LOER NÉRCIDES FRANCO GARCÍA

**SISTEMA HÍBRIDO, MANUAL E AUTOMÁTICO PARA CONTROLE
DE POSIÇÃO E TRAJETÓRIA EM MÁQUINAS DE USINAGEM**

Dissertação apresentada à Escola
Politécnica da Universidade de São Paulo
para obtenção do Título de Mestre em
Engenharia.

São Paulo

2008

LOER NÉRCIDES FRANCO GARCÍA

**SISTEMA HÍBRIDO, MANUAL E AUTOMÁTICO PARA CONTROLE
DE POSIÇÃO E TRAJETÓRIA EM MÁQUINAS DE USINAGEM**

Dissertação apresentada à Escola
Politécnica da Universidade de São Paulo
para obtenção do Título de Mestre em
Engenharia.

Área de Concentração:
Engenharia de Controle e Automação
Mecânica

Orientador:
Prof. Dr. Oswaldo Horikawa

São Paulo

2008

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 4 de janeiro de 2008

Assinatura do autor

Assinatura do orientador

Garcia, Loer Nercides Franco

Sistema híbrido, manual e automático para controle de posição e trajetória em máquinas de usinagem / L.N.F. Garcia. -- ed.ver.-- São Paulo, 2007.

95 p.

Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1.Sistema híbrido de controle 2.Sistema de controle de posição 3.Máquinas de usinagem híbridas 4.Servomecanismos I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos II.t.

À minha amada mãe, Maria Irene, que constantemente me motiva e desde sempre tem se sacrificado pela minha formação.

A meu amado pai, Gonzalo de Jesús, que repousa em melhor vida e presente eternamente em meu coração.

A meu grande Amigo, Guillermo L. Barrera G., que no dia 31 de dezembro de 2005 nos deixa e adormece em campos belos.

AGRADECIMENTOS

A DEUS, por me permitir viver e realizar este outro de seus planos.

À minha amiga Claudia P. Serna G., pela continua motivação e pela ajuda ao facilitar o primeiro contato com meu orientador.

Ao Prof. Dr. Oswaldo Horikawa, pelo aval para minha admissão na Pós-graduação. Pela gestão na consecução dos recursos, tanto para minha manutenção durante a pesquisa, como para o desenvolvimento do projeto. Obrigado pela sua paciência, pela sua dedicação e pela sua valiosa orientação.

À Universidade de São Paulo, pela aceitação e formação recebida.

À CAPES, pelo apoio econômico através da bolsa de estudos.

À FAPESP, pelo financiamento para o desenvolvimento do projeto (Processo No. 05/59963-6. Período de março/2006 a abril/2007).

Aos professores jurados da qualificação, Dr. Marcos Tsuzuki e Dr. Amauri Hassui pelos seus valiosos aportes para alcançar os objetivos do projeto.

Ao Dr. Thiago de Castro Martins, pelas oportunas ajudas e sugestões que permitiram grandes avanços.

Ao meu amigo Freddy A. Franco G., pela motivação e acompanhamento durante meu mestrado.

RESUMO

Com o presente projeto se pretende desenvolver um protótipo de máquina de usinagem para atender a fabricação de pequenos lotes (10 ~ 20 peças), Esta máquina terá características intermediárias entre as máquinas universal (manual) e de controle numérico (CN); conjugando as vantagens funcionais destas duas: alta eficiência e flexibilidade. A máquina projetada terá uma operação manual (trajetória), e seu posicionamento será servo-assistido. Entenda-se “desenvolver” como: a) Propor uma estratégia de usinagem; b) Propor uma configuração de máquina que funcione segundo esta estratégia; c) desenvolver um protótipo e; d) validar a estratégia e a configuração através de testes no protótipo. A estratégia de posicionamento aqui proposta, é baseada em um posicionamento manual, ou seja, o operador define livremente o movimento da ferramenta com relação à peça a se fabricar. Porém, um computador monitora constantemente a posição da ferramenta e não permite a entrada desta numa região, nomeada “Região Proibida”; que corresponde ao perfil da peça. A região proibida é definida diretamente a partir de um desenho de CAD, dispensando o uso de alguma ferramenta computacional de geração automática de trajetória de ferramenta. Deste modo, o operador realiza a usinagem de desbaste. O acabamento é realizado automaticamente pelo computador, que faz com que a ferramenta contorne o perfil da peça, uma vez que o operador envia um sinal de partida. Uma possível configuração de máquina para alcançar esta estratégia é apresentada. E consiste de uma mesa XY servo controlada, adaptada a uma furadeira de bancada, onde operador comanda o movimento da mesa mediante um joystick. Através de ensaios de usinagem, o trabalho demonstra a eficácia da estratégia proposta.

ABSTRACT

Present project aims the development of a machine tool prototype for attend the manufacturing of small lots (10~20 pieces), This machine will have characteristics intermediate between the universal and the numerical control (CN) machines. The desired machine conjugates the functional advantages of both types of machines: high efficiency and flexibility. The machine, thus designed, will have a manual operation (the tool path is defined by the operator), and its positioning will be servo-assisted. Here, "to develop" includes: a) to propose a machining strategy; b) to propose an arrangement of machine that works according to this strategy; c) to develop a prototype and; d) to validate the strategy and the arrangement through tests with the prototype. The positioning strategy, is based on a manual positioning, i.e., the operator defines freely the motion of the tool with respect to the work piece. However, a computer monitores constantly the tool position, not permitting the entrance of it inside the area, here called "prohibited region", corresponding to the profile of the part to be machined. Outside that region, the computer enables operator to move the tool as he wants. The prohibited region is defined directly from a CAD draft, eliminating the necessity of any kind of computer tools for generating automatically the tool path. Thus, the operator executes the rough machining. The finishing path is executed automatically by the computer that makes the tool contour the profile of the part, according to the start-up command given by the operator. In the work, one possible arrangement of machine to achieve this strategy is presented. It consists of servo controlled XY table and adjusted to a bench drill in which, the operator commands the table motion by means a joy-stick. By machining tests, this work shows the efficiency of the proposed strategy.

SUMÁRIO

Página

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SÍMBOLOS

1	INTRODUÇÃO	15
1.1	O tamanho do lote e o tipo de máquina a empregar	15
1.2	Objetivos deste projeto.....	18
2	ESTUDOS PRELIMINARES	19
2.1	Análise de uma usinagem manual	19
2.2	Usinagem manual através do Torno Copiador.	20
3	PROPOSTAS DE ABORDAGEM	22
3.1	Usinagem através de Região Proibida	22
3.2	Alternativa 1 para máquina de usinagem por Região Proibida	22
3.3	Alternativa 2 para máquina de usinagem por Região Proibida	24
3.4	Alternativa 2 expandida para usinagem bidimensional	25
3.5	Vantagem da usinagem através de Região Proibida.....	26
3.6	Comando de movimento e direção em lugar de posição	27
4	REVISÃO BIBLIOGRÁFICA	28
4.1	Trabalhos que abordam a eficiência em máquinas CN isoladamente. 28	
4.2	Trabalhos que abordam a eficiência do sistema produtivo.....	32
4.3	Conclusões desta revisão	34
5	PROTÓTIPO DESENVOLVIDO	35
6	DESENVOLVIMENTO DO ALGORITMO DE CONTROLE.....	38
6.1	Especificação, descrição geral e operação	38
6.2	Pré-processamento	39
6.3	Convenções	41

6.4	Compensação do raio da ferramenta no processo de usinagem	43
6.5	Controle na usinagem de desbaste	47
6.6	Controle na usinagem de acabamento	57
7	TESTES E RESULTADOS	60
7.1	Testes através da verificação da invasão	60
7.2	Testes através dos pontos de intersecção	62
7.3	Discussão dos resultados	71
8	CONCLUSÕES E SUGESTÃO PARA TRABALHOS FUTUROS	73
	REFERÊNCIAS BIBLIOGRÁFICAS	75
	ANEXOS	78
	ANEXO A: Programa Inicial	78
	ANEXO B: Programa "LerDXF"	78
	ANEXO C: Programa "ordenar"	79
	ANEXO D: Programa Inicial de Comunicação "INIC_EXC"	79
	ANEXO E: Programa função "comando"	80
	ANEXO F: Programa "GerapontosRm"	80
	ANEXO G: Função "cross"	87
	ANEXO H: Função "escal"	87
	ANEXO I: Programa Usinagem de desbaste "U_brutam"	87
	ANEXO J: Programa Usinagem de acabamento "U_acabamentom"	94

LISTA DE FIGURAS

	Página
Figura 1 - O tipo de máquina a ser utilizado e o tamanho do lote	16
Figura 2 - Máquina desejada, objeto deste projeto.....	17
Figura 3 - As etapas num posicionamento manual.....	19
Figura 4 - Alteração da velocidade num posicionamento manual.	20
Figura 5 - Esquema de Usinagem com Torno Copiador.	21
Figura 6 - Usinagem através da Região Proibida.	22
Figura 7 - Alternativa 1, Mecanismo de controle unidimensional, operado por manivela.	23
Figura 8 - Proposta de máquina híbrida servo-controlada, operada manualmente.....	24
Figura 9 - A máquina proposta para usinagem no plano.	25
Figura 10 - Uso de uma máquina CN	26
Figura 11 - Uso da máquina proposta	26
Figura 12 - Protótipo da máquina proposta	35
Figura 13 - Configuração dos Módulo	36
Figura 14 - Fotografia da máquina desenvolvida	37
Figura 15 - Exemplo de arquivo DXF, o arquivo texto (a) e o desenho correspondente (b)	39
Figura 16 - Polígono do arquivo “polig-figura.dxf”.....	40
Figura 17 - Coordenadas da ferramenta – Área útil de usinagem.	42
Figura 18 - Convenção para as regiões proibida e permitida.....	43
Figura 19 - Trajetória com pontos muito próximos.....	44
Figura 20 - Vetor V perpendicular ao segmento	44
Figura 21 - Análise para B acima do segmento AC.....	45
Figura 22 - Análise para B abaixo do segmento AC.....	46
Figura 23 - Abordagens de controle para a usinagem de desbaste: (a) por verificação da invasão, (b) por ponto de intersecção	47
Figura 24 - Método Simples.....	48
Figura 25 - Verificação de invasão, critério somatório dos ângulos.....	49
Figura 26 - Polígono com pontos ordenados e intervalos definidos	50
Figura 27 - Fluxograma geral do software para controle através da verificação	

da invasão	51
Figura 28 - Movimentação até o ponto de intersecção.....	52
Figura 29 - Fluxograma geral do software de controle através dos pontos de intersecção	53
Figura 30 - Fluxograma. (a) Ajustar os parâmetros para comunicação “Interfa- ce Specifications”. (b) Zerar o dispositivo “Home Return Operation”...54	
Figura 31 - Fluxograma. Ler arquivos <i>DXF</i> , filtrar e ordenar dados / Gerar re- gião proibida: Matriz de pontos <i>Prp</i>	55
Figura 32 - Fluxograma Usinagem de desbaste.....	56
Figura 33 - Peça após as usinagens (a) de desbaste, (b) de acabamento	58
Figura 34 - Fluxograma Usinagem de acabamento.....	58
Figura 35 - Simulação da usinagem interna e externa em polígono de 7 lados .61	
Figura 36 - Usinagem externa em Triângulo. $R=2,5\text{mm}$	62
Figura 37 - Usinagem interna em Triângulo. $R=2,5\text{mm}$	63
Figura 38 - Triângulo após a usinagem de desbaste interna.....	63
Figura 39 - Usinagem externa em Losango. $R=2,5\text{mm}$	64
Figura 40 - Usinagem interna em Losango. $R=2,5\text{mm}$	64
Figura 41 - Losango após a usinagem de desbaste externa.....	65
Figura 42 - Losango e Triângulo após a usinagem de acabamento.....	65
Figura 43 - Usinagem externa em polígono 7 lados. $R=2,5\text{mm}$	66
Figura 44 - Polígono de 7 lados após a usinagem de desbaste externa	66
Figura 45 - Usinagem interna em polígono 7 lados. $R=2,5\text{mm}$	67
Figura 46 - Polígono de 7 lados após a usinagem de desbaste interna.	67
Figura 47 - Polígono de 7 lados após a usinagem de acabamento.....	68
Figura 48 - Usinagem externa a sigla "USP". $R=2,5\text{mm}$	68
Figura 49 - Sigla "USP" após a usinagem de desbaste.....	69
Figura 50 - Sigla "USP" após a usinagem de acabamento.....	69
Figura 51 - Polígono de 14 lados após a usinagem de desbaste interna.	70
Figura 52 - Polígono de 16 lados após a usinagem desbaste externa.	70
Figura 53 - Polígonos de 14 e 16 lados após a usinagem de acabamento.	71

LISTA DE TABELAS

Página

Tabela 1 - Características Principais dos Módulos	36
Tabela 2 - Resultados da simulação da usinagem: (a) externa em triângulo, (b) interna em losango	61
Tabela 3 - Resultados da simulação da usinagem interna e externa em polígono de 7 lados	61
Tabela 4 - Comparação de medidas	71

LISTA DE ABREVIATURAS E SÍMBOLOS

CNC (ou CN)	Controle Numérico (“Computer Numerical Control”)
PC	Computador Pessoal
CAD	Desenho Assistido pelo Computador (“Computer Aided Design”)
CAM	Manufatura Assistida pelo Computador (“Computer Aided Manufacturing”)
DXF	(“Drawing Interchange Format, or Drawing Exchange Format”)
ASCII	Código Unificado Americano para Intercâmbio de Informação (“American Standard Code for Information Interchange”)
MPG	Gerador de Pulsos Manuais
HSSB	Bus Serial de Alta Velocidade (“High Speed Serial Bus”)
CA	Corrente Alternada
CC	Corrente Contínua
EMST	Parada de Emergência.
dB	decibel, Intensidade de ruído.
bps	Bits por segundo, velocidade de comunicação
RTS	Pedido para emitir (“Request to Send”)
CTS	Desobstruído para emitir (“Clear To Send”)
mm	Milímetros
mm/s	Milímetros por segundo
kgf	Quilograma força
N	Newton, unidade de força
Nm	Newton-metro
W	Watt

µm: Micrômetro

DLL Biblioteca dinamicamente ligada ("Dynamically Linked Library"),

1 INTRODUÇÃO

Ainda hoje, a globalização é um tema atual e persiste uma constante e crescente necessidade de aumento na produtividade e, por conseguinte, um aumento na competitividade nos mercados. Como consequência, persiste a necessidade de desenvolvimento de novas máquinas ou dispositivos de produção adaptáveis às máquinas existentes com o fim de suprir as limitações e desvantagens presentes. Através desta contínua busca por melhorias nos processos de fabricação, foram desenvolvidas diversas máquinas para usinagem que vão desde as convencionais manuais até às mais refinadas máquinas de controle numérico (CN), além das máquinas de uso dedicado.

O desenvolvimento destas tecnologias não tem como objetivo primordial eliminar o emprego destas máquinas do processo produtivo (ou até mesmo do mercado). Em vez disto, é preferível desenvolver máquinas com tecnologia compatível às especificações das peças a fabricar (forma e tamanho de lote principalmente), resultando numa fabricação mais eficiente (refletido no custo).

1.1 O tamanho do lote e o tipo de máquina a empregar

A Figura 1 apresenta um gráfico amplamente conhecido que mostra a relação entre o tamanho do lote e o custo da produção através de três tipos representativos de máquinas de produção: a máquina universal, a máquina de Comando Numérico – CN e a máquina de uso dedicado (ver GROOVER, 2000 [1], por exemplo). De acordo com o tamanho do lote a ser produzido, o gráfico mostra qual o tipo de máquina mais adequada sob o ponto de vista de custo da produção. Os tamanhos de lote apresentados no gráfico, e que indicam a passagem de um tipo de máquina em relação às outras, são meramente ilustrativos servindo somente para se ter uma noção da existência destes limites. Tais limites são variáveis com a evolução tecnológica e o aparecimento de novos tipos de máquinas. Assim, por exemplo, uma nova máquina CN que produza determinado número de peças num menor tempo faria com que o limiar de passagem para a máquina de uso dedicado aumente para valores acima de 100. Por outro lado, uma máquina CN que permite uma

programação ou um preparo mais simplificado reduziria o limite para a máquina universal para um valor inferior a 10.

Mas independente dos números, para lotes grandes (acima de 100 peças, na Fig.1), as máquinas de uso dedicado, se mostram ser as máquinas mais interessantes na produção. Embora o seu custo inicial, assim como o custo necessário para o seu preparo sejam elevados, a sua produtividade é a maior dentre as 3 máquinas, de forma que a sua utilização se torna interessante quanto maior o tamanho do lote. Aqui estão sendo considerados como máquinas de uso dedicado, não somente as linhas transfer, mas todas as variantes de máquinas que em maior ou menor escala dispõem de recursos de automação de modo a elevar a produtividade de um determinado tipo de peça. Para lotes intermediários (10 ~ 100 peças, na Fig.1) a máquina CN é a mais interessante por apresentar um custo inicial e uma produtividade intermediária entre as máquinas universal e a de uso dedicado.

E por final, para lotes reduzidos (inferiores a 10 peças, na Fig.1), a máquina universal, ou seja, a máquina manual, passa a ser interessante já que além do seu custo inicial ser a menor dentre as 3 máquinas, dispensa o programador, uma mão de obra especializada de elevado custo, o trabalho de programação, e todas as atividades necessárias no preparo de uma máquina CN.

Figura 1 - O tipo de máquina a ser utilizado e o tamanho do lote.

Destaque-se ainda que a manufatura através de uma máquina manual é a que apresenta maior versatilidade para, num curto espaço de tempo, adaptar-se a uma nova tarefa, ou seja, flexibilidade.

Este gráfico ilustra um dos motivos do grande sucesso das máquinas CN na manufatura num contexto em que o mercado tende a exigir produtos novos a intervalos cada vez menores de tempo. Portanto, o tamanho dos lotes tende diminuir e a variedade de produtos aumentar. Acredita-se que tal tendência persista e que o tamanho dos lotes irão diminuir ainda mais. Entre outras coisas, isto torna interessante o desenvolvimento de uma máquina com características intermediárias entre uma máquina universal e uma máquina CN, como se observa na Fig. 2. a qual é um detalhamento da Fig. 1 para a região de pequenos lotes.

Se de um lado, deseja-se uma máquina CN com uma programação mais simplificada, deseja-se por outro lado, uma máquina universal com uma produtividade maior.

Há que se ressaltar ainda que uma máquina assim é de grande interesse social por revitalizar a mão de obra humana, oferecendo uma solução para o problema do desemprego decorrente dos elevados níveis de automação que caracterizam as fábricas mais recentes.

Figura 2 - Máquina desejada, objeto deste projeto.

1.2 Objetivos deste projeto

Neste contexto, o presente projeto tem como objetivo propor uma máquina com um nível de automação intermediário e conseqüentemente, de produtividade, entre uma máquina manual e uma máquina CN.

Serão objetivos deste trabalho, desenvolver:

- Uma máquina que não requer programação de código G.
- Uma máquina cuja movimentação é manual e o posicionamento é servo assistido. E desta forma o posicionamento da ferramenta é realizado de forma mais rápida que numa máquina manual contribuindo assim com a agilização da usinagem.

Entenda-se como desenvolver:

- Propor uma estratégia de usinagem;
- Propor uma configuração de máquina que funcione segundo esta estratégia;
- Desenvolver um protótipo;
- Validar a estratégia assim como a configuração através dos ensaios no protótipo;
- Identificar os problemas e as soluções tanto na estratégia quanto na configuração da máquina proposta.

Uma máquina, com as características acima mencionadas, apresentará um desempenho intermediário entre uma máquina manual e uma máquina CN, sendo adequada para a fabricação de lotes especialmente pequenos, a título de ilustração, lotes em torno: de 10 peças. A avaliação rigorosa ao respeito desta adequação requer um estudo permenorizado de custos (custo inicial da máquina, tempo de usinagem, custo de operação etc.), mas isto não será objeto de estudo deste trabalho.

2 ESTUDOS PRELIMINARES

2.1 Análise de uma usinagem manual

Uma simples análise da operação de usinagem mostra uma das etapas em que se despende mais tempo.

A operação de usinagem consiste basicamente na repetição de tarefas de posicionamento da ferramenta numa posição desejada, através da movimentação por manivelas. Ciente do local até onde a ferramenta deve ser conduzida, o operador: (a) numa primeira etapa, movimenta a manivela contabilizando a quantidade deslocada através da escala da manivela, procurando manter a velocidade da ferramenta constante na velocidade ideal de avanço, (b) numa segunda etapa, ao chegar numa posição próxima à posição desejada, passa a movimentar a manivela mais lentamente, de modo a não ultrapassar a posição desejada e (c) numa última etapa, realiza o posicionamento mais minucioso, controlando manualmente a posição até frações de divisões da escala da manivela.

Figura 3 - As etapas num posicionamento manual.

Uma mínima experiência em usinagem mostra claramente que, qualitativamente, a velocidade de movimentação muda de maneira significativa ao longo das etapas acima descritas, conforme ilustra a Fig. 4.

Figura 4 - Alteração da velocidade num posicionamento manual.

A velocidade cai drasticamente nas etapas (b) e (c), o que implica que estas etapas demandam um tempo significativo numa usinagem manual.

É intuitivo que quanto maior a precisão que se exige na usinagem, maior será o tempo despendido na etapa (c) e quanto maior o número de detalhes a serem usinados, maior será a quantidade de etapas do tipo (b) e (c) e menor será a duração relativa das etapas do tipo (a).

Isto vem a sugerir que uma diminuição significativa do tempo de usinagem, ou seja, um aumento da eficiência da usinagem, pode ser atingido aumentando-se a eficiência da usinagem nas fases (b) e (c) acima descritas, ou seja, as etapas de posicionamento relacionadas ao posicionamento de precisão.

2.2 Usinagem manual através do Torno Copiador

Uma antiga solução que se enquadra na proposta deste trabalho são as máquinas (tornos e fresas) copiadoras. Procurando reduzir o tempo em alguns processos de usinagem, principalmente de peças de revolução, e buscando diminuir a responsabilidade do operador em ficar atento para não ultrapassar determinada dimensão, têm-se desenvolvido dispositivos copiadores que são adaptados

principalmente a tornos e, em menor escala, a fresadoras. Os tornos copiadores permitem obter peças a partir da reprodução de uma peça padrão “Limitador”

Figura 5 - Esquema de Usinagem com Torno Copiador.

Um “Apalpador” segue o contorno da peça padrão ao avançar o carro principal e transmite o seu movimento através de um mecanismo hidráulico ou magnético a um carro que realiza movimento independente do fuso pequeno transversal. Geralmente este dispositivo copiador pode ser colocado ou retirado do torno, constituindo-se um dispositivo auxiliar de usinagem que limita mecanicamente os avanços (sistema seguidor). Isto altera drasticamente o que foi ilustrado na Fig. 4, já não existindo mais, nem as etapas de aproximação nem de finalização. O operador pode realizar o avanço rapidamente, numa única velocidade de usinagem. Este trabalho tem como inspiração, estas máquinas copiadoras. Ao passo que nestas máquinas são utilizados máquinas manuais e limitadores mecânicos, propõe-se neste trabalho a utilização de máquinas CN que empregam limitadores não físicos, determinados por software.

3 PROPOSTAS DE ABORDAGEM

3.1 Usinagem através de Região Proibida

Neste trabalho, a estratégia para a agilização da usinagem é através da região proibida (ver Fig. 6), sendo esta região o contorno da peça a ser usinada. A região proibida equivale ao conceito do “Limitador”, utilizado nas máquinas copiadoras. Fora da região proibida, o operador pode movimentar livremente a ferramenta realizando a usinagem de desbaste. Um computador, que possui na memória o perfil da peça a ser usinado, monitora constantemente a posição da ferramenta impedindo a entrada desta dentro da região proibida.

Figura 6 – Usinagem através da Região Proibida.

3.2 Alternativa 1 para máquina de usinagem por Região Proibida

Diversas alternativas podem ser cogitadas para se realizar a usinagem por região proibida. Uma primeira alternativa foi desenvolvida por Horikawa *et al.* (2007) [2]. E consistiu de um sistema de controle de posicionamento híbrido, manual e servo-assistido que tem movimentação só numa direção. A Fig. 7 apresenta um esquema do protótipo desenvolvido com os seus componentes principais. A base é uma guia linear composta por uma mesa e um parafuso de esferas. Num extremo do parafuso é instalada uma manivela e, no outro, um sensor de posição (encoder). Um

Figura 7 – Alternativa 1, Mecanismo de controle unidimensional, operado por manivela.

computador recebe a posição desejada, lê o sinal do encoder, processa o sinal segundo o algoritmo de controle, mostra a posição atual da mesa e gera o sinal para acionar um freio eletromagnético.

Ao início do posicionamento, o operador informa ao computador a posição desejada. Imediatamente, o computador inicia um monitoramento constante da posição da mesa. Até alcançar a posição desejada, o operador movimenta livremente a mesa através da manivela. Na medida em que a mesa se aproxima da posição desejada (Região Proibida), o algoritmo de controle ativa a bobina do freio eletromagnético, fechando gradativamente duas hastes que pressionam umas pastilhas contra o disco da manivela. Quando a distância entre a posição da mesa e a posição desejada é inferior a um valor admissível, a manivela é bloqueada totalmente, liberada só se o operador tenta mover a mesa no sentido contrário. Esta tentativa do operador mover a manivela num sentido ou outro é detectado através de um sensor de intenção. Um protótipo foi construído e através de experimentos de posicionamento, demonstrou-se a validade da estratégia e a possibilidade de realizar posicionamentos rápidos e precisos com a mesa sendo acionada manualmente.

3.3 Alternativa 2 para máquina de usinagem por Região Proibida

Como uma segunda alternativa para a usinagem por região proibida, se propõe um servo-sistema que conjuga o modo de operação de uma máquina CN e uma máquina manual. O servo-sistema é apresentado na Fig. 8.

Figura 8 - Proposta de máquina híbrida servo-controlada, operada manualmente.

A máquina descrita na Fig. 8 funciona da seguinte maneira:

O operador insere a posição de referência no computador e passa a girar a manivela do GMP na direção desejada. A cada intervalo angular fixo que a manivela gira, um pulso é enviado ao servo-motor que, através do fuso, produz um movimento de tamanho fixo (movimento unitário) na mesa. A posição da mesa é monitorada continuamente através de um sensor de rotação acoplado ao eixo do servo-motor e este sinal de posição é enviado ao computador. Enquanto a mesa ainda está distante da posição desejada, o computador permite que os pulsos gerados no GMP passem para o servo motor e convertidos em movimento da mesa. Mas quando a posição da mesa entrar numa faixa de tolerância em torno da posição desejada, o computador para de transmitir os pulsos para o servo motor. Embora o operador continue girando a manivela do GMP, a mesa permanece parada na posição desejada. E assim, o posicionamento é concluído.

Tudo ocorre como se existisse uma barreira além da qual a mesa não pode mais avançar, mesmo que o operador tente. A região além da parede constitui o que está sendo chamado de “região proibida”.

Ao operar numa máquina com esta arquitetura e este funcionamento, a posição desejada é atingida facilmente sem a necessidade de atenção por parte do operador. Ou seja, se eliminam da Fig. 4 as etapas (b) e (c), aumentando assim a velocidade do posicionamento e conseqüentemente da usinagem.

3.4 Alternativa 2 expandida para usinagem bidimensional

Figura 9 - A máquina proposta para usinagem no plano.

Empilhando-se dois sistemas como o da Fig. 8 em direções ortogonais, obtém-se um sistema de posicionamento no plano (ver Fig. 9). O computador passa então a monitorar a posição da mesa no plano e impede agora que a mesa entre numa “região proibida” bidimensional. Esta nova região proibida, nada mais é do que a geometria e as dimensões da peça a ser usinada e que são introduzidas no computador pelo operador.

3.5 Vantagem da usinagem através de Região Proibida

Numa usinagem com máquinas CN, ver Fig. 10, os arquivos contendo dados sobre a geometria e as dimensões (entre outros dados) da peça a ser usinada, fornecidas pelos sistemas CAD – Computer Aided Design, devem passar por uma importante atividade que é a da definição de caminho de corte da ferramenta para assim se chegar ao programa de usinagem em linguagem G. Geralmente esta tarefa é auxiliada por ferramentas computacionais denominadas CAM – Computer Aided Manufacturing.

Figura 10 - Uso de uma máquina CN.

Com o uso da máquina proposta, ver Fig. 11, elimina-se a etapa da definição do caminho de corte da ferramenta e os dados do CAD serão diretamente inseridos no computador de controle da Fig. 9 de modo a definir a região proibida. Fora da região proibida, o operador é livre para movimentar a ferramenta como ele desejar. Um operador visualiza a peça e define através da sua intuição o melhor caminho de corte a ser seguido e o executa manualmente.

Dispensa-se desta forma, o trabalho de preparo de uma máquina CN, especialmente com a sua programação e se realiza um trabalho de forma mais ágil e preciso do que uma usinagem manual.

Figura 11 - Uso da máquina proposta.

O caminho de corte não será otimizado tal qual é, através dos sistemas CAM. Nem a

vida útil da ferramenta será otimizada, assim como é possível nas máquinas CN uma vez que a movimentação é basicamente manual. Mas isto será de pouca relevância pois se estão considerando lotes muito pequenos, quando muito, 10~20 unidades.

3.6 Comando de movimento e direção em lugar de posição

Até aqui, a discussão foi desenvolvida e a proposta da nova máquina feita, assumindo-se que o operador comande o movimento da máquina através de manivelas, simuladas por geradores manuais de pulsos - GMP. Os GMPs, assim como as manivelas de máquinas manuais, atuam como entrada de comando de posição. Contudo, nesta nova máquina, a posição é na realidade, controlada pelo servo sistema. Desta forma, em lugar da manivela comandando a posição, é mais interessante um comando de movimento e direção. E na Fig. 8, em vez do gerador manual de pulsos, poder-se-ia ter um dispositivo para comando de movimento e direção, por exemplo, o joystick utilizado em video-games.

4 REVISÃO BIBLIOGRÁFICA

Uma revisão bibliográfica foi conduzida visando pela identificação de trabalhos com propostas similares aos da “Região Proibida” apresentados neste trabalho. Nenhum trabalho similar e que apresente uma solução para os inconvenientes mencionados no item 2.1 foi encontrado. Porém, foi encontrado um grupo de trabalhos que tratam da Automação Balanceada, com os quais, este trabalho apresenta afinidades quanto aos objetivos finais. A seguir, o cenário de alguns destes trabalhos e das tendências tecnológicas atuais.

4.1 Trabalhos que abordam a eficiência em máquinas CN isoladamente

Atualmente, os trabalhos de pesquisa que procuram melhores desempenhos ao agilizar, otimizar e controlar melhor os processos de fabricação, são liderados por dois frentes de pesquisa. O primeiro está dedicado a obter melhores métodos para a geração da trajetória da ferramenta; o segundo, em métodos para a implementação de controles (de posição ou trajetória) auxiliares, externos às máquinas CN originais (existentes), facilitado em grande parte pelos parâmetros da Arquitetura aberta com os quais vem-se desenvolvendo algumas máquinas CN.

4.1.1 Geração automática da trajetória da ferramenta em máquinas CN

Uma superfície a ser usinada se expressa por uma série de curvas como B-spline, Bézier e NURBS, que compõem a superfície e depois em sistemas CAM as curvas são divididas em um grande número de linhas ou segmentos de arco. Estes comandos divididos de movimentação causam muitos problemas incluindo a dimensão excessiva de dados de CN que torna quase impossível um ajuste local ou modificação da superfície. Existem vários trabalhos propondo estratégias para gerar automaticamente a trajetória da ferramenta e conseqüentemente a parte do programa de CN desde o desenho da peça. Na usinagem de forma livre o software CAM oferece diversos modos de geração para a trajetória da ferramenta, dependendo da geometria da superfície que será usinada, cuja qualidade dependerá da escolha da estratégia de usinagem e dos parâmetros de usinagem. As

estratégias propostas implicam em um trabalho computacional intensivo e ainda nenhuma estratégia computacional universal que atende qualquer tipo de usinagem foi desenvolvida. Dentro destes trabalhos têm-se as propostas como as que seguem.

Yoon (2005) [3], para a geração da trajetória da ferramenta desenvolve um método através da altura de passes constantes (“Iso-Scallop”), para a usinagem de desbaste em três eixos com fresa de ponta redonda. Sendo cada passe subsequente um “offset” do anterior, ao longo da superfície. Apresenta um estudo geométrico e analítico dos aspectos da usinagem, analisando as propriedades locais das funções de distância, que indicam onde o ponto do “scallop” e a ferramenta se centram; usando o algoritmo iterativo de Newton e aproximações de Taylor. O método reduz significativamente o tempo de cálculo necessário para gerar as trajetórias.

Jee e Koo (2003) [4], desenvolveram um eficiente método de geração da trajetória em tempo real através da interpolação de superfícies NURBS. Superfícies que oferecem uma ferramenta muito útil com mais liberdade na geração e modificação da superfície; são fáceis de definir e a alteração das suas formas pode ser localizada em torno de uma região através de pontos de controle ajustados. O método de geração baseia-se numa melhorada estratégia de passes constantes (“Iso-Scallop”) que pode proporcionar uma melhor precisão.

Tam; Xu e Zhou (2002) [5], apresentam uma abordagem para a geração da trajetória “on-line” e a interpolação para a usinagem de superfícies implícitas. Para uma superfície implícita dada, uma vez a direção do plano de corte e dos pontos de corte tem sido escolhidos, trajetórias Iso - planares da ferramenta e pontos interpolados podem ser calculados “on-line”. Isto de acordo com a velocidade de avanço e os requerimentos de altura dos passes constantes. A abordagem permite a determinação correta da posição da ferramenta e sua orientação em cada ponto interpolado.

Jun; Kim e Park (2002) [6], cuja proposta para gerar a trajetória da ferramenta, livre de interferências, e desde um modelo poliédrico baseados em uma curva. Consiste em deslocar (“offset”) o modelo poliédrico, logo, os elementos deslocados são cortados por uma série de planos de movimentação. Os segmentos de curva nos

planos de movimentação são ordenados, aparados e ligados, enquanto a interferência é removida durante o processo de aparamento.

Tsuzuki (1999) [7], quem propõe um método para obter um “bom” acabamento. Mediante uma curva de preenchimento espacial (Hilbert) é determinada a trajetória da ferramenta sobre uma superfície offset calculada com base à ferramenta usada.

Oliveira (1997) [8], implementou um sistema de geração de programas CN para máquinas de 3 eixos, para superfícies de forma livre cujos modelos são criados em um modelador na forma paramétrica Bézier; e gera segmentos de programa CN para usina-las em duas etapas, desbaste e acabamento, com trajetórias tanto em ziguezague, como eqüidistantes ao contorno (offset).

4.1.2 Controles auxiliares de posição, externos para máquinas CN

Os métodos para o controle da posição ou trajetória da ferramenta em máquinas de usinagem estão dirigidas a propor sistemas auxiliares de controle para implementar às máquinas CN existentes. Procurando obter melhorias na precisão e um seguimento em tempo real da trajetória. Devido ao fato de não existirem teorias exatas para predizer os esforços, conjuntamente com a indisponibilidade freqüente da informação do CAD sobre o perfil do mandril, é muito complicado gerar uma parte de programa que se possa carregar numa máquina CN para executar uma tarefa com sucesso. Além disto, as máquinas CN comerciais não têm um controle ativo de eixos (ou seja, não tem a capacidade de responder a entradas de sinais de visão e esforço desde um sensor), operando em um modo de execução de bloco. O qual uma vez que as instruções são carregadas no “buffer” de memória, a execução continua sem se importar com os sinais do sensor, sendo o “Reset” a única forma de parar a execução.

Adicionalmente, com a atenção nos últimos anos pela utilização de sistemas de controle de arquitetura aberta para controlar as máquinas (ROBER; SHIN, 1995 [9] e ONWUBOLU et al, 2002 [10]), estes controles estão sendo implementados ainda mais. Porém, não são muitas as mudanças significativas na trajetória nominal da ferramenta, e a arquitetura de controle que apresentam conserva a funcionalidade

da máquina CN, empregando o diagnóstico e supervisão. Além dos sistemas de acionamento existentes originalmente.

Algumas propostas relacionadas com estes métodos de controle auxiliar externo são apresentadas nos trabalhos a seguir.

Hanafl; Tordon e Katupi (2005) [11] desenvolveram um controlador externo genérico para manter a máquina sob controle em tempo real, o controlador foi interconectado via as entradas do gerador de pulsos manuais (MPG). A máquina é comandada através de um trem de pulsos gerados pelo computador externo, que substituem os pulsos gerados pelo MPGs, mediante os que o operador posiciona manualmente a máquina. Posteriormente, para completar a tarefa, foi implementado um laço de retroalimentação por sensor usando um bus serial de alta velocidade (HSSB), permitindo assim uma intervenção visual, lendo a posição da máquina e alcançando o seguimento ativo do perfil.

Haber e Alique (2004) [12], apresentam uma nova abordagem para o controle da força de corte baseados no princípio do controle do modelo interno (IMC). O objetivo é controlar uma única variável de saída, a força de corte, alterando uma única entrada, a velocidade de avanço. Usaram um modelo neural como modelo interno para determinar a entrada necessária para manter a força de corte constante.

Park; Lim e Choi (2003) [13] desenvolveram uma estratégia de controle para melhorar a precisão da mesa XY de uma máquina CN quando ocorre reversão da velocidade. Devido a que se apresenta um erro pela transição repentina de atrito entre os regimes de pré-resvalamento e resvalamento, um método simples é proposto onde o tempo de transição é estudado para derivar uma relação entre ele e a aceleração a velocidade zero.

Bueno, R. *et al* (2002) [14], desenvolveram um sistema de localização 3D de alta precisão cujo controle está baseado na medição direta da posição 3D da ferramenta através da combinação de um sistema de localização juntamente com um interferômetro que realimentam o controle da posição e permite corrigir a posição da ponta da ferramenta durante o processo de usinagem.

Shawky; Rosenberger e Elbestawi (1998) [15] para compensar mudanças na usinagem conservando a precisão, apresentam o uso de um “Off-set” de posição da ferramenta para ajustar a trajetória comandada.

Rober e Shin (1995) [9] desenvolveram um controlador baseado num PC, com um tratamento levemente diferente conservam os aspectos de monitoramento e segurança do CN, enquanto usam um PC externo para gerar os sinais de atuação.

Al-Kindi; Baul e Gill (1993) [16] propuseram uma solução aos problemas associados com a implicação da parte humana em várias das etapas da fabricação de componentes, mediante a implementação de inspeção baseada em visão computadorizada.

4.2 Trabalhos que abordam a eficiência do sistema produtivo

Já este conjunto de trabalhos aborda não somente uma determinada máquina de produção, como por exemplo uma máquina CN, de forma isolada. Mas sim, discutem e analisam todo o sistema produtivo.

4.2.1 Automação Total

A Automação Total é uma abordagem baseada no uso de máquinas com um alto nível de automação. Sendo desenvolvida através dos conceitos da FMC (Flexible Manufacturing Cell) e do FMS (Flexible Manufacturing System), assuntos descritos, por exemplo, por Zhou (1999) [17] e Tempelmeier e Kuhn (1993) [18]. Ambos os conceitos baseiam-se nas máquinas CN, nos robôs manipuladores e nos dispositivos para manejo de material. Dispositivos altamente automatizados que são integrados e controlados por um computador (DECOTIGNIE; GRÉGOIRE, 1989 [19]). Devido aos altos níveis de automação e integração, os FMC e FMS são capazes de adaptar-se aos requisitos de uma produção cambiante, ver, por exemplo, Groover (2000) [1]; permitindo tempos de entrega curtos, mantendo um baixo inventário.

4.2.2 Sistemas de Produção Ágil

Ainda que o conceito de fabricação ágil foi desenvolvido (GUNASEKARAN, 1998 [20]), o desenho, a implementação e a operação requerem de um vasto conhecimento em planejamento; envolvendo diversas áreas tais como, engenharia, computação, administração de negócios, gerência de operações e ciência de gerenciamento (TEMPELMEIER; KUHN, 1993 [18]). Um trabalho que ilustra a maneira de obter os critérios necessários para o planejamento e controle de sistemas de produção ágil é o trabalho de Hopp; Iravani e Shou (2005) [21]. Critérios que segundo eles são obtidos a través: da caracterização dos benefícios operacionais da automação; da determinação da localização mais conveniente da automação e; da pesquisa da melhor maneira de distribuição dinâmica do trabalho em um processo com equipamento manual e automático.

4.2.3 Automação Balanceada

A idéia da Automação Balanceada foi desenvolvida (CAMARINHA-MATOS; AFSARMANESH, 1995 [22] e RIASCOS; MIYAGI, 2001 [23]) em oposição à tendência da Automação Total. As novas tendências originam grandes desafios, principalmente para as pequenas empresas, que são a grande maioria no mundo. Estas empresas, devido a seus limitados recursos humanos e, sobretudo, financeiros, têm grandes dificuldades para ter acesso e beneficiar-se da tecnologia avançada. Além disto, a herança do passado tem um papel muito importante com relação à fabricação, aos equipamentos, à infra-estrutura, aos processos, à cultura e à habilidade do operador.

A realidade atual mostra, por um lado, o aumento do desemprego nos países desenvolvidos, e pelo outro, a pobre estrutura econômica dos países em desenvolvimento. Isto sugere que para alcançar soluções equilibradas, a automação dos processos produtivos deve-se considerar em um contexto global. A Automação Balanceada procura um nível apropriado de automação em oposição aos delineamentos tanto, puramente antropocêntricos, como da Automação Total.

O presente trabalho, ao propor uma máquina de características intermediárias entre uma máquina universal e uma máquina CN, abre novas possibilidades para a definição de um sistema produtivo que emprega maior volume de mão de obra humana. Portanto, o presente trabalho oferece alternativas para a efetivação das soluções levantadas através da Automação Balanceada.

4.3 Conclusões desta Revisão

Foi apresentado um panorama sobre trabalhos relacionados a máquinas de usinagem e sistemas produtivos. Foi possível encontrar diversos enfoques para o direcionamento dos trabalhos mas não foi encontrado nenhum trabalho que tenha um enfoque similar ao que é apresentado neste trabalho: a proposta duma usinagem por Região Proibida. Contudo o desenvolvimento de uma máquina operada manualmente e no qual, o próprio operador decide a maneira de realizar a usinagem, é consoante com os delineamentos do equilíbrio adequado entre a automação e o uso do trabalho humano, apregoado pela da Automação Balanceada. Aqui não se pretende excluir o operador humano, mas sim, liberá-lo da responsabilidade do controle dimensional da peça para tornar a fabricação mais eficiente.

5 PROTÓTIPO DESENVOLVIDO

Com o objetivo de realizar testes de usinagem para verificar a validade das propostas apresentadas, um protótipo foi desenvolvido em concordância com a proposta da alternativa 2 (item 3.4), e usando comandos de movimento e direção em lugar de comandos de posição. O protótipo, com configuração similar à fresadora, conta para sua movimentação longitudinal e transversal com dois módulos CN comerciais (NSK, Robot modules, ver NSK Ltd, 1997 [24] e [25]), cada um com controle na respectiva direção. Os módulos, aqui nomeados módulo X e módulo Y estão empilhados ortogonalmente como se mostra na Fig. 12. O cabeçote da máquina, isto é, o mandril para a ferramenta e o sistema motriz, foi implementado usando uma furadeira comercial. A movimentação vertical, para a profundidade de corte, realiza-se deslocando o mandril mediante um parafuso fixo ao cabeçote. A estrutura que sustenta todos os elementos foi projetada para que o mandril fique na metade do percurso de ambos os módulos (metade da área efetiva de usinagem 400mm x 350mm). Assim, o protótipo pode realizar um posicionamento bidimensional permitindo um espaço maior para abordar (usinar) por todos os lados à peça fixada e centrada na mesa.

Figura 12 – Protótipo da máquina proposta.

Um controlador, capaz de executar interpolação linear e circular, controla os módulos usando sinais de encoder como sinais de retroalimentação de posição e executando posicionamentos de alta precisão, com uma resolução interna de 2,5µm.

A linguagem de controle é similar ao código G usado nas máquinas CN. Comandos escritos em código ASCII são gerados ou inseridos ao computador e enviados ao controlador através da interface serial (RS232C). O computador recebe comandos de movimentação e direção do operador através de um joystick, de maneira que dependendo do lado para o qual o joystick estiver tombado, se enviarão comandos de movimentação linear ao controlador dos módulos.

Para prover segurança ante uma operação imprevista ou indesejável, sendo uma recomendação para a operação com estes equipamentos, foi implementado um interruptor para a parada de emergência (EMST). As características principais dos módulos são resumidas na Tab. 1 e sua configuração interna mostra-se na Fig. 13. Uma fotografia do protótipo desenvolvido apresenta-se na Fig. 14.

Tabela 1 - Características Principais dos Módulos.

Característica	Módulo X	Módulo Y
Curso (mm)	400	350
Velocidade Max. (mm/s)	1200	1200
Carga Max. (kg)	60	40
Força Axial (N)	300	300
Repetibilidade (mm)	± 0.010	± 0.010
Potência do Motor (W)	300	300
Passo do fuso de esf. (mm)	20	20

Figura 13 - Configuração dos Módulos.

Figura 14 – Fotografia da máquina desenvolvida.

6 DESENVOLVIMENTO DO ALGORITMO DE CONTROLE

Como explicado anteriormente, a idéia proposta neste projeto é usar o conceito da “Região Proibida” para controlar a posição da ferramenta com relação à peça. Neste capítulo são apresentados os passos seguidos para o desenvolvimento do algoritmo que será usado para realizar este controle. Inicialmente especifica-se o alcance, a forma como operará a máquina e os dados iniciais para o controle. Continua-se com o pré-processamento realizado aos dados iniciais, assim como as convenções seguidas para seu manejo. Posteriormente é considerado o raio da ferramenta para definir tanto a região proibida, como a trajetória de usinagem. E, finalmente, se abordam as formas para realizar o controle das usinagens de desbaste e de acabamento; sendo o controle da usinagem de desbaste através da verificação da invasão e através dos pontos de intersecção.

6.1 Especificação, descrição geral e operação

O programa de controle é elaborado de modo que a máquina opere em modo manual ou automático. No modo manual, o operador move a mesa através do joystick, de maneira que a ferramenta move-se na área externa à região proibida, fazendo assim a usinagem de desbaste. Entretanto o algoritmo de controle impede a entrada nesta região (limita a movimentação da mesa). Após a usinagem de desbaste, o modo automático é escolhido. Neste modo, o operador move a ferramenta até tocar a peça, posteriormente, a ferramenta contorna o perfil da peça executando a usinagem de acabamento. O algoritmo de controle é implementado em software Matlab® (Mathworks Inc., USA).

A região proibida é definida apartir do arquivo de desenho da peça fornecido pelo CAD. O programa aqui elaborado, lê este arquivo e extrai a informação necessária para definir a região proibida. Por exemplo, na Fig. 15, o arquivo tipo DXF (formato de arquivo adotado por Autocad®, Autodesk® Inc., USA) apresenta numa coluna a informação das entidades que compõem o desenho, mas de todas estas, as que interessam, são as entidades de linhas e arcos (ou círculos) que formam o perfil da peça. As demais entidades não são levadas em conta, além das linhas de centro,

linhas ocultas, etc, as quais deverão ser realizadas em outra camada (“Layer”) para que possam ser eliminadas do arquivo DXF.

Figura 15 – Exemplo de arquivo DXF, o arquivo texto (a) e o desenho correspondente (b).

O programa de pré-processamento desenvolvido lê o arquivo DXF, identifica cada um dos pares de pontos que determinam cada segmento do polígono formando uma matriz. Posteriormente os pontos são ordenados de forma que definam segmentos contínuos segundo o perfil da peça. Devido a que inicialmente as coordenadas destes pontos são de acordo com as coordenadas do CAD, as coordenadas dos pontos são trasladados de acordo com o sistema de coordenadas da máquina. A nova matriz de pontos, assim obtida, constitui a matriz de pontos de proibição que limitam a zona que a ferramenta não ultrapassará.

Como o objetivo principal neste trabalho é propor e validar uma estratégia de controle de posição, o problema é simplificado e só serão trabalhados desenhos de peças com geometrias poligonais (linhas).

6.2 Pré-processamento

O programa lê o arquivo DXF identificando cada par de pontos que determina cada segmento do polígono. Com estes pontos gera-se uma matriz “*pp*”. Nesta matriz

formam-se pares de pontos para cada aresta cuja posição dentro da matriz dependerá da sequência obtida na realização do desenho; ou seja, a sequência dos pares de pontos nesta matriz nem sempre representará o perfil da peça ao unir por segmentos estes pontos.

Figura 16 - Polígono do arquivo "polig-figura.dxf".

Por exemplo, para o perfil da peça mostrada na Fig. 16, a matriz "*pp*" inicialmente gerada ao realizar a leitura do arquivo DXF foi:

```

-----
>> LerDXF
NOME DO ARQUIVO "DXF" DA PEÇA:
polig-figura.dxf
pp =

60 80
90 60
70 20
90 20
90 20
90 60
50 40
20 20
20 20
20 20
20 70
60 80
20 70
50 40
70 20
-----

```


Observa-se como a seqüência dos pares de pontos não corresponde ao perfil da peça, então para corrigir isto, é necessário ordenar e eliminar alguns pontos de maneira que ao unir de forma consecutiva cada par de pontos da matriz, a trajetória obtida corresponda ao perfil da peça. No programa “ordenar.m” (Anexo C) pode-se observar mediante comparações entre os pontos como é obtida uma nova matriz “*P_ord*” de pontos ordenados. Continuando com o exemplo a nova matriz “*P_ord*” fica assim:

```

-----
>> ordenar

P_ord =

    60    80
    20    70
    20    20
    50    40
    70    20
    90    20
    90    60
-----

```

6.3 Convenções

6.3.1 Coordenadas dos pontos da matriz de proibição com relação às coordenadas da ferramenta

As coordenadas dos pontos da matriz ordenada “*P_ord*” são com respeito à origem escolhida. Agora, para gerar a matriz de proibição, as coordenadas dos pontos de “*P_ord*” devem ser relação às coordenadas da posição da ferramenta (ver Fig. 17); ou seja, deve-se olhar desde a posição da ferramenta “*Pf*”.

Desta maneira com: $P_ord = [P_ord(x_1, y_1), \dots; P_ord(x_i, y_i), \dots; P_ord(x_n, y_n)]$, e as coordenadas da posição da ferramenta $Pf(x_f, y_f)$, a matriz “*Prp*” com os pontos de proibição respeito a “*Pf*” é:

$$Prp = \begin{bmatrix} (x_f - P_ord(x_1)), (y_f - P_ord(y_1)), \dots; (x_f - P_ord(x_i)), (y_f - P_ord(y_i)), \dots \\ \dots; (x_f - P_ord(x_n)), (y_f - P_ord(y_n)) \end{bmatrix}$$

Figura 17 – Coordenadas da ferramenta – Área útil de usinagem.

E continuando com o exemplo do item 6.3.1, com $x_f = 350$ e $y_f = 280$, tem-se:

>> Matriz_proibição

Prp =

290	200
330	210
330	260
300	240
280	260
260	260
260	220

6.3.2 Como mapear as regiões proibida e permitida

A identificação das regiões proibida e permitida é feita através da rotulação dos pontos (ordenação) e assumindo uma convenção. Por exemplo, na Fig.18 com os pontos ordenados, ao percorre-los numa direção assume-se como região proibida a parte que fica à esquerda e como região permitida aquela que fica à direita. No

algoritmo a identificação das regiões é realizada através do sentido da componente “z” no produto vetorial entre o vetor que define o segmento e o vetor normal “V” de qualquer ponto (interno ou externo) a este segmento (ver Fig. 20 do item 6.4). A convenção para o sentido da componente “z” é assumida no momento da determinação da usinagem interna ou externa. Além disso, o operador antes de começar a usinagem, posicionará a ferramenta na região permitida e posteriormente o algoritmo nunca permitirá mudar de região.

Figura 18 – Convenção para as regiões proibida e permitida.

6.4 Compensação do raio da ferramenta no processo de usinagem

Até aqui, assumiu-se uma ferramenta pontual, mas para abordar o problema de forma real se deve considerar seu raio “R” ao gerar tanto a matriz de proibição “Prp”, para o controle da usinagem de desbaste (item 6.5), como a trajetória de usinagem para a usinagem de acabamento (item 6.6). Para isto procede-se da seguinte maneira:

Passo 1: Filtrar os pontos muito próximos de acordo com a geometria da ferramenta

Da Figura 19 tem-se:

$$d = R \cdot \tan(\phi / 2), \text{ sendo } \phi = \arccos[(V_{AB} \cdot V_{BC}) / (|V_{AB}| \cdot |V_{BC}|)]$$

Figura 19 – Trajetória com pontos muito próximos.

Se $d > |V_{AB}| \Rightarrow$ Remover B

Se $d > |V_{BC}| \Rightarrow$ Remover B

Passo 2: Com a matriz “Prp” de novo filtrada, verifica-se a posição dos pontos intermediários encontra-se acima ou abaixo do segmento formado pelos pontos entre os quais eles estão. Assim determina-se o lado onde ficará o novo ponto de proibição (e da trajetória) gerado por este ponto intermediário.

Figura 20 – Vetor V perpendicular ao segmento.

Da Figura 20 tem-se:

$$V = [V_{AC} \otimes (V_{AB} \otimes V_{AC}) / |V_{AC} \otimes (V_{AB} \otimes V_{AC})|] \cdot h$$

$$V = [(C - A) \otimes [(B - A) \otimes (C - A)] / |(C - A) \otimes [(B - A) \otimes (C - A)]|] \cdot h$$

Se $V \cdot H < 0, \Rightarrow B$ abaixo do segmento AC

Se $V \cdot H > 0, \Rightarrow B$ acima do segmento AC

- Para B acima do segmento AC

Da Figura 21: $V_{AB} = (B - A) = (x, y)$, e R : raio da ferramenta

$(B - A)/\|B - A\| = (x, y)/\|B - A\|$, Vetor Unitário na direção de V_{AB}

$(-y, x)/\|B - A\|$, Vetor Unitário perpendicular a V_{AB}

$$V_{AA'} = (A' - A) = [(-y, x)/\|B - A\|] \cdot R$$

$$\Rightarrow A' = V_{AA'} + A = [(-y, x)/\|B - A\|] \cdot R + A$$

Figura 21 - Análise para B acima do segmento AC.

Da mesma maneira tem-se:

$$V_{BC} = (C - B) = (x, y)$$

$(C - B)/\|C - B\| = (x, y)/\|C - B\|$, Vetor Unitário na direção de V_{BC}

$(-y, x)/\|C - B\|$, Vetor Unitário perpendicular a V_{BC}

$$V_{CC'} = (C' - C) = [(-y, x)/\|C - B\|] \cdot R$$

$$\Rightarrow C' = V_{CC'} + C = [(-y, x)/\|C - B\|] \cdot R + C$$

Para a trajetória deve-se gerar o ponto B' deslocado “ d ” a partir do qual a ferramenta sai paralela ao vetor V_{BC} . Onde:

$$d = R \cdot \tan(\phi/2), \text{ sendo } \phi = \arccos[(V_{AB} \cdot V_{BC})/(\|V_{AB}\| \cdot \|V_{BC}\|)]$$

$$\text{e } B' = V_{AA'} + B + d(V_{AB}/\|V_{AB}\|)$$

- Para B abaixo do segmento AC

Da Figura 22: Fazendo a mesma análise que no caso anterior são determinados os pontos A' , C' e B' :

Figura 22 - Análise para B abaixo do segmento AC.

$$\Rightarrow A' = V_{AA'} + A = \left[\frac{(-y, x)}{|(B - A)|} \right] \cdot R + A$$

$$\Rightarrow C' = V_{CC'} + C = \left[\frac{(-y, x)}{|(C - B)|} \right] \cdot R + C$$

$$\Rightarrow B' = V_{AA'} + B - d \left(\frac{V_{AB}}{|V_{AB}|} \right)$$

Para $V \cdot H = 0$, o novo ponto de proibição B' será também determinado através do produto escalar entre os vetores V e Q (ver Fig. 20), com o critério:

Se $V \cdot Q < 0$, $\Rightarrow B$ direita do segmento AC

Se $V \cdot Q > 0$, $\Rightarrow B$ esquerda do segmento AC

E com um procedimento similar à análise anterior para B' acima ou abaixo do segmento AC, é determinado o ponto B' .

Reconsiderando o exemplo do item 6.3.1, com $x_f = 350$ e $y_f = 280$, $R = 2.5\text{mm}$ e uma usinagem externa, a matriz de proibição " Prp " fica:

NOME DO ARQUIVO "DXF" DA PEÇA:
polig-figura.dxf
Raio da Ferramenta em mm?

2.5

Usinagem interna (1), ou Usinagem externa (2)?

1

>> Prp=

```

289.5334 197.3064
332.5000 208.0481
332.5000 264.6713
300.3185 243.2170
281.0355 262.5000
257.5000 262.5000
257.5000 218.6620
-----

```

6.5 Controle na usinagem de desbaste

Como foi dito a usinagem de desbaste realiza-se manualmente pelo operador através de comandos de movimentação e direção desde o joystick. Porém, esta operação é monitorada pelo software de controle que evita a entrada da ferramenta na região proibida. Aqui, apresentam-se duas abordagens para realizar este controle: uma em tempo real através da verificação permanente da invasão da ferramenta em movimento, e outra, através da movimentação direta da ferramenta até a intersecção com a fronteira da região proibida. A Fig. 23 mostra de maneira sucinta as duas abordagens.

Figura 23 – Abordagens de controle para a usinagem de desbaste: (a) por verificação da invasão, (b) por ponto de intersecção.

6.5.1 Através da verificação da Invasão

Um dos problemas importantes tratados no controle da posição da ferramenta, é a verificação da posição da ferramenta relação ao perfil da peça, se a ferramenta está

dentro ou fora do perfil (com uma tolerância desejada), isto é, a verificação da invasão. Este problema é tratado pela Geometria Computacional como o problema de proximidade (ver, por exemplo, Rezende, P., (1994) [26] e Pantaleón, D., (2000) [27]).

Um método simples consiste em gerar uma matriz “*Prp*” com os pontos que conformam o perfil da peça a fabricar, sendo o tamanho da matriz tão grande quanto a resolução requerida na usinagem (a resolução dos módulos usados é de 0,01 mm). E para cada posição alcançada pela ferramenta, a distância desta posição a cada ponto da matriz é calculada. Verificando se esta distância é maior do que um valor de tolerância estabelecido “*T*”, a invasão da ferramenta é verificada. Ver a Fig. 24, onde,

$$|d(\overline{PfP(j)})| \leq T,$$

com

Pf : Posição da ferramenta

P(j): Ponto gerado dentro do segmento *P(i)P(i+1)*

T: Valor de tolerância

E: Espaçamento entre pontos gerados

Figura 24 – Método Simples.

Portanto, quanto maior é a quantidade de pontos que determinam cada segmento do perfil da peça, maior é o número de pontos gerados entre eles, e conseqüentemente maior é o número das comparações a ser feitas. Este método, embora simples, é menos eficiente comparado com outro método que será mencionado mais adiante, oferecendo problemas para uma verificação da invasão em tempo real.

Um segundo método de verificar a invasão deriva-se do critério do somatório dos ângulos (ver Fig. 25). Dada uma região delimitada por três pontos (*P1*, *P2*, e *P3*), é possível verificar se um quarto ponto (*P4*) é externo desta região através do somatório dos ângulos formados entre estes quatro pontos (*P4* com cada ponto *P1*, *P2* e *P3*). Se $\beta + \Phi + \delta < 360^\circ$ o ponto *P4* é externo à região.

Figura 25 - Verificação de invasão, critério somatório dos ângulos.

Usando algum dos métodos mencionados, os pontos que pertencem à matriz e que estão mais próximos à posição da ferramenta (Pf ou $P4$, segundo o critério) ainda não são determinados. As abordagens dos problemas de proximidade recomendam, além do ordenamento dos pontos, definir intervalos para grupos de pontos e finalmente estabelecer os critérios iniciais de comparação entre pontos e intervalos. Assim o conjunto de pontos (ou segmentos) é reduzido a subconjuntos onde o número de operações de comparação obviamente será menor, tornando o processo de verificação de invasão mais eficiente.

Aplicando este procedimento no polígono da Fig. 26, depois de ordenados os pontos de modo a percorrer o perfil da peça segundo uma direção escolhida ($P1, P2, \dots, P5, \dots$) são definidos os intervalos para cada segmento do polígono ($Xp1-Xp2, \dots, Xp4-Xp5, \dots, Yp1-Yp2, \dots, Yp4-Yp5, \dots$). É possível estabelecer os intervalos dentro dos quais a ferramenta (ponto Pf) está, e assim determinar os segmentos mais próximos a Pf (segmentos $P1P2$ e $P4P5$), descartando os demais. Deste modo reduz-se o número de operações desnecessárias na determinação da invasão. Por exemplo, bastará verificar a invasão através dos segmentos $P1P2$ e $P4P5$, usando um terceiro segmento definido por um ponto consecutivo ($P3$) e aplicar o critério do somatório de ângulos.

Figura 26 - Polígono com pontos ordenados e intervalos definidos.

Um terceiro método mais eficiente de verificar a invasão é obtido de forma similar ao apresentado anteriormente. A invasão é verificada através da distância do ponto da ferramenta (Pf) a cada um dos segmentos (ver Fig. 26). Como no método anterior, verifica-se a distância somente nos intervalos de coordenadas “x” (ou “y”) dentro dos quais o ponto Pf está. Usando a geometria vetorial, a distância a cada segmento é determinada assim:

$$d_{12(ou45)} = \left| (Pf - P1(ou4)) \otimes (P2(ou5) - P1(ou4)) \right| / \left| P2(ou5) - P1(ou4) \right| \leq T$$

Este método adota os mesmos procedimentos usados nos critérios de ordenação dos pontos e na eliminação dos intervalos, mas mostra ser mais eficiente ao eliminar parâmetros (o ponto $P3$) e diminuir operações de cálculo (determina uma distância em lugar de três ângulos). A diminuição do número de operações ao passar do método simples aos métodos com ordenação-agrupação, é descrito em Rezende (1994) [26]. Seguindo este terceiro método se desenvolveu um programa baseado no fluxograma da Fig. 27 e se realizaram alguns testes de usinagem, ver Franco e Horikawa (2007) [28].

Figura 27 – Fluxograma geral do software para controle através da verificação da invasão.

6.5.2 Através dos pontos de intersecção

Uma segunda maneira de realizar a usinagem por região proibida consiste em: determinar a direção a ser seguida (leitura do joystick), determinar a linha que passa pela ferramenta e tem a direção definida pelo joystick, procurar o ponto de intersecção desta linha com o contorno da peça (região proibida) e, por final, enviar ao controlador da máquina CN, um comando de posicionamento para o ponto de intersecção. Se durante a operação, o operador interromper o movimento através do joystick, o comando de posicionamento é interrompido.

Figura 28 – Movimentação até o ponto de intersecção.

Aqui a definição dos intervalos se conserva e servirá para eliminar aqueles segmentos cujos pontos de intersecção estão fora do intervalo que define os segmentos. Na Fig. 28 apesar do segmento $P3P4$ estar mais próximo da ferramenta (Pf), os segmentos inicialmente a ser considerados (segundo a direção da movimentação) por ter os seus pontos de intersecção dentro do intervalo para cada segmento, são os segmentos $P4P5$ e $P5P6$. Depois de comparar suas distâncias até a ferramenta, o ponto de intersecção escolhido para realizar a movimentação será o ponto $Pint1$ do segmento $P4P5$. Desta maneira, pode-se eliminar possíveis problemas gerados pelo atraso do controle que ocasionaram imprecisão e as vezes a invasão da região proibida na abordagem anterior; além de ser mais eficiente ao evitar a constante comparação da distância entre cada ponto alcançado pela

ferramenta durante o avanço e o segmento do perfil mais próximo, com o valor de tolerância assumido. A seguir os fluxogramas usados para o desenvolvimento do algoritmo de controle segundo esta abordagem. A Fig. 29 apresenta o fluxograma

Figura 29 - Fluxograma geral do software de controle através dos pontos de intersecção.

geral do software de controle com os grandes módulos que compõem o processo de usinagem, mostrando de forma sucinta a usinagem de desbaste. Seguidamente apresentam-se, em detalhe, alguns destes módulos. A Fig. 30 mostra os ajustes que são necessários para estabelecer a comunicação entre o PC e o controlador, e a forma de zerar a máquina para que todas as operações de movimentação posteriores sejam assim referenciadas com este zero.

Figura 30 - Fluxograma. (a) Ajustar os parâmetros para comunicação "Interface Specifications". (b) Zerar o dispositivo "Home Return Operation".

A figura 31 apresenta o fluxograma para o programa que lê o arquivo DXF e gera a matriz de pontos de proibição ou Região Proibida. E a Fig. 32 o fluxograma para a usinagem de desbaste.

Figura 31 - Fluxograma. Ler arquivos *DXF*, filtrar e ordenar dados / Gerar região proibida: Matriz de pontos Prp .

Figura 32 - Fluxograma Usinagem de desbaste.

Figura 32 – Continuação do Fluxograma Usinagem de desbaste.

6.6 Controle na usinagem de acabamento

Até aqui considerou-se o controle durante a usinagem de desbaste. Depois desta usinagem obtém-se o perfil da peça muito aproximado, embora, como observa-se na Fig. 33(a), pequenas zonas não usinadas sobram no perfil. Sendo necessário realizar uma usinagem de acabamento para obter uma peça com o perfil e as dimensões finais, ver Fig. 33(b). Um módulo de programa é desenvolvido de modo a

Figura 33 – Peça após as usinagens (a) de desbaste, (b) de acabamento.

se gerar automaticamente os comandos de movimentação em forma seqüencial entre os pontos que conformam a região proibida (matriz "*Prp*"), já compensada pelo raio da ferramenta (ver item 6.4).

Figura 34 – Fluxograma Usinagem de acabamento.

A Fig. 34 ilustra o fluxograma do controle usado para esta operação. O operador aproxima a ferramenta sobre a peça. Em seguida, o programa identifica o ponto de “*Prp*” mais próximo à posição atual da ferramenta. Com este ponto, reordena-se a matriz de forma que o seu primeiro ponto seja o ponto identificado. O ponto identificado ainda é adicionado como sendo o último ponto da matriz para fechar o percurso do perfil da peça. Por final, são gerados comandos de interpolação linear entre os pontos que conformam o perfil.

7 TESTES E RESULTADOS

Neste capítulo pretende-se avaliar a estratégia da Região Proibida e da arquitetura desenvolvida para alcançar a eficiência na usinagem. Para testes de usinagem por região proibida, foram utilizadas as abordagens de controle por verificação de invasão e por controle através dos pontos de intersecção. Aqui é nomeado teste de simulação, aquele teste onde são realizadas as trajetórias de usinagem sem a remoção de material (furadeira desligada).

Realizaram-se dois tipos de testes, um de simulação para ambas as abordagens, tanto para a usinagem de desbaste, como para a usinagem de acabamento. Devido a que os testes para o controle por invasão não mostraram um 100% de eficiência, só apresenta-se um resumo dos resultados. Estes resultados foram apresentados em Franco e Horikawa (2007) [28]. No teste de simulação, para visualizar as trajetórias obtidas, em lugar da ferramenta de corte, foi usado um lápis para registrar a trajetória numa folha de papel fixada na mesa. Nas figuras obtidas desta simulação, os caminhos irregulares correspondem às trajetórias da usinagem de desbaste, o caminho “a”, à trajetória do centro da ferramenta durante a usinagem de acabamento, e a trajetória “b” ao perfil da peça (obtido após a usinagem de acabamento). O outro tipo de teste, só para o controle através da intersecção, foi realizando usinagens reais usando uma fresa de 5mm de diâmetro e como matéria-prima placas plásticas de 4.75mm de espessura.

7.1 Testes através da verificação da invasão

Como nos testes o sistema mostrou um atraso devido à limitação na velocidade de comunicação entre o PC e o controlador, aos tempos operacionais, e à inércia mecânica dos equipamentos; uma série de testes foram realizados variando a velocidade de avanço “V” e o valor teórico de tolerância “T” (ver item 6.5.1). Para este procedimento, a porcentagem de sucessos (quando a região proibida não é ultrapassada) e a média do erro (sobre-dimensão, posição alcançada e distante uma certa quantidade do perfil), são determinadas para a usinagem de desbaste. A

Tabela 2 apresenta os resultados da usinagem em triângulo e em losango. A Tab. 3 e a Fig. 35 os resultados da usinagem em polígono de 7 lados.

Tabela 2 – Resultados da simulação da usinagem: (a) externa em triângulo, (b) interna em losango.

Tipo de Simulação	Velocidade "V" (mm/min)	Tolerância "T" (mm)	No. Intentos	No. Sucessos	Média do Erro (mm)	% do sucesso
(a)	1080	1.0	17	0	-	0.00
(a)	1080	2.0	17	3	0.15	17.65
(a)	1080	3.0	17	11	0.54	64.70
(a)	1080	4.0	18	16	1.25	88.90
(a)	1080	5.0	20	20	2.50	100.00
(a)	720	1.0	15	1	0.50	6.70
(a)	720	2.0	19	4	0.25	21.05
(a)	720	3.0	18	14	0.68	77.80
(a)	720	4.0	19	17	1.15	89.50
(a)	720	5.0	20	19	2.63	95.00
(b)	1080	4.0	23	22	1.40	95.65
(b)	720	4.0	25	25	1.10	100.00
(b)	1080	4.5	28	27	2.00	96.40
(b)	720	4.5	28	28	1.95	100.00
(b)	1080	5.0	28	28	2.46	100.00
(b)	720	5.0	28	28	2,48	100.00

Tabela 3 – Resultados da simulação da usinagem interna e externa em polígono de 7 lados.

Figura do teste	Velocidade "V" (mm/min)	Tolerância "T" (mm)	No. Intentos	No. Sucessos	Média do Erro (mm)	% do sucesso
35(a)	720	4.5	46	44	0.90	95.65
35(b)	720	4.5	46	43	1.12	93.50

Figura 35 – simulação da usinagem interna e externa em polígono de 7 lados.

7.2 Testes através dos pontos de intersecção

7.2.1 Usinagem em Triângulo

As Figuras 36 e 37 apresentam a simulação das usinagens externa e interna respectivamente. Observa-se a trajetória da usinagem de desbaste (para diferentes direções de movimentação) e a geração da usinagem de acabamento segundo o raio da ferramenta. A Fig. 38 apresenta a usinagem de desbaste interna real com as rugosidades na borda próprias desta usinagem.

Figura 36 – Usinagem externa em Triângulo. $R = 2,5\text{mm}$.

Figura 37 – Usinagem interna em Triângulo. $R = 2,5\text{mm}$.

Figura 38 – Triângulo após a usinagem de desbaste interna.

7.2.2 Usinagem em Losango

De forma similar ao caso anterior, as Fig. 39 e 40 apresentam a simulação das usinagens externa e interna num losango.

Figura 39 – Usinagem externa em Losango. $R = 2,5\text{mm}$.

Figura 40 – Usinagem interna em Losango. $R = 2,5\text{mm}$.

A Figura 41 apresenta a usinagem de desbaste externa real com as rugosidades típicas deixadas pela usinagem de desbaste.

Figura 41 – Losango após a usinagem de desbaste externa.

Na Figura 42 observa-se o triângulo e o losango sem rugosidades e com a dimensão final após a usinagem de acabamento.

Figura 42 – Losango e Triângulo após a usinagem de acabamento.

7.2.3 Usinagem em polígonos

Realizaram-se testes similares com polígonos irregulares (côncavos e convexos) de maior número de lados. Nas Fig. 43, 45 e 48 observam-se as simulações das usinagens e nas Fig. 44, 46-47 e 49-53 as usinagens reais.

Figura 43 – Usinagem externa em polígono 7 lados. $R = 2,5\text{mm}$.

Figura 44 – Polígono de 7 lados após a usinagem de desbaste externa.

Figura 45 – Usinagem interna em polígono 7 lados. $R = 2,5\text{mm}$.

Figura 46 – Polígono de 7 lados após a usinagem de desbaste interna.

Figura 47 – Polígono de 7 lados após a usinagem de acabamento.

Figura 48 – Usinagem externa a sigla “USP”. $R = 2,5\text{mm}$.

Figura 49 – Sigla “USP” após a usinagem de desbaste.

Figura 50 – Sigla “USP” após a usinagem de acabamento.

Figura 51 – Polígono de 14 lados após a usinagem de desbaste interna.

Figura 52 – Polígono de 16 lados após a usinagem de desbaste externa.

Figura 53 – Polígonos de 14 e 16 lados após a usinagem de acabamento.

7.2.4 Precisão da Usinagem

Na Tabela 4, para avaliar a precisão da usinagem, algumas medidas das peças usinadas são comparadas com as respectivas medidas de desenho.

Tabela 4 – Comparação de medidas.

Medida (Peça – Figura)	Medida de Desenho (mm)	Medida após Usinagem (mm)
A (Losango – Fig. 39), ext.	60.62	60.73
B (Polígono 7 lados – Fig. 43) int	80.00	80.20
B (Polígono 7 lados – Fig. 43) ext	80.00	80.03
C (Polígono 16 lados – Fig. 53) int	70.00	69.80
C (Polígono 16 lados – Fig. 53) ext	70.00	70.16
D (Polígono 16 lados – Fig. 53) int	85.00	84.80
D (Polígono 16 lados – Fig. 53) ext	85.00	85.20

7.3 Discussão dos resultados

7.3.1 Testes através da verificação da invasão

Resultados mostram que a porcentagem de sucesso é fortemente influenciada pela velocidade de avanço “V”. Ao aumentar a velocidade “V” a porcentagem de sucesso

diminui. E conservando a velocidade “ V ”, ao aumentar o valor de tolerância “ T ” a porcentagem de sucesso aumenta. O anterior é determinado pelo atraso, que como já se mencionou, apresenta o sistema.

7.3.2 Testes através dos pontos de intersecção

Seja na usinagem internas ou externa, sejam peças com geometria simples ou complexas e independente da direção de movimentação, o movimento da ferramenta não ultrapassou em nenhum dos casos o contorno da peça desejada.

Observa-se como efetivamente a trajetória para a usinagem de acabamento vai sempre paralela ao perfil da peça e distante desta, segundo o raio da ferramenta. Isto demonstra a eficácia do algoritmo desenvolvido para considerar o raio da ferramenta.

As usinagens foram realizadas com velocidade de avanço constante, onde o operador, após posicionar a ferramenta na região permitida, movimentava o joystick em qualquer direção sem preocupar-se por usinar a peça além da medida final.

Com a usinagem de desbaste obtém-se a forma (contornos) das peças de maneira muito aproximada sobrando pequenas protuberâncias que foram removidos com a usinagem de acabamento; sendo o resultado final, peças com contornos fieis aos esperados e dimensões conforme o projeto.

A pequena diferença observada, entre as medidas de desenho e as medidas reais após a usinagem de acabamento, não são decorrentes do algoritmo de controle e podem ser justificadas ao considerar que a máquina desenvolvida é um protótipo, no qual, aspectos de rigidez podem estar influenciando, além da implementação de um sistema melhor para a fixação da peça a usinar.

8 CONCLUSÕES E SUGESTÃO PARA TRABALHOS FUTUROS

Este trabalho propôs uma nova estratégia de controle para máquinas de usinagem por “Região Proibida”. Sendo esta região a geometria e as dimensões da peça a ser usinada, onde a ferramenta não entrará durante a usinagem realizada pelo operador. Estratégia que abre as possibilidades para o desenvolvimento de máquinas com características entre uma máquina manual e uma máquina CN. Este trabalho sugere que a máquina proposta seja adequada para um perfil de produção que apresente pequenos lotes e grande variedade.

De modo a viabilizar a usinagem por região proibida, propôs-se uma configuração de máquinas e segundo esta configuração foi desenvolvido um protótipo. O protótipo consiste de uma fresadora para usinagem bidimensional em um plano, composta por dois módulos comerciais de CN, um computador com o software de controle e um joystick para a operação manual. A fabricação do protótipo incluiu o desenvolvimento de programas de controle. O primeiro módulo que recebe arquivos de desenhos de fabricação provenientes de um sistema CAD, pre-processando-o de modo a definir a região proibida. Um segundo módulo que realiza o controle de usinagem por região proibida, verificando constantemente se a ferramenta invade esta região. Um terceiro módulo alternativo ao anterior, que realiza o mesmo controle, mas sendo através do posicionamento para o ponto da peça imediatamente à frente da ferramenta, na direção comandada pelo operador através do joystick. Este terceiro módulo se mostrou mais eficaz que o anterior, que apresentou problemas de imprecisão devido a atrasos na comunicação entre o computador de controle e o CN. Por último, um quarto módulo foi desenvolvido para realizar a usinagem de acabamento, fazendo com que a ferramenta contorne a peça automaticamente, levando em consideração o raio da ferramenta.

Com o protótipo desenvolvido demonstrou-se a viabilidade técnica de usinar peças limitando o avanço da ferramenta através da região proibida. Partindo diretamente de um desenho em CAD sem a necessidade da programação em linguagem G para cada desenho de peça. E com os modos de operação manual a critério do operador

para a usinagem de desbaste e o modo automático para realizar a usinagem de acabamento.

Não foi realizado nenhum ensaio comparando uma usinagem manual com a usinagem segundo a máquina proposta neste trabalho. Contudo, verificou-se que o protótipo desenvolvido permite um posicionamento da ferramenta de forma mais ágil que a máquina manual, liberando o operador do controle dimensional e contribuindo para uma usinagem mais eficiente. Além de permitir, de maneira automática, uma usinagem final com acabamento e precisão similares aos obtidos com as máquinas CN.

Recomendações para trabalhos futuros estão encaminhadas em busca de algoritmos mais eficientes, com a finalidade de alcançar a usinagem de peças com geometrias complexas (por exemplo, peças com perfis curvos). Uma sugestão, com relação à abordagem do controle da usinagem através da verificação da invasão, seria avaliar o emprego de algum método baseado em um algoritmo de imagem, por exemplo, a técnica do “Quadtree” (SAMET,1989 [29]). Sendo que para esta abordagem o método desenvolvido não mostrou um 100% de eficiência no posicionamento. Nesta técnica, apesar da discretização do espaço ser lenta, o acesso posterior para a verificação de um ponto (ferramenta) pertencer à região discretizada (peça) pode ser rápido.

REFERÊNCIAS BIBLIOGRÁFICAS

1. GROOVER, Mikell P, "Automation, production systems, and computer integrated manufacturing", Prentice-Hall, 2nd Edition, USA, 2000. 856p
2. HORIKAWA, O., et al. "Simple Architecture for a Manual and Servo Assisted, hybrid system for position control in machine tools", submitted to the 19th International Congress of Mechanical Engineering – COBEM2007, Brasilia, DF, Brazil, November 5-9, 2007.
- 3 YOON, J.,-H., "Fast tool path geration by the Iso-Scallop height method for ball-end milling of sculptured surfaces" International Journal of Production Research, 43(23), pp 4989-4998, Dec., 2005.
- 4 JEE, S. and KOO, T., "Tool-path generation for NURBS surfaces machining", IEEE, Proceeding of the American Control Conference, Denver, 3, pp 2624-2619, 2003.
- 5 TAM, H.-Y, XU, H. and ZHOUE, Z. "Iso-planar interpolation for the machining of implicit surfaces", Computer-Aided Design. 34, pp 125 -136, 2002.
6. JUN, C-S., KIM, D-S. and PARK, S., "A new curve-based approach to polyhedral machining", Computer-Aided Design, 34, pp. 379-389, 2003.
7. TSUZUKI, M. S.G., "Geração Automática da Trajetória da Ferramenta para Acabamento na Usinagem de Superfícies". In: IV Simpósio Brasileiro de Automação Inteligente, São Paulo, pp. 278-284, 1999.
8. OLIVEIRA, Leandro C., "Um sistema de Geração de Trajetórias de Ferramentas em 3 Eixos", Universidade Federal do Rio Grande do Sul, Instituto de Informática. Dissertação, Porto Alegre, 1997,92 p.
9. ROBER, S. J. and SHIN, Y. C., "Modeling and control of CNC machines using a PC-Based open architecture controller", Mechatronics, vol.5, No.4, pp.401-420, 1995.
10. ONWUBOLU, G. C., et al. "Development of a PC-Based computer numerical control drilling machine", Proc. Instn. Mech. Engrs., vol.216, Part B: J. Engineering Manufacture, 2002.
11. HANAFI, D., TORDON, M. and KATUPI, J., "An Active Axis Control System for a Conventional CNC Machine", IEEE, vol.2, pp.1188-1193, 2003.
12. HABER, R. E. and ALIQUE, J. R., "Nonlinear internal model control using neural networks: an application for machining processes", Nueral Computing & Applications, 13(1), pp 47-55, April, 2004.
13. PARK, E-C., LIM, H. and CHOI, C-H., "Position Control of X-Y Table at Velocity Reversal Using Presliding Friction Characteristics", IEEE, vol.11, No.1, pp.24-31, 2003.

14. BUENO, R. et al. "Sistema de localización 3D de alta precisión aplicado al control directo de posición de máquinas –herramienta", Fatronik, IAI-CSIC y AIDO, XIV Congreso de Máquinas – Herramienta y Tecnología de Fabricación, Donostia 14-16 oct, 2002.
15. SHAWKY, A., ROSENBERGER, T. and ELBESTAWI, M., "Inprocess Monitoring and Control of Thickness error in Machining hollow shafts", *Mechatronics*, vol.8, pp.301-322, 1998.
16. AL-KINDI, G.A.H., BAUL, R.M. and GILL, K.F., "Vision – Controlled CNC machines", *Computing & Control Engineering Journal*. Vol. 4, pp. 92-96, 1993.
17. ZHOU, M. C., "Modeling, Simulation, and Control of Flexible Manufacturing Systems: A Petri Net Approach", *World Scientific Technology & Industrial Arts*, New Jersey, USA, 1999, 409 p.
18. TEMPELMEIER, H. and KUHN, H., "Flexible Manufacturing Systems: Decision support for design and operation". Wiley interscience, 1993, 471 p.
19. DECOTIGNIE, J.D. and GRÉGORIE, J.Ch., "Integrating the numerical controller and the FMS", *Proceedings IECON-89*, Philadelphia, USA, November 4-10, pp. 675-680, 1989.
20. GUNASEKARAN, A., "Agile manufacturing: enablers and an implementation framework", *International Journal of Production Research*, Vol.36, Issue 5, pp. 1223 – 1247, 1998.
21. HOOP, W.J., IRAVANI, S.M.R., and SHOU, B., "Serial Agile Production Systems with Automation", *Operation Research*, Vol. 53(5), USA, pp. 852-866, 2005.
22. CAMARINHA-MATOSS, L.M. and AFSARMANESH, H. (Eds.), *Balanced Automation Systems: Architectures and Design Methods*, CHAPMAN & HALL, London, UK, 1995.
23. RIASCOS, L.A.M. and MIYAGI, P.E., "Supervisor system for detection and treatment of failures inbalanced automation systems using Petri nets", *Proceedings of the 2000 IEEE International Conference on Systems, Man, and Cybernetics*, Vol. 4, pp. 2528-2533, 2001.
24. NSK Ltd. Robot Modules System. User's Manual 1. Product Description and Installation. 1st Edition, 4th printing, April 11, 1997. Document Number: K20011-01.
25. NSK Ltd. Robot Modules System. User's Manual 2. Programming and Operation. 1st Edition, 4th printing, April 11, 1997. Document Number: K20011-01.
26. REZENDE, Pedro J. e STOLFI, Jorge. "Fundamentos de Geometria Computacional". Recife UFPE-DI, 1994, 208 p.
27. PANTALEÓN D., Carlos R. "Geometria Computacional de Pontos em Movimento" (Dissertação). Universidade de São Paulo, Brasil, 2000, 93 p.

28. FRANCO, L. and HORIKAWA, O., Manual and servo assisted, hybrid system for position control in machine tools. Submitted to the 19th International Congress of Mechanical Engineering – COBEM2007, Brasilia, DF, Brazil, November 5-9, 2007.
29. SAMET, Hanan. “The Design and Analysis of Spatial Data Structures”. Addison-Wesley, USA, 1989, 493 p.

ANEXOS

A seguir os programas desenvolvidos para a estratégia de usinagem por Região Proibida, através da abordagem por pontos de intersecção.

ANEXO A: Programa Inicial

```
clear all
LerDXF; % Lee o arquivo DXF y cria uma matriz "pp" con os pontos que conforma cada segmento de linha
pause(1);
ordenar; % Os pontos da matriz "pp" sao ordenados segundo a sequencia do perfil e gera-se a nova matriz
        % de pontos ordenados "P_ord".
pause(1);
INIC_EXC;
comando('SVON'); %Servomotores "ON"
comando('HOM'); %Levar ate origem
for i=1:1e3
    fclose(EXC_CONT);
    fopen(EXC_CONT);
    flushinput(EXC_CONT);
    fwrite(EXC_CONT,80); %"POS"
    fwrite(EXC_CONT,79);
    fwrite(EXC_CONT,83);
    fwrite(EXC_CONT,13);
    pause(0.5);
    if EXC_CONT.bytesavailable~=0
        datoin=fread(EXC_CONT, EXC_CONT.bytesavailable, 'uchar');
        POS=[char(datoin(1:17))];
        fclose(EXC_CONT);
        if POS=='X0000.00 Y0000.00'
            comando('SVOF');
            break
        end
    end
end
Ciiu=input('coordenada inferior esquerda da area da usinagem [a,b] em mm?\n');
Csdu=input('coordenada superior direita da area da usinagem [c,d] em mm?\n');
R=input('Raio da Ferramenta em mm?\n');
TU=input('Usinagem interna (1), ou Usinagem externa (2)?\n');
GerapontosRm;
T_usinagem=input('Entre um (1) para Usinagem de desbaste ou dois (2) para Usinagem acabamentoo\n');
if T_usinagem==1
    U_brutam;
end
if T_usinagem==2
    U_acabamentom;
end
```

ANEXO B: Programa “LerDXF”

```
clear all;
arquivo=input('NOME DO ARQUIVO "DXF" DA PEÇA:\n','s');
```

```

[A]= textread(arquivo,'%s'); %Lê "arquivo" e gera matriz "A"
n=size(A);
n=n(1,1);
j=0;
for i=1:n
    if str2double(A(i))==10 & str2double(A(i+2))==20 & str2double(A(i+6))==11 & str2double(A(i+8))==21
        j=j+1;
        xi(j)=str2double(A(i+1));
        yi(j)=str2double(A(i+3));
        xf(j)=str2double(A(i+7));
        yf(j)=str2double(A(i+9));
        ppi(j,:)=[xi(j) yi(j)];
        ppf(j,:)=[xf(j) yf(j)];
    end
end
pp=[ppi(1,:);ppf(1,:)];
for m=2:j
    pp=[pp;ppi(m,:);ppf(m,:)];
end
pp; %Matriz com os pontos inicial e final de cada segmento

```

ANEXO C: Programa “ordenar”

```

%pp: Matriz de pontos lidos do arquivo DXF
n=size(pp);
n=n(1);
for i=1:n/2
    P_ord(i,:)=pp(1,:);
    for j=2:n
        if pp(j,:)==P_ord(i,:) & rem(j,2)==0
            pp=[pp(j-1,:);pp(1:j-2,:);pp(j+1:n,:)];
            break
        end
        if pp(j,:)==P_ord(i,:) & rem(j,2)~=0
            pp=[pp(j+1,:);pp(j,:);pp(1:j-1,:);pp(j+2:n,:)];
            break
        end
    end
end
P_ord; %Matriz com os pontos ordenados segundo sequencia do perfil
n=size(P_ord);
n=n(1,1);

```

ANEXO D: Programa Inicial de Comunicação “INIC_EXC”: Para ajuste de parâmetros e habilitação do modo para controle remoto

```

EXC_CONT = serial('COM1');
set(EXC_CONT, 'BaudRate', 9600);
EXC_CONT.inputbuffersize=256;
set(EXC_CONT,'BytesAvailableActionMode', 'terminator');
set(EXC_CONT,'BytesAvailableFcnMode', 'terminator');
set(EXC_CONT,'DataBits', 8);
set(EXC_CONT,'DataTerminalReady', 'on');
set(EXC_CONT,'FlowControl', 'hardware');
set(EXC_CONT,'Parity', 'none');

```

```

set(EXC_CONT,'ReadAsyncMode', 'continuous');
set(EXC_CONT,'RequestToSend', 'on');
set(EXC_CONT,'StopBits', 2);
set(EXC_CONT,'Terminator', 'CR');
set(EXC_CONT,'Timeout', 5);
fopen(EXC_CONT);
fwrite(EXC_CONT,143); %switching to remote control
fwrite(EXC_CONT,13);
fclose(EXC_CONT);

```

ANEXO E: Programa função “comando”

```

function []=comando(c)
x=double(c);
n=length(x);
EXC_CONT=serial('COM1');
fopen(EXC_CONT);
for i=1:n
    fwrite(EXC_CONT, x(i));
end
fwrite(EXC_CONT, 13);
fclose(EXC_CONT);

```

ANEXO F: Programa “GerapontosRm”

```

%P_ord =[];
%n=size(P_ord);
%n=n(1,1);
for i=1:n
    P_ord3D(i,:)=P_ord(i,:),0];
end
%**** Determinar um dos pontos com "x" mais perto da origem ****%
refmin=300;
a=0;
for i=1:n
    if refmin>P_ord3D(i,1)
        refmin=P_ord3D(i,1);
        a=i;
    end
end
if (a-1)==0
    P_ord3D=[P_ord3D(a,:);P_ord3D(a+1:n,:)];
elseif (a-1)~=0
    P_ord3D=[P_ord3D(a,:);P_ord3D(a+1:n,:);P_ord3D(1:a-1,:)];
end
%*****
H=[0,-1,0];
P_ord3D(n+1,:)=P_ord3D(1,:);
P_ord3D(n+2,:)=P_ord3D(2,:);
M3=(P_ord3D(n,2)-P_ord3D(1,2))/(P_ord3D(n,1)-P_ord3D(1,1));
s=1/0;
%TU=input('Usinagem interna (1), Usinagem externa (2)');
%R=input('Radio da Ferramenta em mm?\n');
if R==0

```



```

Vab=P_ord3D(2,:)-P_ord3D(1,:);
P_ord=P_ord3D;
Ai=[[-Vab(1,2),-Vab(1,1),0]/norm(Vab)]+P_ord3D(1,:);
Vaai=Ai-P_ord3D(1,:);
dA=sqrt((P_ord3D(1,1))^2+(P_ord3D(1,2))^2);
dAi=sqrt((Ai(1,1))^2+(Ai(1,2))^2);
M1(1)=(P_ord3D(1,2)-P_ord3D(2,2))/(P_ord3D(1,1)-P_ord3D(2,1));
if M1(i)>=5000
    M1(i)=s;
elseif M1(i)<=-5000
    M1(i)=-s;
end
M2(1)=(P_ord3D(2,2)-P_ord3D(3,2))/(P_ord3D(2,1)-P_ord3D(3,1));
i=1;
if TU==1
    if (dA<dAi)& M1(i)==0 &M3<0 &M3~=s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dA>dAi)& M1(i)==0 &M3>0 &M3~=s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3>M1(i) &M3~=s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3<M1(i) &M3~=s &M3>0
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)==s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)<0 &M3>M1(i) &M3~=s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)<0 &M3<M1(i) &M3~=s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)>0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)<0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)<0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dAi<dA)& M1(i)==0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dAi>dA)& M1(i)==0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    end
    Vaia=P_ord3D(i,:)-Ai;
    F=cross(Vab,Vaia);
elseif TU==2
    if (dA>dAi)& M1(i)==0 &M3<0 &M3~=s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);

```

```

    Vaai=Ai-P_ord3D(i,:);
elseif (dA<dAi)& M1(i)==0 &M3>0 &M3~=s
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3>M1(i) &M3~=s
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)>0 &M3<M1(i) &M3~=s
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)==s
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)<0 &M3>M1(i) &M3~=s
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)<0 &M3<M1(i) &M3~=s
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)>0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)<0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)<0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (dAi>dA)& M1(i)==0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
elseif (dAi<dA)& M1(i)==0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
    Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)]+P_ord3D(i,:);
    Vaai=Ai-P_ord3D(i,:);
end
Vaia=P_ord3D(i,:)-Ai;
F=cross(Vab,Vaia);
end
elseif R~=0
    if TU==1
        for i=1:n
            Vac=P_ord3D(i+2,:)-P_ord3D(i,:);
            Vab=P_ord3D(i+1,:)-P_ord3D(i,:);
            Vbc=P_ord3D(i+2,:)-P_ord3D(i+1,:);
            M1(i)=(P_ord3D(i,2)-P_ord3D(i+1,2))/(P_ord3D(i,1)-P_ord3D(i+1,1));
            if M1(i)>=5000
                M1(i)=s;
            elseif M1(i)<=-5000
                M1(i)=-s;
            end
            M2(i)=(P_ord3D(i+1,2)-P_ord3D(i+2,2))/(P_ord3D(i+1,1)-P_ord3D(i+2,1));
            h=norm(cross(Vac,Vab))/norm(Vac);
            V=(cross(Vac,cross(Vab,Vac))/norm(cross(Vac,cross(Vab,Vac)))).*h;
            ang=acos(escal(Vab,Vbc)/[norm(Vab)*norm(Vbc)]);
            d=R*tan(ang/2);
            Ai=[[-Vab(1,2),-Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
            Vaai=Ai-P_ord3D(i,:);
        end
    end
end

```

```

K=escal(V,H);
if i==1
    dA=sqrt((P_ord3D(i,1))^2+(P_ord3D(i,2))^2);
    dAi=sqrt((Ai(i,1))^2+(Ai(i,2))^2);
    if (dA<dAi)& M1(i)==0 &M3<0 &M3~=-s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dA>dAi)& M1(i)==0 &M3>0 &M3~=-s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)>0 &M3>M1(i) &M3~=-s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3<M1(i) &M1(i)~=-s &M1(i)~=-s %&M3>0
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)===s %&M3~=-s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)<0 &M3>M1(i) %&M3~=-s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)<0 &M3<M1(i) &M1(i)~=-s
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)>0 &M3===s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3===s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)<0 &M3===s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)<0 &M3===s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dAi<dA)& M1(i)==0 &M3===s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dAi>dA)& M1(i)==0 &M3===s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    end
    Vaia=P_ord3D(i,:)-Ai;
    F=cross(Vab,Vaia);
    if M1(i)==0 &M3>0 &M3~=-s &K<0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
    elseif M1(i)==0 &M3>0 &M3~=-s &K>0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
    elseif M1(i)==0 &M3<0 &M3~=-s &K<0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
    elseif M1(i)==0 &M3<0 &M3~=-s &K>0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
    elseif M1(i)==0 &M3===s &(P_ord3D(n,2)>P_ord3D(i,2))&K<0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
    elseif M1(i)==0 &M3===s &(P_ord3D(n,2)>P_ord3D(i,2))&K>0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
    elseif M1(i)==0 &M3===s &(P_ord3D(n,2)<P_ord3D(i,2))&K<0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;

```

```

elseif M1(i)==0 & M3==s & (P_ord3D(n,2)<P_ord3D(i,2))&K>0
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) & M3>M1(i) & M3~s & K<0 & M1(i)~s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) & M3>M1(i) & M3~s & K>0 & M1(i)~s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) & M3<M1(i) & M3~s & K<0 & M1(i)~s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) & M3<M1(i) & M3~s & K>0 & M1(i)~s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) & M3==s & (P_ord3D(n,2)>P_ord3D(i,2))&K<0 & M1(i)~s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) & M3==s & (P_ord3D(n,2)>P_ord3D(i,2))&K>0 & M1(i)~s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) & M3==s & (P_ord3D(n,2)<P_ord3D(i,2))&K<0 & M1(i)~s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) & M3==s & (P_ord3D(n,2)<P_ord3D(i,2))&K>0 & M1(i)~s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif M1(i)==s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
end
end
if i>=2
    Bi=P_ord3D(i+1,:)+Vaai;
    M4(i)=(P_ordR(i-1,2)-Bi(1,2))/(P_ordR(i-1,1)-Bi(1,1));
    if M4(i)>=5000
        M4(i)=s;
    elseif M4(i)<=-5000
        M4(i)=-s;
    end
    if M1(i)~s & M1(i)~s
        if abs(abs(M4(i))-abs(M1(i)))>0.001
            Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
            Vaai=Ai-P_ord3D(i,:);
            Bi=P_ord3D(i+1,:)+Vaai;
        end
    elseif (M1(i)==s | M1(i)==-s)&(M4(i)~s&M4(i)~s)
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
        Bi=P_ord3D(i+1,:)+Vaai;
    end
    dcbi=sqrt((P_ord3D(i+2,1)-Bi(1,1))^2 + (P_ord3D(i+2,2)-Bi(1,2))^2);
    dcb=sqrt((P_ord3D(i+2,1)-P_ord3D(i+1,1))^2 + (P_ord3D(i+2,2)-P_ord3D(i+1,2))^2);
    if dcbi>dcb
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
    elseif dcbi<dcb
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
    end
end
end
end
if TU==2
    for i=1:n
        Vac=P_ord3D(i+2,:)-P_ord3D(i,:);
        Vab=P_ord3D(i+1,:)-P_ord3D(i,:);
        Vbc=P_ord3D(i+2,:)-P_ord3D(i+1,:);
        M1(i)=(P_ord3D(i,2)-P_ord3D(i+1,2))/(P_ord3D(i,1)-P_ord3D(i+1,1));
        if M1(i)>=5000
            M1(i)=s;
        elseif M1(i)<=-5000

```

```

M1(i)=-s;
end
M2(i)=(P_ord3D(i+1,2)-P_ord3D(i+2,2))/(P_ord3D(i+1,1)-P_ord3D(i+2,1));
h=norm(cross(Vac,Vab))/norm(Vac);
V=(cross(Vac,cross(Vab,Vac))/norm(cross(Vac,cross(Vab,Vac)))).*h;
ang=acos(escal(Vab,Vbc)/[norm(Vab)*norm(Vbc)]);
d=R*tan(ang/2);
Ai=[Vab(1,2),-Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
Vaai=Ai-P_ord3D(i,:);
K=escal(V,H);
if i==1
    dA=sqrt((P_ord3D(i,1))^2+(P_ord3D(i,2))^2);
    dAi=sqrt((Ai(i,1))^2+(Ai(i,2))^2);
    if (dA>dAi)& M1(i)==0 &M3<0 &M3~=-s
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dA<dAi)& M1(i)==0 &M3>0 &M3~=-s
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3>M1(i) &M3~=-s
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)>0 &M3<M1(i) &M1(i)~=-s
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)==s %&M3~=-s
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)<0 &M3>M1(i) %&M3~=-s
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)<0 &M3<M1(i) &M1(i)~=-s
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)>0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)>0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)<P_ord3D(i,1))& M1(i)<0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (Ai(i,1)>P_ord3D(i,1))& M1(i)<0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dAi>dA)& M1(i)==0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    elseif (dAi<dA)& M1(i)==0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))
        Ai=[-Vab(1,2),Vab(1,1),0]/norm(Vab).*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
    end
    Vaia=P_ord3D(i,:)-Ai;
    F=cross(Vab,Vaia);
    if M1(i)==0 &M3>0 &M3~=-s &K>0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
    elseif M1(i)==0 &M3>0 &M3~=-s &K<0
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
    elseif M1(i)==0 &M3<0 &M3~=-s &K>0

```

```

    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif M1(i)==0 &M3<0 &M3~=s &K<0
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif M1(i)==0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))&K>0
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif M1(i)==0 &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))&K<0
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif M1(i)==0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))&K>0
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif M1(i)==0 &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))&K<0
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) &M3>M1(i) &M3~=s &K>0 &M1(i)~=s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) &M3>M1(i) &M3~=s &K<0 &M1(i)~=s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) &M3<M1(i) &M3~=s &K>0 &M1(i)~=s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) &M3<M1(i) &M3~=s &K<0 &M1(i)~=s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))&K>0 &M1(i)~=s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) &M3==s &(P_ord3D(n,2)>P_ord3D(i,2))&K<0 &M1(i)~=s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))&K>0 &M1(i)~=s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
elseif (M1(i)<0 | M1(i)>0) &M3==s &(P_ord3D(n,2)<P_ord3D(i,2))&K<0 &M1(i)~=s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
elseif M1(i)==s
    P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
end
end
if i>=2
    Bi=P_ord3D(i+1,:)+Vaai;
    M4(i)=(P_ordR(i-1,2)-Bi(1,2))/(P_ordR(i-1,1)-Bi(1,1));
    if M4(i)>=5000
        M4(i)=s;
    elseif M4(i)<=-5000
        M4(i)=-s;
    end
    if M1(i)~-s & M1(i)~=s
        if abs(abs(M4(i))-abs(M1(i)))>0.001
            Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
            Vaai=Ai-P_ord3D(i,:);
            Bi=P_ord3D(i+1,:)+Vaai;
        end
    elseif (M1(i)==s | M1(i)~-s)&(M4(i)~=s&M4(i)~-s)
        Ai=[[-Vab(1,2),Vab(1,1),0]/norm(Vab)].*R+P_ord3D(i,:);
        Vaai=Ai-P_ord3D(i,:);
        Bi=P_ord3D(i+1,:)+Vaai;
    end
    dcbi=sqrt((P_ord3D(i+2,1)-Bi(1,1))^2 + (P_ord3D(i+2,2)-Bi(1,2))^2);
    dcb=sqrt((P_ord3D(i+2,1)-P_ord3D(i+1,1))^2 + (P_ord3D(i+2,2)-P_ord3D(i+1,2))^2);
    if dcbi>dcb
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)+(Vab/norm(Vab)).*d;
    elseif dcbi<dcb
        P_ordR(i,:)=Vaai+P_ord3D(i+1,:)-(Vab/norm(Vab)).*d;
    end
end
end
end
end
end

```

```
P_ordR=[P_ordR(1:n,1:2)];
P_ord=P_ordR;
end
```

ANEXO G: Função “cross”

```
function [prodc]=cross(a,b)
prodc=zeros(1,3);
prodc(1,3)=a(1,1)*b(1,2)-b(1,1)*a(1,2);
prodc(1,2)=a(1,1)*b(1,3)-b(1,1)*a(1,3);
prodc(1,1)=a(1,2)*b(1,3)-b(1,2)*a(1,3);
end
```

ANEXO H: Função “escal”

```
function [prodesc]=escal(a,b)
prodesc=a(1,1)*b(1,1)+a(1,2)*b(1,2)+a(1,3)*b(1,3);
end
```

ANEXO I: Programa Usinagem de desbaste “U_brutam”

```
format bank;
Xf=350; Yf=315; %Coordenadas da ferramenta
Prp=[(Xf-P_ord(1:n,1)),(Yf-P_ord(1:n,2))]; %Pontos da reg. proib. com relaç. ferramenta
Prp=[Prp(1:n,1),Prp(1:n,2);Prp(1,1),Prp(1,2)];
n=size(Prp);
n=n(1,1);
comando('ACC 10'); %Ajuste aceleração
comando('VEL 1'); %Ajuste velocidade
comando('SVON'); %Servomotores "ON"
daqregister('joy'); % carregar a dll pra ler joystick
ai=analoginput('joy',3); % O numero e um numero ID do Joystick intalado(ver os ID
% controladores nas opções de jogo no panel de controle)
addchannel(ai,[1 2 3 4 5]);
for i=1:n
    Prp3d(i,:)=[Prp(i,:),0];
end
H=[0,1,0];
Q=[-1,0,0];
for i=1:1e6
    J = getsample(ai);
    if (J(1)>=-1&&J(1)<=1)
        J(1)=0;
    end
    if (J(2)>=-1&&J(2)<=1)
        J(2)=0;
    end
    J(2)=-J(2);
    if J(5)~-10
        comando('STOP');
        comando('SVOF');
        break
    end
end
```

```

if (J(1)==0&&J(2)==0)
    comando('STOP');
end
while J(1)~=0 | J(2)~=0
    a=0;
    b=0;
    c=500;
    ind=1/0;
    n=size(Prp);
    n=n(1,1);
    fclose(EXC_CONT);
    fopen(EXC_CONT);
    flushinput(EXC_CONT);
    fwrite(EXC_CONT,80); %"POS"
    fwrite(EXC_CONT,79);
    fwrite(EXC_CONT,83);
    fwrite(EXC_CONT,13);
    pause(0.08);
    if EXC_CONT.bytesavailable~=0
        datoin=fread(EXC_CONT, EXC_CONT.bytesavailable, 'uchar');
        POS=[char(datoin(1:17))];
        fclose(EXC_CONT);
    end
    for i=1:n-1
        m1=(Prp(i+1,2)-Prp(i,2))/(Prp(i+1,1)-Prp(i,1));
        m2=(J(2)/J(1));
        xint=0;
        yint=0;
        Vac=Prp3d(i+1,:)-Prp3d(i,:);
        Vab=[str2double(POS(2:8)),str2double(POS(11:17)),0]-Prp3d(i,:);
        h=norm(cross(Vac,Vab))/norm(Vac);
        V=(cross(Vac,cross(Vab,Vac))/norm(cross(Vac,cross(Vab,Vac)))).*h;
        K=escal(V,H);
        S=escal(V,Q);
        Vaact=[J(1),J(2),0];
        G=cross(Vac,Vaact);
        if m1>=5000
            m1==ind;
            K==0;
        elseif m1<=-5000
            m1=-ind;
            K==0;
        end
        if (m1>=0&m1<=0.0001)|(m1<=0&m1>=-0.0001)
            m1=0;
            S=0
        end
        if m1~=m2
            if m2~=0 & m2~=ind & m2~-ind
                if m1==ind | m1=-ind
                    xint=Prp(i,1);
                    yint=m2*(xint-str2double(POS(2:8)))+str2double(POS(11:17));
                elseif m1~=ind & m1~-ind
                    xint=(m1*Prp(i,1)-m2*str2double(POS(2:8))+str2double(POS(11:17))-Prp(i,2))/(m1-m2);
                    yint=m1*(xint-Prp(i,1))+Prp(i,2);
                end
            elseif m1~=0 & J(1)~=0 && J(2)==0
                if m1==ind | m1=-ind
                    xint=Prp(i,1);
                    yint=str2double(POS(11:17));
                end
            end
        end
    end
end

```



```

elseif m1~=ind & m1~=-ind
    xint=Prp(i,1)+(str2double(POS(11:17))-Prp(i,2))/m1;
    yint=str2double(POS(11:17));
end
elseif (m1~=ind & m1~=-ind) & J(2)~=0 && J(1)==0
    if m1~=0
        yint=m1*(str2double(POS(2:8))-Prp(i,1))+Prp(i,2);
        xint=str2double(POS(2:8));
    elseif m1==0
        xint=str2double(POS(2:8));
        yint=Prp(i,2);
    end
end
end
if xint~=0|yint~=0
    if (((Prp(i,1)<=xint && xint<=Prp(i+1,1))|(Prp(i,1)>=xint && xint>=Prp(i+1,1)))&&((Prp(i,2)<=yint
&& yint<=Prp(i+1,2))|(Prp(i,2)>=yint && yint>=Prp(i+1,2))))| sqrt((Prp(i,1)-xint)^2+(Prp(i,2)-
yint)^2)<=0.1|sqrt((Prp(i+1,1)-xint)^2+(Prp(i+1,2)-yint)^2)<=0.1
        dist=sqrt((str2double(POS(2:8))-xint)^2+(str2double(POS(11:17))-yint)^2);
        if dist>=0 && dist<2
            c=dist;
            ii=i;
            break
        elseif
((J(1)>0&J(2)==0)&((S>0&K==0)|(S>0&K<0&m1>0)|(K>0&S>0&m1<0)))|((J(1)<0&J(2)==0)&((S<0&K==0
)|(S<0&K>0&m1>0)|(K<0&S<0&m1<0)))|((J(1)==0&J(2)>0)&((S<0&K>0&m1>0)|(K>0&S==0)|(K>0&S>0&
m1<0)))|((J(1)>0&J(2)<0)&((K<0&S>0&m1>0)|(K<0&S==0)|(K<0&S<0&m1<0)))|((J(1)>0&J(2)>0)&((S>0
&K==0)|(S>0&K<0&m1>m2)|(S<0&K>0&m2>m1&m1>0)|(K>0&S==0)|(S>0&K>0&m1<0)))|((J(1)>0&J(2)<
0)&((S>0&K==0)|(K<0&S>0&m1>0)|(K<0&S==0)|(K<0&S<0&m1>m2)|(K>0&S>0&m2>m1&m1<0)))|((J(1)
<0&J(2)<0)&((S<0&K==0)|(K<0&S>0&m2>m1&m1>0)|(K>0&S<0&m1>m2)|(K<0&S==0)|(K<0&S<0&m1<
0)))|((J(1)<0&J(2)>0)&((S<0&K==0)|(K>0&S>0&m2>m1&m1<0)|(K>0&S>0&m2<m1&m1<0)|(K<0&S<0&
m1<m2&m1<0)|(K>0&S==0)|(K>0&S<0&m1>0)))
            if dist<c
                a=xint;
                b=yint;
                c=dist;
            end
        end
    end
end
end
if c<500 && c>=2
    MOV=['MOV X',num2str(a,'%2f'),' Y',num2str(b,'%2f')];
    comando(MOV);
    while J(1)~=0 | J(2)~=0
        A=J(1);
        B=J(2);
        J = getsample(ai);
        J(2)=-J(2);
        if (J(1)>=-1&&J(1)<=1)
            J(1)=0;
        end
        if (J(2)>=-1&&J(2)<=1)
            J(2)=0;
        end
        if (J(1)==0&&J(2)==0)
            comando('STOP');
            break
        end
    end
end

```

```

elseif c==500
    m2=(J(2)/J(1));
    if J(1)<0 && J(2)<0
        Yint=m2*((Xf-Csdu(1,1))-str2double(POS(2:8)))+str2double(POS(11:17));
        if Yint>Yf-Csdu(1,2)
            Xint=Xf-Csdu(1,1);
        else Yint<Yf-Csdu(1,2)
            Yint=Yf-Csdu(1,2);
            Xint=((Yint-str2double(POS(11:17)))/m2)+str2double(POS(2:8));
        end
    elseif J(1)<0 && J(2)>0
        Yint=m2*((Xf-Csdu(1,1))-str2double(POS(2:8)))+str2double(POS(11:17));
        if Yint<Yf-Ciiu(1,2)
            Xint=Xf-Csdu(1,1);
        else Yint>Yf-Ciiu(1,2)
            Yint=Yf-Ciiu(1,2);
            Xint=((Yint-str2double(POS(11:17)))/m2)+str2double(POS(2:8));
        end
    elseif J(1)>0 && J(2)>0
        Yint=m2*((Xf-Ciiu(1,1))-str2double(POS(2:8)))+str2double(POS(11:17));
        if Yint<Yf-Ciiu(1,2)
            Xint=Xf-Ciiu(1,1);
        else Yint>Yf-Ciiu(1,2)
            Yint=Yf-Ciiu(1,2);
            Xint=((Yint-str2double(POS(11:17)))/m2)+str2double(POS(2:8));
        end
    elseif J(1)>0 && J(2)<0
        Yint=m2*((Xf-Ciiu(1,1))-str2double(POS(2:8)))+str2double(POS(11:17));
        if Yint>Yf-Csdu(1,2)
            Xint=Xf-Ciiu(1,1);
        else Yint<Yf-Csdu(1,2)
            Yint=Yf-Csdu(1,2);
            Xint=((Yint-str2double(POS(11:17)))/m2)+str2double(POS(2:8));
        end
    elseif J(1)<0 && J(2)==0
        Xint=Xf-Csdu(1,1);
        Yint=str2double(POS(11:17));
    elseif J(1)>0 && J(2)==0
        Xint=Xf-Ciiu(1,1);
        Yint=str2double(POS(11:17));
    elseif J(2)<0 && J(1)==0
        Yint=Yf-Csdu(1,2);
        Xint=str2double(POS(2:8));
    elseif J(2)>0 && J(1)==0
        Yint=Yf-Ciiu(1,2);
        Xint=str2double(POS(2:8));
    end
    MOV=['MOV X',num2str(Xint,'%2f'),' Y',num2str(Yint,'%2f')];
    comando(MOV);
    while J(1)~=0 | J(2)~=0
        A=J(1);
        B=J(2);
        J = getsample(ai);
        J(2)=-J(2);
        if (J(1)>=-1&&J(1)<=1)
            J(1)=0;
        end
        if (J(2)>=-1&&J(2)<=1)
            J(2)=0;
        end
    end
end

```

```

        if (J(1)==0&&J(2)==0)
            comando('STOP');
            break
        end
    end
end
if c>=0&&c<2
    if ((F(1,3)>0&&G(1,3)>0)|(F(1,3)<0&&G(1,3)<0)|(F(1,3)==0&&G(1,3)==0))
        beep
        beep
        break
    elseif
        ((F(1,3)>=0&&G(1,3)<0)|(F(1,3)<=0&&G(1,3)>0)|(F(1,3)>0&&G(1,3)<=0)|(F(1,3)<0&&G(1,3)>=0))
        if ii==(n-1)
            Prpt=Prp(1:n-1,:);
        elseif ii~=(n-1)
            Prpt=[Prp(ii+1:n-1,:);Prp(1:ii,:)];
        end
        n=size(Prpt);
        n=n(1,1);
        a=0;
        b=0;
        c=500;
        for i=1:n
            Prpt3d(i,:)=[Prpt(i,:),0];
        end
        for i=1:n-1
            m1=(Prpt(i+1,2)-Prpt(i,2))/(Prpt(i+1,1)-Prpt(i,1));
            m2=(J(2)/J(1));
            Vac=Prpt3d(i+1,:)-Prpt3d(i,:);
            Vab=[str2double(POS(2:8)),str2double(POS(11:17)),0]-Prpt3d(i,:);
            h=norm(cross(Vac,Vab))/norm(Vac);
            V=(cross(Vac,cross(Vab,Vac))/norm(cross(Vac,cross(Vab,Vac)))).*h;
            K=escal(V,H);
            S=escal(V,Q);
            if m1>=5000
                m1==ind;
                K==0;
            elseif m1<=-5000
                m1=-ind;
                K==0;
            end
            if (m1>=0&m1<=0.0001)|(m1<=0&m1>=-0.0001)
                m1==0;
                S==0
            end
            if m1~=m2
                if m2~=0 & m2~=ind & m2~-ind
                    if m1==ind | m1=-ind
                        xint=Prpt(i,1);
                        yint=m2*(xint-str2double(POS(2:8)))+str2double(POS(11:17));
                    elseif m1~=ind & m1~-ind
                        xint=(m1*Prpt(i,1)-m2*str2double(POS(2:8))+str2double(POS(11:17))-Prpt(i,2))/(m1-m2);
                        yint=m1*(xint-Prpt(i,1))+Prpt(i,2);
                    end
                elseif m1~=0 & J(1)~=0 && J(2)==0
                    if m1==ind | m1=-ind
                        xint=Prpt(i,1);
                        yint=str2double(POS(11:17));
                    elseif m1~=ind & m1~-ind

```

```

        xint=Prpt(i,1)+(str2double(POS(11:17))-Prpt(i,2))/m1;
        yint=str2double(POS(11:17));
    end
elseif (m1~=ind & m1~-ind) & J(2)~=0 & J(1)==0
    if m1~=0
        yint=m1*(str2double(POS(2:8))-Prpt(i,1))+Prpt(i,2);
        xint=str2double(POS(2:8));
    elseif m1==0
        xint=str2double(POS(2:8));
        yint=Prpt(i,2);
    end
end
end
if (((Prpt(i,1)<=xint && xint<=Prpt(i+1,1))|(Prpt(i,1)>=xint &&
xint>=Prpt(i+1,1)))&&((Prpt(i,2)<=yint && yint<=Prpt(i+1,2))|(Prpt(i,2)>=yint && yint>=Prpt(i+1,2))))|
sqrt((Prpt(i,1)-xint)^2+(Prpt(i,2)-yint)^2)<=0.1|sqrt((Prpt(i+1,1)-xint)^2+(Prpt(i+1,2)-yint)^2)<=0.1
    if
        ((J(1)>0&J(2)==0)&((S>0&K==0)|(S>0&K<0&m1>0)|(K>0&S>0&m1<0)))|((J(1)<0&J(2)==0)&((S<0&K==0)
)|(S<0&K>0&m1>0)|(K<0&S<0&m1<0)))|((J(1)==0&J(2)>0)&((S<0&K>0&m1>0)|(K>0&S==0)|(K>0&S>0&
m1<0)))|((J(1)>0&J(2)<0)&((K<0&S>0&m1>0)|(K<0&S==0)|(K<0&S<0&m1<0)))|((J(1)>0&J(2)>0)&((S>
&K==0)|(S>0&K<0&m1>m2)|(S<0&K>0&m2>m1&m1>0)|(K>0&S==0)|(S>0&K>0&m1<0)))|((J(1)>0&J(2)<
0)&((S>0&K==0)|(K<0&S>0&m1>0)|(K<0&S==0)|(K<0&S<0&m1>m2)|(K>0&S>0&m2>m1&m1<0)))|((J(1)
<0&J(2)<0)&((S<0&K==0)|(K<0&S>0&m2>m1&m1>0)|(K>0&S<0&m1>m2)|(K<0&S==0)|(K<0&S<0&m1<
0)))|((J(1)<0&J(2)>0)&((S<0&K==0)|(K>0&S>0&m2>m1&m1<0)|(K>0&S>0&m2<m1&m1<0)|(K<0&S<0&
m1<m2&m1<0)|(K>0&S==0)|(K>0&S<0&m1>0)))
        dist=sqrt((str2double(POS(2:8))-xint)^2+(str2double(POS(11:17))-yint)^2);
        if dist<c
            a=xint;
            b=yint;
            c=dist;
        end
    end
end
end
if c<500 && c>=2
    MOV=['MOV X',num2str(a,'%0.2f'),' Y',num2str(b,'%0.2f')];
    comando(MOV);
    while J(1)~=0 | J(2)~=0
        A=J(1);
        B=J(2);
        J = getsample(ai);
        J(2)=-J(2);
        if (J(1)>=-1&&J(1)<=1)
            J(1)=0;
        end
        if (J(2)>=-1&&J(2)<=1)
            J(2)=0;
        end
        if (J(1)==0&&J(2)==0)
            comando('STOP');
            break
        end
    end
elseif c==500
    m2=(J(2)/J(1));
    if J(1)<0 && J(2)<0
        Yint=m2*((Xf-Csdu(1,1))-str2double(POS(2:8)))+str2double(POS(11:17));
        if Yint>Yf-Csdu(1,2)
            Xint=Xf-Csdu(1,1);
        else Yint<Yf-Csdu(1,2)

```

```

        Yint=Yf-Csdu(1,2);
        Xint=((Yint-str2double(POS(11:17)))/m2)+str2double(POS(2:8));
    end
elseif J(1)<0 && J(2)>0
    Yint=m2*((Xf-Csdu(1,1))-str2double(POS(2:8)))+str2double(POS(11:17));
    if Yint<Yf-Ciiu(1,2)
        Xint=Xf-Csdu(1,1);
    else Yint>Yf-Ciiu(1,2)
        Yint=Yf-Ciiu(1,2);
        Xint=((Yint-str2double(POS(11:17)))/m2)+str2double(POS(2:8));
    end
elseif J(1)>0 && J(2)>0
    Yint=m2*((Xf-Ciiu(1,1))-str2double(POS(2:8)))+str2double(POS(11:17));
    if Yint<Yf-Ciiu(1,2)
        Xint=Xf-Ciiu(1,1);
    else Yint>Yf-Ciiu(1,2)
        Yint=Yf-Ciiu(1,2);
        Xint=((Yint-str2double(POS(11:17)))/m2)+str2double(POS(2:8));
    end
elseif J(1)>0 && J(2)<0
    Yint=m2*((Xf-Ciiu(1,1))-str2double(POS(2:8)))+str2double(POS(11:17));
    if Yint>Yf-Csdu(1,2)
        Xint=Xf-Ciiu(1,1);
    else Yint<Yf-Csdu(1,2)
        Yint=Yf-Csdu(1,2);
        Xint=((Yint-str2double(POS(11:17)))/m2)+str2double(POS(2:8));
    end
elseif J(1)<0 && J(2)==0
    Xint=Xf-Csdu(1,1);
    Yint=str2double(POS(11:17));
elseif J(1)>0 && J(2)==0
    Xint=Xf-Ciiu(1,1);
    Yint=str2double(POS(11:17));
elseif J(2)<0 && J(1)==0
    Yint=Yf-Csdu(1,2);
    Xint=str2double(POS(2:8));
elseif J(2)>0 && J(1)==0
    Yint=Yf-Ciiu(1,2);
    Xint=str2double(POS(2:8));
end
MOV=['MOV X',num2str(Xint,'%0.2f'),' Y',num2str(Yint,'%0.2f')];
comando(MOV);
while J(1)~=0 | J(2)~=0
    A=J(1);
    B=J(2);
    J = getsample(ai);
    J(2)=-J(2);
    if (J(1)>=-1&&J(1)<=1)
        J(1)=0;
    end
    if (J(2)>=-1&&J(2)<=1)
        J(2)=0;
    end
    if (J(1)==0&&J(2)==0)
        comando('STOP');
        break
    end
end
end
end
end
end

```

```

end
J = getsample(ai);
J(2)=-J(2);
if (J(1)>=-1&&J(1)<=1)
    J(1)=0;
end
if (J(2)>=-1&&J(2)<=1)
    J(2)=0;
end
if J(5)~= -10
    comando('STOP');
    comando('SVOF');
    break
end
if (J(1)==0&&J(2)==0)
    comando('STOP');
end
end
end
Aut_usinar_Acab=input('Realizar Usinagem de acabamento?, Sim=(1), Nao=(0)\n');
if Aut_usinar_Acab==1
    Prp=[Prp(1:n-1,1),Prp(1:n-1,2)];
    n=size(Prp);
    n=n(1,1); %quantidade de pontos
    U_acabamentom;
end

```

ANEXO J: Programa Usinagem de acabamento “U_acabamentom”

```

format bank;
Xf=350; %Coordenada X da origem pra a usinagem
Yf=315; %Coordenada Y da origem pra a usinagem
Prp=[Xf-P_ord(1:n,1),Yf-P_ord(1:n,2)]; %Pontos da reg. proib. com relaç. ferramenta
n=size(Prp);
n=n(1,1); %quantidade de pontos
comando('ACC 10'); %Ajuste aceleração
comando('VEL 1'); %Ajuste velocidade
comando('SVON');
pause(1);
fclose(EXC_CONT);
fopen(EXC_CONT);
flushinput(EXC_CONT);
fwrite(EXC_CONT,80); %"POS"
fwrite(EXC_CONT,79);
fwrite(EXC_CONT,83);
fwrite(EXC_CONT,13);
pause(0.063);
if EXC_CONT.bytesavailable~=0
    datoin=fread(EXC_CONT, EXC_CONT.bytesavailable, 'uchar');
    POS=[char(datoin(1:17))];
    fclose(EXC_CONT);
end
for i=1:n
    D(i)=sqrt((str2double(POS(2:8))-Prp(i,1))^2 + (str2double(POS(11:17))-Prp(i,2))^2);
end
for i=1:n
    D_ord(i)=D(1);
    for j=2:n

```

```

        if D_ord(i)>D(j)
            D=[D(j:n),D(1:j-1)];
            Prp=[Prp(j:n,:);Prp(1:j-1,:)];
            break
        end
    end
end
Prp=[Prp;Prp(1,:)]; %incremento do primeiro ponto pra fechar trajetória
n=size(Prp);
n=n(1,1);
valor_X=[num2str(Prp(:,1), '%.2f')];
valor_Y=[num2str(Prp(:,2), '%.2f')];
vec_MOVX=['MOV X'];
vec_Y=[' Y'];
for i=1:n-1
    vec_MOVX=[vec_MOVX;'MOV X'];
    vec_Y=[vec_Y;' Y'];
end
MOV=[vec_MOVX,valor_X,vec_Y,valor_Y];
for i=1:n
    comando (MOV(i,:))
    for j=1:100
        pause(0.8);
        fopen(EXC_CONT);
        flushinput(EXC_CONT);
        fwrite(EXC_CONT,80); %"POS"
        fwrite(EXC_CONT,79);
        fwrite(EXC_CONT,83);
        fwrite(EXC_CONT,13);
        pause(0.1);
        if EXC_CONT.bytesavailable~=0
            datoin=fread(EXC_CONT, EXC_CONT.bytesavailable, 'uchar');
            POS=[char(datoin(1:17))];
            fclose(EXC_CONT);
        end
        if (str2double(POS(2:8))==str2double(valor_X(i,:)) & (str2double(POS(11:17))==str2double(valor_Y(i,:))))
            break
        end
    end
end
comando('SVOF');

```