



APLICAÇÃO DE RECURSOS DE SISTEMAS DE PROJETO AUXILIADO POR COMPUTADOR PARA INTEGRAÇÃO COM A MANUFATURA

Alexandre Dias da Silva

Fabio Manoel Sá Simões

Fabio Ferreira Guimarães

Departamento de Engenharia Mecânica, Universidade Federal de Uberlândia - UFU

CEP 38.400-902 Uberlândia - MG

Email:adiass@mecanica.ufu.br;fmssimoes@engenheiro.com.br;ffguimaraes@rocketmail.com

Daniel Kupta Knoll

Universidade Federal de Santa Catarina, Departamento de Engenharia Mecânica, Caixa Postal 476, CEP 88040-900, Florianópolis, SC

***Resumo.** Este trabalho apresenta uma metodologia para o desenvolvimento de sistemas CAD/CAM dedicados. A proposta usa recursos de programação do CAD e dois exemplos de facilidades para geração automática de programas para equipamentos de Comando Numérico são implementados. São eles: criação de ciclos fixos de programação, para execução de cavidades elípticas, e definição da geometria do programa através de funções de desenho do CAD. O sistema proposto foi aplicado ao processo de usinagem por fresamento e os resultados mostram a viabilidade da tecnologia para o aumento da potencialidade de sistemas produtivos.*

***Palavras-chave:** Projeto Auxiliado por Computador, CNC, integração CAD/CAM, AutoCAD, Autolisp*

1. INTRODUÇÃO

Ferramentas computacionais são empregadas em modernas técnicas de produção, para promover a integração das fases de projeto e fabricação de produtos. Em sistemas CIM (sigla em inglês para Manufatura Integrada por Computador – *Computer Integrated Manufacturing*), para definição do produto, sistemas CAD (Projeto Auxiliado por Computador - *Computer Aided Design*) oferecem, além de uma grande variedade de recursos gráficos, a possibilidade de automatização de processamento de dados de projeto. Essas características tornam estes sistemas instrumentos de trabalho indispensáveis a qualquer projetista, permitindo-lhe não apenas um aumento de produtividade, mas também estimulando-lhe a criatividade. Entretanto, as especificações de projeto são muitas vezes limitadas, entre outros fatores, por recursos disponíveis no processo de fabricação.

Por outro lado, o desenvolvimento de equipamentos computadorizados aumenta a potencialidade dos processos, principalmente no que se refere à geometria do produto.

Exemplo disso são as chamadas máquinas CNC (Comando Numérico Computadorizado), que possuem um sistema programável para comando e controle dos movimentos de seus componentes. Com esse tipo de equipamento, é possível definir todo o conjunto de trajetórias desejáveis para uma determinada ferramenta, através de um programa codificado (programa CN) e armazenado no comando da máquina. Funções de programação para variadas formas geométricas dão versatilidade ao sistema.

A geração automática do programa CN é uma das funções do módulo do CIM definido como CAM (Fabricação Assistida por Computador - *Computer Aided Manufacturing*). A integração CAD/CAM tem como meta criar o programa CN a partir da geometria definida no CAD. Uma das técnicas aplicadas para executar essa tarefa é explorar a equivalência entre as entidades geométricas do CAD e as funções de programação do comando em equipamentos CNC. Porém, uma das dificuldades nessa abordagem é que poucas são as formas geométricas possíveis de serem definidas no CAD, que têm uma função correspondente no CNC.

O objetivo do presente trabalho é apresentar uma metodologia de integração CAD/CAM, para implementar novas funções de programação a equipamentos CNC. É proposto nesse trabalho uma forma de usar os recursos do CAD para gerar o programa CN, cuja geometria não é possível de ser diretamente definida como função de programação no comando CNC. A técnica foi implementada na forma de aplicativo do *software* AutoCAD R14 Autodesk, (1997), explorando-se seus recursos de programação proporcionados pela linguagem AutoLISP Krammer, (1995). Os elementos geométricos adotados para exemplificar a criação de novas funções de programação são cavidades elípticas e trajetórias criadas no CAD pela função de desenho para definir linhas de formato livre. Para testar a viabilidade do método, o sistema desenvolvido foi aplicado na programação de uma fresadora CNC de três eixos Romi, (1990), empregada no processo de usinagem para fabricação de componentes mecânicos.

2. CICLOS FIXOS DE PROGRAMAÇÃO

A programação de movimentos em equipamentos CNC é obtida através de funções de controle de deslocamentos dos eixos coordenados da máquina. Funções básicas nessa categoria são as interpolações lineares e circulares, onde é possível definir movimentos da ferramenta segundo uma reta ou arco de círculo, respectivamente, envolvendo o acionamento simultâneo de mais de um eixo da máquina. Em máquinas de Comando Numérico, aplicadas ao processo de fabricação por usinagem, outras funções de programação são fornecidas. Estas funções são denominadas de ciclos fixos de programação e se destinam a simplificar o trabalho do programador em tarefas repetitivas. Um exemplo característico é o ciclo de furação. Quando uma série de furos com mesmas características devem ser executados, a especificação dos parâmetros destes furos pode ser feita uma única vez. Estão incluídos entre esses parâmetros: velocidade de aproximação da ferramenta, distância de aproximação, velocidade de avanço na usinagem, profundidade do furo, velocidade de retração, plano de retração e movimentos intermediários, quando necessário. Em um ciclo de furação, para qualquer furo a ser executado, basta ao programador definir a posição de execução do furo através de funções de deslocamentos lineares. Também é possível associar ao ciclo de furação, outras funções como a que determina um conjunto de pontos. Assim, um conjunto de furos distribuídos, por exemplo, segundo uma malha retangular ou ao longo de uma circunferência, podem ser programados em apenas um bloco do programa CN.

Outra forma de programação compacta, disponível na maioria dos equipamentos CNC, é através da função que especifica um ciclo para executar uma cavidade retangular. No processo de usinagem, para produzir uma cavidade, a ferramenta necessita executar operações

de desbaste e acabamento. Para tanto, a operação deve ter início a partir do centro da cavidade, com operações de desbaste até a forma retangular final. Cada operação de desbaste corresponde à execução de retângulos sucessivos, de dimensões crescentes e semelhantes ao retângulo da cavidade desejada. Quando a profundidade da cavidade excede a profundidade de corte admissível da ferramenta, esse conjunto de operações deve ser repetido em diferentes planos.

Executar essa tarefa através de um ciclo fixo, dispensa ao programador a necessidade de cálculo de cada coordenada de todos os pontos a serem percorridos pela ferramenta. Os parâmetros dessa função incluem apenas as coordenadas da diagonal da cavidade, sua profundidade, sobremetal para desbaste e acabamento, número de passes para atingir a profundidade e velocidades de rotação e avanço da ferramenta.

A função criada através desse trabalho permite a execução de cavidades na forma elíptica, seguindo o mesmo formato da função para cavidade retangular. Nesse caso, os parâmetros que definem as dimensões da cavidade são as coordenadas do centro da elipse e o valor de seus raios. O método gera o trecho do programa CN, para execução da cavidade elíptica, contendo apenas interpolações lineares. Isso significa que a geometria desejada é discretizada em segmentos de reta, havendo portanto a necessidade de se introduzir um novo parâmetro para estabelecer a resolução do processo. Quanto menor a resolução estabelecida pelo usuário, maior a quantidade de pontos do programa CN. Porém, maior a precisão da geometria produzida.

3. CICLO DE CAVIDADE ELÍPTICA

No desenvolvimento de equipamentos de Comando Numérico, diferentes técnicas de controle têm sido usadas para programar deslocamentos lineares e circulares envolvendo mais de um de seus eixos de movimento. Uma dessas técnicas é conhecida como circuito de controle DDA (Digital Differential Analyzer), projetada para sistemas de controle na forma de circuitos eletrônicos Koren, (1983). O presente trabalho emprega o algoritmo da lógica DDA para interpolações circulares, adaptado para gerar a função de interpolação elíptica equivalente.

3.1. Interpolação Elíptica

Uma elipse, orientada segundo os eixos de um sistema de coordenadas, é caracterizada pela dimensão dos raios projetados sobre os eixos. Analisando-se no plano XY, as coordenadas no eixo X podem ser calculadas como função da circunferência definida pelo raio da elipse sobre o eixo Y e a relação entre os raios. Desta forma, conforme figura 1, tem-se:

$$(1) \quad X_{elip} = X_{circ} * R_x/R_y$$

Onde:

- X_{elip} => abscissa da elipse
- X_{circ} => abscissa da circunferência de raio R_y
- R_x => raio da elipse sobre o eixo X
- R_y => raio da elipse sobre o eixo Y

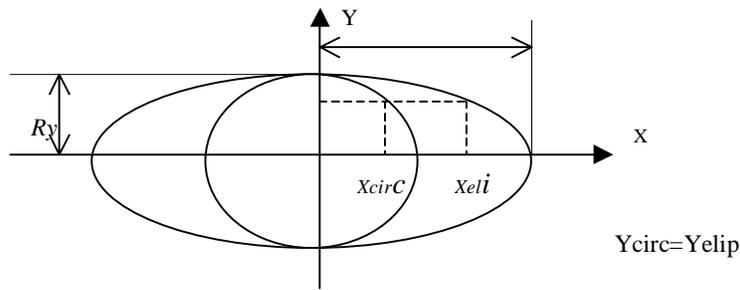


Figura 1 - Definição da elipse

Partindo-se desse princípio, como meio de definição da elipse usou-se o algoritmo de interpolação circular de circuito DDA para obter as coordenadas nos eixos X e Y da elipse. O algoritmo DDA da circunferência no primeiro quadrante, adaptado para geração da forma elíptica, pode ser visto no fluxograma da figura 2. O objetivo da lógica do sistema é identificar a necessidade de deslocamentos incrementais nas direções coordenadas, de tal forma que a ferramenta percorra uma trajetória que tende a seguir a forma elíptica ideal. Esse incremento, é um dado de entrada e caracteriza a resolução do sistema, que é inversamente proporcional à precisão geométrica da cavidade. Nesse fluxograma, as variáveis Xc , Yc , Rx , Ry e Res representam as coordenadas do centro da elipse, os raios segundo os eixos coordenados e a resolução, respectivamente.

Os parâmetros qx , px , qy , py são variáveis que equivalem a registradores no circuito DDA e que têm a função de identificar a necessidade de geração de incrementos na sequência adequada. Conforme observado no fluxograma, as coordenadas X são incrementadas quando o acumulador qx atinge o valor limitado pelo raio da elipse na direção Y (Ry). As coordenadas da elipse ($Xelip$) são calculadas como função da relação entre os raios, conforme observado na equação (1). O acumulador qy tem a mesma função para o eixo Y . Nesse caso, como as coordenadas da elipse na direção Y são iguais às coordenadas da circunferência com este raio, os valores calculados são as próprias coordenadas da elipse nessa direção ($Yelip$).

Considerando $qx = qy = py = 0$ e $px = Ry$ como valores iniciais, e as condições $Xelip = Xc + Rx$ e $Yelip = Yc$, como término do algoritmo, o fluxograma da figura 2 descreve a sequência de pontos coordenados para deslocamentos da ferramenta com origem em $(Xc, Yc + Ry)$ e final do movimento em $(Xc + Rx, Yc)$, no sentido horário. Com isso, obtém-se os pontos da elipse no primeiro quadrante.

Cada alteração nas coordenadas X e/ou Y gera um ponto que deverá ter um bloco correspondente no programa CN. Todo o ponto coordenado gerado pelo algoritmo é armazenado em uma estrutura de dados na forma de lista (*lista_pontos*), que ao final do algoritmo deverá conter todos os pontos necessários para execução da cavidade elíptica.

A elipse completa é obtida a partir dos pontos gerados no primeiro quadrante. A determinação dos pontos no quarto quadrante (X positivo e Y negativo em relação ao centro da elipse) considera apenas a inversão das coordenadas Y do primeiro quadrante. Para cada ponto do primeiro quadrante, o ponto correspondente no quarto quadrante tem os seguintes valores coordenados:

$$(2) \quad P4(i) = (Xelip1(i), -1 * Yelip1(i))$$

Onde: $P4(i)$ => pontos da elipse no quarto quadrante
 $Xelip1(i)$ => abcissa dos pontos no primeiro quadrante
 $Yelip1(i)$ => coordenadas no primeiro quadrante

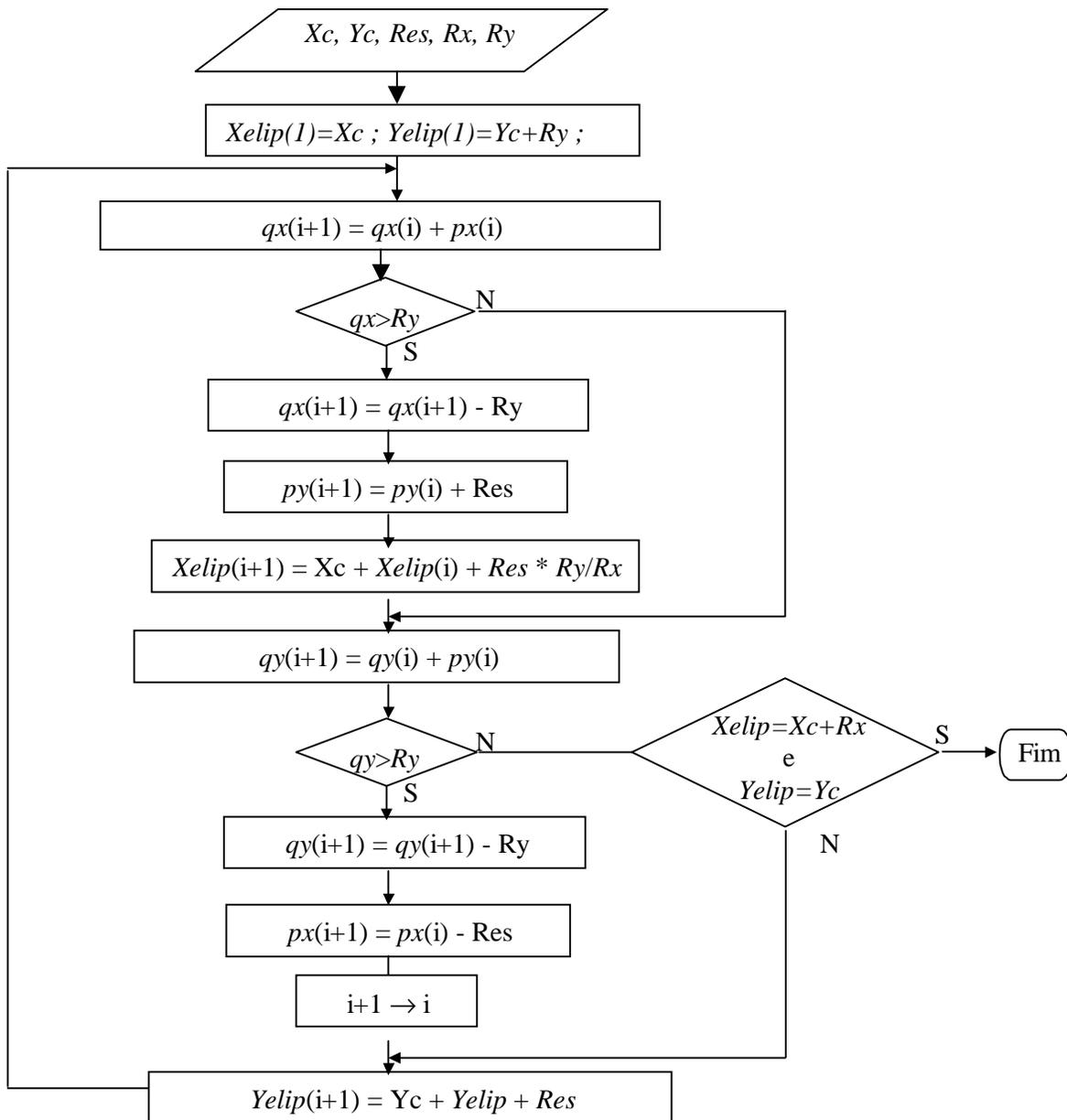


Figura 2 – Lógica DDA para interpolação elíptica

Como o conjunto de pontos na *lista_pontos* são armazenados de tal forma que cada novo ponto é sempre introduzido no início da lista, esses pontos ficam ordenados na sequência inversa à que foram gerados. Para que esta lista contenha também os pontos do quarto quadrante, deve-se seguir esse mesmo princípio. Dessa forma, para cada ponto da *lista_pontos*, o ponto correspondente no quarto quadrante é calculado conforme equação 2 e introduzido no início da mesma lista. Após varrer todos os pontos do primeiro quadrante, a lista resultante conterá os pontos de uma semi-elipse, ordenados da posição $(Xc, Yc - Ry)$ até $(Xc, Yc + Ry)$.

A complementação da *lista_pontos*, para conter os dados da elipse completa, é feita considerando os pontos já obtidos, através de um espelhamento em relação à um eixo orientado na direção Y e que passa pelo centro da elipse. Isso corresponde a um procedimento semelhante ao descrito anteriormente, ou seja:

$$(3) \quad P32(i) = (-1)^i * Xelip14(i), Yelip14(i)$$

onde: $P32(i)$ \Rightarrow pontos da elipse nos quadrantes 3 e 2
 $Xelip14(i)$ \Rightarrow abcissa dos pontos nos quadrantes 1 e 4
 $Yelip14(i)$ \Rightarrow coordenadas nos quadrantes 1 e 4

Como resultado dessa operação, tem-se uma lista de pontos que discretizam uma elipse em segmentos de reta.

3.2. Operações de Desbaste e Acabamento

O item anterior descreve, de forma genérica, a obtenção de pontos de uma elipse. A usinagem de uma cavidade elíptica, no processo de fresamento, requer a definição de pontos que correspondem ao deslocamento do centro da ferramenta. Tem-se que considerar também, que para atingir a forma final, operações de desbaste e acabamento são necessárias. Além dos dados tecnológicos, como velocidades de corte e avanço, os parâmetros de corte que dependem da posição relativa entre peça e ferramenta são a penetração de trabalho (ae) e a profundidade de corte (ap) Stemmer, (1989). Considerando os eixos coordenados da máquina-ferramenta, ae é medido no plano XY e ap na direção do eixo Z (figura 3). Segundo a teoria da usinagem, ambos os parâmetros devem ser reduzidos para operações de acabamento (ae_{acab} e ap_{acab}).

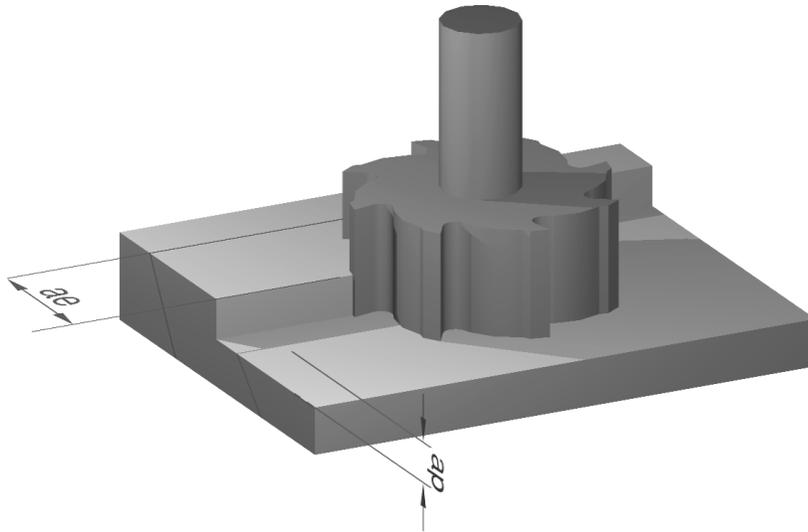


Figura 3 – Parâmetros de usinagem: profundidade de corte (ap); penetração de trabalho (ae)

Para cada operação de desbaste, executados ao longo do eixo Z, o sistema gera uma sequência de desbastes e acabamento no plano XY. As operações de desbastes iniciam com o posicionamento da ferramenta no centro da elipse. Após o avanço na direção Z para definir a profundidade de corte, a ferramenta desloca-se segundo o eixo Y uma distância equivalente à ae . Com início nesse ponto, uma elipse semelhante à elipse da cavidade desejada é executada. Na sequência, novo incremento em Y é definido e o procedimento se repete para todas as

operações de desbaste. Para a operação de acabamento, o último incremento em Y é especificado por *ae_acab*.

Quando a profundidade da cavidade (*ap_total*) é maior que a máxima profundidade admitida pela ferramenta, também na direção Z serão executadas operações de desbaste e acabamento. Nesse caso, toda a sequência estabelecida anteriormente é repetida em diferentes planos Z. Também de forma análoga, o último avanço nessa direção deve ser definido pelo parâmetro *ap_acab*.

As variáveis *ap_acab* e *ae_acab* são dados de entrada do sistema. O número de passes de desbaste, tanto no plano XY como na direção Z, depende de valores selecionados para *ap* e *ae* e é dado por:

$$I = (R - d_{acab}) / d \quad (4)$$

onde: $I \Rightarrow$ número de desbastes

$R = R_y$ e $d = ae$, para desbastes no plano XY

$R = ap_{total}$ e $d = ap$, para desbastes na direção Z

Os valores de *ap* e *ae* são recalculados, para que o resultado da expressão 4 seja ajustado a um número inteiro.

4. ENTIDADE GEOMÉTRICA COMO ENTRADA DE DADOS

A elaboração de um sistema de integração com a fabricação, através do desenvolvimento de aplicativos no CAD, deve permitir que elementos de desenho possam ser transformados no programa CN. A linguagem de programação AutoLISP, adotada nesse trabalho, possui funções para acessar o banco de dados de desenho do CAD. Essa característica foi utilizada para obter as informações necessárias à fabricação, de entidades projetadas no CAD. Para demonstrar a possibilidade desse procedimento, foi desenvolvido uma rotina que converte um elemento definido por função de desenho do CAD no respectivo programa CN. O elemento selecionado é o gerado pela função que cria linhas de formato livre (função *sketch*). Com essa função obtém-se uma sequência de segmentos de reta, conforme movimentação do dispositivo *mouse*. Para qualquer elemento de desenho criado, a correspondente entidade possui informações que permitem sua descrição. Para a entidade *reta*, os parâmetros geométricos são as coordenadas dos pontos iniciais e finais. Os dados de saída da subrotina são um conjunto de pontos alocados em uma lista com a mesma sequência definida durante a elaboração do desenho. No programa CN, cada interpolação linear é definida apenas pelas coordenadas do ponto meta. O ponto de origem do movimento é sempre a posição atual da ferramenta. Na lista de pontos, obtida com o algoritmo, o ponto inicial do primeiro segmento de reta é utilizado para identificar o movimento inicial da ferramenta. A partir desse ponto, os pontos finais de cada entidade são adquiridos para a geração do programa CN.

5. RESULTADOS

A metodologia proposta deve resultar no programa CN, tanto para o caso em que se utiliza algoritmo específico (cavidade elíptica) como pela transformação de uma geometria definida no CAD, esse programa é obtido pela conversão de uma lista de pontos, em código de programação do equipamento CNC. Para efeitos de entrada de dados no Comando via

computador, este programa deve estar na forma de um arquivo texto. Para executar essa tarefa no ambiente AutoLISP, funções de conversão de variáveis e manipulação de arquivos são empregadas. A conversão de variáveis é necessária para transformar os valores reais contidos na lista de pontos coordenados em variáveis do tipo *string*. A cada ponto da lista corresponde um bloco do programa CN. Como esse programa contém apenas interpolações lineares, basta acrescentar a essas variáveis os códigos de funções de movimentação para os correspondentes eixos da máquina.

Na inicialização do arquivo *ProgrCN.txt*, informações que especificam os modos de operação da máquina são introduzidos (unidades, plano de interpolação, definição da origem do sistema de coordenadas, seleção de ferramentas, etc.). Após inserir todos os blocos do programa CN, o código da função que identifica o fim do programa (M02) é acrescentado e o arquivo *ProgrCN* pode então ser enviado ao comando da máquina, ou armazenado para utilização posterior.

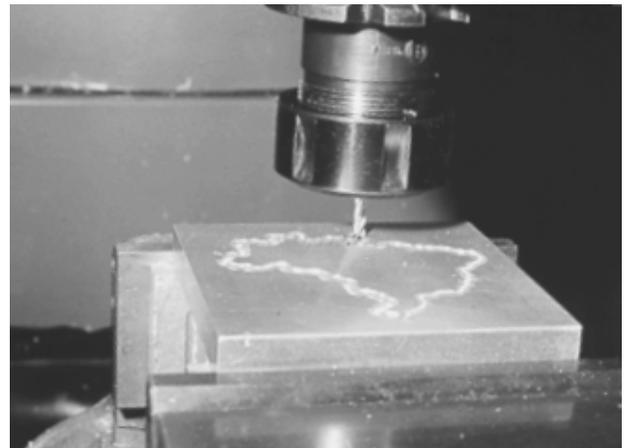
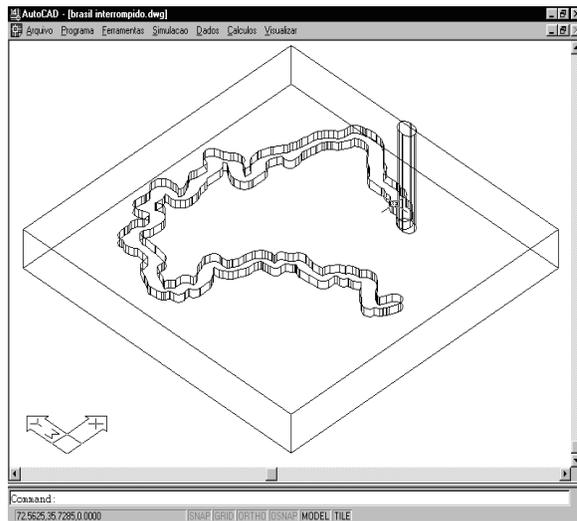


Figura 4 - Simulação de um programa gerado pela função *sketch*.

Figura 5 – Fotografia da máquina em operação: execução do programa CN.

A figura 4 mostra a simulação de um programa (mapa do Brasil), gerado a partir de uma geometria definida no CAD através da função de desenho para formato livre (*sketch*). Essa simulação foi obtida através do sistema descrito em *Da Silva et al, (1997)*. O programa CN do exemplo contém 195 blocos de interpolações lineares e a foto da figura 5 apresenta a máquina em operação para sua execução.

O quadro de diálogo da figura 6 é a forma de entrada de dados da função para realizar o ciclo de cavidade elíptica. A execução da cavidade, para os parâmetros exemplificados no quadro, é mostrado na figura 7. A ferramenta nessa simulação se encontra em posição correspondente à usinagem de acabamento na direção da profundidade da cavidade e no último desbaste no plano XY. O item “desbaste recalculado”, no plano XY ou na direção Z, tem como objetivo manter esse valor constante em todas as operações de desbaste.

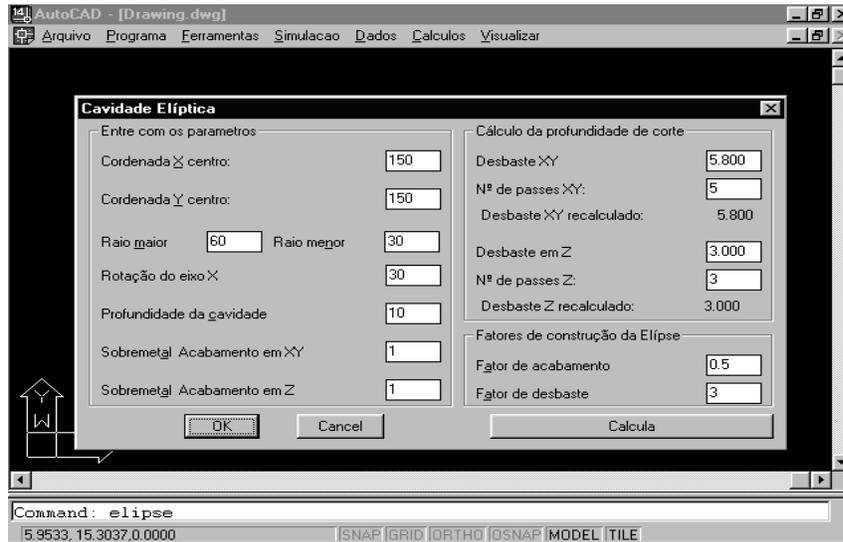


Figura 6 - quadro de diálogo para a função de ciclo de cavidade

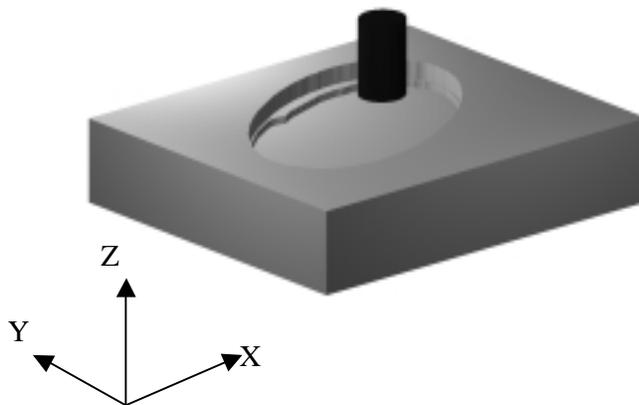


Figura 7 – Simulação da operação de usinagem

6. CONCLUSÃO

Os resultados do trabalho mostram a viabilidade de desenvolvimento de sistemas de integração CAD/CAM, a partir de recursos disponíveis em sistemas CAD. As vantagens que devem ser salientadas do projeto de sistema próprio, se comparado à aquisição de sistemas comerciais, são a versatilidade, o custo e a eficiência. Sistemas comerciais de integração CAD/CAM são programas sofisticados, que fornecem uma ampla gama de facilidades. Porém, a maioria dos usuários usa apenas uma pequena parcela delas. Projetar um sistema dedicado dá versatilidade no sentido em que pode conter apenas funções necessárias e desejáveis. Isso também simplifica o trabalho de programação e tem como consequência um baixo custo de

desenvolvimento. Desconsidera-se aqui o custo de aquisição do CAD, pois admite-se esse embutido em custos de projeto, e não de fabricação. A técnica proposta também elimina as dificuldades dos sistemas CAD/CAM convencionais, referentes ao desenvolvimento de pré-processadores e pós-processadores. Os primeiros são dispensados porque o sistema trabalha no próprio ambiente do CAD, não havendo portanto necessidade de interpretação de arquivos de desenho. Pós-processadores ficam implícitos no sistema, pois características específicas de equipamentos CNC são consideradas na etapa de geração do programa CN. Isso leva, porém, à exigência de adaptações do modelo para aplicações em outros equipamentos.

A principal desvantagem do método proposto é que o programa CN gerado pode atingir tamanho significativo, como consequência da definição ponto a ponto da interpolação elíptica ou de uma resolução muito pequena na elaboração do desenho do CAD. Isso exige meios apropriados para entrada automática do programa na memória do comando CNC. A forma mais indicada para isso é a transmissão direta de dados do computador para o comando da máquina, via porta de comunicação. No trabalho proposto, essa comunicação foi efetuada pelo sistema Dataholder Romi, (1996), fornecido pelo próprio fabricante do equipamento, servindo como interface de comunicação entre computador e comando. Com esse recurso, elimina-se também eventuais problemas ligados às limitações de memória do comando CNC, pois ele permite que o programa CN rode diretamente da memória do computador. Considerando a disponibilidade dos sistemas de comunicação de dados, independente do fabricante do comando, a viabilidade do sistema proposto pode ser comprovada, visto que suas limitações assim são eliminadas.

7. REFERÊNCIAS

- Autodesk Inc., 1997“AutoCAD Release 14 User’s Guide”, .
- Kramer, W., 1995, “Programando em AutoLISP”, Ed. Makron Books, .
- Indústrias ROMI S.A., 1990,“Fresadora CNC Interact 4 – Manual de Programação”, .
- Koren Y., 1983,“Computer Control of Manufacturing Systems”, McGraw-Hill,,
- Stemmer, E. C., “Ferramentas de Corte”, Imprensa Universitária UFSC, 1989.
- König, W., 1990,“Fertigungsverfahren Band 1: Drehen, Fräsen, Bohren”, VDI Verlag, .
- Da Silva, A.D., Simões, F.M.S., Guimarães, F.F., de Paula, A.A., 1997 “Aplicação de um Sistema CAD para Geração e Simulação de Programas para Fresadoras CNC”, COBEM 97.
- Indústrias ROMI S. A.,1996 “Dataholder: Manual de Instruções”.