

TARCISIO ANTONIO HESS COELHO

OTIMIZAÇÃO DO ÂNGULO DE TRANSMISSÃO NA SÍNTESE
DE MECANISMOS PLANOS DE QUATRO BARRAS

Dissertação apresentada à
Escola Politécnica da USP
para obtenção do Título
de Mestre em Engenharia

São Paulo , 1990

P.D. 1193

TARCISIO ANTONIO HESS COELHO

Engº Mecânico, Escola Politécnica da USP, 1987

OTIMIZAÇÃO DO ÂNGULO DE TRANSMISSÃO NA SÍNTESE
DE MECANISMOS PLANOS DE QUATRO BARRAS

Dissertação apresentada à
Escola Politécnica da USP
para obtenção do Título
de Mestre em Engenharia

Orientador : Prof. Dr. Valter Francisco Arruda Alves
Professor Doutor do Departamento de
Engenharia Mecânica da EPUSP.

São Paulo , 1990

A G R A D E C I M E N T O S

Ao Prof. Dr. Valter F. A. Alves pela orientação.

Ao Prof. Dr. Carlos C. Tu pela colaboração
e dedicação.

Aos amigos e colegas do PMC que, direta ou indiretamente,
contribuíram para a realização deste trabalho.

RESUMO

Este trabalho trata da síntese ótima de quadriláteros articulados planos . O objetivo da síntese é a geração de movimento e trajetória para três e quatro posições de precisão. A otimização se refere ao parâmetro ângulo de transmissão. Apresenta-se também, neste trabalho, como os programas de otimização funcionam em exemplos retirados da literatura de projeto de mecanismos.

ABSTRACT

This work deals with optimum synthesis of four-bar linkages. The synthesis objective is motion and path generation for three and four precision positions. The optimization concerns the parameter transmission angle. The effectiveness of the optimization programs has been tested using examples obtained from the mechanism design literature.

ÍNDICE

	Página
Notação.....	i
Índice de figuras.....	ii
Índice de tabelas.....	vi
1. Introdução.....	1
2. Revisão Bibliográfica.....	2
2.1 - Projeto de mecanismos auxiliado por computador....	2
2.1.1 - Década de 50.....	2
2.1.2 - Década de 60.....	3
2.1.3 - Década de 70.....	3
2.1.4 - Década de 80.....	4
2.2 - Otimização no projeto de mecanismos.....	5
2.2.1 - Década de 60.....	5
2.2.2 - Década de 70.....	5
2.2.3 - Década de 80.....	7
3. Solução Analítica para a síntese.....	9
3.1 - Introdução.....	9
3.2 - Alguns fundamentos sobre mecanismos.....	9
3.2.1 - Geração de função.....	11
3.2.2 - Geração de trajetória.....	11
3.2.3 - Geração de movimento.....	13
3.2.4 - Algumas noções da teoria de Burmester.....	14
3.2.5 - Ângulo de transmissão.....	15
3.3 - Modelamento das barras articuladas.....	17
3.4 - Equação da diâada.....	17

3.5 - Os sistemas de equações para solução dos problemas de síntese para algumas posições de precisão.....	20
3.5.1 - Duas posições.....	20
3.5.2 - Três posições.....	21
3.5.3 - Quatro posições.....	21
3.5.4 - Cinco posições.....	22
3.6 - Geração de Movimento, de Trajetória e de Função.....	22
3.6.1 - Geração de Movimento.....	22
3.6.2 - Geração de Trajetória com temporização.....	23
3.6.3 - Geração de Função.....	24
3.7 - Procedimento de solução para três posições de precisão.....	24
3.8 - Procedimento de solução para quatro posições de precisão.....	28
4. Método Generalizado do Gradiente Reduzido.....	31
5. Otimização do ângulo de transmissão.....	35
5.1 - Introdução.....	35
5.2 - Cálculo do ângulo de transmissão.....	35
5.3 - Função objetivo : desvio do ângulo de transmissão.....	36
5.3.1 - Três posições de precisão.....	37
5.3.2 - Quatro posições de precisão.....	38
5.4 - Restrições.....	39
5.4.1 - Três posições de precisão.....	39
5.4.1.1 - Otimização de 4 variáveis.....	40
5.4.1.2 - Otimização de 12 variáveis.....	40
5.4.2 - Quatro posições de precisão.....	40

5.4.2.1 - Otimização de 2 variáveis.....	41
5.4.2.2 - Otimização de 13 variáveis.....	42
6. Problemas Propostos.....	43
6.1 - Introdução.....	43
6.2 - Problema de três posições.....	43
6.2.1 - Otimização com 4 variáveis livres.....	44
6.2.2 - Otimização com 12 variáveis livres.....	45
6.3 - Problema de quatro posições.....	46
6.3.1 - Otimização com 2 variáveis livres.....	47
6.3.2 - Otimização com 13 variáveis livres.....	48
6.4 - Problema da rotação completa.....	49
7. Resultados.....	52
7.1 - Introdução.....	52
7.2 - Resultados da otimização de 3 posições.....	54
7.2.1 - Síntese sem otimização.....	54
7.2.1.1 - Dados das posições de precisão.....	54
7.2.1.2 - Dados do mecanismo obtido.....	55
7.2.2 - Otimização com 4 variáveis livres.....	55
7.2.2.1 - Dados da função de otimização.....	55
7.2.2.2 - Dados das posições de precisão.....	58
7.2.2.3 - Dados do mecanismo obtido.....	59
7.2.3 - Otimização com 12 variáveis.....	62
7.2.3.1 - Dados da função de otimização.....	62
7.2.3.2 - Dados das posições de precisão.....	63
7.2.3.3 - Dados do mecanismo obtido.....	66
7.2.4 - Resultados do ângulo de transmissão.....	67
7.3 - Resultados da otimização de 4 posições.....	75
7.3.1 - Síntese sem otimização.....	75
7.3.1.1 - Dados das posições de precisão.....	75

7.2.1.2 - Dados do mecanismo obtido.....	76
7.3.2 - Otimização com 2 variáveis livres.....	76
7.3.2.1 - Dados da função de otimização.....	76
7.3.2.2 - Dados das posições de precisão.....	79
7.3.2.3 - Dados do mecanismo obtido.....	80
7.3.3 - Otimização com 13 variáveis.....	83
7.3.3.1 - Dados da função de otimização.....	83
7.3.3.2 - Dados das posições de precisão.....	84
7.3.3.3 - Dados do mecanismo obtido.....	85
7.3.4 - Resultados do ângulo de transmissão.....	88
7.4 - Resultados da otimização com rotação completa....	96
7.4.1 - Síntese sem otimização.....	96
7.4.1.1 - Dados das posições de precisão.....	96
7.4.1.2 - Dados do mecanismo obtido.....	96
7.4.2 - Otimização com 4 variáveis	99
7.4.2.1 - Dados da função de otimização.....	99
7.4.2.2 - Dados das posições de precisão.....	100
7.4.2.3 - Dados do mecanismo obtido.....	101
7.4.3 - Resultados do ângulo de transmissão.....	104
7.5 - SYN 3.....	110
7.6 - SYN 4.....	112
7.7 - POS.....	112
7.8 - OPT 3.....	113
7.9 - Linguagens de programação.....	114
7.10 - Computador.....	114
8. Discussão.....	115
8.1 - Introdução.....	115
8.2 - Otimização para três posições de precisão.....	115
8.3 - Otimização para quatro posições de precisão.....	116

8.4 - Otimização para rotação completa.....	117
8.5 - Considerações quanto ao número de pontos e a posição destes pontos no cálculo da função objetivo.....	119
8.6 - Comentários quanto aos exemplos escolhidos.....	127
8.7 - Comentários a respeito do método escolhido para a síntese.....	128
9. Conclusões.....	130
10. Temas de pesquisa futura.....	132
11. Apêndice.....	133
11.1 - Listagem do programa SYN 3 (versão 5).....	133
11.2 - Listagem do programa SYN 4 (versão 5).....	142
11.3 - Listagem do programa POS.....	150
11.4 - Listagem do programa RETA para animação do mecanismo.....	159
11.5 - Listagem do programa de otimização e suas sub-rotinas.....	165
11.6 - Saída da sub-rotina OPT3 para otimização do ângulo de transmissão na síntese para 4 po- sições e 13 variáveis.....	185
12. Referências bibliográficas.....	192

NOTAÇÃO

a letra minúscula representa um escalar ou módulo de um vetor.

a letra sublinhada representa um vetor.

α_j rotação angular do acoplador da posição 1 para a posição j.

β_j rotação angular do elemento conectado à estrutura da posição 1 para a posição j.

w vetor do elemento conectado à estrutura na 1a. posição.

z vetor do acoplador na 1a. posição.

δ_j vetor-deslocamento do ponto P da posição 1 para a posição j.

R_j vetor-posição que localiza a posição j com respeito a um eixo de coordenadas x-y fixo na estrutura.

A_o , B_o articulações fixas.

A , B articulações móveis.

P ponto de interesse do acoplador .

μ ângulo de transmissão.

exp(x) ... função exponencial de x.

ÍNDICE DE FIGURAS

Figura	Título	Página
2.1	Evolução histórica do projeto de mecanismos auxiliado por computador	6
3.1	Quadrilátero articulado.....	10
3.2	Geração de função.....	12
3.3	Geração de trajetória.....	12
3.4	Mecanismo de um projetor de filmes.....	13
3.5	Mecanismo carregador.....	14
3.6	Construção geométrica para a obtenção dos pontos-centro para duas posições de precisão.....	15
3.7	Ângulo de transmissão μ e detalhe dos vetores de força e velocidade no ponto B.....	16
3.8	Notação complexa para representação das barras articuladas.....	18
3.9	Deslocamento do ponto P de P_i para P_j	18
3.10	Quadrilátero articulado.....	23
3.11	Triângulo de solução do sistema não-linear.....	26
5.1	Ângulo de transmissão no quadrilátero articulado.....	35
5.2	Vizinhança de precisão para uma posição do ponto do acoplador.....	41
5.3	Vizinhança de precisão para uma especificação de rotação.....	42

6.1	Posições de precisão do ponto P e ângulos de rotação do acoplador.....	45
6.2	Posições de precisão do ponto P e ângulos de rotação do acoplador.....	48
6.3	Posições de precisão do ponto P e ângulos de rotação do acoplador.....	51
7.1	Mecanismo obtido conforme tabela 7.3.....	56
7.2	Posições de precisão do mecanismo obtido.....	57
7.3	Mecanismo obtido conforme tabela 7.10.....	60
7.4	Posições de precisão do mecanismo obtido.....	61
7.5	Mecanismo obtido conforme tabela 7.17.....	64
7.6	Posições de precisão do mecanismo obtido.....	65
7.7	Gráfico do ângulo de transmissão μ , série B, em função do ângulo β	69
7.8	Gráfico do ângulo de transmissão μ , série C, em função do ângulo β	70
7.9	Gráfico do ângulo de transmissão μ , série D, em função do ângulo β	71
7.10	Gráfico comparativo do ângulo de transmissão das séries B, C e D em função de β	72
7.11	Gráfico comparativo do desvio do ângulo de transmissão das séries B, C e D em função de β	74
7.12	Mecanismo obtido conforme tabela 7.23.....	77
7.13	Posições de precisão do mecanismo obtido.....	78
7.14	Mecanismo obtido conforme tabela 7.30.....	81
7.15	Posições de precisão do mecanismo obtido.....	82
7.16	Mecanismo obtido conforme tabela 7.37.....	86

7.17	Posições de precisão do mecanismo obtido.....	87
7.18	Gráfico do ângulo de transmissão μ , série B, em função do ângulo β	90
7.19	Gráfico do ângulo de transmissão μ , série C, em função do ângulo β	91
7.20	Gráfico do ângulo de transmissão μ , série D, em função do ângulo β	92
7.21	Gráfico comparativo do ângulo de transmissão das séries B, C e D em função de β	93
7.22	Gráfico comparativo do desvio do ângulo de transmissão das séries B, C e D em função de β	95
7.23	Mecanismo obtido conforme tabela 7.43.....	97
7.24	Posições de precisão do mecanismo obtido.....	98
7.25	Mecanismo obtido conforme tabela 7.50.....	102
7.26	Posições de precisão do mecanismo obtido.....	103
7.27	Gráfico do ângulo de transmissão μ , série B, em função do ângulo β	106
7.28	Gráfico do ângulo de transmissão μ , série C, em função do ângulo β	107
7.29	Gráfico comparativo do ângulo de transmissão das séries B e C em função de β	108
7.30	Gráfico comparativo do desvio do ângulo de transmissão das séries B e C em função de β	111
8.1	Mecanismo de 6 barras.....	118
8.2	Círculo trigonométrico.....	122
8.3	Quadrilátero articulado do tipo "retângulo".....	123
8.4	Gráfico de $\cos \mu \times \phi$	124

8.5	Gráfico de $\mu \times \cos \mu$	124
8.6	Gráfico de $\mu \times \phi$	125
8.7	Gráfico do desvio de μ em torno de 90° $\times \phi$	125
8.8	Gráfico do desvio quadrático de $\mu \times \phi$	126
8.9	Posições dos pontos para o cálculo da função objetivo no gráfico do desvio quadrático de $\mu \times \phi$	127

ÍNDICE DE TABELAS

Tabela	Título	Página
3.1	Número máximo de posições de precisão para a diada tendo como valores desconhecidos os vetores w e z quando δ_j e α_j são fixados previamente na equação (3.2).....	20
6.1	Posições do ponto P do acoplador.....	44
6.2	Ângulos de rotação do acoplador.....	44
6.3	Variáveis livres(articulações fixas).....	44
6.4	Posições do ponto P do acoplador.....	46
6.5	Ângulos de rotação do acoplador.....	47
6.6	Ângulos de rotação das alavancas da 1a. e 2a. diadas.....	47
6.7	Posições do ponto P do acoplador.....	50
6.8	Ângulos de rotação do acoplador.....	50
6.9	Variáveis livres(articulações fixas).....	50
7.1	Posições do ponto P do acoplador.....	54
7.2	Ângulos de rotação do acoplador.....	54
7.3	Posições do ponto P do acoplador.....	55
7.4	Ângulos de rotação do acoplador.....	55
7.5	Valor da função na 1a. e última iteração.....	58
7.6	Posições do ponto P do acoplador.....	58
7.7	Erro nas posições do ponto P do acoplador.....	58

7.8	Ângulos de rotação do acoplador.....	59
7.9	Erro nos ângulos de rotação do acoplador.....	59
7.10	Articulações fixas e móveis e ponto P do acoplador.....	59
7.11	Ângulos de rotação da alavanca.....	62
7.12	Valor da função na 1a. e última iteração.....	62
7.13	Posições do ponto P do acoplador.....	63
7.14	Erro nas posições do ponto P do acoplador.....	63
7.15	Ângulos de rotação do acoplador.....	63
7.16	Erro nos ângulos de rotação do acoplador.....	66
7.17	Articulações fixas e móveis e ponto P do acoplador.....	66
7.18	Ângulos de rotação da alavanca.....	66
7.19	Ângulos de transmissão $\times \beta$ para síntese sem otimização (B), com optimização de 4 variáveis (C), com optimização de 12 variáveis (D).....	68
7.20	Desvio do ângulo de transmissão $\times \beta$ para síntese sem optimização (B), com optimização de 4 variáveis (C), com optimização de 12 variáveis (D).....	73
7.21	Posições do ponto P do acoplador.....	75
7.22	Ângulos de rotação do acoplador.....	75
7.23	Articulações fixas e móveis e ponto	

P do acoplador.....	76
7.24 Ângulos de rotação da alavanca.....	76
7.25 Valor da função na 1a. e última iteração.....	79
7.26 Posições do ponto P do acoplador.....	79
7.27 Posições do ponto P do acoplador.....	79
7.28 Ângulos de rotação do acoplador.....	80
7.29 Erro nos ângulos de rotação do acoplador.....	80
7.30 Articulações fixas e móveis e ponto P do acoplador.....	80
7.31 Ângulos de rotação da alavanca.....	83
7.32 Valor da função na 1a. e última iteração.....	83
7.33 Posições do ponto P do acoplador.....	84
7.34 Posições do ponto P do acoplador.....	84
7.35 Ângulos de rotação do acoplador.....	84
7.36 Erro nos ângulos de rotação do acoplador.....	85
7.37 Articulações fixas e móveis e ponto P do acoplador.....	85
7.38 Ângulos de rotação da alavanca.....	85
7.39 Ângulos de transmissão x β para síntese sem otimização (B), com otimização de 2 variáveis (C), com otimização de 13 variáveis (D).....	89
7.40 Desvio do ângulo de transmissão x β para	

síntese sem otimização (B), com otimização	
de 2 variáveis (C), com otimização de 13	
variáveis (D).....	94
7.41 Posições do ponto P do acoplador.....	96
7.42 Ângulos de rotação do acoplador.....	96
7.43 Ângulos de rotação da alavanca.....	96
7.44 Articulações fixas e móveis e ponto	
P do acoplador.....	99
7.45 Valor da função na 1a. e última iteração.....	99
7.46 Posições do ponto P do acoplador.....	100
7.47 Erro nas posições do ponto P do	
acoplador.....	100
7.48 Ângulo de rotação do acoplador.....	100
7.49 Erro nos ângulos de rotação do	
acoplador.....	101
7.50 Articulações fixas e móveis e ponto	
P do acoplador.....	101
7.51 Ângulos de rotação da alavanca.....	101
7.52 Ângulos de transmissão $\times \beta$ para síntese com	
otimização de 4 variáveis (B) e sem	
otimização.....	104
7.53 Ângulos de transmissão $\times \beta$ para síntese com	
otimização de 4 variáveis (B) e sem	
otimização.....	105
7.54 Desvio do ângulo de transmissão $\times \beta$ para	
síntese com otimização de 4 variáveis (B)	
e sem otimização (C).....	109

7.55	Desvio do ângulo de transmissão x β para síntese com otimização de 4 variáveis (B) e sem otimização (C).....	109
------	--	-----

1. INTRODUÇÃO

Define-se [1] mecanismo em projeto mecânico como um dispositivo transformador de movimento, onde dado um movimento de entrada conhecido o mecanismo deve fornecer um movimento de saída desejado. Em geral, o termo mecanismo se refere a mecanismos de barras, engrenagens e camos.

A importância do estudo de mecanismos reside no fato deles serem elementos básicos de qualquer máquina ou equipamento. As aplicações dos mecanismos nos setores industrial e agrícola são inúmeras. Dentre elas, poderiam ser citadas automação industrial, máquinas de levantamento e transporte , máquinas e implementos agrícolas. Mais recentemente, a teoria de mecanismos vem sendo também utilizada para projeto de robôs onde se pode realizar simulação tri-dimensional, estimativa da região de trabalho e análise da resposta dinâmica. Outra área de aplicação seria a biomecânica onde o projeto de articulações e membros artificiais seriam alguns exemplos.

O mecanismo em estudo neste trabalho é um mecanismo plano de 4 barras articuladas mais conhecido como quadrilátero articulado. Neste texto são abordados dois aspectos importantes dentro do projeto desse tipo de mecanismo. Um mais básico, que seria a síntese e outro mais avançado que seria a otimização de um de seus parâmetros, o ângulo de transmissão.

2. REVISÃO BIBLIOGRÁFICA

Nesta revisão bibliográfica, serão abordados os seguintes tópicos :

- Projeto de mecanismos auxiliado por computador e
- Otimização no projeto de mecanismos,

sendo que o período considerado começa na década de 50 e vai até os tempos atuais . Para o desenvolvimento destes assuntos foram tomadas como base as referências [2, 3, 4].

2.1 - Projeto de mecanismos auxiliado por computador

2.1.1 - Década de 50

A maioria das pesquisas para o desenvolvimento de programas de computador para o projeto de mecanismos se originou nas universidades americanas. As equipes que as desenvolveram eram lideradas por Hall na Universidade de Purdue, McLarnan na Universidade Estadual de Ohio, Shigley na Universidade de Michigan, Freudenstein na Universidade de Columbia e Hartenberg e Denavit na Universidade de Northwestern.

O primeiro trabalho que realmente utilizou o computador para síntese de mecanismos de barras foi o de Freudenstein e Sandor [5] em 1959. A solução que apresentaram era baseada na

teoria desenvolvida por Burmester em 1876.

2.1.2 - Década de 60

Nesta época, o computador se tornou um pouco mais acessível aos pesquisadores nas universidades. Assim, alguns problemas de síntese puderam ser resolvidos com a abordagem das posições de precisão. A idéia desta abordagem é que na maioria dos casos de síntese para inúmeras posições pode-se simplificar o problema escolhendo-se algumas posições discretas, por exemplo, três, quatro ou cinco.

Suh e Radcliffe [6] desenvolveram um trabalho para a síntese de mecanismos planos que utilizava técnicas de matrizes de deslocamentos.

2.1.3 - Década de 70

Neste período, houve um grande crescimento na utilização de computadores para o projeto de mecanismos.

Merecem destaque os programas IMP [7], desenvolvido pela Universidade de Wisconsin, KINSYN [8, 9], desenvolvido pelo M.I.T. e o LINCAGES [10], desenvolvido pela Universidade de Minnesota, sendo os dois últimos baseados na teoria de Burmester. O IMP era um programa para a análise cinemática, estática e dinâmica de, em princípio, qualquer tipo de mecanismo. Tanto o KINSYN como o LINCAGES, realizavam a síntese e análise de quaisquer mecanismos planos com juntas articuladas ou deslizantes.

Contudo, estes programas ainda só funcionavam em computadores de grande porte.

Buscou-se nesta década, pela primeira vez, a utilização de saída gráfica no computador para uma melhor e mais rápida visualização dos resultados e uma mais rápida interação homem-computador.

2.1.4 - Década de 80

Em 1984, Whalen e Midha [11] apresentaram um programa de computador para o projeto de mecanismos de 4 , 6 e 8 barras. Contudo, o programa era muito iterativo porque, segundo o texto, se baseia exclusivamente na análise, ou seja, o projetista deve atribuir valores para o comprimento das barras e o programa calcular os resultados. O projetista deve então verificar se aquele mecanismo atende às suas especificações.

Em 1985, Midha e Zhao [12] desenvolveram um método para síntese de mecanismos planos utilizando equações de malha fechada e servindo-se do método de Newton-Raphson para solução de sistemas de equações não-lineares.

O aparecimento dos micro-computadores nessa época facilitou incrivelmente a pesquisa para o projeto de mecanismos. Um dos exemplos foi o trabalho de Peterson, Logan, Erdman e Riley [13], que apresentaram, em 1988, uma versão do LINCAGES para micros da linha IBM PC XT/AT. Outros exemplos foram o MICRAS [14] e os programas de Norton [15]. A filosofia destes programas é

tornar a sua utilização a mais simples possível para o projetista.

Na figura 2.1, é apresentado um quadro que resume o desenvolvimento do projeto de mecanismos auxiliado por computador nos últimos 40 anos.

2.2 - Otimização no projeto de mecanismos

2.2.1 - Década de 60

É neste período que surgem os primeiros trabalhos que utilizaram técnicas de otimização no projeto de mecanismos.

Han [4], em 1966, sugeriu um método geral para otimização em mecanismos. Ele definiu uma função objetivo que representava o erro quadrático mínimo de cada variável de projeto. Assim eram tomados os valores das derivadas parciais da função com respeito às variáveis de projeto, igualadas a zero, e o conjunto resultante de equações não-lineares era resolvido usando o método de Newton. O resultado desta otimização era um mecanismo ótimo do ponto de vista do erro quadrático mínimo.

2.2.2 - Década de 70

Himmelblau [16], em 1972, apresentou no seu livro uma revisão dos principais métodos de otimização para solução de problemas de programação não-linear. Apresentou também uma comparação entre os métodos mais utilizados e concluiu que o Método Generalizado do Gradiente Reduzido (GRG, Generalized

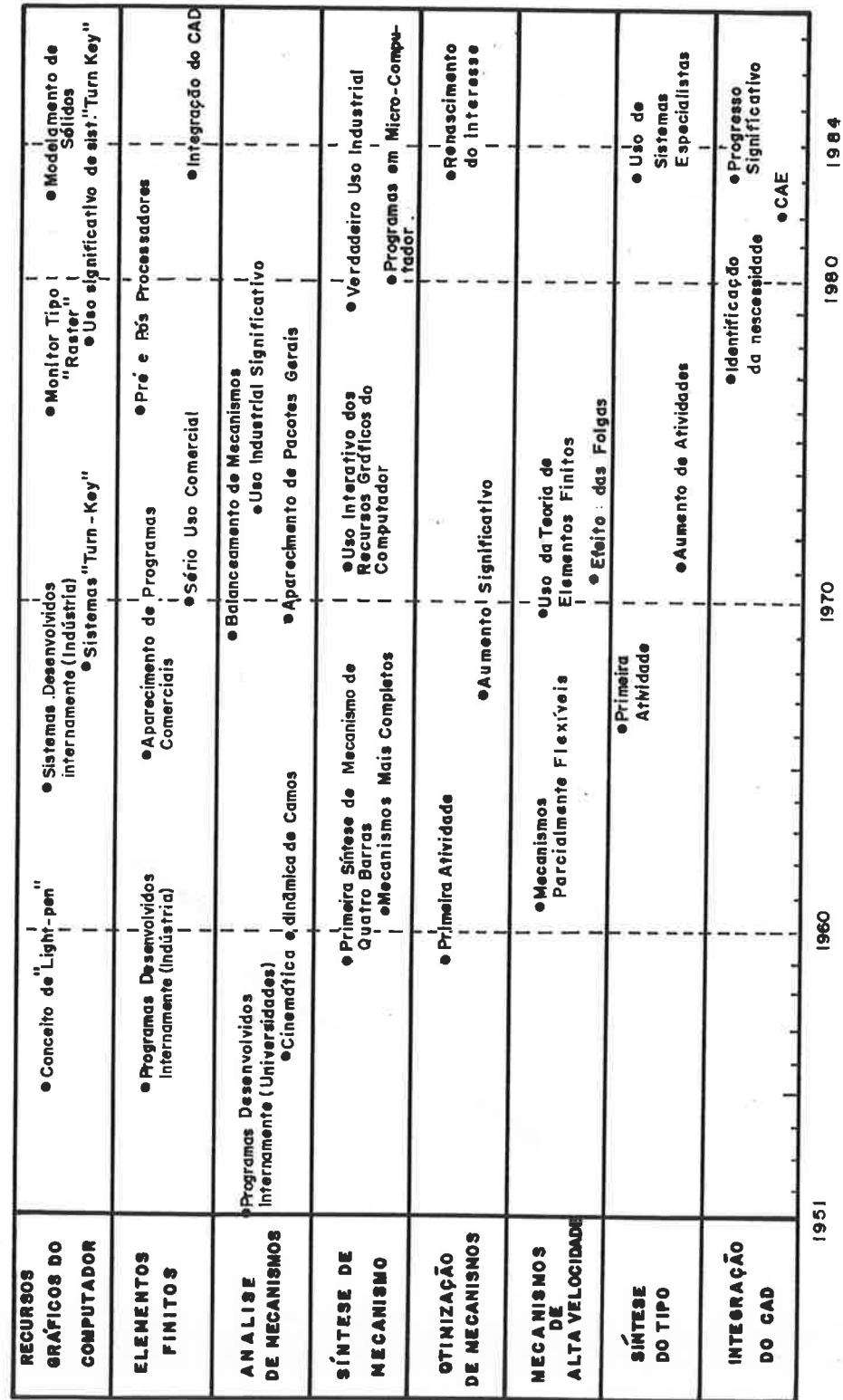


Fig. 2.1 - Evolução histórica do projeto de mecanismos auxiliado por computador [2].

Reduced Gradient method) era o mais eficiente.

Em 1975, Kramer e Sandor [17] apresentaram um método para projeto ótimo de mecanismos planos para geração de trajetória, movimento e função com limites de precisão diferentes nas posições discretas de projeto. Usaram técnicas de programação não-linear para a solução.

Em 1977, Gabriele e Ragsdell [18] apresentaram o GRG como uma ferramenta eficiente e confiável para qualquer otimização em projeto mecânico.

Ainda em 1977, Gupta [19] apresentou um método analítico para otimização do ângulo de transmissão para projeto de geradores de função em quadriláteros articulados. Foi estabelecido um desvio do ângulo de transmissão em torno de 90° , surgindo dois valores extremos, mínimo e máximo, que foram admitidos igualmente distanciados de 90° . Desta imposição surgiram relações que somadas às relações das posições de precisão do mecanismo gerou-se curvas que representavam o lugar geométrico do ótimo ângulo de transmissão.

Em 1978, Suh e Radcliffe [20] apresentaram no seu livro tópicos avançados a respeito dos métodos de otimização disponíveis para o projeto de mecanismos planos e espaciais.

2.2.3 - Década de 80

Em 1985, Avilés, Ajuria e Jalón [21] apresentaram um

método muito geral para otimização na síntese de mecanismos. O método utilizava uma função objetivo para qualquer tipo de mecanismo e para qualquer tipo de síntese, podendo até acoplar diferentes objetivos para síntese.

Também em 1985, Lima [22] apresentou um trabalho sobre otimização do comportamento dinâmico de mecanismos de barras com um único grau de liberdade, com movimento e geometria conhecidos, com o uso de computador.

Em 1988, Rigelman e Kramer [23] apresentaram um método para otimização na síntese de quadriláteros articulados satisfazendo determinadas condições cinemáticas e dinâmicas. O método poderia ser utilizado para geração de trajetória, movimento e função. As condições cinemáticas eram a posição, a velocidade e a aceleração de um ponto do acoplador e as rotações do acoplador e de elementos do seguidor. As condições dinâmicas se referiam à minimização da potência média consumida pelo mecanismo bem como o estabelecimento de um limite para o torque máximo de entrada.

Percebe-se, atualmente, que a tendência para pesquisa nesta área é o desenvolvimento de "sistemas especialistas" para o projeto de mecanismos, prevendo otimização de um grande número de parâmetros de projeto e, ao mesmo tempo, com saídas gráficas cada vez melhores para mais fácil visualização das soluções possíveis.

3. SOLUÇÃO ANALÍTICA PARA A SÍNTSE

3.1 - Introdução

Neste capítulo serão apresentados alguns fundamentos sobre projeto de mecanismos, as hipóteses básicas para a solução do problema da síntese, a equação da diada, a síntese para diferentes tarefas como geração de movimento, de trajetória e de função. Por fim, são desenvolvidos os procedimentos de solução para geração de movimento para 3 e 4 posições de precisão.

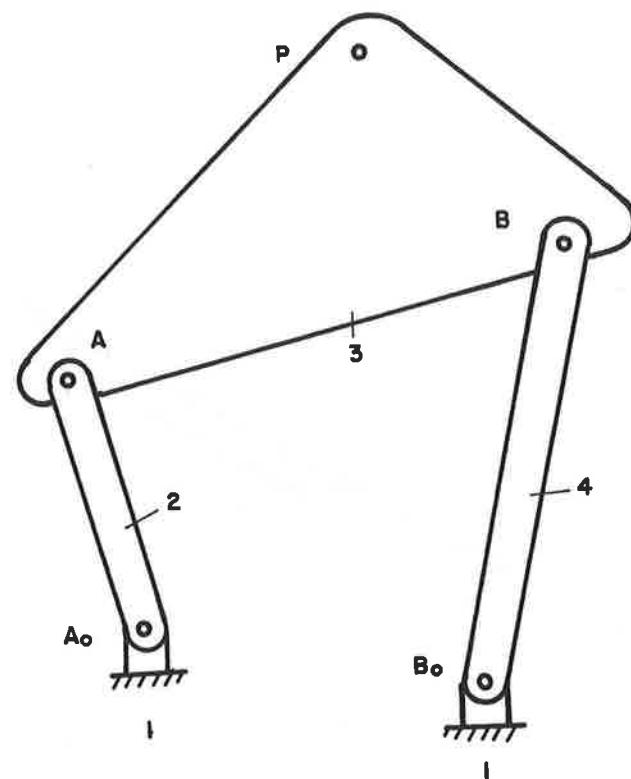
3.2 - Alguns fundamentos sobre mecanismos [24]

Os mecanismos podem ser planos ou espaciais. Este trabalho trata especificamente de um tipo de mecanismo plano de barras articuladas chamado quadrilátero articulado, que é composto de quatro barras e quatro articulações.

Dentre os motivos que poderiam ser citados para o estudo do quadrilátero articulado estão :

- é um mecanismo largamente utilizado,
- é o mecanismo plano mais simples possível,
- muitos mecanismos complexos possuem os quadriláteros articulados como elementos; desta forma, a teoria do mecanismo de quatro barras é útil no projeto destes mecanismos.

O estudo dos mecanismos envolve basicamente dois tópicos, um chamado análise e o outro síntese. A análise trata do estudo cinemático e dinâmico de um dado mecanismo. Nessa área, existe o interesse em determinar as posições, as velocidades, as acelerações e as forças nos elementos que compõem um mecanismo conhecido. Por outro lado, a síntese estuda o problema contrário: dado um movimento, o que se deseja é encontrar o mecanismo que realize esse movimento.



Barra 1 = Elemento Fijo

Barra 2 = Alavanca Motora

Barra 3 = Acoplador

Barra 4 = Alavanca Movida

Fig 3.1 - Quadrilátero Articulado.

As áreas da síntese podem ser agrupadas em duas categorias :

- Síntese do tipo de mecanismo : Dado um desempenho

desejado para uma determinada aplicação, as perguntas que se fazem são as seguintes: qual o tipo de mecanismo mais conveniente ? Barras articuladas ? Camos ? Engrenamentos de rodas dentadas ? Outros temas abordados pela síntese são o número de elementos e o número de graus de liberdade necessário para o mecanismo.

- Síntese Dimensional : Trata da determinação das dimensões mais significativas e da posição inicial de um mecanismo do tipo pré-concebido para realizar uma dada tarefa com um desempenho desejado.

As tarefas mais comuns da síntese são geração de função, trajetória e movimento.

3.2.1 - Geração de função

Relaciona os movimentos de deslimento ou rotação dos elementos de entrada e saída.

3.2.2 - Geração de trajetória

Neste caso, toma-se um ponto fixo num "elemento flutuante" (não diretamente conectado ao elemento fixo) para que realize uma trajetória definida em relação à estrutura fixa de referência. Se os pontos da trajetória estão relacionados com o tempo ou posições do elemento de entrada, a tarefa é dita geração de trajetória com temporização.

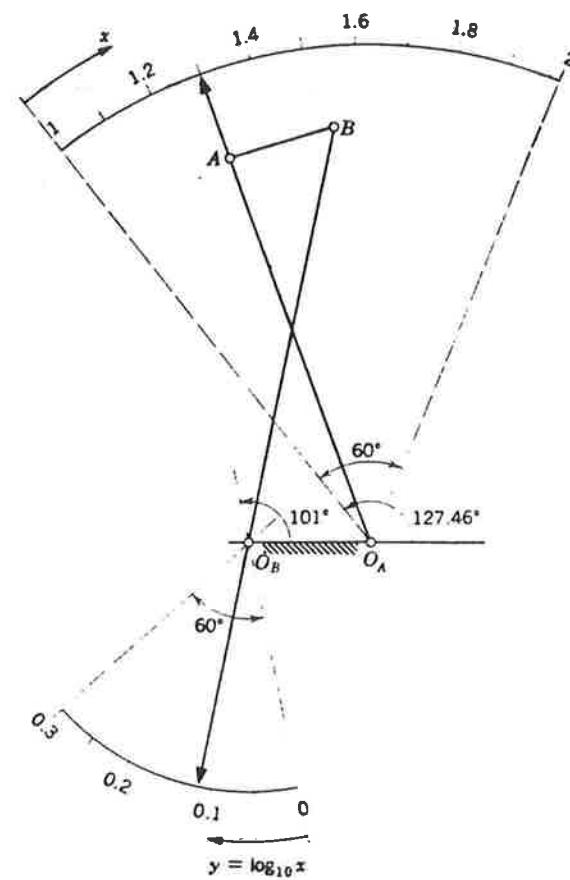


Fig. 3.2 - Geração de função [1].

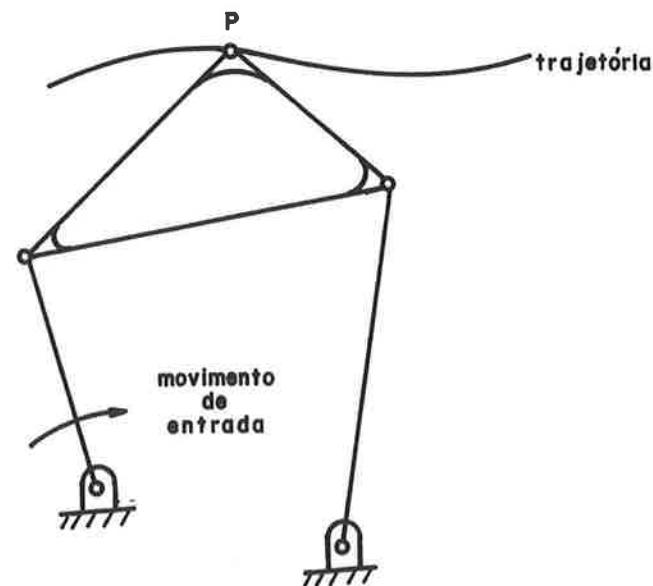


Fig. 3.3 - Geração de Trajetória .

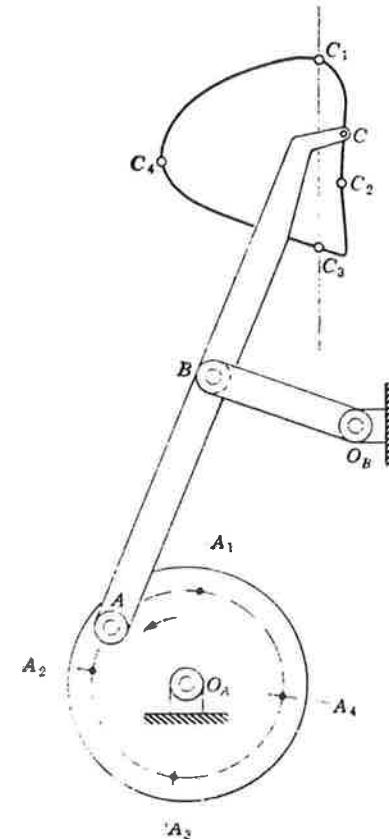


Fig. 3.4 - Mecanismo de um projetor de filmes [1].

3.2.3 - Geração de movimento

Nesta tarefa, um corpo deve ser guiado para realizar um movimento pré-fixado. Isto pode ser visualizado na figura 3.5 .

A solução para os problemas de síntese dimensional pode ser feita graficamente ou analiticamente. Ambas soluções , gráfica e analítica, se baseiam na teoria de Burmester. Porém, a solução analítica é superior à gráfica porque, desde que se disponha de um computador, ela é mais rápida, mais precisa e mais fácil de se manipular.

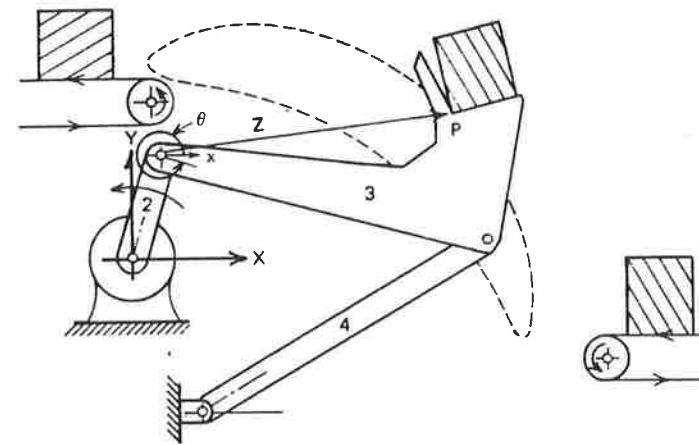


Fig. 3.5 - Mecanismo Carregador [24].

3.2.4 - Algumas noções da teoria de Burmester [25]

Demonstra-se que um conjunto arbitrário de cinco posições de um elemento rígido π , em movimento plano em relação a um sistema de referência E_0 , determina um conjunto de (até quatro) pontos M em E_0 , de tal maneira que cada ponto é o centro de uma circunferência que passa sobre as cinco correspondentes posições de um ponto particular K do elemento π . Os pontos M são chamados pontos-centro ou pontos de Burmester, em homenagem a L. Burmester, que deduziu sua existência a partir de considerações envolvendo geometria projetiva. Os pontos K são chamados pontos-sobre-a-circunferência.

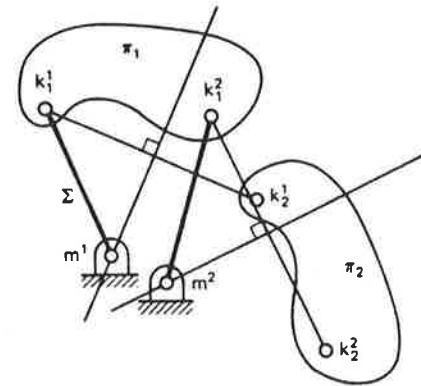


Fig. 3.6 - Construção geométrica para a obtenção dos pontos-centro para duas posições de precisão.

O significado prático dos pontos de Burmester nos mecanismos articulados planos vem do fato de que o movimento relativo de qualquer par de elementos adjacentes é circular. Os pontos de Burmester representam os centros fixos das alavancas do mecanismo. A definição de cada alavanca do mecanismo se dá pela união entre um ponto M e o correspondente ponto K de π . As alavancas guiam o elemento π que se move ao longo das posições de precisão.

3.2.5 - Ângulo de transmissão [1, 24, 26]

Este parâmetro surgiu como critério para indicar quanto bem um dado mecanismo funcionará enquanto ele ainda estiver na prancheta de projeto. O termo "funcionar" significa a eficiência com que um dado movimento é transferido para o elemento de saída (movido); resulta numa operação suave, em que a maior componente da força é utilizada para produzir torque ou força no elemento de saída.

Alt definiu o ângulo de transmissão como sendo o menor ângulo entre as direções dos vetores diferença de velocidades \underline{v}_{BA} do elemento flutuante e velocidade absoluta \underline{v}_B do elemento de saída tomados no ponto da articulação. Como pode ser visto na figura 3.7, este ângulo coincide com aquele formado pelas linhas de centro do acoplador e do seguidor.

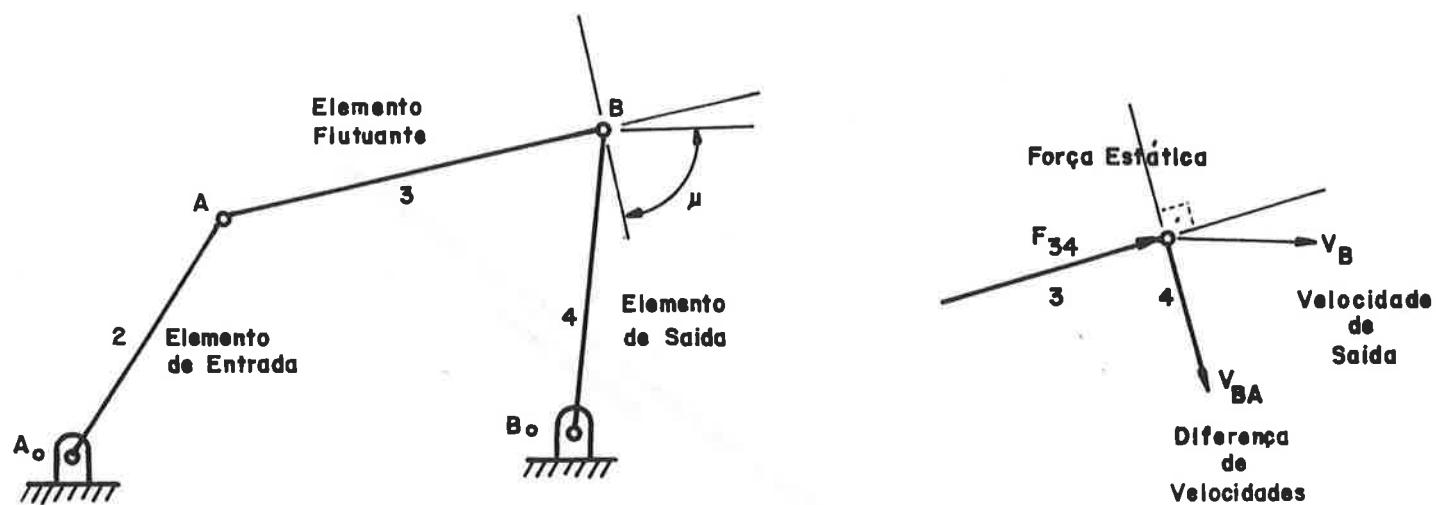


Fig. 3.7 - Ângulo de transmissão μ e detalhe dos vetores de força e velocidade no ponto B.

A importância do ângulo de transmissão pode ser resumida nos seguintes itens abaixo :

- a) A força de transmissão que passa do acoplador para o seguidor é mais efetiva quando o ângulo de transmissão é 90° . É claro que é desejável que este ângulo tenha o menor desvio possível em torno de 90° durante a operação do mecanismo. Recomenda-se, na prática, que a tolerância deste ângulo fique na

faixa de $\pm 45^\circ$, ou seja, valor mínimo 45° e valor máximo 135° .

b) Em algumas máquinas de precisão, quando o ângulo de transmissão é pequeno demais, a precisão do movimento de saída torna-se mais sensível a tolerâncias de fabricação no comprimento dos elementos e folga entre as juntas. Além disso, um ângulo de transmissão baixo pode causar ruído e solavancos indesejáveis em alta velocidade.

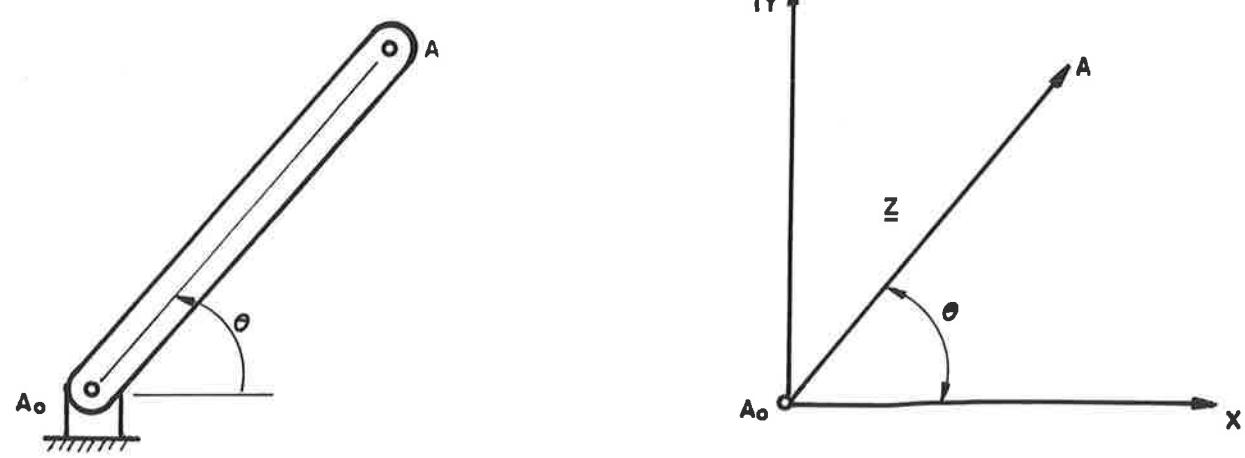
3.3 - Modelamento das barras articuladas

As barras articuladas serão consideradas elementos rígidos e representadas por vetores seguindo a notação complexa.

3.4 - Equação da Diada [10, 13, 25, 27]

A grande maioria dos mecanismos articulados planos pode ser imaginada como combinações de pares de vetores chamados diadas. Por exemplo, o quadrilátero articulado da figura 3.1 pode ser considerado como formado por duas diadas : uma diada do lado esquerdo e outra do lado direito.

Admite-se que um ponto do acoplador chamado P é deslocado de sua posição inicial P_i para uma dada posição P_j .



$$\underline{z} = (A - A_0) = z \cdot (\cos \theta + i \cdot \sin \theta) = z \cdot \exp(i\theta)$$

Fig. 3.8 - Notação complexa para representação das barras articuladas.

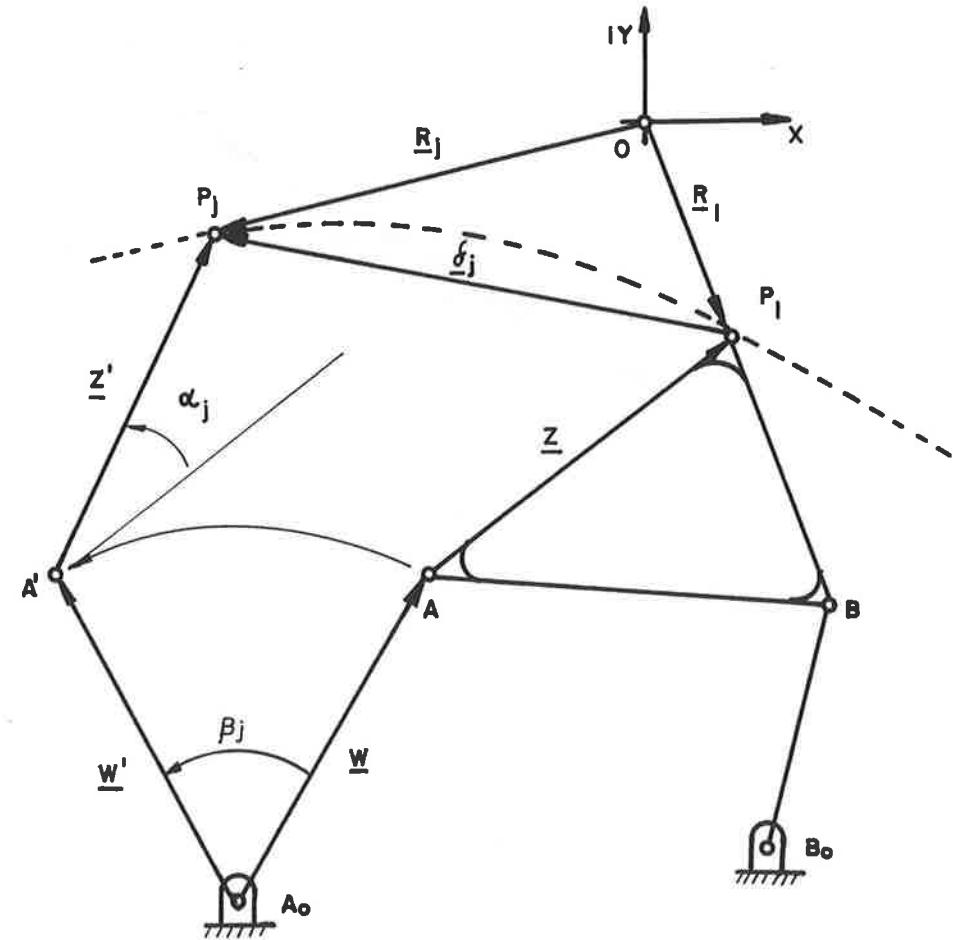


Fig 3.9 - Deslocamento do ponto P de P_1 para P_j .

Sendo OXY o sistema de referência adotado, \underline{R}_i o vetor-posição do ponto P_i , \underline{R}_j o vetor-posição do ponto P_j , a seguinte soma vetorial pode ser realizada :

$$(A - A_o) + (P_i - A) = \underline{R}_i + \underline{R}_j - (P_j - A') = (A' - A_o) = \underline{0}$$

sendo $(A - A_o) = \underline{w}$, $(P_i - A) = \underline{z}$, $(P_j - A') = \underline{z}'$ e

$$(A' - A_o) = \underline{w}', \text{ então}$$

$$\underline{w} + \underline{z} - \underline{R}_i + \underline{R}_j - \underline{z}' - \underline{w}' = \underline{0} \quad (3.1)$$

Sendo β_j a rotação de \underline{w} e α_j a rotação de \underline{z} temos :

$$\underline{w}' = \underline{w} \cdot \exp(i\beta_j) \quad \underline{z}' = \underline{z} \cdot \exp(i\alpha_j)$$

Substituindo \underline{w}' e \underline{z}' em (3.1) vem :

$$\underline{w} + \underline{z} - \underline{R}_i + \underline{R}_j - \underline{z} \cdot \exp(i\alpha_j) - \underline{w} \cdot \exp(i\beta_j) = \underline{0}$$

$$\underline{w} \cdot \exp(i\beta_j) - 1 + \underline{z} \cdot \exp(i\alpha_j) - 1 = \underline{R}_j - \underline{R}_i.$$

$$\text{Como } \underline{R}_j - \underline{R}_i = \delta_j \quad \text{então}$$

$$\underline{w} \cdot \exp(i\beta_j) - 1 + \underline{z} \cdot \exp(i\alpha_j) - 1 = \delta_j \quad (3.2)$$

A equação (3.2) é conhecida como equação da diáda.

3.5 - Os sistemas de equações para solução dos problemas de síntese para algumas posições de precisão

Tendo presente a equação (3.2) é possível construir-se a tabela 3.1 para diferentes posições do ponto P.

Tabela 3.1 - Número máximo de posições de precisão para a diâada, tendo como valores desconhecidos os vetores \underline{w} e \underline{z} quando $\underline{\delta}_j$ e α_j são fixados previamente na equação (3.2).

n.º de posições (n) $j = 2, \dots, n$	n.º de equações escalares	n.º de escalares desconhecidos	n.º de escalares escolhidos livremente	n.º de soluções
2	2	5C $\underline{w}, \underline{z} \in \beta_2$)	3	$\sigma(\infty^3)$
3	4	6C os acima e β_3)	2	$\sigma(\infty^2)$
4	6	7C os acima e β_4)	1	$\sigma(\infty^1)$
5	8	8C os acima e β_5)	0	finito

A seguir serão apresentados os sistemas de equações para a síntese de duas, três, quatro e cinco posições de precisão.

3.5.1 - Duas posições

Tomam-se três escalares, escolhidos arbitrariamente : $\underline{z} \in \beta_2$ (por exemplo), sendo o sistema linear.

$$\underline{w} = \frac{\underline{\delta}_z - \underline{z} \cdot (\exp(i\alpha_z) - 1)}{\exp(i\beta_z) - 1}$$

3.5.2 - Três Posições

Tomam-se dois escalares, escolhidos arbitrariamente.

Alguns exemplos seriam :

- β_2 e β_3 (sistema linear) ou
- w ou z (sistema não-linear).

$$\left\{ \begin{array}{l} \underline{w} \cdot (\exp(i\beta_z) - 1) + \underline{z} \cdot (\exp(i\alpha_z) - 1) = \underline{\delta}_z \\ \underline{w} \cdot (\exp(i\beta_a) - 1) + \underline{z} \cdot (\exp(i\alpha_a) - 1) = \underline{\delta}_a \end{array} \right. \quad (3.3)$$

3.5.3 - Quatro Posições

Toma-se um escalar escolhido livremente. Alguns exemplos seriam :

- β_2 ou $\arg(z)$ (sistema não-linear).

$$\left\{ \begin{array}{l} \underline{w} \cdot (\exp(i\beta_z) - 1) + \underline{z} \cdot (\exp(i\alpha_z) - 1) = \underline{\delta}_z \\ \underline{w} \cdot (\exp(i\beta_a) - 1) + \underline{z} \cdot (\exp(i\alpha_a) - 1) = \underline{\delta}_a \\ \underline{w} \cdot (\exp(i\beta_4) - 1) + \underline{z} \cdot (\exp(i\alpha_4) - 1) = \underline{\delta}_4 \end{array} \right. \quad (3.4)$$

3.5.4 - Cinco Posições

Neste caso não há variáveis livres e o sistema é não-linear.

$$\left\{ \begin{array}{l} \underline{w} \cdot (\exp(i\beta_2) - 1) + \underline{z} \cdot (\exp(i\alpha_2) - 1) = \underline{\delta}_2 \\ \underline{w} \cdot (\exp(i\beta_3) - 1) + \underline{z} \cdot (\exp(i\alpha_3) - 1) = \underline{\delta}_3 \\ \underline{w} \cdot (\exp(i\beta_4) - 1) + \underline{z} \cdot (\exp(i\alpha_4) - 1) = \underline{\delta}_4 \\ \underline{w} \cdot (\exp(i\beta_5) - 1) + \underline{z} \cdot (\exp(i\alpha_5) - 1) = \underline{\delta}_5 \end{array} \right. \quad (3.5)$$

3.6 - Geração de Movimento, de Trajetória e de Função

3.6.1 - Geração de Movimento

O quadrilátero articulado da figura 3.10 deve ser sintetizado para geração de movimento. Como sugerido na seção 3.4, há duas diádas independentes no quadrilátero articulado, uma chamada diada do lado esquerdo e outra do lado direito. Cada diada liga uma articulação fixa a um ponto do acoplador através de uma articulação móvel. A equação da diada do lado esquerdo já foi deduzida mas na notação da figura 3.3 ela fica na seguinte forma

$$\underline{z}_1 \cdot (\exp(i\phi_j) - 1) + \underline{z}_2 \cdot (\exp(i\gamma_j) - 1) = \underline{\delta}_j , \quad j = 2, \dots, 5. \quad (3.6)$$

onde $\underline{\delta}_j$ e γ_j são variáveis especificadas. A equação da diada do lado direito pode ser escrita na forma

$$\underline{Z}_3 \cdot (\exp(i\psi_j) - 1) + \underline{Z}_4 \cdot (\exp(i\gamma_j) - 1) = \underline{\delta}_j ,$$

$$j = 2, \dots, 5. \quad (3.7)$$

onde $\underline{\delta}_j$ e γ_j são especificados. Utilizando-se técnicas matemáticas para resolver este sistema de equações obtém-se \underline{Z}_1 , \underline{Z}_2 , \underline{Z}_3 e \underline{Z}_4 . Os outros dois vetores do quadrilátero articulado são simplesmente

$$\underline{Z}_5 = \underline{Z}_2 - \underline{Z}_4 \quad (3.8)$$

$$\underline{Z}_6 = \underline{Z}_1 + \underline{Z}_5 - \underline{Z}_3 \quad (3.9)$$

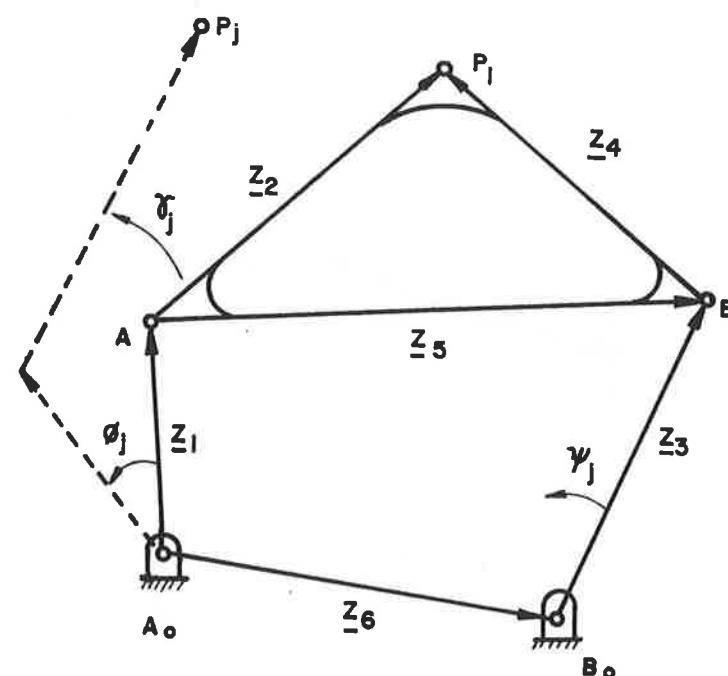


Fig. 3.10 - Quadrilátero articulado.

3.6.2 - Geração de Trajetória com temporização

Agora o quadrilátero articulado da figura 3.10 deve ser sintetizado para geração de trajetória com temporização. As mesmas

equações do item anterior se aplicarão a esse caso (3.6 a 3.9) mas os ângulos especificados serão diferentes. Neste caso ϕ_j é conhecido ao invés de γ_j . Contudo, o ângulo γ_j deve ser o mesmo nas equações (3.6) e (3.7). A partir daí, o procedimento é o mesmo do item anterior.

3.6.3 - Geração de Função

Neste caso, o que se pretende relacionar são os ângulos de rotação da alavanca da 1a. diada (ϕ_j) com os ângulos da alavanca da 2a. diada (ψ_j). Assim estes dois ângulos são especificados no projeto. E o procedimento é o mesmo dos itens anteriores.

3.7 - Procedimento de Solução para Três Posições de Precisão

Para solução deste caso considerar-se-á conhecido o vetor ($P_1 = A_0$), conforme figura 3.9, que será chamado de vetor \underline{R} . Assim, tem-se o seguinte sistema de equações cujas incógnitas seriam w , z , β_2 e β_3 :

$$\left\{ \begin{array}{l} w + z = - \underline{R} \\ w \cdot (\exp(i\beta_2) - 1) + z \cdot (\exp(i\alpha_2) - 1) = \underline{\delta}_2 \\ w \cdot (\exp(i\beta_3) - 1) + z \cdot (\exp(i\alpha_3) - 1) = \underline{\delta}_3 \end{array} \right.$$

Na forma matricial seria

$$\begin{bmatrix} 1 & 1 & \cdot \\ \exp(i\beta_2) - 1 & \exp(i\alpha_2) - 1 & w \\ \exp(i\beta_3) - 1 & \exp(i\alpha_3) - 1 & z \end{bmatrix} = \begin{bmatrix} -R \\ \delta_2 \\ \delta_3 \end{bmatrix}$$

Para que esse sistema seja possível e determinado, o seguinte determinante deve ser zero:

$$\begin{vmatrix} 1 & 1 & -R \\ \exp(i\beta_2) - 1 & \exp(i\alpha_2) - 1 & \delta_2 \\ \exp(i\beta_3) - 1 & \exp(i\alpha_3) - 1 & \delta_3 \end{vmatrix} = 0$$

Resolvendo esse determinante obtém-se o seguinte:

$$\delta_3 \cdot (\exp(i\alpha_2) - 1) - \delta_2 \cdot (\exp(i\alpha_3) - 1) - \delta_3 \cdot (\exp(i\beta_2) -$$

$$1) + \delta_2 \cdot (\exp(i\beta_3) - 1) - R \cdot [(\exp(i\beta_2) - 1) \cdot (\exp(i\alpha_3) -$$

$$1) - (\exp(i\alpha_2) - 1) \cdot (\exp(i\beta_3) - 1)] = 0$$

Chamando de \underline{x} , \underline{u} e \underline{v} os vetores

$$\underline{x} = -R \cdot (\exp(i\alpha_3) - 1) - \delta_3$$

$$\underline{u} = -R \cdot (\exp(i\alpha_2) - 1) + \delta_2$$

$$\underline{v} = -\delta_3 \cdot (\exp(i\alpha_2) - 1) + \delta_2 \cdot (\exp(i\alpha_3) - 1) -$$

$$\delta_3 + \delta_2 - R \cdot (\exp(i\alpha_3) - 1) - R \cdot (\exp(i\alpha_2) - 1)$$

Portanto,

$$x \cdot \exp(i\beta_2) + y \cdot \exp(i\beta_3) = k \quad (3.10)$$

Esta equação é não-linear nas incógnitas β_2 e β_3 . Ela representa um triângulo cujos lados são os módulos dos vetores $x \cdot \exp(i\beta_2)$, $y \cdot \exp(i\beta_3)$ e k .

Equacionando vem

$$\cos \gamma = \frac{x^2 + k^2 - y^2}{2 \cdot x \cdot k}$$

$$0 < \gamma < \pi, \quad \gamma = \arccos(\cos \gamma)$$

$$u^1 = \frac{k \cdot \exp(i \cdot \arg(k) + i \cdot \gamma)}{k} = \exp(i \cdot \arg(k) + i \cdot \gamma)$$

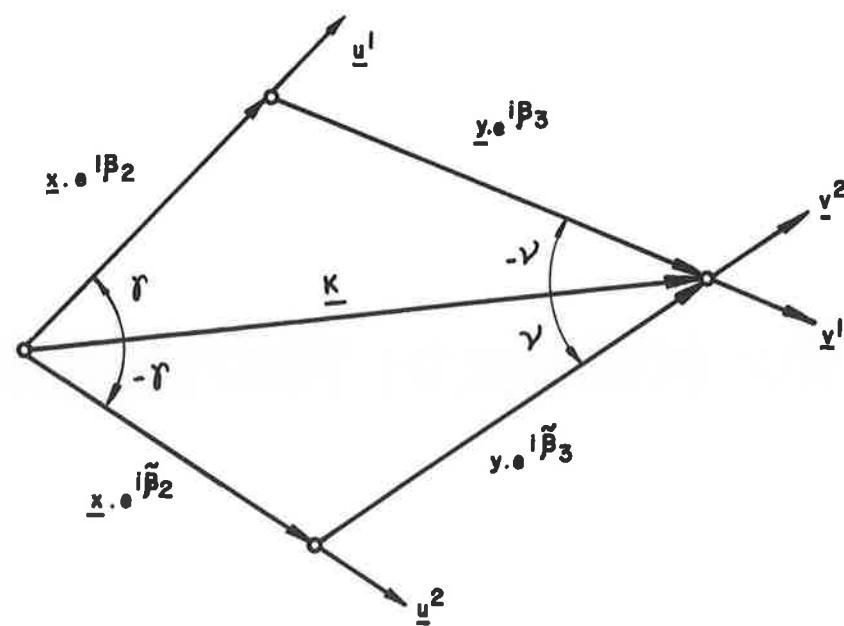


Fig. 3.11 - Triângulo de solução do sistema não-linear.

$$x \cdot \exp(i\beta_2 + i \arg(x)) = x \cdot \underline{u}^1 = x \cdot \exp(i \arg(\underline{k}) + i \gamma)$$

$$\beta_2 = \arg(\underline{k}) + \gamma - \arg(x)$$

$$\cos \nu = \frac{y^2 + k^2 - x^2}{2 \cdot y \cdot k}$$

$$0 < \nu < \pi, \quad \nu = \arccos(\cos \nu)$$

$$\underline{v}^1 = \frac{k \cdot \exp(i \arg(\underline{k}) - i \nu)}{k} = \exp(i \arg(\underline{k}) - i \nu)$$

$$y \cdot \exp(i\beta_3 + i \arg(y)) = y \cdot \underline{v}^1 = y \cdot \exp(i \arg(\underline{k}) - i \nu)$$

$$\beta_3 = \arg(\underline{k}) - \nu - \arg(y)$$

Analogamente,

$$\underline{u}^2 = \exp(i \arg(\underline{k}) - i \gamma) \Rightarrow$$

$$\Rightarrow \tilde{\beta}_2 = \arg(\underline{k}) - \gamma - \arg(x)$$

$$\underline{v}^2 = \exp(i \arg(\underline{k}) + i \nu) \Rightarrow$$

$$\Rightarrow \tilde{\beta}_3 = \arg(\underline{k}) + \nu - \arg(y)$$

Logo, para cada diada, duas soluções são possíveis :

uma com β_2 e β_3 e outra com $\tilde{\beta}_2$ e $\tilde{\beta}_3$. Assim, com as variáveis β_2 e β_3 conhecidas o sistema (3.3) fica linear e w e z ficam determinados.

Calculando-se w e z para a primeira e segunda diadas, ficam determinadas as articulações fixas A_o e B_o , e as articulações móveis, A e B para a primeira posição de precisão. Pode-se perceber que a solução deste sistema fornecerá quatro mecanismos que realizam a síntese com geração de movimento nas três posições de precisão.

3.8 - Procedimento de Solução para Quatro Posições de Precisão

O sistema de equações (3.4) pode ser colocado na forma matricial nas incógnitas w e z .

$$\begin{bmatrix} \exp(i\beta_2) - 1 & \exp(i\alpha_2) - 1 \\ \exp(i\beta_3) - 1 & \exp(i\alpha_3) - 1 \\ \exp(i\beta_4) - 1 & \exp(i\alpha_4) - 1 \end{bmatrix} \begin{bmatrix} w \\ z \end{bmatrix} = \begin{bmatrix} \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} \quad (3.11)$$

As variáveis conhecidas seriam α_2 , α_3 , α_4 , β_2 , δ_2 , δ_3 e δ_4 . Será admitido inicialmente que as únicas incógnitas sejam os vetores w e z . Assim, o sistema possuirá três equações e duas incógnitas w e z e, portanto, uma das equações é combinação linear das outras duas. Chamando M_i , matriz incompleta, a matriz dos coeficientes de w e z , e M , matriz completa, a matriz dos

coeficientes de w e \underline{z} incluindo os termos independentes δ_2 , δ_3 e δ_4 temos :

$$M_i = \begin{bmatrix} \exp(i\beta_2) - 1 & \exp(i\alpha_2) - 1 \\ \exp(i\beta_3) - 1 & \exp(i\alpha_3) - 1 \\ \exp(i\beta_4) - 1 & \exp(i\alpha_4) - 1 \end{bmatrix}$$

$$M = \begin{bmatrix} \exp(i\beta_2) - 1 & \exp(i\alpha_2) - 1 & \delta_2 \\ \exp(i\beta_3) - 1 & \exp(i\alpha_3) - 1 & \delta_3 \\ \exp(i\beta_4) - 1 & \exp(i\alpha_4) - 1 & \delta_4 \end{bmatrix}$$

A partir das matrizes acima, pode-se concluir que a característica p da matriz M_i assume o valor 2. Como o sistema só poderá ter solução se p for igual a q , sendo q a característica da matriz M , então q também deve ser igual a 2.

Assim, o determinante de M deve ser zero. Calculando-se esse determinante obtém-se

$$\Delta_2 \cdot \exp(i\beta_2) + \Delta_3 \cdot \exp(i\beta_3) + \Delta_4 \cdot \exp(i\beta_4) + \Delta_1 = 0$$

onde $\Delta_1 = -\Delta_2 - \Delta_3 - \Delta_4$

$$\Delta_2 = \begin{vmatrix} \exp(i\alpha_3) - 1 & \delta_3 \\ \exp(i\alpha_4) - 1 & \delta_4 \end{vmatrix}$$

$$\Delta_3 = - \begin{vmatrix} \exp(i\alpha_2) - 1 & \delta_2 \\ \exp(i\alpha_4) - 1 & \delta_4 \end{vmatrix}$$

$$\Delta_4 = \begin{vmatrix} \exp(i\alpha_2) - 1 & \delta_2 \\ \exp(i\alpha_3) - 1 & \delta_3 \end{vmatrix}$$

$$\underline{\Delta}_3 \cdot \exp(i\beta_3) + \underline{\Delta}_4 \cdot \exp(i\beta_4) = \underline{\Delta} \quad (3.12)$$

onde $\underline{\Delta} = -\underline{\Delta}_1 - \underline{\Delta}_2 \cdot \exp(i\beta_2)$

Então, adota-se para a variável β_2 um valor arbitrário.

Deve-se notar que a equação (3.12) é não-linear e do mesmo tipo da equação (3.10) para a síntese de 3 posições. Adotando-se o mesmo procedimento de solução determinamos β_3 e β_4 em função de β_2 .

Com β_2 determinado, determina-se β_3 , β_4 , $\tilde{\beta}_3$ e $\tilde{\beta}_4$.

Portanto, para cada diada, haverá duas soluções possíveis. Desta forma, obtém-se quatro mecanismos diferentes que realizam a geração de movimento para quatro posições de precisão.

4. MÉTODO GENERALIZADO DO GRADIENTE REDUZIDO (GRG)

(vide refs. [16, 18, 28])

O problema geral de programação não-linear pode ser apresentado na seguinte forma :

Minimizar a função

$$f(\underline{x}) \quad , \quad \underline{x} = [x_1, x_2, \dots, x_N]^T \quad (4.1)$$

sujeita às restrições

$$g_j(\underline{x}) \geq 0 \quad , \quad j = 1, 2, \dots, J \quad (4.2)$$

$$h_k(\underline{x}) = 0 \quad , \quad k = 1, 2, \dots, K \quad (4.3)$$

sendo \underline{x} ... vetor das variáveis de projeto e

$f(\underline{x})$... função escalar das variáveis de projeto representando um critério de projeto ou objetivo.

As funções $g(\underline{x})$ e $h(\underline{x})$ representam, respectivamente, restrições de desigualdade e igualdade nas variáveis de projeto. As restrições de desigualdade delimitam regiões possíveis dentro do espaço de projeto, enquanto as restrições de igualdade descrevem combinações específicas das variáveis de projeto.

O problema da programação não-linear com restrições po-

de ser reapresentado numa outra forma :

$$\text{Minimizar } f(\underline{x}), \quad \underline{x} = [x_1, x_2, \dots, x_N]^T \quad (4.4)$$

$$\text{sujeita a } h_m(\underline{x}) = 0, \quad m = 1, 2, \dots, M \quad (4.5)$$

$$\text{onde } \underline{x}^{(l)} \leq \underline{x} \leq \underline{x}^{(u)} \quad (4.6).$$

Os vetores $\underline{x}^{(l)}$ e $\underline{x}^{(u)}$ representam os limites inferiores e superiores, respectivamente, do vetor-projeto \underline{x} . As restrições de desigualdade foram consideradas como restrições de igualdade pela introdução de variáveis não-negativas $x_{N+j} \geq 0$ tais que

$$h_{K+j}(\underline{x}) = g_j(\underline{x}) - x_{N+j} \quad (4.7).$$

Estas variáveis não-negativas são incluídas no conjunto original de variáveis de projeto. O parâmetro N então representa o número total de variáveis de projeto mais o número de variáveis não-negativas introduzidas em (4.7). O parâmetro M representa o número total de restrições $M = J + K$.

O método se inicia com a divisão do vetor-projeto \underline{x} em duas classes chamadas variáveis básicas e não-básicas.

$$\underline{x} = [\underline{z}, \underline{y}]^T \quad (4.8)$$

$$\underline{z} = [z_1, z_2, \dots, z_q]^T \dots \text{variáveis não-básicas} \quad (4.9)$$

$$\underline{y} = [y_1, y_2, \dots, y_M]^T \dots \text{variáveis básicas} \quad (4.10)$$

onde $Q = N - M$.

As variáveis básicas serão usadas para satisfazer as restrições e, por isso, serão chamadas de dependentes; as variáveis não-básicas serão independentes.

Serão calculados o diferencial de $f(\underline{x})$ na expressão (4.4) e o diferencial de $h_m(\underline{x})$ na expressão (4.5) com relação às variáveis básicas e não-básicas.

$$df = \nabla_z f(\underline{x})^T d\underline{z} + \nabla_y f(\underline{x})^T d\underline{y} \quad (4.11)$$

$$dh = C d\underline{z} + J d\underline{y} = 0 \quad (4.12)$$

onde

$$\nabla_z f(\underline{x}) = [\partial f / \partial z_1, \partial f / \partial z_2, \dots, \partial f / \partial z_Q]^T \quad (4.13)$$

$$\nabla_y f(\underline{x}) = [\partial f / \partial y_1, \partial f / \partial y_2, \dots, \partial f / \partial y_M]^T \quad (4.14)$$

$$J = [\nabla_y h_1(\underline{x}), \nabla_y h_2(\underline{x}), \dots, \nabla_y h_M(\underline{x})]^T; \quad \text{matriz } M \times M \quad (4.15)$$

$$C = [\nabla_z h_1(\underline{x}), \nabla_z h_2(\underline{x}), \dots, \nabla_z h_M(\underline{x})]^T; \quad \text{matriz } M \times Q \quad (4.16)$$

$$\nabla_y h_m(\underline{x}) = [\partial h_m / \partial y_1, \partial h_m / \partial y_2, \dots, \partial h_m / \partial y_M] \quad (4.17)$$

$$\nabla_{\underline{z}} h_m(\underline{x}) = [\frac{\partial h_m}{\partial z_1}, \frac{\partial h_m}{\partial z_2}, \dots, \frac{\partial h_m}{\partial z_q}]$$

(4.18)

De (4.12) obtém-se

$$d\underline{y} = - J^{-1}C d\underline{z}$$

(4.19)

Substituindo (4.19) em (4.11) obtém-se o gradiente reduzido

$$\nabla_r f(\underline{x}) = \nabla_z f(\underline{x})^T - \nabla_y f(\underline{x})^T J^{-1} C$$

(4.20)

A partir do gradiente reduzido, pode-se determinar a direção de busca de um ponto de mínimo local. O gradiente reduzido define a taxa de variação da função objetivo com relação às variáveis não-básicas. Pela equação (4.19) percebe-se que as variáveis básicas são ajustadas a partir das não-básicas para manter a busca verificando as restrições de projeto.

5. OTIMIZAÇÃO DO ÂNGULO DE TRANSMISSÃO [4, 22]

5.1 - Introdução

Apresenta-se neste capítulo as expressões básicas para o cálculo do ângulo de transmissão e da função objetivo da otimização, bem como a estratégia a ser seguida no seu cálculo no caso de ocorrer travamento do mecanismo. Seguem-se também no texto as restrições impostas às variáveis de projeto para a otimização.

5.2 - Cálculo do ângulo de transmissão

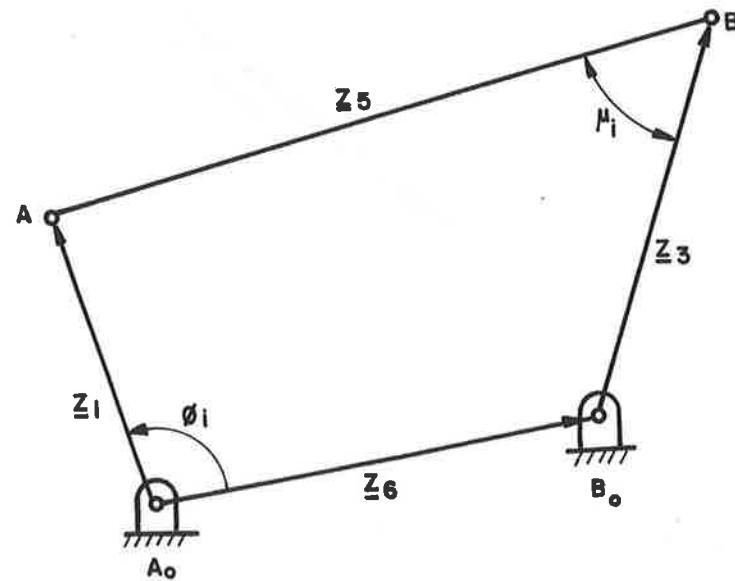


Fig. 5.1 - Ângulo de transmissão no quadrilátero articulado.

$$\text{No } \triangle A_0 A B_0, \quad \phi_i = \arg(z_{1i}) - \arg(z_{6i})$$

onde $\arg(z_{1i}) = \arg(z_1)_{\text{inicial}} + (i - 1) \cdot \Delta\theta$ sendo

$$\Delta\theta = 360^\circ / n \quad i = 1, \dots, n$$

onde n é o número de posições da alavanca motora quando esta realiza uma rotação completa (2π radianos ou 360°).

$$d_i^2 = z_1^2 + z_6^2 - 2 \cdot z_1 \cdot z_6 \cdot \cos\phi_i \quad (5.1)$$

$$\text{No } \Delta ABB_0, \quad d_i^2 = z_5^2 + z_3^2 - 2 \cdot z_5 \cdot z_3 \cdot \cos\mu_i \quad (5.2)$$

Igualando (5.1) e (5.2) vem

$$\cos\mu_i = \frac{z_5^2 + z_3^2 - z_1^2 - z_6^2 + 2 \cdot z_1 \cdot z_6 \cdot \cos\phi_i}{2 \cdot z_5 \cdot z_3} \quad (5.3)$$

como $0 \leq \mu \leq \pi \Rightarrow \mu_i = \arccos(\cos\mu_i)$

$$\mu_i = \mu_i(\phi_i, z_1, z_3, z_5, z_6)$$

5.3 - Função Objetivo : Desvio do Ângulo de transmissão

Adotou-se a seguinte função

$$\text{Desvio}(x) = \sum_{i=1}^n (\mu_i - \pi/2)^2 \quad (5.4)$$

No cálculo da função desvio, admite-se que o mecanismo a

ser sintetizado sempre realize uma rotação completa da alavanca motora. Essa hipótese nem sempre é verdadeira, pois pode haver travamento ($\mu = 0^\circ$ ou 180°) durante a movimentação do mecanismo. Isso pode ser percebido observando-se a expressão (5.3) que pode anular-se ($\mu = 0^\circ$ ou 180°) ou assumir valores maiores que 1 ou menores que -1. Se algum destes casos ocorrer, então pode-se atribuir ao ângulo de transmissão um valor elevado e como consequência, a função desvio assumirá também um valor alto. Como a sub-rotina de otimização busca um mínimo local para a função objetivo, ela se afastará desta situação procurando um vetor \underline{x} que possibilite um rotação completa da alavanca motora.

5.3.1 - Três posições de precisão

O vetor \underline{x} , variáveis de projeto, foi apresentado como dependente dos parâmetros

$$\underline{x} = [\phi_i, z_1, z_3, z_5, z_6]^T, i = 1, \dots, n$$

como todos estes parâmetros do vetor \underline{x} são dependentes dos parâmetros de projeto $A_o, B_o, \alpha_2, \alpha_3, P_1, P_2, P_3$, então pode-se escrever

$$\underline{x} = [A_o, B_o, \alpha_2, \alpha_3, P_1, P_2, P_3]^T \text{ e, portanto,}$$

$$\text{Desvio} = \text{Desvio}(A_o, B_o, \alpha_2, \alpha_3, P_1, P_2, P_3)$$

Numa 1a. fase, pode ser definida uma função objetivo dependente somente das variáveis A_0 e B_0 , os outros parâmetros de projeto são mantidos sem qualquer alteração durante a otimização. A sub-rotina de otimização deve alterar estas variáveis livres, dentro de determinadas restrições, de tal maneira que a função desvio atinja o valor de um mínimo local.

Feita a otimização com 4 variáveis escalares, uma outra função objetivo pode ser elaborada já com otimização de todos os parâmetros independentes.

5.3.2 - Quatro posições de precisão

O vetor \underline{x} , variáveis de projeto, foi apresentado como dependente dos parâmetros

$$\underline{x} = [\phi_i, z_1, z_3, z_5, z_6]^T, i = 1, \dots, n$$

como todos estes parâmetros do vetor \underline{x} são dependentes dos parâmetros de projeto $\beta_{21}, \beta_{22}, \alpha_2, \alpha_3, \alpha_4, P_1, P_2, P_3, P_4$, então pode-se escrever

$$\underline{x} = [\beta_{21}, \beta_{22}, \alpha_2, \alpha_3, \alpha_4, P_1, P_2, P_3, P_4]^T \text{ e, portanto,}$$

$$\text{Desvio} = \text{Desvio}(\beta_{21}, \beta_{22}, \alpha_2, \alpha_3, \alpha_4, P_1, P_2, P_3, P_4)$$

Numa 1a. fase, pode ser definida uma função objetivo

dependente somente das variáveis β_{z1} e β_{z2} , os outros parâmetros de projeto são mantidos sem qualquer alteração durante a otimização. A sub-rotina de otimização deve alterar estas variáveis livres, dentro de determinadas restrições, de tal maneira que a função desvio atinja o valor de um mínimo local.

Feita a otimização com 2 variáveis escalares, uma outra função objetivo pode ser elaborada já com otimização de todos os parâmetros independentes.

5.4 - Restrições

As restrições são de dois tipos, uma considerando a vizinhança de precisão para os pontos de interesse como, por exemplo, o ponto do acoplador que descreve uma trajetória; e outra levando em conta a vizinhança de precisão para os ângulos de rotação como, por exemplo, o do acoplador. Isto pode ser visualizado nas figuras 5.2 e 5.3.

5.4.1 - Três posições de precisão

As restrições serão apresentadas em dois casos, primeiro, com otimização de 4 variáveis, e segundo, com otimização de 12 variáveis.

5.4.1.1 - Otimização de 4 variáveis

As restrições se referem somente às variáveis A_o e B_o .

$$| A_o - A_o^d | \leq d_1$$

$$| B_o - B_o^d | \leq d_2$$

5.4.1.2 - Otimização de 12 variáveis

As restrições se referem às variáveis A_o , B_o , α_2 , α_3 , P_1 , P_2 , P_3 .

$$| A_o - A_o^d | \leq d_1$$

$$| B_o - B_o^d | \leq d_2$$

$$| \alpha_j - \alpha_j^d | \leq d_3, \quad j = 2, 3$$

$$| P_j - P_j^d | \leq d_4, \quad j = 1, \dots, 3$$

onde o índice superior d indica o valor desejado para a variável e os escalares d_1 , d_2 , d_3 , d_4 referem-se aos desvios máximos permitidos nas variáveis de projeto.

5.4.2 - Quatro posições de precisão

De maneira análoga, as restrições se referem a dois casos, primeiro a otimização com 2 variáveis, e segundo, a

otimização com 13 variáveis.

5.4.2.1 - Otimização de 2 variáveis

As restrições se referem às variáveis β_z . O segundo índice se refere à diâda : (1) para a motora e (2) para a movida.

$$|\beta_{z1} - \beta_{z1}^d| \leq d_1 \quad e$$

$$|\beta_{z2} - \beta_{z2}^d| \leq d_2$$

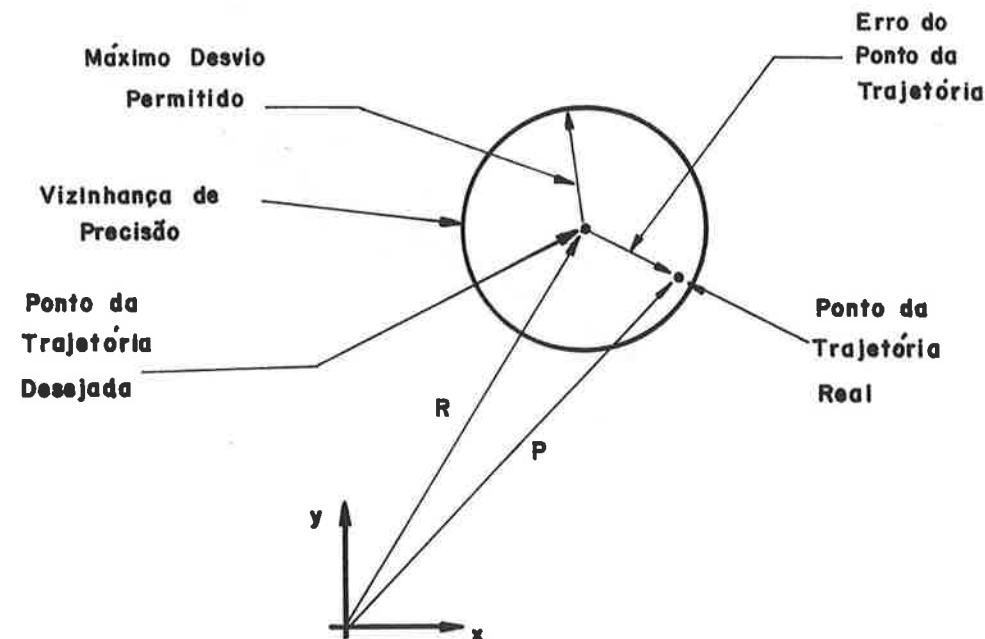


Fig. 5.2 - Vizinhança de precisão para uma posição do ponto do acoplador.

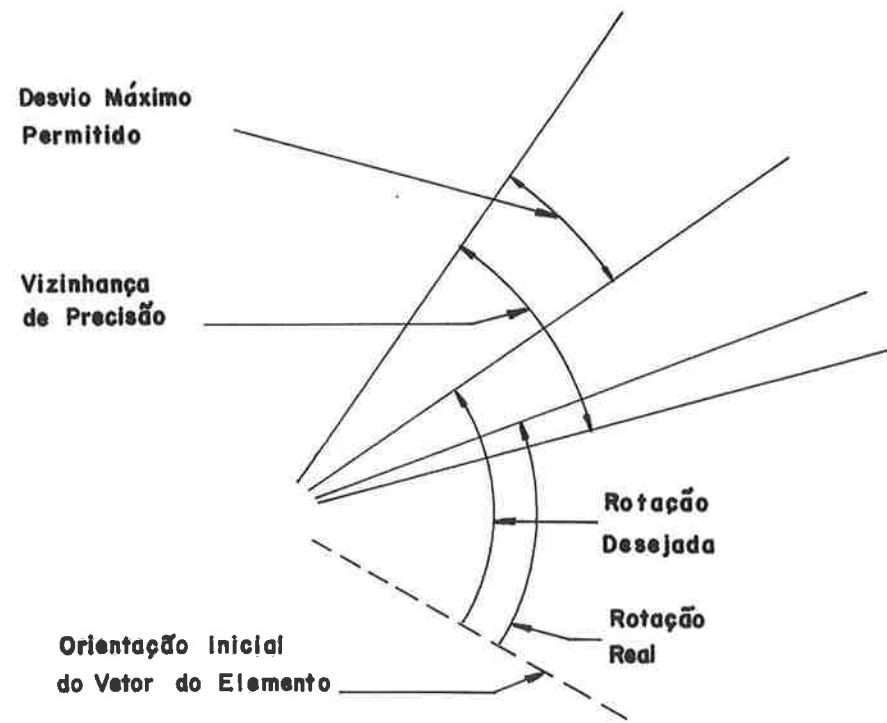


Fig. 5.3 - Vizinhança de precisão para uma especificação de rotação.

5.4.2.2 - Otimização de 13 variáveis

As restrições se referem à β_{21} , β_{22} , α_2 , α_3 , α_4 , P_1 , P_2 , P_3 , P_4 :

$$|\beta_{21} - \beta_{21}^d| \leq d_1 \quad e$$

$$|\beta_{22} - \beta_{22}^d| \leq d_2$$

$$|\alpha_j - \alpha_j^d| \leq d_3, \quad j = 2, \dots, 4$$

$$|P_j - P_j^d| \leq d_4, \quad j = 1, \dots, 4$$

onde o índice superior d indica o valor desejado para a variável e os escalares d_1 , d_2 , d_3 , d_4 referem-se aos desvios máximos permitidos nas variáveis de projeto.

6. PROBLEMAS PROPOSTOS

6.1 - Introdução

Serão apresentados agora os três problemas que este trabalho se propõe a resolver. Estes problemas foram retirados de três publicações na área de projeto de mecanismos [6], [10] e [13]. Em todos os problemas, a tarefa a ser realizada é uma geração de movimento que atenda às especificações de projeto, ou seja, as posições de precisão do ponto P e os ângulos de rotação do acoplador. Pretende-se, além de gerar um mecanismo que atenda a essas especificações, obter um mínimo desvio do ângulo de transmissão em torno de 90° durante toda a movimentação do mecanismo.

6.2 - Problema de três posições [6]

As posições de precisão do ponto P e os ângulos de rotação do acoplador são mostrados na tabela 6.1 e 6.2, respectivamente.

Serão atribuídos valores às variáveis livres A_{ox} , A_{oy} , B_{ox} e B_{oy} . Estes valores são mostrados na tabela 6.3 e coincidem com aqueles sugeridos no problema apresentado por Suh e Radcliffe [6].

Tabela 6.1 - Posições do ponto P do acoplador.

P	x	y
1	1.0	1.0
2	2.0	0.5
3	3.0	1.5

Tabela 6.2 - Ângulos de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_2	0.0
α_3	45.0

Tabela 6.3 - Variáveis livres (articulações fixas).

Pontos	x	y
Ao	5.0	0.0
Bo	0.0	0.0

Apresentam-se nos itens a seguir as restrições das variáveis livres e dos parâmetros de projeto que são dados importantes no que se refere à otimização.

6.2.1 - Otimização com 4 variáveis livres

As variáveis livres neste caso são as coordenadas das

articulações fixas A_{ox} , A_{oy} , B_{ox} e B_{oy} . As restrições vêm a seguir.

$$(A_{ox} - 5)^2 \leq 4$$

$$(A_{oy} - 0)^2 \leq 4$$

$$(B_{ox} - 0)^2 \leq 4$$

$$(B_{oy} - 0)^2 \leq 4$$

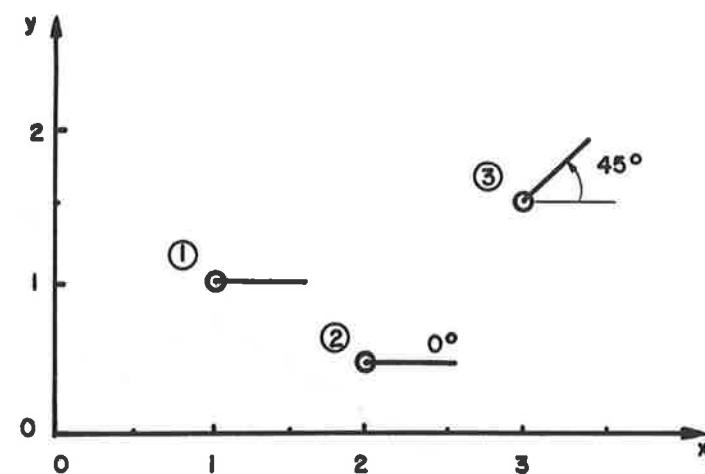


Fig. 6.1 - Posições de precisão do ponto P e ângulos de rotação do acoplador.

6.2.2 - Otimização com 12 variáveis

As variáveis livres neste caso também são as coordenadas das articulações fixas A_{ox} , A_{oy} , B_{ox} e B_{oy} . A essas variáveis acrescentam-se os parâmetros de projeto que são as posições do ponto P, P_{ix} , P_{iy} , P_{zx} , P_{zy} , P_{sx} , P_{sy} e os ângulos de rotação do acoplador α_z e α_s . As restrições vêm a seguir.

$$\begin{aligned}
 C A_{ox} - 5 &\geq 4 \\
 C A_{oy} - 0 &\geq 4 \\
 C B_{ox} - 0 &\geq 4 \\
 C B_{oy} - 0 &\geq 4 \\
 C \alpha_2 - 0^\circ &\geq C 2^\circ \geq 2 \\
 C \alpha_3 - 45^\circ &\geq C 2^\circ \geq 2 \\
 C P_{1x} - 1 &\geq C P_{1y} - 1 \leq C 0.2 \geq 2 \\
 C P_{2x} - 2 &\geq C P_{2y} - 0.5 \geq C 0.2 \geq 2 \\
 C P_{3x} - 3 &\geq C P_{3y} - 1.5 \geq C 0.2 \geq 2
 \end{aligned}$$

6.3 - Problema de quatro posições [10]

As posições de precisão e os ângulos de rotação do acoplador são mostrados na tabela 6.3 e 6.4, respectivamente.

Tabela 6.4 - Posições do ponto P do acoplador.

P	x	y
1	0.0	0.0
2	5.0	8.0
3	10.0	15.0
4	18.0	20.0

Serão atribuídos valores às variáveis livres β_{21} e β_{22} . Estes valores são mostrados na tabela 6.6 e coincidem com aqueles sugeridos no problema apresentado por Gustafson e Erdman [10].

Tabela 6.5 - Ângulos de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_2	10.0
α_3	20.0
α_4	30.0

Tabela 6.6 - Ângulos de rotação das alavancas da 1a. e 2a. diáadas.

Ângulos (graus)	1a. diáada	2a. diáada
β_2	-38.00000	-7.00000

Na sequência, também se apresentam as restrições das variáveis livres e dos parâmetros de projeto.

6.3.1 - Otimização com 2 variáveis livres

As variáveis livres neste caso são os ângulos de rotação das alavancas da 1a. e 2a. diáada, respectivamente, β_{21} e β_{22} . As restrições vem a seguir.

$$(\beta_{21} - 38^\circ)^2 \leq (28.65^\circ)^2$$

$$(\beta_{22} - 7^\circ)^2 \leq (28.65^\circ)^2$$

6.3.2 - Otimização com 13 variáveis livres

As variáveis livres neste caso também são os ângulos de rotação das alavancas da 1a. e 2a. diada, respectivamente, β_{21} e β_{22} . A essas variáveis acrescentam-se as posições de precisão do ponto P, $P_{1x}, P_{1y}, P_{2x}, P_{2y}, P_{3x}, P_{3y}, P_{4x}, P_{4y}$ e os ângulos de rotação do acoplador α_2, α_3 e α_4 . As restrições vem a seguir.

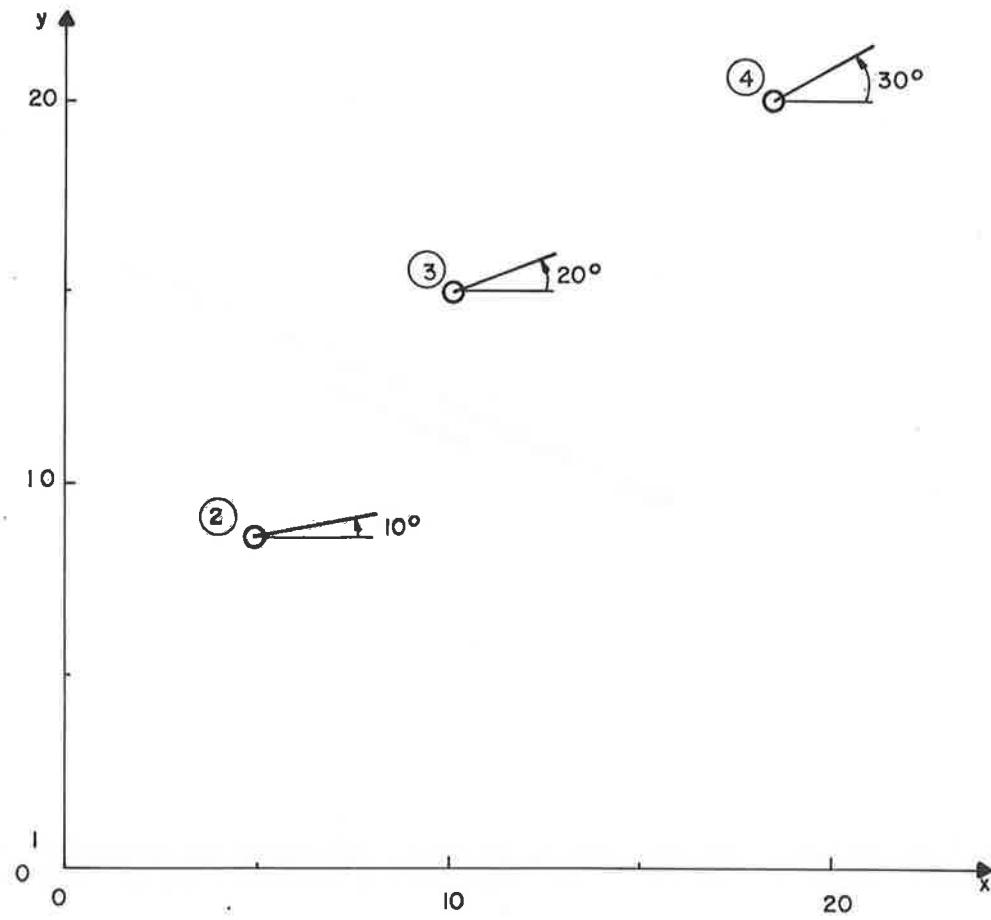


Fig. 6.2 - Posições de precisão do ponto P e ângulos de rotação do acoplador.

$$\begin{aligned}
 C\beta_{21} - 38^\circ &\leq C 28.65^\circ \\
 C\beta_{22} - 7^\circ &\leq C 28.65^\circ \\
 C\alpha_2 - 10^\circ &\leq C 2^\circ \\
 C\alpha_3 - 20^\circ &\leq C 2^\circ \\
 C\alpha_4 - 30^\circ &\leq C 2^\circ \\
 C P_{1x} - 0 &\leq C 0.2 \\
 C P_{2x} - 5 &\leq C 0.2 \\
 C P_{3x} - 10 &\leq C 0.2 \\
 C P_{4x} - 18 &\leq C 0.2
 \end{aligned}$$

6.4 - Problema da rotação completa [13]

Neste problema, além dos objetivos perseguidos nos casos anteriores (geração de movimento, posições de precisão do acoplador, ângulos de rotação do acoplador e mínimo desvio do ângulo de transmissão em torno de 90°), pretende-se transformar um mecanismo que sofra travamento (ângulo μ igual a 0° ou 180°) durante a sua operação, num que possua rotação completa, ou seja, a alavanca motora é uma manivela. As posições de precisão e os ângulos de rotação do acoplador são mostrados nas tabelas 6.7 e 6.8, respectivamente.

Serão atribuídos valores às variáveis livres A_{ox} , A_{oy} , B_{ox} e B_{oy} . Estes valores são mostrados na tabela 6.9 e coincidem com aqueles sugeridos no problema apresentado por Peterson e colaboradores [13].

Tabela 6.7 – Posições do ponto P do acoplador.

P	x	y
1	-19.2100	-1.6200
2	-19.3711	12.1761
3	-13.3333	54.6020

Tabela 6.8 – Ângulo de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_z	-13.86
α_s	-179.01

Tabela 6.9 – Variáveis livres(articulações fixas).

Pontos	x	y
A _o	-14.0	20.8
B _o	12.2	33.9

Neste caso, a otimização a ser desenvolvida se atém somente a 4 variáveis que correspondem às coordenadas das articulações fixas. As restrições vem a seguir.

$$(A_{ox} + 14)^2 \leq 3^2$$

$$\text{C } A_{oy} - 20.8 \text{ } \textcircled{2} \leq 3^2$$

$$\text{C } B_{ox} - 12.2 \text{ } \textcircled{2} \leq 3^2$$

$$\text{C } B_{oy} - 33.9 \text{ } \textcircled{2} \leq 3^2$$

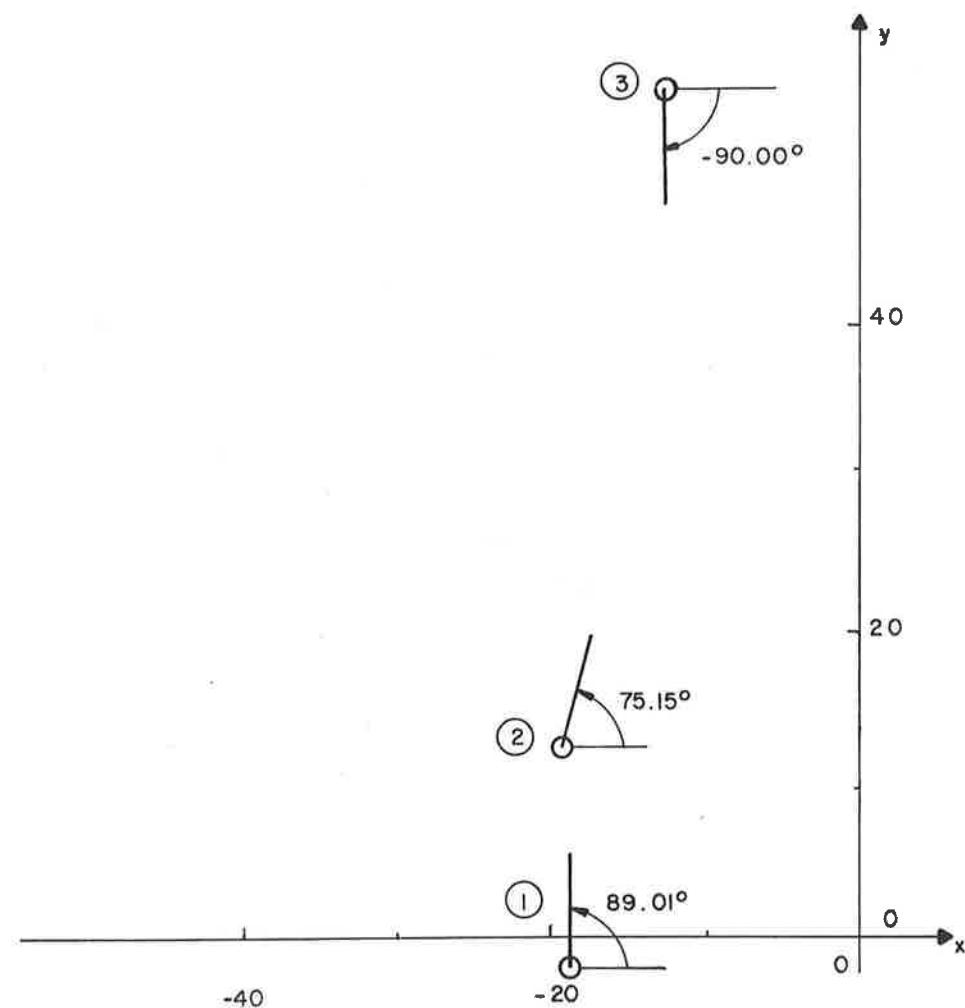


Fig. 6.3 – Posições de precisão do ponto P e ângulos de rotação do acoplador.

7. RESULTADOS

7.1 - Introdução

Inicialmente, foram desenvolvidos dois programas de computador para síntese de um quadrilátero articulado para geração de movimento , sendo um para três e outro para quatro posições de precisão sem qualquer otimização. Para verificação dos resultados, foram elaborados um programa para análise do quadrilátero obtido na síntese e um programa gráfico para melhor visualização do funcionamento do mecanismo.

Para a otimização do ângulo de transmissão foi utilizada uma sub-rotina desenvolvida por G. A. Gabriele e K. M. Ragsdell da Universidade de Missouri-Rolla. Depois realizou-se um acoplamento desta sub-rotina com as sub-rotinas de síntese dos programas anteriores.

No processo de otimização, o problema foi resolvido por partes. Numa primeira fase, permitiu-se uma grande margem de variação para os parâmetros livres e os parâmetros de projeto se mantiveram todos fixos. Posteriormente, permitiu-se a variação de todos os parâmetros, sendo que houve uma margem muito mais estreita com relação aos parâmetros de projeto.

Para três posições de precisão, o número de variáveis

livres é quatro e o número total de variáveis independentes é 12. No caso de quatro posições, o número de variáveis livres é dois e o número total de variáveis independentes é 13.

A seguir, são apresentados os resultados dos problemas apresentados no capítulo de dados, onde se compara a solução tradicional, sem otimização, e as soluções com otimização apenas das variáveis livres e com otimização de todas as variáveis independentes.

Um outro problema resolvido com a otimização foi o de obter um mecanismo que, além de ter um ângulo de transmissão próximo de 90 graus, possuisse um elemento do tipo manivela(que efetue uma rotação completa, 360 graus).

Também se apresenta neste capítulo um resumo do que cada programa realiza, as linguagens de programação e o tipo de computador utilizado .

7.2 - RESULTADOS DA OTIMIZAÇÃO DE 3 POSIÇÕES

7.2.1 - Síntese sem otimização

7.2.1.1 - Dados das posições de precisão

Tabela 7.1 - Posições do ponto P do acoplador.

P	x	y
1	1.00000000	1.00000000
2	2.00000031	0.50000096
3	2.99999823	1.50000065

Tabela 7.2 - Ângulos de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_z	-0.000014
α_s	44.999976

7.2.1.2 - Dados do mecanismo obtido

Tabela 7.3 - Posições de precisão do ponto P do acoplador

Pontos	x	y
Ao	5.000000	0.000000
Bo	0.000000	0.000000
A	3.547723	-1.654555
B	0.994078	3.238155
P	1.000000	1.000000

Tabela 7.4 - Ângulos de rotação do acoplador

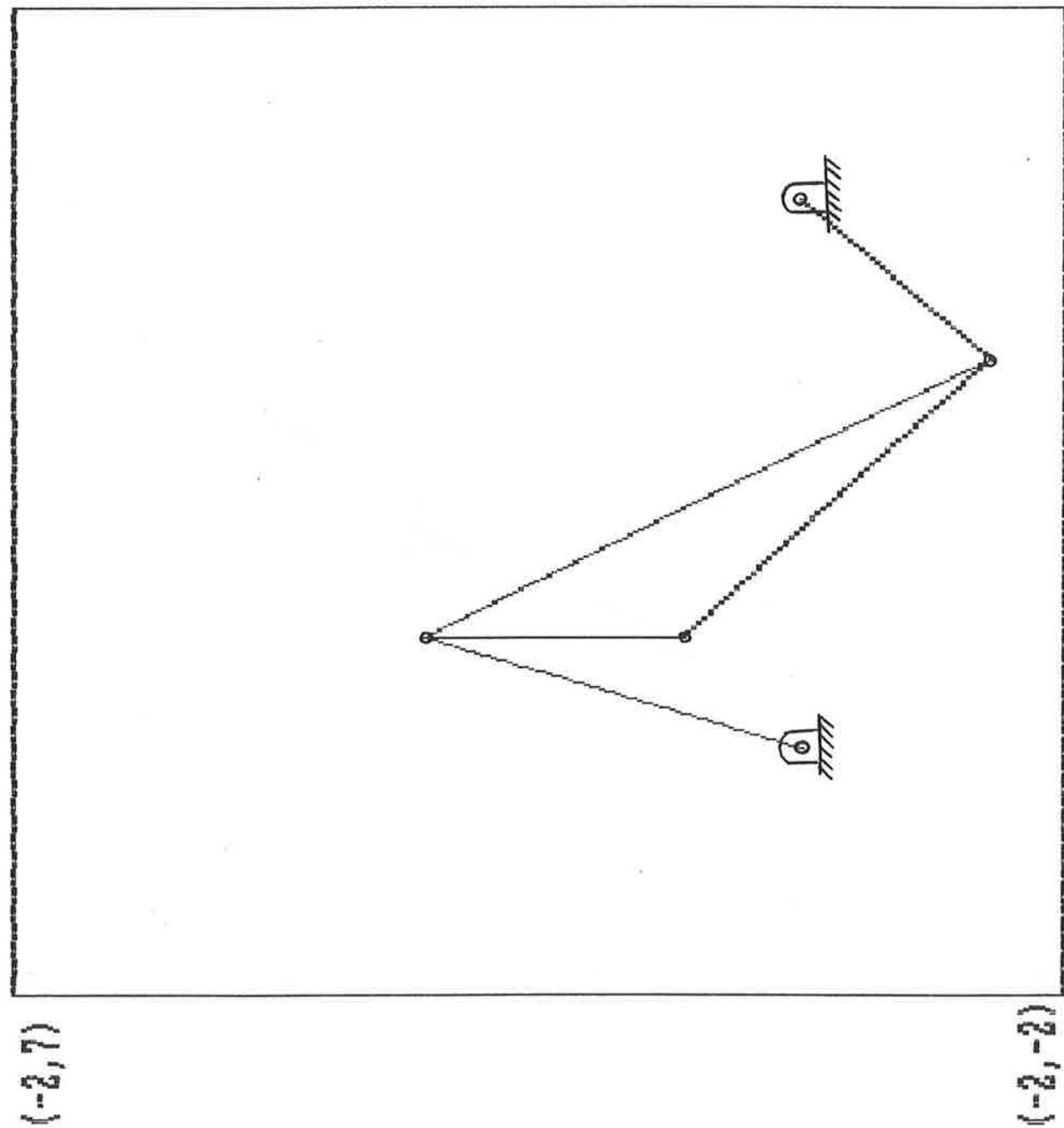
Ângulos (graus)	1a. diada	2a. diada
β_2	29.41966	-18.99830
β_3	171.59336	-7.59229

7.2.2 - Otimização com 4 variáveis livres

7.2.2.1 - Dados da função de otimização

Na tabela 7.5 apresenta-se os valores da função de otimização na primeira e última iteração.

(l', l)

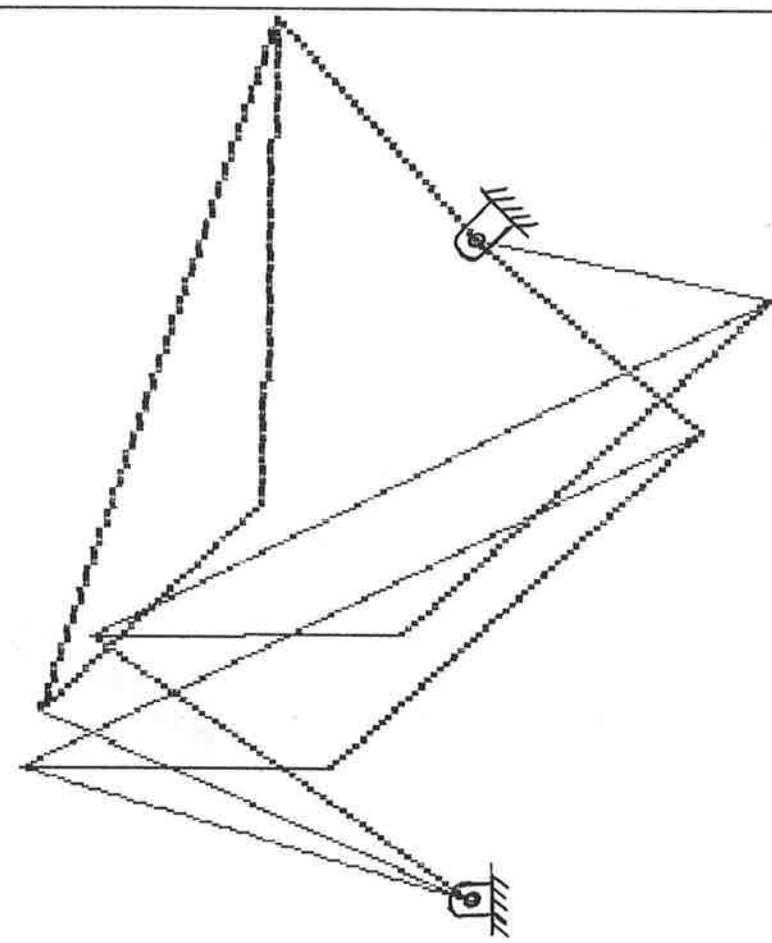


($l, -3$)

Fig. 7.1 - Mecanismo obtido conforme tabela 7.3.

(7,7)

- 1^a Posição
- 2^a Posição
- 3^a Posição



(-2,-2)

(-2,7)

(7,-2)

Fig. 7.2 - Posições de precisão do mecanismo obtido.

Tabela 7.5 - Valor da função na 1a. e última iteração.

Iteração	no. da iteração	Valor da função
Primeira	1	6.22086
Última	13	3.54556

7.2.2.2 - Dados das posições de precisão

Tabela 7.6 - Posições do ponto P do acoplador.

P	x	y
1	1.00000000	1.00000000
2	1.99999991	0.50000178
3	3.00000009	1.49999998

Tabela 7.7 - Erro nas posições do ponto P do acoplador.

Erro posição de P (%)	x	y
1	0.00000	0.00000
2	0.00000	0.00036
3	0.00000	0.00000

Tabela 7.8 - Ângulos de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_2	0. 00000000
α_3	45. 0000034

Tabela 7.9 - Erro nos ângulos de rotação do acoplador.

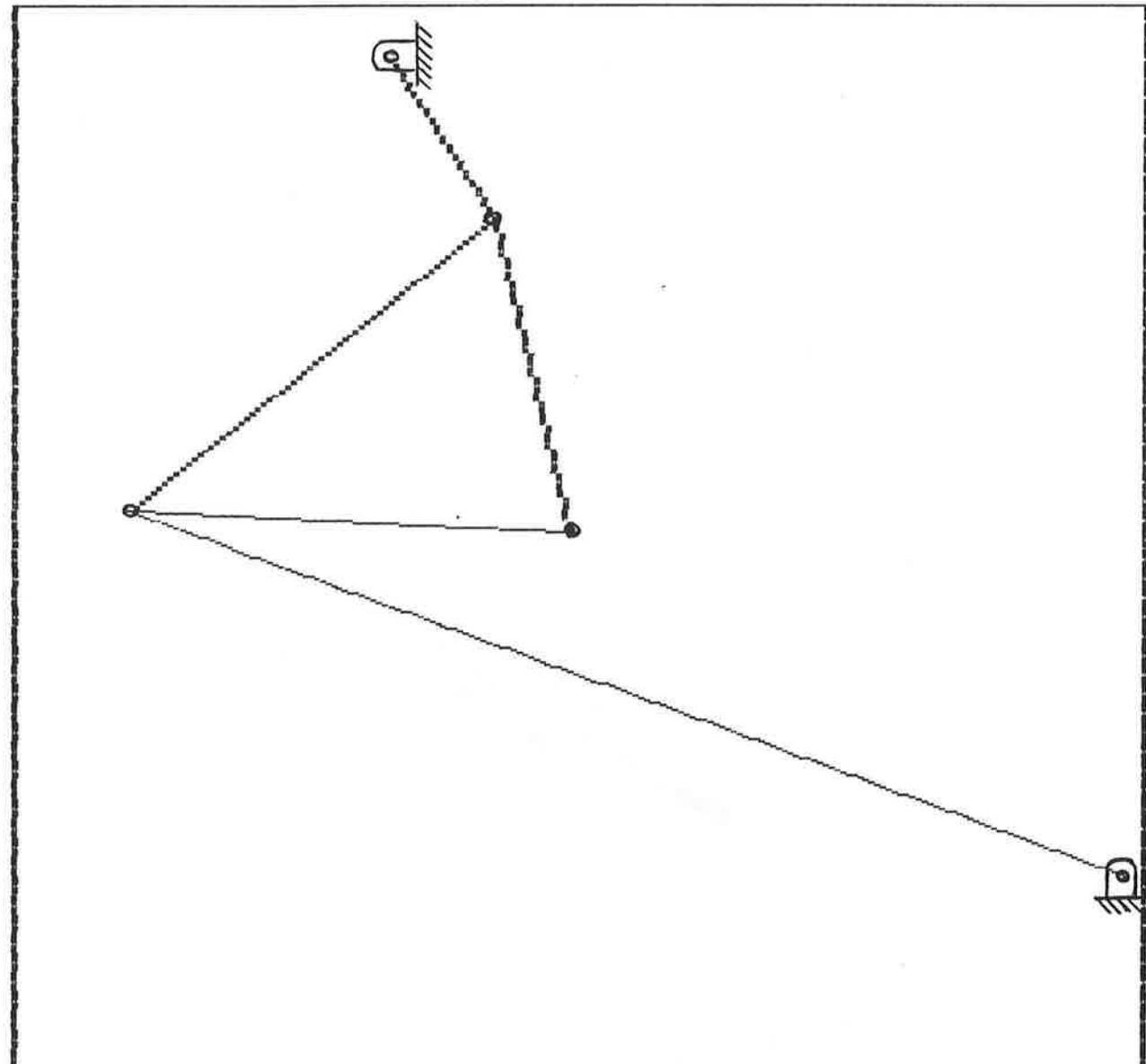
Erro no ângulo do acoplador (%)	
α_2	—
α_3	0. 00000

7.2.2.3 - Dados do mecanismo obtido

Tabela 7.10 - Articulações fixas e móveis e ponto P do acoplador

Pontos	x	y
A _o	3. 696600	2. 000000
B _o	-0. 945518	-2. 000000
A	2. 779405	1. 416804
B	1. 126385	3. 393808
P	1. 000000	1. 000000

(4,4)



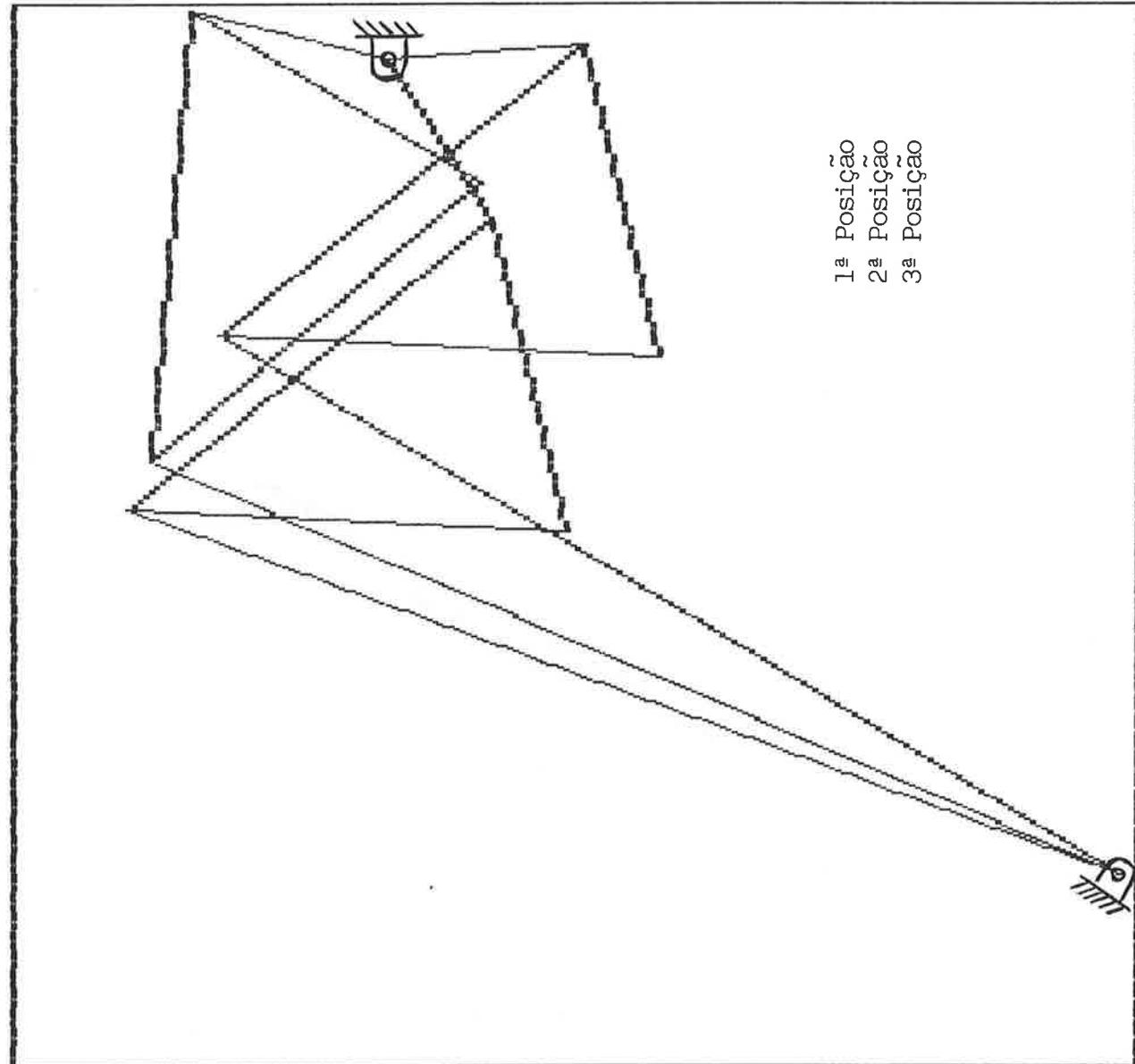
(-3,4)

(-2,-2)

(4,-2)

Fig. 7.3 - Mecanismo obtido conforme tabela 7.10.

(4, 4)



(-2, 4)

(4, -2)

(-2, -2)

Fig. 7.4 – Posições de precisão do mecanismo obtido.

Tabela 7.11 - Ângulos de rotação da alavanca.

Ângulos (graus)	1a. diáada	2a. diáada
β_z	61.93584	-11.10391
β_a	-136.72138	-2.90076

7.2.3 - Otimização com 12 variáveis

7.2.3.1 - Dados da função de otimização

Na tabela 7.12 apresenta-se os valores da função de otimização na primeira e última iteração.

Tabela 7.12 - Valor da função na 1a. e última iteração.

Iteração	no. da iteração	Valor da função
Primeira	1	6.22086
Última	61	3.46505

7.2.3.2 - Dados das posições de precisão

Tabela 7.13 - Posições do ponto P do acoplador.

P	x	y
1	1.19954650	1.01346140
2	1.79999999	0.50000178
3	3.18203337	1.41715061

Tabela 7.14 - Erro nas posições do ponto P do acoplador.

Erro posição de P (%)	x	y
1	19.954	1.3461
2	10.000	0.0000
3	6.068	5.5233

Tabela 7.15 - Ângulos de rotação do acoplador.

Ângulo de rotação do acoplador (graus)
α_2
α_3

(4, 4)

(-2, 4)

(4, -2)

(-2, -2)

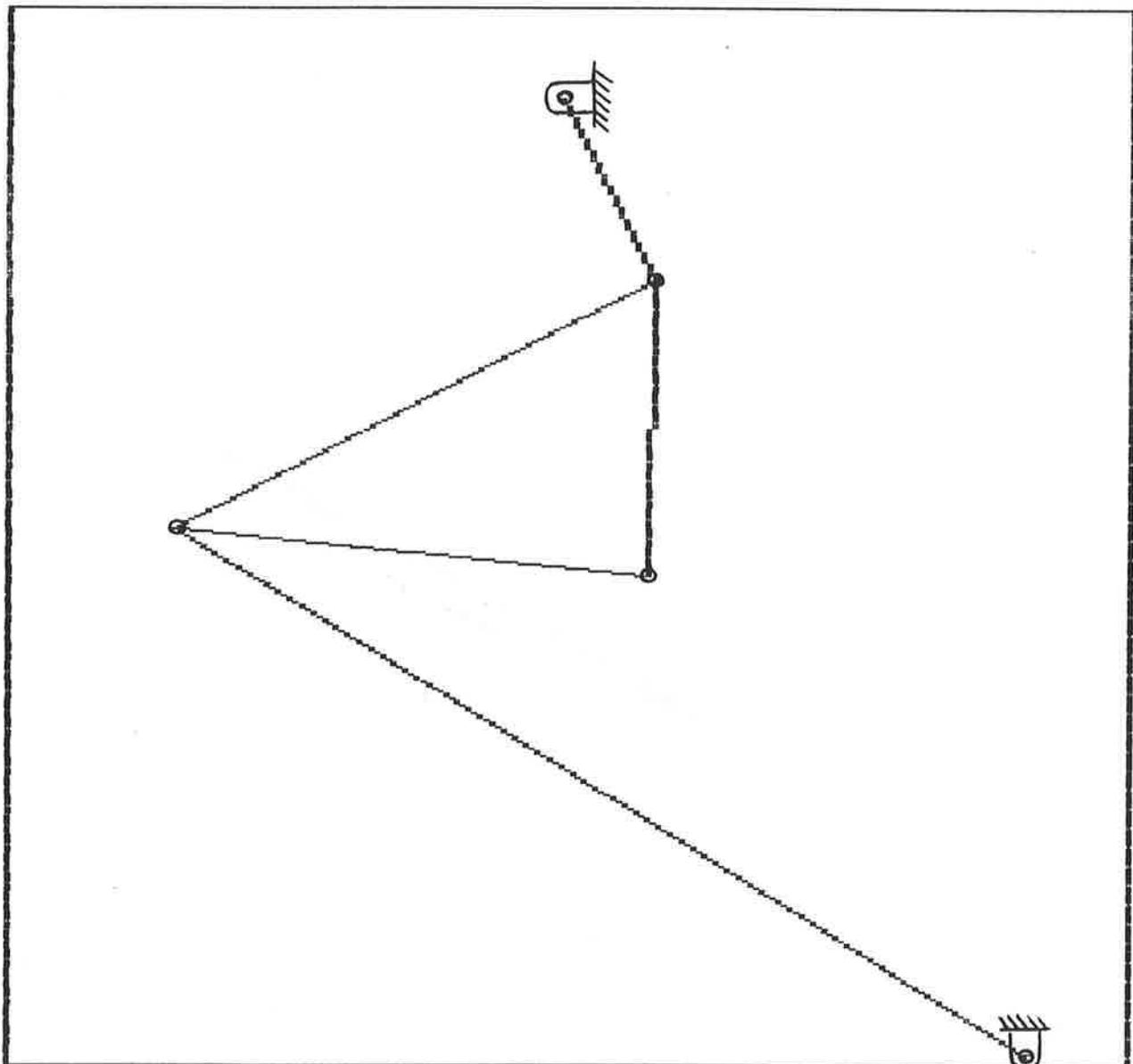
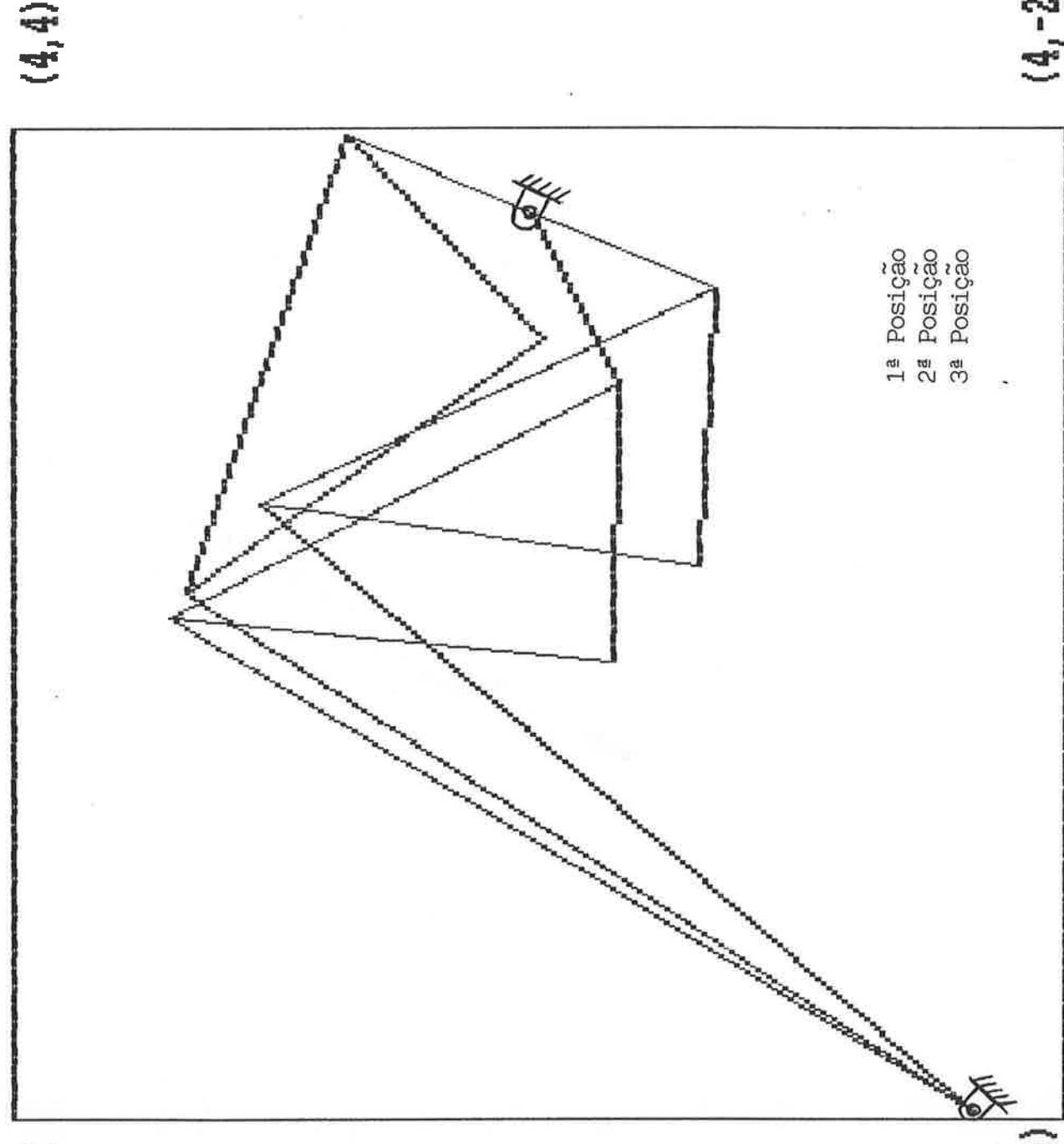


Fig. 7.5 - Mecanismo obtido conforme tabela 7.17.

(4, 4)

(-2, 4)



(4, -2)

(-2, -2)

Fig. 7.6 - Posições de precisão do mecanismo obtido.

Tabela 7.16 - Erro nos ângulos de rotação do acoplador.

Erro no ângulo do acoplador (%)	
α_z	—
α_s	4.44450

7.2.3.3 - Dados do mecanismo obtido

Tabela 7.17 - Articulações fixas e móveis e ponto P do acoplador.

Pontos	x	y
Ao	3.964670	1.485005
Bo	-1.570609	-1.136723
A	2.904778	0.977210
B	1.482859	3.595936
P	1.199547	1.013461

Tabela 7.18 - Ângulos de rotação da alavanca.

Ângulos (graus)	1a. diáda	2a. diáda
β_z	41.26664	-8.83173
β_s	-140.19812	-1.77515

7.2.4 - Resultados do ângulo de transmissão

Apresenta-se nas tabelas e gráficos a seguir , os resultados do ângulo de transmissão e do desvio deste em torno de 90° em função do ângulo β para síntese sem otimização, com otimização de 4 variáveis e com otimização de 12 variáveis.

Tabela 7.19 - Ângulos de transmissão $\times \beta$ para síntese sem otimização (B), com otimização de 4 variáveis (C), com otimização de 12 variáveis (D).

ângulo β (graus)	ângulo B μ (graus)	ângulo C μ (graus)	ângulo D μ (graus)
0	44.63	60.91	61.33
18	56.22	61.03	62.73
36	67.83	63.93	66.63
54	78.84	69.15	72.46
72	88.71	76.01	79.54
90	96.85	83.79	87.19
108	102.57	91.81	94.79
126	105.25	99.45	101.70
144	104.54	106.04	107.29
162	100.54	110.92	110.95
180	93.74	113.49	112.21
198	84.80	113.38	110.88
216	74.39	110.61	107.16
234	63.07	105.57	101.52
252	51.38	98.87	94.58
270	40.02	91.19	86.97
288	30.21	83.16	79.33
306	24.55	75.42	72.27
324	26.15	68.67	66.49
342	33.96	63.60	62.65
360	43.63	60.91	61.33

Ang. de Transmissão - 3 pos.

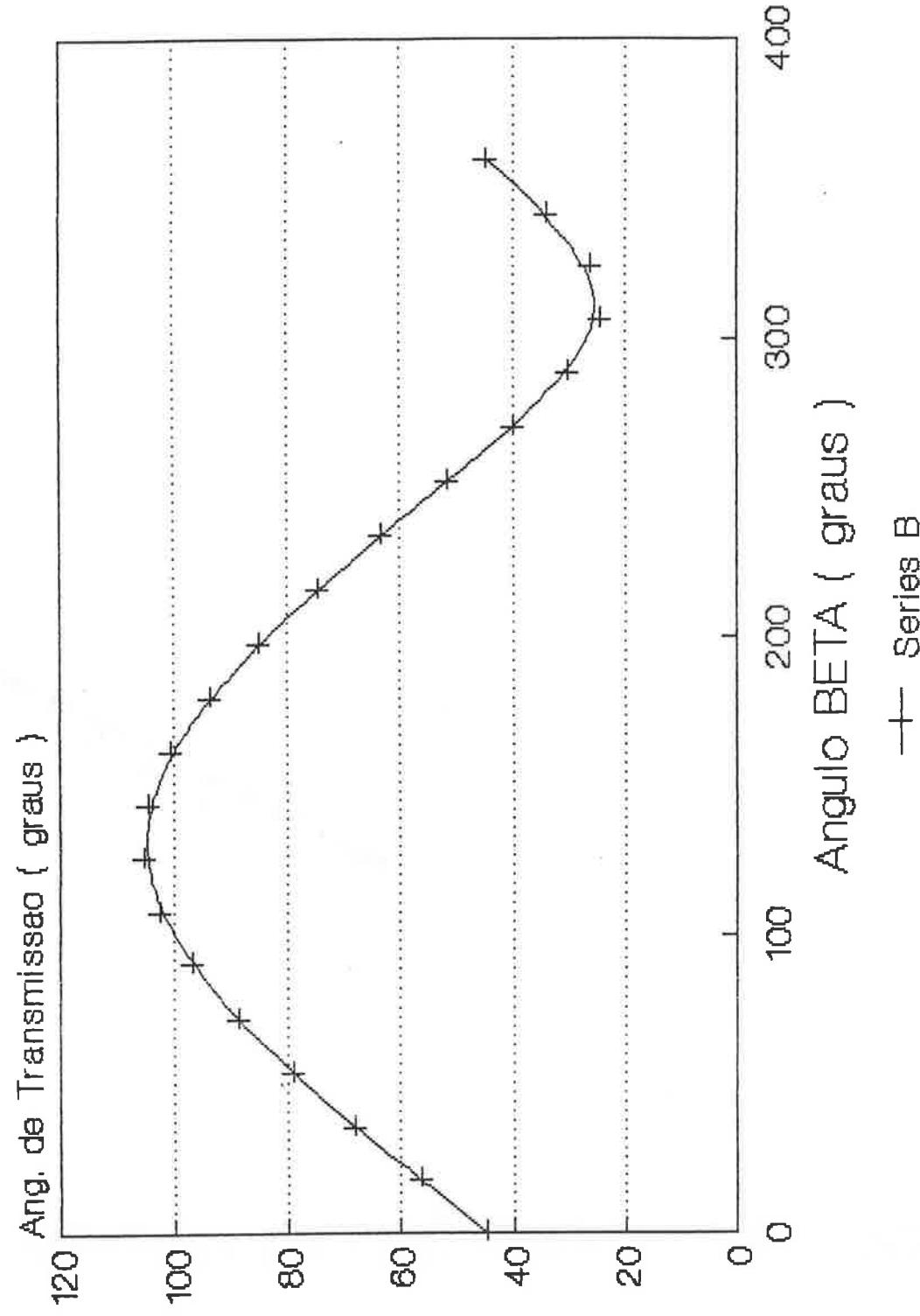


Fig. 7.7 - Gráfico do ângulo de transmissão μ , série B, em função do ângulo β .

Ang. de Transmissão - 3 pos.

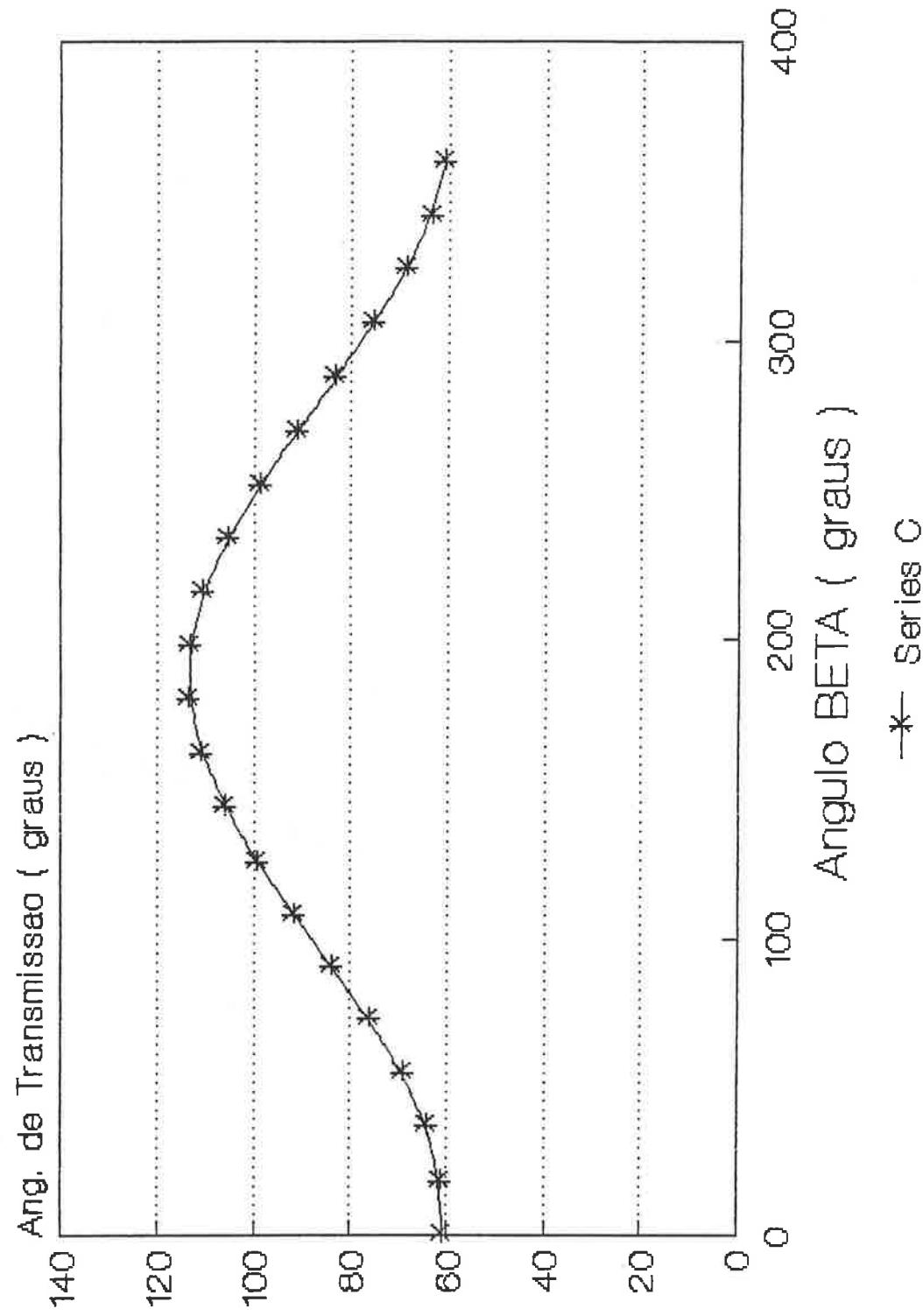


Fig. 7.8 - Gráfico do ângulo de transmissão μ , série C, em função do ângulo β .

Ang. de Transmissão - 3 pos.

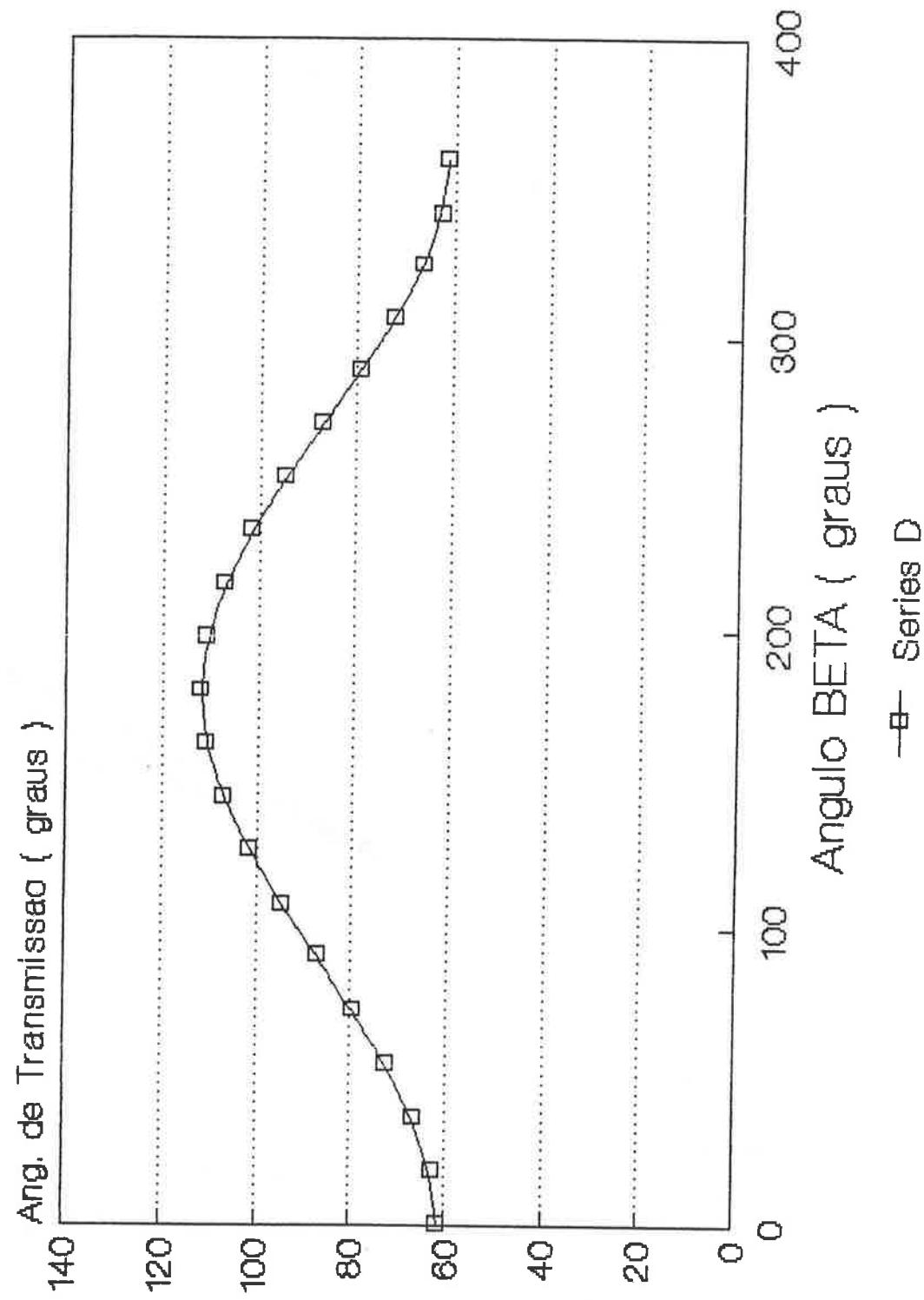


Fig. 7.9 - Gráfico do ângulo de transmissão μ , série D, em função do ângulo β .

Ang. de Transmissão - 3 pos.

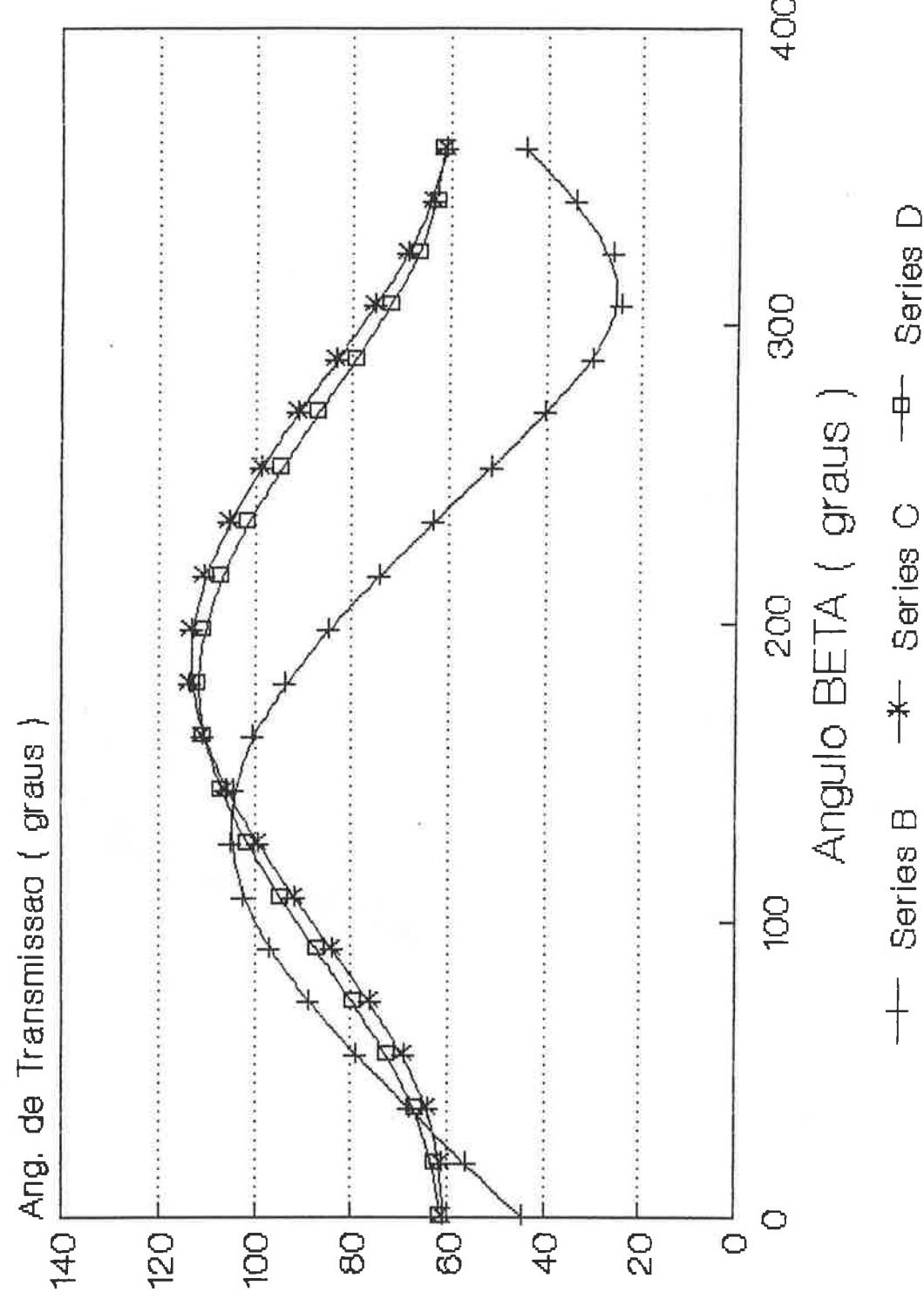


Fig. 7.10 - Gráfico comparativo do ângulo de transmissão das séries B, C e D em função de β .

Tabela 7.20 - Desvio do ângulo de transmissão $\times \beta$ para síntese sem otimização (B), com otimização de 4 variáveis (C), com otimização de 12 variáveis (D).

ângulo β (graus)	desvio B μ (graus)	desvio C μ (graus)	desvio D μ (graus)
0	-45.37	-29.09	-28.67
18	-33.78	-28.97	-27.27
36	-22.17	-26.07	-23.37
54	-11.16	-20.85	-17.54
72	-1.29	-13.99	-10.46
90	6.85	-6.21	-2.81
108	12.57	1.81	4.79
126	15.25	9.45	11.70
144	14.54	16.04	17.29
162	10.54	20.92	20.95
180	3.74	23.49	22.21
198	-5.20	23.38	20.88
216	-15.61	20.61	17.16
234	-26.93	15.57	11.52
252	-38.62	8.87	4.58
270	-49.98	1.19	-3.03
288	-59.79	-6.84	-10.67
306	-65.45	-14.58	-17.73
324	-63.85	-21.33	-23.51
342	-56.04	-26.40	-27.35
360	-45.37	-29.09	-28.67

Desvio do Ang. de Transmissao - 3 pos.

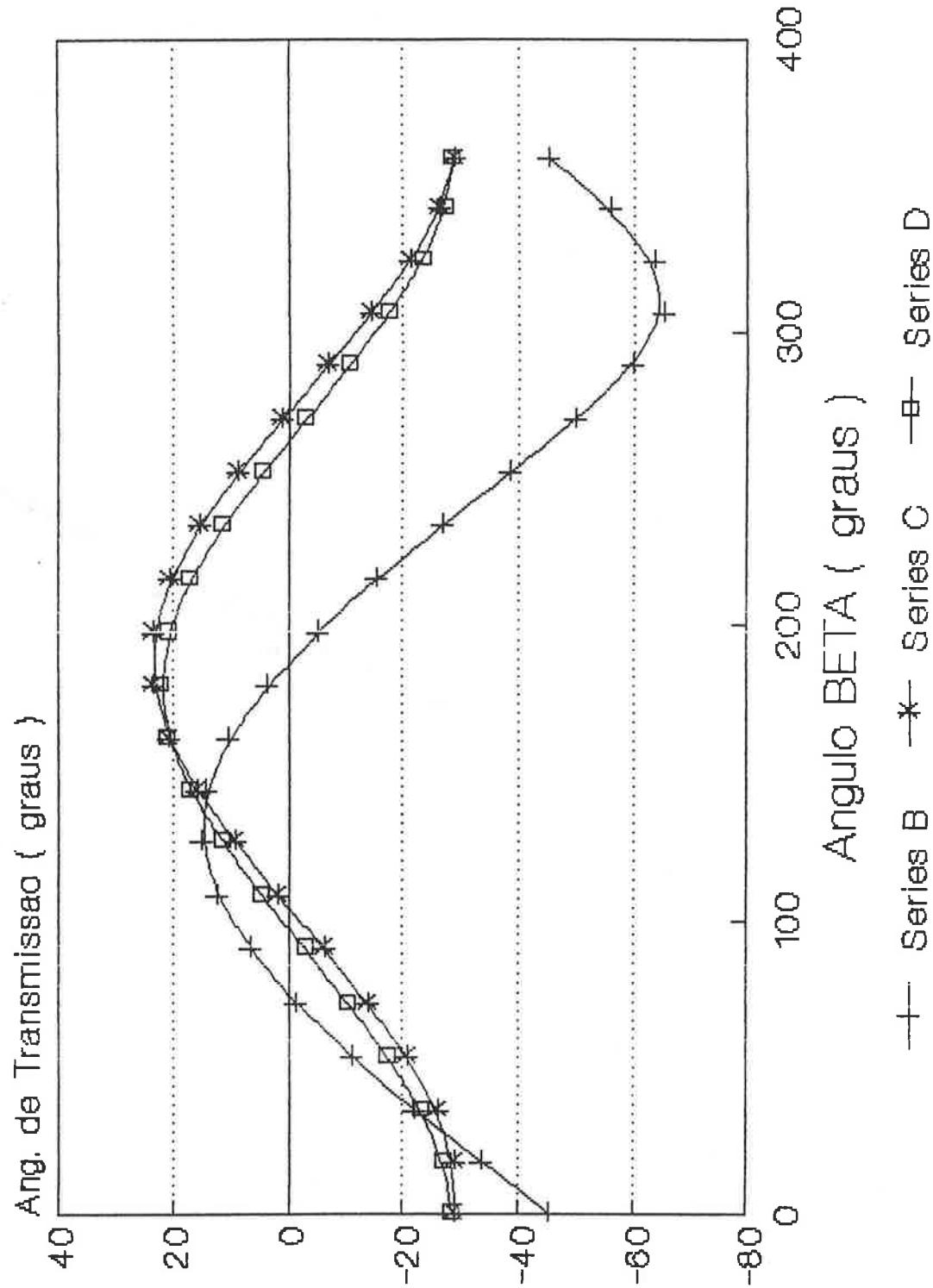


Fig. 7.11 - Gráfico comparativo do desvio do ângulo de transmissão das séries B, C e D em função de β .

7.3 - RESULTADOS DA OTIMIZAÇÃO DE 4 POSIÇÕES

7.3.1 - Síntese sem otimização

7.3.1.1 - Dados das posições de precisão

Tabela 7.21 - Posições do ponto P do acoplador.

P	x	y
1	0. 0000000	0. 0000000
2	4. 9999986	7. 9999982
3	9. 9999982	14. 999998
4	18. 000008	20. 000002

Tabela 7.22 - Ângulos de rotação do acoplador.

Angulo de rotação do acoplador (graus)	
α_2	9. 9999977
α_3	19. 999995
α_4	30. 000007

7.3.1.2 - Dados dos mecanismos obtidos

Tabela 7.23 - Articulações fixas e móveis e o ponto P do acoplador.

Pontos	x	y
A ₀	-20.19528	25.56607
B ₀	-29.16693	42.35537
A	-25.34925	25.37986
B	-37.08592	36.09333
P	0.000000	0.000000

Tabela 7.24 - Ângulos de rotação da alavanca.

Ângulos (graus)	1a. diáda	2a. diáda
β_2	-38.00000	-7.00000
β_3	-65.51381	-1.00010
β_4	-135.60274	34.39482

7.3.2 - Otimização com 2 variáveis livres

7.3.2.1 - Dados da função de otimização

Na tabela 7.25 apresenta-se os valores da função de otimização na primeira e última iteração.

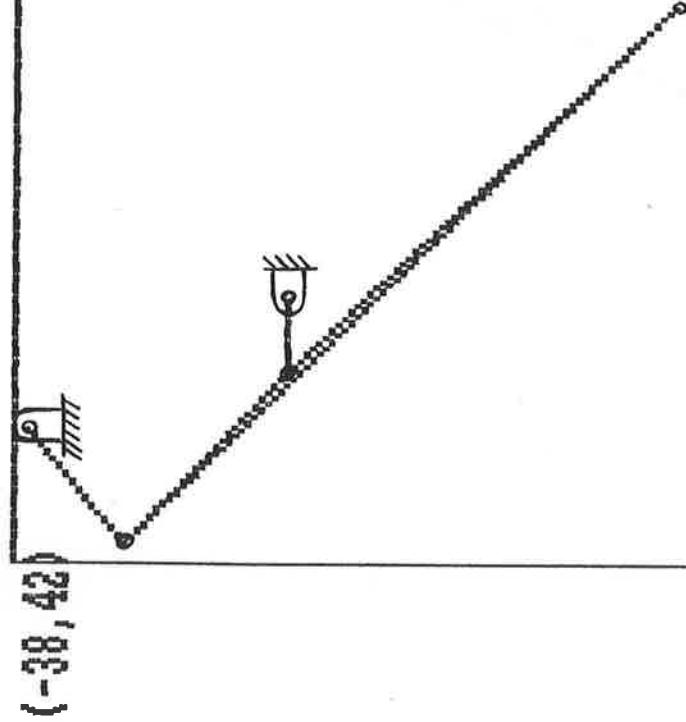


Fig. 7.12 - Mecanismo obtido conforme tabela 7.23.

(42, 42)

(42, -38)

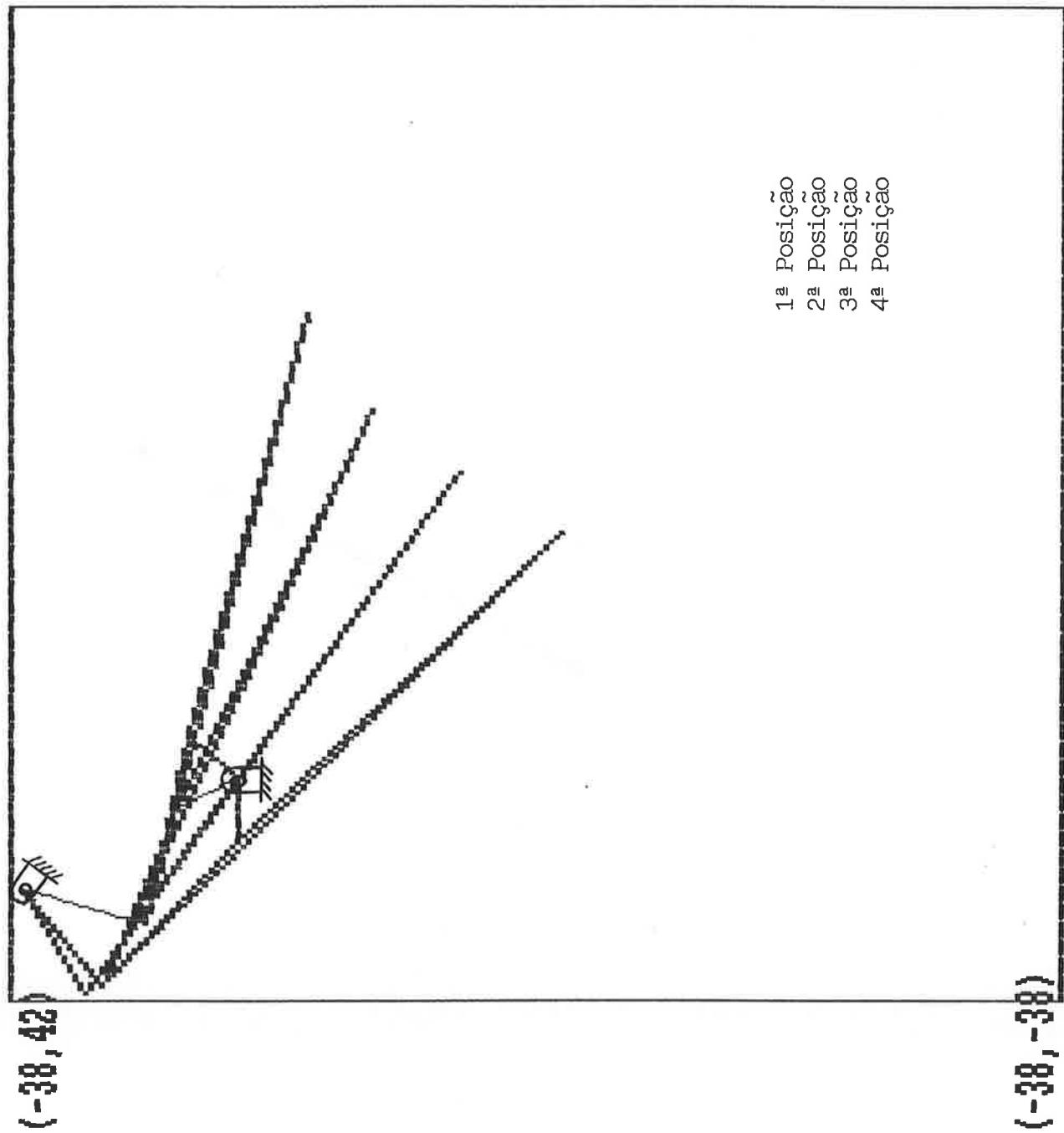


Fig. 7.13 - Posições de precisão do mecanismo obtido.

Tabela 7.25 - Valor da função na 1a. e última iteração.

Iteração	no. da iteração	Valor da função
Primeira	1	4.73997
Última	9	3.72821

7.3.2.2 - Dados das posições de precisão

Tabela 7.26 - Posições do ponto P do acoplador.

P	x	y
1	0.0000000	0.0000000
2	4.9999994	7.9999990
3	9.9999990	14.999999
4	18.000005	19.999995

Tabela 7.27 - Posições do ponto P do acoplador.

Erro posição de P (%)	x	y
1	—	—
2	0.00000	0.00000
3	0.00000	0.00000
4	0.00000	0.00000

Tabela 7.28 - Ângulos de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_2	9. 9999977
α_3	19. 999995
α_4	30. 000007

Tabela 7.29 - Erro no ângulo de rotação do acoplador.

Erro no ângulo de rotação do acoplador (%)	
α_2	0. 00000
α_3	0. 00000
α_4	0. 00000

7.3.2.3 - Dados dos mecanismos obtidos

Tabela 7.30 - Articulações fixas e móveis e ponto P do acoplador.

Pontos	x	y
Ao	-19. 06602	22. 45259
Bo	-28. 60216	43. 22653
A	-25. 20519	23. 10195
B	-41. 53913	33. 58838
P	0. 000000	0. 000000

(43, 43)

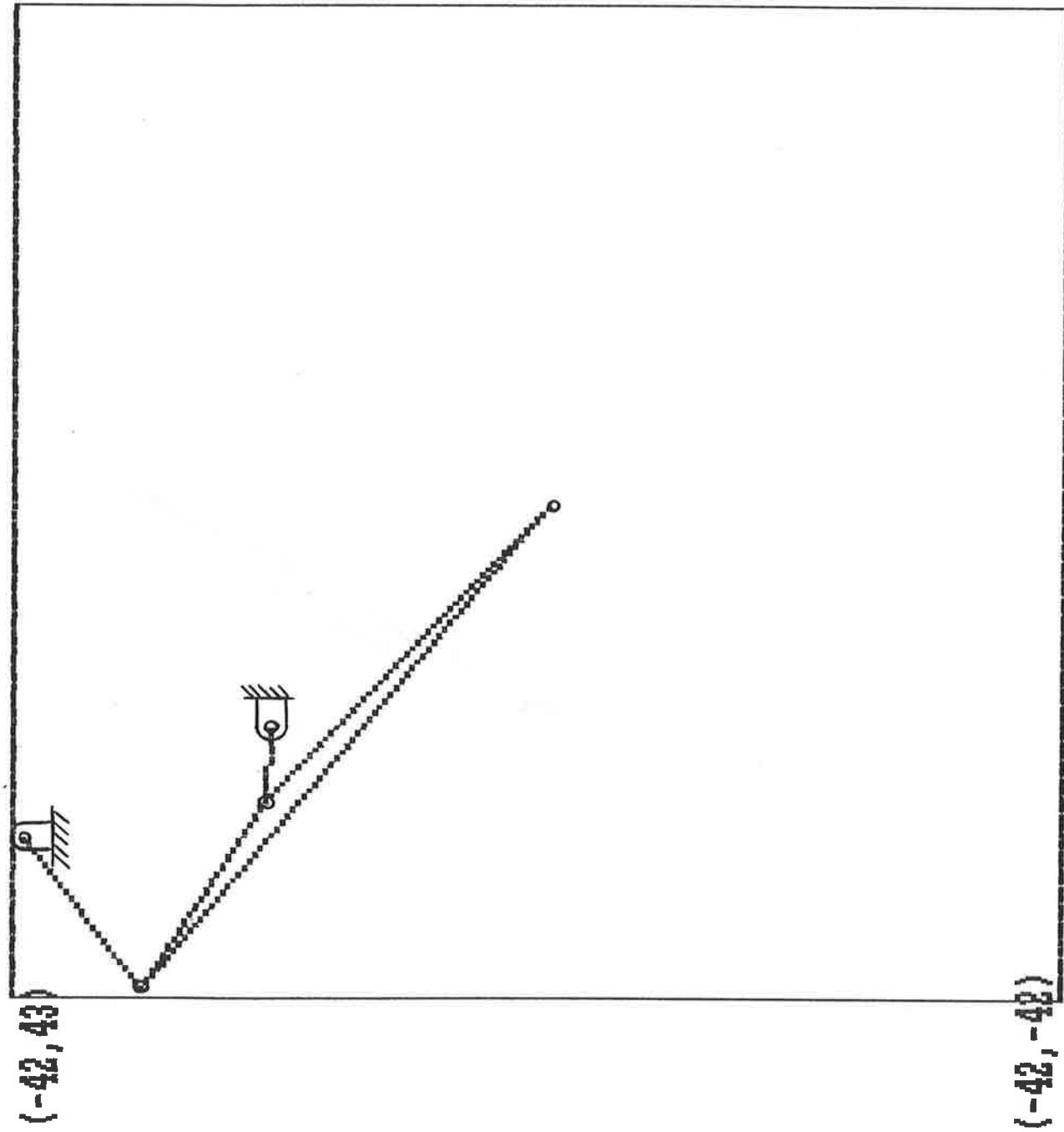


Fig. 7.14 - Mecanismo obtido conforme tabela 7.30.

(43, 43)

(-42, 43)

(-42, -42)

(43, -42)

- 1^a Posição
- 2^a Posição
- 3^a Posição
- 4^a Posição

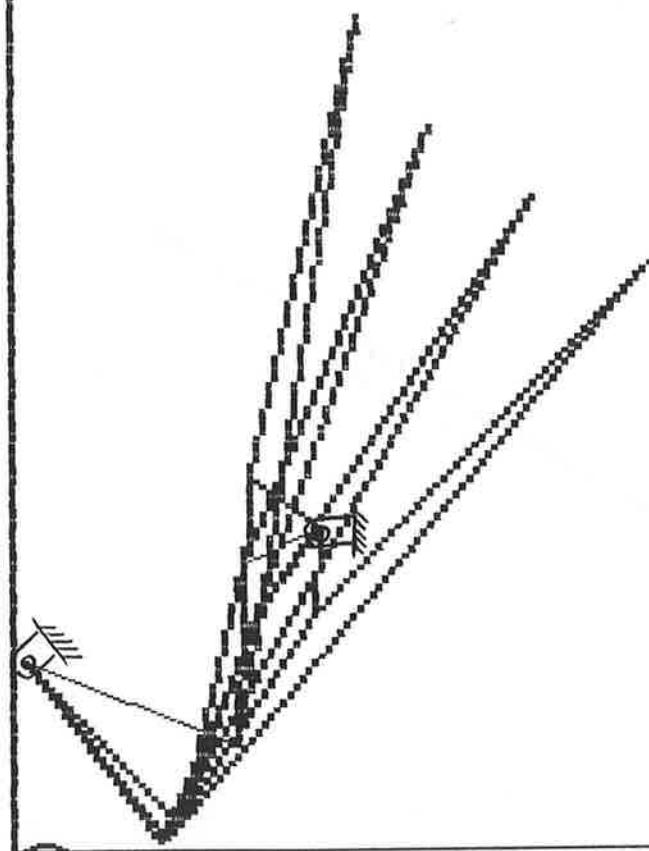


Fig. 7.16 - Posições de precisão do mecanismo obtido.

Tabela 7.31 - Ângulos de rotação da alavanca.

Ângulos (graus)	1a. diada	2a. diada
β_2	-33.39933	-1.21517
β_3	-59.86565	5.67887
β_4	-120.63115	30.84312

7.3.3 - Otimização com 13 variáveis

7.3.3.1 - Dados da função de otimização

Na tabela 7.32 apresenta-se os valores da função de otimização na primeira e última iteração.

Tabela 7.32 - Valor da função na 1a. e última iteração.

Iteração	no. da iteração	Valor da função
Primeira	1	4.73998
Última	10	2.94361

7.3.3.2 - Dados das posições de precisão

Tabela 7.33 - Posições de precisão do ponto P do acoplador.

P	x	y
1	-0.01009255	0.01222560
2	4.99756957	8.00043198
3	10.3678302	15.3386679
4	17.9716621	19.7706959

Tabela 7.34 - Posições do ponto P do acoplador.

Erro posição de P (%)	x	y
1	—	—
2	0.04861	0.00527
3	3.67830	2.25779
4	0.15743	1.14652

Tabela 7.35 - Ângulo de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_2	8.50221924
α_3	17.9998088
α_4	30.5533000

Tabela 7.36 - Erro no ângulo de rotação do acoplador.

Erro no ângulo de rotação do acoplador (%)	
α_2	14.9778
α_3	0.00106
α_4	1.84433

7.3.3.3 - Dados dos mecanismos obtidos

Tabela 7.37 - Articulações fixas e móveis e ponto P do acoplador.

Pontos	x	y
A _o	-27.7225978	24.2535276
B _o	12.9164594	59.4259081
A	-32.5760289	25.1048691
B	-35.4659017	41.7595451
P	-0.01009255	0.01222560

Tabela 7.38 - Ângulos de rotação da alavanca.

Ângulo (graus)	1a. diáda	2a. diáda
β_2	-39.587830	-2.6867320
β_3	-72.639448	-2.7328153
β_4	-163.392986	4.8918326

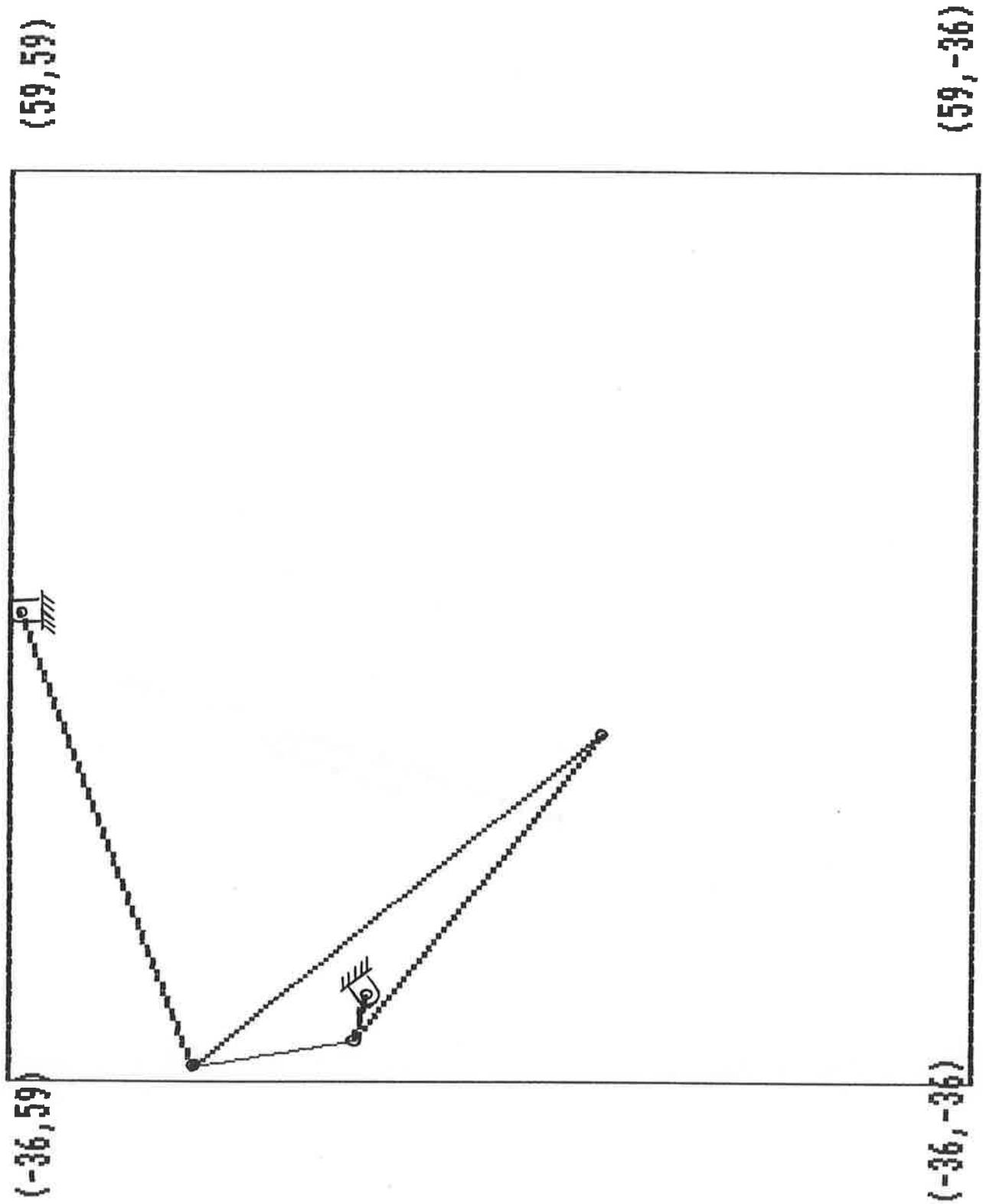


Fig. 7.16 – Mecanismo obtido conforme tabela 7.37.

(59, 59)

(-36, 59)

(-36, -36)

(59, -36)

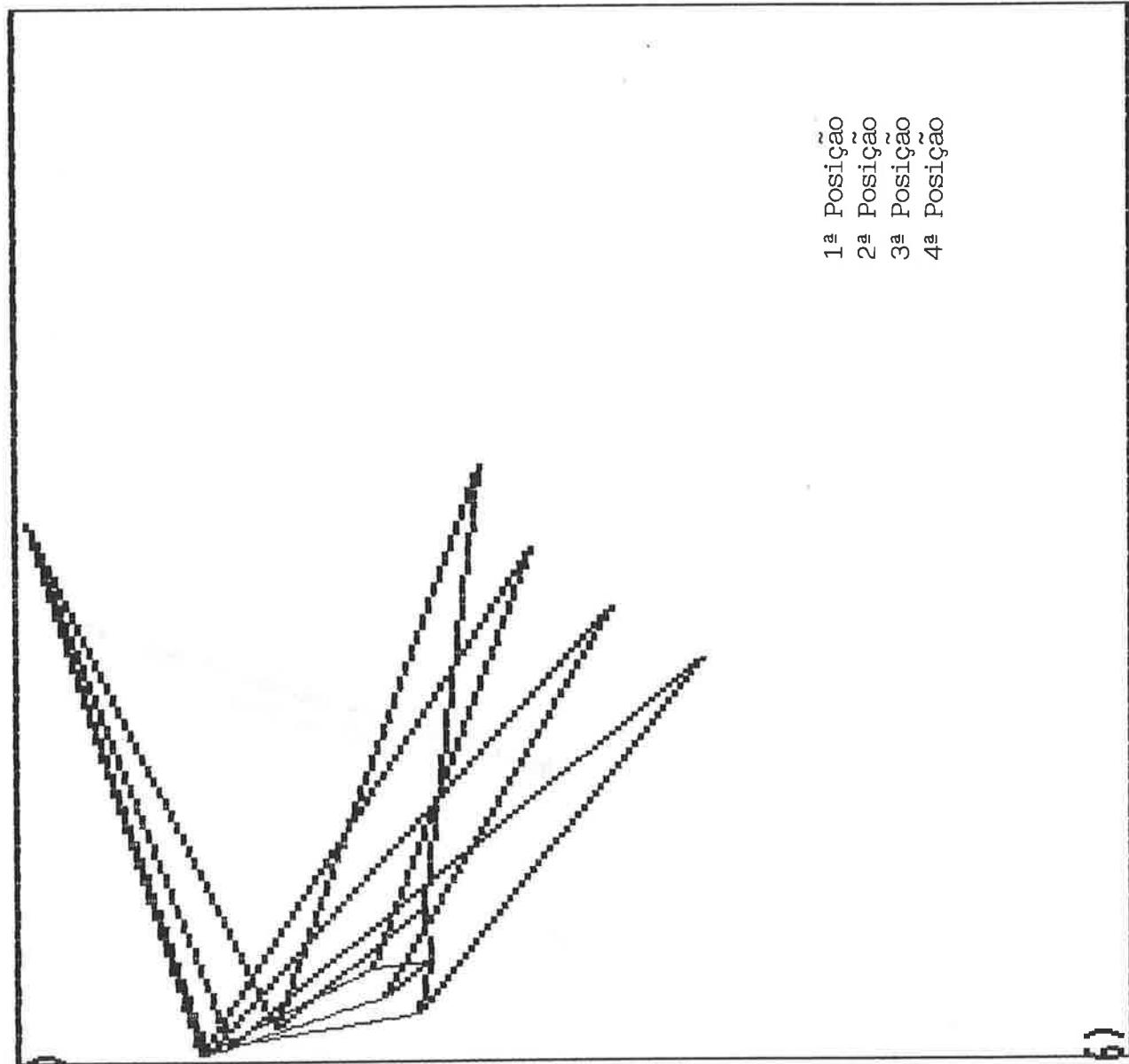


Fig. 7.17 - Posições de precisão do mecanismo obtido.

7.3.4 - Resultados do ângulo de transmissão

Apresentam-se, a seguir, as tabelas e os gráficos do ângulo de transmissão e desvio deste em torno de 90° em função de β para síntese sem otimização, com otimização de 2 variáveis e com otimização de 13 variáveis.

Tabela 7.39 - Ângulo de transmissão x β para síntese sem otimização (B), com otimização de 2 variáveis (C) e com otimização de 13 variáveis (D).

ângulo β (graus)	ângulo μ (graus)	ângulo μ (graus)	ângulo μ (graus)
0	80.73	69.39	100.22
18	91.25	77.23	103.93
36	102.32	85.39	106.22
54	113.23	93.23	106.79
72	123.19	100.10	105.60
90	131.10	105.40	102.77
108	135.52	108.52	98.63
126	135.26	109.07	93.61
144	130.40	106.97	88.19
162	122.21	102.50	82.84
180	112.11	96.20	78.06
198	101.14	88.68	74.31
216	90.11	80.58	72.01
234	79.67	72.52	71.43
252	70.59	65.20	72.64
270	63.76	59.37	75.49
288	60.11	55.83	79.65
306	60.32	55.20	84.69
324	64.34	57.60	90.12
342	71.46	62.58	95.46
360	80.73	69.39	100.22

Ang. de Transmissão - 4 pos.

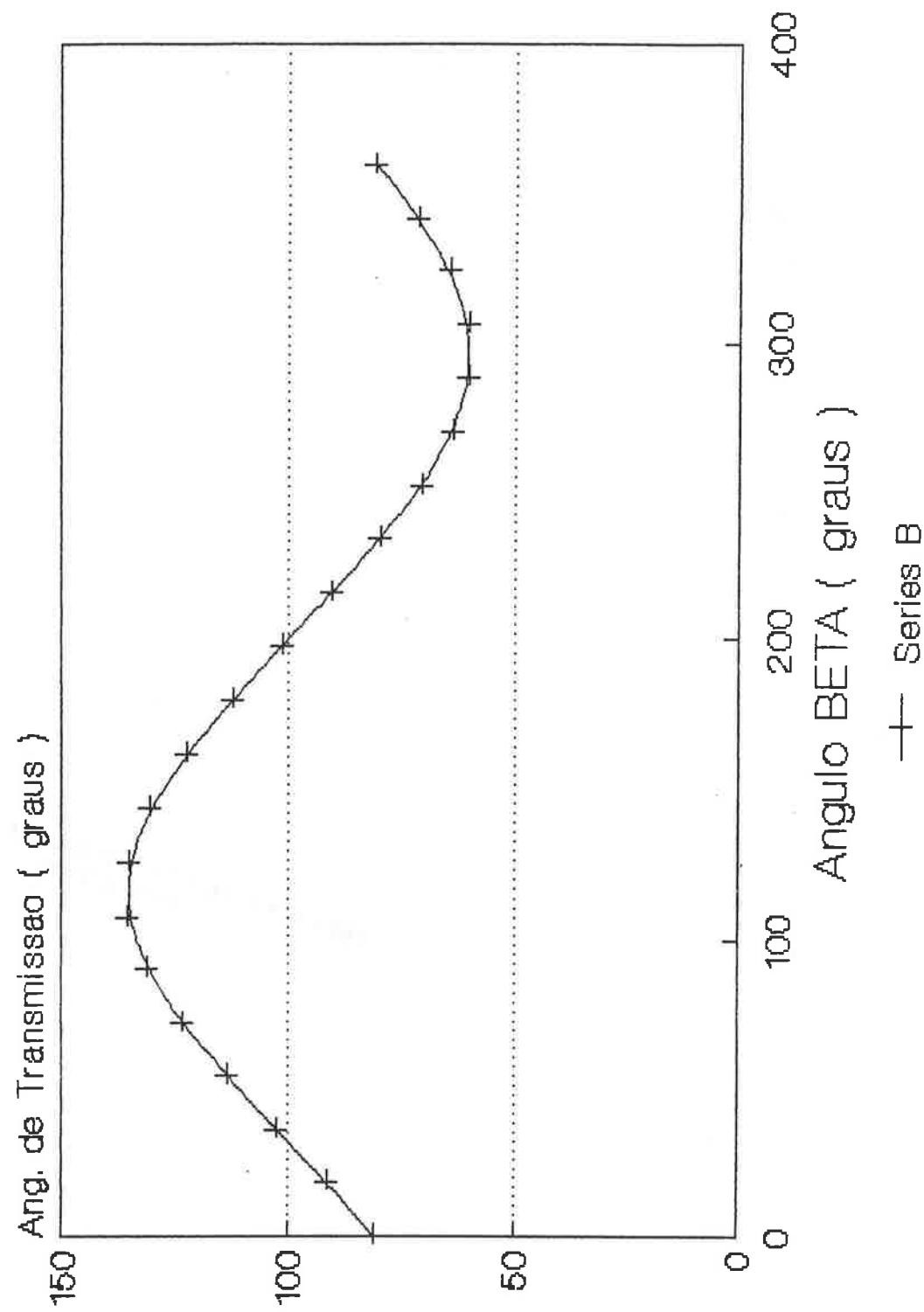


Fig. 7.18 - Gráfico do ângulo de transmissão μ , série B, em função do ângulo β .

Ang. de Transmissão - 4 pos.

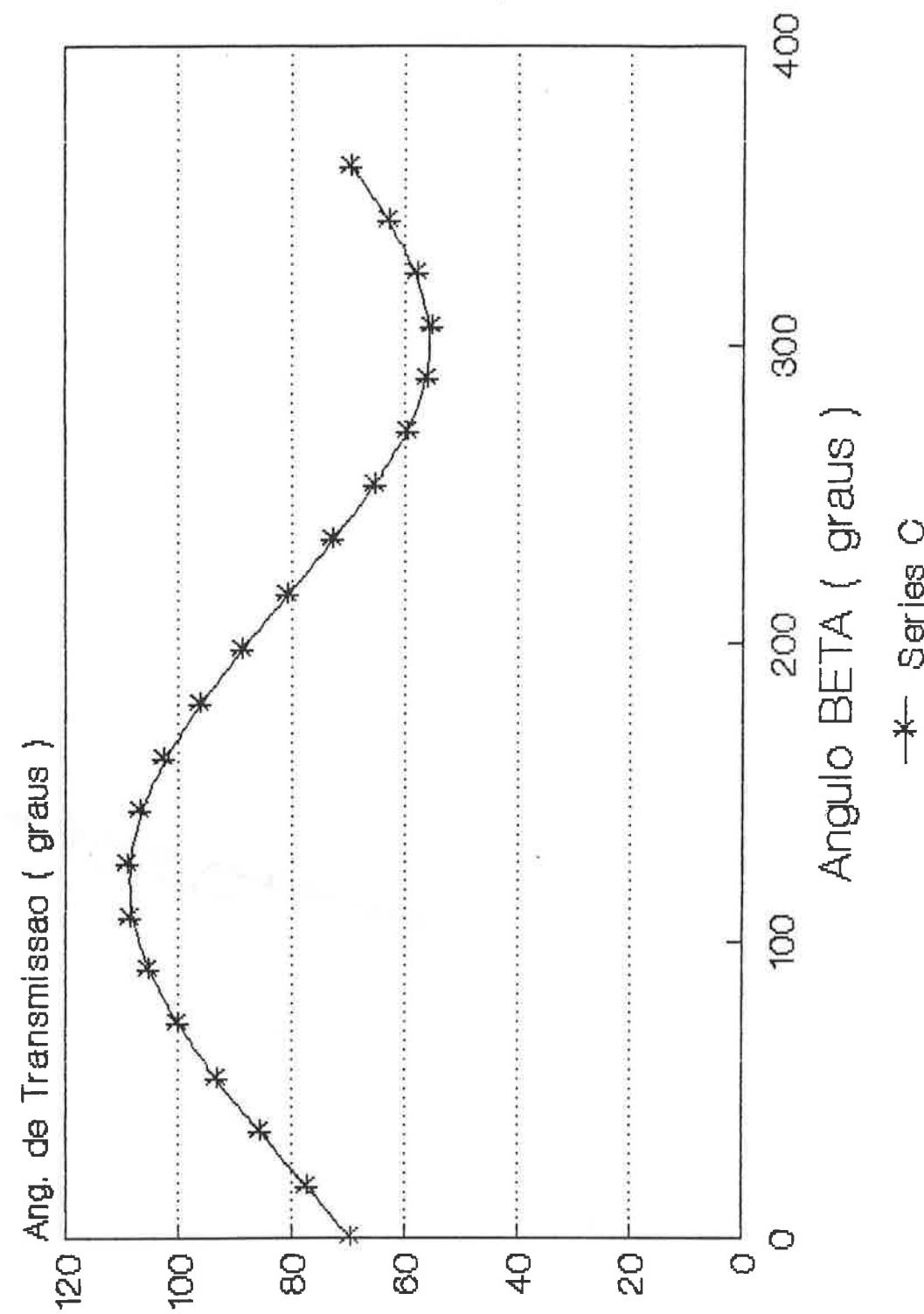


Fig. 7.10 - Gráfico do ângulo de transmissão μ , série C, em função do ângulo β .

Ang. de Transmissão - 4 pos.

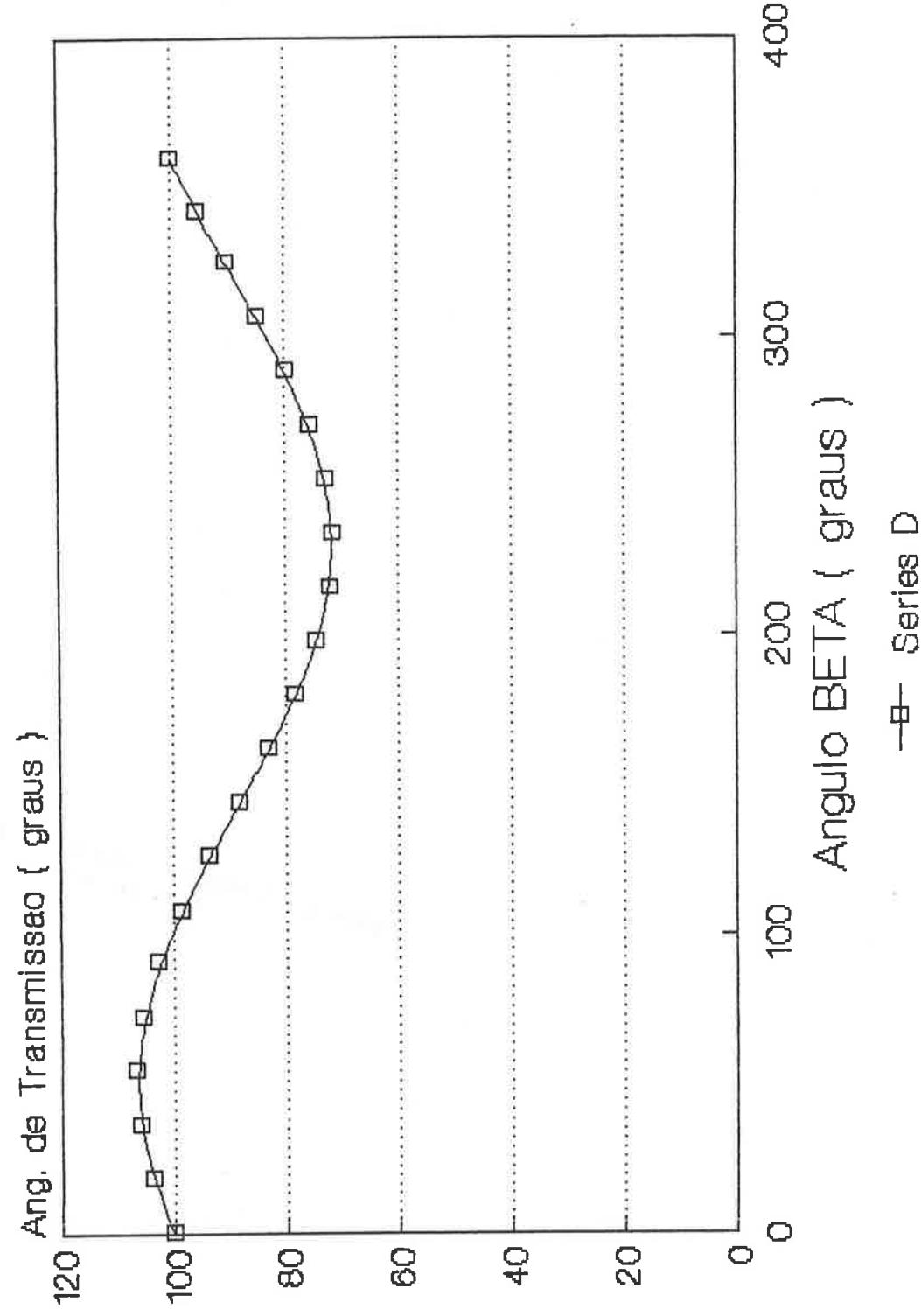


Fig. 7.20 - Gráfico do ângulo de transmissão μ , série D, em função do ângulo β .

Ang. de Transmissão - 4 pos.

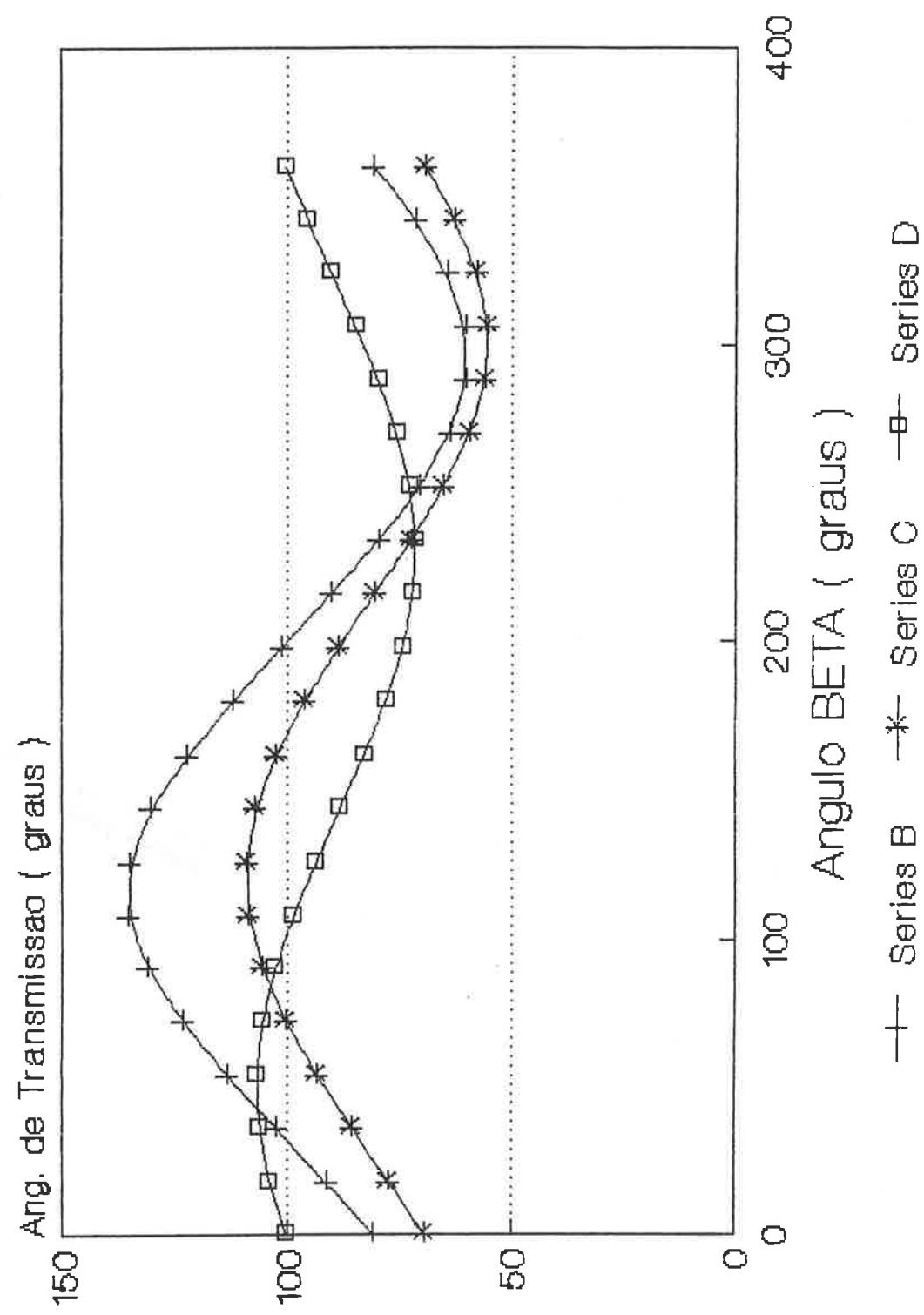


Fig. 7.21 - Gráfico comparativo do ângulo de transmissão das séries B, C e D em função de β .

Tabela 7.40 - Desvio do ângulo de transmissão $\times \beta$ sem otimização (B), com otimização de 2 variáveis (C) e com otimização de 13 variáveis (D).

ângulo β (graus)	desvio B μ (graus)	desvio C μ (graus)	desvio D μ (graus)
0	-9.27	-20.61	10.22
18	1.25	-12.77	13.93
36	12.32	-4.61	16.22
54	23.23	3.23	16.79
72	33.19	10.10	15.60
90	41.10	15.40	12.77
108	45.52	18.52	8.63
126	45.26	19.07	3.61
144	40.40	16.97	-1.81
162	32.21	12.50	-7.16
180	22.11	6.20	-11.94
198	11.14	-1.32	-15.69
216	0.11	-9.42	-17.99
234	-10.33	-17.48	-18.57
252	-19.41	-24.80	-17.36
270	-26.24	-30.63	-14.51
288	-29.89	-34.17	-10.35
306	-29.68	-34.80	-5.31
324	-25.66	-32.40	0.12
342	-18.54	-27.42	5.46
360	-9.27	-20.61	10.22

Desvio do Ang. de Transmissão - 4 pos.

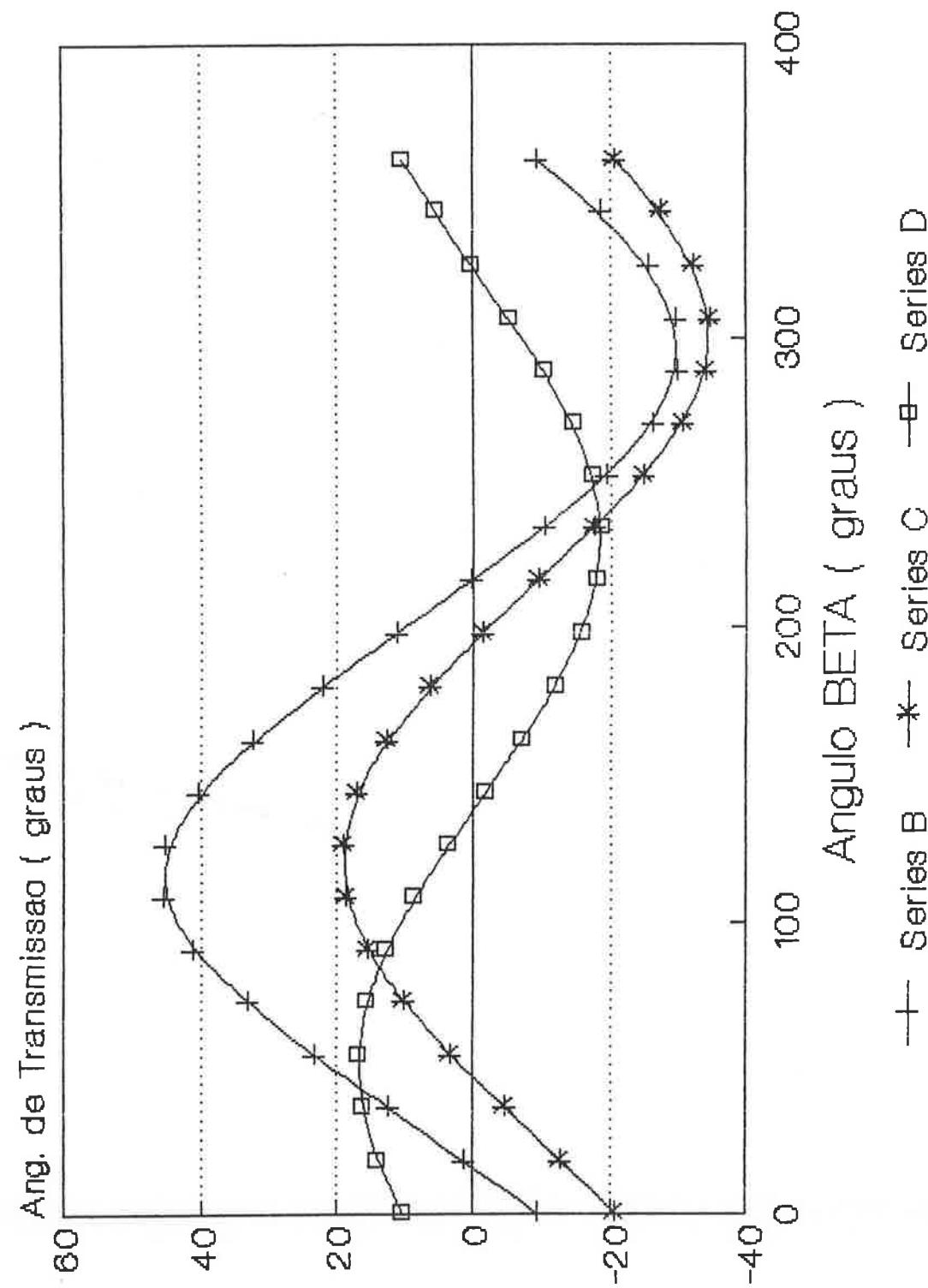


Fig. 7.22 - Gráfico comparativo do desvio do ângulo de transmissão das séries B, C e D em função de β .

7.4 - RESULTADOS DA OTIMIZAÇÃO COM ROTAÇÃO COMPLETA

7.4.1 - Síntese sem otimização

7.4.1.1 - Dados das posições de precisão

Tabela 7.41 - Posições do ponto P do acoplador.

P	x	y
1	-19.2100000	-1.62000000
2	-19.3711053	12.1760984
3	-13.3333191	54.6019961

Tabela 7.42 - Ângulo de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_z	-13.860000
α_s	-179.00996

7.4.1.2 - Dados dos mecanismos obtidos

Tabela 7.43 - Ângulos de rotação da alavanca.

Ângulo (graus)	1a. diáada	2a. diáada
β_z	-43.276053	-59.317292
β_s	-151.661594	124.942337

(55, 55)

(-39, 55)

(55, -39)

(-39, -39)

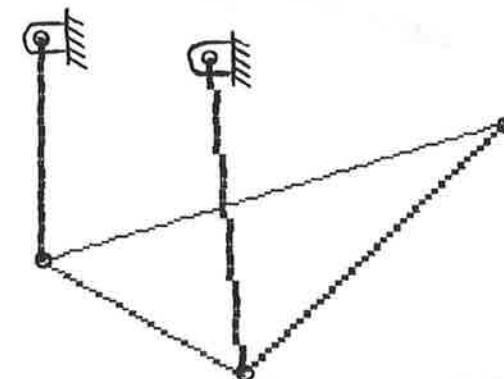
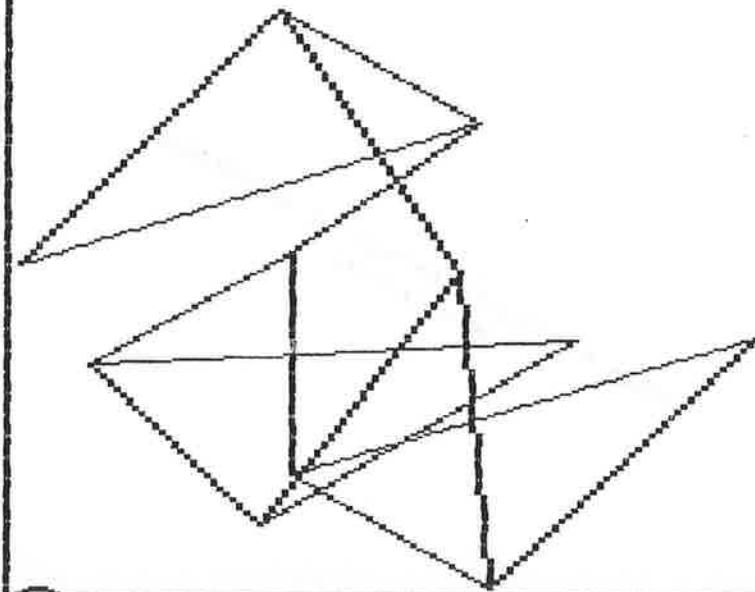


Fig. 7.23 - Mecanismo obtido conforme tabela 7.43.

(55, 55)

(-39, 55)



- 1^a Posição
- 2^a Posição
- 3^a Posição

(-39, -39)

Fig. 7.24 - Posições de precisão do mecanismo obtido.

Tabela 7.44 - Articulações fixas e móveis e ponto do acoplador.

Pontos	x	y
A _o	-14.000000	20.800000
B _o	-12.200000	33.900000
A	-39.3368471	18.3457851
B	-29.9975594	33.8130548
P	-19.2100000	-1.6200000

7.4.2 - Otimização com 4 variáveis

7.4.2.1 - Dados da função de otimização

Na tabela 7.45 apresenta-se os valores da função de otimização na primeira e última iteração.

Tabela 7.45 - Valor da função na 1a. e última iteração.

Iteração	no. da iteração	Valor da função
Primeira	1	406.457
Última	14	4.33038

7.4.2.2 - Dados das posições de precisão

Tabela 7.46 - Posições do ponto P do acoplador.

P	x	y
1	-19.2100000	-1.62000000
2	-19.3711051	12.1760967
3	-13.3333011	54.6019979

Tabela 7.47 - Erro nas posições do ponto P do acoplador.

Erro posição de P (%)	x	y
1	0.00000	0.00000
2	0.00003	0.00002
3	0.00000	0.00000

Tabela 7.48 - Ângulo de rotação do acoplador.

Ângulo de rotação do acoplador (graus)	
α_z	-13.860007
α_x	-179.01000

Tabela 7.49 - Erro nos ângulos de rotação do acoplador.

Erro no ângulo do acoplador (%)	
α_z	0. 00005
α_a	0. 00000

7.4.2.3 - Dados dos mecanismos obtidos

Tabela 7.50 - Articulações fixas e móveis e ponto do acoplador.

Pontos	x	y
Ao	-16. 0592800	23. 7999990
Bo	-9. 1999998	36. 9000020
A	-68. 3707062	27. 5167530
B	-54. 1254547	51. 8707533
P	-19. 2100000	-1. 6200000

Tabela 7.51 - Ângulos de rotação da alavanca.

Ângulo (graus)	1a. diáda	2a. diáda
β_z	-28. 777541	-30. 266270
β_a	-173. 182806	150. 454320

(72, 72)

(72, -68)

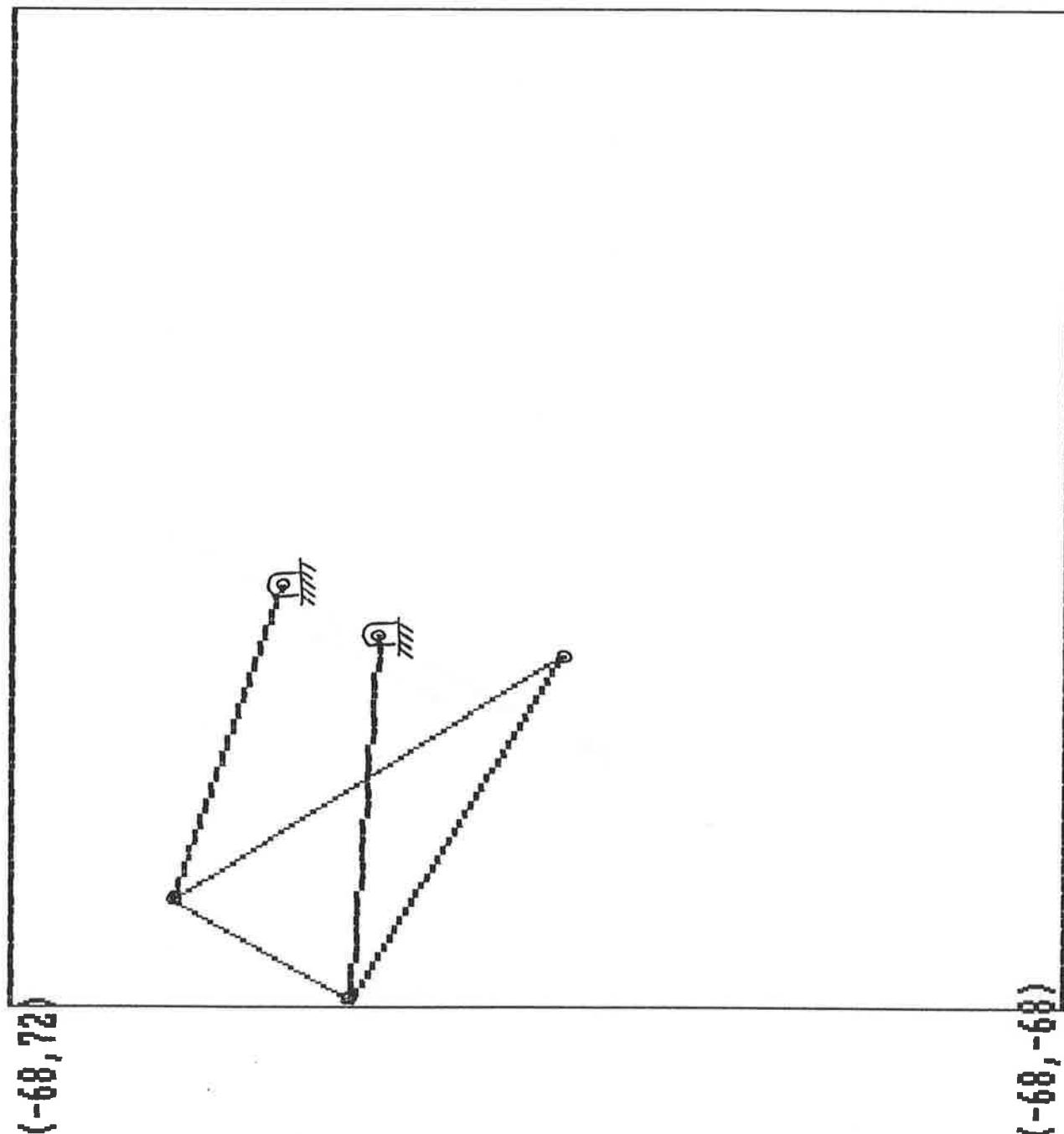
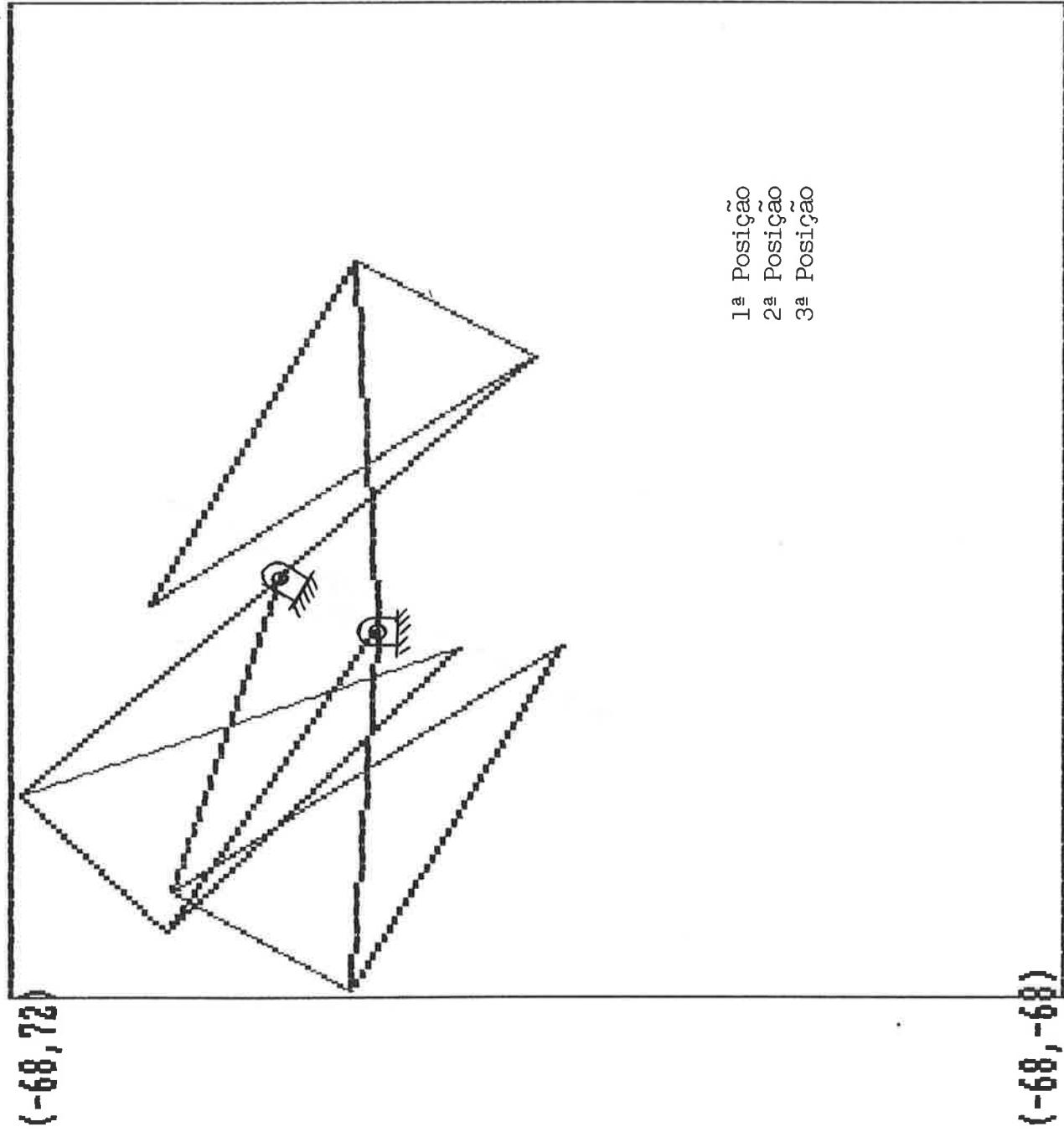


Fig. 7.26 - Mecanismo obtido conforme tabela 7.50.

(72, 72)



(72, -68)

(-68, 72)

(-68, -68)

Fig. 7.26 - Posições de precisão do mecanismo obtido.

7.4.3 - Resultados do ângulo de transmissão

Apresenta-se as tabelas e os gráficos do ângulo de transmissão e do desvio deste em torno de 90° em função de β para síntese sem otimização e com otimização de 4 variáveis.

Tabela 7.52 - Ângulo de transmissão x β para síntese com otimização de 4 variáveis (B) e sem otimização (C).

ângulo β (graus)	ângulo B μ (graus)	ângulo C μ (graus)
0	101.89	121.40
7.5	105.88	130.69
15	109.63	140.93
22.5	113.08	153.17
30	116.17	176.27
36	118.33	
54	122.74	
72	123.48	
90	120.40	
108	114.09	
123.16	107	178.26
125	106.06	166.76
126	105.54	163.96

Tabela 7.53 - Ângulo de transmissão x β para síntese com otimização de 4 variáveis (B) e sem otimização (C).

ângulo β (graus)	ângulo B μ (graus)	ângulo C μ (graus)
144	95.68	133.02
162	85.27	111.35
180	75.04	92.15
198	65.72	74.61
216	58.23	59
234	53.64	46.62
252	52.87	40.18
270	56.08	42.34
288	62.60	52.08
306	71.34	66.26
324	81.32	82.89
342	91.74	101.22
360	101.89	121.40

Ang. de Transm. - 3 pos

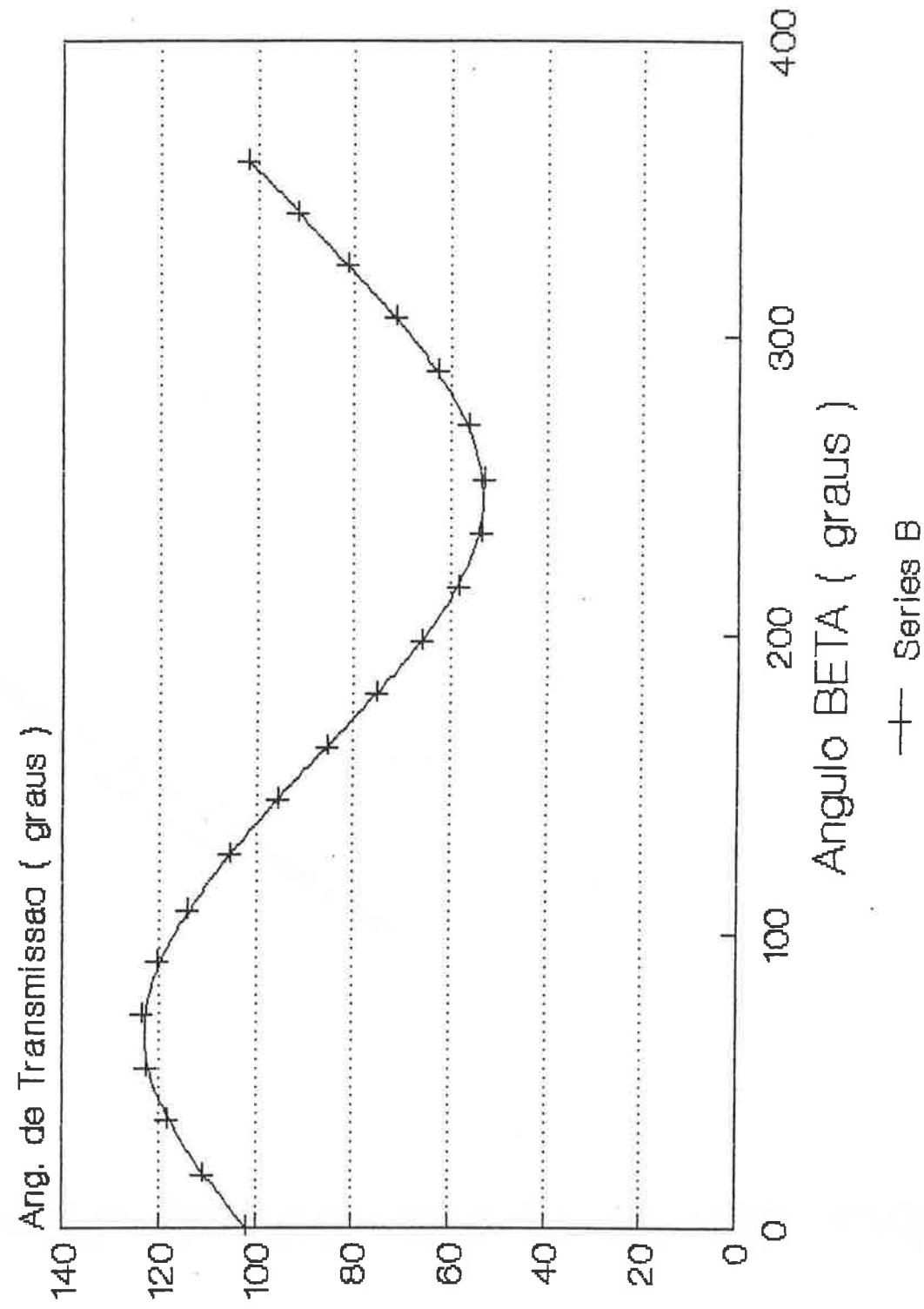


Fig. 7.27 - Gráfico do ângulo de transmissão μ , série B, em função do ângulo β .

Ang. de Transm. - 3 pos

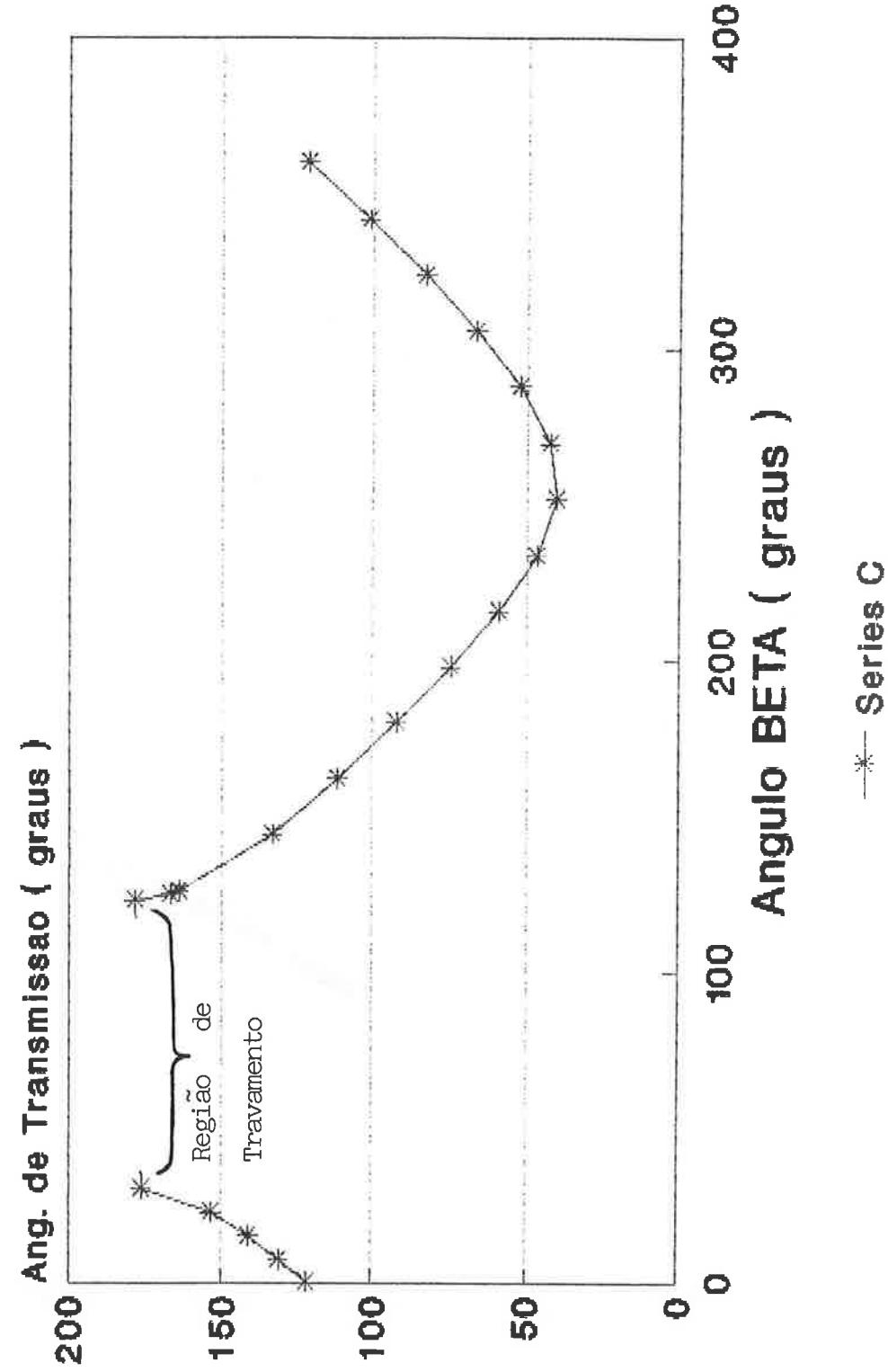


Fig. 7.28 - Gráfico do ângulo de transmissão μ , série C, em função do ângulo β .

Ang. de Transm. - 3 pos

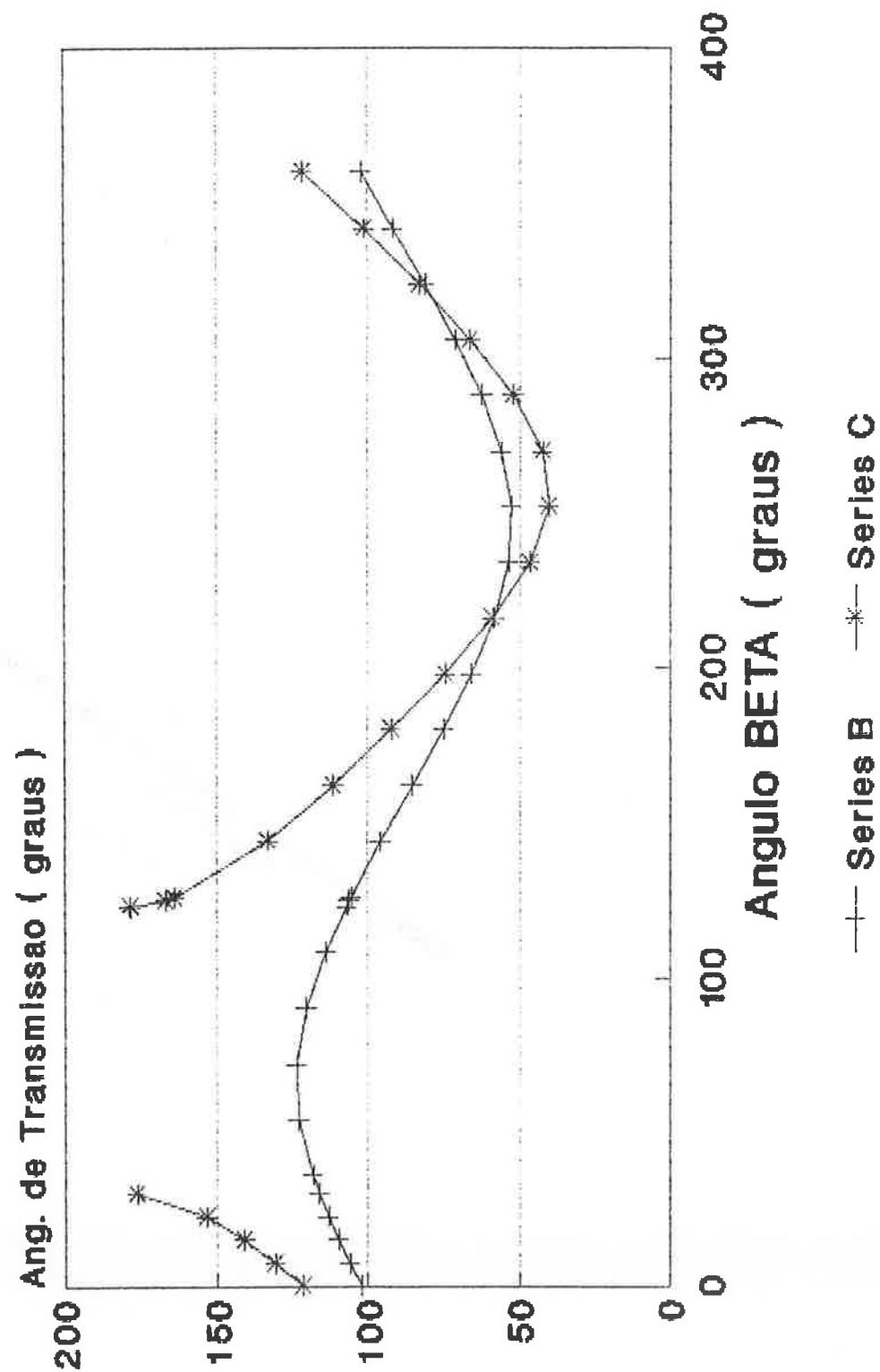


Fig. 7.29 - Gráfico comparativo do ângulo de transmissão das séries B e C em função de β .

Tabela 7.54 - Desvio do ângulo de transmissão x β para síntese com otimização de 4 variáveis (B) e sem otimização (C).

ângulo β (graus)	desvio B μ (graus)	desvio C μ (graus)
0	11.89	31.40
7.5	15.88	40.69
15	19.63	50.93
22.5	23.08	63.17
30	26.17	86.27
36	28.33	
54	32.74	
72	33.48	
90	30.40	
108	24.09	
123.16	17	88.26
125	16.06	76.76
126	15.54	73.96

Tabela 7.55 - Desvio do Ângulo de transmissão x β para síntese com otimização de 4 variáveis (B) e sem otimização (C).

Ângulo β (graus)	desvio B μ (graus)	desvio C μ (graus)
144	5.68	43.02
162	- 4.73	21.35
180	- 14.96	2.15
198	- 24.28	- 15.39
216	- 31.77	- 31
234	- 36.36	- 43.38
252	- 37.13	- 49.82
270	- 33.92	- 47.66
288	- 27.40	- 37.92
306	- 18.66	- 23.74
324	- 8.68	- 7.11
342	1.74	11.22
360	11.89	31.40

7.5 - SYN 3

- Programa para síntese de três posições de precisão.

- Parâmetros de entrada :

a) ângulos do acoplador : α_2 , α_3 ,

b) posições do ponto P : P_1 , P_2 , P_3 ,

c) articulações fixas : A_o , B_o .

Desvio do Ang. de Transm. - 3 pos

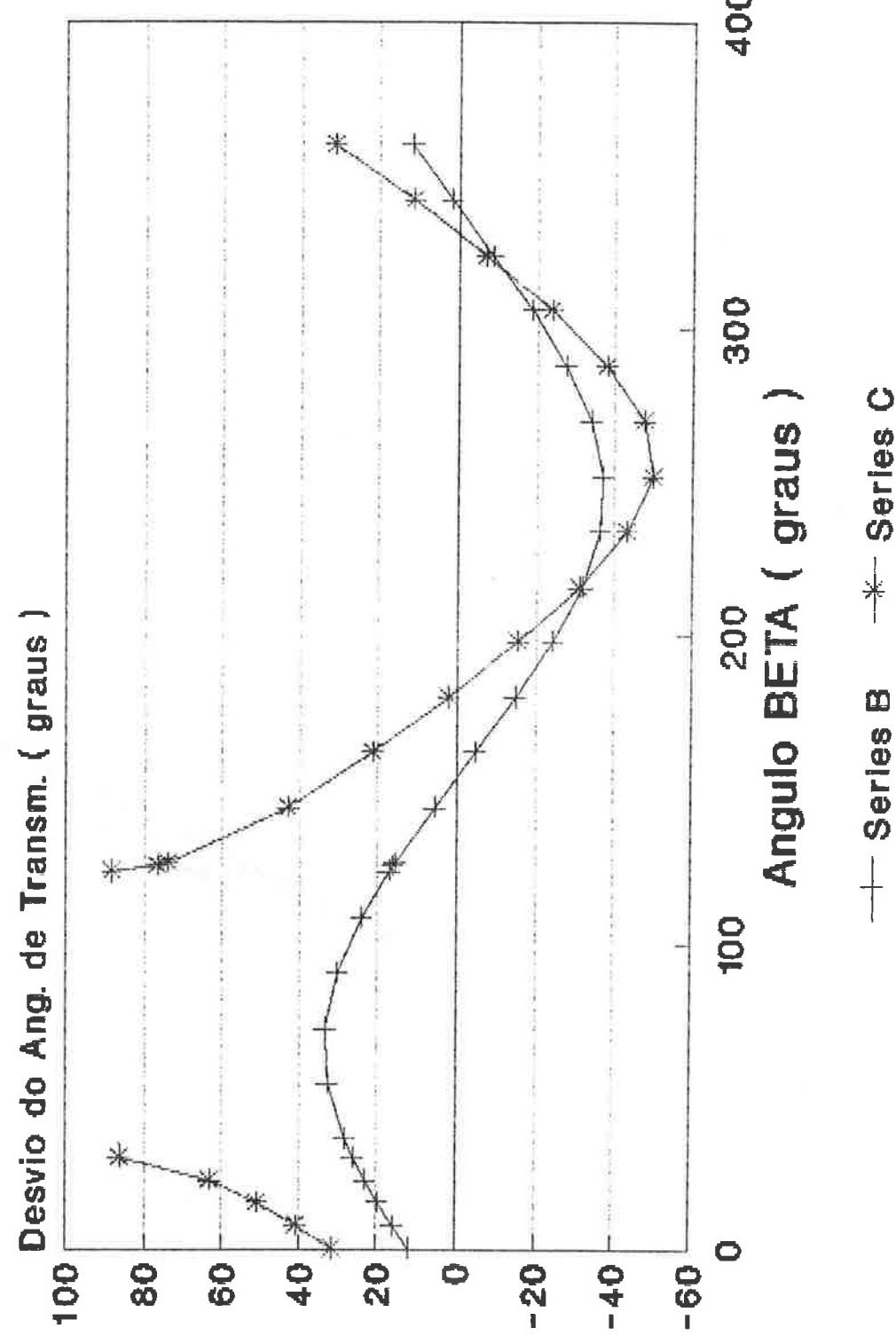


Fig. 7.30 - Gráfico comparativo do desvio do ângulo de transmissão das séries B e C em função de β .

- Parâmetros de saída :

a) articulações móveis : A , B.

7.6 - SYN 4

- Programa para síntese de quatro posições de precisão.

- Parâmetros de entrada :

a) ângulos do acoplador : α_2 , α_3 , α_4 ,

b) posições do ponto P : P_1 , P_2 , P_a , P_4 ,

c) variação do ângulo do seguidor : β_{z1} e β_{z2} .

- Parâmetros de saída :

a) articulações fixas : A_o , B_o e

b) articulações móveis : A, B.

7.7 - POS

- Programa para análise do quadrilátero articulado

obtido na síntese.

- Parâmetros de entrada :

- a) articulações fixas : A_o , B_o ,
- b) articulações móveis : A, B,
- c) ponto P,
- d) número de posições do movimento do quadrilátero e
- e) variações inicial e final do ângulo da manivela.

- Parâmetros de saída :

Em cada posição do movimento o programa fornece :

- a) posição das articulações móveis e do ponto P,
- b) variação do ângulo da manivela e do seguidor,
- c) ângulo de transmissão e
- d) ângulo do acoplador.

7.8 - OPT3

- Programa para determinação dos máximos e mínimos de funções.

- Parâmetros de entrada :

- a) função objetivo ,
- b) restrições de igualdade e desigualdade e
- c) limites inferiores e superiores das variáveis da função .

- Parâmetros de saída :

- a) valor do mínimo da função ,
- b) valor das variáveis para o ponto de mínimo e
- c) valor das restrições no ponto de mínimo .

7.9 - Linguagens de programação

- Fortran 77 , versão 4.01 da MICROSOFT .
- Pascal , Turbo Pascal 5.0 da BORLAND .

7.10 - Computador

- Microcomputadores PROCEDA tipo IBM PC XT/AT .

8. DISCUSSÃO

8.1 - Introdução

Esta discussão, primeiramente, analisa os resultados da otimização para três posições de precisão, para quatro posições de precisão e para se obter um mecanismo com rotação completa da manivela motora.

São apresentadas considerações importantes quanto ao número de pontos e as posições destes pontos para o cálculo da função objetivo de otimização. Também são apresentados comentários com relação aos exemplos e ao método escolhido para a síntese.

8.2 - Otimização para três posições de precisão

O que se pode observar do mecanismo obtido na síntese por Suh e Radcliffe [6] é que este mecanismo possui um ângulo de transmissão mínimo da ordem de 25° e máximo da ordem de 105° conforme tabela 7.1.9 e figura 7.7 . Como o valor recomendado na literatura para fins práticos [1, 24, 26] está num intervalo entre 45° e 135° , então este valor mínimo para μ é um resultado ruim . É fácil perceber que este mecanismo realiza rotação completa da manivela motora e foi utilizado como estimativa inicial para otimização.

Como se pode observar pela figura 7.8 o mecanismo obtido

na otimização com apenas quatro variáveis é muito superior. No caso da otimização com 12 variáveis o resultado obtido não apresentou um melhoramento significativo comparando-se com a otimização de 4 variáveis (vide figuras 7.9 e 7.10) . Isto porque a faixa de variação dos parâmetros independentes foi mantida estreita o suficiente para que o erro das posições de precisão e dos ângulos do acoplador estivesse dentro da faixa de tolerância especificada no item 6.2.2. Num problema em que as posições de precisão tivessem uma folga maior, a otimização com 12 variáveis certamente produzirá resultados superiores em relação à otimização com 4 variáveis.

8.3 - Otimização para quatro posições de precisão

Neste caso, o mecanismo obtido por Erdman e Gustafson [10] não necessitaria de otimização porque o ângulo μ se encontra dentro do intervalo recomendado. Inclusive este mecanismo também realiza rotação completa da manivela motora. Contudo, escolheu-se este para se verificar qual melhoramento seria possível obter com o programa de otimização com relação à distribuição do ângulo de transmissão com a rotação da manivela motora.

Como se pode observar pelas figuras 7.21 e 7.22 a otimização com duas variáveis produziu um resultado superior ao mecanismo obtido pela síntese sem otimização. Mas o resultado mais impressionante ficou para a otimização com 13 variáveis. Neste caso o desvio do ângulo de transmissão em torno de 90° foi muito

pequeno, da ordem de 20° . O erro nas posições está dentro da faixa de tolerância especificada no item 6.3.2.

8.4 - Otimização para rotação completa

Tomou-se, neste caso, o mecanismo apresentado por Peterson e colaboradores [13] como solução para um problema de síntese. O mecanismo sofre travamento após passar pelas posições de precisão. A solução apresentada pelos autores para resolver esse problema foi acrescentar mais duas barras ao mecanismo transformando o quadrilátero articulado num mecanismo de seis barras. Desta forma, uma dessas barras funcionaria como uma manivela o que é extremamente interessante porque é simples conseguirmos um movimento de rotação completa na entrada para o mecanismo. Assim, o mecanismo resultante poderia ser imaginado como dois quadriláteros articulados. O primeiro, já determinado, realizaria uma geração de movimento e o outro, uma geração de função. O primeiro seria um mecanismo de duas alavancas e o outro um mecanismo manivela-alavanca. Os inconvenientes deste tipo de solução são os seguintes:

a) São resolvidos dois problemas de síntese para se obter o mecanismo que atenda às especificações de projeto : um de geração de movimento e outro de geração de função.

b) Ao se acrescentar mais duas barras surgem dois quadriláteros articulados e com eles, dois novos ângulos de transmissão. Isso aumenta a complexidade do problema.

c) Ao se realizar as duas sínteses propostas por esses autores, não se garante nada quanto ao ângulo de transmissão. Ou seja, pode-se obter um mecanismo que atenda às especificações de projeto, cujo movimento de entrada é rotativo mas que tenha ângulos de transmissão ruins.

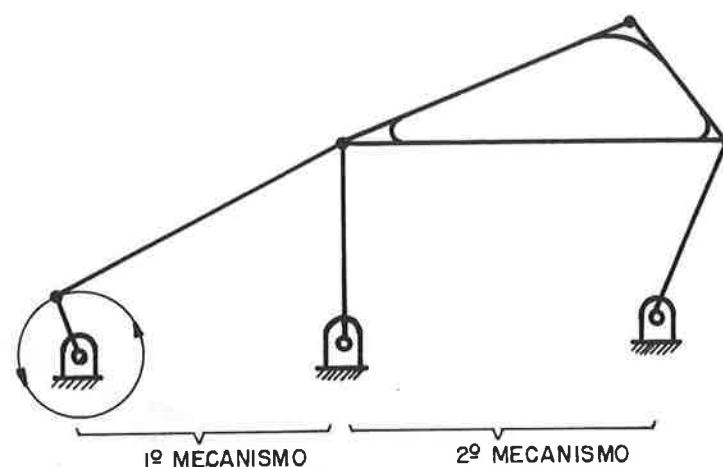


Fig. 8.1 - Mecanismo de 6 barras.

Utilizou-se para otimização a sub-rotina de três posições com uma pequena alteração na função objetivo para que o resultado fosse um mecanismo que realizasse uma rotação completa com ângulo de transmissão ótimo. Esta função objetivo funciona como uma função penalizadora, ou seja, se ocorrer um travamento durante a operação do mecanismo, será atribuído um valor elevado para a função. Desta maneira, como pretende-se obter um mínimo, a sub-rotina de otimização levará a busca para longe desta condição de travamento e muito provavelmente para uma condição de rotação completa da manivela motora. Acredita-se que o programa de

otimização garantirá esse objetivo a menos de uma estimativa inicial ruim.

Como se pode observar pelas figuras 7.29 e 7.30, o mecanismo obtido com a otimização realiza rotação completa e possui ângulos de transmissão dentro do intervalo desejado.

8.5 - Considerações quanto ao número de pontos e a posição destes pontos no cálculo da função objetivo

A função objetivo, conforme apresentado no capítulo 5, é uma somatória do quadrado do desvio dos ângulos de transmissão em torno de 90° . Essa somatória foi calculada inicialmente em 10 pontos igualmente espaçados, sendo cada ponto correspondente à rotação da manivela motora a partir da posição inicial do mecanismo.

Existem dois aspectos a serem considerados para a escolha dos pontos onde a função objetivo é calculada. O primeiro é que o mecanismo pode sofrer travamento na sua movimentação e, portanto, deve existir pelo menos uma região (intervalo entre dois ângulos de manivela) onde o ângulo de transmissão não é definido incluindo 0° ou 180° . Deve-se então escolher um ou mais pontos dentro dessa região, de modo a considerar o travamento do mecanismo no cálculo da função objetivo. É importante lembrar que, como foi mencionado no capítulo 5, se um ponto cair nessa região a

função objetivo assumirá um valor elevado. Caso se esteja analisando um mecanismo que trave e nenhum dos pontos escolhidos cair na região de travamento, então o programa de otimização vai entender que esse mecanismo é bom e não será penalizado.

Para resolver esse problema, analisou-se a equação (5.3) que representa a relação entre o ângulo de transmissão com o ângulo de rotação da alavanca motora e os comprimentos das barras do mecanismo.

$$\cos \mu = \frac{z_3^2 + z_5^2 - z_1^2 - z_6^2 + 2 z_1 z_6 \cos \phi}{2 z_3 z_5}$$

pode-se chamar

$$A = (z_3^2 + z_5^2 - z_1^2 - z_6^2) / (2 z_3 z_5) \quad \text{e}$$

$$B = (z_1 z_6) / (z_3 z_5) \quad \text{então,}$$

$$\cos \mu = A + B \cos \phi \quad (8.1)$$

A expressão de $\cos \mu$, por ser um cosseno, deve estar entre -1 e 1

$$-1 < \cos \mu < 1 \quad (8.2)$$

O intervalo foi considerado aberto porque quando μ for

igual a 0° ou 180° existe também travamento. Esse intervalo representa a região de movimentação do mecanismo. Substituindo (8.1) em (8.2) vem:

$$-1 < A + B \cos \phi < 1$$

$$-A - 1 < B \cos \phi < 1 - A$$

$$\frac{-A - 1}{B} < \cos \phi < \frac{1 - A}{B} \quad (8.3)$$

A desigualdade (8.3) representa o ângulo da alavanca motora em que é possível a movimentação do mecanismo sem travamento. Chamando

$$\frac{-A - 1}{B} = \cos \phi_2 = k_2 \quad \text{e}$$

$$\frac{1 - A}{B} = \cos \phi_1 = k_1$$

pode-se analisar a região de movimentação e travamento através do círculo trigonométrico.

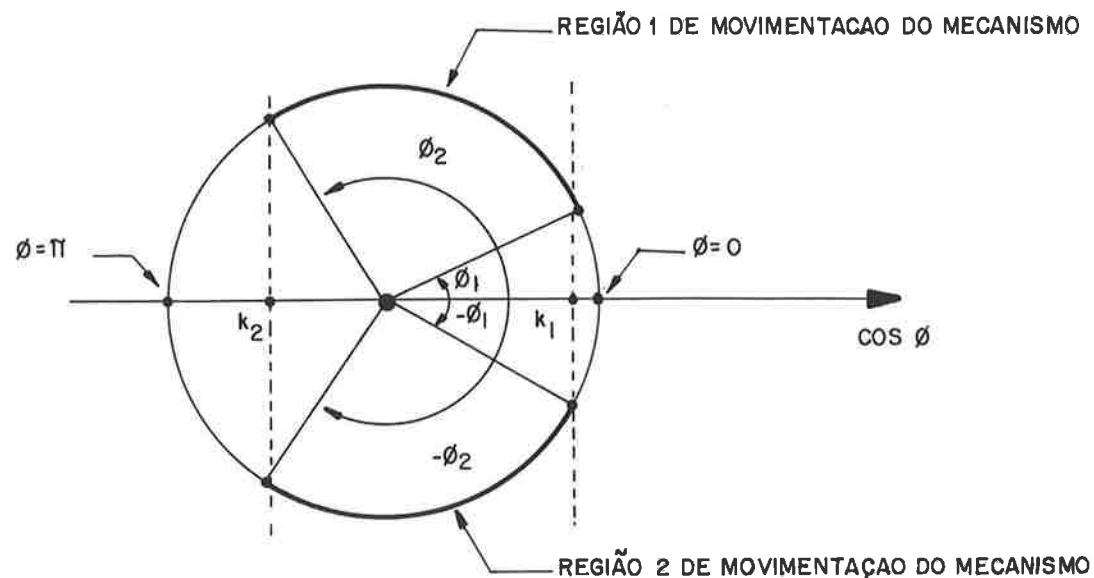


Fig. 8.2 - Círculo trigonométrico.

Pela figura 8.2 pode-se perceber que existem duas regiões de movimentação do mecanismo, a saber

$$(1) \phi_1 < \phi < \phi_2$$

$$(2) -\phi_2 < \phi < -\phi_1$$

Ao mesmo tempo, existem duas regiões de travamento do mecanismo, a saber

$$(1) \phi_2 \leq \phi \leq -\phi_2$$

$$(2) -\phi_1 \leq \phi \leq \phi_1$$

Pela figura 8.2 percebe-se que os centros dos intervalos

de travamento (1) e (2) são, respectivamente, 180° e 0° . Logo, a escolha dos pontos para o cálculo do ângulo de transmissão deve ser tal que, pelo menos os ângulos 0° e 180° estejam incluídos. No caso do quadrilátero articulado do tipo "retângulo", a região de travamento se reduz a dois pontos 0° e 180° , constituindo-se esse mecanismo num caso limite.



Fig. 8.3 - Quadrilátero articulado do tipo "retângulo".

O segundo aspecto a ser considerado é o número de pontos mínimo a ser escolhido de modo a que haja uma boa representatividade da função objetivo.

Admitindo que não haja travamento, o gráfico de $\cos \mu \times \phi$ é o apresentado na figura 8.4, o de $\mu \times \cos \mu$ é o apresentado na figura 8.5 e o de $\mu \times \phi$ é o apresentado na figura 8.6.

No caso do desvio do ângulo de transmissão em torno de 90° em função de ϕ o gráfico é o apresentado na figura 8.7. O desvio quadrático de $\mu \times \phi$ é o apresentado na figura 8.8.

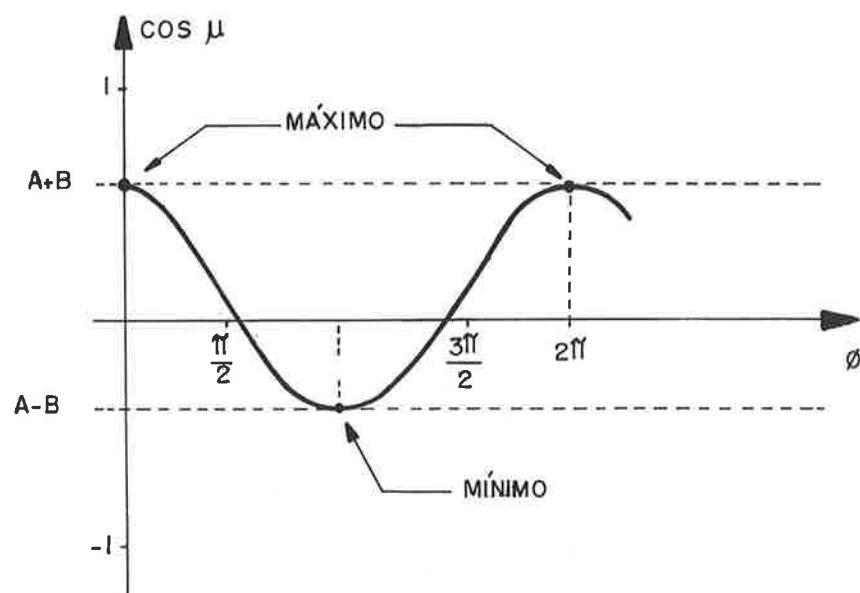


Fig. 8.4 - Gráfico de $\cos \mu \times \phi$.

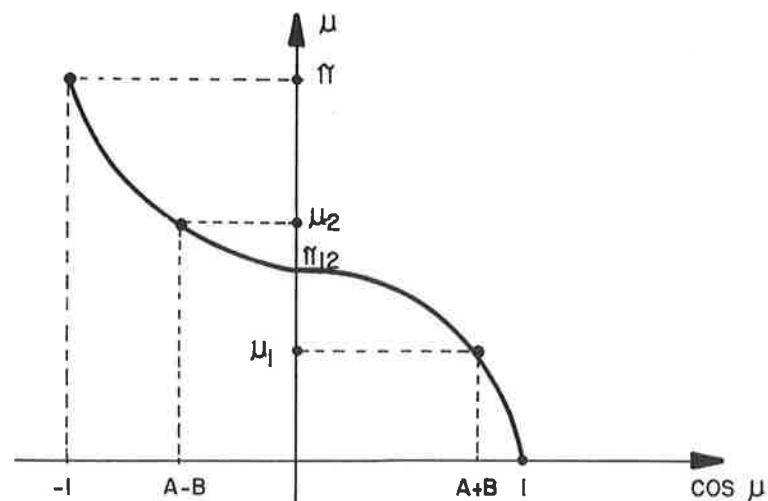


Fig. 8.5 - Gráfico de $\mu \times \cos \mu$.

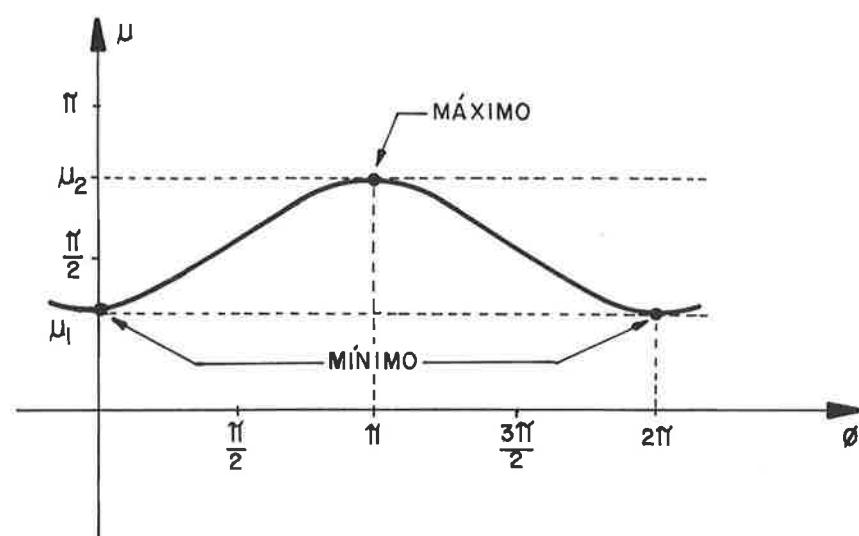


Fig. 8.6 - Gráfico de $\mu \times \phi$.

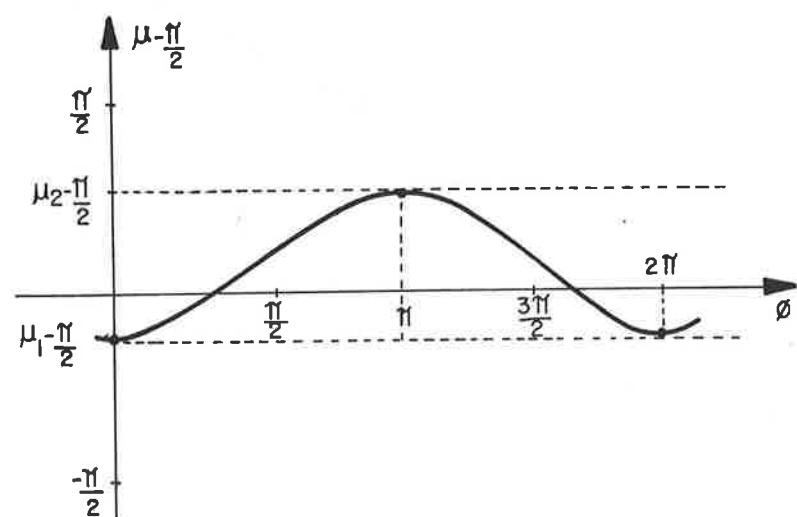


Fig. 8.7 - Gráfico do desvio de μ em torno de $90^\circ \times \phi$.

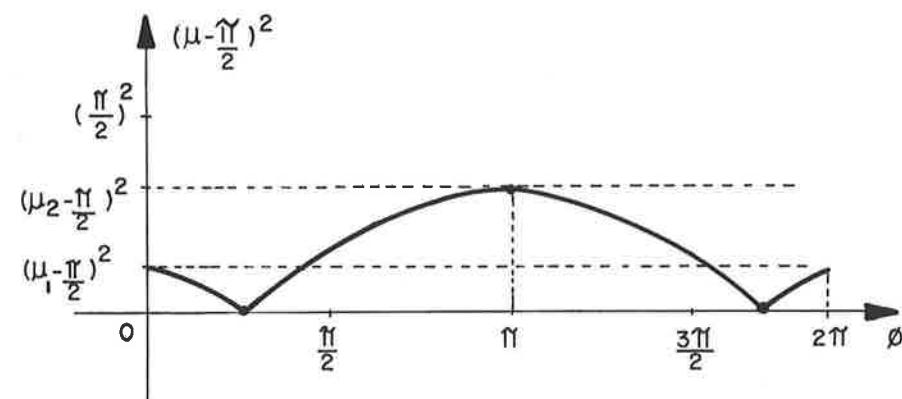


Fig. 8.8 - Gráfico do desvio quadrático de $\mu \times \phi$.

Pela forma do gráfico do desvio quadrático concluiu-se que no mínimo 5 pontos seriam necessários para uma boa representação dessa função. A posição destes pontos é assinalada na figura 8.9. Pode-se perceber que os pontos 2 e 4 são os únicos pontos cujas posições variam e o seu cálculo não é direto. Desta maneira, preferiu-se, por ser mais simples, calcular o desvio quadrático em 10 pontos igualmente espaçados começando em $\phi = 0^\circ$, ao invés de calcular os pontos 2 e 4.

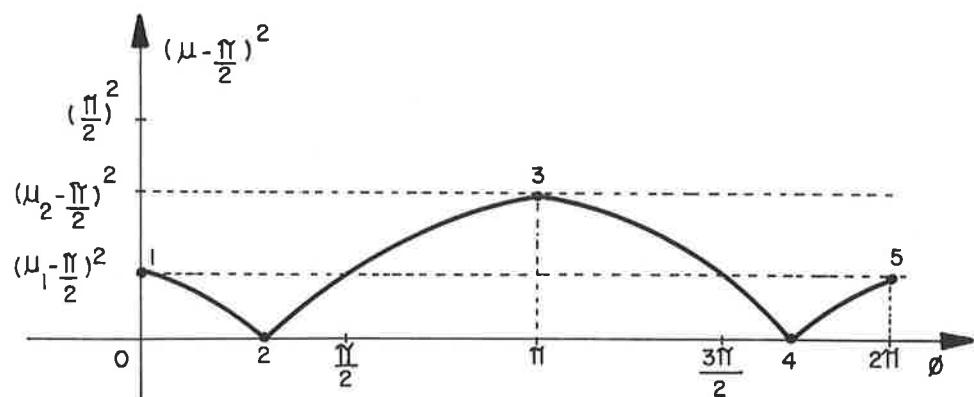


Fig. 8.9 - Posição dos pontos para o cálculo da função objetivo no gráfico do desvio quadrático de μ x ϕ .

Juntando essa conclusão do número de pontos mínimo com o problema de travamento, anteriormente levantado, a escolha de 10 pontos igualmente espaçados começando em $\phi = 0^\circ$ permite, se houver travamento, uma maior penalização do mecanismo que possuir uma maior região de travamento. Em suma, os dois problemas são resolvidos completamente com esse procedimento.

8.6 - Comentários quanto aos exemplos escolhidos

Os exemplos escolhidos para otimização foram tirados de uma publicação recente ([13]) e outras não tão recentes ([6], [10]). A escolha destes exemplos se deve a :

a) foram escritos por autoridades no assunto de nível mundial;

b) foram os exemplos destas publicações que serviram de aferição para os programas de síntese sem otimização desenvolvidos neste trabalho .

8.7 - Comentários a respeito do método escolhido para a síntese

O método escolhido para síntese se baseia na solução analítica desenvolvida por Freudenstein e Sandor [5], e na equação da diada apresentada por Sandor e Erdman [27].

Este método possui muitas vantagens na sua utilização. A primeira delas é que a sua formulação é de fácil visualização. As variáveis envolvidas possuem representação física imediata. Apresenta-se o mecanismo na sua posição inicial e nas posteriores posições de precisão. A formulação surge de equações de malha fechada considerando o mecanismo na primeira posição e numa posição j .

Outra vantagem deste método é que para resolver sistemas não-lineares é possível recorrer-se a soluções geométricas conforme mostrado no capítulo 3, item 3.7 . Isto simplifica a solução deste tipo de sistemas o que torna mais rápidos os

programas de computador que utilizam estes métodos e além disso, fornecem todas as soluções possíveis . Esta vantagem não se consegue quando se utiliza métodos numéricos sofisticados como Newton-Raphson que só fornecem uma solução em cada busca e são muito sensíveis à estimativa inicial, ou seja, pode não haver convergência. O método apresentado por Suh e Radcliffe para síntese de 4 posições necessita de um método do tipo Newton-Raphson.

A última grande vantagem do método da equação da diada sobre o método das matrizes de deslocamento é que o primeiro é mais geral que o segundo. Em outras palavras, é facil resolver diferentes tarefas de síntese com a mesma formulação bastando alterar as variáveis conhecidas e as desconhecidas. O mesmo não ocorre com o método das matrizes de deslocamento que necessita de mais equações para resolver os diferentes problemas de síntese, geração de função, de movimento e de trajetória.

9. CONCLUSÕES

1. Os mecanismos obtidos a partir dos programas de otimização nos diferentes problemas apresentados, revelaram um comportamento superior àqueles propostos na literatura. Estes resultados ficaram evidentes pelo pequeno desvio em torno de 90° apresentado pelo ângulo de transmissão μ nos mecanismos otimizados em comparação com soluções da literatura. No caso da síntese para 3 posições, o desvio na otimização (síntese com 4 e 12 variáveis) foi inferior a 30° e a publicação de Suh e Radcliffe [6] apresentou 65° . Na síntese para 4 posições, o desvio na otimização (síntese com 13 variáveis) foi inferior a 20° e a publicação de Gustafson e Erdman forneceu 46° .

2. Outro aspecto importante, que indicou um resultado superior à uma publicação recente de Peterson e colaboradores [13], foi a otimização do ângulo μ com rotação completa da alavanca motora (manivela). Neste caso, o objetivo era tomar como estimativa inicial um mecanismo que durante a sua movimentação sofreria travamento e procurar um mecanismo que, além de atender os objetivos da síntese e de possuir valores de μ próximos de 90° , fosse do tipo manivela-alavanca, ou seja, que não sofre travamento. Esse objetivo foi alcançado com sucesso no exemplo apresentado.

3. A função objetivo escolhida para a otimização é

penalizadora, ou seja, se a região de travamento do mecanismo durante a busca for relativamente grande esse mecanismo será considerado ruim. Assim como o valor atribuído à função objetivo será grande, o programa altera a direção de busca para a determinação de um mínimo, ou seja, um mecanismo com valores de μ próximos de 90° .

4. Um outro aspecto a ressaltar é que o cálculo da função objetivo, no caso de existir uma região de travamento, sempre considera o tamanho dessa região mesmo que ela se restrinja ao caso limite de um único ponto.

5. A escolha de 10 pontos para o cálculo da função objetivo foi o resultado de um compromisso entre o número mínimo de pontos para que seu valor tenha representatividade e o fato de que a sua penalização seja maior nas regiões de travamento.

6. O método do Gradiente Reduzido, recomendado principalmente para o caso em que a função objetivo e as restrições são não-lineares [16, 18, 28], revelou-se bastante eficiente nos exemplos apresentados. O tempo para o processamento no problema mais complexo (síntese para 4 posições com 13 variáveis) não excedeu 15 minutos .

10. TEMAS DE PESQUISA FUTURA

Como temas de pesquisa para os próximos anos pretende-se continuar o desenvolvimento de programas de computador para :

- a) elaboração de um pacote completo para o projeto de quadriláteros articulados (incluindo análise dinâmica) e de mecanismos planos com mais de quatro barras .
- b) síntese e otimização de mecanismos espaciais.

11. APÊNDICE

11.1 - Listagem do programa SYN 3 (versão 5)

Program Syn35

```
integer i,k,k1,k2,k3,k4

double precision xm, arx, ym, ary, tm, art,
1      w(2,2), mw(2), argw(2),modr(2),argr(2),
1      z(2,2), mz(2), argz(2), R(2,2),
1      xt(2), y(2) , t(2),
1      d(2:3,1:2), md(2:3), argd(2:3),
1      be(2:3,1:2,1:2), alf(2:3),
1      Ao(2), Bo(2), A(2), B(2), P(3,2)

c ****
c      Entrada de dados
c ****

write(*,*) ' P 1 x : '
read(*,*) P(1,1)

write(*,*) ' P 1 y : '
read(*,*) P(1,2)

write(*,*) ' P 2 x : '
read(*,*) P(2,1)

write(*,*) ' P 2 y : '
read(*,*) P(2,2)

write(*,*) ' P 3 x : '
read(*,*) P(3,1)

write(*,*) ' P 3 y : '
read(*,*) P(3,2)

write(*,*) ' alfa 2 : '
read(*,*) alf(2)

write(*,*) ' alfa 3 : '
read(*,*) alf(3)

write(*,*) ' 1a. diada : Articulacao Fixa '
write(*,*) 
write(*,*) ' Ao x : '
read(*,*) Ao(1)
write(*,*) ' Ao y : '
read(*,*) Ao(2)

write(*,*) ' 2a. diada : Articulacao Fixa '
write(*,*) 
write(*,*) ' Bo x : '
read(*,*) Bo(1)
write(*,*) ' Bo y : '
read(*,*) Bo(2)
```

```
c ****
c      Abertura de arquivo de saida de dados
c ****
c
c      open( 6, file = 'c:\tarcisio\saidi1.doc', status = 'old ')
c ****
c      Calculo de DELTA
c ****
c
c      do 10 i = 1,2
c
c      d(2,i) = P(2,i) - P(1,i)
c      d(3,i) = P(3,i) - P(1,i)
c
c 10  continue
c
c      call Polar( d(2,1),d(2,2),md(2),argd(2) )
c      call Polar( d(3,1),d(3,2),md(3),argd(3) )
c
c      write(*,*) '          1 '
c ****
c      Conversao de ALFA de graus p/ rad.
c ****
```

Program Syn35

```
integer i,k,k1,k2,k3,k4

double precision xm, arx, ym, ary, tm, art,
1      w(2,2), mw(2), argw(2),modr(2),argr(2),
1      z(2,2), mz(2), argz(2), R(2,2),
1      xt(2), y(2) , t(2),
1      d(2:3,1:2), md(2:3), argd(2:3),
1      be(2:3,1:2,1:2), alfa(2:3),
1      Ao(2), Bo(2), A(2), B(2), P(3,2)
```

```
c ****
c     Entrada de dados
c ****
```

```
write(*,*) ' P 1 x : '
read(*,*) P(1,1)
```

```
write(*,*) ' P 1 y : '
read(*,*) P(1,2)
```

```
write(*,*) ' P 2 x : '
read(*,*) P(2,1)
```

```
write(*,*) ' P 2 y : '
read(*,*) P(2,2)
```

```
write(*,*) ' P 3 x : '
read(*,*) P(3,1)
```

```
write(*,*) ' P 3 y : '
read(*,*) P(3,2)
```

```
write(*,*) ' alfa 2   : '
read(*,*) alfa(2)
```

```
write(*,*) ' alfa 3   : '
read(*,*) alfa(3)
```

```
write(*,*) ' 1a. diada : Articulacao Fixa '
write(*,*) ' '
write(*,*) ' Ao x : '
read(*,*) Ao(1)
write(*,*) ' Ao y : '
read(*,*) Ao(2)
```

```
write(*,*) ' 2a. diada : Articulacao Fixa '
write(*,*) ' '
write(*,*) ' Bo x : '
read(*,*) Bo(1)
write(*,*) ' Bo y : '
read(*,*) Bo(2)
```

```

c ****
c      Abertura de arquivo de saida de dados
c ****
c
c      open( 6, file = 'c:\tarcisio\saidi.doc', status = 'old' )
c
c ****
c      Calculo de DELTA
c ****
c
c      do 10 i = 1,2
c
c      d(2,i) = P(2,i) - P(1,i)
c      d(3,i) = P(3,i) - P(1,i)
c
c 10 continue
c
c      call Polar( d(2,1),d(2,2),md(2),argd(2) )
c      call Polar( d(3,1),d(3,2),md(3),argd(3) )
c
c      write(*,*) '          1'
c
c ****
c      Conversao de ALFA de graus p/ rad.
c ****
c
c      call Rad( alf(2) )
c      call Rad( alf(3) )
c
c      if( alf(2).lt.1.0E-18.and.alf(2).gt.-1.0E-18 ) then
c          alf(2) = 1.0E-15
c      endif
c
c      if( alf(3).lt.1.0E-18.and.alf(3).gt.-1.0E-18 ) then
c          alf(3) = 1.0E-15
c      endif
c
c      write(*,*) '          2'
c
c ****
c      Calculo dos vetores R
c ****
c
c      R(1,1) = - P(1,1) + Ao(1)
c
c      R(1,2) = - P(1,2) + Ao(2)
c
c      call Polar( R(1,1),R(1,2),modr(1),argr(1) )
c
c      R(2,1) = - P(1,1) + Bo(1)
c
c      R(2,2) = - P(1,2) + Bo(2)
c
c      call Polar( R(2,1),R(2,2),modr(2),argr(2) )
c
c      write(*,*) '          3'

```

```

c ****
c      Resolucao para 1a. e 2a. diadas
c ****

      do 20 k = 1,2

c ****
c      Calculo de XT
c ****

      xt(1) = - d(3,1) - modr(k)*(cos( argr(k) + alf(3) )) + R(k,1)
      xt(2) = - d(3,2) - modr(k)*(sin( argr(k) + alf(3) )) + R(k,2)
      call Polar( xt(1), xt(2), xm, arx )
      write(*,*)'          4'

c ****
c      Calculo de Y
c ****

      y(1) = d(2,1) + modr(k)*(cos( argr(k) + alf(2) )) - R(k,1)
      y(2) = d(2,2) + modr(k)*(sin( argr(k) + alf(2) )) - R(k,2)
      call Polar( y(1), y(2), ym, ary )
      write(*,*)'          5'

c ****
c      Calculo de T
c ****

      t(1) = md(2)*cos( argd(2) + alf(3) ) - modr(k)*
      1   cos( argr(k) + alf(3) ) - md(3)*cos( argd(3) + alf(2) )*
      1   + modr(k)*cos( argr(k) + alf(2) )

      t(2) = md(2)*sin( argd(2) + alf(3) ) - modr(k)*
      1   sin( argr(k) + alf(3) ) - md(3)*sin( argd(3) + alf(2) )*
      1   + modr(k)*sin( argr(k) + alf(2) )

      call Polar( t(1), t(2), tm, art )

      write(*,*)'          6'

c ****
c      Calculo de BETA_2 , BETA_3   e impressao em graus
c ****

      k4 = 2

      call trian( xm, ym, tm, arx, ary, art, be(2,k,1), be(2,k,2),
      1           be(3,k,1), be(3,k,2), k4 )

      if( k4.eq.1 ) then

```

```

        write(6,*)
        write(6,*)' Nao e possivel se determinar um mecanismo '
        write(6,*)' que atenda a essas especificacoes de projeto '
        write(*,*)
        write(*,*)' Nao e possivel se determinar um mecanismo '
        write(*,*)' que atenda a essas especificacoes de projeto '
        write(*,*)
        goto 500

    endif

    write(*,*)'          7'
    c      call Grau ( be(2,k,1) )
    c      call Grau ( be(2,k,2) )
    c      call Grau ( be(3,k,1) )
    c      call Grau ( be(3,k,2) )

    c      write(6,*)' Diada no. : ', k
    c      write(6,*)

    c      write(6,*)' 1a. solucao '
    c      write(6,*)

    c      write(6,*)' beta 2 = ',be(2,k,1)
    c      write(6,*)' beta 3 = ',be(3,k,1)

    c      write(6,*)' 2a. solucao '
    c      write(6,*)

    c      write(6,*)' beta 2 = ',be(2,k,2)
    c      write(6,*)' beta 3 = ',be(3,k,2)

    c      call Rad ( be(2,k,1) )
    c      call Rad ( be(2,k,2) )
    c      call Rad ( be(3,k,1) )
    c      call Rad ( be(3,k,2) )

20 continue

k3 = 0

do 25 k1 = 1,2
  do 30 k2 = 1,2

    k3 = k3 + 1

    write(6,*)
    write(6,*)' Solucao : ', k3
    write(6,*)

    call Grau( be( 2, 1, k1 ) )
    call Grau( be( 3, 1, k1 ) )

```

```

write(6,*)' 1a. diada : '
write(6,*)
write(6,*)' beta 2 = ', be( 2, 1, k1 )
write(6,*)
write(6,*)' beta 3 = ', be( 3, 1, k1 )
write(6,*)

call Rad( be( 2, 1, k1 ) )
call Rad( be( 3, 1, k1 ) )

c *****
c          Calculo de W e Z - 1a. diada
c *****

call sistema( alf(2), alf(3), be(2,1,k1), be(3,1,k1), d(2,1),
1           d(2,2), d(3,1), d(3,2), w(1,1), w(1,2),
1           z(1,1), z(1,2) )

call Grau( be( 2, 2, k2 ) )
call Grau( be( 3, 2, k2 ) )

write(6,*)' 2a. diada : '
write(6,*)
write(6,*)' beta 2 = ', be( 2, 2, k2 )
write(6,*)
write(6,*)' beta 3 = ', be( 3, 2, k2 )
write(6,*)
write(6,*)

call Rad( be( 2, 2, k2 ) )
call Rad( be( 3, 2, k2 ) )

c *****
c          Calculo de W e Z - 2a. diada
c *****

call sistema( alf(2), alf(3), be(2,2,k2), be(3,2,k2), d(2,1),
1           d(2,2), d(3,1), d(3,2), w(2,1), w(2,2),
1           z(2,1), z(2,2) )

c *****
c          Calculo de Ao, Bo, A, B
c *****

do 40 i = 1,2
  A(i) = P(1,i) - z(1,i)
  B(i) = P(1,i) - z(2,i)
40 continue

c *****
c          Impressao de Ao, Bo, A, B, P
c *****

```

```
write(6,*), Ax = ',A(1),', Ay = ',A(2)
write(6,*), Ax = ',A(1),', Ay = ',A(2)
write(6,*), Bx = ',B(1),', By = ',B(2)
write(6,*), Bx = ',B(1),', By = ',B(2)
write(6,*), Px = ',P(1,1),', Py = ',P(1,2)

30 continue
25 continue

500 write(*,*), Os resultados do SYN35 estao no arquivo '
write(*,*)
write(*,*),      C:\ tarcisio\ said1.doc '
write(*,*)

stop
end
```

11.2 - Listagem do programa SYN 4 (versão 5)

Program Syn45

```
integer i,j,k,k1,k2,k3,k4

Double Precision alf(2:4), d(2:4,1:2), dm(2:4), argd(2:4),
1      de(4,2), dem(4), argdem(4),
1      del(2), edmo, argdel,
1      be(2:4,1:2,1:2) ,
1      ze(1:6,1:2), tt(1:6), argze(1:6),
1      Ao(2), Bo(2), A(2), B(2), F(4,2),
1      w(2,2), z(2,2),
1      thet(4)

open(5,file='c:\tarcisio\entr1.doc',status='old')
open(6,file='c:\tarcisio\said2.doc',status='old')

c ****
c ***** Entrada de Dados *****
c *****

do 12 i = 1,4

  read(5,*) thet(i)
c   write(*,*)' theta( ,i, )= ',thet(i)

12 continue

do 22 i = 1,4
  do 32 j = 1,2

    read(5,*) P(i,j)
c     write(*,*)' P( ,i, ,j, )= ',P(i,j)

32 continue
22 continue

  do 52 j = 1,2

    read(5,*) be(2,j,1)
c     write(*,*)' beta ( 2, ,j, )= ',be(2,j,1)
    be(2,j,2) = be(2,j,1)

52 continue

c ****
c ***** Calculo de ALFA e DELTA *****
c *****

  alf(2) = thet(2) - thet(1)
  alf(3) = thet(3) - thet(1)
  alf(4) = thet(4) - thet(1)

  do 72 i = 2,4
    write(*,*)' alfa( ,i, )= ',alf(i)
72 continue
```

```

do 62 j =1,2

d(2,j) = P(2,j) - P(1,j)
d(3,j) = P(3,j) - P(1,j)
d(4,j) = P(4,j) - P(1,j)
write(6,*)' d(2,',j,',')= ',d(2,j)
write(6,*)' d(3,',j,',')= ',d(3,j)
c      write(*,*)' d(4,',j,',')= ',d(4,j)

62 continue

call rad(alf(2))
call rad(alf(3))
call rad(alf(4))

if( alf(2).lt.1.0E-6.and.alf(2).gt.-1.0E-6 ) then
               alf(2) = 1.0E-5
endif

if( alf(3).lt.1.0E-6.and.alf(3).gt.-1.0E-6 ) then
               alf(3) = 1.0E-5
endif

if( alf(4).lt.1.0E-6.and.alf(4).gt.-1.0E-6 ) then
               alf(4) = 1.0E-5
endif

call rad(be(2,1,1))
call rad(be(2,2,1))
call rad(be(2,1,2))
call rad(be(2,2,2))

do 53 i=2,4

call Polar( d(i,1), d(i,2), dm(i), argd(i) )
c   call Grau(argd(i))
c   write(*,*)' dm(',i,',')= ',dm(i),' arg(',i,',')= ',argd(i)

53 continue

c **** Calculo de Delta_2 ****
c
c     de(2,1) = (cos(alf(3))-1)*d(4,1) - sin(alf(3))*d(4,2)
c     1 - (cos(alf(4))-1)*d(3,1) + sin(alf(4))*d(3,2)
c
c     de(2,2) = sin(alf(3))*d(4,1) + d(4,2)*cos(alf(3)-1)
c     1 - sin(alf(4))*d(3,1) - cos(alf(4)-1)*d(3,2)
c
c     de(2,1) = dm(4)*cos(argd(4)+alf(3))-d(4,1)+d(3,1)
c     1 -dm(3)*cos(argd(3)+alf(4))

```

```

      de(2,2) = dm(4)*sin(argd(4)+alf(3))-d(4,2)+d(3,2)
1           -dm(3)*sin(argd(3)+alf(4))

c **** Calculo de Delta_3 ****
c
c     de(3,1) = (cos(alf(4))-1)*d(2,1) - sin(alf(4))*d(2,2)
c     1       - (cos(alf(2))-1)*d(4,1) + sin(alf(2))*d(4,2)
c
c     de(3,2) = sin(alf(4))*d(2,1) + (cos(alf(4))-1)*d(2,2)
c     1       - sin(alf(2))*d(4,1) - (cos(alf(2))-1)*d(4,2)

c
c     de(3,1) = dm(2)*cos(argd(2)+alf(4))-d(2,1)+d(4,1)
c     1       -dm(4)*cos(argd(4)+alf(2))
c
c     de(3,2) = dm(2)*sin(argd(2)+alf(4))-d(2,2)+d(4,2)
c     1       -dm(4)*sin(argd(4)+alf(2))

c **** Calculo de Delta_4 ****
c
c     de(4,1) = (cos(alf(2))-1)*d(3,1) - sin(alf(2))*d(3,2)
c     1       - (cos(alf(3))-1)*d(2,1) + sin(alf(3))*d(2,2)
c
c     de(4,2) = sin(alf(2))*d(3,1) + d(3,2)*(cos(alf(2))-1)
c     1       - (cos(alf(3))-1)*d(2,2) - sin(alf(3))*d(2,1)

c
c     de(4,1) = dm(3)*cos(argd(3)+alf(2))-d(3,1)+d(2,1)
c     1       -dm(2)*cos(argd(2)+alf(3))
c
c     de(4,2) = dm(3)*sin(argd(3)+alf(2))-d(3,2)+d(2,2)
c     1       -dm(2)*sin(argd(2)+alf(3))

c **** Calculo de Delta_1 ****
c
c     de(1,1) = - de(2,1) - de(3,1) - de(4,1)
c     de(1,2) = - de(2,2) - de(3,2) - de(4,2)

c     write(6,*) ' de11 = ', de(1,1)
c     write(6,*) ' de12 = ', de(1,2)

c
c     do 100 i=1,4
c
c     write(*,*)' de(1,i,1)= ',de(i,1),' de(1,i,2)= ',de(i,2)
c
c 100 continue

```

```

call Polar ( de(1,1), de(1,2), dem(1), argdem(1))
call Polar ( de(2,1), de(2,2), dem(2), argdem(2))
call Polar ( de(3,1), de(3,2), dem(3), argdem(3))
call Polar ( de(4,1), de(4,2), dem(4), argdem(4))

do 152 j=1,2

c ***** Calculo de Delta *****
del(1) = - de(1,1) - dem(2)*cos(argdem(2)+be(2,j,1))
del(2) = - de(1,2) - dem(2)*sin(argdem(2)+be(2,j,1))

call Polar (del(1),del(2),edmo,argdel)

do 153 i = 2,4

      write(6,*)' dem(',i,',') = ', dem(i)
      write(6,*)
      call Grau( argdem(i) )
      write(6,*)' argdem (',i,',') = ', argdem(i)
      call Rad( argdem(i) )

153 continue

      write(6,*)'dem(3)= ',dem(3)
      write(6,*)'dem(4)= ',dem(4)
      write(6,*)'edmo = ',edmo
      write(6,*)' argdel = ', argdel

c ***** Calculo de Beta3, Beta4 *****
k4 = 2

call trian ( dem(3), dem(4), edmo, argdem(3), argdem(4), argdel,
1           be(3,j,1), be(3,j,2), be(4,j,1), be(4,j,2), k4 )

if( k4.eq.1 ) then

      write(6,*)
      write(6,*)' Nao e possivel se determinar um mecanismo '
      write(6,*)' que atenda a essas especificacoes de projeto '
      write(*,*)
      write(*,*)' Nao e possivel se determinar um mecanismo '
      write(*,*)' que atenda a essas especificacoes de projeto '
      write(*,*)
      goto 500

endif

call checan ( be(2,j,1))
call checan ( be(2,j,2))
call checan ( be(3,j,1))

```

```

call checan ( be(3,j,2))
call checan ( be(4,j,1))
call checan ( be(4,j,2))

call Grau ( be(2,j,1))

call Grau ( be(3,j,1))
call Grau ( be(3,j,2))
call Grau ( be(4,j,1))

call Grau ( be(4,j,2))

c      write(6,*)
c      write(6,*)
c      write(6,*)
c      write(6,*)
c      write(6,*)

```

152 continue

```

call Rad( be(2,1,1))
call Rad( be(2,2,1))
call Rad( be(3,1,1))
call Rad( be(3,2,1))
call Rad( be(3,1,2))
call Rad( be(3,2,2))
call Rad( be(4,1,1))
call Rad( be(4,2,1))
call Rad( be(4,1,2))
call Rad( be(4,2,2))

```

k3 = 0

```

do 250 k1 = 1,2
  do 300 k2 = 1,2

```

k3 = k3 + 1

```

write(6,*)
write(6,*)
write(6,*)
call Grau( be( 2, 1, k1 ) )
call Grau( be( 3, 1, k1 ) )
call Grau( be( 4, 1, k1 ) )

write(6,*)
write(6,*)
write(6,*)
write(6,*)
write(6,*)
write(6,*)

```

```

write(6,*)

call Rad( be( 2, 1, k1 ) )
call Rad( be( 3, 1, k1 ) )
call Rad( be( 4, 1, k1 ) )

c ****
c          Calculo de W e Z - 1a. diada
c ****

call sistema( alf(2), alf(3), be(2,1,k1), be(3,1,k1), d(2,1),
1           d(2,2), d(3,1), d(3,2), w(1,1), w(1,2),
1           z(1,1), z(1,2) )

call Grau( be( 2, 2, k2 ) )
call Grau( be( 3, 2, k2 ) )
call Grau( be( 4, 2, k2 ) )

write(6,*)' 2a. diada : '
write(6,*)
write(6,*)' beta 2 = ', be( 2, 2, k2 )
write(6,*)
write(6,*)' beta 3 = ', be( 3, 2, k2 )
write(6,*)
write(6,*)' beta 4 = ', be( 4, 2, k2 )
write(6,*)

call Rad( be( 2, 2, k2 ) )
call Rad( be( 3, 2, k2 ) )
call Rad( be( 4, 2, k2 ) )

c ****
c          Calculo de W e Z - 2a. diada
c ****

call sistema( alf(2), alf(3), be(2,2,k2), be(3,2,k2), d(2,1),
1           d(2,2), d(3,1), d(3,2), w(2,1), w(2,2),
1           z(2,1), z(2,2) )

c ****
c          Calculo de Ao, Bo, A, B
c ****

do 400 i = 1,2

A(i) = P(1,i) - z(1,i)

Ao(i) = A(i) - w(1,i)

B(i) = P(1,i) - z(2,i)

Bo(i) = B(i) - w(2,i)

400 continue

c ****

```

```
c           Impressao de Ao, Bo, A, B, P
c ****
c
c      write(6,*),< AoX = ',Ao(1),', AoY = ',Ao(2)
c      write(6,*),< Ax = ',A(1),', Ay = ',A(2)
c      write(6,*),< Bx = ',Bo(1),', Boy = ',Bo(2)
c      write(6,*),< Bx = ',B(1),', By = ',B(2)
c      write(6,*),< Px = ',P(1,1),', Py = ',P(1,2)
c
300  continue
250  continue
c
500  write(*,*),< Os resultados do SYN45 estao no arquivo '
c      write(*,*),
c      write(*,*),<          C:\ tarcisio\ said2.doc '
c      write(*,*)
c
c
stop
end
```

11.3 - Listagem do programa POS

Program Pos53

```
integer i, j, k, k1, nbet, index, valor, indice, valor2

Double Precision Ao(2), Bo(2), A( 200, 2 ), B( 200, 2 ),
1      P( 200, 2 ), be( 2:200, 1:2 ), fi,
1      zel( 6, 2, 200 ), mze(6), argzel( 6, 200 ),
1      dia( 200, 2 ), mdia( 200 ),
1      argdia( 200 ), theta( 200 ), c, d,
1      u(2), v(2), w(2),
1      vet1, vet2, pv, mi( 200 )

c
c **** Entrada de Dados ****
c
c Abertura de arquivo para entrada de dados
open( 5, file = 'c:\tarcisio\enpos.doc' , status = 'old' )

c Leitura de Ao, Bo, A, B, P
do 10 i=1,2
c     write(*,*)' Ao( ,i, )='
read(5,*) Ao(i)
10 continue
do 11 i=1,2
c     write(*,*)' Bo( ,i, )='
read(5,*) Bo(i)
11 continue
do 12 i=1,2
c     write(*,*)' A( ,i, )='
read(5,*) A(1,i)
12 continue
do 13 i=1,2
c     write(*,*)' B( ,i, )='
read(5,*) B(1,i)
13 continue
do 14 i=1,2
c     write(*,*)' P( ,i, )='
read(5,*) P(1,i)
```

```

14 continue

c
c  Leitura do No. de Posicoes de Precisao
c
c      write(*,*)'  No. de posicoes ( max 50 ) : '
read(5,*) nbet

c
c  Leitura dos BETAS
c
do 20 k = 2,nbet

c      write(*,*)'  be( ',k,' )= '
read(5,*) be(k,1)
call Rad( be(k,1))

20 continue

c *****
c      Pergunta : deseja alterar os indices no caso de
c                  desmontagem
c *****

      write(*,*)'  Deseja alterar os indices no caso de '
      write(*,*)'  desmontagem ? Tecle [1] p/ alterar '
      write(*,*)'  e [2] p/ manter '
read(*,*) valor2

c *****
c      Abertura de arquivos para saida de resultados
c *****

open( 6, file = 'c:\tarcisio\lepos2.doc', status = 'old' )

c *****
c      Calculo dos ZE ( 1 a 6 )
c *****

do 30 j = 1,2

ze(1,j,1) = A(1,j) - Ao(j)

c      write(*,*)'  ze( 1,' ,j,' )= ', ze(1,j,1)

ze(2,j,1) = P(1,j) - A(1,j)

c      write(*,*)'  ze( 2,' ,j,' )= ', ze(2,j,1)

ze(3,j,1) = B(1,j) - Bo(j)

c      write(*,*)'  ze( 3,' ,j,' )= ', ze(3,j,1)

ze(4,j,1) = P(1,j) - B(1,j)

```

```

c      write(*,*)' ze( 4,' ,j,')= ', ze(4,j,1)
ze(5,j,1) = B(1,j) - A(1,j)
c      write(*,*)' ze( 5,' ,j,')= ', ze(5,j,1)
ze(6,j,1) = Bo(j) - Ao(j)
c      write(*,*)' ze( 6,' ,j,')= ', ze(6,j,1)

30 continue

c **** Conversao dos ZE ( 1 a 6 ) de coord. cartes. p/ polares
c ****
do 40 i = 1,6
    call Polar( ze(i,1,1), ze(i,2,1), mze(i), argze(i,1))
40 continue

write(*,*)' mze(5)= ',mze(5)

c **** Calculo do angulo entre Z_2 e Z_5 ( fi ) e mi
c ****
fi = argze(2,1) - argze(5,1)
call Grau( fi)
write(*,*)' fi= ',fi
call Rad( fi)

c     theta(1) = argze(5,1) + fi
be( 1, 1) = 0.0
mi( 1 ) = argze( 3, 1 ) - argze( 5, 1 )
call checan( mi( 1 ) )
call Grau( mi( 1 ) )
write(*,*)' mi_1 = ', sngl( mi( 1 ) )

c **** Calculo de A, B, P para cada BETA
c ****
do 50 k = 2,nbet

```

```

c ****
c          Calculo de U
c ****
c
c      u( 1 ) = Bo( 1 ) - A( k-1, 1 )
c
c      write(6,*)' Bo_1 = ', Bo( 1 )
c
c      write(6,*)' A( ,k-1, , 1 )= ', A( k-1, 1 )
c
c      u( 2 ) = Bo( 2 ) - A( k-1, 2 )
c
c      write(6,*)' Bo_2 = ', Bo( 2 )
c
c      write(6,*)' A( ,k-1, , 2 )= ', A( k-1, 2 )
c
c      write(6,*)' u1 = ', u( 1 )
c      write(6,*)' u2 = ', u( 2 )
c
c ****
c          Calculo de V
c ****
c
c      v( 1 ) = B( k-1, 1 ) - A( k-1, 1 )
c
c      write(6,*)' B( ,k-1, , 1 )= ', B( k-1, 1 )
c
c      write(6,*)' A( ,k-1, , 1 )= ', A( k-1, 1 )
c
c      v( 2 ) = B( k-1, 2 ) - A( k-1, 2 )
c
c      write(6,*)' B( ,k-1, , 2 )= ', B( k-1, 2 )
c
c      write(6,*)' A( ,k-1, , 2 )= ', A( k-1, 2 )
c
c      write(6,*)' v1 = ', v( 1 )
c      write(6,*)' v2 = ', v( 2 )
c
c ****
c          Produto Vetorial de U por V
c ****
c
c      call Pvet( u( 1 ), u( 2 ), v( 1 ), v( 2 ), pv )
c
c      write(6,*)' pv = ', pv
c
c ****
c      Condicoes para determinacao do indice do quadrilatero
c ****
c
c      if( pv.gt.0 ) then
c
c          index = 1
c
c      else

```

```

        index = 2

    endif

    write(*,*) ' indice = ', index

c ****
c      Alteracao do usuario no valor do indice ( caso o angulo
c      de transmissao "mi" se aproxime de 0 ou 180 )
c ****

    if( valor2.eq.1 ) then

        write(*,*)' Voce deseja alterar o valor do indice ? '
        write(*,*)'
        write(*,*)' Tecle [ 1 ] p/ alterar e [ 2 ] p/ manter '
        read(*,*) valor

        if( valor.eq.1 ) then

            write(*,*)' Entre o novo valor do indice : '
            read(*,*) index

        endif
    endif

    write(6,*)' indice = ', index

    argze(1,k) = argze(1,1) + be(k,1)

    call Cartes( mze(1), argze(1,k), ze(1,1,k), ze(1,2,k))

c ****
c      Calculo da diagonal DIA
c ****

    do 60 j = 1,2

        dia(k,j) = ze(6,j,1) - ze(1,j,k)

60    continue

c ****
c      Conversao de DIA de coord. cartes. p/ polares
c ****

    call Polar( dia(k,1), dia(k,2), mdia(k), argdia(k))

    write(*,*)' mdia( ,k, )= ',mdia(k)

c ****
c      Determinacao de Z_3 e Z_5 ( coord. cartes. )
c ****

```

```

c ****
indice = 0

call Trian4( mdia(k), mze(5), mze(3), argdia(k),
1           ze(5,1,k), ze(5,2,k), ze(3,1,k),
1           ze(3,2,k), index, indice )

c ****
c      Verificacao de travamento do mecanismo
c ****

if( indice.eq.1 ) then

  write(*,*)
  write(*,*)' O mecanismo trava na posicao: ', k
  write(*,*)

  goto 150

endif

c ****
c      Determinacao das Articulacoes Moveis : A e B
c ****

do 70 j = 1,2

  A(k,j) = Ao(j) + ze(1,j,k)

  B(k,j) = Bo(j) + ze(3,j,k)

70 continue

c ****
c      Conversao de Z_3 e Z_5 de coord. cartes. p/ polares
c ****

call Polar( ze(5,1,k), ze(5,2,k), mze(5), argze(5,k))
call Polar( ze(3,1,k), ze(3,2,k), mze(3), argze(3,k))

write(*,*)' argze5(' ,k,')= ',argze(5,k)

c ****
c      Calculo do argumento de Z_2
c ****

argze(2,k) = argze(5,k) + fi

c ****
c      Conversao de Z_2 de coord. polares p/ cartes.
c ****

call Cartes( mze(2), argze(2,k), ze(2,1,k), ze(2,2,k))

```

```

c ****
c      Calculo de P em coord. cartes.
c ****
do 80 j = 1,2
      P(k,j) = A(k,j) + ze(2,j,k)
80 continue

c ****
c      Calculo de Theta ( angulo do acoplador )
c ****
theta(k) = argze(5,k) + fi
c ****
c      Calculo do Angulo de transmissao ( mi )
c ****
mi( k ) = argze( 3, k ) - argze( 5, k )
call checan( mi( k ) )
call Grau( mi( k ) )
write(*,*) ' mi_ ',k,' = ', mi( k )

c ****
c      Calculo de BETA ( 2a. diada )
c ****
be(k,2) = argze(3,k) - argze(3,1)
call checan( be(k,2) )
call Grau( be(k,2) )
call checan( be(k,1) )
call Grau( be(k,1) )
50 continue

150 if( indice.eq.1 ) then
      k1 = k - 1
      else
      k1 = nbet
      endif
      do 90 k = 1, k1

```

```

c ****
c           Impressao de A , B e P
c ****

      do 100 j = 1,2

      write(6,*)' A(‘,k,’,‘,j,’)= ‘,A(k,j)
      write(6,*)' B(‘,k,’,‘,j,’)= ‘,B(k,j)
      write(6,*)' P(‘,k,’,‘,j,’)= ‘,P(k,j)

100  continue

c      call Grau( theta(k))
      call Grau( argze(2,k))

c      write(*,*)' theta(‘,k,’)= ‘, theta(k)

c ****
c           Impressao de Theta , mi e BETA ( 1a. e 2a. diada )
c ****

      write(6,*)
      write(6,*)' theta(‘,k,’)= ‘, argze(2,k)
      write(6,*)
      write(6,*)' mi(‘,k,’)= ‘, mi( k )
      write(6,*)
      write(6,*)' beta(‘,k,’,1)= ‘,be(k,1)
      write(6,*)
      write(6,*)' beta(‘,k,’,2)= ‘,be(k,2)
      write(6,*)

90   continue

      if( indice.eq.1 ) then

          k1 = k1 + 1

          write(6,*)
          write(6,*)' *****
          write(6,*)'      O mecanismo trava na posicao: ‘, k1
          write(6,*)' *****
          write(6,*)

      endif

      write(*,*)' Os resultados do POS_52 estao no arquivo : '
      write(*,*)'
      write(*,*)'      c:\tarcisio\lepos2.doc '
      write(*,*)

      stop
      end

```

**11.4 - Listagem do programa RETA para animação do
mecanismo**

```

Program Reta;
uses graph,crt;
Var
  nmaxx, nminx, nmaxy,nminy : real;
  npos,x,y,i,j,k :integer;
  A, B, P : array[1..200,1..2] of real;
  Ai, Bi, Pi : array[1..200,1..2] of integer;
  Ao, Bo : array[1..2] of real ;
  Aio, Bio : array[1..2] of integer;
  texto1, texto2, texto3, texto4,
  texto5, texto6, texto7, texto8 : string;
  arq : text;
  z:char;

Procedure Escala( var Px, Py, nmaxx, nminx, nmaxy, nminy : real );
begin
  Px := 437 * ( Px - nminx )/( nmaxx - nminx ) + 90 ;
  Py := -190 * ( Py - nminy )/( nmaxy - nminy ) + 195 ;
end;

Procedure Minmax( var Px, Py, nmaxx, nminx, nmaxy, nminy : real );
begin
  if( Px > nmaxx ) then nmaxx := Px ;
  if( Px < nminx ) then nminx := Px ;
  if( Py > nmaxy ) then nmaxy := Py ;
  if( Py < nminy ) then nminy := Py ;
end ;

begin
  Assign( arq, 'c:\tarcisio\lepos.doc' );
  Reset( arq );
  readln ( arq, npos);
  x:=4;
  y:=1;
  clrscr;
  for i:= 1 to 2 do
  begin
    { write( 'Ao(' ,i,')=' ); }
    readln( arq, Ao[i] );

```

```

end;
for i:= 1 to 2 do
begin
{ write( 'Bo( ,i, )=' ); }
readln( arq, Bo[i] );
end;

for j:= 1 to npos do
begin
  for i:= 1 to 2 do
    begin
      readln( arq, A[j,i] );
      readln( arq, B[j,i] );
      readln( arq, P[j,i] );
    end;
  end;

{ Ao[1] := 0.159   ;
Ao[2] := 0.49   ;
Bo[1] := -0.432   ;
Bo[2] := 1.91   ;
A[1,1] := 1      ;
A[1,2] := -1.19e-7   ;
B[1,1] := -0.856   ;
B[1,2] := -2.28   ;
P[1,1] := 2      ;
P[1,2] := 0      ; }

nmaxx := 1;
nminx := 1;
nmaxy := 1;
nminy := 1;

for i:= 1 to npos do
begin

Minmax( A[i,1], A[i,2], nmaxx, nminx, nmaxy, nminy );
Minmax( B[i,1], B[i,2], nmaxx, nminx, nmaxy, nminy );
Minmax( P[i,1], P[i,2], nmaxx, nminx, nmaxy, nminy );
end;

```

```

Minmax( Ao[1], Ao[2], nmaxx, nminx, nmaxy, nminy ) ;
Minmax( Bo[1], Bo[2], nmaxx, nminx, nmaxy, nminy ) ;
if( nmaxx > nmaxy ) then
    nmaxy := nmaxx
else
    nmaxx := nmaxy ;
if( nminx > nminy ) then
    nminx := nminy
else
    nminy := nminx ;

for i:= 1 to npos do
begin
    Escala( A[i,1], A[i,2], nmaxx, nminx, nmaxy, nminy );
    Ai[i,1] := round( A[i,1] );
    Ai[i,2] := round( A[i,2] );
    Escala( B[i,1], B[i,2], nmaxx, nminx, nmaxy, nminy );
    Bi[i,1] := round( B[i,1] );
    Bi[i,2] := round( B[i,2] );
    Escala( P[i,1], P[i,2], nmaxx, nminx, nmaxy, nminy );
    Pi[i,1] := round( P[i,1] );
    Pi[i,2] := round( P[i,2] );
end;
Escala( Ao[1], Ao[2], nmaxx, nminx, nmaxy, nminy );
Ao[1] := round( Ao[1] );
Ao[2] := round( Ao[2] );
Escala( Bo[1], Bo[2], nmaxx, nminx, nmaxy, nminy );
Bo[1] := round( Bo[1] );
Bo[2] := round( Bo[2] );

```

```

Initgraph(y,x,'');

SetLineStyle( 0, 0, 1 );
{ outtextxy(110,110,'retangulo'); }

rectangle( 87, 3, 530, 198 );
{
  str( round( nminx ),texto2 );
  texto1 := '(' + texto2 ;
  OutTextXY( 50, 50, texto1 ); }

str( round(nminx), texto1 );
str( round(nminy), texto2 );
str( round(nmaxy), texto3 );
str( round(nmaxx), texto4 );

texto5 := '(' + texto1 + ',' + texto2 + ')';
texto6 := '(' + texto4 + ',' + texto2 + ')';
texto7 := '(' + texto4 + ',' + texto3 + ')';
texto8 := '(' + texto1 + ',' + texto3 + ')';

OutTextXY( 30, 190, texto5 );
OutTextXY( 550, 190, texto6 );
OutTextXY( 550, 5, texto7 );
OutTextXY( 30, 5 , texto8 );

{
  OutTextxy( 90, 197, texto1 );
  OutTextxy( 80, 195, texto2 );
  OutTextxy( 527, 197, texto4 );
  OutTextxy( 537, 195, texto2 );
  OutTextxy( 537, 5, texto3 );
  OutTextxy( 527, 3, texto4 );
  OutTextxy( 90, 3, texto1 );
  OutTextxy( 80, 5 , texto3 ); }

k := 100;

for i:= 1 to npos do
begin
  setcolor( 3 );
  Line( Aio[1], Aio[2], Ai[i,1], Ai[i,2] );
  Line( Ai[i,1], Ai[i,2], Pi[i,1], Pi[i,2] );
  Line( Pi[i,1], Pi[i,2], Bi[i,1], Bi[i,2] );
  Line( Ai[i,1], Ai[i,2], Bi[i,1], Bi[i,2] );
}

```

```
        Line( Bi[i,1], Bi[i,2], Bio[1], Bio[2] );
        { PutPixel( Pi[i,1], Pi[i,2], 3 ); }
        { z:= readkey; }
        { sound( round(100*ln(100 + k)/ln(10)) ); }
        delay(150);
        { nosound; }
        if( i < npos ) then
        begin
            setcolor( 0 );
            Line( Aio[1], Aio[2], Ai[i,1], Ai[i,2] );
            Line( Ai[i,1], Ai[i,2], Pi[i,1], Pi[i,2] );
            Line( Pi[i,1], Pi[i,2], Bi[i,1], Bi[i,2] );
            Line( Ai[i,1], Ai[i,2], Bi[i,1], Bi[i,2] );
            Line( Bi[i,1], Bi[i,2], Bio[1], Bio[2] );
        end;
        k := k + 100 ;
    end;
    z := readkey ;
    Closegraph ;
end.
```

**11.5 - Listagem do programa de Otimização e suas
sub-rotinas**

```

Program Main
c
c scale = no. de variaveis + no. de restricoes
c
Implicit double precision ( A-H, O-Z )

Dimension x( 4 ), xmax( 4 ), xmin( 4 ), scale( 8 ),
1      con( 4 )
Common / Optdat/ D( 750 )
Common / Pari/ xtol, gtol, eps, epsls, epsbd, ipr, maxm, idata,
1           lfunct, ne, lne, ni, lni, ncon, nscale, isearc,
1           maxnew, idif, nout

c ***** Fator de escala para funcao objetivo F *****
c
c fscale = 1.0

c ***** Indica em qual unidade serao impressos os parametros
c ***** de saida

nout = 6

c ***** Fatores de escala serao calculados pela OPT3 *****
c
nscale = 2

c ***** Impressao do resumo das Saidas Iniciais e Finais *****
c
idata = 1

c ***** Imprime um resumo das iteracoes no final de cada iteracao
c
ipr = 1

c ***** Parametro de convergencia das variaveis X ( a cada
c ***** iteracao e' checado )
c
xtol = 0.1E-6

c ***** Parametro de convergencia do gradiente reduzido *****
c
gtol = 0.1E-3

c ***** Parametro de controle da busca linear *****
c
epsls = 0.1E-5

c ***** Parametro que define o limite em que uma variavel
c ***** e igual a uma de suas fronteiras
c
epsbd = 0.1E-7

c ***** Calculo da derivada para frente *****
c
idif = 1

```

```
c ***** Calculo da derivada para centro *****
      idif = 2

c ***** Parametro do incremento da derivada *****
c      eps = epsmach ( k ) ** ( 1.0D0/3.0D0 )
      eps = 0.1E-3

c ***** Metodo de Broyden-Fletcher-Goldfarb-Shanno Quasi-Newton
      isearc = 1

c ***** No. de variaveis de projeto *****
      n = 4

c ***** No. total de restricoes *****
      ncon = 4

c ***** No. de restricoes lineares de igualdade *****
      lne = 0

c ***** No. de restricoes de desigualdade *****
      ni = 4

c ***** No. de restricoes lineares de desigualdade *****
      lni = 0

c ***** No. total de restricoes de igualdade *****
      ne = 0

c ***** ??????
      lbd = 0

c ***** Funcao Objetivo NAO-LINEAR *****
      lfunct = 0

c ***** No. maximo de iteracoes *****
      maxm = 20

c ***** No. maximo de passos no metodo de Newton *****
      maxnew = 10

c ***** Tamanho do problema *****

```

```

isize = 500

c ***** Dados de Entrada ( chute inicial ) *****
c *****
c ***** x1, x2, x3 e x4 *****

c      write(*,*) ' 1a. diada : Articulacao Fixa '
c      write(*,*) 
c      write(*,*) ' Aox : '
c      read(*,*) x(1)
x(1) = 5

c      write(*,*) ' Aoy : '
c      read(*,*) x(2)
x(2) = 0

c      write(*,*) ' 2a. diada : Articulacao Fixa '
c      write(*,*) 
c      write(*,*) ' Box : '
c      read(*,*) x(3)
x(3) = 0

c      write(*,*) ' Boy : '
c      read(*,*) x(4)
x(4) = 0

c ***** Limite maximo de cada variavel *****
data xmax/ 20.,20.,20.,20. /

c ***** Limite minimo de cada variavel *****
data xmin/ -20.,-20.,-20.,-20. /

open( 6, file = 'd:\tarcisio\saída.doc', status = 'old' )

call opt3( x, xmax, xmin, scale, fscale, n, isize )

write(6,*)
write(6,*)' Os resultados estao no arquivo : '
write(6,*)
write(6,*)'      d:\tarcisio\saída.doc '
write(6,*)

stop

end

```

```

Function F( x )

c ****
c      FUNCAO    " FUNC34.for"
c *****

Implicit double precision (A-H,O-Z)

dimension x( 1 )

integer i,k,k1,k2,k3,k4

double precision xm, arx, ym, ary, tm, art,
1           w(2,2), mw(2), argw(2),modr(2),argr(2),
1           z(6,2), mz(6), argz(6), R(2,2),
1           xt(2), y(2) , t(2),
1           d(2:3,1:2), md(2:3), argd(2:3),
1           be(2:3,1:2,1:2), alf(2:3),
1           Ao(2), Bo(2), A(2), B(2), P(3,2), jeito

common /one/ nfe, nce

f = 1

c ****
c      Entrada de dados
c *****

P(1,1) = 1.0
P(1,2) = 1.0
P(2,1) = 2.0
P(2,2) = 0.5
P(3,1) = 3.0
P(3,2) = 1.5
alf(2) = 0.0
alf(3) = 45.0

c ****
c      Abertura de arquivo de saida de dados
c *****

c      open( 6, file = 'c:\tarcisio\saíd1.doc', status = 'old ' )

c ****
c      Calculo de DELTA
c ****

```

```

do 10 i = 1,2
d(2,i) = P(2,i) - P(1,i)
d(3,i) = P(3,i) - P(1,i)

10 continue

call Polar( d(2,1),d(2,2),md(2),argd(2) )
call Polar( d(3,1),d(3,2),md(3),argd(3) )

c ****
c      Conversao de ALFA de graus p/ rad.
c ****

call Rad( alf(2) )
call Rad( alf(3) )

if( alf(2).lt.1.0E-6.and.alf(2).gt.-1.0E-6 ) then
    alf(2) = 1.0E-5
endif

if( alf(3).lt.1.0E-6.and.alf(3).gt.-1.0E-6 ) then
    alf(3) = 1.0E-5
endif

c ****
c      Calculo dos vetores R
c ****

R(1,1) = - P(1,1) + x(1)
R(1,2) = - P(1,2) + x(2)
call Polar( R(1,1),R(1,2),modr(1),argr(1) )
R(2,1) = - P(1,1) + x(3)
R(2,2) = - P(1,2) + x(4)
call Polar( R(2,1),R(2,2),modr(2),argr(2) )

c ****
c      Resolucao para 1a. e 2a. diadas
c ****

do 20 k = 1,2
c ****
c      Calculo de XT
c ****

xt(1) = - d(3,1) - modr(k)*(cos( argr(k) + alf(3) )) + R(k,1)

```

```

xt(2) = - d(3,2) - modr(k)*(sin( argr(k) + alf(3) )) + R(k,2)
call Polar( xt(1), xt(2), xm, arx )

c *****
c      Calculo de Y
c *****

y(1) = d(2,1) + modr(k)*(cos( argr(k) + alf(2) )) - R(k,1)
y(2) = d(2,2) + modr(k)*(sin( argr(k) + alf(2) )) - R(k,2)
call Polar( y(1), y(2), ym, ary )

c *****
c      Calculo de T
c *****

t(1) = md(2)*cos( argd(2) + alf(3) ) - modr(k)*
1   cos( argr(k) + alf(3) ) - md(3)*cos( argd(3) + alf(2) )
1   + modr(k)*cos( argr(k) + alf(2) )

t(2) = md(2)*sin( argd(2) + alf(3) ) - modr(k)*
1   sin( argr(k) + alf(3) ) - md(3)*sin( argd(3) + alf(2) )
1   + modr(k)*sin( argr(k) + alf(2) )

call Polar( t(1), t(2), tm, art )

c *****
c      Calculo de BETA_2 , BETA_3 e impressao em graus
c *****

write(*,*) ' *** 5 **** '
k4 = 2

call trian( xm, ym, tm, arx, ary, art, be(2,k,1), be(2,k,2),
1           be(3,k,1), be(3,k,2), k4 )

write(*,*) ' *** 6 **** '

if( k4.eq.1 ) then
    jeito = 100.0
    goto 500
endif

20 continue

```

```

c ****
c      Possiveis solucoes :
c ****

c      [ 1 ] ... k1 = 1 , k2 = 1
c      [ 2 ] ... k1 = 1 , k2 = 2
c      [ 3 ] ... k1 = 2 , k2 = 1
c      [ 4 ] ... k1 = 2 , k2 = 2

c ****
c      Utilizando a Solucao # 2
c ****

k1 = 1
k2 = 2

c      call Grau( be( 2, 1, k1 ) )
c      call Grau( be( 3, 1, k1 ) )

c      write(6,*)' 1a. diada : '
c      write(6,*)
c      write(6,*)' beta 2 = ', be( 2, 1, k1 )
c      write(6,*)
c      write(6,*)' beta 3 = ', be( 3, 1, k1 )
c      write(6,*)

c      call Rad( be( 2, 1, k1 ) )
c      call Rad( be( 3, 1, k1 ) )

c ****
c      Calculo de W e Z - 1a. diada
c ****

call sistema( alf(2), alf(3), be(2,1,k1), be(3,1,k1), d(2,1),
1           d(2,2), d(3,1), d(3,2), w(1,1), w(1,2),
1           z(1,1), z(1,2) )

c      call Grau( be( 2, 2, k2 ) )
c      call Grau( be( 3, 2, k2 ) )

c      write(6,*)' 2a. diada : '
c      write(6,*)
c      write(6,*)' beta 2 = ', be( 2, 2, k2 )
c      write(6,*)
c      write(6,*)' beta 3 = ', be( 3, 2, k2 )
c      write(6,*)
c      write(6,*)

c      call Rad( be( 2, 2, k2 ) )
c      call Rad( be( 3, 2, k2 ) )

c ****
c      Calculo de W e Z - 2a. diada
c ****

call sistema( alf(2), alf(3), be(2,2,k2), be(3,2,k2), d(2,1),

```

```

1           d(2,2), d(3,1), d(3,2), w(2,1), w(2,2),
1           z(2,1), z(2,2) )

c *****
c      Calculo de z1, z3, z5, z6, argz1, argz6
c *****

c      z1
c      call Polar( w( 1, 1 ), w( 1, 2 ), mz(1), argz(1) )
c      z3
c      call Polar( w( 2, 1 ), w( 2, 2 ), mz(3), argz(3) )
c      z5
c      z( 5, 1 ) = z( 1, 1 ) - z( 2, 1 )
c      z( 5, 2 ) = z( 1, 2 ) - z( 2, 2 )
c      call Polar( z( 5, 1 ), z( 5, 2 ), mz(5), argz(5) )
c      z6
c      z( 6, 1 ) = w( 1, 1 ) + z( 5, 1 ) - w( 2, 1 )
c      z( 6, 2 ) = w( 1, 2 ) + z( 5, 2 ) - w( 2, 2 )
c      call Polar( z( 6, 1 ), z( 6, 2 ), mz(6), argz(6) )

c *****      Calculo da funcao      *****
c
c      call Erro34( argz(1), argz(6), mz(1), mz(3),
1           mz(5), mz(6), jeito )

500 write(*,*)' Por incrivel que pareca : F = ', jeito

      f = jeito

      nfe = nfe + 1

      return

      end

```

```

Subroutine Const( x, con )

Implicit double precision ( A-H, O-Z )

Dimension x( 1 ), con( 1 )

Common / one/ nfe, nce
1           / igrdcl/ igrd

nce = nce + 1

c *****
c   1a. restricao : 4 - ( x(1) - 5 )**2 >= 0
c *****
c   con( 1 ) = 4 - ( x(1) - 5 )**2
c *****
c   2a. restricao : 4 - ( x(2) - 0 )**2 >= 0
c *****
c   con( 2 ) = 4 - ( x(2) - 0 )**2
c *****
c   3a. restricao : 4 - ( x(3) - 0 )**2 >= 0
c *****
c   con( 3 ) = 4 - ( x(3) - 0 )**2
c *****
c   4a. restricao : 4 - ( x(4) - 0 )**2 >= 0
c *****
c   con( 4 ) = 4 - ( x(4) - 0 )**2

      return

      end

```

```

Subroutine Erro34( argz1, argz6, z1, z3, z5, z6, jeito )
integer i, n
double precision argz1, z1, z3, z5, z6, dth, pi,
1           m( 25 ), th( 25 ), argz6, jeito
pi = 3.141592654
n = 10
c   n = 5
c   n = 20
c   n = 3
c   n = 2
c   n = 1
dth = 2 * pi / n
do 100 i = 1, n
    th( i ) = argz1 + ( i-1 ) * dth - argz6
    m( i ) = angtr34( th( i ), z1, z3, z5, z6 )
100 continue
jeito = 0.0
do 200 i = 1, n+1
    jeito = jeito + ( m( i ) - pi/2.0 )**2
200 continue
write(*,*)' ERRO = ', jeito
return
end

```

```

Function angtr34 ( ang, z1, z3, z5, z6 )
double precision ang, z1, z3, z5, z6, pi,
1           d, d2, smi, cmi, t

pi = 3.141592654

c *****
c      Calculo do modulo da diagonal
c *****

d2 = z1**2 + z6**2 - 2 * z1 * z6 * cos( ang )

if( d2.lt.0.0 ) then

c      write(*,*) ' **** 7 **** '
t = - 10

goto 100

endif

d = sqrt( d2 )

c      write(*,*) ' **** 8 **** '

c      write(*,*) ' d = ', d

c *****
c      Calculo do angulo de transmissao
c *****

cmi = ( z3**2 + z5**2 - d**2 )/( 2 * z3 * z5 )

c      if( cmi.gt.1.or.cmi.lt.-1.or.cmi.eq.0 ) then

if( cmi.ge.1.or.cmi.le.-1 ) then

t = - 10

goto 100

endif

c      write(*,*) ' co-seno de mi = ', cmi
smi = sqrt( 1 - cmi*cmi )

c      write(*,*) ' seno de mi = ', smi
t = atg( smi, cmi )

```

```
call grau( t )
write(*,*), mi =', t
call rad( t )
100 angtr34 = t
return
end
```

```

Subroutine trian(x1,x2,x3,ag1,ag2,ag3,b2,b2t,b3,b3t,j)
integer j .

Double Precision cg, sg, g,
1      x1, x2, x3, ag1, ag2, ag3,
1      b2, b2t, b3, b3t,
1      f, cf, sf

c *****
c ***** Calculo de BET #2, #3      *****
c *****

cg = ( x1*x1 + x3*x3 - x2*x2 )/( 2.0*x1*x3 )
if( cg.gt.1.or.cg.lt.-1 ) then
    j = 1
    goto 50
endif

sg = sqrt( 1-cg*cg )
g = atg ( sg,cg )
b2 = ag3 + g - ag1
b2t = ag3 - g - ag1
cf = ( x3*x3 + x2*x2 - x1*x1 )/( 2.0*x2*x3 )
if( cf.gt.1.or.cf.lt.-1 ) then
    j = 1
    goto 50
endif

c     write(*,*) ' cos fi = ',cf
sf = sqrt ( 1-cf*cf )
c     write(*,*) ' sen fi = ',sf
f = atg ( sf,cf )
c     write(*,*)' fi = ',f
b3 = ag3 - f - ag2
b3t = ag3 + f - ag2

```

```
call checan ( b2 )
call checan ( b2t )
call checan ( b3 )
call checan ( b3t )

c      write(*,*) ' Bet2 = ',b2
c      write(*,*) ' Bet2 til = ',b2t
c      write(*,*) ' Bet3 = ',b3
c      write(*,*) ' Bet3 til = ',b3t

50  return
end
```

```

Subroutine Trian4( a, b, c, arga, bx, by, cx, cy, i, j )

integer i, j

Double Precision a, b, c, arga, ax, ay, bx, by,
1           cx, cy, cg, sg, gama

c      write(*,*)' a='
c      read(*,*) a
c      ,a
c      write(*,*)' b='
c      read(*,*) b
c      ,b
c      write(*,*)' c='
c      read(*,*) c
c      ,c

c *****
c      Calculo de co-seno de Gama
c *****

cg = ( a*a + b*b - c*c )/(2*a*b)

write(*,*)' cg= ',cg

c *****
c      Verificacao: cg > 1 ? - Condicao de travamento
c *****

if( cg.gt.1.or.cg.lt.-1 ) then

    j = 1

    goto 100

endif

c *****
c      Calculo de Seno de Gama
c *****

sg = sqrt( 1 - cg*cg )

c      write(*,*)' sg= ',sg

c *****
c      Calculo de Gama
c *****

gama = atg( sg, cg)

c      call Grau( gama)

c      write(*,*) gama

c      call Rad( gama)

```

```

c **** Testes ****
c      write(*,*)' arga= '
c      read(*,*) arga
c      call Rad( arga)

c **** Conversao de a de coord. polares p/ cartes.
c ****

      call Cartes( a, arga, ax, ay)

c **** Testes do indice i p/ escolha da solucao correta
c

c      Calculo de bx e by
c ****

      if( i.eq.1 ) then
          bx = b * cos( arga + gama )
          by = b * sin( arga + gama )
      else
          if( i.eq.2 ) then
              bx = b * cos( arga - gama )
              by = b * sin( arga - gama )
          endif
      endif

      write(*,*)' bx= ',bx
      write(*,*)' by= ',by

c **** Calculo de Cx e Cy
c ****

      cx = bx - ax
      write(*,*)' cx= ',cx
      cy = by - ay
      write(*,*)' cy= ',cy

```

```
100    return  
c      stop  
end
```

```
Subroutine Grau(th)
Double Precision th,pi
```

```
pi=3.141592654
th=180.*th/pi
```

```
return
end
```

```
Subroutine Rad(th)
Double Precision th,pi
```

```
pi=3.141592654
th=pi*th/180.
```

```
return
end
```

```
Subroutine Polar (vx,vy,v,th)
Double Precision vx,vy,v,th
```

```
v = sqrt(vx*vx + vy*vy)
```

```
th=atg(vy,vx)
```

```
return
end
```

```

Subroutine Cartes (v,th,vx,vy)
Double Precision v,th,vx,vy

vx = v*cos(th)
vy = v*sin(th)

return
end

Subroutine Pvet ( ax, ay, bx, by, vet )
double precision ax, ay, bx, by, vet

c *****
c           Calculo de VET
c *****

vet = ax * by - ay * bx

return
end

Function Atg(vy,vx)
Double Precision vy,vx,pi

pi=3.141592654

if(vy.ne.0.0.and.vx.ne.0.0) then
  Atg=atan2(vy,vx)
elseif(vy.eq.0.0.and.vx.ne.0.0)then
  if(vx.gt.0.0) then
    Atg=0.0
  else
    Atg=pi
  endif
else
  if(vy.ne.0.0.and.vx.eq.0.0) then
    if(vy.gt.0.0) then
      Atg=pi/2
    else
      Atg=-pi/2
    endif
  endif
endif

return
end

```

**11.6 - Saída da sub-rotina OPT3 para otimização
do ângulo de transmissão na síntese para
4 posições e 13 variáveis**

```
*****
*****  

**          **  

**          OPT 3          **  

**          **  

**      VERSION 2 (DOUBLE PRECISION): LAST REVISION FEBRUARY 1989. **  

**      COPYRIGHT, PURDUE RESEARCH FOUNDATION, 1976.           **  

**      COPYRIGHT, K. M. RAGSDELL, 1989.           **  

**          **  

*****  

*****
```

1

STARTING INFORMATION

OBJECTIVE FUNCTION.....	LFUNCT =	0 NL
NUMBER OF DESIGN VARIABLES.....	N =	13
NUMBER OF EQUALITY CONSTRAINTS....	NE =	0
LINEAR OF EQUALITY CONSTRAINTS...	LNE =	0
NUMBER OF INEQUALITY CONSTRAINTS...NI =	9	
LINEAR OF INEQUALITY CONSTRAINTS.LNI =	0	
LINE SEARCH CRITERIA.....EPSLS =	.10000E-05	
BOUND TIGHTNESS.....EPSBD =	.10000E-07	
VARIABLE CONVERGENCE CRITERIA....XTOL =	.10000E-06	
GRADIENT CONVERGENCE CRITERIA....GTOL =	.10000E-03	
FORWARD DIFFERENCE PARAMETER.....EPS =	.10000E-03	
OUTPUT PARAMETER.....IPR =	1	
INITIAL OUTPUT PARAMETER.....IDATA =	1	
MAXIMUM NUMBER OF ITERATIONS.....MAXM =	15	
SCALING PARAMETER.....NSCALE =	2 AUTO	
SEARCH PARAMETER.....ISEARC =	1 BFS	
MAXIMUM NEWTON ITERATIONS.....MAXNEW =	10	
NUMERICAL DIFFERENCING OPTION....IDIF =	2 CENT	

OBJECTIVE FUNCTION F(X) = .473998E+01

INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.66322513E+00		.25000000E+00
2	-.12217305E+00		.25000000E+00
3	.17453293E+00		.12187082E-02
4	.34906586E+00		.12187082E-02
5	.52359879E+00		.12187082E-02
6	.00000000E+00		.25000000E+00
7	.00000000E+00		.25000000E+00
8	.50000000E+01		.25000000E+00
9	.80000000E+01		.25000000E+00
10	.10000000E+02		
11	.15000000E+02		
12	.18000000E+02		
13	.20000000E+02		

INDEX	LOWER BOUND	UPPER BOUND	AUTO SCALE FACTOR
1	-.31415930E+01	.31415930E+01	.10000000E+01
2	-.31415930E+01	.31415930E+01	.10000000E+01
3	-.31415930E+01	.31415930E+01	.10000000E+01
4	-.31415930E+01	.31415930E+01	.10000000E+01
5	-.31415930E+01	.31415930E+01	.10000000E+01
6	-.10000000E+02	.30000000E+02	.10000000E+01
7	-.10000000E+02	.30000000E+02	.10000000E+01
8	.00000000E+00	.30000000E+02	.10000000E+01
9	.00000000E+00	.30000000E+02	.10000000E+01
10	.00000000E+00	.30000000E+02	.10000000E+02
11	.00000000E+00	.30000000E+02	.10000000E+02
12	.00000000E+00	.30000000E+02	.10000000E+02
13	.00000000E+00	.30000000E+02	.10000000E+02

CONSTRAINT SCALE FACTORS

.100000E+01
 .100000E+01
 .100000E-01
 .100000E-01
 .100000E-01
 .100000E+01
 .100000E+01
 .100000E+01
 .100000E+01

TOP OF D ARRAY AT 811

1

OPTIMIZATION USING THE REDUCED GRADIENT METHOD OPT3.2

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .105690E+03

OBJECTIVE F(X) =	.4404453015E+01	FUNCTION EVALUATIONS	38
ITERATION 1: ITERATIONS SINCE BFS RESET		0	CONSTRAINT
EVALUATIONS 56			

INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.66353066E+00		.24999991E+00
2	-.12174328E+00		.24999982E+00
3	.17640668E+00		.12151862E+00
4	.34715123E+00		.12150419E+00
5	.52423305E+00		.12183074E+00
6	.11827222E-04		.25000000E+00
7	.21647804E-04		.25000000E+00
8	.49999819E+01		.24998004E+00
9	.79999451E+01		.24999887E+00
10	.10000746E+02		
11	.15004405E+02		
12	.18000033E+02		
13	.19998939E+02		

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .339110E+02

OBJECTIVE F(X) =	.3616187188E+01	FUNCTION EVALUATIONS*****
------------------	-----------------	---------------------------

ITERATION 2: ITERATIONS SINCE BFS RESET 1 CONSTRAINT EVALUATIONS*****

INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.67386620E+00		.24988676E+00
2	-.10544276E+00		.24972010E+00
3	.16434009E+00		.11148738E+00
4	.33729993E+00		.10802680E+00
5	.53351268E+00		.11204466E+00
6	-.14534803E-03		.24999985E+00
7	.36290263E-03		.24999908E+00
8	.50004702E+01		.20880443E+00
9	.79991627E+01		.24765772E+00
10	.99809636E+01		
11	.15202072E+02		
12	.18012342E+02		
13	.19953203E+02		

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .461260E+02

OBJECTIVE F(X) = .3390793518E+01 FUNCTION EVALUATIONS*****
ITERATION 3: ITERATIONS SINCE BFS RESET 2 CONSTRAINT EVALUATIONS*****

INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.67430363E+00		.24987726E+00
2	-.10418789E+00		.24967654E+00
3	.16415369E+00		.11110404E+00
4	.33709922E+00		.10755046E+00
5	.53347521E+00		.11211881E+00
6	.18055218E-03		.24998732E+00
7	.35568996E-02		.24999915E+00
8	.50004315E+01	L	.48161236E-09
9	.79991844E+01		.23588113E+00
10	.99547054E+01		
11	.15497944E+02		
12	.18029803E+02		
13	.19884975E+02		

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .331483E+02

OBJECTIVE F(X) = .3339872398E+01 FUNCTION EVALUATIONS*****
ITERATION 4: ITERATIONS SINCE BFS RESET 0 CONSTRAINT EVALUATIONS*****

INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.67431566E+00		.24987700E+00
2	-.10414120E+00		.24967485E+00
3	.16495149E+00		.11269604E+00
4	.33655721E+00		.10622385E+00
5	.53355914E+00		.11195234E+00
6	.19587659E-03		.24998730E+00
7	.35576688E-02		.24999915E+00
8	.50004146E+01	L	-.88698308E-08
9	.79991775E+01		.23586385E+00
10	.99558892E+01		
11	.15498050E+02		

12 .18029599E+02
13 .19884848E+02

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .242769E+01

ITERATION	OBJECTIVE F(X) = .3203569503E+01	FUNCTION EVALUATIONS*****	
	5: ITERATIONS SINCE BFS RESET	1 CONSTRAINT EVALUATIONS*****	
INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.68302748E+00		.24960786E+00
2	-.84708483E-01		.24859641E+00
3	.15949446E+00		.99264067E-01
4	.32526157E+00		.65205801E-01
5	.53758197E+00		.10232122E+00
6	-.57448165E-02		.24994284E+00
7	.49149433E-02		.24999750E+00
8	.50006867E+01	L	-.16919518E-08
9	.79985771E+01		.23372614E+00
10	.99468465E+01		
11	.15497167E+02		
12	.18032361E+02		
13	.19876604E+02		

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .470394E+02

ITERATION	OBJECTIVE F(X) = .3140115613E+01	FUNCTION EVALUATIONS*****	
	6: ITERATIONS SINCE BFS RESET	2 CONSTRAINT EVALUATIONS*****	
INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.68535330E+00		.24951034E+00
2	-.75579018E-01		.24782900E+00
3	.15532545E+00		.84989360E-01
4	.31961768E+00		.35150540E-01
5	.53988966E+00		.95335468E-01
6	-.65480178E-02		.24990721E+00
7	.70648129E-02		.24999834E+00
8	.50002544E+01	L	.50892794E-08
9	.79987357E+01		.21635229E+00
10	.99671186E+01		
11	.15498918E+02		
12	.18033373E+02		
13	.19819628E+02		

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .307691E+02

SKIPPING BFS UPDATE

ITERATION	OBJECTIVE F(X) = .3062376592E+01	FUNCTION EVALUATIONS*****	
	7: ITERATIONS SINCE BFS RESET	3 CONSTRAINT EVALUATIONS*****	
INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.68595323E+00		.24948343E+00
2	-.68389493E-01		.24710733E+00
3	.15690394E+00		.90803002E-01

4	.32493988E+00	.63663883E-01
5	.53190485E+00	.11497373E+00
6	-.11219203E-01	.24978692E+00
7	.93385043E-02	.24999842E+00
8	.49987514E+01	L -.63669855E-09
9	.79998628E+01	.22936172E+00
10	.10175794E+02	.
11	.15468077E+02	.
12	.17997045E+02	.
13	.19856370E+02	.

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .198673E+02

ITERATION	OBJECTIVE F(X) = .2984594338E+01	FUNCTION EVALUATIONS*****	
	8: ITERATIONS SINCE BFS RESET	4 CONSTRAINT EVALUATIONS*****	
INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.68921852E+00		.24932434E+00
2	-.55438162E-01		.24554646E+00
3	.15281670E+00		.74724064E-01
4	.32054188E+00		.40508354E-01
5	.53167941E+00		.11534309E+00
6	-.10630026E-01		.24976941E+00
7	.10843969E-01		.24999608E+00
8	.49980295E+01	L -.57328129E-08	.
9	.80001819E+01		.21534341E+00
10	.10307077E+02		.
11	.15394593E+02		.
12	.17983587E+02		.
13	.19814562E+02		.

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .165175E+02

SKIPPING BFS UPDATE

ITERATION	OBJECTIVE F(X) = .2943612885E+01	FUNCTION EVALUATIONS*****	
	9: ITERATIONS SINCE BFS RESET	5 CONSTRAINT EVALUATIONS*****	
INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.69093796E+00		.24923199E+00
2	-.46892318E-01		.24433282E+00
3	.14839171E+00		.53549776E-01
4	.31415599E+00	L -.29802644E-09	.
5	.53325579E+00		.11254736E+00
6	-.10092545E-01		.24974868E+00
7	.12225596E-01		.24999390E+00
8	.49975691E+01	L -.69647216E-08	.
9	.80004321E+01		.19662193E+00
10	.10367831E+02	.	.
11	.15338674E+02	.	.
12	.17971656E+02	.	.
13	.19770708E+02	.	.

MAXIMUM ELEMENT OF RELATIVE REDUCED GRADIENT = .106856E+02

OBJECTIVE F(X) =	.2943612885E+01	FUNCTION EVALUATIONS*****	
ITERATION 10: ITERATIONS SINCE BFS RESET	0	CONSTRAINT EVALUATIONS*****	
INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.69093796E+00		.24923199E+00
2	-.46892318E-01		.24433282E+00
3	.14839171E+00		.53549776E-01
4	.31415599E+00	L	-.29802644E-09
5	.53325579E+00		.11254736E+00
6	-.10092545E-01		.24974868E+00
7	.12225596E-01		.24999390E+00
8	.49975691E+01	L	-.69647216E-08
9	.80004321E+01		.19662193E+00
10	.10367831E+02		
11	.15338674E+02		
12	.17971656E+02		
13	.19770708E+02		

RELATIVE CHANGE IN X VECTOR <= XTOL

*** OPTIMUM FOUND TO BE ***

*****	+ FINAL DESIGN +	*****	
OBJECTIVE F(X) =	.2943612885E+01	TOTAL FUNCTION EVALUATIONS*****	
ITERATION 10		TOTAL CONSTRAINT EVALUATIONS*****	
INDEX	X(I)	EQUALITY H(I)	INEQUALITY G(I)
1	-.69093796E+00		.24923199E+00
2	-.46892318E-01		.24433282E+00
3	.14839171E+00		.53549776E-01
4	.31415599E+00	L	-.29802644E-09
5	.53325579E+00		.11254736E+00
6	-.10092545E-01		.24974868E+00
7	.12225596E-01		.24999390E+00
8	.49975691E+01	L	-.69647216E-08
9	.80004321E+01		.19662193E+00
10	.10367831E+02		
11	.15338674E+02		
12	.17971656E+02		
13	.19770708E+02		

LAGRANGE MULTIPLIERS OF ACTIVE CONSTRAINTS

INEQUALITY CONSTRAINT	4 = .193864E+01
INEQUALITY CONSTRAINT	8 = .853262E-01

12. REFERÉNCIAS BIBLIOGRÁFICAS

1. Hartenberg, R. S. e Denavit, J. , *Kinematic Synthesis of Linkages*, Mc Graw Hill, 1964.
2. Erdman, A. G., *Computer-Aided Design of Mechanisms : 1984 and beyond*, Mechanism and Machine Theory, vol.20, No.4, pp 245-249, 1985.
3. Fox, R. L. e Gupta, K. C., *Optimization Technology as Applied to Mechanism Design*, Journal of Engineering for Industry, pp 657-663, Maio de 1973.
4. Root, R. R. e Ragsdell, K. M., *A Survey of Optimization Methods Applied to the Design of Mechanisms* , Journal of Engineering for Industry, pp 1036-1041, Agosto de 1976.
5. Freudenstein, F. e Sandor, G. N., *Synthesis of Path-generating Mechanisms by Means of a Programmed Digital Computer*, Journal of Engineering for Industry, ser.B, vol.81, No.2, pp 159-168, Maio de 1959.
6. Suh, C. H. e Radcliffe, C. W., *Synthesis of Plane Linkages with Use of Displacement Matrix*, Journal of Engineering for Industry, ser. B, vol.89, No.2, pp 206-214, Maio de 1967.

7. Shett, P. N. e Uicker, J. J., Jr., *IMP (Integrated Mechanism Program) : A Computer-Aided Design System for Mechanisms and Linkages*, Journal of Engineering for Industry, Setembro de 1971.
8. Kaufman, R. E., *KINSYN Phase II : A Human Engineering Computer System for Kinematic Design and a New Least-Square Synthesis Operator*, Mechanism and Machine Theory, vol.8 , No. 4, 1974.
9. Rubel, A. J. e Kaufman, R. E., *KINSYN III : A new Human-Engineering System for Interactive Computer-Aided Design of Planar-Linkages*, Journal of Engineering for Industry, pp 440-448, Maio de 1977.
10. Erdman, A. G. e Gustafson, J. E., *Linkages: Linkage Interactive Computer Analysis and Graphically Enhanced Synthesis Package*, ASME Paper No. 77-DET-5, pp 1-9, 1977.
11. Whalen, P. V. e Midha, A., *User-Friendly Interactive Computer-Aided Design of Four-Bar Based Linkages*, Computers in Engineering, West Lafayette IN, pp 154-161, 1984.
12. Midha, A. e Zhao, Z. L., *Synthesis of Planar Linkage Via Loop Closure and Nonlinear Equations Solution*, Proceedings of the 8th. OSU Applied Mechanisms Conference, St. Louis,

- Missouri, Mechanism and Machine Theory, vol. 20, No.6, pp
491-502, 1985.
13. Peterson, R., Logan, L., Erdman, A. G., Riley, D. R., *Three Precision Point Synthesis of Four-Bar linkage : An Example using the LINCAGES-4 Program*, Computers in Engineering, San Francisco CA, pp 91-96, 1988.
14. Ardayfio, D. D., Kapur, R., Yang, S. B., Watson, W. A., *MICRAS: Microcomputerinteractive codes for robot analysis and simulation*, Mechanism and machine Theory, vol. 20, No. 4, pp 271-284, 1985.
15. Norton, R. L., *Four-Bar and geared five bar linkage analysis programs for the Apple computer*, Mechanism and Machine Theory, vol. 20, No. 4, pp 313-320, 1985.
16. Himmelblau, D. M., *Applied Nonlinear Programming*, Mc Graw Hill, 1972.
17. Kramer, S. N. e Sandor, G. N., *Selective Precision Synthesis - A General Method of Optimization for Planar Mechanisms*, Journal of Engineering for Industry, vol. 97, No. 2, pp 689-701, 1975
18. Gabriele, G. A. e Ragsdell, K. M., *The Generalized Reduced Gradient Method : A Reliable Tool for Optimal Design*,

- Journal of Engineering for Industry, vol. 99, pp 394-400,
1977.
19. Gupta, K. C., *Design of Four-Bar Function Generation with
Mini-Max Transmission Angle*, Journal of Engineering for
Industry, pp 360-366, Maio de 1977.
20. Suh, C. H. e Radcliffe, C. W., *Kinematics and Mechanism
Design*, John Wiley & Sons, 1978.
21. Avilés, R., Ajuria, M. B. e Jalón, J. G., *A Fairly General
Method for Optimum Synthesis of Mechanisms*, Mechanism and
Machine Theory, vol. 20, No. 4, pp 321-328, 1985.
22. Lima, C. S., *Optimum Dynamic Synthesis of Single Degree of
Freedom Planar Linkages*, tese de doutorado apresentada na
Universidade da Flórida, Maio de 1985.
23. Rigelman, G. A. e Kramer, S. N., *A Computer-Aided Design
Technique for the Synthesis of Planar Four-Bar Mechanisms
Satisfying Specified Kinematic and Dynamic Conditions*,
Journal of Mechanisms, Transmissions, and Automation in
Design, vol. 110, pp 263-268, Setembro de 1988.
24. Erdman, A. G., Sandor, G. N., *Mechanism Design : Analysis and
Synthesis*, vol. 1, Prentice-Hall, 1984.

25. Freudenstein, F. e Sandor, G. N., *On the Burmester Points of a Plane*, Journal of Applied Mechanics, vol. 28, ser. E, No. 1, pp 41-49, Março de 1961.
26. Tao, D. C., *Applied Linkage Synthesis*, Addison-Wesley, 1964.
27. Sandor, G. N. e Erdman, A. G., *Advanced Mechanism Design : Analysis and Synthesis*, vol. 2, Prentice-Hall, 1984.
28. Gabriele, G. A. e Ragsdell, K. M., *OPT 3.2-A Nonlinear Programming Code in Fortran Implementing the Generalized Reduced Gradient Method*, Design Productivity Center, University of Missouri-Rolla, 1989.