

ARDUINO FRANCESCO LAURICELLA

**MECANISMO E ESTRATÉGIA DE CONTROLE DE UMA PIPETADORA
MATRICIAL PARA A DETERMINAÇÃO DE EXPRESSÃO GENÉTICA**

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

São Paulo
2004

ARDUINO FRANCESCO LAURICELLA

**MECANISMO E ESTRATÉGIA DE CONTROLE DE UMA PIPETADORA
MATRICIAL PARA A DETERMINAÇÃO DE EXPRESSÃO GENÉTICA**

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

Área de Concentração:
Engenharia Mecânica

Orientador:
Prof. Dr. Raul Gonzalez Lima

São Paulo
2004

RESUMO

Um “microarrayer” é um robô que move agulhas localizadas na denominada “cabeça de impressão” para manipular amostras contendo material biológico com rapidez e precisão, construindo micro-arranjos de amostras depositadas sobre um substrato que posteriormente serão analisados na intenção de descobrir, por exemplo, a seqüência de DNA de determinada molécula ou detectar uma doença genética. É um equipamento de fundamental importância em um laboratório de análise genética. Este trabalho apresenta um mecanismo com elevada rigidez e baixa inércia que permite movimentos rápidos e com precisão. A solução adotada tem dois mecanismos de quatro barras acopladas por articulações, mas com movimentos independentes. Esta solução é inovadora em relação aos “microarrayers” que existem no mercado, os quais movimentam a “cabeça” em eixos cartesianos utilizando guias deslizantes com esferas recirculantes. É simulada também uma estratégia de controle de tempo mínimo com uma única descontinuidade.

ABSTRACT

A “microarrayer” is a robot that moves needles that are located at the called printer head to manipulate samples that contain biological material. It builds microarrays of samples laid over a substrate that will be analyzed to determine, for example, a DNA sequence or detect a genetic disease. The microarrayer is one of the most important equipments in a genetic laboratory. This work presents a mechanism that is stiff and has low inertia allowing fast and accurate movements. The solution has two four links mechanisms coupled by articulations and with independent movements. This is a new design compared to others microarrays which have cartesian movements, using sliding guides with recirculating spheres. Also, time control strategy is simulated, allowing a single control discontinuity.

À minha querida esposa, Christiane,
que tem sido a grande razão e
incentivo na realização deste
trabalho.

AGRADECIMENTOS

Ao meu distinto orientador Professor Doutor Raul Gonzalez Lima, por toda a atenção dispensada, sugestões e idéias transferidas e por todo o trabalho e paciência que teve comigo e, principalmente, pela oportunidade que me foi oferecida em ser o meu orientador.

À Professora Doutora Idagene Cestari, por todo o apoio dispensado, e pelo oferecimento da utilização da estrutura do Instituto do Coração de S. Paulo.

A ambos expresso minha gratidão pela compreensão ao momento delicado pelo qual passei no final deste trabalho.

Aos colegas e amigos de departamento, por terem criado um ambiente agradável e favorável ao desenvolvimento deste trabalho.

ÍNDICE

Lista de figuras	
Lista de tabelas	
Resumo	
1. INTRODUÇÃO	1
1.1. Aplicações do “microarrayer”	1
1.2. Tecnologias para o transporte de amostras	1
1.3. Características e componentes do “microarrayer”	3
1.4. Geometria da agulha ou pino metálico	4
1.5. Material de confecção da agulha ou pino metálico	5
1.6. Densidade das amostras	5
1.7. Problemas nas soluções atuais	6
1.8. O que seria desejável	7
CAPÍTULO 2: OBJETIVOS	8
CAPÍTULO 3: REVISÃO BIBLIOGRÁFICA	9

	ii
3.1. Evolução do desenvolvimento do “microarrayer”	10
3.2. Tecnologia de impressão com agulhas	10
3.3. Velocidade das análises	16
3.4. Obtenção das imagens	17
3.5. Configurações atuais	18
CAPÍTULO 4: METODOLOGIA	19
4.1. Modelamento dinâmico do manipulador	19
4.2. Controle do manipulador	27
4.2.1. Trajetória polinomial	28
4.2.2. Trajetória em linha reta	29
4.2.3. Trajetória de tempo mínimo	30
4.2.4. Dinâmica inversa	32
4.2.4.1. Trajetória polinomial	32
4.2.4.2. Trajetória em linha reta	33
4.2.4.3. Trajetória de tempo mínimo	34
4.2.5. Dinâmica direta	36
4.2.6. Estrutura básica	36
CAPÍTULO 5: RESULTADOS	38

	iii
5.1. Dinâmica inversa e direta	38
5.1.1. Trajetória polinomial	39
5.1.2. Trajetória em linha reta	44
5.1.3. Trajetória de tempo mínimo	49
5.2. Comparação de trajetórias	55
CAPÍTULO 6: CONCLUSÃO	59
CAPÍTULO 7: REFERÊNCIAS BIBLIOGRÁFICAS	60
ANEXO 1	I
ANEXO 2	XII

LISTA DE FIGURAS

Fig. 3.1: Placa de crescimento contendo 96 poços.	11
Fig. 3.2: Vista frontal de um elevador de placas 384.	12
Fig. 3.3: Esquema da cabeça de impressão e compartimento de lavagem.	13
Fig. 3.4: Esquema geral de um “microarray”.	15
Fig. 4.1: Movimento da cabeça de impressão.	20
Fig. 4.2: Posições dos centros de gravidade dos mecanismos 1 e 2.	22
Figura 5.1: Posições inicial e final do manipulador juntamente com a trajetória da cabeça de impressão – trajetória polinomial.	39
Figura 5.2: Posição angular dos mecanismos em função do tempo – trajetória polinomial	40
Figura 5.3: Torques transmitidos pelos motores em função do tempo, obtidos pelo método da dinâmica inversa – trajetória polinomial.	41
Figura 5.4: Velocidade e aceleração dos mecanismos em função do tempo – trajetória polinomial.	41
Figura 5.5: Velocidade linear da cabeça de impressão em função do tempo gasto na tarefa – trajetória polinomial.	42
Figura 5.6: Coordenadas da cabeça de impressão, obtidas pelos métodos das dinâmicas direta e inversa – trajetória polinomial.	42
Figura 5.7: Desvios das coordenadas da cabeça de impressão em relação ao método da dinâmica direta – trajetória polinomial.	43
Figura 5.8: Trechos finais das trajetórias obtidas pelos métodos das dinâmicas inversa e direta – trajetória polinomial.	43
Figura 5.9: Posições inicial e final do manipulador juntamente com as trajetórias da cabeça de impressão obtidas pelas dinâmicas direta e inversa – trajetória retilínea.	44
Figura 5.10: Posição angular dos mecanismos em função do tempo – trajetória retilínea.	45

	v
Figura 5.11: Torques transmitidos pelos motores em função do tempo, obtidos pelo método da dinâmica inversa – trajetória retilínea.	46
Figura 5.12: Velocidade e aceleração em função do tempo – trajetória retilínea.	46
Figura 5.13: Coordenadas da cabeça de impressão em função do tempo – trajetória retilínea.	47
Figura 5.14: Velocidade linear da cabeça de impressão em função do tempo – trajetória retilínea.	47
Figura 5.15: Desvios das coordenadas da cabeça de impressão em relação ao método da dinâmica direta – trajetória retilínea.	48
Figura 5.16: Trechos finais das trajetórias obtidas pelos métodos das dinâmicas inversa e direta – trajetória retilínea.	48
Figura 5.17: Diagramas de fases dos mecanismos – trajetória retilínea.	49
Figura 5.18: Posições inicial e final do manipulador juntamente com a trajetória da cabeça de impressão – trajetória de tempo mínimo.	50
Figura 5.19: Posição angular dos mecanismos em função do tempo – trajetória de tempo mínimo.	50
Figura 5.20: Torques transmitidos pelos motores – trajetória de tempo mínimo.	51
Figura 5.21: Velocidade e aceleração em função do tempo decorrido na tarefa – trajetória de tempo mínimo.	52
Figura 5.22: Diagramas de fases dos mecanismos – trajetória de tempo mínimo.	52
Figura 5.23: Coordenadas da cabeça de impressão – trajetória de tempo mínimo	53
Figura 5.24: Desvios das coordenadas da cabeça de impressão em relação ao método da dinâmica direta – trajetória de tempo mínimo.	53
Figura 5.25: Trechos finais das trajetórias obtidas pelos métodos das dinâmicas inversa e direta – trajetória de tempo mínimo.	54
Figura 5.26: Velocidade linear da cabeça de impressão em função do tempo decorrido na tarefa – trajetória de tempo mínimo.	54
Figura 5.27: Tipos de trajetórias obtidas pelo método da dinâmica inversa.	55
Figura 5.28: Tipos de trajetórias obtidas pelo método da dinâmica direta.	56
Figura 5.29: Velocidade escalar da “cabeça de impressão”.	57

LISTA DE TABELAS

Tabela 1.1: Densidade de “spots” em função da distância centro a centro.	6
Tabela 4.1: Condições para solução não ótima.	28
Tabela 4.2: Condições para solução ótima.	31
Tabela 4.3: Grandezas mantidas constantes.	36
Tabela 4.4. Densidades, massas e momentos de inércia.	37
Tabela 5.1: Coordenadas iniciais e finais da cabeça de impressão – trajetória polinomial.	39
Tabela 5.2: Torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa – trajetória polinomial.	40
Tabela 5.3: Coordenadas iniciais e finais da cabeça de impressão – trajetória retilínea.	44
Tabela 5.4: Torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa – trajetória retilínea.	45
Tabela 5.5: Coordenadas iniciais e finais da cabeça de impressão – trajetória de tempo mínimo.	49
Tabela 5.6: Torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa – trajetória de tempo mínimo.	51
Tabela 5.7: Comparação entre os valores máximos de torques e acelerações angulares, mantendo-se o tempo de execução da tarefa constante.	56
Tabela 5.8: Velocidade máxima da cabeça de impressão para cada tipo de trajetória.	57
Tabela 5.9: Desvio da trajetória obtida pela dinâmica inversa em relação à obtida pela dinâmica direta.	58

LISTA DE SÍMBOLOS

- θ : posição angular do mecanismo 1.
 α : posição angular do mecanismo 2.
 τ : torque.
 ω_1 : velocidade angular do mecanismo 1.
 ω_2 : velocidade angular do mecanismo 2.
 G : força que deriva o potencial.
 G_1 : posição do centro de gravidade do mecanismo 1.
 G_2 : posição do centro de gravidade do mecanismo 2.
 H : matriz de inércia do manipulador.
 I : momento de inércia.
 J : matriz de transformação linear.
 M_1 : massa da barra 1.
 m_1 : massa do mecanismo 1.
 M_2 : massa da barra 2.
 m_2 : massa do mecanismo 2.
 q : coordenada generalizada.
 t : tempo.
 t_f : tempo final.
 t_s : “switching time”.
 U : energia potencial do manipulador.
 x : abscissa da posição da cabeça do manipulador.
 x_G : abscissa do centro de gravidade.
 y : ordenada da posição da cabeça do manipulador.
 y_G : ordenada do centro de gravidade.

1. INTRODUÇÃO

1.1. Aplicações do “microarrayer”

O estudo dos genes de diversas espécies animais e dos seres humanos tem permitido avanços imensuráveis no tratamento de doenças adquiridas e de males congênitos bem como na descoberta de novos fármacos e em pesquisas de materiais tóxicos [1-2].

Dado que cada tipo de gene é caracterizado por uma determinada seqüência de bases nitrogenadas (adenina, guanina, citosina e timina) na molécula de DNA (ácido desoxirribonucléico), faz-se necessária uma manipulação precisa de amostras para se obterem resultados genéticos confiáveis.

Uma análise biológica de uma seqüência de DNA requer a coleta de uma amostra de pequeno volume, na ordem de 10^{-10} litros. Ou seja, os valores de volumes envolvidos em ensaios genéticos não podem ser captados por sistemas convencionais de manipulação de líquidos. Verifica-se, portanto, a necessidade da criação de sistemas de aquisição de amostras capazes de operar com volumes exíguos, como ocorrem nos micro-manipuladores (ou “microarrayers”) [3-28].

Assim, o “microarrayer” é um equipamento capaz de coletar as referidas amostras biológicas e depositá-las sobre substratos sólidos com alta velocidade.

1.2. Tecnologias para o transporte de amostras

Existem basicamente duas tecnologias distintas para se transportar o material genético da micro-célula para o substrato sólido onde as análises serão executadas.

Na primeira, não há contato entre a agulha ou pino que retira a gota da micro-célula e o substrato. Na segunda, que será objeto desta dissertação, há o contato direto entre a agulha de impressão de coleta e o suporte para análise.

A tecnologia empregada na construção dos “microarrays” de impressão de pontos permite a análise simultânea de milhares de seqüências de DNA através das seguintes etapas:

- o Coleta de amostra biológica contendo a seqüência de DNA a ser estudada da micro-célula, utilizando-se um pino ou uma micro-agulha;
- o Deposição da amostra sobre a superfície de uma placa de microscopia plana;
- o Obtenção da quantidade e da seqüência de bases através de análise a laser.

Em uma placa de dimensões de cerca de 18 mm por 54 mm são depositadas aproximadamente 9000 amostras. Cada amostra (“spot”) colocada sobre a placa apresenta diâmetro entre 75 μm e 360 μm .

A eficiência desta tecnologia pode ser atestada pelos trabalhos, dentre outros, da Affymetrix [29] e de um grupo de pesquisadores da Universidade de Stanford [30], comandado por Patrick Brown e Ronald Davis.

No caso da Affymetrix, milhares de genes presentes em uma amostra biológica são depositados sobre um substrato sólido, ocupando uma área de alguns centímetros quadrados. Através de técnicas fotográficas de fluorescência, é possível verificar-se a ordenação das seqüências que sofrem hibridização e, conseqüentemente, determinar o número de seqüências de DNA contido na amostra. A localização e a intensidade do sinal fluorescente estão relacionadas com a quantidade de DNA presente na amostra.

1.3. Características e componentes do “microarrayer”

O “microarrayer” é um robô que move agulhas ou pinos metálicos localizados na denominada “cabeça de impressão”. Há um mecanismo que permite o mergulho dos pinos em reservatórios de DNA, coletando volumes na ordem de micro litros e, finalmente, transferindo o fluido para uma placa denominada “chip” de DNA.

Os componentes de um “microarrayer” são basicamente os seguintes [20, 28, 31-37]:

- o plataforma estável e rígida para impressão,
- o braço robótico com movimento em três eixos,
- o servo controlador,
- o controle computacional,
- o chapa de montagem,
- o placa,
- o estações de lavagem e de secagem,
- o controle ambiental,
- o cabeça de impressão.

A plataforma para a impressão deve fornecer um suporte eficiente para evitar vibrações, com rugosidade inferior a 100 μm . O braço robótico deve ser preciso no posicionamento do pino metálico, com um erro menor ou igual a 10 μm .

O servo-controlador eletrônico é composto por um circuito digital de alta velocidade e deve controlar com acuidade a direção do mecanismo. O controle computacional consiste de um sistema controlado por um computador pessoal, que opera segundo a aplicação de um programa específico. Esse sistema deve transmitir instruções de alto nível para o servo-controlador e supervisionar o desempenho do equipamento.

A chapa de montagem deve apresentar rugosidade inferior a 100 μm . A placa de microscopia deve ser imobilizada sobre a superfície para que seja possível a impressão de milhares de pontos durante um período que pode atingir vários dias.

As estações de lavagem e de secagem são destinadas à limpeza dos pinos ao final de cada ciclo de operação do equipamento, evitando a contaminação cruzada, segundo o protocolo[38]. Esse sistema inclui lavagem com água e secagem a vácuo.

O controle ambiental refere-se aos particulados presentes no ar ambiente, que podem contaminar a placa. A umidade do ar deve ser controlada para não haver taxa excessiva de evaporação da solução (causada por baixa umidade do ar) ou para que os pontos não assumam forma irregular (causada por alta umidade do ar).

A cabeça de impressão pode conter vários pinos organizados em um arranjo matricial. O movimento vertical do pino pode ser conduzido pela gravidade.

1.4. Geometria da agulha ou pino metálico

De modo geral, os pinos (ou agulhas) metálicos podem seguir as seguintes especificações:

- Pinos sólidos: é o arranjo mais simples, no qual pinos confeccionados em aço são presos ordenadamente, em linhas e colunas, à cabeça de impressão. Nesse caso, a amostra é coletada por tensão superficial. Essa situação apresenta a vantagem da facilidade na limpeza, mas a desvantagem de que os pinos sólidos executam uma única impressão para cada coleta de material. Pela baixa velocidade por ciclo, esses tipos de pinos favorecem a evaporação do fluido.

- Pinos vazados: a coleta de material é realizada por capilaridade, facilitando a penetração do material genético no interior do pino e proporcionando a impressão contínua de material nas placas. O número de elementos que podem ser impressos por ciclo depende do tamanho do reservatório, da umidade ambiente e da solução na qual o material está dissolvido. A retenção de particulados no capilar do pino pode causar contaminação da amostra e obstrução no canal, gerando pontos de impressão irregulares.
- Anel e pino: pode ser utilizada uma combinação dessas duas formas geométricas, na qual os pinos apresentam um pequeno anel em suas extremidades. O anel funciona como um pequeno reservatório, armazenando 0,5 a 1,0 μL da solução coletada da placa fonte. O pino sólido passa pelo centro do menisco formado pela solução e auxilia na distribuição uniforme de amostra. No entanto, esse sistema conjugado consome grande quantidade de solução, embora seja menos afetado pela presença de particulados na solução ou por diferenças de viscosidade.

1.5. Material de confecção da agulha ou pino metálico

Os pinos devem ser confeccionados em material durável, inerte (que não sofra oxidação em ambiente salino) e duro. Geralmente os pinos são fabricados em aço inoxidável. A parte externa do pino deve ser lisa, não apresentar afinidade com a solução de DNA e não ser hidrofóbica ou eletricamente carregada.

1.6. Densidade das amostras

O número de “spots” por unidade de área depende da distância centro a centro entre “spots” consecutivos, c.

A densidade de “spots” é calculada por:

$$\text{Número de spot/área} = \frac{1}{c^2} \quad (1.1)$$

A equação 1.1 mostra que a densidade dos “spots” é muito sensível a pequenas variações da distância centro a centro. Na tabela 1.1 são apresentadas algumas simulações.

Tabela 1.1: Densidade de “spots” em função da distância centro a centro.

Diâmetro Spot	Centro-Centro	Densidade	Slide	Número de spots/slide
d (µm)	c (µm)	Número de “spots”/cm ²	A (cm ²)	N
25	50	40.000	18,75(25 × 75)mm	750.000
75	120	6944	12,50(25 × 50)mm	86.806

1.7. Problemas nas soluções atuais

Os modelos de “microarrays” fabricados atualmente, que operam com movimento cartesiano (denominados xyz), têm seu desempenho afetado por:

- o **Atrito:** É necessário utilizar motores com torques relativamente altos para movimentar a cabeça de impressão devido ao atrito nas correias de transmissão.
- o **Custo:** Um “microarray”, com todos os componentes, custa em torno de U\$ 30.000,00. Esse preço que pode ser considerado alto para a realidade dos laboratórios de pesquisa do Brasil.
- o **Vibração:** As altas velocidades exigidas e a necessidade de um posicionamento que garanta uma grande repetibilidade fazem com que surjam vibrações que devem ser rapidamente anuladas, caso isto não ocorra o desempenho do microarrayer não será o esperado.

1.8. O que seria desejável

É evidente que o caminho a ser seguido deve priorizar o custo. Além disso, o projeto a ser implementado deve contemplar as seguintes características:

- o Solução tecnológica: deve ser inovadora e vislumbrar possibilidades futuras de aperfeiçoamento. O projeto deve permitir diversos tipos de movimento da cabeça de impressão.
- o Espaço de trabalho: amplo, no sentido que seja possível uma rotina de tarefas com diferentes níveis de altura.
- o Atrito: mínimo, o que possibilita o uso de motores de menor torque e conseqüentemente um menor consumo de energia.
- o Vibrações: devem ser minimizadas, possibilitando a execução de uma rotina de tarefas em um intervalo de tempo menor.
- o Velocidade: deve ser da ordem de 0,5 m/s, mas ajustável conforme o tipo de trabalho a ser executado.
- o Materiais: todo o conjunto deve ser fabricado com materiais leves, com elevado módulo de elasticidade.

2. OBJETIVOS

O objetivo principal deste trabalho é propor um mecanismo, juntamente com estratégias de controle, de uma pipetadora matricial para determinação de expressões genéticas.

Cada estratégia de controle de malha aberta será simulada numericamente por métodos distintos. O intervalo de tempo de execução de uma tarefa deve ser minimizado o que exige uma estratégia de controle de tempo mínimo. Uma malha de controle fechada deverá se incumbir do posicionamento fino.

Espera-se que esse estudo permita o desenvolvimento de um produto competitivo em termos de custo, precisão e velocidade.

3. REVISÃO BIBLIOGRÁFICA

Os parâmetros de construção de um “microarrayer” dependem das características específicas dos materiais biológicos e do estudo mecânico do equipamento. A combinação desses dois itens resulta em um equipamento de desempenho satisfatório e flexível.

Neste capítulo serão descritos os procedimentos envolvendo a tecnologia de impressão com agulhas e os parâmetros de projeto do equipamento. Também será discutido no apêndice desse trabalho o método físico-matemático utilizado no estudo dinâmico dos manipuladores.

3.1. Evolução do desenvolvimento do “microarrayer”

Por volta de 1980, foi desenvolvida a química combinatória com o objetivo de descobrir novas drogas, utilizando-se a síntese paralela de compostos e sistemas de elevado grau de seleção.

A química combinatória constrói um banco de compostos a partir da síntese de moléculas lineares, como os peptídeos, e da seleção aleatória de compostos bio-ativos, usando para isso sistemas biológicos sensíveis, como reações antígeno-anticorpo.

Ou seja, inicia-se uma fase na qual a velocidade de obtenção de grandes quantidades de resultados é condição primordial para a elaboração de um vasto arquivo de padrões para analogias com casos estudados.

Em 1988, Fodor, trabalhava com química combinatória na síntese de biomoléculas de alta densidade. Suas experiências em campos interdisciplinares como a bioquímica e a biofísica, o levaram a combinar tecnologias de diferentes áreas do conhecimento, como a eletrônica, a biologia e a química combinatória, no estudo de oligonuclédeos de alta densidade [39-40].

As tecnologias combinadas envolviam métodos gráficos de fotolito usados na fabricação de semicondutores e a síntese de ácidos nucleicos [41-43], já demonstrando potencial comercial.

Em 1993, Fodor desenvolveu uma tecnologia para oligonucleotídeos de alta densidade. A partir da confecção de materiais de alta densidade de RNA ou DNA, em 1994, a Affymax comercializou as primeiras edições do “microarrayer”, compatibilizando a eletrônica e a genética [44].

Como consequência dessa evolução metodológica, atualmente é possível a construção de um banco de padrões para comparação de expressões genéticas de elevada magnitude, possibilitando o mapeamento genético de diversas espécies. Nesse contexto, equipamentos que produzem micro-arranjos de amostras são fundamentais para o desenvolvimento da área genético-farmacêutica.

3.2 Tecnologia de impressão com agulhas

Nesta secção serão descritos os elementos que constituem um sistema que emprega a tecnologia de impressão de material genético com agulhas, bem como os procedimentos gerais utilizados.

Amostras contendo material genético purificado são colocadas em poços de uma placa de crescimento, conhecida como “placa Elisa”. Essa placa é uma matriz 8x12, na qual podem ser depositadas até 96 amostras, conforme ilustrado na figura 3.1.

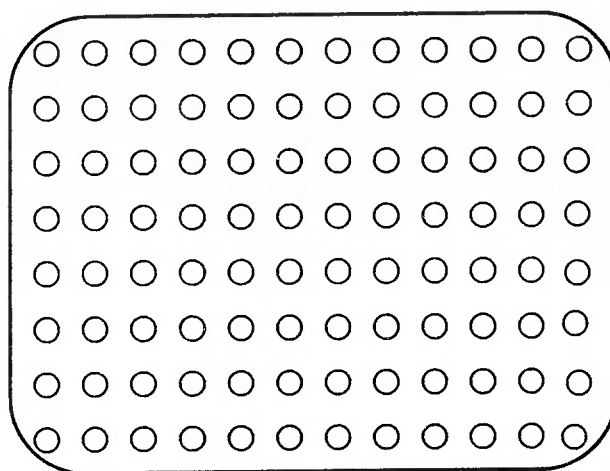


Figura 3.1. Placa de crescimento contendo 96 poços.

Após o preenchimento dos poços de quatro placas de crescimento, o conteúdo de cada poço é transferido para poços de outra placa, denominada “placa 384”. Essa placa apresenta linhas numeradas desde 1 até 24 e colunas classificadas desde A até P, resultando em 384 posições. Cada posição é ocupada por um volume de cerca de 15 μL de amostra genética e mais 15 μL de solução tampão para manter a adesão do material.

A placa 384 é colocada horizontalmente em um compartimento conhecido como “elevador” do microarrayer. Um “elevador” típico abriga cerca de 12 placas 384, conforme ilustrado na figura 3.2.

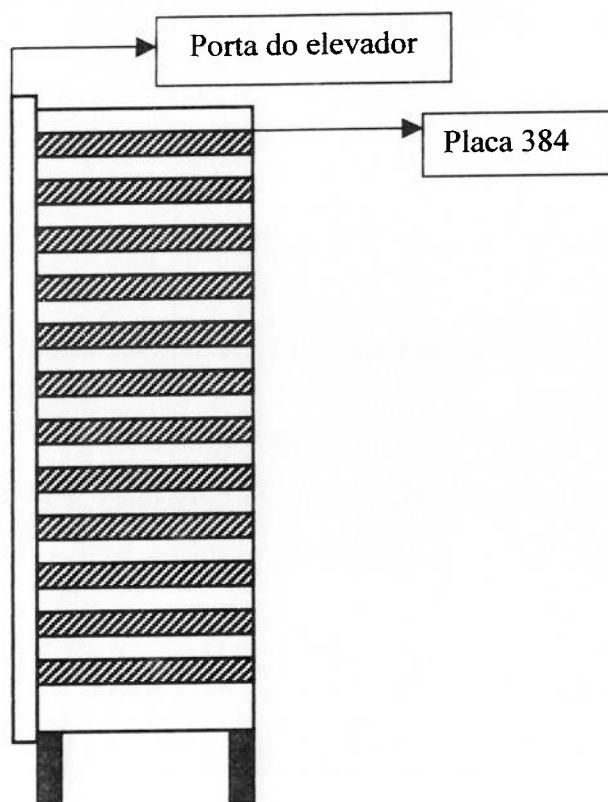


Figura 3.2. Vista frontal de um “elevador” de placas 384.

Seleciona-se uma das 12 placas e a mesma é posicionada em uma área de coleta de material pelas agulhas de impressão.

As agulhas encontram-se conectadas à cabeça do microarrayer que, por sua vez é conectado ao braço do robô. Tome-se como exemplo uma cabeça contendo 12 agulhas arranjadas segundo uma linha. Antes do início de cada fase de coleta, as agulhas são lavadas no chamado “compartimento de lavagem” em uma solução contendo KOH, etanol e água. Existe também um estágio de descanso das agulhas, contendo água, que apenas é utilizado em operações de parada do equipamento. Após a lavagem, as agulhas são secas com nitrogênio extra-seco. A figura 3.3 ilustra a situação descrita.

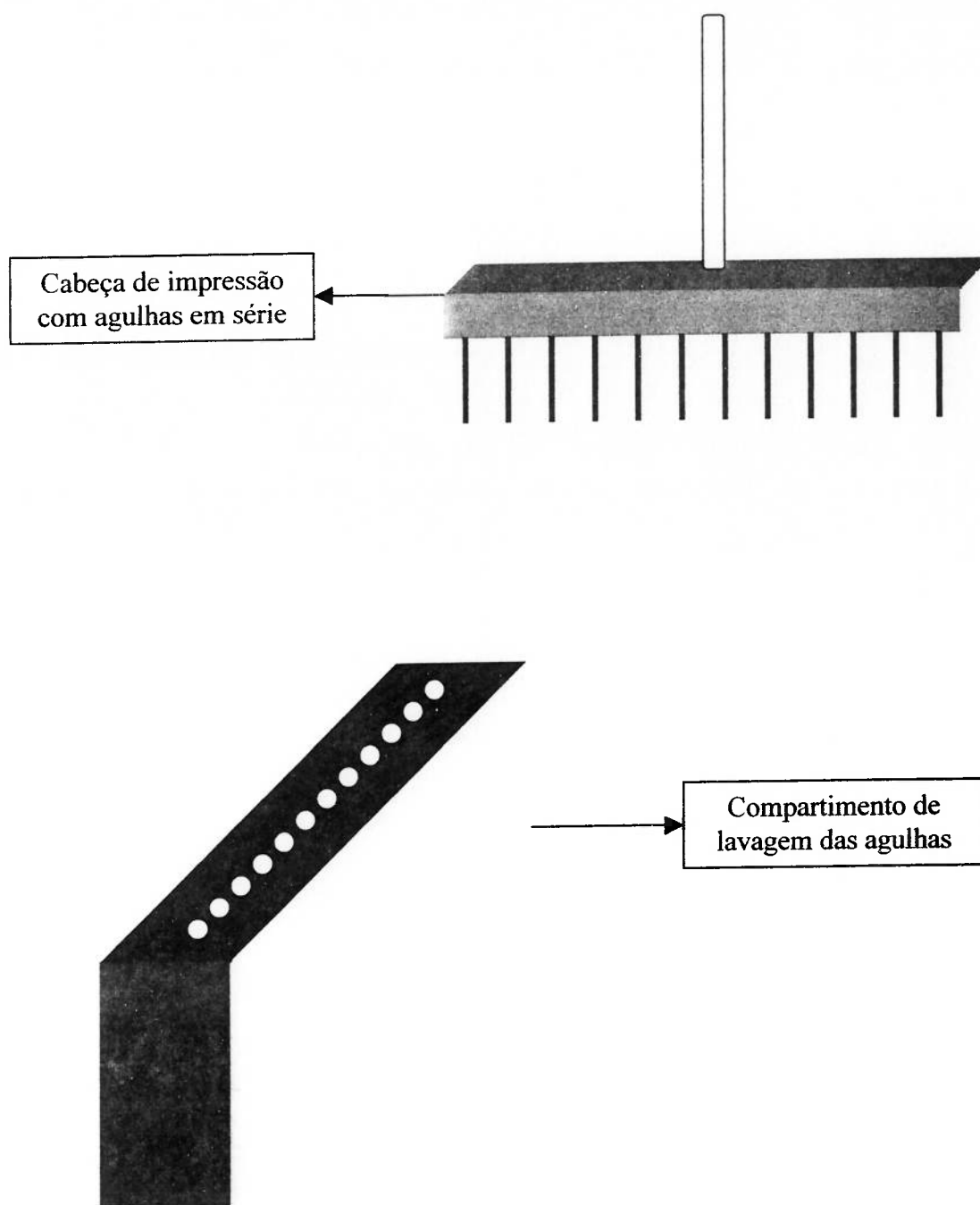


Figura 3.3. Esquema da cabeça de impressão e compartimento de lavagem.

As agulhas secas e limpas mergulham nas doze primeiras linhas da placa 384 em operação no momento. A figura 3.4 ilustra um esquema geral de um microarrayer.

Cada micro-agulha é mergulhada em um poço da placa 384 contendo amostra da solução biológica, captando um volume exíguo de líquido no seu interior, da ordem de nano-litros (10^{-9} L) ou pico-litros (10^{-12} L).

A micro-agulha é então movida pelo mecanismo eletrônico que comanda o movimento da cabeça e é colocada sobre a superfície de uma matriz ou “slide” fotográfico, denominado “chip” de DNA, na superfície do qual é depositada a amostra de solução biológica, com capacidade para mais de 9000 deposições.

O diâmetro que cada amostra (“spot”) ocupa na superfície do “chip” é definido pelo balanço das energias livres da amostra, do substrato sólido (“chip”) e da micro-agulha. Geralmente, as micro-agulhas apresentam pontas planas e o diâmetro da agulha está diretamente relacionado com o volume de amostra coletada. Em alguns casos, o uso de uma ponta de agulha curva pode facilitar a tarefa de coleta.

Uma amostragem completa envolve a coleta de mais de duas mil amostras de solução, resultando em um procedimento bastante lento. A fim de minimizar esta dificuldade, têm sido projetados manipuladores capilares, na forma de tesouras e agulhas com fendas, capazes de conduzir maiores volumes de soluções. No entanto, a desvantagem dessas geometrias é a necessidade de realização de trabalho mecânico para se depositar a amostra sobre o substrato.

Algumas empresas têm desenvolvido os projetos mais utilizados de “microarrayers”, nos quais as agulhas dos manipuladores operam de modo semelhante a canetas esferográficas a tinta. A massa de amostra localizada na ponta da agulha é colocada em contato com o substrato antes que a própria ponta da agulha o faça, fazendo com que o substrato puxe uma pequena quantidade de solução para fora do reservatório destinado ao fluido quando a agulha é retirada da superfície.

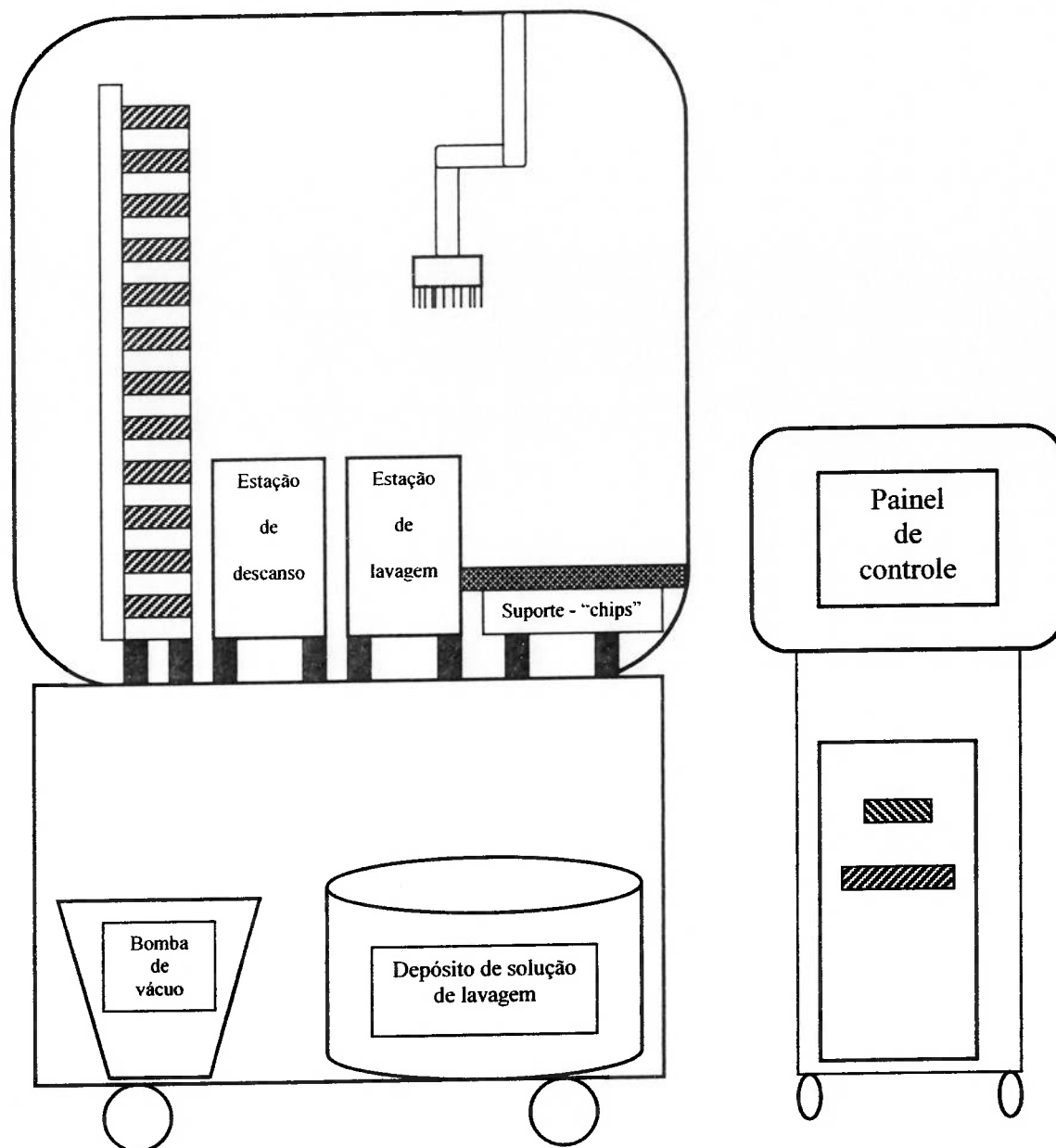


Figura 3.4: Esquema geral de um "microarrayer".

Resumindo, as agulhas do “microarrayer” coletam amostras (da ordem de microlitros) com alta densidade de DNA ou RNA presentes em uma “placa 384” e as depositam em posições precisas de “chips”. As agulhas são lavadas e secas para minimização de possíveis contaminações. A umidade da área de trabalho é mantida no máximo em 60%.

Os “microarrays” que utilizam impressão com agulhas rígidas ou pinos são relativamente comuns, não havendo o contato direto entre a ponta da agulha e o suporte sólido aonde será depositado o material genético [26].

3.3. Velocidade das análises

A evolução e a velocidade do mapeamento genético estão diretamente relacionadas com o desenvolvimento da tecnologia dos “microarrayer”. Antes da aplicação dessa tecnologia predominava a idéia de “um gene, um experimento”.

A utilização de um sistema robótico-computacional na análise genética permite, atualmente, a execução de até milhares de análises paralelas [4, 44-45]. Essa velocidade pode superar a frequência com que são realizadas as análises químicas e biológicas.

Por exemplo, a procura de mutações genéticas no DNA e a busca de expressões genéticas específicas de um determinado tecido [46] requerem a análise simultânea de cerca de 10.000 amostras. Esses arranjos, chamados de arranjos de alta densidade, são fabricados por tecnologias de fotolito e gráficas, similares às usadas em microeletrônica.

Muitas vezes, há a necessidade da construção de gráficos que comparam expressões genéticas normais com expressões genéticas de tecidos defeituosos. Esses gráficos podem ser obtidos combinando-se a micro-dissecação de captura a “laser” [47], a amplificação da imagem de uma célula isolada de DNA e a análise da matriz.

Uma limitação na velocidade de produção dos resultados é a capacidade de transferência de líquidos dos seus recipientes para as placas. Uma solução atual é a conexão de reservatórios por capilares. O uso de reservatórios maiores aumenta a produção e facilita a introdução da agulha, segundo as dimensões do projeto do bocal.

3.4. Obtenção de imagens

As técnicas que usam um ou mais “lasers” e analisam imagens simultaneamente oferecem resultados menos confiáveis do que as obtidas por câmeras específicas.

Uma câmera CCD, com dispositivo de espalhamento acoplado, exhibe imagens seqüenciais que cobrem uma extensa área superficial [48]. A câmera CCD apresenta elevada eficiência na produção de imagens, mas pode haver saturação de fluorescência, acarretando uma resposta não-linear do equipamento. Recentemente, empresas têm desenvolvido câmeras CCD que coletam mais fótons por “pixel” sem haver saturação, além de outros sistemas óticos. Os ruídos de fundo têm sido eliminados pela utilização de “lasers” que operam com baixa freqüência.

A qualidade das imagens depende da convergência entre a captura e a impressão da mesma [49], o que pode ser obtido por microscópios que trabalham com comprimentos de onda próximos ao do infra-vermelho, com níveis mais restritos de focos [50].

A técnica RLS (Ressonance Light Scattering), por exemplo, toma vantagens das propriedades de espalhamento da luz por nano-partículas em suspensão. Quando irradiadas por luz branca, as partículas espalham a luz em diferentes comprimentos de onda, dependendo do tamanho, da forma e da composição da partícula [51].

3.5. Configurações atuais

Os modelos atuais que operam em três eixos (XYZ) trabalham com módulos, incluindo pipetadoras eletrônicas (agulhas), sistema de impressão e câmera CCD, e são reconfiguráveis em minutos.

A cabeça do robô pode conter de 12 a 48 agulhas, arranjadas em linha ou em linhas e colunas. A velocidade de um ciclo de coleta de amostras é da ordem de um segundo. As resoluções atingem 1,25 μm , nos eixos X e Y, e 0,25 μm segundo o eixo Z, com repetibilidade de 3 μm .

O diâmetro de cada “spot” da placa que recebe uma amostra (“chip” de DNA) varia de, aproximadamente, 25 μm a 75 μm . A distância entre os centros de dois “spots” consecutivos é de 120 μm . Cada “chip”, medindo cerca de 18mm x 54 mm, pode receber 9216 amostras. Para “spots” com diâmetro de 25 μm , a densidade dos mesmos em uma placa é de cerca de 10 “spots” por mm^2 . A produção de arranjos genéticos é da ordem de 1 segundo por “spot” por pino. O sistema de pipetadora eletrônica coleta volumes de 0,1 a 200 μL .

4. METODOLOGIA

4.1. Modelamento dinâmico do manipulador

Será mostrado, utilizando-se o método Lagrangeano [53-56], como são obtidos os torques que modificam a configuração do manipulador em função de sua inércia e das variáveis que definem sua posição no tempo. A notação das matrizes de inércia está detalhada no anexo 1.

O manipulador que posiciona a cabeça de impressão com as agulhas está apresentado esquematicamente na figura 4.1. A cabeça de impressão se movimenta no plano $x0y$. O plano de trabalho $x0z$ possui um movimento independente. São dois mecanismos de quatro barras com um ponto de acoplamento. A posição da cabeça é definida pelos ângulos θ e α indicados na figura 4.1. Todo o conjunto é considerado um corpo rígido e os atritos são desprezados. Assume-se que as massas estão uniformemente distribuídas. O movimento da cabeça é governado pelos torques dos motores 1 e 2. Observar que as barras de massa M_1 e M_2 , indicadas na figura 4.1, estão sempre nas posições horizontal e vertical respectivamente. Os motores que movimentam o manipulador estão fixos na base da pipetadora o que permite obter seus torques sem considerar a inércia dos motores.

Para efeito de obtenção das equações dos torques dos motores que irão governar o movimento do manipulador, faz-se a equivalência dos mecanismos da figura 4.1 com os da figura 4.2. Para cada mecanismo, as barras paralelas estão unidas e a barra que as une é interpretada como uma massa puntiforme, uma vez que este ato de movimento é sempre de translação.

O correspondente momento de inércia é calculado em relação ao centro de massa de cada corpo rígido. Adotam-se como coordenadas generalizadas os ângulos θ e α ,

$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} \theta \\ \alpha \end{bmatrix} \quad (4.1)$$

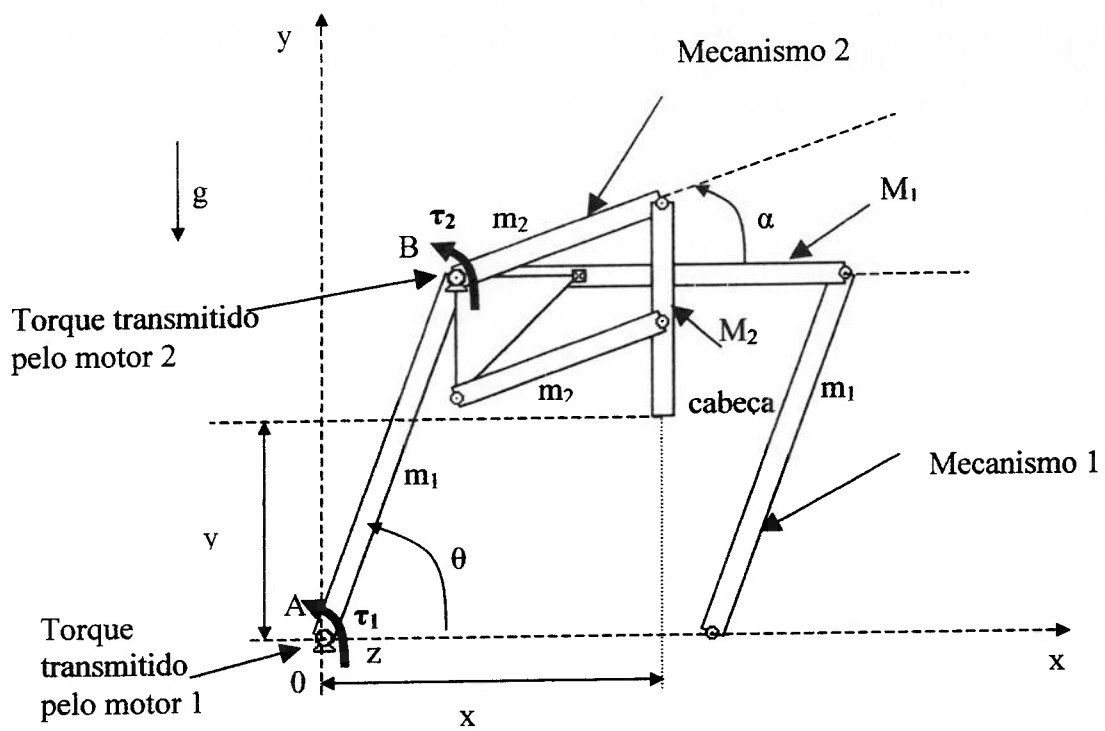


Figura 4.1: Movimento da cabeça de impressão.

As correspondentes derivadas de primeira e segunda ordem são:

$$\dot{q} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \quad (4.2)$$

$$\ddot{q} = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} = \begin{bmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix} \quad (4.3)$$

Na figura 4.1, nota-se que os ângulos θ e α são definidos em relação a duas direções fixas em relação ao referencial inercial, de forma que as velocidades angulares ω_1 e dos mecanismos são:

$$\omega_1 = [1 \ 0] \dot{q} = [1 \ 0] \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} = \dot{\theta} \quad (4.4)$$

$$\omega_2 = [0 \ 1] \dot{q} = [0 \ 1] \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} = \dot{\alpha} \quad (4.5)$$

A partir das equações 4.4 e 4.5 identificam-se as expressões das matrizes de transformação angulares [57-59]:

$$\bar{\omega}_1 = J_A^1 \dot{q} = J_A^1 \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} = [1 \ 0] \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \therefore J_A^1 = [1 \ 0] \quad (4.6)$$

$$\bar{\omega}_2 = J_A^2 \dot{q} = J_A^2 \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} = [0 \ 1] \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \therefore J_A^2 = [0 \ 1] \quad (4.7)$$

Na figura 4.2, G_1 e G_2 definem as posições dos centros de gravidade dos mecanismos 1 e 2 respectivamente.

As massas dos mecanismos e as respectivas posições dos centros de gravidade G_1 e G_2 medidas ao longo das barras são definidas pelas seguintes equações:

$$m_{G1} = 2 * m_1 + M_1 \quad \text{e} \quad m_{G2} = 2 * m_2 + M_2 \quad (4.8)$$

$$L_{G1} = \frac{2.m_1.\frac{L_1}{2} + M_1.L_1}{2.m_1 + M_1} \quad \text{e} \quad L_{G2} = \frac{2.m_2.\frac{L_2}{2} + M_2.L_2}{2.m_2 + M_2} \quad (4.9)$$

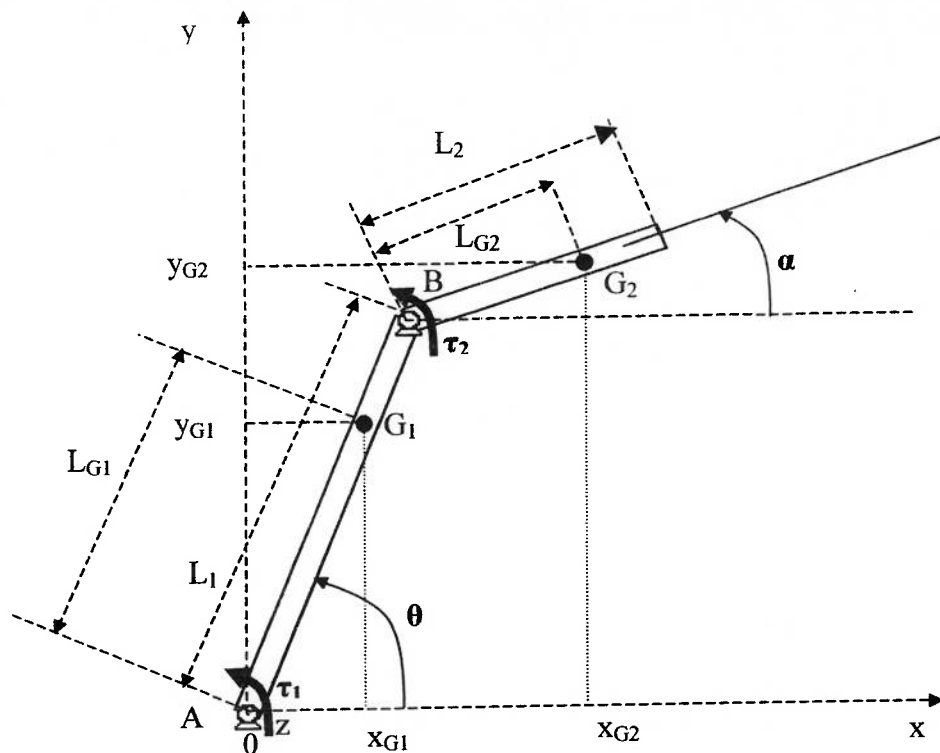


Figura 4.2: Posições dos centros de gravidade dos mecanismos 1 e 2.

As coordenadas cartesianas dos baricentros de cada mecanismo são representadas pelas seguintes equações:

$$x_{G1} = L_{G1} \cdot \cos(\theta) \quad (4.10)$$

$$y_{G1} = L_{G1} \cdot \text{sen}(\theta) \quad (4.11)$$

$$x_{G2} = L_1 \cdot \cos(\theta) + L_{G2} \cdot \cos(\alpha) \quad (4.12)$$

$$y_{G2} = L_1 \cdot \text{sin}(\theta) + L_{G2} \cdot \text{sin}(\alpha) \quad (4.13)$$

As componentes cartesianas das velocidades dos baricentros podem ser obtidas mais facilmente por derivação direta das equações 4.8, 4.9, 4.10 e 4.11, resultando nas equações seguintes:

$$\dot{x}_{G1} = -L_{G1} \cdot \text{sin}(\theta) \cdot \dot{\theta} \quad (4.14)$$

$$\dot{y}_{G1} = +L_{G1} \cdot \cos(\theta) \cdot \dot{\theta} \quad (4.15)$$

$$\dot{x}_{G2} = -L_1 \cdot \text{sen}(\theta) \cdot \dot{\theta} - L_{G2} \cdot \text{sin}(\alpha) \cdot \dot{\alpha} \quad (4.16)$$

$$\dot{y}_{G2} = +L_1 \cdot \cos(\theta) \cdot \dot{\theta} + L_{G2} \cdot \cos(\alpha) \cdot \dot{\alpha} \quad (4.17)$$

As equações 4.14, 4.15, 4.16 e 4.17 podem ser apresentadas na forma matricial, o que possibilita obter a expressão da matriz de transformação linear para cada mecanismo. Ou seja,

$$\vec{v}_{G1} = \begin{bmatrix} -L_{G1} \cdot \text{sen}(\theta) & 0 \\ +L_{G1} \cdot \text{cos}(\theta) & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} = J_L^1 \cdot \dot{\vec{q}} \quad \therefore$$

$$J_L^1 = \begin{bmatrix} -L_{G1} \cdot \text{sen}(\theta) & 0 \\ +L_{G1} \cdot \text{cos}(\theta) & 0 \end{bmatrix} \quad (4.18)$$

$$\vec{v}_{G2} = \begin{bmatrix} -L_1 \cdot \text{sen}(\theta) & -L_{G2} \cdot \text{sin}(\alpha) \\ +L_1 \cdot \text{cos}(\theta) & +L_{G2} \cdot \text{cos}(\alpha) \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\alpha} \end{bmatrix} = J_L^2 \cdot \dot{\vec{q}} \quad \therefore$$

$$J_L^2 = \begin{bmatrix} -L_1 \cdot \text{sen}(\theta) & -L_{G2} \cdot \text{sen}(\alpha) \\ +L_1 \cdot \text{cos}(\theta) & +L_{G2} \cdot \text{cos}(\alpha) \end{bmatrix} \quad (4.19)$$

Com as equações básicas já definidas, podem ser realizadas as substituições na equação que fornece a matriz de inércia H do manipulador expressa por,

$$H = \sum_{i=1}^n (m_i \cdot (J_L^i)^T \cdot J_L^i + (J_A^i)^T \cdot I_i \cdot J_A^i) \quad (4.20)$$

Assim, resulta-se em:

$$H = \{m_{G1} \cdot J_L^{(1)T} \cdot L_L^1 + J_A^{(1)T} \cdot I_{G1} \cdot J_A^1\} + \{m_{G2} \cdot J_L^{(2)T} \cdot J_L^2 + J_A^{(2)T} \cdot I_{G2} \cdot J_A^2\} \quad (4.21)$$

Realizando-se as substituições das matrizes de transformação na equação 4.21,

$$H = \begin{bmatrix} (m_{G1} \cdot L_{G1}^2 + I_{G1}) + m_{G2} \cdot L_1^2 & m_{G2} \cdot L_1 \cdot L_{G2} \cdot \text{cos}(\theta - \alpha) \\ m_{G2} \cdot L_1 \cdot L_{G2} \cdot \text{cos}(\theta - \alpha) & (m_{G2} \cdot L_{G2}^2 + I_{G2}) \end{bmatrix} \quad (4.22)$$

Na seqüência, obtém-se os termos da equação:

$$h_{ijk} = \frac{\partial H_{ij}}{\partial q_k} - \frac{1}{2} \cdot \frac{\partial H_{jk}}{\partial q_i} \quad (4.23)$$

Resultando-se em:

$$h_{111} = \frac{d}{d\theta} H_{11} - \frac{1}{2} \cdot \frac{d}{d\theta} H_{11} \quad \therefore h_{111} = 0 ;$$

$$h_{112} = \frac{d}{d\alpha} H_{11} - \frac{1}{2} \cdot \frac{d}{d\theta} H_{12} \quad \therefore h_{112} = \frac{1}{2} \cdot m_{G2} \cdot L_1 \cdot L_{G2} \cdot \sin(\theta - \alpha)$$

$$h_{121} = \frac{d}{d\theta} H_{12} - \frac{1}{2} \cdot \frac{d}{d\theta} H_{21} \quad \therefore h_{121} = -\frac{1}{2} \cdot m_{G2} \cdot L_1 \cdot L_{G2} \cdot \sin(\theta - \alpha)$$

$$h_{122} = \frac{d}{d\alpha} H_{12} - \frac{1}{2} \cdot \frac{d}{d\theta} H_{22} \quad \therefore h_{122} = m_{G2} \cdot L_1 \cdot L_{G2} \cdot \sin(\theta - \alpha)$$

$$h_{211} = \frac{d}{d\theta} H_{21} - \frac{1}{2} \cdot \frac{d}{d\alpha} H_{11} \quad \therefore h_{211} = -m_{G2} \cdot L_1 \cdot L_{G2} \cdot \sin(\theta - \alpha)$$

$$h_{212} = \frac{d}{d\alpha} H_{21} - \frac{1}{2} \cdot \frac{d}{d\alpha} H_{12} \quad \therefore h_{212} = \frac{1}{2} \cdot m_{G2} \cdot L_1 \cdot L_{G2} \cdot \sin(\theta - \alpha)$$

$$h_{221} = \frac{d}{d\theta} H_{22} - \frac{1}{2} \cdot \frac{d}{d\alpha} H_{21} \quad \therefore h_{221} = -\frac{1}{2} \cdot m_{G2} \cdot L_1 \cdot L_{G2} \cdot \sin(\theta - \alpha)$$

$$h_{222} = \frac{d}{d\alpha} H_{22} - \frac{1}{2} \cdot \frac{d}{d\alpha} H_{22} \quad \therefore \quad h_{222} = 0 \quad (4.24)$$

A energia potencial do manipulador é obtida a partir da figura 4.2. resultando em:

$$U = -m_{G1} \cdot g \cdot L_{G1} \cdot \text{sen}(\theta) - m_{G2} \cdot g \cdot [L_1 \cdot \text{sen}(\theta) + L_{G2} \cdot \text{sen}(\alpha)] \quad (4.25)$$

Portanto, as forças que derivam do potencial são:

$$G_1 = \frac{dU}{d\theta} = -m_{G1} \cdot g \cdot L_{G1} \cdot \cos(\theta) - m_{G2} \cdot g \cdot L_1 \cos(\theta)$$

$$G_2 = \frac{dU}{d\alpha} = -m_{G2} \cdot g \cdot L_{G2} \cos(\alpha) \quad (4.26)$$

As equações de Lagrange para cada coordenada generalizada resultam,

$$\tau_1 = [m_{G1} \cdot L_{G1}^2 + I_{G1} + m_{G2} \cdot L_1^2] \ddot{\theta} + m_{G2} \cdot L_1 \cdot L_{G2} \cdot \cos(\theta - \alpha) \ddot{\alpha} + m_{G2} \cdot L_1 \cdot L_{G2} \cdot \cos(\theta - \alpha) \dot{\alpha}^2 + g \cdot \cos(\theta) [m_{G1} \cdot L_{G1} + m_{G2} \cdot L_1]$$

(4.27)

$$\tau_2 = [m_{G2} \cdot L_{G2}^2 + I_{G2}] \ddot{\alpha} + m_{G2} \cdot L_1 \cdot L_{G2} \cdot \cos(\theta - \alpha) \ddot{\theta} - m_{G2} \cdot L_1 \cdot L_{G2} \cdot \text{sen}(\theta - \alpha) \dot{\theta}^2 + m_{G2} \cdot g \cdot L_{G2} \cdot \cos(\alpha)$$

(4.28)

4.2. Controle do manipulador

As trajetórias a serem descritas pela cabeça do manipulador são definidas a partir das seguintes condições:

- a) estabelecimento das coordenadas cartesianas das posições iniciais $x(0)$ e $y(0)$ e finais $x(t_f)$ e $y(t_f)$ da trajetória.
- b) fixação das componentes cartesianas de velocidade para o início $\dot{x}(0)$ e $\dot{y}(0)$ e fim $\dot{x}(t_f)$ e $\dot{y}(t_f)$ da trajetória.
- c) definição dos torques de partida dos motores necessários para superar os torques gravitacionais.
- d) caso a trajetória não seja ótima no tempo, deve ser definido o tempo de percurso t_f .

As seguintes trajetórias serão analisadas:

Trajétória polinomial: com os estados inicial e final previamente determinados juntamente com o tempo de percurso, determina-se a correspondente equação da trajetória da cabeça do manipulador.

Trajétória em linha reta: com os estados inicial e final previamente determinados juntamente com o tempo de percurso faz-se com que a cabeça do manipulador descreva uma trajetória linear.

Trajétória de tempo mínimo: com estado inicial e final previamente determinados, bem como as máximas acelerações durante o movimento, determina-se a equação da trajetória que estabelece um tempo mínimo t_f .

4.2.1 Trajetória polinomial

Estabelece-se um intervalo de tempo para executar a trajetória.

Para as posições inicial e final, impõe-se velocidade nula para a cabeça do manipulador. Essas condições estão expressas na tabela 4.1.

Tabela 4.1: Condições para solução não ótima.

	posição	velocidade
inicial	$q(0)$	$\dot{q}(0) = 0$
final	$q(t_f)$	$\dot{q}(t_f) = 0$

Para descrever o movimento desta trajetória é requerido um polinômio cúbico, com quatro coeficientes independentes. As equações de posição e velocidade são:

$$q = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (4.29)$$

$$\dot{q} = a_1 + 2.a_2 t + 3.a_3 t^2 \quad (4.30)$$

onde q denota uma coordenada generalizada.

Combinando-se estas equações com as condições de contorno, obtém-se:

$$q(0) = a_0 \quad \dot{q}(0) = a_1 \quad (4.31)$$

$$q(t_f) = a_0 + a_1 t_f + a_2 t_f^2 + a_3 t_f^3 \quad \dot{q}(t_f) = a_1 + 2.a_2 t_f + 3.a_3 t_f^2 \quad (4.32)$$

Estas quatro equações podem ser combinadas em uma equação matricial:

$$(4.33) \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} q(0) \\ \dot{q}(0) \\ q(t_f) \\ \dot{q}(t_f) \end{bmatrix} \quad \therefore \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} q(0) \\ 0 \\ q(t_f) \\ 0 \end{bmatrix}$$

Disto resulta:

$$a_0 = q(0) \qquad a_1 = 0 \qquad (4.34)$$

$$a_2 = 3 \cdot \frac{\{q(t_f) - q(0)\}}{t_f^2} \qquad a_3 = -2 \cdot \frac{\{q(t_f) - q(0)\}}{t_f^3} \qquad (4.35)$$

4.2.2 Trajetória em linha reta

Estabelece-se um intervalo de tempo para executar a trajetória. Fixa-se a posição inicial e final da cabeça do manipulador. A trajetória linear é expressa pela equação:

$$y = A + B.x, \text{ onde} \qquad (4.36)$$

$$A = y(0) - \left\{ \frac{y(t_f) - y(0)}{x(t_f) - x(0)} \right\} \cdot x(0) \text{ e } B = \frac{y(t_f) - y(0)}{x(t_f) - x(0)} \qquad (4.37)$$

As coordenadas cartesianas da cabeça do manipulador são,

$$x = L_1 \cdot \cos(\theta) + L_2 \cdot \cos(\alpha) \quad (4.38)$$

$$y = L_1 \cdot \sin(\theta) + L_2 \cdot \sin(\alpha) - L_2 \quad (4.39)$$

Substituindo-se as equações 4.38 e 4.39 na equação 4.36 obtém-se,

$$[L_1 \cdot \sin(\theta) + L_2 \cdot \sin(\alpha) - L_2] = \left[y(0) - \frac{y(t_f) - y(0)}{x(t_f) - x(0)} x(0) \right] + \left[\frac{y(t_f) - y(0)}{x(t_f) - x(0)} \right] [L_1 \cdot \cos(\theta) + L_2 \cdot \cos(\alpha)] \quad (4.40)$$

Observa-se, pela equação 4.40, que as variáveis θ e α não são independentes. Portanto, podem ser fixadas condições cinemáticas que permitam obter a equação horária de uma variável e, assim, obter a outra.

4.2.3 Trajetória de tempo mínimo

Nesta seção, procura-se a história de controle de tempo mínimo, conhecidos o estado inicial e o estado final, admitindo-se que exista uma única descontinuidade na história de controle. Esta trajetória é chamada “Bang-Bang” [51], com condição de tempo mínimo t_f entre duas posições fixas $q_+(0)$ e $q_-(t_f)$, com aceleração em módulo constante em todo intervalo de tempo com velocidade nula nas posições inicial e final, ou seja, $\dot{q}(0) = 0$ e $\dot{q}(t_f) = 0$.

A solução ótima impõe que a aceleração tenha um máximo $+\ddot{q}_+$ com um apropriado “switching time” t_s a partir do qual a aceleração instantaneamente atinja um mínimo \ddot{q}_- . Nestas condições deduzem-se as expressas na tabela 4.2.

Tabela 4.2: Condições para solução ótima.

Intervalo de tempo	Posição	Velocidade
$0 \leq t < t_s$	$q_+ = \frac{1}{2} \ddot{q}_+ t^2 + q_+(0)$	$\dot{q}_+ = \ddot{q}_+ t$
$t_s < t \leq t_f$	$q_- = \frac{1}{2} \ddot{q}_- t^2 - \ddot{q}_- t_f t + q_-(t_f) + \frac{1}{2} \ddot{q}_- t_f^2$	$\dot{q}_- = \ddot{q}_- (t - t_f)$

Com a condição de continuidade de posição e velocidade para o instante t_s , obtém-se:

$$\frac{1}{2} \ddot{q}_+ t_s^2 + q_+(0) = \frac{1}{2} \ddot{q}_- t_s^2 - \ddot{q}_- t_f t_s + q_-(t_f) + \frac{1}{2} \ddot{q}_- t_f^2 \quad (4.41)$$

$$\ddot{q}_+ t_s = \ddot{q}_- (t_s - t_f) \quad (4.42)$$

Das equações 4.29 e 4.30 resultam

$$t_s = \frac{\ddot{q}_-}{\ddot{q}_- - \ddot{q}_+} t_f \quad (4.43)$$

$$t_f = \sqrt{2 \cdot \frac{(\ddot{q}_- - \ddot{q}_+)}{\ddot{q}_- \cdot \ddot{q}_+} \{q_-(t_f) - q_+(0)\}} \quad (4.44)$$

4.2.4. Dinâmica inversa

As equações dinâmicas 4.27 e 4.28 permitem, através dos torques exercidos pelos motores nas juntas, obter a trajetória. O procedimento que será utilizado na presente seção é o inverso. A partir de uma trajetória serão obtidos os torques.

4.2.4.1. Trajetória polinomial

Neste caso, a posição dos mecanismos do manipulador é definida pela equação:

$$\theta = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (4.45)$$

Considerando que: $\dot{\theta}(0) = \dot{\theta}(t_f) = 0$, então os coeficientes valem:

$$a_0 = \theta(0)$$

$$a_1 = 0$$

$$a_2 = 3 \cdot \frac{\{\theta(t_f) - \theta(0)\}}{t_f^2}$$

$$a_3 = -2 \cdot \frac{\{\theta(t_f) - \theta(0)\}}{t_f^3}$$

Analogamente, na outra coordenada generalizada:

$$\alpha = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (4.46)$$

Considerando que: $\dot{\alpha}(0) = \dot{\alpha}(t_f) = 0$, então os coeficientes valem:

$$a_0 = \alpha(0)$$

$$a_1 = 0$$

$$a_2 = 3 \cdot \frac{\{\alpha(t_f) - \alpha(0)\}}{t_f^2} \qquad a_3 = -2 \cdot \frac{\{\alpha(t_f) - \alpha(0)\}}{t_f^3}$$

Lembrando que o intervalo de tempo t_f deve ser previamente definido.

Com as equações 4.45 e 4.46 da trajetória já definidas e introduzidas nas equações de movimento 4.27 e 4.28, pode ser obtida a história de controle.

4.2.4.2. Trajetória em linha reta

A cabeça do manipulador descreve uma trajetória linear entre os pontos de coordenadas generalizadas $\theta(0), \alpha(0), \theta(t_f)$ e $\alpha(t_f)$ com um tempo de percurso t_f previamente definido. Impõe-se que coordenada generalizada θ segue a equação

$$\theta = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \qquad (4.47)$$

Considerando que: $\dot{\theta}(0) = \dot{\theta}(t_f) = 0$, então os coeficientes valem:

$$a_0 = \theta(0)$$

$$a_1 = 0$$

$$a_2 = 3 \cdot \frac{\{\theta(t_f) - \theta(0)\}}{t_f^2}$$

$$a_3 = -2 \cdot \frac{\{\theta(t_f) - \theta(0)\}}{t_f^3}$$

Da equação (4.40) obtém-se, após manipulação algébrica, que

$$\alpha = \text{asin} \left[\frac{-2.k - \sqrt{\Delta}}{2.(B^2 + 1)} \right] \text{ onde } k \text{ e } \Delta \text{ são expressos por} \quad (4.48)$$

$$k = \frac{L_1 \cdot \text{sen}(\theta) - B.L_1 \cdot \text{cos}(\theta) - L_2 - A}{L_2} \quad (4.49)$$

$$\Delta = 4.k^2 - 4.(B^2 + 1)(k^2 - B^2) \quad (4.50)$$

Introduzindo a equação 4.47 na equação 4.48, obtém-se a equação horária da coordenada generalizada α . O tempo de percurso t_f pode ser ajustado de forma que $\dot{\alpha}(t_f) = 0$.

4.2.4.3. Trajetória de tempo mínimo

No exemplo que será descrito, a posição do manipulador é definida pelas equações:

$$0 \leq t \leq t_s^{\theta} \quad \theta_+(t) = \frac{1}{2} \ddot{\theta}_+ t^2 + \theta_+(0) \quad (4.51)$$

$$t_s^{\theta} \leq t \leq t_f \quad \theta_-(t) = \frac{1}{2} \ddot{\theta}_- t^2 - \ddot{\theta}_- t_f t + \theta_-(t_f) + \frac{1}{2} \ddot{\theta}_- t_f^2 \quad (4.52)$$

$$0 \leq t \leq t_s^{\alpha} \quad \alpha_+(t) = \frac{1}{2} \ddot{\alpha}_+ t^2 + \alpha_+(0) \quad (4.53)$$

$$t_s^\alpha \leq t \leq t_f \quad \alpha_-(t) = \frac{1}{2} \ddot{\alpha}_- t^2 - \ddot{\alpha}_- t_f t + \alpha_-(t_f) + \frac{1}{2} \ddot{\alpha}_- t_f^2 \quad (4.54)$$

Com estes valores, pode-se calcular o “switching time” t_s e o tempo mínimo t_f utilizando as equações 4.51 e 4.52, obtendo-se, para cada movimento do manipulador, as equações:

$$t_f^\theta = \sqrt{2 \cdot \frac{(\ddot{\theta}_- - \ddot{\theta}_+)}{\ddot{\theta}_- \ddot{\theta}_+} \{\theta_-(t_f) - \theta_+(0)\}} \quad t_s^\theta = \frac{\ddot{\theta}_-}{\ddot{\theta}_- - \ddot{\theta}_+} t_f^\theta \quad (4.55)$$

$$t_f^\alpha = \sqrt{2 \cdot \frac{(\ddot{\alpha}_- - \ddot{\alpha}_+)}{\ddot{\alpha}_- \ddot{\alpha}_+} \{\alpha_-(t_f) - \alpha_+(0)\}} \quad t_s^\alpha = \frac{\ddot{\alpha}_-}{\ddot{\alpha}_- - \ddot{\alpha}_+} t_f^\alpha \quad (4.56)$$

É importante observar que os mecanismos devem executar suas tarefas no mesmo intervalo de tempo, o que implica que as acelerações angulares devem manter uma relação determinada entre si, como se pode observar pelas equações:

$$t_f^\theta = t_f^\beta \quad (4.57)$$

$$\frac{(\ddot{\theta}_- - \ddot{\theta}_+)}{\ddot{\theta}_- \ddot{\theta}_+} \{\theta_-(t_f) - \theta_+(0)\} = \frac{(\ddot{\alpha}_- - \ddot{\alpha}_+)}{\ddot{\alpha}_- \ddot{\alpha}_+} \{\alpha_-(t_f) - \alpha_+(0)\} \quad (4.58)$$

$$\frac{(\ddot{\theta}_- - \ddot{\theta}_+)}{\ddot{\theta}_- \ddot{\theta}_+} \cdot \frac{\ddot{\alpha}_- \ddot{\alpha}_+}{(\ddot{\alpha}_- - \ddot{\alpha}_+)} = cte \quad (4.59)$$

$$cte = \frac{\{\alpha_-(t_f) - \alpha_+(0)\}}{\{\theta_-(t_f) - \theta_+(0)\}} \quad (4.60)$$

O sistema de equações 4.57 a 4.60 permite escolher três das quatro acelerações generalizadas, $\ddot{\theta}_+$, $\ddot{\theta}_-$, $\ddot{\alpha}_+$ e $\ddot{\alpha}_-$. A particular escolha destes valores implicitamente define o torque máximo dos motores.

4.2.5. Dinâmica direta

Para verificar se as histórias de controle determinadas na seção 4.2.3 estão corretas, o manipulador e os motores serão modelados em ambiente computacional, com campo da gravidade terrestre, utilizando o programa “Interactive Physics” [60].

A trajetória da cabeça de impressão, a aceleração, a velocidade e a posição dos mecanismos podem ser calculadas numericamente e comparadas com os dados cinemáticos da dinâmica inversa.

4.2.6. Estrutura básica

As barras do manipulador deverão ser esbeltas, sendo que a geometria mais adequada é a tubular de secção retangular fechada com paredes finas. As grandezas mantidas constantes estão relacionadas na tabela 4.3.

Tabela 4.3: Grandezas mantidas constantes.

Comprimentos		Larguras		Espessura
L_1 (m)	L_2 (m)	a (m)	b (m)	e (m)
0,32	0,18	0,0185	0,05	0,0015

O momento de inércia em relação ao centro de massa das barras é:

$$I = \frac{m}{12} \cdot \{L^2 + a^2\} \quad (4.61)$$

Os momentos de inércia em relação ao centro de massa e as massas dos elementos para o Duralumínio estão indicados na tabela 4.4.

Tabela 4.4: Densidades, massas e momentos de inércia.

Material	Densidade (kg/m ³)	m ₁ (kg)	m ₂ (kg)	I ₁ (kg.m ²)	I ₂ (kg.m ²)
Duralumínio	2700	0,16980	0.09550	0,001450	0.00026

5. RESULTADOS

5.1. Dinâmica inversa e direta

Os resultados obtidos pela aplicação do método da dinâmica inversa e da dinâmica direta serão expressos para trajetórias polinomial, retilínea e de tempo mínimo. O material utilizado é o duralumínio. No método da dinâmica inversa define-se a trajetória a ser descrita pela cabeça de impressão que é utilizada na obtenção dos correspondentes torques nos motores. Com estes torques, faz-se uma simulação de dinâmica direta com o programa computacional “Interactive Physics“ [60], comparando-se as trajetórias obtidas nos dois métodos. O detalhamento dos modelos é idêntico e uma aproximação das trajetórias indica que a história de controle foi calculada corretamente. Nestes resultados, que serão apresentados a seguir, faz-se uma comparação dos três tipos de trajetória entre si e com a dinâmica direta.

Em todas as situações, a posição inicial e a posição final desejada do manipulador são as mesmas. O mecanismo manipulador possui, com será mostrado, uma grande versatilidade quanto às trajetórias, velocidades e acelerações. Apesar de ser possível um movimento mais amplo, a área de trabalho da cabeça do manipulador será restrita por um retângulo conforme ilustrado na figura 5.1. O tempo final adotado no controle com trajetória polinomial e no controle com trajetória retilínea é o tempo final obtido no controle de tempo mínimo, para que a comparação de torques máximos faça sentido.

5.1.1. Trajetória polinomial

A tabela 5.1 e figura 5.1 apresentam as posições inicial e final dos mecanismos, juntamente com o tempo decorrido, nos métodos das dinâmicas inversa e direta. As figuras 5.2 a 5.7 mostram a dependência temporal, respectivamente, da posição angular do manipulador, dos torques transmitidos, das velocidades e acelerações dos mecanismos, da velocidade linear da cabeça de impressão, das coordenadas da cabeça de impressão e dos desvios das coordenadas da cabeça de impressão em relação ao método da dinâmica direta. A figura 5.8 ilustra os trechos finais ampliados das trajetórias obtidas pelos métodos das dinâmicas inversa e direta. A tabela 5.2 exibe os torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa.

Tabela 5.1: Coordenadas iniciais e finais da cabeça de impressão – trajetória polinomial

Tempo	Dinâmica	Posição inicial				Posição final			
		$x(0)$ (m)	$y(0)$ (m)	$\theta(0)$ (°)	$\alpha(0)$ (°)	$x(t_f)$ (m)	$y(t_f)$ (m)	$\theta(t_f)$ (°)	$\alpha(t_f)$ (°)
0,6778	Inversa	0,05	0,07	112,86	-14,43	0,250000	0,000000	68,99	-41,27
0,6778	Direta	0,05	0,07	112,86	-14,43	0,250590	0,000190	68,88	-41,26

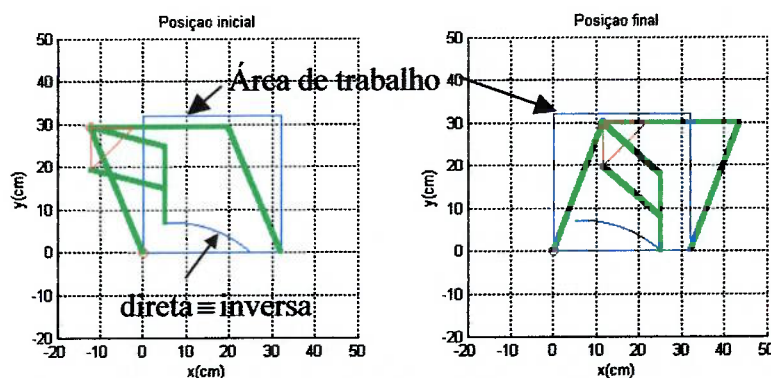


Figura 5.1: Posições inicial e final do manipulador juntamente com a trajetória da cabeça de impressão – trajetória polinomial.

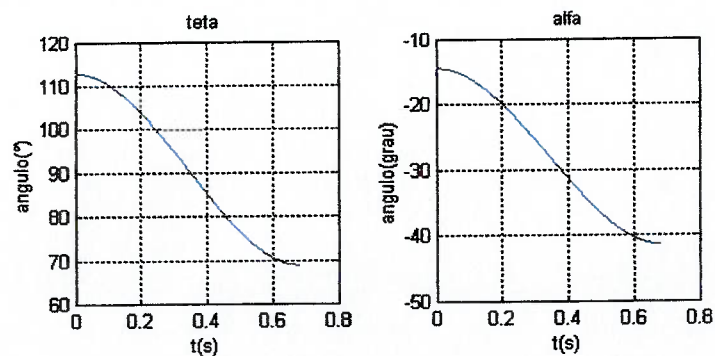


Figura 5.2: Posição angular dos mecanismos em função do tempo – trajetória polinomial. A tangente nos pontos extremos é horizontal, indicando velocidade angular nula.

Tabela 5.2: Torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa – trajetória polinomial.

Trajectoria polinomial	Mecanismo 1		Mecanismo2	
t_f (s)	τ_1 (mN.m) máximo	$\ddot{\theta}$ (rad / s ²) máximo	τ_2 (mN.m) máximo	$\ddot{\alpha}$ (rad / s ²) máximo
0,6778	1082	10	362	6,1

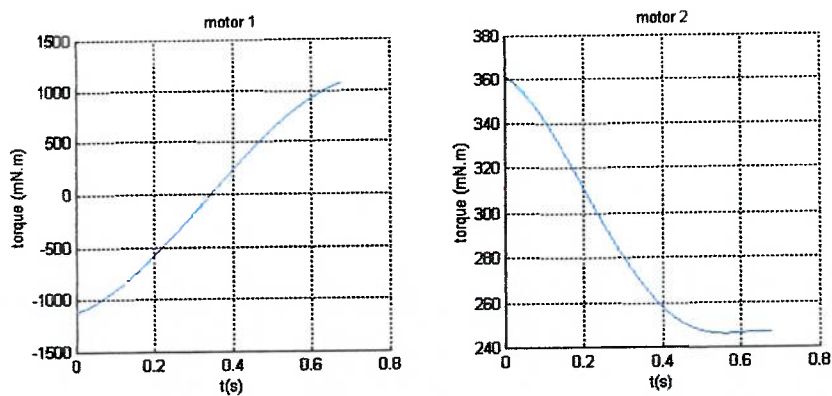


Figura 5.3: Torques transmitidos pelos motores em função do tempo, obtidos pelo método da dinâmica inversa – trajetória polinomial.

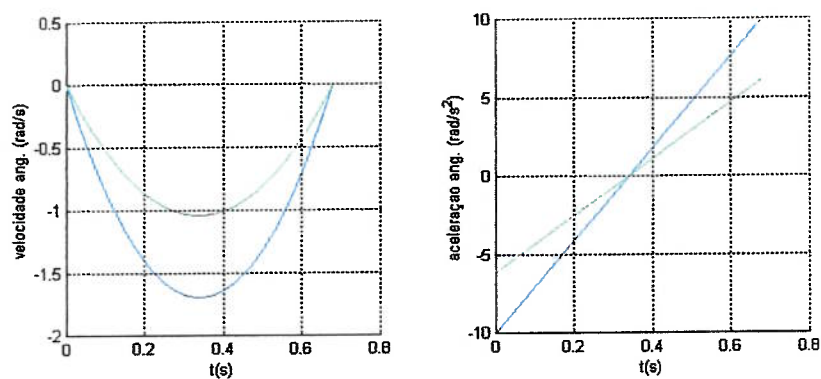


Figura 5.4: Velocidade e aceleração dos mecanismos em função do tempo – trajetória polinomial. Nesta trajetória polinomial cúbica, a aceleração angular varia linearmente com o tempo.

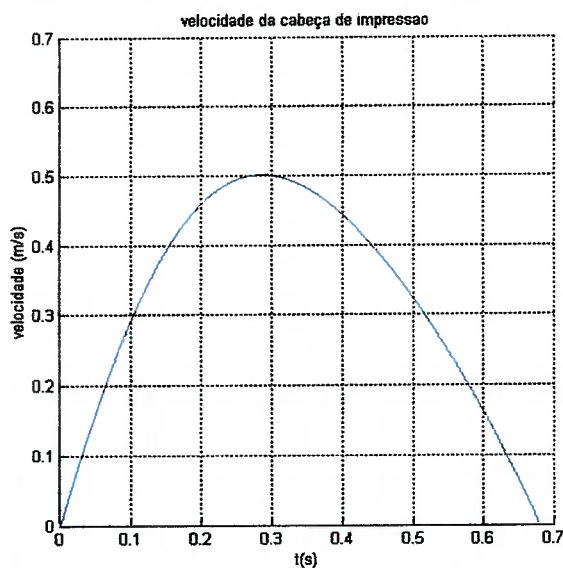


Figura 5.5: Velocidade linear da cabeça de impressão em função do tempo gasto na tarefa – trajetória polinomial. Nos pontos extremos, a velocidade é nula.

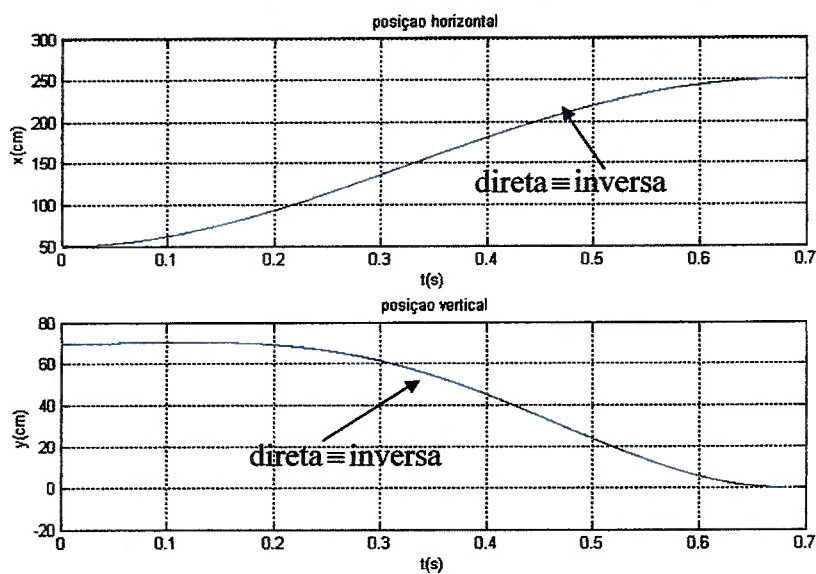


Figura 5.6: Coordenadas da cabeça de impressão, obtidas pelos métodos da dinâmica inversa e direta – trajetória polinomial. Nesta escala, os diagramas são praticamente coincidentes.

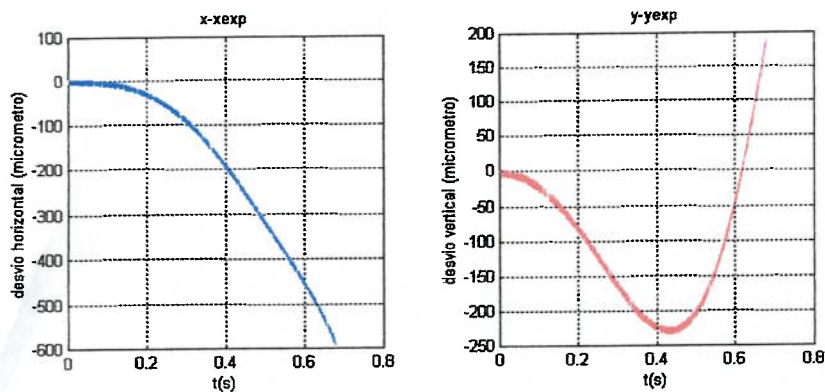


Figura 5.7: Desvio das coordenadas da cabeça de impressão em relação ao método da dinâmica direta – trajetória polinomial.

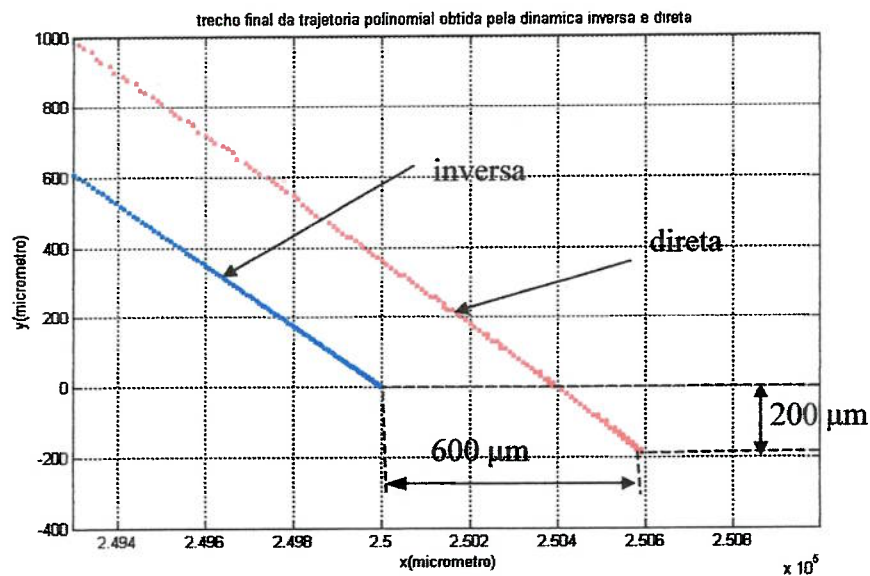


Figura 5.8: Trechos finais ampliados das trajetórias obtidas pelos métodos das dinâmicas inversa e direta – trajetória polinomial.

5.1.2. Trajetória em linha reta

A tabela 5.3 e figura 5.9 apresentam as posições inicial e final dos mecanismos e o tempo decorrido, nos métodos das dinâmicas inversa e direta. As figuras 5.10 a 5.15 mostram a dependência temporal, respectivamente, da posição angular do manipulador, dos torques transmitidos, das velocidades e acelerações dos mecanismos, das coordenadas da cabeça de impressão, da velocidade linear da cabeça de impressão e dos desvios das coordenadas da cabeça de impressão em relação ao método da dinâmica direta. A figura 5.16 ilustra os trechos finais ampliados das trajetórias obtidas pelos métodos das dinâmicas inversa e direta. A figura 5.17 mostra os diagramas de fase dos mecanismos. A tabela 5.4 exibe os torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa.

Tabela 5.3: Coordenadas iniciais e finais da cabeça de impressão – trajetória retilínea.

Tempo (s)	Dinâmica	$x(0)$ (m)	$y(0)$ (m)	$\theta(0)$ (°)	$\alpha(0)$ (°)	$x(t_f)$ (m)	$y(t_f)$ (m)	$\theta(t_f)$ (°)	$\alpha(t_f)$ (°)
0,6778	Inversa	0,05	0,07	112,86	-14,43	0,250000	0,000000	68,99	-41,27
0,6778	Direta	0,05	0,07	112,86	-14,43	0,249792	0,006870	69,87	-39,12

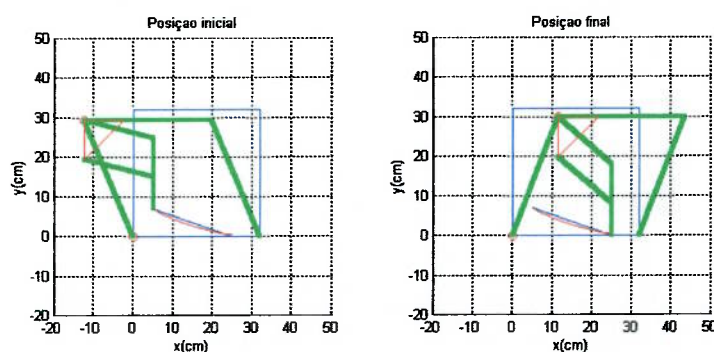


Figura 5.9: Posições inicial e final do manipulador juntamente com a trajetória da cabeça de impressão obtidas pela dinâmica direta e inversa – trajetória retilínea.

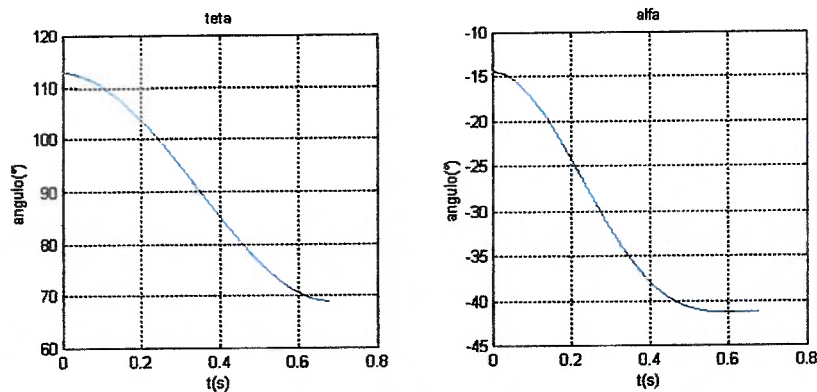


Figura 5.10: Posição angular dos mecanismos em função do tempo – trajetória retilínea. A variável generalizada α foi obtida pela equação 4.48.

Tabela 5.4: Torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa – trajetória retilínea.

Trajetória em linha reta	Mecanismo 1		Mecanismo2	
	$\tau_1 (mN.m)$ máximo	$\ddot{\theta} (rad / s^2)$ máximo	$\tau_2 (mN.m)$ máximo	$\ddot{\alpha} (rad / s^2)$ máximo
$t_f (s)$	1118	10,0	325	5,21

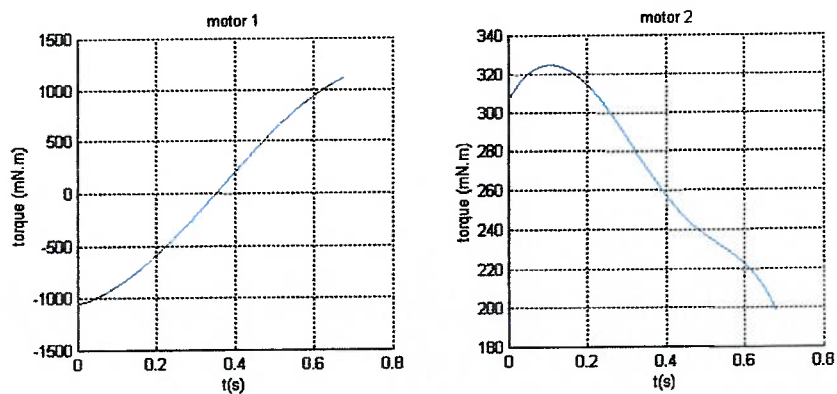


Figura 5.11: Torques transmitidos pelos motores em função do tempo, obtidos pelo método da dinâmica inversa – trajetória retilínea.

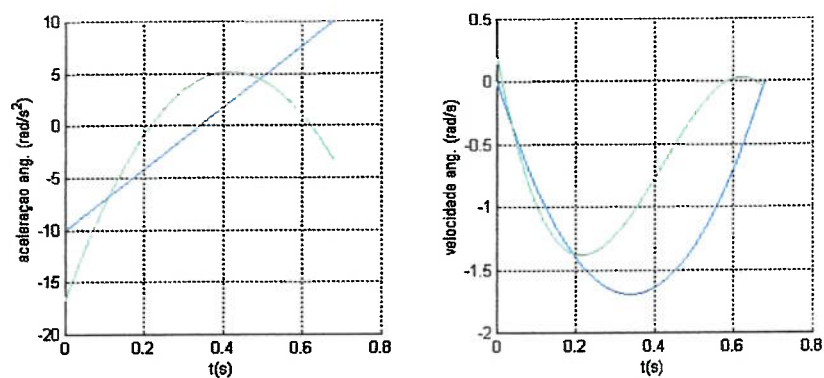


Figura 5.12: Velocidade e aceleração dos mecanismos em função do tempo – trajetória retilínea. A aceleração angular α não varia linearmente com o tempo, visto que esta variável foi obtida pelo ajuste de uma função de quarto grau.

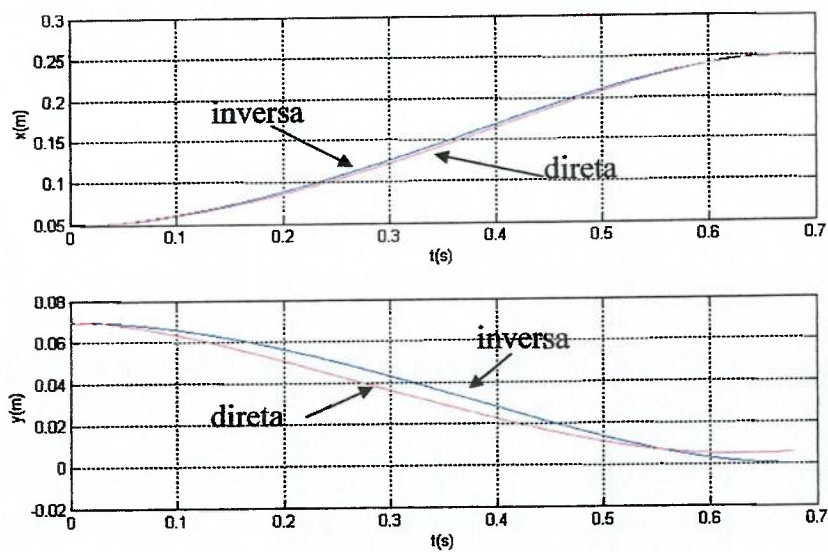


Figura 5.13: Coordenadas da cabeça de impressão em função do tempo – trajetória retilínea. Nesta trajetória linear surgem desvios maiores.

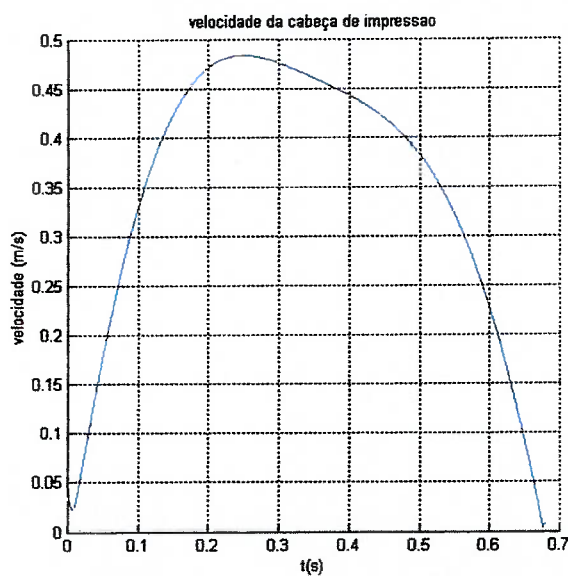


Figura 5.14: Velocidade linear da cabeça de impressão em função do tempo – trajetória retilínea.

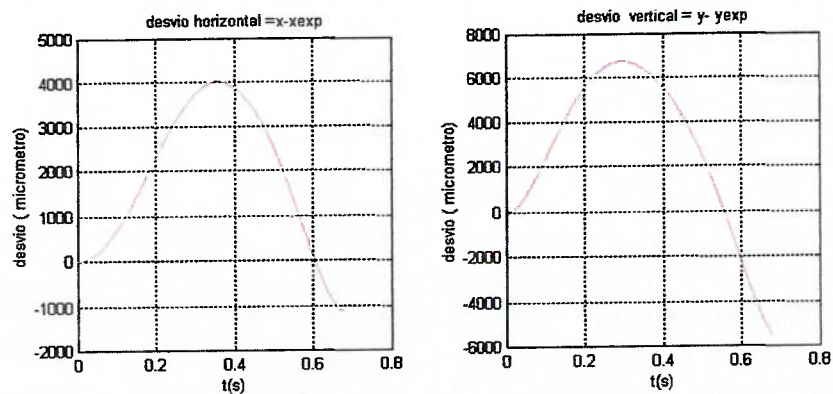


Figura 5.15: Desvio das coordenadas da cabeça de impressão em relação ao método da dinâmica direta – trajetória retilínea.

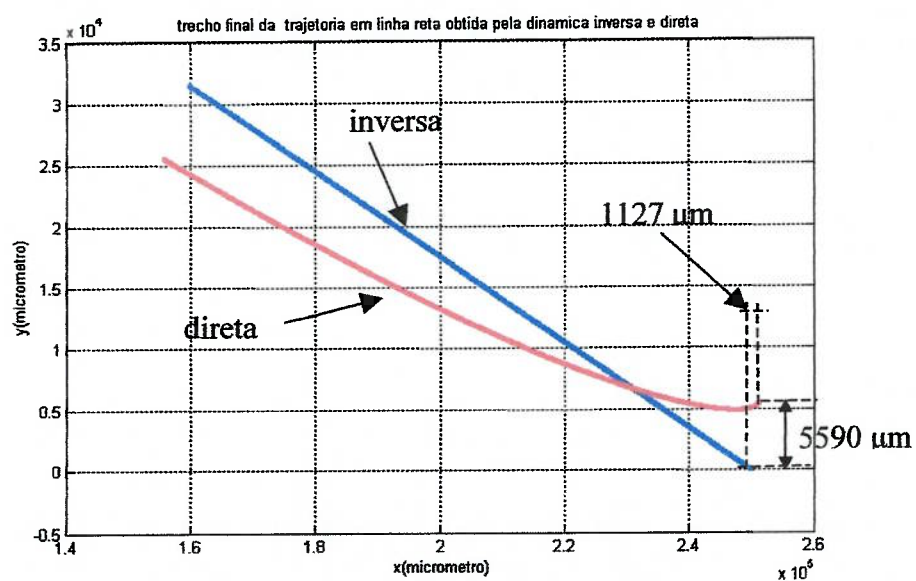


Figura 5.16: Trechos finais ampliados das trajetórias obtidas pelo método da dinâmica inversa e direta – trajetória retilínea.

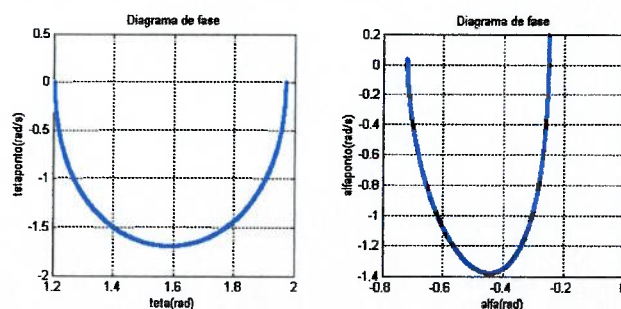


Figura 5.17: Diagramas de fase dos mecanismos – trajetória retilínea.

5.1.3. Trajetória de tempo mínimo

A tabela 5.5 e figura 5.18 apresentam as posições inicial e final dos mecanismos e o tempo decorrido, nas dinâmicas inversa e direta. As figuras 5.19, 5.20, 5.21, 5.23, 5.24 e 5.26 mostram a dependência temporal, respectivamente, da posição angular do manipulador, dos torques transmitidos, das velocidades e acelerações dos mecanismos, das coordenadas da cabeça de impressão, dos desvios das coordenadas da cabeça de impressão em relação ao método da dinâmica direta e da velocidade linear da cabeça de impressão. A figura 5.25 ilustra os trechos finais ampliados das trajetórias obtidas pelos métodos das dinâmicas inversa e direta. A figura 5.22 mostra os diagramas de fase dos mecanismos. A tabela 5.6 exibe os torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa.

Tabela 5.5: Coordenadas iniciais e finais da cabeça de impressão – tempo mínimo.

Tempo (s)	Dinâmica	$x(0)$ (m)	$y(0)$ (m)	$\theta(0)$ (°)	$\alpha(0)$ (°)	$x(t_f)$ (m)	$y(t_f)$ (m)	$\theta(t_f)$ (°)	$\alpha(t_f)$ (°)
0,6778	Inversa	0,05	0,07	112,86	-14,43	0,250000	0,000000	68,99	-41,27
0,6778	Direta	0,05	0,07	112,86	-14,43	0,249799	0,000096	69,03	-41,27

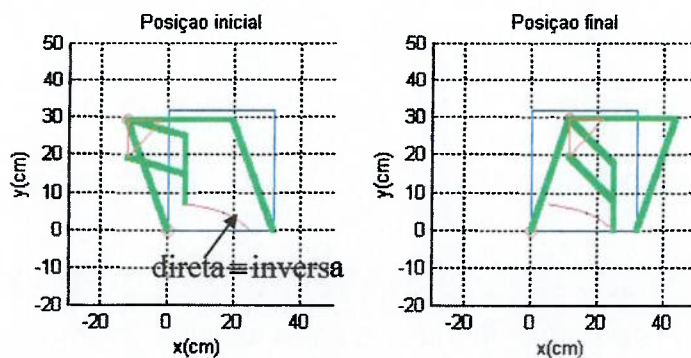


Figura 5.18: Posições inicial e final do manipulador juntamente com a trajetória da cabeça de impressão – trajetória de tempo mínimo. Observa-se que nesta escala as trajetórias dos métodos da dinâmica inversa e direta “coincidem”.

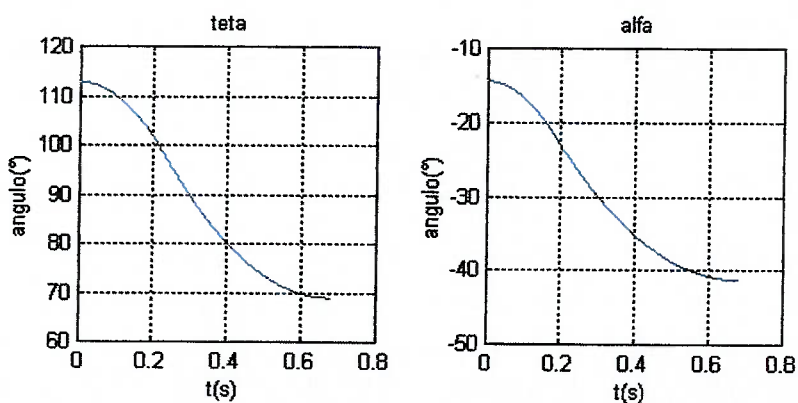


Figura 5.19: Posição angular dos mecanismos em função do tempo – trajetória de tempo mínimo. Observar a tangência nos pontos extremos.

Tabela 5.6: Torques e acelerações angulares máximos transmitidos pelos motores e o tempo gasto na tarefa – trajetória de tempo mínimo.

Trajectoria de tempo mínimo	Mecanismo 1		Mecanismo2	
t_f (s)	τ_1 (mN.m) máximo	$\ddot{\theta}$ (rad / s ²) máximo	τ_2 (mN.m) máximo	$\ddot{\alpha}$ (rad / s ²) máximo
0,6778	847	5,00	366	2,74

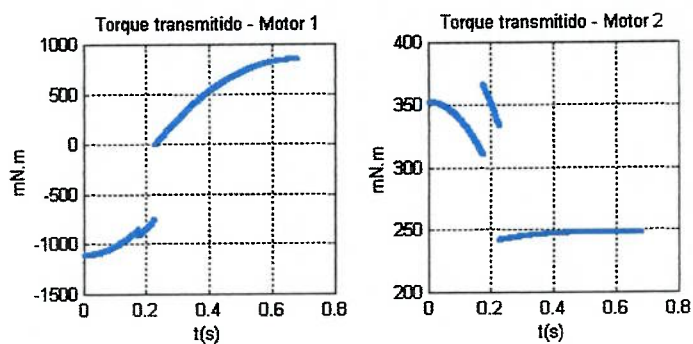


Figura 5.20: Torques transmitidos pelos motores – trajetória de tempo mínimo. As descontinuidades são devidas ao “switching time” dos mecanismos.

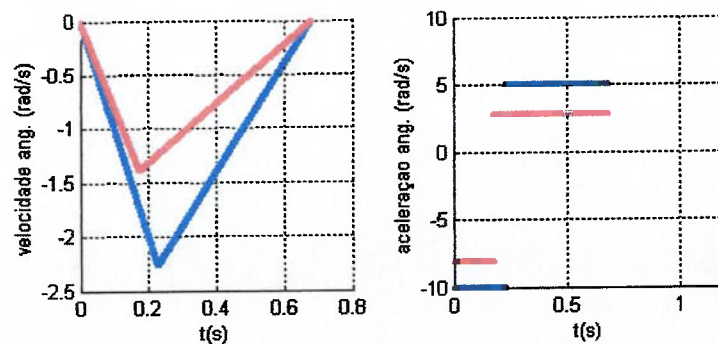


Figura 5.21: Velocidade e aceleração dos mecanismos em função do tempo decorrido na tarefa – trajetória de tempo mínimo. Observar o instante em que ocorre o “switching time” para cada mecanismo.

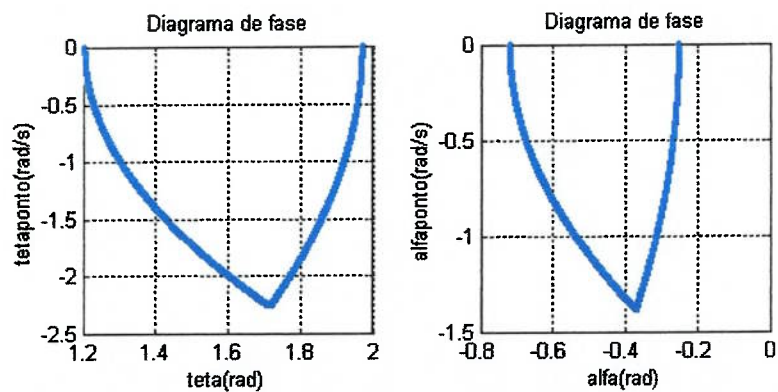


Figura 5.22: Diagramas de fase dos mecanismos – trajetória de tempo mínimo.

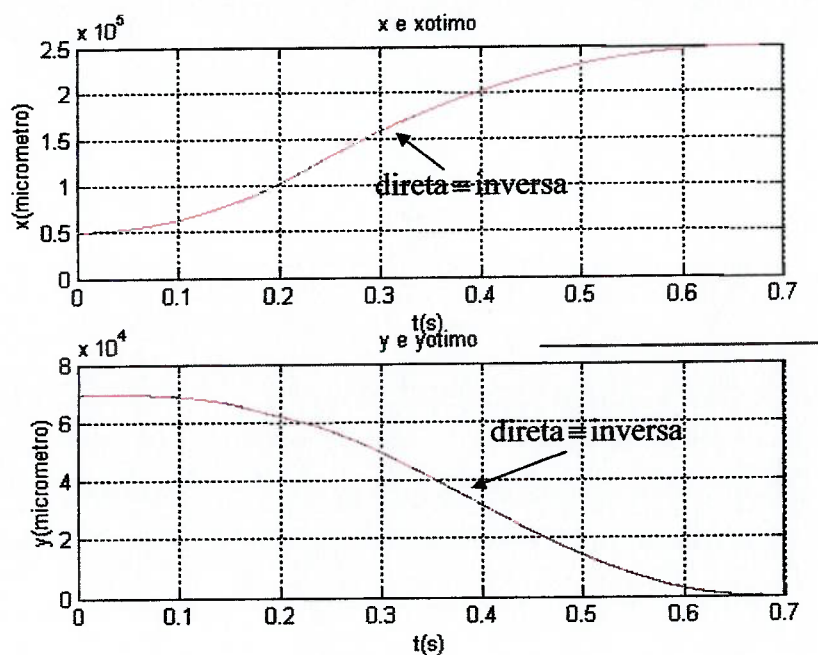


Figura 5.23: Coordenadas da cabeça de impressão – trajetória de tempo mínimo. Os diagramas do método da dinâmica inversa e direta são coincidentes nesta escala.

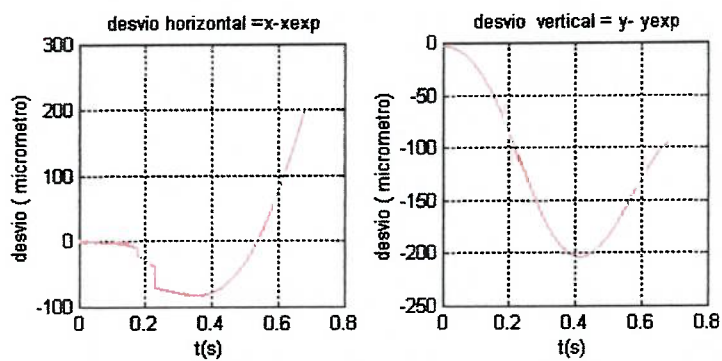


Figura 5.24: Desvios das coordenadas da cabeça de impressão em relação ao método da dinâmica direta – trajetória de tempo mínimo.

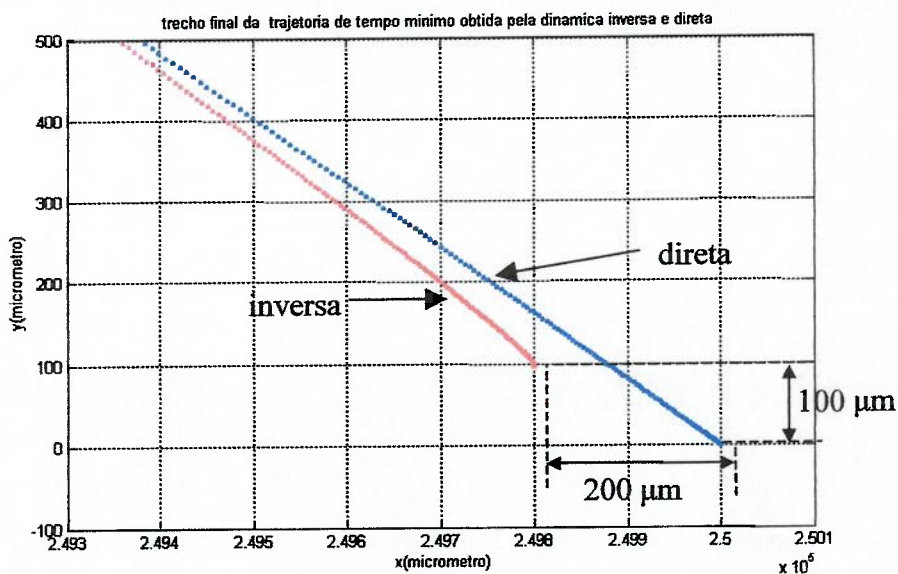


Figura 5.25: Trechos finais ampliados das trajetórias obtidas pelos métodos das dinâmicas inversa e direta – trajetória de tempo mínimo.

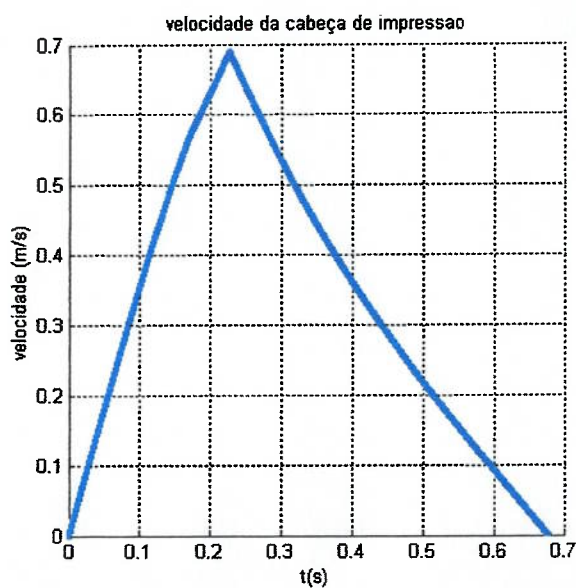


Figura 5.26: Velocidade linear da cabeça de impressão em função do tempo decorrido na tarefa – trajetória de tempo mínimo.

5.2. Comparação de trajetórias

Os tipos de trajetórias analisadas pelos métodos da dinâmica inversa e direta estão nas figuras 5.27 e 5.28 respectivamente. As trajetórias denominadas polinomial e de tempo mínimo apresentam uma tendência de aproximação, apesar de modeladas segundo critérios diferentes. A figura 5.29 mostra a velocidade escalar da “cabeça de impressão” em função do tempo. Na tabela 5.7 faz-se uma comparação entre os valores máximos dos torques e das acelerações angulares, para cada tipo de trajetória, mantendo o tempo de execução da tarefa constante. A tabela 5.8 apresenta a velocidade máxima da “cabeça de impressão” para cada tipo de trajetória. Na tabela 5.9 comparam-se os desvios entre trajetórias nominais e calculadas. Observa-se erro significativo nestes desvios no controle retilíneo e desvios pequenos no controle de tempo mínimo e polinomial. Provavelmente há propagação de erro numérico na determinação de $\alpha(t)$ através da equação 4.40.

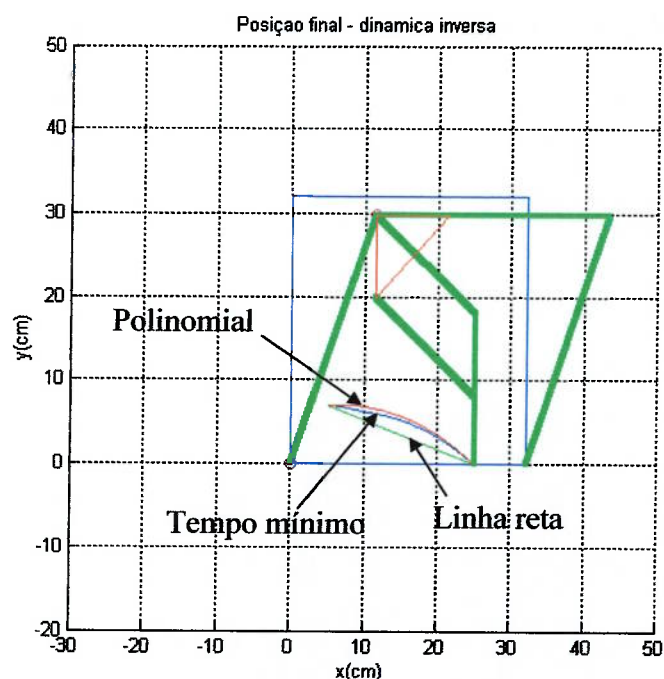


Figura 5.27: Tipos de trajetórias obtidas pelo método da dinâmica inversa.

Tabela 5.7: Comparação entre os valores máximos dos torques e acelerações angulares, mantendo-se o tempo de execução da tarefa constante.

Tempo/tarefa	Mecanismo 1		Mecanismo 2	
	$\tau_1 (mN.m)$ máximo	$\ddot{\theta} (rad/s^2)$ máximo	$\tau_2 (mN.m)$ máximo	$\ddot{\alpha} (rad/s^2)$ máximo
$t_f = 0,6778$ s				
Polinomial	1082	10	362	6,1
Linha reta	1118	10,0	325	5,21
Tempo mínimo	847	5,00	366	2,74

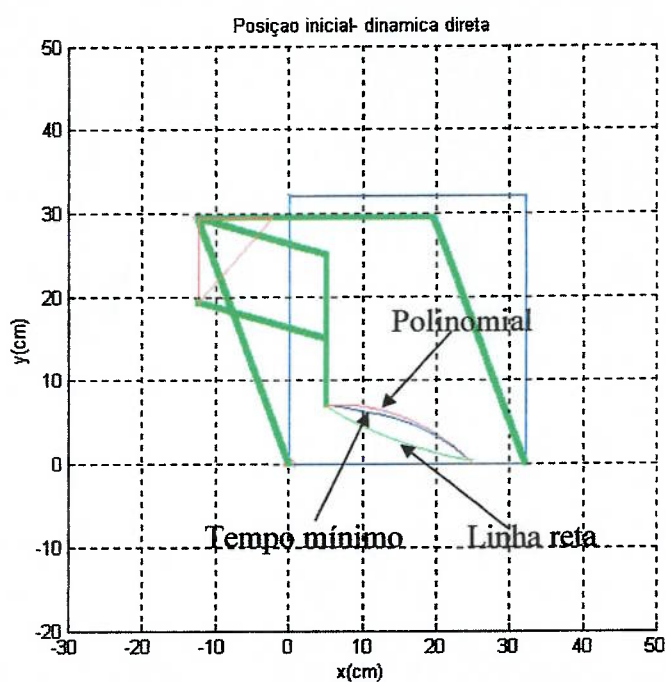


Figura 5.28: Tipos de trajetórias obtidas pelo método da dinâmica direta.

Tabela 5.8: Velocidade máxima da “cabeça de impressão” para cada tipo de trajetória.

Tempo de execução da tarefa: $t = 0,6778$ s	Tempo $t(s)$	Velocidade máxima da “cabeça de impressão” $v_{m\acute{a}xima}(m/s)$
Polinomial	0,285	0,501
Linha reta	0,248	0,484
Tempo mínimo	0,226	0,691

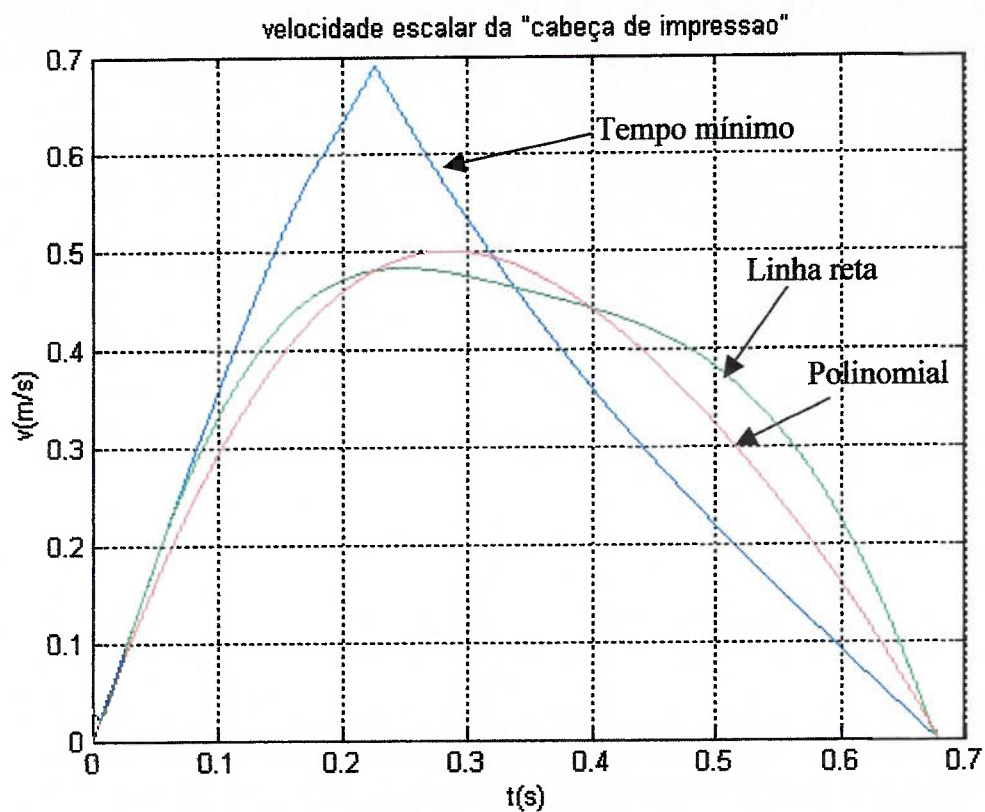


Figura 5.29: Velocidade escalar da “cabeça de impressão” para os tipos de trajetória estudados.

Tabela 5.9: Desvio da trajetória obtida pela dinâmica inversa em relação à obtida pela dinâmica direta.

Tempo de execução da tarefa: $t = 0,6778$ s	Desvio (micrometro) médio		Desvio (micrometro) final	
	horizontal	vertical	horizontal	vertical
Polinomial	187	118	590	190
Linha reta	2066	3880	1127	5590
Tempo mínimo	52	120	201	96

Comparando-se os resultados obtidos com tipos diferentes de controle, verifica-se pela tabela 5.9 que, para as trajetórias de tempo mínimo e polinomial, o desvio final é da mesma ordem de grandeza da distância entre “spots” de um arranjo genético. Destaca-se também que estas trajetórias são muito próximas.

As trajetórias obtidas pela dinâmica inversa e direta apresentam ótima concordância, indicando que não há erros grosseiros de modelagem. Na tabela 5.7, nota-se que o torque máximo transmitido pelo segundo motor é aproximadamente igual para todos os tipos de trajetórias estudadas. Na trajetória de tempo mínimo, o torque máximo transmitido pelo primeiro motor é menor, comparado com as trajetórias polinomial e linha reta. O inverso ocorre com a velocidade máxima da “cabeça de impressão”, conforme a tabela 5.8.

6. CONCLUSÃO

Foi proposto um mecanismo com elevada rigidez e baixa inércia, no qual os atuadores estão localizados na base da pipetadora. Foi desenvolvida uma estratégia de controle, em malha aberta, de tempo mínimo e esta estratégia foi simulada numericamente e seu efeito foi verificado sem levar em consideração a flexibilidade de redutores e o atrito nas articulações, motores e redutores, atestando que a história de controle foi corretamente determinada.

Um controle de malha fechada adicional deve ainda ser implementado para elevar a precisão de posicionamento até 10 μm requeridos pela técnica utilizada na determinação de expressão genética, posto que a história de controle determinada não leva em consideração o atrito, folgas e flexibilidade nos redutores.

7. REFERÊNCIAS BIBLIOGRÁFICAS

1. Lander, E. S.; Linton, L. M.; Birren, B.; Nusbaum, C.; Zody, M. C.; Baldwin, J.; Devon, K.; Dewar, K.; Doyle, M.; et al.; "Initial sequencing and analysis of the human genome", *Nature*, v. 409, p. 860-921, 2001.
2. Sarlie, T.; Perou, C. M., Tibshirani, R.; Aas, T.; Geisler, S.; Johnsen, H.; Hastie, T.; Eisen, M. B.; et al.; "Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications"; *Proc. Natl. Acad. Sci.*, n.98, p. 10869-10874, 2001.
3. Gwynne, P.; Page, G.; "Microarray analysis: the next revolution in molecular biology", *Science*, 1999.
4. Lockhart, D. J.; Winzeler, E. A.; "Genomics, gene expression and DNA arrays"; *Nature*, v. 405, p. 827-836, 2000.
5. Schreiber, S. L.; MacBeath G.; "Printing proteins as microarrays for high-throughput function determination"; *Science*, v. 289, p. 1760-1763, 2000.
6. Fritz, J.; Baller, M. K.; Lang, H. P.; Rothuizen, H.; Vettiger, P.; Meyer, E.; Guntherodt, H.; Gerber, C.; Gimzewski, J. K.; "Translating biomolecular recognition into nanomechanics"; *Science*, v. 288, p. 316-318, 2000.
7. Cortese, J. D.; "Array of options: instrumentation to exploit the DNA microarray explosion", *The Scientist*, v. 14, p. 26, 2000.
8. Cortese, J. D.; "The array of today: biomolecule arrays become the 21 st century's test tube", *The Scientist*, v. 14, p. 26, 2000.
9. Afshari, C. A.; Nuwaysir, E. F.; Barrett, J. C.; "Application of complementary DNA microarray technology to carcinogen identification, toxicology and drug safety evaluation", *Cancer Res.*, v. 59, p. 4759-4760, 1999.
10. Baldwin, D.; Crane, V.; Rice, D.; "A comparison of gel-based, nylon filter and microarray techniques to detect differential RNA expression in plants", *Curr. Opin. Plant Biol.*, v. 2, p. 96-103, 1999.

11. Taton, T. A.; Mirkin, C. A.; Lethsinger, R. L.; "Scanometric DNA array detection with nanoparticles probes", *Science*, v. 289, p. 1757-1760, 2000.
12. Shoemaker, D. D.; Schadt, E. E.; Armour, Y. D.; Garrett, P.; McDonagh, P. D.; et al; "Experimental annotation of the human genome using microarray technology"; *Nature*, v. 409, n. 6822, p. 922-927, 2001.
13. Kane, M. D.; Jatkoe, T. A.; Stumpf, C. R.; Lu, J.; Thomas, J. D.; Madore, S. J.; "Assessment of the sensivity and specificity of oligonucleotide (50 mer) microarrays"; *Nucleic Acids Res.*, v. 28, n. 22, p. 4552-4557, 2000.
14. Marx; J.; "DNA arrays reveal cancer in its many forms"; *Science*, v. 289, p. 1670-1672, 2000.
15. Pollack, J. R.; Perou, C. M.; Alizadeh, A. A.; Eisen, M. B.; et al.; "Genome-wide analysis of DNA copy-number changes using DNA microarrays", *Nat. Genet*, v. 23, p. 41-46, 1999.
16. Bittner, M. L.; Khan, J.; Saal, L. H.; et al.; "Expression profiling in cancer using DNA microarrays", *Electrophoresis*, v. 20, p. 223-229, 1999.
17. Gerhold, D.; Rushmore, T.; Caskey, C. T.; "DNA chips: promising toys have become powerful tools", *Trends Biochem. Sci*, v. 24, p. 168-173, 1999.
18. Ekins, R.; Chu, F. W.; "Microarrays: their origins and applicattions", *Trends in Biotechnology*, v. 17, p. 217-218, 1999.
19. Schena, M.; Davis, R. W.; "Genes, Genomes and Chips. DNA microarrays: a practical approach", Oxford University Press, Oxford, UK, 1999.
20. Schena, M.; Davis, R. W.; "Parallel analysis with biological chips", Academic Press, San Diego, 1998.
21. Service, R. F.; "Microchip arrays put DNA on the spot", *Science*, v. 282, p. 396-399, 1998.
22. Service, R. F.; "Coming soon: the pocket DNA sequencer", *Science*, v. 282, p. 399-401, 1998.
23. Kricka, L.; "Revolution on a square centimeter", *Nature Biotechnology*, v. 16, p. 513, 1998.

24. Ramsay, G.; "DNA chips states of the art", *Nature Biotechnology*, v. 16, p. 40-44, 1998.
25. Marshall, A.; Hodgson, J.; "DNA chips: an array of possibilities", *Nature Biotechnology*, v. 16, p. 27-31, 1998.
26. Blanchard, A. P.; "Genetic Engineering", v. 20, p. 11-123, Plenum Press, New York, 1998.
27. Shalon, D.; Smith, S. J.; Brown, P. O.; "A DNA microarray system for analyzing complex DNA samples using two-color fluorescent probe hybridization", *Genome Res.*, v. 6, p. 639-645, 1996.
28. Schena, M.; Shalon, D.; Davis, R. W.; Brown, P. O., "Quantitative monitoring of gene expression patterns with a complementary DNA microarray", *Science*, v. 270, p. 467-470, 1995.
29. Affymetrix, Inc. Santa Clara, California, USA.
30. www.stanford.edu.
31. Wang, B. B.; Yue, H.; Eastman, P. S.; et al; "An evaluation of the performance of DNA microarrays for detecting changes in global RNA expression", *Nucleic Acid Res.*, 29, E41, 2001.
32. Zammattéo, N.; Jeanmart, L.; Hamels, S.; et al.; "Comparison between different strategies of covalent attachment of DNA to glass surfaces to build DNA microarrays", *Anal. Biochem.*, 280, p. 143-150, 2000.
33. Call, D. R.; Chandler, D. P., Brockman, F.; "Fabrication of DNA microarrays using unmodified oligonucleotide probes", *Bio Techniques*, v. 30, p. 368-376, 2001.
34. Rogers, Y. H., Bogdanov, V., Anderson, R.; et al; "Immobilization of oligonucleotides onto a glass support via disulfide bonds: a method for preparation of DNA microarrays", *Anal. Biochem.*, 266, p. 23-30, 1999.
35. Hedge, P.; Qi, R.; Gay, C.; et al.; "A concise guide to DNA microarray analysis", *Bio Techniques*, 29, 548-556, 2000.

36. Cass, A. E.; Halliwell, C. M.; "A factorial analysis of sinalization conditions for the immobilization of oligonucleotides on glass surfaces", *Anal. Chem.*, 73, p. 2476-2483, 2001.
37. Beier, M.; Diehl, F.; Grahlmann, S.; et al.; "Manufacturing DNA microarrays of high spot homogeneity and reduced background signal", *Nucleic Acids Res.*, 29, E38, 2001.
38. Holloway, A.; Murphy, M.; Massimi, A.; et al.; "Care and cleaning of quill pins".
39. Hamill, B. J. "Apparatus for the chemical synthesis of oligonucleotides", *USP*, 4, 728, 502.
40. Maskos, U.; Southern, E. M.; "Parallel analysis of oligodeoxyribonucleotide (oligonucleotide) interactions. I. Analysis of factors influencing oligonucleotide duplex formation", *Nucleic Acids Res.*, v. 20, p. 1675-1678, 1992.
41. Fodor, S. P., Read, J. L.; Pirrung, M. c. et al. "Light-direct, spatially addressable parallel chemical synthesis", *Science*, v. 251, p. 767-773, 1991.
42. Pease, A. C.; Solas, D. et al, "Light-generated oligonucleotide arrays for rapid DNA sequence analysis", *Proc. Natl. Acad. Sci. USA*, v. 91, p. 5022-5026, 1994.
43. McGall, G. H.; Barone, D. et al, "The efficiency of light-directed synthesis of DNA arrays on glass substrates", *J. Am. Chem. Soc.*, v. 119, p. 5081-5090, 1997.
44. Fodor, S. P.; Stryer, L.; et al. "Arrays of materials attached to a substrate", *USP*, 5, 744, 305.
45. Schena, M. et al.; "Microarrays: biotechnology's discovery platform for functional genomics", *Trends in Biotechnoly*, v. 16, p. 301-306, 1998.
46. Schena, M.; "Microarray biochip technology", Eaton Publishing, 2000.
47. Luo, Z.; Geschwind, D. H.; "Microarray applications in neuroscience", *Neurobiology of Disease*, v. 8, p. 183-193, 2001.
48. Emmert-Buck, M. R. et al; "Laser capture microdissection"; *Science*, v. 274, p. 998-1001, 1996.
49. Cortese, J. D.; "Microscopy paraphernalia", *The Scientist*, 14:26, 2000.

50. Ramdas, L. et al. "Comparative evaluation of laser-based microarray scanners", *Bio Techniques*, v. 31, p. 546-552, 2001.
51. Waddell, E. et al. "High-resolution near-infrared imaging of DNA microarrays with time-resolved acquisition of fluorescence lifetimes", *Analytical Chemistry*, v. 72, p. 5907-5917, 2000.
52. Yguerabide, J. et al, "Light-scattering submicroscopic particles as highly fluorescent analogs and their use as a tracer labels in clinical and biological applications", *Analytical Biochemistry*, v. 262, p. 137-156, 1998.
53. Goldstein, H.; "Classical Mechanics", Wesley Series in Physics, 1980.
54. Barroso, W.; "La mécanique de Lagrange: principes e méthodes", Editions Karthala, Paris, 1994.
55. Chow, T. L.; "Classical Mechanics", John Wesley and Sons, 1995.
56. Thornton, S. T.; Marion, J. B.; "Classical dynamics of particles and systems", Harcourt Brace Company, 1995.
57. Asada, H.; Slotine, J. E.; "Robot analysis and control", John Wesley and Sons, 1986.
58. Mabie, H. H.; Reinholtz, C. F.; "Mechanisms and dynamics of machinery", John Wesley and Sons, 1987.
59. Spong, M. W.; Vidyasagar, M.; "Robot dynamics and control", John Wesley and Sons, 1989.
60. MSC.Software Corporation, California, 2000.

ANEXO 1

Dinâmica dos manipuladores

A análise do comportamento dinâmico de mecanismos manipuladores permite que o sistema seja descrito em termos da variação com o tempo de sua configuração em função dos torques exercidos pelos atuadores.

Essa descrição é expressa por um conjunto de equações diferenciais chamadas de equações do movimento que governa a resposta dinâmica do mecanismo manipulador. Dois métodos podem ser usados para obter as equações de movimento: o método de Newton-Euler e a formulação Lagrangeana.

Na formulação de Newton-Euler, as equações de movimento são derivadas dos teoremas do centro de massa (TCM) e do teorema do momento angular (TMA) que relacionam força e momento, além de torque e momento angular. Essa análise resulta em equações que envolvem forças de acoplamento e torques entre as partes do sistema, assim como forças de fronteira, que podem ser eliminadas, de forma a obter equações dinâmicas na forma fechada.

Nesta formulação, as equações não são expressas em termos de variáveis independentes e não incluem a aplicação de torques nas juntas explicitamente. São necessárias também operações algébricas para obter as equações dinâmicas na forma fechada, o que envolve um procedimento bastante complexo que requer intuição física.

Uma alternativa para a formulação de Newton-Euler aplicada a manipuladores dinâmicos é a formulação Lagrangeana, que descreve o comportamento de sistemas dinâmicos em termos de trabalho e energia utilizando coordenadas generalizadas.

Nesse caso, todas as forças que não trabalham e as forças de fronteira envolvidas no sistema são automaticamente eliminadas. As equações resultantes são compactas e expressas em termos de torques e deslocamentos nas articulações e podem ser derivadas sistematicamente em qualquer sistema de coordenadas.

No presente trabalho foi utilizado o método de Lagrange.

Formulação lagrangeana para manipuladores dinâmicos

Sejam $q_1, \dots, q_i, \dots, q_n$ as coordenadas generalizadas que localizam completamente o sistema dinâmico, sendo q_i uma matriz 3×1 . Sejam T e U respectivamente a energia cinética e energia potencial armazenada no sistema dinâmico. Define-se a Lagrangeana L por [53-56]:

$$L(q_i, \dot{q}_i) = T - U \quad (1)$$

Note que, desde que a energia cinética e a energia potencial são funções de q_i e \dot{q}_i ($i=1, \dots, n$), a Lagrangeana também o é. Usando a Lagrangeana, obtém-se as equações de movimento do sistema dinâmico representadas por:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i \quad (i=1, \dots, n) \quad (2)$$

Onde Q_i é a força generalizada correspondente a coordenada generalizada q_i .

A força generalizada pode ser identificada considerando o trabalho virtual produzido pelas forças não conservativas que atuam no sistema.

Energia cinética de um manipulador

A energia cinética de um braço do mecanismo relativamente ao laboratório que é considerado um referencial inercial é:

$$T_i = \frac{1}{2} \cdot m_i \vec{v}_i^T \cdot \vec{v}_i + \frac{1}{2} \cdot \vec{\omega}_i^T \cdot J_i \cdot \vec{\omega}_i \quad (3)$$

Na equação (2.3) \vec{v}_{ci} e $\vec{\omega}_i$ são matrizes 3×1 da velocidade do centro de massa e da velocidade angular, respectivamente, de um braço do mecanismo no referencial inercial expressos respectivamente por:

$$\vec{v}_{ci} = J_L^i \cdot \dot{\vec{q}} \quad (4)$$

$$\vec{\omega}_i = J_A^i \cdot \dot{\vec{q}} \quad (5)$$

Onde J_L^i e J_A^i são, respectivamente, as matrizes de transformação para as velocidades linear e angular, respectivamente, de um braço.

Cada braço possui massa m_i e momento de inércia I_i , representado no referencial inercial centrado no centro de massa do braço.

O primeiro termo da equação (2.3) representa a energia cinética de translação de um braço e o segundo termo sua energia cinética de rotação relativamente ao seu centro de massa.

A energia cinética total do sistema é dada por:

$$T = \sum_{i=1}^n T_i \quad (6)$$

Substituindo-se a equação (2.3) na equação (2.4) obtém-se:

$$T = \frac{1}{2} \sum_i^n (m_i \dot{\vec{q}}^T \cdot J_L^{(i)T} \cdot J_L^{(i)} \cdot \dot{\vec{q}} + \dot{\vec{q}}^T \cdot J_A^{(i)T} \cdot I_i \cdot J_A^{(i)} \cdot \dot{\vec{q}}) = \frac{1}{2} \dot{\vec{q}} \cdot H \cdot \dot{\vec{q}}^T \quad (7)$$

Onde H é uma matriz que representa a inércia do manipulador dinâmico, determinado pela equação:

$$H = \sum_{i=1}^n (m_i \cdot (J_L^i)^T \cdot J_L^i + (J_A^i)^T \cdot I_i \cdot J_A^i) \quad (8)$$

A matriz H incorpora propriedades de todo o mecanismo e envolve matrizes jacobianas que variam com a configuração dos braços do mecanismo.

Seja H_{ij} as componentes da matriz H. A equação (5) pode ser reescrita na forma escalar obtendo-se:

$$T = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n H_{ij} \dot{q}_i \dot{q}_j \quad (9)$$

Equações de movimento de Lagrange

É necessário obter a energia potencial U do manipulador e as forças generalizadas. Seja \vec{g} o vetor 3×1 representando a aceleração da gravidade em relação ao sistema de referencia inercial. A energia potencial do manipulador é dada por:

$$U = \sum_{i=1}^n m_i \cdot \vec{g}^T \cdot \vec{r}_{0,c_i} \quad (8)$$

A posição do centro de massa C_i depende da configuração dos braços e a energia potencial é função de $q_1, \dots, q_i, \dots, q_n$.

Utilizando a energia cinética total e a energia potencial total obtém-se as equações de movimento de Lagrange. Utilizando as equações (2) e a equação (7) obtém-se:

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_i} \right) = \frac{d}{dt} \left(\sum_{j=1}^n H_{ij} \cdot \dot{q}_j \right) = \sum_{j=1}^n H_{ij} \cdot \ddot{q}_j + \sum_{j=1}^n \frac{dH_{ij}}{dt} \cdot \dot{q}_j \quad (9)$$

Onde H_{ij} é uma função de q_1, \dots, q_n , portanto sua derivada é dada por:

$$\frac{d}{dt} H_{ij} = \sum_{k=1}^n \frac{\partial H_{ij}}{\partial q_k} \cdot \frac{dq_k}{dt} = \sum_{k=1}^n \frac{\partial H_{ij}}{\partial q_k} \cdot \dot{q}_k \quad (10)$$

O segundo termo da equação (2) inclui termos de derivadas parciais de energia cinética dada por:

$$\frac{\partial T}{\partial q_i} = \frac{\partial}{\partial q_i} \left(\frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n H_{jk} \cdot \dot{q}_j \cdot \dot{q}_k \right) = \frac{1}{2} \sum_{j=1}^n \sum_{k=1}^n \frac{\partial H_{jk}}{\partial q_i} \cdot \dot{q}_j \cdot \dot{q}_k$$

(11)

O termo gravitacional G_i é obtido através das derivadas parciais da energia potencial:

$$G_i = \frac{\partial U}{\partial q_i} = \sum_{j=1}^n m_j \cdot \vec{g}^T \cdot \frac{\partial \vec{r}_{0,cj}}{\partial q_i} = \sum_{j=1}^n m_j \cdot \vec{g}^T \cdot J_{Li}^{(j)}$$

(12)

Desde que as derivadas parciais do vetor posição $\vec{r}_{0,cj}$ em relação a q_i é a mesma que a i -ésima vetor coluna da matriz de transformação $J_{Li}^{(j)}$ e substituindo as equações (9) e (12) na equação (2) obtém-se:

$$\sum_{j=1}^n H_{ij} \cdot \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n h_{ijk} \cdot \dot{q}_j \cdot \dot{q}_k + G_i = Q_i \quad (i=1, \dots, n)$$

(13)

$$\text{Onde } h_{ijk} = \frac{\partial H_{ij}}{\partial q_k} - \frac{1}{2} \cdot \frac{\partial H_{jk}}{\partial q_i}$$

(14)

O primeiro termo da equação (13) representa o torque de inércia, incluindo torques de interação, o segundo termo é devido aos efeitos de Coriolis e centrífugos, e o último termo é devido ao torque gravitacional. É importante observar que os torques de inércia interativos $H_{ij} \cdot \ddot{q}_j$ ($j \neq i$) correspondem a elementos fora da diagonal do tensor de inércia do manipulador e que os torques de Coriolis e centrífugos $h_{ijk} \cdot \dot{q}_j \cdot \dot{q}_k$ aparecem porque a configuração do mecanismo manipulador é dependente no tempo.

ANEXO 2

Os programas MATLAB que foram utilizados para a simulação da dinâmica inversa são descritos a seguir.

Programa 1 – Trajetória polinomial

```

close all
clear all
format long e
load xnaootimo
load ynaootimo
L1=.0185;
L2=L1
c1=.32;
c2=.18;
g=9.81;
bb1=.05;
dens=2700;%kg/m^3
ea= 1.5E-3; %m;
eb=1.5E-3; %m;
A1=2*(bb1-2*eb)*ea+2*L1*eb; % 0<= ea <=a/2 e 0<= eb <= b/2
A2=A1
m1=dens*A1*c1;
m2=dens*A2*c2;
M1=m1;
M1=m1/2;
M2=m2
mG1=2*m1+M1;
LG1=(2*m1*c1/2+M1*c1)/mG1;
IG1=2*m1/12*(c1^2+L1^2)+2*m1*(LG1-c1/2)^2+M1*(c1-LG1)^2;
I1=mG1*LG1^2+IG1
mG2=2*m2+M2;
LG2=(2*m2*c2/2+M2*c2)/mG2;
IG2=2*m2/12*(c2^2+L2^2)+2*m2*(LG2-c2/2)^2+M2*(c2-LG2)^2;
I2=mG2*LG2^2+IG2
x0=0.05;%m
y0=.25;%m
[alfa0,teta0]=solve('.32*cos(teta0)+.18*cos(alfa0)=0.05','.32*sin(teta0)+.18*sin(alfa0)=.25')
teta0=double(teta0(1))
alfa0=double(alfa0(1))
xf=.25;%m
yf=.18;%m
[alfaf,tetaf]=solve('.32*cos(tetaf)+.18*cos(alfaf)=.25','.32*sin(tetaf)+.18*sin(alfaf)=.18')
tetaf=double(tetaf(1))
alfaf=double(alfaf(1))
t0=0;
tf=0.6778;%s
tetaponto0=0;

```

```

tetapontof=0;
alfaponto0=0;
alfapontof=0;
Ateta=[ 1 t0 t0^2 t0^3 ; 0 1 2*t0 3*t0^2 ; 1 tf tf^2 tf^3; 0 1 2*tf 3*tf^2];
uteta= [teta0; tetaponto0; tetaf; tetapontof];
ateta=Aateta\uteta;
a0=ateta(1);
a1=ateta(2);
a2=ateta(3);
a3=ateta(4);
Aalfa=[ 1 t0 t0^2 t0^3 ; 0 1 2*t0 3*t0^2 ; 1 tf tf^2 tf^3; 0 1 2*tf 3*tf^2];
ualfa= [alfa0; alfaponto0; alfaf; alfapontof];
balfa=Aaalfa\ualfa;
b0=balfa(1);
b1=balfa(2);
b2=balfa(3);
b3=balfa(4);
t=0:.0002:tf
teta=a0+a1.*t+a2.*t.^2+a3*t.^3;
tetaponto=a1+2*a2.*t+3*a3*t.^2;
tetapontoponto=2*a2+6*a3*t;
alfa=b0+b1.*t+b2*t.^2+b3*t.^3;
alfaponto=b1+2*b2*t+3*b3*t.^2;
alfapontoponto=2*b2+6*b3*t;
x =c1*cos(teta)+c2*cos(alfa);
y= c1*sin(teta)+c2*sin(alfa)-c2;
vx=-c1*tetaponto.*sin(teta)-c2*alfaponto.*sin(alfa);
vy=-c1*tetaponto.*cos(teta)+c2*alfaponto.*cos(alfa);
v=(vx.^2+vy.^2).^5;
I=find(v==max(v))
vmax=v(I)
T1=(mG1*LG1^2+IG1+mG2*c1^2)*tetapontoponto+mG2*c1*LG2*cos(teta-
alfa).*alfapontoponto+mG2*c1*LG2*sin(teta-
alfa).*alfaponto.^2+g*cos(teta)*(mG1*LG1+mG2*c1)% -k1*abs((teta-
abs(teta(1))))+C1*abs(tetaponto)
T2=mG2*c1*LG2*cos(teta-alfa).*tetapontoponto+(mG2*LG2^2+IG2)*alfapontoponto-
mG2*c1*LG2*sin(teta-alfa).*tetaponto.^2+mG2*g*LG2*cos(alfa)%
-k2*abs((alfa-abs(alfa(1))))+C2*abs(alfaponto)
torque1=[t;T1]
save torq1.dat torque1 -ascii
torque2=[t;T2]
save torq2.dat torque2 -ascii
figure(1)
subplot(1,2,1)
plot(t,teta/pi*180)
axis equal
axis square
xlabel('t(s)')
ylabel('angulo(°)')

```

```
title('teta ')
grid
subplot(1,2,2)
plot(t,alfa/pi*180)
axis equal
axis square
xlabel('t(s)')
ylabel('angulo(grau)')
title('alfa')
grid
figure(2)
subplot(1,2,1)
axis square
hold on
plot(t,T1*1000)
xlabel('t(s)')
ylabel('torque (mN.m)')
title('motor 1')
grid on
subplot(1,2,2)
axis square
hold on
plot(t,T2*1000)
xlabel('t(s)')
ylabel('torque (mN.m)')
title(' motor 2')
grid on
figure(3)
subplot(2,1,1)
plot(t,x*1000,'r',t,xnaootimo(1:length(t))*1000,'b')
grid
xlabel('t(s)')
ylabel('x(cm)')
title(' posição horizontal')
subplot(2,1,2)
plot(t,y*1000,'r',t,ynaootimo(1:length(t))*1000,'b')
grid
xlabel('t(s)')
ylabel('y(cm)')
title(' posição vertical')
teta0grau=teta0/pi*180
alfa0grau=alfa0/pi*180
tetafgrau=tetaf/pi*180
alfafgrau=alfaf/pi*180
L=c1
D=.1
figure(5)
subplot(1,2,1)
hold on
```

```

plot(0*100,0*100,'ro',c1*cos(teta0)*100,c1*sin(teta0)*100,'ro')
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0])
line([0,c1*cos(teta0)*100],[0,c1*sin(teta0)*100],'LineWidth',3,'Color','g')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c1)*100],[(c1*sin(teta0))*100,c1*sin(teta0)*100],'LineWidth',3,'Color','g')
line([(c1*cos(teta0)+c1)*100,(c1)*100],[(c1*sin(teta0))*100,(0)*100],'LineWidth',3,'Color','g')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0))*100,(c1*sin(teta0)+c2*sin(alfa0))*100],'LineWidth',3,'Color','g')
line([(c1*cos(teta0)+c2*cos(alfa0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)+c2*sin(alfa0))*100,(c1*sin(teta0)+c2*sin(alfa0)-c2)*100],'LineWidth',3,'Color','g')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0)+c2*sin(alfa0)-D)*100],'LineWidth',3,'Color','g')
line([(c1*cos(teta0))*100,(c1*cos(teta0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0))*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r')
xlabel('x(cm)')
ylabel('y(cm)')
title('Posição inicial')
axis equal
axis square
plot(x*100,y*100,'r',xnaootimo*100,ynaootimo*100,'b')
axis([-20 50 -20 50])
grid on
subplot(1,2,2)
hold on
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0])
plot(0*100,0*100,'ro',c1*cos(tetaf)*100,c1*sin(tetaf)*100,'ro')
line([0,c1*cos(tetaf)*100],[0,c1*sin(tetaf)*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c1)*100],[(c1*sin(tetaf))*100,c1*sin(tetaf)*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf)+c1)*100,(c1)*100],[(c1*sin(tetaf))*100,(0)*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf)+c2*sin(alfaf))*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf)+c2*cos(alfaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)+c2*sin(alfaf))*100,(c1*sin(tetaf)+c2*sin(alfaf)-c2)*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf)+c2*sin(alfaf)-D)*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf))*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r')
xlabel('x(cm)')

```

```

ylabel('y(cm)')
title('Posição final')
axis equal
axis square
axis([-20 50 -20 50])
plot(x*100,y*100,'r',xnaootimo*100,ynaootimo*100,'b')
grid on
figure(6)
subplot(1,2,2)
hold on
axis square
plot(t,tetapontoponto,t,alfapontoponto)
xlabel('t(s)')
ylabel('aceleração ang. (rad/s^2)')
grid on
subplot(1,2,1)
hold on
axis square
plot(t,tetaponto,t,alfaponto)
xlabel('t(s)')
ylabel('velocidade ang. (rad/s)')
grid on
figure(7)
hold on
axis square
plot(t,v)
xlabel('t(s)')
ylabel('velocidade (m/s)')
title('velocidade da cabeça de impressao')
grid on
T1(1)*1000
T1(length(T1))*1000
T2(1)*1000
T2(length(T2))*1000
[T1(1) T1(length(T1))]
[T2(1) T2(length(T2))]
[tetapontoponto(1) tetapontoponto(length(tetapontoponto))]
[alfapontoponto(1) alfapontoponto(length(alfapontoponto))]
max(x'-xnaootimo(1:length(x')))*100
max(y'-ynaootimo(1:length(x')))*100
y=y'
x=x'
figure(8)
subplot(1,2,1)
plot(t,(x-xnaootimo)*1e6,'b')
axis equal
axis square
xlabel('t(s)')
ylabel('desvio horizontal (micrometro)')

```



```
title('x-xexp')
grid
subplot(1,2,2)
plot(t,(y-ynaootimo)*1e6,'r')
axis equal
axis square
xlabel('t(s)')
ylabel('desvio vertical (micrometro)')
title('y-yexp')
grid
n=500
i=length(t)-n:length(t)
figure(9)
plot(x(i)*1000000,y(i)*1000000,'b',xnaootimo(i)*1000000,ynaootimo(i)*1000000,'r')
% axis equal
% axis square
xlabel(' x(micrometro)')
ylabel(' y(micrometro)')
title(' trecho final da trajetoria nao otima obtida pela dinamica inversa e direta ' )
axis([ 2.493*1e5 2.51*1e5 -400 1000])
grid
torquemedio1= sum(abs(T1))/length(T1)
torquemedio2= sum(abs(T2))/length(T2)
torqueinicial1=T1(1)
torquefinal1=T1(length(T1))
torqueinicial2=T2(1)
torquefinal2=T2(length(T2))
tetapontopontoinicial=tetapontoponto(1)
tetapontopontofinal=tetapontoponto(length(tetapontoponto))
alfapontopontoinicial=alfapontoponto(1)
alfapontopontofinal=alfapontoponto(length(alfapontoponto))
tetapontomedio=sum(abs(tetapontoponto))/length(tetapontoponto)
alfapontomedio=sum(abs(alfapontoponto))/length(alfapontoponto)
xnao=x
ynaoo=y
save xnao
save ynao
```

Programa 2 – Trajetória em linha reta

```

close all
clear all
format long e
load xlinhareta
load ylinhareta
load xotimo
load yotimo
load xnaootimo
load ynaootimo
load xreta
load yreta
load xotim
load yotim
load xnao
load ynao
L1=.0185;
L2=L1
c1=.32;
c2=.18;
g=9.81;
bb1=.05;
dens=2700;%kg/m^3
ea= 1.5E-3; %m;
eb=1.5E-3; %m;
A1=2*(bb1-2*eb)*ea+2*L1*eb; % 0<= ea <=a/2 e 0<= eb <= b/2
A2=A1
m1=dens*A1*c1;
m2=dens*A2*c2;
M1=m1;
M1=m1/2;
M2=m2
mG1=2*m1+M1;
LG1=(2*m1*c1/2+M1*c1)/mG1;
IG1=2*m1/12*(c1^2+L1^2)+2*m1*(LG1-c1/2)^2+M1*(c1-LG1)^2;
I1=mG1*LG1^2+IG1
mG2=2*m2+M2;
LG2=(2*m2*c2/2+M2*c2)/mG2;
IG2=2*m2/12*(c2^2+L2^2)+2*m2*(LG2-c2/2)^2+M2*(c2-LG2)^2;
I2=mG2*LG2^2+IG2
x0=0.05;%m
y0=.25;%m
[alfa0,teta0]=solve('0.32*cos(teta0)+.18*cos(alfa0)=0.05','0.32*sin(teta0)+.18*sin(alfa0)=.25')
teta0=double(teta0(1))
alfa0=double(alfa0(1))
xf=.25;%m

```

```

yf=.18;%m
[alfaf,tetaf]=solve('.32*cos(tetaf)+.18*cos(alfaf)=.25','.32*sin(tetaf)+.18*sin(alfaf)=.18')
tetaf=double(tetaf(1))
alfaf=double(alfaf(1))
t0=0;
tf=.6778
tetaponto0=0;
tetapontof=0;
Ateta=[ 1 t0 t0^2 t0^3 ; 0 1 2*t0 3*t0^2 ; 1 tf tf^2 tf^3; 0 1 2*tf 3*tf^2];
uteta= [teta0; tetaponto0; tetaf; tetapontof];
ateta=Ateta\uteta;
a0=ateta(1);
a1=ateta(2);
a2=ateta(3);
a3=ateta(4);
t=0:.0002:tf;
teta=a0+a1.*t+a2.*t.^2+a3*t.^3;
tetaponto=a1+2*a2.*t+3*a3*t.^2;
tetapontoponto=2*a2+6*a3*t;
A=y0-c2-(yf-y0)/(xf-x0)*x0
B=(yf-y0)/(xf-x0)
k1=(c1.*sin(teta)-B*c1.*cos(teta)-c2-A)/c2;
delta=(4*k1.^2-4*(B^2+1).*(k1.^2-B^2)).^.5
alfa=asin((-2*k1-delta)./(2*(B^2+1)))
figure(1)
plot(t,alfa)
axis equal
axis square
xlabel('t(s)')
ylabel('angulo(rad)')
title('alfa')
grid
b0=-.25573
b1=.1926
b2=-8.317
b3=17.43
b4=-10.432
alfaa=b0+b1.*t+b2*t.^2+b3*t.^3+b4*t.^4
alfaponto=b1+2*b2*t+3*b3*t.^2+4*b4*t.^3;
alfapontoponto=2*b2+6*b3.*t+12*b4*t.^2;
x=c1*cos(teta)+c2*cos(alfa);
y=c1*sin(teta)+c2*sin(alfa)-c2;
vx=-c1*tetaponto.*sin(teta)-c2*alfaponto.*sin(alfa);
vy=-c1*tetaponto.*cos(teta)+c2*alfaponto.*cos(alfa);
v=(vx.^2+vy.^2).^5;
I=find(v==max(v))
vmax=v(I)
T1=(mG1*LG1^2+IG1+mG2*c1^2)*tetapontoponto+mG2*c1*LG2*cos(teta-
alfaa).*alfapontoponto+mG2*c1*LG2*sin(teta-

```

```

alfa).*alfaponto.^2+g*cos(teta)*(mG1*LG1+mG2*c1)%
k1*abs((teta-abs(teta(1))))+C1*abs(tetaponto)
T2=mG2*c1*LG2*cos(teta-alfa).*tetapontoponto+(mG2*LG2^2+IG2)*alfapontoponto-
mG2*c1*LG2*sin(teta-alfa).*tetaponto.^2+mG2*g*LG2*cos(alfa)%
-k2*abs((alfa-abs(alfa(1))))+C2*abs(alfaponto)
torque1=[t;T1]'
save tor1.dat torque1 -ascii
torque2=[t;T2]'
save tor2.dat torque2 -ascii
figure(2)
subplot(1,2,1)
plot(t,teta/pi*180)
axis equal
axis square
xlabel('t(s)')
ylabel('angulo(°)')
title('teta')
grid
subplot(1,2,2)
plot(t,alfa/pi*180)
axis equal
axis square
xlabel('t(s)')
ylabel('angulo(°)')
title('alfa')
grid
figure(3)
subplot(1,2,1)
hold on
plot(t,T1*1000)
axis square
xlabel('t(s)')
ylabel('torque (mN.m)')
title('motor 1')
grid on
subplot(1,2,2)
hold on
plot(t,T2*1000)
axis square
xlabel('t(s)')
ylabel('torque (mN.m)')
title(' motor 2')
grid on
figure(4)
subplot(2,1,1)
plot(t,x)
xlabel('t(s)')
ylabel('x(m)')
grid

```

```

subplot(2,1,2)
plot(t,y)
xlabel('t(s)')
ylabel('y(m)')
grid
teta0grau=teta0/pi*180
alfa0grau=alfa0/pi*180
tetafgrau=tetaf/pi*180
alfafgrau=alfaf/pi*180
L=c1
D=.1
figure(5)
subplot(1,2,1)
hold on
plot(0*100,0*100,'ro',c1*cos(teta0)*100,c1*sin(teta0)*100,'ro')
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0])
line([0,c1*cos(teta0)*100],[0,c1*sin(teta0)*100],'LineWidth',3,'Color','g')

line([(c1*cos(teta0))*100,(c1*cos(teta0)+c1)*100],[c1*sin(teta0)*100,c1*sin(teta0)*100],
'LineWidth',3,'Color','g')

line([(c1*cos(teta0)+c1)*100,(c1)*100],[c1*sin(teta0)*100,(0)*100],'LineWidth',3,'Color','g')

line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[c1*sin(teta0)*100,(c1*sin(teta0)+c2*sin(alfa0))*100],
'LineWidth',3,'Color','g')
line([(c1*cos(teta0)+c2*cos(alfa0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[c1*sin(teta0)+c2*sin(alfa0)*100,(c1*sin(teta0)+c2*sin(alfa0)-c2)*100],
'LineWidth',3,'Color','g')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[c1*sin(teta0)-D*100,(c1*sin(teta0)+c2*sin(alfa0)-D)*100],
'LineWidth',3,'Color','g')
line([(c1*cos(teta0))*100,(c1*cos(teta0))*100],[c1*sin(teta0)-D*100,(c1*sin(teta0))*100],
'LineWidth',1,'Color','r')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[c1*sin(teta0)-D*100,(c1*sin(teta0))*100],
'LineWidth',1,'Color','r')

line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[c1*sin(teta0)*100,(c1*sin(teta0))*100],
'LineWidth',1,'Color','r')
xlabel('x(cm)')
ylabel('y(cm)')
title('Posiçao inicial')
axis equal
axis square
plot(x*100,y*100)
axis([-20 50 -20 50])
grid on
subplot(1,2,2)
hold on
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0])
plot(0*100,0*100,'ro',c1*cos(tetaf)*100,c1*sin(tetaf)*100,'ro')

```

```

line([0,c1*cos(tetaf)*100],[0,c1*sin(tetaf)*100],'LineWidth',3,'Color','g')

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c1)*100],[(c1*sin(tetaf))*100,c1*sin(tetaf)*100],
'LineWidth',3,'Color','g')

line([(c1*cos(tetaf)+c1)*100,(c1)*100],[(c1*sin(tetaf))*100,(0)*100],'LineWidth',3,'Color','g')

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf)+c2*sin(alfaf))*100],
'LineWidth',3,'Color','g')
line([(c1*cos(tetaf)+c2*cos(alfaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)+c2*sin(alfaf))*100,(c1*sin(tetaf)+c2*sin(alfaf)-c2)*100],
'LineWidth',3,'Color','g')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf)+c2*sin(alfaf)-D)*100],
'LineWidth',3,'Color','g')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf))*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r')

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r')
xlabel('x(cm)')
ylabel('y(cm)')
title('Posiçao final')
axis equal
axis square
axis([-20 50 -20 50])
plot(x*100,y*100)
grid on
figure(6)
subplot(1,2,1)
hold on
axis square
plot(t,tetapontoponto,t,alfapontoponto)
xlabel('t(s)')
ylabel('aceleração ang. (rad/s^2)')
grid on
subplot(1,2,2)
hold on
axis square
plot(t,tetaponto,t,alfaponto)
xlabel('t(s)')
ylabel('velocidade ang. (rad/s)')
grid on
figure(7)
hold on
axis square
plot(t,v)
xlabel('t(s)')

```

```

ylabel('velocidade (m/s)')
title('velocidade da cabeça de impressao')
grid on
T1(1)*1000
T1(length(T1))*1000
T2(1)*1000
T2(length(T2))*1000
[T1(1) T1(length(T1))]
[T2(1) T2(length(T2))]
[tetapontoponto(1) tetapontoponto(length(tetapontoponto))]
[alfapontoponto(1) alfapontoponto(length(alfapontoponto))]
figure(8)
subplot(1,2,1)
hold on
plot(0*100,0*100,'ro',c1*cos(teta0)*100,c1*sin(teta0)*100,'ro')
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0])
line([0,c1*cos(teta0)*100],[0,c1*sin(teta0)*100],'LineWidth',3,'Color','g')

line([(c1*cos(teta0))*100,(c1*cos(teta0)+c1)*100],[(c1*sin(teta0))*100,c1*sin(teta0)*100],
'LineWidth',3,'Color','g')

line([(c1*cos(teta0)+c1)*100,(c1)*100],[(c1*sin(teta0))*100,(0)*100],'LineWidth',3,'Color','g')

line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0))*100,(c1*sin(teta0)+c2*sin(alfa0))*100],
'LineWidth',3,'Color','g')
line([(c1*cos(teta0)+c2*cos(alfa0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)+c2*sin(alfa0))*100,(c1*sin(teta0)+c2*sin(alfa0)-c2)*100],
'LineWidth',3,'Color','g')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0)+c2*sin(alfa0)-D)*100],
'LineWidth',3,'Color','g')
line([(c1*cos(teta0))*100,(c1*cos(teta0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],
'LineWidth',1,'Color','r')
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],
'LineWidth',1,'Color','r')

line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0))*100,(c1*sin(teta0))*100],
'LineWidth',1,'Color','r')
xlabel('x(cm)')
ylabel('y(cm)')
title('Posição inicial')
axis equal
axis square
plot(x*100,y*100,'b',xlinhareta*100,ylinhareta*100,'r')
axis([-20 50 -20 50])
grid on
subplot(1,2,2)
hold on
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0])
plot(0*100,0*100,'ro',c1*cos(tetaf)*100,c1*sin(tetaf)*100,'ro')

```

```

line([0,c1*cos(tetaf)*100],[0,c1*sin(tetaf)*100],'LineWidth',3,'Color','g')

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c1)*100],[(c1*sin(tetaf))*100,c1*sin(tetaf)*100],
'LineWidth',3,'Color','g')

line([(c1*cos(tetaf)+c1)*100,(c1)*100],[(c1*sin(tetaf))*100,(0)*100],'LineWidth',3,'Color',
'g')

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf))*100,(c1*sin
n(tetaf)+c2*sin(alfaf))*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf)+c2*cos(alfaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(teta
f)+c2*sin(alfaf))*100,(c1*sin(tetaf)+c2*sin(alfaf)-c2)*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf)+c2*sin(alfaf)-D)*100],'LineWidth',3,'Color','g')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf))*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r')
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r')

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf))*10
0],'LineWidth',1,'Color','r')
xlabel('x(cm)')
ylabel('y(cm)')
title('Posição final')
axis equal
axis square
axis([-20 50 -20 50])
plot(x*100,y*100,'b',xlinhareta*100,ylinhareta*100,'r')
grid on
figure(9)
subplot(1,2,1)
plot(teta,tetaponto,'.b')
xlabel('teta(rad)')
ylabel('tetaponto(rad/s)')
title('Diagrama de fase')
axis equal
axis square
grid on
subplot(1,2,2)
plot(alfa,alfaponto,'.b')
xlabel('alfa(rad)')
ylabel('alfaponto(rad/s)')
title('Diagrama de fase')
axis equal
axis square
grid on
load xlinhareta
figure(10)
subplot(1,2,1)

```



```

plot(t,(x'-xlinhareta)*1e6,'r')
axis equal
axis square
grid
xlabel('t(s)')
ylabel('desvio ( micrometro)')
title (' desvio horizontal =x-xexp')
load ylinhareta
subplot(1,2,2)
plot(t,(y'-ylinhareta)*1e6,'r')
axis equal
axis square
grid
xlabel('t(s)')
ylabel('desvio ( micrometro)')
title (' desvio vertical = y- yexp')
n=1500
i=length(t)-n:length(t)
figure(11)
plot(x(i)*1000000,y(i)*1000000,'.b',xlinhareta(i)*1000000,ylinhareta(i)*1000000,'.r')
% axis equal
% axis square
xlabel(' x(micrometro)')
ylabel(' y(micrometro)')
title(' trecho final da trajetoria otima obtida pela dinamica inversa e direta ')
% axis([ 2.493*1e5 2.501*1e5 -100 500])
grid
figure(12)
% subplot(1,2,1)
hold on
plot(0*100,0*100,'ro',c1*cos(teta0)*100,c1*sin(teta0)*100,'ro');
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0]);
line([0,c1*cos(teta0)*100],[0,c1*sin(teta0)*100],'LineWidth',3,'Color','g');

line([(c1*cos(teta0))*100,(c1*cos(teta0)+c1)*100],[c1*sin(teta0)*100,c1*sin(teta0)*100],
'LineWidth',3,'Color','g');

line([(c1*cos(teta0)+c1)*100,(c1)*100],[c1*sin(teta0)*100,(0)*100],'LineWidth',3,'Color','g');

line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[c1*sin(teta0)*100,(c1*
sin(teta0)+c2*sin(alfa0))*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0)+c2*cos(alfa0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[c1*sin(teta0)+c2*sin(alfa0)*100,(c1*sin(teta0)+c2*sin(alfa0)-c2)*100],
'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[c1*sin(teta0)-D)*100,(c1*sin(teta0)+c2*sin(alfa0)-D)*100],
'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0))*100],[c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],
'LineWidth',1,'Color','r');

```

```

line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0)-
D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');

line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0))*100,(c1*sin(teta0))*
100],'LineWidth',1,'Color','r');
xlabel('x(cm)')
ylabel('y(cm)')
title('Posiçao inicial- dinamica direta')
axis equal
axis square
axis([-30 50 -20 50])
plot(xotimo*100,yotimo*100,'b')
hold on
plot(xnaootimo*100,ynaootimo*100,'r')
hold on
plot(xlinhareta*100,ylinhareta*100,'g')
grid on
xreta=x
yreta=y
save xreta
save yreta
figure(13)
% subplot(1,2,2)
hold on
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0]);
plot(0*100,0*100,'ro',c1*cos(tetaf)*100,c1*sin(tetaf)*100,'ro');
line([0,c1*cos(tetaf)*100],[0,c1*sin(tetaf)*100],'LineWidth',3,'Color','g');

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c1)*100],[(c1*sin(tetaf))*100,c1*sin(tetaf)*100
],'LineWidth',3,'Color','g');

line([(c1*cos(tetaf)+c1)*100,(c1)*100],[(c1*sin(tetaf))*100,(0)*100],'LineWidth',3,'Col
or','g');

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf))*100,(c1*si
n(tetaf)+c2*sin(alfaf))*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf)+c2*cos(alfaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(teta
f)+c2*sin(alfaf))*100,(c1*sin(tetaf)+c2*sin(alfaf)-c2)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf)+c2*sin(alfaf)-D)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf))*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r');

line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf))*10
0],'LineWidth',1,'Color','r');
xlabel('x(cm)')
ylabel('y(cm)')

```

```
title('Posição final - dinamica inversa')
axis equal
axis square
axis([-30 50 -20 50])
plot(xotim*100,yotim*100,'b')
hold on
plot(xnao*100,ynao*100,'r')
hold on
plot(xreta*100,yreta*100,'g')
grid on
torqueinicial1=T1(1)
torquefinal1=T1(length(T1))
torqueinicial2=T2(1)
torquefinal2=T2(length(T2))
tetapontopontoinal=tetapontoponto(1)
tetapontopontofinal=tetapontoponto(length(tetapontoponto))
alfapontopontoinal=alfapontoponto(1)
alfapontopontofinal=alfapontoponto(length(alfapontoponto))
tetapontomedio=sum(abs(tetapontoponto))/length(tetapontoponto)
alfapontomedio=sum(abs(alfapontoponto))/length(alfapontoponto)
xreta=x
yreta=y
save xreta
save yreta
```

Programa 3 – Trajetória de tempo mínimo

```

close all
clear all
format long e
load xotimo
load yotimo
load xnaotimo
load ynaotimo
g=9.81;
L1=.0185;
L2=L1
c1=.32;
c2=.18;
bb1=.05;
dens=2700;%kg/m^3   duraluminio
ea= 1.5E-3; %m;
eb=1.5E-3; %m;
A1=2*(bb1-2*eb)*ea+2*L1*eb; %  0<= ea <=a/2   e 0<= eb <= b/2
A2=A1
m1=dens*A1*c1;
m2=dens*A2*c2;
M1=m1;
M1=m1/2
M2=m2;
mG1=2*m1+M1;
LG1=(2*m1*c1/2+M1*c1)/mG1;
IG1=2*m1/12*(c1^2+L1^2)+2*m1*(LG1-c1/2)^2+M1*(c1-LG1)^2;
I1=mG1*LG1^2+IG1;
mG2=2*m2+M2;
LG2=(2*m2*c2/2+M2*c2)/mG2;
IG2=2*m2/12*(c2^2+L2^2)+2*m2*(LG2-c2/2)^2+M2*(c2-LG2)^2;
I2=mG2*LG2^2+IG2;
x0=0.05;%m
y0=.25;%m
[alfa0,teta0]=solve('.32*cos(teta0)+.18*cos(alfa0)=.05','.32*sin(teta0)+.18*sin(alfa0)=.25');
teta0=double(teta0(1));
alfa0=double(alfa0(1));
xf=.25;%m
yf=.18;%m
[alfaf,tetaf]=solve('.32*cos(tetaf)+.18*cos(alfaf)=.25','.32*sin(tetaf)+.18*sin(alfaf)=.18');
;
tetaf=double(tetaf(1));
alfaf=double(alfaf(1));
tetaponto0=0;
tetapontof=0;

```

```

alfaponto0=0;
alfapontof=0;
cte=(alfa0-alfaf)/(teta0-tetaf);
tetapontopontod=-10
tetapontopontoe=5
alfapontopontod=-8
d1=tetapontopontod
e1=tetapontopontoe
d2=alfapontopontod
alfapontopontoe=(d1*d2*e1*cte)/(d1*d2+e1*d1*cte-e1*d2);
e2=alfapontopontoe;
tfteta=(2*(d1-e1)/(e1*d1)*(teta0-tetaf))^.5;
tsteta=e1/(e1-d1)*tfteta;
tfalfa=(2*(d2-e2)/(e2*d2)*(alfa0-alfaf))^.5;
tsalfa=e2/(e2-d2)*tfalfa;
delta=.0002;
if tsalfa<= tsteta
ta=0:delta:tsalfa
tetaa=1/2*d1*ta.^2+teta0;
tetapontoa=d1.*ta;
tetapontopontoa=d1+0.*ta;
alfaa=1/2*d2*ta.^2+alfa0;
alfapontoa=d2.*ta;
alfapontopontoa=d2+0.*ta;
tb=tsalfa:delta:tsteta;
tetab=1/2*d1*tb.^2+teta0;
tetapontob=d1.*tb;
tetapontopontob=d1+0.*tb;
alfab=1/2*e2.*tb.^2+(d2-e2)*tsalfa*tb+alfaf-1/2*e2*tfalfa^2-(d2-e2)*tsalfa*tfalfa;
alfapontob=e2.*tb+(d2-e2)*tsalfa;
alfapontopontob=e2+0.*tb;
tc=tsteta:delta:tfteta;
tetac=1/2*e1.*tc.^2+(d1-e1)*tsteta*tc+tetaf-1/2*e1*tfteta^2-(d1-e1)*tsteta*tfteta;
tetapontoc=e1.*tc+(d1-e1)*tsteta;
tetapontopontoc=e1+0.*tc;
alfac=1/2*e2.*tc.^2+(d2-e2)*tsalfa*tc+alfaf-1/2*e2*tfalfa^2-(d2-e2)*tsalfa*tfalfa;
alfapontoc=e2.*tc+(d2-e2)*tsalfa;
alfapontopontoc=e2+0.*tc;
t=[ta tb tc]';
teta=[tetaa tetab tetac]';
tetaponto=[tetapontoa tetapontob tetapontoc]';
tetapontoponto=[tetapontopontoa tetapontopontob tetapontopontoc]';
alfa=[alfaa alfab alfac]';
alfaponto=[alfapontoa alfapontob alfapontoc]';
alfapontoponto=[alfapontopontoa alfapontopontob alfapontopontoc]';
else
ta=0:delta:tsteta;
tetaa=1/2*d1*ta.^2+teta0;
tetapontoa=d1.*ta;

```

```

tetapontopontoa=d1+0.*ta;
alfaa=1/2*d2*ta.^2+alfa0;
alfapontoa=d2.*ta;
alfapontopontoa=d2+0.*ta;
tb=tsteta:delta:tsalfa;
tetab=1/2*e1.*tb.^2+(d1-e1)*tsteta*tb+tetaf-1/2*e1*tfteta^2-(d1-e1)*tsteta*tfteta;
tetapontob=e1.*tb+(d1-e1)*tsteta;
tetapontopontob=e1+0.*tb;
alfab=1/2*d2*tb.^2+alfa0;
alfapontob=d2.*tb;
alfapontopontob=d2+0.*tb;
tc=tsalfa:delta:tfteta;
tetac=1/2*e1.*tc.^2+(d1-e1)*tsteta*tc+tetaf-1/2*e1*tfteta^2-(d1-e1)*tsteta*tfteta;
tetapontoc=e1.*tc+(d1-e1)*tsteta;
tetapontopontoc=e1+0.*tc;
alfac=1/2*e2.*tc.^2+(d2-e2)*tsalfa*tc+alfaf-1/2*e2*tfalfa^2-(d2-e2)*tsalfa*tfalfa;
alfapontoc=e2.*tc+(d2-e2)*tsalfa;
alfapontopontoc=e2+0.*tc;
t=[ta tb tc]';
teta=[tetaa tetab tetac]';
tetaponto=[tetapontoa tetapontob tetapontoc]';
tetapontoponto=[tetapontopontoa tetapontopontob tetapontopontoc]';
alfa=[alfaa alfab alfaf]';
alfaponto=[alfapontoa alfapontob alfapontoc]';
alfapontoponto=[alfapontopontoa alfapontopontob alfapontopontoc]';
end
T1=(mG1*LG1^2+IG1+mG2*c1^2)*tetapontoponto+mG2*c1*LG2*cos(teta-
alfa).*alfapontoponto+mG2*c1*LG2*sin(teta-
alfa).*alfaponto.^2+g*cos(teta)*(mG1*LG1+mG2*c1) ; % -k1*abs((teta-
abs(teta(1))))+C1*abs(tetaponto)
T2=mG2*c1*LG2*cos(teta-alfa).*tetapontoponto+(mG2*LG2^2+IG2)*alfapontoponto-
mG2*c1*LG2*sin(teta-alfa).*tetaponto.^2+mG2*g*LG2*cos(alfa) ; %
-k2*abs((alfa-abs(alfa(1))))+C2*abs(alfaponto)
% T1e=(mG1*LG1^2+IG1+mG2*c1^2)*tetapontopontoe+mG2*c1*LG2*cos(tetae-
alfae).*alfapontopontoe+mG2*c1*LG2*sin(tetae-
alfae).*alfapontoe.^2+g*cos(tetae)*(mG1*LG1+mG2*c1)% -
k1*abs((teta-abs(teta(1))))+C1*abs(tetaponto)
% T2e=mG2*c1*LG2*cos(tetae-
alfae).*tetapontopontoe+(mG2*LG2^2+IG2)*alfapontopontoe-mG2*c1*LG2*sin(tetae-
alfae).*tetapontoe.^2+mG2*g*LG2*cos(alfae)%
-k2*abs((alfa-abs(alfa(1))))+C2*abs(alfaponto)
momento1=[t,T1]
save momento1.dat momento1 -ascii
momento2=[t,T2]
save momento2.dat momento2 -ascii
x=c1*cos(teta)+c2*cos(alfa);
y= c1*sin(teta)+c2*sin(alfa)-c2;
vx=-c1*tetaponto.*sin(teta)-c2*alfaponto.*sin(alfa);
vy=-c1*tetaponto.*cos(teta)+c2*alfaponto.*cos(alfa);

```

```

v=(vx.^2+vy.^2).^5;
figure(1)
subplot(1,2,1)
plot(t,teta/pi*180,'b')
axis equal
axis square
xlabel('t(s)')
ylabel('angulo(°)')
title('teta')
grid
subplot(1,2,2)
plot(t,alfa/pi*180,'b')
axis equal
axis square
xlabel('t(s)')
ylabel('angulo(°)')
title('alfa')
grid
figure(2)
subplot(2,1,1)
plot(t,x*1000000,'b',t,xotimo*1000000,'r')
xlabel('t(s)')
ylabel('x(micrometro)')
title(' x e xexp ')
grid
subplot(2,1,2)
plot(t,y*1000000,'b',t,yotimo*1000000,'r')
xlabel('t(s)')
ylabel('y(micrometro)')
title(' y e yexp ')
grid
L=c1;%m
D=.1
figure(3)
subplot(1,2,1)
hold on
plot(0*100,0*100,'ro',c1*cos(teta0)*100,c1*sin(teta0)*100,'ro')
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0]);
line([0,c1*cos(teta0)*100],[0,c1*sin(teta0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c1)*100],[(c1*sin(teta0))*100,c1*sin(teta0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0)+c1)*100,(c1)*100],[(c1*sin(teta0))*100,(0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0))*100,(c1*sin(teta0)+c2*sin(alfa0))*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0)+c2*cos(alfa0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)+c2*sin(alfa0))*100,(c1*sin(teta0)+c2*sin(alfa0)-c2)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0)+c2*sin(alfa0)-D)*100],'LineWidth',3,'Color','g');

```

```

line([(c1*cos(teta0))*100,(c1*cos(teta0))*100],[(c1*sin(teta0)-
D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0)-
D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0))*100,(c1*sin(teta0))*
100],'LineWidth',1,'Color','r');
xlabel('x(cm)')
ylabel('y(cm)')
title('Posiçao inicial')
axis equal
axis square
axis([-30 50 -20 50])
plot(x*100,y*100,x*100,y*100,'LineWidth',1,'Color','b')
grid on
subplot(1,2,2)
hold on
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0]);
plot(0*100,0*100,'ro',c1*cos(tetaf)*100,c1*sin(tetaf)*100,'ro');
line([0,c1*cos(tetaf)*100],[0,c1*sin(tetaf)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c1)*100],[(c1*sin(tetaf))*100,c1*sin(tetaf)*100
],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf)+c1)*100,(c1)*100],[(c1*sin(tetaf))*100,(0)*100],'LineWidth',3,'Col
or','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf))*100,(c1*si
n(tetaf)+c2*sin(alfaf))*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf)+c2*cos(alfaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(teta
f)+c2*sin(alfaf))*100,(c1*sin(tetaf)+c2*sin(alfaf)-c2)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf)+c2*sin(alfaf)-D)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf))*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf)-
D)*100,(c1*sin(tetaf))*100],'LineWidth',1,'Color','r');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf))*10
0],'LineWidth',1,'Color','r');
xlabel('x(cm)')
ylabel('y(cm)')
title('Posiçao final')
axis equal
axis square
axis([-30 50 -20 50])
plot(x*100,y*100,x*100,y*100,'LineWidth',1,'Color','b');
grid on
m1
m2;
figure(4)
subplot(1,2,1)
plot(t,T1*1000,'.b')
xlabel('t(s)')

```



```
ylabel('mN.m')
title('Torque transmitido - Motor 1')
axis equal
axis square
grid on
subplot(1,2,2)
plot(t,T2*1000,'.b')
xlabel('t(s)')
ylabel('mN.m')
title('Torque transmitido - Motor 2')
axis equal
axis square
grid on
teta0grau=teta0/pi*180
alfa0grau=alfa0/pi*180
m1;
M1;
m2;
M2;
T1;
T2;
tetafgrau=tetaf/pi*180
alfafgrau=alfaf/pi*180
tfteta
tfalfa
s1='LineWidth',1,'Color','r';
s2='LineWidth',1,'Color','b';
figure(5)
subplot(1,2,1)
hold on
axis square
plot(t,tetaponto,'.b',t,alfaponto,'.r')
xlabel('t(s)')
ylabel('(rad/s)')
title(' velocidade angular')
grid on
subplot(1,2,2)
hold on
axis square
plot(t,tetapontoponto,'.b',t,alfapontoponto,'.r')
axis([0 1.2 -10 10])
xlabel('t(s)')
ylabel('(rad/s^2)')
title(' aceleração angular')
grid on
figure(6)
hold on
axis square
plot(t,v)
```

```

hold on
xlabel('t(s)')
ylabel('velocidade (m/s)')
title('velocidade da cabeça de impressao')
grid on
figure(6)
subplot(1,2,1)
plot(teta,tetaponto,'b')
xlabel('teta(rad)')
ylabel('tetaponto(rad/s)')
title('Diagrama de fase')
axis equal
axis square
grid on
subplot(1,2,2)
plot(alfa,alfaponto,'b')
xlabel('alfa(rad)')
ylabel('alfaponto(rad/s)')
title('Diagrama de fase')
axis equal
axis square
grid on
figure(7)
subplot(1,2,1)
hold on
plot(0*100,0*100,'ro',c1*cos(teta0)*100,c1*sin(teta0)*100,'ro');
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0]);
line([0,c1*cos(teta0)*100],[0,c1*sin(teta0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c1)*100],[(c1*sin(teta0))*100,c1*sin(teta0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0)+c1)*100,(c1)*100],[(c1*sin(teta0))*100,(0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0))*100,(c1*sin(teta0)+c2*sin(alfa0))*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0)+c2*cos(alfa0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)+c2*sin(alfa0))*100],[(c1*sin(teta0)+c2*sin(alfa0)-c2)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0)+c2*sin(alfa0)-D)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0))*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');
xlabel('x(cm)')
ylabel('y(cm)')
title('Posicao inicial')
axis equal
axis square

```

```

axis([-30 50 -20 50])
plot(x*100,y*100,'b',xotimo*100,yotimo*100,'r')
grid on
subplot(1,2,2)
hold on
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0]);
plot(0*100,0*100,'ro',c1*cos(tetaf)*100,c1*sin(tetaf)*100,'ro');
line([0,c1*cos(tetaf)*100],[0,c1*sin(tetaf)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c1)*100],[(c1*sin(tetaf))*100,c1*sin(tetaf)*100],
'LineWidth',3,'Color','g');
line([(c1*cos(tetaf)+c1)*100,(c1)*100],[(c1*sin(tetaf))*100,(0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf)+c2*sin(alfaf))*100],
'LineWidth',3,'Color','g');
line([(c1*cos(tetaf)+c2*cos(alfaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)+c2*sin(alfaf))*100,(c1*sin(tetaf)+c2*sin(alfaf)-c2)*100],
'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf)+c2*sin(alfaf)-D)*100],
'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf))*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf)-D)*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[(c1*sin(tetaf))*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r');
xlabel('x(cm)')
ylabel('y(cm)')
title('Posição final')
axis equal
axis square
axis([-30 50 -20 50])
plot(x*100,y*100,'b',xotimo*100,yotimo*100,'r')
grid on
figure(8)
subplot(1,2,1)
plot(t,(x-xotimo)*1e6,'r')
axis equal
axis square
grid
xlabel('t(s)')
ylabel('desvio ( micrometro)')
title (' desvio horizontal =x-xexp' )
subplot(1,2,2)
plot(t,(y-yotimo)*1e6,'r')
axis equal
axis square
grid
xlabel('t(s)')
ylabel('desvio ( micrometro)')
title (' desvio vertical = y- yexp' )

```

```

n=500
i=length(t)-n:length(t)
figure(9)
plot(x(i)*1000000,y(i)*1000000,'b',xotimo(i)*1000000,yotimo(i)*1000000,'r')
% axis equal
% axis square
xlabel(' x(micrometro)')
ylabel(' y(micrometro)')
title(' trecho final da trajetoria otima obtida pela dinamica inversa e direta ')
axis([ 2.493*1e5 2.501*1e5 -100 500])
grid
figure(10)
subplot(1,2,1)
hold on
plot(0*100,0*100,'ro',c1*cos(teta0)*100,c1*sin(teta0)*100,'ro');
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0]);
line([0,c1*cos(teta0)*100],[0,c1*sin(teta0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c1)*100],[(c1*sin(teta0))*100,c1*sin(teta0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0)+c1)*100,(c1)*100],[(c1*sin(teta0))*100,(0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0))*100,(c1*sin(teta0)+c2*sin(alfa0))*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0)+c2*cos(alfa0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)+c2*sin(alfa0))*100,(c1*sin(teta0)+c2*sin(alfa0)-c2)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+c2*cos(alfa0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0)+c2*sin(alfa0)-D)*100],'LineWidth',3,'Color','g');
line([(c1*cos(teta0))*100,(c1*cos(teta0))*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0)-D)*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');
line([(c1*cos(teta0))*100,(c1*cos(teta0)+D)*100],[(c1*sin(teta0))*100,(c1*sin(teta0))*100],'LineWidth',1,'Color','r');
xlabel('x(cm)')
ylabel('y(cm)')
title('Posição inicial')
axis equal
axis square
axis([-30 50 -20 50])
plot(xotimo*100,yotimo*100,'b')
hold on
plot(xnaootimo*100,ynaootimo*100,'r')
grid on
subplot(1,2,2)
hold on
line([0,0,L*100,L*100,0],[0,c1*100,c1*100,0,0]);
plot(0*100,0*100,'ro',c1*cos(teta0)*100,c1*sin(teta0)*100,'ro');

```

```

line([0,c1*cos(tetaf)*100],[0,c1*sin(tetaf)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c1)*100],[c1*sin(tetaf)*100,c1*sin(tetaf)*100],
'LineWidth',3,'Color','g');
line([(c1*cos(tetaf)+c1)*100,(c1)*100],[c1*sin(tetaf)*100,(0)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[c1*sin(tetaf)*100,(c1*sin(tetaf)+c2*sin(alfaf))*100],
'LineWidth',3,'Color','g');
line([(c1*cos(tetaf)+c2*cos(alfaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[c1*sin(tetaf)+c2*sin(alfaf)*100,
(c1*sin(tetaf)+c2*sin(alfaf)-c2)*100],'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+c2*cos(alfaf))*100],[c1*sin(tetaf)-D)*100,(c1*sin(tetaf)+c2*sin(alfaf)-D)*100],
'LineWidth',3,'Color','g');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf))*100],[c1*sin(tetaf)-D)*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[c1*sin(tetaf)-D)*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r');
line([(c1*cos(tetaf))*100,(c1*cos(tetaf)+D)*100],[c1*sin(tetaf))*100,(c1*sin(tetaf))*100],
'LineWidth',1,'Color','r');
xlabel('x(cm)')
ylabel('y(cm)')
title('Posiçao final')
axis equal
axis square
axis([-30 50 -20 50])
plot(xotimo*100,yotimo*100,'b')
hold on
plot(xnaotimo*100,ynaotimo*100,'r')
grid on
tsteta
tfteta
tsalfa
tsalfa
d1
e1
d2
e2
torqueinicial1=T1(1)
torquefinal1=T1(length(T1))
torqueinicial2=T2(1)
torquefinal2=T2(length(T2))
[(x-xotimo)*1e6 (y-yotimo)*1e6]
torquemedio1= sum(abs(T1))/length(T1)
torquemedio2= sum(abs(T2))/length(T2)
torqueinicial1=T1(1)
torquefinal1=T1(length(T1))
torqueinicial2=T2(1)
torquefinal2=T2(length(T2))
tetapontoponto inicial=tetapontoponto(1)
tetapontoponto final=tetapontoponto(length(tetapontoponto))

```

```
alfapontoponto inicial=alfapontoponto(1)
alfapontoponto final=alfapontoponto(length(alfapontoponto))
tetapontomedio=sum(abs(tetapontoponto))/length(tetapontoponto)
alfapontomedio=sum(abs(alfapontoponto))/length(alfapontoponto)
figure(11)
hold on
axis square
plot(t,v,'b')
xlabel('t(s)')
ylabel('velocidade (m/s)')
title('velocidade da cabeça de impressao')
grid on
```