

Uma Contextualização da Álgebra Linear e dos Quatérnios

Prof.Dr. Claus Haetinger^a

Centro Universitário UNIVATES

Centro de Ciências Exatas e Tecnológicas

Porto Alegre, 28 de Agosto de 2006

^aParcialmente apoiado pelo PPGMAT - UFRGS

Resumo

Trata-se de uma proposta de utilização de um robô do tipo braço mecânico construído a partir de materiais de baixo custo como ferramenta de apoio ao ensino-aprendizado de Matemática a nível de graduação.

Desenvolvemos inicialmente um sistema de controle algébrico bidimensional para os modelos cinemáticos direto e inverso do braço mecânico, facilitando a programação das suas trajetórias, descrevendo as transformações de coordenadas associadas ao movimento de cada articulação do robô.

Posteriormente, mostraremos uma representação não-usual dos quatérnios, simplificando as fórmulas do sistema de controle para os modelos cinemáticos direto e inverso do braço mecânico.

Os quatérnios têm contribuído para pesquisas em computação gráfica e robótica por poderem ser usados no controle de rotações em 3D.

1. Motivação

A automação é cada vez mais presente nas indústrias do Vale do Taquari-RS.

Os cursos da UNIVATES capacitam pessoas a trabalhar nestas indústrias.

Os resultados desta atividade têm servido como subsídio para trabalhos e projetos na área de robótica, visando sua aplicação também no setor industrial da região.

Um dos desafios do educador é tornar o processo ensino-aprendizagem mais significativo.

O mundo fora da sala de aula é muito mais atrativo para o aluno do que as atividades que ele realiza dentro dela, o que tem reduzido seu interesse pelo estudo, impelindo o professor a procurar novas estratégias para inovar suas aulas.

Objetivamos criar ambientes de aprendizagem mais atrativos onde os alunos interagem com materiais concretos e computadores.

A redução dos custos da eletrônica vem permitindo fazer a aplicação da robótica em diferentes áreas do conhecimento.

Para preencher a lacuna descrita acima, foram construídos dois robôs no desenvolvimento de uma pesquisa que contou com bolsistas BIC.

Neste primeira etapa contamos com a colaboração do Prof. Werner Haetinger. Na etapa 3D tivemos a participação do Prof. Marcelo Malheiros.

Os robôs foram totalmente projetados e construídos na UNIVATES, desde a parte mecânica e eletrônica até o *software*.

Descrevemos inicialmente o desenvolvimento de um sistema de controle algébrico bidimensional para o modelo cinemático, pesquisando as TL associadas ao movimento das articulações do robô, dinamizando as disciplinas de Álgebra Linear, Cálculo III, Geometria Analítica, Linguagem C e Métodos Numéricos nos cursos de Engenharia de Computação, Engenharia de Produção e Engenharia de Controle e Automação da UNIVATES.

- Álgebra Linear, Cálculo III e Geometria Analítica: modelagem matemática dos movimentos do robô articulado, via matrizes de rotação, de transformações de coordenadas e de equações diferenciais elementares.
- Linguagem C: programação dos movimentos do braço mecânico.
- Métodos Numéricos: sintonia fina da movimentação do braço, estipulando trajetórias ideais para problemas relacionados.

1.1 O Paradoxo da Álgebra Linear Básica

Por um lado, o Professor não aborda aplicações mais profundas porque os alunos não possuem muitos conhecimentos técnicos necessários e, por outro, os alunos cursam a disciplina de modo técnico sem ver as aplicações.

1.2 Motivação Matemática

O controle de mecanismos articulados é fortemente baseado em operações matemáticas.

A Matemática, a Eletrônica e a Informática são bases da robótica, que vem sendo empregada cada vez mais nas atividades de controle automatizado, principalmente em processos industriais.

Na indústria, os robôs são usados para manipular e movimentar objetos com precisão, bem como em ambientes insalubres ou contaminados, por exemplo contendo gases tóxicos.

2 Descrição do Robô

- Pré-Construção: simulação das dimensões e forma via planilha eletrônica.
- Pré-Protótipos: construção de vários pré-protótipos, utilizando materiais recicláveis, para testar a viabilidade da montagem e definir os tipos de mecanismos para transmissão de força às partes do robô.
- Aquisição dos materiais para a montagem do robô.

2.1 O Primeiro Robô

Motores de c.c. da linha automotiva (ferro-velho):

- motor para giro do pulso: de uma impressora antiga;
- estrutura: em madeira (bolsistas) com equipamentos disponíveis na instituição;
- engrenagens: em tecnil (tornearia local);
- eixos das juntas: cubos de bicicleta;
- placas controladoras: componentes eletrônicos reciclados (UNIVATES).

Os motores de corrente contínua facilitam uma determinação mais precisa de posições do que os motores de passo, enquanto que os motores de campo magnético constante (ímã permanente) permitem um torque e uma velocidade relativamente lineares, facilitando a modelagem.

O robô é alimentado com apenas uma tensão fixa de 12 V, de modo que não é possível alterar a sua velocidade no momento.

2.2 O Segundo Robô

Recebemos apoio da FAPERGS, permitindo a aquisição de motores melhores.

Solicitamos a fabricação de eixos e peças sob medida.

Adquirimos componentes eletrônicos novos.

Ambos os robôs são idênticos em sua forma, porém o segundo tem uma estrutura mais robusta.

Ver complementação com fotos no arquivo anexo,
slides 1 a 15.

Cada robô do tipo braço mecânico possui:

- uma *base* fixa;
- um *suporte vertical* giratório;
- um *braço superior*;
- um *antebraço*;
- um *pulso* giratório;
- uma *garra* magnética em sua extremidade.

São, portanto, 4 graus de liberdade.

As *juntas* de revolução do antebraço e do braço são coplanares.

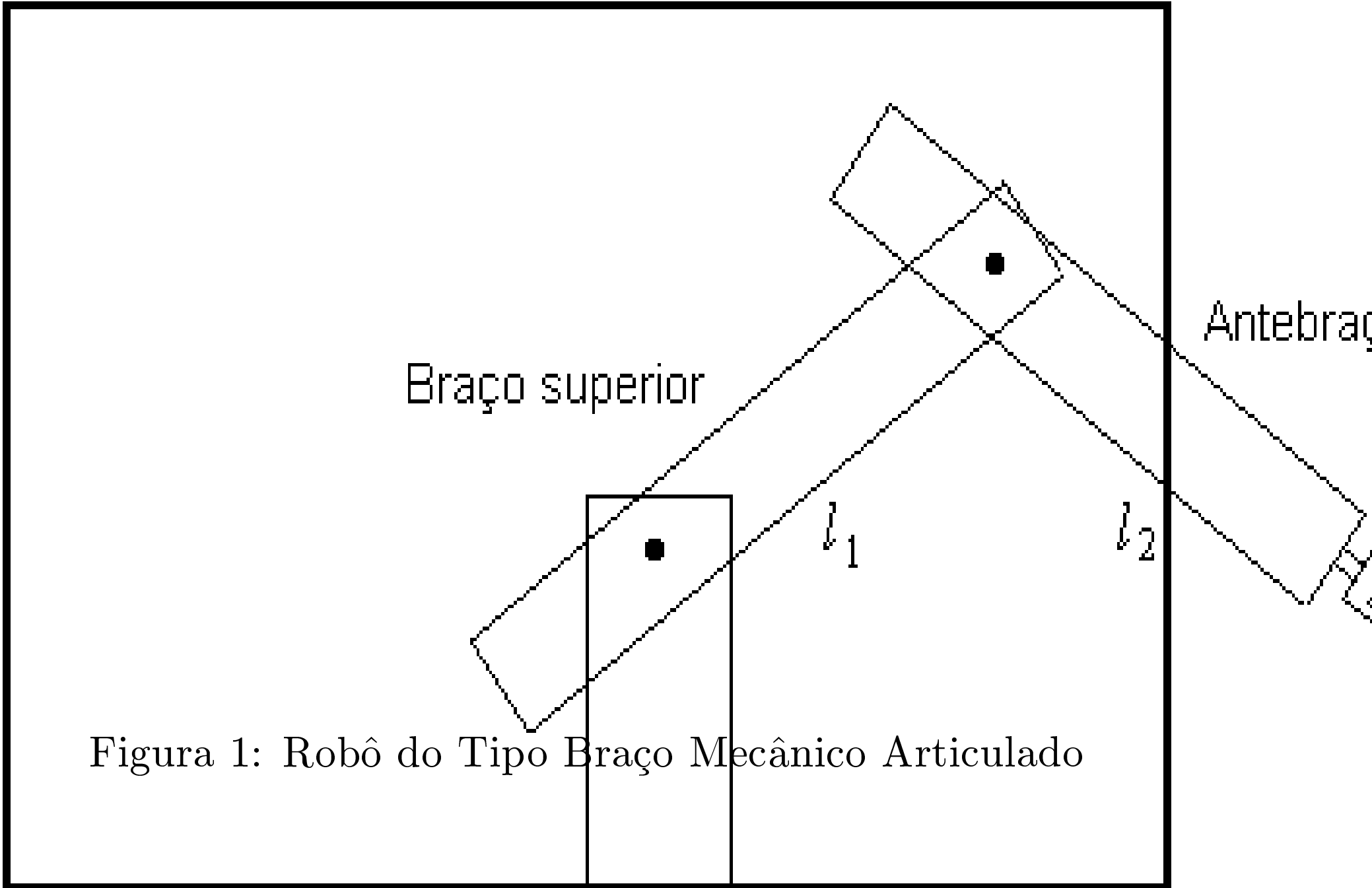


Figura 1: Robô do Tipo Braço Mecânico Articulado

Ver complementação com fotos no arquivo anexo,
slides 16 a 31.

Modelar os movimentos do robô via Álgebra Linear é importante para simular as tarefas a serem executadas pelo robô dentro da sua célula de trabalho.

Um robô deve se mover eficientemente de uma posição conhecida para outra de destino específica, visando alcançar os objetos a serem manipulados.

Para chegar a estes estágios sofisticados de automação é necessário haver um sistema computacional complexo associado ao robô.

Daí a importância das trajetórias ideais.

Há pouca literatura em português sobre robótica e sobre a modelagem de trajetórias ideais em tempo-real.

É difícil controlar um robô por computador, devido à complexidade e variedade de ambientes, plataformas, sensores e movimentos envolvidos.

Não desenvolvemos algoritmos complexos de processamento de imagens, nem de inteligência artificial.

Apenas capacitamos o braço a operar com um controle mais complexo e pré-determinado.

Usando um computador e um *software* específico, podemos variar os ângulos de controle para o robô atuar sobre uma certa região do plano XOY .

Enquanto não está plenamente desenvolvida a modelagem matemática, a programação do robô tem sido feita com algoritmos heurísticos e baseados no tempo de acionamento de cada um dos seus motores.

Desta forma a programação exige muitas tentativas e repetições dos movimentos até que se consiga alcançar o efeito desejado.

3 A Modelagem Matemática

Sejam θ_1 e θ_2 os ângulos de rotação do braço no plano (*ângulos de controle*), e sejam l_1 e l_2 os comprimentos dos *segmentos* (*elos, hastes, ou partes rígidas*): braço superior e antebraço, respectivamente.

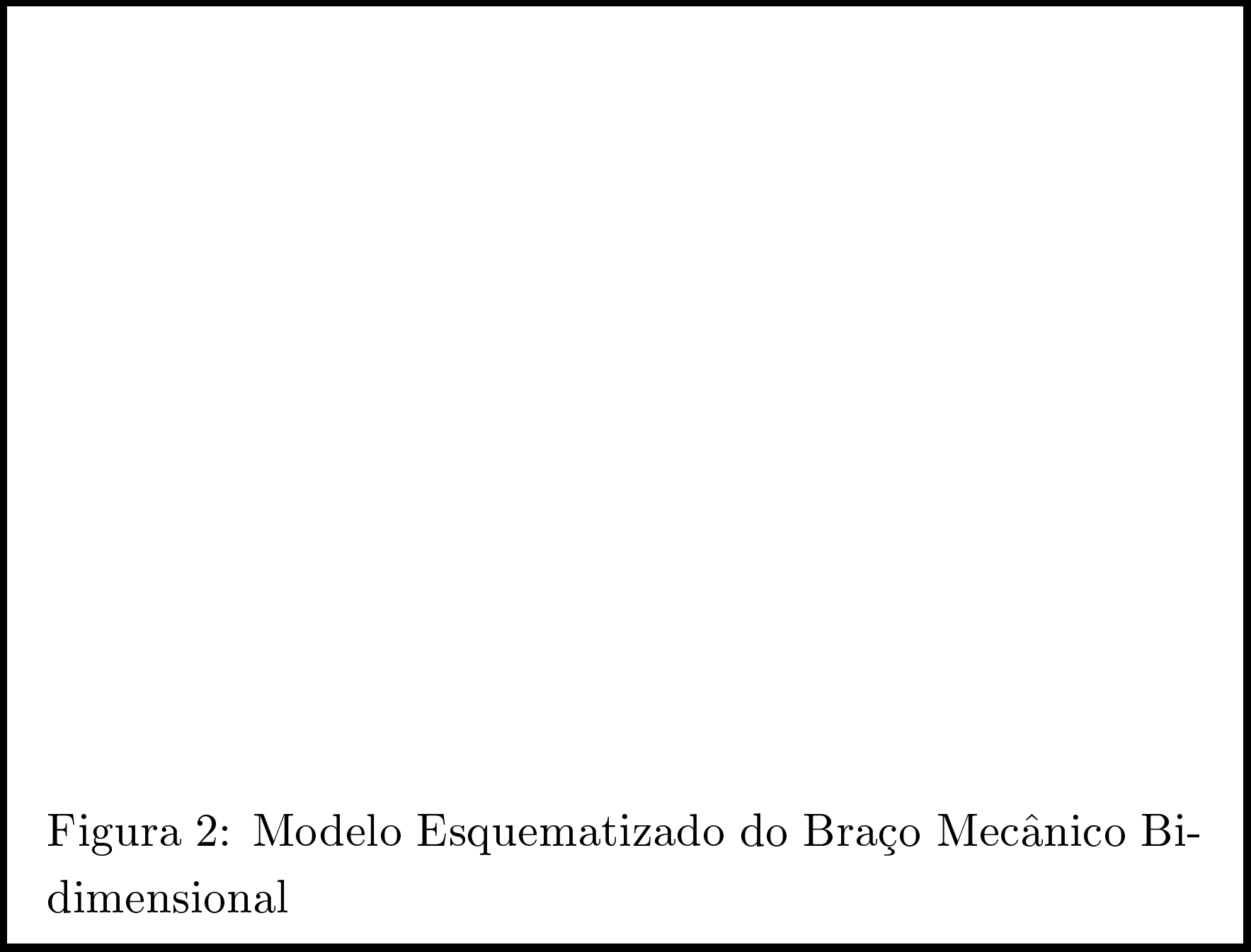


Figura 2: Modelo Esquematizado do Braço Mecânico Bi-dimensional

Sejam $P = (x, y)$ um ponto do plano XOY que queremos alcançar com o robô.

Dizemos que P é o *terminal efetuator*.

Então temos as equações paramétricas do avanço neste modelo simplificado.

$$\begin{aligned}x &= l_1 \cdot \cos(\theta_1) + l_2 \cdot \cos(\theta_1 + \theta_2) \\y &= l_1 \cdot \sin(\theta_1) + l_2 \cdot \sin(\theta_1 + \theta_2).\end{aligned}\tag{1}$$

Sejam os ângulos de controle em função do tempo t .

Suponhamos que o braço se estenda horizontalmente ao longo do eixo positivo $O\vec{X}$ a partir de $t = 0$ e que os elos 1 e 2 girem a taxas constantes de $\omega_1 \frac{rad}{s}$ e $\omega_2 \frac{rad}{s}$, respectivamente.

Então $\theta_1 = \theta_1(t) = \omega_1 t$ e $\theta_2 = \theta_2(t) = \omega_2 t$, donde as *equações cinemáticas* são descritas por

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\omega_1 t) & \cos((\omega_1 + \omega_2)t) \\ \sin(\omega_1 t) & \sin((\omega_1 + \omega_2)t) \end{bmatrix} \cdot \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} \cdot \quad (2)$$

Exemplo: utilizando *softwares* de baixo custo^a, podemos simular situações e verificar a área de atuação e o padrão de movimento do terminal efetuator num determinado intervalo de tempo, à medida que alteramos os valores de ω_i , l_i , $i = 1, 2$.

^aSugerimos alguns como *Graphmatica*, *Winplot*, *Scilab* ou *Gnuplot*.

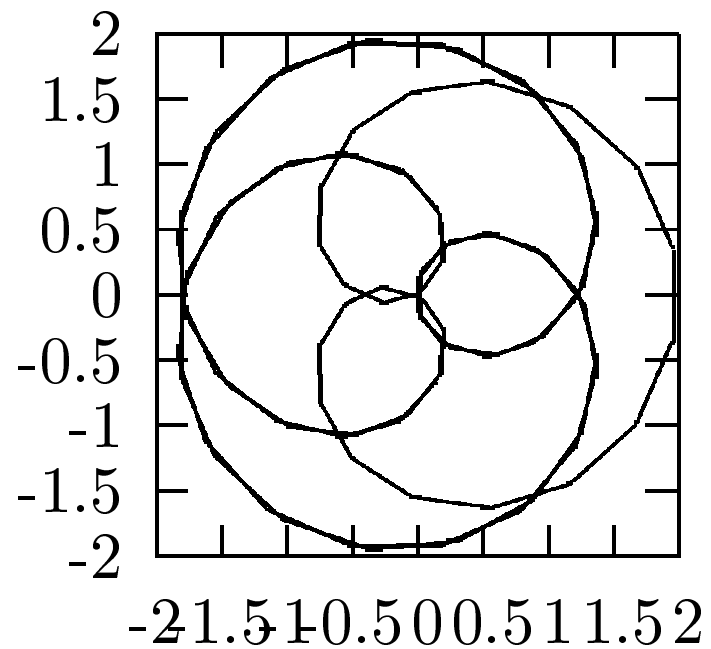


Figura 3: Padrão de Movimento do Efetuador com $l_1 = l_2 = 1$, $\omega_1 = 2 \frac{rad}{s}$ e $\omega_2 = 3 \frac{rad}{s}$

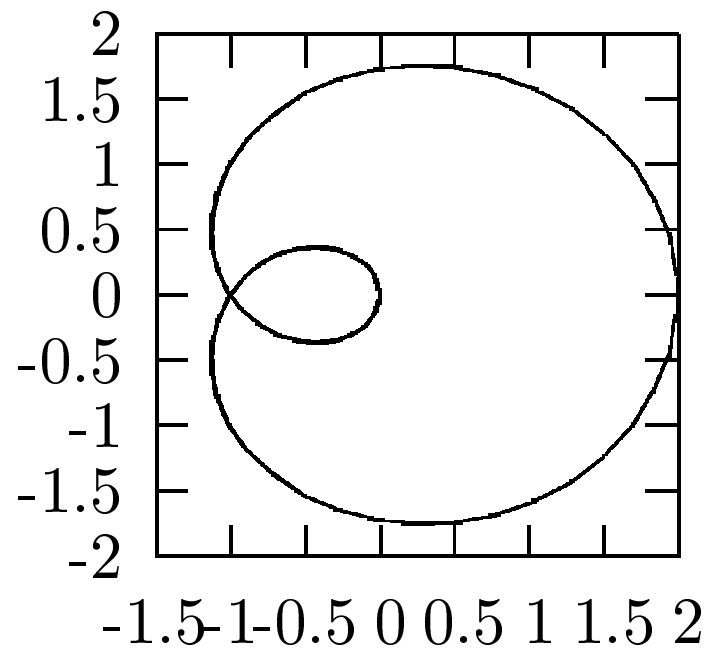


Figura 4: Padrão de Movimento do Efetuador com $l_1 = l_2 = 1$ e $\omega_1 = \omega_2 = 1 \frac{rad}{s}$

4 Os Modelos Cinemáticos Direto e Inverso

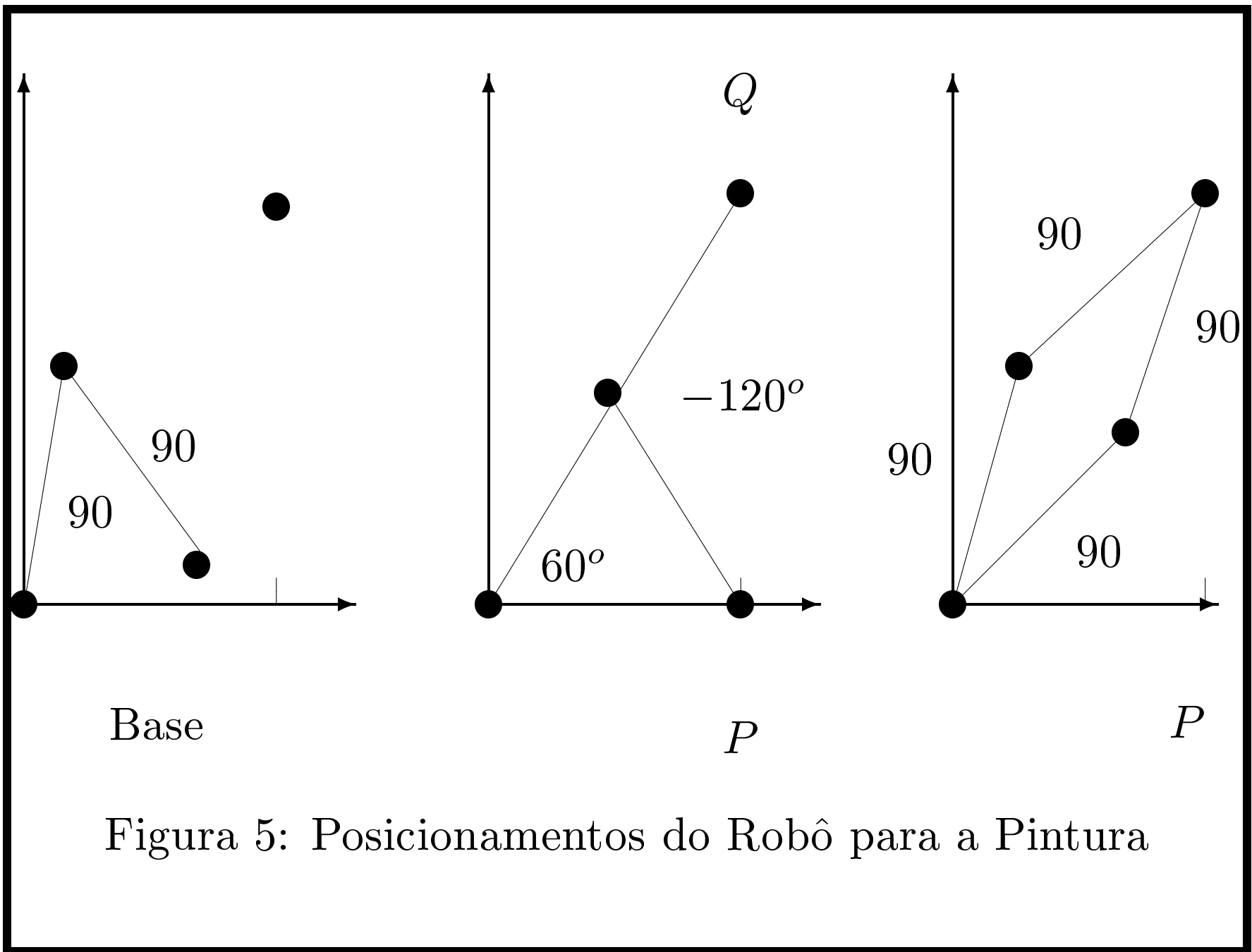
O robô deve pintar uma chapa metálica de 90cm de largura por 150cm de altura, com movimentos verticais, da base para cima.

Após a pintura de cada faixa, deve retornar à base e mover-se horizontalmente para a nova faixa.

Digamos que tenhamos optado por um robô com dois elos de igual comprimento $l_1 = l_2 = 90\text{cm}$, com base próxima ao canto esquerdo inferior da chapa (Figura 5a).

Iniciamos pintando o lado direito de um ponto $P = (90, 0)$ até um ponto $Q = (90, 150)$.

É fácil determinar a posição do efetuador no ponto P , escolhendo ângulos de controle $\theta_1 = \frac{\pi}{3}$ e $\theta_2 = -\frac{2\pi}{3}$ (Figura 5b).



Base

P

P

Figura 5: Posicionamentos do Robô para a Pintura

Em geral, o problema todo consiste em encontrar os ângulos de controle correspondentes a um ponto $Q = (u, v)$ qualquer.

Para tal, valemo-nos das equações cinemáticas:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \cos(\theta_1) & \cos(\theta_1 + \theta_2) \\ \sin(\theta_1) & \sin(\theta_1 + \theta_2) \end{bmatrix} \cdot \begin{bmatrix} l_1 \\ l_2 \end{bmatrix}. \quad (3)$$

Portanto, para pôr o terminal efetuator no ponto $Q = (u, v)$, os ângulos controladores devem satisfazer as equações

$$\begin{aligned} u &= l_1 \cdot \cos(\theta_1) + l_2 \cdot \cos(\theta_1 + \theta_2) \\ v &= l_1 \cdot \sin(\theta_1) + l_2 \cdot \sin(\theta_1 + \theta_2), \end{aligned} \tag{4}$$

onde l_1 e l_2 são os comprimentos do braço e do antebraço, respectivamente.

Após alguns cálculos e manipulações algébricas e trigonométricas, obtemos os ângulos procurados:

$$\theta_{1_1} = \arccos \left(\frac{-B + \sqrt{B^2 - 4AC}}{2A} \right) \text{ rad}$$

(5)

ou

$$\theta_{1_2} = \arccos \left(\frac{-B - \sqrt{B^2 - 4AC}}{2A} \right) \text{ rad},$$

onde

$$A = 4u^2 l_1^2 + 4v^2 l_1^2,$$

$$B = -4ul_1 \cdot (u^2 + v^2 + l_1^2 - l_2^2), \quad e$$

$$C = (u^2 + v^2 + l_1^2 - l_2^2)^2 - 4v^2 l_1^2.$$

Para calcular estes ângulos em graus, basta multiplicá-los pelo fator $\frac{180}{\pi}$.

Basta agora substituir cada um destes ângulos na primeira equação de (4) e resolvê-la, obtendo

$$\begin{aligned}\theta_{2_1} &= -\theta_{1_1} + \arccos\left(\frac{u-l_1 \cdot \cos(\theta_{1_1})}{l_2}\right) \text{ rad}, \\ \text{ou} & \\ \theta_{2_2} &= -\theta_{1_2} + \arccos\left(\frac{u-l_1 \cdot \cos(\theta_{1_2})}{l_2}\right) \text{ rad}.\end{aligned}\tag{6}$$

É fácil ver que uma vez definida a posição do terminal efetuator, sempre haverá duas posições ideais para os elos (Figura 5c).

Agora vejamos como representar graficamente o braço mecânico.

Uma vez definido o ponto extremo do segundo elo do braço, $Q = (u, v)$, e encontrados os ângulos θ_{1_1} e θ_{2_1} , podemos determinar o ponto extremo do primeiro elo, $P = (a, b)$, via equações paramétricas:

$$a = l_1 \cdot \cos(\theta_{1_1})$$

$$b = l_1 \cdot \sin(\theta_{1_1}).$$

A equação da reta que passa pelos pontos $O = (0, 0)$

e $P = (a, b)$, é obtida de $\begin{vmatrix} x & y & 1 \\ a & b & 1 \\ 0 & 0 & 1 \end{vmatrix} = 0$. Ou seja,

primeiro elo do braço é dado por

$$r_1 : y = \frac{b}{a} \cdot x, \quad x \in [0, a].$$

Já a equação da reta que passa pelos pontos

$$P = (a, b) \text{ e } Q = (u, v), \text{ é obtida de } \begin{vmatrix} x & y & 1 \\ a & b & 1 \\ u & v & 1 \end{vmatrix} = 0.$$

Ou seja, segundo elo do braço é dado por

$$r_2 : y = \frac{(v - b)}{u - a} \cdot x + \frac{(ub - av)}{u - a}, \quad x \in [a, u].$$

Passemos ao estudo de como programar o computador para pintar. Por exemplo, para efetuar a pintura no sentido vertical, desde o ponto $S = (u, 0)$ até o ponto $Q = (u, v)$, com velocidade constante, basta

considerar o gráfico de
$$\begin{cases} r_1, & \text{se } 0 \leq x \leq a \\ r_2, & \text{se } a \leq x \leq u \end{cases} .$$

Observe que existe uma limitação para as coordenadas u e v , dados os comprimentos l_1 e l_2 , e dado o fato de que o caso estudado é bidimensional.

Finalmente, passamos ao estudo de como programar o computador para pintar o segmento de reta vertical entre dois pontos.

Sabemos que, além de mover o terminal efetuator sobre a reta vertical, precisamos controlar sua velocidade.

No nosso caso, o robô se movimenta com velocidade constante v_{vert} .

Precisamos agora determinar as taxas de rotação $\frac{d\theta_1}{dt}$ e $\frac{d\theta_2}{dt}$ (em $\frac{rad}{s}$) de modo que $\frac{du}{dt} = 0$ e $\frac{dv}{dt} = v_{vert}$.

A primeira condição garante que o terminal efetuator se mova verticalmente (pois não há velocidade horizontal), enquanto a segunda garante um movimento para cima à velocidade de $v_{vert} \frac{cm}{s}$.

Diferenciando as equações cinemáticas de avanço (3), obtemos:

$$\frac{du}{dt} = -l_1 \cdot \sin(\theta_1) \cdot \frac{d\theta_1}{dt} - l_2 \cdot \sin(\theta_1 + \theta_2) \cdot \left(\frac{d\theta_1}{dt} + \frac{d\theta_2}{dt} \right)$$

$$\frac{dv}{dt} = l_1 \cdot \cos(\theta_1) \cdot \frac{d\theta_1}{dt} + l_2 \cdot \cos(\theta_1 + \theta_2) \cdot \left(\frac{d\theta_1}{dt} + \frac{d\theta_2}{dt} \right) \quad (7)$$

Utilizando novamente as equações cinemáticas, conseguimos simplificar (7) e, então, utilizar as condições iniciais $\frac{du}{dt} = 0$ e $\frac{dv}{dt} = v_{vert}$ para resolvê-las, obtendo

$$-v \cdot \frac{d\theta_1}{dt} - l_2 \cdot \sin(\theta_1 + \theta_2) \cdot \frac{d\theta_2}{dt} = 0 \quad (8)$$

$$u \cdot \frac{d\theta_1}{dt} + l_2 \cdot \sin(\theta_1 + \theta_2) \cdot \frac{d\theta_2}{dt} = v_{vert}.$$

As equações (8) são utilizadas nos algoritmos seguintes:

1. **Modelo Cinemático Direto:** descrição da posição e da orientação da garra em função das variáveis da junta. Associa-se a cada elo do robô um sistema de coordenadas.
 - (a) em um dado instante t , o robô relata os ângulos de controle θ_1 e θ_2 de seus elos ao computador;
 - (b) o computador usa as equações cinemáticas de avanço em (3) para calcular as coordenadas u e v do terminal efetuator;
 - (c) de posse destes dados, os valores de θ_1 , θ_2 , u e v são substituídos em (8) para produzir duas equações nas incógnitas $\frac{d\theta_1}{dt}$ e $\frac{d\theta_2}{dt}$;
 - (d) o computador resolve as equações e determina as taxas de rotação exigidas para os elos.

2. **Modelo Cinemático Inverso:** o modelo consiste na determinação das variáveis da junta a partir da posição e orientação da garra. Existem duas soluções para cada posição dada.
- (a) em um dado instante t , o robô relata as coordenadas u e v do terminal efetuator;
 - (b) o computador usa as equações (5) e (6) para calcular os ângulos de controle θ_1 e θ_2 de seus elos;
 - (c) de posse destes dados, os valores de θ_1 , θ_2 , u e v são substituídos em (8) para produzir duas equações nas incógnitas $\frac{d\theta_1}{dt}$ e $\frac{d\theta_2}{dt}$;
 - (d) o computador resolve as equações e determina as taxas de rotação exigidas para os elos.

Exemplo: com os dados do exemplo anterior ($\theta_1 = \frac{\pi}{3}$ e $\theta_2 = -2\frac{\pi}{3}$), obtemos as taxas de rotação (em $\frac{rad}{s}$) para os elos do terminal efetuator se moverem para cima à velocidade de $1\frac{cm}{s}$. Então as equações (8) se escrevem como

$$-0 \cdot \frac{d\theta_1}{dt} + 90 \cdot \frac{\sqrt{3}}{2} \cdot \frac{d\theta_2}{dt} = 0$$

$$90 \cdot \frac{d\theta_1}{dt} - 90 \cdot \frac{\sqrt{3}}{2} \cdot \frac{d\theta_2}{dt} = 1.$$

Donde $\frac{d\theta_1}{dt} = \frac{1}{90} \approx 0,0111 \frac{rad}{s}$ e
 $\frac{d\theta_2}{dt} = \frac{1}{45 \cdot \sqrt{3}} = \frac{\sqrt{3}}{135} \approx 0,0128 \frac{rad}{s}$.

Com *softwares* licenciados (Excel, Mathcad, Matlab, Turbo C), ou livres (Scilab), pode-se facilmente implementar este algoritmo.

Mostrar programas em *DevC++* em que são feitas as simulações descritas acima: arquivos `bracla.exe`, `bragra2.exe` e `brapin4.exe`.

5 Conclusões e Atividades Futuras do Caso 2D

O uso do robô torna mais palpável ao aluno a compreensão dos conceitos da Álgebra Linear e da Geometria Analítica, mostra com mais riqueza as aplicações das Equações Diferenciais, permite uma interação entre teoria e prática nas disciplinas de Linguagem de Programação e Métodos Numéricos, além de consistir numa concretização de um problema real que fará parte da vida profissional do estudante.

No momento estamos pesquisando os modelos descritos acima no caso em que o robô é do tipo serpente, i.e., apresenta várias juntas interligadas.

Neste caso, no modelo cinemático inverso, existem várias maneiras de colocar a garra na mesma posição com a mesma orientação em função da posição das juntas.

Então surge um novo problema: determinar qual é a melhor solução.

Outro problema é que existem singularidades (às vezes não há soluções).

Já no modelo cinemático direto sempre existem soluções (sempre tenho uma posição da garra resultante da posição das juntas).

Contudo, há limitações de posição das juntas e do espaço de trabalho do robô que às vezes impossibilitam soluções na cinemática inversa.

Além disso, o problema é não-linear.

O modelo direto também não era, mas dependia apenas de senos e cossenos, mas pode ser linearizado por substituição de variáveis.

Na cinemática inversa não mais é possível expressar o problema na forma de matrizes.

A vantagem da cinemática inversa admitir várias soluções é que podemos desviar de obstáculos.

Como trabalhos futuros, pretendemos desenvolver um estudo do caso tridimensional e suas implicações do ponto de vista teórico matricial, bem como as implementações computacionais necessárias.

Ademais, procuraremos desenvolver também o modelo dinâmico.

Na seção seguinte, mostraremos o desenvolvimento da pesquisa no caso 3D.

Apresentar slides do arquivo complementar, do 32
ao 34.

6 O Anel \mathbb{H} dos Quatérnios

W.R. Hamilton (1805-1865) fundamentou os complexos como pares ordenados de reais.

Desenvolveu uma álgebra que permitia trabalhar com vetores no plano, para poder resolver a questão de criar uma álgebra para 3-uplas que permitissem trabalhar com vetores 3D, em dinâmica.

A grosso modo, quatérnios são uma extensão de \mathbb{C} . Ao invés de apenas i , temos três diferentes números tais que $i = j = k = \sqrt{-1}$.

O sistema numérico de quatérnios é importante em pelo menos dois contextos:

No desenvolvimento da álgebra – mostrou que as leis fundamentais sugeridas pelos sistemas conhecidos até 1843 não eram algo a ser aceito sempre, *a priori*.

O conjunto \mathbb{H} foi o primeiro exemplo conhecido onde a ordem dos fatores altera o produto, i.e., a primeira álgebra não-comutativa, permitindo estender o conjunto das álgebras até então conhecidas.

Sejam $\mathbb{H} = \{a + b\iota + cj + dk \mid a, b, c \in \mathbb{R}\}$ o conjunto de todas as CL reais de ι, j, k ; e $q_1 = (a_1, b_1, c_1, d_1), q_2 = (a_2, b_2, c_2, d_2) \in \mathbb{H}$.

Definimos

$$q_1 + q_2 = (a_1 + a_2) + (b_1 + b_2)\iota + (c_1 + c_2)j + (d_1 + d_2)k.$$

A multiplicação é definida por distributividade, considerando as regras: $\iota^2 = j^2 = k^2 = -1, \iota j = -j\iota = k, jk = -kj = \iota, k\iota = -\iota k = j$, donde $q_1 \star q_2 = (a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2) + (a_1b_2 + a_2b_1 + c_1d_2 - d_1c_2)\iota + (a_1c_2 + a_2c_1 + d_1b_2 - b_1d_2)j + (a_1d_2 + a_2d_1 + b_1c_2 - c_1b_2)k$.

Tecnicamente, \mathbb{H} é uma anel associativo não-comutativo (mais ainda: um anel de divisão).

7 Uma Representação Não Usual de \mathbb{H}

Podemos representar \mathbb{H} de várias maneiras como CL de $1, i, j$, e k :

1. Como um vetor dos quatro coeficientes nesta CL;
2. Como um escalar para o coeficiente de 1 e um vetor para os coeficientes dos termos imaginários.

Em 2., $q = a + bi + cj + dk \in \mathbb{H}$ é escrito como (a, \vec{v}) , onde \vec{v} é o vetor 3D $\vec{v} = (b, c, d)$.

Assim, para $q_1 = (a_1, \vec{v}_1)$ e $q_2 = (a_2, \vec{v}_2)$, temos:
 $q_1 + q_2 = (a_1 + a_2, \vec{v}_1 + \vec{v}_2)$, e
 $q_1 \star q_2 = (a_1 a_2 - \vec{v}_1 \cdot \vec{v}_2, a_1 \vec{v}_2 + a_2 \vec{v}_1 + \vec{v}_1 \times \vec{v}_2)$ (produto escalar e produto vetorial, respectivamente).

Por conseguinte, se $\vec{0} = (0, 0, 0)$, então $(a, \vec{0}) \in \mathbb{R}$; e se $\vec{v} = (b, 0, 0)$, então $(a, \vec{v}) \in \mathbb{C}$.

A notação acima é não usual para matemáticos, mas é a preferida por engenheiros que trabalham com Computação Gráfica e Robótica, por motivos que veremos a seguir.

Mais ainda, para $q = (a, \vec{v}) \in \mathbb{H}$, temos que:

$-q = (-a, -\vec{v})$ é o simétrico de q ;

$\bar{q} = (a, -\vec{v})$ é o conjugado de q ;

$q^{-1} = \left(\frac{a}{a^2 + |\vec{v}|^2}, -\frac{\vec{v}}{a^2 + |\vec{v}|^2} \right)$ é o inverso de q ($q \neq 0$).

Quatérnios da forma $(a, \vec{0})$ são usualmente denotados por $a \in \mathbb{R}$. Se c é um escalar, então $c \star q = (ca, c\vec{v})$. Assim, $q^{-1} = \frac{(a, -\vec{v})}{a^2 + |\vec{v}|^2}$, onde $|\vec{v}|$ é a norma de \vec{v} , i.e., $|\vec{v}| = \sqrt{b^2 + c^2 + d^2}$. Finalmente, definimos a norma de q como $|q| = \sqrt{a^2 + |\vec{v}|^2}$.

8 Representando Rotações por Quatérnios

Podemos utilizar a multiplicação de quatérnios para efetuar uma rotação por um eixo unitário arbitrário \vec{u} de um ângulo θ .

O quatérnio que representa esta rotação é $q = (a, \vec{v})$, onde $a = \cos\left(\frac{\theta}{2}\right)$ e $\vec{v} = \vec{u} \sin\left(\frac{\theta}{2}\right)$.

Note que $|q| = 1$, donde $q^{-1} = \bar{q}$.

Representaremos um ponto $P \in \mathbb{R}^3$ por $p = (0, \vec{P}) \in \mathbb{H}$. A rotação desejada é dada por

$$P_{rot} = q \star p \star q^{-1}.$$

9 Quatérnios Realmente Produzem 3D-Rotações?

Segundo K. Shoemake, (\mathbb{H}, \star) preserva normas, i.e., $N(p \star q) = N(p)N(q)$, donde $N(q^{-1}) = N(q)^{-1}$. Então $N(q \star p \star q^{-1}) = N(p)$. Portanto a aplicação $p \rightsquigarrow q \star p \star q^{-1}$ é uma transformação ortogonal em 4D.

Por outro lado, a multiplicação por escalar comuta, donde $q \star a \star q^{-1} = a$. Então a aplicação $p \rightsquigarrow q \star p \star q^{-1}$ é uma transformação ortogonal do vetor 3D de p .

Trata-se de um movimento contínuo da identidade a cada possível ação. Isto exclui reflexões, portanto a aplicação $p \rightsquigarrow q \star p \star q^{-1}$ é uma 3D-rotação.

10 Aplicações

Façamos duas rotações da garra do robô. Sejam $q_1, q_2 \in \mathbb{H}$ seus representantes unitários. A rotação composta é dada por

$$q_2 \star q_{1_{rot}} = (q_2 \star q_1) \star p \star (q_2 \star q_1)^{-1}.$$

A representação de rotações por quatérnios (4 n^{os}) é mais compacta que a por matrizes ortogonais (9 n^{os}).

Dados um eixo e um ângulo, é fácil construir o quatérnio correspondente, e reciprocamente. Ambos são bastante difíceis com matrizes ou ângulos de Euler.

Em *games* de computador e outras aplicações, interessam as rotações “suaves”, i.e., a imagem deve rotacionar lentamente e não de um passo só.

Isto pode ser obtido escolhendo uma curva em quatérnios, com um ponto final sendo a transformação identidade 1 e o outro sendo a rotação pretendida, o que é bem mais problemático com outros tipos de representações de rotações.

Ao compor muitas rotações num computador, erros de arredondamento acumularão.

Um quatérnio pode sofrer algum pequeno erro e ainda representar uma rotação.

Uma matriz que sofrer um pequeno erro não mais será ortogonal e não mais representará uma rotação por completo. É muito difícil voltar a fazer uma matriz destas ser ortogonal.

A matriz ortogonal que corresponde ao quatérnio unitário $q = a + bi + cj + dk$ é dada por:

$$\begin{pmatrix} a^2 + b^2 - c^2 - d^2 & 2bc - 2ad & 2ac + 2bd \\ 2ad + 2bc & a^2 - b^2 + c^2 - d^2 & 2cd - 2ab \\ 2bd - 2ac & 2ab + 2cd & a^2 - b^2 - c^2 + d^2 \end{pmatrix}.$$

Os quatérnios são muito utilizados em Computação Gráfica e associados à Análise Geométrica para representar rotações e orientações de objetos em \mathbb{R}^3 .

Também são usados em Teoria de Controle, Processamento de Sinais, Física e Mecânica Orbital.

Podemos ainda utilizar quatérnios para efetuar interpolações lineares entre matrizes de rotação, usando cinemática inversa.

No artigo associado a esta comunicação descrevemos algumas destas aplicações.

11 Representando Quatérnios por Matrizes

Há pelo menos duas maneiras de fazê-lo, usando homomorfismos de quatérnios-matrizes (preservando somas e multiplicações).

1. Usando $\mathcal{M}_2(\mathbb{C})$: o quatérnio $q = a + bi + cj + dk$ é dado por

$$\begin{pmatrix} a - di & -b + ci \\ b + ci & a + di \end{pmatrix}.$$

Esta representação apresenta boas propriedades:

- Todo número complexo corresponde a matrizes com entradas reais;
- O quadrado da norma do quatérnio é o determinante da matriz associada;
- O conjugado do quatérnio é a transposta conjugada da matriz associada;
- Para quatérnios unitários, fornece o grupo de isomorfismos entre a 3D-esfera \mathcal{S}_3 e $SU(2)$, importante em mecânica quântica no trato com *spins*.

2. Usando $\mathcal{M}_4(\mathbb{R})$: o quatérnio $q = a + bi + cj + dk$ é dado por

$$\begin{pmatrix} a & -b & d & -c \\ b & a & -c & -d \\ -d & c & a & -b \\ c & d & b & a \end{pmatrix}.$$

Neste caso, o conjugado do quatérnio é a transposta da matriz associada.

12 Conclusões e Ações Futuras no Caso 3D

Com estas representações podemos descrever os modelos cinemáticos dos 3D-movimentos do nosso robô, generalizando a nossa 2D-descrição anterior.

Como trabalhos futuros estudaremos os sistemas de equações não-lineares a várias variáveis, usando Bases de Gröbner para determinar a posição da garra do robô em cada ponto de sua trajetória, bem como os modelos dinâmicos da sua movimentação.

Mostrar slides 35 ao 40 do arquivo complementar
anexo.

Mostrar os vídeos anexos, arquivos mov00028.mpg e mov00038.mpg.

Concluir com os slides 41 ao 44 do arquivo complementar anexo.

Claus Haetinger

Centro de Ciências Exatas e Tecnológicas

Centro Universitário UNIVATES

95900-000, Lajeado-RS

e-mail: chaet@univates.br

URL <http://ensino.univates.br/~chaet>