

BOLSISTA: ALI ANTONIO MANFRIN ABDALLAH
ORIENTADOR: Prof. Dr. DALBERTO DIAS DA COSTA

**DESENVOLVIMENTO DE UMA INTERFACE PARA
INTERPRETAÇÃO E REPRESENTAÇÃO DE INFORMAÇÕES
GEOMÉTRICAS ORIUNDAS DE UM SISTEMA CAD**

Relatório Técnico-Científico entregue à Co-
ordenação de Iniciação Científica da Pró-
Reitoria de Pesquisa e Pós-Graduação da
Universidade Federal do Paraná, referente
ao período de agosto de 2000 a agosto de
2001.

CURITIBA
Agosto de 2001

BOLSISTA: ALI ANTONIO MANFRIN ABDALLAH
ORIENTADOR: Prof. Dr. DALBERTO DIAS DA COSTA

**DESENVOLVIMENTO DE UMA INTERFACE PARA
INTERPRETAÇÃO E REPRESENTAÇÃO DE INFORMAÇÕES
GEOMÉTRICAS ORIUNDAS DE UM SISTEMA CAD**

Bolsa PIBIQ/CNPq
Nº BANPESQ/THALES: 1997005032

Relatório Técnico-Científico entregue à Co-ordenação de Iniciação Científica da Pró-Reitoria de Pesquisa e Pós-Graduação da Universidade Federal do Paraná, referente ao período de agosto de 2000 a agosto de 2001.

CURITIBA
Agosto de 2001

SUMÁRIO

RESUMO.....	iv
1. INTRODUÇÃO	1
2. MODELO COMPUTACIONAL DO CAPP.....	2
3. INTEGRAÇÃO CAD/CAPP	3
4. DESCRIÇÃO DA METODOLOGIA PROPOSTA	5
4.1 OBTENÇÃO DA DIREÇÃO PREFERENCIAL DE SECCIONAMENTO	5
4.2 DEFINIÇÃO DOS PONTOS DE SECCIONAMENTO.....	7
4.3 EXTRAÇÃO DAS SEÇÕES	8
4.4 RELAÇÕES DE PERTINÊNCIA ENTRE CONTORNOS FECHADOS	8
4.5 CLASSIFICAÇÃO DAS SEÇÕES.....	12
4.6 ALGORITMO PARA FRESAMENTO PERIMETRAL DE UM CONTORNO	13
4.7 OPERAÇÕES DE TORNEAMENTO	18
4.8 OPERAÇÕES DE FURAÇÃO	19
4.9 ANÁLISE DOS RESULTADOS	19
5. INTERFACE COM O AUTOCAD.....	21
6. PROGRAMAÇÃO ORIENTADA A OBJETOS.....	23
7. EXEMPLO	24
8. CONCLUSÕES.....	25
9. REFERÊNCIAS BIBLIOGRÁFICAS.....	26
10. BIBLIOGRAFIA AUXILIAR	27
11. ASSINATURAS	28

RESUMO

A seleção das máquinas-ferramenta a serem utilizadas na fabricação de um determinado produto é uma das tarefas mais importantes no planejamento de usinagem. Devido à sua complexidade, é necessário um profissional extremamente qualificado e experiente, visto que uma escolha adequada resulta, entre outras coisas, em melhor qualidade e maior produtividade. O objetivo deste trabalho é desenvolver um módulo que propicie a ligação entre um sistema de planejamento de processos (CAPP) e um sistema CAD comercial. Com esta interface será possível recuperar informações geométricas do produto, transformando-as numa representação cinemática que auxilie na escolha automática das máquinas ferramentas. A metodologia empregada consiste na análise de vistas e seções do modelo tridimensional e na identificação, para cada seção, das direções preferenciais dos movimentos de aproximação, avanço e corte. Empregando regras de produção a peça é dividida em volumes intermediários e, para cada região, são extraídas seções representativas, resultando sempre em círculos e/ou conjuntos de linhas e arcos ligados entre si formando uma figura fechada. Cada seção é analisada individualmente e classificada em *contorno simples*, *múltiplo contorno simples*, *contorno composto* ou *múltiplo contorno composto*. Em seguida são aplicados algoritmos que associam conjuntos de movimentos cinemáticos do par peça-ferramenta capazes de gerar a geometria final. Um protótipo implementado em *MS Visual Basic*[®] permite o reconhecimento automático de regiões usináveis por fresamento perimetral, furação e peças típicas de revolução, além da extração de parâmetros como volume e dimensões máximas. Estas informações serão analisadas pela máquina de inferência do CAPP, que confrontando as restrições impostas por outros recursos (ferramentas, dispositivos de fixação, condições de corte) e metas do sistema de manufatura (tempo e custo), selecionará as máquinas-ferramenta mais adequadas.

1. INTRODUÇÃO

Os processos de fabricação por meios mecânicos, em especial os processos de usinagem, têm participação significativa em nossa vida cotidiana. Como exemplo, pode-se citar dentre vários, a produção de eletrodomésticos, eletroeletrônicos, automóveis, equipamentos cirúrgicos e equipamentos para a indústria alimentícia.

Desde a revolução industrial que tais processos, também conhecidos como processos de manufatura, vêm sendo executados pela combinação entre homem, máquina e materiais. Até hoje a participação humana aplica-se à mão-de-obra direta, ou seja, movimentação dos materiais e, principalmente, com conhecimentos (o *saber fazer*).

A partir da metade deste século, com o surgimento e constante expansão dos sistemas de informação baseados em computadores, observa-se uma redução gradativa da contribuição humana para a execução de processos de fabricação. Desde o surgimento da tecnologia CNC (*Comando Numérico Computadorizado*) e, mais recentemente, com o avanço da robótica, que a mão-de-obra direta vêm desaparecendo das fábricas. Operações, tais como a carga e descarga de peças numa máquina-ferramenta, o controle da trajetória de ferramentas, soldagem e vazamento de metais em fundições são, em fábricas modernas, realizadas por máquinas programáveis.

Pelo lado do *saber fazer*, a participação humana ainda é condição necessária. Isto é fortemente evidenciado numa atividade conhecida como planejamento de processos. Esta pode ser definida como a determinação sistemática dos métodos utilizados para a fabricação de um produto de forma econômica e competitiva. Tal tarefa é delegada a técnicos experientes – engenheiros na maioria dos casos. Segundo Da Costa (1997), outro fator que deve ser considerado é que um planejamento completo, além de levar muito tempo, requer para isto um profissional com larga experiência. No entanto, esta experiência não pode ser transmitida a um profissional mais jovem em um curto intervalo de tempo. E ainda, quando esse especialista se aposenta, ele leva consigo todo esse conhecimento.

Entretanto, a comunidade científica vem sustentando a tese de que o ser humano poderá ser substituído em atividades de planejamento. Isto decorre, em parte, pelo expressivo progresso em técnicas de inteligência artificial para representação de conhecimentos e na solução para problemas de busca.

Nesta mesma linha, devido à escassez de técnicos experientes, as indústrias americanas e européias vêm investindo grandes somas num campo da pesquisa conhecido como *Planejamento de Processos Auxiliado por Computador* (CAPP) [Van Houten, 1990 e Elmaraghy, 1993].

Apesar de alguns problemas de planejamento, entre eles o planejamento de processos de fabricação, serem considerados intratáveis por completo, isto é, serem *NP-completos* [Sarna e Wright, 1998], os avanços realizados até agora são promissores.

A automação do planejamento de processos representa um desafio para a comunidade científica devido à grande quantidade de detalhes a serem definidos antes que o plano possa ser executado. Além disso, o conhecimento necessário a este detalhamento dificilmente pode ser modelado matematicamente [Elmaraghy,

1993]. Segundo Alting e Zhang (1989), o planejamento de processos para peças fabricadas por usinagem pode ser sintetizado nas seguintes etapas, independente se realizado pelo ser humano ou auxiliado por computador:

1. Interpretação das informações geométricas do produto;
2. Seleção dos processos de usinagem;
3. Seleção das máquinas-ferramenta;
4. Determinação dos dispositivos de fixação e superfícies de referência;
5. Determinação da seqüência de operações;
6. Seleção dos dispositivos de inspeção;
7. Determinação das tolerâncias de produção;
8. Determinação das condições de corte;
9. Cálculo dos tempos de usinagem.

Tradicionalmente, o plano de processos é gerado na forma de um roteiro que permitirá à engenharia de produção executar a fabricação de um determinado lote de produtos. Num ambiente de manufatura integrada por computador este roteiro não precisa ser gerado em papel. Parte deste roteiro será transformada num programa CN (controle numérico) e enviado à máquina-ferramenta, ou seja, representa o outro sentido do fluxo de informação entre o CAPP e o chão-de-fábrica.

2. MODELO COMPUTACIONAL DO CAPP

Devido à complexidade da construção de um CAPP, é interessante, do ponto de vista computacional, utilizar uma estrutura modular, criando sistemas para processar entradas específicas e gerar parâmetros de saída para uma máquina de inferência que reúna, processe e pondere estas informações globalmente.

A estrutura computacional pode ser observada no esquema da figura 1. É importante ressaltar a ligação entre a máquina de inferência e os módulos individuais, onde as setas duplas indicam que as informações fluem nos dois sentidos, ou seja, depois de processados todos os dados a máquina de inferência busca sempre otimizar os resultados, chamando os módulos especialistas tantas vezes quanto for necessário para reprocessamento de informações.

Seguindo esta metodologia, Da Costa (1997) propôs em seu doutoramento a construção de um módulo especialista dirigido à escolha das condições de corte. Nishimuni (2000) desenvolveu um modelo para a representação de máquinas-ferramenta, definindo uma estrutura orientada a objetos para classificação e organização de modelos comerciais, possibilitando armazenar os recursos físicos de uma empresa. Dando continuidade a essa proposta, este trabalho busca desenvolver um módulo especialista que constitua o elo de ligação entre os sistemas CAD/CAPP.

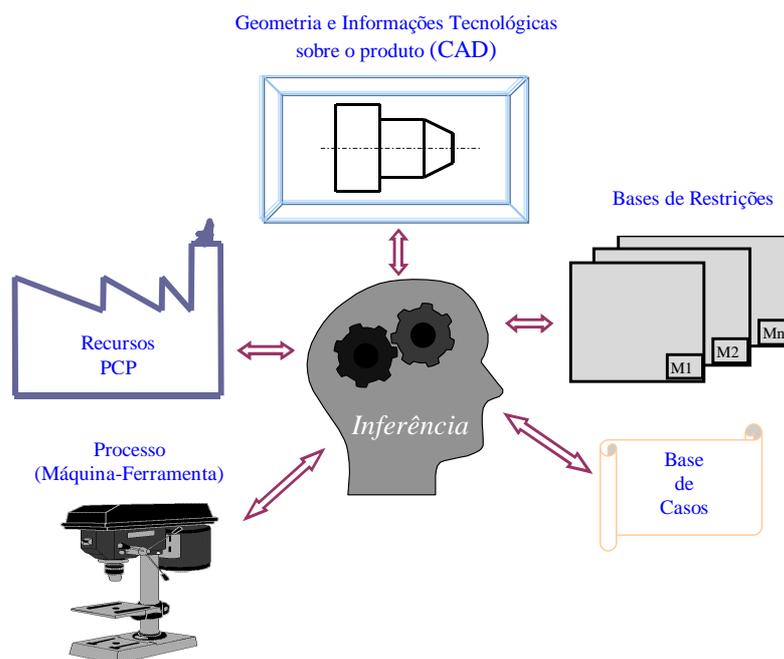


Figura 1 – Estrutura computacional modular de um CAPP

3. INTEGRAÇÃO CAD/CAPP

Os sistemas CAD tradicionais são essencialmente modeladores geométricos, não fornecendo a princípio informações que possam ajudar na etapa de planejamento de processos. As informações são armazenadas em geometria de baixo nível e entidades topológicas organizadas hierarquicamente (*BRep – Boundary Representation*) ou em árvores binárias de operações geométricas aplicadas a primitivas simples (*CSG – Constructive Solid Geometry*). Desta forma os modeladores geométricos propiciam um mecanismo para validar matematicamente a geometria definida pelo usuário.

Buscando uma integração maior entre sistemas CAD e a etapa de planejamento foi introduzido o conceito de *features*, que pode ser definido como um determinado conjunto de entidades geométricas ao qual são associados atributos e informações que possuam significado tecnológico. Seguindo esse raciocínio, foram desenvolvidos sistemas CAD para projeto mecânico que permitiam desenhar componentes através de *features*, porém tal atitude acaba forçando o projetista a utilizar um conjunto finito de ferramentas, inibindo a criatividade e limitando o projeto às *features* disponíveis. Para contornar estes problemas diversos pesquisadores têm proposto uma abordagem diferente, o reconhecimento e extração automática de *features* a partir de um modelo puramente geométrico, conseguindo resultados satisfatórios.

Dentre os vários métodos já propostos pode-se classificar as técnicas em dois grandes grupos [Shah, 1992]:

- *Reconhecimento de regiões usináveis*: São aplicados algoritmos que definem um volume inicial que envolva completamente a peça a fabricada, identificando então volumes intermediários que subtraídos do sólido inicial resultem na geometria final da peça;
- *Reconhecimento de Features*: Este método compara o modelo geométrico da peça com *features* genéricas pré-definidas e armazenadas em um banco de dados. O processo de reconhecimento consiste em identificar padrões topológicos/geométricos e na determinação dos parâmetros (por exemplo, diâmetro do furo, profundidade do *pocket*, largura do rasgo, etc.).

A abordagem utilizada neste trabalho envolve características do primeiro grupo citado, consistindo na análise de figuras planas extraídas do objeto tridimensional. Uma das técnicas utilizadas é o *seccionamento*, amplamente difundido na área de prototipagem rápida por deposição de material. Em usinagem, Chen (2001) desenvolveu um sistema robotizado para a obtenção de protótipos baseado na análise de arquivos SLC (*StereoLithography Contour*), que não passam de seções equidistantes extraídas do modelo sólido da peça. A grande vantagem desta abordagem está na simplicidade da análise de primitivas bidimensionais, podendo ser aplicada para obter informações úteis à manufatura.

Embora o reconhecimento de *features* proporcione resultados mais expressivos, sua eficácia é condicionada pela quantidade de *features* cadastradas, e, inevitavelmente, o conjunto de *features* possíveis de serem modeladas é infinito.

4. DESCRIÇÃO DA METODOLOGIA PROPOSTA

4.1 OBTENÇÃO DA DIREÇÃO PREFERENCIAL DE SECCIONAMENTO

O modelo tridimensional da peça em estudo precisa, inicialmente, respeitar algumas restrições de modo a não comprometer as análises posteriores. O arquivo gerado no sistema CAD, por exemplo, deve conter somente um sólido, além de estar devidamente orientado com os eixos do sistema de coordenadas. Esta condição é absolutamente necessária porque uma das propriedades que será frequentemente utilizada é a *Enclosed Box*¹, cujas dimensões são dadas em relação aos eixos X, Y e Z.

A princípio poderíamos arbitrar uma direção qualquer e começar a extrair seções. Esta, porém, não é a melhor alternativa. Caso essa direção não traga informações significativas, inevitavelmente teremos que voltar ao princípio, definir outra direção e executar todos os testes novamente, desperdiçando desta forma um tempo computacional precioso. Pensando nisso, a definição de qual é a melhor direção para iniciarmos as investigações é feita classificando-se o possível estado bruto do componente. Extraindo a *Enclosed Box* do sólido, podemos definir números adimensionais que relacionam as dimensões da caixa imaginária de acordo com os eixos principais e classificá-la de acordo com a provável direção do movimento de corte.

Sejam LW, LT e WT parâmetros definidos da seguinte maneira:

$$LW = \frac{\text{comprimento}}{\text{largura}} \quad (1)$$

$$LT = \frac{\text{comprimento}}{\text{espessura}} \quad (2)$$

$$WT = \frac{\text{largura}}{\text{espessura}} \quad (3)$$

Considerando que a unidade principal utilizada no arquivo contendo o desenho seja milímetros:

- a) Se $0,95 \leq WT \leq 1$ e $LW > 1,1$ então o provável estado bruto é *barra*, cujo movimento de corte na maioria dos casos é na direção do seu comprimento;
- b) Se $\text{espessura} \leq 3$ e $WT \geq 1$ então o estado bruto é definido como *chapa*. Neste caso o movimento de corte corresponde à direção da espessura, e dificilmente existirão operações cujo movimento de corte seja perpendicular a esta direção. Para estas situações é preciso apurar se o processo de fabricação mais indicado não é conformação mecânica;

¹ Caixa imaginária que engloba o sólido

- c) Se $3 < espessura \leq 25$ e $WT \geq 1$ estão o estado bruto escolhido é *placa*. O movimento de corte ocorre na direção da espessura, similarmente ao caso de chapa. Como a espessura é maior, a hipótese de que existam algumas operações nos planos contendo esta direção não pode ser descartada;
- d) Se nenhuma das regras anteriores for satisfeita, o estado bruto é classificado como *bloco*. Neste caso, é verificada a quantidade de círculos presentes em cada uma das vistas ortogonais que representam o sólido. Estas primitivas representam essencialmente furos, cuja direção do movimento de avanço deve ser a direção de seccionamento, independente se serão usinados por *formação* ou *geração*.

É importante salientar que os critérios anteriores buscam apenas definir uma provável direção que traga resultados conclusivos. Como a modelagem é baseada em conceitos heurísticos, a classificação do estado bruto nem sempre leva a uma direção correta, embora as chances de acerto sejam grandes.

Outro ponto a ser verificado é o percentual de material que precisa ser removido para se chegar ao estado acabado, definido como:

$$dVol = \left(1 - \frac{\text{volume real do sólido}}{\text{volume da caixa imaginária que envolve o sólido}} \right) \cdot 100 \quad (4)$$

Caso o valor de $dVol$ seja muito grande, provavelmente a usinagem não é o processo de fabricação mais adequado (levando-se em conta produção seriada do componente). Considerando que as perdas de material na forma de cavaco seriam elevadas, talvez outro processo (como fundição) seja a melhor alternativa. Para ilustrar a aplicação deste critério, vejamos o exemplo da figura 2. Para este caso $dVol$ é aproximadamente igual a 88% e, através de uma rápida inspeção visual, percebemos que o processo de fabricação mais indicado é conformação mecânica (puncionamento do furo e dobramento da chapa).

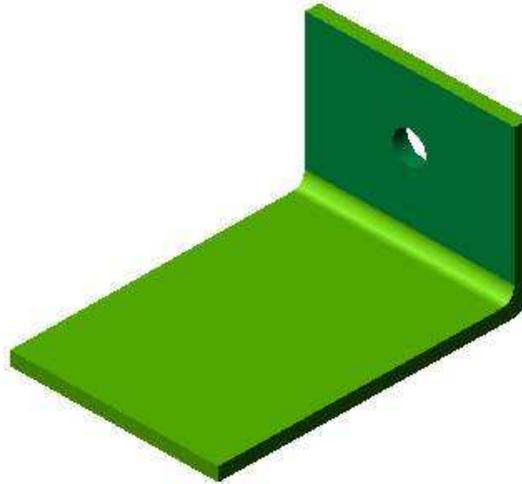


Figura 2 - Exemplo da aplicação do critério da remoção de material

4.2 DEFINIÇÃO DOS PONTOS DE SECCIONAMENTO

Uma vez determinada a direção onde ocorrerá o seccionamento, poderíamos simplesmente estipular um incremento qualquer, como 1 mm, e extrair seções paralelas nesta direção a partir de uma extremidade do sólido até atingir o seu fim. Todavia, este não é o melhor procedimento, pois inevitavelmente desperdiçaríamos um enorme tempo computacional na extração e análise de seções redundantes. Isto significa dizer que se o componente em estudo fosse uma barra circular de 1 m de comprimento, teríamos mil seções completamente iguais compostas por um círculo cada, o que definitivamente não é interessante.

Buscando evitar esse tipo de problema, foi desenvolvido um algoritmo para a obtenção dos pontos onde ocorrem variações bruscas de seção, assegurando desta forma que a cada dois desses pontos a seção qualitativamente é constante.

Escolhendo uma das duas vistas que contenham a direção de seccionamento (as vistas ortogonais serão sempre nos planos XY, YZ e XZ), uma lista contendo as coordenadas dos pontos onde existe mudança de seção é criada. Tais pontos são escolhidos com base nas primitivas presentes na vista:

- **Linhas:** são tomados os pontos inicial e final;
- **Círculos:** são tomados os dois pontos extremos segundo a direção de seccionamento;
- **Arcos:** primeiramente é avaliado se o arco intercepta a direção de seccionamento. Caso seja verdadeiro, tais pontos são extraídos.

A listagem final é então gerada após a exclusão dos valores repetidos. A figura 3 ilustra graficamente o procedimento adotado.

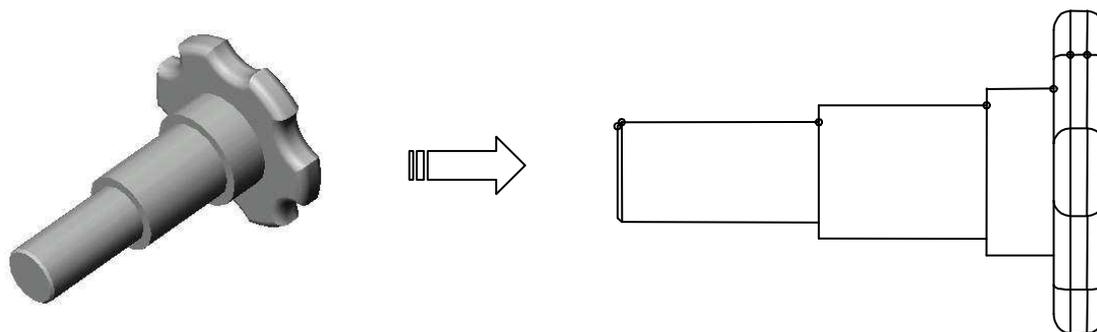


Figura 3 – Exemplo ilustrativo dos pontos de seccionamento

4.3 EXTRAÇÃO DAS SEÇÕES

A princípio poderíamos extrair uma seção em cada um dos pontos de seccionamento. Mas como a maioria dos pontos foi calculada em tempo de execução, podem existir erros de arredondamento que, por menores que sejam, podem causar problemas no momento de obter a respectiva seção. Isto poderia levar a um ponto fora do sólido (causaria erro, travando o programa) ou à perda de algum detalhe da peça. Para evitar esses inconvenientes, foi definido que serão extraídas duas seções para cada ponto: uma imediatamente anterior e outra imediatamente posterior, excluindo-se obviamente os pontos inicial e final do sólido. Além disso, para cada região uma terceira seção é tomada, localizada na metade do intervalo definido a cada dois pontos. A finalidade deste procedimento é avaliar a variação em tamanho das seções na região em questão, visto que em forma elas são incondicionalmente constantes.

O próximo passo é organizar os dados de cada seção. Como o arquivo contendo as primitivas é simplesmente uma lista, é preciso estabelecer quantos contornos existem e separá-los. O algoritmo desenvolvido é recursivo, isto é, toma uma primitiva qualquer e verifica qual outra compartilha um ponto comum, já que não existem três primitivas que compartilham o mesmo ponto. Conforme as primitivas adjacentes são encontradas, estas vão sendo excluídas das próximas buscas.

4.4 RELAÇÕES DE PERTINÊNCIA ENTRE CONTORNOS FECHADOS

Na análise das seções transversais da peça, é de fundamental importância a determinação das relações de pertinência entre as figuras fechadas obtidas a partir do seccionamento. Quando a seção é formada por uma única figura, fica claro que o contorno formado pelo conjunto de linhas e retas delimita a região que contém o material, sendo desnecessária a utilização do algoritmo proposto a seguir. A identificação da posição dos contornos, classificados em *internos* ou *externos*, permite ao algoritmo principal decidir qual rotina deve ser chamada e efetuar

os testes corretos. No caso da figura ser uma circunferência, por exemplo, a região da peça representada por esta seção pode ser obtida por revolução ou fresamento perimetral (dependendo do comprimento da região na direção perpendicular ao seccionamento) se a figura é externa, ou então pode ser furo, caso seja interna. Isto leva a tomadas de decisões completamente diferentes, já que a cinemática necessária para chegar ao estado final é completamente diferente.

A abordagem utilizada neste projeto consiste em investigar, através de uma representação vetorial, se um ponto está contido dentro de um polígono. Uma das vantagens de utilizar vetores é que segmentos de linha podem ser facilmente fixados a partir de dois pontos, ao contrário do que ocorre quando são usadas equações lineares, que representam retas de comprimento infinito.

Um segmento de reta é definido por dois pontos, P_1 e P_2 , ou por dois vetores, \mathbf{V}_1 e \mathbf{V}_2 . Conforme mostrado na figura 4, um terceiro vetor \mathbf{V}_3 pode ser formado pela subtração de \mathbf{V}_2 a partir de \mathbf{V}_1 :

$$\mathbf{V}_3 = \mathbf{V}_2 - \mathbf{V}_1 \quad (5)$$

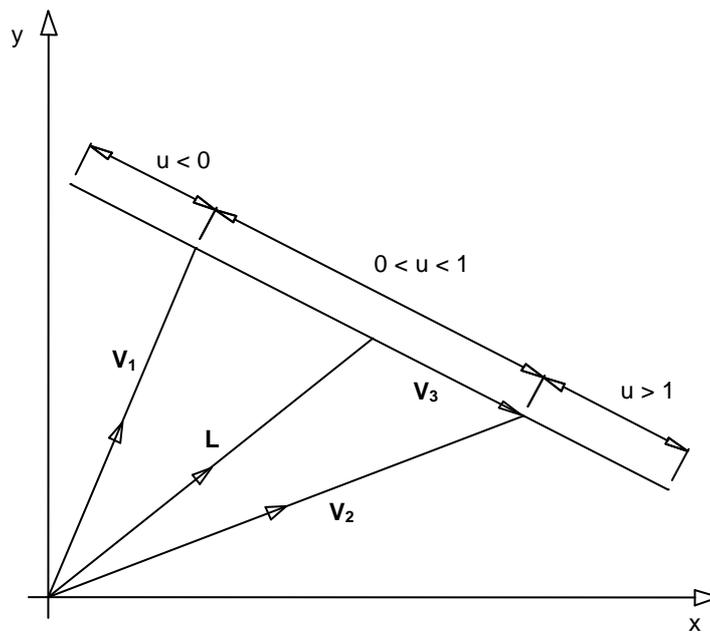


Figura 4 – Representação gráfica da análise vetorial

É importante ressaltar que \mathbf{V}_3 pode ser tratado como um vetor representativo do segmento em questão, assumindo que a origem do sistema de coordenadas seja reposicionada no ponto inicial do segmento.

Um outro vetor \mathbf{L} , com início na origem do sistema de coordenadas e término em um ponto qualquer de \mathbf{V}_3 , pode ser definido pela seguinte equação vetorial:

$$\mathbf{L} = \mathbf{V}_1 + u(\mathbf{V}_2 - \mathbf{V}_1) \quad (6)$$

onde o valor da constante escalar u indica a localização do ponto final de L . De fato, qualquer valor de u definirá um segmento diferente, mas a faixa de interesse é entre 0 e 1.

Estas propriedades da equação vetorial podem ser usadas para determinar se dois segmentos de retas se interceptam, existindo uma clara distinção para o caso de retas infinitas, que sempre se interceptarão em algum ponto desde que não sejam paralelas. Por exemplo, considere dois segmentos de reta, o primeiro se estendendo do ponto P_{11} ao ponto P_{12} e o segundo de P_{21} a P_{22} . Suas equações são:

$$L_1 = P_{11} + u(P_{12} - P_{11}) \quad (7)$$

$$L_2 = P_{21} + v(P_{22} - P_{21}) \quad (8)$$

A intersecção ocorre quando $L_1 = L_2$:

$$P_{11} + u(P_{12} - P_{11}) = P_{21} + v(P_{22} - P_{21}) \quad (9)$$

que pode ser rescrita em duas equações, uma para cada coordenada cartesiana:

$$x_{11} + u(x_{12} - x_{11}) = x_{21} + v(x_{22} - x_{21}) \quad (10)$$

$$y_{11} + u(y_{12} - y_{11}) = y_{21} + v(y_{22} - y_{21}) \quad (11)$$

Desde que os dois segmentos não sejam paralelos, este sistema de equações pode ser resolvido, existindo intersecção quando

$$0 < u < 1 \text{ e } 0 < v < 1$$

A partir da modelagem anterior, podemos determinar quando um ponto qualquer está dentro ou fora de um dado polígono. Os testes realizados são os seguintes:

1. Desenhemos uma linha orientada numa direção qualquer (para facilitar, a direção tomada é horizontal) a partir do ponto de teste até o infinito;
2. Em seguida, são verificadas quantas intersecções existem com os segmentos de retas que definem o polígono: se o número de intersecções é *ímpar*, o ponto é interno ao polígono; caso o número de intersecções seja *par*, o ponto é externo ao polígono.

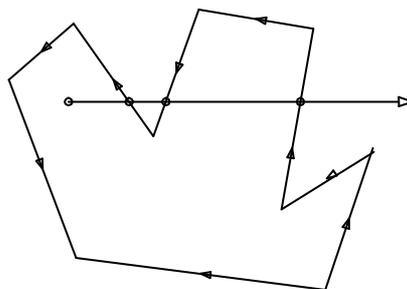


Figura 5 – Se o número de intersecções for ímpar, o ponto está contido no polígono

Como os testes anteriores são válidos somente para um ponto em relação a um polígono formado unicamente por segmentos de reta, são necessárias algumas transformações nos dados geométricos provenientes da seção. Para tanto, caso a mesma seja uma circunferência ou contenha arcos, o algoritmo aproxima estas primitivas por segmentos de retas. No algoritmo foi definido que cada arco ou círculo seja decomposto em 15 segmentos, valor que permite uma excelente aproximação. Caso seja desejado, pode-se aumentar esse valor de modo a aumentar a precisão, em detrimento do tempo de processamento.

Os pontos de teste, por sua vez, são escolhidos da seguinte maneira:

- a) Para cada um dos contornos que compõem a seção, um ponto representativo é definido e armazenado na memória;
- b) O critério para a tomada do ponto é baseado na verificação de uma primitiva qualquer pertencente à figura. Caso seja uma circunferência (é o próprio contorno), o ponto escolhido dista o valor do raio na horizontal, ou seja,

$$\begin{aligned}x_t &= x_c + r \\y_t &= y_c\end{aligned}$$

onde x_c e y_c referem-se às coordenadas do seu centro e r é o raio. Caso a primitiva seja uma linha ou arco, o ponto escolhido é o ponto inicial, uma vez que este ponto obrigatoriamente está sobre o contorno.

A nível de programação, existem alguns casos especiais que, se não mapeados e corrigidos, causarão erros. Isto acontece se a linha horizontal semi-infinita passar por um ou mais vértices do polígono, conforme a figura 6. Para obter os resultados corretos, o vértice A_1 deve ser considerado como não fornecendo nenhuma intersecção (ou fornecendo duas intersecções), enquanto os vértices A_2 , A_3 e A_4 proporcionam uma intersecção cada. A solução para este problema é identificar os vértices que passam pela linha horizontal e providenciar um teste especial. Tal teste verifica as coordenadas y do vértice anterior e posterior, definindo o número de intersecções que precisam ser fornecidas (assume-se que os vértices vizinhos não são colineares).

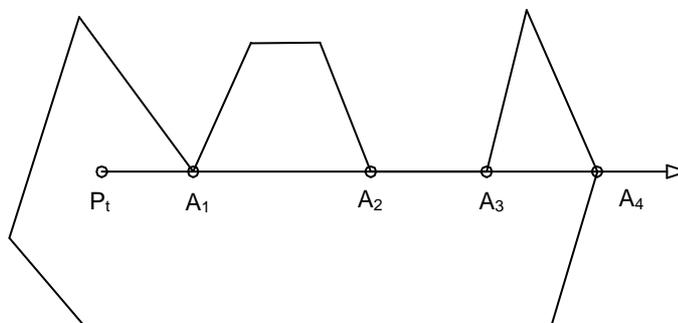


Figura 6 – Casos especiais do teste de pertinência

Após realizar estas etapas, o algoritmo determina, através de processamento recursivo, qual figura contém outra(s) e armazena os resultados numa matriz de dimensões $n \times n$ onde n é a quantidade de figuras fechadas que constituem a seção. Esta matriz vai sendo preenchida de acordo a seguinte notação:

$$M_{ij} = 1 \quad (\text{a figura } i \text{ contém a figura } j)$$

$$M_{ij} = 0 \quad (\text{a figura } i \text{ não contém a figura } j)$$

4.5 CLASSIFICAÇÃO DAS SEÇÕES

Os resultados guardados na forma matricial propiciam muitas facilidades no momento de classificar a seção, racionando o uso da memória e diminuindo o tempo de processamento. Imaginemos uma seção composta por quatro figuras, denominadas F_1 , F_2 , F_3 e F_4 , cuja matriz resultado seja a seguinte:

$$M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

A classificação da seção é feita somando os elementos de cada linha e verificando, de acordo com os índices, qual figura contém as outras. Para a primeira linha, o somatório é igual a zero, logo, F_1 não contém nenhuma outra figura. Da segunda linha temos somatório igual a 2, logo F_2 contém duas figuras, respectivamente F_1 e F_3 . Para as duas linhas restantes, verificamos que estas também não contém outras figuras.

Aplicando o mesmo raciocínio agora para as colunas, vemos que F_1 e F_3 estão contidas em F_2 , e que F_2 e F_4 não são contidas por ninguém. Vale ressaltar que a diagonal principal da matriz é sempre nula, pois uma figura não pode conter ela mesma.

Confrontando os resultados dessas duas análises, conclui-se que F_4 é um contorno isolado (não contém ninguém nem está contido em outra figura) e que F_2 abriga F_1 e F_3 .

De acordo com esta metodologia, as seções são então enquadradas nos seguintes casos:

- **Contorno Simples (CS):** é situação mais ordinária, onde a seção é composta por um único contorno que diferencia a região que contém material do meio externo;
- **Contorno Composto (CC):** é um contorno simples contendo uma ou mais figuras no seu interior, que, por sua vez, representam regiões onde deverá ocorrer remoção de material.
- **Múltiplo Contorno Simples (MCS):** quando existem dois ou mais CS na mesma seção;

- **Múltiplo Contorno Composto (MCC):** é o caso MCS quando pelo menos um dos contornos é composto.

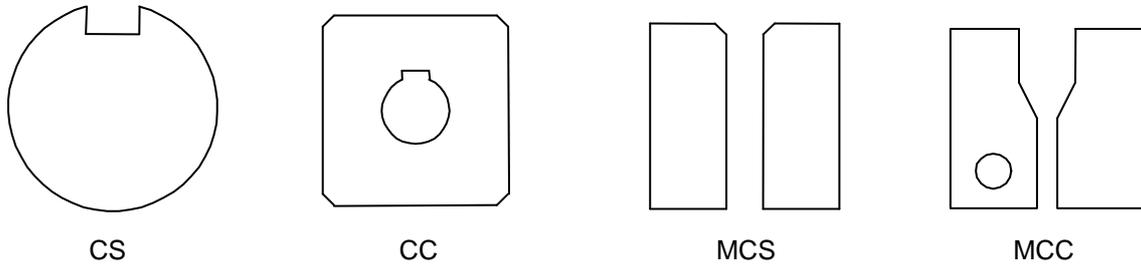


Figura 7 – Exemplos da classificação de seções adotada

Esta classificação pode ser estendida para outras situações, como no caso de seções que possuam *Contorno Hipercomposto*, ou seja, figuras que estão contidas por outra também contém outras figuras. Este caso não foi implementado devido à complexidade da análise cinemática, o que demandaria muito tempo de programação, ficando como sugestão para trabalhos futuros que desejem aprofundar-se no assunto.

4.6 ALGORITMO PARA FRESAMENTO PERIMETRAL DE UM CONTORNO

Para que um contorno qualquer possa ser usinado por fresamento perimetral, é necessário que o mesmo possua somente arestas convexas. Quando existirem arestas côncavas, estas devem ser formadas por arcos para que possam ser executadas por essa operação, sendo que o raio do arco, na maioria dos casos, limita o raio da ferramenta de corte. Desta forma, caso exista uma aresta côncava formada por duas linhas, é impossível obter o contorno através da translação da ferramenta ao redor da peça, o que implicaria em um raio igual a zero.

Para exemplificar a metodologia desenvolvida, imaginemos que a figura 8 foi obtida a partir de uma seção da peça.

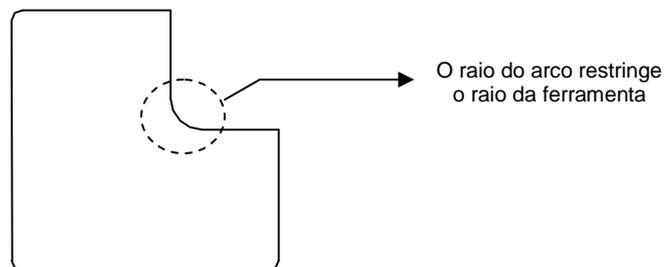


Figura 8 – Exemplo de restrições que definem se o contorno pode ser obtido por fresamento perimetral

Visualmente podemos concluir que esta seção pode ser usinada por fresamento perimetral, desde que a ferramenta possua um raio menor ou igual ao raio do arco que forma uma aresta côncava entre as duas faces perpendiculares. Mas como um algoritmo computacional deve proceder para chegar a esta conclusão? Até o momento, os algoritmos já desenvolvidos forneceriam as seguintes informações:

- a) Classificação do contorno: CS (contorno simples);
- b) Primitivas parcialmente ordenadas, ou seja, partindo de uma linha ou arco qualquer teríamos as primitivas subsequentes que fechariam a figura, em um sentido não definido.

Para analisar se a operação de fresamento é passível de ser executada, é necessário levar o ordenamento das primitivas a um nível mais alto, de forma que do ponto de vista computacional seja possível identificar a região interna que contém o material da peça, e a região externa, onde a ferramenta pode transitar. Além disso, a aresta de corte não pode mais ser considerada como sendo um ponto, e sim como uma região do espaço que deve trafegar ao redor da peça, gerando mais uma restrição.

Essa nova classificação consiste nas seguintes etapas:

Passo 1: O contorno deve ser representado por pontos ordenados, que interligados entre si devem fechar a figura. Além desse conjunto de pontos, é preciso conhecer a primitiva presente a cada dois pontos. Caso seja um arco, é necessário ainda guardar o seu centro.

Passo 2: A próxima etapa consiste na determinação do sentido de geração da figura, que até o momento não está definido (pode ser tanto horário como anti-horário). A partir desse momento torna-se imprescindível trabalhar como os dados na forma vetorial, visto que vetores condensam informações como módulo, direção e sentido, podendo ser resgatadas a qualquer momento por funções analíticas previamente implementadas. Para ilustrar o procedimento adotado, vamos desconsiderar a presença de arcos e imaginar a seção mais simples possível, ou seja, um triângulo equilátero conforme a figura 9.

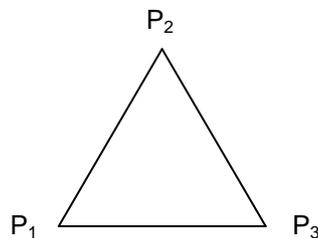


Figura 9

Como entre os três pontos só existem retas, criamos então três vetores que somados resultem em um vetor nulo (ou seja, fecham a figura):

$$\vec{V}_1 = P_2 - P_1$$

$$\vec{V}_2 = P_3 - P_2$$

$$\vec{V}_3 = P_1 - P_3$$

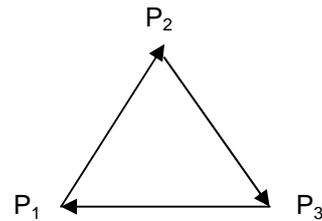
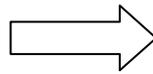


Figura 10

Para descobrirmos o sentido de geração do contorno é necessário escolher um ponto e verificar se ele está dentro ou fora da figura. A partir de um vetor qualquer, o ponto de teste é definido como sendo o ponto situado a uma distância equivalente à metade do valor do módulo a partir do ponto inicial do vetor e defasado de 1 grau à direita do vetor escolhido. Caso este ponto esteja dentro da figura, então o sentido de geração é horário; caso esteja situado fora da figura, o sentido é anti-horário.

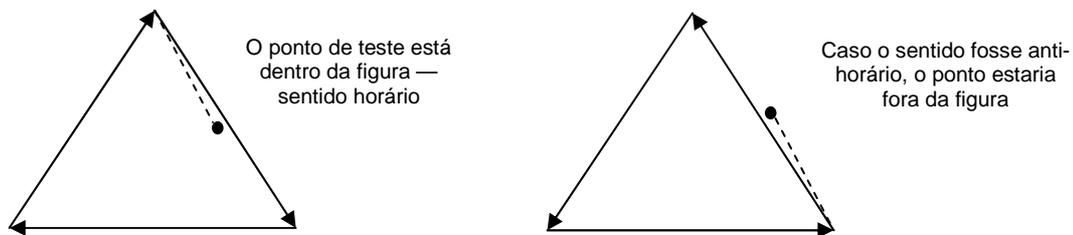


Figura 11

Passo 3: Uma vez determinado o sentido da figura, precisamos adotar o sentido de translação da ferramenta para que execução dos testes. Nos algoritmos implementados o sentido escolhido foi o anti-horário. Desta forma, se do passo anterior descobriu-se que a figura está no sentido horário, o algoritmo automaticamente inverte a ordem dos pontos e recalcula os vetores.

Passo 4: A próxima etapa é calcular o ângulo de cada um dos vetores faz com um vetor de referência, no caso, um vetor unitário orientado segundo o eixo x positivo. De posse dos valores de todos os ângulos, estes são então analisados dois a dois e cada uma das arestas do polígono é classificada em côncava ou convexa de acordo com a seguintes regra:

```
Se ang_final ≥ ang_inicial E ang_final ≤ ang_inicial + 180 Então
    Aresta CONVEXA
Caso contrário
    Aresta CÔNCAVA
```

onde *ang_inicial* refere-se ao ângulo do vetor inicial e *ang_final* ao ângulo do vetor subsequente.

Antes de aplicar a regra anterior é necessário corrigir os valores dos ângulos. Quando o vetor inicial está no terceiro ou quarto quadrante, por exemplo, em

alguns casos o vetor final pode estar no primeiro quadrante (tendo ângulo entre 0 e 90 graus) e formar uma aresta convexa. Nestas situações é preciso somar 360 ao valor do ângulo final de forma que o valor total seja maior que 360 graus.

Passo 5: Como nem toda seção resultará somente em linhas, é necessário considerar a presença de arcos na figura. Seguindo o mesmo raciocínio anterior, no qual a aresta é classificada de acordo com a variação angular sofrida a cada par de vetores, o tratamento vetorial utilizado consiste na determinação dos vetores tangentes ao ponto inicial e final de cada arco, obedecendo o sentido anti-horário de geração pré-definido. Tomemos como exemplo a seção representada pela figura 12.

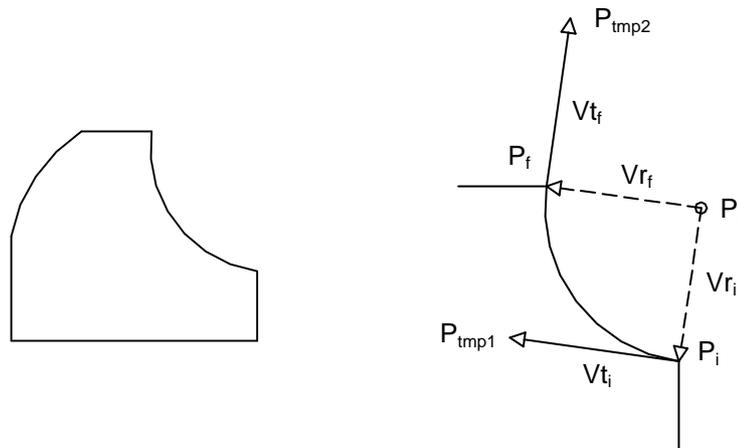


Figura 12

Quando a presença de um arco no contorno é detectada, uma rotina automaticamente cria os vetores V_{t_i} e V_{t_f} , tangentes ao ponto inicial e final do arco, respectivamente. Tais vetores são definidos pelos pontos P_{tmp1} e P_{tmp2} , calculados em função do centro do arco e dos pontos P_i e P_f através das seguintes fórmulas:

$$X_{tmp} = X_{P_i} + \|P_i - P_c\| \cdot \cos(\text{angulo}[P_i - P_c] \pm 90) \quad (12)$$

$$Y_{tmp} = Y_{P_i} + \|P_i - P_c\| \cdot \text{sen}(\text{angulo}[P_i - P_c] \pm 90) \quad (13)$$

Para que o vetor tangente esteja apropriadamente orientado, deve ser somado 90 graus ao ângulo de V_{r_i} ou V_{r_f} se a concavidade do arco aponta para o exterior da figura, ou, caso contrário, deve ser subtraído 90 graus se a concavidade apontar para o interior. Essa questão é solucionada computacionalmente pela verificação da posição do ponto médio que une os pontos inicial e final do arco em relação ao contorno. Cabe salientar que desses vetores calculados só interessa a direção, sendo que o valor dos seus módulos serão iguais ao raio do arco.

Aplicando a mesma metodologia proposta para a caso de contornos compostos somente por linhas, podemos, neste ponto, identificar se o arco e as arestas formadas por ele com a primitiva anterior e posterior são côncavas ou convexas.

Passo 6: Para completar a análise, resta averiguar os pontos fracos do modelo matemático, ou seja, situações que causarão erros e fornecerão resultados falsos, e definir, em função dos arcos côncavos, qual deve ser o maior raio permissível da ferramenta de corte, caso se chegue à conclusão de que o contorno pode ser obtido por fresamento perimetral.

De acordo com a figura 13, observamos que existem dois arcos diferentes que, segundo a representação computacional adotada, são idênticos. Isto acontece porque a análise é baseada somente no ângulo que os vetores fazem com o eixo horizontal.

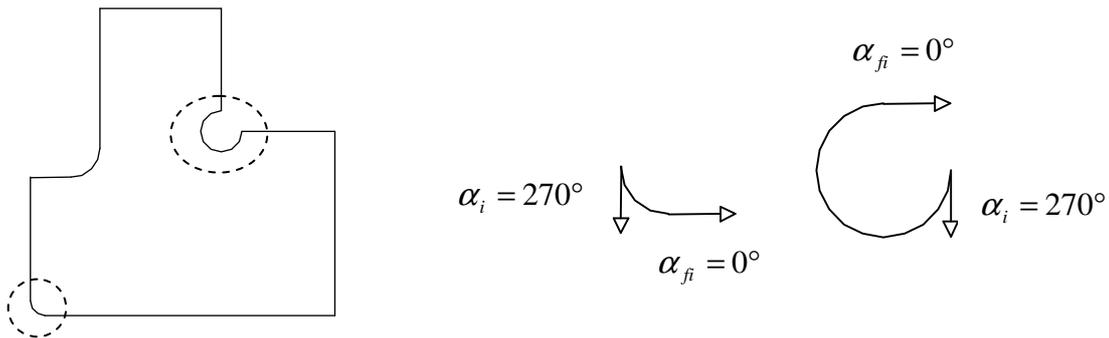


Figura 13 - Exemplos de arcos diferentes que possuem a mesma representação

Buscando contornar este problema, para cada arco do contorno um rotina específica verifica a extensão do arco, isto é, checa se o mesmo varre mais de 180 graus, através da seguinte regra:

$$\frac{\text{ang_final} - \text{ang_inicial}}{360} > 0,5 \quad (14)$$

onde *ang_final* é devidamente corrigido caso seu valor seja menor que *ang_inicial*.

Se o resultado do teste anterior for falso, o arco é menor que uma meia-circunferência e algoritmo já desenvolvido prossegue. Se o teste for verdadeiro, significando que o arco cobre mais de 180 graus, então um vetor auxiliar é calculado. Este vetor é posicionado no ponto que divide o arco ao meio, tangenciando o mesmo, de acordo com a figura 14.

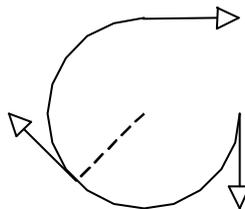


Figura 14 – Representação gráfica do vetor auxiliar

De posse desse vetor auxiliar, o algoritmo geral pode ser aplicado, testando separadamente o vetor inicial com o vetor auxiliar e este com o vetor final.

A determinação do maior raio admissível para a ferramenta é feita guardando o raio dos arcos classificados como côncavos e, no final, verificando qual é o menor. Além deste procedimento, quando um arco maior que meia-circunferência for identificado pelo critério anterior, a dimensão que restringe a ferramenta não é o seu raio, e sim a metade da distância entre seus pontos inicial e final.

Outro ponto que deve ser levado em consideração é o fato de que, muitas vezes, o projetista esquece de desenhar no modelo da peça os raios de concórdância entre as faces côncavas, visto que o profissional responsável pela produção intuitivamente saberá que ali existirá um raio. Em vista disso, é imprescindível que o desenho seja coerente com as restrições produtivas, evitando que a análise computacional forneça resultados inválidos.

4.7 OPERAÇÕES DE TORNEAMENTO

Existem diversas situações onde, do ponto de vista cinemático, determinadas regiões da peça poderão ser usinadas por fresamento perimetral. Todavia do ponto de vista prático essa alternativa nem sempre é a mais viável, seja por demandar um maior tempo de corte ou por limitações estruturais da ferramenta. Desse modo, é interessante testar se a seção pode ser gerada pela revolução da peça associada a um movimento de avanço da ferramenta de corte na direção de secionamento. Esta opção assegura, entre outras coisas, melhor acabamento superficial e uma redução do tempo de usinagem, sendo portanto, mais interessante economicamente.

O algoritmo computacional para realizar este teste é fundamentado na presença de arcos na seção, caso esta não seja uma circunferência. Quando a seção é circular, a conclusão imediata é que a possibilidade de torneamento é a mais provável, desde que as regiões anterior e posterior não gerem restrições quanto à translação do conjunto ferramenta e porta-ferramenta. No extremo oposto, se a seção é composta somente por linhas, conclui-se que se trata de uma região prismática e essa opção é imediatamente descartada.

Verificadas as situações anteriores, para que uma seção não circular possa ser parcialmente obtida por torneamento é necessário que pelo menos um dos arcos presentes na seção defina uma circunferência que contenha todas as demais primitivas. É importante ressaltar que a regra anterior não assegura que a região em estudo é tipicamente de revolução, apenas afirma que cinematicamente a operação é exeqüível.

O procedimento inicial consiste em armazenar na memória o raio de todos os arcos presentes no contorno. Em seguida são definidos os pontos de teste, de acordo com a primitiva:

- **Linhas:** ponto inicial e final;
- **Círculos:** uma vez que círculos são intrinsecamente contorno fechados, estes estão obrigatoriamente dentro do contorno externo em estudo. Assim sendo, o seu centro é o ponto de teste;

- **Arcos:** novamente é necessário um processo de discretização, ou seja, uma aproximação por segmentos de reta. No programa implementado, os arcos são quebrados em 15 linhas, valor que pode ser aumentado para uma eventual melhora na precisão.

Seja m o número de arcos e n a quantidade de pontos de teste. Para que a seção admita torneamento externo, a inequação

$$(X_{i_j} - X_{c_i})^2 + (Y_{i_j} - Y_{c_i})^2 \leq R_i^2 \quad (15)$$

deve sempre ser satisfeita, onde i varia de 1 a m e j varia de 1 a n .

Para verificar se um contorno interno à outra figura pode ser parcialmente obtido por torneamento interno, a análise é a mesma, invertendo o sinal da inequação para maior ou igual.

4.8 OPERAÇÕES DE FURAÇÃO

A identificação de furos é a etapa mais simples, consistindo na verificação de círculos internos à seção. Quando mapeados, resta definir a maneira como serão obtidos (em função da ferramenta):

- **Por formação:** o diâmetro do furo é igual ao diâmetro da broca ou fresa, sendo usinado num único passe;
- **Por geração:** o diâmetro da ferramenta é menor que o diâmetro do furo. Através de ciclos pré-definidos, a fresa vai removendo material até alcançar a dimensão final. Pode existir uma etapa de pré-furação para aliviar os esforços na operação de fresamento.

4.9 ANÁLISE DOS RESULTADOS

Após a execução de todos os algoritmos já descritos, é preciso validar as conclusões obtidas para cada região da peça. Esta tarefa, porém, necessita de informações de outros módulos do CAPP, fugindo do escopo deste trabalho. Na escolha final das máquinas-ferramenta, deve-se levar em conta os seguintes aspectos:

1. É necessário criar uma base de dados detalhada das ferramentas de corte disponíveis, contendo todas as dimensões características que as identificam. Nesta etapa, a ferramenta não pode mais ser considerada pontual, e sim como um volume que restringe a obtenção da geometria final. Deve-se levar em conta

ainda o volume de dispositivos como porta-ferramentas, que podem tocar na peça durante a operação.

2. É preciso determinar as superfícies de referência, os dispositivos de fixação e o número de fixações. Esta é uma das etapas mais complexas, senão a mais difícil, visto que muitas vezes existe um projeto paralelo para definir um dispositivo de fixação adequado. Modelar computacionalmente o raciocínio humano para essas tarefas é extremamente complicado, mas buscando-se desenvolver um CAPP completo esta parte não pode ser esquecida.
3. A máquina onde será executado o acabamento é definida em função de informações tecnológicas como tolerância e rugosidade máxima. Esses parâmetros indicam a necessidade de operações de retificação, por exemplo.
4. A massa do componente e as dimensões máximas segundo os eixos coordenados, extraídas pelo programa implementado, limitam quais equipamentos poderão ser utilizadas. Deve-se, enfim, consultar um banco dados contendo modelos comerciais e escolher, de acordo com os recursos físicos disponibilizados pela fábrica, as máquinas-ferramenta que produzirão o componente.

5. INTERFACE COM O AUTOCAD

Os algoritmos desenvolvidos para controlar o *AutoCad*² podem ser divididos em duas categorias: automação através das tecnologia *ActiveX* e *VBA (Visual Basic for Applications)*, oferecidas pelo próprio sistema, ou através de *scripts*³, gerados em tempo de execução pelo programa implementado em *Visual Basic*.

A principal entrada de dados para os algoritmos de raciocínio geométricos são os arquivos *dxf*. Desta forma, uma atenção especial foi dispensada visando interpretá-los corretamente, visto algumas peculiaridades existentes. Como nos interessa somente círculos, arcos e linhas provenientes das representações planas do sólido, ao abrir o arquivo seqüencial correspondente vamos direto ao marcador *ENTITIES*, a partir do qual estão localizadas todos os objetos que possuam representação gráfica. Buscando pelos códigos adequados⁴, podemos extrair todos os parâmetros que interessam e armazená-los em um banco de dados apropriado. Tais parâmetros são definidos de acordo a primitiva:

- **Linhas:** ponto inicial, ponto final e vetor de extrusão;
- **Arcos:** centro, raio, ângulo inicial, ângulo final e vetor de extrusão;
- **Círculos:** centro, raio e vetor de extrusão.

Antes de disponibilizar os dados anteriores para os algoritmos implementados, é necessário reprocessá-los, transportando-os para um sistema de coordenadas absoluto, e calcular alguns parâmetros adicionais que serão extremamente úteis.

O primeiro passo é avaliar os vetores de extrusão. Quando a primitiva não estiver no plano XY, o AutoCad automaticamente gera um novo sistema de coordenadas individual, isto é, cada primitiva possui o seu, visando poupar espaço nos arquivos. A transposição para o sistema de coordenadas absoluto é realizada de acordo com o seguinte algoritmo:

```
N = vetor de extrusão
Wy = (0,1,0)
Wz = (0,0,1)
```

— Definir um eixo X e um eixo Y normais a N, denominados respectivamente Ax e Ay.

```
Se (abs(Nx) < 1/64) e (abs(Ny) < 1/64) então
    Ax = Wy X N (onde "X" é produto vetorial)
Caso contrário
    Ax = Wz X N.
```

— Ay é então definido como:

```
Ay = N X Ax
```

² *AutoCad*[®] é um sistema CAD comercial da *AutoDesc Inc.*

³ Arquivos texto contendo uma lista de comandos e seus respectivos argumentos.

⁴ Para maiores detalhes, consultar o *Guia de Referência DXF*, presente no próprio AutoCad.

Visando otimizar o código e evitar que o mesmo cálculo seja executado desnecessariamente, para arcos os pontos inicial e final são calculados nesta etapa. Pelo mesmo motivo, após a mudança das coordenadas para um referencial absoluto são determinados os coeficientes angulares de todas as linhas.

Por meio dos procedimentos anteriores, asseguramos que os dados extraídos dos arquivos *dxf* estão corretos e não causarão problemas nas etapas subsequentes.

6. PROGRAMAÇÃO ORIENTADA A OBJETOS

Para facilitar a comunicação com futuros módulos do CAPP e com a própria máquina de inferência, todos os resultados obtidos por este projeto foram transformados em objetos. Desta forma, temos uma estrutura hierárquica onde os elementos (contornos, seções, vistas, etc.) herdam propriedades dos níveis superiores, facilitando a recuperação e manipulação de informações. Esta organização é constituída essencialmente pelos objetos propriamente ditos, suas propriedades e por métodos. Os métodos nada mais são que funções aplicáveis a um determinado grupo de objetos visando obter uma dada propriedade. A figura abaixo demonstra a estrutura utilizada e as inter-relações entre métodos e objetos.

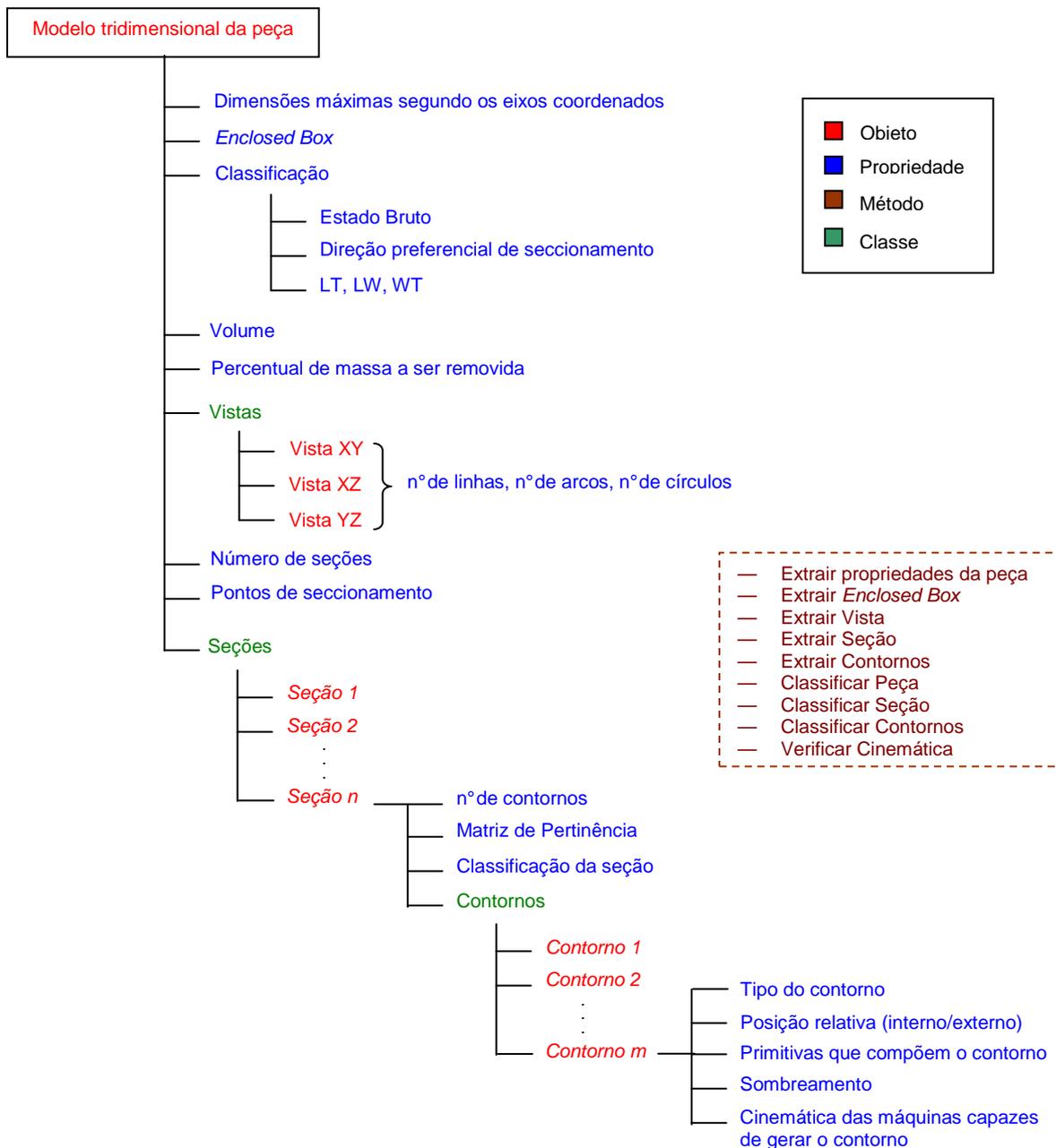


Figura 15 - Estrutura orientada a objeto empregada

7. EXEMPLO

Para ilustrar os resultados obtidos, a peça representada pela figura abaixo foi analisada pelo protótipo implementado. As regiões agrupadas pelos colchetes possuem seção qualitativamente igual (variam em tamanho), sendo obtidas pelas mesmas operações.

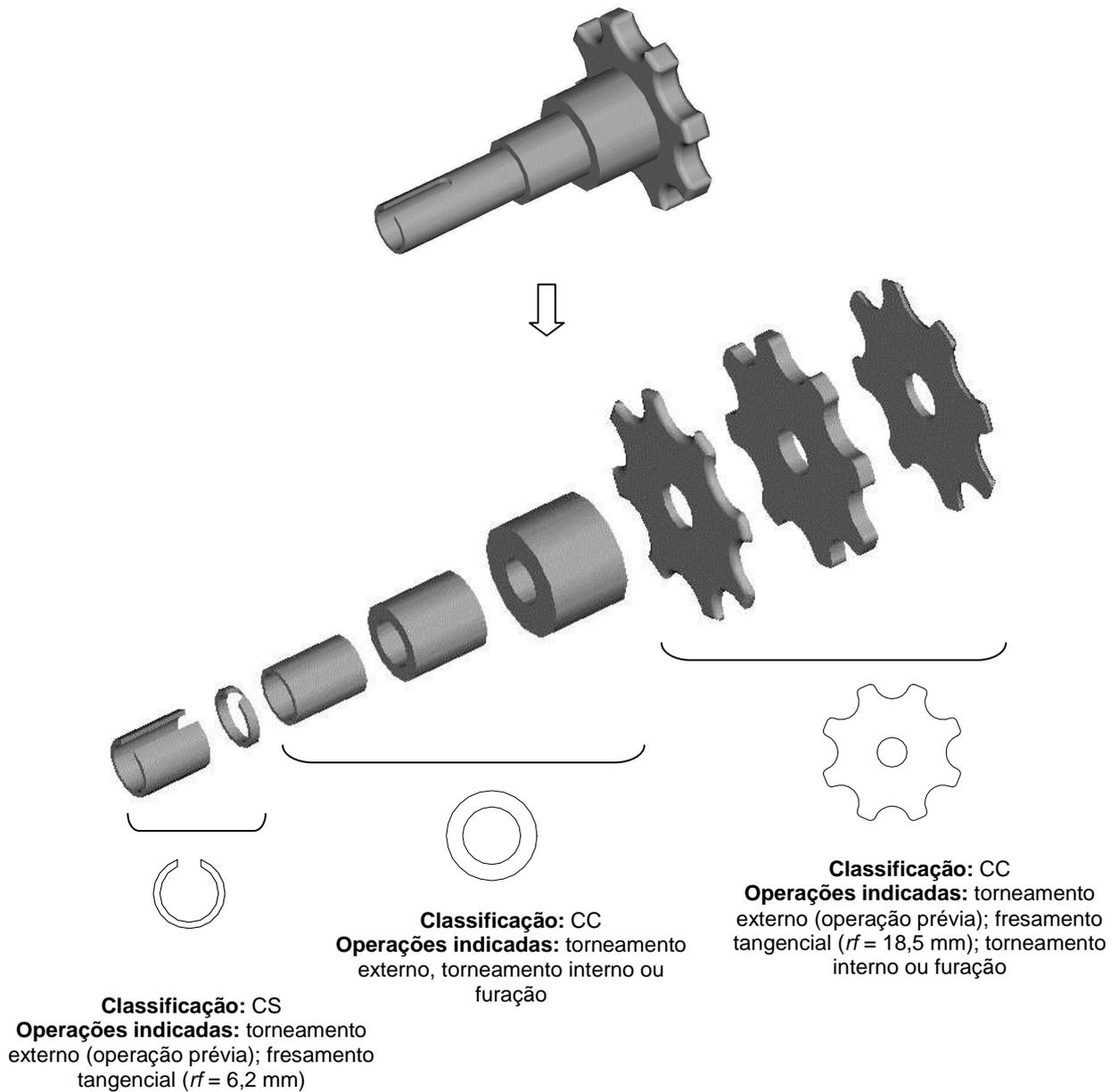


Figura 16 – Exemplo de resultados obtidos a partir do modelo sólido da peça. rf é o raio mínimo da fresa para que o contorno possa ser usinado

8. CONCLUSÕES

A análise de modelos tridimensionais através suas representações bidimensionais mostrou-se eficiente e aplicável a um grande número de situações, possibilitando indicar quais máquinas-ferramenta são capazes de usinar a peça. A partir de dados puramente geométricos como linhas, arcos e círculos foi possível reconhecer regiões usináveis por operações de fresamento, torneamento e furação. Porém a implementação computacional de algoritmos para raciocínio geométrico baseados na análise de figuras planas mostrou-se bastante trabalhosa, visto o baixíssimo nível dos dados de entrada. Além disso, estas análises consomem um enorme tempo de processamento, tornando a execução do programa demasiadamente demorada quando o componente em estudo é razoavelmente complexo. Em vista disso, é interessante adotar uma modelagem híbrida, associando à análise de seções bidimensionais o processamento de primitivas tridimensionais como planos e superfícies analíticas fornecidas por uma estrutura *B-Rep*. Outra ferramenta que pode ser utilizada visando reduzir os tempos de programação e de processamento é a extração de *features* previamente criadas. Isso facilitaria o rastreamento de conjuntos geométricos padronizados como roscas e engrenagens.

Os resultados alcançados, entretanto, não são definitivos. Como foram desconsiderados aspectos tecnológicos como tolerâncias dimensionais e rugosidade, não se pode afirmar em que máquina a operação de acabamento deve ser realizada. O tratamento destas informações, oriundas do modelo CAD, pode gerar novas restrições que levem a um processo de retificação, por exemplo. Além disso, não é possível modelar matematicamente algumas características das máquinas, como rigidez e histerese dos fusos de medição. Tais fatores são decorrentes da montagem e estado de conservação do equipamento, influenciando diretamente no acabamento final. Outro ponto que foi deixado de lado é a extração e análise de elipses, presentes nas representações planas de furos inclinados em relação aos planos ortogonais de referência. Situações como essas não comprometem o estudo em fase experimental, mas desejando-se uma análise completa e profunda não podem ser relegadas a segundo plano.

Outro inconveniente de trabalhar com representações planas do sólido é a dependência de um sistema CAD para efetuar operações como geração de vistas e seções. Como este trabalho tem caráter experimental, a necessidade de um programa externo para realizar estas tarefas não é impeditiva. Na construção de softwares comerciais utilizando esta metodologia, entretanto, seria imperativo gerar um sistema independente, com rotinas próprias para essas finalidades.

Além de permitir selecionar as máquinas-ferramenta, ou no mínimo eliminar algumas opções, os resultados obtidos pelo programa implementado em *MS Visual Basic*[®] são de extrema importância para outras etapas do planejamento, como a determinação das superfícies de referência, seqüenciamento das operações e geração do código CN. A abordagem empregada, utilizando programação orientada a objetos, facilita a recuperação de informações como topologia e propriedades dos contornos, proporcionando uma nova representação que otimiza o acesso e processamento realizado pela máquina de inferência.

9. REFERÊNCIAS BIBLIOGRÁFICAS

1. ALTING, Leo; ZHANG, Hongchao. **Computer Aided Process Planning: the state-of-the-art survey**. International Journal of Production and Research, v. 27, n. 4, p. 553-585, 1989.
2. CHEN, Y. H.; SONG, Y. **The development of a layer based machining system**. Computer-Aided Design, n. 33, p. 331-342, 2001.
3. DA COSTA, Dalberto Dias. **Proposta de uma Metodologia para Construção de um Módulo Especialista Dirigido à Escolha das Condições de Corte**. Tese de Doutorado, Unicamp, 1997.
4. ELMARAGHY, Hoda A. **Evolution and Future Perspectives of CAPP**. Annals of the CIRP, v. 42, n. 2, p. 739-751, 1993.
4. NISHIMUNI, Érico. **Construção de uma base de recursos baseada na infraestrutura disponível nas empresas do estado do Paraná**. Relatório de Bolsa de Iniciação Científica, 2000.
5. SARMA, Sanjay E.; WRIGHT, Paul K. **Using Mechanical hardware to simplify process planning**. Computer Integrated Manufacturing Systems, v. 11, n. 3, p. 147-155, 1998.
6. SHAH, Jami J. **Features in Design and Manufacturing**. Intelligent Design and Manufacturing, 1992.
7. VAN HOUTEN, F. J. A. M. **PART: A Computer Aided Process Planning System**. Tese de Doutorado, 1991.

10. BIBLIOGRAFIA AUXILIAR

1. ALLADA, Venkat; ANAND, Sam. **Feature-based modelling approaches for integrated manufacturing: state-of-the-art survey and future research directions.** International Journal of Computer Integrated Manufacturing, v. 8, n. 6, p. 411-440, 1995.
2. BURGER, Peter; GILLIES, Duncan. **Interactive Computer Graphics: Functional, Procedural and Device-Level methods.** 1992
3. CHANG, Tien-Chien; WYSK, Richard A. **An Introduction to Automated Process Planning Systems.** Prentice-Hall, 1985.
4. DHANDE, S. G.; KARUNAKARAN, K. P.; MISRA, B. K. **Geometric Modeling of Manufacturing Processes Using Symbolic and Computational Conjugate Geometry.** Journal of Engineering for Industry, v. 117, p. 288-296, 1995.
5. PEKLENIK, J.; SLUGA, A. **Contribution to Development of a Generative CAPP-System Based on Manufacturing Process Topology.** Annals of the CIRP, v. 38, n. 1, p. 407-412, 1989.
6. SILER, Brian; SPOTTS, Jeff. **Usando Visual Basic 6.0.** Rio de Janeiro : Campus,1999.

11. ASSINATURAS

Agosto de 2001

Ali Antonio Manfrin Abdallah
Bolsista PIBIC/CNPq

Dalberto Dias da Costa
Prof. Orientador