6 arestas de corte, temos a força distribuída ao longo dos 360° , sendo portanto mais uniforme e tendo um máximo de 10 Kgf , - 3 vezes menor que no caso anterior.

Na alínea *b*), as várias linhas inferiores representam a força aplicada a cada dente, a linha superior representa o somatório das forças dos vários dentes que se encontram em contacto com o material, em cada instante.

A força de corte por dente têm sempre uma direcção perpendicular ao raio da ferramenta, sendo aplicada no ponto (ou linha) de contacto com o material. Quando se usa uma ferramenta com 1 ou 2 dentes, a força total é sempre a força exercida por um dos dentes, pois em qualquer instante que consideramos, somente um dos dentes se encontra em contacto com o material.

Se a ferramenta tiver 3 ou mais dentes, a força total de corte é o somatório vectorial das várias forças de corte dos dentes em contacto com o material, neste caso poderá existir alguma deflexão da ferramenta, pois do somatório vectorial das forças resultará uma força cuja direcção não será perpendicular ao raio da ferramenta.

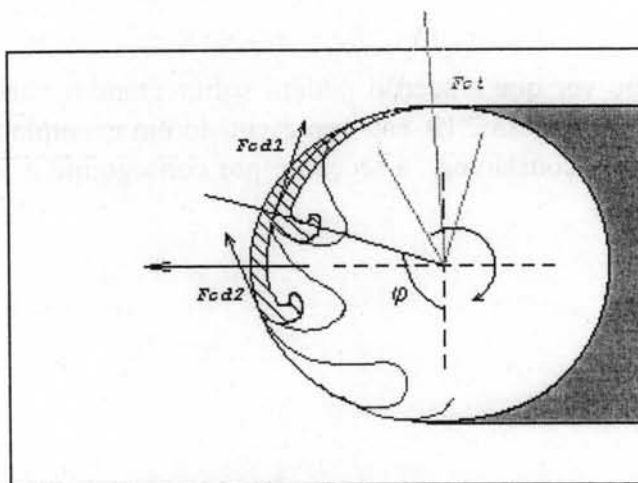


Figura 2.9 - Esquema de forças na ferramenta

Pela observação da Figura 2.9 podemos ver como seria a força aplicada ao eixo da ferramenta. Trata-se apenas de um exemplo, pois a ferramenta tem mais dentes que não foram considerados no somatório das forças, mas dá para ter uma ideia da direcção da força total, esta, não será nunca perpendicular ao raio da ferramenta, exceptuando os casos especiais de 0° e 180° em que as respectivas forças são nulas.

2.2.2 Força de avanço

No momento do corte, e para além da força de corte, existem outras forças aplicadas á ferramenta, das quais se destaca a força de avanço. Esta força será mais uma componente a acrescentar no calculo da força total.

Apesar da sua importância para a determinação do desgaste da ferramenta, esta força não foi considerada no nosso estudo pelo simples facto que pretendemos usar como sensor da força de corte um sensor que permite medir, indirectamente a força de corte, medindo o binário do motor.

Podemos definir a força de avanço como a força exercida pela ferramenta sobre o material, segundo a direcção de avanço. Esta força permite definir o estado de afiação da ferramenta, pois esta é função da capacidade da ferramenta penetrar no material. Esta força poderá também dar uma ideia da deflexão da ferramenta, segundo a sua direcção de avanço. No entanto não influencia o corte do material.

2.2.3 Influencia da geometria da peça na força de corte

Considerou-se até aqui que a força de corte se mantém constante ao longo de todo a trajectória, mas tal só é verdade se a secção se mantiver constante, e esta depende da geometria da peça.

Podemos facilmente ver que a secção poderá sofrer grandes variações ao longo da maquinação de uma peça. Na Figura 2.10, está representado um exemplo de como mantendo todos os parâmetros de corte constantes, a secção e por conseguinte a força, podem variar com a geometria da peça.

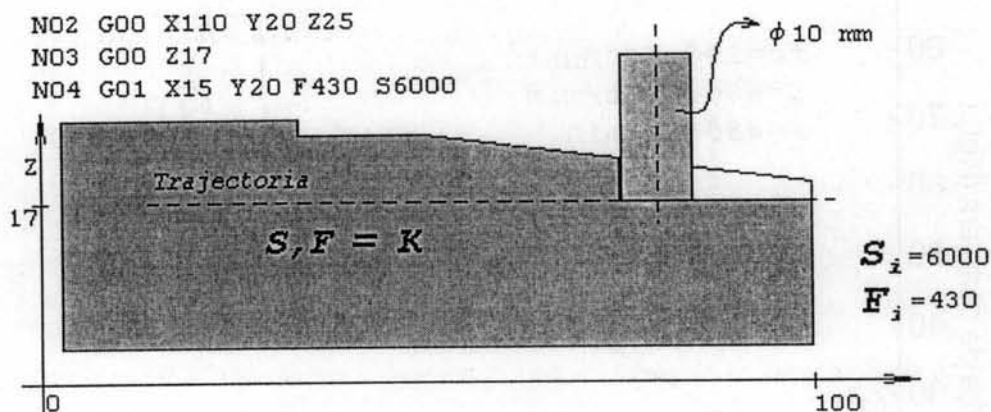


Figura 2.10 - Exemplo da variação da geometria da peça ao longo da trajectória

O exemplo da Figura 2.10, que não passa de um exemplo ilustrativo do que poderá acontecer, é, de certa forma, exagerado, quando comparado com a realidade, mas permite-nos uma melhor demonstração da influencia da variação da geometria do material a maquinar.

Neste exemplo considerou-se que o material a maquinar tem uma geometria irregular, pretendendo-se obter uma geometria regular. No entanto o contrario também poderá acontecer, tendo o material a maquinar uma geometria regular e a peça a obter uma geometria irregular, mantendo-se o problema.

No caso da Figura 2.10, estamos a considerar que a geometria do material a maquinar varia ao longo da trajectória, de tal forma que se mantivermos o parâmetros constantes ao longo do corte, a força de corte variará como podemos observar no Gráfico 2.5.

A variação da força de corte, deve-se simplesmente ao aumento da secção de corte, que é variável e função apenas da profundidade de corte, que por sua vez vai aumentando. A curva apresentada no Gráfico 2.5, foi calculada a partir das equações definidas atrás, mais precisamente da equação 2.19.

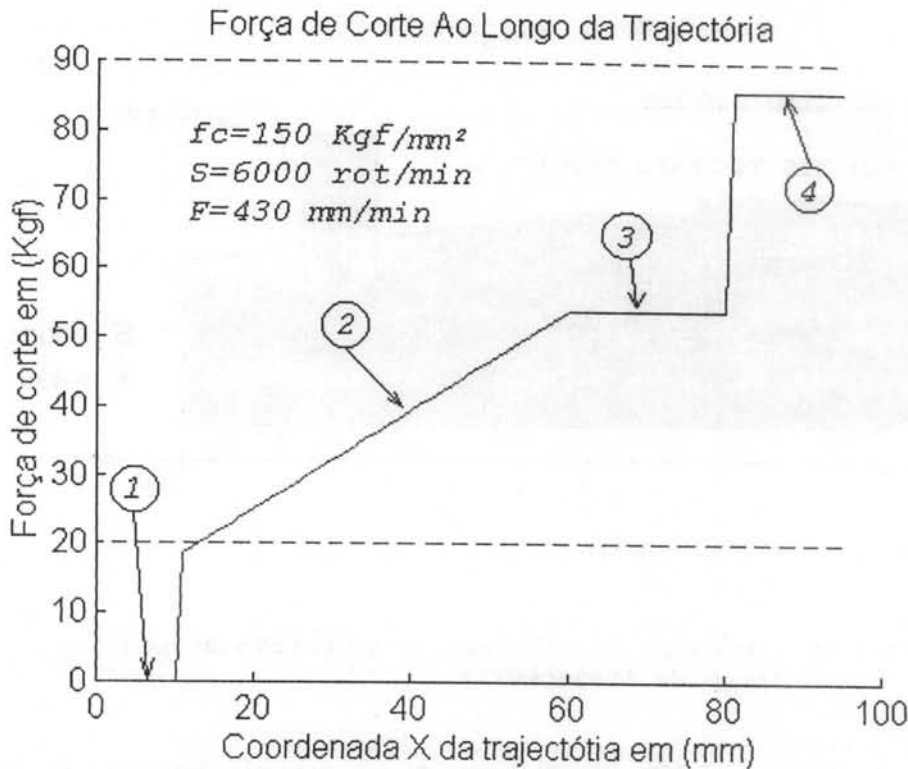


Gráfico 2.5 - Variação da força de corte ao longo da trajectória

2.2.3.1 Variação da força de corte

Como podemos observar, a força de corte está constantemente a variar ao longo das trajectórias definidas no programa. A trajectória definida no programa é composta pelos movimentos de 1 a 4.

Durante o movimento 1, assinalado no Gráfico 2.5, a força de corte é nula, devido ao facto de a ferramenta se encontrar a maquinar fora do material. Em relação ao movimento assinalado pelo numero 2, a força sofre um aumento brusco no início, devido á entrada da ferramenta no material, aumentando lentamente em função do aumento da profundidade de corte.

Ao longo do movimento assinalado pelo numero 3, a força de corte é constante, e o seu valor igual ao ultimo valor da força do movimento anterior. No movimento 4, o importante é o início da trajectória, pois é neste ponto que a força de corte varia bruscamente, mantendo-se constante ao longo do resto da trajectória.

Os movimentos de 1 até 4 dão-se de uma forma continuada, constituindo um movimento único, aqui dividido apenas para podermos observar o que se passa em cada troço da peça, correspondente ás variações do material.

Como podemos deduzir do exposto, a força de corte sofre variações de várias naturezas ao longo de uma trajectória, tendo um numero infindável de variações possíveis, só dependendo da geometria do material e da peça a maquinar.

Nos dias de hoje, produzem-se materiais especiais cujo a dureza varia ao longo da secção do material, como exemplo temos alguns materiais cujo o núcleo têm uma dureza inferior à dureza da superfície.

No corte de material deste tipo temos ainda um outro factor que influencia a força de corte, ou seja, a força específica de corte poderá variar ou ao longo da maquinação ou segundo a direcção de maquinação. Estas variações afectam de igual modo o desempenho do corte.

2.2.3.2 Uso da ferramenta de geometria variável

A optimização das operações de maquinação depende, entre outros factores, da eficiência das ferramentas de corte. Esta eficiência passa pela selecção do material da ferramenta, da geometria e da utilização das condições de corte correctas [2]. Destes factores, depende a vida útil da ferramenta de corte e, como consequência a estratégia a implementar para a sua substituição ou tratamento.

As ferramentas são fabricadas para suportarem uma determinada força de corte, e esta será a força que optimizará o uso da ferramenta. A força especificada para a ferramenta não deve ser ultrapassada, com o prejuízo de se desgastar a ferramenta rapidamente.

O uso da ferramenta com forças muito inferiores á força especificada, faz com que se esteja a desaproveitar a ferramenta, e por conseguinte a aumentar o tempo de maquinação, não implicando isso, que seja proporcionalmente menor o desgaste.

Quando se tem uma geometria variável do género da apresentada na Figura 2.10, e seguindo o processo convencional, o responsável pelo desenvolvimento do programa de maquinação terá que escolher parâmetros que permitam o uso de determinada ferramenta. Perante as variações da força de corte existentes, o problema poderá ser abordado de duas maneiras diferentes, consoante a importância dada á optimização da maquinação.

- escolha dos parâmetros de corte de modo a satisfazer o caso mais desfavorável.
- ajuste das trajectória do programa á geometria do material.

No primeiro caso, trata-se de verificar qual será ao longo de toda a trajectória, o ponto onde a ferramenta exerce a maior força. Ao seleccionar os parâmetros de corte para o pior caso, significa que se está a desaproveitar a ferramenta ao longo do resto da trajectória.

Assim, não existe a preocupação do programador em otimizar a ferramenta, quando muito o programador não escolhe o pior caso, mas sim um ponto intermédio da trajectória para definir os parâmetros. A ferramenta é otimizada nas zonas em que a força é menor, no entanto em zonas onde a força é maior, está a desgastar-se a ferramenta rapidamente e a provocar-se um mau acabamento á peça.

No segundo caso, o programador já teve alguma preocupação em otimizar a operação, conseguindo isso dividindo a trajectória apresentada na Figura 2.10, em várias sub-trajectórias onde a força se mantém constante, ou seja, zonas cuja profundidade de corte se mantém, constante.

A subdivisão, no caso da Figura 2.10, poderia ser feita segundo os diferentes troços da trajectórias indicadas, ajustando para cada uma delas os parâmetros de corte. Note-se que existe, no entanto, o troço da trajectória assinalada pelo numero 2, onde isso se torna difícil, devido ao plano ser inclinado, sendo a única solução para este caso, criar um numero bastante elevado de sub-trajectórias, para este troço de maquinação.

Um numero elevado de sub-trajectórias provoca programas de maquinação bastante grandes e um esforço de projecto elevado para se otimizar o corte.

Nesta tese, o autor tenta demonstrar que existe uma outra possibilidade mais eficiente para resolver este problema.

2.2.4 Influência do estado da ferramenta no corte

Durante a execução de qualquer operação de maquinação, a ferramenta de corte vai-se desgastando de forma mais ou menos rápida, o que conduz, inevitavelmente, à necessidade da sua substituição ou tratamento. Este facto tem implicações de natureza económica, que condicionam a selecção dos parâmetros de maquinação.

Pode referir-se desde já que, se a vida da ferramenta for excessivamente curta, o processo terá de ser interrompido, com uma frequência exagerada, para tratamento ou substituição da ferramenta, o que acarreta aumento do tempo de produção e por conseguinte o aumento dos custos de produção.

No extremo oposto, se pretendermos prolongar excessivamente a vida da ferramenta, seremos obrigados a usar parâmetros de corte que conduzem, a tempos de produção demasiado elevados, com idênticas consequências anti-económicas.

Verifica-se assim que o conhecimento da evolução do desgaste das ferramentas está na base das opções tecnológicas que condicionam os aspectos económicos da produção por maquinação[3].

O estado da ferramenta, mais precisamente o seu desgaste, também influencia a força de corte exercida. Conforme se vai desgastando a ferramenta, esta começa a exercer uma maior força para cortar o material, influenciando assim a força de corte e a força de avanço.

Neste caso a força de corte é a variável independentemente dos parâmetros de corte e da geometria da peça, e o trabalho desenvolvido pelo programador a otimizar o programa de maquinação será um pouco inglório, pois ele não poderá prever o estado da ferramenta ao longo da maquinação ou ao fim de maquinar N peças.

Mesmo que fosse possível ao programador saber qual o desgaste da ferramenta em função do numero de horas de maquinação, este teria que desenvolver vários programas de maquinação para manter a mesma força de corte, conforme se desgastava a ferramenta, à medida que a máquina ia produzindo as peças.

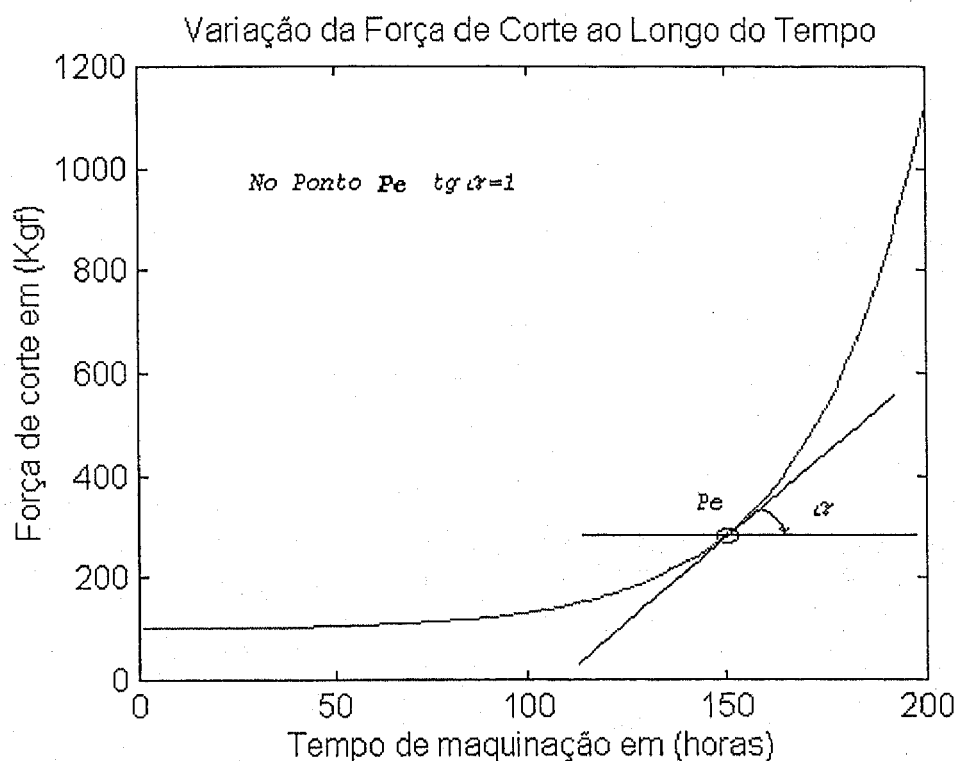


Gráfico 2.6 - A força exercida pela ferramenta ao longo do seu tempo de maquinação

O facto da ferramenta ter um certo desgaste, como já vimos, provoca um aumento da força de corte e da força de avanço. O aumento desta provoca um aumento do atrito e por conseguinte um aumento da temperatura acentuando o desgaste da ferramenta e exigindo da máquina mais potência para executar a maquinação da mesma secção, o que tende a baixar o rendimento geral da máquina.

O aumento da força de avanço também é prejudicial ao corte, pois provoca um aumento da flexão da ferramenta segundo a direcção de avanço, resultando em mau acabamento. Se a força de avanço continuar a aumentar poderá mesmo provocar a quebra da ferramenta.

A força de corte aumenta devido ao desgaste da ferramenta seguindo uma curva mais ou menos idêntica à indicada no Gráfico 2.6.

A deflexão da curva pode aparecer mais cedo ou mais tarde conforme a ferramenta se encontra a funcionar nas condições normais ou não. Como podemos observar, a partir do momento em que a ferramenta atinge um determinado nível de desgaste, a força de corte começa a aumentar de tal forma que provoca um desgaste cada vez maior, conduzindo ao seu fim aproxima-se de uma forma bastante acelerada.

Se levarmos uma ferramenta à exaustão, para além do seu tempo de vida, o que deverá ser evitado, a força de corte pode ser de tal forma grande, que ultrapassa a tensão de rotura da ferramenta, partindo-a, implicando normalmente a destruição da peça que se encontra a maquinar.

Se considerarmos um ponto na curva, onde a derivada é igual a 1, como exemplo, podemos afirmar que é possível deslocar esse ponto ao longo do eixo dos tempos, consoante se usa a ferramenta correcta ou incorrectamente.

O desgaste da ferramenta depende do tempo de maquinação e da velocidade de corte, isto é, a ferramenta desgasta-se ao longo da operação, e a sua taxa de desgaste depende da velocidade empregue no processo. Quanto mais rápido se trabalha, maior é o desgaste.

Se relacionarmos o desgaste com a quantidade de material removido, podemos afirmar que a quantidade de material removido por uma ferramenta até esta ter de ser de novo tratada diminui quando esta está a operar a velocidades acima das normais. Por outro lado a quantidade de material removido não aumenta proporcionalmente quando se usam velocidades de corte inferiores á normal. O Gráfico 2.7 dá uma ideia de como varia o desgaste da ferramenta em função dos parâmetros de corte (velocidade).

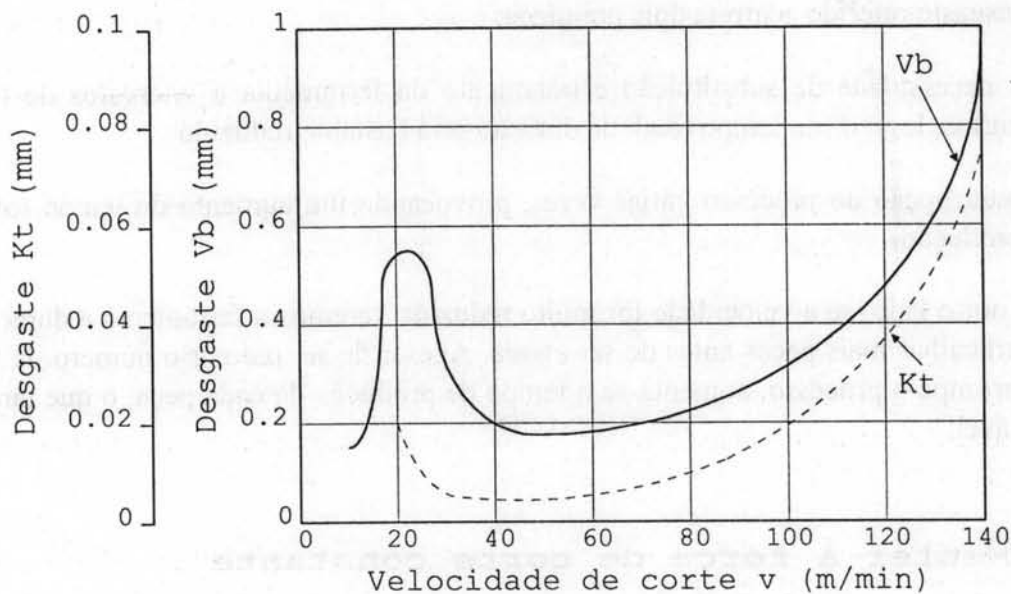


Gráfico 2.7 - Curva de Desgaste/Velocidade de corte para um tempo de corte constante de 60 minutos [3]

Se fixarmos o tempo de corte podemos representar o desgaste da ferramenta em função da velocidade. As curvas obtidas, desgaste-velocidade de corte, contêm informações importantes e estão directamente relacionadas com os mecanismos envolvidos na degradação das ferramentas.

O exemplo do Gráfico 2.7 foi obtido a partir das curvas de desgaste, considerando um tempo de corte de 60 minutos. A forma desta curva é típica da maquinação de aços de construção e o pico de desgaste à volta dos 20 m/min está associado ao comportamento da aresta postíça de corte.

Resumindo, existe um ponto óptimo de funcionamento para ferramenta, e operar com valores acima ou abaixo deste ponto não trás qualquer vantagem, antes pelo contrario, em qualquer das duas situações o rendimento será sempre inferior.

A forma correcta de definir a duração da ferramenta é relacioná-la com a quantidade de material removido. No entanto, esta definição não nos serve, pois a ferramenta terá que trabalhar diferentes tipos de material e em situações muito diversas, e isto significa que teremos de continuar a relacionar a duração da ferramenta com o tempo.

O desgaste da ferramenta não é linear com o tempo, pois a operação pode demorar mais tempo, mas com parâmetros de corte melhor seleccionados, e a ferramenta poderá ter um desgaste menor.

É no entanto inevitável o desgaste da ferramenta, logo há que procurar um compromisso, entre o desgaste e a duração da maquinação. Para tempos reduzidos, a ferramenta terá que trabalhar em condições mais duras, provocando um desgaste maior.

O desgaste referido acarreta dois prejuízos:

- a necessidade de substituição e tratamento da ferramenta a intervalos de tempo curtos, logo o seu tempo total de duração será bastante reduzido
- interrupção do processo várias vezes, provocando um aumento do tempo total de produção

Por outro lado, se a velocidade for muito reduzida, teremos a ferramenta a durar mais tempo e a trabalhar mais peças antes de ser afiada. Apesar de se reduzir o numero de vezes que se interrompe o processo, aumenta-se o tempo de produção de cada peça, o que também não é desejável.

2.3 Manter a força de corte constante

Como vimos atrás, a ferramenta é projectada para exercer sobre o material a cortar uma determinada força, força essa que será escolhida de tal modo que seja optimizada a duração da ferramenta. Esta força está também limitada pela força que a ferramenta é capaz de suportar, sem que a sua flexão influenciará o processo de maquinação.

O aumento da força de corte da ferramenta, para além do valor considerado como óptimo, provoca um desgaste muito mais rápido da ferramenta, o que é equivalente a fazer deslocar o ponto referido atrás (Pe), na direcção do zero do eixo dos tempos. Este desgaste deve-se não só ao aumento da força, mas também ao aumento da temperatura na ferramenta, devido ao aumento do atrito. Provoca-se um desgaste, que por sua vez faz aumentar ainda mais o atrito, e por conseguinte, um maior desgaste, entrando-se num ciclo em que a ferramenta se deteriora muito rapidamente.

Facilmente se deduz, pelo atrás exposto, que se mantivermos a ferramenta a funcionar com a força para a qual ela foi projectada, estamos a trabalhar na zona óptima de funcionamento, tanto a nível de desgaste como de acabamento superficial da peça. Este facto minimiza os custos envolvidos na produção de peças.

No exemplo mostrado na Figura 2.10, se quisermos aproveitar a ferramenta o mais possível, teremos que saber qual a força especificada para a ferramenta e ajustar constantemente os parâmetros de corte, de modo a manter a força constante, ao longo de toda a trajectória. A variação da secção, ou do desgaste da ferramenta, pode ser compensado de tal modo que a força exercida pela ferramenta, não provoque o seu desgaste rápido.

Ao manter a força de corte constante, reduzindo os parâmetros de maquinação, o ponto Pe atrás referido nunca chega a ser atingido, mantendo-se o desgaste da ferramenta sempre num nível baixo, pois a temperatura não terá tendência a aumentar, e provocar um ciclo de desgaste rápido, aumentado-se assim o tempo de vida útil da ferramenta.

O aumento do tempo de vida da ferramenta, usando esta técnica, não sacrifica muito o tempo de maquinação, sendo certo que caso isso se comece a verificar, ou quando a força de corte medida for de tal modo grande que leve a que os parâmetros de corte saiam da zona normal de funcionamento, significará que a ferramenta está a precisar de ser substituída.

O controlo da força de corte não influencia apenas o acabamento da peça e o desgaste da ferramenta, influenciando também o tempo de maquinação. Podemos ver através de um exemplo, como isso pode acontecer.

Vamos considerar que pretendemos maquinar uma secção idêntica à secção a tracejado na Figura 2.11,

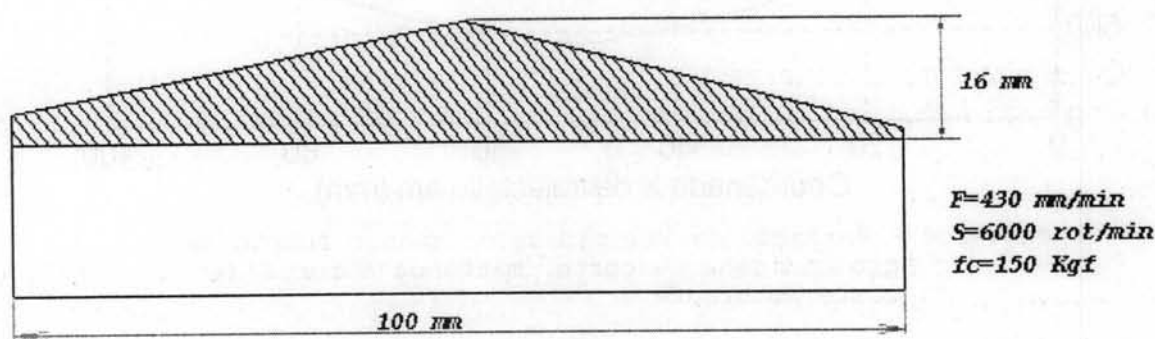


Figura 2.11 - Exemplo do cálculo de custos para o corte da secção a tracejado

Num processo tradicional, se pretendemos maquinar tal secção, teríamos que seleccionar os parâmetros de corte, para a maior profundidade de corte, que neste caso é de 16 mm. O parâmetros seleccionados estão os representados na Figura 2.11, parâmetros estes que nos dão uma ideia teórica, do valor da força de corte indicada para o conjunto ferramenta/material.

Usando a equação 2.9, podemos calcular a força de corte ideal, cujo resultado seria **172 Kgf**, se mantivermos esta força de corte constante ao longo de toda a maquinação.

Se fizermos mais uma simplificação, que é considerar que a velocidade de rotação se mantém constante, isto é, $S=6000 \text{ rot/min}$, de modo a apenas termos que alterar a velocidade de avanço, podemos calcular qual o valor desta ao longo da trajectória usando as equações referidas atrás. O Gráfico 2.8, mostra como deveria variar a velocidade de avanço, ao longo da trajectória de corte, função apenas da profundidade de corte, enquanto se mantém a velocidade de rotação e a força de corte constante e iguais respectivamente a **6000 rot/min** e **172 Kgf**.

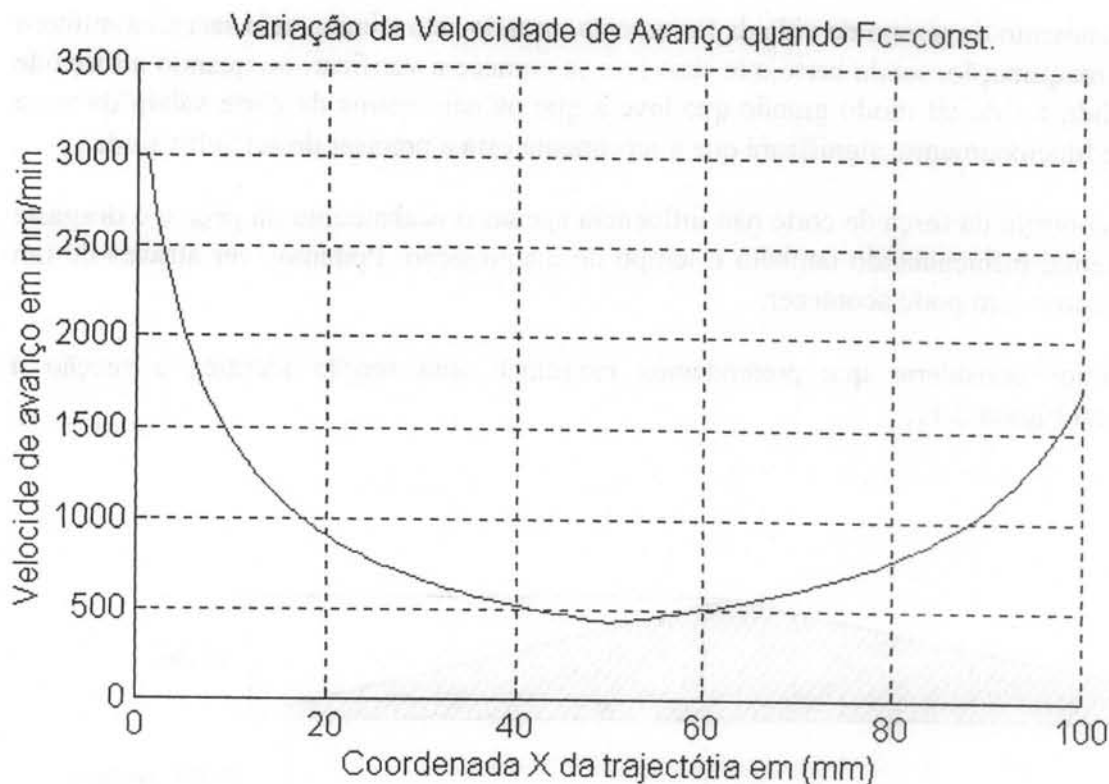


Gráfico 2.8 - Variação da Velocidade de avanço função da profundidade de corte, mantendo a força de corte constante

Em ambos os casos, poderemos calcular qual seria o tempo gasto na maquinação da secção.

Para o primeiro caso, este calculo é bastante simples. Como temos uma velocidade de avanço constante de **430 mm/min**, e um espaço a percorrer de **100 mm**, para estes valores o tempo do percurso seria de **13.95 s**.

No segundo caso, este calculo torna-se mais complexo, pois, a velocidade de avanço não é constante. Para se resolver o problema, correu-se a um pequeno programa em MatLab, cujo objectivo é o calculo do somatório de tempos parciais (tempo gasto a percorrer uma unidade comprimento δx considerando que a velocidade de avanço durante esse período se mantém constante), obtendo-se assim o tempo total do percurso. O resultado foi de **8.29 s** ou seja 40.6% mais rápido.

Estes são valores de grande importância para a industria, pois estão directamente relacionados com o custo final do produto.

Estes cálculos e gráficos são apenas exemplos ilustrativos do funcionamento do processo de maquinação, tendo como objectivo expor de uma forma simplificada esse funcionamento.

É igualmente possível também usar o controlo da força, para compensar excesso de vibrações de maquinação, que normalmente provocam um mau acabamento e um deterioramento da máquina e da ferramenta. Esta será uma hipótese a estudar durante a implementação do sistema.

2.4 Conclusões

No processo de maquinação, estão envolvidos um numero bastante grande de parâmetros que o influenciam

- o acabamento superficial da peça
- o desgaste da ferramenta usada.

Os dois factores anteriores influenciam

- a qualidade final da peça
- o tempo envolvido na maquinação.

Um dos factores mais importantes, e que não é controlado normalmente no processo de maquinação, é a força de corte da ferramenta. Esta força de corte influencia tanto o acabamento superficial da peça, como o desgaste da ferramenta ao longo do processo de maquinação, e influencia também o tempo de maquinação.

Se for possível controlar a força de corte, mais precisamente, manter esta força constante e igual à força especificada para a ferramenta, obteremos

- um acabamento superficial de melhor qualidade
- uma maior duração da vida útil da ferramenta de corte
- um tempo de maquinação menor.

Todos estes factores, quando optimizados, permitem reduzir o custo de produção.

É importante lembrar que a maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições. A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

2.4 Conclusões

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

A maquinação é um problema de otimização, que envolve a busca por uma solução ótima, dada uma função objetivo e um conjunto de restrições.

3. Sistema de Controlo

3. Sistema de Controlo..... 57

3.1 Sumário	58
3.2 O controlo	58
3.3 O controlo da força de corte	61
3.3.1 O Decisor	61
3.3.2 O Actuador	63
3.3.3 Controlo não linear	66
3.3.4 Variáveis de Controlo	70
3.4 Arquitectura do controlador	73
3.4.1 Principais problemas de controlo	74
3.4.2 O Actuador	74
3.4.3 Conclusões	76

3.1 Sumário

Este capítulo inicia-se com uma ligeira introdução ao controlo, e à sua evolução ao longo do tempo, e a uma aplicação numa área específica, ainda com uma dificuldade de controlo relativamente elevada.

Segue-se uma descrição, do ponto de vista do controlo, do problema que se pretende resolver, bem como do processo a controlar. Também, e de uma forma lata, será exposta a arquitectura a usar pelo controlador.

Com esse objectivo, são descritos os problemas associados aos sistemas não lineares, bem como possíveis técnicas a usar neste tipo de sistemas.

Também se aborda o controlo intrínseco á máquina e os parâmetros que lhe estão associados. Será descrito o controlo que se pretende implementar e as suas dependências.

3.2 O controlo

Desde os tempos mais remotos que se utilizam sistemas de controlo. Sistemas simples e mecânicos eram já empregues na antiguidade para controlar dispositivos de água, por exemplo. Estes sistemas naturalmente evoluíram, e com a descoberta do motor (a vapor, e posteriormente, o motor de combustão), começaram a surgir sistemas de controlo mecânico cada vez mais complexos, de que é exemplo o controlo de velocidade exposto na Figura 3.1.

Com o evoluir do motor eléctrico e o desenvolvimento, nos tempos mais recentes, da electrónica e, ainda mais próximo de nós, do microprocessador, o homem chegou a pensamentos abrangentes quanto ao controlo, trabalhou teorias, desenvolveu métodos e criou controladores padrão, que se adaptam á maioria das situações de controlo.

É possível hoje em dia comprar controladores genéricos, em que é necessário apenas configurar os seus parâmetros, para que se adaptem ao controlo do sistema em causa.

Contudo, e porque na realidade existem processos e situações que têm um grau de dificuldade acrescido, é impossível, para determinados sistemas, seleccionar um conjunto de parâmetros capazes de adaptar os referidos controladores genéricos a tais processos.

Assim sendo, o controlo desses processos implicará a procura de soluções específicas, envolvendo técnicas ainda em desenvolvimento. Neste tipo de controlo inclui-se o designado normalmente de controlo adaptativo, significando esta designação que o controlador, ao longo do seu funcionamento, se vai modificando de modo a adaptar-se a situações de controlo não previstas.

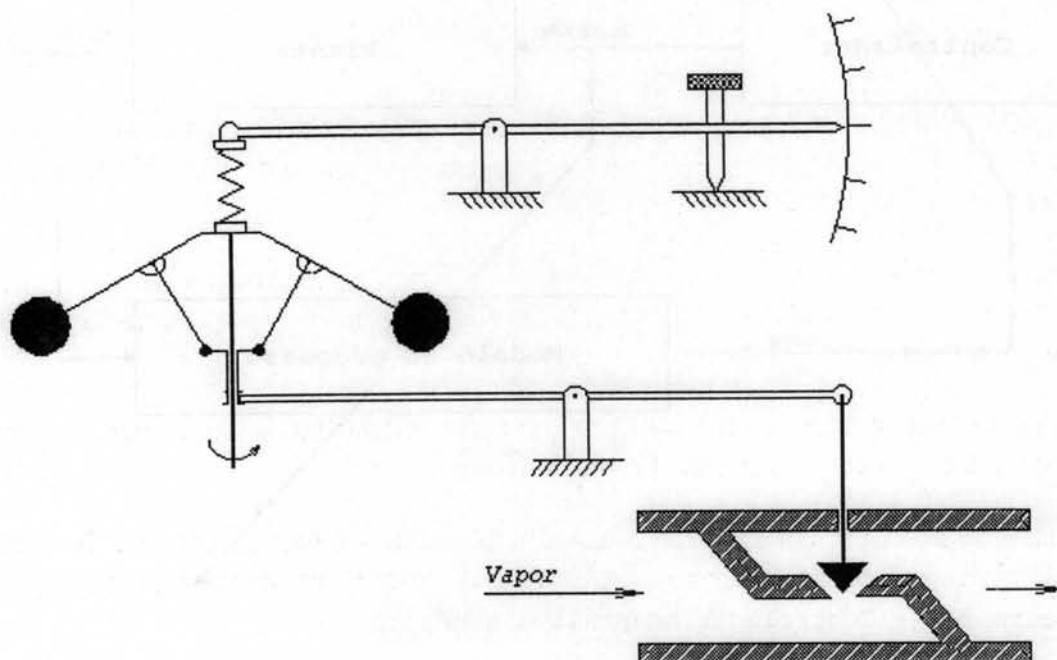


Figura 3.1 - Controlo de velocidade de uma máquina a vapor

Duas técnicas de controlo têm mostrado potencialidade no campo do controlo adaptativo, a Lógica *Fuzzy* e as Redes Neurais. Trabalhos científicos recentes [5] demonstram como estas técnicas podem ser aplicadas ao controlo de um modo bem sucedido.

Nesta tese, o autor procura aprofundar técnicas que permitam o controlo do processo de corte.

A Figura 3.2 representa um sistema de controlo adaptativo genérico. Um sinal de referencia é aplicado ao controlador, que, por sua vez, excita a planta, aplicando à sua entrada um sinal de controlo.

A planta reage ao sinal de entrada, produzindo uma saída, saída esta aplicada ao modelo do processo (modelo da planta) que se pretende controlar. O modelo do processo permite gerar um erro, identificativo de quanto a saída do controlador se afastou do valor pretendido. Esse sinal de erro serve para modificar os parâmetros do controlador, adaptando este a uma nova situação

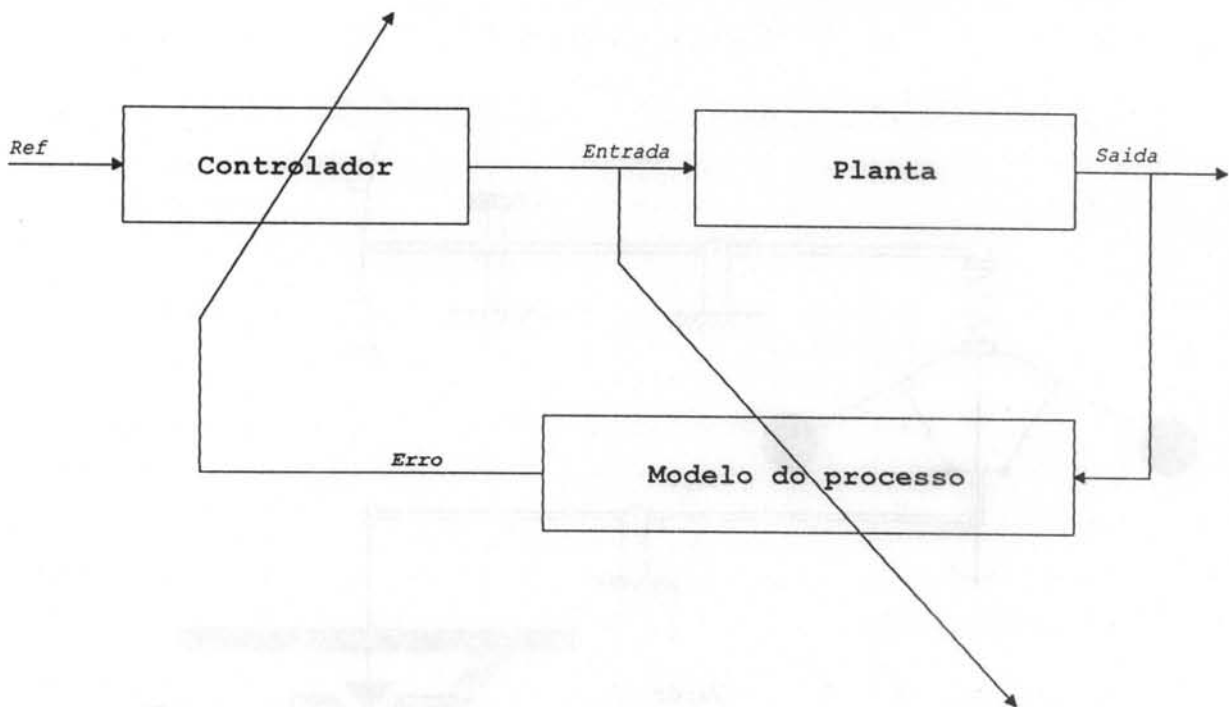


Figura 3.2 - Controlador adaptativo genérico

O modelo do processo, deve ser capaz de; perante um determinado valor da saída da planta, gerar o respectivo valor de entrada que lhe deu origem, isto é, o modelo do processo, deve identificar o valor apresentado a entrada.

O modelo do processo, também designado de identificador, pode ser implementado analiticamente caso se conheça a função de transferência da planta.

Por exemplo, se tivermos $y(t) = f(u(t))$ e se f for conhecida e invertível, temos $u(t) = f^{-1}(y(t))$, sendo portanto f^{-1} a função que descreve o identificador.

Para processos complexos, a função f não é conhecida, nem tão pouco invertível, pelo que é necessário recorrer a outras técnicas, como por exemplo a Redes Neurais Artificiais, para implementar o identificador.

Quando se implementa o identificador usando Redes Neurais Artificiais, é necessário recolher informação da entrada da planta, para que a rede seja treinada de modo a aproximar o mais possível a função de transferência da planta.

3.3 O controlo da força de corte

Do ponto de vista da teoria de controlo, o processo de corte pode ser considerado complexo, existindo um conhecimento reduzido acerca deste e das possibilidades do seu controlo. Até ao momento ainda não foi encontrada uma solução aceitável para lidar com tais processos [10].

O Instituto de Automação Industrial, que já referimos, está neste momento a desenvolver um controlador para o processo de corte, baseado em Lógica Fuzzy.

Dada a cooperação existente entre aquele instituto e o CCP - Centro de CIM do Porto, pareceu de grande interesse estudar a utilização de Redes Neurais Artificiais como aproximação ao problema de controlo destes processos, e comparar os resultados.

Para definir este tipo de controlo recorreremos, a modelos do processo, a modelos de controlo e à sua adaptação recorrendo à aprendizagem.

No capítulo anterior, tentou-se transmitir a ideia de como se processa o corte numa máquina de fresagem e quais os parâmetros envolvidos. Também no capítulo anterior, mostrou-se quais as vantagens que resultariam do controlo da força de corte.

O problema de controlar a força de corte põe-se a dois níveis hierárquicos distintos

- a um nível superior, temos o problema de definir qual a força de corte, que melhor otimiza o processo de corte. Vamos designar este nível de *Decisor*
- a um nível inferior, é necessário um controlador, que mantenha essa força de corte constante, independentemente, das variações da geometria da peça, do desgaste da ferramenta, das variações da dureza do material ou outras perturbações do sistema. A este nível vamos chamar de *Actuador*.

3.3.1 O Decisor

O controlador desenvolvido para o Decisor, decidirá qual a melhor força a usar no corte, isto é, a força de corte que minimizará os custos de maquinação.

Os custos de maquinação envolvidos são função do

- tempo de maquinação
- desgaste da ferramenta
- qualidade final da superfície de acabamento

Existem provavelmente outros parâmetros, que influenciarão os custos de maquinação, mas estes não estão ligados directamente ao processo, mas sim a custos marginais da empresa, tais como custos de manutenção, custos de secretaria etc. Este custos marginais da empresa poderão ser ignorados, em questões de minimização dos custos para a selecção da força de corte, pois são constantes e independentes do processo de corte.

A função custo define uma superfície tridimensional, que terá pelo menos um ponto onde existe um mínimo da função, mas nada nos garante que não apresente vários pontos onde existam mínimos locais. Será preocupação do sistema Decisor, encontrar pelo menos um dos mínimos que optimize a operação. Uma das características principais deste sistema é a sua capacidade de evoluir, adaptando-se cada vez melhor ao processo.

No custo de produção da peça, consideramos como parâmetros principais o tempo e o desgaste da ferramenta, sendo que o tempo depende essencialmente da qualidade final que pretendemos obter.

O desgaste da ferramenta e a qualidade final pretendida, da superfície, influenciam o tempo de produção. O tempo de maquinação, o desgaste da ferramenta e a qualidade final da superfície dependem da força de corte exercida pela ferramenta sobre o material.

É a força de corte que define o acabamento superficial e o tempo de maquinação e que influencia o desgaste da ferramenta. Forças de corte elevadas implicam tempos reduzidos e rápido desgaste da ferramenta. Forças de corte reduzidas, implicam um menor desgaste da ferramenta e tempos de produção elevados.

A qualidade implica uma determinada força de corte, que, caso seja demasiado elevada, pode provocar vibrações e flexão da ferramenta, diminuindo a qualidade final de acabamento. Forças de corte demasiado reduzidas, poderão provocar um mau acabamento por sobreaquecimento do material.

A acrescentar a isto, os custos poderão não ser uniformemente distribuídos, isto é, em função do tempo de maquinação, o custo terá sempre limites inferiores e limites superiores que impedirão a função de ser linear.

Temos, portanto, uma função de custo de maquinação bastante complexa, função esta que pretendemos minimizar, actuando na força de corte.

O controlo da força de corte permite-nos escolher o tempo de maquinação, em função da qualidade que pretendemos e do mínimo desgaste da ferramenta.

Só o facto de se manter a força de corte constante em qualquer situação permite-nos minimizar o desgaste da ferramenta e produzir a peça no melhor tempo possível.

Se seleccionássemos a força de corte em função apenas do material e da ferramenta, e se mantivéssemos esta força constante ao longo de todo o processo, já se obteria um tempo de maquinação inferior ao tempo obtido pelos processos tradicionais. Logo teríamos um custo de produção inferior, podendo este ser ou não o mínimo possível.

3.3.2 O Actuator

O controlador de nível inferior tem como função manter a força de corte constante e igual à força de corte de referência, seleccionada pelo Decisor, independentemente da geometria, do desgaste da ferramenta, da variação da dureza do material ou de perturbações do processo.

Existem três parâmetros que influenciam a força de corte, no processo de maquinação

- a secção, relacionada com a geometria da peça
- o desgaste da ferramenta, relacionado com o material e os parâmetros de corte
- a força específica de corte, relacionada com a dureza do material.

A maioria das variações da força de corte deve-se a alterações da geometria da peça. Estas alterações também apresentam normalmente derivadas bastante grandes.

As alterações da força, provocadas pelo desgaste da ferramenta, são limitadas e prolongam-se por períodos de tempo relativamente longos, sendo a sua derivada de fraca amplitude.

A força específica de corte é um parâmetro que normalmente se mantém constante, e que só no caso de certos materiais especiais, apresenta variações, normalmente segundo uma das direcções de corte, podendo no entanto apresentar variações em várias direcções. Exemplos de materiais deste tipo, são aqueles cujo núcleo apresenta uma dureza diferente da superfície.

Para o controlador manter a força de corte constante, é necessário medir esta força, de modo a ser realimentada e comparada com a força de corte de referência, gerando-se assim um erro. O controlador terá que actuar sobre a máquina, alterando-lhe os parâmetros de maquinação, de modo a manter a força de corte constante.

A máquina usa, como parâmetros de maquinação, as coordenadas do movimento, a velocidade de rotação e de avanço da ferramenta. As coordenadas fazem parte da definição da peça, logo não poderão ser alteradas, restando-nos, portanto, como parâmetros de controlo a velocidade de avanço e de rotação.

Como ficou demonstrado no capítulo anterior, a força de corte depende da secção do material a cortar. Para o controlo poderemos definir a força de corte da seguinte maneira:

$$Fc(t) = fc(t) \times Sec(t) \times d_f(t) \quad (3.24)$$

em que:

$fc(t)$ - força específica de corte

$Sec(t)$ - secção de corte

$d_f(t)$ - desgaste da ferramenta.

$fc(t)$ é uma função que normalmente é constante ao longo do tempo e depende apenas das características do material.

$d_f(t)$ é função da ferramenta e, normalmente não entra na equação de cálculo da força de corte, pela dificuldade que existe em saber num determinado instante qual é realmente o estado da ferramenta. $d_f(t)$ depende também da lubrificação e da temperatura de maquinação, mas, no entanto, sabe-se que a força de corte aumenta ao longo do tempo devido ao desgaste da ferramenta.

Embora no capítulo anterior, a equação da força de corte fosse ligeiramente diferente desta, pretendeu-se que a equação (3.1), fosse o mais genérica possível. No entanto, permite-nos observar que a sua dependência da secção de corte, torna possível alterar a força de corte, apenas alterando a secção.

Se pretendemos controlar a secção de modo a manter a força de corte constante, basta-nos alterar a velocidade de rotação e avanço da ferramenta. Esta relação está representada na equação (3.2)

$$Sec(t) = \frac{F(t)}{S(t)} \times p(t) \quad (3.25)$$

em que

$F(t)$ - Velocidade de avanço função do tempo

$S(t)$ - Velocidade de rotação da ferramenta

$p(t)$ - Profundidade do corte.

Portanto, se pretendermos que $Fc(t) = Const$, teremos uma secção a variar em função da força específica de corte, desgaste da ferramenta e da própria geometria da peça.

Se temporalmente for controlada a velocidade de rotação e avanço da ferramenta, é possível alterar a secção da apara, segundo os nossos desejos.

O Gráfico 3.1, tenta demonstrar a influencia dos vários factores na força de corte, no processo de maquinação.

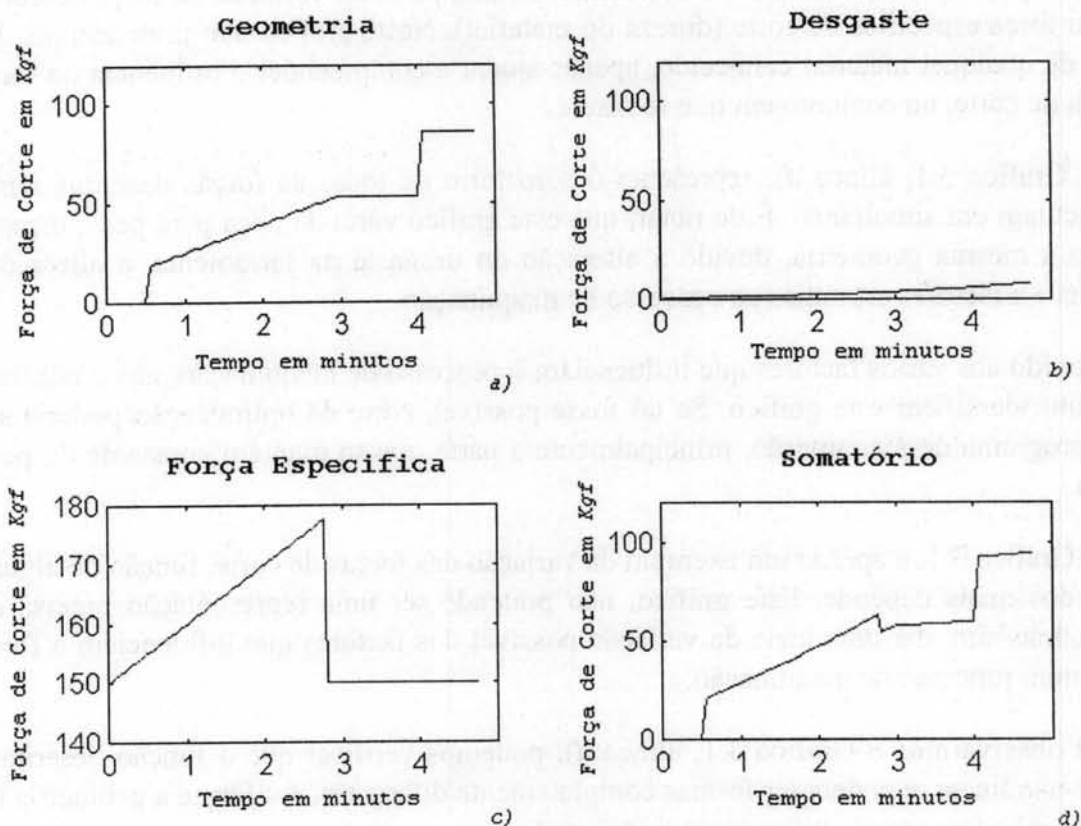


Gráfico 3.1- Exemplo da variação dos parâmetros que influenciam a força de corte

O Gráfico 3.1, alínea *a)*, representa a força de corte, função da geometria da peça. Este gráfico é referente ao exemplo do capítulo 1.

Pelo exposto no capítulo 1, conhece-se empiricamente qual é a variação da força, função do desgaste da ferramenta, observando-se tratar-se de uma função monótona crescente e, com um tempo de acção elevado, quando comparado com tempo associado ao processo de maquinação de uma só peça.

Sendo assim, a influência na força de corte, provocada pelo desgaste da ferramenta, estende-se por várias peças e, portanto é reduzida quando se está a trabalhar ao nível do tempo de maquinação de uma peça. No entanto, esta força modifica-se de peça para peça, alterando-se assim a função da força de corte, ao longo da produção do lote de várias peças. Esta variação está representada na alínea *b)*.

Se observarmos o Gráfico 2.6, do capítulo anterior, onde está representado o desgaste da ferramenta, pode notar-se que a afirmação no parágrafo anterior só é válida até que o desgaste atinja um determinado nível, a partir do qual a força de corte aumenta exponencialmente com o tempo. Isto poderia mesmo acontecer á escala temporal, de produção de uma peça.

Na alínea *c*), é representado um exemplo de uma possível variação da força de corte, função da força específica de corte (dureza do material). Neste gráfico não pretendemos dar exemplo de qualquer material conhecido, apenas ajudar a compreender a influência da força específica de corte, no conjunto em que se insere.

O Gráfico 3.1, alínea *d*), representa o somatório de todas as forças descritas atrás, quando actuam em simultâneo. É de notar, que este gráfico varia de peça para peça, mesmo mantendo a mesma geometria, devido à alteração do desgaste da ferramenta, a alterações ambientais e a factores estranhos ao processo de maquinação.

Devido aos vários factores que influenciam o processo de maquinação, não é possível de imediato identificar este gráfico. Se tal fosse possível, parte da optimização poderia ser feita no programa de maquinação, principalmente a parte que se mantém constante de peça para peça.

O Gráfico 3.1, é apenas um exemplo da variação das forças de corte, função de alguns factores, dos quais depende. Este gráfico, não pretende ser uma representação precisa da realidade, mas sim, dar uma ideia da variação possível dos factores que influenciam a força de corte, num processo de maquinação.

Se observarmos o Gráfico 3.1, alínea *d*), podemos verificar que a função descrita é altamente não linear, e poderá ter formas completamente diferentes, conforme a geometria da peça, material e ferramenta utilizada na maquinação.

A função, descrita no parágrafo anterior, varia de peça para peça, e além disso, existe o facto de este processo ter uma componente empírica bastante elevada, não sendo possível identificar na totalidade, e analiticamente, qualquer das funções descritas nas várias alíneas do Gráfico 3.1.

O que acabamos de descrever significa que o conhecimento da planta concreta que pretendemos controlar é reduzido e empírico, não se podendo aplicar os métodos tradicionais de controlo.

3.3.3 Controlo não linear

A maior parte dos sistemas do mundo real apresentam, da perspectiva de controlo, um comportamento não linear e, também muitas vezes, dificilmente se conseguem modelizar, usando as leis da Física [5].

Os principais problemas ligados ao controlo, prendem-se com a necessidade de conhecimento preciso da planta a controlar e, com a sua natureza não linear.

A divisão entre sistemas lineares e não lineares, sugere uma classificação para os sistemas não lineares que não existe porque as equações diferenciais de um sistema não

linear são virtualmente desprovidas de um método geral de resolução, e uma solução exacta só poderá existir para um grupo particular que partilhe algumas propriedades [4].

Uma revisão detalhada, acerca de sistemas de controlo não lineares, é uma tarefa difícil, já que esta área é muito diversificada e está em continuo desenvolvimento, e não teria cabimento neste trabalho.

Os sistemas não lineares têm uma dinâmica muito rica, sendo a sua análise mais complexa. As equações diferenciais de sistemas não lineares, ao contrário das equações dos sistemas lineares, não podem genericamente serem resolvidas analiticamente.

Algumas diferenças entre sistemas lineares e não lineares são [4]

- as ferramentas clássicas, como as transformadas de Laplace e Fourier, não podem ser aplicadas aos sistemas não lineares
- o princípio da sobreposição não se aplica a sistemas não lineares, e portanto não existe forma possível de encontrar resposta a um sinal de entrada particular, se conhecermos a resposta a outro sinal
- os sistemas não lineares não são paramétricos, não existindo coeficientes *standards*, como os que existem para sistemas lineares, pelo que o problema da identificação é não trivial.
- os sistemas não lineares exibem propriedades locais, pelo que, enquanto um sistema linear é globalmente estável ou globalmente instável, por exemplo, um sistema não linear poderá ter regiões de estabilidade.

Resumindo, não existe um procedimento sistemático para o projecto de sistemas de controlo não lineares, e, em alternativa, existem várias técnicas de análise, cada uma aplicada a um conjunto particular de sistema de controlo não linear.

Para a resolução destes problemas, recorre-se presentemente a técnicas de controlo adaptativo, como sejam os controladores baseados em Lógica Fuzzy, e as Redes Neurais Artificiais.

Em linguagem corrente, podemos dizer que o controlo adaptativo significa mudar o comportamento de acordo com as novas circunstâncias. Intuitivamente, um controlador adaptativo é aquele que consegue modificar o seu comportamento, como resposta às mudanças de dinâmica do processo e ao carácter das perturbações [6].

As Redes Neurais Artificiais, têm demonstrado ser uma técnica bastante promissora para resolver este tipo de problema, através do uso de modelos construídos a partir dos dados medidos. As redes neurais artificiais, têm como princípio de funcionamento a adaptabilidade, função do conhecimento que vão adquirindo com a experiência em funcionamento.

Para se poder projectar um controlador, terá sempre que se ter um conhecimento da planta que se pretende controlar, conhecimento este que poderá já existir, ou que terá que ser transmitido ao controlador durante o processo de controlo.

Num sistema de controlo teremos sempre os blocos representados na Figura 3.2. Nesta figura podemos identificar o bloco controlador e o bloco a ser controlado, por nós normalmente designado por planta. Também podemos identificar quais os sinais em jogo.

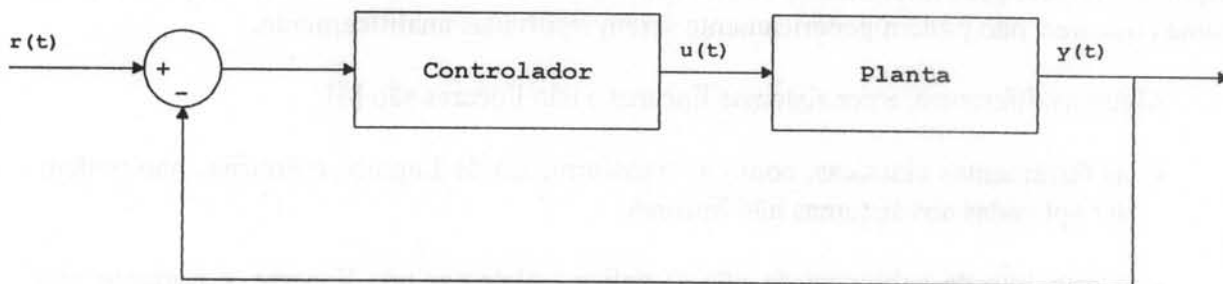


Figura 3.3 - Diagrama de um controlador genérico

Para se desenhar um controlador, necessitamos de saber qual o sinal $u(t)$ a aplicar à planta de modo a que $y(t) = r(t)$. Podemos no entanto dizer que genericamente a planta pode ser definida na seguinte equação (3.3)

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-n), u(t), u(t-1), \dots, u(t-m)) \quad (3.26)$$

A partir da equação (3.3) podemos obter a equação (3.4) que nos define o sinal de entrada u em função do sinal de saída y e de anteriores sinais de entrada u

$$u(t) = f^{-1}(y(t), y(t-1), \dots, y(t-n), u(t-1), u(t-2), \dots, u(t-m)) \quad (3.27)$$

Em sistemas não lineares a função f atrás definida não é conhecida e mesmo quando se pode definir uma função f que aproxime o processo essa função normalmente não tem função inversa.

Em muitos sistemas de controlo não linear, a função f é definida através de uma tabela de pontos que são recolhidos experimentalmente. No entanto, uma das características dos sistemas não lineares é a função f poder estar constantemente a modificar-se, impossibilitando assim um correcto tabelamento da função.

O sistema anterior está definido de uma forma simples, pois poderão surgir perturbações exteriores como exemplificado na Figura 3.4

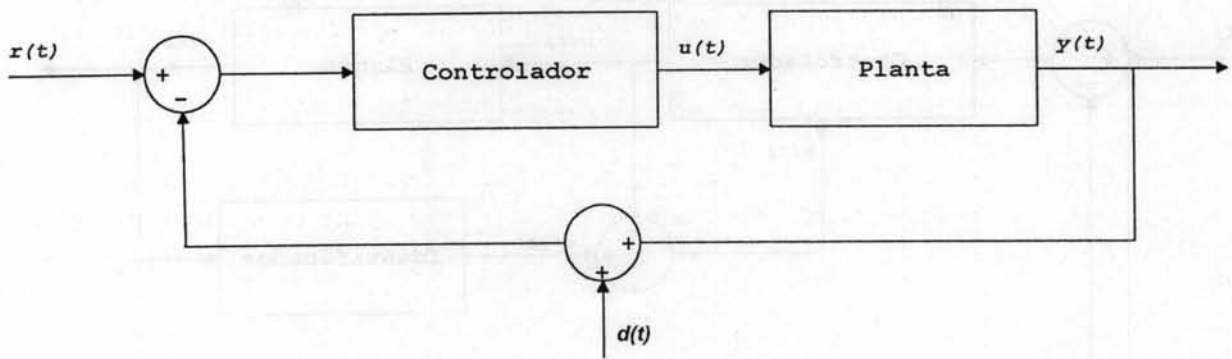


Figura 3.4 - Diagrama de blocos de um sistema de controlo onde se introduz perturbações na malha de realimentação

Para este sistema, as equações definidas atrás transformam-se nas seguintes (3.5), (3.6).

$$y(t) = f(y(t-1), y(t-2), \dots, y(t-n), u(t), u(t-1), \dots, u(t-m), d(t), \dots, d(t-k)) \quad (3.28)$$

$$u(t) = f^{-1}(y(t), y(t-1), \dots, y(t-n), u(t-1), u(t-2), \dots, u(t-m), d(t), \dots, d(t-k)) \quad (3.29)$$

Se tivermos um conjunto de amostras significativas recolhidas da planta experimentalmente, é possível, usando redes neurais, implementar um bloco que define a função f ou a função $g = f^{-1}$. Este bloco designam-se normalmente por Identificador.

Tendo as funções f e g definidas, é possível implementar um controlador, baseado em redes neurais, em que os vectores de treino da rede são obtidos a partir da referência e do resultado da função g , quando se aplica a esta a referência, pois pretendemos que $y(t) = r(t)$.

Num sistema não linear, se um sinal u , aplicado à entrada da planta, gerar um sinal y à sua saída, o mesmo sinal u , aplicado à entrada, poderá gerar um outro sinal y à saída, diferente, consoante o estado em que se encontra a planta nessa altura. Isto quer dizer que, o controlador terá que se adaptar a esta situação. Para isso o controlador necessita de saber, em cada instante, qual o estado da sua saída. Terá portanto de ser realimentado internamente.

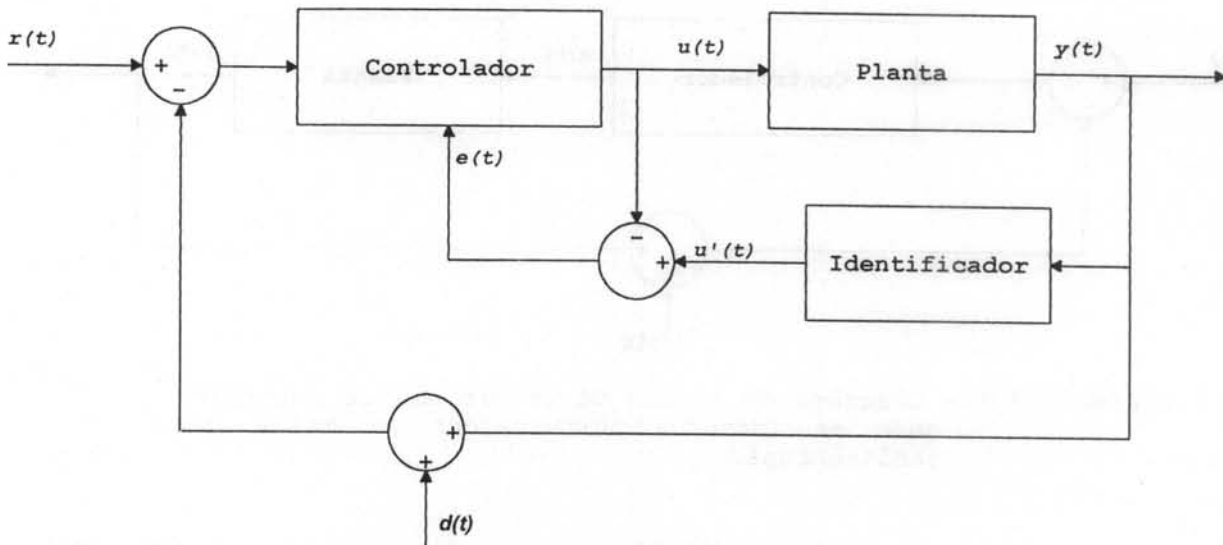


Figura 3.5 - Controlador com o bloco de identificação

A partir do momento em que se implementa um controlador, baseado em redes neuronais, é possível adaptar dinamicamente os pesos da rede neuronal do controlador de modo a satisfazer a condição anterior. Para isso, usamos o identificador do seguinte modo:

- para um sinal u apresentado pelo controlador á entrada da planta, é gerado um sinal y à sua saída
- aplicando o sinal y da saída da planta, à entrada do identificador, obtemos o sinal u' à saída deste
- o sinal u' assim obtido, quando comparado com o sinal u da saída do controlador, gera um sinal de erro que será usado para adaptar os parâmetros da rede neuronal do controlador.

Podemos observar na Figura 3.5 como seria constituído um controlador baseado nesta técnica.

Na Figura 3.5 temos apenas um diagrama representativo do que seria o sistema de controlo. Note-se no entanto que o bloco designado por identificador deve traduzir as equações referidas atrás, isto é, além do sinal $y(t)$ deve também ter como entrada o sinal $u(t)$ atrasado e se for caso disso, também o sinal $d(t)$ atrasado.

3.3.4 Variáveis de Controlo

Numa máquina de controlo numérico, existe já implementado um sistema de controlo de velocidade e posicionamento bastante complexo. A máquina controla com grande precisão a posição da peça, a velocidade de rotação da ferramenta e a velocidade de avanço.

O controlo do posicionamento tem implicações apenas no que diz respeito à geometria da peça. Este controlo define a precisão com que a máquina é capaz de executar determinadas operações, significando que a qualidade do controlo é definida à priori, definindo, de certa forma, as características e o custo da máquina.

O controlo da velocidade de avanço está, de certa forma, relacionado com o controlo de posicionamento, e assim sendo, o controlo da velocidade de avanço é preciso e intrínseco à máquina.

O controlo da velocidade de rotação, não será tão preciso como o controlo da velocidade de avanço, mas no entanto, nas máquinas modernas, este controlo recorre a sistemas electrónicos de controlo de velocidade de motores, já bastante desenvolvidos, o que significa que terá uma precisão bastante aceitável.

O controlo da velocidade de rotação é portanto, um controlo também intrínseco à máquina, não precisando de nos preocupar com tal.

Para se exercer o controlo da força de corte, como foi já dito atrás, é forçoso controlar a secção da apara do material.

Se nos reportarmos à equação (3.2), verificamos que podemos alterar a secção, alterando a velocidade de avanço, a velocidade de rotação e a profundidade de corte. Como a profundidade de corte é dependente da geometria da peça, tem a sua variação própria e independente dos nossos desejos, ficamos assim limitados à possibilidade de alterar a razão velocidade de avanço / velocidade de rotação.

A alteração da razão velocidade avanço / velocidade de rotação define a variação da secção da apara.

No entanto, o nosso controlador terá que controlar a velocidade de avanço e a velocidade de rotação individualmente, pois a máquina usa estes dois parâmetros separadamente.



FRESA 4 NAVALHAS HSCo

Vc (m/min)		24			32			18			25			80			100		
G.Mat. Desig.		1						2						6					
		AÇO CONSTRUÇÃO						AÇO C. TRATADO						LIGAS LEVES					
NÃO DIN		60/70 Kgf/mm²						80/100 Kgf/mm²						Al/Cu/CuZn					
		1.1191 / 1.1730						1.7225											
Diametro mm	FIG.	RPM		AVANÇO				RPM		AVANÇO				RPM		AVANÇO			
				mm/min	mm/Rot					mm/min	mm/Rot					mm/min	mm/Rot		
2	1	3822	61	0.012				2866	48	0.012				12739	255	0.016			
	2	5096	31	0.008				3981	23	0.008				15924	102	0.008			
4	1	1911	71	0.028				1433	56	0.028				6369	287	0.036			
	2	2548	46	0.024				1990	34	0.024				7962	178	0.028			
6	1	1274	109	0.064				955	85	0.064				4246	403	0.076			
	2	1699	66	0.052				1327	50	0.052				5308	255	0.060			
8	1	955	178	0.140				717	139	0.140				3185	621	0.156			
	2	1274	115	0.120				995	86	0.120				3981	446	0.140			
10	1	764	204	0.200				573	159	0.200				2548	764	0.240			
	2	1019	122	0.160				796	92	0.160				3185	510	0.200			
12	1	637	255	0.300				478	199	0.300				2123	955	0.360			
	2	849	153	0.240				663	115	0.240				2654	679	0.320			
16	1	478	255	0.400				358	199	0.400				1592	1035	0.520			
	2	637	162	0.340				498	122	0.340				1990	637	0.400			
20	1	382	265	0.520				287	207	0.520				1274	1019	0.640			
	2	510	168	0.440				398	126	0.440				1592	662	0.520			
25	1	637	467	0.550				478	365	0.550				2123	1725	0.650			
	2	849	287	0.450				663	215	0.450				2654	1168	0.550			
32	1	478	430	0.675				358	336	0.675				1592	1642	0.825			
	2	637	275	0.575				498	206	0.575				1990	1115	0.700			



Fig 1

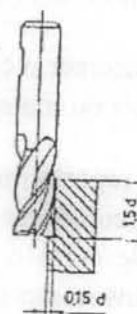


Fig 2

FACTORES
CORREÇÃO
AVANÇO

- 1.2 Para fresas curtas.
- 0.7 Para fresas médias.
- 0.5 Para fresas longas.

FACTORES
CORREÇÃO
Vc e AVANÇO

- 0.9 a 1.1 Dependendo da carga de rotura material.
- 0.9 Deficiente lixação.
- 0.85 a 1.1 Dependendo da qualidade da refrigeração.



Figura 3.6 - Velocidades de avanço e corte para materiais diferentes e espessuras de corte diferentes

Para uma determinada ferramenta e um determinado material, não existe uma expressão analítica que seja capaz de descrever a relação velocidade de avanço / velocidade de rotação.

A selecção de um determinada velocidade de avanço e uma velocidade de rotação, para um material e ferramenta, baseia-se em tabelas de valores obtidos experimentalmente e normalmente fornecidos pelos fabricantes de ferramentas. Um exemplo de uma dessas tabelas encontra-se na Figura 3.6.

Note-se que não existe uma relação capaz de definir a relação velocidade de avanço / velocidade de rotação, nem para o mesmo material e ferramenta, apenas e só função da profundidade de corte.

Na tabela, a alteração da força de corte para os diferentes casos não está explícita, mas podemos atribuí-la à variação da profundidade de corte representada nas Figura 1 e Figura 2 apresentadas ao lado da tabela.

Quando nos referimos a um controlador da força de corte para uma máquina, isto quer dizer que esta terá de actuar qualquer que seja a ferramenta, material ou qualquer outros distúrbio que possa ocorrer no momento da maquinação. As tabelas atrás referidas são função do material, da ferramenta e da profundidade de corte, pelo que qualquer outro factor que possa influenciar o processo terá de ser corrigido pela experiência do projectista ou do operador da máquina. Essa experiência poderá vir a fazer parte do controlador implementado.

Como indicação para o controlador é usada a força de corte que a máquina exerce em cada instante. A força de corte é obtida de modo indirecto, a partir da leitura da corrente eléctrica consumida pelo motor. A corrente eléctrica consumida pelo motor é proporcional ao binário desenvolvido, sendo o binário proporcional á força de corte.

Este método indirecto de medir a força de corte, poderá apresentar resultados errados, caso se entre na zona de saturação do motor. No entanto, isto não deverá acontecer, pois nesse instante a força exercida seria de tal modo elevada, que algo de errado estaria a acontecer.

A força assim medida é o somatório de todas as forças que sejam perpendiculares ao raio da ferramenta, nomeadamente

- forças de corte exercidas pelos vários dentes da ferramenta, que se encontram em contacto com o material
- forças devido ao desgaste da ferramenta
- forças de atrito e de inércia do sistema
- forças resultantes de qualquer perturbação exterior ao processo.

3.4 Arquitectura do controlador

Como foi já discutido atrás, o controlador foi dividido em dois níveis hierárquicos a saber,

- Decisor - nível hierárquico superior
- Actuador - nível hierárquico inferior.

O controlador que se pretende desenvolver, basear-se-á em Redes Neurais Artificiais, capazes de modelizar o funcionamento da máquina, evoluindo e adaptando-se através da aprendizagem do comportamento do processo.

3.4.1 Principais problemas de controlo

Temos vindo a descrever os principais problemas de controlo existentes, e vamos agora resumi-los

- a força de corte é uma função altamente não linear
- a função da força, de corte altera-se completamente de peça para peça, material para material e de ferramenta para ferramenta
- a planta é quase completamente desconhecida
- há duas variáveis de controlo independentes
- a selecção da força de corte que minimiza os custo de produção, maximiza a qualidade e o tempo de vida da ferramenta.

Um sistema de controlo capaz de resolver todos estes problemas terá que se basear em técnicas de controlo adaptativo, o que sugeriu o uso de Redes Neurais Artificiais, como técnica base para o desenvolvimento do controlador.

Também deve ficar claro que o controlador a implementar pretende apenas otimizar a solução. Não se espera que este seja capaz de resolver todos os problemas descritos atrás em termos absolutos.

3.4.2 O Actuador

O controlador, que vem a ser descrito, poderá ser representado pelo diagrama da Figura 3.7. Neste diagrama, podemos ver representado o Decisor, o Actuador e dois blocos de identificação.

Como se trata de um controlo adaptativo, o objectivo dos blocos de identificação é simular o processo e comparar os resultados com os valores reais. O erro resultante será usado para alterar os parâmetros do controlador, de modo a adaptá-lo á nova situação.

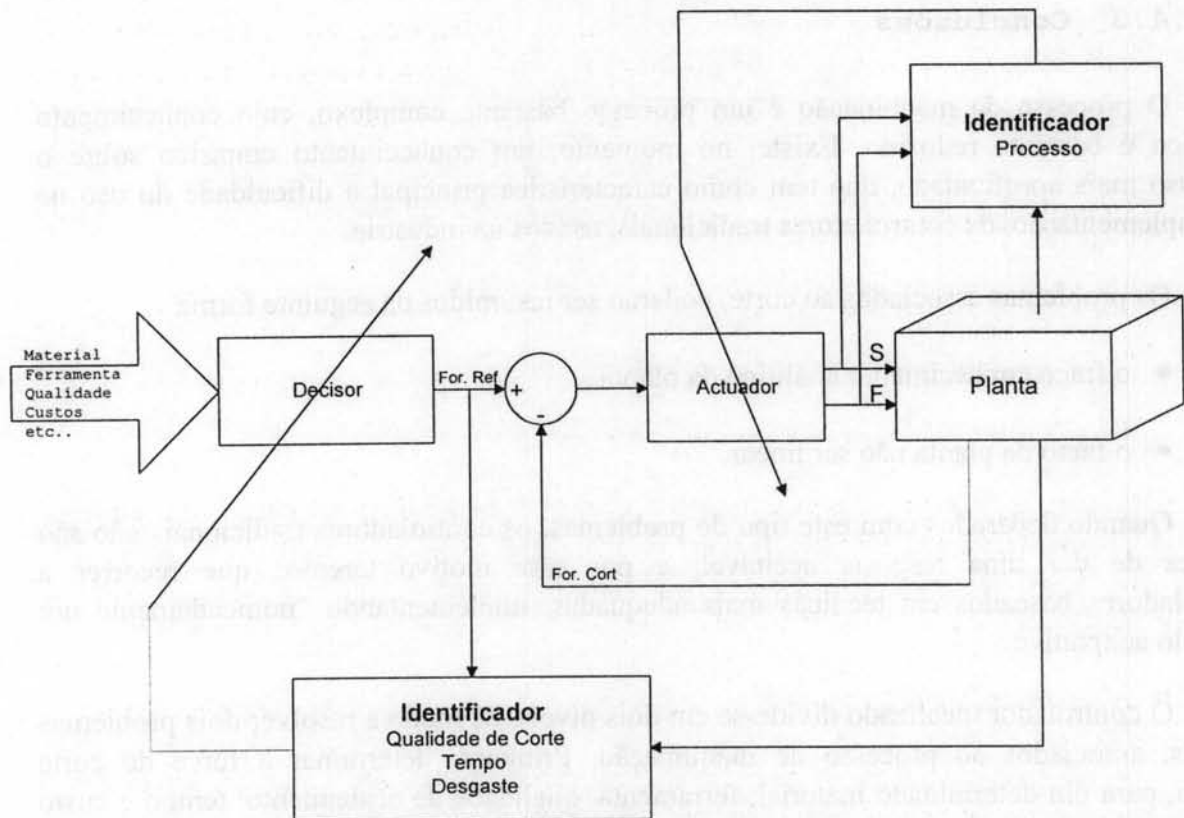


Figura 3.7 - Diagrama de blocos do Controlador

O Decisor será baseado numa rede neuronal do tipo *feed forward*. A esta rede será fornecido um conjunto de parâmetros, dos quais depende a maquinação e, como resultado, esta dará a indicação da força de referência a ser usada na maquinação em causa.

O Controlador será baseado numa rede neuronal do tipo recorrente. A esta rede será fornecido o erro resultante da comparação entre a força de referencia e a força de corte exercida pela máquina em cada instante. Esta rede debitará continuamente, como resultado, o valor da velocidade de rotação e velocidade de avanço a usar no processo de maquinação, em cada instante.

O identificador relacionado com o Decisor, recolherá um conjunto de parâmetros relacionados com a maquinação e o resultado final. A informação recolhida servirá para alterar os parâmetros do controlador de modo que este seja capaz de decidir, cada vez melhor, qual a força a ser usada para um determinado conjunto de parâmetros de entrada.

O identificador do Actuador têm como função gerar um erro, que permita ao Controlador saber se está a executar o controlo correctamente, adaptando os seu parâmetros de modo a minimizar o erro.

3.4.3 Conclusões

O processo de maquinação é um processo bastante complexo, cujo conhecimento analítico é bastante reduzido. Existe, no momento, um conhecimento empírico sobre o processo mais aprofundado, que tem como característica principal a dificuldade do uso na sua implementação, de controladores tradicionais, usados na indústria.

Os problemas associados ao corte, poderão ser resumidos da seguinte forma

- o fraco conhecimento analítico da planta
- o facto da planta não ser linear.

Quando deparados com este tipo de problemas, os controladores tradicionais não são capazes de dar uma resposta aceitável, e por esse motivo teremos que recorrer a controladores baseados em técnicas mais adequadas, implementando nomeadamente um controlo adaptativo.

O controlador idealizado divide-se em dois níveis, de modo a resolver dois problemas básicos, associados ao processo de maquinação. Primeiro, determinar a força de corte óptima, para um determinado material, ferramenta, qualidade de acabamento, tempo e custo de produção. Segundo, manter a força de corte óptima, constante ao longo de toda a operação.

Devido às suas características especiais, as redes neuronais servirão como base para o desenvolvimento dos referidos controladores, sendo o controlador de alto nível será implementado numa rede neuronal do tipo *feed forward*, e o controlador de baixo nível implementado numa rede do tipo recorrente.

4. Redes Neurais

4. Redes Neurais	77
4.1 Sumário	78
4.2 Redes Neurais	78
4.3 Evolução das Redes Neurais Artificiais	79
4.4 Biologicamente	81
4.4.1 Evolução do Conhecimento do Cérebro	81
4.4.1.1 O Neurónio	82
4.4.1.2 Estrutura do neurónio	82
4.4.1.3 Funcionamento do Neurónio	84
4.4.1.4 Modelo do Neurónio	84
4.5 Sistema Artificial	85
4.5.1 Algumas propriedades	86
4.5.1.1 Paralelismo	86
4.5.1.2 Capacidade de adaptação	86
4.5.1.3 Memória distribuída	87
4.5.1.4 Capacidade de generalização	87
4.5.1.5 Facilidade de desenvolvimento	87
4.5.2 Limitações das Redes Neurais	88
4.6 Redes Neurais. Teoria/Prática	88
4.6.1 Redes Neurais Artificiais - Fundamentos	88
4.6.1.1 Neurónios - unidades de processamento	89
4.6.1.2 Conexões entre neurónios	91
4.6.1.3 Regras e funções de activação da saída	92
4.6.2 Topologia das Redes Neurais Artificiais	93
4.6.3 Treino de Redes Neurais Artificiais	94
4.6.3.1 Paradigmas da aprendizagem	97
4.6.3.2 Regra para alteração dos pesos das conexões	97
4.7 Conclusões	98

4.1 Sumário

O presente capítulo pretende apenas dar uma ideia genérica do que são as redes neuronais, de onde provêm, como se desenvolveram e qual é normalmente o seu campo de aplicação.

É importante também, que se descreva o processo biológico das redes neuronais, de onde provem a ideia associada às Redes Neurais Artificiais.

Este capítulo inicia-se com uma breve introdução às redes neuronais, artificiais e biológicas, onde se comparará a eficiência do sistema biológico, com a dos computadores tradicionais.

A evolução do conhecimento, acerca das Redes Neurais Artificiais, é o tema seguinte.

O capítulo continua com a descrição do sistema artificial. São expostas as propriedades mais importantes relacionadas com as Redes Neurais Artificiais, vistas como sistemas de computação paralela. Também serão enumerados os limites de tais sistemas.

Finalizaremos, com a descrição de alguns aspectos teórico-práticos relacionados com as Redes Neurais Artificiais.

4.2 Redes Neurais

Todos os animais desempenham tarefas de reconhecimento de uma forma natural e muito eficiente. Se pensarmos apenas no sistema de visão, verificamos que quase todos os animais são capazes de reconhecer formas tridimensionais, movimentos, e alguns ainda as cores.

Embora os computadores superem as redes neuronais, biológicas ou artificiais, em tarefas baseadas nas operações de cálculo aritmético, estas representam uma nova e promissora geração de sistemas de processamento de informação [8].

O enorme poder das Redes Neurais Artificiais reside no tipo de aproximação que fazem aos problemas, muitas vezes intratáveis por recurso a técnicas de computação tradicionais, e na sua capacidade de generalização: As Redes Neurais Artificiais conseguem, tal como um sistema biológico, realizar uma tarefa de reconhecimento, mesmo em circunstâncias em que não possuem a totalidade dos dados.

Se compararmos o funcionamento lógico de uma máquina, um computador, com o do ser Humano, podemos verificar que o computador é muito rápido a fazer cálculos aritméticos (milhões de vezes mais rápido que o ser Humano). No entanto, o ser Humano, é capaz de reconhecer uma cara familiar em menos de 200 ms, enquanto que o computador levaria provavelmente algumas horas, dependendo das suas potencialidades. O caso seria ainda mais complicado, para o computador, se a imagem da cara se apresentasse, por exemplo, de um ângulo ligeiramente diferente do previamente armazenado.

Isto permite-nos concluir que o funcionamento biológico do cérebro humano consegue tratar grande quantidade de informação mais rapidamente que o computador.

As Redes Neurais Artificiais podem ser caracterizadas como Modelos Computacionais com a particularidade de terem a capacidade de adaptar-se ou aprender, generalizar, organizar dados, e cujas operações se baseiam no processamento paralelo.

Segundo o Dr. Robert Hoccht-Wielsen [6] uma rede neuronal será

“... um sistema computacional, constituído por um número de sinapses e elementos de processamento, altamente interconectados, que processam a informação segundo a dinâmica de resposta a entradas externas...”

Estes são sistemas neurais artificiais. Sistemas neurais artificiais ou redes neurais, são sistemas físicos celulares, que adquirem, armazenam e utilizam conhecimentos experimentais.

O objectivo último desta tese não é, note-se,, o conhecimento teórico profundo das redes neurais. Estamos mais interessados no funcionamento das redes neurais do ponto de vista prático, em aplicações reais, como a que estamos a estudar.

Em função dos problemas que se tentou resolver, este estudo foi sendo direccionado para áreas relativamente restritas e isoladas, nomeadamente para os algoritmos de treino das redes.

Não obstante, ou por isso mesmo, faz-se inicialmente uma breve resenha dos princípios biológicos e artificiais, e dos fundamentos teóricos e práticos, sobre que assentam as redes neurais.

4.3 Evolução das Redes Neurais Artificiais

O interesse pelas redes neurais surgiu com a introdução do neurónio simplificado (*perceptrão*) por McCulloch e Pitts em 1943 [7].

McCulloch e Pitts, estudaram este modelo e demonstraram, que tinha capacidade de cálculo de certas funções lógicas.

Hebb, num estudo publicado em 1949 [16], demonstrou a importância das conexões, entre sinapses no processo de conhecimento.

Rosenblatt, em 1959 [19], descreveu o primeiro modelo operacional, conjugando as ideias de Hebb, McCulloch e Pitts. O seu *perceptrão*, inspirado no estudo do sistema de visão, podia aprender a calcular algumas funções lógicas, modificando as conexões entre as suas próprias sinapses. Este modelo estimulou a investigação em redes de neurónios, provocando mesmo um optimismo exagerado.

Em 1969, dois matemáticos, Minsky e Papert, demonstraram teoricamente as limitações do *perceptrão* [17]. O efeito foi dramático, e os investigadores começaram a perder o interesse nas redes neuronais, tendo nomeadamente começado a estudar a possibilidade de usar uma linguagem simbólica em problemas relacionados com inteligência artificial.

Os anos oitenta mostraram contudo um renovado interesse pelas redes neuronais (também conhecidos como *connectionist model* ou *parallel distributed processing*).

Este interesse nas redes neuronais é largamente devido à contribuição de Hopfield, que demonstrou a analogia entre redes neuronais e certos sistemas físicos, num estudo em 1982 [18]. Mais recentemente e desde 1985, novos modelos matemáticos deram origem a um grande alargamento dos limites do *perceptrão*.

O que contribuiu, para que as redes neuronais tivessem a importância que têm hoje, foi a publicação de alguns resultados teóricos importantes, o mais notável dos quais foi a descoberta do algoritmo *back-propagation*. O desenvolvimento de *hardware* com novas capacidades de processamento também contribuiu decisivamente para aumentar as capacidades e importância das Redes Neuronais Artificiais.

As primeiras aplicações práticas começam a surgir, e a interessar um grande número de estudantes, investigadores e engenheiros industriais. Contudo, como resultado da multidisciplinaridade do assunto, é muito difícil sistematizar todo o conhecimento acerca das redes neuronais de uma forma coerente. Literalmente, milhares de *papers* foram escritos sobre este assunto em jornais que vão desde a biologia fisiológica à matemática, física e electrónica, cada um com uma aproximação ao problema, segundo a sua área de especialização.

Como resultado da intensa actividade de investigação que se desenvolve neste domínio, já existem no mercado algumas soluções para a implementação de redes neuronais, recorrendo a circuitos integrados dedicados à realização de redes massiças de interconexão em paralelo.

O esperado sucesso dos semicondutores, com uma tecnologia de fabrico já amadurecida, aliado à vulgarização dos computadores pessoais, com desempenhos inimagináveis há apenas alguns anos atrás, tem imposto, também aqui, um domínio de soluções mais tradicionais sobre o processamento paralelo.

No entanto, as potencialidades desta área de investigação, que se reconhecem de imediato no paralelismo inerente, na velocidade de processamento e na inexistência de interferências, conferem-lhe um lugar privilegiado no desenvolvimento das redes neurais, evidenciado pela miríade de trabalhos que têm sido publicados nos últimos anos.

Existem actualmente, implementadas na industria, soluções baseadas em Redes Neurais Artificiais. São soluções basicamente ligadas a problemas não resolúveis sequencialmente pelos algoritmos actuais, isto porque, ou não existem esses algoritmos, ou porque os algoritmos são de tal modo pesados, computacionalmente, que não produzem resultados em tempo útil.

As áreas, na industria, onde as Redes Neurais Artificiais têm tido maior aplicação são

- controlo de sistemas com certas particularidades desconhecidas
- reconhecimento de padrões.

No primeiro caso, o sucesso das redes neurais deve-se às dificuldades encontradas, quando não se conhece a planta a controlar e quando os sistemas são não lineares. No segundo caso, deve-se principalmente à velocidade de processamento, que se pode imprimir às Redes Neurais Artificiais, com o uso de *hardware* especializado.

Existem ainda outras aplicações, ligadas a sistemas de decisão, onde há necessidade de assimilar dados, a partir da experiência. Estas aplicações abrangem muitos tipos de problemas diferentes, não estando exclusivamente relacionadas com a a solução de problemas de índole industrial.

4.4 Biologicamente

4.4.1 Evolução do Conhecimento do Cérebro

Há cerca de 3000 anos, já existia a preocupação de conhecer a função do cérebro, embora alguns filósofos pensassem que a inteligência e o sentimento residiam no coração. O cérebro, ao ser irrigado pelo sangue, produziria o *espírito animal*, que animava o corpo [9].

Só no séc. XVIII, é que ficou claro que o cérebro era o centro de comando do corpo. A partir do séc. XIX, o estudo do cérebro encaminhou-se para três aspectos distintos: modo de operação, localização das funções e, mais recentemente, processos químicos envolvidos.

4.4.1.1 O Neurónio

As células nervosas, as quais chamamos neurónios, são os elementos fundamentais do sistema nervoso central. Este sistema é constituído por cerca de cinco biliões de neurónios. Os neurónios tem um conjunto de pontos comuns a outras células, na sua maneira geral de organização e no seus sistemas bioquímicos [9].

Os neurónios tem cinco funções básicas

- receber os sinais que provêm dos neurónios vizinhos
- integrar estes sinais
- emitir impulsos nervosos
- conduzir esses impulsos
- transmitir tais impulsos para os neurónios que detêm a capacidade de recebe-los.

4.4.1.2 Estrutura do neurónio

O neurónio é constituído por três partes: o corpo da célula, as dendrites, e o axon, ou axónio. Um exemplo de um neurónio pode ser visto na Figura 4.1.

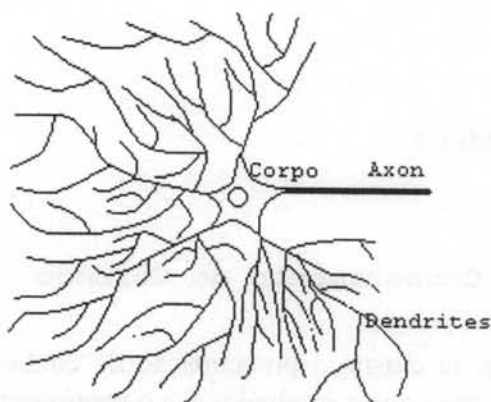


Figura 4.1 - Esquema do Neurónio Biológico

4.4.1.2.1 O corpo

O corpo da célula contém o núcleo, onde se processam as transformações bioquímicas necessárias para sintetizar enzimas e outras moléculas, necessárias à própria vida do neurónio. A sua forma depende da sua posição no cérebro, e a maioria, situada no neocortex, tem uma forma piramidal. O corpo da célula tem apenas alguns microns de diâmetro.

4.4.1.2.2 As dendrites

Cada neurónio tem uma estrutura de dendrites à sua volta, que faz lembrar uma cabeleira. As dendrites são extensões tubulares finas e cruzadas, com algumas dezenas de microns de comprimento, que ramificam em forma de uma árvore à volta do corpo da célula. São as dendrites as principais receptoras do neurónio e servem para fazer a ligação aos sinais exteriores.

4.4.1.2.3 O Axónio

O axónio, ou fibra nervosa, é a ligação pelo qual o neurónio transmite os seus sinais. Difere das dendrites na forma e nas propriedades da sua membrana externa. O axónio é mais comprido que as dendrites, podendo variar entre um milímetro e um metro de comprimento. A sua extremidade, pela qual comunica com outros neurónios, é ramificada.

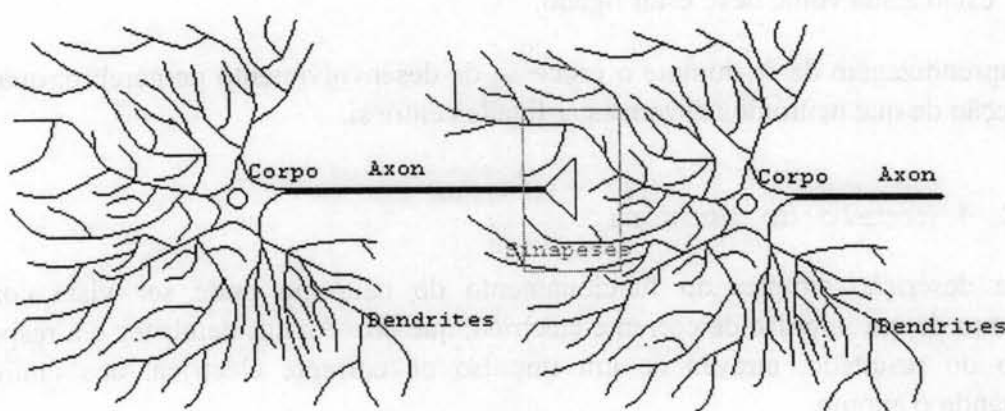


Figura 4.2 - Esquema de comunicação entre neurónio - sinapses

4.4.1.2.4 A Conexão

Os neurónios são conectados uns aos outros, de uma forma espacial e complexa, de modo a formar o sistema nervoso central. Na Figura 4.2 podemos observar a conexão entre dois neurónios, que tem o nome de sinapse.

4.4.1.3 Funcionamento do Neurónio

A função específica do neurónio depende das propriedades da sua membrana externa.

As cinco funções descritas atrás poderão ser processadas pelo neurónio do seguinte modo:

- os impulsos eléctricos são propagados ao longo do axónio e das dendrites
- nas extremidades são libertadas substâncias
- as dendrites reagem a estas substâncias e transmitem as reacções ao corpo da célula
- esta reage aos impulsos transmitidos pelas dendrites, e geram, ou não, um novo impulso eléctrico
- finalmente, o corpo do neurónio reconhece a qual dos neurónios, entre os que estão à sua volta, deve estar ligado.

A aprendizagem dá-se durante o processo de desenvolvimento de cérebro, quando é feita a selecção de que neurónios devem estar ligados entre si.

4.4.1.4 Modelo do Neurónio

Uma descrição simples do funcionamento do neurónio pode ser vista como o processamento de um impulso de corrente eléctrica, que provem das dendrites e a respectiva transmissão do resultado, através de um impulso de corrente eléctrica, aos neurónios, vizinhos usando o axónio.

A explicação biológica classicamente dada para esta operação, é que a célula produz uma saída, que é função da soma dos sinais que chegam através das dendrites. Se a soma, excede um determinado valor, a resposta do neurónio é um novo impulso, que se propaga ao longo do seu axónio; se a soma é inferior, o neurónio mantém-se inactivo. O impulso propagado entre neurónios é um fenómeno eléctrico.

Podemos observar o modelo de neurónio, na Figura 4.3.

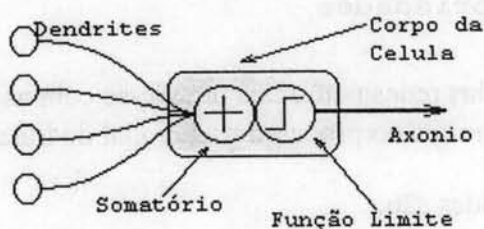


Figura 4.3 - Representação do modelo de um neurónio

4.5 Sistema Artificial

Se considerarmos o cérebro Humano uma máquina complexa de cálculo, podemos compara-lo, em algumas das suas características, com um computador vulgar.

Se compararmos a velocidade de cálculo, o computador é bastante mais rápido, mas se se comparar a capacidade de tratar grandes capacidades de informação (como seja uma imagem tridimensional), verificamos que o cérebro humano executa esta tarefa muito mais eficientemente, quer em velocidade quer em consumo energético.

Na realidade, o consumo de energia do ser Humano é de cerca de 10^{-16} joules por segundo por operação, enquanto que o valor equivalente, para os mecanismos artificiais mais desenvolvidos de hoje, é de cerca de 10^{-6} joules por segundo por operação.

Podemos comparar estes dois sistemas também quanto à precisão de cálculo. Neste domínio, os computadores são mais precisos que o ser Humano. Todos os cálculos produzidos pelo computador têm uma precisão que é maior ou menor, em função do algoritmo usado, sendo o cálculo baseado, sempre em sistemas que produzem um resultado preciso.

O ser humano, por seu lado, ao processar a informação faz sempre uma aproximação ao resultado, aproximação esta, que poderá ser maior ou menor, função apenas do seu conhecimento prévio e da sua capacidade de generalizar.

A extraordinária eficiência, sob todos os pontos de vista, dos sistemas biológicos, não deve no entanto constituir por si só, um objectivo a atingir pelos sistemas artificiais. Não só o conhecimento existente sobre os primeiros se encontra longe de atingir o estágio necessário para o efeito, como esse tipo de atitude se revelou com frequência inconsequente no passado, ao conduzir à frustração muitos de quantos se dedicaram ao desenvolvimento dos segundos.

As tendências mais recentes, nesta área de investigação, apontam no sentido de desenvolvimento de sistemas paralelos, baseados na filosofia de sistemas biológicos, que possam processar informação, com uma eficiência comparável à destes.

4.5.1 Algumas propriedades

O presente interesse pelas redes neuronais prende-se com as suas propriedades, e a sua capacidade de ultrapassar os limites do processo tradicional de tratar a informação.

As principais propriedades são:

- paralelismo
- capacidade de adaptação
- uso distribuído da memória
- capacidade de generalização
- fácil desenvolvimento.

Estas propriedades provêm da modelização directa do sistema nervoso Humano. A capacidade de generalização, que corresponde a um alto nível fisiológico de interpretação, relaciona-se directamente com o modo de operar do cérebro.

4.5.1.1 Paralelismo

O paralelismo é fundamental na arquitectura de uma rede neuronal, quando esta é considerada como um conjunto de unidades, que operam simultaneamente.

Este paralelismo no processamento de dados é interessante, devido às limitações do método sequencial no processamento de problemas que envolvem grandes quantidades de informação.

Através do uso de arquitecturas de *hardware* em paralelo, consegue-se um grande incremento de velocidade de calculo, mas isso exige que o problema seja resolvido de uma forma não convencional.

O estudo do modo de operação das redes neuronais, que são intrinsecamente sistemas paralelos, talvez dê origem a novas técnicas de especificar os problemas, permitindo que estes sejam processados em paralelo.

4.5.1.2 Capacidade de adaptação

Esta propriedade das redes neuronais, manifesta-se em primeiro pela capacidade de aprender, que permite às redes neuronais tomar em consideração os novos dados e restrições, que lhe chegam do mundo exterior.

Além disso, parece que algumas redes neuronais, têm a capacidade de se auto-organizarem, assegurando a sua estabilidade, como um sistema dinâmico.

Esta capacidade de adaptação, é particularmente interessante para problemas que envolvam situações que são desconhecidas à partida. Isto quer dizer que a rede é capaz de ter em atenção uma alteração no problema que tenta resolver, ou mesmo que será capaz de aprender a resolver o problema de uma forma diferente.

4.5.1.3 Memória distribuída

Numa rede neuronal, a memória corresponde a um mapa de activações de neurónios, mapa este que é, de certa forma, a codificação dos factos que estão armazenados. A memória assim distribuída, por várias unidades, torna-se uma propriedade valiosa, de elevada resistência ao ruído.

Em primeiro lugar, a perda de informação individual não causa necessariamente a perda de dados de um conjunto, ao contrário do que acontece num computador tradicional, em que a perda de um dado individual causa uma perda permanente, não sendo este capaz de recuperar o conjunto.

Numa rede neuronal, a destruição de uma unidade de memória, apenas muda marginalmente o mapa de activações dos neurónios, mantendo-se a sua funcionalidade base. Dai a imunidade ao ruído.

4.5.1.4 Capacidade de generalização

A capacidade de generalização de uma rede neuronal é demonstrada pela capacidade que a rede tem depois de aprender a partir de exemplos, de ser capaz de reconhecer um padrão que não faça parte desses exemplos.

Esta capacidade é crucial, e a sua importância foi recentemente demonstrada, pela dificuldade de obter regras para sistemas de inteligência artificial.

Muitos problemas são resolvidos por peritos, de uma maneira mais ou menos intuitiva. Isto torna bastante difícil definir uma base de conhecimento, a partir de regras. Além disso, é importante considerar um sistema que seja capaz de adquirir as suas próprias regras, simplesmente a partir de exemplos, o que permitirá que este apreenda um comportamento de imitação, fazendo com que o problema se auto-resolva.

4.5.1.5 Facilidade de desenvolvimento

A simulação computacional de uma rede neuronal, para um pequeno exemplo, é simples e requer apenas um curto espaço de tempo para desenvolvimento.

Para aplicações mais complexas, simuladores ou cartas *plug-in* aceleradoras poderão ser úteis. Alguns simuladores são neste momento extremamente fáceis de utilizar.

O desenvolvimento, usando Redes Neurais Artificiais, é bastante simples. Basta, em princípio, formular correctamente o problema e conseguir um conjunto de exemplos representativos.

4.5.2 Limitações das Redes Neurais

Apesar das características anteriormente descritas, as redes neurais apresentam certas limitações.

Uma destas limitações prende-se com a operacionalidade do sistema, já que quando se pretende simular o paralelismo das redes neurais, estas são normalmente simuladas em máquinas sequenciais. Deste modo, o tempo requerido para o processamento aumenta muito com o tamanho do problema. Isto é particularmente verdadeiro, quando se simulam redes neurais que usam máquinas de Boltzmann ou as redes neurais que usam *back-propagation*.

Presentemente, trabalha-se em técnicas de implementação de redes neurais, directamente em *hardware*. Apesar destas técnicas estarem numa fase de desenvolvimento ainda muito embrionária, espera-se que venham a permitir explorar verdadeiramente o paralelismo associado às redes neurais.

Não sendo propriamente uma limitação, o uso de redes neurais não é recomendado na implementação de algoritmos perfeitamente dominados. Nestes casos, não deveram ser usadas pois os resultados serão menos perfeitos.

4.6 Redes Neurais. Teoria/Prática

4.6.1 Redes Neurais Artificiais - Fundamentos

As redes neurais artificiais descritas neste capítulo, são todas variações de processos distribuídos paralelos. A arquitectura de cada uma das redes é baseada em blocos que representam os processos.

Neste capítulo, são descritas as unidades de processamento, seguindo-se uma referência às diferentes topologias e estratégias de aprendizagem.

Uma rede neuronal consiste num conjunto de unidades de processamento simples que comunicam enviando sinais, de umas para as outras, através de um sistema de pesos das conexões.

Podemos definir um conjunto de aspectos que podem ser distinguidos num sistema de computação paralelo e distribuído

- um conjunto de unidades de processamento (neurónios, células)
- um estado de activação para cada unidade (a_j), que determina a saída dessa mesma unidade
- conexões entre unidades (geralmente cada conexão é definida por um peso (W_{ij}) que determina o efeito que o sinal da unidade j tem sobre a unidade i).
- regra de propagação, que determina a saída efectiva da unidade i , a partir das suas entradas externas
- função de activação (F_j), que determina o nível de activação de saída, baseada na entrada $i_i(t)$ e no estado corrente $a_i(t)$.
- uma entrada externa, normalmente designando de *offset* (θ_j), para cada unidade
- ambiente no qual o sistema está inserido e deve operar (este ambiente, deverá fornecer os sinais de entrada, e se necessário os sinais de erro).

A Figura 4.4, representa os elementos aqui descritos.

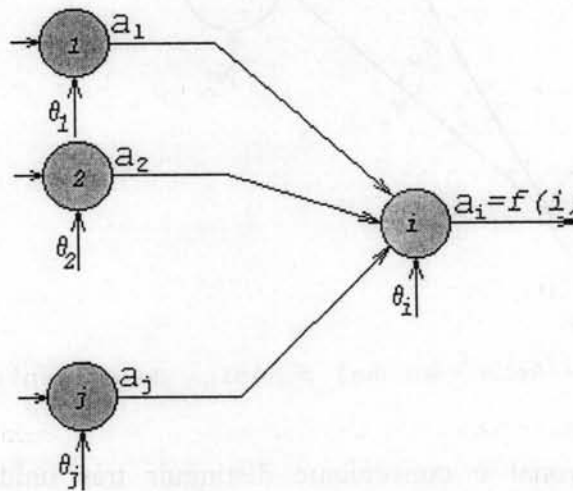


Figura 4.4 - Componentes básicos de uma rede neuronal (neurónios e respectivas ligações)

4.6.1.1 Neurónios - unidades de processamento

Cada neurónio executa uma tarefa simples, que consiste em receber as entradas provindas dos neurónios vizinhos, ou de fontes externas e usar estas para calcular um valor de saída, que será propagado para neurónios vizinhos

$$i = \sum_{k=1}^j w_{ik} a_k + \theta_i \quad a_i = f(i) \quad (4.30)$$

O sistema é, no seu processo básico, de processamento paralelo, isto porque vários neurónios podem produzir a sua saída, em simultâneo.

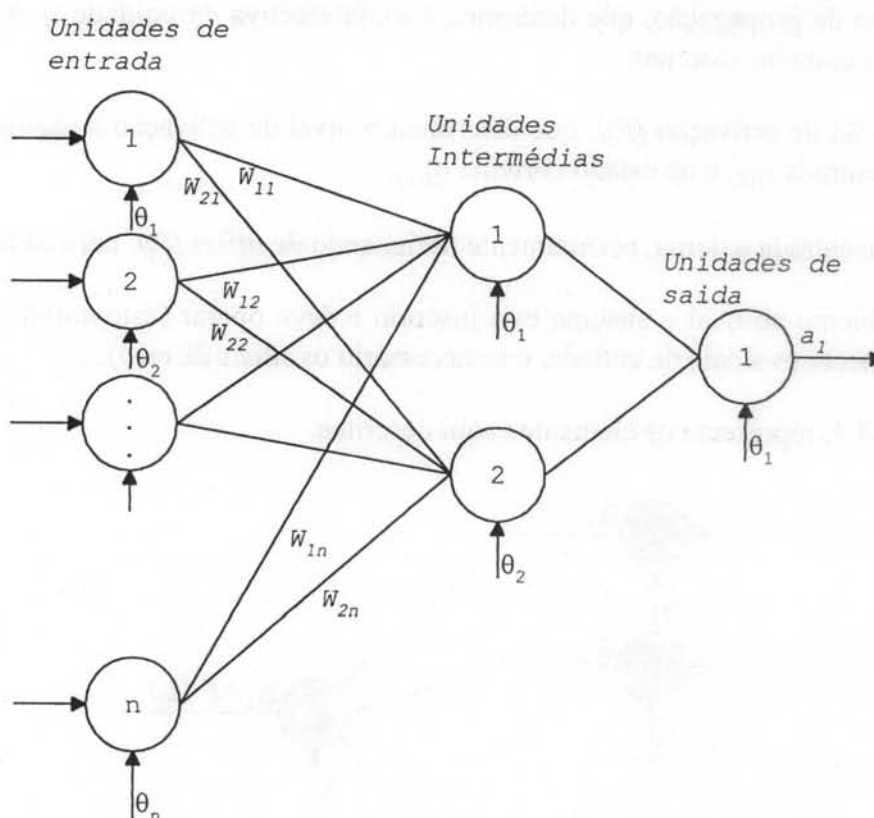


Figura 4.5 - Rede neuronal genérica de dois níveis

Numa rede neuronal é conveniente distinguir três unidades de processamento (neurónios), exemplificadas na Figura 4.5.

- unidades de entrada, cujo objectivo é receber os dados provindos do exterior do sistema
- unidades de saída, cujo objectivo é produzir os dados de saída (as acções ou decisões tomadas pela rede)
- unidades intermédias, também conhecidas por unidades escondidas (*hidden units*), cujas entradas e saídas, se encontram no interior do sistema. Estas unidades têm como objectivo, aumentar os graus de liberdade do sistema.

Numa Rede Neuronal Artificial, as unidades são dispostas segundo níveis, definidos a partir do número de grupos de unidades (na Figura 4.5, segundo a vertical). Para se definir o número de níveis existentes numa Rede Neuronal Artificial, conta-se o número de níveis intermédios mais o nível de saída.

Numa rede neuronal genérica, as unidades envolvidas, durante o funcionamento da rede, podem apresentar os seus resultados, sincronamente ou assincronamente. Em alguns casos, o sistema assíncrono tem algumas vantagens, principalmente, se pensarmos nos casos em que os dados de entrada não variam todos em simultâneo, bastando portanto, a alteração de um dado de entrada, para alterar a saída.

4.6.1.2 Conecções entre neurónios

As conexões entre unidades definem os pesos a aplicar a cada saída do neurónio, influenciando o neurónio seguinte ao qual esta conectado.

Cada neurónio, pode ainda ter uma entrada independente, que permite criar um deslocamento base, em relação ao zero, da função de transferência de cada neurónio. Esta entrada não garante que se aumentem as capacidades do neurónio, mas também não as diminui.

O valor total de entrada do neurónio i , é simplesmente a soma pesada, de todas as saídas dos outros neurónios, que estão ligados ao neurónio i , mais a respectiva entrada independente (*bios*)

$$i_i(t) = \sum_{k=1}^j w_{ik}(t) a_k(t) + \theta_i(t) \quad (4.31)$$

Uma unidade cuja regra de propagação, seja a apresentada atrás, designa-se de *sigma units*

Outras regras de propagação têm sido definidas, nomeadamente a regra introduzida por Feldman e Ballard [11], conhecida como *sigma-pi unit*, e cuja equação é

$$i_i(t) = \sum_j w_{ij}(t) \prod_k a_{jk}(t) + \theta_i(t) \quad (4.32)$$

4.6.1.3 Regras e funções de activação da saída

É necessário também definir uma regra que permita calcular a saída do neurónio em função dos seus valores de entrada, isto é, uma função que use o total de entrada $i_i(t)$ e calcule a respectiva saída $a_i(t+1)$. Neste calculo, poderá usar-se ou não $a_i(t)$, e as equações são

$$a_i(t+1) = f_i(\sum_j w_{ij} a_j(t) + \theta_i(t)) = f_i(i_i(t)) \quad (4.33)$$

$$a_i(t+1) = f_i(a_i(t), i_i(t)) \quad (4.34)$$

Esta função poder-se-á definir de muitas formas diferentes

- funções que apenas permitem gerar saídas do tipo, sim ou não (0, 1), normalmente designada de *hard limit*;
- funções lineares ou semi-lineares, cuja saída é uma função linear de entrada ou é uma função linear de entrada com valores máximo e mínimo definidos,
- funções não lineares normalmente designadas de *sigmoid* e cuja equação é a seguinte:

$$a_i = f(i_i) = \frac{1}{1 + e^{-i_i}} \quad (4.35)$$

Em algumas aplicações, usa-se a função tangente hiperbólica, ou outras funções com andamento análogo. Na figura 5 estão representadas as funções mais usadas.

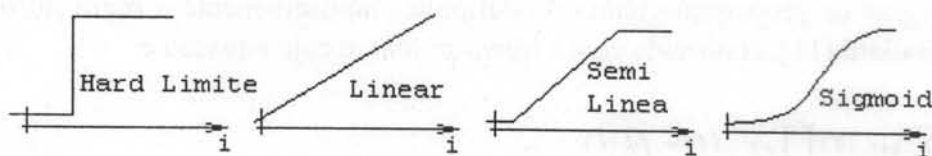


Figura 4.6 - Funções normalmente usadas como activação nos neurónios

Em alguns casos, o valor de saída da unidade, poderá ser uma função estocástica, e neste caso não temos uma activação da saída, determinista, mas uma activação que depende de um valor de probabilidade p

$$p(a_i \leftarrow 1) = \frac{1}{1 + e^{-i_i/T}} \quad (4.36)$$

em que T é o coeficiente de temperatura, parâmetro que determina a derivada da função de probabilidade.

Em alguns casos, poderá ser interessante definir uma função nova, se isso nos for útil. Teremos no entanto que verificar se tal função poderá ser usada nos algoritmos de treino que temos à disposição.

4.6.2 Topologia das Redes Neurais Artificiais

As Redes Neurais Artificiais são implementadas basicamente em uma de duas topologias

- *Feed-Forward Networks*, em que o sentido de circulação dos dados é estritamente da entrada para a saída, não existindo qualquer *feed-back*, podendo os dados passar por várias unidades em níveis intermédios
- *Recurrent Neural Networks*, redes neurais recorrentes, uma topologia de redes que contêm *feed-back*.

Ao contrario das *Feed-Forward Networks*, nestas a dinâmica de rede é importante. Em alguns casos os valores de activação entram num processo de relaxamento, de tal modo que atingem um estado estável, em que as activações de saída não se alteram no futuro.

A Figura 4.7, representa alguns exemplos de topologias de Redes Neurais Artificiais. Na alínea *a)*, temos a topologia vulgar, de uma rede neuronal artificial do tipo *Feed-Forward* (dois níveis). Nas, alíneas *b1)*, *b2)* e *b3)*, temos vários tipos de redes neurais recorrentes, em que na alínea *b1)*, temos o que podemos designar, verdadeiramente, de rede neuronal recorrente, em que todos os neurónios estão interligados entre si. Na alínea *b2)*, por sua vez, temos o modelo de uma rede neuronal recorrente, usada por Hopfield [12], em que a saída está ligada à entrada. Embora esta rede esteja representada com dois níveis, a estrutura da rede usada por Hopfield, era de um só nível. Finalmente, em *b3)*, temos o modelo usado por Elman [13], em que a realimentação é feita, não na saída da rede, mas sim na saída dos neurónios do segundo nível, e em que temos, portanto, os neurónios do segundo nível ligados à entrada.

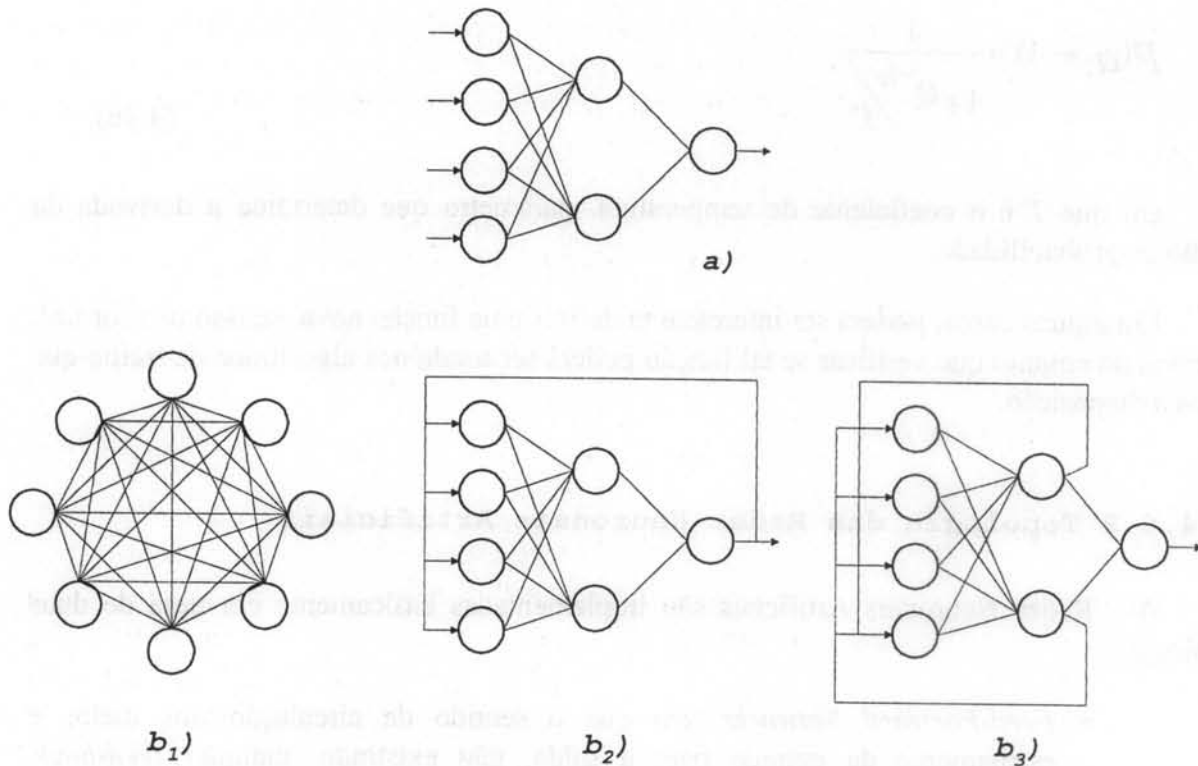


Figura 4.7 - Topologias de Redes Neurais Artificiais,
a) "Feed-forward"; b_{1,2,3}) "Recurrent"

As redes neurais designadas de *Elman neural networks*, são definidas através de dois níveis, com realimentação da saída do nível intermédio para a entrada. O algoritmo de treino usado é o algoritmo *Back-propagation*. Estas redes são normalmente usadas para o reconhecimento de padrões temporais e espaciais.

As redes neurais normalmente designadas de *Hopfield neural networks*, apresentam uma estrutura definida apenas por um nível, em que a saída, é realimentada para a entrada. O objectivo destas redes, é o armazenamento de vectores estáveis, aos quais a rede recorre quando solicitada com entradas idênticas. Esta rede comporta-se basicamente como uma memória.

4.6.3 Treino de Redes Neurais Artificiais

A rede neuronal terá de ser configurada, isto é, definida a sua estrutura e os pesos das conexões, de modo a que a aplicação de um vector de dados de entrada, produza as saídas desejadas.

O problema de definição da estrutura da rede neuronal, não é trivial. Vários autores têm-se preocupado com este assunto, sem que tenham chegado a definições conclusivas [14].

Normalmente, este problema é resolvido recorrendo à prática, simulando algumas arquiteturas diferentes. Pode-se começar por definir uma estrutura tipo, em que o número de neurónios no nível intermédio é igual duas vezes o número de neurónios de entrada vezes o número de neurónios de saída.

A selecção do número de neurónios do nível intermédio é bastante importante. Embora, não se possa afirmar que exista uma regra prática absoluta para determinar o número de neurónios neste nível, teremos que ter em atenção o seguinte

- um número reduzido de neurónios no nível intermédio, fará com que a rede tenha dificuldades em convergir, podendo o erro não atingir o valor esperado, mesmo ao fim de um período bastante longo de treino. Isto dará origem ao que designaremos de, incapacidade de se adaptar à função que tenta aproximar (*underfitting*), Gráfico 4.2.
- um número muito elevado de neurónios no nível intermédio, leva a que a rede apresente, um elevado grau de liberdade. Isto significa que a rede convergirá rapidamente, mas, no entanto, poder-se-á dar o que se designa de indefinição dos pontos intermédios (não pertencentes ao conjunto de treino) da função que se tenta aproximar (*overfitting*) Gráfico 4.1.

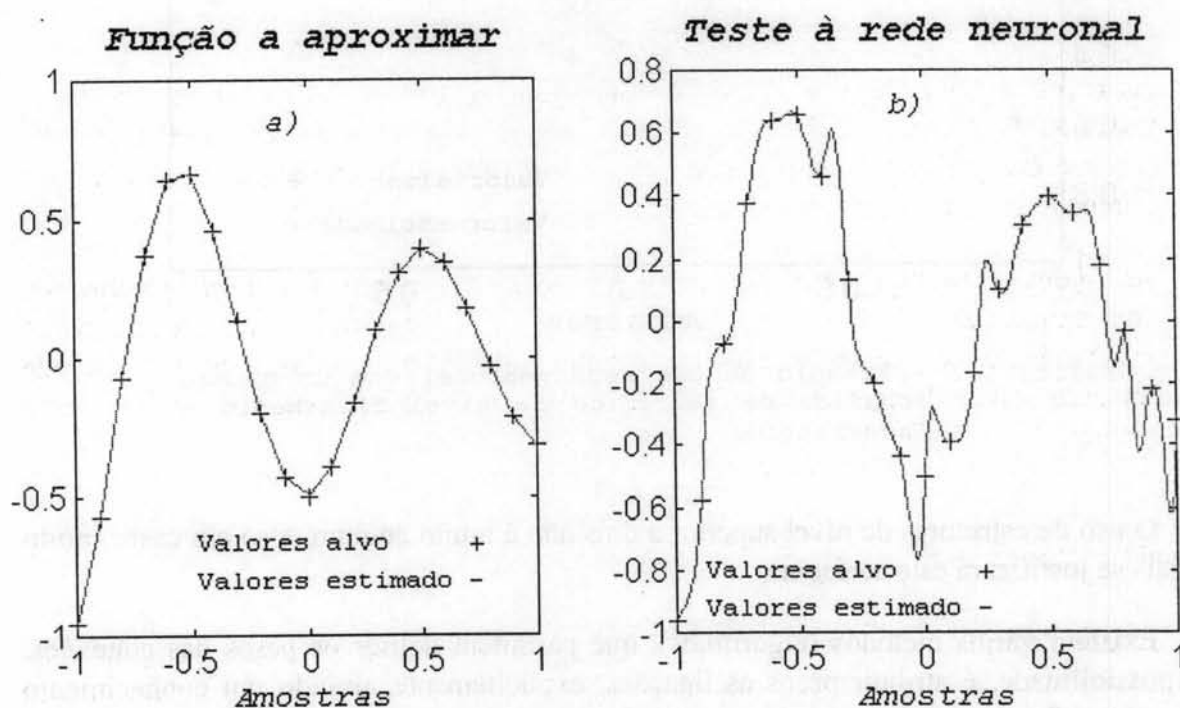


Gráfico 4.1- Exemplo de uma rede que apresenta um número elevado de neurónios no nível intermédio. a) Resposta da rede para os valores que serviram de treino, b) resposta da rede para um conjunto de amostras maior. - Overfitting

Se observarmos o Gráfico 4.1, podemos verificar que neste caso - *overfitting* - a rede é capaz de responder a todos os valores apresentados no treino correctamente (alínea a) do gráfico). No entanto quando se apresentam valores intermédios, em relação aos valores apresentados no treino, a rede tenta generalizar, apresentando valores errados e fora da curva que se pretendia (alínea b) do gráfico).

O Gráfico 4.2, é representativo do caso inverso, em que a rede não consegue responder correctamente aos valores para a qual foi treinada, isto é, a rede não consegue alterar os seus pesos de modo a poder aproximar a função que foi proposta.

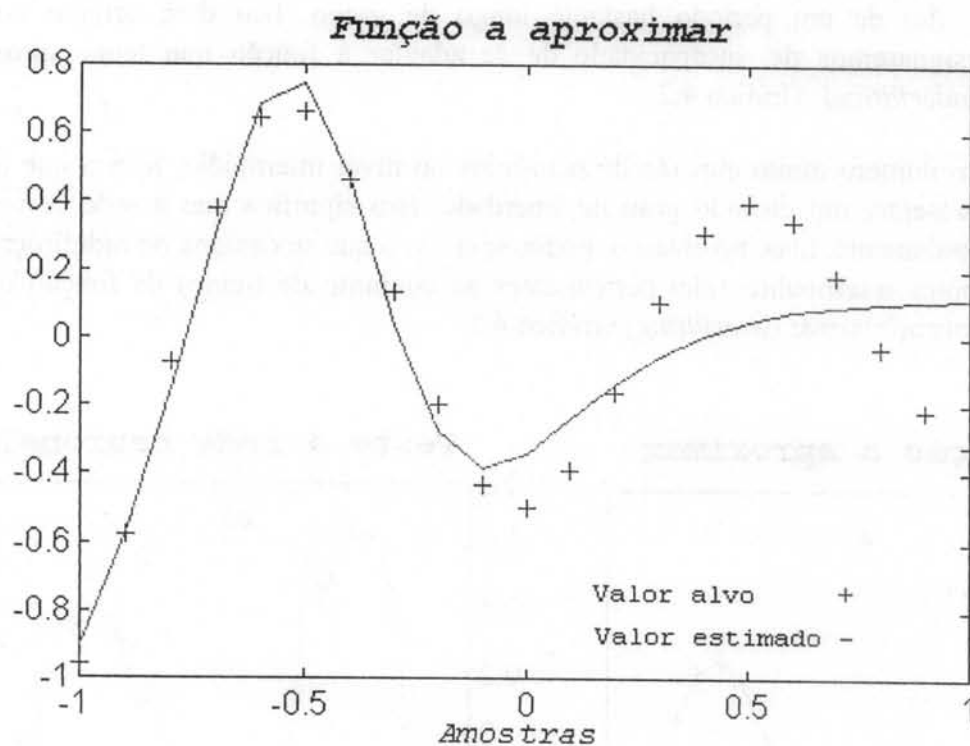


Gráfico 4.2 - Exemplo de uma rede neuronal com um número reduzido de neurónios no nível intermédio - *Underfitting*

O uso de estruturas de nível superior a dois não é muito comum, e só em casos muito especiais se justificará este acréscimo.

Existem vários métodos (algoritmos), que permitem definir os pesos das conexões. Uma possibilidade, é atribuir pesos às ligações, explicitamente, usando um conhecimento prévio. Outra forma, é treinar a rede, usando uma regra de aprendizagem, de forma que esta assimile um determinado comportamento, a partir de exemplos.

4.6.3.1 Paradigmas da aprendizagem

Para se treinar uma Rede Neuronal Artificial, usando uma das regras de aprendizagem, é necessário fornecer um conjunto de exemplos, com os vectores de entradas e respectivos vectores de saída.

Podemos dividir a aprendizagem em duas situações distintas

- Aprendizagem supervisionada, em que a rede é treinada, servindo-se de vectores de entrada e respectivas saídas. Os pares de valores de entrada - saída são fornecidos por um *professor* ou por um sistema capaz de fornecer o valor de saída para cada entrada.
- Aprendizagem não supervisionada, em que a unidade de saída da rede é treinada para responder a uma classe de padrões que serviram de entrada. Neste tipo de aprendizagem o sistema é suposto descobrir estatisticamente aspectos importantes na população de entrada. Ao contrário da aprendizagem supervisionada, neste caso não existe à priori um conjunto de classes no qual um determinado padrão deve ser classificado. Por seu lado o sistema deve desenvolver a sua própria classe de representação dos estímulos de entrada. Resumindo, não necessita do *professor*.

4.6.3.2 Regra para alteração dos pesos das conexões

Ambos os paradigmas discutidos acima resultam num ajustamento dos pesos das conexões entre unidades, de acordo com uma regra de modificação. Virtualmente, todas as regras de aprendizagem, para estes tipo de modelos, podem ser consideradas como uma variante, da regra de aprendizagem designada de Hebbian, sugerida por Hebb no seu livro *Organization of Behaviour* (1949).

A ideia básica é, que se duas unidades, i e j , estão activas em simultâneo, a sua conexão deve ser reforçada. Se i tem a sua entrada ligada à saída de j , a regra de Hebbian simplesmente diz que se deve modificar o peso da conexão (w_{ij}) segundo a seguinte equação

$$\Delta w_{ij} = \gamma a_i a_j \quad (4.37)$$

em que γ , é uma constante de proporcionalidade e positiva, que representa a taxa de aprendizagem. Uma outra regra, também comumente usada, ao contrario da anterior, é usada a diferença entre a actual activação da unidade e o valor desejado para esta é

$$\Delta w_{ij} = \gamma (d_i - a_i) a_j \quad (4.38)$$

em que d_i , representa o valor desejado para a activação, sendo este valor fornecido pelo *professor*. Esta regra é também muitas vezes designada de regra de “*Widrow-Hoff*” ou regra *delta*. Muitas variantes destas regras têm sido publicadas nos últimos anos.

Estas são as regras mais vulgarmente usadas no treino das Redes Neurais Artificiais.

4.7 Conclusões

Este capítulo é fundamentalmente uma descrição das redes neuronais, abrangendo, desde um pouco de história, até aos fundamentos básicos das Redes Neurais Artificiais, passando pela analogia biológica e por algumas comparações interessante do ponto de vista da eficiência.

Poderemos no entanto resumir as redes neuronais no seguinte

- neurónio, unidade base do sistema, biológico e artificial
- sistema de memória e processamento distribuídos
- paralelismo inerente ao processo
- capacidade de assimilar determinadas características, baseando-se em regras de aprendizagem e exemplos das entradas e saídas
- capacidade importante de generalização para novas situações.

As características descritas levam-nos a crer tratar-se de um sistema capaz de se adaptar aos sistemas de controlo não linear, de plantas desconhecidas.

A grande vantagem das Redes Neurais Artificiais, é a sua capacidade de assimilar comportamentos, a partir de exemplos desse mesmo comportamento.

5. Implementação

5. Implementação	99
5.1 Sumário	100
5.2 Descrição	100
5.2.1 O Bloco de nível superior - Decisor	101
5.2.1.1 O Controlador do bloco Decisor	102
5.2.1.2 O Identificador do Bloco Decisor	105
5.2.2 O Bloco de nível inferior - Actuador	106
5.2.2.1 O Adaptador	108
5.2.2.2 O Identificador	114
5.2.2.3 O Controlador	121
5.2.2.4 O Sensor	125
5.2.2.5 Testes	126
5.3 Conclusões	129

5.1 Sumário

Neste capítulo, tentaremos descrever os problemas que encontrámos na implementação de um controlador da força de corte, baseado em redes neuronais.

Tal como já foi descrito no Capítulo 3, o Controlador foi dividido em dois grandes blocos, um destinado a otimizar a força de corte, e o outro destinado a manter esta força de corte constante.

Definiremos agora como será constituído cada um destes blocos, quais os seus componentes básicos e as suas funções.

Genericamente, num controlo adaptativo os componentes básicos serão:

- o Identificador, que tem como função a modelização do sistema que se pretende controlar
- o Controlador propriamente dito, com a capacidade de se adaptar ao controlo do processo.

Tentaremos mostrar também qual a importância do Identificador num bloco de controlo como o que se pretende desenvolver.

A tecnologia usada para se desenvolver o Controlador e o Identificador, será baseada em Redes Neuronais Artificiais.

5.2 Descrição

A implementação de qualquer sistema baseado em Redes Neuronais Artificiais terá sempre duas etapas. A primeira consiste em seleccionar um conjunto de exemplos representativos do sistema, de tal modo que permitam á rede neuronal assimilar o seu comportamento. A segunda é definir a estrutura da rede, treiná-la e implementá-la.

Definir a estrutura da Rede Neuronal Artificial, significa, definir:

- o número de níveis da rede
- o número de neurónios por nível
- as funções de activação por nível.

Em alguns casos, será necessário definir também o numero de entradas regressivas, entradas que representam ou um sinal atrasado no tempo, ou iterações da rede.

5.2.1 O Bloco de nível superior - Decisor

Na implementação deste bloco, resolveu-se impor algumas restrições iniciais ao sistema e só no futuro proceder à sua generalização, conforme será justificado ao longo do capítulo.

Por questão de simplificação, decidiu-se também utilizar apenas um material e uma ferramenta no desenvolvimento base do bloco de nível superior.

O bloco de nível superior, como se pode ver na Figura 5.1, baseia-se em dois sub-blocos, o Controlador e o Identificador.

Este bloco foi designado de Decisor, porque a sua principal função, é decidir qual a força óptima de corte. No entanto, não deixa de ter características de controlador, na medida em que se variar algumas das variáveis de que depende, poderá ajustar a força de corte de acordo com essa variação.

Uma vez que as variáveis das quais depende normalmente só são actualizadas, ou no inicio ou no fim da maquinação, o Decisor tem um funcionamento principalmente estático, na medida em que decide no inicio da operação qual a força de corte a usar no processo e espera pelo seu fim para verificar qual o erro cometido.

A excepção será no caso em que exista realimentação em tempo real de variáveis como a temperatura, vibrações da máquina, etc. Neste caso, o Decisor comportar-se-à como um controlador de nível superior.

O controlador usado pelo Decisor, baseia-se numa Rede Neuronal Artificial do tipo *feed-forward*. Este controlador deverá ter como entradas

- material da peça - definido através da sua tensão de rotura
- tipo de acabamento - definido através de uma escala de 0 a 10
- material da ferramenta - definido pelos de tipos de material
- diâmetro da ferramenta
- numero de dentes da ferramenta
- tempo máximo de operação.

A saída da rede será a força de corte óptima, para ser usada nesta operação.



O bloco Decisor é ainda composto por um Identificador do sistema, que permite alterar os pesos da rede neuronal do controlador. Este Identificador tem como função observar a qualidade final, o tempo de operação e outros factores de maquinação que sejam influenciados pela força de corte, como por exemplo a temperatura, vibrações etc., e gerar a força de corte que lhes deu origem.

A força de corte gerada pelo sistema de identificação, quando comparada com a força de corte gerada pelo controlador, dá origem a um erro, erro este que pode ser propagado da saída para a entrada, na rede do controlador, de modo a ajustar os pesos associados a esta rede. A propagação do erro neste sentido é conseguida usando um algoritmo já conhecido, designado de *back-propagation*. Esta operação servirá para adaptar a Rede Neuronal Artificial do Controlador, de modo a que esta possa contemplar mais esta informação.

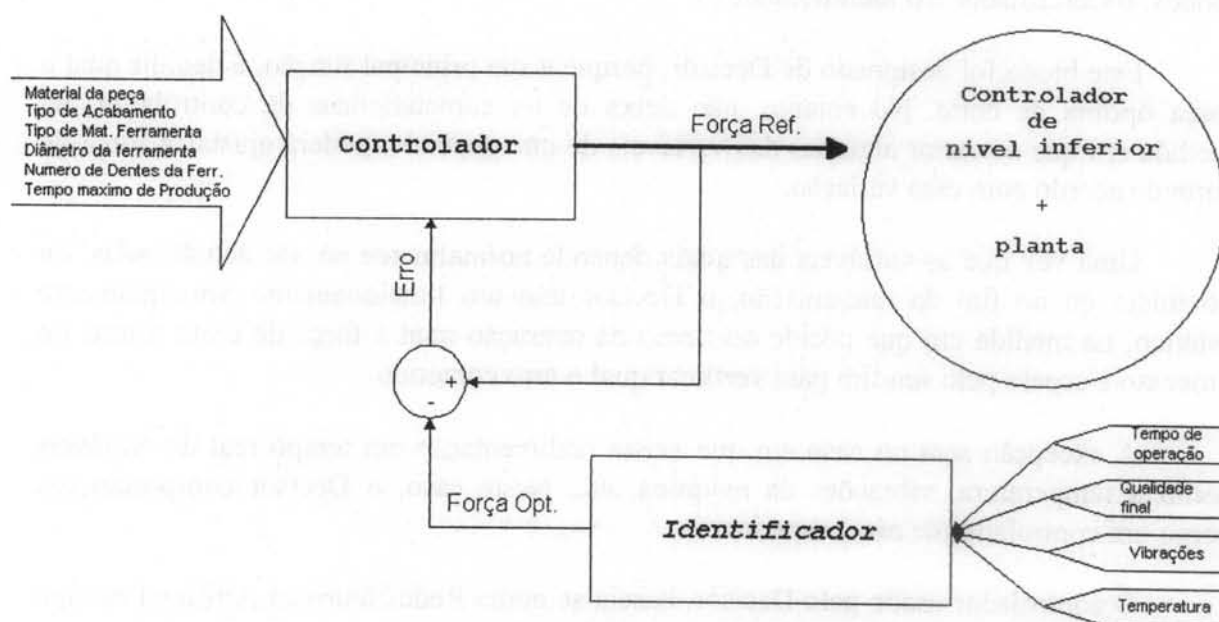


Figura 5.1 - Diagrama do Bloco Superior

A Figura 5.1, representa o diagrama do Decisor, podendo-se observar como se interligam os diferentes componentes que o implementam. A força de referencia (Força Ref.) representada na Figura 5.1, será usada pelo Actuador, como sinal de referencia. O Identificador receberá os seus sinais da planta que se pretende controlar.

5.2.1.1 O Controlador do bloco Decisor

O Controlador implementado neste bloco exhibe algumas dificuldades, associadas ao tipo de informação que vai utilizar. A informação a ser tratada, neste nível, é de dois tipos, qualitativa (qualidade - melhor, pior, etc.) e quantitativa (tempo de operação, tensão de rotura do material, etc.).

Como é fácil de compreender, esta informação é difícil de recolher e de quantificar, de modo a poder ser usada por um sistema computacional.

Apesar das Redes Neurais Artificiais serem capazes de tratar este tipo de informação, é sempre necessário quantificá-la de modo a que possa ser aceite por um sistema electrónico ou de cálculo.

Em consequência do que acabamos de referir, é fácil concluir que a informação qualitativa necessária à implementação do controlador, não pode ser recolhida totalmente e quantificada. No entanto, as propriedades de adaptação associadas às Redes Neurais Artificiais, permitiram-nos criar uma rede e transmitir-lhe apenas parte do conhecimento.

É necessário esclarecer que qualquer sistema de controlo, que tenha como propriedade principal a adaptação, necessita de ter algum conhecimento prévio do sistema em causa, antes de ser posto em funcionamento ligado ao sistema real.

No nosso caso, devido a falta de informação qualitativa, preparamos o Controlador, usando apenas informação quantitativa, de modo a ser aplicado ao sistema real, com algum conhecimento prévio. O restante conhecimento será transmitido à rede do controlador, durante o seu funcionamento.

Os dados usados para implementar o Controlador, foram os dados que foi possível de se recolher em de tabelas ou calcular numericamente

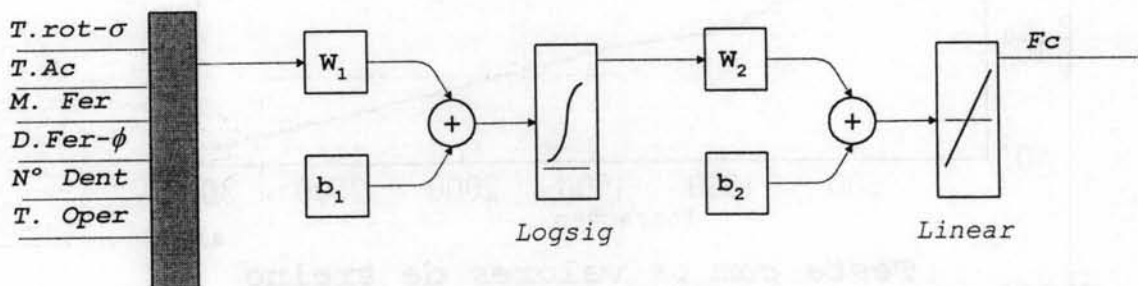


Figura 5.2 - Rede neuronal representativa do controlador associado ao sistema Decisor

Às entradas, referentes ao dados qualitativos, foram atribuídos valores que permitem, no futuro, a ligação destas entradas aos sinais correctos. Os valores a atribuir a estas entradas será zero, pois este valor na rede neuronal significa que estas entradas não influenciam a saída. Quaisquer que sejam dos pesos das ligações destas entradas aos restantes neurónios a o resultado será sempre zero.

A Rede Neuronal Artificial, representativa do Controlador do bloco Decisor, encontra-se na Figura 5.2.

Se considerarmos que pretendemos adaptar a rede para um material e uma ferramenta, poderemos colocar as entradas T. Ac; T. Op; M. Fer em zero, treinando a rede para um determinado material e ferramenta.

Ficaremos portanto com uma rede neuronal com a seguinte estrutura:

- seis entradas, algumas ligadas a zero
- um nível intermédio com 60 neurónios e com a função *logsig* como função de activação
- Uma saída (um neurónio) cujo função de disparo é linear (*purelin*).

A Rede Neuronal Artificial foi simulada e treinada no programa MATLAB usando o algoritmo de treino *back-progation*.

O treino durou cerca de 3000 iterações, atingindo o erro (quadrado do erro) o valor de 0.019.

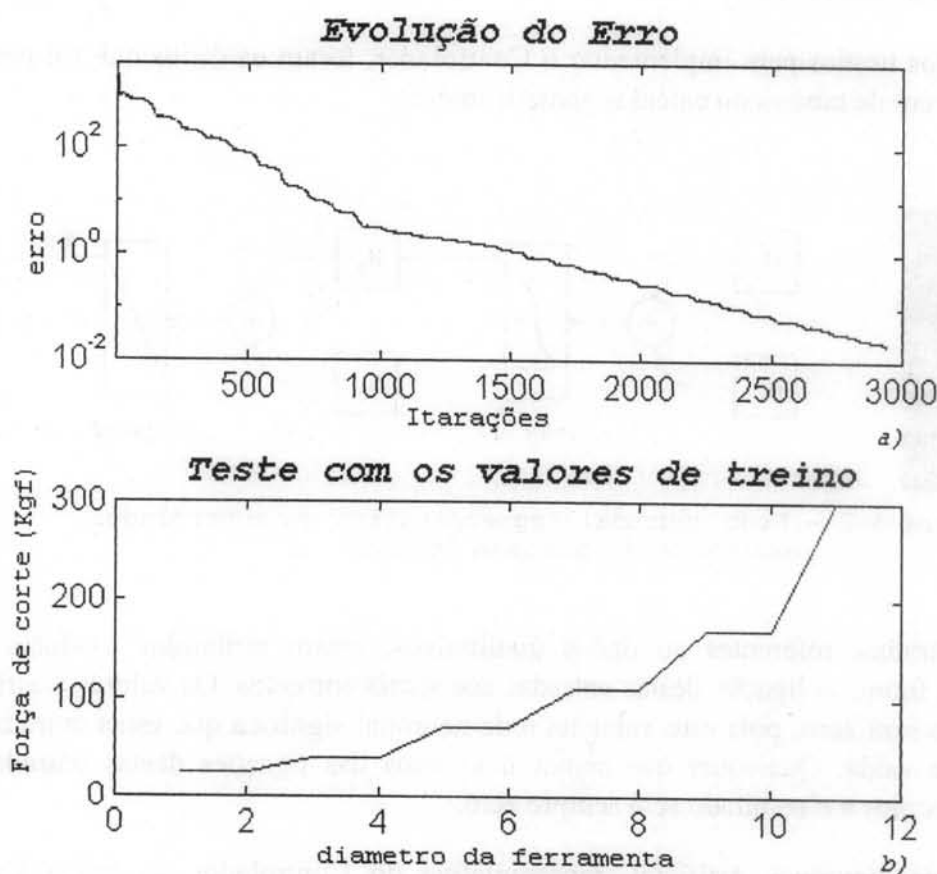


Gráfico 5.1 - Resultado do treino da Rede Neuronal Artificial que implementa o Controlador

O Gráfico 5.1, alínea *a*), mostra como evoluiu o treino da Rede Neuronal Artificial associada ao Controlador. Na alínea *b*) temos a comparação entre o valor estimado pela Rede Neuronal Artificial (a traço interrompido) e o valores usados para o treino (traço contínuo).

Nesta caso, é quase impossível distinguir, o valor estimado do valor real, o que se deve principalmente ao facto da Rede Neuronal Artificial ser sobre-dimensionada, de modo a que no futuro posso englobar as variáveis que de momento foram deixadas em zero. Tal facto permitiu que a rede convergisse para um erro bastante reduzido.

5.2.1.2 O Identificador do Bloco Decisor

O Identificador, tal como o Controlador, também recorre a informação qualitativa, o que origina algumas dificuldades na sua implementação.

Na realidade não foi possível obter dados suficientes, das experiências realizadas, que permitam definir a estrutura para uma Rede Neuronal Artificial que implemente o Identificador.

A qualidade final, o tempo de operação, as vibrações e temperatura, são dados apenas passíveis de serem recolhidos do sistema em funcionamento. O que significa que a Rede Neuronal Artificial que implementa o Identificador terá de ser treinada com valores resultantes do funcionamento do sistema.

O facto deste Identificador pertencer ao bloco do Decisor, permite-nos colocar o sistema em funcionamento, substituindo a resposta do Identificador pela resposta do operador da máquina. Este terá de vigiar, no início da operação, quais os valores gerados pelo Controlador, de modo a que se possa identificar o erro que servirá para adaptar o Controlador e em simultâneo o Identificador *on-line*.

Propomos, no entanto, uma estrutura base para a Rede Neuronal Artificial que implementará o Identificador. Essa mesma estrutura pode ser vista na Figura 5.3.

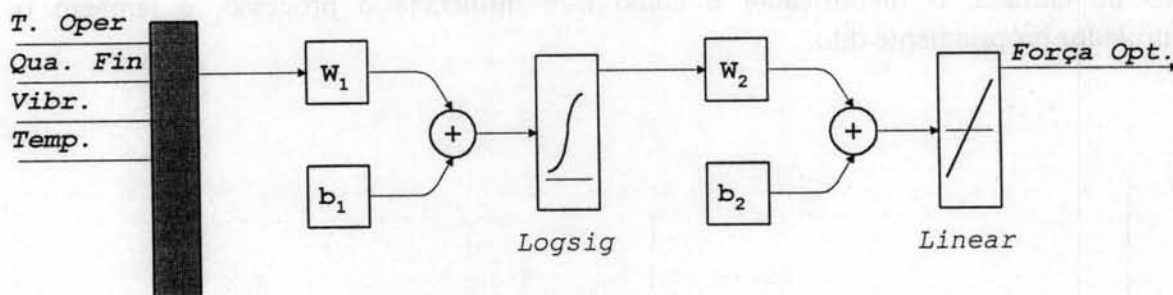


Figura 5.3 - uma possível estrutura para a Rede Neuronal Artificial que poderá implementar o Identificador

Sugere-se que no nível intermédio esta rede tenha um número elevado de neurónios, na ordem dos 200. Esta preocupação deve-se ao facto de a informação a ser tratada por esta rede ser variada e pouco relacionável.

Será importante, durante o funcionamento real, verificar se novas variáveis, não consideradas de momento, poderão definir melhor quanto se aproxima a força de corte seleccionada pelo Controlador da força de corte óptima. Também se poderá verificar se algumas das variáveis usadas têm pouca influência no processo, o que permitiria eventualmente simplificar a rede por eliminação de tais variáveis.

5.2.2 O Bloco de nível inferior - Actuador

Este bloco foi designado de Actuador, pelo simples facto de se tratar do bloco que actua directamente sobre a máquina. O Actuador na realidade é um sistema de controlo que engloba o Controlador propriamente dito, mais o Identificador.

O Actuador, é constituído pelo controlador, pelo Identificador e por um bloco intermédio entre o Controlador e a planta. Este bloco intermédio, que pode ser designado de Adaptador, tem como função, transformar o sinal recebido do controlador, em dois outros sinais, usados pela máquina. O controlador gera o sinal F/S , o Adaptador deve desdobrar este sinal nos sinais F e S .

O Adaptador é necessário porque para a mesma relação F/S , deverão ser gerados sinais F e S que poderão ser diferentes, em função dos materiais e da ferramenta utilizados. Quando a razão F/S muda, mesmo sem que se altere o material, o aumento ou diminuição do F e do S , poderá não ser linear, tão pouco proporcional. O valor F/S também pode ser designado de *mrr - material remove rate*.

O Adaptador, para poder gerar os sinais F e S , necessita assim da informação sobre a ferramenta e o material que se pretende maquinar.

A Figura 5.4, representa o Actuador, que deverá manter a força de corte da planta constante. Na figura, podemos identificar a planta a controlar, o Adaptador e os respectivos sinais de entrada, o Identificador e como este influencia o processo, e também o Controlador propriamente dito.

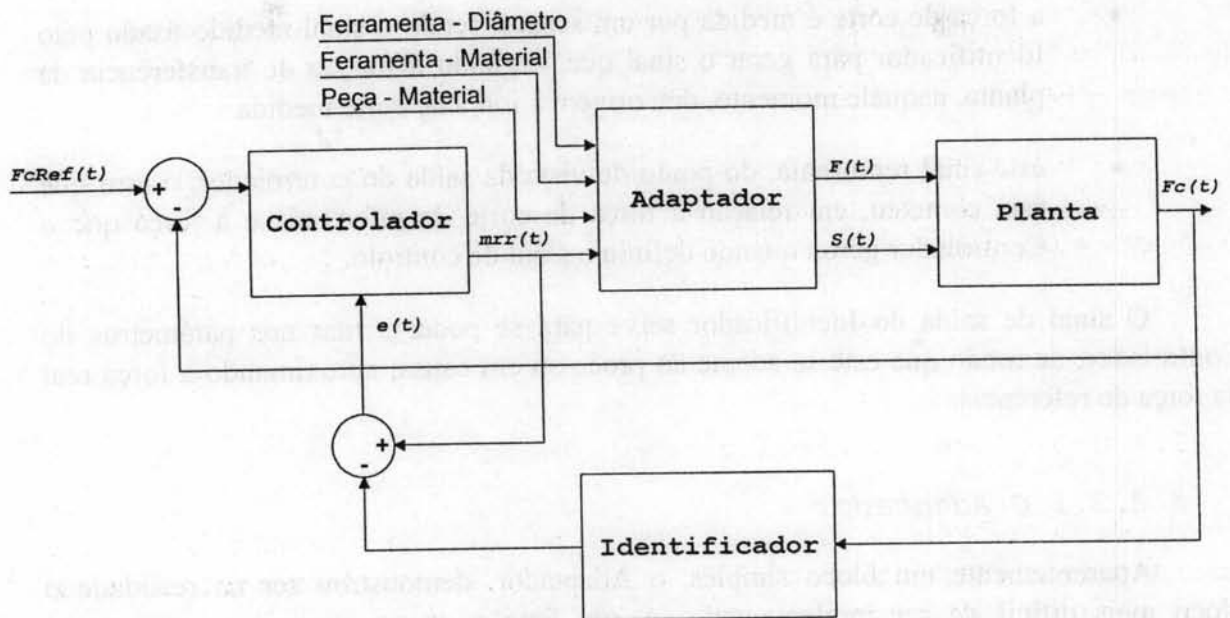


Figura 5.4 - Diagrama do bloco do nível inferior

A ideia ilustrada na Figura 5.4, é a seguinte

- o Identificador modeliza a planta, sendo capaz de gerar o sinal de entrada da planta, sabendo qual é a saída
- o controlador gera um sinal, a partir da força de corte de referência, designado de $mrr(t)$, que não é mais que a razão (F/S), dos sinais S e F a serem aplicados à planta
- o Adaptador, conhecedor das características do processo (material, ferramenta, etc.) e do respectivo sinal de entrada ($mrr(t)$), gera os dois sinais F e S , a aplicar à planta.

O princípio de funcionamento do Actuador pode ser descrito da seguinte forma

- o sinal de referencia ($FcRef$), é aplicado ao controlador
- o Controlador gera um sinal $mrr(t)$, de tal modo que a força de corte exercida pela máquina, seja igual à força de referência
- o Adaptador recebe o sinal $mrr(t)$, e a partir do conhecimento que tem do processo, gera os sinais F e S , a serem usados pela máquina
- o sinal aplicado à máquina, provoca uma determinada força de corte, que poderá ser igual, ou não, à força de corte de referência

- a força de corte é medida por um sensor, sendo o sinal medido usado pelo Identificador para gerar o sinal que, segundo a função de transferência da planta, naquele momento, deu origem à força de corte medida
- este sinal representa, do ponto de vista da saída do controlador, o erro que este cometeu, em relação à força de corte de referência e a força que o Controlador gerou quando definiu o sinal de controlo.

O sinal de saída do Identificador serve para se poder actuar nos parâmetros do Controlador, de modo que este se adapte ao processo em causa, aproximando a força real da força de referência.

5.2.2.1 O Adaptador

Aparentemente um bloco simples, o Adaptador, demonstrou ser na realidade o bloco mais difícil de ser implementado. A sua função, como atrás já se afirmou, é determinar o valor de $S(t)$ e $F(t)$, conhecendo a sua razão, e informações sobre os material a maquinar, e sobre a ferramenta.

Aparentemente, esta seria uma função bastante simples de implementar, mas, no entanto, verificou-se ao longo do seu desenvolvimento, que a situação era bem diferente. Na realidade, não podemos afirmar que a informação usada, e da qual depende os valores de $S(t)$ e $F(t)$, pode ser correlacionada.

Para implementar o Adaptador, tivemos que recorrer a ferramentas de cálculo capazes de correlacionar a informação. Uma das ferramentas capazes de executar tal tarefa são as Redes Neurais Artificiais.

Redes Neurais Artificiais com diferentes estruturas, foram treinadas, na tentativa de se criar um Adaptador com as seguintes características de entrada:

- material a maquinar (codificado pela sua tensão de rotura)
- diâmetro do ferramenta
- material da ferramenta (codificado 1- HSCo e 2- HSCo TiCN);
- número de dentes da ferramenta
- tipo de ferramenta (codificado 1 - Fresa normal; 2 - Fresa 55G Hélice e 3 - Fresa ripa)
- $mrr(t)$ ou $F(t)/S(t)$

e com as seguintes saídas

- $S(t)$ - velocidade de rotação

- $F(t)$ - velocidade de avanço.

No entanto, e depois de várias tentativas de treino de um conjunto de diferentes estruturas para a Rede Neuronal Artificial, verificou-se que esta não convergia, isto é, tinha dificuldades em aproximar (correlacionar) os valores dos exemplos usados no treino.

Para se tentar resolver este problema, criaram-se finalmente redes diferentes para estimar a velocidade de rotação (S) e a velocidade de avanço (F).

Basicamente, as Redes Neurais Artificiais ficaram com o número de neurónios do nível intermédio independente, o que permite ter um número de neurónios neste nível diferente para uma e outra rede.

Por outro lado, também os neurónios de saída são independentes, o que permite que a Rede Neuronal Artificial seja capaz de convergir mais rapidamente, aproximando os valores dos exemplos de treino.

Mesmo assim, a rede neuronal não convergiu em tempo útil, de modo que foram feitas várias tentativas de modificar o conjunto de exemplos de treino.

Modificou-se em primeiro lugar a forma como as características do material eram apresentadas à rede, tendo os valores sofrido uma modificação de escala. Esta alteração não foi suficiente para se conseguir que a rede convergisse, porque o conjunto de todos estes elementos tornou a correlação entre eles uma tarefa difícil. Seria necessário recolher um conjunto mais vasto de exemplos, onde fosse possível encontrar valores comuns que permitissem correlacionar todos os elementos.

A solução finalmente encontrada obrigou a reduzir o número de entradas da rede neuronal. Ficamos assim com uma rede neuronal para estimar a velocidade de avanço, com as seguintes características:

- duas entradas ($mrr(t)$ e tensão de rotura do material a maquinar - σ) (ver Figura 5.5)
- Um nível intermédio de 100 neurónios e função de activação *tansig*
- Uma saída ($F(t)$), cuja função de activação é linear (*purelin*).

A rede é do tipo *feed-forward*, treinada com o algoritmo *back-propagation*

A estrutura definida para esta Rede Neuronal Artificial, Figura 5.5, convergiu rapidamente no início e muito lentamente na parte final do treino. O erro atingido ao fim de 5000 iterações foi de 0.085.

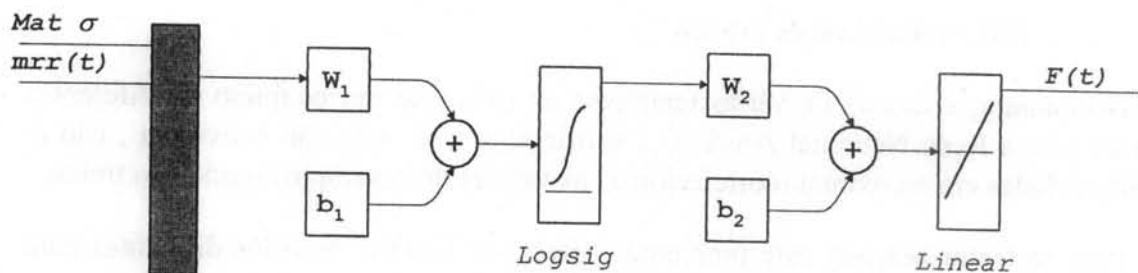


Figura 5.5 - Estrutura da Rede Neuronal Artificial que implementa a parte do Adaptador que estima a velocidade de avanço

O Gráfico 5.2, permite-nos analisar, o que se passou durante o treino. No gráfico menor, inserido no interior, poderá verificar como a curva do erro evolui ao longo das 5000 iterações. O gráfico maior, compara os valores estimados pela Rede Neuronal Artificial (traço interrompido) com os valores reais usados no treino (linha continua).

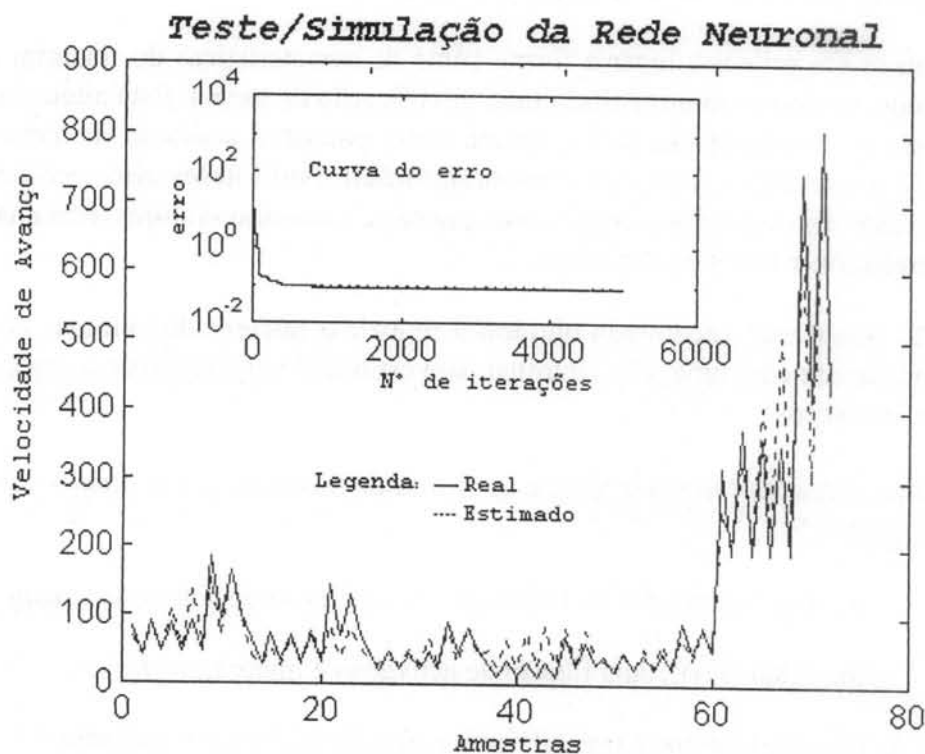


Gráfico 5.2- Teste realizado a Rede Neuronal Artificial que implementa a parte do Adaptador que estima a velocidade de avanço

A comparação entre os valores reais e os estimados pela Rede Neuronal Artificial, representados no Gráfico 5.2, é feita de uma forma mais clara no Gráfico 5.3. Como podemos observar, para valores elevados de velocidade de avanço o erro é maior. O erro foi calculado como sendo a diferença entre o valor real e o valor estimado pela rede, para os valores do conjunto de exemplos usados no treino.

O facto de termos um erro negativo, significa que o valor estimado ultrapassa o valor real.

No Gráfico 5.3, está marcado, através de uma linha a traço interrompido o valor de $\pm 10\%$ e podemos verificar que o erro, na sua maior parte da gama, se encontra abaixo deste valor.

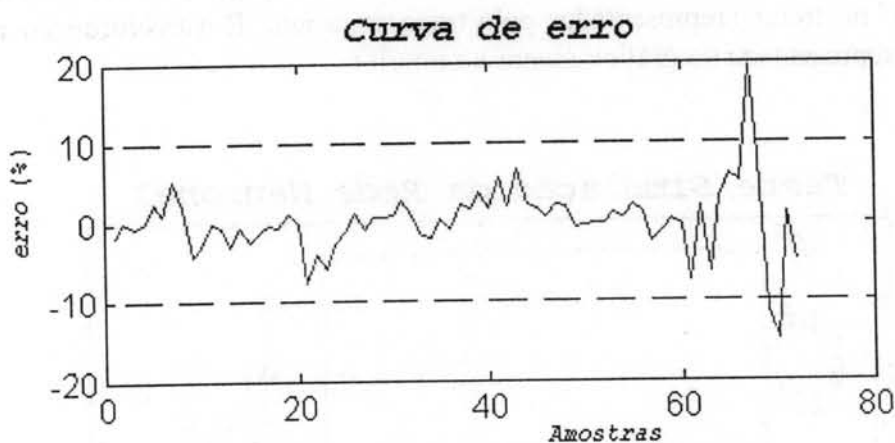


Gráfico 5.3- Erro resultante da simulação da Rede Neuronal Artificial que implementa a parte do Adaptador que estima o valor da velocidade de avanço

A Rede Neuronal Artificial usada para estimar a velocidade de rotação difere da anterior apenas no número de neurónios do nível intermédio.

A Figura 5.6, representa a estrutura desta Rede Neuronal Artificial, que tem como função implementar a parte do Adaptador que estima o valor de velocidade de rotação.

Verificámos maiores dificuldades na convergência da rede que deveria estimar a velocidade de rotação. Por este motivo a Rede Neuronal Artificial destinada a estimar a velocidade de rotação usou 200 neurónios no seu nível intermédio.

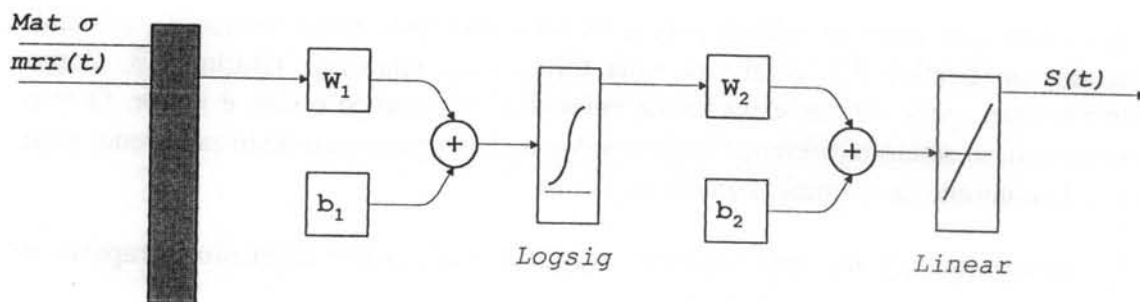


Figura 5.6 - Estrutura da Rede Neuronal Artificial que implementa a parte do Adaptador que estima a velocidade de rotação

O Gráfico 5.4, representa a comparação entre os valores estimados pela Rede Neuronal Artificial, representados pela linha a traço interrompido, e os valores dos exemplos usados no treino, representados pelo traço contínuo,. E a evolução do erro ao longo do treino, representada no gráfico menor no interior.

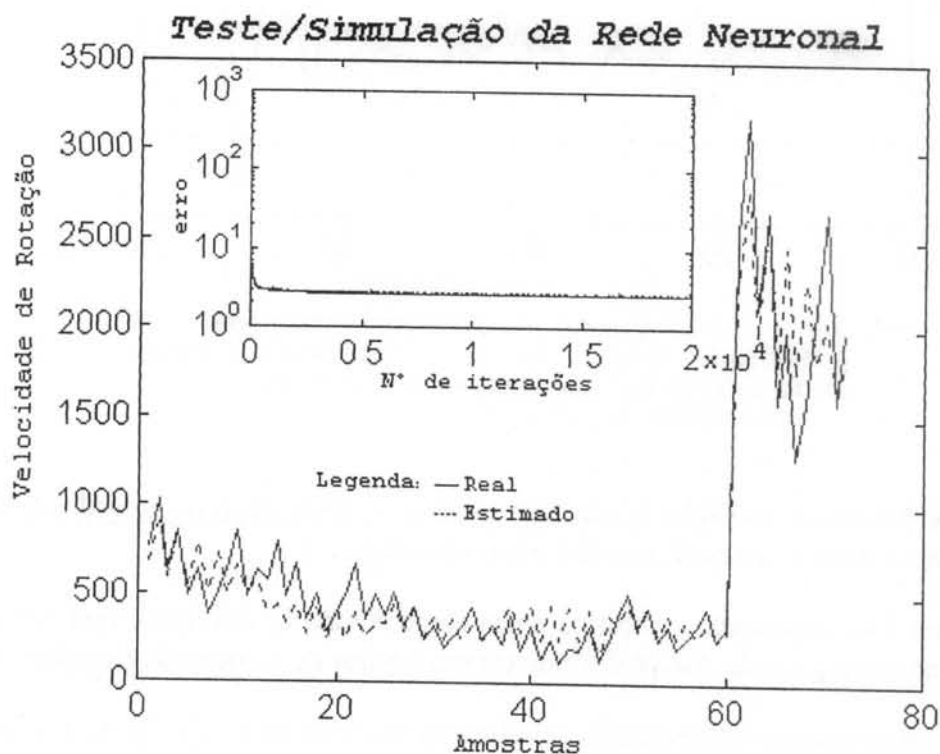


Gráfico 5.4- Teste realizado à Rede Neuronal Artificial que implementa a parte do Adaptador que estima a velocidade de rotação

O Gráfico 5.5, representa o erro que a Rede Neuronal Artificial comete ao estimar o valor da velocidade de rotação. Como podemos verificar o valor do erro, é na maior parte das situações, inferior a $\pm 10\%$.

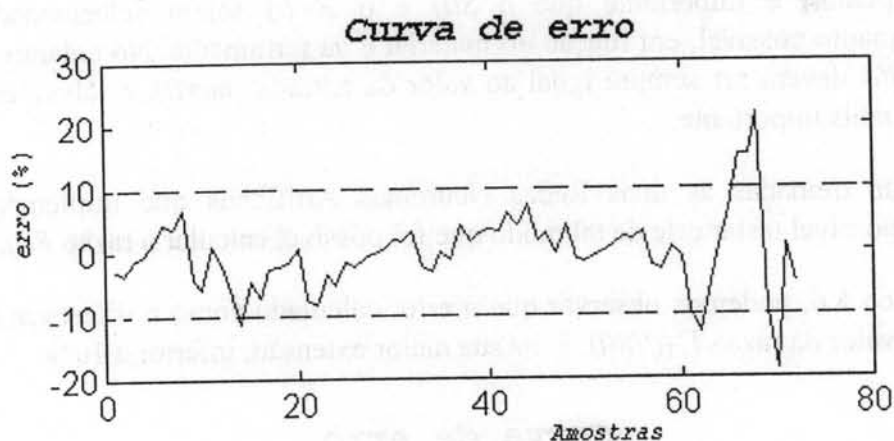


Gráfico 5.5- Erro resultante da simulação da Rede Neuronal Artificial que implementa a parte que estima o valor da velocidade de rotação

Na Figura 5.7, podemos verificar como é constituído o Adaptador. A entrada é comum às duas Redes Neurais Artificiais, e cada uma destas estima o valor para o qual foi treinada.

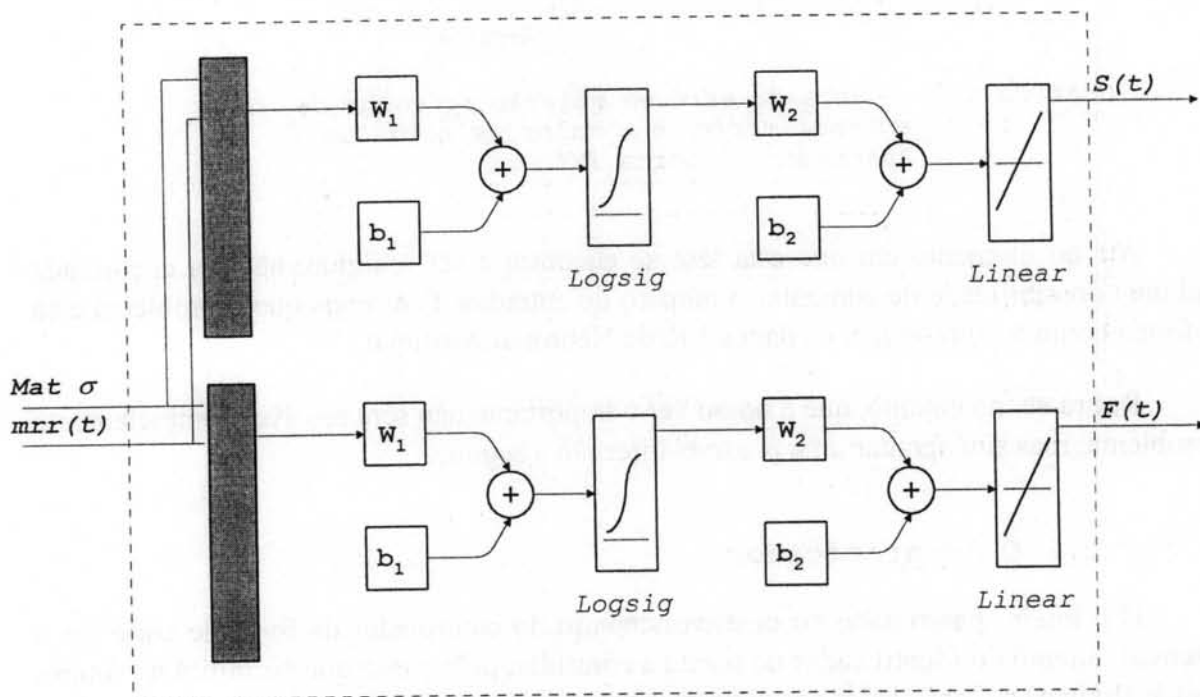


Figura 5.7 - A constituição do bloco designado de Adaptador

No Adaptador, é importante que o $S(t)$ e o $F(t)$ sejam seleccionados tão acertadamente quanto possível, em função do material e da ferramenta. No entanto o valor da razão $F(t)/S(t)$ deverá ser sempre igual ao valor da entrada $mrr(t)$, e talvez este seja mesmo o factor mais importante.

Depois de treinadas as duas Redes Neurais Artificiais que implementam o Adaptador, foi possível testar este de tal modo que foi possível calcular a razão $F(t)/S(t)$.

No Gráfico 5.6, podemos observar que o erro, calculado como a diferença entre o valor $mrr(t)$ e o valor da razão $F(t)/S(t)$, é, na sua maior extensão, inferior a 10%.

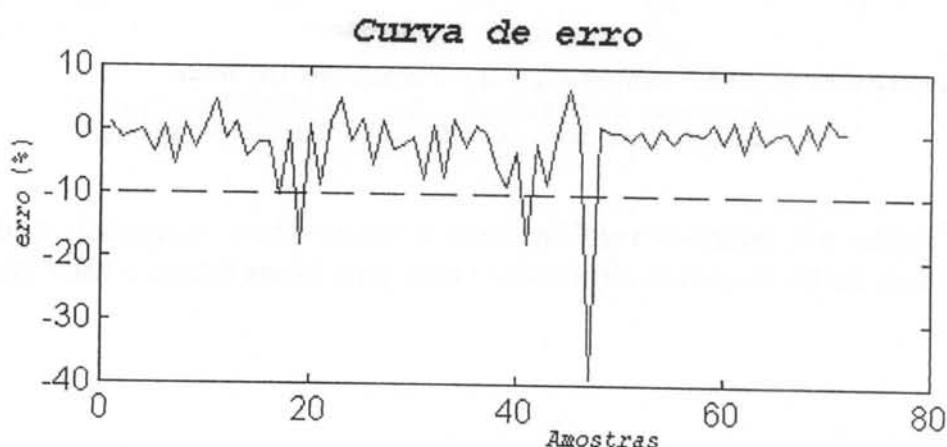


Gráfico 5.6 - Curva do erro em relação ao valor de entrada $mrr(t)$ e o valor de saída do Adaptador na forma F/S

Até ao momento em que esta tese se encontra a ser redigida, não foi encontrada nenhuma possibilidade de aumentar o número de entradas. Pensamos que o problema está na forma como se apresentam os dados à Rede Neuronal Artificial.

Refira-se, no entanto, que a nosso ver o importante não será resolver completamente o problema, mas sim apontar uma possível direcção a seguir.

5.2.2.2 O Identificador

O primeiro passo dado no desenvolvimento do controlador da força de corte foi o desenvolvimento do Identificador da planta a controlar, pois é este que permitirá no futuro, adaptar (treinar) o Controlador aos diferentes processos.

As equações (5.1) (5.2), que definem o princípio de funcionamento da planta do nosso sistema, são também características dos sistemas não lineares

$$\begin{aligned}
 Fc(t) = f(Fc(t-1), Fc(t-2), \dots, Fc(t-n), mrr(t), \\
 mrr(t-1), \dots, mrr(t-m), d(t), d(t-1), \dots, d(t-k))
 \end{aligned}
 \tag{5.1}$$

$$mrr(t) = f^{-1}(Fc(t), Fc(t-1), \dots, Fc(t-n), mrr(t-1), mrr(t-2), \dots, mrr(t-m), d(t), d(t-1), \dots, d(t-k)) \quad (5.2)$$

O Identificador desenvolvido tenta reproduzir a equação (5.2), ou, mais precisamente, simula a função f^{-1} . O Adaptador, do ponto de vista do Identificador, faz parte da planta, e Identificador terá de ser capaz de se adaptar aos erros que possam ser gerados pelo Adaptador.

Como podemos verificar pela equação (5.4), determinar dinamicamente qual o valor da razão F/S , ou os valores de $F(t)$ e $S(t)$, que permitirão manter a força de corte ($Fc(t)$) constante é uma tarefa difícil. Isto significa que teremos de usar algoritmos de cálculo que tenham a possibilidade de assimilar o comportamento do problema, como é o caso das Redes Neurais Artificiais.

$$Fc(t) = \frac{F(t)}{S(t)} \times fc(t) \times p(t) + i(t) \quad (5.3)$$

ou, de outra forma,

$$Fc(t) = \frac{F(t)}{S(t)} \times \left(fc(t) \times p(t) + \frac{S(t)}{F(t)} i(t) \right) \quad (5.4)$$

O factor $fc(t) \times p(t)$, da equação (5.4), representa os distúrbios devidos à variação da geometria da peça.

O factor $i(t)$, representa os distúrbios devidos a outros factores (desgaste da ferramenta, variações de temperatura, alteração da força específica de corte, etc.)

O factor $fc(t) \times p(t) + \frac{S(t)}{F(t)} i(t)$, representa, segundo a equação (5.4), os distúrbios que a planta sofre ao longo do processo de maquinação.

Se mantivermos $F(t)$ e $S(t)$, constantes, a força de corte $Fc(t)$, sofrerá todas as influências descritas atrás. No entanto temos a possibilidade de alterar dinamicamente, $F(t)$ e $S(t)$, o que nos permite manter a força de corte ($Fc(t)$) constante.

A decisão do Identificador gerar o valor da razão F/S , e não os valores de S e F , permite que o Identificador seja mais simples, e ao mesmo tempo, torna o Controlador independente do Adaptador. Isso é conseguido, porque o Identificador, do ponto de vista do Controlador, vai ser capaz de englobar o Adaptador na planta.

Os sistemas, cujas técnicas básicas usadas, se baseiam em sistemas adaptativos, terão de ter um conhecimento inicial do processo (ter já uma certa adaptação ao sistema em causa), antes de poderem ser implementados em sistemas físicos reais.

A maior parte dos sistemas físicos necessitam de alguma informação para começarem a trabalhar. Se essa informação inicial for deficitária ou incorrecta, poderá levar o sistema à destruição. Isto poderá fazer parte da adaptação do sistema de controlo ao sistema real, no entanto poderá ser desastroso. São situações que em alguns sistemas, não poderão ser toleradas, contudo.

O desenvolvimento do Identificador usando Redes Neurais Artificiais, implica que se disponha de alguns exemplos do funcionamento do sistema, de modo a que a Rede Neuronal Artificial possa ser treinada.

Se consideramos, que os maiores distúrbios se devem a alterações da geometria da peça, e que o nosso Identificador, necessita apenas de um conhecimento inicial, podemos fazer a simplificação apresentada na equação (5.5)

$$mrr(t) = \frac{F(t)}{S(t)} = \frac{Fc(t)}{fc(t) \times p(t)} \quad (5.5)$$

A partir do momento, que o Identificador possa usar valores reais, recolhidos da própria planta, este poderá assimilar melhor o comportamentos desta e dos seus distúrbios.

As equações (5.3) e (5.5), traduzem o problema associado à tecnologia de corte descrito no capítulo 2. Do ponto de vista do controlo, temos $mrr(t)$ como sinal de entrada e $Fc(t)$ como sinal de saída da planta a controlar.

A Figura 5.8 representa o Identificador da planta, onde se pode observar que temos como entrada, a força de corte $Fc(t)$, temos o sinal $d(t)$, que representa todas as perturbações à planta e o sinal $mrr(t)$ que é realimentado da saída para a entrada.

Quando se pretende implementar o Identificador com base numa Rede Neuronal Artificial, primeiro temos de definir a estrutura desta e o numero de regressores a usar. O numero de regressores, é o numero de vezes, que o mesmo sinal entra na rede, atrasado de igual numero de iterações (um exemplo de um regressor é o sinal $mrr(t-1)$).

No nosso caso, a estrutura da rede e o número de regressores foi definido experimentalmente, recorrendo-se ao teste de várias estruturas diferentes. Isto significa que talvez não tenhamos definido a melhor estrutura, mas definimos com certeza uma das melhores.

Comprovou-se também junto da máquina que a descrição do problema feita no capítulo 1 estava de acordo com o que se passava realmente na máquina.

Os resultados obtidos junto da máquina apenas diferem dos resultados teóricos, pelo facto de apresentarem ruído e algumas perturbações não identificadas. No entanto, pudemos identificar perfeitamente, no sinal, os efeitos referidos no capítulo 2. A ausência de figuras ilustrativas deve-se ao facto da análise ter sido feita com um osciloscópio, no momento de maquinação de várias peças, e de não ter sido possível o seu registo.

Numa primeira aproximação, ao Identificador, e pela razão que acabamos de expor, usaram-se os exemplos descritos nos capítulos 2 e 3 como geradores de conjuntos de valores. Estes valores serviram para definir a estrutura e o número de regressores que iriam ser usados pela rede neuronal, que implementa o Identificador.

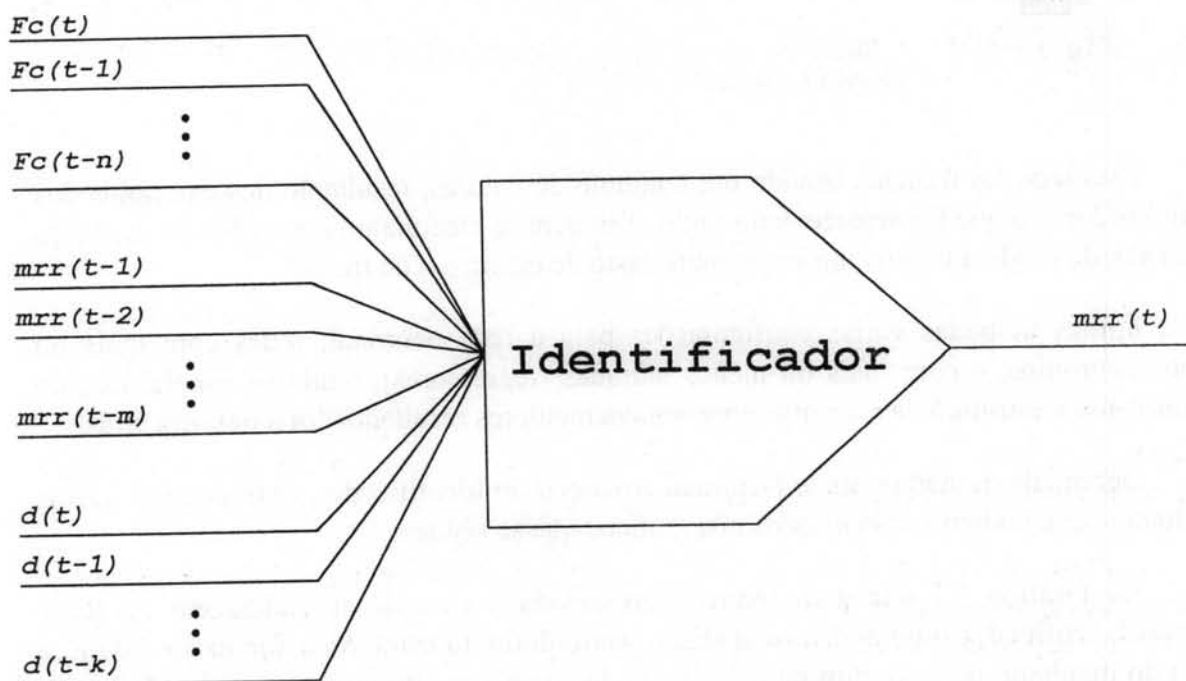


Figura 5.8 - Diagrama genérico do Identificador

Na Figura 5.9 está representada a estrutura Rede Neuronal Artificial, na qual se baseou o bloco de identificação. Como podemos observar esta é constituída por:

- sete entradas
- um nível intermédio, com a função *tansig* como função de disparo e 30 neurónios neste nível
- uma saída, e apenas um neurónio, cujo função de disparo é linear (*purelin*).

Esta rede neuronal é do tipo *feed-forward*, e usou-se o algoritmo de treino do tipo *back-propagation*.

Foi usado o programa MATLAB para a implementação e simulação do Identificador.

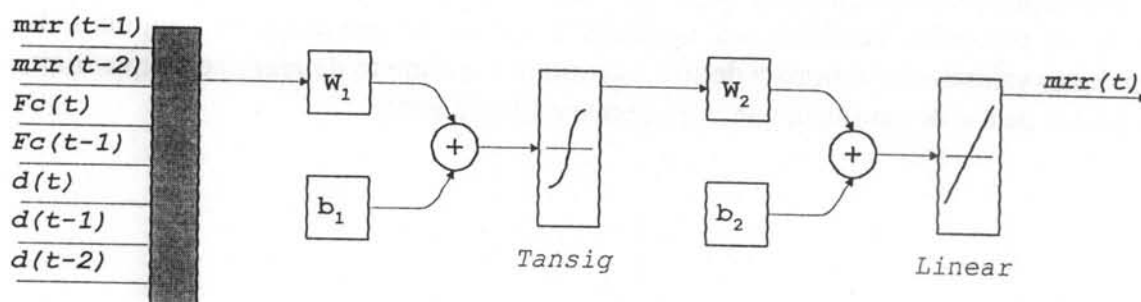


Figura 5.9 - Rede neuronal que implementa o Identificador

Esta rede foi treinada usando um conjunto de valores, resultante dos exemplos dos capítulos 2 e 3, a que foi acrescentado ruído. Também se efectuaram operações de mudança de escala, de modo a termos um leque mais vasto de exemplos de treino.

Foram treinadas várias configurações para a rede neuronal, redes com mais ou menos neurónios, e com mais ou menos entradas (regressores), tendo-se concluindo, no entanto, que a estrutura da rede que apresentava melhores resultados foi a descrita atrás.

Depois de treinada a Rede Neuronal Artificial do Identificador, obtiveram-se vários resultados, que podem ser analisados nos gráficos que se seguem.

No Gráfico 5.7 alínea *a*) temos representada a taxa de aprendizagem da Rede Neuronal Artificial. Como podemos analisar, o erro definido como *Sum Sqr Error (SSE)*, a soma do quadrado do erro, diminuiu ao longo das iterações, atingindo um valor de 0.4 ao fim de 15000 iterações. Na alínea *b*) podemos verificar que o resultado do Identificador ($mrr(t)$ estimado) se aproxima bastante do sinal usado para o treino ($mrr(t)$ real).

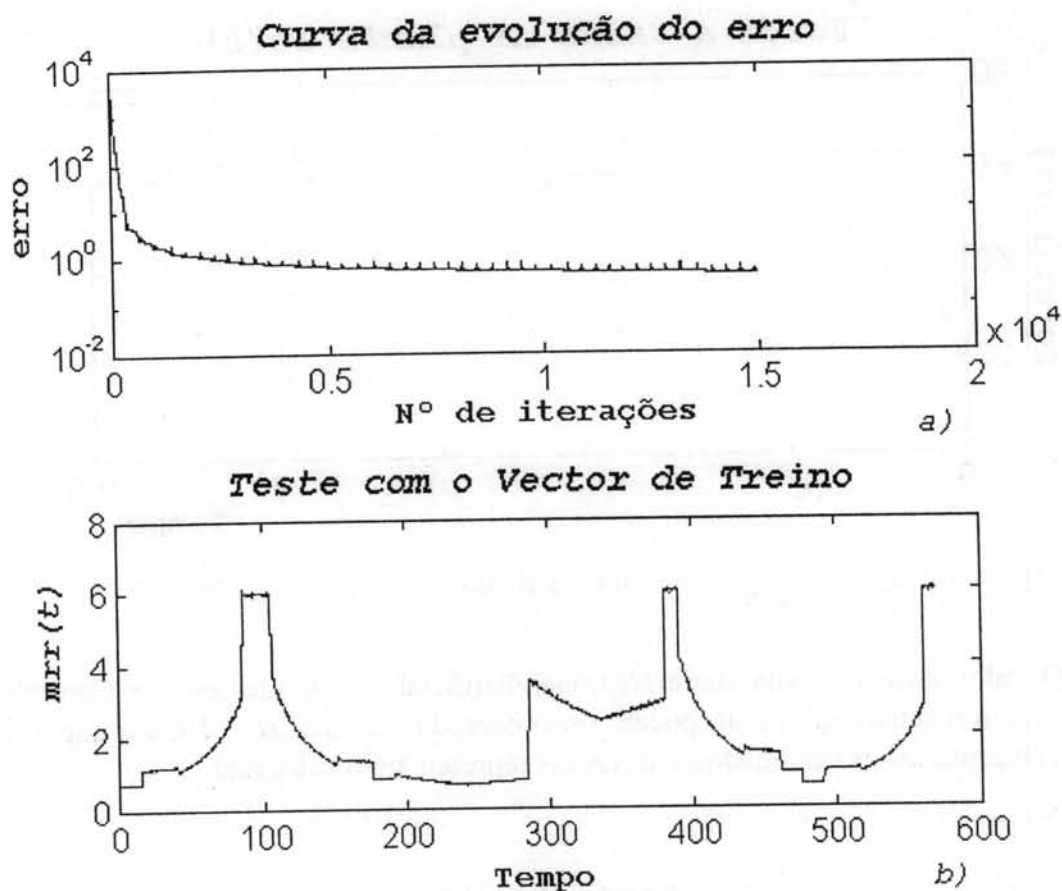


Gráfico 5.7 - Resultado do treino da rede neuronal a) Evolução do quadrado do erro b) Comparação do valor mrr estimado pela rede e o verdadeiro valor de mrr (usado para treinar a rede)

É natural que o valor estimado pela rede se aproxime bastante do resultado, quando nos estamos a servir dos sinais usados no treino da rede. Para verificarmos o comportamento do Identificador em situações diferentes da do treino, foi realizado um outro teste, com um conjunto de valores diferentes dos usados no treino da Rede Neuronal Artificial.

O Gráfico 5.8 representa as possíveis flutuações do valor da saída da planta, devido a alterações da força de referência, que serão aplicadas ao Identificador. A este conjunto acrescentou-se também um conjunto de possíveis perturbações ao sistema Gráfico 5.9.

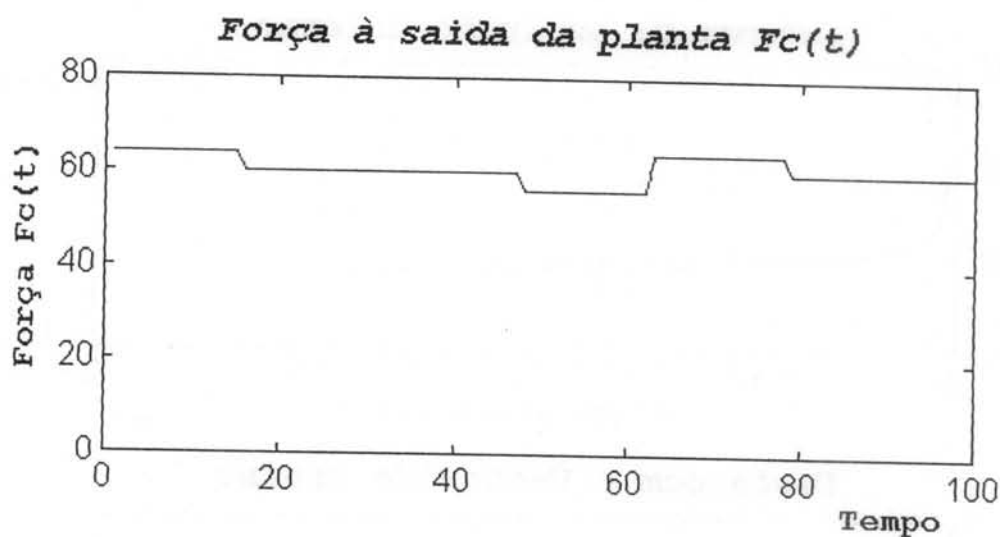


Gráfico 5.8 - Força de corte medida à saída da planta F_c

O valor estimado pela Rede Neuronal Artificial que implementa o Controlador ($mrr(t)$) e o respectivo valor real, podem ser observados no Gráfico 5.10, em que a linha sólida representa valores estimados, e as cruzes representam o valor real.

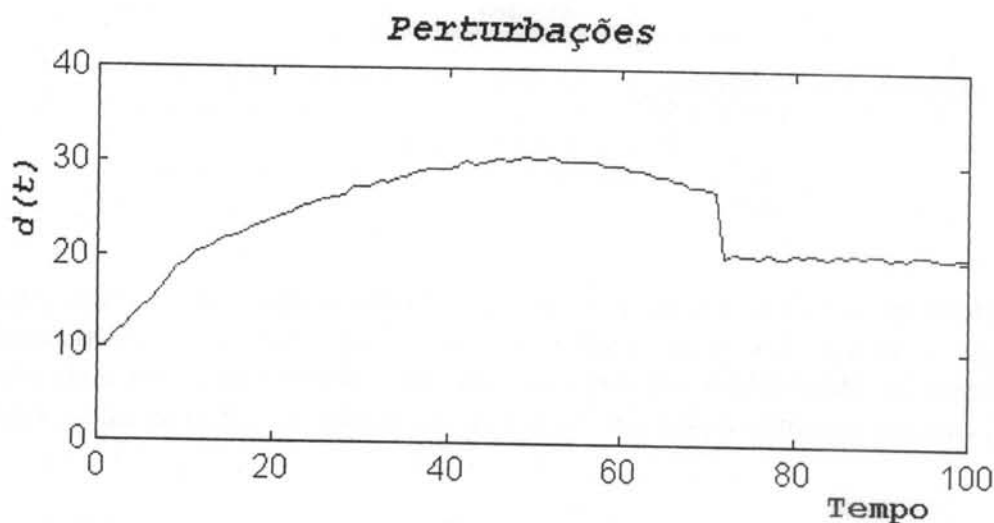


Gráfico 5.9- Perturbações apresentadas ao sistema

Analisando o Gráfico 5.10, podemos verificar que, mesmo para sinais diferentes dos usados no treino, a rede neuronal, apresenta resultados suficientemente precisos para serem usados pelo sistema de controlo.

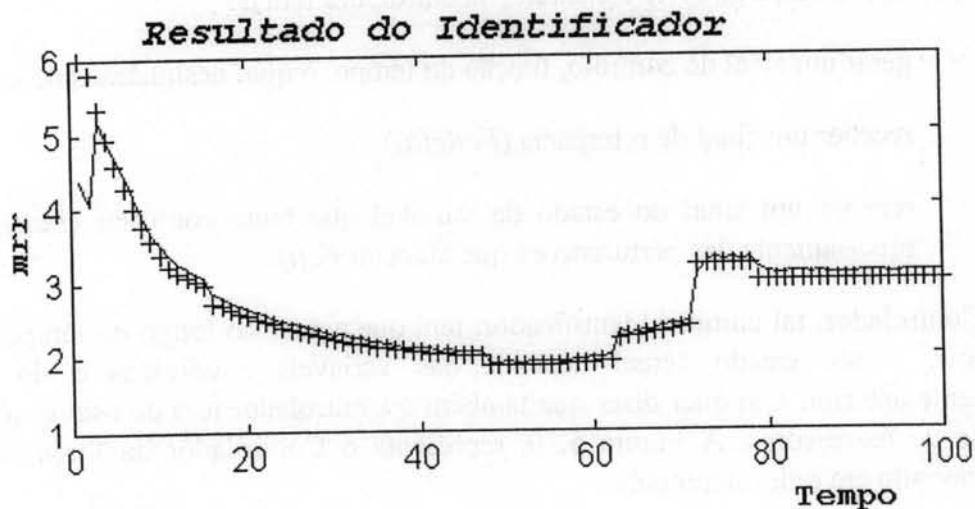


Gráfico 5.10- Comparação entre o sinal verdadeiro (representado por cruzes) e o sinal estimado pelo Identificador (representado a solido)

Como já discutimos noutro capítulo, os exemplo utilizados de treino da Rede Neuronal Artificial, são constituídos por valores de entrada e os respectivos valores de saída.

Assim sendo, ao longo do funcionamento do sistema temos constantemente valores apresentados à entrada da planta e os respectivos valores de saída desta. Isto significa que poderemos continuar a treinar a Rede Neuronal Artificial, na qual se baseia o Identificador, adaptando este o melhor possível ao funcionamento da planta.

O Identificador poderá continuar a evoluir ao longo da sua vida, ficando com uma precisão cada vez maior.

5.2.2.3 O Controlador

Ao contrário do que parece à primeira vista, o principal problema de manter a força de corte constante, não se deve às propriedades da função de transferencia da planta, mas sim aquilo que designamos de perturbações.

As perturbações são as alterações que a força de corte sofre devido a modificações das condições do processo da maquinação (desgaste, variações da geometria da peça, alterações de temperatura, etc.).

Genericamente, o nosso controlador terá de manter a força de corte constante, e igual ao valor da força de referência, a partir da realimentação que é feita por um sensor capaz de detectar as perturbações ao sistema. Este sensor é genericamente um sensor da força de corte.

Para manter a força de corte constante, o controlador terá de

- gerar um sinal de controlo, função do tempo, o qual designámos de $mrr(t)$
- receber um sinal de referência ($FcRef(t)$)
- receber um sinal do estado da variável que tenta controlar ($Fc(t)$), mais precisamente das perturbações que afectam $Fc(t)$.

O Controlador, tal como o Identificador, terá que actuar ao longo do tempo, o que significa que o seu estado actual depende das variáveis envolvidas e do estado imediatamente anterior. Isto quer dizer que também o Controlador terá de usar o que atrás designamos de regressores. A Figura 5.10, representa o Controlador da força de corte genérico, baseado em redes neuronais.

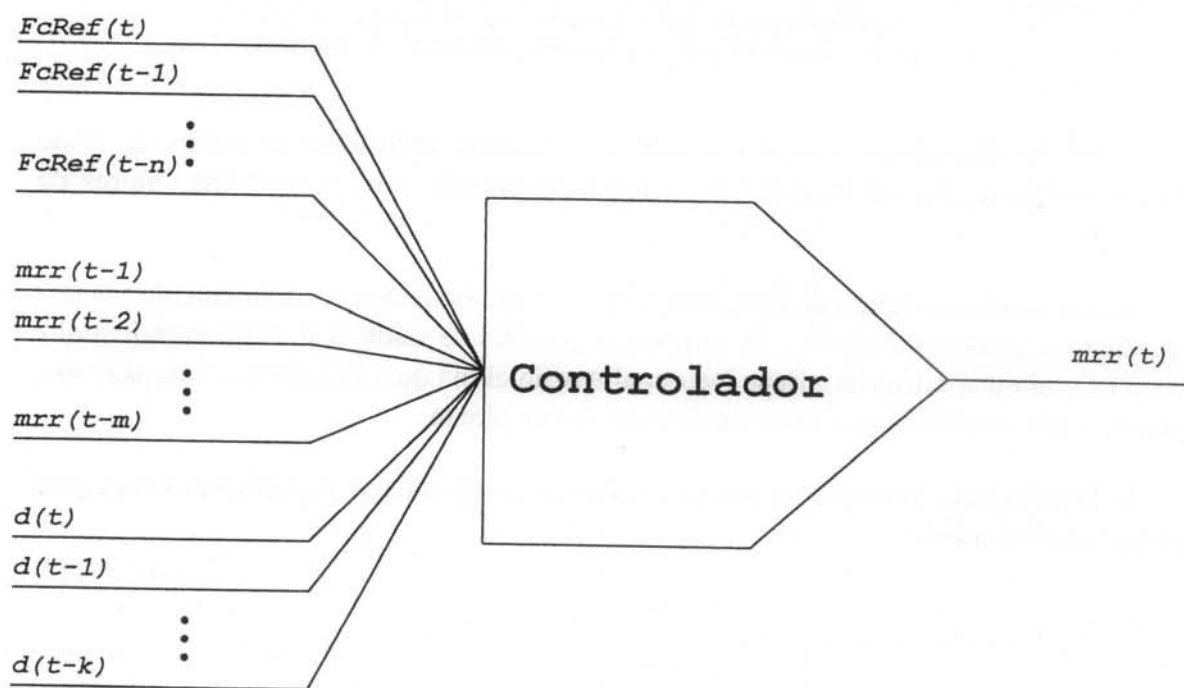


Figura 5.10 - Diagrama genérico do controlador

No controlador apresentado na Figura 5.10, temos um conjunto de entradas, que representam as variáveis envolvidas no processo e os seus respectivos regressores, e uma saída que será o sinal de controlo a ser aplicado ao Adaptador.

A Figura 5.11, representa a Rede Neuronal Artificial em que se baseou o bloco de controlo. A sua estrutura é constituída por

- sete entradas (podemos observar o numero de regressores na Figura 5.11)
- um nível intermédio, com a função *tansig* como função de disparo e 60 neurónios;

- uma saída (apenas um neurónio), cujo função de disparo é linear (*purelin*).

Esta rede neuronal é do tipo recorrente, (*Elman Network*) e usou-se o algoritmo de treino do tipo *back-propagation*, normalmente aplicado a este tipo de redes.

O Controlador também foi implementado e simulado no programa MATLAB.

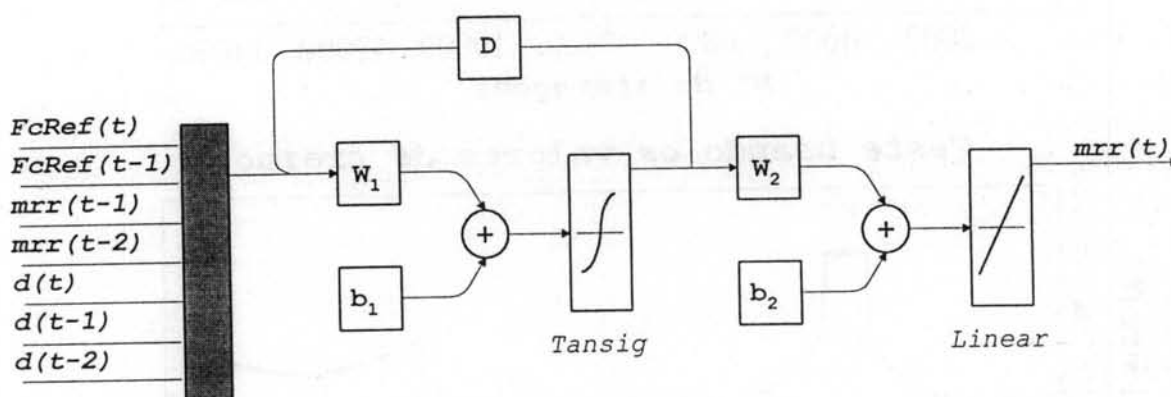


Figura 5.11 - Rede neuronal que implementa o Controlador

A estrutura encontrada, para a rede e apresentada na Figura 5.11, servirá de base à implementação do controlador real.

O numero de neurónios, que se apresentam nesta estrutura, é demonstrativo de que a regra, para calculo do numero de neurónios apresentada no capítulo anterior, nem sempre é válida.

Este numero, foi definido a partir de várias tentativas de treino da rede. A primeira estrutura para a rede tinha apenas 20 neurónios mas, no entanto, a rede não convergia. Nas tentativas seguintes, o numero foi sendo aumentado. Conseguiu-se, que a rede convergisse e atingisse um valor aceitável, ao fim de 15000 iterações, quando se usaram 60 neurónios, no nível intermédio. Tentativas com um numero de neurónios superior a 60, não demonstraram aumento significativo da velocidade de convergência.

No Gráfico 5.11, na alínea *a*), podemos observar, a curva do erro, que atingiu o valor de 0.55 no fim de 15000 iterações. Na alínea *b*) temos a curva representativa dos valores de saída, usados no treino, quase totalmente sobreposta à curva de resposta da rede (valor estimado) para a mesma entrada.

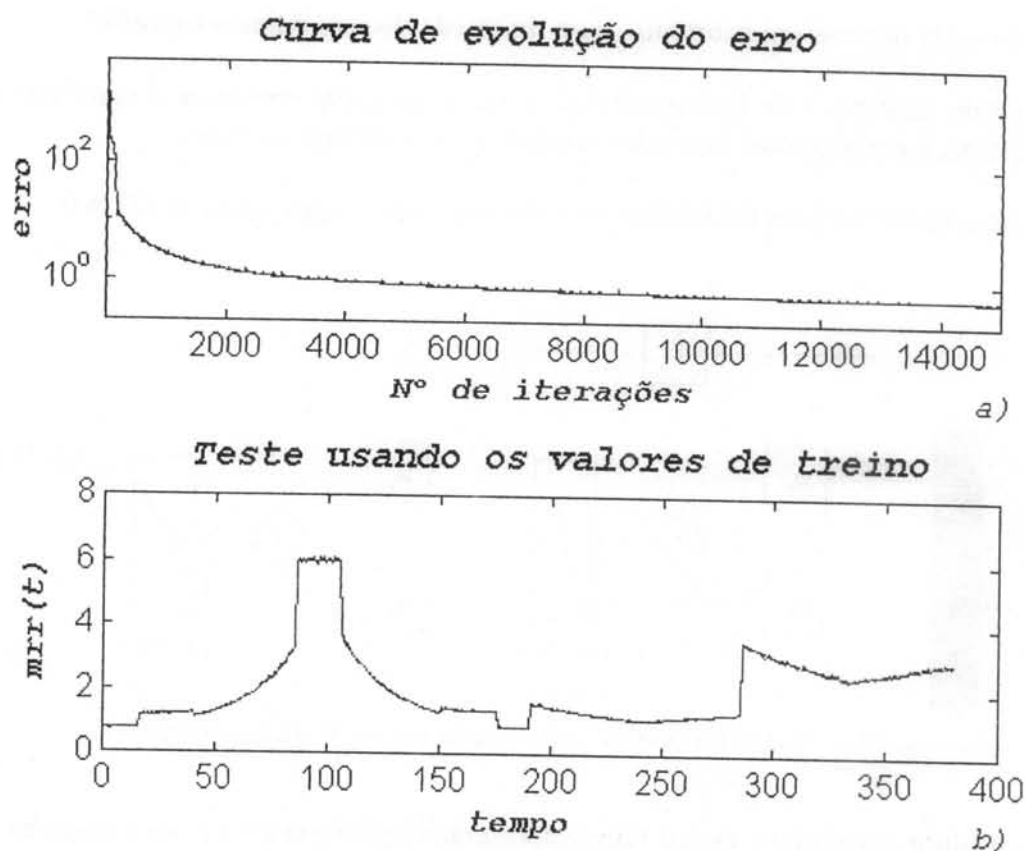


Gráfico 5.11 - a) - Curva de aprendizagem do controlador. b) - Simulação usando os valores de treino

Se permitirmos que a rede seja treinada durante um numero maior de iterações (o que acontecerá quando estiver em funcionamento), o erro terá tendência a diminuir, fazendo com que os valores representados no Gráfico 5.12, se aproximem ainda mais.

Tal como no caso anterior, não pudemos deixar de testar a rede com valores diferentes dos usados no treino. O Gráfico 5.12, representa os resultados obtidos neste teste. No gráfico os valores representados pelo símbolo '+', são os valores alvo da rede, e os representados pela linha a solido, são os valores estimados pela rede. Como podemos observar, estes valores aproximam-se.

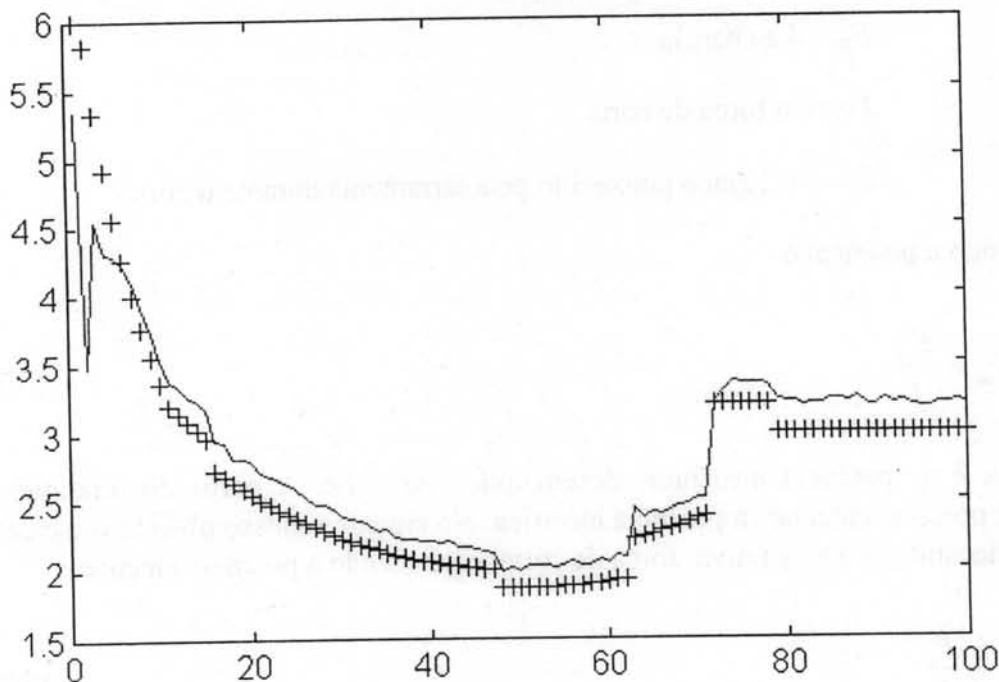


Gráfico 5.12- Teste à rede neuronal, usando valores diferentes dos usados no treino

O controlador atrás definido servirá de base a uma possível implementação real. A estrutura definida terá provavelmente de ser alterada e adaptada à realidade. No entanto, teremos uma base para acreditarmos que é possível implementar o Controlador num sistema real.

5.2.2.4 O Sensor

Para realizar a realimentação, poderíamos usar vários tipos de sensores de força, mas, no entanto, recorreu-se a uma medida indirecta da força de corte. A força de corte que se exerce no material é medida, usando um sensor de corrente, aplicado à alimentação do motor que acciona a ferramenta.

Este sensor, mede a corrente eléctrica consumida pelo motor, sendo esta corrente proporcional ao binário desenvolvido pelo motor. A força de corte, é função do binário, e do diâmetro da ferramenta.

Por outro lado, conhecendo a corrente eléctrica consumida, conhecemos também a energia consumida. A força exercida pode então ser calculada a partir da energia.

Sabemos que a energia mecânica é proporcional à força, equação (5.6)

$$E_m = Fc \times e_c \quad (5.6)$$

em que:

E_m - é a energia

F_c - é a força de corte

e_c - é o espaço percorrido pela ferramenta durante o corte.

E logo a potência é

$$P_m = \frac{E_m}{t} \quad (5.7)$$

Esta é a potência mecânica, desenvolvida no corte. A partir do rendimento da máquina é possível calcular a potência eléctrica. No entanto o nosso objectivo é calcular a potência mecânica, e a respectiva força de corte, conhecendo a potência eléctrica

$$\eta = \frac{P_m}{P_e} \quad (5.8)$$

$$F_c = \frac{\eta \times P_e \times t}{e_c} \quad (5.9)$$

Para o nosso sistema, só interessa saber qual o valor médio da força de corte entre amostragens. Para isso basta-nos determinar o espaço percorrido durante os intervalos entre amostras, e isso poderá ser calculado função das velocidades de avanço e de rotação.

Note-se que a máquina é capaz de manter a força de corte constante, se não existirem perturbações ao sistema. A força medida, em condições normais, deverá ser igual à força de referência. Os desvios registados devem-se a perturbações ao sistema por alteração das condições de corte. Estas perturbações são traduzidas por alteração da força de corte medida pelo sensor.

São estas perturbações que o Controlador deve ser capaz de corrigir, mantendo a força de corte igual à força de referência.

5.2.2.5 Testes

Depois de implementados e testados individualmente cada um dos componentes do Actuador, a fase seguinte foi treinar e testar o conjunto.

Com esse objectivo, foi implementado no programa MATLAB o diagrama do Actuador (Figura 5.4), onde se simulou e treino o seu funcionamento.

O Gráfico 5.13, representa os valores à saída do Adaptador quando este recebe à entrada o valor $mrr(t)$ gerado pelo controlador e a informação de que o material a maquinar tem uma tensão de rotura igual a 20.

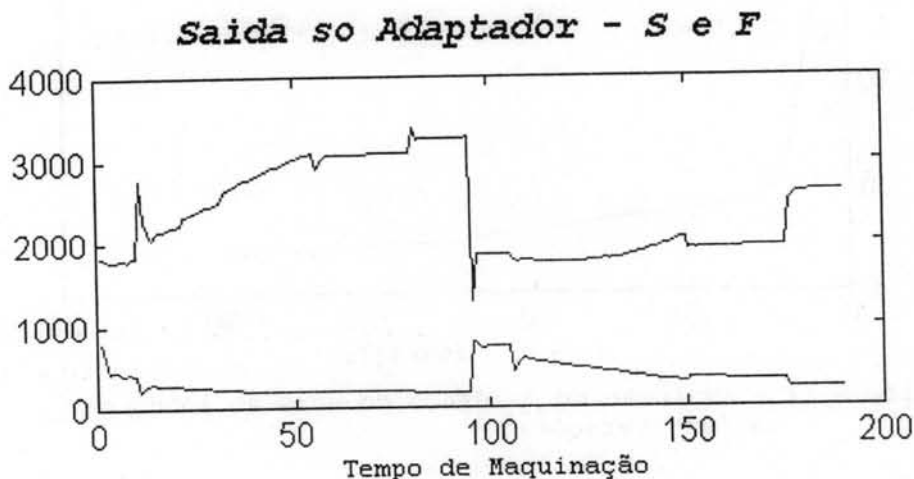


Gráfico 5.13- Os valores de F e S à saída do Adaptador

Durante a simulação os valores gerados pelo Adaptador foram usados como informação para o Identificador, que por sua vez, definia o respectivo valor de $mrr(t)$. Este valor foi então usado para adaptar (treinar) o Controlador.

Foi implementado o algoritmo de treino definido por *Elman*. Este algoritmo é usado para treinar Redes Neurais Artificiais Recorrentes, e é basicamente o algoritmo *back-propagation* adaptado às redes neurais recorrentes. O erro foi calculado como a quadrado da diferença entre o valor gerado pelo controlador e o valor gerado pelo Identificador.

Desta forma o Controlador foi-se adaptando ao processo. No Gráfico 5.14, podemos observar como evoluiu o quadrado do erro do controlador ao longo das iterações realizadas. Seria possível melhorar a qualidade do controlador se se realizasse um numero maior de iterações, naturalmente.

Não foi possível treinar o Identificador, pois tratava-se de uma simulação, e logo os dados à saída da planta eram também simulados. Na realidade, os dados usado à saída da planta eram os mesmo que foram usados para treinar o Identificador, não fazendo assim sentido que se usassem os mesmo dados para treinar o Identificador.

Quando o processo for implementado na sua totalidade, e conforme já descrito atrás, será possível continuar a treinar o Identificador e o Controlador.

O treino do Adaptador constitui um problema maior, pois a informação não é captada directamente da planta, nem do sistema, e logo o Adaptador terá de ser treinado independentemente do sistema. Uma vez treinado o Adaptador, mesmo com um erro acima do desejado, o restante sistema é capaz de se adaptar a este englobando-o dentro da planta.

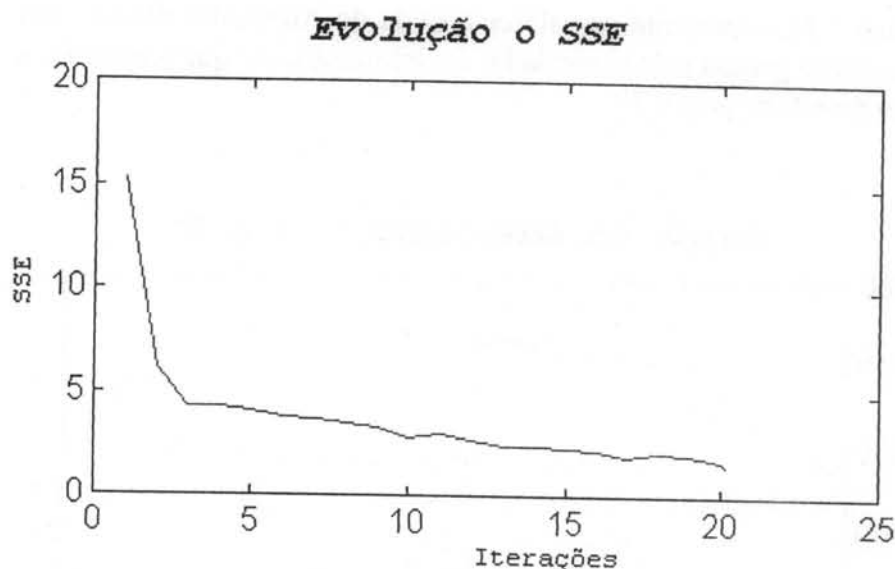


Gráfico 5.14 - Evolução do quadrado do erro ao longo das várias iterações

O Gráfico 5.15, representa o erro na saída do controlador ao fim da primeira e da vigésima iteração, quando a este é aplicado o sinal de entrada usada nos exemplos definidos atrás. Como podemos observar o erro diminui consideravelmente da primeira para a vigésima iteração.

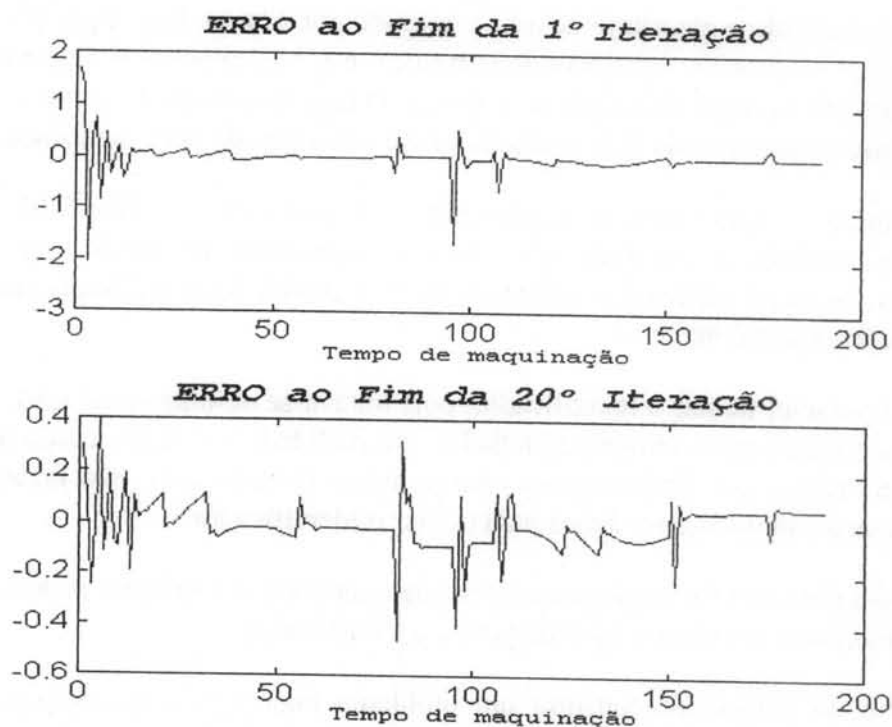


Gráfico 5.15 - Evolução do erro da primeira para a vigésima iteração

5.3 Conclusões

A conclusão mais importante neste momento, como mostram os resultados, é que é possível implementar este tipo de controlo baseado em redes neuronais. Embora tenhamos tido algumas dificuldades e o problema não se encontre totalmente resolvido, todos os dados apontam no sentido de ser possível tal implementação.

Objectivamente, todo o trabalho realizado serviu somente para demonstrar que é possível implementar o controlo do processo de corte baseado em Redes Neuronais Artificiais.

Quando se implementar o sistema, todas as considerações feitas acerca da força de corte, das perturbações, da linearidade etc. terão pouca importância, se formos capazes de criar os mecanismos que permitam ao sistema evoluir através da aprendizagem.

Todas as considerações feitas, foram importantes exactamente para termos uma noção do sistema, de tal modo que fôssemos capazes de criar tais mecanismos.

Verificamos que a ideia base é criar um sistema capaz de recolher informação do processo, de tal modo que seja capaz de identificar o erro e adaptar-se. No nosso sistema, os componentes principais são os controladores e os identificadores que interactivam de modo a gerar informação que permita a ambos evoluírem.

O Identificador recolhe informação da planta e gera a informação que permite ao Controlador adaptar-se. No entanto é o Controlador que gera informação a aplicar à planta de modo a controlar o processo e gerar informação para que o Identificador possa ser adaptado à planta.

Se a importância do Controlador não pode ser questionada, pois é este que controla o processo, também não podemos menosprezar a importância do Identificador, que permite que o Controlador se adapte correctamente.

Podemos assim concluir que o Identificador é uma das peças importantes no sistemas de controlo adaptativo.

Durante o desenvolvimento do controlo do processo de corte tivemos algumas dificuldades para a classificação da informação, tanto no Decisor onde se verificou que a maior parte da informação é do tipo qualitativo, como no Adaptador onde tivemos dificuldades em conseguir que a Rede Neuronal Artificial convergisse com o conjunto total de elementos que pretendíamos utilizar.

Também verificamos que a divisão do sistema de controlo em dois níveis era inevitável, pois temos basicamente dois tipos de informação em jogo neste tipo de controlo. Uma informação maioritariamente qualitativa, no controlo de nível superior, e uma informação maioritariamente quantitativa, no controlo de nível inferior.

As estruturas das Redes Neurais Artificiais definidas para os diferentes tipos de bloco deverão ser suficientes para tratar os problemas inerentes a cada uma das tarefas. No entanto, e se justificar poderemos alterar estas estruturas de tal modo que se possa contemplar a nova informação.

6. Conclusões

6. Conclusões	131
6.1 Sumário	132
6.2 O problema	132
6.3 A tecnologia	133
6.4 A solução	135
6.5 Vantagens e desvantagens	137
6.6 O futuro	138

6.1 Sumário

Neste capítulo tentaremos apresentar, de uma forma organizada, as conclusões a que chegamos no final do trabalho realizado, focando, nomeadamente, os seguintes aspectos desse trabalho

- o problema
- a tecnologia
- a solução
- as vantagens e desvantagens
- a futuro do controlo da força de corte.

6.2 O problema

Um dos problemas inerentes ao processo de corte usado na maquinação, advém das perturbações que a força de corte sofre devido a alterações do processo de maquinação.

Estas alterações devem-se basicamente a alterações da geometria da peça, desgaste da ferramenta, alterações de temperatura, etc..

As alterações a que força de corte está sujeita, fazem com que os parâmetros que a definem (S , F) sejam programados de uma forma conservadora, mesmo pelos programadores mais experimentados pelo que normalmente as máquinas trabalham abaixo das possibilidades disponíveis, o que se traduz em perdas de produtividade.

Em alguns casos, e na tentativa de aumentar a produtividade, os parâmetros S e F , são fixados em valores superiores aos recomendados, resultando daí prejuízos para a ferramenta e para o acabamento final da peça.

A verdadeira solução deste problema passa por manter-se a força de corte constante, e se for possível igual à força de corte óptima definida para a ferramenta, ao longo de todo o processo de maquinação.

Se tal for conseguido, poderemos garantir que

- a qualidade final do produto será superior à que seria possível atingir com o processo tradicional
- o tempo de produção será reduzido, aumentando-se assim a produtividade da oficina

- as ferramentas terão o seu tempo de vida prolongado, reduzindo-se os gastos com as ferramentas
- o programador não terá que se preocupar em otimizar o programa de fabrico deste ponto de vista
- resultará uma redução de custos associados à produção.

No entanto, o controlo da força de corte no processo de fresagem vertical é uma tarefa bastante difícil, devido, essencialmente a dois factores

- o pouco conhecimento existente sobre a planta a controlar (processo de fresagem vertical)
- o facto de a planta não ser um sistema linear.

Assim, tudo indica que, se pretendermos controlar a força de corte, teremos de implementar um controlo do tipo adaptativo.

6.3 A tecnologia

Para se implementar um controlo do tipo adaptativo, teremos de recorrer a uma tecnologia capaz de evoluir e assimilar o comportamento do sistema a controlar.

Numa primeira análise, tínhamos à disposição duas possíveis tecnologias, Logica Fuzy e Redes Neurais Artificiais, que têm vindo a demonstrar serem capazes de implementar controladores adaptativos com excelente desempenho.

Estando o Instituto de Automática Industrial, de Madrid, com quem o CCP mantém colaboração nesta área, envolvido num projecto de I&D cujo objectivo é controlar a força de corte usando Lógica Fuzy, entendeu-se que seria de investigar a utilização de Redes Neurais Artificiais na implementação do controlador de força de corte, e posteriormente comparar resultados.

O controlo adaptativo baseia-se na possibilidade de alteração dos parâmetros do Controlador em tempo real. Para isso, necessitamos de usar um sistema capaz de identificar determinadas alterações na planta e transmitir ao Controlador informação suficiente para que este, alterando os seus parâmetros, se adapte à planta em causa.

As Redes Neurais Artificiais são sistemas computacionais, de processamento paralelo, capazes de assimilar comportamentos a partir de um conjunto de exemplos.

Esta capacidade, mais a capacidade das Redes Neurais Artificiais, de generalização, extrapolando situações não previstas no conjunto de exemplos de treino, permite implementar um controlo não passível de ser implementado por processos tradicionais.

As Redes Neurais Artificiais implementam o controlo aprendendo qual o valor do sinal a aplicar à planta, para que esta produza uma determinada saída. Isso é conseguido treinando a rede com exemplos do comportamento da planta.

As Redes Neurais Artificiais implementam um controlo adaptativo se for possível *on-line* treinar a rede, recorrendo a exemplos extraídos do processo, ou seja, se for possível saber qual deverá ser a saída da rede para se gerar uma determinada saída da planta.

O Controlador recebe normalmente o sinal de referência e gera um sinal aplicar à planta. Ao gerar esse sinal, o Controlador espera que a saída da planta seja igual ao sinal de referência. Se o Controlador estiver adaptado correctamente à planta, a saída da planta deverá ser igual à referência.

A planta recebe o sinal do Controlador e produz uma determinada saída, que poderá ser diferente da esperada pelo Controlador. No entanto, o Controlador, porque se baseia em Redes Neurais Artificiais, necessita de ter uma informação de erro em relação à sua saída, de modo a que se possa usar um algoritmo de treino capaz de alterar os pesos da Rede Neuronal Artificial adaptando esta à nova situação.

Para se gerar o sinal de saída do Controlador, ou seja, o sinal de entrada da planta, necessitamos de criar um sistema capaz de gerar essa informação a partir da saída da planta. Isto quer dizer que necessitamos de ter um sistema capaz de modelizar o inverso da planta a controlar. A este sistema damos o nome de Identificador.

O Identificador necessita modelizar o inverso da planta, pelo que é necessário conhecer a função de transferência da planta e a função inversa, de modo a podermos conhecer a sua entrada, conhecendo a sua saída.

Na maioria dos casos, a função de transferência da planta é totalmente desconhecida, podendo mesmo ter características não lineares.

Para se poder modelizar uma planta nestas condições, o Identificador deve ser capaz de extrair as características da planta a partir de um conjunto de valores de entradas e respectivos valores de saídas.

Devido a esta última característica do Identificador, este terá também de ser implementado usando uma Rede Neuronal Artificial. O Identificador, ao estar ligado entre a entrada e saída do sistema que pretende modelizar, recolhe informação suficiente para modelizar (treinar a Rede Neuronal Artificial) a planta.

Resumindo, um Controlador baseado em Redes Neurais Artificiais, será sempre constituído pelo Controlador propriamente dito e pelo Identificador, interligados como se mostra na a Figura 6.1, de tal modo que depois de colocarmos o sistema em funcionamento, aqueles módulos evoluirão de modo a controlar a planta, adaptando-se a novas situações de controlo.

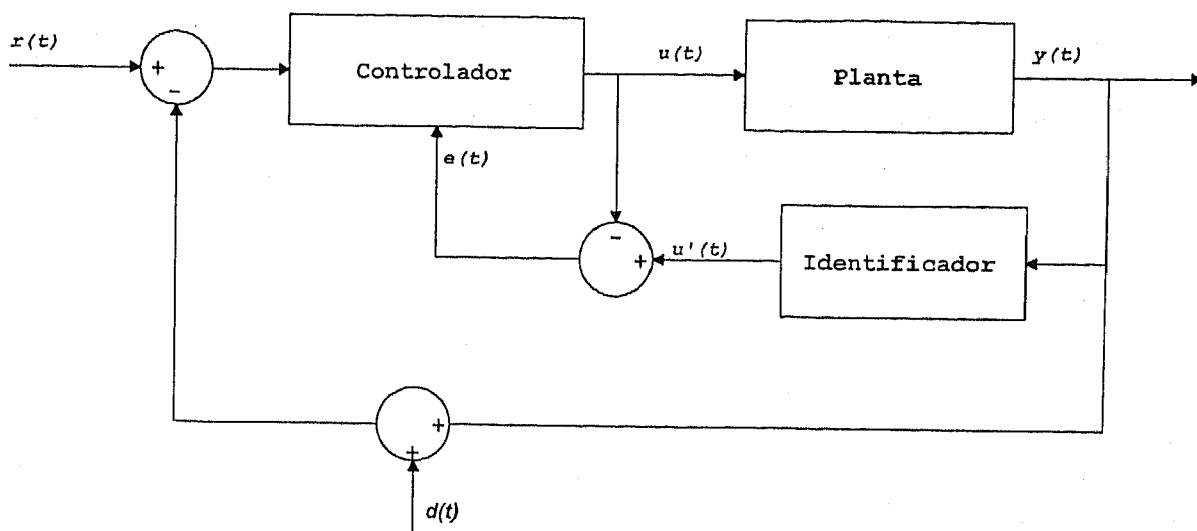


Figura 6.1 - Sistema de controlo baseado em redes neurais

6.4 A solução

O problema de controlo foi dividido em dois grandes blocos, cada bloco associado ao tipo de informação que deve tratar.

Assim definiu-se um bloco, designado de Decisor, hierarquicamente de nível superior, cuja função é decidir qual a força óptima de corte a ser usada no processo de maquinação. Esta decisão deverá ser tomada em função das características do processo, tais como, o material a maquinar, o tipo, material, e numero de dentes da ferramenta, a qualidade final, o tempo de maquinação pretendido, etc..

O Decisor tem também a possibilidade de alterar dinamicamente a força óptima de corte, em função da temperatura e vibrações detectadas no processo.

Hierarquicamente, existe também um Controlador de nível inferior, que designamos de Actuador.

O Actuador deve o seu nome, à sua função de actuar sobre a planta (máquina), com o objectivo de manter a força de corte igual à força óptima de corte, definida pelo Decisor.

Este bloco, é o bloco principal do sistema de controlo, pois se é possível ao operador ou programador da máquina, definir, recorrendo a tabelas fornecidas pelos fabricantes de ferramentas e à sua experiência, qual deverá ser a força óptima de corte (tarefa do Decisor), será impossível ao operador da máquina actuar sobre a máquina em tempo real, de modo a manter a força de corte igual à força óptima de corte definida.

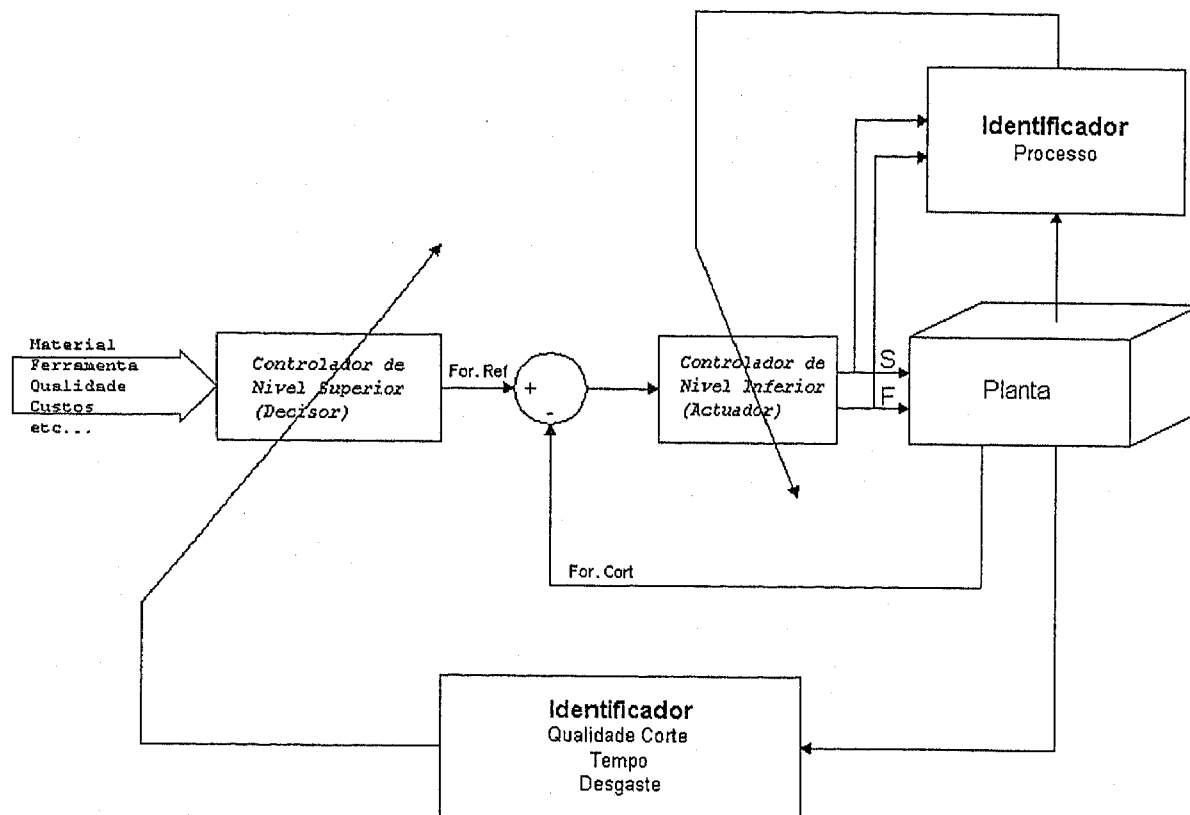


Figura 6.2 - Diagrama genérico do Controlador

No diagrama genérico do Controlador, que se encontra na Figura 6.2, poderemos identificar os blocos que referimos atrás.

Ambos os blocos se baseiam na mesma tecnologia, Redes Neurais Artificiais, para implementar o controlo adaptativo.

Durante a elaboração desta tese deu-se maior importância ao Controlador de nível inferior, por dois motivos: em primeiro lugar porque a informação a tratar é maioritariamente quantitativa, e em segundo porque se trata de uma tarefa impossível de executar pelo homem.

O Controlador de nível inferior (Actuador) foi assim completamente desenvolvido e testado, enquanto que o Controlador de nível superior (Decisor) não ultrapassou uma fase de aproximação à solução possível.

Não quisemos com isto menosprezar a importância de se definir qual a força óptima de corte, mas achamos simplesmente que esta só teria importância depois de termos em funcionamento um sistema capaz de manter essa força constante.

Todo o trabalho desenvolvido nesta tese, se baseou na informação recolhida junto à máquina, e na simulação de Redes Neurais Artificiais, usando o programa MATLAB.

A implementação real do sistema poderá exigir algumas alterações, mas, no entanto, estamos confiantes nos resultados obtidos, e acreditamos que controlar a força de corte através do uso de Redes Neurais Artificiais é possível e eficaz.

6.5 Vantagens e desvantagens

A solução encontrada, recorre a um controlo adaptativo baseado em redes neuronais, pelo que qualquer afirmação sobre vantagens e desvantagens não poderá ser entendida em termos comparativos com outras tecnologias. Realmente, não dispomos de elementos suficientes acerca de outras tecnologias que nos permitam fazer tais comparações. Podemos afirmar que se trata de um sistema de controlo bastante evoluído e com potencialidades ainda por explorar.

O sistema desenvolvido apresenta algumas vantagens que podemos resumir da seguinte forma

- controlo completamente adaptativo
- não necessita de intervenção Humana para evoluir
- o sistema que modeliza a planta continuará a fazer evoluir o modelo de uma forma automática
- caso seja necessário uma resposta em tempo reduzido, é possível implementar o sistema em *hardware* específico para o efeito, com velocidades de resposta acima das disponíveis nos sistemas de computação correntes em oficinas.

O uso de Rede Neuronal Artificial na implementação do sistema de controlo apresenta também algumas desvantagens

- requer a disponibilidade de um conjunto de exemplos que sejam representativos do sistema que se pretende modelizar
- em alguns casos a Rede Neuronal Artificial, tem dificuldades em convergir, principalmente se os exemplos usados não representarem correctamente o processo
- o tempo de treino da rede, até que esta apresente resultados aceitáveis, poderá ser elevado.

- a estrutura da rede deve ser bem definida, para que não se dê o caso de existir *underfitting* ou *overfitting*.

6.6 O futuro

Pensamos que será importante, como continuação do trabalho realizado até ao momento, que se implementem e testem os modelos das Redes Neurais Artificiais em sistemas (máquinas) reais. Isso permitiria validar a estrutura das Redes Neurais Artificiais usadas para implementar os diferentes blocos, e treiná-los com valores reais.

No momento actual os produtores de máquinas ferramenta tentam resolver ainda um conjunto de problemas ligados à integração das máquinas em células flexíveis de fabrico.

No entanto será inevitável que todas as marcas produtoras de máquinas CNC caminhem nesta direcção do controlo *on-line* da força de corte. A concorrência do mercado assim o forçará.

Referências

- [1] Acácio Teixeira da Rocha; *Tecnologia Mecânica - Volume III*; Coimbra Editora; 1977.
- [2] J. Paulo Davim; *Princípios da Maquinagem*; ALMEDINA; 1995.
- [3] Vitor Polónia; *A Escolha da máquina - Ferramenta*; FEUP - DEP. MAT. PROC. TEC.; 1983.
- [4] Jesus A. D. Rivera; *Input/Output Linearization of Control Affine Systems Using Neural Networks*; A dissertation submitted to the cybernetics department in partial fulfilment of the requirements for the degree of doctor of philosophy; July 1996.
- [5] Ole Ravn, Paul H Sørensen, Magnus Nørgaard; *What is Adaptive Control*; Neural Network Project; IAU; 1996.
- [6] Astrom & Wittenmark; *Adaptive Control*; Addison-Wesley; 1995.
- [7] W.S. McCulloch & W. Pitts *A logical calculus of the ideas immanent in neurons activity*; bulletin of mathematical biophysics 5 (115 -133); 1943
- [8] Jacek M. Zurada; *Introduction to artificial neural systems*; west publishing company; 1992
- [9] Eric Davalo and Patrick Naïm; *Neural Networks*; Macmillan; 1991
- [10] CCP & IAI; *MATIMAC- Machine Tool: Intelligent Monitoring and Control*; 1996
- [11] J.A. Feldman & D.H. Ballard; *Connectionist Models and their properties*; Cognitive Science, 6; 1982
- [12] Demuth Howard & Beale Mark; *Neural Networks Toolbox User's Guide*; The Math Works Inc; 1994.
- [13] Elman, J. L.; *Finding structure in time*; Cognitive Science, 14; 1990.
- [14] Steve Lawrence, C. Lee Giles, Ah Chung Tsoi; *What Size Neural Networks Gives Optimal Generalization? Convergence Properties of Backpropagation*; Technical Report, NEC Research Institute; 1996.

-
- [15] Ben J. A. Kröse & P. Patrick Van Der Smagt; *An introduction to neural network*; University of Amsterdam Faculty of Mathematics & Computer Science; 1993
- [16] D. O. Hebb; *The Organization of Behaviour*; Wiley; 1949
- [17] M. Minsky & S. Papert; *Perceptrons: An Introduction to Computational Geometry*; MIT Press; 1969
- [18] J. J. Hopfield; *Neural Network and Physical Systems with Emergent Collective Computational abilities*; Proceedings of National Academy of Sciences 79; 1982
- [19] F. Rosenblatt; *Principles of Neurodynamics*; Spartan Books New York; 1959

7. Script's usados no MatLab

7.1 Valores usados no treino do Controlador do bloco Decisor

```
y1=[24;  
    24;  
    40;  
    40;  
    70;  
    70;  
    110;  
    110;  
    170;  
    170;  
    300;  
    300;];  
u1=[65 0 0 10 2 0;  
    65 0 0 10 2 0;  
    65 0 0 12 2 0;  
    65 0 0 12 2 0;  
    65 0 0 16 2 0;  
    65 0 0 16 2 0;  
    65 0 0 20 2 0;  
    65 0 0 20 2 0;  
    65 0 0 25 2 0;  
    65 0 0 25 2 0;
```

```
65 0 0 32 2 0;  
65 0 0 32 2 0;];  
  
u=[u1]'; %Valores de S«entrada  
y=[y1]'; % Valores de saída (Força de corte)  
  
save cont_s Ficheiro onde ficou gravado estes valores
```

7.2 Script usado para treinar o Controlador do bloco Decisor

```
clear all  
load cont_s;  
p=[0 65;0 10;0 10;5 35;0 8;0 10];  
y1=y./10;  
[w1,b1,w2,b2]=initff(p,60,'logsig',1,'purelin');  
[w1,b1,w2,b2,te,tr]=trainbpx(w1,b1,'logsig',w2,b2,'purelin',u,y1,[10 15000 .02]);  
a1=simuff(u,w1,b1,'logsig',w2,b2,'purelin');  
clf  
plot(a1.*10)  
hold on  
pause  
plot(y)  
hold off
```

7.3 Script usado para testar o Controlador do bloco Decisor

```
load cont_s
clg
%whitebg;
u(1,:)=u(1,:)+20;
subplot(211),semilogy(tr(1,:)),axis([1 3000 .01 4000]);;
a1=simuff(u,w1,b1,'logsig',w2,b2,'purelin');
subplot(212)
plot(a1.*10)
hold on
plot(y+10)
```

7.4 Script usado para gerar a maior parte dos dados usados no treino das Redes Neurais Artificiais do bloco de nível inferior

```
x=1:1:95;
whitebg;

load sf
S=6000;
F=430;
a=F./S;
p=x;
p(1:10)=0*ones(1,10);
p(11:60)=1+1/15*x(11:60);
p(61:80)=1+1/15.*x(60).*ones(1,20);
```

```

p(81:95)=8.*ones(1,15);

Sec=p.*a(1:95);
Fc=150*Sec;
t=x./20;
plot(t,Fc),axis([0 5 0 120]) ;
%subplot(221),plot(t,Fc),axis([0 5 0 120]) ;
[a,b]=max(Fc);
pause;
x=1:1:95;
y=0+6.^(x./80);
subplot(222),plot(t,y),axis([0 5 0 120]) ;
fc=x;
fc(1:55)=150+1/2.*x(1:55);
fc(56:95)=150*ones(1,40);
Ffc=Sec.*fc;
subplot(223), plot(t,fc),axis([0 5 140 180]) ;
fsoma=y.*sign(Fc)+Ffc;
subplot(224),plot(t,fsoma),axis([0 5 0 120])
save fsoma fsoma;

```

7.5 Script usado para treinar o Identificador do bloco de nível inferior

```

load iden_i1;
p=[0 150;0 150;0 150;0 150;0 150;0 150;0 5;0 5;0 5];
fs(1:95)=fsoma(95:-1:1);

```

```

d=[fs,fsoma a*2 b*2 a b fsoma fs];
d=d+1;

FcRef=70*ones(1,95);
y=[FcRef,FcRef+10, FcRef-30,...
    FcRef+5, FcRef+20, FcRef-10];

u=y./d;
u(find(u>6))=6*ones(size(find(u>6)));

d=d+2.*rand(size(d));
d=d+3.*rand(size(d));
d=d+0.5.*rand(size(d));
Y=delaysig(y,1,3);
D=delaysig(d,1,3);
U=delaysig(u,1,3);
in=[Y;D;U];
[w1,b1,w2,b2]=initff(p,30,'tansig',1,'purelin');
[w1,b1,w2,b2,te,tr]=trainbpx(w1,b1,'tansig',w2,b2,'purelin',in,u,[150 15000 .02 .001
1.02 .6 .95 1.1]);
a1=simuff(in,w1,b1,'tansig',w2,b2,'purelin');
clf
plot(a1)
hold on
plot(u)
hold off

```

7.6 Script usado para testar o Identificador

```

load iden_111
whitebg;
subplot(211),semilogy(tr(1,:));
a1=simuff(in,w1,b1,'tansig',w2,b2,'purelin');
subplot(212)
plot(a1)
hold on
plot(u)
clg
y=60.*(ones(1,100))+2.*sign(sin((1:100)./10))+2.*sign(sin((1:100)./5));
a=10:20;
b=20+10.*sin((1:60)./25);
c=20*ones(1,29);
d=[a,b,c];
u=y./d;
u(find(u>6))=6*ones(size(find(u>6)));

d=d+.5.*rand(size(d));
d=d+0.2.*rand(size(d));

plot(y),axis([0 100 0 80]);
pause
plot(d),axis([0 100 0 40]);
pause
Y=delaysig(y,0,1);
D=delaysig(d,0,2);
U=delaysig(u,1,2);
in=[Y;D;U];
a1=simuff(in,w1,b1,'tansig',w2,b2,'purelin');
plot(a1)
hold on
plot(u,'b+')

```

7.7 Script usado para treinar o Controlador do nível inferior

```
clear all
load iden_il;
p=[0 150;0 150;0 150;0 150;0 150;0 5;0 5];
fs(1:95)=fsoma(95:-1:1);

d=[fs,fsoma, a*2, b*2, a b];
d=d+1;

FcRef=70*ones(1,95);
y=[FcRef,FcRef+10,...
   FcRef-5, FcRef+5];

u=y./d;
u(find(u>6))=6*ones(size(find(u>6)));

d=d+2.*rand(size(d));
d=d+0.5.*rand(size(d));
Y=delaysig(y,0,1);
D=delaysig(d,0,2);
U=delaysig(u,1,2);
in=[Y;D;U];
pause
[w1,b1,w2,b2]=initelm(p,60,1);
[w1,b1,w2,b2,te,tr]=trainelm(w1,b1,w2,b2,in,u,[150 15000 .02]);
a1=simuelm(in,w1,b1,w2,b2);
clf
```

```
plot(a1)
hold on
plot(u)
hold off
```

7.8 Script usado para testar o Controlador

```
load cont_i
clg
%whitebg;
subplot(211),semilogy(tr(1,:)),axis([1 15000 .2 5000]);;
a1=simuelm(in,w1,b1,w2,b2);
subplot(212)
plot(a1)
hold on
plot(u)
pause
clg
y=60.*(ones(1,100))+2.*sign(sin((1:100)./10))+2.*sign(sin((1:100)./5));
a=10:20;
b=20+10.*sin((1:60)./25);
c=20*ones(1,29);
d=[a,b,c];
u=y./d;
u(find(u>6))=6*ones(size(find(u>6)));
```

```
d=d+.5.*rand(size(d));
d=d+0.2.*rand(size(d));

plot(y),axis([0 100 0 80]);
pause
plot(d),axis([0 100 0 40]);;
pause
Y=delaysig(y,0,1);
D=delaysig(d,0,2);
U=delaysig(u,1,2);
in=[Y;D;U];
a1=simuelm(in,w1,b1,w2,b2);
plot(a1)
hold on
plot(u,'b+')
```

7.9 Scripts usado para treinar/testar o Adaptador

7.9.1 estimar a velocidade de avanço

7.9.1.1 treinar

```
%clear all
load adapt_if
%whitebg;
%subplot(211),semilogy(te(1,:));
a1=simuff(u,w1,b1,'logsig',w2,b2,'purelin');
```

```
%subplot(212)
plot(a1.*1000,':')
hold on
plot(y(2,:))
```

7.9.1.2 Testar

```
%clear all
load adapt_if
%whitebg;
%subplot(211),semilogy(te(1,:));
a1=simuff(u,w1,b1,'logsig',w2,b2,'purelin');
%subplot(212)
plot(a1.*1000,':')
hold on
plot(y(2,:))
```

7.9.2 Estimar a velocidade de Rotação

7.9.2.1 Treinar

```
clear all
load adapt_i;
p=[0 200;0 5];
ys1=ys./1000;
[w,b]=initlin(us,ys1);
[w,b]=svelin(us,ys1);
```

```
pause
[w1,b1,w2,b2]=initelm(p,100,1);
[w1,b1,w2,b2,tr,te]=trainbpx(w1,b1,w2,b2,us,ys1,[50 20000 .02 .0001]);
a1=simuff(us,w1,b1,w2,b2);
save adapt_is
clf
plot(a1)
hold on
plot(ys)
hold off
```

7.9.2.2 Testar

```
clear all
load adapt_is
%whitebg;
%subplot(211),semilogy(te(1,:));
a1=simuff(u,w1,b1,'logsig',w2,b2,'purelin');
%subplot(212)
plot(a1.*1000,':')
hold on
plot(y(1,:))
```

7.10 Script usado para treinar o conjunto (Controlador, Identificador e adaptador)

```
clear all
load cont_t
load adv_f
```

```

load adv_s
load idv

d=fsoma;
d=[d d];
d_1=0;
d_2=0;

mrr_1=0;
mrr_2=0;

Fc=[FcRef-30 FcRef+10];
Fc_1=0;
errmea=zeros(1,21);
errsqr=zeros(1,21);

for k=1:20,
for j=1:190,
in=[Fc(j); Fc_1;d(j);d_1;d_2;mrr_1;mrr_2];
mrr(j)=simuelm(in,w1c,b1c,w2c,b2c);

u=[20;mrr(j)./14];
F(j)=simuff(u,w1f,b1f,'logsig',w2f,b2f,'purelin');

S(j)=simuff(u,w1s,b1s,'logsig',w2s,b2s,'purelin');

mrr_e(j)=simuff(in,w1i,b1i,'tansig',w2i,b2i,'purelin');

erro=mrr-mrr_e;
mrr_2=mrr_1;
mrr_1=14*(F(j)./S(j));
%mrr_1=mrr(j);

```

```

d_2=d_1;
d_1=d(j);
%-----
%-----Treino do Controlador-----
%-----
clear a1 a2 s1 s2 dw1c dw2c r q tr d1 d2 w e SSE
tp = nndef(tp,[5 500 0.01 0.001 1.05 0.5 0.95 1.04]);
df = tp(1);
me = tp(2);
eg = tp(3);
lr = tp(4);
lr_inc = tp(5);
lr_dec = tp(6);
mc = tp(7);
er = tp(8);

% INITIALIZE MATRIX SIZES
% =====

[r,q] = size(in);
s1 = length(b1c);
s2 = length(b2c);

a1 = zeros(s1,q);
a2 = zeros(s2,q);

dw1c = w1c*0; db1c = b1c*0;
dw2c = w2c*0; db2c = b2c*0;

tr = zeros(1,me+1);

```

```

% SIMULATE NETWORK FOR Q TIMESTEPS

init_a1 = zeros(s1,1);
[a1,a2] = simuelm(in,w1c,b1c,w2c,b2c,init_a1);

% CALCULATE ERRORS

e = mrre(j) - a2;
SSE = sumsqr(e);
%tr(1) = SSE;

% CALCULATE DERIVATIVES

d2 = deltaln(a2,e);
d1 = deltatan(a1,d2,w2c);

% PLOTTING
%clf
%h = ploterr(tr(1),eg);
%message = sprintf('TRAINELM: %%g/%%g, lr = %%g, epochs, SSE =
%%g.\n',me);
%fprintf(message,0,lr,SSE)

% INITIALIZE MOMENTUM & TRAIN

current_mc = 0;
for i=1:me

% ERROR CHECK

if SSE < eg, i=i-1; break, end

```

```
% CALCULATE WEIGHT CHANGES
```

```
[dw1c,db1c] = learnbpm([in; init_a1 a1(:,1:(q-1))],d1,lr,current_mc,dw1c,db1c);  
[dw2c,db2c] = learnbpm(a1,d2,lr,current_mc,dw2c,db2c);  
current_mc = mc;
```

```
% CALCULATE NEW WEIGHTS
```

```
new_w1c = w1c + dw1c; new_b1c = b1c + db1c;  
new_w2c = w2c + dw2c; new_b2c = b2c + db2c;
```

```
% SIMULATE NEW NETWORK FOR Q TIMESTEPS
```

```
[new_a1,new_a2] = simuelm(in,new_w1c,new_b1c,new_w2c,new_b2c,init_a1);
```

```
% CALCULATE ERRORS
```

```
new_e = mrre(j) - new_a2;  
new_SSE = sumsqr(new_e);
```

```
% MOMENTUM & ADAPTIVE LEARNING RATE
```

```
if new_SSE > SSE*er  
    lr = lr * lr_dec;  
    current_mc = 0;  
else  
    if new_SSE < SSE, lr = lr * lr_inc; end
```

```
% ACCEPT NEW NETWORK
```

```
w1c = new_w1c; b1c = new_b1c; w2c = new_w2c; b2c = new_b2c;  
a1 = new_a1; a2 = new_a2; e = new_e; SSE = new_SSE;
```

```

% CALCULATE NEW DERIVATIVES

d2 = deltaln(a2,e);
d1 = deltatan(a1,d2,w2c);
end

% PLOTTING
% tr(i+1) = SSE;
% if rem(i,df) == 0
%   fprintf(message,i,lr,SSE)
%   delete(h);
%   h = ploterr(tr(1:i+1),eg);
% end
end

% PLOTTING
%if rem(i,df) ~= 0
%   fprintf(message,i,lr,SSE)
%   ploterr(tr(1:i+1),eg);
%   str = sprintf('TRAINELM: %g/%g, me =%g/%g, lr = %g/%g, epochs, SSE = %g.\n',me);
%   fprintf(str,i,me,lr,SSE);
%end

%fprintf('- OK\n');
%-----
%-----Fim do Treino-----
%-----
end;

save cont_t FcRef b1c d fsoma w1c y b2c fs w2c

```

```
errmea(k)=mean (erro);  
errsqr(k)=sumsqr(erro);  
plot(errsqr(1:k));  
drawnow;  
end;  
Fc_1=Fc(j);  
plot(mrr./14,':')  
hold on  
plot(mrre./14,'--')  
plot(F./S,'-')  
plot(erro./14)
```

7.11 Seguem-se vários *Scripts* usados para criarem os gráficos que aparecem nesta tese

```
x=1:50;  
whitebg;  
clf reset;  
set(gcf,'Units','pixels','backingstore','off');  
set(gcf,'position',[20,20,600,380],'NumberTitle','off');  
  
p1=2+14/50.*x;  
p2=16-12/50.*x;  
p=[p1 p2];  
F=430;  
S=6000;  
fc=150;  
Fc=fc.*F/S.*p;
```

```
%plot(Fc)
hold on
fs=172./(fc.*p);
Feed=fs.*S;
plot(Feed);grid
y=1:100;
t=1./Feed.*60;
a=sum(t)
xlabel('Coordenada X da trajetória em (mm)');
ylabel('Velocidade de avanço em mm/min');
title ('Variação da Velocidade de Avanço quando Fc=const.');
```

```
x=1:1:200;
whitebg;
y=100+4.^(x./40);
plot(y);
xlabel('Tempo de maquinação em (horas)');
ylabel('Força de corte em (Kgf)');
title ('Variação da Força de Corte ao Longo do Tempo');
```

```
x=1:1:95;
whitebg;
S=6000;
F=430;
a=430/6000;
y=20.*ones(size(x));
z=90.*ones(size(x));
p=x;
```

```
p(1:10)=0*ones(1,10);
p(11:60)=1+1/15*x(11:60);
p(61:80)=1+1/15.*x(60).*ones(1,20);
p(81:95)=8.*ones(1,15);

Sec=p*a;
Fc=150*Sec;
subplot(211),plot(Fc);
[a,b]=max(Fc);
xlabel('Coordenada X da trajetória em (mm)');
ylabel('Força de corte em (Kgf)');
title ('Força de Corte Ao Longo da Trajetória');

x=1:1:200;
whitebg;
y=100+4.^(x./40);
subplot(212),plot(y);
xlabel('Tempo de maquinação em (horas)');
ylabel('Força de corte em (Kgf)');
title ('Variação da Força de Corte ao Longo do Tempo');

clear all;
x=0:pi/180:2*pi;
whitebg;
S=6000;
F=430;
a=430/6000;
y(1:180)=x(1:180);
y(181:361)=0.*ones(1,181);
```

```

size(y)
p=8.*F/S.*(y./(2*pi)).*sin(y);
size(p)
Fc=180*p;
subplot(211),plot(Fc);
ylabel('Força de corte em (Kgf)');
title ('Força de Corte Ao Longo da Revolução 1 Dente');

y(1,1:180)=x(1:180);
y(1,181:361)=0.*ones(1,181);
y(2,61:240)=x(1:180);
y(2,241:361)=0.*ones(1,121);
y(2,1:60)=0.*ones(1,60);
y(3,121:300)=x(1:180);
y(3,301:361)=0.*ones(1,61);
y(3,1:120)=0.*ones(1,120);
y(4,181:361)=x(1:181);
y(4,1:180)=0.*ones(1,180);
y(5,241:361)=x(1:121);
y(5,1:60)=x(121:180);
y(5,181:240)=0.*ones(1,60);
y(6,301:361)=x(1:61);
y(6,1:120)=x(61:180);
y(6,121:300)=0.*ones(1,180);
p=y;
p(1:6,:)=8.*F/S.*(y(1:6,:)./(2*pi)).*sin(y(1:6,:))./6.*(sin(y(1:6,:))>0);

Fc=180*p;
for j=1:6,
hold on
subplot(212),plot(Fc(j,:));

```

```
end
fct=sum(Fc);
plot(fct);
[a,b]=max(Fc);
xlabel('Angulo de rotação (°)');
ylabel('Força de corte em (Kgf)');
title ('Força de Corte Ao Longo da Revolução 6 Dentes');
```

```
x=0:pi/180:pi;
whitebg;
for l=.005:.01:.1
y =l.*(x./(2*pi)).*sin(x);
hold on
plot(y);
[a,b]=max(y);
t=num2str(l*6000);
text(b,a+.002,t);
end
text(b+18,a+.002,'mm/min');
xlabel('Angulo de Rotação em graus');
ylabel('Espessura da Apara em mm');
title ('Espessura da Apara para S=6000 RPM');
```

```
x=0:pi/180:pi;
whitebg;
for l=.005:.01:.1
y =l.*(x./pi).*sin(x)./4;
hold on
```

```

plot(y);axis([-1 180 0 6e-2]);
[a,b]=max(y);
a=a*2;
t=num2str(1*6000);
text(150,a+.02,t);
end
text(150+18,a+.02,'mm/min');
xlabel('Angulo de Rotação em graus');
ylabel('Espessura da Apra em mm');
title ('Esp. da Apra - S=6000 RPM fresa - 4 dentes');

```

7.12 Script que permitiu armazenar os dados para treino da rede

```

y1=[764 82;
    1019 41;
    637 93;
    849 48;
    478 89;
    637 46;
    382 92;
    510 46;
    637 187;
    849 96;
    478 166;
    637 100;];
i=y1(:,2)./y1(:,1);
u1=[65 10 1 0 2 1 0 0;
    65 10 1 0 2 1 0 0;
    65 12 1 0 2 1 0 0;

```

```
        65 12 1 0 2 1 0 0;
        65 16 1 0 2 1 0 0;
        65 16 1 0 2 1 0 0;
    65 20 1 0 2 1 0 0;
        65 20 1 0 2 1 0 0;
        65 25 1 0 2 1 0 0;
        65 25 1 0 2 1 0 0;
        65 32 1 0 2 1 0 0;
        65 32 1 0 2 1 0 0;
    ];

    u1=[u1 i];

    y2=[573 64;
        796 31;
        478 73;
        663 36;
        358 70;
        498 34;
        287 72;
        398 34;
        478 146;
        663 72;
        358 129;
        498 75;];
    i=y2(:,2)./y2(:,1);
    u2=[90 10 1 0 2 1 0 0;
        90 10 1 0 2 1 0 0;
        90 12 1 0 2 1 0 0;
        90 12 1 0 2 1 0 0;
        90 16 1 0 2 1 0 0;
        90 16 1 0 2 1 0 0;
        90 20 1 0 2 1 0 0;
```

```

90 20 1 0 2 1 0 0;
90 25 1 0 2 1 0 0;
90 25 1 0 2 1 0 0;
90 32 1 0 2 1 0 0;
90 32 1 0 2 1 0 0;
];

```

```
u2=[u2 i];
```

```
y3=[382 41;
```

```
510 21;
```

```
318 47;
```

```
425 24;
```

```
239 45;
```

```
318 23;
```

```
191 46;
```

```
255 23;
```

```
318 93;
```

```
425 48;
```

```
239 83;
```

```
318 50;];
```

```
i=y3(:,2)./y3(:,1);
```

```
u3=[120 10 1 0 2 1 0 0;
```

```
120 10 1 0 2 1 0 0;
```

```
120 12 1 0 2 1 0 0;
```

```
120 12 1 0 2 1 0 0;
```

```
120 16 1 0 2 1 0 0;
```

```
120 16 1 0 2 1 0 0;
```

```
120 20 1 0 2 1 0 0;
```

```
120 20 1 0 2 1 0 0;
```

```
120 25 1 0 2 1 0 0;
```

```
120 25 1 0 2 1 0 0;
```

```
120 32 1 0 2 1 0 0;
```

```
120 32 1 0 2 1 0 0;  
];  
u3=[u3 i];  
  
y4=[223 31;  
382 12;  
186 35;  
318 14;  
139 33;  
239 13;  
111 34;  
191 13;  
186 70;  
318 28;  
139 62;  
239 29;];  
i=y4(:,2)./y4(:,1);  
u4=[90 10 1 0 2 1 0 0;  
90 10 1 0 2 1 0 0;  
90 12 1 0 2 1 0 0;  
90 12 1 0 2 1 0 0;  
90 16 1 0 2 1 0 0;  
90 16 1 0 2 1 0 0;  
90 20 1 0 2 1 0 0;  
90 20 1 0 2 1 0 0;  
90 25 1 0 2 1 0 0;  
90 25 1 0 2 1 0 0;  
90 32 1 0 2 1 0 0;  
90 32 1 0 2 1 0 0;  
];  
u4=[u4 i];
```

```

y5=[382 41;
    510 21;
    318 47;
    425 24;
    239 45;
    318 23;
    191 46;
    255 23;
    318 93;
    425 48;
    239 83;
    318 50;];
i=y5(:,2)./y5(:,1);
u5=[200 10 1 0 2 1 0 0;
    200 10 1 0 2 1 0 0;
        200 12 1 0 2 1 0 0;
        200 12 1 0 2 1 0 0;
        200 16 1 0 2 1 0 0;
        200 16 1 0 2 1 0 0;
    200 20 1 0 2 1 0 0;
        200 20 1 0 2 1 0 0;
        200 25 1 0 2 1 0 0;
        200 25 1 0 2 1 0 0;
        200 32 1 0 2 1 0 0;
        200 32 1 0 2 1 0 0;
    ];
u5=[u5 i];

y6=[2548 318;
    3185 191;
    2123 372;
    2654 191;

```

```
1592 358;
1990 191;
1274 350;
1592 191;
2123 743;
2654 414;
1592 796;
1990 430;];
i=y6(:,2)./y6(:,1);
u6=[20 10 1 0 2 1 0 0;
    20 10 1 0 2 1 0 0;
        20 12 1 0 2 1 0 0;
        20 12 1 0 2 1 0 0;
        20 16 1 0 2 1 0 0;
        20 16 1 0 2 1 0 0;
    20 20 1 0 2 1 0 0;
        20 20 1 0 2 1 0 0;
        20 25 1 0 2 1 0 0;
        20 25 1 0 2 1 0 0;
        20 32 1 0 2 1 0 0;
        20 32 1 0 2 1 0 0;
    ];
u6=[u6 i];

y7=[1720 183;
    2293 93;
    1433 210;
    1911 107;
    1075 201;
    1433 103;
    860 206;
    1146 103;
```

```

1433 420;
1911 215;
1075 373;
1433 226;];
i=y7(:,2)./y7(:,1);
u7=[65 10 0 1 2 1 0 0;
    65 10 0 1 2 1 0 0;
    65 12 0 1 2 1 0 0;
    65 12 0 1 2 1 0 0;
    65 16 0 1 2 1 0 0;
    65 16 0 1 2 1 0 0;
    65 20 0 1 2 1 0 0;
    65 20 0 1 2 1 0 0;
    65 25 0 1 2 1 0 0;
    65 25 0 1 2 1 0 0;
    65 32 0 1 2 1 0 0;
    65 32 0 1 2 1 0 0;
];

```

```
u7=[u7 i];
```

```

y8=[1306 143;
    1783 71 ;
    1088 163;
    1486 82;
    816 156;
    1115 78 ;
    653 161;
    892 78 ;
    1088 327 ;
    1486 163;
    816 290 ;
    1115 171;];

```

```
i=y8(:,2)./y8(:,1);
u8=[90 10 0 1 2 1 0 0;
    90 10 0 1 2 1 0 0;
    90 12 0 1 2 1 0 0;
    90 12 0 1 2 1 0 0;
    90 16 0 1 2 1 0 0;
    90 16 0 1 2 1 0 0;
    90 20 0 1 2 1 0 0;
    90 20 0 1 2 1 0 0;
    90 25 0 1 2 1 0 0;
    90 25 0 1 2 1 0 0;
    90 32 0 1 2 1 0 0;
    90 32 0 1 2 1 0 0;
    ];
u8=[u8 i];

y9=[860 92;
    1146 46;
    171 105;
    955 54;
    537 100;
    717 52;
    430 103;
    573 52;
    717 210;
    955 107;
    537 186;
    717 113;];
i=y9(:,2)./y9(:,1);
u9=[115 10 0 1 2 1 0 0;
    115 10 0 1 2 1 0 0;
    115 12 0 1 2 1 0 0;
```

```

115 12 0 1 2 1 0 0;
115 16 0 1 2 1 0 0;
115 16 0 1 2 1 0 0;
115 20 0 1 2 1 0 0;
115 20 0 1 2 1 0 0;
115 25 0 1 2 1 0 0;
115 25 0 1 2 1 0 0;
115 32 0 1 2 1 0 0;
115 32 0 1 2 1 0 0;
];

```

```
u9=[u9 i];
```

```
y10=[510 69;
```

```
860 28;
```

```
425 79;
```

```
717 32;
```

```
318 75;
```

```
537 31;
```

```
255 77;
```

```
430 31;
```

```
425 158;
```

```
717 64;
```

```
318 140;
```

```
537 67;];
```

```
i=y10(:,2)./y10(:,1);
```

```
u10=[90 10 0 1 2 1 0 0;
```

```
90 10 0 1 2 1 0 0;
```

```
90 12 0 1 2 1 0 0;
```

```
90 12 0 1 2 1 0 0;
```

```
90 16 0 1 2 1 0 0;
```

```
90 16 0 1 2 1 0 0;
```

```
90 20 0 1 2 1 0 0;
```

```
90 20 0 1 2 1 0 0;
90 25 0 1 2 1 0 0;
90 25 0 1 2 1 0 0;
90 32 0 1 2 1 0 0;
90 32 0 1 2 1 0 0;
];
u10=[u10 i];

y11=[860 92;
1146 46;
717 105;
955 54;
537 100;
717 52;
430 103;
573 52;
717 210;
955 107;
537 186;
717 113;];
i=y11(:,2)./y11(:,1);
u11=[200 10 0 1 2 1 0 0;
200 10 0 1 2 1 0 0;
200 12 0 1 2 1 0 0;
200 12 0 1 2 1 0 0;
200 16 0 1 2 1 0 0;
200 16 0 1 2 1 0 0;
200 20 0 1 2 1 0 0;
200 20 0 1 2 1 0 0;
200 25 0 1 2 1 0 0;
200 25 0 1 2 1 0 0;
200 32 0 1 2 1 0 0;
```

```

                200 32 0 1 2 1 0 0;
            ];

    u11=[u11 i];

    y12=[5732 717;
        7166 430;
        4777 836;
        5971 430;
        3583 806;
        4479 430;
        2866 788;
        3583 430;
        4777 1672;
        5971 932;
        3583 1791;
        4479 967;];

    i=y12(:,2)./y12(:,1);
    u12=[20 10 0 1 2 1 0 0;
        20 10 0 1 2 1 0 0;
                20 12 0 1 2 1 0 0;
                20 12 0 1 2 1 0 0;
                20 16 0 1 2 1 0 0;
                20 16 0 1 2 1 0 0;
        20 20 0 1 2 1 0 0;
                20 20 0 1 2 1 0 0;
                20 25 0 1 2 1 0 0;
                20 25 0 1 2 1 0 0;
                20 32 0 1 2 1 0 0;
                20 32 0 1 2 1 0 0;
    ];

    u12=[u12 i];

```

```
y13=[764 82;  
      1019 41;  
      637 93;  
      849 48;  
      478 89;  
      637 46;  
      382 92;  
      510 46;  
      637 187;  
      849 96;  
      478 166;  
      637 100;];  
i=y13(:,2)./y13(:,1);  
u13=[65 10 1 0 3 1 0 0;  
      65 10 1 0 3 1 0 0;  
      65 12 1 0 3 1 0 0;  
      65 12 1 0 3 1 0 0;  
      65 16 1 0 3 1 0 0;  
      65 16 1 0 3 1 0 0;  
      65 20 1 0 3 1 0 0;  
      65 20 1 0 3 1 0 0;  
      65 25 1 0 3 1 0 0;  
      65 25 1 0 3 1 0 0;  
      65 32 1 0 3 1 0 0;  
      65 32 1 0 3 1 0 0;  
      ];  
u13=[u13 i];  
  
y14=[573 64;  
      796 31;  
      478 73;  
      663 36;
```

```

358 70;
498 34;
287 72;
398 34;
478 146;
663 72;
358 129;
498 75;];
i=y14(:,2)./y14(:,1);
u14=[90 10 1 0 3 1 0 0;
90 10 1 0 3 1 0 0;
90 12 1 0 3 1 0 0;
90 12 1 0 3 1 0 0;
90 16 1 0 3 1 0 0;
90 16 1 0 3 1 0 0;
90 20 1 0 3 1 0 0;
90 20 1 0 3 1 0 0;
90 25 1 0 3 1 0 0;
90 25 1 0 3 1 0 0;
90 32 1 0 3 1 0 0;
90 32 1 0 3 1 0 0;
];
u14=[u14 i];

y15=[382 41;
510 21;
318 47;
425 24;
239 45;
318 23;
191 46;
255 23;

```

```
318 93;
425 48;
239 83;
318 50;];
i=y15(:,2)./y15(:,1);
u15=[115 10 1 0 3 1 0 0;
     115 10 1 0 3 1 0 0;
           115 12 1 0 3 1 0 0;
           115 12 1 0 3 1 0 0;
           115 16 1 0 3 1 0 0;
           115 16 1 0 3 1 0 0;
115 20 1 0 3 1 0 0;
           115 20 1 0 3 1 0 0;
           115 25 1 0 3 1 0 0;
           115 25 1 0 3 1 0 0;
           115 32 1 0 3 1 0 0;
           115 32 1 0 3 1 0 0;
];
u15=[u15 i];

y16=[223 31;
     382 12;
     186 35;
     318 14;
     139 33;
     239 13;
     111 34;
     191 13;
     186 70;
     318 28;
     139 62;
     239 29;];
```

```

i=y16(:,2)./y16(:,1);
u16=[90 10 1 0 3 1 0 0;
     90 10 1 0 3 1 0 0;
           90 12 1 0 3 1 0 0;
           90 12 1 0 3 1 0 0;
           90 16 1 0 3 1 0 0;
           90 16 1 0 3 1 0 0;
     90 20 1 0 3 1 0 0;
           90 20 1 0 3 1 0 0;
           90 25 1 0 3 1 0 0;
           90 25 1 0 3 1 0 0;
           90 32 1 0 3 1 0 0;
           90 32 1 0 3 1 0 0;
];

```

```
u16=[u16 i];
```

```
y17=[382 41;
```

```
510 21;
```

```
318 47;
```

```
425 24;
```

```
239 45;
```

```
318 23;
```

```
191 46;
```

```
255 23;
```

```
318 93;
```

```
425 48;
```

```
239 83;
```

```
318 50;];
```

```
i=y17(:,2)./y17(:,1);
```

```
u17=[200 10 1 0 3 1 0 0;
```

```
200 10 1 0 3 1 0 0;
```

```
200 12 1 0 3 1 0 0;
```

```

                200 12 1 0 3 1 0 0;
                200 16 1 0 3 1 0 0;
                200 16 1 0 3 1 0 0;
    200 20 1 0 3 1 0 0;
                200 20 1 0 3 1 0 0;
                200 25 1 0 3 1 0 0;
                200 25 1 0 3 1 0 0;
                200 32 1 0 3 1 0 0;
                200 32 1 0 3 1 0 0;
    ];
    u17=[u17 i];

    y18=[2548 318;
        3185 191;
        2123 372;
        2654 191;
        1592 358;
        1990 191;
        1274 350;
        1592 191;
        2123 743;
        2654 414;
        1592 796;
        1990 430;];
    i=y18(:,2)./y18(:,1);
    u18=[20 10 1 0 3 1 0 0;
        20 10 1 0 3 1 0 0;
                20 12 1 0 3 1 0 0;
                20 12 1 0 3 1 0 0;
                20 16 1 0 3 1 0 0;
                20 16 1 0 3 1 0 0;
        20 20 1 0 3 1 0 0;

```

```

                20 20 1 0 3 1 0 0;
                20 25 1 0 3 1 0 0;
                20 25 1 0 3 1 0 0;
                20 32 1 0 3 1 0 0;
                20 32 1 0 3 1 0 0;
            ];
    u18=[u18 i];

    y19=[1720 183;
        2293 93;
        1433 210;
        1911 107;
        1075 201;
        1433 103;
        860 206;
        1146 103;
        1433 420;
        1911 215;
        1075 373;
        1433 226;];
    i=y19(:,2)./y19(:,1);
    u19=[65 10 0 1 3 1 0 0;
        65 10 0 1 3 1 0 0;
                65 12 0 1 3 1 0 0;
                65 12 0 1 3 1 0 0;
                65 16 0 1 3 1 0 0;
                65 16 0 1 3 1 0 0;
        65 20 0 1 3 1 0 0;
                65 20 0 1 3 1 0 0;
                65 25 0 1 3 1 0 0;
                65 25 0 1 3 1 0 0;
                65 32 0 1 3 1 0 0;
    ];

```

```
        65 32 0 1 3 1 0 0;  
    ];  
    u19=[u19 i];  
  
    y20=[1306 143;  
        1783 71 ;  
        1088 163;  
        1486 82;  
        816 156;  
        1115 78 ;  
        653 161;  
        892 78 ;  
        1088 327 ;  
        1486 163;  
        816 290 ;  
        1115 171;];  
    i=y20(:,2)./y20(:,1);  
    u20=[90 10 0 1 3 1 0 0;  
        90 10 0 1 3 1 0 0;  
        90 12 0 1 3 1 0 0;  
        90 12 0 1 3 1 0 0;  
        90 16 0 1 3 1 0 0;  
        90 16 0 1 3 1 0 0;  
        90 20 0 1 3 1 0 0;  
        90 20 0 1 3 1 0 0;  
        90 25 0 1 3 1 0 0;  
        90 25 0 1 3 1 0 0;  
        90 32 0 1 3 1 0 0;  
        90 32 0 1 3 1 0 0;  
    ];  
    u20=[u20 i];
```

```

y21=[860 92;
      1146 46;
      717 105;
      955 54;
      537 100;
      717 52;
      430 103;
      573 52;
      717 210;
      955 107;
      537 186;
      717 113;];
i=y21(:,2)./y21(:,1);
u21=[115 10 0 1 3 1 0 0;
      115 10 0 1 3 1 0 0;
      115 12 0 1 3 1 0 0;
      115 12 0 1 3 1 0 0;
      115 16 0 1 3 1 0 0;
      115 16 0 1 3 1 0 0;
      115 20 0 1 3 1 0 0;
      115 20 0 1 3 1 0 0;
      115 25 0 1 3 1 0 0;
      115 25 0 1 3 1 0 0;
      115 32 0 1 3 1 0 0;
      115 32 0 1 3 1 0 0;
      ];

```

```

u21=[u21 i];

```

```

y22=[510 69;
      860 28;
      425 79;
      717 32;

```

```
318 75;
537 31;
255 77;
430 31;
425 158;
717 64;
318 140;
537 67;];
i=y22(:,2)./y22(:,1);
u22=[90 10 0 1 3 1 0 0;
      90 10 0 1 3 1 0 0;
      90 12 0 1 3 1 0 0;
      90 12 0 1 3 1 0 0;
      90 16 0 1 3 1 0 0;
      90 16 0 1 3 1 0 0;
      90 20 0 1 3 1 0 0;
      90 20 0 1 3 1 0 0;
      90 25 0 1 3 1 0 0;
      90 25 0 1 3 1 0 0;
      90 32 0 1 3 1 0 0;
      90 32 0 1 3 1 0 0;
      ];
u22=[u22 i];

y23=[860 92;
      1146 46;
      717 105;
      955 54;
      537 100;
      717 52;
      430 103;
      573 52;
```

```
717 210;
955 107;
537 186;
717 113;];
i=y23(:,2)./y23(:,1);
u23=[200 10 0 1 3 1 0 0;
      200 10 0 1 3 1 0 0;
      200 12 0 1 3 1 0 0;
      200 12 0 1 3 1 0 0;
      200 16 0 1 3 1 0 0;
      200 16 0 1 3 1 0 0;
      200 20 0 1 3 1 0 0;
      200 20 0 1 3 1 0 0;
      200 25 0 1 3 1 0 0;
      200 25 0 1 3 1 0 0;
      200 32 0 1 3 1 0 0;
      200 32 0 1 3 1 0 0;
      ];
u23=[u23 i];

y24=[5732 717;
      7166 430;
      4777 836;
      5971 430;
      3583 806;
      4479 430;
      2866 788;
      3583 430;
      4777 1672;
      5971 932;
      3583 1791;
      4479 967;
```

```
];  
i=y24(:,2)./y24(:,1);  
u24=[20 10 0 1 3 1 0 0;  
      20 10 0 1 3 1 0 0;  
      20 12 0 1 3 1 0 0;  
      20 12 0 1 3 1 0 0;  
      20 16 0 1 3 1 0 0;  
      20 16 0 1 3 1 0 0;  
      20 20 0 1 3 1 0 0;  
      20 20 0 1 3 1 0 0;  
      20 25 0 1 3 1 0 0;  
      20 25 0 1 3 1 0 0;  
      20 32 0 1 3 1 0 0;  
      20 32 0 1 3 1 0 0;  
];  
u24=[u24 i];
```

```
y25=[1720 459;  
      2293 275;  
      1433 573;  
      1911 344;  
      1075 573;  
      1433 365;  
      860 596;  
      1146 378;  
      1433 1051;  
      1911 645;  
      1705 967;  
      1433 618;  
];
```

```

i=y25(:,2)./y25(:,1);
u25=[65 10 1 0 4 1 0 0;
      65 10 1 0 4 1 0 0;
      65 12 1 0 4 1 0 0;
      65 12 1 0 4 1 0 0;
      65 16 1 0 4 1 0 0;
      65 16 1 0 4 1 0 0;
      65 20 1 0 4 1 0 0;
      65 20 1 0 4 1 0 0;
      65 25 1 0 4 1 0 0;
      65 25 1 0 4 1 0 0;
      65 32 1 0 4 1 0 0;
      65 32 1 0 4 1 0 0;
];
u25=[u25 i];

```

```

y26=[5732 1720;
      7166 1146;
      4777 2150;
      5971 1529;
      3583 2329;
      4479 1433;
      2866 2293;
      3583 1490;
      4777 3881;
      5971 2627;
      3583 3695;
      4479 2508;];
i=y26(:,2)./y26(:,1);
u26=[20 10 1 0 4 1 0 0;
      20 10 1 0 4 1 0 0;
      20 12 1 0 4 1 0 0;

```

```
20 12 1 0 4 1 0 0;
20 16 1 0 4 1 0 0;
20 16 1 0 4 1 0 0;
20 20 1 0 4 1 0 0;
20 20 1 0 4 1 0 0;
20 25 1 0 4 1 0 0;
20 25 1 0 4 1 0 0;
20 32 1 0 4 1 0 0;
20 32 1 0 4 1 0 0;
];
u26=[u26 i];

y27=[764 122;
1019 61;
637 153;
849 64;
478 140;
637 67;
382 143;
510 69;
637 306;
849 140;
478 303;
637 131;];
i=y27(:,2)./y27(:,1);
u27=[65 10 1 0 2 0 1 0;
65 10 1 0 4 0 1 0;
65 12 1 0 2 0 1 0;
65 12 1 0 2 0 1 0;
65 16 1 0 4 0 1 0;
65 16 1 0 2 0 1 0;
65 20 1 0 4 0 1 0;
```

```

        65 20 1 0 2 0 1 0;
        65 25 1 0 4 0 1 0;
        65 25 1 0 4 0 1 0;
        65 32 1 0 2 0 1 0;
        65 32 1 0 2 0 1 0;
    ];

```

```

u27=[u27 i];

```

```

y28=[2548 573;

```

```

    3185 265;

```

```

    2123 690;

```

```

    2654 340;

```

```

    1592 677;

```

```

    1990 350;

```

```

    1274 732;

```

```

    1592 382;

```

```

    2123 1645;

```

```

    1654 764;

```

```

    1592 1642;

```

```

    1990 796;];

```

```

i=y28(:,2)./y28(:,1);

```

```

u28=[20 10 1 0 4 0 1 0;

```

```

    20 10 1 0 4 0 1 0;

```

```

        20 12 1 0 4 0 1 0;

```

```

        20 12 1 0 3 0 1 0;

```

```

        20 16 1 0 2 0 1 0;

```

```

        20 16 1 0 2 0 1 0;

```

```

    20 20 1 0 4 0 1 0;

```

```

        20 20 1 0 2 0 1 0;

```

```

        20 25 1 0 4 0 1 0;

```

```

        20 25 1 0 4 0 1 0;

```

```

        20 32 1 0 2 0 1 0;

```

```

                20 32 1 0 2 0 1 0;
            ];

u28=[u28 i];

y29=[1720 275;
      2293 138;
      1433 344;
      1911 143;
      1075 315;
      1433 150;
      860 321;
      1146 155;
      1433 688;
      1911 315;
      1075 681;
      1433 296;];

i=y29(:,2)./y29(:,1);
u29=[65 10 0 1 2 0 1 0;
      65 10 0 1 2 0 1 0;
           65 12 0 1 2 0 1 0;
           65 12 0 1 4 0 1 0;
           65 16 0 1 4 0 1 0;
           65 16 0 1 2 0 1 0;

      65 20 0 1 4 0 1 0;
           65 20 0 1 4 0 1 0;
           65 25 0 1 4 0 1 0;
           65 25 0 1 2 0 1 0;
           65 32 0 1 2 0 1 0;
           65 32 0 1 4 0 1 0;
      ];

u29=[u29 i];

```

```

y30=[5732 1290;
      7166 596;
      4777 1553;
      5971 764;
      3583 1523;
      4479 788;
      2866 1648;
      3583 860;
      4777 3702;
      5971 1720;
      3583 3695;
      4479 1791;];
i=y30(:,2)./y30(:,1);
u30=[20 10 0 1 2 0 1 0 ;
      20 10 0 1 4 0 1 0 ;
      20 12 0 1 4 0 1 0 ;
      20 12 0 1 4 0 1 0 ;
      20 16 0 1 2 0 1 0 ;
      20 16 0 1 4 0 1 0 ;
      20 20 0 1 4 0 1 0 ;
      20 20 0 1 2 0 1 0 ;
      20 25 0 1 2 0 1 0 ;
      20 25 0 1 2 0 1 0 ;
      20 32 0 1 4 0 1 0 ;
      20 32 0 1 4 0 1 0 ;
      ];

```

```
u30=[u30 i];
```

```

u=[u1; u2; u3; u4; u5; u6;
   u7; u8; u9; u10; u11; u12;
   u13; u14; u15; u16; u17; u18;
   u19; u20; u21; u22; u23; u24;

```

```
u25; u26; u27; u28; u29; u30;  
]';  
y=[y1; y2; y3; y4; y5; y6;  
y7; y8; y9; y10; y11; y12;  
y13; y14; y15; y16; y17; y18;  
y19; y20; y21; y22; y23; y24;  
y25; y26; y27; y28; y29; y30;  
]';
```

```
t=u(9,:);  
[t1,I]=sort(t);  
inu=u(:,I);  
outy=y(:,I);
```

```
save adapt_i
```


8. Exemplo de algumas tabelas usadas

8.1 Fresa de 4 navalhas



FRESA 4 NAVALHAS HSCo

Vc (m/min)		24			32			18			25			80			100				
G.Mat. Desig. Nº DIN		1 AÇO CONSTRUÇÃO 60/70 Kgf/mm² 1.1191 / 1.1730			2 AÇO C. TRATADO 60/100 Kgf/mm² 1.7225			6 LIGAS LEVES Al/Cu/CuZn													
Diâmetro mm	FIG.	RPM			AVANÇO			RPM			AVANÇO			RPM			AVANÇO				
		mm/min mm/Rev			mm/min mm/Rev			mm/min mm/Rev			mm/min mm/Rev			mm/min mm/Rev							
2	1	3822	61	0.012	2866	48	0.012	12739	255	0.016	2	2	5096	31	0.008	3981	23	0.008	15924	102	0.008
	2																				
4	1	1911	71	0.028	1433	56	0.028	6369	287	0.036	4	2	2548	46	0.024	1990	34	0.024	7962	178	0.028
	2																				
6	1	1274	109	0.064	955	85	0.064	4246	403	0.076	6	2	1609	66	0.052	1327	50	0.052	5308	255	0.060
	2																				
8	1	955	178	0.140	717	139	0.140	3185	621	0.156	8	2	1274	115	0.120	995	86	0.120	3981	446	0.140
	2																				
10	1	764	204	0.200	573	159	0.200	2548	764	0.240	10	2	1019	122	0.160	796	92	0.160	3185	510	0.200
	2																				
12	1	637	255	0.300	478	199	0.300	2123	955	0.360	12	2	849	153	0.240	663	115	0.240	2654	679	0.320
	2																				
16	1	478	255	0.400	358	199	0.400	1592	1035	0.520	16	2	637	162	0.340	498	122	0.340	1990	637	0.400
	2																				
20	1	382	265	0.520	287	207	0.520	1274	1019	0.640	20	2	510	168	0.440	398	126	0.440	1592	662	0.520
	2																				
25	1	637	467	0.550	478	365	0.550	2123	1725	0.650	25	2	849	287	0.450	663	215	0.450	2654	1168	0.550
	2																				
32	1	478	430	0.675	358	336	0.675	1592	1642	0.825	32	2	637	275	0.575	498	206	0.575	1990	1115	0.700
	2																				

FACTORES
CORREÇÃO
AVANÇO

- 1.2 Para fresas curtas.
- 0.7 Para fresas médias.
- 0.5 Para fresas longas.

FACTORES
CORREÇÃO
Vc e AVANÇO

- 0.9 a 1.1 Dependendo da carga de rotura material.
- 0.9 Deficiente fixação.
- 0.05 a 1.1 Dependendo da qualidade da refrigeração.

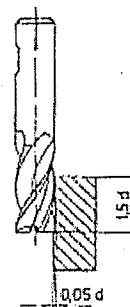


Fig 1

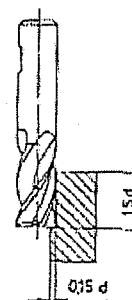


Fig 2



8.2 Fresa de 2/3 navalhas



FRESA 2/3 NAVALHAS HSCo TiCN

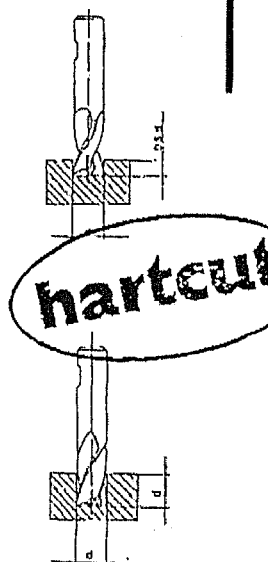
Vc (m/min)	54	72	41	56	180	225	
G.Mat. Desig.	1 AÇO CONSTRUÇÃO 60/70 Kgf/mm² 1.1191 / 1.1730		2 AÇO C. TRATADO 80/100 Kgf/mm² 1.7225		6 LIGAS LEVES Al/Cu/CuIn		
Nº DIN							
Diâmetro mm	FIG.	RPM	AVANÇO mm/min mm/Rev	RPM	AVANÇO mm/min mm/Rev	RPM	AVANÇO mm/min mm/Rev
2	1	8599	92 0.008	6529	71 0.008	28662	358 0.010
	2	11465	52 0.006	8917	39 0.006	35828	258 0.009
4	1	4299	126 0.022	3264	98 0.022	14331	466 0.026
	2	5732	64 0.015	4459	49 0.015	17914	301 0.021
6	1	2866	183 0.048	2176	143 0.048	9554	717 0.060
	2	3822	95 0.033	2972	72 0.033	11943	430 0.045
8	1	2150	166 0.058	1632	129 0.058	7166	627 0.070
	2	2866	84 0.039	2279	64 0.039	8957	430 0.060
10	1	1720	183 0.080	1306	143 0.080	5732	717 0.100
	2	2293	93 0.054	1783	71 0.054	7166	430 0.075
12	1	1433	210 0.110	1088	163 0.110	4777	836 0.140
	2	1911	107 0.075	1486	82 0.075	5971	430 0.090
16	1	1075	201 0.140	816	156 0.140	3583	806 0.180
	2	1433	103 0.096	1115	78 0.096	4479	430 0.120
20	1	860	206 0.180	653	161 0.180	2866	788 0.220
	2	1146	103 0.120	892	78 0.120	3583	430 0.150
25	1	1433	420 0.220	1088	327 0.220	4777	1672 0.280
	2	1911	215 0.150	1486	163 0.150	5971	932 0.195
32	1	1075	373 0.260	816	290 0.260	3583	1791 0.400
	2	1433	226 0.210	1115	171 0.210	4479	967 0.270

FACTORES
CORREÇÃO
AVANÇO

- 1.2 Para fresas curvas.
- 0.7 Para fresas médias.
- 0.5 Para fresas longas.

FACTORES
CORREÇÃO
Vc e AVANÇO

- 0.9 a 1.1 Dep
- 0.9 Deleção
- 0.85 a 1.1 De

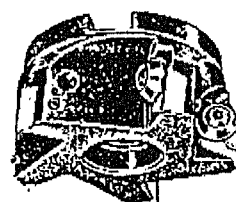


8.3 Roca de Esquadrar



ROCA DE ESQUADRAR - TPUN

Vc (m/min)	86	113	68	90	150	225			
G.Mat. Desig. Nº DIN	1 Aço Construção 60/70 Kgf/mm² 1.1191 / 1.1730		2 Aço C. Tratado 80/100 Kgf/mm² 1.7225		6 Alumínio > 10% Si				
Diametro mm	RPM AVANÇO mm/min mm/Rot			RPM AVANÇO mm/min mm/Rot					
40	687	202	0.23	537	161	0.23	1194	363	0.20
	896	278	0.41	717	218	0.41	1791	403	0.34
50	549	258	0.36	430	206	0.36	955	430	0.30
	717	363	0.66	573	284	0.66	1433	516	0.54
63	436	256	0.45	341	205	0.45	758	409	0.36
	569	327	0.75	455	256	0.75	1137	500	0.66
80	343	252	0.56	269	202	0.56	597	403	0.45
	448	322	0.94	358	252	0.94	896	493	0.83
100	275	242	0.68	215	193	0.68	778	387	0.54
	358	309	1.13	287	242	1.13	717	473	0.99
125	220	193	0.68	172	155	0.68	382	310	0.54
	287	247	1.13	229	193	1.13	513	378	0.99



9. Bibliografia

- [1] Tavares Paulo José da Silva; *Classificadores Ópticos Neurais*; Tese de Mestrado Universidade do Porto; 1996
- [2] Gurney Kevin; *Computers and simbols versus nets and neurons*; Paper Ucl Press Limiteded;
- [3] Buckley J. James & Reilly D. Kevin & Penmetcha Krishnamraju V; *Back Propagation and Genetics Algorithmics for Training Fuzzy Neural Nets*; F. Herrera and J. L. Verdegay, Eds; 1996
- [4] Sarrate-Estruch Ramon, Aguilar Martin Josep, Nicol Charlie; *EAT, a Real-Time Window-Based Event Generation Tool: an Industrial Aplication Example* ; Dept. d'ESAI; 1995
- [5] Hashem Cherif, Keller Paul E., Kouzes Richard T. Kangas Lars J.; *Neural Networks Based Data Analysis for Chemical Sensor Arrays* ; Pacific Northwest Laboratory; 1994
- [6] Technology Brief; *Turbine Engine Diagnostics Using Artificial Neural Networks*; Pacific Northwest Laboratory; 1995
- [7] Technology Brief; *Nuclear Sectral Analisys via Artificial Neural Networks for Waste Handling*; Pacific Northwest Laboratory; 1995
- [8] Tebo Albert; *Artificial Neural Networks: A Developing Science*; OE Reports; 1994
- [9] Zbikowski R., Hunt K.J. Dzielinski A., Murray-Smith R., Gawthrop P.J.; *Revue of Advances in Neural Adaptive Control System*; University of Glasgow, Daimler-Benz; 1994
- [10] Hunt K.J., Irwin G. W., Warwick K.; *Neural Network Engineering in Dynamic Control Systems*; Springer-Verlag; 1995



-
- [11] Rivera Jesus Alberto Delgado; *Input/Output Linearization of Control Affine Systems Using Neural Networks*; A Dissertation Submitted to the Cybernetics Department In Partial Fulfilment of the Requirements for the Degree of doctor of Philosophy; 1996
- [12] G. Horne Bill, Giles C. Lee; *An Experimental Comparision of Recurrent Neural Networks*; NEC Research Institut; 1995
- [13] Madl J.; *Optimization of Cutting Conditions in Machining*; Paper
- [14] M. Kubalcik, V. Bobal ; *Adaptative Pole Placement Controller*; Paper
- [15] J. Kundrák; *Automatic Determination of Cutting Data for Aplication of CBN Tools*; Paper
- [16] Kalmar Péter, Szalay Tibor, Markos Sándor; *Aplication of a Neuro-Fuzzy System in Cutting Process Monitoring*; Paper
- [17] Giles C. Lee; *Using Recurrent Neural Networks To Learn the Structure of Interconnection Networks*; Paper PhD Thesis, Prinseton University; 1993
- [18] Giles C. Lee, B. G. Horn, T. Lin ; *Learning a class of Large Finit State Machines With a Recorrent Neural Network*; Technical Report, University of Maryland; 1994
- [19] Tino Peter, Horn Bill G., Giles C. Lee, C. Colling Wood Pete; *Finit State Machines and Recorrent Neural Networks - Automata and Dynamical Systems Approach*; Technical Repport, University of Maryland; 1995
- [20] Tae - Yong Kim; *Cutting Force Regulation by Using an Adaptive Robust Controller*; Journal of the Korean Society of Precision Engineering; 1995
- [21] Tae - Yong Kim, Deok-ki Choi, Chong-Nan Chu, Jongwon Kim; *Indirect Cutting Force Sensing and Application to Monitoring and Control of Machining Process*; Proceedings of the First World Congress on Intelligence Manufacturing Process & Systems; 1995
- [22] Tae - Yong Kim, Jongwon Kim; *Cutting Force Regulation by Using an Adaptive Servocontroller*; Proceedings of the First S. M. Wu Symposium on Manufacturing Science; 1994

- [23] Tae - Yong Kim, Deok-ki Choi, Chong-Nan Chu, Jongwon Kim; *Indirect Cutting Force Measurement by Using Servodrive Current Sensing and It's application to Monitoring and Control of Machining Process*; Journal of The Korean Society of Precision Engineering; 1996
- [24] Ogata Katsuhiko; *Engenharia de Controlo Moderno*; Prentice/Hall ; 1982
- [25] Kappen Bert, Gielen Stan; *Neural Networks: Artificial Intelligence and Industrial Applications*; Proceedings of the Third Annual SNN Symposium on Neural Networks, Nijmegen The Netherlands; 1995
- [26] William L. Brogan ; *Modern Control Theory*; Prentice-Hall; 1991