

# Usinagem de superfícies definidas em CAD

**Dilson José Aguiar de Souza**

Mestre pela Universidade Federal de Santa Maria - UFSM  
Programa de Pós-Graduação em Engenharia de Produção - PPGE  
Santa Maria, RS  
djasouza@gmail.com

## Resumo

O texto apresenta uma das três alternativas da pesquisa da dissertação de mestrado, realizada junto ao Programa de Pós-Graduação em Engenharia de Produção, da Universidade Federal de Santa Maria. Esse trabalho tem por objetivo demonstrar o desenvolvimento de ciclos avançados de usinagem com o uso de extração de pontos de entidades geométricas em 3D definidas em CAD, no auxílio de geração automática de códigos CN, para operações de desbaste e acabamento como um sistema CAD/CAM. Para essa alternativa pesquisada, fez-se a utilização da linguagem AutoLISP e como plataforma a utilização do software AutoCAD2000®. O programa CN foi utilizado na usinagem de uma matriz associada a um sólido gerado por malhas de pontos, em um Centro de Usinagem CNC. Os resultados mostram que a utilização do sistema é viável para uso em pequenas e médias empresas.

**Palavras-chave:** ciclos avançados, CAD, CNC, malha de pontos.

## Abstract

This text presents one of the options proposed for the Master's thesis research of the graduate program of Production Engineering of the Federal University of Santa Maria (UFSM, Brazil). The aim of this work is to demonstrate the development of advanced cycles for milling, through the use of point extraction of 3D geometric entities defined by CAD, to help in the automatic generation of CN codes, for cutting and finishing operations as a CAD/CAM system. With this aim, we used AutoLISP language and the platform of AutoCAD2000®. The CN program was used in the milling of a matrix, associated to a solid, generated by point webs, in a CNC Milling Center. The results show that the use of the system is feasible for small and medium-sized companies.

**Key words:** advanced cycles, CAD, CNC, point webs.

## 1. Introdução

A pesquisa de mestrado teve por objetivo propor uma metodologia para elaboração de rotinas em ambiente CAD que possibilitem a definição e armazenamento de ciclos avançados de usinagem, para geração automática de códigos CN (Controle Numérico). Para concretização disso, foram criadas e implementadas três alternativas: rotinas avançadas baseada em ciclos básicos de comando numérico em forma de banco de dados; rotinas para desenvolvimento de ciclos avançados (matriz de furos, furação circular, oblongos e cavidades irregulares), baseada em dados de entidades do CAD (Desenho Auxiliado por Computador) e processamento entidades definidas em 3D no CAD (usinagem de superfícies).

Das três alternativas pesquisadas foi escolhido o processamento de entidades definidas em 3D no CAD para expor nesse trabalho, apresentando a fundamentação teórica, a seqüência do desenvolvimento da extração de pontos para o desbaste e acabamento de uma superfície geométrica 3D gerada em ambiente CAD e os resultados dos testes finais, demonstrados em usinagem de peças exemplo em um Centro de Usinagem CNC (Controle Numérico Computadorizado).

A justificativa da pesquisa dessa alternativa é devido ao custo considerado elevado de sistemas CAD/CAM, principalmente para pequenas e médias empresas, custos estes devido a complexidade e versatilidade dos pacotes computacionais os quais tem por objetivo transformar dados de projeto em informações para a fabricação do produto gerando automaticamente os programas CN e também, a dificuldade de utilizá-los em setores industriais, pela carência de operadores qualificados.

Será apresentado o sistema desenvolvido, que visa mostrar a viabilidade de implementação da metodologia e motivar sua utilização, principalmente por pequenas e médias empresas, devido a seu baixo custo e conformidade com aplicações específicas em diversos setores da indústria.

## **2. Fundamentação Teórica**

### **2.1 Estrutura de um programa CN**

Um programa CN é composto por uma série de instruções seqüenciais relacionadas com a peça a ser usinada. Cada instrução preparatória é especificada em um formato que possa ser aceito, interpretado e processado por um sistema CNC, difundida atualmente como código G, disponível nos padrões EIA RS-274 e ISO 6983-88 (Pereira, 2003).

Os comandos de um programa CN são os responsáveis pelo acionamento de uma máquina CNC, através da informação de todas as etapas de fabricação para uma determinada operação de uma peça. Uma linha de comando de um programa CN pode ter informações sobre o movimento da ferramenta (movimento rápido, interpolação, etc.), informações tecnológicas (velocidade, avanço, etc.), ou informações que acionam funções auxiliares (ligar fluido de corte, eixo árvore, torre de ferramentas, etc.). A aquisição dessas informações envolve, sobretudo, os dados da peça a ser usinada, levando-se em conta as limitações da máquina, as características do CNC e da ferramenta.

Existem quatro termos básicos utilizados em programação CN. Estas palavras são a chave para entender a terminologia geral CN: caractere, palavra, bloco e programa (ROMI, 2002).

### **2.2 Ciclos padrões de usinagem**

Ciclos padrões de usinagem são subprogramas tecnológicos que permitem realizar universalmente uma certa operação de usinagem, tal como roscar com macho ou fresar um alojamento retangular (Siemens, 1997). Como exemplo tem-se para o comando MACH8 (ROMI, 1997), os ciclos G25 (quadrado ou retângulo de furos), G26 (fresamento de alojamento retangular, quadrado ou circular), G74 (rotação do sistema de coordenadas). Já para os comandos Sinumerik (Siemens, 1997) tem-se: ROT (rotação do sistema de coordenadas), HOLES1 (linha de furos), POCKET1 (fresamento de alojamento retangular).

Muitas funções de programação manual de máquinas CNC correspondem a ciclos de usinagem que simplificam o trabalho do programador. Esses ciclos concentram diversas operações que são especificadas através de parâmetros. Um exemplo típico é a execução de um alojamento ou cavidade retangular em fresadoras. Nesse caso, a usinagem é executada em diversos passes de desbaste para remoção do material interno à cavidade, e posterior operação de acabamento no contorno da mesma. Essas operações devem se repetir em diferentes planos, se a profundidade da cavidade for maior que a profundidade de corte da

ferramenta. Toda essa seqüência de trajetórias da ferramenta é gerada automaticamente pelo comando da máquina.

A tarefa do programador do equipamento é apenas introduzir no programa *CN* a função de ciclo de usinagem correspondente, preenchendo os valores dos parâmetros para as condições desejadas (Groover, 2001).

Em equipamentos modernos, ciclos mais complexos de usinagem podem ser desenvolvidos pelo próprio usuário. Para tanto, recursos avançados de programação são disponibilizados pelos fabricantes do comando da máquina. Essa técnica possibilita incluir no programa *CN* recursos como variáveis computacionais, expressões matemáticas, desvios condicionais e chamada de sub-rotinas. Apesar do potencial desses dispositivos de programação para aumentar a versatilidade do equipamento, seu uso não é amplamente explorado. Por isso, fazem-se programas para usar esses recursos.

Empresas de médio e grande porte optam, como estratégia para executar usinagens complexas, pela aplicação de sistemas de integração *CAD/CAM*, que, apesar de exigir maior investimento, são mais eficientes (Nanfara *et al.*, 1999). Para pequenas e micro empresas, o principal obstáculo para o uso de programação avançada é a dificuldade de manter profissionais qualificados.

### 2.3 Principais aplicações do CNC

Os comandos numéricos computadorizados são aplicados principalmente em: lotes médios e pequenos (que justifiquem os tempos de preparação); lotes grandes onde se queira maior eficiência em toda operação; peças com geometrias complexas; peças de grande precisão dimensional; peças com alto grau de acabamento da superfície; peças em que, se utilizando o sistema convencional, requeiram um grande número de refugos e retrabalhos; peças em que, se utilizando o sistema convencional, requeiram um ferramental de alto investimento ou de elevada complexibilidade de projeto e construção; peças em que exista um número variado e grande de usinagem, que requeiram um processo que seja desmembrado em diversas máquinas, e que se possam substituir todas essas máquinas e realizar esta operação em apenas uma máquina *CNC*. Comando Numérico Computadorizado é um equipamento eletrônico capaz de receber informações, compilar essas informações e transmiti-las em forma de comando à máquina operatriz, fazendo com que esta realize as operações na seqüência programada, sem a intervenção do operador (Machado, 1990).

### 2.4 Sistemas CAD

Sistema *CAD*, definido como o envolvimento do computador em atividades de desenho, muito freqüentemente é associado ao uso de sistema de gráficos de computador interativo. Há muitas vantagens em utilizar um sistema de *CAD*, tais como:

- aumento da produtividade do desenhista. Sistemas *CAD* ajudam o desenhista a reduzir seu tempo de desenho, editando, corrigindo e atualizando-os;
- melhora a qualidade do desenho permitindo ao desenhista executar mais, criando análise e considerando um número maior de alternativas de desenho;
- melhora a documentação dos desenhos. A produção intensa de um sistema de *CAD* é superior a

manual, com menor ocorrência de erros;

- criação de um banco de dados industrial relativo a materiais, dimensões e dados de fabricação.

## 2.5 Sistemas CAD/CAM

Sistemas CAD/CAM envolvem o uso de sistemas de computador para realizar certas funções em desenho e fabricação. Sistemas de CAD envolvem o uso de computadores para facilitar a função de desenho e sistemas de CAM para apoiar atividades de manufatura. Esses dois foram unidos em uma definição comum denominada CAD/CAM porque o segundo ato é como uma atividade continuada do primeiro, sendo ambos integrados em desenho e manufatura.

Um sistema CAD/CAM integrado pode ser definido como o sistema em que a fabricação e o projeto foram ligados através de dados e informações acumuladas no processo de projeto e transmitidos diretamente para os procedimentos de manufatura (Simões, 1999). Essas informações são elaboradas através de um pós-processador, sendo que é necessário um para cada comando, para gerar o programa CNC correspondente (IFAO, 1984). Redução em tempos de usinagem com otimização de códigos NC são uma das prioridades principais da Manufatura Auxiliada por Computador (CAM – *Computer Aided Manufacturing*) por desenvolvedores de pacotes de programas (Kaymakci *et al.*, 2006).

Por outro lado, funções de integração CAD/CAM também podem ser desenvolvidas na forma de aplicativos desenvolvidos no ambiente computacional de projeto (sistemas CAD). Para utilização dos sistemas CAD/CAM comerciais é necessário investir no desenvolvimento de rotinas para executar funções específicas de integração CAD/CAM (Da Silva e Bevilacqua, 2005). Implementar rotinas em CAD permite adaptá-las às características específicas do equipamento onde o sistema será aplicado (Da Silva e Souza, 2003; Souza, 2004).

## 2.6 Linguagens de Programação em CAD

A finalidade da linguagem de programação em CAD é a personalização do programa, para atender necessidades específicas, para usar melhorias desenvolvidas por terceiros e também para possibilitar o desenvolvimento próprio e a posterior comercialização.

A linguagem para uso de criação de rotinas avançadas de usinagem deve ter como possibilidade:

- capacidade de realizar operações matemáticas;
- ler informações do desenho (acesso a banco de dados das entidades);
- desenhar através de comandos;
- permitir entrada e saída de dados;
- processamento de arquivos.

Dentro dessas características, temos a linguagem *AutoLISP*, que foi utilizada para o desenvolvimento das rotinas, entre as várias linguagens existentes em ambientes CAD.

A *AutoLISP* deriva-se da *LISP* e é a segunda mais antiga linguagem de programação de alto nível ainda usada por computadores modernos. A *LISP* permite programação concisa para problemas complexos. Ela roda somente dentro do AutoCAD, por isso ele pode ser considerado o sistema operacional da *AutoLISP*.

A *AutoLISP* permite manipulação de listas, definir novas funções, trabalhar com entrada e saída de dados, efetuar operações matemáticas, expressões condicionais, *looping*, processamento externo de arquivos, acessar o banco de dados de desenho do AutoCAD, modificar dados, executar qualquer comando do CAD, chamar quadros de diálogo, etc.

As estruturas de dados são tanto programas quanto dados, tornando-as relativamente simples para construir ou modificar sub-rotinas dentro de um programa. A *AutoLISP* possui características de linguagem de programação estruturada (Matsumoyo, 2001).

A vantagem de usar uma linguagem baseada em *LISP*, é que ela é um processador de listas. Ele se encaixa naturalmente dentro de um sistema CAD, pois é baseado em pontos e vetores. Pontos podem ser realmente considerados listas, já que dois ou três números reais são combinados para formar um ponto de referência (X, Y, Z) para definir sua posição no espaço, o que representa uma lista de três números. Por exemplo, uma linha é definida por dois pontos. Então ela é uma lista contendo duas listas que descrevem os valores de coordenada do ponto.

A Linguagem *AutoLISP* evoluiu de forma a conter a maioria dos comandos básicos, tipos de entrada e capacidades da linguagem de programação *LISP*, assim como funções especiais adicionais para operações dentro do sistema gráfico do AutoCAD®.

Os dados estendidos podem ser anexados a uma seqüência de linhas informando à aplicação que estas linhas são constituídas de um determinado material que representam um determinado lado da peça. Estes dados são usados em combinação com os dados geométricos dos objetos para calcular as necessidades em termos de materiais, propriedades de encaixe para aproveitamento de chapas, caminhos de usinagem e outros dados de que o aplicativo necessite como informação. Esses dados estendidos fornecem aos desenvolvedores de aplicativos, um sistema de recuperação de informações com localização indexada (Kramer e Kramer, 1995).

Desde a criação da *LISP*, ela tem sido utilizada por cientistas da computação nos campos da inteligência artificial e do desenvolvimento de linguagem. Esta permite programação de objetivos simples para controle de robótica, caminhos de usinagem e é algumas vezes utilizada em sistemas de CLP avançado para controle de células de trabalho (Kramer e Kramer, 1995).

## 2.7 Ciclos de usinagem avançados

Ciclos de usinagem avançados são rotinas elaboradas para facilitar as tarefas de programadores de máquinas de comando numérico e usuários de CAD, pois elas envolvem repetições de tarefas que teriam que ser feitas individualmente para aplicações específicas ou que não estejam disponíveis no comando da máquina e que tenham que ser implementadas.

Sua implementação pode ser através de:

- recursos de programação do comando (programação parametrizada);
- sistemas CAD/CAM (estão prontas);
- rotinas em CAD;
- software independente.

No caso em estudo, os ciclos são implementados sob a forma de rotinas em CAD envolvendo a

repetição de comandos e ações que venham a satisfazer as necessidades do usuário. Estas rotinas são desenvolvidas através da utilização de linguagens de programação. Pode-se, desta forma, utilizar os próprios comandos existentes de uma máquina *CNC* ou através de extração de dados de entidades desenhadas (linhas, arcos, círculos) através de linguagem de programação.

### **3. Sistema para extração de pontos de uma superfície geométrica definida em CAD**

Para a seqüência do desenvolvimento da extração de pontos para o desbaste e acabamento de uma superfície geométrica gerada 3D em ambiente *CAD*, foi utilizado o recurso de elaboração de rotinas, através da linguagem de programação *AutoLISP* em plataforma *CAD*. A seguir são apresentados os passos utilizados no sistema do ciclo avançado para a extração dos pontos e a sua aplicação na usinagem de uma superfície 3D definida em *CAD*.

#### **3.1 Compensação para eixo Z considerando operação de acabamento**

Um procedimento bastante usual de definição de superfícies em sistemas computacionais de auxílio ao projeto – *CAD* é o recurso de desenho por especificação de malha de pontos. Nesses sistemas, os valores coordenados de cada ponto da malha que definem a superfície devem ser convertidos em dados geométricos que serão inseridos no programa que comanda os movimentos da máquina (programa *CN*). Na programação de equipamentos *CNC*, a geometria estabelecida no programa define a forma final da peça. Os posicionamentos da ferramenta contidos no programa *CN* referem-se a um ponto de referência da mesma. Na operação de fresamento, esse ponto é o centro da fresa. Portanto, para eliminar erros geométricos, é necessário realizar a compensação do raio da ferramenta, ou seja, a ferramenta deve percorrer a linha programada tangenciando-a pelo lado direito, ou esquerdo, conforme especificado.

Em Centros de Usinagem *CNC*, essa compensação é feita automaticamente através de funções específicas de programação da máquina. Neste estudo, para usinagem de superfícies 3D nesses equipamentos, a função de compensação do raio da ferramenta não é aplicável, pois a direção de compensação é normal à superfície e, portanto, difere do plano de trabalho da máquina (plano XY ou perpendicular ao eixo da ferramenta). A metodologia adotada nesta rotina é determinar a trajetória da ferramenta, para usinar uma superfície 3D definida em sistema *CAD*, através de uma malha de pontos. O processo considera operações de desbaste e acabamento com uma fresa de extremidade esférica, utilizada em um Centro de Usinagem *CNC* de três eixos. A malha é composta de faces retangulares, agrupadas no *CAD*, formando uma superfície que será a guia da ferramenta. Partindo dessa malha, da lista de pontos 3D dessa superfície e de uma esfera (sólido que representa a ferramenta, com raio definido pelo usuário) é iniciada a operação de compensação do raio. O procedimento apresentado a seguir é feito para todos os pontos de cada face, conforme descrito no fluxograma da Figura 1. Para cada ponto, a esfera que representa a ferramenta é posicionada inicialmente acima da superfície projetada, afastada de um incremento *I*, que é

de duas vezes o raio da esfera (Figura 2a). Esses pontos são obtidos, utilizando-se as coordenadas X e Y do ponto da lista, sendo a coordenada Z acrescida do incremento I. Com este posicionamento da esfera, a operação de interseção com o sólido "peça" é realizada. Como esse afastamento inicial representa uma limitação do sistema para valores de compensação do raio da ferramenta, a ocorrência de interseção na primeira operação é interpretada como seleção inadequada da ferramenta de corte. Nesse caso, o programa é encerrado sendo exibida uma mensagem de alerta. Não havendo interseção, a etapa seguinte é aproximar a entidade "ferramenta" da entidade "peça".

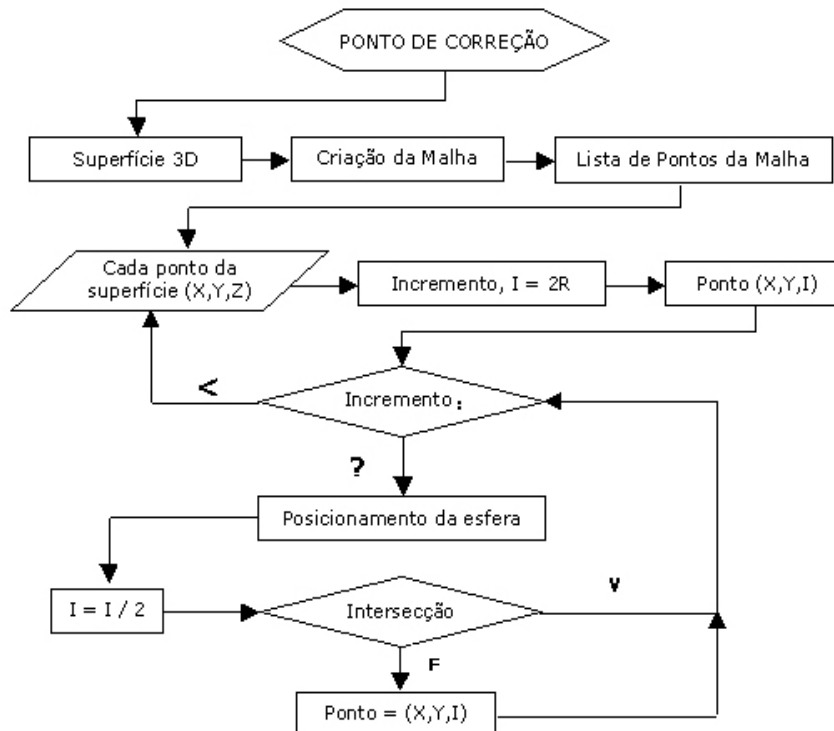


Figura 1 – Fluxograma demonstrativo da correção dos pontos compensar o raio da ferramenta.

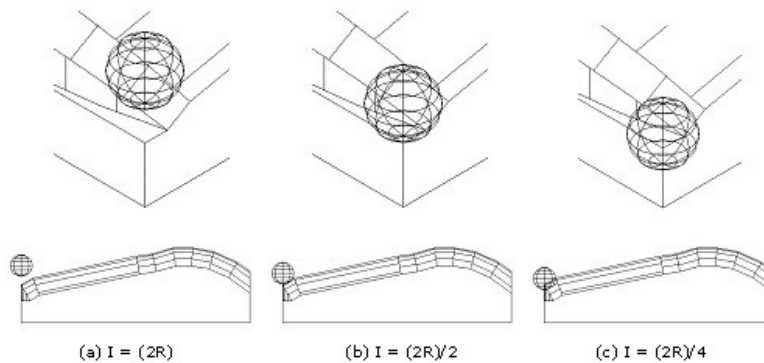


Figura 2 - Vistas isométricas e laterais das etapas de posicionamento da esfera.

O próximo posicionamento da esfera é definido alterando-se a coordenada Z do ponto anterior. A cada teste de interseção, o incremento I é dividido por dois, e subtraído da coordenada Z do ponto anteriormente definido (Figura 2b). Com o novo posicionamento da esfera, outra operação de interseção é realizada e o processo se repete até que ocorra um contato entre o sólido “peça” e a esfera “ferramenta”. Esse contato é identificado pela condição de criação de um novo sólido no desenho, como resultado da interseção da esfera com a peça.

Ao identificar uma interseção não nula (Figura 2c), a correspondente posição da esfera é desprezada e o ponto anterior volta a ser considerado como posição atual da ferramenta, sendo o novo incremento (metade do anterior) subtraído da coordenada Z deste ponto. Esse procedimento ocorre até que o incremento seja menor que o limite definido pelo usuário. O valor desse limite está diretamente relacionado com a precisão do ponto de tangência entre os dois sólidos processados. É o maior afastamento entre os sólidos, cuja posição relativa pode ser considerada como tangente.

Depois de encerrado o processo, a posição final da esfera corresponde ao ponto em que a coordenada Z da ferramenta deve ser compensada. Os pontos processados definem a nova trajetória (trajetória compensada) para a ferramenta na operação de acabamento e, são inseridos e armazenados na forma de lista de dados (lista para acabamento).

### **3.2 Otimização da trajetória da ferramenta**

A trajetória é realizada pela exclusão dos pontos comuns entre as faces e pela reorganização, através de uma lista auxiliar, onde os pontos de cada face adjacente identificadas são inseridos, conforme será detalhada a seguir. A otimização da trajetória da ferramenta é realizada através da seqüência de criação da malha plana, onde as combinações possíveis estão divididas em quatro casos:

Caso 1: Seqüência de pontos da face 1

Seqüência = P1, P2, P3, P4                      (Pontos = P1, P2, P3, P4)

Caso 2: Seqüência de pontos da face 2

Seqüência = P5, P6, P7, P8                      (Pontos coincidentes = P2 e P5; P3 e P8; P7 e P14)

Caso 3: Seqüência de pontos da face 3

Seqüência = P9, P10, P11, P12                      (Pontos coincidentes = P4 e P9; P3, P8 e P10; P11 e P16)

Caso 4: Seqüência de pontos da face 4

Seqüência = P13, P14, P15, P16                      (Pontos coincidentes = P3, P8, P10 e P13; P7 e P14; P11 e P16)

A lógica deste método é retirar os pontos coincidentes para poder trabalhar na organização da lista otimizada e poder definir uma seqüência de usinagem com menor tempo.

Estes pontos são retirados baseados no banco de dados da entidade *3DFace*, demonstrados na Figura 3.



|                                |                                                   |
|--------------------------------|---------------------------------------------------|
| (-1 . <Entity name: 1689dc0>)  | ; nome da entidade                                |
| (0 . "3DFACE")                 | ; tipo da entidade                                |
| (330 . <Entity name: 1689cf8>) | ; marcador de subdivisão de classe                |
| (67 . 0)                       | ; verifica ( <i>model</i> ou <i>paper space</i> ) |
| (410 . "Model")                | ; <i>model</i>                                    |
| (8 . "malha")                  | ; nome do <i>layer</i>                            |
| (100 . "AcDbFace")             | ; classe <i>face</i>                              |
| (10 323.826 268.854 0.0)       | ; ponto inferior esquerdo                         |
| (11 338.826 268.854 0.0)       | ; ponto inferior direito                          |
| (12 338.826 278.854 0.0)       | ; ponto superior direito                          |
| (13 323.826 278.854 0.0)       | ; ponto superior esquerdo                         |

Figura 3 – Dados de uma entidade Malha 3Dface.

Como a ferramenta só percorre as linhas determinadas pelos pontos de cada face, há uma restrição em relação ao comprimento desta face, podendo fazer com que partes da superfície não sejam usinadas (Figura 4a).

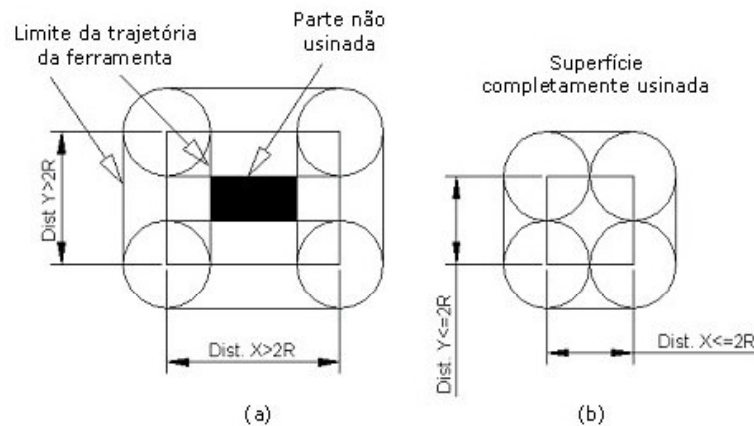


Figura 4 – Demonstração entre as dimensões em X e Y, para que toda a face seja usinada.

Essa restrição é determinada em função do raio R da ferramenta. Para que isso não ocorra, foi limitada a entrada de dados do usuário para que a largura de corte não seja maior que o duas vezes o raio da ferramenta, fazendo com que o comprimento (eixo X) e a largura (eixo Y) da malha sejam divididos pelo número de nós do eixo X e do eixo Y, comparando os resultados com o valor do raio da ferramenta, para que não haja partes sem serem usinadas na peça, conforme pode ser observado na Figura 4b.

### 3.3 Percurso para a operação de desbaste

A operação de desbaste é necessária quando a diferença entre a altura da peça bruta e a superfície a ser usinada é maior que a máxima profundidade de corte admitida para a ferramenta. Nesse caso, um ou

mais passes de usinagem são necessários para remover o excesso de material, antes da execução da operação de acabamento da superfície de projeto.

Neste trabalho, as passadas de desbastes são executados em planos paralelos ao plano horizontal (Figura 5) e afastados entre si de uma distância igual à profundidade de corte da ferramenta especificada pelo usuário. A estratégia de usinagem adotada foi o de zigue-zague (Figura 6).

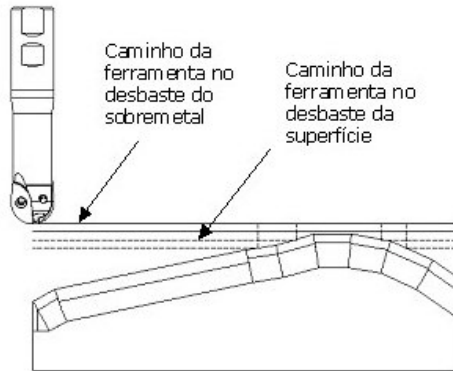


Figura 5: Percurso da ferramenta durante o desbaste

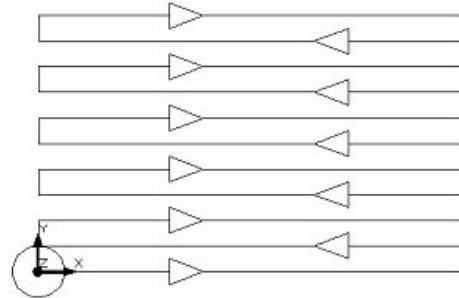


Figura 6: Estratégia de usinagem adotada para desbaste e acabamento

A lista de pontos para cada plano de desbaste é obtida a partir da lista de pontos que contém a compensação do raio da ferramenta (lista para acabamento), depois de realizado o processo de otimização da trajetória. Os pontos da lista de desbaste diferem da lista original apenas no valor da coordenada Z. Como cada plano de desbaste tem altura constante, a coordenada Z de todos os pontos é substituída por essa altura. Assim,

$(X, Y, Z)$  - ponto genérico da superfície  
 $\downarrow$   
 $(X, Y, Z_i)$  - ponto correspondente no plano de desbaste i.

Em cada plano de desbaste, são excluídos os pontos em que a coordenada Z da lista para acabamento for maior ou igual à altura do respectivo plano de desbaste. O fluxograma da Figura 7 mostra a lógica de implementação do algoritmo para a geração da lista de pontos da operação de desbaste.

### 3.4 Percurso para a operação de acabamento

Após o processamento dos pontos, ocorre a união das listas de acabamento otimizado e desbaste, sendo assim, os dados que definem a trajetória da ferramenta são enviados para o Centro de Usinagem CNC de três eixos. No desbaste, o movimento ocorre no plano XY e somente a troca de plano de desbaste (profundidade de corte) é no eixo Z. Já no acabamento, o percurso é realizado com a movimentação simultânea dos dois eixos (no caso eixos X e Z), seguindo o perfil da peça (Figura 8).

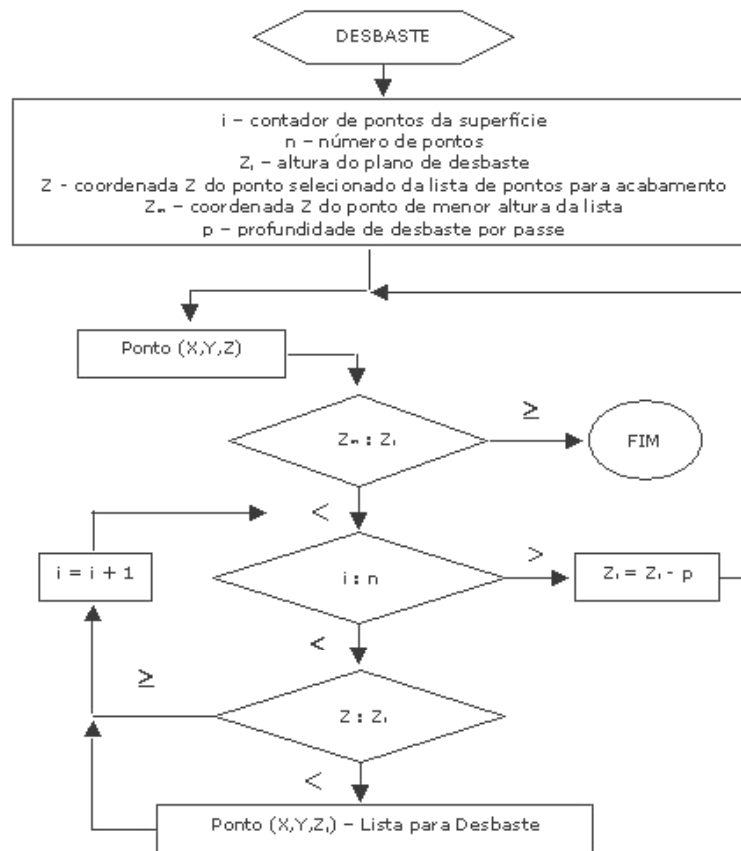


Figura 7: Fluxograma demonstrativo do processamento para os pontos de desbaste.

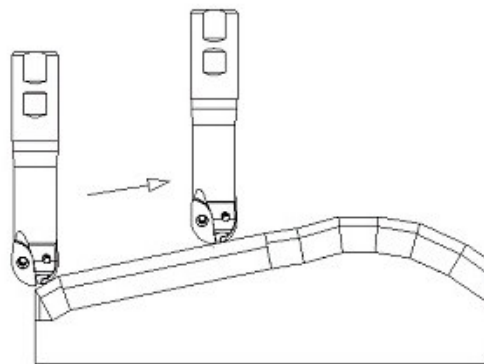


Figura 8: Percurso da ferramenta durante o acabamento seguindo o perfil da peça

#### 4. Usinagem de superfícies 3D

Para analisar o desempenho do sistema, foi definido no CAD um sólido que reproduz a parte superior de um *mouse* de computador, em que é criada, a partir do desenho, a malha plana que determinará os pontos a serem seguidos durante a usinagem. A peça foi usinada em um tarugo de bronze. Para reproduzir essa superfície pelo processo de usinagem em um Centro de Usinagem CNC, foi criada uma malha de pontos, mostrada na Figura 9a, através de pontos clicados nas extremidades da peça a usinar, no plano XY, a uma distância de recuo que é o ponto mais alto de recuo no eixo Z da ferramenta representada sob a forma de uma esfera nas Figuras 9b e 9c. Depois de criada a malha plana, o sistema utilizará os próprios pontos de definição da superfície, nas funções de interpolação linear do programa CN. Os movimentos obtidos dessa forma correspondem à operação de acabamento no processo de usinagem. As posições de movimentação da ferramenta no desbaste precisam ser calculadas nessa rotina. Esses pontos definem outros planos de movimento, em número e afastamentos entre si determinados como função da quantidade de material a ser removido em cada ponto da malha (diferença de altura entre a superfície livre da peça bruta e a superfície 3D) e da profundidade de corte admissível para a ferramenta.

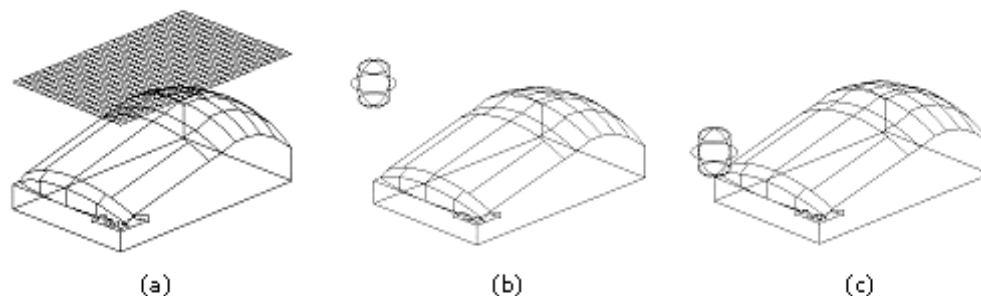


Figura 9 – Criação da malha (a), posição inicial (b) e posição final (c).

A malha utilizada para teste do sistema foi definida por 25 linhas e 25 colunas, com um total de 574 faces retangulares planas, formando 2.304 nós.

Para comprovar o resultado do processamento para o exemplo proposto, o programa CN gerado pelo sistema foi aplicado para a usinagem da matriz associada ao sólido. O resultado da usinagem, em forma de ressaltos, foi executada em um Centro de Usinagem CNC Discovery 560, através de operação de desbaste mostrada nas Figuras 12a a 12o e de acabamento, mostrada na Figura 12p.

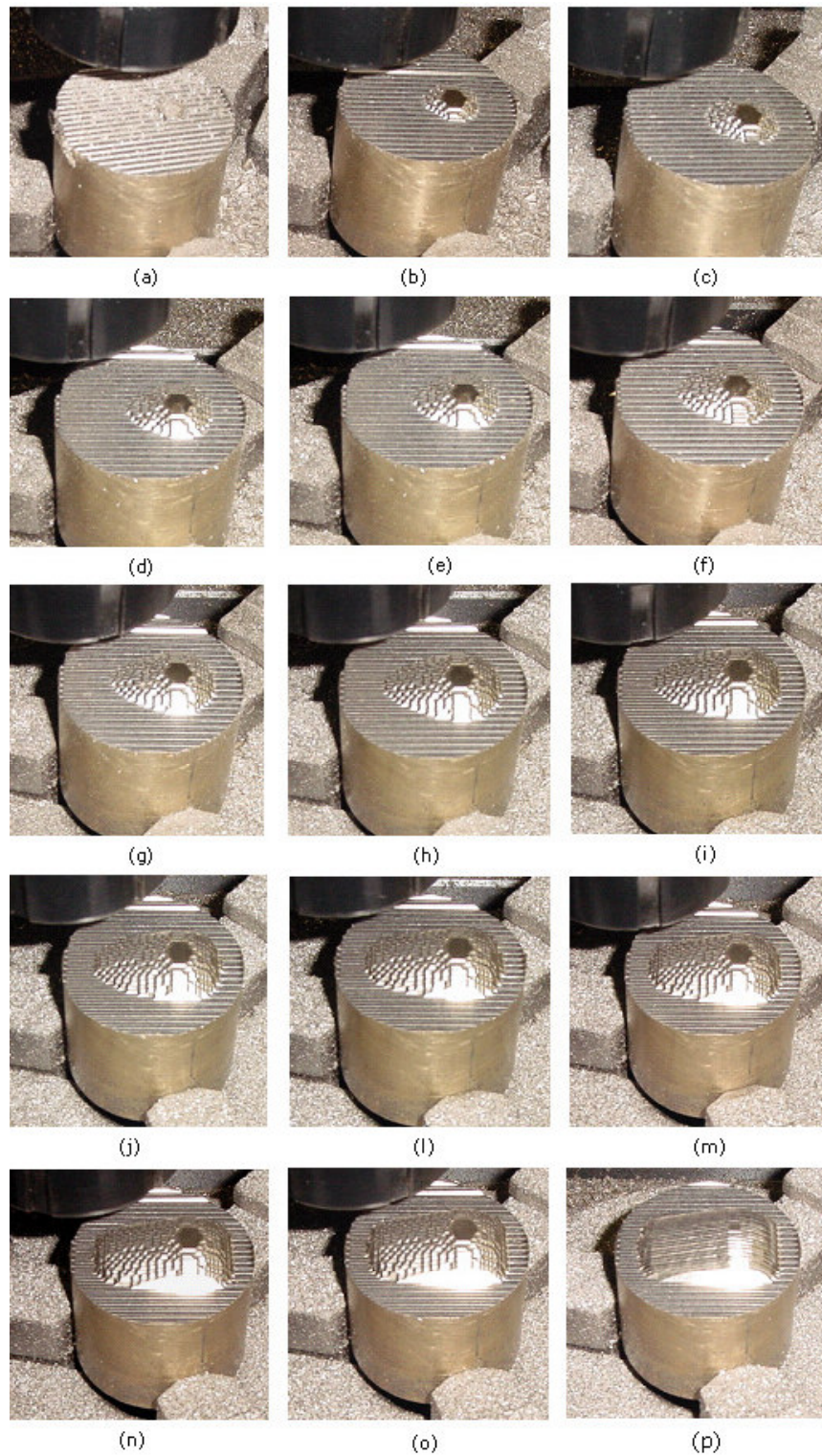


Figura 12 – Seqüência das passadas durante a usinagem.

## 5. Considerações finais

Uma alternativa para o desenvolvimento de ciclos avançados é explorar funções de processamento de entidades de desenho em sistemas CAD. Os algoritmos propostos constituem, portanto, funções de integração CAD/CAM. As vantagens que devem ser salientadas da alternativa apresentada, se comparadas à alternativa de aquisição de um sistema comercial para esse fim, são a simplicidade de operação e custo, já que ciclos avançados podem ser elaborados pelo usuário do sistema CAD e desenvolvidos considerando as particularidades do processo de fabricação que será aplicado. Sistemas CAD/CAM são programas sofisticados, que fornecem uma ampla gama de facilidades. Porém, para um ambiente produtivo com limitação no número de equipamentos CNC, apenas uma pequena parcela delas é necessária. Assim, elaborar as próprias rotinas avançadas dá versatilidade, pois pode conter apenas funções necessárias e desejáveis. Isso também simplifica o trabalho de programação de máquinas e tem como consequência um baixo custo de desenvolvimento. Desconsidera-se aqui o custo de aquisição do CAD, pois se considera embutido em custos de projeto e não, de fabricação.

Como o sistema não considera a aplicação de pós-processadores, características específicas de equipamentos CNC devem ser consideradas nas rotinas para geração do programa CN, não necessita de um pós-processador para cada tipo de máquina. Superfícies de formas livres podem ser usinadas com os códigos NC obtidos por pós-processamento (Wei e Lin, 2005). Isso leva, porém, à exigência de pequenas adaptações do modelo desenvolvido para aplicações em outros comandos numéricos de equipamentos CNC.

Também no caso de pequenas e médias empresas que possuam processos definidos de fabricação, a utilização de ciclos avançados de programação tem suas vantagens, pois esses podem ser definidos em relação à linha de fabricação das peças.

Mesmo sem a comprovação de resultados pela aplicação do sistema no meio produtivo, pode-se afirmar que a técnica utilizada, se convenientemente aplicada, leva a um aumento de produtividade em sistemas de manufatura com equipamentos CNC, através da redução do tempo de programação e a diminuição da probabilidade de ocorrência de erros. A facilidade de comunicação de dados entre o comando de uma máquina CNC e microcomputadores, aliada à contínua queda dos preços e avanços tecnológicos desses, são justificativas que reforçam a validade da proposta do trabalho.

## 6. Referências

- DA SILVA, A.D. e SOUZA, D.J.A. de. 2003. Geração de Ciclos Avançados de Usinagem a partir de Ciclos Básicos disponíveis em equipamentos CNC. *In: Congresso Brasileiro de Engenharia de Fabricação – COBEF 2003, 2*, Uberlândia. Anais do COBEF 2003, Uberlândia, ABCM, 8 p.
- DA SILVA, A.D. e BEVILACQUA, J.C. 2005. Implementação de Pós-Processadores em Sistemas CAD/CAM Dedicados. *In: COBEF - Congresso Brasileiro de Engenharia de Fabricação, 3*, 2005, Joinville. Anais do COBEF 2003, Uberlândia, ABCM, 10 p.

- GROOVER, M.P. 2001. *Automation, Production Systems, and Computer-Integrated Manufacturing*. New Jersey, Prentice-Hall, 856 p.
- IFAO - Institut für Angewandte Organisationsforschung. 1984. *Comando Numérico CNC: Técnica Operacional. Curso Básico*. São Paulo, EPU, 173 p.
- KAYMAKCI, M; LAZOGLU I. e MURTEZAOGLU, Y. 2006. Machining of complex sculptured surfaces with feed rate scheduling. *International Journal of Machining Research*, **1**:157-175.
- KRAMER, W. e KRAMER, D. 1995. *Programando em AutoLisp*. São Paulo, Makron Books, 274 p.
- MACHADO, A. 1990. *Comando Numérico Aplicado às Máquinas-Ferramenta*. 4ª ed., São Paulo, Ícone, 312 p.
- MATSUMOTO, E.Y. 2001. *AutoLISP 2002: Linguagem de Programação do AutoCAD*. São Paulo, Érica, 184 p.
- NANFARA, F.; UCCELLO; T. e MURPHY, D. 1999. *The CNC Workshop – A Multimedia Introduction to Computer Numerical Control*. Ontario, Canadá, Addison-Wesley, 378 p.
- PEREIRA, A.G. 2003. *Desenvolvimento e Avaliação de um Editor para Programação CN em Centros de Usinagem*. Curitiba, PR. Dissertação de mestrado. Universidade Federal do Paraná - UFPR, 122 p.
- ROMI. 1997. *Manual de Programação Mach8*. Discovery, 308, 131 p.
- ROMI. 2002. *Manual de Programação e Operação do Comando Siemens 810D*. Discovery, 560, 102 p.
- SIEMENS. 1997. *Instrução de Programação Sinumerik 840D/810D/FM-NC*. Ciclos, 245 p.
- SIMÕES, F.M.S. 1999. *Implementação de um Sistema CAD/CAM para Fresadora CNC a partir de Funções CAM Integradas no CAD*. Uberlândia, MG. Dissertação de mestrado. Universidade Federal de Uberlândia - UFU, 110 p.
- SOUZA, D.J.A. de. 2004. *Técnicas de implementação de ciclos expandidos e avançados de usinagem para máquinas com comando numérico*. Santa Maria, RS. Dissertação de mestrado. Universidade Federal de Santa Maria – UFSM, 113 p.
- WEI, E.J. e LIN, M.C. 2005. Study on general analytical method for CNC machining the free-form surfaces. *Journal of Materials Processing Technology*, **168**:408-413.

Submissao: 01/04/2007

Aceite: 02/05/2007