ANTENOR DE FREITAS FILHO

OTIMIZAÇÃO DOS ÂNGULOS DE TRANSMISSÃO NA SÍNTESE DE MECANISMOS PLANOS ARTICULADOS, MOVIDOS POR ACIONADORES PRISMÁTICOS

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

Orientador : Prof. Dr. Tarcísio Antônio Hess Coelho
Professor Doutor do Departamento de
Engenharia Mecânica da EPUSP

São Paulo , 1999

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

Gostaria de agradecer ao meu orientador Prof. Dr. Tarcísio A. Hess Coelho cujo apoio, perseverança e dedicação foram decisivos para a conclusão do presente trabalho.

À minha esposa Josiane e ao meu filho Gabriel que compreenderam o sacrifício a que foram submetidos para que esta dissertação fosse elaborada.

À minha mãe e à memória de meu pai que sempre apoiaram as minhas decisões.

E finalmente aos amigos que direta ou indiretamente deram suas parcelas de colaboração.

Resumo

Este trabalho trata da síntese ótima de mecanismos articulados planos movidos por acionadores prismáticos. O objetivo da síntese é a geração de movimento com especificação de três posições de precisão. Para a realização da otimização, empregou-se o método do Gradiente Reduzido Generalizado. Os parâmetros otimizados são os ângulos de transmissão dos mecanismos acionador e posicionador. São apresentados três exemplos onde se comprova a eficácia dos programas desenvolvidos

Abstract

This work treats optimum synthesis of planar linkages with prismatic actuators.

The synthesis goal is motion generating prescribing three precision positions. In order to perform optimization, the Generated Gradient Method is applied. The optimized parameters are transmission angles of positioning and actuating mechanisms. Three aplication exemples are presented so the effectiveness of developed problems is verified.

Índice

Lista de Símbolos

$\overline{AA_0}$	letras com barras, componentes (barras) do mecanismo
a	minúscula normal, escalar
a	minúscula negrito, vetor
w e **z**	vetores da 1ª díada do quadrilátero articulado
w' e **z**'	vetores da 2ª díada do quadrilátero articulado
F(x)	função objetivo
ν_1	ângulo de orientação do mecanismo acionador em relação ao posicionador
ν_2	ângulo do mecanismo posicionador em relação ao acionador
μ	ângulo de transmissão do quadrilátero articulado (mecanismo posicionador)
η	ângulo de transmissão entre o cilindro e a engrenagem (pinhão) – mecanismo acionador
α_j	rotação angular do acoplador da posição 1 para a j
β_j	rotação angular do elemento conectado à estrutura da posição 1 para a j
i	número complexo, seu valor é $\sqrt{-1}$

D_0 e M	articulação fixa da engrenagem maior (coroa)
$\exp(\alpha)$	função exponencial de α, indica e^{α}
A_0, B_0 e C_0	articulações fixas
A, B e C	articulações móveis
P	ponto de interesse do acoplador
$\mathbf{R}_j$	vetor-posição que localiza a posição j com respeito a um eixo de coordenadas X-Y fixo na estrutura

Índice de Figuras

Capítulo 1 – Introdução

1.1– Considerações preliminares

Define-se mecanismo em projeto mecânico como um dispositivo transformador de movimento, sendo que, dado um movimento de entrada disponível, o mecanismo deve fornecer um movimento de saída desejado (SANDOR; ERDMAN, 1984). Em geral, o termo mecanismo refere-se a mecanismos de barras, engrenagens e camo-seguidor.

A importância do estudo de mecanismos reside no fato deles serem elementos básicos de qualquer máquina ou equipamento. As suas aplicações nos setores industrial e agrícola são inúmeras. Dentre elas podem ser citadas: automação industrial, máquinas de levantamento e transporte, máquinas e implementos agrícolas. Com relação à automação industrial, a teoria de mecanismos vem sendo utilizada para projeto de manipuladores robóticos, de modo que se possa realizar simulação tridimensional, estimativa da região de trabalho e análise da resposta dinâmica. Outra área de aplicação mais recente é a bio-mecânica onde o projeto de próteses de mãos, de joelhos e pés seriam alguns exemplos.

O conhecimento acumulado até o presente momento sobre o projeto de mecanismos se concentrou principalmente em como modificações realizadas na geometria de um dado mecanismo alteram as suas características cinemáticas e dinâmicas. As características cinemáticas estão relacionadas, por exemplo, com posicionamento de precisão, tempos de avanço e retorno. Por outro lado, as características dinâmicas lidam, por exemplo, com a redução dos esforços de acionamento e balanceamento (COELHO, 1997).

Tradicionalmente, os métodos de projeto – síntese – de mecanismos são analíticos e envolvem a busca de expressões matemáticas que, uma vez obtidas, tornam possível por substituição direta dos parâmetros conhecidos, a determinação de incógnitas como os comprimentos das barras e posições das articulações fixas. Contudo, estes métodos apresentam limitações quando se pretende satisfazer, simultaneamente, diferentes especificações de projeto sujeitas a um grande número de restrições físicas. Devido a isso, existe atualmente um interesse crescente pelo emprego de métodos numéricos iterativos para otimização das características cinemáticas e dinâmicas dos mecanismos por oferecerem uma perspectiva mais promissora e superior se comparados aos métodos analíticos.

1.2 - Objetivos

Esta pesquisa tem por objetivo estudar a síntese cinemática de mecanismos articulados planos movidos por acionadores prismáticos. As especificações de síntese prevêem o posicionamento de precisão da barra acopladora, segundo o critério de otimização das variáveis ângulos de transmissão. Utilizar-se-á a técnica numérica conhecida como "Método do Gradiente Reduzido Generalizado", para determinação da geometria ideal do mecanismo, de modo a atender ao critério mencionado e outros requisitos de ordem construtiva. Para verificação dos modelos matemático e computacional, serão abordados três exemplos de aplicação que permitirão avaliar a eficácia da metodologia aplicada.

1.3 - Sobre a divisão do trabalho

A seguir será feito um resumo do conteúdo dos capítulos subseqüentes.

O capítulo 2 se inicia com uma revisão da literatura apresentando conceitos básicos sobre mecanismos, tais como, suas partes constitutivas, seus graus de mobilidade e as tarefas da síntese cinemática. Este capítulo ainda contém comentários sobre várias publicações científicas envolvendo "projeto de mecanismos auxiliado por computador" e "otimização no projeto de mecanismos".

Após a introdução da teoria, serão apresentadas no capítulo 3, a formulação do problema que esta dissertação se propõe a resolver, a abordagem utilizada e as equações principais para se desenvolver um projeto de um mecanismo articulado, incluindo a equação da díada.

O capítulo 4 trata do "Método do Gradiente Reduzido Generalizado", sobre o qual é baseado toda a otimização desenvolvida nessa dissertação. Também é apresentado um exemplo de aplicação envolvendo a minimização de uma função de duas variáveis.

No capítulo 5 é apresentada a estrutura dos programas desenvolvidos, além de uma descrição mais detalhada de cada uma de suas sub-rotinas.

A seguir apresenta-se, no capítulo 6, três exemplos de aplicação dos programas desenvolvidos. Os mecanismos são sintetizados sem a sub-rotina de otimização e, depois, com a sub-rotina de otimização. São apresentados gráficos comparativos antes e depois da otimização para melhor visualização das alterações ocorridas.

No capítulo 7 é realizada uma discussão dos resultados obtidos no capítulo anterior, bem como da metodologia empregada.

No capítulo 8 são apresentadas as conclusões e os trabalhos que futuramente poderão ser desenvolvidos nesta linha de pesquisa.

No capítulo 9 são apresentadas, como apêndice, noções básicas sobre a notação polar complexa e a listagem dos programas desenvolvidos.

Finalmente, no capítulo 10, encontram-se todas as referências bibliográficas utilizadas.

Capítulo 2 – Revisão da Literatura

2.1 – Introdução

Por questões de ordem didática, a revisão da literatura se inicia com algumas considerações básicas sobre mecanismos. A seguir, trata-se da síntese de mecanismos, abordando-se os seguintes tópicos: projeto de mecanismos auxiliado por computador e otimização no projeto de mecanismos.

2.2 – Principais partes de um mecanismo

Um mecanismo é formado por peças ou barras e pelas conexões entre as peças (HARTENBERG; DENAVIT, 1964). Uma dessas peças é chamada de base ou peça fixa pois não realiza qualquer movimento. Outras peças são chamadas de motoras porque estão ligadas diretamente aos motores ou acionadores do mecanismo. As demais peças são ditas peças movidas.

As conexões são os elementos de ligação entre as peças. São também chamados de pares cinemáticos (SUH; RADCLIFFE, 1978). Há diversos tipos de pares cinemáticos. Existem os pares de revolução (ou de rotação), os prismáticos (ou de translação), os helicoidais, os cilíndricos, os esféricos, os planos, dentre outros. Cada tipo de par cinemático permite um determinado número de graus de liberdade entre as peças por ele conectadas. Como cada tipo de par cinemático possui uma forma geométrica básica, mas pode assumir diferentes formas construtivas, adotam-se algumas representações esquemáticas para facilitar o seu desenho na estrutura do mecanismo.

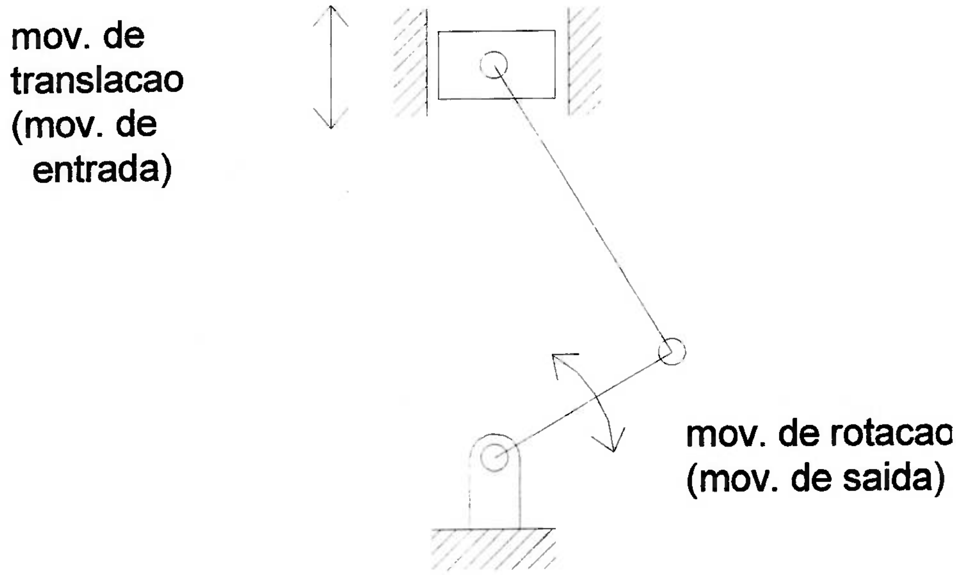


Fig. 2.2.1 - Mecanismo biela-manivela com indicação do movimento de entrada e de saída.

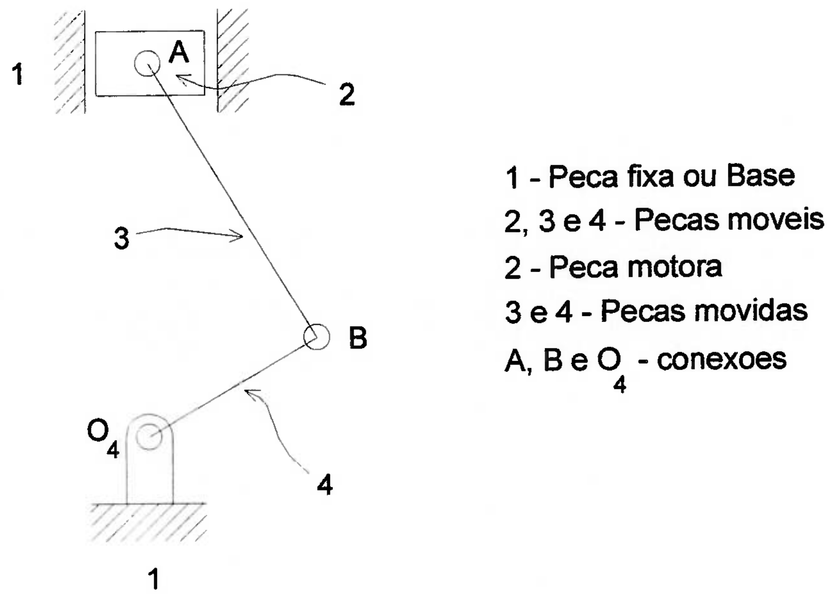


Fig 2.2.2 – Mecanismo biela-manivela com indicação das peças com nomes e conexões.

2.3 - Grau de liberdade e grau de mobilidade

Grau de liberdade é o número total de movimentos disponíveis e independentes de uma peça em relação à outra, permitidos pelo par cinemático que une as duas peças.

Grau de mobilidade (HARTENBEG; DENAVIT,1964) é o número de variáveis independentes necessárias para definir completamente a posição relativa de todas as peças de um mecanismo.

Para a determinação do grau de mobilidade (G M) de um mecanismo, utilizam-se dois critérios conhecidos como critérios de Grübler (SHIGLEY; UICKER, 1981), um para mecanismos planos e outro para espaciais.

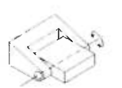
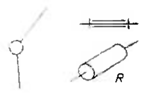
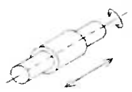
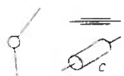
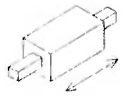
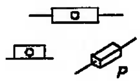
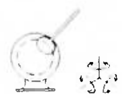
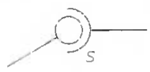
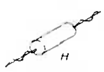
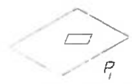
1. Revolução (R)			1
2. Cilíndrico (C)			2
3. Prismático (P)			1
4. Esférico (S)			3
5. Helicoidal (H)			1
6. Plana (P_L)			3

Fig. 2.3.1 - Tipos de pares cinemáticos (SUH; RADCLIFFE,1978)

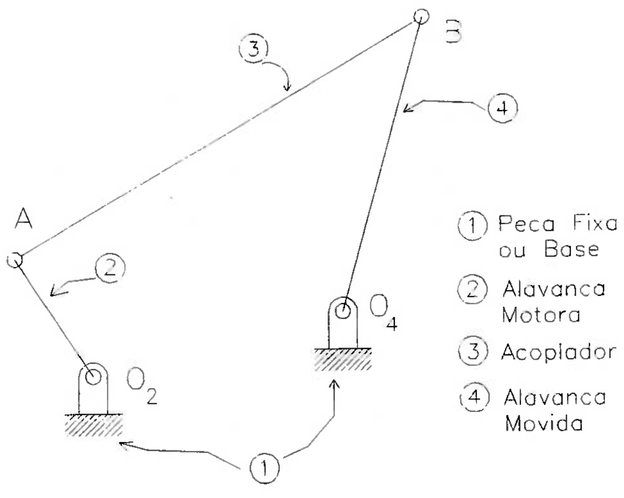


Fig. 2.3.2 - Mecanismo quadrilátero articulado

Para mecanismos planos,

$$G.\,M. = 3\,.\,(n - 1) - 2\,.\,n_{p1} - n_{p2} \qquad (2.3.1)$$

e para espaciais,

$$G.\,M. = 6.\,(\,n - 1\,) - 5.n_{p1} - 4.n_{p2} - 3.n_{p3} - 2.n_{p4} - n_{p5} \qquad (2.3.2)$$

sendo que **n** é o número total de peças e $\mathbf{n_{pj}}$ o número total de pares que permitem **j** ($1 \leq \mathbf{j} \leq 5$) graus de liberdade. Para o quadrilátero articulado, $n = 4$, $n_{p1} = 4$, $n_{p2} = 0$, portanto, $G.M. = 1$.

2.4 - Tipos de mecanismos

Tradicionalmente, incluem-se no grupo dos mecanismos (SUH; RADCLIFFE, 1978) os formados por um conjunto de barras conectadas por diferentes tipos de pares cinemáticos, os engrenamentos de rodas dentadas e aqueles que possuem camo e seguidor.

2.5 - Mecanismos planos e espaciais ou tridimensionais (HARTENBERG, DENAVIT, 1964)

Os mecanismos podem ser planos ou espaciais. Os mecanismos planos são aqueles cujas peças realizam movimentos planos que sejam paralelos a um plano comum. Os mecanismos espaciais, ou contêm pelo menos uma de suas peças que execute movimento espacial, ou ainda possuem um número de peças que desenvolvam movimentos planos que não sejam paralelos a um plano comum.

2.6- Análise e síntese; Cinemática e dinâmica

Entende-se por análise o estudo do comportamento estático, cinemático, cineto-estático, dinâmico, ou ainda cineto-elasto-dinâmico, de um determinado tipo de mecanismo assumindo que sejam conhecidos os pares cinemáticos, a forma de suas peças, bem como suas dimensões.

A análise estática (ERDMAN; SANDOR, 1972) envolve a obtenção das forças e/ou momentos internos ao mecanismo a partir de forças e/ou momentos externos conhecidos aplicados ao mecanismo, assumindo que não haja qualquer movimento deste, ou seja, equilíbrio. A análise cinemática (ERDMAN; SANDOR, 1972) trata da obtenção das posições, velocidades e acelerações de pontos e peças do mecanismo, admitindo que o movimento da peça motora seja conhecido e que as peças sejam corpos rígidos. A análise cineto-estática (ERDMAN; SANDOR, 1984) envolve a obtenção das forças e/ou momentos internos ao mecanismo sujeito ou não a forças e/ou momentos externos, a partir dos resultados da análise cinemática. A análise dinâmica (ERDMAN; SANDOR, 1984) trata da determinação do movimento (posições, velocidades e acelerações) de pontos e peças do mecanismo a partir de uma posição inicial conhecida, em que uma ou mais peças estejam sujeitas à ação de forças ou momentos externos conhecidos (por exemplo, força da gravidade). Assume-se que as peças sejam corpos rígidos (ERDMAN; SANDOR, 1972) . A análise cineto-elasto-dinâmica (ERDMAN; SANDOR, 1972) é semelhante à análise dinâmica nos seus objetivos, mas considera que as peças do mecanismo sejam elásticas. Além disso, os efeitos de deformação elástica sobre as forças de inércia são incluídos neste tipo de análise.

Entende-se por síntese a determinação do tipo de mecanismo, seus pares cinemáticos, as formas e dimensões de suas peças de modo que ele satisfaça a especificações de projeto previamente estabelecidas (HARTENBEG; DENAVIT, 1964). A síntese pode ser cinemática ou dinâmica.

A síntese cinemática envolve a satisfação de especificações de posições - que é o caso mais comum - ou ainda de velocidades ou acelerações de uma ou mais peças, ou mesmo de pontos escolhidos do mecanismo. A síntese cinemática se distingüe por três tarefas básicas (SUH; RADCLIFFE, 1978): projeto de posicionadores ou geradores de movimento, de geradores de trajetória e de geradores de função.

A síntese dinâmica (LIMA, 1985) (SALEM; MANOOCHEHRI, 1990) trata da satisfação de especificações do tipo: minimizar o torque do motor, diminuir a flutuação desse torque, balanceamento do mecanismo, entre outras.

2.7 – Tarefas da síntese cinemática

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

A geração de função trata da determinação de um mecanismo que relacione os movimentos de entrada e saída de suas peças segundo uma determinada função.

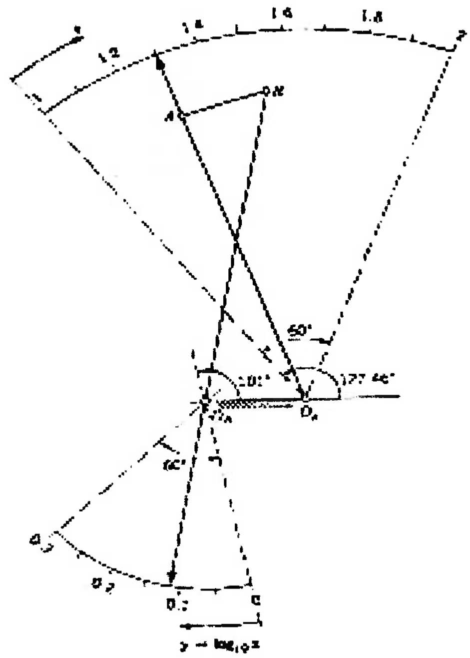


Fig. 2.7.1 – Geração de função (HARTENBERG; DENAVIT, 1964)

No caso da geração de trajetória, toma-se um ponto pertencente a uma peça móvel não diretamente conectada à peça fixa, 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.

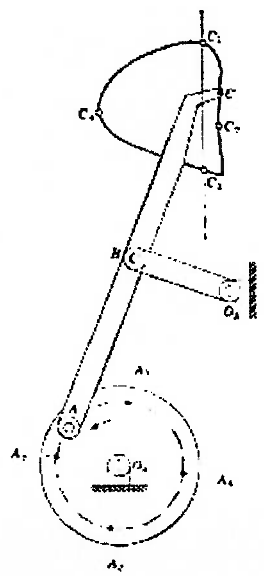


Fig 2.7.2 – Geração de trajetória (HARTENBERG; DENAVIT, 1964).

Quanto à geração de movimento, uma peça deve ser guiada ou posicionada para realizar movimento pré-fixado. Isto pode ser visualizado na fig. 2.7.3.

A solução para os problemas de síntese cinemática pode ser obtida gráfica ou analiticamente. Ambas soluções, gráfica e analítica, se baseiam na teoria de Burmester.

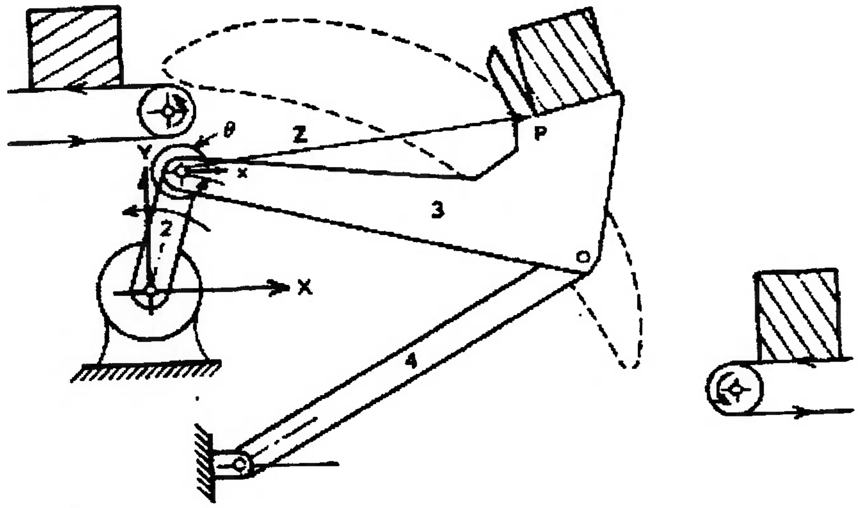


Fig. 2.7.3 – Gerador de movimento (ERDMAN; SANDOR, 1984).

2.8 - Algumas noções da teoria de Burmester

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

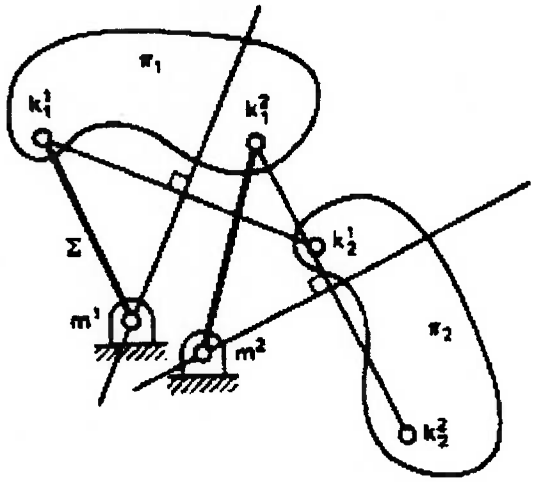


Fig. 2.8.1 – Construção geométrica para a obtenção dos pontos centro para duas posições de precisão (ERDMAN; SANDOR, 1984).

O significado prático dos pontos de Burmester nos mecanismos articulados planos vem do fato de que o movimento relativo de qualquer par de peças 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 a peça que se move ao longo das posições de precisão.

2.9 - Ângulo de transmissão

Este parâmetro surgiu como critério para indicar a qualidade de funcionamento de um dado mecanismo. O termo "funcionar" significa a eficiência com que um dado movimento é transferido para o elemento de saída (movido), que resulta numa operação suave, em que a maior componente da força é utilizada para produzir torque ou força de saída.

TAO (1964) definiu o ângulo de transmissão num quadrilátero articulado, como sendo o menor ângulo entre as direções dos vetores $\vec{V}_{BA}$ e $\vec{V}_B$ tomados no ponto da articulação. Como pode ser visto na fig. 2.9.1, este ângulo coincide com aquele formado pelas linhas de centro do acoplador e da alavanca movida.

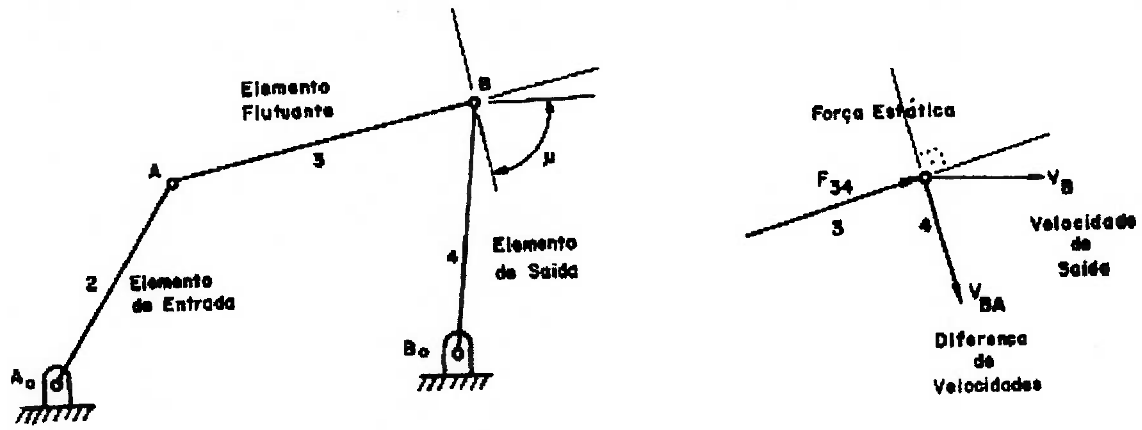


Fig. 2.9.1 – Ângulo de transmissão μ e detalhes dos vetores de força e velocidade no ponto B

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

a)A força de transmissão que passa do acoplador para a alavanca movida é mais efetiva quando o ângulo de transmissão é de 90°. É claro que é desejável que este ângulo tenha o menor desvio possível de 90° durante a operação do mecanismo. Recomenda-se, na prática, que o ângulo de tolerância fique na faixa de ± 45°, ou seja, valor mínimo 45° e máximo 135° (TAO, 1964).

b) Em algumas máquinas de precisão, quando o ângulo de transmissão é pequeno demais, a precisão do movimento de saída torna-se mais sensível à tolerância de fabricação no comprimento das peças e folgas nos pares cinemáticos. Além disso, um ângulo de transmissão baixo, menor que 45° ou maior que 135°, pode causar ruído e solavancos indesejáveis em alta velocidade.

2.10 - Projeto de mecanismo auxiliado por computador

A maioria das pesquisas para o desenvolvimento de programas de computador para o projeto de mecanismos se iniciaram na década de 50 em 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 quadriláteros articulados foi o de FREUDENSTEIN ; SANDOR (1959). A solução que apresentaram empregava notação polar complexa e baseava-se na teoria elaborada por Burmester em 1876.

Na década de 60, o computador se tornou um pouco mais acessível aos pesquisadores nas universidades. Assim, alguns problemas de síntese puderam ser resolvidos mais facilmente 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 ; RADCLIFFE (1967) desenvolveram um trabalho para síntese de mecanismos planos que empregava matrizes de deslocamentos, para equacionamento das rotações e translações de suas peças.

Com o passar do tempo houve um grande crescimento na utilização de computadores para o projeto de mecanismos. Merecem destaque os programas IMP (SHETT;UICKER, 1971), desenvolvido na Universidade de Wisconsin, KINSYN (KAUFMAN, 1974), desenvolvido no MIT e o LINCAGES (ERDMAN;GUSTAFSON, 1977), desenvolvido na 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 mecanismos planos. Tanto o KINSYN como o LINCAGES, realizavam a síntese e análise de mecanismos planos com pares cinemáticos de rotação e translação. Contudo, estes programas ainda só funcionavam em computadores de grande porte.

Uma realização importante na década de 70, é que buscou-se, pela primeira vez, a utilização de saída gráfica no computador para uma melhor e mais fácil visualização dos resultados e uma interação homem-computador mais rápida.

Em 1984, WHALEN e MIDHA apresentaram um programa de computador para o projeto de mecanismos de 4, 6 e 8 barras. Contudo o programa era muito iterativo porque, segundo os autores, se baseava exclusivamente na análise, ou seja, o projetista atribuía valores para o comprimento das barras e o programa calculava os resultados. Após isso, o projetista deveria então verificar se aquele mecanismo atenderia às suas especificações.

Em 1985, MIDHA e ZHAO desenvolveram um método para síntese de mecanismos planos utilizando equações de vetores de posição e servindo-se do método de Newton-Raphson para solução de um sistema de equações não lineares iterativamente.

A disseminação dos microcomputadores nessa época facilitou incrivelmente o desenvolvimento e o acesso a programas de computador para o projeto de mecanismos. Um dos exemplos foi o trabalho de PETERSON et al. (1988), que apresentaram uma versão do LINCAGES para micros da linha PC . Outros exemplos foram o MICRAS (1985) e os programas de NORTON (1985). A filosofia destes programas era tornar a sua utilização a mais simples possível para o projetista.

Na década de 1990, já encontramos os computadores suficientemente capazes e disseminados nas universidades e indústrias, com uma grande diversidade de programas que auxiliam na criação de novos sistemas mecânicos.

Atualmente a Realidade Virtual (RV) está surgindo como uma nova tecnologia baseada em computadores que possibilita ao usuário imergir em um mundo artificial criado pelo computador, e interagir com esse mundo como se fosse real. A RV tem aplicações em projeto de mecanismos, especialmente os tri-dimensionais. A RV poderá auxiliar no projeto desses mecanismos da seguinte forma:

- facilitar a entrada das especificações e dos parâmetros de projeto e de suas restrições;
- verificação da adequação do mecanismo calculado às especificações.

Um grande número de programas foram elaborados para o projeto de mecanismos com a abordagem de análise (tentativa-e-erro). Porém existem, ainda, poucos programas com a abordagem de síntese.

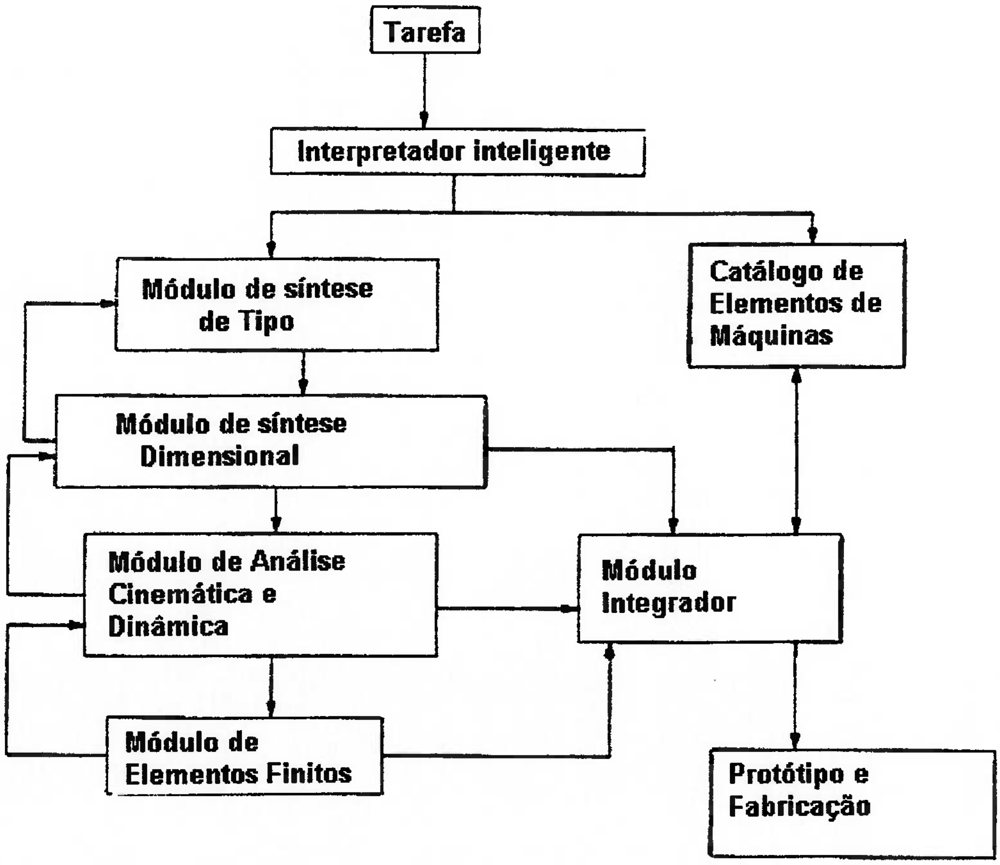


Fig. 2.10.1 – Abordagem modular para síntese de mecanismos (ERDMAN, 1995)

Podemos imaginar um conjunto ideal de programas para o projeto de mecanismos. Todos os módulos devem estar conectados e terem facilidade na troca de informações. A fig. 2.10.1 mostra um conjunto de tal estratégia (ERDMAN, 1995). Uma necessidade específica ou um problema passaria por um interpretador inteligente que determinaria se há alguma "solução armazenada" para o problema em particular ou se será necessário uma nova síntese. No primeiro caso, a solução seria encontrada em um banco de dados (CD ROM, por exemplo) onde um mecanismo ou seu componente esteja disponível e catalogado. Para situações onde não fosse encontrada uma solução nesta fonte, o interpretador inteligente enviaria o problema para o módulos de síntese de tipo e dimensional.

Uma vez encontrada a solução cinemática, os módulos de análise cinemática e dinâmica seriam acionados para verificação do comportamento do mecanismo. Idealmente, todos os módulos estariam integrados de modo a obter um mecanismo adequado que, posteriormente, possa ser fabricado.

2.11 - Otimização no projeto de mecanismos

Em 1966, HAN sugeriu (ROOT; RAGSDELL, 1976) 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 dessa otimização era um mecanismo ótimo do ponto de vista do erro quadrático mínimo.

HIMMELBLAU (1972) apresentou no seu livro uma revisão dos principais métodos de otimização para solução de problemas de programação não linear. Incluiu no seu texto uma comparação entre os métodos mais utilizados e concluiu que o método Gradiente Reduzido Generalizado (GRG, Generalized Reduced Gradient method) era o mais eficiente quando a função objetivo e as restrições do problema são não-lineares.

KRAMER e SANDOR (1975) introduziram um método, chamado de "síntese de precisão seletiva" (conforme fig. 2.11.1), para projeto ótimo de mecanismos planos para geração de trajetória, movimento e função com limites de precisão diferentes nas posições discretas de projeto. Eles não utilizaram a abordagem de posições de precisão, mas de vizinhança de precisão. Com isso puderam especificar um número maior de posições, satisfazendo melhor os requisitos de projeto. Eles empregavam técnicas de programação não linear para solução.

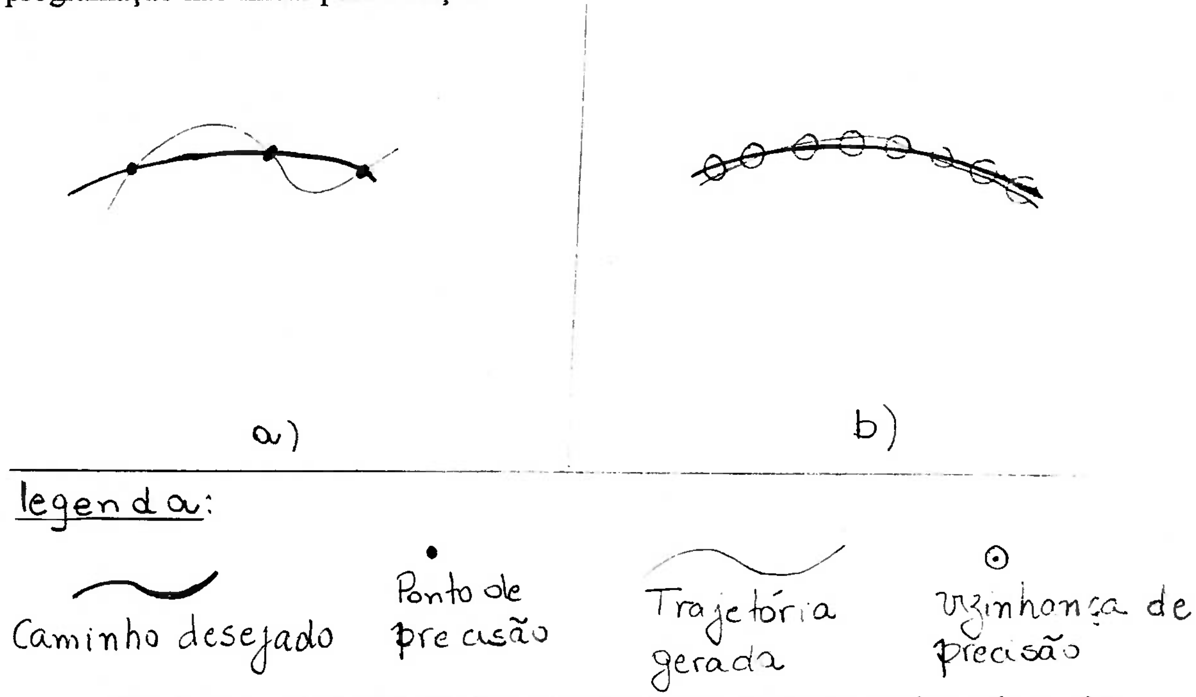


Fig. 2.11.1 – Trajetória de um mecanismo, a) método das posições de precisão; b) método de precisão seletiva.

GABRIELE e RAGSDELL (1977) apresentaram o GRG como uma ferramenta eficiente e confiável para otimização em projeto de engenharia, que é um método numérico para resolucão de problemas de programação não-linear ("nonlinear programming, NLP). Antes desse método, utilizava-se um outro chamado de função de penalização, que transformava um problema NLP num problema LP ("linear programming"). Tal procedimento era aceitável quando a função objetivo e as restrições fossem quase lineares. Caso contrário, o incremento nos valores das variáveis de projeto deveria ser pequeno para a utilização correta do LP. Neste caso, o tempo para a convergência era demasiadamente longo.

GUPTA (1977) apresentou um método analítico para otimização do ângulo de transmissão para projeto de geradores de função em quadrilátero 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°. Com essa imposição aliada às relações das posições de precisão do mecanismo geraram-se curvas que representavam o lugar geométrico do ângulo de transmissão ótimo.

LIMA (1985) publicou um trabalho sobre otimização no balanceamento de mecanismos planos de barras articuladas com um único grau de liberdade, com movimento e geometria conhecidos.

RIGELMAN e KRAMER (1988) 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 da alavanca movida. 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.

KRISHNAMURTY e TURCIC (1992) propuseram que a solução dos problemas de síntese de mecanismos fosse realizada através de técnicas de programação não-linear de alvo ("nonlinear goal programming, NGP). Os problemas de síntese utilizando métodos numéricos normalmente propõem uma função objetivo que satisfaça um determinado critério de projeto, sendo que os demais critérios sejam considerados nas restrições; ou ainda, que uma única função objetivo englobe todos os critérios. No método proposto por esses autores, os valores das variáveis de projeto são determinados, inicialmente, otimizando o objetivo que tenha prioridade "1". Uma vez alcançado um valor mínimo para aquele objetivo, o objetivo com prioridade "2" passará a ser minimizado enquanto que se garante que o valor mínimo do primeiro objetivo seja mantido. O processo é repetido até que os valores mínimos de todas as funções objetivo sejam alcançados.

TODOROV (1997) apresentou um método de síntese do mecanismo de seis barras de Watt (fig. 2.11.2), utilizado para a geração de trajetória visando a manipulação de carga numa região determinada.

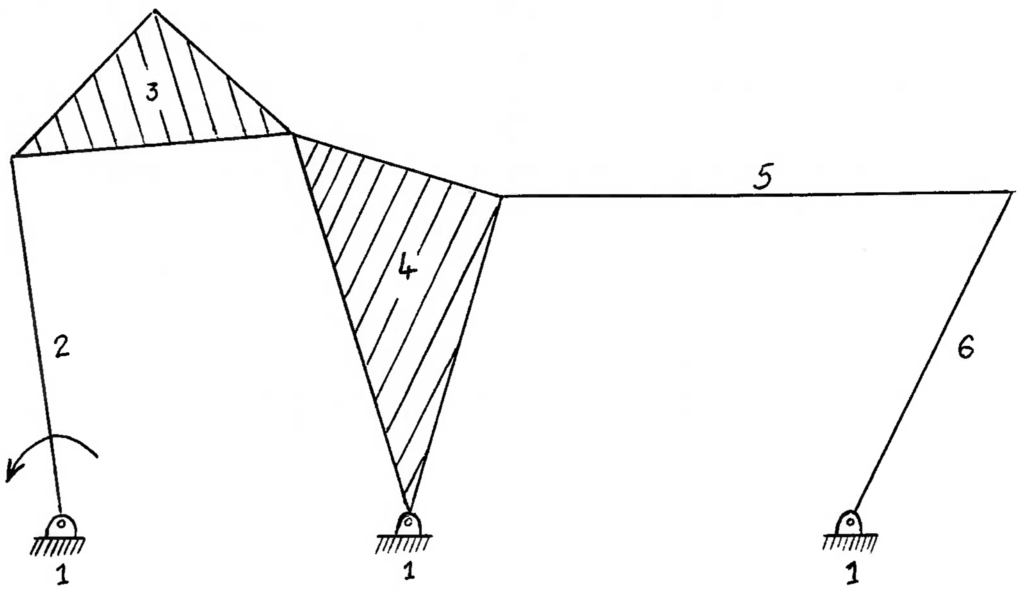


Fig. 2.11.2 – Mecanismo de seis barras de Watt.

Capítulo 3 – Formulação do Problema e seu Modelo Matemático

3.1 – Introdução

A figura 3.1.1 apresenta o mecanismo que esta dissertação se propõe a estudar: um mecanismo articulado plano movido por um acionador prismático.

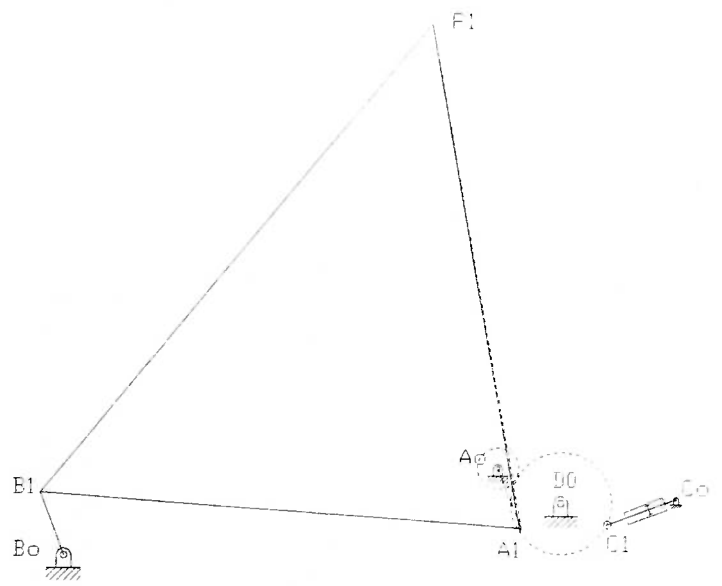


Fig. 3.1.1 - Mecanismo articulado plano movido por um acionador prismático

O mecanismo articulado plano é aquele definido pelas peças $\overline{A_0A_1}$, $\overline{A_1B_1}$, $\overline{B_1B_0}$ e $\overline{B_0A_0}$. Este tipo de mecanismo é conhecido também como quadrilátero articulado.

Fazendo parte da articulação fixa A_0 está uma engrenagem chamada de engrenagem menor ou pinhão que está engrenada a uma engrenagem maior ou coroa fixa em D_0.

A articulação C pertencente à engrenagem maior está ligada à haste de um acionador prismático $\overline{C_1C_0}$ que, construtivamente, pode ser um cilindro hidráulico ou pneumático.

O funcionamento do mecanismo completo decorre naturalmente do entendimento de suas partes. O acionador prismático ao deslocar-se provoca o afastamento ou aproximação de C_1 a C_0 e, a seguir, causa um deslocamento angular da engrenagem maior. O giro desta engrenagem, por sua vez, provoca um giro na engrenagem menor e na própria peça $\overline{A_1A_0}$.

O deslocamento angular da peça $\overline{A_1A_0}$ causa um deslocamento da peça $\overline{A_1B_1}$ (rotação e translação) que, por sua vez, causa uma determinada trajetória do ponto $\underline{P_1}$, pertencente à peça $\overline{A_1B_1}$. A $\overline{A_1B_1}$ peça provoca um deslocamento angular da peça $\overline{B_1B_0}$.

A seguir será descrito o problema que esta dissertação se propõe resolver. Pretende-se, numa etapa inicial, realizar a síntese de um mecanismo articulado plano gerador de movimento, ou seja, que guie a peça $\overline{A_1B_1}$ em três posições específicas.

Desta forma, o objetivo é a determinação dos comprimentos das peças $\overline{A_0A_1}$, $\overline{A_1B_1}$, $\overline{B_1B_0}$ e $\overline{B_0A_0}$. A seção 3.2 trata desta questão da síntese do mecanismo quadrilátero articulado plano.

Em segundo lugar, pretende-se sintetizar o mecanismo acionador. Nesta etapa, que é tratada na seção 3.3, o objetivo é a determinação dos raios das engrenagens maior e menor, e as coordenadas de D_0, C_1 e C_0.

Em terceiro lugar, busca-se introduzir dois critérios de otimização às duas etapas anteriores de síntese. Os dois critérios envolvem a otimização das variáveis ângulo de transmissão, μ e η, dos mecanismos posicionador e acionador, respectivamente. Esta etapa é descrita na seções 3.4 e 3.5.

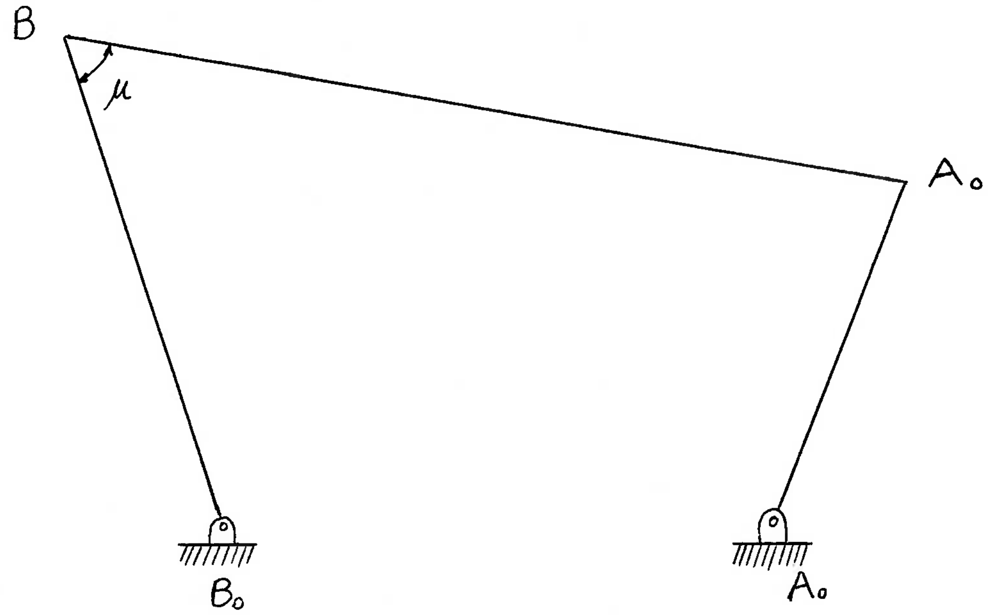


Fig. 3.1.2 – Ângulo de transmissão μ do mecanismo posicionador.

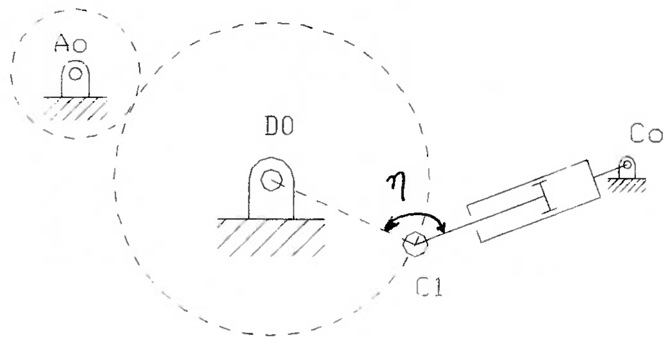


Fig. 3.1.3 – Ângulo de transmissão η do mecanismo acionador

3.2 – Síntese do mecanismo posicionador

Nesta seção serão apresentadas as hipóteses básicas para a solução do problema da síntese, a equação da díada, a síntese para a tarefa de geração de movimento. Por fim, serão desenvolvidos os procedimentos de solução para geração de movimento para 3 posições de precisão com especificação das articulações fixas.

3.2.1 - Modelagem das barras articuladas

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

3.2.2 - Equação da díada (ERDMAN; SANDOR, 1984)

A grande maioria dos mecanismos articulados planos pode ser imaginada como combinações de pares de peças ou barras chamadas díadas. Por exemplo, o quadrilátero articulado da figura 3.4 pode ser considerado como formado por duas díadas: uma do lado esquerdo e outra do lado direito. A díada do lado esquerdo contém as peças $\overline{A_1A_0}$ e $\overline{PA_1}$ e a díada do lado direito, as peças $\overline{B_1B_0}$ e $\overline{PB_1}$.

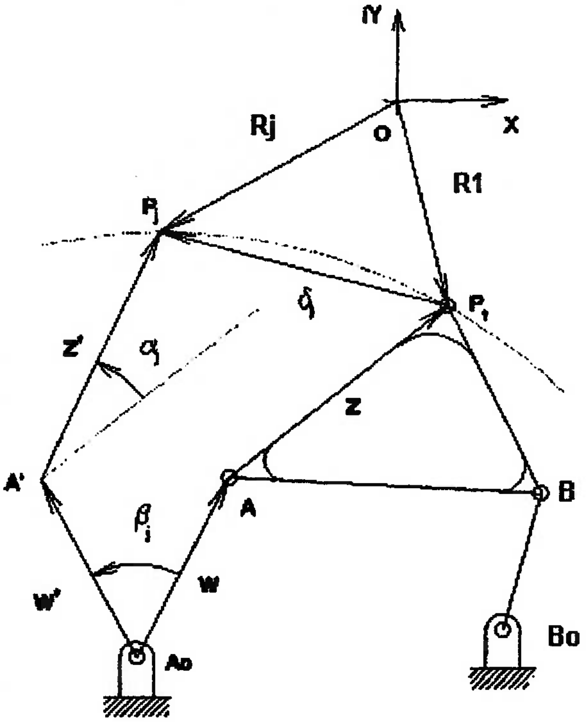


Fig 3.2.2.1 - Deslocamento do ponto P de P1 para Pj

Admite-se que um ponto P na peça acopladora $\overline{AB}$ seja deslocado de sua posição inicial P_1 para uma posição P_J.

Sendo O X iY o sistema de referencia adotado, $\mathbf{R}_1$ o vetor-posição do ponto P_1, $\mathbf{R}_j$ o vetor posição do ponto P_J, a seguinte soma vetorial pode ser realizada:

$$(A - A_0) + (P_1 - A) - \mathbf{R}_1 + \mathbf{Rj} - (P_j - A') - (A' - A_0) = 0$$

sendo $(A - A_0) = \mathbf{w}$, $(P_1 - A) = \mathbf{z}$, $(P_j - A') = \mathbf{z}'$ e $(A' - A_0) = \mathbf{w}'$, então:

$$\mathbf{w} + \mathbf{z} - \mathbf{R}_1 + \mathbf{R}_j - \mathbf{z}' - \mathbf{w}' = 0 \qquad (3.2.2.1)$$

Sendo β_j a rotação de **w** e α_j a rotação de **z** temos:

$$\mathbf{w}' = \mathbf{w} . \exp(i\beta_j) \qquad \mathbf{z}' = \mathbf{z} . \exp(i\alpha_j)$$

substituindo $\mathbf{w}'$ e $\mathbf{z}'$ em (3.1) vem:

$$\mathbf{w} + \mathbf{z} - \mathbf{R}_1 + \mathbf{R}_j - \mathbf{z} \cdot \exp(i\alpha_j) - \mathbf{w} \cdot \exp(i\beta_j) = 0$$

$$\mathbf{w} \cdot (\exp(i\beta_j) - 1) + \mathbf{z} \cdot (\exp(i\alpha_j) - 1) = \mathbf{R}_j - \mathbf{R}_1$$

como $\mathbf{R}_j - \mathbf{R}_1 = \delta_j$ então:

$$\mathbf{w} \cdot (\exp(i\beta_j) - 1) + \mathbf{z} \cdot (\exp(i\alpha_j) - 1) = \delta_j \qquad (3.2.2.2)$$

A equação (3.2.2.2) é conhecida como equação da Díada.

3.2.3 - Os sistemas de equações para solução da equação da díada na síntese para algumas posições de precisão.

Levando em conta a equação (3.2.2.2) é possível construir-se a tabela 3.2.3.1 para diferentes posições do ponto P.

Tabela 3.2.3.1 - Número máximo de posições de precisão para a díada, tendo como valores desconhecidos dos vetores **w** e **z** quando δ_j e α_j são especificados previamente na equação (3.2.2.2).

número de posições (n) j = 2, 3, ..., n	número de equações escalares	número de escalares desconhecidos	número de escalares escolhido livremente	número de soluções
2	2	5 (**w**, **z**, e β_2)	3	∞^3
3	4	6 (os acima e β_3)	2	∞^2
4	6	7 (os acima e β_4)	1	∞
5	8	8 (os acima e β_5)	0	finito

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

3.2.3.1 - Duas posições

Tomando - se três escalares, escolhidos arbitrariamente: $\underline{\mathbf{z}}$ e β_2 (por exemplo), calcula-se o vetor **w** :

$$\mathbf{w} = \frac{\delta_2 - \mathbf{z}(\exp(i\alpha_2) - 1)}{\exp(i\beta_2) - 1}$$

3.2.3.2 - 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).

$$\mathbf{w}(\exp(i\beta_2) - 1) + \mathbf{z}(\exp(i\alpha_2) - 1) = \delta_2$$

$$\mathbf{w}(\exp(i\beta_3) - 1) + \mathbf{z}(\exp(i\alpha_3) - 1) = \delta_3$$

3.2.4 - Geração de movimento

O quadrilátero articulado da figura 3.2.4.1 deve ser sintetizado para geração de movimento. Como pode ser observado nesta figura, há duas díadas independentes no quadrilátero articulado, uma chamada díada do lado esquerdo e outra do lado direito.

Cada díada liga uma articulação fixa a um ponto do acoplador através de uma articulação móvel. A equação da díada do lado esquerdo, 1ª díada, já foi deduzida, e de acordo com a figura 3.5 assume a seguinte forma:

$$\mathbf{z}_2 (\exp(i\beta_j) - 1) + \mathbf{z}_5 (\exp(i\,\alpha_j) - 1) = \delta_j \quad , \; j = 2, \ldots, 5$$

onde δ_j e α_j são variáveis especificadas. A equação da díada do lado direito, 2ª díada, pode ser escrita na forma:

$$\mathbf{z}_4 (\exp(i\,\beta_j) - 1) + \mathbf{z}_6 (\exp(i\,\alpha_j) - 1) = \delta_j$$

onde δ_j e α_j são especificados. Utilizando - se técnicas matemáticas para resolver este sistema de equações obtém - se $\mathbf{z}_2$, $\mathbf{z}_4$, $\mathbf{z}_5$, e $\mathbf{z}_6$. Os outros dois vetores do quadrilátero articulado são simplesmente:

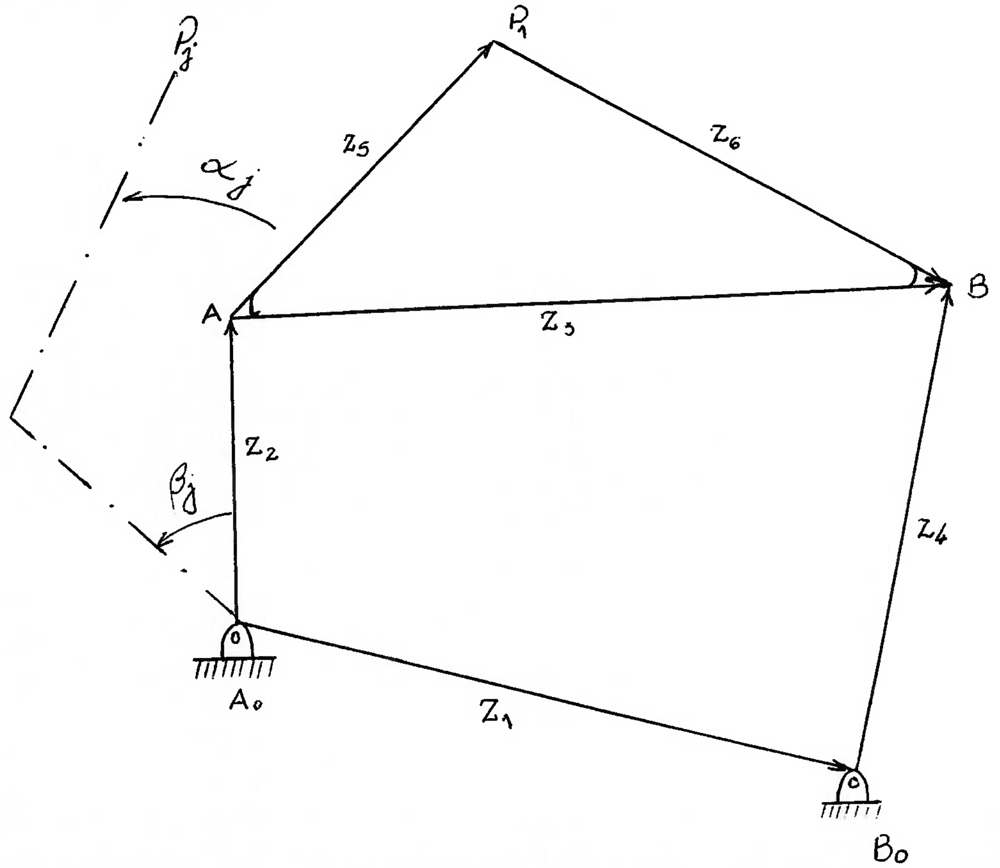


Fig.3.2.4.1 - Quadrilátero articulado.

$$\mathbf{z}_3 = \mathbf{z}_5 - \mathbf{z}_6$$

$$\mathbf{z}_1 = \mathbf{z}_2 + \mathbf{z}_3 - \mathbf{z}_4$$

3.2.5 - Procedimento de solução para três posições de precisão com especificação da articulação fixa

Para solução deste caso considerar-se-á conhecido o vetor (P_1 - A_0), conforme a figura 3.2.5.1, que será chamado de vetor $\mathbf{R}_1$. Assim, tem-se o seguinte sistema de equações cujas incógnitas seriam **w**, **z**, β_2 e β_3:

$$\mathbf{w} + \mathbf{z} = \mathbf{R}_1$$

$$\mathbf{w}.\exp(i\beta_2) + \mathbf{z}.\exp(i\alpha_2) = \mathbf{R}_2$$

$$\mathbf{w}.\exp(i\beta_3) + \mathbf{z}.\exp(i\alpha_3) = \mathbf{R}_3$$

$$\mathbf{z} = \mathbf{R}_1 - \mathbf{w}$$

$$\mathbf{w}\exp(i\beta_2) + (\mathbf{R}_1 - \mathbf{w})\exp(i\alpha_2) = \mathbf{R}_2$$

$$\mathbf{w}(\exp(i\beta_2) - \exp(i\alpha_2)) = \mathbf{R}_2 - \mathbf{R}_1\exp(i\alpha_2)$$

$$\mathbf{w} = \frac{\mathbf{R}_2 - \mathbf{R}_1\exp(i\alpha_2)}{\exp(i\beta_2) - \exp(i\alpha_2)} \qquad (3.2.5.1)$$

$$\mathbf{w}(\exp(i\beta_3) + (\mathbf{R}_1 - \mathbf{w})\exp(i\alpha_3) = \mathbf{R}_3$$

$$\mathbf{w}(\exp(i\beta_3) - \exp(i\alpha_3)) = \mathbf{R}_3 - \mathbf{R}_1\exp(i\alpha_3)$$

$$\mathbf{w} = \frac{\mathbf{R}_3 - \mathbf{R}_1\exp(i\alpha_3)}{\exp(i\beta_3) - \exp(i\alpha_3)} \qquad (3.2.5.2)$$

igualando as equações 3.2.5.1 e 3.2.5.2 vem:

$$\frac{\mathbf{R}_2 - \mathbf{R}_1 \exp(i\alpha_2)}{\exp(i\beta_2) - \exp(i\alpha_2)} = \frac{\mathbf{R}_3 - \mathbf{R}_1 \exp(i\alpha_3)}{\exp(i\beta_3) - \exp(i\alpha_3)}$$

$$(\mathbf{R}_2 - \mathbf{R}_1 \exp(i\alpha_2))\,(\exp(i\beta_3) - \exp(i\alpha_3)) = (\mathbf{R}_3 - \mathbf{R}_1 \exp(i\alpha_3))\,(\exp(i\beta_2) - \exp(i\alpha_2))$$

$$(\mathbf{R}_2 - \mathbf{R}_1 \exp(i\alpha_2)) \exp(i\beta_3) - (\mathbf{R}_3 - \mathbf{R}_1 \exp(i\alpha_3)) \exp(i\beta_2) =$$
$$= (\mathbf{R}_2 - \mathbf{R}_1 \exp(i\alpha_2))\exp(i\alpha_3) - (\mathbf{R}_3 - \mathbf{R}_1 \exp(i\alpha_3)) \exp(i\alpha_2)$$

Chamando de **x**, **y**, **k** os vetores :

$$\mathbf{x} = -(\mathbf{R}_3 - \mathbf{R}_1 \exp(i\alpha_3))$$

$$\mathbf{y} = \mathbf{R}_2 - \mathbf{R}_1 \exp(i\alpha_2)$$

$$\mathbf{k} = \mathbf{R}_2 \exp(i\alpha_3) - \mathbf{R}_3 \exp(i\alpha_2)$$

Portanto,

$$\mathbf{x} \exp(i\beta_2) + \mathbf{y} \exp(i\beta_3) = \mathbf{k}$$

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

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

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

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

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

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

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

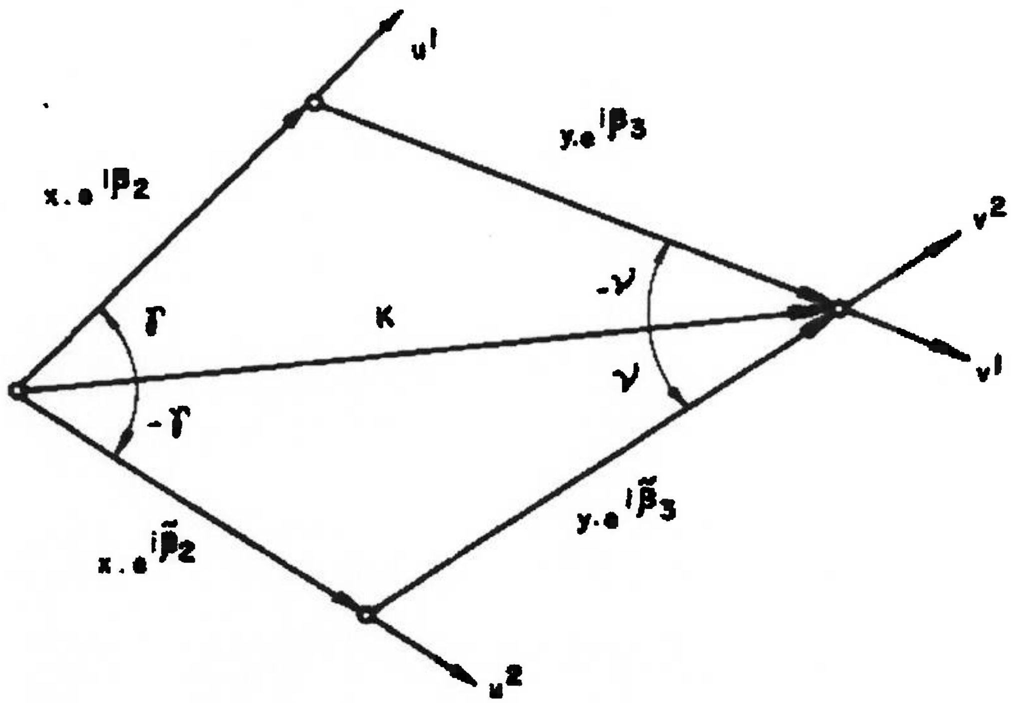


Fig 3.2.5.1 - Triângulo de solução do sistema não linear

Analogamente,

$\mathbf{u}^2 = \exp(i.\arg(\mathbf{k}) - i\gamma) \Rightarrow \tilde{\beta}_2 = \arg(\mathbf{k}) - \gamma - \arg(\mathbf{x})$

$\mathbf{v}^2 = \exp(i.\arg(\mathbf{k}) - i\nu) \Rightarrow \tilde{\beta}_3 = \arg(\mathbf{k}) + \nu - \arg(\mathbf{y})$

Logo, para cada díada, são possíveis duas soluções: 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, as equações da sub-seção 3.2.3.2 se tornaram lineares e **w** e **z** ficam determinados.

Calculando-se **w** e **z** para a primeira e segunda díadas, ficam determinadas 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.

3.3 – Síntese do mecanismo acionador

Na figura 3.3.1 pode-se observar o esquema cinemático do mecanismo acionador.

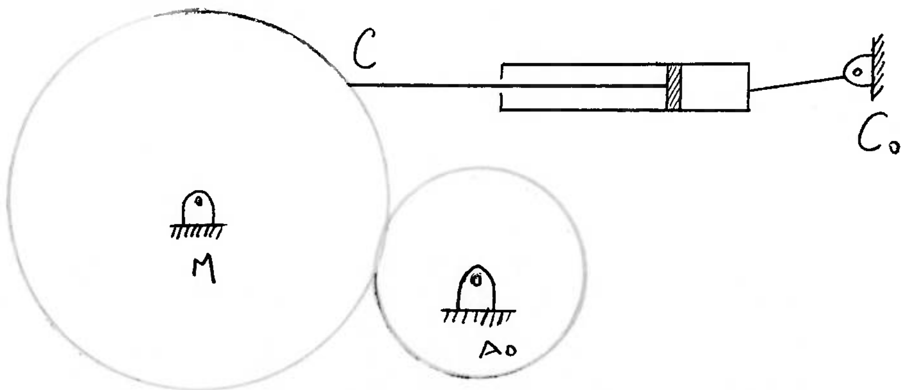


Fig. 3.3.1 – Mecanismo acionador

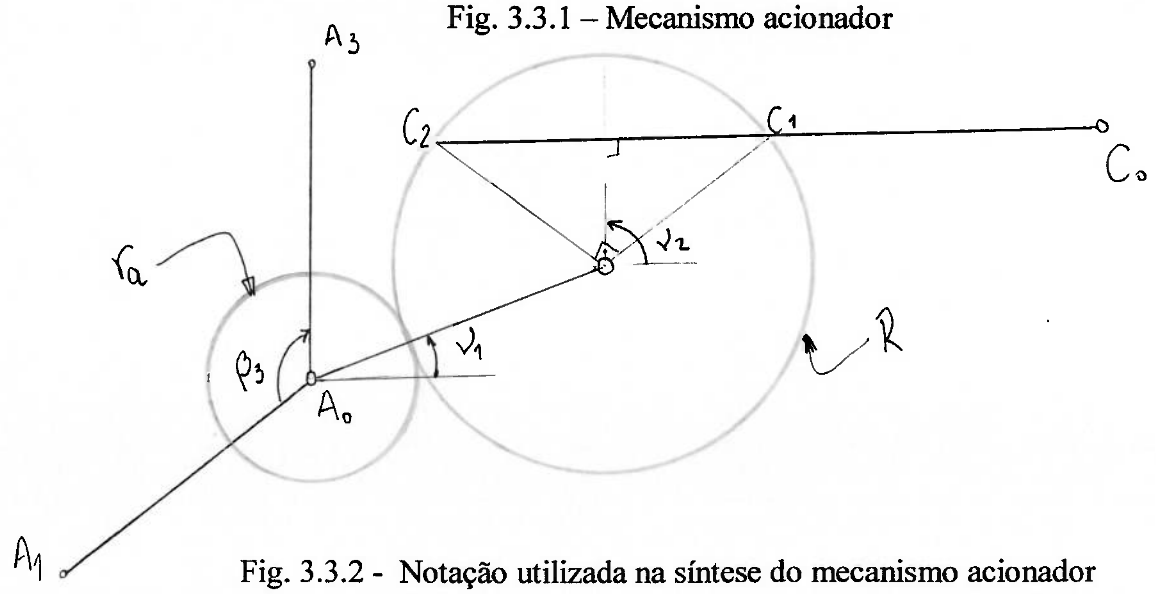


Fig. 3.3.2 - Notação utilizada na síntese do mecanismo acionador

Para realizar a síntese do mecanismo acionador, são necessários os seguintes dados:

- os valores das coordenadas da articulação fixa C_0, que são fornecidos pelo projetista;
- os valores de ra e R, que são calculados da seguinte maneira:

$$ra = \frac{\overline{A_0A_1}}{3}$$

$$\frac{R}{\beta_3} = \frac{ra}{\frac{\pi}{2}}$$

ou seja, o raio da engrenagem maior R está para o da engrenagem menor ra assim como o ângulo β_3 está para $\pi/2$ (90°),

$$R = \frac{2\beta_3 \, ra}{\pi}$$

sendo β_3 em radianos.

Para $\beta_2 < 0$, que é o caso indicado pela figura 3.3.2, obtem-se a seguinte equação vetorial:

$$C_0 = A_0 + (R + ra)\exp(i\nu_1) + R\frac{\sqrt{2}}{2}\exp(i\nu_2) + 3.R\frac{\sqrt{2}}{2}\exp\left(i\left(\nu_2 - \frac{\pi}{2}\right)\right)$$

$$C_{0x} = A_{0x} + (R + ra)\cos(\nu_1) + R\frac{\sqrt{2}}{2}\cos(\nu_2) + 3.R\frac{\sqrt{2}}{2}\cos\left(\nu_2 - \frac{\pi}{2}\right)$$

$$C_{0y} = A_{0y} + (R + ra)\,\mathrm{sen}(\nu_1) + R\frac{\sqrt{2}}{2}\,\mathrm{sen}(\nu_2) + 3.R\frac{\sqrt{2}}{2}\,\mathrm{sen}\left(\nu_2 - \frac{\pi}{2}\right)$$

Da trigonometria podemos escrever que:

$$\cos \nu_2 = -\operatorname{sen}\left(\nu_2 - \frac{\pi}{2}\right)$$

e

$$\operatorname{sen} \nu_2 = \cos\left(\nu_2 - \frac{\pi}{2}\right)$$

Fazendo as substituições encontramos:

$$C_{0x} = A_{0x} + (R + ra)\cos(\nu_1) + R\frac{\sqrt{2}}{2}\cos(\nu_2) + 3.R\frac{\sqrt{2}}{2}\operatorname{sen}\nu_2$$

$$C_{0x} = A_{0x} + (R + ra)\cos \nu_1 + R\frac{\sqrt{2}}{2}(\cos \nu_2 + 3.\operatorname{sen} \nu_2)$$

$$C_{0y} = A_{0y} + (R + ra)\operatorname{sen} \nu_1 + R\frac{\sqrt{2}}{2}\operatorname{sen} \nu_2 - 3.R\frac{\sqrt{2}}{2}\cos \nu_2$$

$$C_{0y} = A_{0y} + (R + ra)\operatorname{sen} \nu_1 + R\frac{\sqrt{2}}{2}(\operatorname{sen} \nu_2 - 3.\cos \nu_2)$$

Nestas equações escalares, as incógnitas são ν_1 e ν_2 sendo que C_{0x} e C_{0y} são especificados. A resolução deste sistema não-linear de duas equações e duas incógnitas pode ser obtida utilizando-se o método numérico de Newton-Raphson. Desta forma criam-se duas funções $F_1(\nu_1, \nu_2)$ e $F_2(\nu_1, \nu_2)$, que devem ser anuladas.

$$F_1(\nu_1, \nu_2) = A_{ox} - C_{ox} + R\frac{\sqrt{2}}{2}(3\operatorname{sen}(\nu_2) + \cos(\nu_2)) + (R + ra)\cos \nu_1$$

$$F_2(\nu_1, \nu_2) = A_{oy} - C_{oy} + R\frac{\sqrt{2}}{2}(-3\cos(\nu_2) + \operatorname{sen}(\nu_2)) + (R + ra)\operatorname{sen} \nu_1$$

$$\begin{bmatrix} \dfrac{\partial \mathbf{F}_1^{(k)}(\nu_1,\nu_2)}{\partial \nu_1} & \dfrac{\partial \mathbf{F}_1^{(k)}(\nu_1,\nu_2)}{\partial \nu_2} \\ \dfrac{\partial \mathbf{F}_2^{(k)}(\nu_1,\nu_2)}{\partial \nu_1} & \dfrac{\partial \mathbf{F}_2^{(k)}(\nu_1,\nu_2)}{\partial \nu_2} \end{bmatrix} \cdot \begin{bmatrix} \Delta_1^{(k)} \\ \Delta_2^{(k)} \end{bmatrix} = \begin{bmatrix} -\mathbf{F}_1^{(k)}(\nu_1,\nu_2) \\ -\mathbf{F}_2^{(k)}(\nu_1,\nu_2) \end{bmatrix}$$

onde k é a k-ésima iteração.

Calculando-se os jacobianos de uma das matrizes acima, obtem-se:

$$\text{jaco}\ (1,1) = \frac{\partial F_1(\nu_1,\nu_2)}{\partial \nu_1} = -(R + ra)\ \text{sen}\ \nu_1$$

$$\text{jaco}\ (1,2) = \frac{\partial F_1(\nu_1,\nu_2)}{\partial \nu_2} = R\ \frac{\sqrt{2}}{2}(3\cos\ \nu_2 - \text{sen}\ \nu_2)$$

$$\text{jaco}\ (2,1) = \frac{\partial F_2(\nu_1,\nu_2)}{\partial \nu_1} = (R + ra)\cos\ \nu_1$$

$$\text{jaco}\ (2,2) = \frac{\partial F_2(\nu_1,\nu_2)}{\partial \nu_2} = R\ \frac{\sqrt{2}}{2}(3\ \text{sen}\ \nu_2 + \cos\ \nu_2)$$

$$\begin{bmatrix} \text{jaco}(1,1) & \text{jaco}(1,2) \\ \text{jaco}(2,1) & \text{jaco}(2,2) \end{bmatrix} \cdot \begin{bmatrix} \Delta_1 \\ \Delta_2 \end{bmatrix} = \begin{bmatrix} -F_1 \\ -F_2 \end{bmatrix}$$

$$\Delta_1 = \frac{\begin{vmatrix} -F_1 & jaco(1,2) \\ -F_2 & jaco(2,2) \end{vmatrix}}{\begin{vmatrix} jaco(1,1) & jaco(1,2) \\ jaco(2,1) & jaco(2,2) \end{vmatrix}} = \frac{-F_1 jaco(2,2) + F_2 jaco(1,2)}{jaco(1,1) jaco(2,2) - jaco(2,1) jaco(1,2)}$$

$$\Delta_2 = \frac{\begin{vmatrix} jaco(1,1) & -F_1 \\ jaco(2,1) & -F_2 \end{vmatrix}}{\begin{vmatrix} jaco(1,1) & jaco(1,2) \\ jaco(2,1) & jaco(2,2) \end{vmatrix}} = \frac{-F_2 jaco(1,1) + F_1 jaco(2,1)}{jaco(1,1) jaco(2,2) - jaco(2,1) jaco(1,2)}$$

$$\nu_1^{(k+1)} = \nu_1^k + \Delta_1^k$$

$$\nu_2^{(k+1)} = \nu_2^k + \Delta_2^k$$

$$\Delta = \left[\left(\frac{\Delta_1}{\nu_1} \right)^2 + \left(\frac{\Delta_2}{\nu_2} \right)^2 \right]^{\frac{1}{2}}$$

O critério de convergência de convergência é dado a seguir:

$$\Delta \leq 1 \,.\, 10^{-7}$$

Analogamente, para $\beta_2 > 0$, as funções $F_1(\nu_1, \nu_2)$ e $F_2(\nu_1, \nu_2)$ são as seguintes:

$$F_1(\nu_1,\nu_2) = A_{ox} - C_{ox} + R\frac{\sqrt{2}}{2}(-3\,\mathrm{sen}(\nu_2) + \cos(\nu_2)) + (R + ra)\cos\nu_1$$

$$F_2(\nu_1,\nu_2) = A_{oy} - C_{oy} + R\frac{\sqrt{2}}{2}(\mathrm{sen}(\nu_2) + 3\cos(\nu_2)) + (R + ra)\,\mathrm{sen}\,\nu_1$$

Os jacobianos serão:

$$\mathrm{jaco}\ (1,1) = \frac{\partial F_1(\nu_1,\nu_2)}{\partial \nu_1} = -(R + ra)\,\mathrm{sen}\,\nu_1$$

$$\mathrm{jaco}\ (1,2) = \frac{\partial F_1(\nu_1,\nu_2)}{\partial \nu_2} = -R\frac{\sqrt{2}}{2}(\mathrm{sen}(\nu_2) + 3\cos(\nu_2))$$

$$\mathrm{jaco}\ (2,1) = \frac{\partial F_2(\nu_1,\nu_2)}{\partial \nu_1} = (R + ra)\cos\nu_1$$

$$\mathrm{jaco}\ (2,2) = \frac{\partial F_2(\nu_1,\nu_2)}{\partial \nu_2} = R\frac{\sqrt{2}}{2}(\cos(\nu_2) - 3\sin(\nu_2))$$

$$\Delta_1 = \frac{-(\mathbf{F}_1(\nu_1,\nu_2).\,\mathrm{Jaco}\ (2,2) - \mathbf{F}_2(\nu_1,\nu_2).\,\mathrm{Jaco}\ (1,2))}{\mathrm{Jaco}(1,1).\,\mathrm{Jaco}\ (2,2) - \mathrm{Jaco}(2,1).\,\mathrm{Jaco}\ (1,2)}$$

$$\Delta_2 = \frac{-(\mathbf{F}_2(\nu_1,\nu_2).\,\mathrm{Jaco}\ (1,1) - \mathbf{F}_1(\nu_1,\nu_2).\,\mathrm{Jaco}\ (2,1))}{\mathrm{Jaco}\ (1,1).\,\mathrm{Jaco}\ (2,2) - \mathrm{Jaco}\ (2,1).\,\mathrm{Jaco}\ (1,2)}$$

$$\Delta = \left[\left(\frac{\Delta_1}{\nu_1} \right)^2 + \left(\frac{\Delta_2}{\nu_2} \right)^2 \right]^{\frac{1}{2}}$$

O critério de convergência é dado a seguir:

$$\Delta \leq 1 \,.\, 10^{-7}$$

3.4 – Ângulos de transmissão dos mecanismos posicionador e acionador

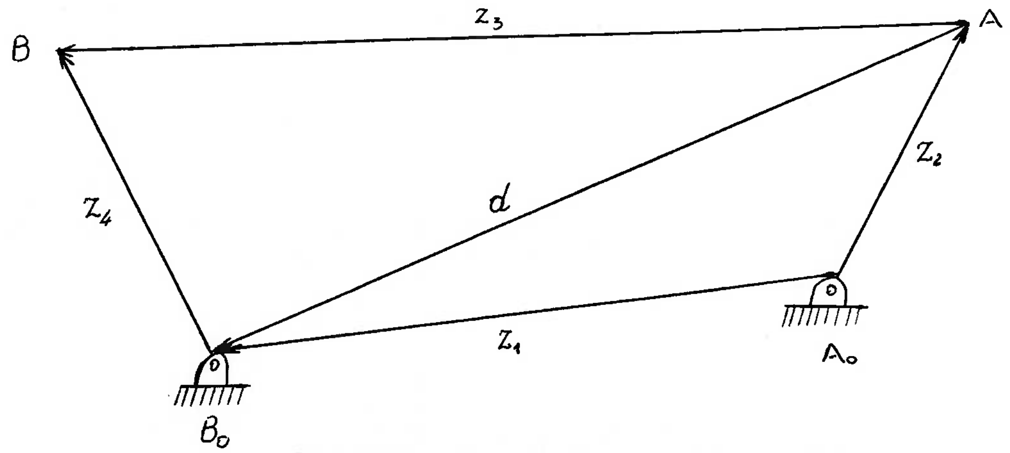


Fig. 3.4.1 – Notação do mecanismo posicionador

Da figura acima pode-se deduzir a seguinte equação vetorial:

$$\mathbf{z}_2 + \mathbf{d} = \mathbf{z}_1 \Rightarrow \mathbf{d} = \mathbf{z}_1 - \mathbf{z}_2$$

$$\mathbf{d} = z_1(\cos\theta_1 + i\,\text{sen}\,\theta_1) - z_2(\cos\theta_2 + i\,\text{sen}\,\theta_2)$$

Reagrupando em reais e imaginários temos:

$$\mathbf{d} = (z_1 \cos\theta_1 - z_2 \cos\theta_2) + i(z_1 \,\text{sen}\,\theta_1 - z_2 \,\text{sen}\,\theta_2)$$

Para determinação do ângulo μ, aplica-se a lei dos cossenos ao triângulo AB_0B:

$$d^2 = z_4^2 + z_3^2 - 2z_3z_4 \cos\mu$$

$$\cos \mu = \frac{z_3^2 + z_4^2 - d^2}{2 z_3 z_4}$$

$$\mu = a\cos\left(\frac{z_3^2 + z_4^2 - d^2}{2 z_3 z_4}\right)$$

No caso do mecanismo acionador, o ângulo η de transmissão será o ângulo D_0CC_0.

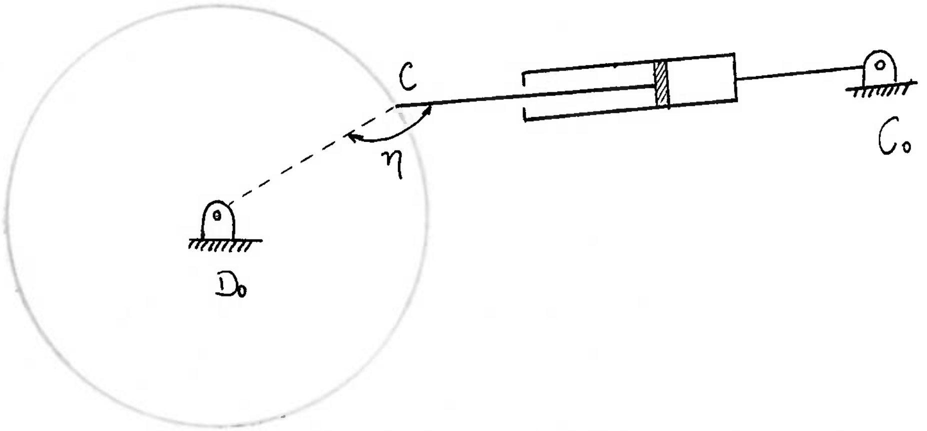


Fig. 3.4.2 – Ângulo de transmissão do mecanismo acionador

Como para determinação do ponto C_0 (vide seção 3.3) escolheu-se uma geometria de mecanismo tal que $\eta_1 = 135°$ e $\eta_3 = 45°$, esta variável, ângulo de transmissão, encontra-se dentro do intervalo recomendado e, portanto, pode-se considerá-la otimizada.

3.5 – Otimização do ângulo de transmissão μ do mecanismo posicionador

Para o mecanismo posicionador, parte-se de um conjunto de variáveis conhecidas como variáveis de projeto. Estas variáveis são componentes de um vetor de projeto **x** Nos problemas dessa dissertação as variáveis de projeto serão as seguintes:

$$\mathbf{x} = [x_1, x_2, x_3, \ldots, x_{10}]^T$$

onde:

$x_1 = A_{0x}$ = Coordenada x da articulação fixa A_0

$x_2 = A_{0y}$ = Coordenada y da articulação fixa A_0

$x_3 = B_{0x}$ = Coordenada x da articulação fixa B_0

$x_4 = B_{0y}$ = Coordenada y da articulação fixa B_0

$x_5 = P_{1x}$ = Coordenada x do ponto P na 1ª posição de precisão

$x_6 = P_{1y}$ = Coordenada y do ponto P na 1ª posição de precisão

$x_7 = P_{2x}$ = Coordenada x do ponto P na 2ª posição de precisão

$x_8 = P_{2y}$ = Coordenada y do ponto P na 2ª posição de precisão

$x_9 = P_{3x}$ = Coordenada x do ponto P na 3ª posição de precisão

$x_{10} = P_{3y}$ = Coordenada y do ponto P na 3ª posição de precisão

As variáveis de projeto são definidas num dado domínio de projeto, representado por seus valores máximo e mínimo. A função f(**x**) será chamada de função-objetivo:

$$f(\mathbf{x}) = \sum_{i=1}^{n} (\mu_i - \frac{\pi}{2})^2$$

e deverá ser minimizada. O valor de "n" indica o número de posições representativas no curso do movimento do mecanismo de "1"para "3"; este intervalo foi subdividido em 20 partes iguais. O método de otimização numérica será o GRG, descrito no capítulo 4.

A filosofia da otimização prevê a alteração dos valores das variáveis de projeto de modo a minimizar a função-objetivo.

Capítulo 4 - Método do Gradiente Reduzido Generalizado

4.1 - Introdução

Os métodos de otimização podem ser classificados sob vários critérios (HIMMELBLAU, 1972). O primeiro critério define se o método resolve problemas de otimização com ou sem restrições (vínculos). Um problema de otimização envolve a minimização de uma função, que é chamada de função-objetivo. A função-objetivo depende de um certo número de variáveis, denominadas variáveis de projeto, e é válida num determinado domínio definidos pelos limites inferior e superior de cada variável, e pelas suas restrições. Estas restrições nada mais são do que equações se/ou inequações envolvendo as variáveis de projeto.

O segundo critério estabelece se o problema a ser resolvido pelo método é de programação linear ou não-linear. Ele se baseia na natureza da função-obetivo e das restrições (se existirem), ou seja, se são lineares ou não-lineares. Assim, se tanto a função objetivo quanto as restrições forem lineares, a programação é dita linear. Por outro lado, a programação não-linear permite que tanto a função-objetivo como as restrições sejam de natureza linear ou não-linear. Portanto, a programação não-linear engloba a linear.

O terceiro critério de classificação define se o método é baseado ou não no cálculo de derivadas. O método do Gradiente Reduzido Generalizado, por exemplo, é baseado no cálculo de derivadas. Os métodos não baseados no cálculo de derivadas são chamados métodos de Busca.

4.2 - Características do método do gradiente reduzido e generalizado

O método do gradiente reduzido generalizado (GRG) é uma técnica numérica de programação não-linear. Este método define a sua direção de busca a partir do cálculo do vetor gradiente reduzido da função-objetivo, ou seja, um vetor que possui um número menor de componentes que o próprio vetor gradiente da função-objetivo. De qualquer forma, como o método baseia-se no cálculo de um gradiente, esta técnica envolve o cálculo de derivadas de 1a. Ordem. A seguir, será apresentada a teoria básica do método do Gradiente Reduzido Generalizado (GABRIELE; RAGSDELL, 1977) e (1989).

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

minimizar a função

$$f(\mathbf{x}), \quad \mathbf{x}=[x_1, x_2, ..., x_n]^T$$

sujeito às restrições

$$g_j(\mathbf{x}) \geq 0 \qquad j = 1, 2, 3, ..., J$$
$$h_k(\mathbf{x}) = 0 \qquad k = 1, 2, 3, ..., K$$
$$\text{onde } x^{(l)} \leq x \leq x^{(u)}$$

Com relação à notação, pode-se dizer que x representa o vetor das variáveis de projeto e $f(\mathbf{x})$ é uma função escalar das variáveis de projeto e representa um critério de projeto ou objetivo. As funções $g(x)$ e $h(x)$ representam, respectivamente, as restrições de desigualdade e de igualdade com relação às variáveis de projeto. As restrições de desigualdade delimitam as possíveis regiões dentro do espaço de projeto, enquanto as restrições de igualdade definem combinações específicas das variáveis de projeto. Os vetores $x^{(l)}$ e $x^{(u)}$ contém, respectivamente, os limites inferiores e superiores de cada componente do vetor **x**.

A seguir será apresentada uma outra forma de definição do problema geral da programação não-linear:

$$\text{minimizar} \qquad f(\mathbf{x}), \qquad \mathbf{x}=[x_1, x_2, ..., x_N]^T$$

sujeita a

$$h_m(\mathbf{x}) = 0 \quad m = 1,2,3,\ldots,M$$
$$\mathbf{x}^{(l)} \leq \mathbf{x} \leq \mathbf{x}^{(u)}$$

As restrições de desigualdade foram incluídas no grupo das restrições de igualdade por meio da introdução de variáveis de folga não-negativas $x_{N+j} \geq 0$ de modo que

$$h_{K+j}(\mathbf{x}) = g_j(\mathbf{x}) - \mathbf{x}_{N+j} \qquad (4.2.1)$$

Estas variáveis de folga são incluídas no conjunto original das variáveis de projeto. O parâmetro M refere-se ao número total de restrições, $M = J + K$.

O método inicia-se com a decomposição do vetor **x** em dois novos vetores: **z** e **y**. As componentes do vetor **z** serão chamadas de variáveis não-básicas ou independentes, e as componentes do vetor **y** serão chamadas de variáveis básicas ou dependentes. As variáveis básicas serão utilizadas para satisfazer as restrições.

$$\mathbf{x} = [\mathbf{z},\mathbf{y}]^T \qquad (4.2.2)$$

$$\mathbf{z} = [\, z_1, z_2, \ldots, z_Q]^T$$

$$\mathbf{y} = [y_1, y_2, \ldots, y_M]^T$$

onde $Q = N - M$. Serão calculados, a seguir, os diferenciais da função $f(\mathbf{x})$ e das restrições de igualdade $h_m(\mathbf{x})$ com relação às variáveis básicas e não-básicas.

$$df = \nabla_z f(\mathbf{x})^T d\mathbf{z} + \nabla_y f(\mathbf{x})^T d\mathbf{y} \qquad (4.2.3)$$

$$d\mathbf{h} = C.d\mathbf{z} + J.d\mathbf{y} = 0 \qquad (4.2.4)$$

onde

$$\nabla_z f(\mathbf{x}) = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{z}_1}, \frac{\partial \mathbf{f}}{\partial \mathbf{z}_2}, \ldots, \frac{\partial \mathbf{f}}{\partial \mathbf{z}_Q} \right]^T \qquad (4.2.5)$$

$$\nabla_y f(\mathbf{x}) = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{y}_1}, \frac{\partial \mathbf{f}}{\partial \mathbf{y}_2}, \ldots, \frac{\partial \mathbf{f}}{\partial \mathbf{y}_M} \right]^T \qquad (4.2.6)$$

$$J = [\nabla_y h_1(\mathbf{x}), \nabla_y h_2(\mathbf{x}), \ldots, \nabla_y h_M(\mathbf{x})]^T; \quad \text{matriz MxM} \qquad (4.2.7)$$

$$C = [\nabla z\, h_1(\mathbf{x}), \nabla z\, h_2(\mathbf{x}), \ldots, \nabla z\, h_M(\mathbf{x})]^T; \quad \text{matriz MxQ} \qquad (4.2.8)$$

$$\nabla_y h_m(\mathbf{x}) = \left[\frac{\partial \mathbf{h}_m}{\partial \mathbf{y}_1}, \frac{\partial \mathbf{h}_m}{\partial \mathbf{y}_2}, \ldots, \frac{\partial \mathbf{h}_m}{\partial \mathbf{y}_M} \right] \qquad (4.2.9)$$

$$\nabla_z h_m(\mathbf{x}) = \left[\frac{\partial \mathbf{h}_m}{\partial \mathbf{z}_1}, \frac{\partial \mathbf{h}_m}{\partial \mathbf{z}_2}, \ldots, \frac{\partial \mathbf{h}_m}{\partial \mathbf{z}_Q} \right] \qquad (4.2.10)$$

Observando-se a expressão do diferencial de **h** pode-se deduzir que :

$$dy = -J^{-1}.\, C\, .dz \qquad (4.2.11)$$

Substituindo este resultado na expressão do diferencial de f(**x**) vem:

$$df = (\nabla_z f(\mathbf{x})^T - \nabla_y f(\mathbf{x})^T J^{-1}.C).\, dz \qquad (4.2.12)$$

e, define-se o Gradiente Reduzido Generalizado da função f, $\vec{\nabla}_r f(\mathbf{x})^T$, como sendo:

$$\nabla_r f(\mathbf{x})^T = \nabla_z f(\mathbf{x})^T - \nabla_y f(\mathbf{x})^T J^{-1} C \qquad (4.2.13)$$

A expressão do Gradiente Reduzido Generalizado indica a taxa de variação da função-objetivo com relação às variáveis não-básicas, e a equação (4.11) refere-se ao ajuste a ser feito nas variáveis básicas de modo a respeitar-se as restrições do problema. A partir do vetor gradiente reduzido generalizado da função f, pode-se determinar a direção de busca em direção a um ponto de mínimo local.

4.3 - Exemplo

Determinar o mínimo de $f(\mathbf{x}) = x_1^2 + x_2^2$, sujeita à restrição $h(\mathbf{x}) = 2.x_1 + x_2 - 1 = 0$.

Solução:

Pode-se notar que N = 2, J = 0,K = 1, e Q = N - M = 1.

Como só existe uma restrição de igualdade, haverá somente uma variável básica e uma não-básica. A escolha de qual será básica e qual a não-básica é arbitrária . Escolhe-se por exemplo, $\mathbf{z} = x_1$, como variável não-básica e $\mathbf{y} = x_2$, como básica.

A partir das expressões dos diferenciais da função **f** e da igualdade **h**, obtem-se

$$d\mathbf{f} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}_1}.d\mathbf{x}_1 + \frac{\partial \mathbf{f}}{\partial \mathbf{x}_2}.d\mathbf{x}_2 \qquad (4.3.1)$$

e

$$dh = \frac{\partial h}{\partial \mathbf{x}_1}.d\mathbf{x}_1 + \frac{\partial h}{\partial \mathbf{x}_2}.d\mathbf{x}_2 = 0$$

$$d\mathbf{x}_2 = -\frac{\left[\dfrac{\partial h}{\partial \mathbf{x}_1}\right]}{\left[\dfrac{\partial h}{\partial \mathbf{x}_2}\right]}.d\mathbf{x}_1 \qquad (4.3.2)$$

Substituindo a expressão de dx_2 no diferencial de **f** vem

$$df = \left[\frac{\partial f}{\partial x_1} - \frac{\partial f}{\partial x_2} . \frac{\partial h}{\partial x_1} . \left(\frac{\partial h}{\partial x_2} \right)^{-1} \right] . d x_1 \qquad (4.3.3)$$

Onde o gradiente reduzido e generalizado será:

$$\nabla_r f(x)^T = \frac{df}{d x_1} = \frac{\partial f}{\partial x_1} - \frac{\partial f}{\partial x_2} . \frac{\partial h}{\partial x_1} . \left(\frac{\partial h}{\partial x_2} \right)^{-1} \qquad (4.3.4)$$

Resolvendo agora o exemplo em questão, pode-se obter que

$$\frac{df}{d x_1} = 2 . x_1 - 4 . x_2 \qquad dx_2 = -2.dx_1$$

Para uma dada iteração k, pode-se escrever que

$$x_1^{(k+1)} = x_1^{(k)} + \Delta x_1^{(k)} \qquad \Delta x_1^{(k)} = -\lambda^{(k)} . \frac{df^{(k)}}{d x_1}$$

$$x_2^{(k+1)} = x_2^{(k)} + \Delta x_2^{(k)} \qquad \Delta x_2^{(k)} = -2.\Delta x_1^{(k)}$$

$$\frac{df^{(k)}}{dx_1} = 2.x_1^{(k)} - 4.x_2^{(k)}$$

Adotando-se como estimativa inicial, o ponto $x^{(0)} = [1,-1]^T$, chega-se ao seguinte resultado apresentado na tabela 4.3.1. O escalar $\lambda^{(k)}$ é um número não-negativo e corresponde a um fator que , multiplicado pelo vetor gradiente reduzido generalizado, e ajustado convenientemente, pode provocar uma aceleração ou desaceleração da otimização.

O ponto de mínimo pelo método otimizado foi $\mathbf{x}^* = [0{,}4 \;\; 0{,}2]^T$. A busca termina neste ponto pois o vetor gradiente reduzido e generalizado assume o valor zero após a primeira iteração.

Tabela 4.3.1 - Resultado da otimização

k	0	1
f	2	0,2
x_1	1	0,4
x_2	-1	0,2
$\frac{df}{dx_1}$	6	0
Δx_1	-0,6	0
Δx_2	1,2	0
λ	0,1	-----

Resolvendo-se este problema analiticamente, é possível concluir que $\mathbf{x}^* = [0{,}4 \;\; 0{,}2]^T$ é realmente o ponto de mínimo.

Capítulo 5 – Estrutura do Modelo Computacional

5.1 – Introdução

Neste capítulo serão apresentados os programas que foram desenvolvidos e suas principais características , mencionando as suas variáveis de entrada e saída. Os programas foram elaborados em linguagem Fortran 77 e executados em sistema operacional LINUX, distribuição "Red Hat".

5.2 – Fluxograma dos programas

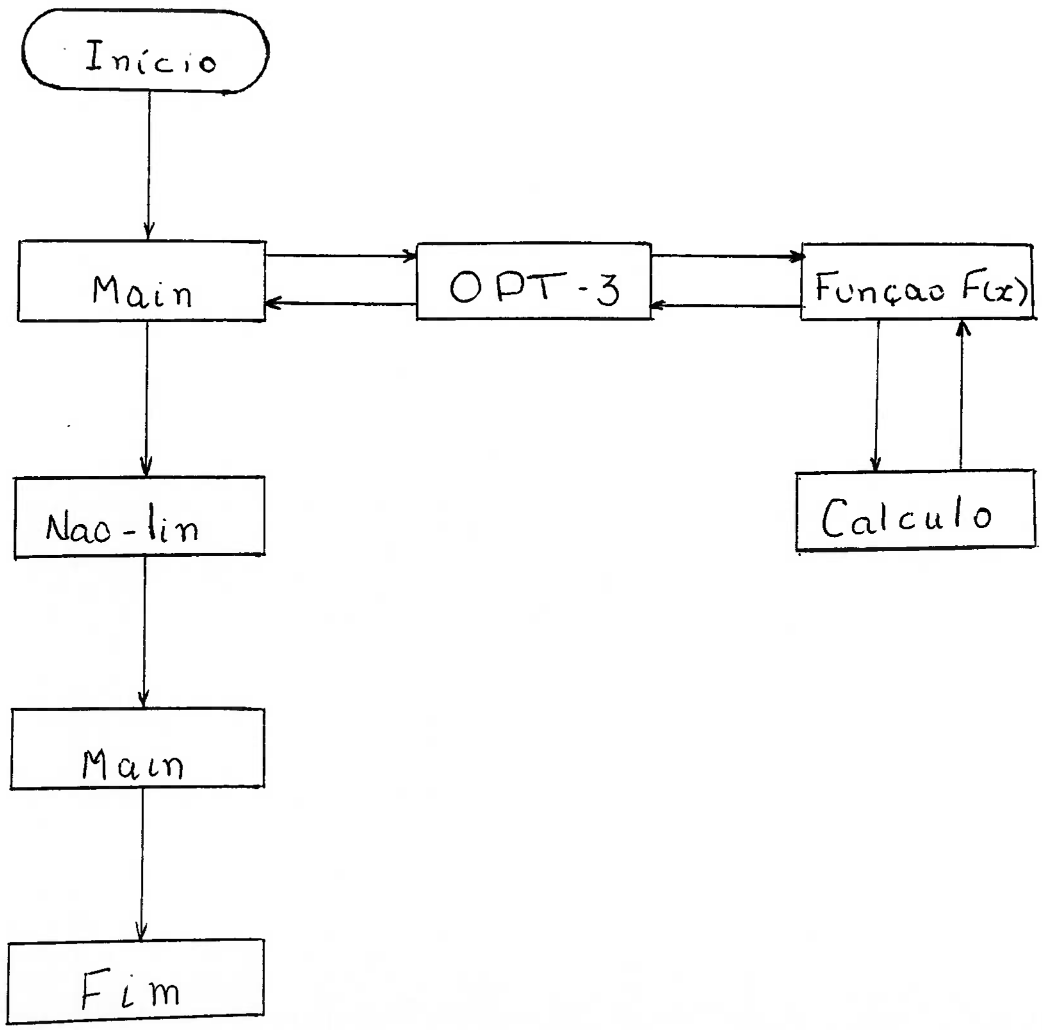


Fig. 5.2.1 – Fluxograma dos programas

5.3 - Descrição das principais funções de cada programa

5.3.1 – Programa principal "MAIN"

Com esta sub-rotina dá-se o início e o fim do programa. Essa sub-rotina tem três objetivos:

1º – Obtenção das características geométricas do mecanismo quadrilátero articulado sem a otimização, ou seja, realizar a síntese e calcular os ângulos de transmissão iniciais.

Para tanto deve-se fornecer, como variável de entrada, as componentes x e y das variáveis de projeto: articulações fixas A_0 e B_0 e dos pontos da trajetória P_1, P_2 e P_3.

Obs.: Esta função do programa 'MAIN' não foi apresentada no fluxograma do início desse capítulo uma vez que seus resultados são somente para aferição do programa de otimização e não fornece dados para o mecanismo final.

2º –Síntese do mecanismo posicionador otimizado. Para isso o programa deverá fornecer, além das variáveis anteriores, articulações fixas e pontos da trajetória, os valores máximos e mínimos dessas variáveis, o número de variáveis de projeto (no nosso caso é 10) além de outros três parâmetros (scale, fscale e isize) para a sub-rotina 'OPT-3'.

Após a entrada destes dados o programa 'MAIN' realiza a síntese do quadrilátero articulado com seu ângulo de transmissão otimizado, determinando algumas variáveis de saída, como as coordenadas x e y dos pontos A e B na primeira posição de precisão.

3º – Síntese do mecanismo acionador. Para isso o programa 'MAIN' já obteve o quadrilátero articulado otimizado, que fornece as seguintes informações: o valor do ânguloβ_2, comprimento $\overline{A_0A_1}$ da barra do quadrilátero, além das coordenadas de A_0. Para 'chamar' a sub-rotina 'NÃO-LIN', o programa 'MAIN' deverá fornecer os valores dos raios R e ra das engrenagens – determinados na sub-rotina 'CALCULO' – além das coordenadas do ponto fixo C_0, que é fornecida (atribuída) pelo projetista.

E, finalmente, o programa 'MAIN' determina os ângulos ν_1 e ν_2.

5.3.2 - Sub-rotina "OPT-3"

Esta sub-rotina, que se baseia no método do Gradiente Reduzido Generalizado (GRG), para minimização da função-objetivo, é responsável pela alteração dos valores das variáveis de projeto.

5.3.3 - Função F(x)

Esta função calcula a função-objetivo a ser otimizada. Além disso, analisa o comportamento cinemático dos mecanismos calculados durante o processo de otimização, verificando se os mesmos não passarão posições que poderão causar travamento durante o deslocamento entre as posições pré-estabelecidas .

5.3.4 - Sub-rotina "Cálculo"

É nesta sub-rotina que são calculadas as dimensões das barras dos mecanismos, os valores do ângulo de transmissão do quadrilátero articulado e fornecidos também os dados para a sub-rotina 'NÃO-LIN', que são: os valores da coordenada fixa Co, e os raios ra e R das engrenagens.

5.3.5 - Sub-rotina "Não-lin"

Os dados iniciais desta sub-rotina são os valores das coordenadas de Co e os raios ra e R, das engrenagens conforme descrito no ítem anterior. Com essas informações calculam-se os ângulos ν_1 e ν_2, variáveis de saída, mencionados na seção 3.3.

Capítulo 6 - Exemplos de aplicação

6.1 –Introdução

Neste capítulo, os programas desenvolvidos serão aplicados em três exemplos. Em cada exemplo serão apresentados os dados que caracterizam o mecanismo sem a otimização e o mecanismo otimizado.

6.2 – Exemplo 1

6.2.1 – Síntese sem otimização

Tabela 6.2.1.1 – Coordenadas dos pontos notáveis do mecanismo sem a otimização

	x	y
Ao	5	0
Bo	0	0
A_1	3,5477	-1,6545
B_1	0,994	3,238
P_1	1	1
P_2	2	0,5
P_3	3	1,5

Tabela 6.2.1.2 – Valores dos ângulos β para a 1ª e 2ª díadas

Ângulos (graus)	1ª. díada	2ª. díada
β_2	29,42	-19
β_3	171,59	-7,59

$f(x)=2,847$

$\alpha_2 = 0°$

$\alpha_3 = 45°$

Tabela 6.2.1.3- Valores máximos e mínimos das coordenadas dos pontos Ao, Bo, P_1, P_2 e P_3

	Máximo	Mínimo
A_{0x}	7.0	3.0
A_{0y}	2.0	-2.0
B_{0x}	2.0	-2.0
B_{0y}	2.0	-2.0
P_{1x}	1.2	0.8
P_{1y}	1.2	0.8
P_{2x}	2.2	1.8
P_{2y}	0.7	0.3
P_{3x}	3.2	2.8
P_{3y}	1.7	1.3

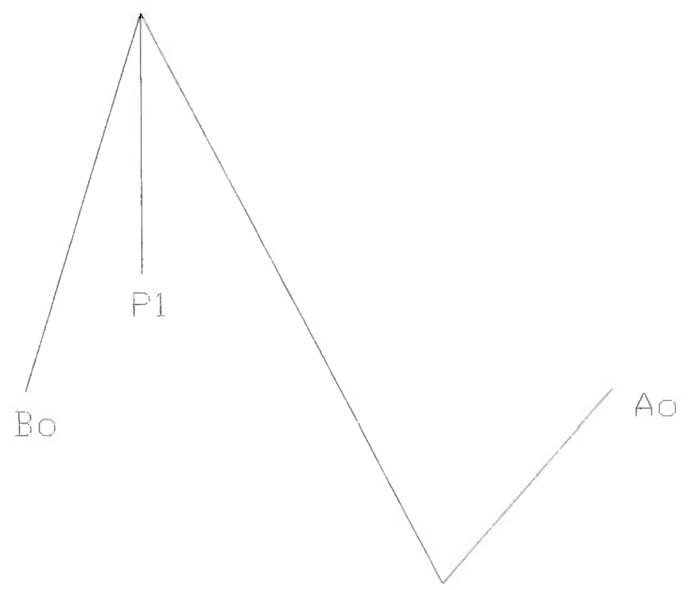


Fig. 6.2.1.1 – Exemplo-1: primeira posição, mec. sem otimização

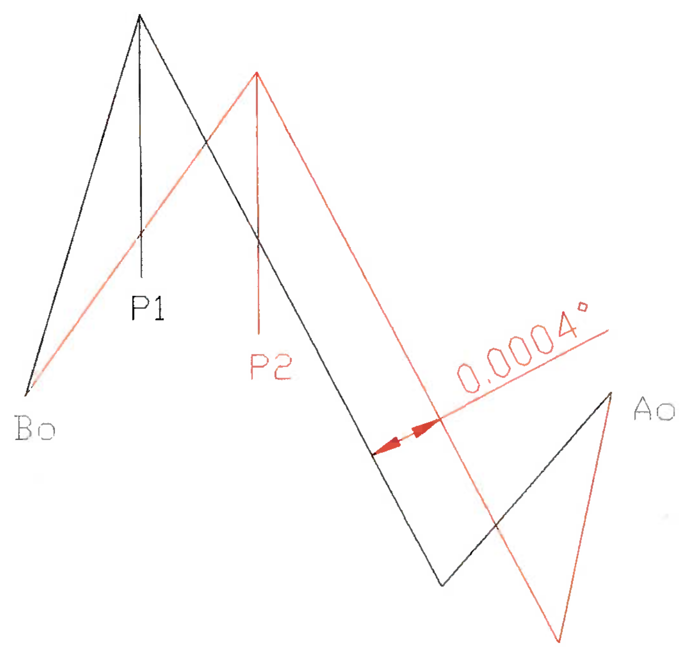


Fig. 6.2.1.2 – Exemplo-1: primeira e segunda posições, mec. sem otimização

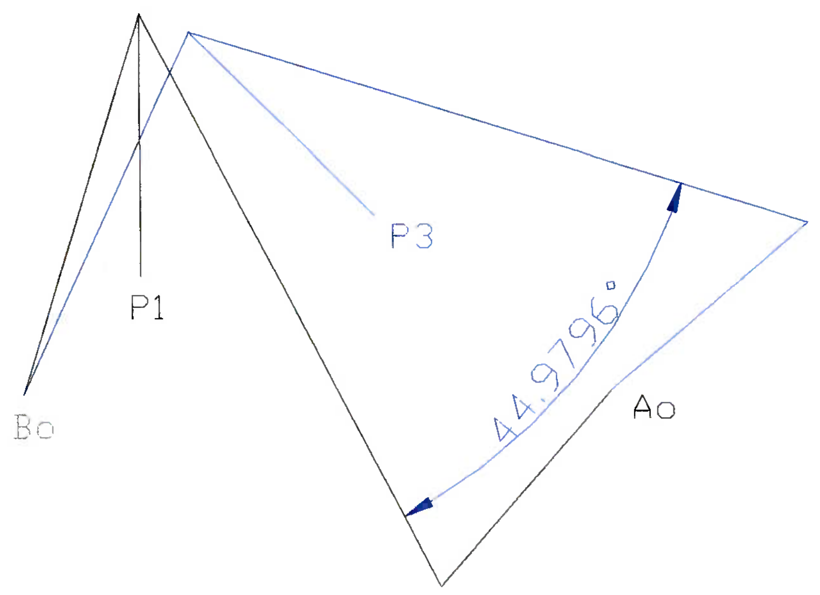


Fig. 6.2.1.3 – Exemplo-1: primeira e terceira posições, mec. sem otimização

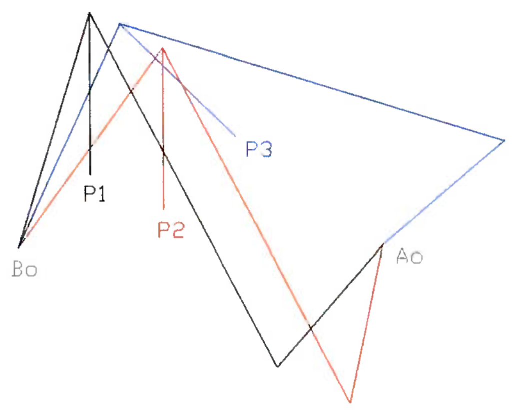


Fig 6.2.1.4. – Exemplo-1: com as três posições, mec. sem otimização

6.2.2 – Síntese com otimização

Tabela 6.2.2.1 – Coordenadas de pontos notáveis do mecanismo otimizado

	x	y
Ao	7	0
Bo	0	0
A_1	5,3	-1,9
B_1	2,62	4,55
P_1	1,2	0,8
P_2	1,8	0,4
P_3	2,8	1,47

Tabela 6.2.2.2 – Valores dos ângulos β para a 1ª e 2ª díadas

Ângulos (graus)	1ª. díada	2ª. díada
β_2	16,36	-7,9
β_3	-151,58	17,26

f(x)=1,5211

ni1 = 1.24

ni2 = 90.62

Co = (6.248;1.008)

R = 1.425

ra = 0.8463

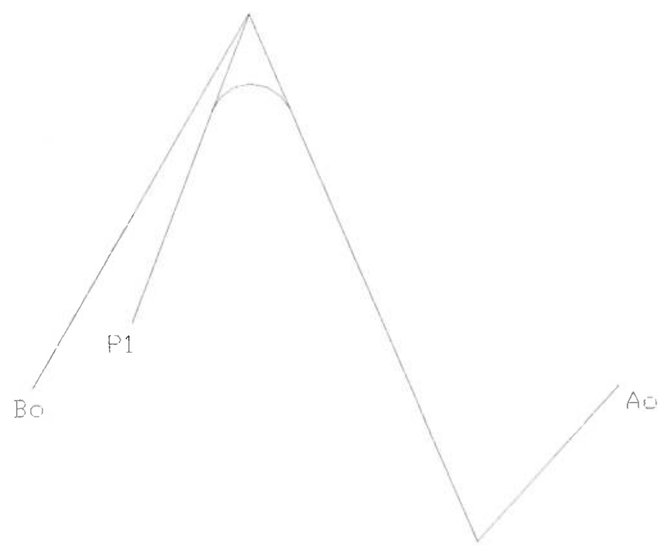


Fig. 6.2.2.1 –Exemplo –1: primeira posição, mecanismo otimizado

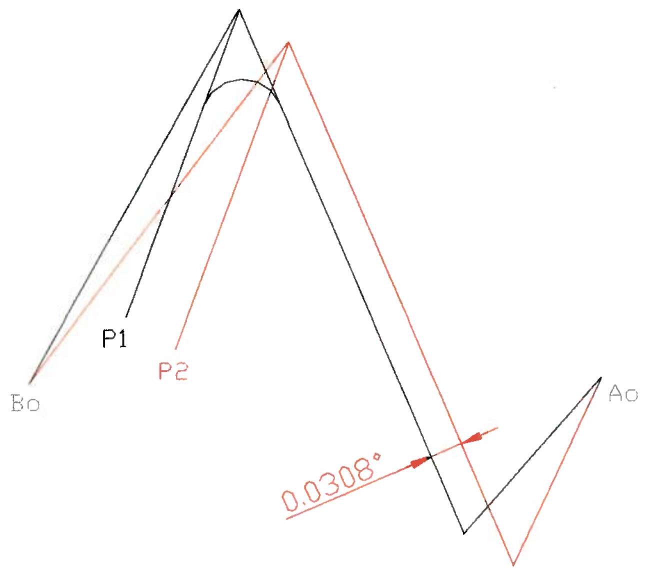


Fig. 6.2.2.2 - Exemplo-1: primeira e segunda posições, mecanismo otimizado

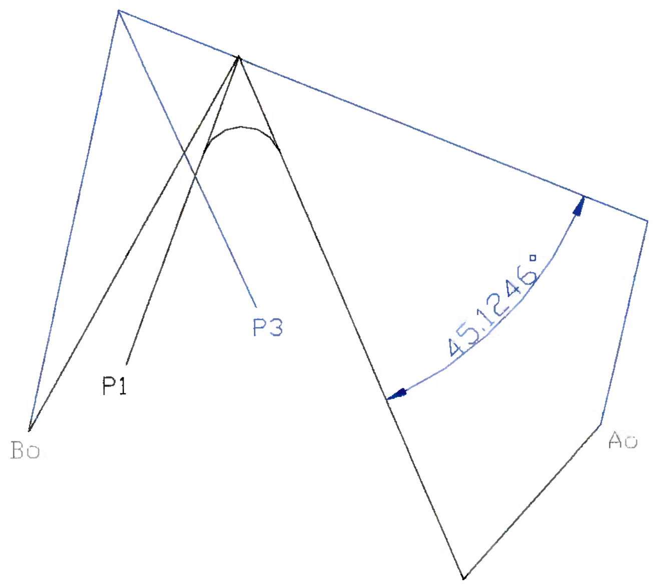


Fig. 6.2.2.3 - Exemplo-1 primeira e terceira posições, mecanismo otimizado

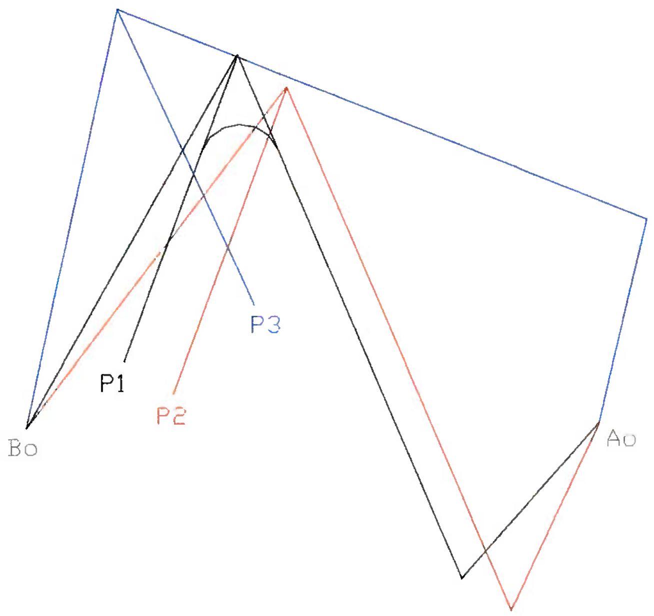


Fig 6.2.2.4 – Exemplo-1: nas três posições, mecanismo otimizado

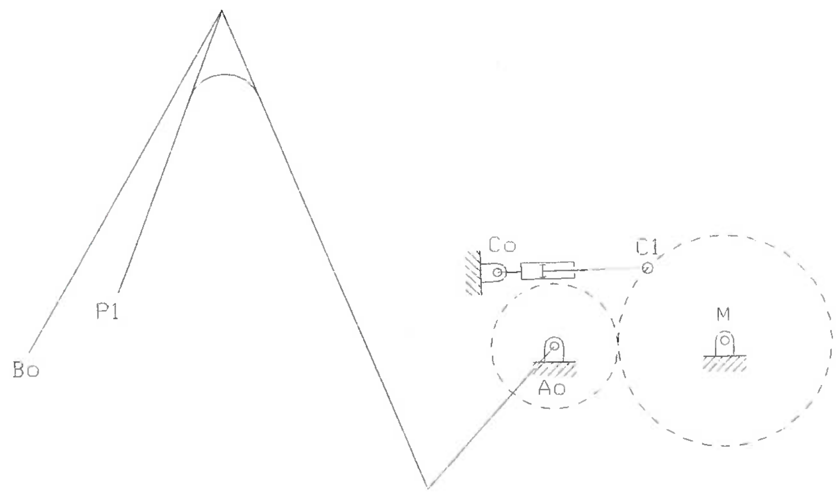


Fig 6.2.2.5 – Exemplo-1: mecanismo otimizado com o sistema de acionamento

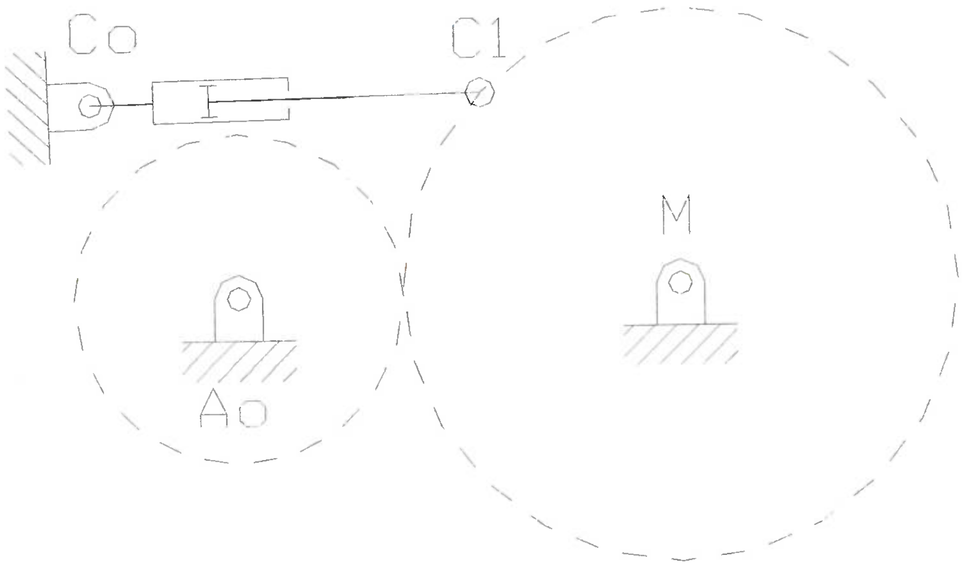


Fig 6.2.2.6- Detalhe do mecanismo acionador

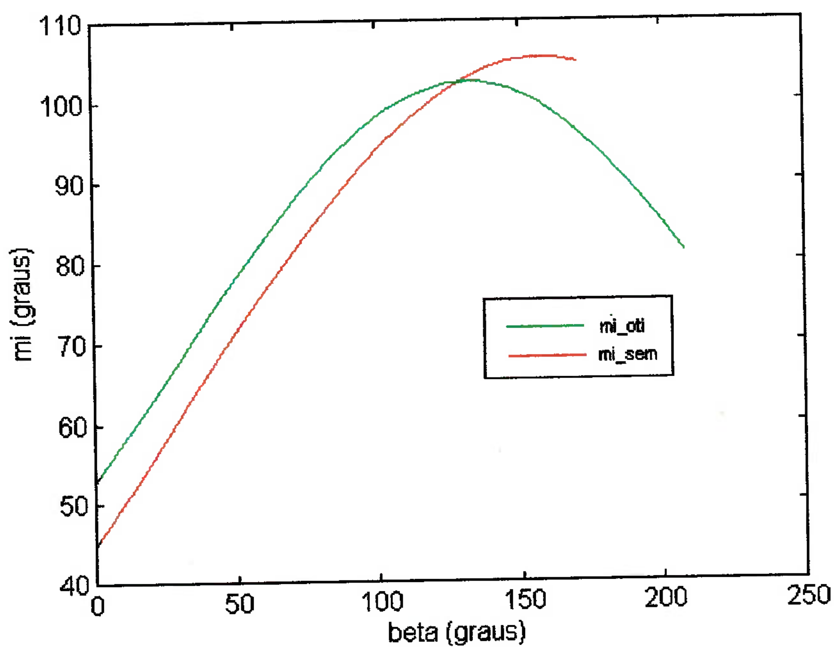


Fig. 6.2.2.7 –Gráfico do ângulo μ em função de β para os mecanismos sem e com otimização

Tabela 6.2.2.3 – Valores do ângulo μ

mi (graus)	sem	oti
1	44.6266585	52.7732844
2	49.1594895	57.7635434
3	53.762791	63.0344276
4	58.386484	68.4251611
5	62.9893347	73.7987576
6	67.5352626	79.0324104
7	71.9908066	84.0100199
8	76.3233525	88.6171121
9	80.4998355	92.7384072
10	84.4858382	96.258509
11	88.2449625	99.0659762
12	91.7385905	101.061006
13	94.9259664	102.165516
14	97.7648115	102.333347
15	100.212437	101.557308
16	102.227493	99.8702856
17	103.772243	97.3401104
18	104.815192	94.0601375
19	105.333685	90.1387075
20	105.316009	85.6904786
21	104.762572	80.8310177

6.3 – Exemplo 2

6.3.1 – Síntese sem otimização

Tabela 6.3.1.1 – Coordenadas de pontos notáveis do mecanismo sem otimização

	x	y
Ao	-15.9	23.5
Bo	-8.9	36.5
A_1	-58.19	25.02
B_1	-43.02	46
P_1	-19.21	-1.62
P_2	-19.37	12.18
P_3	-13.33	54.60

Tabela 6.3.1.2 – Valores dos ângulos β para a 1ª e 2ª díadas

Ângulos (graus)	1ª. díada	2ª. díada
β_2	-32.30	-35.67
β_3	-170.97	140.32

$$f(x)= 6.7445$$

$$\alpha_2 = 13°$$

$$\alpha_3 = 179°$$

Tabela 6.3.1.3 – Valores máximos e mínimos das coordenadas dos pontos Ao, Bo, P_1, P_2 e P_3

	Máximo	Mínimo
A_{0x}	-15.0	-17.0
A_{0y}	24.5	21.0
B_{0x}	-8.0	-11.0
B_{0y}	37.5	33.0
P_{1x}	-17.0	-21.0
P_{1y}	1.0	-4.0
P_{2x}	-17.0	-21.0
P_{2y}	14.0	10.0
P_{3x}	-11.0	-15.0
P_{3y}	57.0	52.0

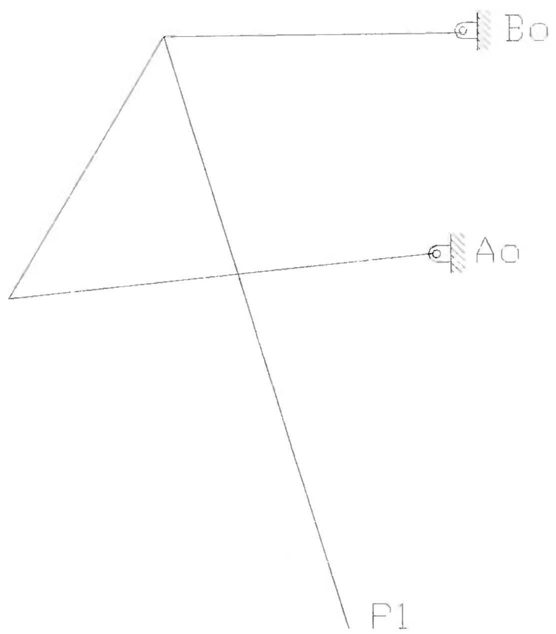


Fig. 6.3.1.1 – Exemplo-2: primeira posição, mec. sem otimização

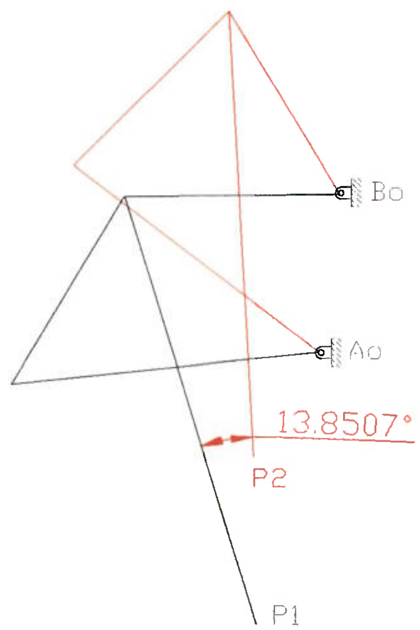


Fig. 6.3.1.2 – Exemplo-2: primeira e segunda posições, mec. sem otimização

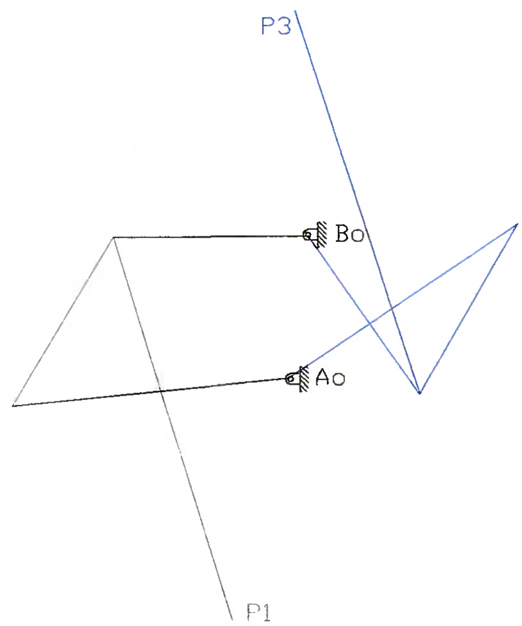


Fig. 6.3.1.3 – Exemplo-2: primeira e terceira posições, mec. sem otimização

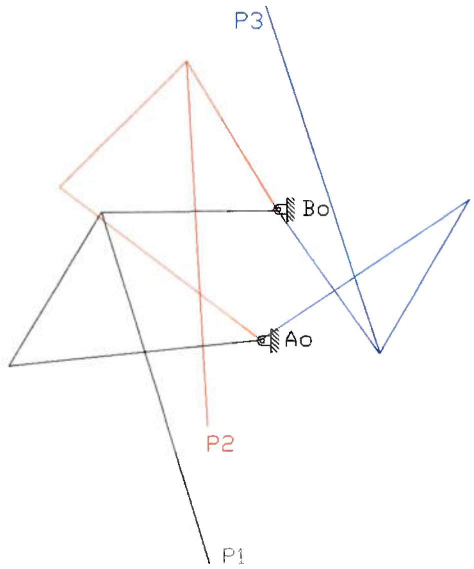


Fig. 6.3.1.4 – Exemplo-2: com as três posições, mec. sem otimização

6.3.2 – Síntese com otimização

Tabela 6.3.2.1 – Coordenadas de pontos notáveis do mecanismo otimizado

	x	y
Ao	-15	21
Bo	-8.92	37.5
A_1	-73.45	23.16
B_1	-65.74	60.08
P_1	-18.73	-1.61
P_2	-19.5	11.25
P_3	-12.77	56.29

Tabela 6.3.2.2 – Valores dos ângulos β para a 1ª e 2ª díadas

Ângulos (graus)	1ª. díada	2ª. díada
β_2	-25.83	-25.73
β_3	-166.58	157.92

f(x)= 0.4384

ni 1= 270.23°

ni2 = 200.06°

Co = (-65; 28.56)

R = 36

ra = 19.5

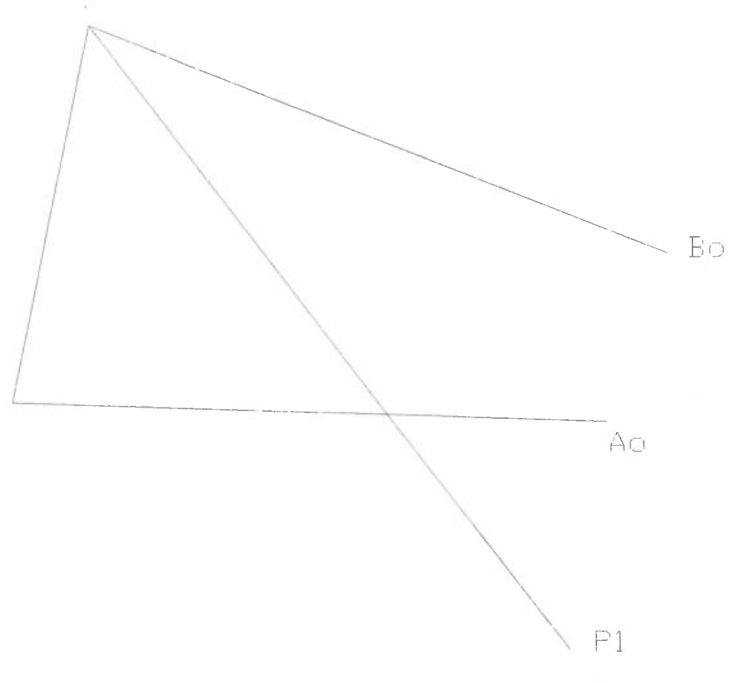


Fig. 6.3.2.1 –Exemplo –2: primeira posição, mecanismo otimizado

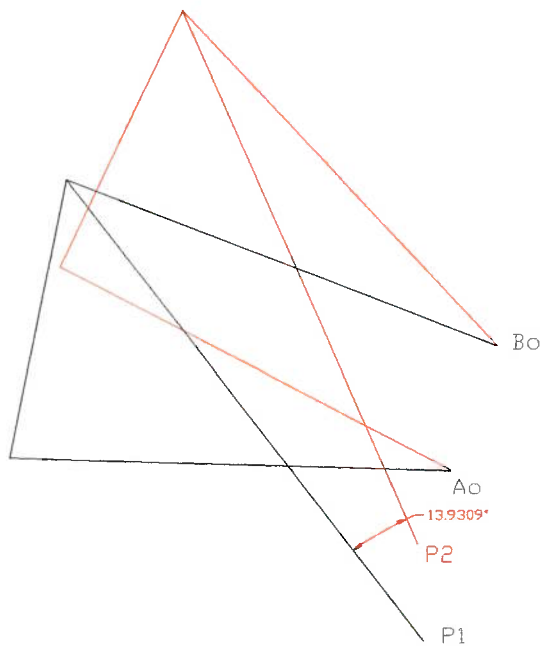


Fig. 6.3.2.2 - Exemplo-2: primeira e segunda posições, mecanismo otimizado

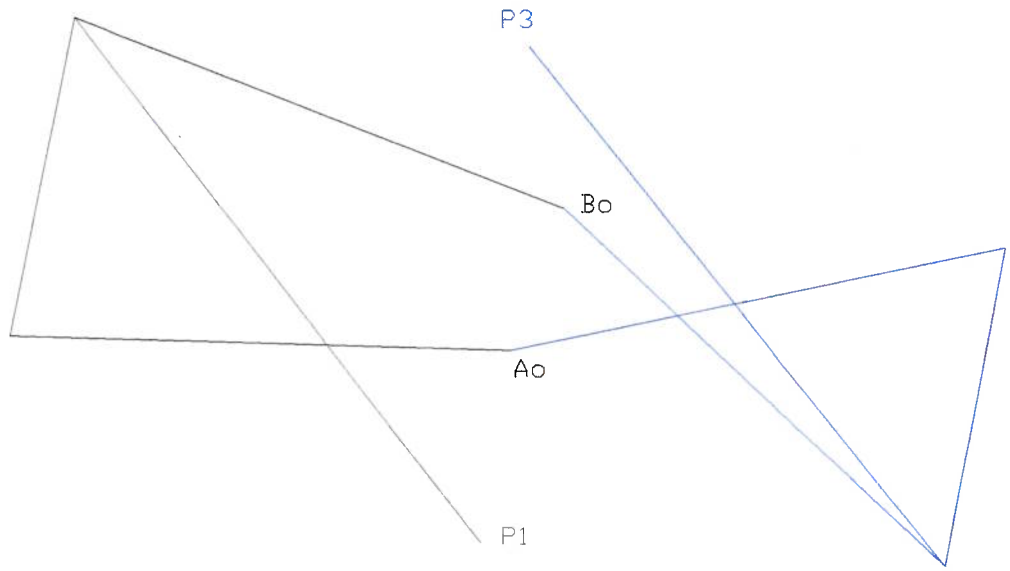


Fig. 6.3.2.3 - Exemplo-2: primeira e terceira posições, mecanismo otimizado

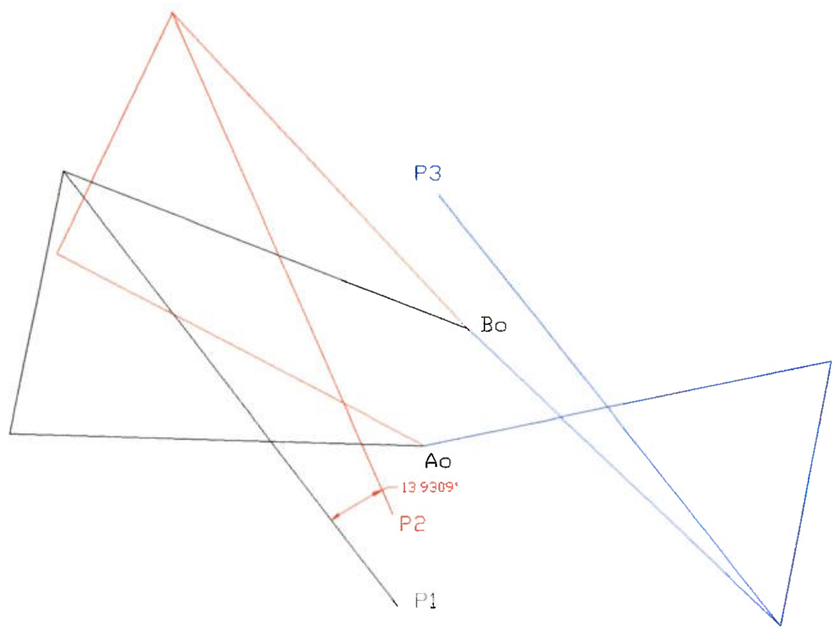


Fig 6.3.2.4 – Exemplo-2: nas três posições, mecanismo otimizado

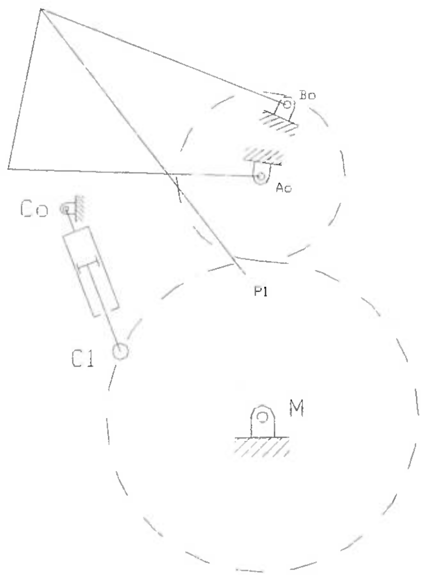


Fig 6.3.2.5 – Exemplo-2: mecanismo otimizado com o sistema de acionamento

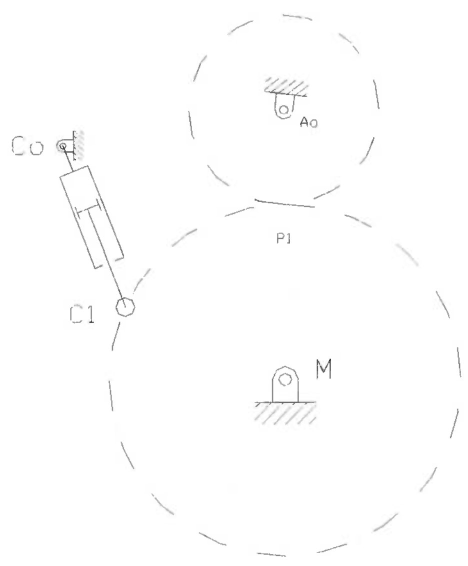


Fig 6.3.2.6 - Detalhe do mecanismo de acionamento

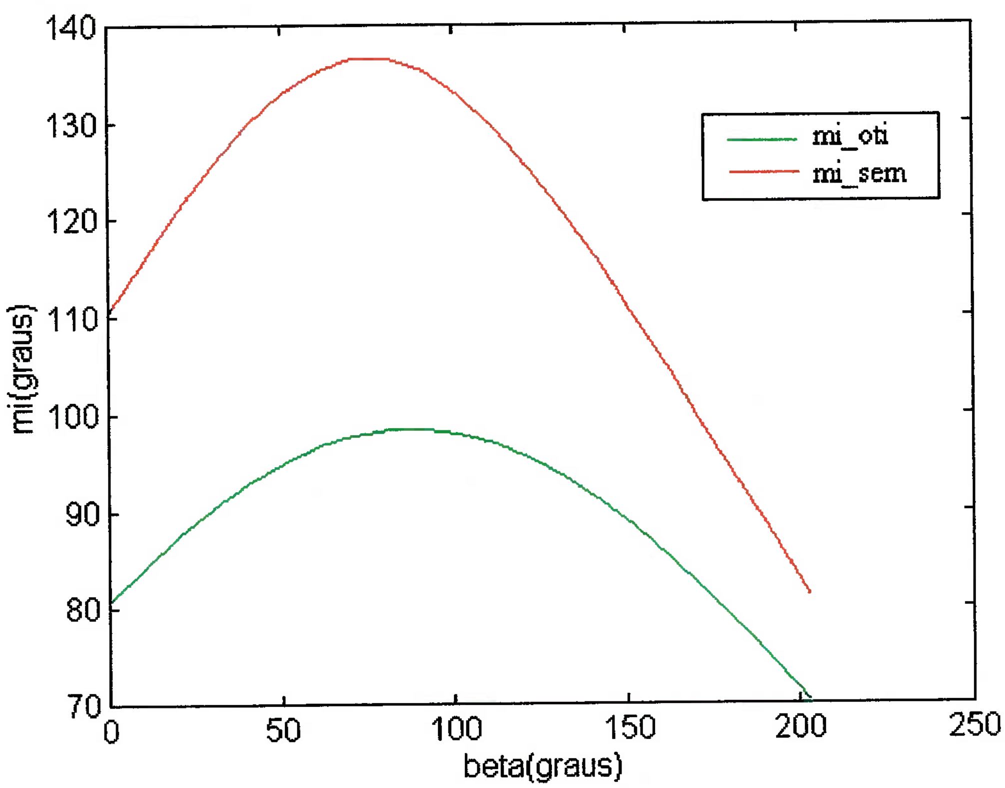


Fig. 6.3.2.7 –Gráfico do ângulo μ em função de β para os mecanismos sem e com otimização

Tabela 6.3.3.3 – Valores do ângulo μ

mi (graus)	sem otim.	otimizado
1	110.34	80.70
2	115.79	84.18
3	120.93	87.40
4	125.63	90.30
5	129.72	92.83
6	133.04	94.94
7	135.37	96.59
8	136.57	97.73
9	136.52	98.42
10	135.23	98.42
11	132.80	97.94
12	129.41	96.94
13	125.26	95.42
14	120.52	93.42
15	115.35	90.99
16	109.88	88.18
17	104.20	85.04
18	98.41	81.62
19	92.57	77.98
20	86.76	74.18
21	81.06	70.26

6.4 - Exemplo 3

6.4.1 – Síntese sem otimização

Tabela 6.4.1.1 – Cooredenadas de pontos notáveis do mecanismo sem otimização

	X	Y
Ao	1.2	0.8
Bo	0.2	0.8
A1	1.106	0.727
B1	0.142	0.9605
P1	1.2	2.5
P2	0.8	2.5
P3	0.4	2.5

Tabela 6.4.1.2 – Valores dos ângulos β para a 1ª e 2ª díadas

Ângulos (°)	1ª díada	2ª díada
β_2	143.30	- 149.47
β_3	- 130.29	178.10

$$F(x) = 5.1355$$

$$\alpha_2 = 20°$$

$$\alpha_3 = 30°$$

Tabela 6.4.1.3 – Valores máximos e mínimos das coordenadas dos pontos Ao, Bo, P_1, P_2 e P_3

	Máximo	Mínimo
A_{0x}	1.5	1.0
A_{0y}	1.0	0.7
B_{0x}	0.8	0.0
B_{0y}	1.0	0.7
P_{1x}	1.5	1.0
P_{1y}	2.6	2.4
P_{2x}	1.0	0.6
P_{2y}	2.6	2.4
P_{3x}	0.6	0.3
P_{3y}	2.6	2.4

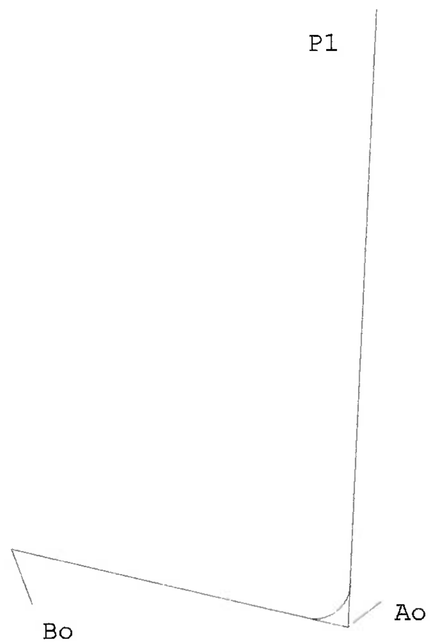


Fig. 6.4.1.1 – Exemplo-3: primeira posição, mec. sem otimização

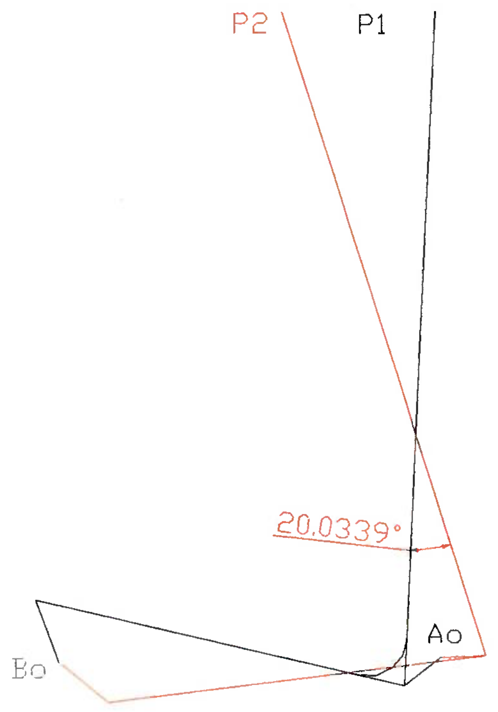


Fig. 6.4.1.2 – Exemplo-3: primeira e segunda posições, mec. sem otimização

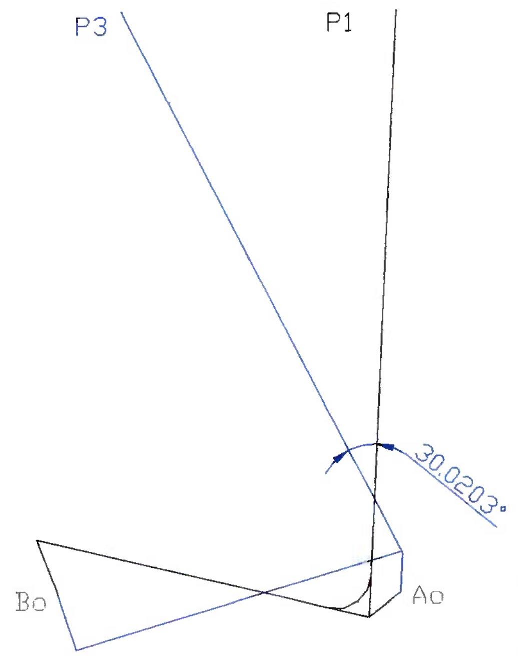


Fig. 6.4.1.3 – Exemplo-3: primeira e terceira posições, mec. sem otimização

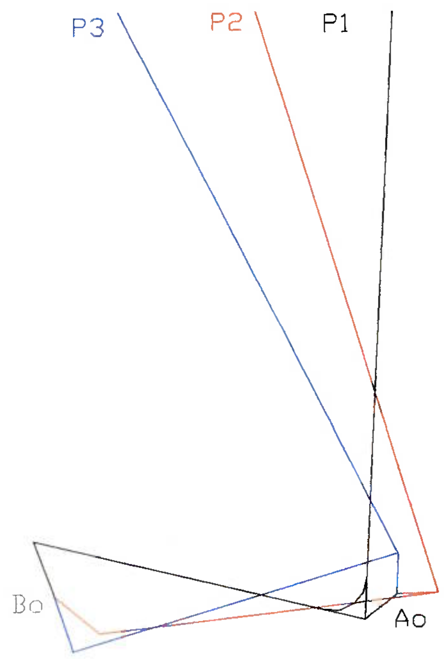


Fig. 6.4.1.4 – Exemplo-3: com as três posições, mec. sem otimização

6.4.2 – Síntese com otimização

Tabela 6.4.2.1 – Coordenadas dos pontos notáveis do mecanismo otimizado

Pontos	X	Y
Ao	1.5	1
B o	0.04	0.7
A1	1.58	0.8
B1	-0.04	0.91
P1	1.26	2.502
P2	0.834	2.45
P3	0.44	2.52

Tabela 6.4.2.2 – Valores dos ângulos β para a 1ª e 2ª díadas

Ângulos (°)	1ª díada	2ª díada
β_2	58.08	- 172.67
β_3	- 218.23	175.12

F(**x**) = 0.1366

ni 1= -28.17 °

ni 2 = 290.67°

Co = (2.1; 0.9)

R = 4.983

ra = 1.379

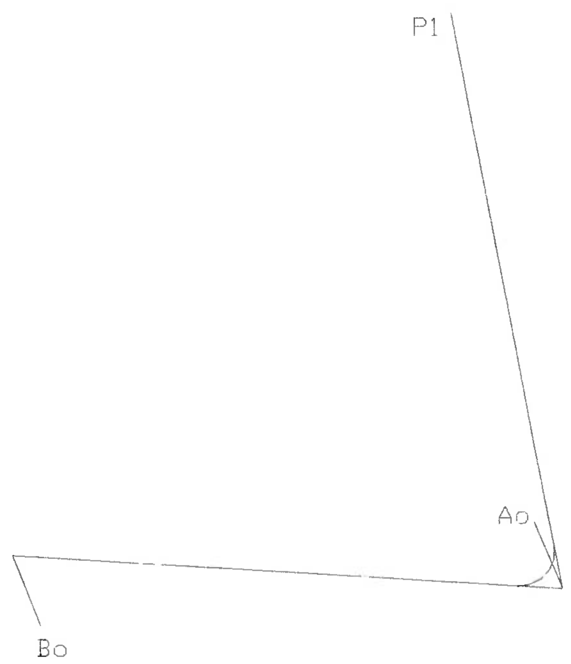


Fig. 6.4.2.1 – Exemplo-3: primeira posição, mec. otimizado

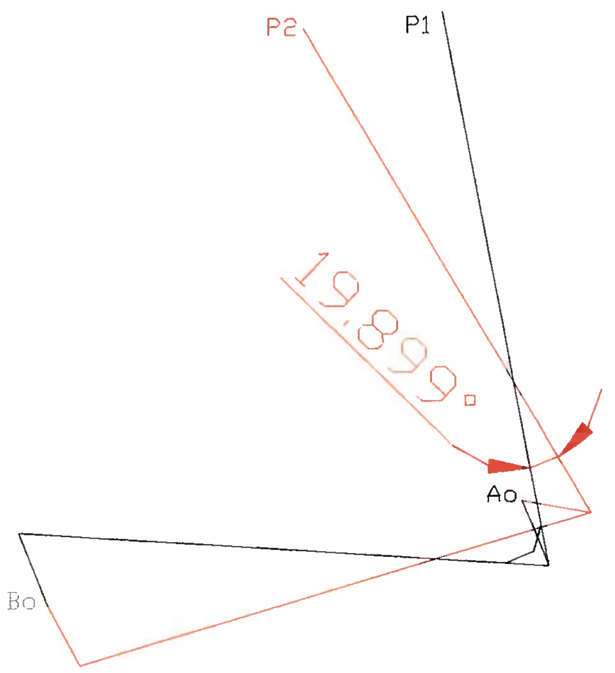


Fig. 6.4.2.2 – Exemplo-3: primeira e segunda posições, mec. otimizado

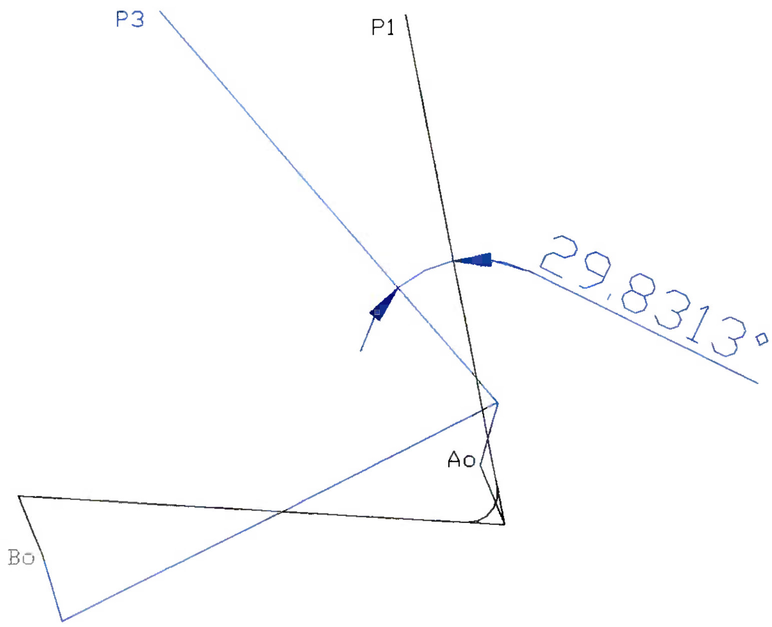


Fig. 6.4.2.3 – Exemplo-3: primeira e terceira posições, mec. otimizado

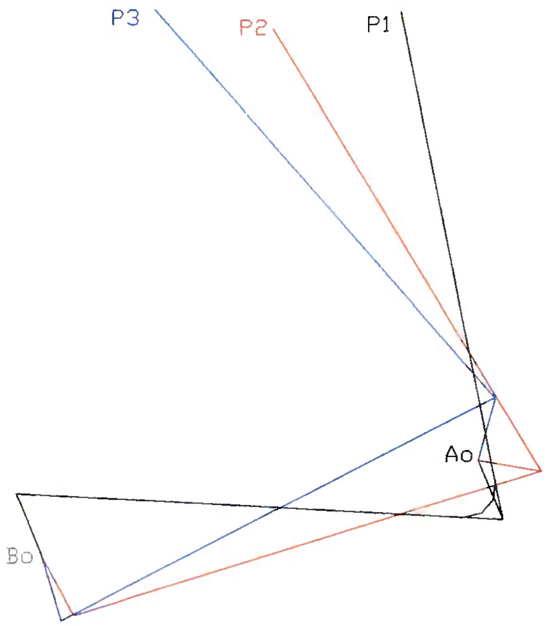


Fig. 6.4.2.4 – Exemplo-3,: com as três posições, mec. otimizado

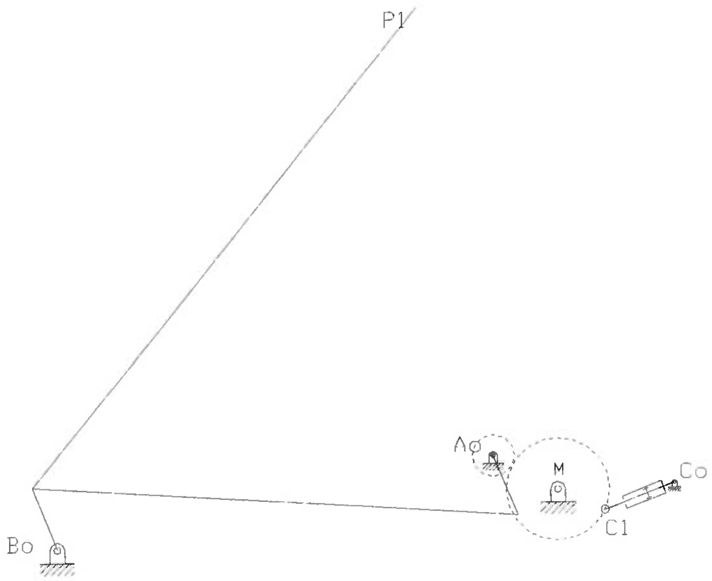


Fig 6.4.2.5 - Exemplo- 3: mecanismo otimizado com o sistema de acionamento

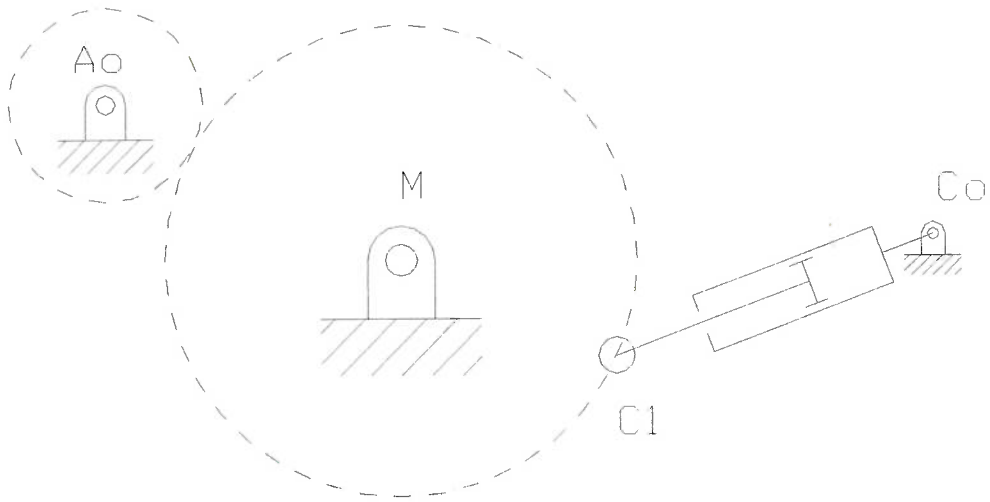


Fig 6.4.2.6 - Detalhe do mecanismo de acionamento

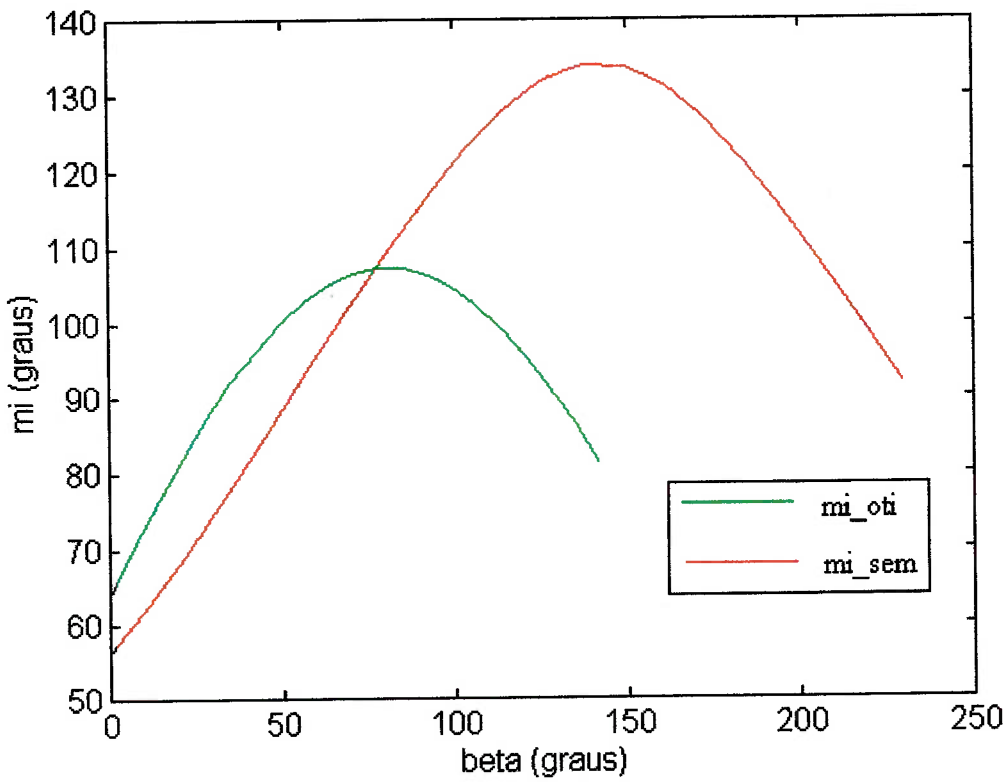


Fig. 6.4.2.7 –Gráfico do ângulo μ em função de β para os mecanismos sem e com otimização

Tabela 6.4.2.3 – Valores do ângulo μ

mi (graus)	sem otim.	otimizado
1	56.42	64.03
2	62.81	70.60
3	69.97	76.69
4	77.61	82.32
5	85.53	87.49
6	93.54	92.17
7	101.46	96.31
8	109.14	99.88
9	116.34	102.81
10	122.80	105.05
11	128.17	106.54
12	132.03	107.25
13	133.96	107.16
14	133.70	106.27
15	131.29	104.61
16	127.04	102.21
17	121.39	99.13
18	114.72	95.42
19	107.39	91.15
20	99.64	86.36
21	91.68	81.10

Capítulo 7 - Discussão

7.1 – Introdução

Inicialmente será descrita e comentada a metodologia empregada durante a realização deste trabalho.

Também serão apresentadas as principais dificuldades que surgiram em cada um dos três exemplos apresentados.

7.2 – Descrição e comentários da metodologia empregada

Conforme apresentado no capítulo 3, o problema da síntese de mecanismos planos articulados movidos por acionadores prismáticos é dividido em dois tópicos: síntese do mecanismo posicionador e síntese do mecanismo acionador.

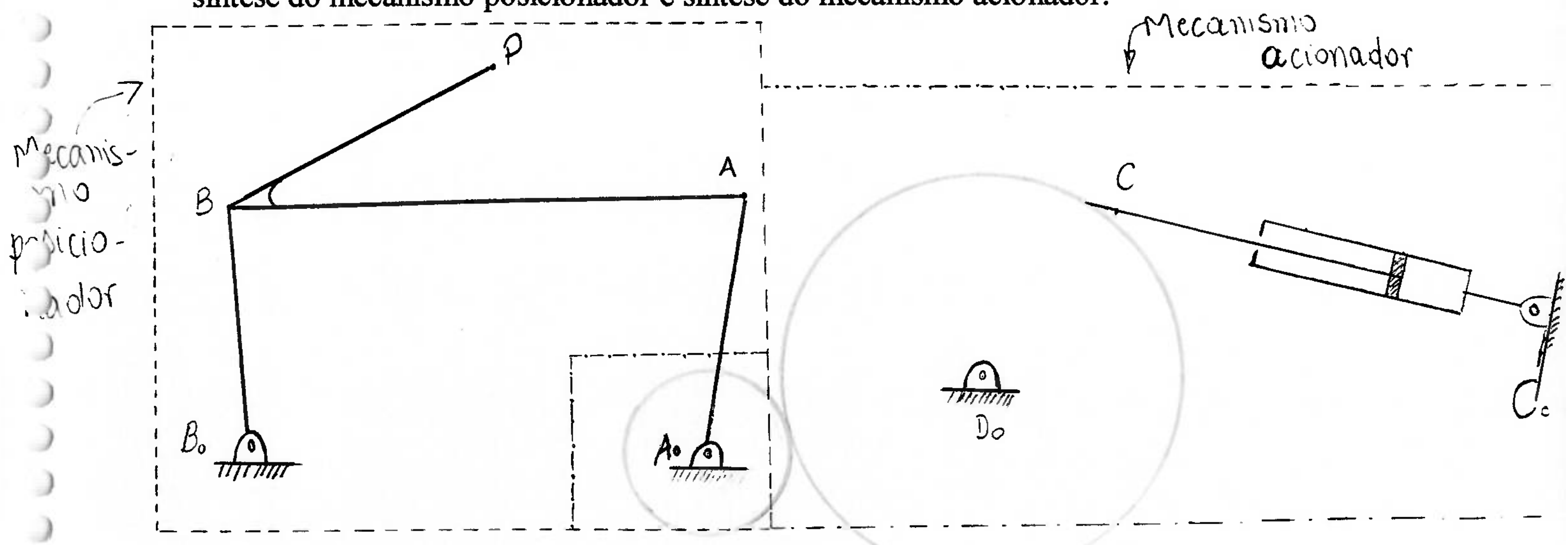


fig. 7.2.1- Mecanismos posicionador a acionador.

Com relação ao mecanismo posicionador, que na verdade é um quadrilátero articulado plano, o problema de sua síntese é definido para realização da tarefa chamada de "geração de movimento". Nesta tarefa, a peça $\overline{AB}$ deve ser orientada em 3 posições de precisão. Para tanto, basta especificar 3 posições ocupadas por um ponto P pertencente a essa peça e dois deslocamentos angulares α_2 e α_3.

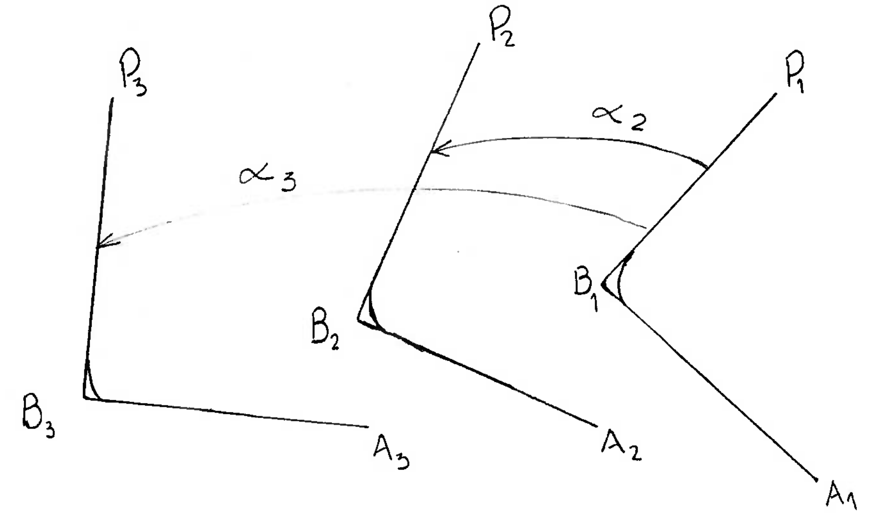


Fig. 7.2.2 – Variáveis especificadas para síntese do mecanismo posicionador

Sobre a determinação do quadrilátero, é necessário calcular suas duas díadas, de modo a obter os comprimentos $\overline{\mathbf{A_0A_1}}$ e $\overline{\mathbf{A_1P_1}}$ e depois $\overline{\mathbf{B_0B_1}}$ e $\overline{\mathbf{B_1P_1}}$. Na modelagem matemática foi empregada a notação polar complexa (ver cap. 9), por se tratar de uma notação mais compacta e normalmente utilizada na descrição da operação de rotação de um vetor. Como pôde ser visto no capítulo 3, no cálculo de cada díada surgem duas respostas matematicamente viáveis, portanto, nos dá, com a combinação de duas a duas, quatro mecanismos de solução para a mesma síntese. O critério adotado para a escolha de uma das alternativas foi o comprimento das barras. Dentre os quatro mecanismos possíveis um deles possuía barras menores e outros três barras maiores, a diferença entre estas barras em termos de ordem de grandeza era, para as menores 10^1 e para as maiores $>10^4$, o que implicava em três mecanismos com pelo menos uma das barras muito grande e somente um deles com as dimensões compatíveis para uma possível construção física. Esse último era o mecanismo escolhido para realizar a síntese.

Com o quadrilátero articulado definido, calcula-se a variável ângulo de transmissão μ nas três posições de precisão. Para uma melhor representação do comportamento do ângulo μ ao longo da movimentação do mecanismo, procedeu-se a uma sub-divisão do intervalo definido por β_3 em vinte sub-intervalos iguais. O ângulo de transmissão pode ser obtido da equação matemática a seguir, conforme descrito na seção 3.8:

$$\mu = a\cos\left(\frac{z_3^2 + z_4^2 - d^2}{2z_3z_4}\right)$$

No caso do co-seno de μ assumir valores maiores que 1 ou menores que -1, significando que não é possível a montagem do triângulo z_3, z_4 e d, penaliza-se a função-objetivo, f(**x**), ou seja, atribui-se a ela um valor elevado para que a sub-rotina OPT-3 procue por outros valores do vetor **x** dentro do domínio previamente estabelecido.

Com relação à síntese do mecanismo acionador, admite-se que o projetista especifique previamente as coordenadas do ponto C_0, ou seja, a articulação fixa do acionador prismático. A seguir, resolvendo-se um sistema de equações não-lineares, pode-se determinar os ângulos v_1 e v_2 que definem a orientação das peças do mecanismo acionador em relação ao posicionador. Uma observação deve ser feita quanto à determinação dos ângulos v_1 e v_2 . Ao sintetizar-se o mecanismo posicionador, sabe-se qual o sentido de rotação da alavanca $\overline{AA_0}$, se horário ou anti-horário, que na verdade é definido pela variável β_2. Sabendo-se o sentido da rotação da alavanca $\overline{AA_0}$, a modelagem matemática assume que o acionador prismático esteja recolhido na primeira posição (posição inicial) e distendido na terceira posição (posição final)

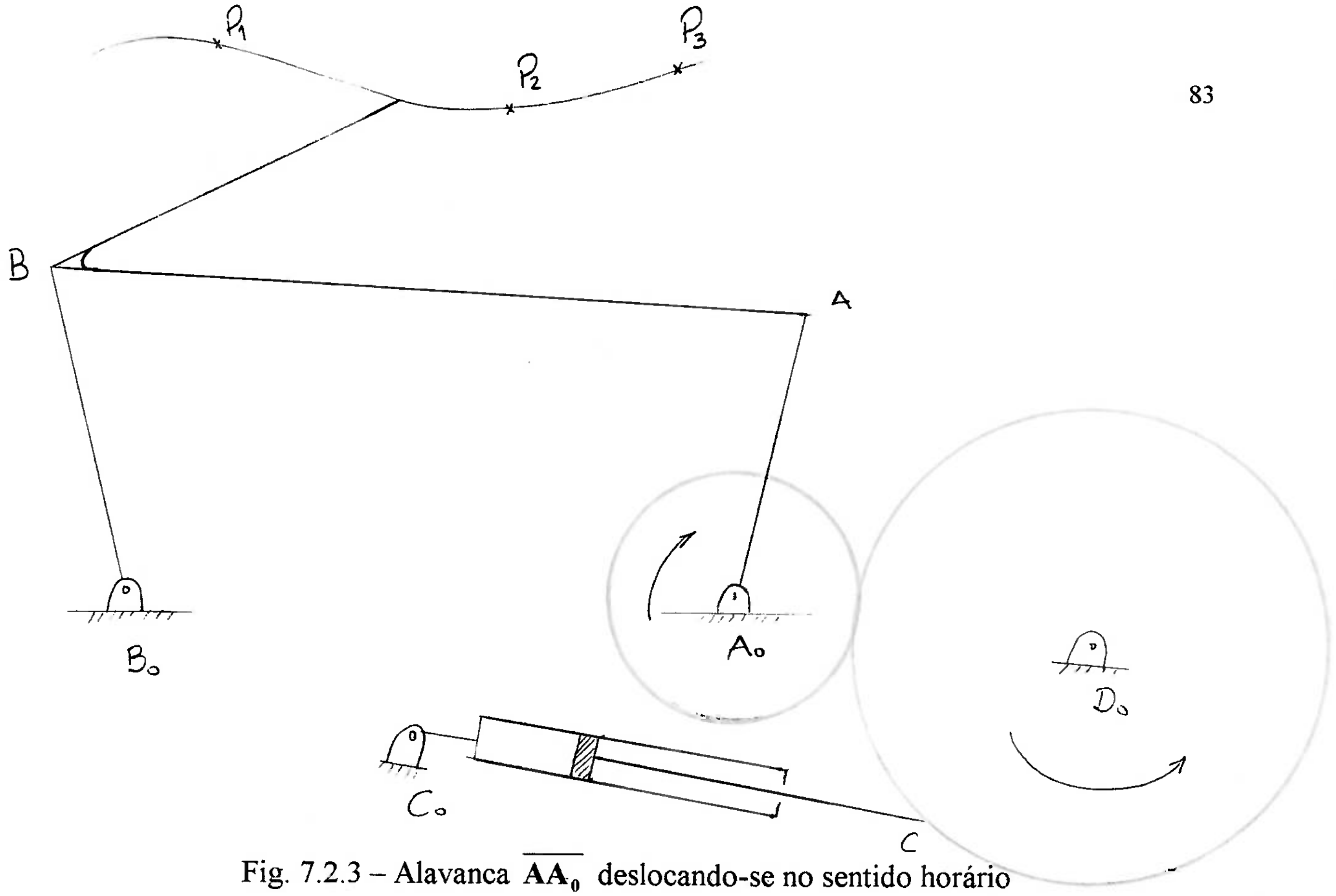


Fig. 7.2.3 – Alavanca $\overline{\mathbf{AA}_0}$ deslocando-se no sentido horário

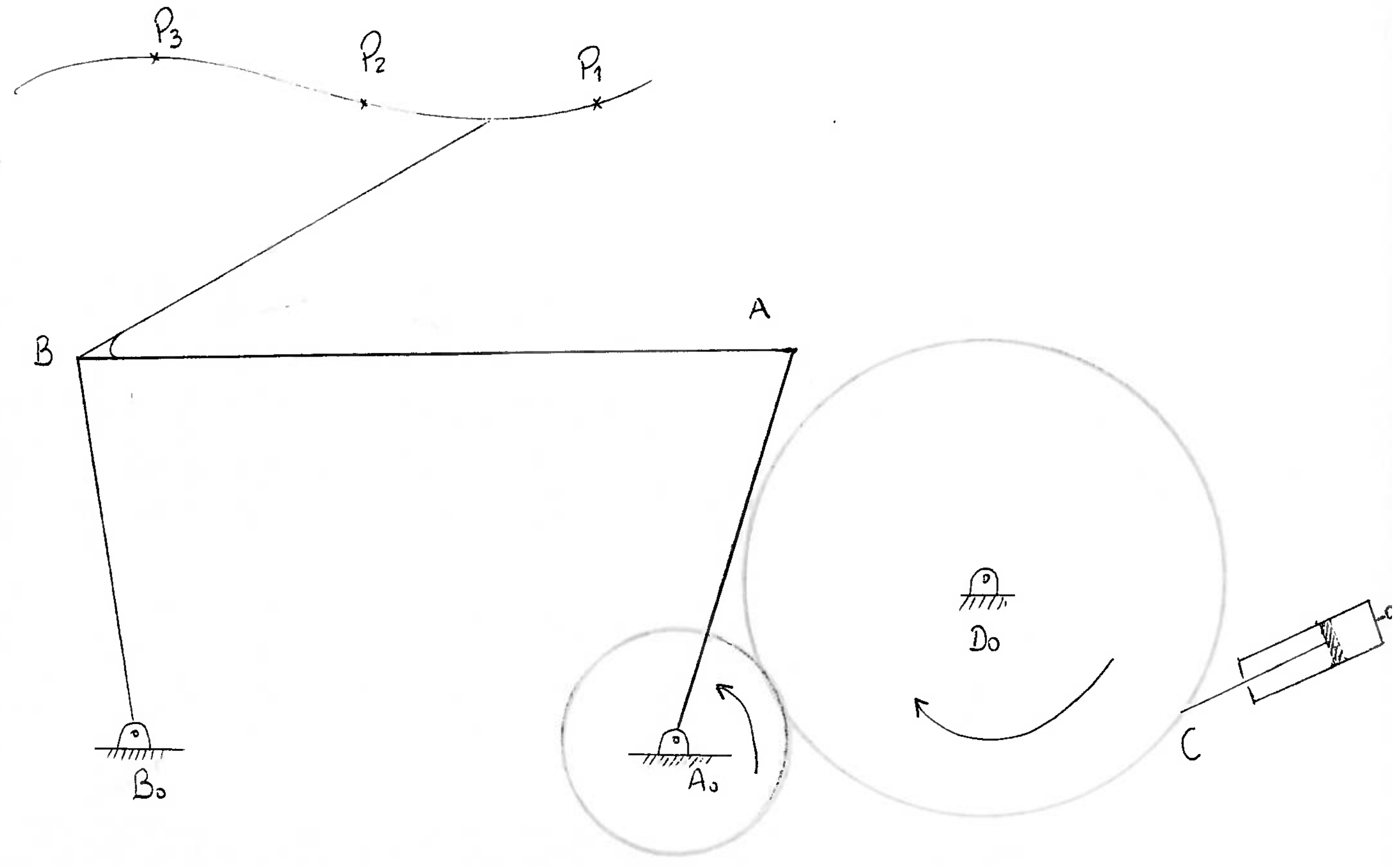


Fig. 7.2.4 – Alavanca $\overline{\mathbf{AA}_0}$ deslocando-se no sentido anti-horário

Deve-se ainda acrescentar que o procedimento seguido para a síntese do mecanismo acionador, garante que o ângulo η esteja sempre definido num intervalo angular recomendado pela literatura, ou seja,

$$45^\circ \leq \eta \leq 135^\circ$$

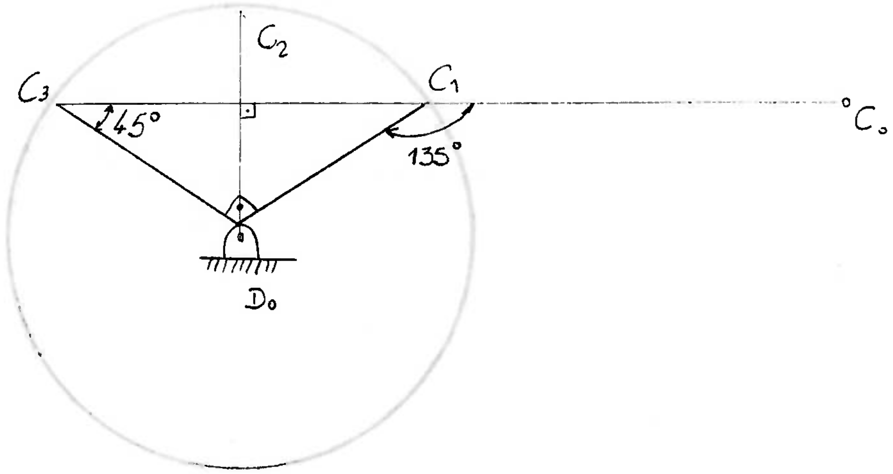


Fig. 7.2.5 – Indicação dos valores extremos de η

Com relação à função-objetivo, deve-se mencionar que, inicialmente, sua expressão era a seguinte:

$$\mathbf{F(x)} = \sum_{i=1}^{n}\left(\mathbf{w}_1\left(\mu_i - \frac{\pi}{2}\right)^2 + \mathbf{w}_2\left(\eta_i - \frac{\pi}{2}\right)^2\right)^{\frac{1}{2}}$$

sendo μ_i e η_i os ângulos de transmissão dos mecanismos posicionador e acionador, respectivamente, numa posição qualquer "i". O vetor **x** de projeto incluía as variáveis A_0, B_0, P_1, P_2, P_3 e C_0.

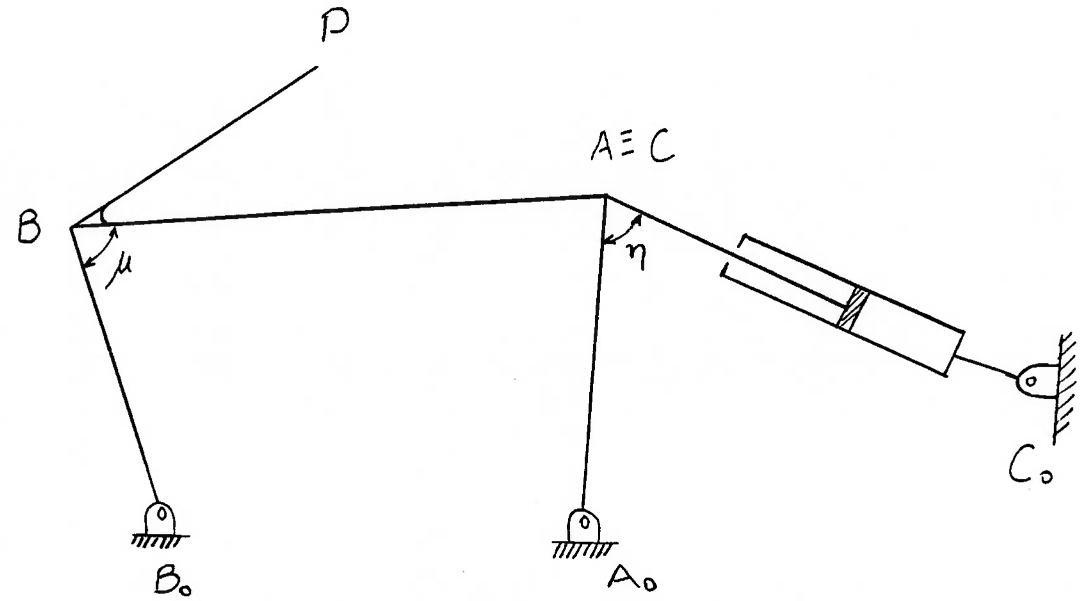


Fig. 7.2.6 – Ângulos de transmissão μ_i e η_i

Os escalares w_1 e w_2 eram pesos atribuídos aos desvios do ângulo de transmissão μ e η. Desta forma, se o mecanismo completo otimizado caracterizava-se, por exemplo, por ângulos μ excelentes e ângulos η ruins, poderia-se alterar os valores de w_1 e w_2 de modo a aumentar a importância do ângulo η sobre μ para uma dada situação.

Contudo, abandonou-se essa abordagem por ela não prever e resolver situações em que o acionador prismático, no seu ciclo de movimento, apresentava alternância no seu deslocamento, como, por exemplo, distenção-contração-distenção.

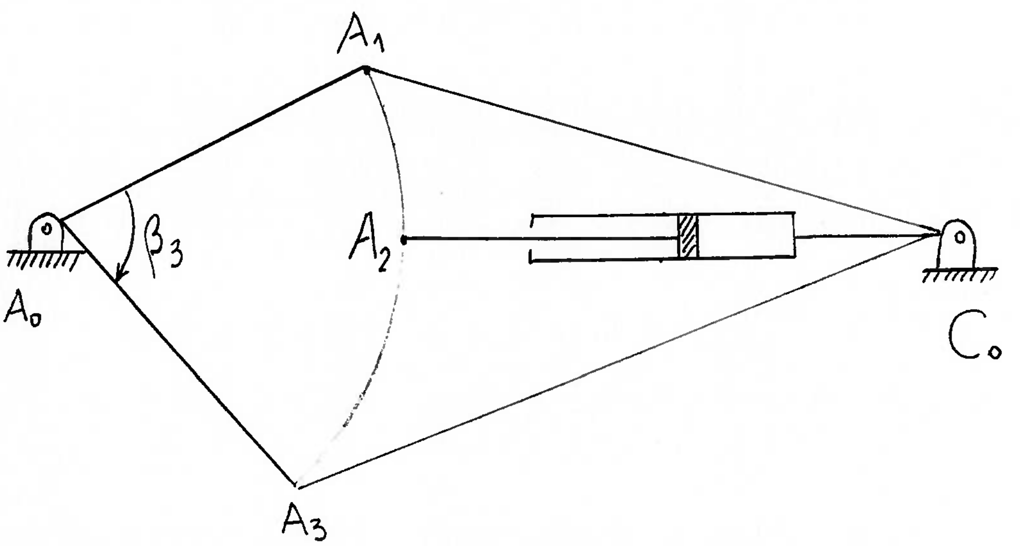


Fig. 7.2.7 – Alternância no deslocamento do acionador prismático

7.3 – Descrição das ocorrências durante a resolução dos casos exemplo

Com relação ao primeiro caso estudado, o mecanismo inicial apresentava-se bem comportado como pode ser visto pelos valores da função-objetivo que inicialmente era de 2,84 e houve uma diminuição de apenas 46,5%, para 1,52, o que nos diz que com relação à variável ângulo de transmissão, não foi possível obter uma melhora expressiva como nos dois casos seguintes.

O segundo exemplo de aplicação, que baseia-se num problema proposto por PERTERSON et al. (1988), apresentou uma dificuldade adicional porque, dependendo dos valores iniciais para as coordenadas das articulações fixas, A_o e B_o, o mecanismo sintetizado travava durante sua movimentação. Para que fosse superada essa dificuldade realizou-se um estudo preliminar para avaliação do intervalo de travamento. Desta forma, observou-se melhor os valores dos ângulos β_2 e β_3 das duas díadas do mecanismo, considerado como estimativa inicial. Após a realização da otimização, é possível verificar um desvio muito menor do ângulo μ em relação a 90° (conforme a fig. 6.2.11), sendo que o valor da função-objetivo teve uma redução de 93,5%, passando de 6,74 para 0,44.

Com relação à síntese do mecanismo acionador, observou-se que dependendo da estimativa inicial dos ângulos ν_1 e ν_2 a convergência se dava numa ou noutra solução. Este comportamento é razoável porque, geometricamente, pode-se observar que o problema definido na seção 3.3, possui mais de uma solução.

Com relação ao terceiro exemplo de aplicação, deve-se mencionar que se baseia em um problema real, ou seja, projetar um mecanismo que posicione a caçamba basculante de um caminhão, sendo que o centro de gravidade da caçamba deve descrever uma trajetória retilínea e horizontal.

No decorrer da solução deste exemplo, surgiu a questão da especificação da inclinação da caçamba, que seria a definição do ângulo α. Iniciou-se com os valores $\alpha_2 = 0°$ e $\alpha_3 = 20°$, resultando em dimensões das barras do mecanismo sintetizado que não eram compatíveis com as do chassis, portanto estes dados foram desconsiderados.

Como segunda estimativa, atribuiu-se os valores $\alpha_2 = 20°$ e $\alpha_3 = 30°$. Um ponto positivo, constatado, foi que as dimensões das barras ficaram compatíveis com as do chassis, além disso, houve uma redução significativa do valor da função-objetivo, passando de 5,13 para 0,13, o que significa uma potência menor para acionar todo o conjunto.

Numa terceira estimativa realizada, utilizou-se os valores de $\alpha_2 = 20°$ e $\alpha_3 = 35°$. Porém nessa situação, apesar de ter havido compatibilidade das dimensões, tivemos um aumento significativo no valor da função-objetivo, sendo que seu valor otimizado foi maior que 5,13 (valor sem otimização no caso da segunda estimativa), o que pode significar um sobre dimensionamento do posicionador prismático. Com valores do ângulo α_3 maiores não foi possível calcular um mecanismo que atendesse às restrições de projeto.

Capítulo 8 – Conclusões e Temas de Trabalhos Futuros

8.1 – Conclusões

1 – Apresentou-se neste trabalho uma metodologia para a síntese cinemática para mecanismos articulados planos, movidos por acionadores prismáticos. O procedimento de solução prevê a decomposição do mecanismo completo em dois outros sub-sistemas: mecanismo posicionador e mecanismo acionador.

2 – Desenvolveram-se modelos matemáticos e computacionais para realização da síntese cinemática tanto do mecanismo posicionador quanto do acionador.

3 – A síntese cinemática considerada neste trabalho prevê a otimização de duas variáveis do mecanismo do mecanismo completo: seus ângulos de transmissão.

4 – Para otimização da variável ângulo de transmissão do mecanismo posicionador, acoplou-se o seu modelo computacional a uma rotina de otimização baseada no método do Gradiente Reduzido Generalizado.

5 – Para otimização da variável ângulo de transmissão do mecanismo acionador, desenvolveu-se uma expressão analítica em função de determinados parâmetros do mecanismo que garante que esta variável se mantenha dentro do intervalo recomendado pela literatura.

6 – A eficácia da metodologia desenvolvida foi verificada através de simulações realizadas com os modelos computacionais em três exemplos, observou-se a otimização do dois ângulos de transmissão.

8.2 – Trabalhos futuros

Como temas de pesquisa para os próximos anos pretendemos iniciar a utilização de inteligência artificial (redes neurais, algoritmo genético, etc.) para a obtenção de síntese e otimização de mecanismos mais complexos tais como:

1- Mecanismos espaciais, como plataforma de Stewart.

2- E o comportamento cineto-elasto-dinâmico tanto de mecanismos plano quanto dos espaciais.

Capítulo 9- Apêndices

9.1- Introdução

Neste capítulo iremos apresentar a notação polar complexa utilizada e a listagem dos programas computacionais desenvolvidos.

9.2 – Notação vetorial polar complexa

A notação polar complexa é útil na análise de mecanismos planos. O método é baseado na representação polar de um número complexo da seguinte forma:

$$e^{i\theta} = \cos\theta + i\,\mathrm{sen}\theta$$

Assim, os componentes x e y de um vetor **a** podem ser encontrados da seguinte forma:

$$\mathbf{a} = ae^{i\theta}$$

$$\mathbf{a} = a(\cos\theta + i\ \mathrm{sen}\theta)$$

$$\mathbf{a} = a_x + a_y$$

O produto do vetor unitário $e^{i\varphi}$ vezes o vetor $ae^{i\theta}$ é igual a $e^{i\varphi}(ae^{i\theta}) = a(e^{i(\theta+\varphi)})$. Desta forma notamos que o vetor $\mathbf{a} = ae^{i\theta}$ roda de um ângulo φ quando ele é multiplicado pelo vetor unitário $e^{i\theta}$.

A multiplicação do vetor $\mathbf{a} = ae^{i\theta}$ pelo vetor complexo $i = \sqrt{-1}$ é equivalente a rodar o vetor de $\pi/2$ radianos no sentido positivo de θ, isto é:

$$ie^{i\theta} = i(\cos\theta + i\,\mathrm{sen}\theta) = i\cos\theta - \mathrm{sen}\theta$$

$$ie^{i\theta} = e^{i(\theta + \pi/2)}$$

Da mesma forma:

$$i(ie^{i\theta}) = i^2e^{i\theta} = -\,e^{i\theta} = e^{i(\theta+\pi)}$$

As seguintes identidades trigonométricas também são importantes:

$$\cos\theta = \frac{e^{i\theta} + e^{-i\theta}}{2}$$

$$\operatorname{sen}\theta = -i\frac{e^{i\theta} - e^{-i\theta}}{2}$$

9.3 – Análise cinemática usando a notação polar complexa

Em um quadrilátero de quatro barras a análise das posições é baseada nas equações formuladas a partir de suas barras. O vetor do polígono por ser formado de duas maneiras: (1) de maneira contínua, do começo ao fim, ou (2) de forma segmentada, por duas metades ou díadas. Segundo SUH e RADCLIFFE a segunda notação é melhor quando é desejável ter a rotação de ambas as díadas, entrada e saída, em torno de centros fixos.

A equação de posicionamento do mecanismo de quatro barras é baseada vetores de trajetória independentes que descrevem a posição do ponto B, como mostra a figura 9.3.1:

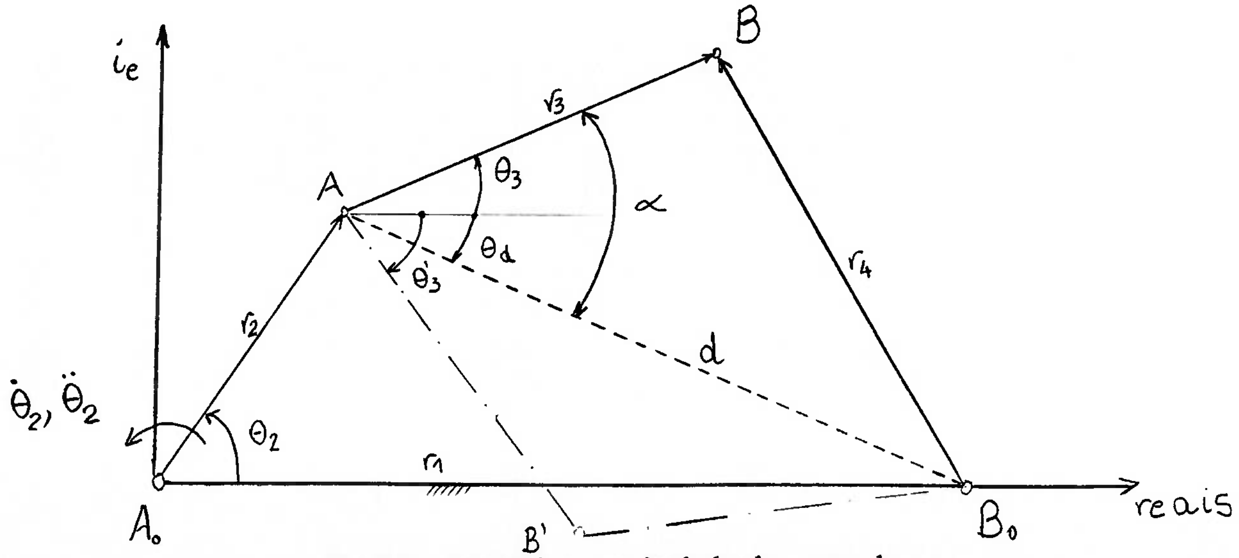


Fig 9.3.1 Mecanismo articulado de quatro barras

$$\mathbf{r}_B = r_2 e^{i\theta}{}_2 + r_3 e^{i\theta}{}_3 = \mathbf{r}_1 + r_4 e^{i\theta}{}_4$$

Assumindo que sejam conhecidos o vetor $\mathbf{r}_1$, e as barras r_2, r_3 e r_4 além do ângulo θ_2, faz-se a análise de posicionamento da seguinte maneira: nós primeiro determinamos o comprimento da diagonal d,

$$\mathbf{d}e^{i\theta_d} + \mathbf{r}_2 e^{i\theta_2} = \mathbf{r}_1 \qquad (9.3.1)$$

Podemos eliminar o ângulo θ_d isolando $de^{i\theta}{}_d$ e multiplicando cada lado da equação pelo seu complexo conjugado.

$$(de^{i\theta_d})(de^{-i\theta_d}) = (r_1 - r_2 e^{i\theta_2})(r_1 - r_2 e^{-i\theta_2})$$

O que nos fornece:

$$d^2 = r_1{}^2 + r_2^2 - r_1 r_2 (e^{i\theta_2} + e^{-i\theta_2})$$

$$d = \sqrt{r_1{}^2 + r_2{}^2 - 2r_1 r_2 \cos\theta_2}$$

Resolvendo a equação 9.3.1 nas suas partes real e imaginária teremos:

$$r_2\cos\theta_2 = r_1 - d\cos\theta_d$$

$$r_2\text{sen}\theta_2 = 0 - d\,\text{sen}\theta_d$$

O que nos dá:

$$\text{sen}\,\theta_d = -\frac{r_2\,\text{sen}\,\theta_2}{d}$$

$$\cos\theta_d = -\frac{r_1 - r_2\cos\theta_2}{d}$$

O ângulo θ_3 pode ser encontrado através da equação:

$$r_3 e^{i\theta_3} = de^{i\theta_d} + r_4 e^{i\theta_4}$$

$$\cos(\theta_3 - \theta_d) = \frac{r_3^2 - d^2 - r_4^2}{2dr_3}$$

9.4 – Listagem dos programas desenvolvidos

9.4.1 – Programa "Main"

```
PROGRAM MAINI

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

      Common / Optdat/ D(750)

      Common / Pari/ xtol, gtol, eps, epsls, epsbd, ipr, maxm, idata,
     'lfunct, ne, lne, ni, lni, ncon, nscale, isearc, maxnew,
     ' idif, nout, bex3, R2

      Double Precision Valor, X(10), xmax(10), xmin(10), scale(10)
     ' , con(1),A1,A2,A3,A4,A5,A6,B1,B2,B3,B4,B5,B6,Cxo,Cyo,R,ra
     ' ,yni1,yni2


      open(9, file = 'sol.doc',  status = 'old')
      open(13,file = 'dadI.doc',status = 'old')
      open(14, file = 'dad1I.doc',status = 'old')
      open(16,file = 'dad2I.doc',status = 'old')

c *******  fator de escala para a funcao objetivo F *******

      fscale =1

c ***** Indica em qual unidade serao impressos os parametros
c ***** de saida
      nout = 9

c ******* Fatores de escala serao calculados pelo OPT-3 ********
      nscale = 2

c ****** Impressao do resumo de saidas inicial e final *******
      idata = 1

c ****** Imprime um resumo das iteracoes no final de cada iteracao
      ipr =0

c ****** Parametro de convergencia das variaveis X (a cada
c ****** iteracao e checado)
      xtol = 0.1E-8

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

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

c ****** Parametro que define o limite em que uma variavel
```

```
c ****** e igual a uma de suas fronteiras
      epsbd = 0.1E-7

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

c ****** Parametro de incremento da derivada ********
      eps = 0.1E-3

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

c ******* No. de variaveis de projeto   *********
      n = 10

c ****** No. total de restricoes ********
      ncon = 0

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

c ****** No. de restricoes de desigualdade   ********
      ni = 0

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

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

c ********* ????????
      ibd = 0

c ****** Funcao objetivo nao-linear *******
      Ifunct = 0

c ****** No. maximo de iteracoes *******
      maxm = 40

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

c ******  Tamanho do problema *******
      isize = 750


      Rewind (13)
      READ(13,*) A1,A2,A3,A4
      X(1) = A1
      X(2) = A2
      X(3) = A3
      X(4) = A4


      Rewind(14)
      READ(14,*) B1,B2,B3,B4,B5,B6
      X(5) = B1
      X(6) = B2
      X(7) = B3
      X(8) = B4
      X(9) = B5
```

```
      X(10) = B6

c ***** Limite maximo de cada variavel *********
      data xmax/ 7, 2, 2, 2,
     ' 1.2, 1.2, 2.2, .7, 3.2, 1.7/

c ****** Limite minimo de cada variavel *********
      data xmin/ 3, -2, -2, -2,
     ' .8, .8, 1.8, 0.3, 2.8, 1.3/



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

      func = F(x)

      Valor = F(X)

      rewind(40)
      read(40,*) Axo,Ayo,R,ra,Cxo,Cyo,bex2


      call  naolin(Axo,Ayo,R,ra,Cxo,Cyo,bex2)

      rewind(41)
      read(41,*) yNI1,yNI2

      print *, 'NI1 =',yNI1
      print *, 'NI2 =',yNI2



      write(*,*) '******************************'
      write(*,*) 'NI(1) =',yNI1
      write(*,*) 'NI(2) =',yNI2
      write(*,*) '******************************'


      print *, 'A funcao vale: ',valor


      STOP
      END


      subroutine Const (x, con )

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

      Dimension x(1), con(1)

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

      nce = nce + 1

      return
      end
```

```
SUBROUTINE GRAU (th)

DOUBLE PRECISION TH, PI

PI = 4.0*ATAN(1.0)

TH = 180*TH/PI

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 POLAR (VX,VY,V,TH)

DOUBLE PRECISION VX,VY,V,TH


TH = ATAN2(VY,VX)

V = SQRT(VX**2 + VY**2)

RETURN
END



SUBROUTINE RAD(TH)

DOUBLE PRECISION TH, PI

PI = 4.0*ATAN(1.0)

TH = PI*TH/180

RETURN
END


SUBROUTINE CHECAN4 (A)

DOUBLE PRECISION A,PI

PI = 4.0*ATAN(1.0)
```

```
      IF (A.GT.2*PI.OR.A.LT.-2*PI) THEN
         A = A - INT(A/(2*PI))*2*PI
      ENDIF

      IF (A.GT.PI) THEN
         A = A - 2*PI

      ENDIF

      IF (A.LT.-2*PI) THEN
         A = A + 2*PI

      ENDIF

      RETURN
      END



      SUBROUTINE TRIAN(X1,X2,X3,AG1,AG2,AG3,B2,B2T,B3,B3T,J)

      DOUBLE PRECISION CG,SG,G,
     '              X1,X2,X3,AG1,AG2,AG3,
     '              B2,B2T,B3,B3T,
     '              F,CF,SF

C *******  CALCULO DE BET #2, #3 ********

      CG = ( X1**2 + X3**2 - X2**2)/(2.0*X1*X3)

      IF (CG.GT.1.OR.CG.LT.-1) THEN
         J = 1
         GOTO 50
      ENDIF

      SG = SQRT( 1 - CG**2)

      G = ATg(SG,CG)

      B2 = AG3 + G - AG1
      B2T = AG3 - G - AG1

      CF = (X3**2 + X2**2 - X1**2)/(2.0*X2*X3)

      IF (CG.GT.1.OR.CF.LT.-1) THEN
         J = 1
         GOTO 50
      ENDIF

      SF = SQRT( 1 - CF**2)
      F = ATg (SF,CF)

      B3 = AG3 - F - AG2
      B3T = AG3 + F - AG2

      CALL CHECAN4(B2)
      CALL CHECAN4 (B2T)
      CALL CHECAN4 (B3)
      CALL CHECAN4(B3T)
```

```
50    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



      Subroutine Sistema( alf2, alf3, bet2, bet3, del2x,
     1                    del2y, del3x, del3y, wx, wy,
     1                    zx, zy )

      double precision ax, ay, bx, by, cx, cy, dx, dy,
     1     bet2, bet3, alf2, alf3,
     1     del2x, del2y, del3x, del3y,
     1     DDx, DDy, DD1x, DD1y, DD2x, DD2y,
     1     mDD, mDD1, mDD2, argDD, argDD1, argDD2,
     1     wx, wy, zx, zy,
     1     w, z, argw, argz


      ax = cos( bet2 ) - 1

      ay = sin( bet2 )

      bx = cos( alf2 ) - 1

      by = sin( alf2 )

      cx = cos( bet3 ) - 1
```

```
      cy = sin( bet3 )

      dx = cos( alf3 ) - 1

      dy = sin( alf3 )

      DDx = ax * dx - ay * dy - bx * cx + by * cy

      DDy = ax * dy + ay * dx - by * cx - bx * cy

      DD1x = del2x * dx - del2y * dy - del3x * bx + del3y * by

      DD1y = del2x * dy + del2y * dx - del3x * by - del3y * bx

      DD2x = del3x * ax - del3y * ay - del2x * cx + del2y * cy

      DD2y = del3x * ay + del3y * ax - del2y * cx - del2x * cy

      call Polar ( DDx, DDy, mDD, argDD )
      call Polar ( DD1x, DD1y, mDD1, argDD1 )
      call Polar ( DD2x, DD2y, mDD2, argDD2 )

      w = mDD1 / mDD

      argw = argDD1 - argDD

      z = mDD2 / mDD

      argz = argDD2 - argDD

      call checan4( argw )

      call checan4( argz )

      call Cartes( w, argw, wx, wy )

      call Cartes( z, argz, zx, zy )


      return
      end
```

9.4.2 – Função F(x)

```
FUNCTION F(X)

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

      double precision xm, arx, ym, ary, tm, art,x(10),
     1           w(2,2), mw(2), argw(2),modr(2),argr(2),
     1           z(2,2), mz(2), argz(2), R(2,2),pi,
     1           xt(2), y(2) , t(2),theta2,otim,pa1,pa2,
     1           d(2:3,1:2), md(2:3), argd(2:3),pa3,pa4,
     1           bet(2:3,1:2,1:2), alf(2:3),otimo,
     1           Ao(2), Bo(2),Co(2), A(2,4), B(2,4), P(3,2)
```

```
      open(18, file = 'saiI.doc',status ='old')


      pi = 4.0*atan(1.0)

      Ao(1) = x(1)
      Ao(2) = x(2)
      Bo(1) = x(3)
      Bo(2) = x(4)


      Rewind(16)
      READ(16,*) C1,C2
      alf(2) = C1
      alf(3) = C2

      P(1,1) = x(5)
      P(1,2) = x(6)
      P(2,1) = x(7)
      P(2,2) = x(8)
      P(3,1) = x(9)
      P(3,2) = x(10)

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) + Ao(1)

      R(1,2) = - P(1,2) + Ao(2)

      call Polar( R(1,1),R(1,2),modr(1),argr(1) )

      R(2,1) = - P(1,1) + Bo(1)

      R(2,2) = - P(1,2) + Bo(2)

      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 *******************************************************************

      k4 = 2


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

      if( k4.eq.1 ) then

c        write(7,*)
c        write(7,*)'  Nao e possivel se determinar um mecanismo '
c        write(7,*)'   que atenda a essas especificacoes de projeto '
c        write(*,*)
c        write(*,*)'  Nao e possivel se determinar um mecanismo '
c        write(*,*)'   que atenda a essas especificacoes de projeto '
c        write(*,*)
         goto 500

      endif


   20 continue

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


      k3 = 0

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

        k3 = k3 + 1


      write(18,*)
      write(18,*)'  Solucao : ', k3
      write(18,*)

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

      write(18,*)'    1a. diada  : '
      write(18,*)
      write(18,*)'  beta 2 =   ', bet( 2, 1, k1 )
      write(18,*)
      write(18,*)'  beta 3 =   ', bet( 3, 1, k1 )
      write(18,*)
```

```
      call Rad( bet( 2, 1, k1 ) )
      call Rad( bet( 3, 1, k1 ) )


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

      bex3 = bet(3,1,k1)

      call sistema( alf(2), alf(3), bet(2,1,k1), bet(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( bet( 2, 2, k2 ) )
      call Grau( bet( 3, 2, k2 ) )

      write(18,*)'    2a. diada   : '
      write(18,*)
      write(18,*)'  beta 2 =   ', bet( 2, 2, k2 )
      write(18,*)
      write(18,*)'  beta 3 =   ', bet( 3, 2, k2 )
      write(18,*)
      write(18,*)

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

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

      bex3 = bet(3,2,k2)

      call sistema( alf(2), alf(3), bet(2,2,k2), bet(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,k3) = P(1,i) - z(1,i)

      B(i,k3) = P(1,i) - z(2,i)

   40 continue

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



      write(18,*) ' Aox = ',Ao(1),'    Aoy = ',Ao(2)
```

```
      write(18,*) ' Box = ',Bo(1),'    Boy = ',Bo(2)

      write(18,*) '  Ax = ',A(1,k3),'    Ay = ',A(2,k3)

      write(18,*) '  Bx = ',B(1,k3),'    By = ',B(2,k3)

      write(18,*) ' Px = ',P(1,1),'     Py = ',P(1,2)

pa3 = bet(2,k1,k2)*180/pi
      pa4 = bet(3,k1,k2) *180/pi


      if (A(1,k3) .lt. 500 .and. A(1,k3) .gt. -500 .and. A(2,k3) .lt.
500
     '.and. A(2,k3) .gt. -500 .and. B(1,k3) .lt. 500 .and. B(1,k3)
.gt.
     ' -500 .and. B(2,k3) .lt. 500 .and. B(2,k3) .gt. -500) then

       call calculo(Ao(1),Ao(2),Bo(1),Bo(2),A(1,k3),A(2,k3),
     ' B(1,k3),B(2,k3), k3,bet(2,1,k1),bet(3,1,k1),w(1,1),w(1,2),
     ' p(1,1),p(1,2),z(1,1),z(1,2), otim)


      endif


   30 continue
   25 continue


      close(18)

  500

       F = otim

      return
      end
```

9.4.3 – Sub-rotina "Calculo"

```
subroutine calculo(Axo,Ayo,Bxo,Byo,Ax1,Ay1,Bx1,By1,K3,
     ' bex2,bex3,Wx,Wy,Px,Py,Zx,Zy,oti )

      integer k3,i,j,k,t

      Dimension phi(21),betax(3),beta(1:21),c(0:21),z(21)

      double precision
Axo,Ayo,Bxo,Byo,Ax1,Ay1,Bx1,By1,A(1:2,1:4),B(1:2,1:
     '4),oti,R1,R2,R3,R4,D,Dx,Dy,F,pi,thetay2,bex2,bex3,theta1,u,
     '
mi,pa,pa1,pa2,theta2i,pa5,Cxo,Cyo,R5,R6,v,eta(21),add1,add2,R21,
```

```
'Ax3,Ay3,mod,Wx,Wy,Px,Py,Zx,Zy,R,ra,xNI1,xNI2, bet3,theta2


 open(41, file = 'ni.out', status ='old')
 open(40, file = 'arq.out', status ='old')
 open(15, file = 'betaI.doc',status = 'old')
 open(10, file = 'barraCI.out',status = 'old')
 open(11, file = 'miotiI.out',status = 'old')
 open(12, file = 'etaotiI.out',status = 'old')
 open(19, file = 'Co.out', status = 'old')

 pi = 4.0*atan(1.0)

 A(1,k3) = Ax1
 A(2,k3) = Ay1
 B(1,k3) = Bx1
 B(2,k3) = By1

 bet3 = bex3


 if(bex2.gt.0.and.bex3.lt.0) then

    bet3 =bex3 + 2*pi

 endif


 if(bex2.lt.0.and.bex3.gt.0) then

    bet3 =bex3 - 2*pi

 endif


 theta2i = atan2((Ay1-Ayo),(Ax1-Axo))

 theta1 = atan2((Byo - Ayo),(Bxo - Axo))

 c(0) = 10

 oti = 0

 j = 2

 i = 0

 R1 = sqrt((Bxo - Axo)**2 + (Byo - Ayo)**2)

 R2 = sqrt((Ax1 - Axo)**2 + (Ay1 - Ayo)**2)

 R3 = sqrt((Bx1 - Ax1)**2 + (By1 - Ay1)**2)

 R4 = sqrt((Bx1 - Bxo)**2 + (By1 - Byo)**2)

 R5 = sqrt((Ax1 - Cxo)**2 + (Ay1 - Cyo)**2)

 R6 = sqrt((Axo - Cxo)**2 + (Ayo - Cyo)**2)

 do 100 i= 1,21,1
```

```
      Dx = R1 * cos(theta1) - R2 * cos(theta2)
      Dy = R1 * sin(theta1) - R2 * sin(theta2)

      D = sqrt(Dx**2 + Dy**2)

      u =(R3**2 + R4**2 - D**2)/(2*R3*R4)
 if(u.ge.1.or.u.le.-1) then

         p = 100
         goto 200

       endif

       phi(i)  = acos(u)


      write(11,*) 'mi(',i,') =',(phi(i) )*180/pi ,';'


       p = (phi(i) - pi/2)**2


       theta2=  theta2i +add2*i


       Ax1 = R2 * cos(theta2) + Axo
       Ay1 = R2 * sin(theta2) + Ayo


       R5 = sqrt((Ax1 - Cxo)**2 + (Ay1 - Cyo)**2)
       R2 = sqrt((Ax1 - Axo)**2 + (Ay1 - Ayo)**2)


 200  oti = oti + p

 100  continue

      print *, 'funcao =',oti

c************  dados para o sistema nao-linearp*********
      ra = r2/3
      R  = (2*ra*bex3)/pi
      Cxo = 6.248
      Cyo = 1.008
c**********************************************

      rewind(19)

      write(19,*) Axo
      write(19,*) Ayo
      write(19,*) R
      write(19,*) ra
      write(19,*) Cxo
      write(19,*) Cyo
      write(19,*) bex2

      return
      end
```

```
c****************************************************************

      subroutine naolin (A1,A2,xR,xra,C1,C2,bet2)

      dimension   F(2,3000), xjaco(2,2), del(2,3000)

      double precision pi, R, ra, Axo,Ayo,Cxo,Cyo, xnitol, t ,
     '     A1,A2,xR,xra,xni1,xni2,C1,C2,bet2,  XNI(2,3000)

      integer K

      open(50, file = 'NI1.doc', status = 'old')
      open(51, file = 'NI2.doc', status = 'old')
      open(19, file = 'Co.out', status = 'old')

      pi = 4.0*atan(1.0)

      xnitol = 1E-5

      write(*,*) 'Beta2 - naolin =',bet2*180/pi
      write(*,*) 'Axo =',A1
      write(*,*) 'Ayo =',A2
      write(*,*) 'Cxo =',C1
      write(*,*) 'Cyo =',C2
      write(*,*) 'R =',xR
      write(*,*) 'ra =',xra

      Axo = A1
      Ayo = A2
      R = xR
      ra = xra
      Cxo = C1
      Cyo = C2


c ******** inicio iteracoes *********
      xNI(1,1) = 20
      xNI(2,1) = 65
c ***********************************

      XNI(1,1) = xNI(1,1)*pi/180
      XNI(2,1) = xNI(2,1)*pi/180


      do 10 k = 1,3000

         write(50,*) 'NI(1,',k,') =',XNI(1,k)*180/pi
         write(50,*) 'NI(2,',k,') =',XNI(2,k)*180/pi

       if(bet2.lt.0) then

         t = 2

         F(1,k) = R*1.414/2*(3*cos(XNI(2,k)-pi/2)+cos(XNI(2,k)))+
     '          (R + ra) * cos(XNI(1,k)) + Axo - Cxo
         F(2,k) = R*1.414/2*(3*sin(XNI(2,k)-pi/2)+sin(XNI(2,k)))+
     '          (R + ra) * sin(XNI(1,k)) + Ayo - Cyo

         xjaco(1,1) = (R+ra) * (-sin(XNI(1,k)))
         xjaco(1,2) = R*1.414/2*(-3*sin(XNI(2,k)-pi/2)-sin(XNI(2,k)))
         xjaco(2,1) = (R+ra) * cos(XNI(1,k))
```

```
      xjaco(2,2) = R*1.414/2*(3*cos(XNI(2,k)-pi/2)+cos(XNI(2,k)))

      else

      t = 2

      F(1,k) = R*1.414/2*(3*cos(XNI(2,k)+pi/2)+cos(XNI(2,k)))+
     '      (R + ra) * cos(XNI(1,k)) + Axo - Cxo
      F(2,k) = R*1.414/2*(3*sin(XNI(2,k)+pi/2)+sin(XNI(2,k)))+
     '      (R + ra) * sin(XNI(1,k)) + Ayo - Cyo

      xjaco(1,1) = (R+ra) * (-sin(XNI(1,k)))
      xjaco(1,2) = R*1.414/2*(-3*sin(XNI(2,k)+pi/2)-sin(XNI(2,k)))
      xjaco(2,1) = (R+ra) * cos(XNI(1,k))
      xjaco(2,2) = R*1.414/2*(3*cos(XNI(2,k)+pi/2)+cos(XNI(2,k)))

    endif


    del(1,k) = -(xjaco(2,2) * F(1,k) - xjaco(1,2) * F(2,k))/
     '         (xjaco(1,1)*xjaco(2,2) - xjaco(2,1)*xjaco(1,2))


   del(2,K) = - (xjaco(1,1)*F(2,K) - xjaco(2,1)*F(1,K))/
     '         (xjaco(1,1)*xjaco(2,2) - xjaco(2,1)*xjaco(1,2))


   del(1,k) = del(1,k)/200
   del(2,k) = del(2,k)/200


   delNI = (del(1,k)**2 + del(2,k)**2)**.5

   write(50,*) 'NI(1,',k,') =',xni(1,k)*180/pi
   write(50,*) 'NI(2,',k,') =',xni(2,k)*180/pi
   write(50,*) 'delNI =',delNI
   write(50,*) 'delta(1',k,') =',del(1,k)
   write(50,*) 'delta(2',k,') =',del(2,k)
   write(50,*) 'F(1,',k,') = ',F(1,k)
   write(50,*) 'F(2,',k,') =',F(2,k)
   write(50,*)  'jaco 1,1 =',xjaco(1,1)
   write(50,*)  'jaco 1,2 =',xjaco(1,2)
   write(50,*)  'jaco 2,1 =',xjaco(2,1)
   write(50,*)  'jaco 2,2 =',xjaco(2,2)
   write(50,*) '**************************'

   if( delNI.le.xnitol) then
     goto 50
   else

      XNI(1,k+1) = XNI(1,k) + del(1,K)
      xNI(2,k+1) = XNI(2,k) + del(2,K)

      goto 10
   endif

10 continue


50 write(*,*) '******************************'
```

```
write(*,*) 'NI(1,',k,') =',xNI(1,k)*180/pi
write(*,*) 'NI(2,',k,') =',xNI(2,k)*180/pi
write(*,*) '*****************************'


close (50)
close (51)
close (19)

return
end
```

Capítulo 10 – Referência Bibliográfica

1. Erdman, A. G. e Gustafson, J. E., Lincages: Linkage Interactive Computer Analysis and Graphically Enhenced Syntesis Package, ASME Paper Nº 77-DET-5, pp1-9, 1977.

2. Erdman, A. G. e Sandor, G. N., Mechanism Design: Analysis and Synthesis, vol. 1, Prentice-Hall, 1984.

3. Erdman, A. G., Computer-Aided Mechanism Design: Now and the Future, Transaction of the ASME, vol. 117 , July 1995.

4. Freudenstein, F. e Sandor, G. N., Syntesis of Path-gerantig Mechanism by Means of a Programmed Digital Computer, Journal of Engineering for Industry, ser.B; vol. 81, Nº 2, pp 159-168, Maio de 1959.

5. 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.

6. Gupta, K. C., Design of Four-Bar Function Generation With Min-Max Transmition Angle, Journal for Engineering for Industry, pp 360-366, Maio de 1977.

7. Hartenberg, R. S. e Denavit, J., Kinematic Syntesis of Linkages, McGraw Hill, 1964.

8. Himmelbau, D. M., Applied Nonlinear Programming, McGraw Hill, 1972

9. Kaufman, R. E., KINSYN Phase II : A Human-Engineering Computer System for Kinematic Design and a new Last-Square Synthesis Operator, Mechanism and Machine Design, vol. 8, Nº 4, 1974.

10. Krishnamurty, S. e Turcic, D. A., Optimal Syntesis of Mechanisms Using Nonlinear Goal Programming Techniques, Mechanism and Machine Theory, vol. 27, Nº 05 , pp 599-612, 1992

11. Lima, C. S., Optimun Dynamic Synthesis of Single Degree of Freedom Planar Linkages, tese de doutorado apresentado na Universidade da Florida, Maio de 1985.

12. Midha; A. e Zhao, Z. L., Synthesis of Planar Linkage Via Loop Closure and Nonlinear Equations Solution, Proceeding of the 8th OSU Aplied Mechanism 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 Exemple Using the LINCAGE-4 Program, Computers in Engineering, San Francisco CA, pp 91-96, 1988.

14. Ragsdell; K. M. e Gabriele; G. A., The Generalized Reduced Gradient Method : A Reliabel Tool for Optimal Design, Journal of Engineering for Industry, vol. 99; pp 394-400, 1977.

15. 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 Mechanism, Transmissions, and Automation in Design, vol. 110, pp 263-268, Setembro de 1988.

16. Root, R. R. e Ragsdell, K. M., A survey of Optimization Methods Applied for the Design of Mechanism, Journal of Engineering and Industry, pp 1036-1041, Agosto de 1976

17. Salem, N. H. e Manoochehri, S., AutoCAD Based Computer-Aided Mechanism Design System, paper presented at the conference Cam, Gears, Robot and Mechanism Design, Chicago, IL, pp 331-339, 1990.

18. Sandor, G. N. e Erdman, A. G., Advanced Mechanism Design : Analysis and Synthesis, vol. 2, Prentice Hall, 1984.

19. Shett; P. N., e Uicker, J. J., Jr. IMP (Integrated Mechanism Program) : A Computer Aid Design System for Mechanism and Linkages, Journal for Engineering for Industry, setembro de 1971.

20. Shigley, J. E. e Uicker, J. J., Theory of Machines and Mechanism, Internatinal Student Edition, McGraw Hill, Singapure, 1981.

21. Suh, C. H. e Radcliffe, C. W., Kinematic and Machine Design, John Wiley & Sons, New York, 1978.

22. Tao, D. C., Applied Linkages Syntesis, Addison-Wesley, 1964

23. Todorov, T. S., Synthesis of Watt's Six-Link Mechanism for manipulation Action in Relative Space, Mechanism and Machine Theory, vol. 32, issue - 5, july, 1997

24. Whalen, P. V. e Midha, A., User-Frendly Interactive Computer-Aided Design of Four-Bar Based Linkages, Computers in Engineering, West Lafayet IN, pp 154-161, 1984.