

**UTILIZAÇÃO DO CRITÉRIO DE SUPERFÍCIES
DESENVOLVIDAS PARA AUMENTAR PRECISÃO
NA FABRICAÇÃO DO CASCO DE EMBARCAÇÕES
DE MÉDIO E PEQUENO PORTES**

RAMÓN HERNÁN VELASCO STOLL

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

**São Paulo
1995**

OK

São Paulo
1995

Orientador:
Prof. Dr. Odair Caltabeloti

Área de Concentração:
Engenharia Naval

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

**UTILIZAÇÃO DO CRITÉRIO DE SUPERFÍCIES
DESENVOLVIDAS PARA AUMENTAR PRECISÃO
NA FABRICAÇÃO DO CASCO DE EMBARCAÇÕES
DE MÉDIO E PEQUENO PORTES**

RAMÓN HERNÁN VELASCO STOLL

À minha Esposa:
*Olga, a companheira da minha vida,
por tudo seu amor, compreensão e
estímulo.*

DEDALUS - Acervo - EPMN



31600010053

AGRADECIMENTOS

Ao meu orientador, Prof. Dr. Odair Calitabeloti, pelo apoio proporcionado e por estar sempre disposto a me ajudar a solucionar minhas dúvidas desta dissertação.

Ao Prof. Dr. Celio Tamiguchi pela ajuda prestada durante os inícios e desenvolvimento deste trabalho.

Ao meu amigo Jairo pelas sugestões, constante disposição e boa vontade em me ajudar no desenvolvimento do programa e finalização do trabalho.

Ao Francisco "Chico" que me oriento nas etapas difíceis do trabalho e ao personal de estagiários do Departamento de Transportes do IPT.

A todos os amigos do Departamento de Engenharia Naval que direta ou indiretamente colaboraram no desenvolvimento deste projeto.

SUMÁRIO

Lista de figuras e tabelas
Resumo
Abstract
Nomenclatura

CAPÍTULO 1

1	1.1	INTRODUÇÃO
1	1.1	Generalidades

CAPÍTULO 2

5	2.	REVISÃO BIBLIOGRÁFICA
5	2.1	Conceito de Plano de Linhas
6	2.2	Conceito de Plano de Expansão do Chapamento
12	2.3	Princípios Fundamentais das Superfícies Desenvolvidas
12	2.3.1	Superfícies Regradas
13	2.3.2	Definição de Tangência
13	2.3.3	Superfície Desenvolvida
15	2.4	Prova da Existência de uma Superfície Desenvolvida
16	2.5	Superfícies de Casco Desenvolvidas
17	2.6	Aproximação do Método Gráfico

CAPÍTULO 3

20	3.	METODOLOGIA EMPREGADA PARA DESENHO DO PLANO DE EXPANSÃO E UMA PROVA DE SUPERFÍCIE DESENVOLVIDA
23	3.1	Metodologia Empregada para o Desenho do Plano de Expansão de Chapamento
25	3.1.1	Dados de Entrada
26	3.1.2	Resultados Obtidos
26	3.1.3	Desenvolvimento de uma Rotina na Linguagem LISP
27	3.2	Metodologia Empregada para Superfícies Desenvolvidas

CAPÍTULO 4

4.	ANÁLISE TEÓRICA PARA COMPROVAR A	
28	EXISTÊNCIA DE SUPERFÍCIES DESENVOLVIDAS	
28	Representação Matemática das Linhas do Casco	4.1
30	Descrição Vetorial da Ruling	4.2
34	Proposta de uma Estratégia de Busca	4.3

CAPÍTULO 5

5.	APLICAÇÃO DO PROGRAMA	
39	Aplicação do Teste de Superfícies Desenvolvidas para um Tronco	5.1
42	de Cone	5.2
44	Dados de Entrada do Programa "EXCHAP"	5.2.1
44	Dados de Entrada para Superfícies Desenvolvidas	5.2.2
44	Dados de Entrada para o Desenho de Expansão de Chapamento ..	5.3
45	Resultados do Programa "EXCHAP"	5.3.1
46	Resultado do Teste de Superfície Desenvolvida	5.3.2
48	Resultado do Desenho do Plano de Expansão de Chapamento	

CAPÍTULO 6

6.	CONCLUSÕES E SUGESTÕES	
52	Resumo do Trabalho	6.1
52	Comentários Gerais	6.2

55	REFERÊNCIAS BIBLIOGRÁFICAS	
----	----------------------------------	--

APÊNDICES

Apêndice I	Demonstração dos Teoremas Necessários à	
Apêndice II	Existência de Superfícies Desenvolvidas	
Apêndice III	Listagem dos Programas Desenvolvidos	
	Arquivos Empregados no Programa	

LISTA DE FIGURAS E TABELAS

FIGURAS

7	Figura 2.1	Plano de linhas típico de um navio
8	Figura 2.2	Metodologia para a retificação das balizas
11	Figura 2.3	Plano do expansão de chapamento
12	Figura 2.4	Superfície regrada, obtida por uma reta geradora
13	Figura 2.5	Definição de tangencia
14	Figura 2.6	Superfície desenvolvida
18	Figura 2.7	Rulings traçadas entre duas quinas de um navio pesqueiro
19	Figura 2.8	Rulings traçadas entre uma quina e o fundo de um navio pesqueiro
21	Figura 3.1	Fluxograma principal do programa "EXCHAP"
24	Figura 3.2	Representação de uma curva através de segmentos retos
31	Figura 4.1	Representação vetorial da rulling
34	Figura 4.2	Representação da área de regressão
35	Figura 4.3	Deformação do plano tangente
37	Figura 4.4	Busca do valor do senφ
40	Figura 5.1	Plano de linhas e balizas da embarcação proposta
41	Figura 5.2	Atração geral da embarcação pesqueira
43	Figura 5.3	Rulings obtidas num tronco de cone
47	Figura 5.4	Rulings obtidas no programa "EXCHAP"
49	Figura 5.5	Expansão de chapamento obtido no programa "EXCHAP"
50	Figura 5.6	Expansão de chapamento real do navio pesqueiro

TABELAS

45	Tabela 5.1	Fluxo dos arquivos do programa "EXCHAP"
51	Tabela 5.2	Áreas e Perímetros dos Planos de Expansão do Chapamento

O objetivo do presente trabalho é apresentar os critérios em que se baseia o conceito de superfícies desenvolvidas e gerar um programa que permita verificar a existência de superfícies desenvolvidas contidas nos costados que conformam o casco de um navio. Como complementação, são apresentados os conceitos preliminares e metodologia empregada para a obtenção da expansão de chapameento de um navio.

A fim de se poder aproveitar os grandes desenvolvimentos na área da tecnologia da construção, é importante que nos estágios iniciais de projeto sejam levados em conta alguns critérios visando dar maiores facilidades nas etapas de construção, conseguindo assim poupar insumos, materiais e horas de trabalho, refletidos na diminuição dos custos de produção, permitindo empregar com maior eficiência grandes máquinas automatizadas e interfacçadas com os recursos de controle disponíveis em computador. Um critério válido neste ponto de vista, aplicado para navios de médio e pequeno portes, é projetar suas superfícies laterais de forma a obter superfícies desenvolvidas.

RESUMO

ABSTRACT

Initial stages of a naval construction project should take advantage of the great technological advances in the field by using computerized design in combination with automated machinery. Reduced construction costs will be achieved by working more efficiently while saving materials costs and time. This is a basic criterion to apply to small and medium sized vessels when designing their sides to form a surrounding shell.

Objectives of this work are to present the basic concept of a surrounding shell, and to generate software to (validate or verify) its (presence or existence) in the hull of a ship. A complementary method for achieving expansion of the shell plating in a ship is also presented.

NOMENCLATURA

a	Longitude de um vetor
C	Curva inferior no espaço
D	Curva superior no espaço
n1	Vetor normal
P1	Ponto sobre a curva C
Q1	Ponto sob D sobre um plano tangente T1
Q2	Ponto sob D sobre um plano tangente T2
R	Linha regrada da superfície (Ruling).
S	Superfície desenvolvida
T1	Plano tangente desde o ponto P
T2	Plano tangente desde o ponto Q
tc	Vetor unitário tangente em C
td	Vetor unitário tangente em D
X	Produto vectorial de dois vetores
x	Eixo do sistema de coordenadas horizontal
y	Eixo do sistema de coordenadas de profundidade
z	Eixo do sistemas de coordenadas altura
$\epsilon(P1, Q1)$	Direção unitária do segmento P1Q1
g	Longitude de arco C
$\Delta\sigma$	Distancia desde Q1 (a um plano tangente T2)
Q2	Longitude de arco D
σ	Longitude de arco D
ϕ	Angulo entre as retas tangentes na s curvas C1 e C2

Estas novas técnicas, assim como o aperfeiçoamento em outras, melhoram os processos e métodos na produção naval, tendo-se sempre como objetivo poupar o máximo de insumos, e obter incremento na eficiência da produção. O fato é que a característica do mercado da construção naval nestes dias é marcada por severa competitividade, sendo a principal meta de um estaleiro entregar produtos com a mais alta qualidade no menor custo dentro dos prazos estabelecidos. As técnicas de engenharia desenvolvidas num estaleiro são

Os grandes avanços na tecnologia e o uso das novas técnicas de produção têm permitido melhorar e elevar as taxas de produção na construção naval, mediante o emprego de grandes máquinas automatizadas e interfacçadas com os recursos de controle disponíveis em computador, tais como CAD, CAM e CAE, conseguindo-se montar excelentes aplicações nas tarefas de projeto e manufatura. Além disso, com a reimplimentação dos conceitos de construção modular e trabalhos por zonas na construção naval, consegue-se obter maior qualidade dos produtos intermediários refletindo na melhoria dos acabamentos e uma redução nos tempos de entrega dos navios.

CAPITULO 1

1. INTRODUÇÃO

1.1 GENERALIDADES

importantes para a obtenção de redução de mão-de-obra nas tarefas que envolvem a construção do casco, nas quais os blocos manufaturados tem prioridade.

Além de trabalhar com êxito os métodos de construção modular, incorporando ao mesmo tempo os grupos tecnológicos GT, para o emprego destas metodologias de produção, é indispensável que o estaleiro consiga obter precisão e qualidade nas etapas de fabricação de blocos.

Mas, apesar da qualidade e dos níveis de produção alcançados, o fato de ainda acontecer erros na etapa de edificação prova que o problema de precisão ainda persiste. De acordo com as estatísticas apresentadas por GAO [1], no curso da fabricação de um bloco, os trabalhos de correção, tais como marcações e inspeções, corte oxiacetilênico, posicionamento da solda, lixamento e rechapeamento, e outros trabalhos auxiliares, causados principalmente pela imprecisão nas partes da construção do casco, ocupam mais de 20-22% no total do trabalho da fabricação do bloco. Logo, obter melhorias nos graus de precisão das partes que compõem o casco é o ponto crucial para elevar a produtividade na etapa de construção do casco do navio.

Atualmente, dentro dos surpreendentes desenvolvimentos ocorridos na área da computação, em especial dentro do projeto assistido por computador CAD, dispõe-se de tecnologia amplamente utilizada na construção do casco, podendo ser integrada às máquinas de corte numérico NC disponíveis nos estaleiros nas últimas duas décadas, levando a precisão da construção das partes planas do navio.

A precisão das chapas, de outro lado, é ainda uma situação desfavorável ligada diretamente aos limitados métodos de aproximação existentes, a maioria

partindo da idéia que toda a chapa do navio é expansível. O resultado é que, durante a etapa de edificação, ajustes finais como cortes e lixamentos, são difíceis de evitar, sendo difícil elevar a taxa de produtividade dos blocos, em condições extremas, a qualidade do produto é perdida.

Porém, existem três tarefas básicas para as correções necessárias sobre as chapas para conformação de estruturas. Os problemas descritos nos parágrafos anteriores, apresentam-se com maior ênfase para embarcações de médio e pequeno portes, tais como navios pesqueiros, embarcações de planeio, etc., que têm como característica comum a presença de quinas pronunciadas no casco. Uma forma de contribuir para redução dos erros é procurar trabalhar com superfícies desenvolvidas.

FERRIS [2] apresentou em 1967 uma série padrão de superfícies desenvolvidas. Segundo o autor não existe dúvida que uma chapa plana pode ser mais facilmente trabalhada numa superfície desenvolvida do que em outra com bojos, curvaturas ou torções e tendo ainda a vantagem de reduzir as tensões residuais na forma final. CALKINS [3] afirma que nestas superfícies é necessário só dobrar a chapa durante a construção, sem necessidade de aplicar outros processos que produzam tais torções, bojos e encolhimentos, permitindo que o material seja cortado diretamente da chapa plana, conseguindo aproveitar ao máximo os recursos automatizados de corte numérico NCC. Os baixos custos na construção, a facilidade de reparação e emprego de ferramentas simples para construção são as principais vantagens das superfícies desenvolvidas. Em função dos custos é evidente a preferência destas aos cascos chapeados.

O objetivo do presente trabalho é estudar um critério desenvolvido para testar a existência de superfícies desenvolvidas ou expandidas. Previamente serão discutidos os conceitos preliminares e uma metodologia empregada para a obtenção da expansão de chapameento de um navio.

CAPÍTULO 2

2. REVISÃO BIBLIOGRÁFICA

2.1 CONCEITO DE PLANO DE LINHAS

Para levar adiante um projeto de construção de um navio, prescindem-se de muita informação inicial, porém um plano que defina geometria e características da embarcação é dado por um plano de linhas. Este plano geralmente é preparado no tempo conceitual de projeto, para ter definidas as características de capacidade requerida, deslocamento e propulsão de acordo com os requisitos do armador.

Um plano de linhas, segundo EYRES [4], é um desenho, numa escala adequada, das linhas moldadas do navio arranjadas em plano, perfil e seção transversal. As seções transversais do navio estão igualmente espaçadas em balizas desenhadas entre as perpendiculares da proa e da popa, formando o plano do corpo principal da embarcação (*body plan*). Geralmente são divididas em dez seções com um traçado de meias balizas nas partes onde o casco muda de forma acentuada. A meia seção transversal é desenhada sempre que o navio é simétrico, desde a linha central. Assim mesmo a meia seção de proa é desenhada no lado direito da linha central e a meia seção de popa do lado esquerdo.

Planos preliminares são desenhados no início para correções no deslocamento, ângulo de trim, capacidade, etc. E assim mesmo devem ser dispostos num plano de elevação para obtenção da forma do casco da embarcação. No final, tem-se um plano exato da embarcação, plano de linhas completo, mostrando o perfil e contorno do navio, assim como o plano de formas de linhas de água em diferentes alturas acima da linha de base.

Obtido o desenho do plano de linhas, é preciso que o desenhista faça uma compilação dos pontos que definem o navio, isto é, uma tabela de cotas (*table off-sets*), que é uma lista de meias bocas, alturas das cobertas e contornos etc., em cada ponto das balizas. Um exemplo de tabelas de cotas para um navio pesqueiro é apresentado no apêndice III, nos arquivos "*Bocas.dar*" e "*Alturas.dar*". A desvantagem desta tabela de cotas é que a resolução obtida não é boa, pois o espaçamento entre balizas é muito amplo; portanto, para maior precisão, é necessário trabalhar com uma tabela de pontos completa em cada caverna.

Atualmente é comum que os grandes estaleiros possuam sistemas integrados de desenho computadorizado, já que a criação conceitual da forma do casco e as possibilidades de se ter uma boa precisão, são requisitos indispensáveis para atingir níveis de produção ótimos e competitivos. A forma do casco é mantida num sistema computadorizado como um modelo tridimensional "*wirframe model*", que permite visualizar as linhas moldadas do navio. ZEID [5], define um "*wirframe model*" como uma representação geométrica de um objeto, que pode ser representada matematicamente no computador através de pontos, retas, arcos e círculos, cones e curvas. A figura 2.1 apresenta um plano de linhas típico de um navio.

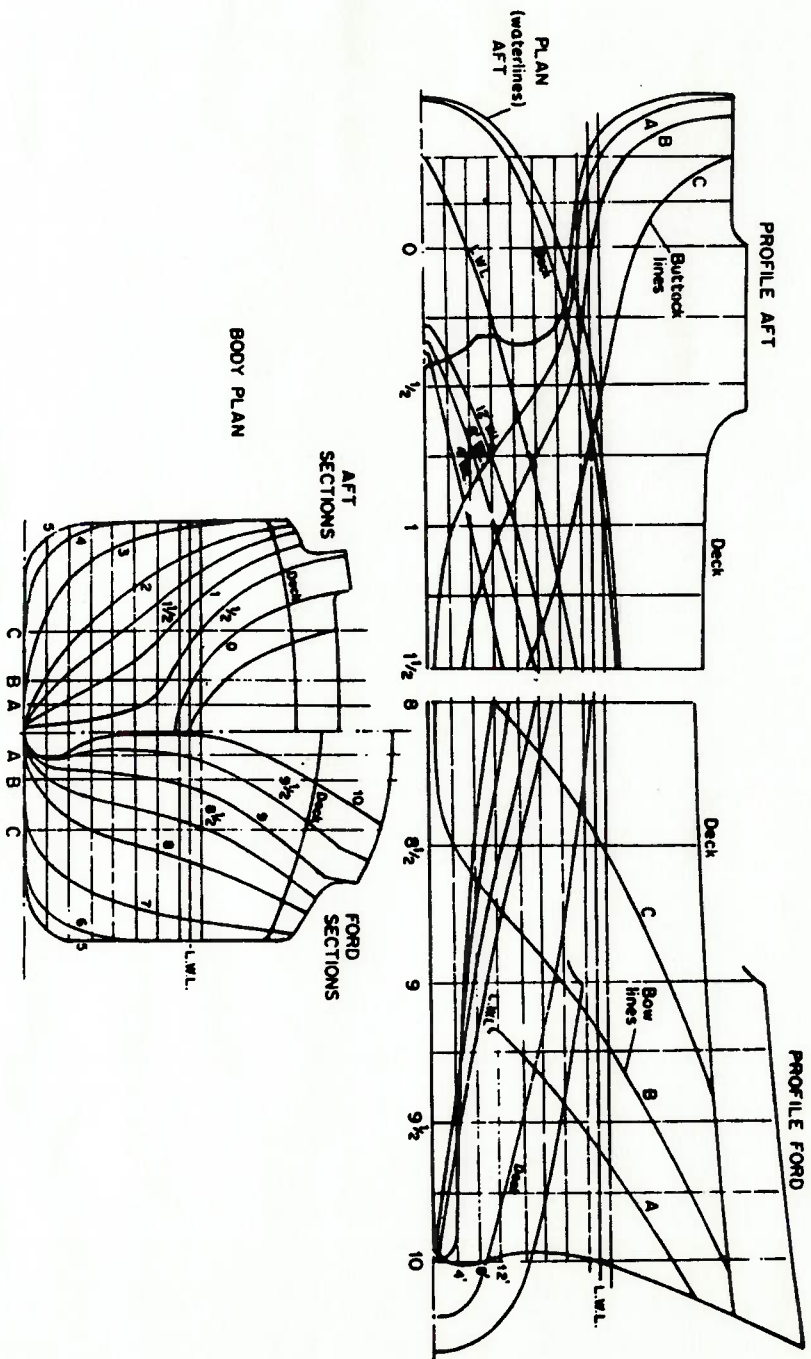


Figura 2.1 - Plano de linhas típico de um navio

2.2 CONCEITO DE PLANO DE EXPANSÃO DO CHAPEAMENTO

O plano de expansão de chapameento é requisito básico da construção do navio, visto que permite o arranjo das chapas e a obtenção de dados para a fabricação de cada uma delas. A metodologia típica usada na construção do plano de expansão é indicada por BARABANOV [6] através do desenho das balizas em sua própria posição longitudinal sobre a linha da quilha como linha base. O roteiro seguido é :

- se desenha uma grade contendo a linha de base e balizas do navio.
- com auxílio do plano de linhas se traça o perfil da embarcação.
- começando-se de uma das extremidades, espelho de popa ou proa, as balizas são retificadas, determinando a altura das quinas e traço do convés. Isto é feito tomando-se as medidas desde a linha central da quilha até o final da borda, para cada segmento que conformam o casco do navio, quinas e traço do convés são incluídos, como mostra a figura 2.2.

• as medições destes segmentos obtidos desde o desenho transversal no plano de linhas, são transferidas para as balizas correspondentes no plano de expansão, unindo os pontos correspondentes, obtendo a expansão do casco, com uma figura do casco com distorção.

Neste plano são desenhadas, após os elementos estruturais, cavernas, longitudinalais, anteparas, emendas de chapas, etc. e as demais linhas no modelo, tais como convés, fundo interno além das dimensões das chapas. A seguir, são desenhadas as linhas de cavernas e anteparas transversais. Geralmente cada chapa é designada por uma letra, sendo que as mais próximas da linha base são marcadas com "A" e as seguintes com B, C, D, etc. As chapas em cada linha de contorno são numeradas 1,2,3,etc. iniciando na proa. Ambos os lados têm a

mesma numeração, com uma especificação para distinção entre bombarde e estibordo.

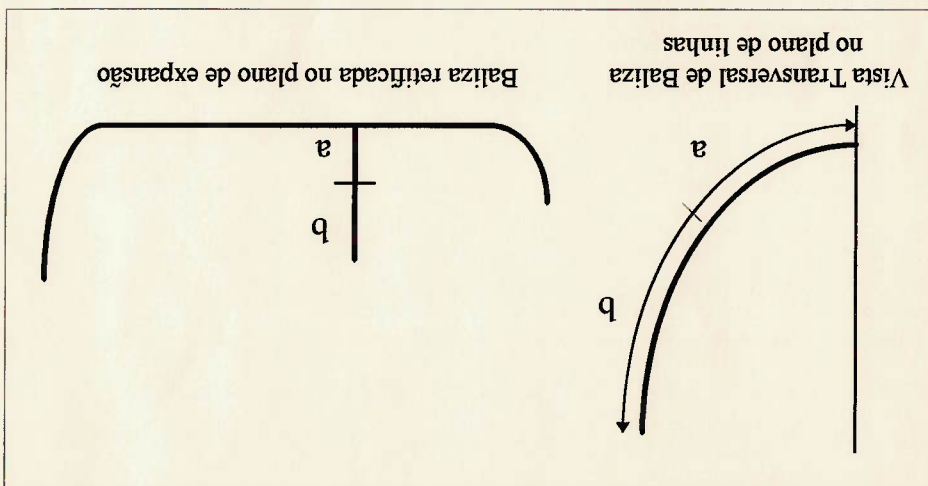


Figura 2.2 - Metodologia para a retificação das balizas

O principal propósito de um plano de expansão de chapamento é dar ideia de largura máxima da chapa que poderá ser usada sem deformações. Outra vantagem é a segurança de eliminar as diferentes interferências e dificuldades que se apresentam, por exemplo bordos de certas chapas sobre cavernas e anteparas. Essas costuras são eliminadas dentro de compartimentos e tanques, dado que o trabalho de solda não é feito rapidamente. Conexões de válvulas no casco podem ser ubicadas sem apresentar interferências que comprometam estruturas, evitando complicações.

De acordo com DORMIDONTOV[7] as finalidades do plano de expansão do chapamento são as seguintes:

- obter uma estimativa do material necessário para a construção do casco
- preparar a lista de encomenda de matéria-prima a ser adquirida

- orientar na divisão do navio em blocos
- visualizar a disposição geral das chapas.

Com a metodologia anteriormente descrita, só podem ser obtidas dimensões transversais corretamente, porém permite uma aproximação dos comprimentos das chapas tomando o espaçamento entre as cavernas como referência. A mesma metodologia permite obter um plano de expansão do convés, essencial para arranjo das chapas e estimativa de material.

O arranjo das chapas tendo por base um modelo tridimensional poderia ser representado num desenho de duas dimensões, chamado plano de expansão. Todas as dimensões verticais neste desenho são medidas ao redor do contorno do navio em vez da parte inicial de sua projeção. Esta técnica mostrará costado e fundo do navio como uma peça contínua.

Na figura 2.3 mostra-se um plano de expansão típico para um navio tanque, com a expansão do casco. Observe-se também que todas as chapas estão rotuladas em seqüência, visando facilitar sua identificação e ajudar nas tarefas de produção.

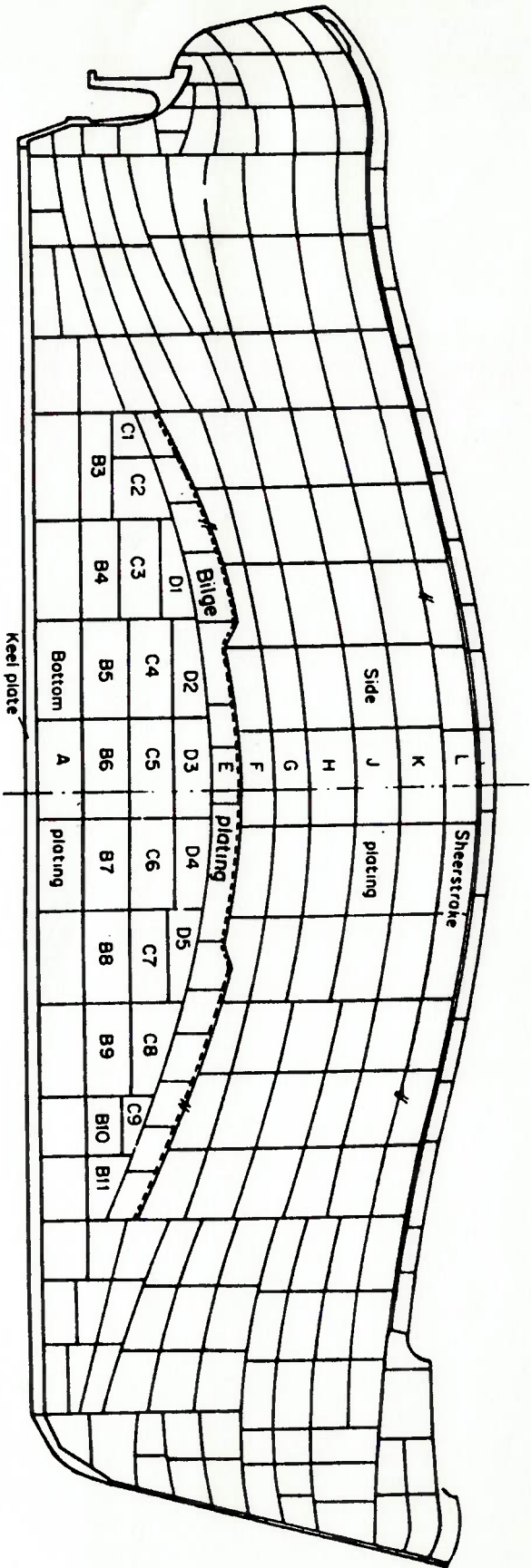


Figura 2.3 - Plano de expansão de chapamento

2.3 PRINCÍPIOS FUNDAMENTAIS DAS SUPERFÍCIES DESENVOLVIDAS

Para o emprego de superfícies desenvolvidas é preciso alguns conceitos básicos definidos pela geometria diferencial.

2.3.1 SUPERFÍCIES REGRADAS

Uma superfície regrada (*Ruled Surface*) é aquela gerada pelo varrimento de uma reta chamada *geratriz*, cuja direção é determinada por valores sucessivos de um parâmetro avançando continuamente ao longo de uma curva chamada *diretriz*, e o ângulo formado pela interseção entre *geratriz* e *diretriz* é diferente de zero. A figura 2.4 representa a definição.

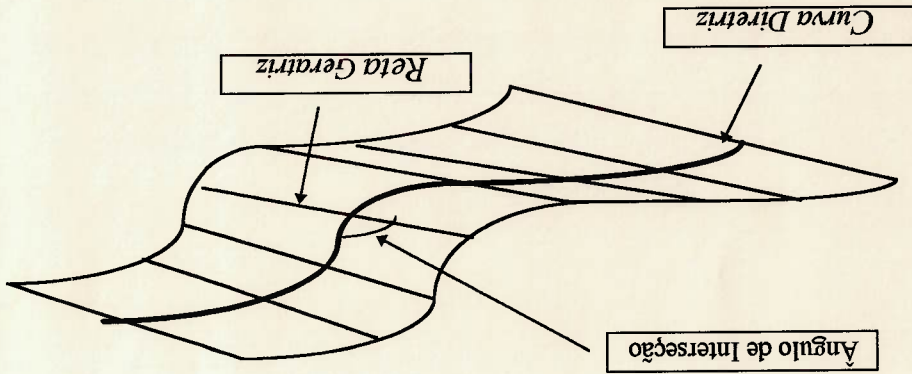


Figura 2.4 - Superfície Regrada, obtida por uma reta geradora

2.3.2 DEFINIÇÃO DE TANGÊNCIA

Se um plano T e uma superfície S coincidem num ponto P , conforme a figura 2.5, e têm uma normal comum através de P , então T é tangente a S no ponto P . Por extensão, se T e S coincidem numa sucessão de pontos, que determinam uma curva C , e se as normais comuns para T e S que se encontram em cada um desses pontos em C são paralelas, então T é tangencial a S ao longo da curva C . Evidentemente, as normais para T e S serão normais para C e para outras curvas ou linhas em qualquer S ou T que toque P .

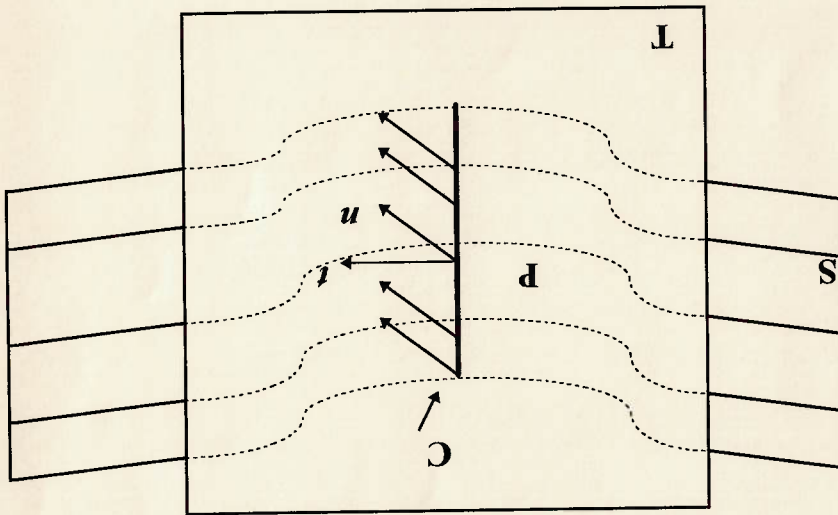


Figura 2.5 - Definição de tangência

2.3.3 SUPERFÍCIE DESENVOLVIDA

Uma superfície desenvolvida é uma superfície regradada que tem o mesmo plano tangencial sob uma única geradora. FARIN [17].

A partir da definição de superfície regradada, sabe-se que a diretriz deve encostar na superfície. Das definições de tangência e de superfície desenvolvida, observa-se que cada plano tangencial à superfície deve ser também tangencial à diretriz e para outras curvas na superfície que têm interseção com a reta geratriz. Uma observação neste ponto é que, se um plano é colocado tangente a uma superfície desenvolvida em qualquer ponto, a tangencialidade estará ao longo da linha. Esta linha levará o nome de "Linha Regrada da Superfície" (*ruling of the surface*); De agora em diante esta linha será identificada como "*Ruling*" para evitar erros induzidos pela tradução de conceitos. A continuidade dessas "*rulings*" mostrará que uma superfície desenvolvida poderá ser trasladada dentro de um plano, desenhando-se estas "*rulings*" em sua verdadeira magnitude isométrica, isto é, sem sofrer nenhuma deformação, sobre a projeção reta da curvatura gaussiana da superfície, então todas as dimensões da superfície poderiam ser mapeadas diretamente num plano.

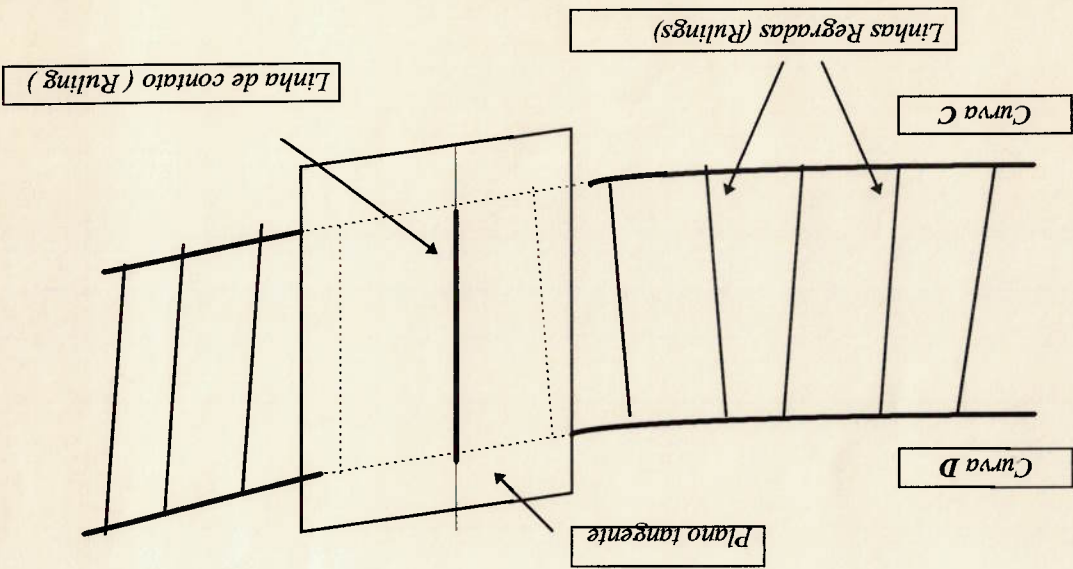


Figura 2.6 - Superfície desenvolvida

Tais superfícies podem ser descritas algebricamente escrevendo a equação de uma família de planos com um parâmetro, excluindo os planos paralelos um ao outro. Graficamente a superfície pode ser descrita pelo desenho de uma diretriz e um número suficiente de "rulings", onde, para mais de uma curva conhecida no espaço que se ajusta à superfície, qualquer uma destas ou todas podem ser usadas como diretriz. Os planos tangenciais na superfície serão tangentes a essas curvas. Isto é conhecido axiomáticamente para qualquer plano e poderá ser reproduzido graficamente pela interseção de duas retas ou por duas retas paralelas. A figura 2.6 representa uma superfície desenvolvida.

2.4 PROVA DA EXISTÊNCIA DE UMA SUPERFÍCIE DESENVOLVIDA

Dadas duas curvas no espaço, (figura 2.6) é possível a construção de uma superfície desenvolvida contendo ambas as curvas. Muitas superfícies regradas contendo ambas as curvas poderão existir, sem que seja possível desenvolvê-las. Para provar a existência de uma superfície desenvolvida é necessário o teorema da unicidade, demonstrado no apêndice I, onde mediante o emprego das relações vetoriais aplicadas em três pontos sobre duas curvas no espaço e dois planos tangentes nestes pontos, fica demonstrado que existe um e só um plano que cumpra as condições que geram estes tipos de superfícies.

2.5 SUPERFÍCIES DE CASCO DESENVOLVIDAS

Após definidas as informações necessárias do plano de expansão do chapameento, tem-se a etapa de fabricação dos blocos e seções, que conformaram o casco do navio. Porém, a obtenção de chapas com melhor precisão evitará erros posteriores e possíveis trabalhos de remendo. Neste sentido, visando boas aproximações na precisão, há métodos para expandir seções da chapa e obter a superfície desenvolvida respectiva.

Conforme definição dada por DORMIDONTOV [7], superfície desenvolvida é aquela cuja representação gráfica apresenta uma deformação no plano das verdadeiras dimensões e formas das partes projetadas do corpo do navio, para uma precisão específica.

Mas, apesar dos esforços dos métodos existentes, a precisão dificilmente era obtida, porque os métodos praticados para ajuste de superfícies desenvolvidas para curvas aceitáveis é um processo trabalhoso. O método é baseado em que qualquer superfície desenvolvida deva ser cônica ou cilíndrica. Qualquer uma dessas combinações poderá ser ajustada somente se eles puderem ser encontrados, e essa tarefa é uma sucessão de tentativas até chegar a um ajuste aproximado.

Segundo KILGORE [8] com tais métodos não pode ser encontrado um ajuste exato, sendo solução comum modificar as curvas originais do casco para conseguir os ajustes necessários, aceitos de forma aleatória. Porém os desenhos preliminares são apenas uma aproximação grosseira. Antes de escolher volumes, centros, parâmetros e formas desejadas de acordo com as exigências hidrodinâmicas e razões de estética, o projetista determina os resultados finais

logo após finalizado o esboço, podendo então fazer modificações ou voltar ao início do trabalho, em casos de resultados não satisfatórios.

Um método para gerar diretamente, desde o início, superfícies desenvolvidas a partir de pares de curvas no espaço foi apresentado por KILGORE [8]. O par de curvas no espaço poderia ser aquele representativo na forma transversal do navio, que indica mudanças no contorno, por exemplo, entre o perfil do navio na linha central e uma quina, uma quina e um bordo de coberta ou entre duas quinas. O método deveria prover superfícies para ajuste destas curvas, antes de fazer alterações às curvas originais para ajustar às superfícies, isto é, que os limites das curvas adotados no início do desenho deveriam aparecer, tanto quanto possível, inalterados no desenho final. Além disso, o método baseia-se nas propriedades de superfícies desenvolvidas.

2.6 APROXIMAÇÃO DO MÉTODO GRÁFICO

O procedimento gráfico de uma superfície desenvolvida é feito para descrever os planos que se encostam tangencialmente à superfície. Na maioria dos casos esses planos tangenciais poderão ser localizados diretamente e com precisão, mas em outros, a exata localização será uma tarefa cansativa em que se obtém solução aproximada.

Os fundamentos apresentados constituem as bases do método proposto por KILGORE [8], para gerar diretamente a forma do casco a partir de pares de curvas representativas de sua forma. O método proposto encontra as "ruilings" de uma superfície desenvolvida limitada por duas curvas típicas ou representativas do navio através de procedimentos gráficos manuais sobre o plano de linhas. O problema da precisão dependerá da qualidade do desenho final e do nível de habilidade e esforço investido pelo projetista no processo.

As figuras 2.7 e 2.8 mostram as rulings obtidas por KILGORE[8] empregando o método gráfico.

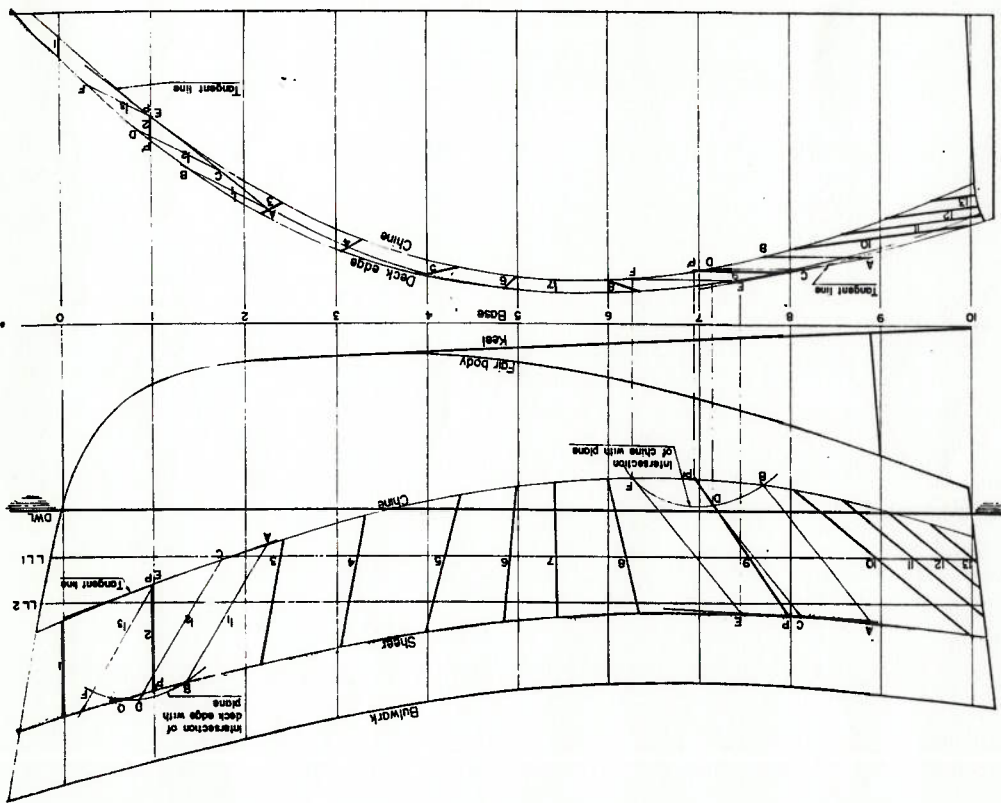
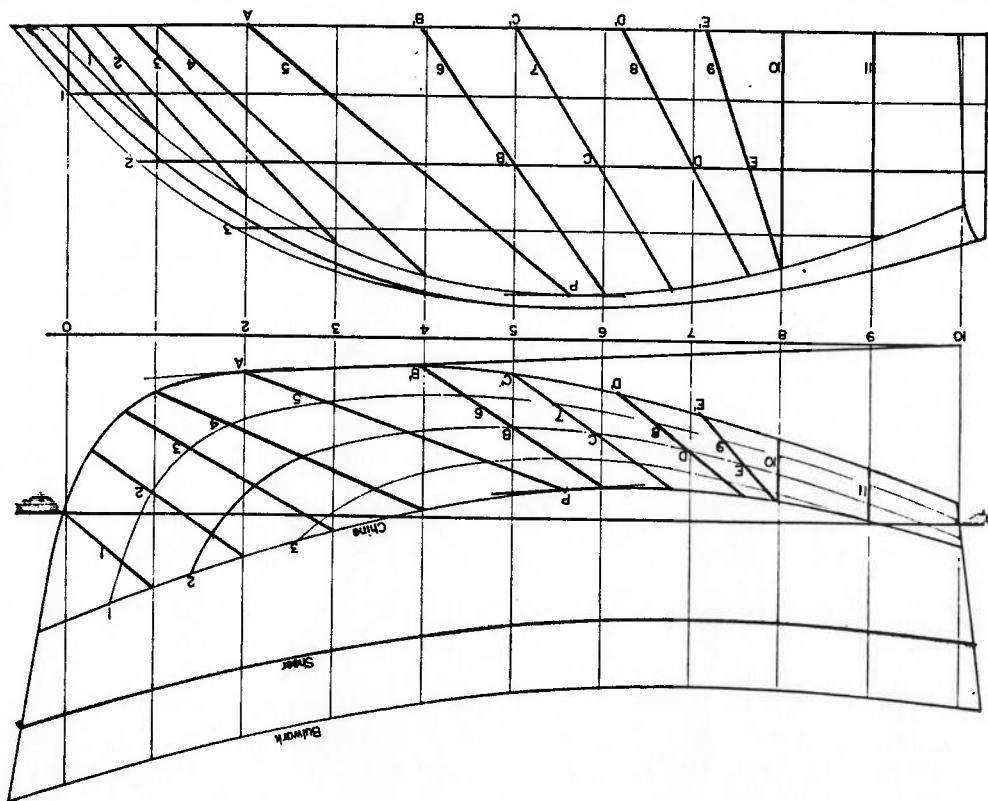


Figura 2.7 - Rulings traçadas entre duas quinas de um navio pesqueiro

Figura 2.8 - Rulings traçadas entre uma guias e o fundo de um navio pesqueiro



CAPÍTULO 3

3. METODOLOGIA EMPREGADA PARA DESENHO DO PLANO DE EXPANSÃO E UMA PROVA DE SUPERFÍCIE DESENVOLVIDA

A metodologia proposta para o desenho do plano de expansão de chapameento e teste da existência de uma superfície desenvolvida consiste no estabelecimento de uma sistemática composta por um programa desenvolvido em linguagem PASCAL, para execução em microcomputadores compatíveis com IBM-PC, sob ambiente operacional DOS, seguindo as diretrizes de iteração amigável com o usuário e prevendo como opções de saída arquivos em formato ASCII para análise e interface com o software de desenho AUTOCAD dentro do aplicativo AUTOLISP para o desenho do plano de expansão.

Anexo à dissertação está um disquete contendo a versão executável do programa "EXCHAP".

(b) O fluxograma do programa "EXCHAP" é apresentado na figura 3.1(a) e

Figura 3.1(a) - Fluxograma Principal do Programa EXCHAP.

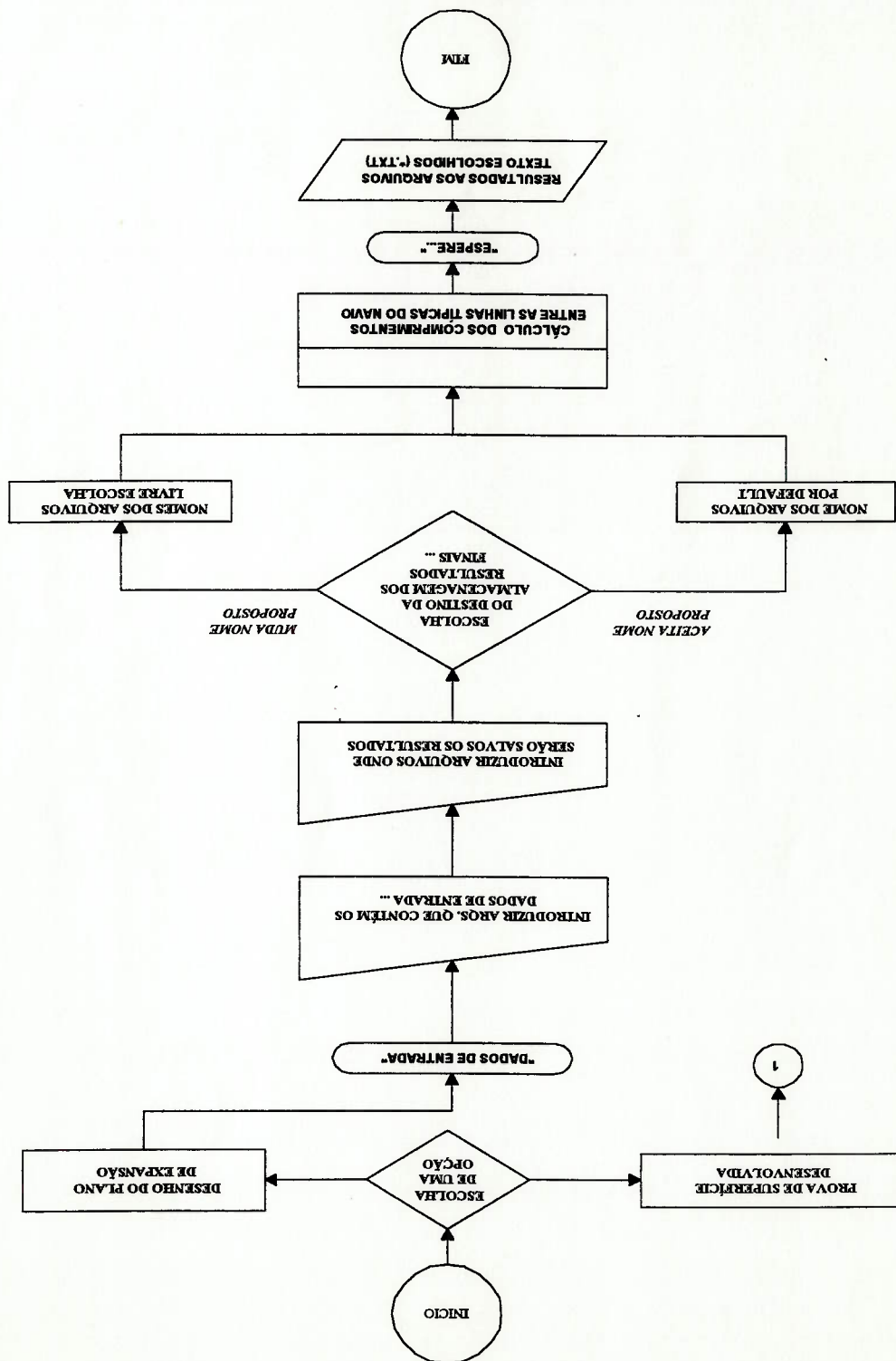
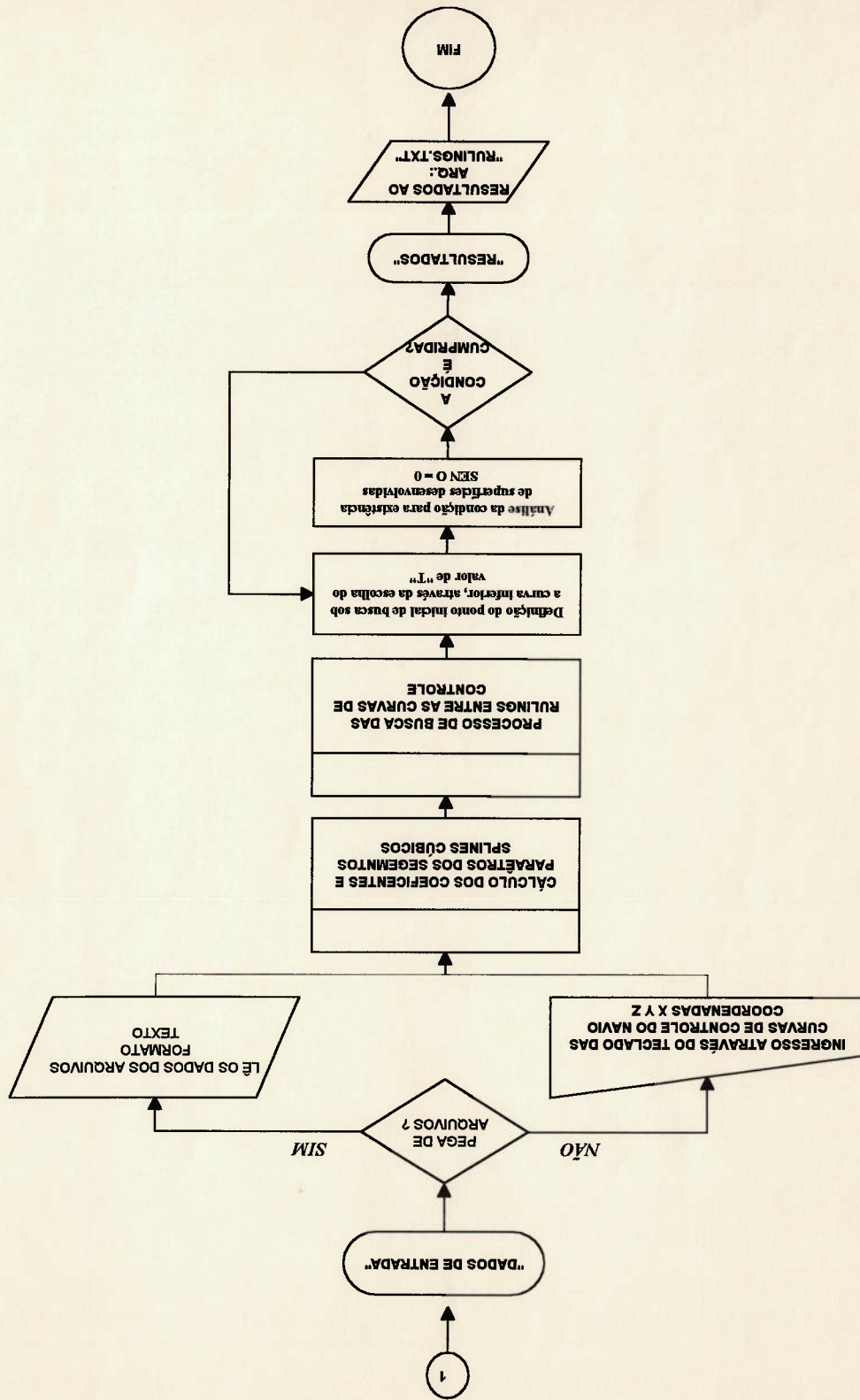


Figura 3.1(b) - Fluxograma Principal do Programa EXCHAP.



3.1 METODOLOGIA EMPREGADA PARA O DESENHO DO PLANO DE EXPANSÃO DE CHAPEAMENTO

Uma metodologia prática foi implementada para obtenção do plano de expansão de chapameento, fazendo uso da definição de um vetor no espaço, através da norma euclidiana ou magnitude de um vetor.

A idéia elementar é partir de três ordenadas de pontos no espaço, que representam as projeções e geometria do navio dentro de um sistema de coordenadas cartesianas, onde se representam nos três eixos x, y, z o comprimento longitudinal, a meia boca e as alturas, respectivamente. Por definição, o módulo de qualquer vetor é a distância entre dois pontos no espaço, que na verdade formam uma reta. Pelo teorema fundamental de Pitágoras, pode-se deduzir que o módulo de um vetor a :

$$|a| = \sqrt{ax^2 + ay^2 + az^2} \quad (3.1)$$

A estratégia planejada para abordar os cálculos é que partindo de cotas de projeções da geometria, consegue-se suavizar da melhor forma possível as linhas da curvatura de casco através de segmentos de retas sucessivas, previamente calculados como o módulo de um vetor definido entre dois pontos.

Como sugestão pode-se partir das tabelas de off-sets de desenho de qualquer navio para aproveitamento de maior quantidade de pontos. Uma representação para suavizar as curvas através de pontos intermediários, é apresentada na figura 3.2.

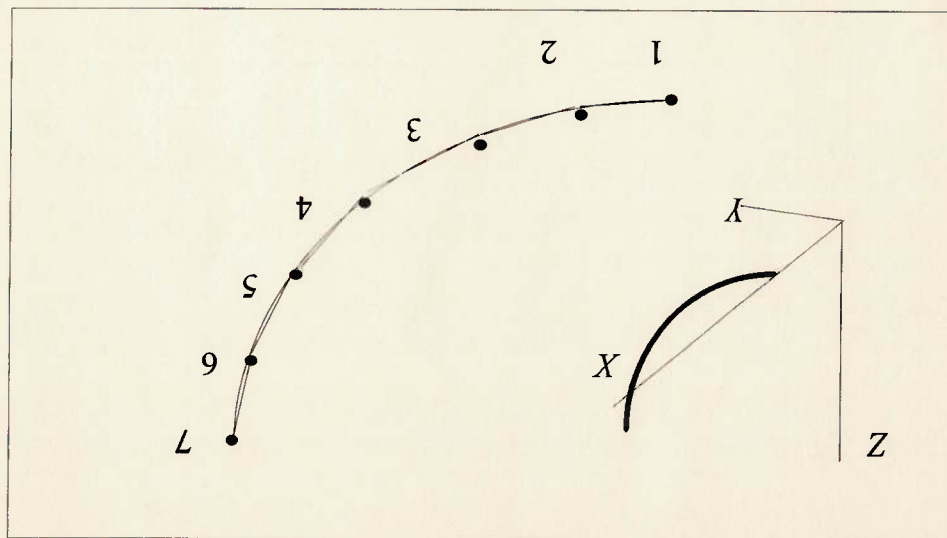


Figura 3.2- Representação de uma curva através de segmentos retos

Aplicando este princípio básico e sabendo das limitações próprias de um desenho de expansão de chapamento, como uma projeção em função da região cilíndrica da seção média do navio indicada por KUO [9], é possível montar um processo numérico para determinar as distâncias e comprimentos de cada baliza com a curvatura transversal sobre o eixo longitudinal. A vantagem deste processo é poder alterar a quantidade de pontos aumentando a precisão do desenho.

Para este caso, é preciso fornecer sempre sete cotas para definir a curvatura da baliza ou trabalhar cada segmento, mas sempre com sete pontos interiores. No exemplo, estas sete cotas são constituídas como os pontos da interseção entre uma baliza e as curvas principais do navio, posto que a forma transversal do casco é reta a aproximação feita através do módulo de um vetor tem uma boa aproximação.

3.1.1 DADOS DE ENTRADA

Os dados necessários para a execução do programa que desenha o plano de expansão de chapameento devem ser fornecidos em arquivos-texto, formatos ASCII no início. Cada sequência no programa é apresentada com uso de telas, criando um ambiente agradável ao usuário. A nível de demonstração, são fornecidos quatro arquivos com todas as informações, divididos em:

- **ALTURAS.DAT** : Correspondem às coordenadas XZ extraídas das tabelas de projeções, *off-sets*, para cada posição denotada pelas balizas ou cavernas do navio.
- **BOCAS.DAT** : Correspondem às coordenadas XY extraídas das tabelas de projeções, *off-sets*, para cada posição denotada pelas balizas ou cavernas do navio.

- **PERFIL.DAT** : Correspondem às coordenadas XYZ que permitem plotar o perfil do navio sob a linha central, tendo os respectivos dados para cada posição denotados pelas balizas ou cavernas do navio. O valor da coordenada Y deve ser igual a zero ($Y=0$), podendo empregar este mesmo arquivo para desenhar o perfil no plano.

- **INICIO.DAT** : Correspondem às coordenadas XYZ extraídas das tabelas de projeções ou do plano de linhas, para a posição inicial, onde nasce a linha, na proa do navio de cada linha típica assumida do navio.

3.1.2 RESULTADOS OBTIDOS

No término da parte do programa que desenha um plano de expansão, serão obtidos sete arquivos contendo as coordenadas de cada curva típica do casco expandida sobre o plano. De acordo com as opções iniciais podem ser mudados os nomes de cada arquivo de saída para melhor identificação.

No capítulo 5, para o caso demonstrativo, serão apresentados os resultados com maiores detalhes.

3.1.3 DESENVOLVIMENTO DE UMA ROTINA NA LINGUAGEM LISP

LISP é uma linguagem de programação dos primórdios da computação, que trabalha em forma sequencial por meio de listas, as quais representam comandos básicos de desenho do AUTOCAD ou outras tarefas a serem executadas. A dificuldade desta linguagem é o fato de serem listas sequenciais e não terem recursos poderosos para uma programação de alto nível como a linguagem PASCAL ou C+. A validade desta linguagem está na interface e capacidade que tem o AUTOCAD para aceitá-la, realizando as tarefas repetitivas com rapidez, segurança e eficiência.

No apêndice II, são apresentados as rotinas desenvolvidas para a leitura das curvas e o desenho no software AUTOCAD ver 12.0.

3.2 METODOLOGIA EMPREGADA PARA SUPERFÍCIES DESENVOLVIDAS

Tendo-se como dados iniciais duas curvas, são necessários três vetores para encontrar uma "ruling": os dois vetores tangentes em cada ponto extremo e o vetor da "ruling". Representando as curvas de controle por Splines, os vetores tangentes em qualquer ponto da curva são fáceis de calcular. O vetor da "ruling" é determinado pelo cálculo da diferença entre as coordenadas dos pontos nos extremos, DE BOOR[10] e LEITE[11],

As "rulings" serão formadas usando-se um método iterativo de ponto fixo. Um ponto sobre uma curva é fixado e o outro ponto na curva oposta deverá satisfazer à condição final de busca. Uma vez encontrado o ponto livre, a "ruling" é definida. Desta forma sucessivas "rulings" podem ser geradas até que todas são encontradas.

Logo, conhecendo todas as "rulings" pode-se analisar se a superfície que as contém cumpre a condição de não haver cruzamento, para ser considerada desenvolvida.

CAPÍTULO 4

4. ANÁLISE TEÓRICA PARA COMPROVAR A

EXISTÊNCIA DE SUPERFÍCIES DESENVOLVIDAS

A seguir é apresentada a metodologia proposta por NOLAN [13], que dá solução analítica para caso mais geral de superfícies desenvolvidas, baseada no método gráfico de KILGORE [8], para determinação das "rulings" de um plano tangente à superfície lateral da embarcação.

4.1 REPRESENTAÇÃO MATEMÁTICA DAS LINHAS DO CASCO

Antes de abordar a metodologia para teste de superfície desenvolvida, é preciso definir matematicamente as curvas que a contêm, conforme capítulo 2. Porém, deve-se escolher as curvas adequadas que melhor representem as variações do casco da embarcação, por exemplo à quilha, quinas nos costados do navio e outras. Uma vez escolhidas e definidas as linhas características de um navio ou curvas de controle, é preciso obter as equações matemáticas que melhor representam seu contorno físico.

Para a representação matemática destas curvas de controle existem duas classificações: os tipos **Paramétrico** e **Não-Paramétrico**. As dificuldades encontradas no uso das representações não-paramétricas limitam seu uso na definição do caso. Estas foram identificadas por CALKINGS [3].

No trabalho apresentado por NOLAN [13], foram utilizadas equações de tipo não-paramétrico, trabalhadas com o polinômio interpolador de THEILHEIMER [14]. Para os tempos atuais este enfoque apresenta as seguintes desvantagens:

- Trabalhar com representações não-paramétricas, isto é, definindo os valores de uma função $Y(X)$ para valores geométricos de X , é pouco apropriado para aplicação no computador.

- O polinômio emprega muito o uso de memória, pois origina procedimentos de cálculos extensos.

- O método do polinômio de Theilheimer é bastante desatualizado, pois data de 1961.

Neste trabalho, o tratamento abordado para representar as curvas de controle é feito mediante uma Spline Cúbica Paramétrica Natural, ROGERS [12]. Após obtenção das representações matemáticas convenientes, pode-se efetuar o teste de superfície desenvolvida.

4.2 DESCRIÇÃO VETORIAL DA RULING

No capítulo 2 foi definida como "ruling", uma reta formada por pontos em contato a um plano tangente à superfície. Sempre que o plano é tangente a qualquer parte da superfície ao longo da "ruling", será também tangente em qualquer curva encostada da superfície que cruze a mesma "ruling". Portanto, podendo-se encontrar dois pontos em cada uma das curvas encostadas sob a superfície e afirmando-se que suas respectivas tangentes nesses pontos encostam-se no mesmo plano, então esses dois pontos definem a "ruling".

Os vetores normais podem ser calculados pelo produto vetorial entre o vetor contido no plano determinado pela "ruling" e cada vetor tangente correspondente a sua curva, como segue:

$$N1 = r \times T1 \quad (4.1)$$

$$N2 = r \times T2 \quad (4.2)$$

Concluindo que $N1$ e $N2$ são paralelos, significando que ambos estão no mesmo plano, então r é uma "ruling". De fato, esta afirmação é verificada quando:

$$N1 \times N2 = 0 \quad (4.3)$$

$$(r \times T1) \times (r \times T2) = 0 \quad (4.4)$$

A figura 4.1 mostra uma representação vetorial da "ruling" formada entre duas curvas quaisquer.

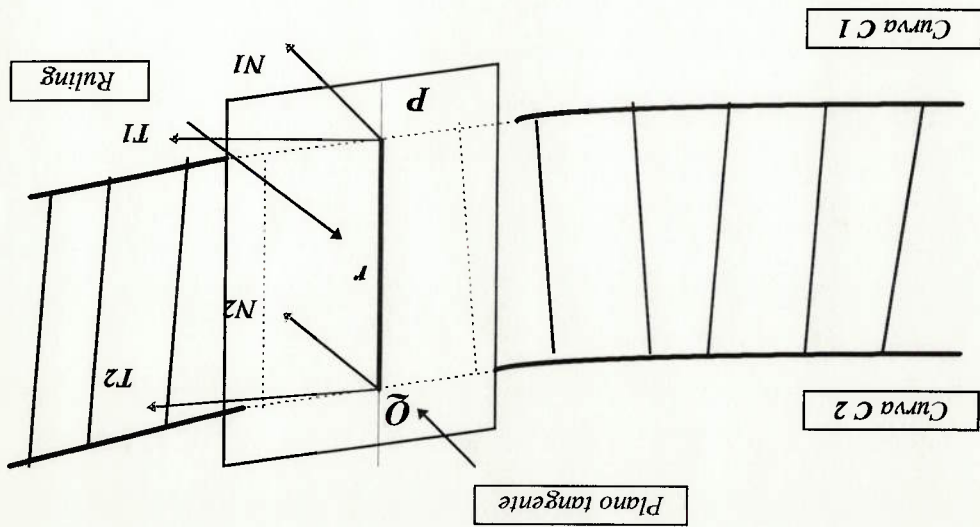


Figura 4.1 - Representação vetorial da ruling

onde:

$$T1(t) = \text{Vetor tangente sob o plano tangente desde o ponto } P$$

$$T2(t) = \text{Vetor tangente sob o plano tangente desde o ponto } Q$$

$$T1(t) = x'1(t)i + y'1(t)j + z'1(t)k = \text{Primeiras derivadas ou componentes do vetor tangente } T1(t).$$

$$T2(t) = x'2(t)i + y'2(t)j + z'2(t)k = \text{Primeiras derivadas ou componentes do vetor tangente } T2(t).$$

$$i, j, k = \text{Índices para os eixos de coordenadas } x, y \text{ e } z.$$

$$r(t) = \text{vetor ao longo da "ruling":}$$

Uma vez estabelecidas as condições iniciais, as mesmas devem ser expressas em função das curvas limites que contém a diretriz. Tais curvas são $C1$ e $C2$. Para NOLAN [13] essas curvas seriam funções de terceiro grau de xy com

um máximo de três raízes reais, e as definições mediante o polinômio proposto por THEILHEIMER [14] sendo em princípio um ajuste das curvas empregando splines cúbicas, com equações não-paramétricas. No presente estudo se empregará representação spline cúbica paramétrica das curvas $C1$ e $C2$. Em forma de matriz, os pontos de cada curva são expressos como:

$$\begin{aligned} C1(t) &= [x1(t), y1(t), z1(t)] \\ C2(t) &= [x2(t), y2(t), z2(t)] \end{aligned} \quad (4.5)$$

onde $x1(t)$, $y1(t)$ e $z1(t)$ são componentes do ponto avaliado num parâmetro t . Os valores requeridos para determinar uma "ruling" são:

- As tangentes em ambas as curvas sob os pontos finais.
- O vetor ao longo da reta que une os pontos extremos ou o vetor ao longo da "ruling".
- As normais ao plano nesses pontos extremos.

Os vetores tangentes $T1$ e $T2$ em P e Q são definidos por :

$$\begin{aligned} T1(t) &= f'1(t) = x'1(t) i + y'1(t) j + z'1(t) k \\ T2(t) &= f'2(t) = x'2(t) i + y'2(t) j + z'2(t) k \end{aligned} \quad (4.6)$$

O vetor da "ruling", definido pela a reta traçada entre os pontos P e Q , é

$$r(t) = [x1(t) - x2(t)] i + [y1(t) - y2(t)] j + [z1(t) - z2(t)] k \quad (4.7)$$

Substituindo adequadamente as equações (4.6) e (4.7) dentro das equações (4.1) e (4.2) obtêm-se os vetores unitários de N_1 e N_2 para o plano tangente formado por T_1 e r em P e T_2 e r em Q , em função do parâmetro t , os mesmos definidos por:

$$N_1(t) = [r(t) \times T_1(t)] / [|r(t) \times T_1(t)|] \quad (4.8)$$

$$N_2(t) = [r(t) \times T_2(t)] / [|r(t) \times T_2(t)|]$$

onde

$$N_1(t), N_2(t) = \text{Normais nos pontos } P \text{ e } Q$$

$$r(t) \times T_1(t) = \text{Produto vetorial do vetor } r(t) \text{ e } T_1(t)$$

$$|r(t) \times T_1(t)| = \text{Magnitude de } r(t) \times T_1(t)$$

A condição principal para existência de uma superfície desenvolvida requer que as normais contidas em cada extremo da "ruling" (r) sejam paralelas. A equação (4.3) expressa em sua forma paramétrica como o produto vetorial das normais é, como segue:

$$N_1(t) \times N_2(t) = 0 \quad (4.9)$$

Se a "ruling" definida ao longo das curvas de controle não tem interseções em nenhum ponto, a superfície é desenvolvida. As áreas definidas pelos pontos de interseção entre as "rulings" são conhecidas como *áreas de regressão*. O limite da área de regressão é chamado de *borda da regressão*. Ao longo da superfície onde acontece a regressão, a superfície só é desenvolvida entre o borde da regressão e a curva dada. Uma área de regressão conformada por interseção de "rulings" é ilustrada na figura 4.2.

Pode-se concluir, então, que para qualquer superfície definida por um par de curvas seja desenvolvida, as "rulings" não devem cruzar-se ao longo da mesma superfície. Dentro das áreas de regressão, as curvas de controle poderão ser modificadas até que as geratrizes deixem de se intersectar.

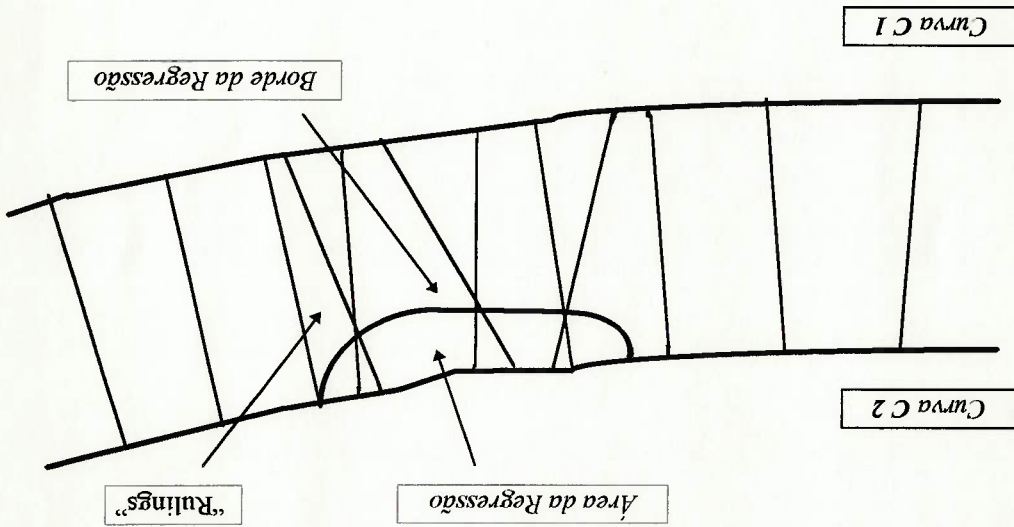


Figura 4.2 - Representação da área de regressão

4.3 PROPOSTA DE UMA ESTRATÉGIA DE BUSCA

Uma estratégia de busca pode ser estabelecida empregando a definição de produto vetorial, onde fazendo-se um arranjo conveniente podem ser relacionados os vetores unitários normais com a magnitude igual ao seno do ângulo entre eles, de acordo com:

$$| N1 \times N2 | = | N1 | \cdot | N2 | \cdot \text{Sen}(\phi)$$

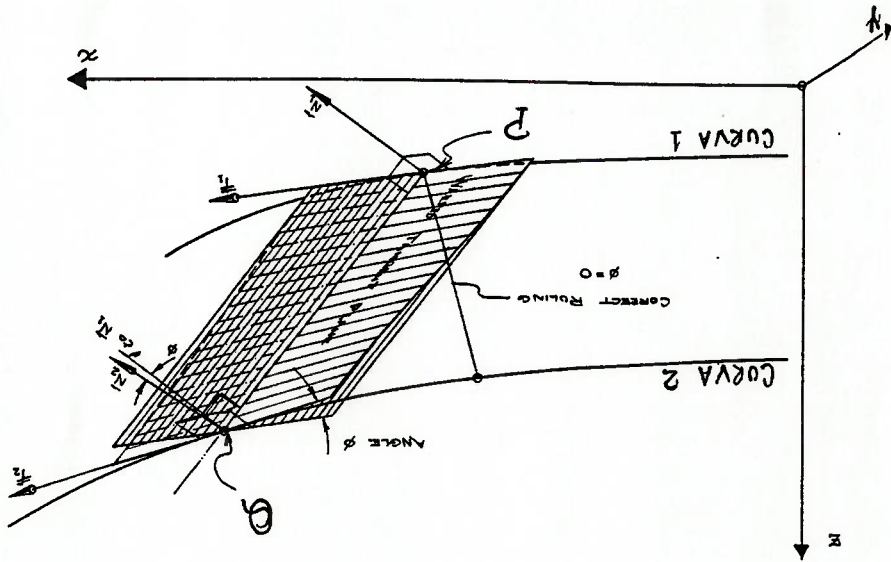


Figura 4.3 - Deformação do Plano Tangente

Através da figura 4.3, esta representação vetorial é apresentada considerando um extremo da "ruling" como ponto fixo sob a curva que a contém. O problema se simplifica à busca do ponto de interseção no extremo oposto sob outra curva que cumpra a condição de "ruling", isto é, o produto vetorial entre as normais deve ser zero. A técnica pode ser resumida na fixação de um extremo da "ruling", assumindo outros valores ao longo da curva oposta, resolvendo o seno do ângulo de inclinação de acordo com :

No ângulo ϕ tem-se uma interpretação física, sendo nesse ângulo onde deve dar-se uma deformação no plano tangente, objetivando uma tangência em ambas as curvas que geraram a "ruling".

$$\text{Seno}(\phi) = \frac{|N1 \times N2|}{|N1| \cdot |N2|}$$

(4.9)

$$\text{Seno}(t) = \frac{|N1(t)| \cdot |N2(t)|}{|N1(t) \times N2(t)|}$$

(4.10)

Para obter os valores de $x1(t)$, $y1(t)$ e $z1(t)$, devem ser feitos interpolações desde as curvas de controle, representadas por Splines. O vetor operação, isto é, $(x2(t), y2(t), z2(t))$, pode ser carregado sequencialmente. As soluções imaginárias e reais obtidas que não satisfizerem a condição podem ser descartadas, de acordo com o valor de t . No presente critério de busca pode-se obter eficiência e precisão, pela procura exaustiva de soluções.

Além disso, deve ser considerado o valor absoluto da função $\text{Seno}(\phi(t))$. Os valores oscilam entre zero e um, para qualquer valor de t . É fácil conhecer o momento em que a função é tangente à "ruling" com um valor de $\text{Seno}(\phi) = 0$, onde a função se encosta. Isto pode ser visualizado num plano tangente a uma superfície desenvolvida, fixado num extremo da geratriz e livre no outro extremo, variando até a solução.

No final o plano deve permanecer tangente em ambas as curvas, e desde que o vetor tangente na curva se desloque de maneira contínua, o $\text{Seno}(\phi)$ deve também mudar de forma contínua, tanto quanto t é válido. Uma representação do afirmado está na figura 4.4.

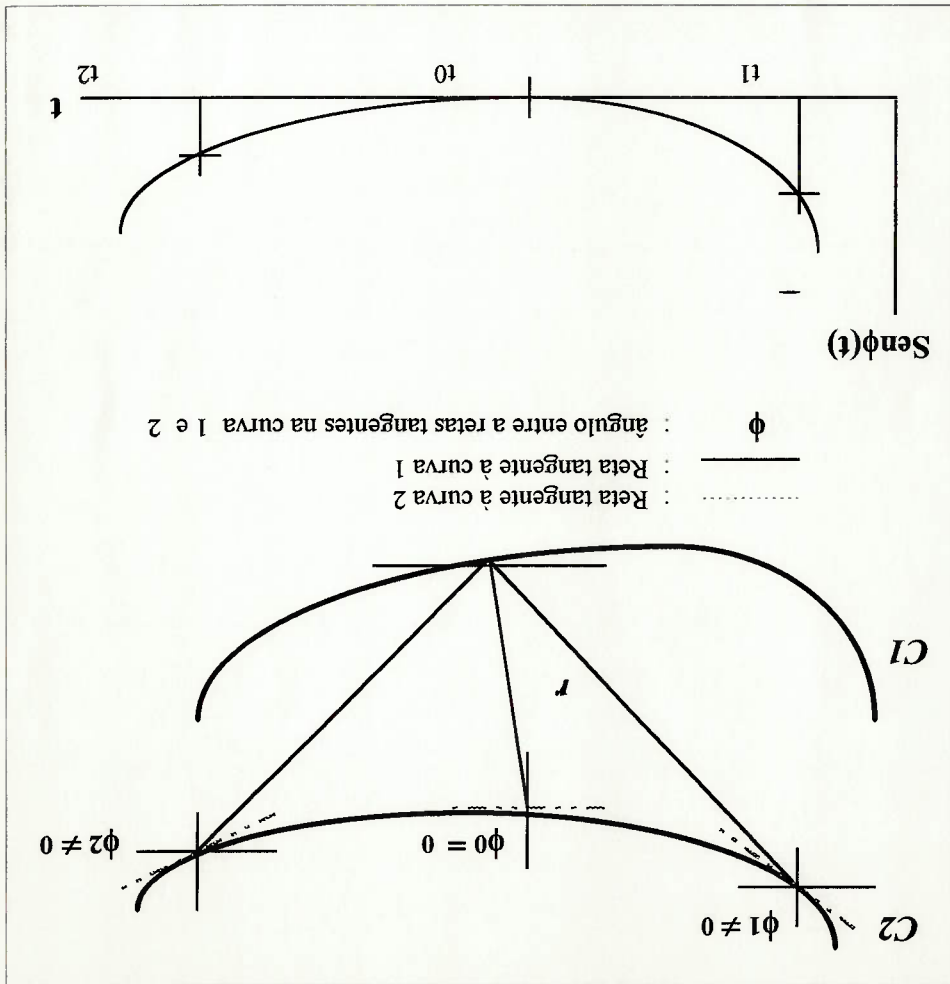


Figura 4.4 - Busca do Valor do Sen ϕ

Na procura do *Sen* ϕ (t), pode-se confirmar que apresenta uma forma côncava para cima próxima da solução, isto porque a interseção da tangente para o *Sen* (ϕ) no eixo do parâmetro t converge rapidamente.

A fim de obter um método de procura direto, é feita uma derivação numérica para determinar a inclinação da tangente com o *Sen* (ϕ) sob algum ponto de início da curva 1, fazendo uma escolha inicial do valor do parâmetro t .

obtendo-se os valores das coordenadas respectivas ao ponto inicial. A tangente, então, é extrapolada para o $Seno(\phi) = 0$, onde a interseção com o eixo das t dará um novo ponto de partida. O procedimento é repetido até que o valor do $Seno(\phi)$ fique próximo a zero. Com a finalidade de aproximar zero se trabalhará com uma margem de erro de 0,001. A busca dos zeros da função é conhecida também como método de “*Newton ou das Tangentes*”; HUMES [16] apresenta este método com clareza. A ausência de uma solução para qualquer segmento indicará que o critério de superfície desenvolvida não foi cumprido.

Uma das limitações apresentadas por este método de busca é a necessidade que o ponto inicial se encontre próximo a solução; porém dentro do processo de busca das “*rulings*” pode acontecer que o ponto inicial escolhido não apresente uma solução satisfatória. Para superar estes problemas, será necessário escolher um ponto de início mais perto da solução, partindo-se de um novo valor do parâmetro t para a curva onde se fixa o extremo da “*ruling*”:

CAPÍTULO 5

5. APLICAÇÃO DO PROGRAMA

No capítulo 3 foi desenvolvido o programa "EXCHAP" que proporciona duas ferramentas de interesse: o teste de superfícies desenvolvidas aplicável para os estágios de projeto e um processo computadorizado para o desenho do plano de expansão de um navio, necessário no início da construção.

Nos capítulos anteriores foi descrita a importância de formas do casco desenvolvidas para facilitar a construção, porém o programa permite a verificação da existência da superfície desenvolvida contida entre duas curvas típicas na lateral do casco da embarcação. No capítulo 4 foram expostos os principais critérios para a comprovação de uma superfície desenvolvida baseada na metodologia proposta por NOLAN [13]. A metodologia e o processo montado no computador para desenhar o plano de expansão foram expostos nos capítulos 2 e 3, respectivamente, seguindo-se o método tradicionalmente empregado nos estaleiros.

Neste capítulo, a parte do programa desenvolvido para teste da existência de uma superfície desenvolvida empregará uma embarcação cujo plano de linhas e balizas é mostrado na figura 5.1. Com a finalidade de proporcionar maior clareza sobre estes conceitos, é se apresentada uma aplicação do programa para um tronco de cone. Para desenhar a expansão de chapamento foram utilizados dados de outra embarcação pesqueira, cujo arranjo geral é mostrado na figura 5.2.

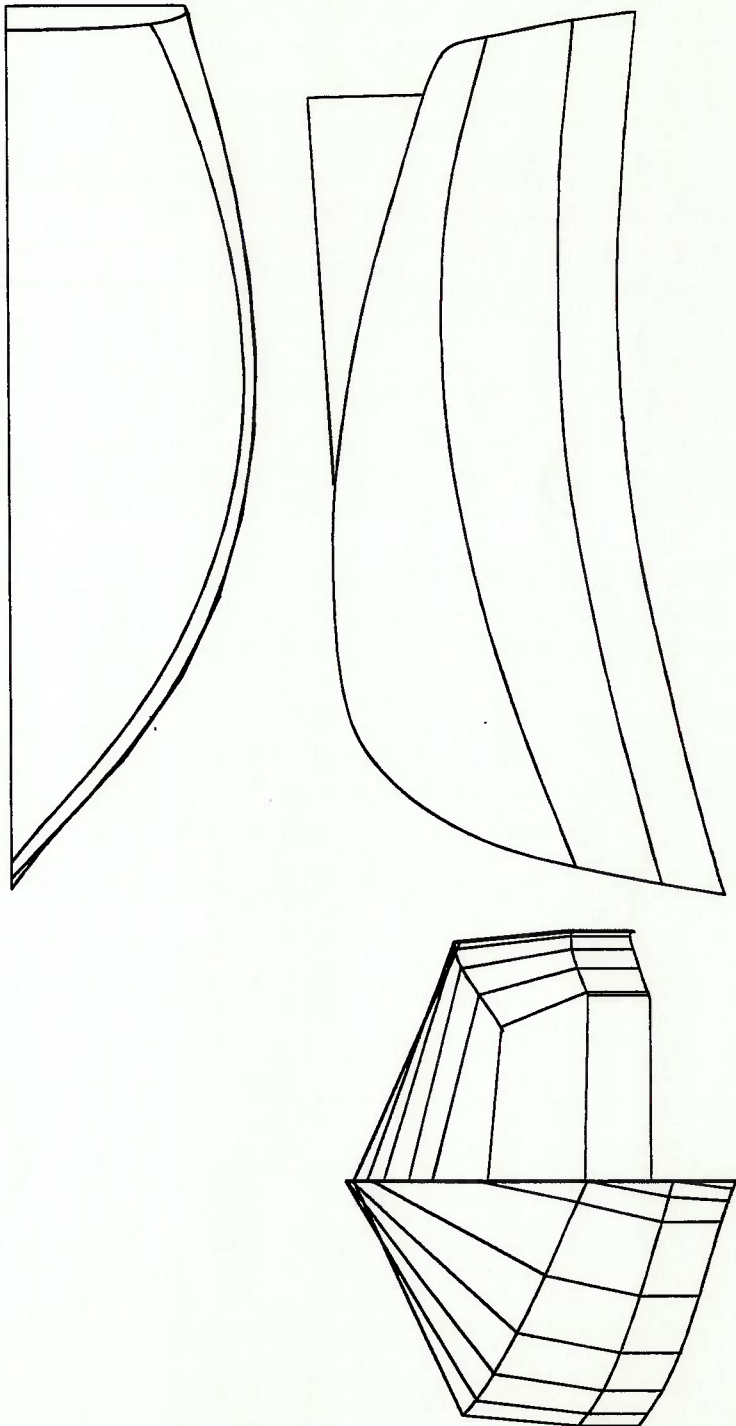


Figura 5.1 - Plano de Linhas e Balizas da Embarcação Proposta

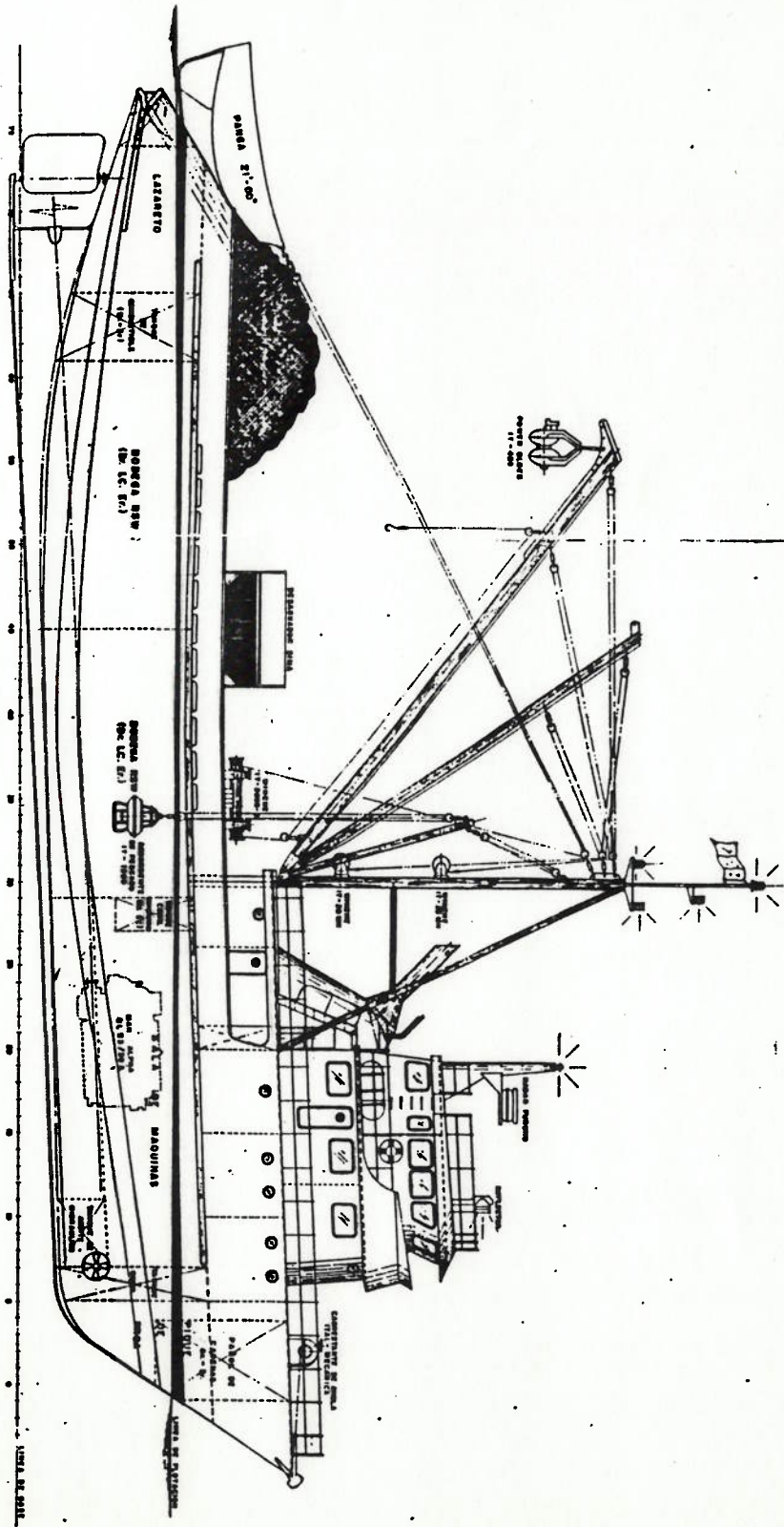


Figura 5.2 - Arranjo geral da embarcação pesqueira

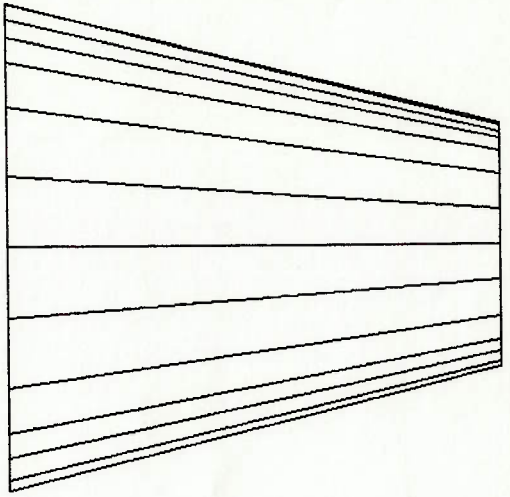
5.1 APLICAÇÃO DO TESTE DE SUPERFÍCIES DESENVOLVIDAS PARA UM TRONCO DE CONE

O teste de superfícies desenvolvidas foi feito para uma porção de tronco de cone de 120 graus, com a finalidade de comprovar os critérios revisados nos capítulos anteriores e conseguir maior clareza no emprego da ferramenta proposta.

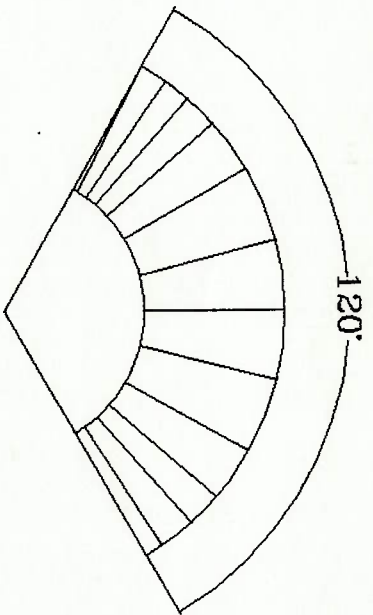
Sabe-se por geometria elementar que a figura geométrica do cone é composta por uma superfície desenvolvida, isto é, pode-se construir sua superfície lateral perfeitamente a partir um desenho feito no plano. O teste comprovou a validade da afirmação, mediante busca das "ruilings" sobre a área lateral. As curvas de controle empregadas foram os respectivos arcos das circunferências que conformam as bases do tronco de cone.

A figura 5.3 mostra os resultados obtidos graficamente sobre as duas vistas da porção de tronco de cone proposto, As "ruilings" foram encontradas e desenhadas sobre a área lateral. O fato dessas "ruilings" não terem cruzamentos entre si, comprova a existência de uma superfície desenvolvida.

O mesmo critério será empregado a seguir na parte lateral do casco de uma embarcação, onde a diferença em adotar os dois arcos de circunferência da base do tronco de cone como as curvas de controle, será a escolha de duas que sejam as mais características. O critério para definir a existência de superfície desenvolvida é rigorosamente ligado ao não cruzamento das "ruilings" ao longo do casco do navio.



Desenho das Rulings



Porcao de Tronco do Cone

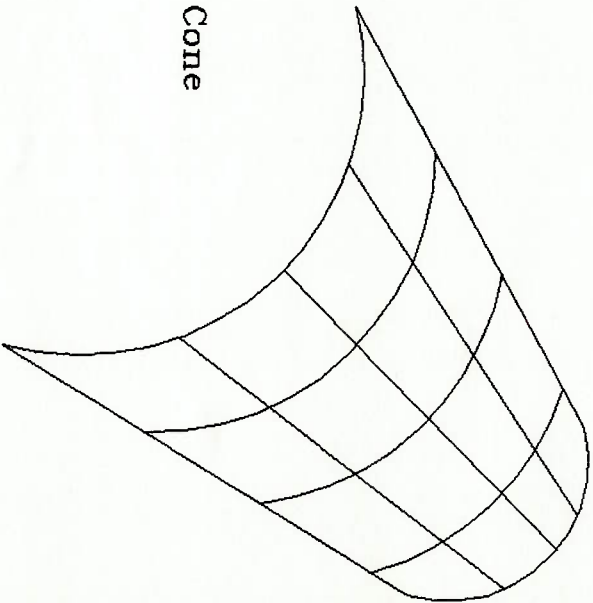


Figura 5.3 - Rulings obtidas num tronco de cone

5.2 DADOS DE ENTRADA PROGRAMA "EXCHAP"

O programa "EXCHAP" é composto de duas opções principais que permitiram verificar a existência de uma superfície desenvolvida e um desenho da expansão de chapamento de um navio. Porém, para cada parte do programa tem-se dois navios de cascos diferentes, isto é, feitos com a finalidade de apresentar estas aplicações de forma a mais clara possível.

5.2.1 DADOS DE ENTRADA PARA SUPERFÍCIES DESENVOLVIDAS

Para efetuar o teste é necessário definir as curvas de controle em que serão traçadas as "ruilings". Porém, os dados de entrada devem definir estas curvas em função de suas coordenadas x , y e z . O ingresso dos dados é fornecido mediante arquivos-texto no formato ASCII. Os arquivos empregados na demonstração são os seguintes:

- Perfik.dat
- QuinaInt.dat
- QuinaSup.dat
- Borda.dat

5.2.2 DADOS DE ENTRADA PARA O DESENHO DE EXPANSÃO DE CHAPAMENTO

Conforme apresentado no capítulo 3, os dados de entrada referentes à embarcação pesqueira da figura 5.2 devem ser obtidos através dos arquivos com as coordenadas x, y, z que contém a geometria da embarcação.

Os arquivos contendo os dados são apresentados no anexo III.

5.3 RESULTADOS DO PROGRAMA "EXCHAP"

Em seguida mostra-se os resultados obtidos no programa "EXCHAP" para as embarcações escolhidas. Os resultados dos cálculos são guardados como arquivos-texto, contendo as coordenadas das linhas e respectivos desenhos. A tabela 5.1 mostra os arquivos de entrada e saída para cada subrotina empregada no programa "EXCHAP". Para desenhar os resultados foi necessário o software AUTOCAD ver 12.0, mediante emprego de rotinas em linguagem de programação AUTOLISP aceitas pelo CAD.

Tabela 5.1 Fluxo de Arquivos do Programa EXCHAP

SAÍDA	ENTRADA	
Rulings.txt	QuinaInf.dat (CURVA 1)	SUPERFICIE DESENVOLVIDA
	QuinaSup.dat (CURVA 2)	EXPANSÃO CHAPAMENTO
Linha_1.txt, Linha_2.txt, Linha_3.txt, Linha_4.txt, Linha_5.txt, Linha_6.txt, Linha_7.txt	Alturas.dat, Bocas.dat, Perfil.dat, Inicio.dat	

Para maior clareza, os resultados do programa "EXCHAP" são apresentados em forma gráfica. Pelo fato de ter-se coordenadas das diferentes linhas expandidas, como também as "rulings", é necessário fazer as análises em função dos desenhos obtidos.

5.3.1 RESULTADO DO TESTE DE SUPERFÍCIE DESENVOLVIDA

A figura 5.4 apresenta a mesma embarcação proposta por KILGORE [8] com os desenhos, das "rulings" contidas no casco obtidos no presente programa. A busca das "rulings" é feita através de duas curvas de controle, sendo que a figura mostra estas retas contidas em duas porções do navio. A primeira delas foi determinada entre as duas quinas principais, contidas na rodada inicial do programa, e as outras entre a quina inferior e o fundo da embarcação (Perfilk.dat), com os arquivos fornecidos para este fim.

Uma vez rodado o programa, gerou-se um arquivo chamado "RULINGS.TXT" contendo as coordenadas das "rulings" correspondentes à superfície do casco. Porém, sempre que se rodar o programa para duas curvas diferentes, o nome deste arquivo deve ser trocado, ou guardados os resultados obtidos graficamente, com a finalidade de não perder os cálculos efetuados.

Fazendo-se uma comparação entre as "rulings" determinadas pelo método gráfico de Kilgore, mostradas nas figuras 2.7 e 2.8 e as obtidas no programa "EXCHAP", pode-se comprovar que existe uma boa aproximação com os resultados obtidos, as "rulings" encontradas matematicamente são muito próximas das desenhadas pelo método gráfico. Além disso, a validade da prova é feita através do desenho e comprovação visual de não cruzamento das "rulings" que neste caso foi cumprida. Portanto, o teste implementado no computador comprovou que as áreas laterais do casco submetidas à análise são superfícies desenvolvidas.

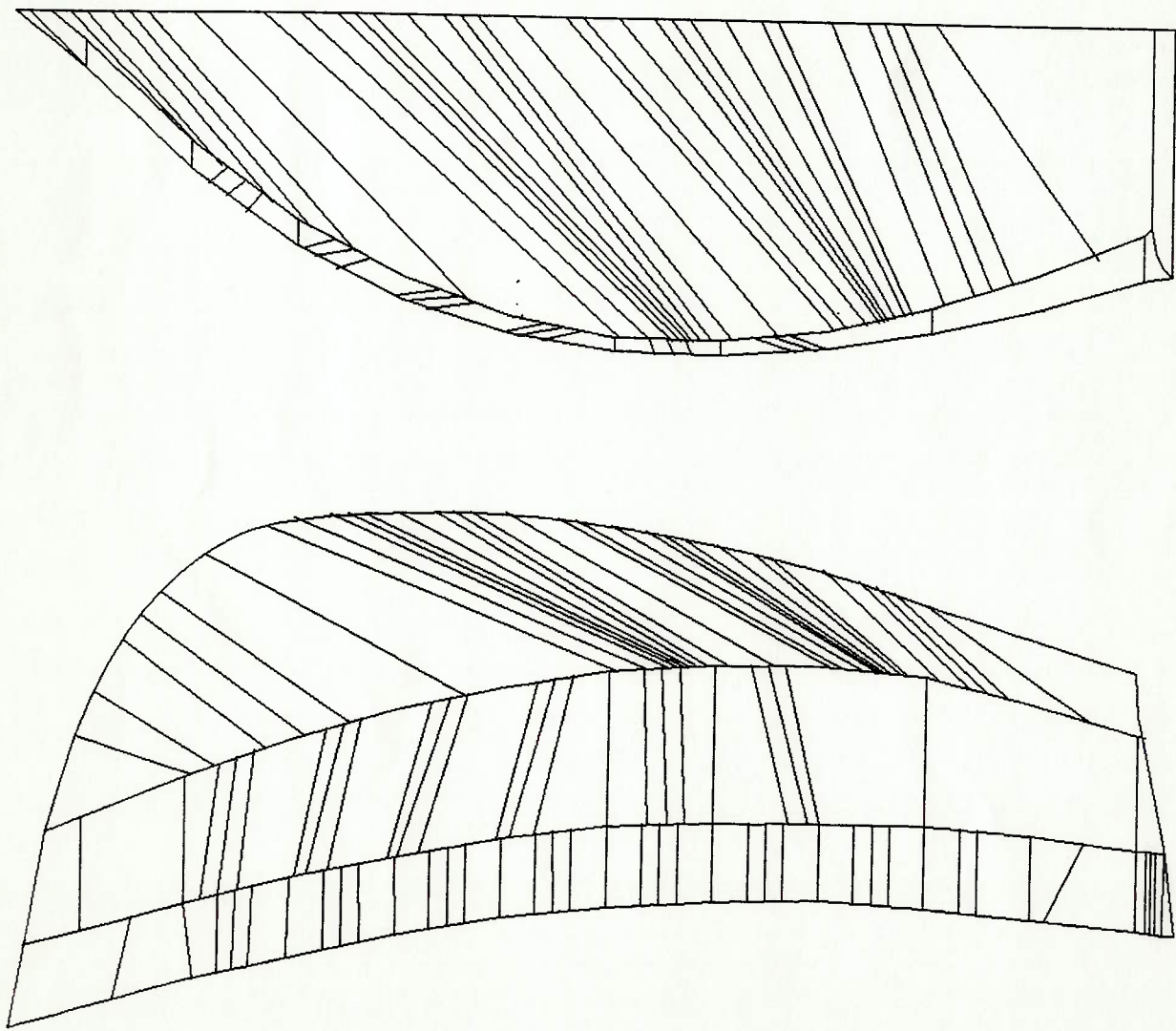


Figura 5.4 - Rulings Obtidas no Programa "EXCHAP"

5.3.2. RESULTADO DO DESENHO DO PLANO DE EXPANSÃO DE CHAPEAMENTO

A figura 5.5 apresenta o desenho obtido no presente programa, e a figura 5.6 um plano de expansão seguindo o método tradicional. Com a finalidade de efetuar uma análise comparativa entre os desenhos do plano de expansão de chapapeamento, na tabela 5.2 são apresentadas as respectivas áreas e perímetros para um arranjo de chapas mostrado nas figuras citadas.

Os cálculos das áreas e perímetros foram feitos empregando as ferramentas disponíveis no software AUTOCAD ver 12.0. A quantidade de chapa estimada de acordo com dados obtidos do plano de expansão feito da forma tradicional foi de 315,47 m² e a do plano de expansão desenhado com o programa "EXCHAP" foi de 316,92 m², o que dá erro de 1,45 m² de chapa a mais, representando 0,45 %, que para índices de construção é aceitável. Além disso, deve-se ter em conta que o plano de chapapeamento é feito para estimar a quantidade e fazer o arranjo das chapas, porém em função do desenho obtido do programa, cumpre com os requerimentos iniciais.

A precisão do desenho poderá ser melhorada em função dos dados de entrada do programa, jogando uma maior quantidade das coordenadas que representam a geometria do navio.

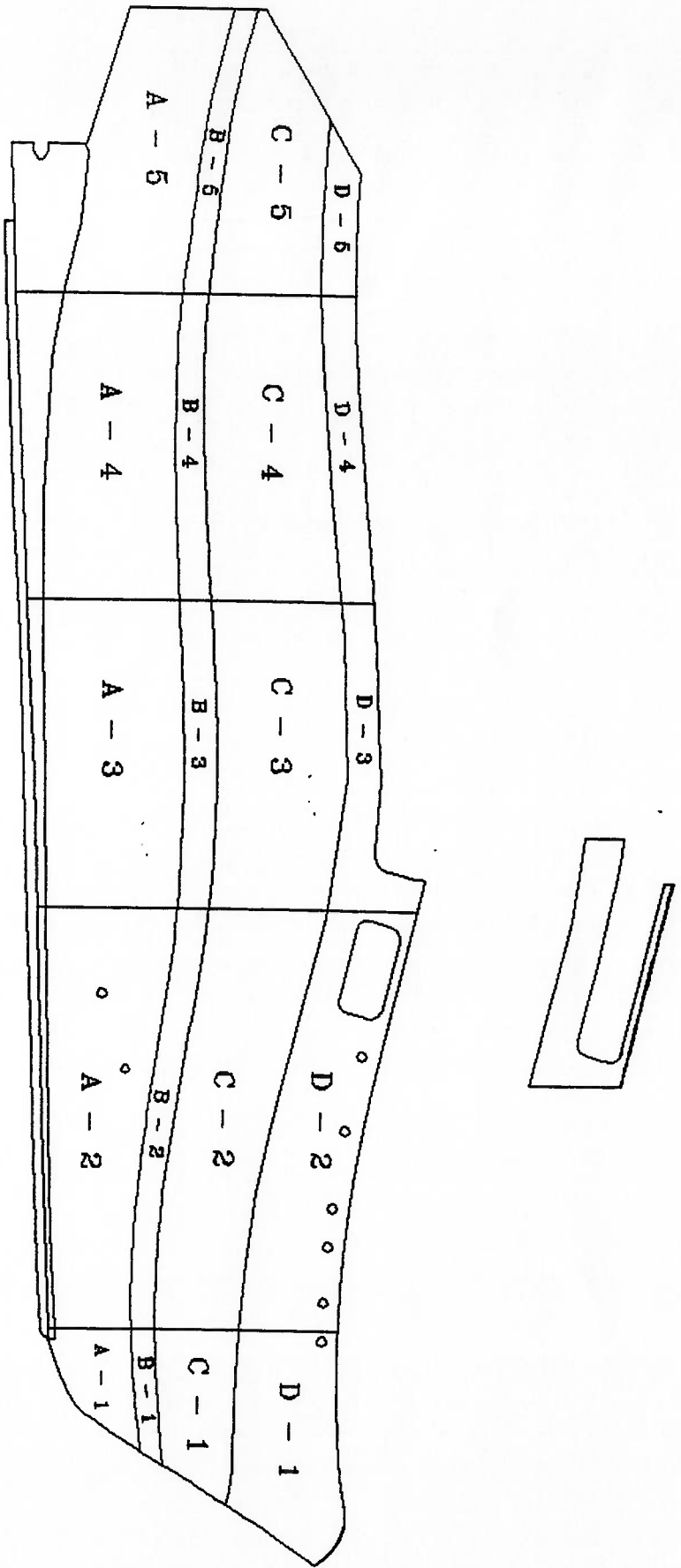


Figura 5.6 - Expansão de chapamento do navio pesqueiro

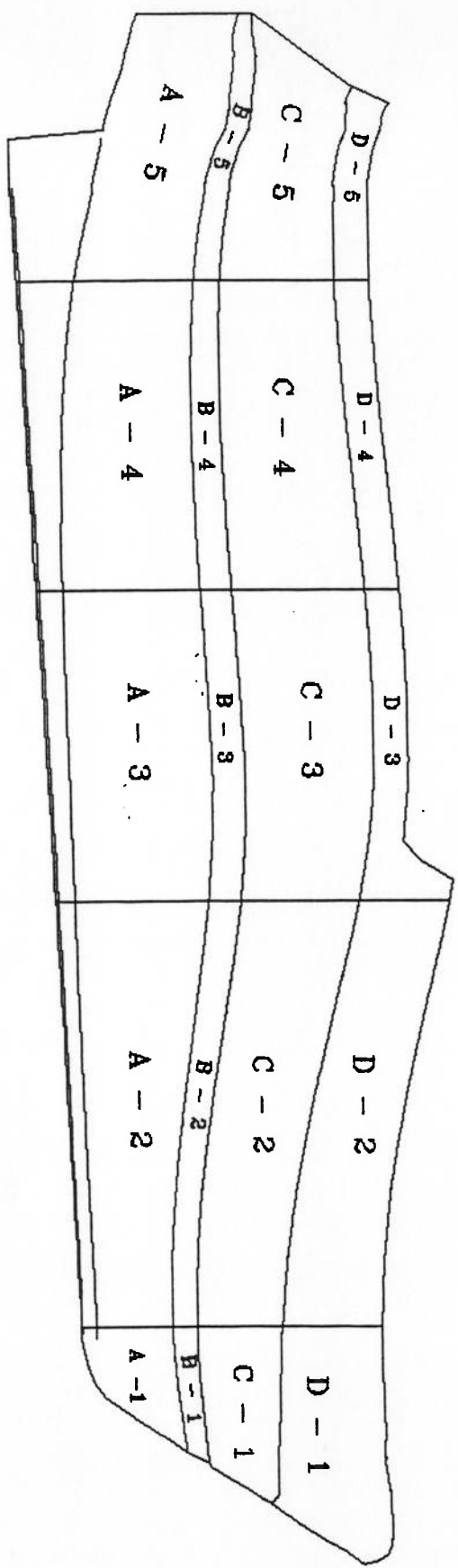


Figura 5.5 - Expansão de chapamento obtido no programa "EXCHAP "

Tabela 5.2 Áreas e Perímetros dos Planos de Expansão do Chapamento

Programa "EXCHAP"		Expansão Tradicional		Programa "EXCHAP"	
CHAPA	ÁREA	PERÍMETRO	ÁREA	PERÍMETRO	PERÍMETRO
A-1	3,6422	9,6397	4,0408	10,0931	27,9915
A-2	31,0115	27,9920	31,0086	27,9915	23,3957
A-3	28,8202	23,4132	29,0043	23,3957	23,0176
A-4	27,4759	23,0339	27,4505	20,4708	20,2424
A-5	23,9063	21,5601	20,4708	2,0793	8,2246
B-1	2,1239	8,3205	2,0793	8,4310	17,7698
B-2	8,2775	24,0071	8,4310	6,8682	15,3132
B-3	6,8093	17,9250	6,8682	6,1747	12,5625
B-4	6,2026	17,7830	6,1747	3,6570	28,0320
B-5	4,9610	16,8313	3,6570	26,9653	22,8962
C-1	8,2085	12,5782	8,1328	26,4703	22,7948
C-2	30,1379	28,0842	30,1520	17,7346	18,4886
C-3	26,9105	22,8794	26,9653	14,4107	16,6064
C-4	26,1924	22,7222	26,4703	28,1497	27,7633
C-5	16,5356	19,0663	17,7346	13,4919	19,6735
D-1	14,3230	16,5626	14,4107	7,4338	18,1613
D-2	28,5778	27,8956	28,1497	10,5521	11,8500
D-3	10,4022	20,5597	13,4919	4,7900	
D-4	7,3580	18,1054	7,4338		
D-5	3,5932	10,5521	4,7900		
ÁREA TOTAL	315,4695			316,9163	

CAPÍTULO 6

6. CONCLUSÕES E SUGESTÕES

O objetivo deste capítulo é fazer uma apreciação crítica do trabalho realizado, apontando as conclusões mais importantes obtidas no seu desenvolvimento e apresentar, por outro lado, algumas sugestões para a continuação do tema. É apresentado um resumo da pesquisa desenvolvida destacando a importância das superfícies desenvolvidas na construção naval.

6.1 RESUMO DO TRABALHO

Foi estabelecido como objetivo do trabalho a revisão do critério de superfícies desenvolvidas, e de outros conceitos básicos da construção naval, desenvolvendo-se uma sistemática para o desenho do plano de expansão de chapamento para um navio e verificação da existência de superfícies desenvolvidas aplicadas à superfície lateral de um casco.

A parte inicial do trabalho consistiu da definição dos conceitos necessários de plano de linhas de navio e plano de expansão de chapamento, destacando sua importância no início da construção. Foi também definida uma superfície desenvolvida através das definições geométricas clássicas e sua aplicação no meio naval, através da aplicação no casco do navio. A seguir, foi conduzido um estudo sobre os métodos manuais existentes para obter o desenho do plano de chapamento e para testar a existência de superfície desenvolvida numa porção lateral do casco de um navio.

Em continuação, desenvolveu-se um procedimento computadorizado de cálculo das coordenadas respectivas para a obtenção do chapameento de um navio segundo teoria proposta por Baravanov e a verificação de superfícies desenvolvidas baseada no método gráfico de Kilgore, culminando no desenvolvimento do programa "EXCHAP".

6.2 COMENTÁRIOS GERAIS

Através do estudo apresentou-se um critério de superfícies desenvolvidas aplicadas à superfície lateral de um casco, como alternativa para aumento da precisão na construção de embarcações de médio e pequeno portes, que requer dos projetistas navais o uso destes conceitos visando maiores facilidades, tanto nos momentos iniciais dos estágios da construção, como é o corte das chapas planas, como nas etapas de fabricação e montagem através da simplificação dos processos de conformação das chapas.

O presente trabalho fornece ferramenta computacional para testar efetivamente a existência de uma superfície desenvolvida contida entre duas curvas de controle. As limitações deste teste estão na forma de representar estas curvas, podendo ser implementado outro tipo de representação além da Spline Cúbica Natural utilizada. Esta prova auxilia no projeto do navio, modificando linhas e curvas da forma da embarcação de maneira eficaz, obtendo um produto final dentro dos requisitos descritos.

Este teste não permite modificações das curvas, mas, somente uma verificação exaustiva da superfície lateral do casco do navio.

A outra ferramenta proposta dentro do programa "EXCHAP", é a obtenção do plano de expansão de chapameento de uma embarcação, desenhada através do AUTOCAD ver 12.0 de forma rápida e que possibilita o manejo dos dados da geometria do navio dentro do computador. A precisão alcançada no caso exemplo apresentado foi ótima, posto que o erro obtido foi de 1,45 m² de chapa a mais, representando 0,45 %, perfeitamente aceitável para os índices de construção. Tendo-se em conta que a utilidade do plano de chapameento é fazer a estimativa da quantidade e arranjo das chapas, comprovou-se que o programa rodado para os dados fornecidos com as coordenadas do navio permite ter resultados que cumprem com os requisitos iniciais da construção naval.

Como trabalho futuro, planeja-se a complementação do presente estudo, modificando as curvas de forma a obter superfícies desenvolvidas correspondentes. A grande vantagem deste tipo de superfície é poder expandir a chapa no plano e aproveitar ao máximo as facilidades do controle de corte numérico NCC. Um processo neste sentido foi implementado por CLEMENTS [26], cortando chapas tal como serão empregadas no casco, poupando material e tempo, o evitando retrabalhos devido às imperfeições de corte no início da produção.

No caso do desenho do plano de expansão de chapameento, o processo pode ser melhorado pela adoção de tratamento matemático das balizas do navio através de splines ou outras de melhor performance.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1]-GAO, JIE-GU. *Hull construction precision management*. CSSC Published China, Jul. 1988.
- [2]-FERRIS, L.W. *A standart series of developable surfaces*. *Marine Technology*, V. 5 n. 1, p. 52-62, Jan. 1968.
- [3]-CALKINS, D. E. et. al. *Small craft hull form surface definition in a high-level computer graphic design environment*, *Transaction of the Society of Naval Architects and Marine Engineers*, V. 97, 1989, p. 85-114.
- [4]-EYRES, D.J. *Ship construction* 3. ed. Oxford, Newres, 1988, 339 p.
- [5]-ZEID, I. *CAD/CAM: theory and practice*. 2. ed. New York, Mc-Graw Hill, 1991, 1052 p.
- [6]-BARABANOV N. *Structural design of sea-going ships*. 2. ed. Moscou, Peace Publishers, 1973
- [7]-DORMIDONTOV V.K. *Shipbuilding technology*. 2. ed. Moscou, Mir Publishers, 1973
- [8]-KILGORE, U. *Developable hull surfaces*. In: Jan-Olof Traung. *Fishing boat of the world*; London, Fishing News, 1967, V. 3, 346 p.
- [9]-KVO, CHENGUI, *Computer methods for ship surface design*. London; Longman, 1971, 224 p.
- [10]-DE BOOR, C. *A practical guide to splines*. New York, Springer, 1978, 392 p.
- [11]-LEITE, J.B. *Funções spline e aplicações à interpolações e integrações numéricas*. São Carlos, 1977, 190 p. Dissertação (Mestrado)- Instituto de Ciências Matemática de São Carlos, Universidade de São Carlos.
- [12]-ROGERS, D. F.; ADAMS, J. A. *Mathematical elements for computer graphics*. New York: McGraw-Hill, 1976, 239 p.
- [13]-NOLAN, T. J. *Computer aided design of developable hull surfaces*. *Marine Technology*; V. 8 n. 2, p.233-242. April 1971.
- [14]-THEILHEIMER, F. *The fairing of the ship lines on a high-speed electronic computer*. DTMB Report 1474, 1961.
- [15]-RICHARD L. BURDEN J. DOUGLAS FAIRES *Analisis Numérico*. Mexico, Grupo editorial Iberoamerica, 1985, 725 p.
- [16]-HUMES, A. F. C. et. al. *Nogões básicas de calculo numerico*. São Paulo, IMELUSP, 1981.
- [17]-FARIN, G. *Curves and surfaces for computer aided geometric design, a practical guide*; 3. ed. Boston, Academic Press, 1993, 473 p.
- [18]-BAKER E. *Introduction to steel shipbuilding*. 2. ed. New York, Mc-Graw-Hill, 1953, 398 p.
- [19]-COLVIN, T.E. *Steel boat building: from plans to bare hull*. Camden International Marine Publishing, 1985, V. 1.
- [20]-STORCH R.L.; HAMMON C.P.; BUNCH H.M. *Ship production*. Centreville, Cornell Maritime Press, 1988, 455 p.

- [21]-THOMAS R.M. *AutoCAD avanzado: verso 12*. Rio de Janeiro, Campus, 1994
- [22]-ZIMBARG E. *AutoCad avanzado*. 2. ed. São Paulo, Erica 1990
- [23]-YAMAGUCHI FUJIO, *Curves and Surfaces in computer aided geometric design*. Berlin, Springer, 1988. 378 p.
- [24]-VANDERGRAFT J.S. *Introduction to numerical computations*. 2 ed. New York, Academic, 1983. 72 p.
- [25]-CHAPRA C. S. *Numerical methods for engineers with personal computer applications*. New York, McGraw-Hill, 1985. 570 p.
- [26]-CLEMETS J. C. *Developed Plate Expansion Using Geodesics. Marine Technology*; V. 21 n. 4, p.384-388. Oct. 1984.
- [27]-HEAD, G.O. *Autosp in plain english: a practical guide for non - programmers*. 5 ed. Sta. North Carolina, s.d. 312 p.
- [28]-O'BRIEN, S. *Turbo pascal 6: manual de referencia*. Madrid, Mc-Graw Hill/ Interamericana de España. 1991.
- [29]-KONG, M. *Lenguaje de programación pascal*. 6 ed. Lima, Pontificia Universidad Católica del Perú. 1989. 180 p.

APÊNDICES

APÊNDICE I

Demonstração dos Teoremas Necessários para a Existência de Superfícies Desenvolvidas

Teorema da Unicidade	ii
Teorema do Paralelismo das Tangentes	iii

APÊNDICE II

Listagem dos Programas Desenvolvidos

Listagem do Programa "EXCHAP"	v
Listagem das Rotinas em AUTOLISP	lvii
Rotina "EXPANSAO.LSP"	lvii
Rotina "RULINGXY.LSP"	lviii
Rotina "RULINGXZ.LSP"	lix

APÊNDICE III

Arquivos Empregados no Programa

Arquivos com os Dados de Entrada	lxi
--	-----

Apêndice I Demonstração dos teoremas necessários para a existência de superfícies desenvolvidas

Teorema da Unidade

O teorema que permite provar a existência da superfície desenvolvida é da unidade, onde seu enunciado é:

“ Se duas curvas no espaço encostam-se em qualquer superfície desenvolvida, essas pares de curvas encostam-se em uma e só uma superfície desenvolvida.”

Para a prova deste teorema, sejam C e D duas curvas com longitudes de arco de δ e σ respectivamente, tais que poderiam encostar-se em (ao menos uma) superfície desenvolvida, como indica-se na figura (a). Não se considerará o caso trivial, onde para certas linhas e pontos e combinação de C e D não sejam coplanares. Propõe-se como hipóteses que o teorema é falso, por exemplo existe mais de uma superfície desenvolvida contendo as curvas C e D .

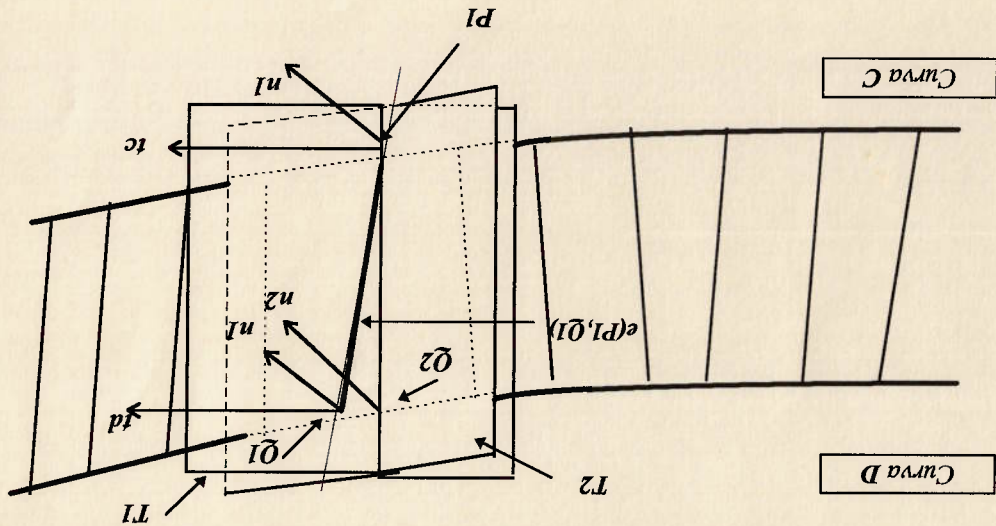


Figura (a) Teorema da unidade

Num ponto $P1$ sob C , constrói-se um plano tangente $T1$, tangente a C e seja $T1$ tangencial a D num ponto $Q1$, então isto faz possível a hipóteses que C e D

encostem-se no menos uma superfície desenvolvida. Uma linha de P_1 a Q_1 se encostara em T_1 .

Obtem-se uma direção unitária desta linha dada por $\hat{e}(P_1, Q_1)$. Sejam tc e td os vetores unitários tangentes para C e D . Agora se o vetor normal a T_1 é n_1 então cumprem-se as relações do produto vetorial que se apresentam em (i).

$$tc(P_1) \times \hat{e}(P_1, Q_1) = n_1 = td(Q_1) \times \hat{e}(P_1, Q_1) \quad (i)$$

Avançando-se ao longo de D a distância $\Delta\sigma$ desde Q_1 , supondo outro plano T_2 que é tangente a D em Q_2 , e supondo-se T_2 tangencial a C no ponto P_1 . Isto é possível baseado na hipóteses que C e D encostam-se em mais de uma superfície desenvolvida.

A normal a T_2 é n_2 . dando-se essa relação pela expressão em (ii) análoga à anterior (i).

$$tc(P_1) \times \hat{e}(P_1, Q_2) = n_2 = td(Q_2) \times \hat{e}(P_1, Q_2) \quad (ii)$$

Mas, desde que ambos planos T_1 e T_2 sejam tangentes a C em P_1 , se tem por hipótese a expressão (iii) um produto vetorial entre as normais obtendo-se o vetor tangente no ponto P_1 .

$$n_1 \times n_2 = tc(P_1) \quad (iii)$$

As afirmações estabelecidas acima são equivalentes à relação que se apresenta em (iv), para o ponto tangente em P_1 , o que é absurdo, e portanto, fica demonstrado:

$$tc(P_1) \times \hat{e}(P_1, Q_1) \times td(Q_1) \times \hat{e}(P_1, Q_2) = tc(P_1) = tc(P_1) \times \hat{e}(P_1, Q_2) \times td(Q_1) \times \hat{e}(P_1, Q_1) \quad (iv)$$

Teorema de paralelismo de tangentes para curvas planas paralelas
 O teorema que complementa as propriedades de uma superfície desenvolvida é de paralelismo de tangentes, que foi aplicado no método gráfico de Kilgore U. seu enunciado é:

“Se duas ou mais curvas planas contidas numa superfície desenvolvida S , são determinadas por a intersecção de S com dois o mais planos paralelos, nenhum dos quais contem qualquer linha traçada de S , então as tangentes

para todas essas curvas planas em suas respectivas intersecções com qualquer linha traçada de S são paralelas”

Sejam os planos paralelos R_1 e $R_2 \dots R_n$ intersecção da superfície desenvolvida S , de uma maneira que permita o corte das curvas planas $D_1 \dots D_n$. Se S é contínua, poderá se ter um plano tangencial T_i ao longo de qualquer traçado L_i , e em S para todas as curvas D_j tem um ponto em L_i que possui uma tangente encostada em T_i . O plano das curvas $D_1 \dots D_n$ cortadas em S por $R_1 \dots R_n$ tem tais tangentes t_1, \dots, t_n , em suas intersecções com L_i . Agora ambos t_i e t_2 se encostam em T_i , mas eles também se encostam respectivamente em R_1 e R_2 e desta maneira são idênticos com as linhas de intersecção de um plano T_i com dois planos paralelos. Consequentemente eles aparecem paralelos em todas as vistas.

Apêndice II

1. LISTAGEM DO PROGRAMA "EXCHAP"

Program EXPCHAP;

{* Autor : Ramon Hernan Velasco Stoll *}
{* Fecha : 14.04.95 *}

{* Controle de Teclado, Tela e Modo Gráfico *}

Uses Crt, Graph, Dos, Vars, Telas, Desenha, Surface, Rulings;

```
{*  
*****  
* CORPO DO PROCEDIMIENTO DESENHA EXPANSÃO *  
*****  
}
```

Procedure DESENHA_EXPANSAO_CHAPEAMENTO;

```
Begin  
  PantallaMenu_1;  
  Lectura1;  
  Arreglo_perfil;  
  PantallaControlArquivosSalda;  
  Expansao_Tradicional;  
  PantallaResultados1;  
End;
```

```
{*  
*****  
* CORPO DO PROCEDIMIENTO SUPERFICIE DESENVOLVIDA *  
*****  
}
```

Procedure SUPERFICIE_DESENVOLVIDA;

```
Begin  
  Dados_Iniciais;  
  N:=N1;  
  contador:=1;
```

```
For contador:=1 to 2 do  
  begin  
    Matriz_Spline;  
    Solucion;  
    N:=N2;
```



```

end;
Kuling_Search;
PantallaResultados2;
End;
{*****}
{* CORPO DO PROGRAMA *}
{*****}
BEGIN
ChrScr;
PantallaInicial;
Repeat
Pantalla_Escolla_Metodo;
Repeat
metodo:=readkey;
Until (metodo = 'e') or (metodo = 'E') or
(metodo = 's') or (metodo = 'S') or
(metodo=#27);
Case metodo of
'E','e' : begin
gotoxy(23,17);
writeln('T');
TextColor(Yellow+Blink);
gotoxy(2,19);
delim;
write ('Aguarde um momento...');
Desenha_expansao_chapamento;
end;
'S','s' : begin
gotoxy(23,17);
writeln('S');
TextColor(Yellow+Blink);
gotoxy(2,19);
delim;
write ('Aguarde um momento...');
Superficie_desenvolvida;
end;
until metodo = #27;
PantallaFinal;

```


END.

```

}
*****
{ UNIDADE COM AS VARIÁVEIS DO PROGRAMA EXPCHAP }
{ UNIT VARS }
*****
}

```

Unit Vars:

Interface

Uses Crt,graph,Dos;

Const

```

Filename1='Radbaliz.txt';
Filename2='Radlmha.txt';
Filename3='Curva1.txt';
Filename4='Curva2.txt';
Filename5='Param_t1.txt';
Filename6='Param_t2.txt';
Filename10='MtxSp_X.txt';
Filename11='MtxSp_Y.txt';
Filename12='MtxSp_Z.txt';
Filename13='CofSpIn1.txt';
Filename14='CofSpIn2.txt';
Fnaux='Pertifl.txt';
Max=30;
Max1=31;
campoixo=10;
deftxo=4;

```

Type

```

Matriz=Array[1..Max,1..Max] of Real;
Vector=Array[1..Max] of Real;
Valor=Real;

```

Var

```

fh,fm,fbal,ft,fp,fi,fauz,fc1,fc2 : text;
f1,f2,f3,f4,f5,f6,f7 : text;
fhh,fhm,fbal,fbt,fbp,fbt : string[20];
fh1,fh2,fh3,fh4,fh5,fh6,fh7 : string[20];
snh,snm,snp,smi,snbal,snbal2,scampo,sdec : string;
l,j,k,contador,campo,dec : integer;

```

```

nbal,nbal2,vector_i,position
X1,X2,A,B,C
Y11,Y12,Z11,Z12,R
Rbal,Rlwl,O
Distance,Siguiente
g,metodo
Px,Py,Pz
V
Mx
Det,f,p,q
Fspln_X,Fspln_Y,Fsplne,
Fspln_Z,Csplne1,Csplne2
Param_t1,Param_t2
Derivando
: Boolean;
N,N1,N2
: Integer;
curva1,curva2
Px,P1y,P1z,Cvector,
P2x,P2y,P2z
: Vector;
T1x,T1y,T1z
T2x,T2y,T2z
: Vector;
Amx
: Matriz;
}
Implementation
End.
}
*****
{
TELAS PARA MANEJO EXTERNO DO PROGRAMA }
{ UNIT TELAS }
*****
}
Unit Telas;
Interface
Uses Crt,Vars,Graph,Dos;
Procedure LimpiarPantallaError;
Procedure PantallaError;
Procedure PantallaInicial;
Procedure Pantalla_Escolla_Metodo;
Procedure PantallaMenu_1;
Procedure PantallaMenu_2;

```



```
Procedure PantallaControlArquivosSaida;
Procedure PantallaResultados1;
Procedure PantallaResultados2;
Procedure PantallaFinal;
Procedure AwaitingUser;

Implementation

{
*****
}

Procedure AwaitingUser;
var Ch : char;
begin
  TextColor(Yellow+Blink);
  write (' Pressione ENTER para continuar. ');
  readln;
  TextColor(Red+Blink);
  write ('
AGUARDE UM MOMENTO !!!);
end;
}
*****
}

Procedure PantallaInicial;
{ * Tela de Apresentação - Modo Gráfico *
var
  Gd, Gm : Integer;
  Font : word;
  Direction : word;
  CharSize : word;
  s : char;
begin
  Gd := VGA;
  Gm := 2;
  InitGraph(Gd, Gm, c:\tp\bgr');
  MoveTo(0,0);
  SetTextStyle(1,0,2);
  SetColor(Blue);
  SetTextJustify(CenterText,CenterText);
  OutTextXY(340,80,'ESCOLA POLITECNICA DA UNIVERSIDADE DE
SAO PAULO);
  OutTextXY(350,100,'Departamento de Engenharia Naval e Oceânica');
  SetTextStyle(1,0,7);
```



```

SetColor(Blue);
SetBkColor(Cyan);
SetTextJustify(CenterText,CenterText);
OutTextXY(Succ(GetMaxX) div 2, Succ(GetMaxY) div 2,
'EXPCHAP ver 1.0');
SetTextStyle(1,0,1);
SetColor(red);
SetTextJustify(LeftText,RightText);
OutTextXY(60,300,'Ramón Hernán Velasco Stoll');
OutTextXY(60,340,'Programa Aplicativo para AUTOCAD ver 12,0');
OutTextXY(60,360,'Desenho do Plano de Expansão de Chapamento');
SetColor(Yellow+Blink);
SetTextJustify(CenterText,CenterText);
OutTextXY(Succ(GetMaxX) div 2, GetMaxY-26, '< Pressione qualquer
tecla para Continuar >');
g := ReadKey;
ClearDevice;
CloseGraph
end;
}
Procedure Pantalla_Escolla_Metodo;
begin
clrscr;
window(5,4,75,23);
textbackground(cyan);
textcolor(blue);
clrscr;
gotoxy(3,3);
write('EXCHAP VER 1.0 EXPANSAO DE CHAPEAMENTO
);
textcolor(yellow);
gotoxy(2,5);
write('Escolha uma das opcoes para desenhar uma expansao de chapamento');
gotoxy(2,6);
write('do navio ou rodar um teste de superficies desenvolvidas. ');
gotoxy(2,8);
write('(E) Calcula a expansao retificando só as balizas em relaçao a');
gotoxy(2,9);
write('linha base, sendo o mais usado. ');
gotoxy(2,12);
write('(S) Efetua um teste baseado em "SUPERFICIES DESENVOLVIDAS"
sobre');
gotoxy(2,13);

```

```

write(' uma metodologia proposta por U. Kilgore');
gotoxy(2,15);
write('Para sair do programa teclee ESC...');
gotoxy(2,17);
write('Digite a sua opção : ');
gotoxy(23,17);
end;
}
*****
Procedure PantallaMenu_1;
{ * Inicia o programa pidoendo o arquivo texto de entrada
  e define outros parâmetros a serem usados * }
begin
  clrscr;
  window(5,4,75,23);
  textbackground (cyan);
  textcolor (blue);
  clrscr;
  gotoxy(3,3);
  write(' EXCHAP VER 1.0 EXPANSAO DE CHAPAMENTO
);
  textcolor (yellow);
  gotoxy(2,5);
  write('A seguir devem ser fornecidos (4) arquivos que descrevem a forma e');
  gotoxy(2,6);
  write('geometria completa do navio escolhido. Os conteúdos de cada um sao');
  gotoxy(2,7);
  write('tabelas com as coordenadas nos planos "XY" como meias bocas e
  "XZ"');
  gotoxy(2,8);
  write('como as alturas desde a linha base. Assim também os arquivos com os');
  gotoxy(2,9);
  write('pontos do perfil e outro com pontos das coordenadas de inicio na);
  gotoxy(2,10);
  write('proa do navio de cada linha "Típica" (borda,convés,quinas,etc.);');
  writeh;
  gotoxy(2,12);
  write('A seguir digite o nome dos arquivos de seu navio após de cada linha. ');
  gotoxy(2,13);
  write('Para o caso exemplo sao fornecidos (4) arq. com extensao <.dat>');
  gotoxy(2,14);
  write('A fim de obter-se a maior precisao possivel,logue um maior numero');
  gotoxy(2,15);

```


write('de balizas, até um máximo de (250), as linhas típicas são máximo

write('Todos os arquivos devem ser de formato texto (ASCII).');

writeLn;

gotoxy(2,18);

write('Arquivo de dados no plano " XY " <bocas.dat>:');

readLn(fm);

if length(fm)=0 then fm:='bocas.dat';

gotoxy(2,18);

write('Arquivo de dados no plano " XZ " <alturas.dat>:');

readLn(fh);

if length(fh)=0 then fh:='alturas.dat';

gotoxy(2,18);

write('Arquivo de dados das coord. perfil <perfil.dat>:');

readLn(fp);

if length(fp)=0 then fp:='perfil.dat';

gotoxy(2,18);

write('Arquivo de dados das coord. de inicio <inicio.dat>:');

readLn(fi);

if length(fi)=0 then fi:='inicio.dat';

gotoxy(2,18);

write('Número de balizas (250) <117>:');

readLn(sbal);

val(sbal,nsbal,vector_i);

if (vector_i > 0) or (length(sbal)=0) then nsbal:=117;

gotoxy(2,18);

write('Número do campo numérico <10>:');

readLn(scampo);

val(scampo,campo,vector_i);

if (vector_i > 0) or (length(scampo)=0) then campo:=10;

gotoxy(2,18);

write('Número de casa decimais <4>:');

readLn(sdec);

val(sdec,dec,vector_i);

if (vector_i > 0) or (length(sdec)=0) then dec:=4;

gotoxy(2,18);

delline;

AwaitUser;

gotoxy(2,18);


```

end;
delline;
gotoxy(2,18);
}
Procedure PantallaMenu_2;
{* Inicia o programa pidoendo o arquivo texto de entrada
e define outros parâmetros a serem usados *}
begin
  clrscr;
  window(5,4,75,23);
  textbackground (cyan);
  textcolor (blue);
  clrscr;
  gotoxy(3,3);
  write(' EXCHAP VER 1.0 SUPERFICIES DESENVOLVIDAS ');
  textcolor (yellow);
  gotoxy(2,5);
  write('A seguir devem ser fornecidos informações que definen uma superfície');
  gotoxy(2,6);
  write('parcial entre duas quinas ou curvas típicas num navio escolhido. ');
  gotoxy(2,7);
  write('Estas duas curvas típicas ou de control teram as coordenadas "X,Y,Z"');
  gotoxy(2,8);
  write('descrebendo o comportamento das curvas no espaço. ');
  gotoxy(2,9);
  write('As curvas de control podem ser a linha de borda,convês,quinas,etc. ');
  gotoxy(2,11);
  write('Escolha uma das opções para o ingresso dos dados iniciais');
  gotoxy(2,13);
  write(' ( A ) Ingresso de dados através de arquivos ASCII, formato ( *.txt )');
  gotoxy(2,15);
  write(' ( T ) Ingresso de dados através do teclado. ');
  gotoxy(2,17);
  write('Para sair do programa teclee ESC... ');
  gotoxy(2,19);
  write('Digite a sua opção : ');
  gotoxy(23,19);
end;
}
Procedure PantallaControlArquivosSaida;

```

{* Da nome nos arquivos de saída conforme desejo do usuário *

```

begin
  clrscr;
  window(5,4,75,23);
  textbackground(cyan);
  textcolor(blue);
  clrscr;
  gotoxy(3,3);
  write(' EXCHAP VER 1.0 EXPANSAO DE CHAPEAMENTO ');
  textcolor(yellow);
  gotoxy(2,5);
  write('As coordenadas dos pontos de chapejamento sao calculadas');
  gotoxy(2,6);
  write('respecto a cada linha tipica pertencente ao navio e seram guardados');
  gotoxy(2,7);
  write('em arquivos de texto, porém ASCII, no seguinte formato: ');
  gotoxy(2,8);
  write('Tem-se que para uma linha l gera-se o arquivo "LINHA_1.TXT"');
  gotoxy(2,9);
  write('linha2 gera-se "LINHA_2.TXT", linha3 gera-se "LINHA_1.TXT"');
  gotoxy(2,10);
  write('da mesma forma para as restantes até completar as sete (7) linhas');
  write('');
  gotoxy(2,12);
  write('Caso voce deseje nomear os arquivos, digite o nome escolhido');
  gotoxy(2,13);
  write('ou tecla enter para aceitar o default. Exemplo: ');
  textcolor(blue);
  gotoxy(2,15);
  write('Arquivo com a primeira linha tipica <LINHA_1.TXT> : borda.txt ');
  gotoxy(2,16);
  write('ENTER!!! ');
  textcolor(yellow);
  gotoxy(2,18);
  write('Arquivo com a primeira linha tipica <LINHA_1.TXT> : ');
  readln(m1);
  if length(m1)=0 then m1:=LINHA_1.TXT;
  gotoxy(2,18);
  delme;
  write('Arquivo com a segunda linha tipica <LINHA_2.TXT> : ');
  readln(m2);
  if length(m2)=0 then m2:=LINHA_2.TXT;
  gotoxy(2,18);
  delme;

```



```

sound(trunc(exp(ln(440)+(10-1)/12*ln(2)))));
delay(1000);
nosound;
chrscr;
window(7,6,73,21);
write('EXPCHAP FOI EXECUTADO COM SUCESSO !!!');
write;
write('Para ler as linhas de expansao no ACAD rode EXPANSAO.TSP');
write('Informando o nome de cada arquivo gerado no programa. ');
write('Cada linha típica foi gerada independentemente, porém');
write('deve-se rodar uma por vez, até o máximo de sete (7). ');
write('Exemplo: LINHA_1.TXT, LINHA_2.TXT, etc. ');
write;
write('Para obter o desenho do perfil do navio rode EXPANSAO.TSP');
write('Introduzindo o nome do arquivo PERFIL.TXT gerado no programa. ');
write('A mudança no campo numérico e casas decimais dos arquivos');
write('gerados podem induzir erros no momento de desenho no CAD. ');
write;
write('Os arquivos gerados, devem-se copiar no C:\ACAD\ no início. ');
gotoxy(2,18);
write('digite ENTER para reiniciar ou ESC para sair...');
metodo:=readkey;
end;
}
*****
Procedure PantallaResultados2;
begin
window(5,4,75,23);
textbackround(red);
textcolor(yellow);
sound(880);
delay(200);
nosound;
delay(30);
sound(880);
delay(200);
sound(trunc(exp(ln(440)+(10-1)/12*ln(2)))));
delay(1000);
nosound;
chrscr;
window(7,6,73,21);
write('EXPCHAP FOI EXECUTADO COM SUCESSO !!!');
write;
write('O teste de superficies desenvolvidas consiste na busca ');

```

```
writeln('das rullings, através de duas curvas no espaço as quais');
writeln('foram definidas no início. Pode-se verificar a seguir ');
writeln('que as respetivas rullings não poderam-se cruzar entre ');
writeln('sim mesmas, para estabelecer a superfície desenvolvida. ');
writeln;
writeln('Para obter o desenho de cada rulling rode RULLINGS.LSP ');
writeln('introduzindo o nome do arquivo RULLINGS.TXT gerado no ');
writeln('A mudança no campo numérico e casas decimais dos arquivos');
writeln('gerados podem induzir erros no momento de desenho no AUTOCAD. ');
writeln;
writeln('Os arquivos gerados, devem-se copiar no C:\ACAD\ no início. ');
gotoxy(2,18);
write('digite ENTER para reiniciar ou ESC para sair...');
metodo:=readkey;
end;
```

Procedure PantallaFinal;
 * Visualiza Tela Final no Modo Gráfico *

var Gd, Gm : Integer;

```
begin
    Gd := VGA;
    Gm := 2;
    InitGraph(Gd, Gm, c:\tp\bgr);
    MoveTo(0,0);
    SetTextStyle(1,0,5);
    SetColor(Blue);
    SetBkColor(Cyan);
    SetTextJustify(CenterText,CenterText);
    OutTextXY(Succ(GetMaxX) div 2, Succ(GetMaxY) div 2,
    'FIM DO PROGRAMA');
    g:=ReadKey;
    ClearDevice;
    CloseGraph;
end;
```

END.

 { DESENHO DO PLANO DE EXPANSÃO DE CHAPEAMENTO }
 { UNIT DESENHA }


```

{
    Unit DESENHA;
    Interface
    {
        * Controle do Teclado, Tela e Modo Gráfico *
        Uses Crt,Graph,Dos,Vars,Telas;
        Procedure Arranjo_perfil;
        Procedure LecturaI;
        Procedure Expansao_Tradicional;
        Implementation
    }
    Procedure Arranjo_perfil;
    Procedure Arranjo_perfil;
    Procedure LecturaI;
    Procedure Expansao_Tradicional;
}
{
    * Comienza a lectura dos arquivos texto especificados *
    Procedure LecturaI;
    var
        IJ : Integer;
}
}

```



```

begin
  { Preparar el archivo 'perfil.dat' para lectura }
  assign(fp,fp);
  reset(fp);
  { Preparar el archivo 'bocas.dat' para lectura }
  assign(fm,fm);
  reset(fm);
  { Preparar el archivo 'alturas.dat' para lectura }
  assign(fh,fh);
  reset(fh);
  { Cria el archivo 'Radbaliz.txt' para almacenar calculos }
  assign(fb,flename1);
  rewrite(fb);
  { Lê as primeira columna dos arquivos "alturas.dat" e "bocas.dat" }
  while Not Eof(fh) do
    readln(fp,X1,Y11[1],Z11[1]);
    readln(fm,X2,Y12[1],Y12[2],Y12[3],Y12[4],Y12[5],Y12[6],Y12[7]);
    readln(fh,X2,Z12[1],Z12[2],Z12[3],Z12[4],Z12[5],Z12[6],Z12[7]);
    Distance:=False;
    Siguiente:=False;
    Contador:=0;
  begin
    For i:= 1 to 7{nlvl} do
      Begin
        If ((Y12[i]>0) and (Z12[i]>0)) and (R[i]=0) then
          begin
            B:= Y11[1]-Y12[i];
            C:= Z11[1]-Z12[i];
            Distance:=True;
            R[i]:= SQR(SQR(B)+SQR(C));
            write(fb,R[i]:campo:dec,#32);
            contador:=contador+1;
          end
        else
          begin
            If (siguiente=False) then
              begin
                R[i]:=0;
                write(fb,R[i]:campo:dec,#32);
                contador:=contador+1;
              end
            end
          end
        end;
      end
    end
  end

```

```

end;
If (Distance=True) and ((Y12[i+1]>0) and (Z12[i+1]>0)) then

```

```

begin
B:= Y12[i]-Y12[i+1];
C:= Z12[i]-Z12[i+1];
R[i+1]:= SQR(SQR(B)+SQR(C));
If ((Y12[i]>0) and (Z12[i]>0)) then

```

```

begin
write(fb1,R[i+1]:campo:dec,#32);

```

```

contador:=contador+1;
end;

```

```

Siguiente:=True;

```

```

}
*****

```

```

If ((Y12[i]=0) and (Z12[i]=0)) and
((Y12[i-1]>0) and (Z12[i-1]>0)) and
(contador > 7{nlvl}) then

```

```

begin
B:= Y12[i-1]-Y12[i+1];
C:= Z12[i-1]-Z12[i+1];
R[i]:=0;
write(fb1,R[i]:campo:dec,#32);
contador:=contador+1;
R[i+1]:= SQR(SQR(B)+SQR(C));
write(fb1,R[i+1]:campo:dec,#32);
contador:=contador+1;
end;

```

```

If ((Y12[i]=0) and (Z12[i]=0)) and
((Y12[i-1]=0) and (Z12[i-1]=0)) and
(contador > 7{nlvl}) then

```

```

begin
R[i]:=0;
write(fb1,R[i]:campo:dec,#32);
contador:=contador+1;
end;

```

```

If ((Y12[i]=0) and (Z12[i]=0)) and
((Y12[i+1]=0) and (Z12[i+1]=0)) and
(contador > 7{nlvl}) then
begin
R[i]:=0;

```



```

write(fb1,R[i]:campo:#32);
contador:=contador+1;
R[i+1]:=0;
write(fb1,R[i+1]:campo:#32);
contador:=contador+1;
end;
IF ((Y12[i]=0) and (Z12[i]=0)) and
((Y12[i+1]>0) and (Z12[i+1]>0)) and
(contador > 7{nlwl}) then
begin
B:= Y12[i]-1]-Y12[i+1];
C:= Z12[i]-1]-Z12[i+1];
IF ((Y12[i]-1]=0) and (Z12[i]-1]=0)) and
(contador > 7{nlwl}) then
begin
B:= Y12[i]-2]-Y12[i+1];
C:= Z12[i]-2]-Z12[i+1];
end;
R[i]:=0;
write(fb1,R[i]:campo:#32);
contador:=contador+1;
R[i+1]:= SQR(T(SQR(B)+SQR(C)));
write(fb1,R[i+1]:campo:#32);
contador:=contador+1;
end;
IF (((Y12[i]=Y12[i+1]) and ((Z12[i]=Z12[i+1])) and
((Y12[i+1]>0) and (Z12[i+1]>0)) and
((Y12[i]>0) and (Z12[i]>0)))) then R[i+1]:=1;
End;
writem(fb1);
end;
end;
} Termine de trabalho nos arquivos
close(fm);
close(fh);
close(fb1);

```



```

end;
{
*****
}

```

```

Procedure Expansao_Tradicional;

```

```

var
  X11,dbal      : Array [1..7] of Real;
  X12,Y12,Z12  : Real;
  CoXexp,CoYexp : Real;
  ponto,fm     : Boolean;
  i,j          : Integer;

```

```

BEGIN

```

```

  { Lê a primeira linha do arquivo "inicio.dat" como ponto de partida
  e escreve dentro do arquivo linha1, onde seram guardados os calculos
  }
  { Prepara o arquivo 'inicio.dat' para leitura
  }
  assign(f,fm);
  reset(f);

```

```

  for i:= 1 to 7{nlvl} do

```

```

    begin { Comenzo do FOR }

```

```

      readln(f,X11[i],Y11[i],Z11[i]);

```

```

    case i of

```

```

      1 : begin
          assign(f1,fm1);
          rewrite(f1);
          write(f1,X11[1]:campo:dec);
          write(f1,Y11[1]:campo:dec);
          write(f1,Z11[1]:campo:dec);
          writeln(f1);
          close(f1);
        end;

```

```

      2: begin
          assign(f2,fm2);
          rewrite(f2);
          write(f2,X11[2]:campo:dec);
          write(f2,Y11[2]:campo:dec);
          write(f2,Z11[2]:campo:dec);

```

```

write(h2);
close(h2);
end;

3: begin
    assign(h3, f13);
    rewrite(h3);
    write(h3, X11[3]:campo:dec);
    write(h3, Y11[3]:campo:dec);
    write(h3, Z11[3]:campo:dec);
    write(h3);
    close(h3);
end;

4: begin
    assign(h4, f14);
    rewrite(h4);
    write(h4, X11[4]:campo:dec);
    write(h4, Y11[4]:campo:dec);
    write(h4, Z11[4]:campo:dec);
    write(h4);
    close(h4);
end;

5: begin
    assign(h5, f15);
    rewrite(h5);
    write(h5, X11[5]:campo:dec);
    write(h5, Y11[5]:campo:dec);
    write(h5, Z11[5]:campo:dec);
    write(h5);
    close(h5);
end;

6: begin
    assign(h6, f16);
    rewrite(h6);
    write(h6, X11[6]:campo:dec);
    write(h6, Y11[6]:campo:dec);
    write(h6, Z11[6]:campo:dec);
    write(h6);
    close(h6);
end;

7: begin

```



```

assign(h7,fnl7);
rewrite(h7);
write(h7,X11[7]:campo:dec);
write(h7,Y11[7]:campo:dec);
write(h7,Z11[7]:campo:dec);
write(h7);
end;

end; { final del CASE }
end; { final del FOR }
close(h);
{ termina lectura do inicio.dat }
for i:= 1 to 7{nlvl} do
begin { inicio do FOR para controle da leitura de dados }
Rbal:=0;
Rlwl:=0;
fim:=false;
ponto:=false;
{ Preparar o arquivo 'radbaliz.txt' para leitura }
assign(fb,filename1);
reset(fb);
{ Preparar o arquivo 'perfl.dat' para leitura }
assign(fp,mp);
reset(fp);
for j:= 1 to nbal do
begin
If (Ponto=false) then
begin
begin Repeat
begin
readln(fp,X12,Y12,Z12);
readln(fb,dbal[1],dbal[2],dbal[3],dbal[4],dbal[5],dbal[6],dbal[7]);
j:=j+1;
ponto:=true;
end
until (X11[i]=X12)
end
end
end;

```



```

end;
readln(fp, X12, Y12, Z12);
readln(fb, dbal[1], dbal[2], dbal[3], dbal[4], dbal[5], dbal[6], dbal[7]);

Case i of
1: Rbal:= dbal[i];
2: Rbal:= dbal[i-1]+dbal[i];
3: Rbal:= dbal[i-2]+dbal[i-1]+dbal[i];
4: Rbal:= dbal[i-3]+dbal[i-2]+dbal[i-1]+dbal[i];
5: Rbal:= dbal[i-4]+dbal[i-3]+dbal[i-2]+dbal[i-1]+dbal[i];
6: Rbal:= dbal[i-5]+dbal[i-4]+dbal[i-3]+dbal[i-2]+dbal[i-1]+dbal[i];
7: Rbal:= dbal[i-6]+dbal[i-5]+dbal[i-4]+dbal[i-3]+dbal[i-2]+
dbal[i-1]+dbal[i];
end;
CoXexp:= X12;
CoYexp:= Z12+Rbal;
If (dbal[i]=0) then fim:=true;
}
{ Processo de armazenamento para cada arquivo Linha [i].txt }
If (fim<>true) then
begin
write(fp, CoXexp:campo:dec);
write(fp, Y11[i]:campo:dec);
write(fp, CoYexp:campo:dec);
write(fp);
close(fp);
end;
1: begin
append(fb);
write(fb, CoXexp:campo:dec);
write(fb, Y11[i]:campo:dec);
write(fb, CoYexp:campo:dec);
write(fb);
close(fb);
end;
2: begin
append(fb2);
write(fb2, CoXexp:campo:dec);
write(fb2, Y11[i]:campo:dec);
write(fb2, CoYexp:campo:dec);
write(fb2);
close(fb2);
end;

```

```
end;
3: begin
  write(H3,CoXexp:campo:dec);
  write(H3,Y11[I]:campo:dec);
  write(H3,CoYexp:campo:dec);
  write(H3);
  close(H3);
end;
4: begin
  append(H4);
  write(H4,CoXexp:campo:dec);
  write(H4,Y11[I]:campo:dec);
  write(H4,CoYexp:campo:dec);
  write(H4);
  close(H4);
end;
5: begin
  append(H5);
  write(H5,CoXexp:campo:dec);
  write(H5,Y11[I]:campo:dec);
  write(H5,CoYexp:campo:dec);
  write(H5);
  close(H5);
end;
6: begin
  append(H6);
  write(H6,CoXexp:campo:dec);
  write(H6,Y11[I]:campo:dec);
  write(H6,CoYexp:campo:dec);
  write(H6);
  close(H6);
end;
7: begin
  append(H7);
  write(H7,CoXexp:campo:dec);
  write(H7,Y11[I]:campo:dec);
  write(H7,CoYexp:campo:dec);
  write(H7);
  close(H7);
```



```

textcolor (yellow);
gotoxy(2,5);
write('Digite o número dos pontos da curva_1 (N1): ');
readln(N1);
gotoxy(2,6);
write('Ingresse as coordenadas Xi, Yi e Zi ');
assign(curva1,filename3);
rewrite(curva1);
for i:=1 to N1 do
begin
  gotoxy(2,8);
  write('Digite o valor para a Coordenada : P1x (,I) ');
  Read(P1x[i]);
  Write(curva1,P1x[i]);
  gotoxy(2,9);
  write('Digite o valor para o Coordenada : P1y (,I) ');
  Read(P1y[i]);
  Write(curva1,P1y[i]);
  gotoxy(2,10);
  write('Digite o valor para o Coordenada : P1z (,I) ');
  Read(P1z[i]);
  Write(curva1,P1z[i]);
  gotoxy(2,8);
  delme;
  delme;
  delme;
end;
Close(curva1);

chrscr;
window(5,4,75,23);
textbackground (cyan);
textcolor (blue);
chrscr;
gotoxy(3,3);
write(' EXCHAP VER 1.0 SUPERFICIES DESENVOLVIDAS ');
textcolor (yellow);
gotoxy(2,5);
write('Digite o número dos pontos da curva_2 (N2): ');
readln(N2);
gotoxy(2,6);
write('Ingresse as coordenadas Xi, Yi e Zi ');
assign(curva2,filename4);
rewrite(curva2);
for i:=1 to N2 do

```

```

begin
  Gotoxy(2,8);
  Write('Digite o valor para a Coordenada : P2x (,1,)' );
  Read(P2x[i]);
  Write(curva2,P2x[i]);
  Gotoxy(2,9);
  Write('Digite o valor para o Coordenada : P2y (,1,)' );
  Read(P2y[i]);
  Write(curva2,P2y[i]);
  Gotoxy(2,10);
  Write('Digite o valor para o Coordenada : P2z (,1,)' );
  Read(P2z[i]);
  Write(curva2,P2z[i]);
  Gotoxy(2,8);
  delime;
  delime;
  delime;
end;
Close(curva2);
readkey;

end;
}
*****
Procedure Dados_Arquivos;
begin
  clrscr;
  window(5,4,75,23);
  textbackground (cyan);
  textcolor (blue);
  clrscr;
  gotoxy(3,3);
  write(' EXCHAP VER 1.0 SUPERFICIES DESENVOLVIDAS ');
  textcolor (yellow);
  gotoxy(2,5);
  write('A seguir devem ser fornecidos (2) arquivos que descrevem a forma e);
  gotoxy(2,6);
  write('geometria completa das curvas típicas do navio. ');
  gotoxy(2,8);
  write('Para o caso exemplo sao fornecidos (2) arq. com extensao < .dat > e);
  gotoxy(2,9);
  write('no formato texto (ASCII), os quais contem duas quinas de um navio. ');
  write;
end;

```



```

gotoxy(2,11);
write('Número de balizas na curva 1 (Max 30) < I3> ');
readln(snb1);
val(snb1,N1,vector_1);
if (vector_1 > 0) or (length(snb1)=0) then N1:=13;
gotoxy(2,11);
write('Arquivo com dados da curva 1 < Kilgore1.dat > ');
readln(fm);
if length(fm)=0 then fm:='Kilgore1.dat';
assign(fc1,fm);
reset(fc1);
assign(curva1,filename3);
rewrite(curva1);
For i:=1 to N1 do
begin
Read(fc1,P1x[i],P1y[i],P1z[i]);
WriteLn(curva1,P1x[i]:campo fixo:decfixo,
P1y[i]:campo fixo:decfixo,P1z[i]:campo fixo:decfixo);
end;
Close(fc1);
Close(curva1);
gotoxy(2,11);
write('Número de balizas na curva 2 (Max 30) < I3> ');
readln(snb2);
val(snb2,N2,vector_2);
if (vector_2 > 0) or (length(snb2)=0) then N2:=13;
gotoxy(2,11);
write('Arquivo com dados da curva 2 < Kilgore2.dat > ');
readln(fmh);
if length(fmh)=0 then fmh:='Kilgore2.dat';
assign(fc2,fmh);
reset(fc2);
assign(curva2,filename4);
rewrite(curva2);
For i:=1 to N2 do
begin
Read(fc2,P2x[i],P2y[i],P2z[i]);
WriteLn(curva2,P2x[i]:campo fixo:decfixo,
P2y[i]:campo fixo:decfixo,P2z[i]:campo fixo:decfixo);
end;

```



```
end;  
Close(fc2);  
Close(curva2);  
gotoxy(2,15);  
AwaitUser;  
gotoxy(2,15);  
Delline;  
end;  
}*****  
Procedure Dados_Iniciais;  
begin  
  Clscr;  
  Repeat  
    PantallaMenu_2;  
    metodo:=readkey;  
  Until ( metodo = 'a') or ( metodo = 'A') or  
        ( metodo = 't') or ( metodo = 'T') or  
        ( metodo=#27);  
  Case metodo of  
    'A','a' : begin  
      gotoxy(23,19);  
      writeLn('A');  
      Dados_Arquivos;  
    end;  
    'T','t' : begin  
      gotoxy(23,19);  
      writeLn('T');  
      Dados_Teclado;  
    end;  
  end;  
end;  
}*****  
* Procedimento para gerar a matriz interpoladora Spline, aplicando-se  
a condição Natural, derivadas nos extremos igual a zero.*  
}*****  
Procedure Matriz_Spline;
```

```

Var      pos,dummy,renjlon      : integer;
          T                      : Vector;
          Fechadura              : Boolean;

begin
  { Condicion inicial N >= 3 }
  pos:=1;
  renjlon:=0;
  dummy:=0;
  Fechadura:=true;
  assign(Fspln_X,filename10);
  rewrite(Fspln_X);
  assign(Fspln_Y,filename11);
  rewrite(Fspln_Y);
  assign(Fspln_Z,filename12);
  rewrite(Fspln_Z);
  IF N=N1 then
    begin
      assign(curva1,filename3);
      reset(curva1);
      For i:=1 to N do
        begin
          Readln(Curva1,Px[i],Py[i],Pz[i]);
        end;
      Close(curva1);
    end;
  IF N=N2 then
    begin
      assign(curva2,filename4);
      reset(curva2);
      For i:=1 to N do
        begin
          Readln(Curva2,Px[i],Py[i],Pz[i]);
        end;
      Close(curva2);
    end;
  For K:=1 to 3 do

```



```

BEGIN
  pos:=1;
  renjlon:=0;
  dummy:=0;
  i:=3;
  Fechadura:=true;
  { * Condições para escrever a primeira coluna *
  Mx[1,1]:=1;
  Mx[1,2]:=0.5;
  Case K of
    1: begin
      Write(Fspim_X,Mx[1,1]:campo:deftxo,#32);
      Write(Fspim_X,Mx[1,2]:campo:deftxo,#32);
    end;
    2: begin
      Write(Fspim_Y,Mx[1,1]:campo:deftxo,#32);
      Write(Fspim_Y,Mx[1,2]:campo:deftxo,#32);
    end;
    3: begin
      Write(Fspim_Z,Mx[1,1]:campo:deftxo,#32);
      Write(Fspim_Z,Mx[1,2]:campo:deftxo,#32);
    end;
  end;
  For j:=3 to N do
    begin
      Mx[1,j]:=0;
    Case K of
      1: Write(Fspim_X,Mx[1,j]:campo:deftxo,#32);
      2: Write(Fspim_Y,Mx[1,j]:campo:deftxo,#32);
      3: Write(Fspim_Z,Mx[1,j]:campo:deftxo,#32);
    end;
  end;
  T[2]:= sqrt(sqrt(Px[2]-Px[1])+sqrt(Py[2]-Py[1])+sqrt(Pz[2]-Pz[1]));
  IF N=N1 then
    begin
      Assign(Param_t1,filenam5);
      rewrite(Param_t1);
      writeh(Param_t1,T[2]:campo:deftxo);
    end;
  end;

```



```

close(Param_t1)
end;
IF N=N2 then
begin
Assign(Param_t2,filename6);
rewrite(Param_t2);
writeln(Param_t2,T[2]:campo:deftxo);
close(Param_t2)
end;
Case K of
1: begin
Mx[1,n+1]:=3/(2*T[2])*(Px[2]-Px[1]);
Write(Fsplm_X,Mx[1,n+1]:campo:deftxo,#32);
writeln(Fsplm_X);
end;
2: begin
Mx[1,n+1]:=3/(2*T[2])*(Py[2]-Py[1]);
Write(Fsplm_Y,Mx[1,n+1]:campo:deftxo,#32);
writeln(Fsplm_Y);
end;
3: begin
Mx[1,n+1]:=3/(2*T[2])*(Pz[2]-Pz[1]);
Write(Fsplm_Z,Mx[1,n+1]:campo:deftxo,#32);
writeln(Fsplm_Z);
end;
end;
*****
} * Condições para escrever as linhas internas *
}
For j:= 3 to N do
begin
For dummy:= 1 to (posi-1) do
begin
Mx[i,j]:=0;
If fechadura=true then
begin
Case K of
1: Write(Fsplm_X,Mx[i,j]:campo:deftxo,#32);
2: Write(Fsplm_Y,Mx[i,j]:campo:deftxo,#32);
3: Write(Fsplm_Z,Mx[i,j]:campo:deftxo,#32);
end;
end;
end;
end;
For j:= 3 to N do
begin
For dummy:= 1 to (posi-1) do
begin
Mx[i,j]:=0;
If fechadura=true then
begin
Case K of
1: Write(Fsplm_X,Mx[i,j]:campo:deftxo,#32);
2: Write(Fsplm_Y,Mx[i,j]:campo:deftxo,#32);
3: Write(Fsplm_Z,Mx[i,j]:campo:deftxo,#32);
end;
end;
end;
end;
end;
end;

```

```

end;
T[i-1]:=sqrt(sqrt(Px[i]-1]-Px[i-2]))+sqrt(Py[i]-1]-Py[i-2])+
sqrt(Pz[i]-1]-Pz[i-2]));
T[i]:=sqrt(sqrt(Px[i]-1]-Px[i-1])+sqrt(Py[i]-1]-Py[i-1])+
sqrt(Pz[i]-1]-Pz[i-1]));
IF N=N1 then
begin
Assign(Param_t1,filename5);
append(Param_t1);
writeIn(Param_t1,T[i]:campoFixo:decFixo);
close(Param_t1)
end;
IF N=N2 then
begin
Assign(Param_t2,filename6);
Append(Param_t2);
writeIn(Param_t2,T[i]:campoFixo:decFixo);
close(Param_t2)
end;
Mx[i,j-2]:=T[i];
Mx[i,j-1]:=2*(T[i-1]+T[i]);
Mx[i,j]:=T[i-1];
Case K of
1: If (fechadura=true) then
begin
Write(FspIn_X,Mx[i,j-2]:campoFixo:decFixo,#32);
Write(FspIn_X,Mx[i,j-1]:campoFixo:decFixo,#32);
Write(FspIn_X,Mx[i,j]:campoFixo:decFixo,#32);
end;
2: If (fechadura=true) then
begin
Write(FspIn_Y,Mx[i,j-2]:campoFixo:decFixo,#32);
Write(FspIn_Y,Mx[i,j-1]:campoFixo:decFixo,#32);
Write(FspIn_Y,Mx[i,j]:campoFixo:decFixo,#32);
end;
3: If (fechadura=true) then
begin
Write(FspIn_Z,Mx[i,j-2]:campoFixo:decFixo,#32);
Write(FspIn_Z,Mx[i,j-1]:campoFixo:decFixo,#32);
end;
end;

```



```

Write(Fsplm_Z,Mx[i,j]:campo:deftxo,#32);
end;
end;
dummy:=j+1;
While (dummy<=N) do
begin
Mx[i,dummy]:=0;
Case K of
1: Write(Fsplm_X,Mx[i,dummy]:campo:deftxo,#32);
2: Write(Fsplm_Y,Mx[i,dummy]:campo:deftxo,#32);
3: Write(Fsplm_Z,Mx[i,dummy]:campo:deftxo,#32);
end;
dummy:=dummy+1;
end;
end;
dummy:=j+1;
Case K of
1: begin
Mx[n,n+1]:=(3/(T[i-1]*T[i]))*(sqrt(T[i-1])*(Px[j]-Px[j-1]))+
sqrt(T[i])*(Px[j-1]-Px[j-2]));
Write(Fsplm_X,Mx[n,n+1]:campo:deftxo,#32);
Write(Fsplm_X);
renqon:=renqon+1;
If (renqon=N-2) or (K<>1) then Fechadura:=false;
end;
2: begin
Mx[n,n+1]:=(3/(T[i-1]*T[i]))*(sqrt(T[i-1])*(Py[j]-Py[j-1]))+
sqrt(T[i])*(Py[j-1]-Py[j-2]));
Write(Fsplm_Y,Mx[n,n+1]:campo:deftxo,#32);
Write(Fsplm_Y);
renqon:=renqon+1;
If (renqon=N-2) or (K<>2) then Fechadura:=false;
end;
3: begin
Mx[n,n+1]:=(3/(T[i-1]*T[i]))*(sqrt(T[i-1])*(Pz[j]-Pz[j-1]))+
sqrt(T[i])*(Pz[j-1]-Pz[j-2]));
Write(Fsplm_Z,Mx[n,n+1]:campo:deftxo,#32);
Write(Fsplm_Z);
renqon:=renqon+1;
If (renqon=N-2) or (K<>3) then Fechadura:=false;
end;
end;
posi:=posi+1;

```



```

end;
close(Fsplm_X);

end;

end;
end;
WriteIn(Fsplm_Z);
Write(Fsplm_Z, Mx[n,n+1]:campo:deftxo,#32);
Mx[n,n+1]:=6/T[n]*(Pz[n]-Pz[n-1]);
Write(Fsplm_Z, Mx[n,n]:campo:deftxo,#32);
Write(Fsplm_Z, Mx[n,n-1]:campo:deftxo,#32);
3: begin
end;
end;
WriteIn(Fsplm_Y);
Write(Fsplm_Y, Mx[n,n+1]:campo:deftxo,#32);
Mx[n,n+1]:=6/T[n]*(Py[n]-Py[n-1]);
Write(Fsplm_Y, Mx[n,n]:campo:deftxo,#32);
Write(Fsplm_Y, Mx[n,n-1]:campo:deftxo,#32);
2: begin
end;
end;
WriteIn(Fsplm_X);
Write(Fsplm_X, Mx[n,n+1]:campo:deftxo,#32);
Mx[n,n+1]:=6/T[n]*(Px[n]-Px[n-1]);
Write(Fsplm_X, Mx[n,n]:campo:deftxo,#32);
Write(Fsplm_X, Mx[n,n-1]:campo:deftxo,#32);
1: begin
Case K of
T[n]:=sqrt(sqrt(Px[n]-Px[n-1])+sqrt(Py[n]-Py[n-1])+sqrt(Pz[n]-Pz[n-1]));
Mx[n,n-1]:=2;
Mx[n,n]:=4;
end;
end;
3: Write(Fsplm_Z, Mx[n,j]:campo:deftxo,#32);
2: Write(Fsplm_Y, Mx[n,j]:campo:deftxo,#32);
1: Write(Fsplm_X, Mx[n,j]:campo:deftxo,#32);
Case K of
Mx[n,j]:=0;
begin
For j:=1 to N-2 do
*****
} * Condições para escrever a ultima linha da matriz *
}
end;

```

```
close(Fspline_Y);
close(Fspline_Z);
end;
{
  * Procedimento para resolver a matriz determinante que calcula
  os coeficientes de la Spline Cubica Natural. *
}
{
  *****
}
Procedure Solucion;
  Procedure Leer_Matriz(Var A:Matriz);
    Var I,J : Integer;
  begin
    Case K of
      1: begin
          Assign(Fspline,filename10);
          Reset(Fspline);
          end;
      2: begin
          Assign(Fspline,filename1);
          Reset(Fspline);
          end;
      3: begin
          Assign(Fspline,filename12);
          Reset(Fspline);
          end;
    end;
    For I:= 1 to N do
      begin
        For J:=1 to N+1 do Read(Fspline,a[i,j]);
        end;
      end;
    end;
  end;
}
Procedure Triangular(Var A:Matriz);
  Var I,J,K : Integer;
```



```

}
*****
end;

Procedure Pivote(I: Integer);
Var V, T : Real;
    Piv, J : Integer;
begin
    Piv:=I;
    V:=ABS(A[I,I]);
    For J:=I+1 to N do
        T:=ABS(A[J,I]);
        If (V<T) Then
            begin
                V:=T;
                Piv:=J;
            end;
        end;
        If (V=0) Then DET:=0
        Else
            begin
                For J:=I to N+1 do
                    V:=A[I,J];
                    A[I,J]:=A[Piv,J];
                    A[Piv,J]:=V;
                end;
                DET:=A[I,I]*DET;
                If (I<Piv) Then DET:=-DET;
            end;
end;

}
*****
begin
    For I:=1 to N do
        begin
            Pivote(I);
            If (DET<>0) Then
                For J:=I+1 to N do
                    A[J,K]:=A[J,K]-A[J,I]*A[I,K]/A[I,I];
                A[J,I]:=0
        end;
end;

```



```
end;
{
*****
}
begin
  IF N=N1 then
    assign(Cspline1, filename13);
    rewrite(Cspline1);
    close(Cspline1);
  end;
  IF N=N2 then
    begin
      assign(Cspline2, filename14);
      rewrite(Cspline2);
      close(Cspline2);
    end;
    For K:=1 to 3 do
      begin
        Leer_Matriz(Amx);
        DET:=1;
        Triangular(Amx);
        IF (DET=0)
          Then writeLn('Determinante do sistema , zero ')
            else Resolver(Amx);
        end;
      end;
    end;
  end;
{
*****
}
end;
{
*****
}
{ * PROCEDIMENTO PARA GERAR AS RULINGS NECESSARIAS
  PARA O TESTE DE SUPERFICIES DESENVOLVIDAS *
  { * UNIT RULINGS *
  }
  *****
}
Unit Rulings;
Interface
Uses Cr, Vars, Surface;
Procedure Ruling_Search;
```



```

Implementation
{
*****

```

```

Procedure Kuling_Search;

```

```

const h=0.001;

```

```

var

```

```

b1,b2,b3,b4 : real;

```

```

Xt,Yt,Zt : vector;

```

```

Xdt,Ydt,Zdt : vector;

```

```

b21,b22,b23,b24 : real;

```

```

X2t,Y2t,Z2t : vector;

```

```

Xdt2,Ydt2,Zdt2 : vector;

```

```

r1 : text;

```

```

ra,rb,rc : real;

```

```

i1,i2,k1,i2,j2,k2 : integer;

```

```

T,T2,TfTp,Ta,Tb : real;

```

```

Rxt1,Ryt1,Rzt1,Pnx,Pny,Pnz,

```

```

Rxt2,Ryt2,Rzt2,Pdx,Pdy,Pdz,

```

```

Vn1,Vn2,Vn1n2,Vd1d2,Seno,

```

```

Senod : real;

```

```

Pex1,Pex2,Pey1,Pey2,Pez1,

```

```

Pez2 : real;

```

```

Kuling_Found,Kaiz,fechadura,

```

```

Stop : Boolean;

```

```

BEGIN

```

```

clrscr;

```

```

window(5,4,75,23);

```

```

textbackground(cyan);

```

```

textcolor(blue);

```

```

clrscr;

```

```

gotoxy(3,3);

```

```

write(' EXCHAP VER 1.0 SUPERFICIES DESENVOLVIDAS ');

```

```

textcolor(yellow);

```

```

gotoxy(1,5);

```

```

Assign(Param_t1,Filename5);

```

```

reset(Param_t1);

```

```

{ Lectura de parâmetros na curva 1 }

```

```

For i:=2 to N1 do readln(Param_t1,Px[i]);

```

```

Close(Param_t1);

```

```

Assign(Param_t2,Filename6);

```



```

reset(Param_t2);
{ Lectura de parametros na curva 2
  For i:=2 to readm(Param_t2,Py[i]);
  Close(Param_t2);
  Assign(Csplme1,Filename13);
  reset(Csplme1);
  For i:=1 to N1 do read(Csplme1,T1x[i]);
  readm(Csplme1);
  For i:=1 to N1 do read(Csplme1,T1y[i]);
  readm(Csplme1);
  For i:=1 to N1 do read(Csplme1,T1z[i]);
  close(Csplme1);
  Assign(Csplme2,Filename14);
  reset(Csplme2);
  For i:=1 to N2 do read(Csplme2,T2x[i]);
  readm(Csplme2);
  For i:=1 to N2 do read(Csplme2,T2y[i]);
  readm(Csplme2);
  For i:=1 to N2 do read(Csplme2,T2z[i]);
  close(Csplme2);
  } CALCULO DO PONTO FIXO SOBRE A CURVA 1 }
Assign(r1,'Ruings.txt');
rewrite(r1);
}
writeIn('Ingrese o valor do parametro T da Iera Curva);
writeIn('para 0.333 (1/3), 0.5 (1/2) ou 0.667 (2/3));
write('Escolha um só valor para T:= ');
read(Tp);
fechadura:=False;
For i:=1 to N1-1 do
  Raiz:=False;
  Fechadura:=False;
  Stop:=False;
  Px[1]:=0;
  T:=Px[i+1]*(TP);
  If i=N1 then T:=0;
  For k:=1 to 3 do
    Begin
      Ruing_Found:=False;
      Raiz:=False;
      Fechadura:=False;
      Stop:=False;
      Px[1]:=0;
      T:=Px[i+1]*(TP);
      If i=N1 then T:=0;
    End
  End

```

```

begin
case k of
1: If T > 0 then
begin
b1:=P1x[i];
b2:=T1x[i];
b3:=(3*(P1x[i+1]-P1x[i]))/(sqrt(px[i+1])/px[i+1])-
(T1x[i+1]/px[i+1]));
b4:=(2*(P1x[i]-P1x[i+1]))/(sqrt(px[i+1])*px[i+1]) +
((T1x[i+1]/px[i+1]));
Xt[i]:=b1+b2*t+b3*sqrt(t)+b4*sqrt(t)*t;
Xdt[i]:=b2 + b3*t*2 + b4*sqrt(t)*3;
end;
2: If T > 0 then
begin
b1:=P1y[i];
b2:=T1y[i];
b3:=(3*(P1y[i+1]-P1y[i]))/(sqrt(px[i+1])/px[i+1])-
(T1y[i+1]/px[i+1]));
b4:=(2*(P1y[i]-P1y[i+1]))/(sqrt(px[i+1])*px[i+1]) +
((T1y[i+1]/px[i+1])) + ((T1y[i+1]))/(sqrt(px[i+1])*t;
Yt[i]:=b1+b2*t+b3*sqrt(t)+b4*sqrt(t)*t;
Ydt[i]:=b2 + b3*t*2 + b4*sqrt(t)*3;
end;
3: If T > 0 then
begin
b1:=P1z[i];
b2:=T1z[i];
b3:=(3*(P1z[i+1]-P1z[i]))/(sqrt(px[i+1])/px[i+1])-
(T1z[i+1]/px[i+1]));
b4:=(2*(P1z[i]-P1z[i+1]))/(sqrt(px[i+1])*px[i+1]) +
((T1z[i+1]/px[i+1])) + ((T1z[i+1]))/(sqrt(px[i+1])*t;
Zt[i]:=b1+b2*t+b3*sqrt(t)+b4*sqrt(t)*t;
Zdt[i]:=b2 + b3*t*2 + b4*sqrt(t)*3;
end;
end; { fim del CASE Curva-1 }
end; { fm del For K= 1 a 3 }
}
}
*****
Calculo do valor de (t) para a condicon Curva 2.
*****

```



```

For i1:=1 to N2-1 do
begin
    Escolha de um valor apropriado de T2 assim também prova da existência
    {
    *****
    da raiz no intervalo.
    *****
    }
    If (Fechadura=False) then
begin
    Chrscr;
    window(5,4,75,23);
    textbackground (cyan);
    textcolor (blue);
    gotoxy(3,3);
    write( ' EXCHAP VER 1.0 SUPERFICIES DESENVOLVIDAS ');
    textcolor (yellow);
    gotoxy(2,5);
    WriteLn('Busca da Raizng ',i1);
    gotoxy(2,6);
    WriteLn('Analisando raices para el segmento ',i1);
    For j2:=1 to 2 do
begin
    { For de prova da existencia }
    If j2=1 then T2:=0 else T2:=Py[i1+1];
    For k1:=1 to 3 do
begin
    case k1 of
1: If (I1 > N2) then
begin
        b21:=P2x[i1];
        b22:=T2x[i1];
        b23:=(3*(P2x[i1+1]-P2x[i1]))/(sqrt(py[i1+1])/py[i1+1])-
        (T2x[i1+1]/py[i1+1]));
        b24:=(2*(P2x[i1]-P2x[i1+1]))/(sqrt(py[i1+1])*py[i1+1]) +
        ((T2x[i1])/sqrt(py[i1+1])) + ((T2x[i1+1])/sqrt(py[i1+1]))));
        X2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
        Xdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
    end;
2: If (I1 > N2) then
begin
        b21:=P2x[i1];
        b22:=T2x[i1];
        b23:=(3*(P2x[i1+1]-P2x[i1]))/(sqrt(py[i1+1])/py[i1+1])-
        (T2x[i1+1]/py[i1+1]));
        b24:=(2*(P2x[i1]-P2x[i1+1]))/(sqrt(py[i1+1])*py[i1+1]) +
        ((T2x[i1])/sqrt(py[i1+1])) + ((T2x[i1+1])/sqrt(py[i1+1]))));
        X2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
        Xdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
    end;
end;
end;
end;

```



```

b21:=P2y[i1];
b22:=T2y[i1];
b23:=(3*(P2y[i1+1]-P2y[i1]))/(sqrt(py[i1+1])/py[i1+1])-
(T2y[i1+1]/py[i1+1]));
b24:=(2*(P2y[i1]-P2y[i1+1]))*(py[i1+1]) +
((T2y[i1])/(sqrt(py[i1+1])))) + ((T2z[i1+1])/(sqrt(py[i1+1]))));
Y2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
Ydt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;

3: IF (I1 > N2) THEN
begm
b21:=P2z[i1];
b22:=T2z[i1];
b23:=(3*(P2z[i1+1]-P2z[i1]))/(sqrt(py[i1+1])/py[i1+1])-
(T2z[i1+1]/py[i1+1]));
b24:=(2*(P2z[i1]-P2z[i1+1]))*(py[i1+1]) +
((T2z[i1])/(sqrt(py[i1+1])))) + ((T2z[i1+1])/(sqrt(py[i1+1]))));
Z2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
Zdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;

end; { fm del CASE Curva-2 }
end; { fm del For K1= 1 a 3 }

} Calculo do vetor R(t)=(Xt[i]-X2t[i1])+(Yt[i]-Y2t[i1])+(Zt[i]-Z2t[i1])}
IF (i1 > N2) THEN
begm
Ra:=(Xt[i]-X2t[i1]);
Rb:=(Yt[i]-Y2t[i1]);
Rc:=(Zt[i]-Z2t[i1]);
} Calculo do vetor R(t) X T1[i]}
Rxt1:=(Rb*Zdt[i] - Rc*Ydt[i]);
Ryt1:=(Rc*Xdt[i] - Ra*Zdt[i]);
Rzt1:=(Ra*Ydt[i] - Rb*Xdt[i]);
} Calculo do Módulo do vetor R(t) X T1[i]}
Vn1:=sqrt( sqr(Rxt1) + sqr(Ryt1) + sqr(Rzt1) );
} Calculo do vetor R(t) X T2[i]}

```

```

Rxt2:=(Rb*Zdt2[i1] - Rc*Ydt2[i1]);
Ryt2:=(Rc*Xdt2[i1] - Ra*Zdt2[i1]);
Rzt2:=(Ra*Ydt2[i1] - Rb*Xdt2[i1]);
} Calculo do Módulo do vetor R(t) X T2[i1]
Vn2:=sqrt( sqrt(Rxt2) + sqrt(Ryt2) + sqrt(Rzt2) );
} Calculo do produto NI X N2
Pnx:=(((Ryt1/Vn1)*(Rzt2/Vn2))-((Rzt1/Vn1)*(Ryt2/Vn2)));
Pny:=(((Rzt1/Vn1)*(Rxt2/Vn2))-((Rxt1/Vn1)*(Rzt2/Vn2)));
Pnz:=(((Rxt1/Vn1)*(Ryt2/Vn2))-((Ryt1/Vn1)*(Rxt2/Vn2)));
} gotoxy(2,7);
Writeln('Valor de t2 em ',J2,' ponto e ',T2:10:4);
} gotoxy(2,8);
Writeln('Valor de la F(t2) no ponto ',J2,' e ', e );
} gotoxy(2,9);
Writeln('Valor de Pnx:=',Pnx:10:4);
} gotoxy(2,10);
Writeln('Valor de Pny:=',Pny:10:4);
} gotoxy(2,11);
Writeln('Valor de Pnz:=',Pnz:10:4);
}
If J2=1 then
  begin
    Pex1:=Pnx;
    Pey1:=Pny;
    Pez1:=Pnz;
  end;
If J2=2 then
  begin
    Pex2:=Pnx;
    Pey2:=Pny;
    Pez2:=Pnz;
  end;
end;
{ End de la condicion N2<I1 }
end;
{ End del For de prova da existencia }
gotoxy(2,18);
Writeln('Valor de Pnx(t2):=',(Pex1*Pex2):10:4);
gotoxy(2,19);
Writeln('Valor de Pny(t2):=',(Pey1*Pey2):10:4);
gotoxy(2,20);

```



```

WriteM('Valor de Pnz(0)*Pnz(2):=',(Pez1*Pez2):10:4);
textcolor(red+Blink);
If ((Pex1*Pex2<=0) or (Pey1*Pey2<=0)) or ((Pez1*Pez2<=0))
then begin
gotoxy(25,21);
writeM;
Write('EXISTE UNA RAZ NO INTERVALO!!!');
tehadura:=true;
raiz:=true;
end
else raiz:=false;
textcolor(yellow);
Readkey;
end;
}
If (RAIZ=True) then
begin
Ta:=0;
Tb:=Py[i+1];
If (Pex1*Pex2<=0) and (Stop=false) then
begin
tf:=1/2*(ta+tb);
Stop:=true;
end;
If (Pey1*Pey2<=0) and (Stop=false) then
begin
tf:=1/2*(ta+tb);
Stop:=true;
end;
If (Pez1*Pez2<=0) and (Stop=false) then
begin
tf:=1/2*(ta+tb);
Stop:=true;
end;
T2:=TF;
For k1:=1 to 3 do
begin

```



```

case k1 of
1: begin
b21:=P2x[i1];
b22:=T2x[i1];
b23:=(3*(P2x[i1+1]-P2x[i1]))/(sqrt(py[i1+1])/py[i1+1])-
(T2x[i1+1]/py[i1+1]));
b24:=(2*(P2x[i1]-P2x[i1+1]))/(sqrt(py[i1+1])*py[i1+1]) +
((T2x[i1])/(sqrt(py[i1+1])) + ((T2x[i1+1])/(sqrt(py[i1+1])))));
X2t[i1]:=b21+b22*t2+b23*sqrt(t2)*t2;
Xdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;

2: begin
b21:=P2y[i1];
b22:=T2y[i1];
b23:=(3*(P2y[i1+1]-P2y[i1]))/(sqrt(py[i1+1])/py[i1+1])-
(T2y[i1+1]/py[i1+1]));
b24:=(2*(P2y[i1]-P2y[i1+1]))/(sqrt(py[i1+1])*py[i1+1]) +
((T2y[i1])/(sqrt(py[i1+1])) + ((T2y[i1+1])/(sqrt(py[i1+1])))));
Y2t[i1]:=b21+b22*t2+b23*sqrt(t2)*t2;
Ydt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;

3: begin
b21:=P2z[i1];
b22:=T2z[i1];
b23:=(3*(P2z[i1+1]-P2z[i1]))/(sqrt(py[i1+1])/py[i1+1])-
(T2z[i1+1]/py[i1+1]));
b24:=(2*(P2z[i1]-P2z[i1+1]))/(sqrt(py[i1+1])*py[i1+1]) +
((T2z[i1])/(sqrt(py[i1+1])) + ((T2z[i1+1])/(sqrt(py[i1+1])))));
Z2t[i1]:=b21+b22*t2+b23*sqrt(t2)*t2;
Zdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;

end;
{ fim del CASE Curva-2 }
end;
{ fim del For K1= 1 a 3 }
} Calculo do vetor R(t)=(Xt[i]-X2t[i1])-(Yt[i]-Y2t[i1])-(Zt[i]-Z2t[i1]) }
} Calculo do vetor R(t) X T1[i] }

```

```

Rxt1:=(Rb*Zdt[i] - Rc*Ydt[i]);
Ryt1:=(Rc*Xdt[i] - Ra*Zdt[i]);
Rzt1:=(Ra*Ydt[i] - Rb*Xdt[i]);
} Calculo do Módulo do vector R(t) X T1[i] }

Vn1:=sqrt( sqrt(Rxt1) + sqrt(Ryt1) + sqrt(Rzt1) );

} Calculo do vector R(t) X T2[i] }

Rxt2:=(Rb*Zdt2[i] - Rc*Ydt2[i]);
Ryt2:=(Rc*Xdt2[i] - Ra*Zdt2[i]);
Rzt2:=(Ra*Ydt2[i] - Rb*Xdt2[i]);
} Calculo do Módulo do vector R(t) X T2[i] }

Vn2:=sqrt( sqrt(Rxt2) + sqrt(Ryt2) + sqrt(Rzt2) );

} Calculo do producto N1 X N2 }

Pnx:=(((Ryt1/Vn1)*(Rzt2/Vn2))-((Rxt1/Vn1)*(Ryt2/Vn2)));
Pny:=(((Rxt1/Vn1)*(Rzt2/Vn2))-((Ryt1/Vn1)*(Rxt2/Vn2)));
Pnz:=(((Rxt1/Vn1)*(Ryt2/Vn2))-((Ryt1/Vn1)*(Rxt2/Vn2)));
Vn1n2:=sqrt( sqrt(Pnx) + sqrt(Pny) + sqrt(Pnz) );
Vn1:=sqrt( sqrt(Rxt1/Vn1) + sqrt(Ryt1/Vn1) + sqrt(Rzt1/Vn1) );
Vn2:=sqrt( sqrt(Rxt2/Vn2) + sqrt(Ryt2/Vn2) + sqrt(Rzt2/Vn2) );
} Funcion del Sen(t) }
Sen:= ( Vn1n2/(Vn1*Vn2) );

end;
Stop:=false;
}
If (RAIZ=True) then
begin
T2:=0; { A PRIMERA VEZ EM CADA NOVO INTERVALO }
{ T2:= Py[i+1]/3; }
If ((T2>=0) or (Py[i+1]>=T2)) then Rulng_Found:=False;
If (Stop=True) then Rulng_Found:=True;
While (Rulng_Found=False) do
begin
If ((T2>=0) and (T2<=Py[i+1])) then
end;
}

```



```

begin
* DETERMINAR EL VALOR DEL PARAMETRO A EMPLLEAR *
{ Cálculo de las coordenadas para T2 }
For k1:=1 to 3 do
begin
case k1 of
1: If (I1 > N2) then
begin
b21:=P2x[i1];
b22:=T2x[i1];
b23:=(3*(P2x[i1+1]-P2x[i1]))/(sqrt(py[i1+1])/py[i1+1])-
(T2x[i1+1]/py[i1+1]));
b24:=(2*(P2x[i1]-P2x[i1+1]))/(sqrt(py[i1+1])*py[i1+1]) +
((T2x[i1])/(sqrt(py[i1+1])) + ((T2x[i1+1])/(sqrt(py[i1+1])))));
X2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
Xdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;
2: If (I1 > N2) then
begin
b21:=P2y[i1];
b22:=T2y[i1];
b23:=(3*(P2y[i1+1]-P2y[i1]))/(sqrt(py[i1+1])/py[i1+1])-
(T2y[i1+1]/py[i1+1]));
b24:=(2*(P2y[i1]-P2y[i1+1]))/(sqrt(py[i1+1])*py[i1+1]) +
((T2y[i1])/(sqrt(py[i1+1])) + ((T2y[i1+1])/(sqrt(py[i1+1])))));
Y2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
Ydt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;
3: If (I1 > N2) then
begin
b21:=P2z[i1];
b22:=T2z[i1];
b23:=(3*(P2z[i1+1]-P2z[i1]))/(sqrt(py[i1+1])/py[i1+1])-
(T2z[i1+1]/py[i1+1]));
b24:=(2*(P2z[i1]-P2z[i1+1]))/(sqrt(py[i1+1])*py[i1+1]) +
((T2z[i1])/(sqrt(py[i1+1])) + ((T2z[i1+1])/(sqrt(py[i1+1])))));
Z2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
Zdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;
end;
{ fin del CASE Curva-2 }
end;
{ fin del For K1= 1 a 3 }

```



```

{ Calculo do vetor R(t)=(Xt[i]-X2t[i1])+(Yt[i]-Y2t[i1])+(Zt[i]-Z2t[i1]) }
If (i1 > N2) then
begin
Ra:=(Xt[i]-X2t[i1]);
Rb:=(Yt[i]-Y2t[i1]);
Rc:=(Zt[i]-Z2t[i1]);
{ Calculo do vetor R(t) X T1[i] }
Rxt1:=(Rb*Zdt[i] - Rc*Ydt[i]);
Ryt1:=(Rc*Xdt[i] - Ra*Zdt[i]);
Rzt1:=(Ra*Ydt[i] - Rb*Xdt[i]);
{ Calculo do Módulo do vetor R(t) X T1[i] }
Vn1:=sqr( sqr(Rxt1) + sqr(Ryt1) + sqr(Rzt1) );
{ Calculo do Módulo do vetor R(t) X T2[i] }
Rxt2:=(Rb*Zdt2[i] - Rc*Ydt2[i]);
Ryt2:=(Rc*Xdt2[i] - Ra*Zdt2[i]);
Rzt2:=(Ra*Ydt2[i] - Rb*Xdt2[i]);
{ Calculo do Módulo do vetor R(t) X T2[i] }
Vn2:=sqr( sqr(Rxt2) + sqr(Ryt2) + sqr(Rzt2) );
{ Calculo do produto N1 X N2 }
Pnx:=(((Ryt1/Vn1)*(Rzt2/Vn2))-((Rzt1/Vn1)*(Ryt2/Vn2)));
Pny:=(((Rzt1/Vn1)*(Rxt2/Vn2))-((Rxt1/Vn1)*(Rzt2/Vn2)));
Pnz:=(((Rxt1/Vn1)*(Ryt2/Vn2))-((Ryt1/Vn1)*(Rxt2/Vn2)));
Vn1n2:=sqr( sqr(Pnx) + sqr(Pny) + sqr(Pnz) );
Vn1:=sqr( sqr(Rxt1/Vn1) + sqr(Ryt1/Vn1) + sqr(Rzt1/Vn1) );
Vn2:=sqr( sqr(Rxt2/Vn2) + sqr(Ryt2/Vn2) + sqr(Rzt2/Vn2) );
Seno:=( Vn1n2/(Vn1*Vn2) ); { Função del Sen(t) }
* Calculo de las coordenadas para T2+h *
T2:=T2+h;
For k1:=1 to 3 do
begin

```

```

case k1 of
1: IF (I1 > N2) then
begin
b21:=P2x[i1];
b22:=T2x[i1];
b23:=(3*(P2x[i1+1]-P2x[i1]))/(sqrt(py[i1+1]))-(2*T2x[i1])/py[i1+1])
-(T2x[i1+1]/py[i1+1]);
b24:=(2*(P2x[i1]-P2x[i1+1]))/(sqrt(py[i1+1]))*py[i1+1]) +
((T2x[i1])/sqrt(py[i1+1])) + ((T2x[i1+1])/sqrt(py[i1+1])));
X2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
Xdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;
2: IF (I1 > N2) then
begin
b21:=P2y[i1];
b22:=T2y[i1];
b23:=(3*(P2y[i1+1]-P2y[i1]))/(sqrt(py[i1+1]))-(2*T2y[i1])/py[i1+1])
-(T2y[i1+1]/py[i1+1]);
b24:=(2*(P2y[i1]-P2y[i1+1]))/(sqrt(py[i1+1]))*py[i1+1]) +
((T2y[i1])/sqrt(py[i1+1])) + ((T2y[i1+1])/sqrt(py[i1+1])));
Y2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
Ydt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;
3: IF (I1 > N2) then
begin
b21:=P2z[i1];
b22:=T2z[i1];
b23:=(3*(P2z[i1+1]-P2z[i1]))/(sqrt(py[i1+1]))-(2*T2z[i1])/py[i1+1])
-(T2z[i1+1]/py[i1+1]);
b24:=(2*(P2z[i1]-P2z[i1+1]))/(sqrt(py[i1+1]))*py[i1+1]) +
((T2z[i1])/sqrt(py[i1+1])) + ((T2z[i1+1])/sqrt(py[i1+1])));
Z2t[i1]:=b21+b22*t2+b23*sqrt(t2)+b24*sqrt(t2)*t2;
Zdt2[i1]:=b22 + b23*t2*2 + b24*sqrt(t2)*3;
end;
end; { fim del CASE Curva-2 }
end; { fim del For KI= 1 a 3 }
} Calculo do vetor R(t)=(Xt[i]-X2t[i1])+(Yt[i]-Y2t[i1])+(Zt[i]-Z2t[i1]) }
Ra:=(Xt[i]-X2t[i1]);
Rb:=(Yt[i]-Y2t[i1]);
Rc:=(Zt[i]-Z2t[i1]);

```



```

} Calculo do vetor R(t) X T1[i] }
Rxt1:=(Rb*Zdt[i] - Rc*Ydt[i]);
Ryt1:=(Rc*Xdt[i] - Ra*Zdt[i]);
Rzt1:=(Ra*Ydt[i] - Rb*Xdt[i]);
} Calculo do Módulo do vetor }
Vn1:=sqrt( sqr(Rxt1) + sqr(Ryt1) + sqr(Rzt1));
} Calculo do vetor R(t) X T2[i] N1 }
Rxt2:=(Rb*Zdt2[i] - Rc*Ydt2[i]);
Ryt2:=(Rc*Xdt2[i] - Ra*Zdt2[i]);
Rzt2:=(Ra*Ydt2[i] - Rb*Xdt2[i]);
} Calculo do Módulo do vetor N2 }
Vn2:=sqrt( sqr(Rxt2) + sqr(Ryt2) + sqr(Rzt2) );
} Calculo do produto N1 X N2 }
Pdx:=(((Ryt1/Vn1)*(Rzt2/Vn2))-((Rzt1/Vn1)*(Ryt2/Vn2)));
Pdy:=(((Rxt1/Vn1)*(Rzt2/Vn2))-((Rzt1/Vn1)*(Rxt2/Vn2)));
Pdz:=(((Rxt1/Vn1)*(Ryt2/Vn2))-((Ryt1/Vn1)*(Rxt2/Vn2)));
Vd1d2:=sqrt( sqr(Pdx) +sqr(Pdy) +sqr(Pdz) );
Vn1:=sqrt( sqr(Rxt1/Vn1) + sqr(Ryt1/Vn1) + sqr(Rzt1/Vn1) );
Vn2:=sqrt( sqr(Rxt2/Vn2) + sqr(Ryt2/Vn2) + sqr(Rzt2/Vn2) );
} Função avaliada del Sen(t+h) }
Senod:=( Vd1d2/(Vn1*Vn2) );
} * CALCULO DA DERIVADA NUMERICA *
Senod:=( Senod - Seno ) / h;
T2:=T2-h;
TF:=( Seno)/(Senod);
} * PRUEBA DE NEWTON *
chrscr;
window(5,4,75,23);
textbackground (cyan);
textcolor (blue);
gotoxy(3,3);
write( EXCHAP VER 1.0 SUPERFICIES DESENVOLVIDAS );
textcolor (yellow);

```



```

gotoxy(1,5);
IF (abs(TF-T2)<h) then
  begin
    TextColor(blue);
    writeln('SE ENCONTRO A RULING Nº1, iii);
    writeln('Segmento de prueba Nº',i);
    TextColor(yellow);
    writeln('Ponto X1 :=',Xt[i]:10:4);
    writeln('Ponto Y1 :=',yt[i]:10:4);
    writeln('Ponto Z1 :=',zt[i]:10:4);
    writeln('Valor do Parametro T Inicial :=',t:10:4);
    writeln('Ponto X2 :=',X2t[i]:10:4);
    writeln('Ponto Y2 :=',y2t[i]:10:4);
    writeln('Ponto Z2 :=',z2t[i]:10:4);
    writeln('Valor do parametro T2:=',TF:10:4);
    writeln('Coord.(x) de N1XN2 :=',Pnx:10:4);
    writeln('Coord.(y) de N1XN2 :=',Pny:10:4);
    writeln('Coord.(z) de N1XN2 :=',Pnz:10:4);
    writeln('r1,x1,xt[i]:10:4);
    writeln('r1,yt[i]:10:4);
    writeln('r1,zt[i]:10:4);
    writeln('r1,x2t[i]:10:4);
    writeln('r1,y2t[i]:10:4);
    writeln('r1,z2t[i]:10:4);
    Ruling_Found:=True;
    Raiz:=True;
    Fechadura:=True;
    Stop:=true;
  readkey;
  end
  else
  begin
    TextColor(blue);
    writeln('Busca da Ruling Nº',i);
    writeln('Segmento de prova Nº',i);
    TextColor(yellow);
    writeln('Ponto X1 :=',Xt[i]:10:4);
    writeln('Ponto Y1 :=',yt[i]:10:4);
    writeln('Ponto Z1 :=',zt[i]:10:4);
    writeln('Valor do Parametro T Inicial :=',t:10:4);
    writeln('Ponto X2 :=',X2t[i]:10:4);
    writeln('Ponto Y2 :=',y2t[i]:10:4);
    writeln('Ponto Z2 :=',z2t[i]:10:4);
    writeln('Valor do Parametro T2 Inicial :=',t2:10:4);
  end
end

```

```

writeh('Valor do Parametro T2 Final =',tf,10:4);
writeh('Valor do vetor [N1XN2] :=',VnIn2:10:4);
writeh('Valor do vetor [N1XN2]/[N1].[N2] :=',seno:10:4);
writeh('Valor da 1er Deriv. [N1XN2]/[N1].[N2] :=',senod:10:4);
writeh('Coord.(x) de N1XN2 :=',Pnx:10:4);
writeh('Coord.(y) de N1XN2 :=',Pny:10:4);
writeh('Coord.(z) de N1XN2 :=',Pnz:10:4);
T2:=TF;
readkey;
end;
end; { End del IF11 > N1 }
end { Condicion del parametro <= }
else begin
  Ruling_Found:=True;
  Fechadura:=False;
end;
IF11=N2 then Ruling_Found:=True;
end; { End del While }
end; { If para existencia de la raiz }
end; { fin del For II = 1 a N2 }
End; { fin del For I = 1 t N1 }
close(r1);
END; { fin del procedimiento }
}*****
End. { FIN DE LA UNIT }

```

2. LISTAGEM DAS ROTINAS EM AUTOLISP

2.1 ROTINA "EXPANSAO.LSP"

Rotina necessária para desenhar as curvas geradas da expansão de chapamento no programa AUTOCAD.

```
(defun c:expansao(/ s x y z arq p l c d k b nbal)
(prompt "\n LEITURA DAS CURVAS GERADAS NO PROGRAMA
EXPCHAP Ver 1.0 ")
(arq (getint "\n Entre diretório de origem e o nome do arquivo : "))
(c (getint "\n Entre tamanho do campo numérico : "))
(d (getint "\n Entre numero de casas decimais : "))
(nbal (getint "\n Entre numero de balizas : "))
(r (open arq "r"))
;je primeiro ponto
(s (read-line f))
(x (atoi(substr s 1 c)))
(y (atoi(substr s (+ c 1) c)))
(z (atoi(substr s (+ (* 2 c) 1) c)))
;inicializa contador das balizas
(b 1)
(command "LAYER" "M" b ""))
(setq p l (list x z))
(command p l)
(setq k 1)
(while (/= s nil)
(setq k (+ k 1))
(x (atoi(substr s 1 c)))
(s (read-line f))
(x (atoi(substr s 1 c)))
(y (atoi(substr s (+ 1 c) c)))
(z (atoi(substr s (+ (* 2 c) 1) c)))
(setq p l (list x z))
(command p l)
(setq k 1)
(if (= k nbal)
(progn
(command ""))
(setq k 0))
))

```



```
(setq pnt2 (list x2 y2 ))
(command "LINE" pnt1 pnt2 "")
(setq ba (+ ba 1))
(if (= ba nbali)
    (setq t nil)
    )
)
(close fi)
)
```

2.3 ROTINA "RULINGXZ.LSP"

Rotina necessária para desenhar as rulingsgeradas no plano XZ do programa EXCHAP no programa AUTOCAD.

```
(defun c:rulingxz/ fi si x1 y1 z1 x2 y2 z2 arq pnt1 pnt2 ca de ki ba t nbali)
(prompt "n LEITURA DAS RULINGS GERADAS NO PROGRAMA
EXPCAP Ver 1.0 ")
(prompt "n A seguir devese introduzir as curvas geradas, uma por vez")
(setq arq (getstring "n Entre diretório de origem e o nome do arquivo : "))
(setq ca (getint "n Entre tamanho do campo numerico : "))
(setq de (getint "n Entre numero de casas decimais : "))
(setq nbali (getint "n Entre numero de balizas : "))
)
(setq fi (open arq "r"))
(setq t 1)
(setq nbali (* - nbali 1) 2))
(setq ba 0)
(while t
    (setq x1 (atof(substr si 1 ca)))
    (setq y1 (atof(substr si (+ ca 1) ca)))
    (setq z1 (atof(substr si (+ (* 2 ca) 1) ca)))
    (setq x2 (atof(substr si 1 ca)))
    (setq y2 (atof(substr si (+ 1 ca) ca)))
    (setq z2 (atof(substr si (+ (* 2 ca) 1) ca)))
    (command "LAYER" "M" ba "")
    (setq pnt1 (list x1 z1 ))
    (setq pnt2 (list x2 z2 ))
    (command "LINE" pnt1 pnt2 "")
)
```

```
(setq ba (+ ba 1))  
(if (= ba nbah)  
    (setq t nil)  
    )  
(close fi)  
)
```


APÊNDICE III

1. ARQUIVOS COM OS DADOS DE ENTRADA

1.1 ROTINA EXPANSÃO DE CHAPEAMENTO

Arquivo: BOCAS.DAT

Balizas	Zapata	Alétriz	QuinaInt	QuinaSup	CubFpal	CubSup	Borda
0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001
0.0826	0.0000	0.0000	0.0000	0.0000	0.0000	0.5326	0.5334
0.3175	0.0000	0.0000	0.0000	0.0000	0.0000	1.0398	1.0414
0.6731	0.0000	0.0000	0.0000	0.0000	0.0000	1.4456	1.4478
0.7493	0.0000	0.0000	0.0000	0.0000	0.0000	1.4986	1.4986
1.0986	0.0000	0.0000	0.0000	0.0000	0.0000	1.7780	1.7780
1.2573	0.0000	0.0000	0.0000	0.0000	0.0000	1.8574	1.8701
1.3589	0.0000	0.0000	0.0000	0.0000	0.0000	1.9177	1.9177
1.6002	0.0000	0.0000	0.0000	0.0000	0.0001	2.0258	2.0288
1.7653	0.0000	0.0000	0.0000	0.0000	0.3810	2.1146	2.1273
1.9685	0.0000	0.0000	0.0000	0.0000	0.5525	2.1908	2.1908
2.2733	0.0000	0.0000	0.0000	0.0000	0.7303	2.2860	2.2987
2.5781	0.0000	0.0000	0.0000	0.0000	0.8763	2.3749	2.3749
2.6670	0.0000	0.0000	0.0000	0.0001	0.8890	2.3727	2.3749
2.7813	0.0000	0.0000	0.0000	0.0318	0.9589	2.4321	2.4448
2.9718	0.0000	0.0000	0.0001	0.0762	1.0465	2.4870	2.4892
3.2893	0.0000	0.0000	0.0381	0.1524	1.1684	2.5654	2.5781
3.7973	0.0000	0.0000	0.1207	0.2731	1.3716	2.6924	2.7051
4.3053	0.0000	0.0000	0.2286	0.4128	1.5685	2.8194	2.8321
4.4164	0.0000	0.0000	0.2667	0.4509	1.6193	2.8512	2.8512
4.8133	0.0000	0.0000	0.3683	0.5588	1.7526	2.9337	2.9464
5.2113	0.0000	0.0000	0.5017	0.7176	1.9304	3.0480	3.0607
5.8293	0.0000	0.0000	0.6414	0.8827	2.0955	3.1560	3.1687
5.9944	0.0001	0.0001	0.6858	0.9271	1.8288	2.7005	3.1801
6.2548	0.0254	0.0762	0.7747	1.0351	2.2352	3.2512	3.2512
6.3373	0.0254	0.0794	0.7874	1.0541	2.2543	3.2639	3.2766
6.8453	0.0254	0.0889	0.9335	1.2129	2.4003	3.3528	3.3655
7.3533	0.0254	0.0953	1.0859	1.3843	2.5527	3.4417	3.4544
7.8613	0.0318	0.1016	1.2319	1.5494	2.6924	3.5306	3.5433
8.0931	0.0318	0.1080	1.3145	1.6383	2.7559	3.5687	3.5687
8.3693	0.0318	0.1143	1.3780	1.7082	2.8258	3.6132	3.6259
8.8773	0.0318	0.1207	1.5304	1.8796	2.9591	3.6957	3.7084
9.3853	0.0318	0.1334	1.6828	2.0447	3.0798	3.7656	3.7687
9.8933	0.0318	0.1397	1.8352	2.2098	3.2004	3.8354	3.8449
9.9314	0.0413	0.1441	1.8542	2.2289	3.2131	3.8418	3.8418
10.4013	0.0413	0.1492	1.9812	2.3749	3.3147	3.9053	3.9148
10.9093	0.0413	0.1651	2.1336	2.5337	3.4227	3.9688	3.9751

Balizas	11.4173	0.0413	0.1778	2.2733	2.6861	3.5243	4.0259	4.0323
Zapata	11.7697	0.0413	0.1778	2.3813	2.8004	3.6005	4.0640	4.0640
Aletriz	11.9253	0.0413	0.1778	2.4130	2.8448	3.6259	4.0831	4.0894
Quinalm	12.4333	0.0413	0.1905	2.5591	2.9972	3.7211	4.1402	4.1434
Quinasup	12.9413	0.0508	0.1969	2.6988	3.1496	3.8164	4.1910	4.1942
Cubpal	13.4493	0.0508	0.2032	2.8321	3.2957	3.9021	4.2323	4.2323
Cubsup	13.6081	0.0508	0.2032	2.8829	3.3465	3.9370	4.2482	4.2482
Borda	13.9573	0.0508	0.2159	2.9655	3.4417	3.9878	4.2799	4.2799
	14.4653	0.0508	0.2223	3.0925	3.5751	4.0640	4.3244	4.3244
	14.9733	0.0508	0.2350	3.2195	3.7084	4.1402	4.3688	4.3688
	15.4464	0.0508	0.2413	3.3338	3.8354	4.2101	4.4006	4.4006
	15.4813	0.0603	0.2445	3.3338	3.8418	4.2101	4.4006	4.4006
	15.9893	0.0603	0.2540	3.4481	3.9624	4.2799	4.4450	4.4450
	16.4973	0.0603	0.2604	3.5560	4.0767	4.3434	4.4768	4.4768
	17.0053	0.0603	0.2667	3.6576	4.1783	4.3942	4.5085	4.5085
	17.2847	0.0603	0.2794	3.7084	4.2355	4.4323	4.5212	4.5212
	17.5133	0.0603	0.2794	3.7465	4.2736	4.4450	4.5339	4.5339
	18.0213	0.0667	0.2921	3.8227	4.3561	4.4958	4.5593	4.5593
	18.5293	0.0667	0.2985	3.8989	4.4387	4.5339	4.5657	4.5657
	19.0373	0.0667	0.3080	3.9624	4.5022	4.5657	4.5847	4.5847
	19.5453	0.0667	0.3175	4.0132	4.5530	4.5974	4.6101	4.6101
	20.0533	0.0667	0.3302	4.0513	4.5974	4.6165	4.6228	4.6228
	20.5613	0.0730	0.3397	4.0831	4.6292	4.6323	4.6355	4.6355
	20.9614	0.0730	0.3429	4.0958	4.6482	4.6482	4.6482	4.6482
	21.0693	0.0730	0.3493	4.0989	4.6482	4.6482	4.6482	4.6482
	21.5773	0.0730	0.3556	4.0989	4.6482	4.6482	4.6482	4.6482
	22.0853	0.0730	0.3651	4.1021	4.6482	4.6482	4.6482	4.6482
	22.5933	0.0730	0.3747	4.1021	4.6450	4.6450	4.6450	4.6450
	23.1013	0.0762	0.3874	4.0989	4.6419	4.6419	4.6419	4.6419
	24.1173	0.0762	0.4159	4.0831	4.6260	4.6260	4.6260	4.6260
	24.6253	0.0762	0.4286	4.0735	4.6165	4.6165	4.6165	4.6165
	24.6380	0.0762	0.4318	4.0704	4.6165	4.6165	4.6165	4.6165
	25.1333	0.0762	0.4382	4.0577	4.6038	4.6038	4.6038	4.6038
	25.6413	0.0762	0.4509	4.0354	4.5847	4.5847	4.5847	4.5847
	26.1493	0.0762	0.4572	4.0132	4.5657	4.5657	4.5657	4.5657
	26.6573	0.0762	0.4667	3.9910	4.5466	4.5466	4.5466	4.5466
	27.1653	0.0762	0.4794	3.9688	4.5276	4.5276	4.5276	4.5276
	28.1813	0.0762	0.4953	3.9180	4.4831	4.4831	4.4831	4.4831
	28.3147	0.0762	0.4953	3.9116	4.4768	4.4768	4.4768	4.4768
	28.6893	0.0762	0.5048	3.8926	4.4577	4.4577	4.4577	4.4577
	29.1973	0.0762	0.5080	3.8608	4.4260	4.4260	4.4260	4.4260
	29.7053	0.0762	0.5144	3.8354	4.3942	4.3942	4.3942	4.3942
	30.2133	0.0762	0.5207	3.8005	4.3656	4.3656	4.3656	4.3656
	30.7213	0.0762	0.5271	3.7687	4.3339	4.3339	4.3339	4.3339
	31.2293	0.0762	0.5302	3.7402	4.3053	4.3053	4.3053	4.3053

Balizas	Zapata	Aletriz	Quinalm	Quinasup	Cubfpal	Cubsup	Borda
31.7373	0.0762	0.5334	3.7021	4.2672	4.2672	0.0000	4.2672
31.9913	0.0762	0.5334	3.6830	4.2482	4.2482	0.0000	4.2482
32.2453	0.0762	0.5334	3.6576	4.2291	4.2291	0.0000	4.2291
32.7533	0.0762	0.5334	3.6322	4.1910	4.1910	0.0000	4.1910
33.2613	0.0762	0.5334	3.5878	4.1466	4.1466	0.0000	4.1466
33.7693	0.0762	0.5334	3.5433	4.1021	4.1021	0.0000	4.1021
33.8296	0.0762	0.5334	3.5433	4.1021	4.1021	0.0000	4.1021
34.2773	0.0762	0.5334	3.5052	4.0640	4.0640	0.0000	4.0640
34.7853	0.0762	0.5334	3.4608	4.0196	4.0196	0.0000	4.0196
35.2933	0.0762	0.5207	3.4163	3.9751	3.9751	0.0000	3.9751
35.6680	0.0762	0.5048	3.3846	3.9434	3.9434	0.0000	3.9434
35.8013	0.0762	0.4953	3.3719	3.9307	3.9307	0.0000	3.9307
36.2077	0.0762	0.4572	3.3274	3.8862	3.8862	0.0000	3.8862
36.3093	0.0000	0.4509	3.3274	3.8799	3.8799	0.0000	3.8799
36.8173	0.0000	0.3683	3.2766	3.8291	3.8291	0.0000	3.8291
37.3253	0.0000	0.2286	3.2322	3.7846	3.7846	0.0000	3.7592
37.5063	0.0000	0.1270	3.2068	3.7656	3.7656	0.0000	3.7338
37.7317	0.0000	0.0000	3.1750	3.7338	3.7275	0.0000	3.7211
37.8333	0.0000	0.0000	3.1750	3.7275	3.6957	0.0000	3.6132
38.0873	0.0000	0.0000	3.1496	3.6957	3.6767	0.0000	3.4576
38.3413	0.0000	0.0000	3.1242	3.6767	3.6576	0.0000	3.2258
38.4048	0.0000	0.0000	3.1115	3.6576	3.6576	0.0000	2.9718
38.4556	0.0000	0.0000	3.0988	3.6195	3.5814	0.0000	0.0000
38.8493	0.0000	0.0000	3.0671	3.5814	3.5814	0.0000	0.0000
39.3573	0.0000	0.0000	3.0099	3.5687	3.4290	0.0000	0.0000
39.8653	0.0000	0.0000	2.9528	3.5116	3.0099	0.0000	0.0000
39.9415	0.0000	0.0000	2.9210	3.4798	0.0000	0.0000	0.0000
40.3733	0.0000	0.0000	2.8893	3.4481	0.0000	0.0000	0.0000
40.8813	0.0000	0.0000	2.8258	3.3846	0.0000	0.0000	0.0000
41.3893	0.0000	0.0000	2.7623	3.3211	0.0000	0.0000	0.0000
41.6179	0.0000	0.0000	2.7178	3.2512	0.0000	0.0000	0.0000
41.7449	0.0000	0.0000	2.7051	3.1623	0.0000	0.0000	0.0000
41.9227	0.0000	0.0000	2.6924	3.0226	0.0000	0.0000	0.0000
42.0370	0.0000	0.0000	2.6797	2.8702	0.0000	0.0000	0.0000
42.1005	0.0000	0.0000	2.6670	2.6670	0.0000	0.0000	0.0000

Arquivo: ALTURAS.DAT

Balizas	Zapata	Aletriz	Quinalnt	Quinasup	Cubfpal	Cubsup	Borda
0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	8.5344	8.5344
0.0826	0.0000	0.0000	0.0000	0.0000	0.0000	8.5201	8.5328
0.3175	0.0000	0.0000	0.0000	0.0000	0.0000	8.5185	8.5312
0.6731	0.0000	0.0000	0.0000	0.0000	0.0000	8.5122	8.5249
0.7493	0.0000	0.0000	0.0000	0.0000	0.0000	8.5011	8.5233
1.0986	0.0000	0.0000	0.0000	0.0000	0.0000	8.4979	8.5201
1.2573	0.0000	0.0000	0.0000	0.0000	0.0000	8.4963	8.5185
1.3589	0.0000	0.0000	0.0000	0.0000	0.0000	8.4868	8.5090
1.6002	0.0000	0.0000	0.0000	0.0000	0.0000	8.4582	8.4709
1.7653	0.0000	0.0000	0.0000	0.0000	0.0000	8.4836	8.5058
1.9685	0.0000	0.0000	0.0000	0.0000	0.0000	8.4741	8.4963
2.2733	0.0000	0.0000	0.0000	0.0000	0.0000	8.4646	8.4868
2.5781	0.0000	0.0000	0.0000	0.0000	0.0000	8.4614	8.4836
2.6670	0.0000	0.0000	0.0000	0.0000	0.0000	8.4265	8.4392
2.7813	0.0000	0.0000	0.0000	0.0000	0.0000	8.4455	8.4677
2.9718	0.0000	0.0000	0.0000	0.0000	0.0000	8.4138	8.4265
3.2893	0.0000	0.0000	0.0000	0.0000	0.0000	8.4265	8.4487
3.7973	0.0000	0.0000	0.0000	0.0000	0.0000	8.4074	8.4296
4.3053	0.0000	0.0000	0.0000	0.0000	0.0000	8.3820	8.4042
4.4164	0.0000	0.0000	0.0000	0.0000	0.0000	8.3725	8.3947
4.8133	0.0000	0.0000	0.0000	0.0000	0.0000	8.3630	8.3852
5.3213	0.0000	0.0000	0.0000	0.0000	0.0000	8.3376	8.3598
5.8293	0.0000	0.0000	0.0000	0.0000	0.0000	8.3185	8.3407
5.9944	1.1176	1.4986	3.4036	3.9878	5.8928	8.2962	8.3185
6.2548	1.1049	1.4859	3.3782	3.9688	5.8877	8.2936	8.3158
6.3373	1.0922	1.4732	3.3655	3.9561	5.8928	8.2931	8.3153
6.8453	1.0859	1.4605	3.2893	3.8862	5.8801	8.2741	8.2963
7.3533	1.0668	1.4351	3.2195	3.8100	5.8547	8.2423	8.2645
7.8613	1.0414	1.4097	3.1433	3.7338	5.8293	8.2233	8.2455
8.0931	1.0287	1.3970	3.1052	3.6957	5.8166	8.2074	8.2296
8.3693	1.0160	1.3780	3.0671	3.6576	5.8103	8.2042	8.2264
8.8773	0.9906	1.3526	2.9972	3.5751	5.7912	8.1788	8.2010
9.3853	0.9684	1.3272	2.9210	3.4989	5.7658	8.1471	8.1693
9.8933	0.9462	1.3018	2.8448	3.4227	5.7468	8.1280	8.1502
9.9314	0.9398	1.2954	2.8321	3.4163	5.7404	8.1058	8.1280
10.4013	0.9208	1.2700	2.7559	3.3465	5.7277	8.1026	8.1248
10.9093	0.8954	1.2446	2.6670	3.2639	5.7023	8.0772	8.0994
11.4173	0.8763	1.2192	2.5908	3.1941	5.6833	8.0582	8.0804
11.7697	0.8509	1.2065	2.5337	3.1369	5.6642	8.0423	8.0645
11.9253	0.8509	1.1938	2.5019	3.1115	5.6579	8.0328	8.0550
12.4333	0.8255	1.1748	2.4257	3.0353	5.6388	8.0137	8.0359
12.9413	0.8001	1.1494	2.3432	2.9591	5.6198	7.9883	8.0105

Balizas	13.4493	0.7811	1.1303	2.2606	2.8893	5.5944	7.9629	7.9883
Zapata	13.6081	0.7811	1.1176	2.2352	2.8575	5.5880	7.9502	7.9756
Aletriz	13.9573	0.7557	1.1113	2.1781	2.8131	5.5690	7.9439	7.9693
QuinaInt	14.4653	0.7366	1.0859	2.0892	2.7305	5.5436	7.9185	7.9439
QuinaSup	14.9733	0.7112	1.0605	2.0066	2.6543	5.5245	7.8994	7.9248
	15.4464	0.6858	1.0478	1.9304	2.5845	5.4991	7.8867	7.9121
	15.4813	0.6858	1.0414	1.9241	2.5781	5.4991	7.8740	7.8994
	15.9893	0.6668	1.0287	1.8415	2.5019	5.4737	7.8550	7.8804
	16.4973	0.6414	1.0097	1.7653	2.4321	5.4547	7.8359	7.8613
	17.0053	0.6160	0.9906	1.6828	2.3559	5.4356	7.8105	7.8359
	17.2847	0.6033	0.9811	1.6383	2.3241	5.4229	7.7978	7.8232
	17.5133	0.5906	0.9716	1.6066	2.2860	5.4102	7.7915	7.8169
	18.0213	0.5715	0.9462	1.5304	2.2162	5.3912	7.7724	7.7978
	18.5293	0.5461	0.9271	1.4669	2.1527	5.3721	7.7531	7.7724
	19.0373	0.5271	0.9144	1.4034	2.0828	5.3531	7.7332	7.7531
	19.5453	0.5017	0.8954	1.3462	2.0320	5.3404	7.7141	7.7332
	20.0533	0.4763	0.8763	1.2954	1.9812	5.3213	7.6950	7.7141
	20.5613	0.4540	0.8636	1.2478	1.9241	5.3054	7.6759	7.6950
	20.9614	0.4286	0.8446	1.2192	1.8860	5.2959	7.6568	7.6759
	21.0693	0.4255	0.8382	1.2065	1.8796	5.2959	7.6377	7.6568
	21.5773	0.4064	0.8192	1.1811	1.8479	5.2896	7.6186	7.6377
	22.0853	0.3810	0.8001	1.1557	1.8161	5.2832	7.5995	7.6186
	22.5933	0.3620	0.7811	1.1367	1.7971	5.2832	7.5804	7.5995
	23.1013	0.3366	0.7652	1.1303	1.7812	5.2800	7.5613	7.5804
	24.1173	0.2858	0.7366	1.1303	1.7653	5.2864	7.5422	7.5613
	24.6253	0.2635	0.7303	1.1367	1.7717	5.2896	7.5231	7.5422
	24.6380	0.2604	0.7239	1.1430	1.7717	5.2896	7.5040	7.5231
	25.1333	0.2381	0.7112	1.1557	1.7812	5.2959	7.4849	7.5040
	25.6413	0.2159	0.6985	1.1748	1.7939	5.3054	7.4658	7.4849
	26.1493	0.1937	0.6922	1.2033	1.8161	5.3150	7.4467	7.4658
	26.6573	0.1683	0.6890	1.2351	1.8383	5.3245	7.4276	7.4467
	27.1653	0.1461	0.6858	1.2700	1.8669	5.3340	7.4085	7.4276
	28.1813	0.0984	0.7049	1.3653	1.9368	5.3594	7.3894	7.4085
	28.3147	0.0905	0.7074	1.3780	1.9495	5.3658	7.3703	7.3894
	28.6893	0.0762	0.7112	1.4097	1.9749	5.3721	7.3512	7.3703
	29.1973	0.0540	0.7303	1.4669	2.0130	5.3785	7.3321	7.3512
	29.7053	0.0286	0.7493	1.5177	2.0511	5.3912	7.3130	7.3321
	30.2133	0.0064	0.7747	1.5748	2.0955	5.3975	7.2939	7.3130
	30.7213	-0.0191	0.8128	1.6383	2.1400	5.4102	7.2748	7.2939
	31.2293	-0.0413	0.8636	1.7018	2.1908	5.4197	7.2557	7.2748
	31.7373	-0.0667	0.9081	1.7653	2.2416	5.4356	7.2366	7.2557
	31.9913	-0.0778	0.9398	1.8034	2.2670	5.4420	7.2175	7.2366
Borda	6.3564	6.3500	6.3341	6.3246	6.3119	6.3056	6.2929	6.2865
CubSup	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
CubPal	5.4420	5.4356	5.4197	5.4102	5.3975	5.3912	5.3785	5.3721
QuinaSup	2.2670	2.2416	2.1908	2.1400	2.0955	2.0511	2.0130	1.9749
QuinaInt	1.8034	1.7653	1.7018	1.6383	1.5748	1.5177	1.4669	1.4097
Aletriz	0.9398	0.9081	0.8636	0.8128	0.7747	0.7493	0.7303	0.7112
Zapata	-0.0778	-0.0667	-0.0413	-0.0191	0.0064	0.0286	0.0540	0.0762
Balizas	31.9913	31.7373	31.2293	30.7213	30.2133	29.7053	29.1973	28.6893

Balizas	32.2453	-0.0889	0.9652	1.8352	2.2987	5.4483	0.0000	6.3627
	32.7533	-0.1111	1.0351	1.9114	2.3559	5.4610	0.0000	6.3754
	33.2613	-0.1365	1.1113	1.9876	2.4130	5.4674	0.0000	6.3818
	33.7693	-0.1588	1.1938	2.0701	2.4765	5.4801	0.0000	6.3945
	33.8296	-0.1619	1.2192	2.0828	2.4892	5.4801	0.0000	6.3945
	34.2773	-0.1810	1.2954	2.1463	2.5400	5.4991	0.0000	6.4135
	34.7853	-0.2064	1.4161	2.2352	2.6099	5.5118	0.0000	6.4262
	35.2933	-0.2286	1.5367	2.3241	2.6797	5.5245	0.0000	6.4389
	35.6680	-0.2477	1.6510	2.4003	2.7432	5.5372	0.0000	6.4516
	35.8013	-0.2540	1.6764	2.4194	2.7559	5.5404	0.0000	6.4548
	36.2077	-0.2794	1.7844	2.4892	2.8004	5.5372	0.0000	6.4326
	36.3093	0.0000	1.8161	2.5146	2.8321	5.5563	0.0000	6.4707
	36.8173	0.0000	1.9622	2.6099	2.9115	5.5721	0.0000	6.4865
	37.3253	0.0000	2.0892	2.7115	2.9909	5.5880	0.0000	6.5024
	37.5063	0.0000	2.1336	2.7496	3.0163	5.5880	0.0000	6.5024
	37.7317	0.0000	2.1717	2.7813	3.0480	5.5880	0.0000	6.4897
	37.8333	0.0000	2.8067	2.8067	3.0671	5.6007	0.0000	6.5151
	38.0873	0.0000	2.8448	2.8448	3.0937	5.5880	0.0000	6.5024
	38.3413	0.0000	2.9147	2.9147	3.1560	5.6198	0.0000	6.5342
	38.4048	0.0000	2.8956	2.8956	3.1369	5.6007	0.0000	6.5151
	38.4556	0.0000	2.9210	2.9210	3.1560	5.6083	0.0000	0.0000
	38.8493	0.0000	3.0099	3.0099	3.2385	5.6325	0.0000	0.0000
	39.3573	0.0000	3.1115	3.1115	3.3274	5.6515	0.0000	0.0000
	39.8653	0.0000	3.2131	3.2131	3.4100	5.6642	0.0000	0.0000
	39.9415	0.0000	3.2131	3.2131	3.4036	0.0000	0.0000	0.0000
	40.3733	0.0000	3.3020	3.3020	3.4798	0.0000	0.0000	0.0000
	40.8813	0.0000	3.4036	3.4036	3.5814	0.0000	0.0000	0.0000
	41.3893	0.0000	3.5052	3.5052	3.5179	0.0000	0.0000	0.0000
	41.6179	0.0000	3.5433	3.5433	3.7084	0.0000	0.0000	0.0000
	41.7449	0.0000	3.5814	3.5814	3.7211	0.0000	0.0000	0.0000
	41.9227	0.0000	3.6068	3.6068	3.7135	0.0000	0.0000	0.0000
	42.0370	0.0000	3.6195	3.6195	3.6830	0.0000	0.0000	0.0000
	42.1005	0.0000	3.6576	3.6576	0.0000	0.0000	0.0000	0.0000

Eixo X	Eixo Y	Eixo Z
0.0001	0.0000	8.5344
0.0826	0.0000	8.4074
0.3175	0.0000	8.0772
0.6731	0.0000	7.4803
0.7493	0.0000	7.3914
1.0986	0.0000	6.8072
1.2573	0.0000	6.6040
1.3589	0.0000	6.4262
1.6002	0.0000	6.0706
1.7653	0.0000	5.8166
1.9685	0.0000	5.4864
2.2733	0.0000	4.9784
2.5781	0.0000	4.4958
2.6670	0.0000	4.4387
2.7813	0.0000	4.1910
2.9718	0.0000	3.8545
3.2893	0.0000	3.4036
3.7973	0.0000	2.6162
4.3053	0.0000	1.8542
4.4164	0.0000	1.7145
4.8133	0.0000	1.4478
5.3213	0.0000	1.2446
5.8293	0.0000	1.1430
5.9944	0.0000	1.1176
6.2548	0.0000	1.1049
6.3373	0.0000	1.0922
6.8453	0.0000	1.0859
7.3533	0.0000	1.0668
7.8613	0.0000	1.0414
8.0931	0.0000	1.0287
8.3693	0.0000	1.0160
8.8773	0.0000	0.9906
9.3853	0.0000	0.9684
9.8933	0.0000	0.9462
9.9314	0.0000	0.9398
10.4013	0.0000	0.9208
10.9093	0.0000	0.8954
11.4173	0.0000	0.8763
11.7697	0.0000	0.8636
11.9253	0.0000	0.8509
12.4333	0.0000	0.8255
12.9413	0.0000	0.8001
13.4493	0.0000	0.7811

Arquivo: PERFIL.DAT

Eixo X	Eixo Y	Eixo Z
13.6081	0.0000	0.7747
13.9573	0.0000	0.7557
14.4653	0.0000	0.7366
14.9733	0.0000	0.7112
15.4464	0.0000	0.6858
15.4813	0.0000	0.6858
15.9893	0.0000	0.6668
16.4973	0.0000	0.6414
17.0053	0.0000	0.6160
17.2847	0.0000	0.6096
17.5133	0.0000	0.6033
18.0213	0.0000	0.5715
18.5293	0.0000	0.5461
19.0373	0.0000	0.5271
19.5453	0.0000	0.5017
20.0533	0.0000	0.4763
20.5613	0.0000	0.4540
20.9614	0.0000	0.4286
21.0693	0.0000	0.4255
21.5773	0.0000	0.4064
22.0853	0.0000	0.3810
22.5933	0.0000	0.3620
23.1013	0.0000	0.3366
24.1173	0.0000	0.2858
24.6253	0.0000	0.2635
24.6380	0.0000	0.2604
25.1333	0.0000	0.2381
25.6413	0.0000	0.2159
26.1493	0.0000	0.1937
26.6573	0.0000	0.1683
27.1653	0.0000	0.1461
28.1813	0.0000	0.0984
28.3147	0.0000	0.0810
28.6893	0.0000	0.0762
29.1973	0.0000	0.0540
29.7053	0.0000	0.0286
30.2133	0.0000	0.0064
30.7213	0.0000	-0.0191
31.2293	0.0000	-0.0413
31.7373	0.0000	-0.0667
31.9913	0.0000	-0.0778
32.2453	0.0000	-0.0889
32.7533	0.0000	-0.1111
33.2613	0.0000	-0.1365
33.7693	0.0000	-0.1588

Eixo X	Eixo Y	Eixo Z
--------	--------	--------

33.8296	0.0000	-0.1619
34.2773	0.0000	-0.1810
34.7853	0.0000	-0.2064
35.2933	0.0000	-0.2286
35.6680	0.0000	-0.2477
35.8013	0.0000	-0.2540
36.2077	0.0000	-0.2794
36.3093	0.0000	-0.2794
36.8173	0.0000	-0.2794
37.3253	0.0000	-0.2794
37.5063	0.0000	-0.2794
37.7317	0.0000	2.1794
37.8333	0.0000	2.1944
38.0873	0.0000	2.2670
38.3413	0.0000	2.3590
38.4048	0.0000	2.3622
38.4556	0.0000	2.3749
38.8493	0.0000	2.5162
39.3573	0.0000	2.6924
39.8653	0.0000	2.8321
39.9415	0.0000	2.8702
40.3733	0.0000	2.9734
40.8813	0.0000	3.1623
41.3893	0.0000	3.3242
41.6179	0.0000	3.4036
41.7449	0.0000	3.4544
41.9227	0.0000	3.5052
42.0370	0.0000	3.5560
42.1005	0.0000	3.5814

Arquivo: INICIO.DAT

Eixo X	Eixo Y	Eixo Z
--------	--------	--------

5.9944	0.0000	1.1176
5.9944	0.0000	1.4986
2.9718	0.0000	3.8545
2.6670	0.0000	4.4387
1.6002	0.0000	6.0706
0.0001	0.0000	8.5344
0.0001	0.0000	8.5344
0.0001	0.0000	Zapata

1.2 ROTINA BUSCA DAS RUINIGS

Arquivo: BORDA.DAT

Eixo X	Eixo Y	Eixo Z
0.0000	0.0000	7.9000
0.2000	0.1517	7.8200
0.5000	0.3836	7.7500
1.0000	0.7800	7.6000
2.5000	2.1900	7.2000
4.0000	3.2800	6.8500
5.5000	4.0100	6.5000
7.0000	4.4500	6.2200
8.5000	4.6800	6.0500
10.0000	4.7100	5.9500
11.5000	4.6000	5.9000
13.0000	4.3500	5.9900
14.5000	4.0000	6.1000
16.0000	3.5500	6.2500
16.4000	3.4900	6.2900

Arquivo: QUNASUP.DAT

Eixo X	Eixo Y	Eixo Z
0.2000	0.0001	6.7200
1.0000	0.7800	6.5000
2.5000	2.1900	6.1000
4.0000	3.2800	5.7100
5.5000	4.0100	5.4000
7.0000	4.4500	5.1000
8.5000	4.6800	4.9000
10.0000	4.7100	4.8200
11.5000	4.6000	4.8000
13.0000	4.3500	4.8100
14.5000	4.0000	4.9100
16.0000	3.5500	5.1000
16.4000	3.4900	5.1200

Eixo X	Eixo Y	Eixo Z
0.5000	0.0000	5.1000
1.0000	0.0000	3.0000
2.5000	0.0000	1.0000
4.0000	0.0000	0.6100
5.5000	0.0000	0.5200
7.0000	0.0000	0.5000
8.5000	0.0000	0.6500
10.0000	0.0000	0.9000
11.5000	0.0000	1.2200
13.0000	0.0000	1.7000
14.5000	0.0000	2.1500
15.9500	0.0000	2.6000
16.0000	0.0000	3.2000
16.4000	0.0000	5.0500
16.5500	0.0000	6.3000

Arquivo: PERFILK.DAT

Eixo X	Eixo Y	Eixo Z
0.5000	0.0001	5.1000
1.0000	0.4000	4.9000
2.5000	1.8000	4.3000
4.0000	2.9000	3.7500
5.5000	3.6800	3.3000
7.0000	4.2000	2.9700
8.5000	4.4800	2.7000
10.0000	4.5000	2.5900
11.5000	4.3500	2.5800
13.0000	3.9900	2.7100
14.5000	3.5000	3.0500
16.0000	2.9000	3.4800
16.1000	2.8100	3.5000

Arquivo: QUINTAINF.DAT