

TÉCNICAS DE CONTROLE DE ROBÔS MÓVEIS

- **Controladores Cinemáticos:**

- Baseados apenas no modelo cinemático. A dinâmica a ser controlada é simplesmente o sistema não linear:

$$\mathbf{q}' = {}^q\mathbf{T}_V \cdot \mathbf{V} \quad \text{ou,} \quad \mathbf{q}' = {}^q\mathbf{T}_V \cdot {}^V\mathbf{T}_W \cdot \mathbf{W}$$

- O modelo dinâmico (incluindo as equações do movimento de Newton e Euler, bem como o modelo dinâmico dos atuadores) é desprezado.
- Pressupõem que as velocidades das rodas ω_D ω_E (ou, equivalentemente, v e ω) podem ser impostas instantaneamente.
- Válido para aplicações em que os efeitos dinâmicos podem ser desprezados. Exemplo: baixas velocidades e acelerações, baixa inércia.
- Duas categorias: Controle de Trajetória e Estabilização.

- **Controle de Trajetória:**

- Problema mais simples do que o problema de estabilização.
- A referência a ser seguida é apenas a posição $(x^*(t), y^*(t))$.
- A orientação desejada é definida implicitamente pela trajetória seguida e pelas restrições não holonômicas.

$$\tan(\theta) = y'/x' \Rightarrow d(\tan(\theta))/dt = \theta'/\cos^2(\theta) = [x'.y'' - x''.y']/ (x')^2$$

$$\Rightarrow \theta' = [x'.y'' - x''.y'] / (x'/\cos(\theta))^2 = [x'.y'' - x''.y'] / [(x')^2 + (y')^2]$$

- As referências em espaço de configuração podem ser mapeadas para comandos de velocidade linear e angular:

$$v^* = \pm [(x^*)^2 + (y^*)^2]^{1/2} \quad \omega^* = [x^*.y^{*''} - x^{*''}.y^*] / [(x^*)^2 + (y^*)^2]$$

Controle por Linearização por Realimentação Dinâmica

- O Controle por Linearização por Realimentação Dinâmica (DFL - *Dynamic Feedback Linearization*) envolve Realimentação PD e Compensação do Modelo Não Linear.
- Um estado adicional é acrescentado ao sistema, derivando o modelo cinemático $\mathbf{q}' = {}^q\mathbf{T}_v \cdot \mathbf{V}$:

$$x' = v \cdot \cos\theta \quad \Rightarrow \quad x'' = \cos\theta \cdot v' - \text{sen}\theta \cdot v \cdot \omega$$

$$y' = v \cdot \text{sen}\theta \quad \Rightarrow \quad y'' = \text{sen}\theta \cdot v' + \cos\theta \cdot v \cdot \omega$$

$$\Rightarrow \begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} \cos\theta & -v \cdot \text{sen}\theta \\ \text{sen}\theta & v \cdot \cos\theta \end{bmatrix} \cdot \begin{bmatrix} v' \\ \omega \end{bmatrix}$$

- Trajetória desejada: $(x^*(t), y^*(t))$ e suas 1ª e 2ª derivadas.
- Requer medição da configuração $[x \ y \ \theta]^T$ e das velocidades x', y' , bem como a determinação de v .
- Lei de Controle:

- Realimentação Proporcional Derivativa:

$$x_c'' = x^{*''} + K_{dx}(x^{*'} - x') + K_{px}(x^* - x)$$

$$y_c'' = y^{*''} + K_{dy}(y^{*'} - y') + K_{py}(y^* - y)$$

onde x_c'' e y_c'' são as acelerações de comando em espaço de configuração, K_{px} e K_{py} são ganhos proporcionais e K_{dx} e K_{dy} são ganhos derivativos positivos.

- Compensação do Modelo Não Linear:

$$\Rightarrow \begin{bmatrix} v_c' \\ \omega_c \end{bmatrix} = \begin{bmatrix} \cos\theta & \text{sen}\theta \\ -\text{sen}\theta/v & \cos\theta/v \end{bmatrix} \cdot \begin{bmatrix} x_c'' \\ y_c'' \end{bmatrix}$$

- Integração - velocidade de comando: $v_c = \int v_c' dt$

- Em malha fechada:

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} \cos\theta & -v.\text{sen}\theta \\ \text{sen}\theta & v.\text{cos}\theta \end{bmatrix} \cdot \begin{bmatrix} v_c' \\ \omega_c \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} \cos\theta & -v.\text{sen}\theta \\ \text{sen}\theta & v.\text{cos}\theta \end{bmatrix} \begin{bmatrix} \cos\theta & \text{sen}\theta \\ -\text{sen}\theta/v & \text{cos}\theta/v \end{bmatrix} \cdot \begin{bmatrix} x_c'' \\ y_c'' \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_c'' \\ y_c'' \end{bmatrix} = \begin{bmatrix} x_c'' \\ y_c'' \end{bmatrix}$$

$$\Rightarrow \begin{aligned} x_c'' - x'' &= 0 \\ y_c'' - y'' &= 0 \end{aligned}$$

Assim, a dinâmica do erro $[\Delta x \ \Delta y]^T = [(x^* - x) \ (y^* - y)]^T$ é:

$$s^2.\Delta x(s) + K_{dx}.s.\Delta x(s) + K_{px}.\Delta x(s) = 0$$

$$s^2.\Delta y(s) + K_{dy}.s.\Delta y(s) + K_{py}.\Delta y(s) = 0$$

- Observações:
 - A trajetória desejada deve ser duplamente derivável.
 - O estado inicial do integrador deve ser $v_c \neq 0$.
 - A trajetória desejada deve ser persistente, de modo a evitar singularidades.
 - Uma possível modificação para contornar a divisão por zero quando v tende a zero é substituir v por $(v + \delta)$, onde δ é um coeficiente positivo e muito menor do que a velocidade nominal.

- **Controle Estabilizante:**

- Problema mais complexo do que o problema de controle de trajetória.
- A referência a ser alcançada é uma configuração (x^*, y^*, θ^*) .
- O sistema é subatuado (apenas duas entradas, v e ω ou ω_D e ω_E , para controlar três saídas: x' , y' e θ').

Controle por Coordenadas Polares (*Polar Coordinate Control*)

- Sistema descrito pelo modelo não linear: $\mathbf{q}' = {}^q\mathbf{T}_v \cdot \mathbf{V}$
- Baseado na transformação para coordenadas polares da configuração do robô, a qual é singular na origem:

$$R = [x^2 + y^2]^{1/2}$$

$$\gamma = \tan^{-1}(y/x)$$

$$\delta = \gamma + \theta$$

- Configuração final desejada: $\mathbf{q}^* = [x^*, y^*, \theta^*]^T$. Sem perda de generalidade, assumiremos $\mathbf{q}^* = [0 \ 0 \ 0]^T$.
- Medição apenas da configuração $[x \ y \ \theta]^T$.
- Regulador estável no sentido de Lyapunov, definido pela lei de controle:

$$v = K_R \cdot R \cdot \cos\gamma$$

$$\omega = K_\gamma \cdot \gamma + K_R \cdot (\gamma + K_\delta \cdot \delta) \cdot (\sin\gamma \cdot \cos\gamma) / \gamma$$

onde K_R , K_γ e K_δ são ganhos constantes e positivos.

- Observação: a lei de controle deve ser desligada ou modificada na vizinhança da origem, de modo a evitar a singularidade introduzida por $\tan^{-1}(y/x)$.

- **Controladores Dinâmicos:**

- Baseados no modelo completo, incluindo as relações cinemáticas, as equações de movimento de Newton e Euler e a dinâmica de atuadores.
- As entradas de controle são fornecidas diretamente pelos atuadores (tensões de armadura dos motores direito e esquerdo e_D e e_E , ou entradas equivalentes).
- As velocidades das rodas ω_D ω_E (ou, equivalentemente, as velocidades do robô v e ω) não são impostas instantaneamente, mas apresentam resposta de acordo com a dinâmica do sistema.
- Válido para aplicações em que os efeitos dinâmicos não podem ser desprezados. Exemplo: altas velocidades e acelerações, alta inércia.
- Duas categorias: Controle de Trajetória e Estabilização.

- **Controle de Trajetória:**

- Problema mais simples do que o problema de estabilização.
- A referência a ser seguida é apenas a posição $(x^*(t), y^*(t))$.
- A orientação desejada é definida implicitamente pela trajetória seguida e pelas restrições não holonômicas.

$$\tan(\theta) = y'/x' \Rightarrow d(\tan(\theta))/dt = \theta'/\cos^2(\theta) = [x'.y'' - x''.y']/x'^2$$

$$\Rightarrow \theta' = [x'.y'' - x''.y']/(x'/\cos(\theta))^2 = [x'.y'' - x''.y']/[(x')^2 + (y')^2]$$

- As referências em espaço de configuração podem ser mapeadas para comandos de velocidade linear e angular:

$$v^* = \pm[(x^*)^2 + (y^*)^2]^{1/2} \quad \omega^* = [x^*.y^{*''} - x^{*''}.y^*]/[(x^*)^2 + (y^*)^2]$$

Controle de Velocidade – Compensação Não Linear + PI

- Sistema descrito pelo modelo não linear:

$$\boldsymbol{\tau}_V = \mathbf{M}_V \cdot \mathbf{V}' + \mathbf{B}_V \cdot \mathbf{V}$$

- Trajetória desejada: $(x^*(t), y^*(t))$ e suas 1ª e 2ª derivadas. $\theta^*(t)$ e suas derivadas são calculados de modo a respeitar as restrições não holonômicas.
- Mapeamento das velocidades e acelerações desejadas em espaço de configuração para velocidades e acelerações no referencial do robô, $\mathbf{V}^* = ({}^q\mathbf{T}_V)^T \cdot \mathbf{q}^{*'}$ e $\mathbf{V}^{*'} = ({}^q\mathbf{T}_V)^T \cdot \mathbf{q}^{*''}$.
- Medição da configuração $\mathbf{q} = [x \quad y \quad \theta]^T$ e velocidade em espaço de configuração: $\mathbf{q}'' = [x'' \quad y'' \quad \theta'']^T$.
- Mapeamento das velocidades medidas em espaço de configuração para o referencial do robô: $\mathbf{V} = ({}^q\mathbf{T}_V)^T \cdot \mathbf{q}'$
- Lei de Controle incluindo Realimentação PI e Compensação do Modelo Não Linear:

- Realimentação Proporcional Integral:

$$\mathbf{V}_c' = \mathbf{V}^{*'} + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt$$

onde $\Delta \mathbf{V} = (\mathbf{V}^* - \mathbf{V})$, \mathbf{V}_c' é a aceleração de comando, \mathbf{K}_p é uma matriz diagonal de ganhos proporcionais positivos e \mathbf{K}_i é uma matriz diagonal de ganhos integrais positivos.

- Compensação do Modelo Não Linear:

$$\boldsymbol{\tau}_V = \mathbf{M}_V^* \cdot \mathbf{V}_c' + \mathbf{B}_V^* \cdot \mathbf{V}$$

onde \mathbf{M}_V^* e \mathbf{B}_V^* são as matrizes de parâmetros dinâmicos disponíveis, que constituem o modelo nominal. São uma estimativa dos parâmetros reais \mathbf{M}_V e \mathbf{B}_V .

- Em malha fechada:

$$\mathbf{M}_V^* \cdot (\mathbf{V}^{*'} + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) + \mathbf{B}_V^* \cdot \mathbf{V} = \mathbf{M}_V \cdot \mathbf{V}' + \mathbf{B}_V \cdot \mathbf{V}$$

$$\Rightarrow \mathbf{M}_V^* \cdot (\mathbf{V}^{*'} + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) = \mathbf{M}_V \cdot \mathbf{V}' + \mathbf{B}_V \cdot \mathbf{V} - \mathbf{B}_V^* \cdot \mathbf{V}$$

$$\Rightarrow \mathbf{M}_V^* \cdot (\mathbf{V}^{*'} + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) = \mathbf{M}_V \cdot \mathbf{V}' + (\mathbf{B}_V - \mathbf{B}_V^*) \cdot \mathbf{V}$$

$$\Rightarrow \mathbf{M}_V^* \cdot (\mathbf{V}^{*'} + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) = \mathbf{M}_V \cdot \mathbf{V}' + \Delta_{\mathbf{B}_V} \cdot \mathbf{V}$$

onde $\Delta_{\mathbf{B}_V} = (\mathbf{B}_V - \mathbf{B}_V^*)$ é o erro de modelagem da matriz \mathbf{B}_V^* .

Subtraindo $\mathbf{M}_V^* \cdot \mathbf{V}'$ de ambos lados da equação acima:

$$\Rightarrow \mathbf{M}_V^* \cdot (\mathbf{V}^{*'} + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) - \mathbf{M}_V^* \cdot \mathbf{V}' = \mathbf{M}_V \cdot \mathbf{V}' - \mathbf{M}_V^* \cdot \mathbf{V}' + \Delta_{\mathbf{B}_V} \cdot \mathbf{V}$$

$$\Rightarrow \mathbf{M}_V^* \cdot (\mathbf{V}^{*'} - \mathbf{V}' + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) = (\mathbf{M}_V - \mathbf{M}_V^*) \cdot \mathbf{V}' + \Delta_{\mathbf{B}_V} \cdot \mathbf{V}$$

$$\Rightarrow \mathbf{M}_V^* \cdot (\Delta \mathbf{V}' + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) = \Delta_{\mathbf{M}_V} \cdot \mathbf{V}' + \Delta_{\mathbf{B}_V} \cdot \mathbf{V}$$

onde $\Delta_{\mathbf{M}_V} = (\mathbf{M}_V - \mathbf{M}_V^*)$ é o erro de modelagem da matriz \mathbf{M}_V^* .

$$\Rightarrow (\Delta \mathbf{V}' + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) = (\mathbf{M}_V^*)^{-1} \cdot (\Delta_{\mathbf{M}_V} \cdot \mathbf{V}' + \Delta_{\mathbf{B}_V} \cdot \mathbf{V})$$

Verifica-se que, se o modelo nominal disponível descrever exatamente o modelo real do robô, (ou seja, $\Delta_{\mathbf{M}_V} = (\mathbf{M}_V - \mathbf{M}_V^*) = \mathbf{0}$ e $\Delta_{\mathbf{B}_V} = (\mathbf{B}_V - \mathbf{B}_V^*) = \mathbf{0}$), então:

$$\Rightarrow (\Delta \mathbf{V}' + \mathbf{K}_p \cdot \Delta \mathbf{V} + \mathbf{K}_i \cdot \int \Delta \mathbf{V} dt) = \mathbf{0}$$

Ou seja, o sistema em malha fechada seria linear e desacoplado.

- **Controle Estabilizante:**

Controle por Referência Variável

- Configuração final desejada: $\mathbf{q}^* = [x^* \ y^* \ \theta^*]^T$.
- Requer apenas medição da configuração $\mathbf{q} = [x \ y \ \theta]^T$.
- Controlador baseado na realimentação de erro nas variáveis:

$$\mathbf{L}(t) = [l(t) \ \theta(t)]^T = \int \mathbf{V}(t).dt$$

onde $l(t) = \int v(t).dt$ é o comprimento percorrido pelo robô no intervalo de tempo de integração e $\theta(t) = \int \omega(t).dt$ é o ângulo de orientação do robô. Tomando a transformada de Laplace, temos:

$$s.\mathbf{L}(s) = \mathbf{V}(s)$$

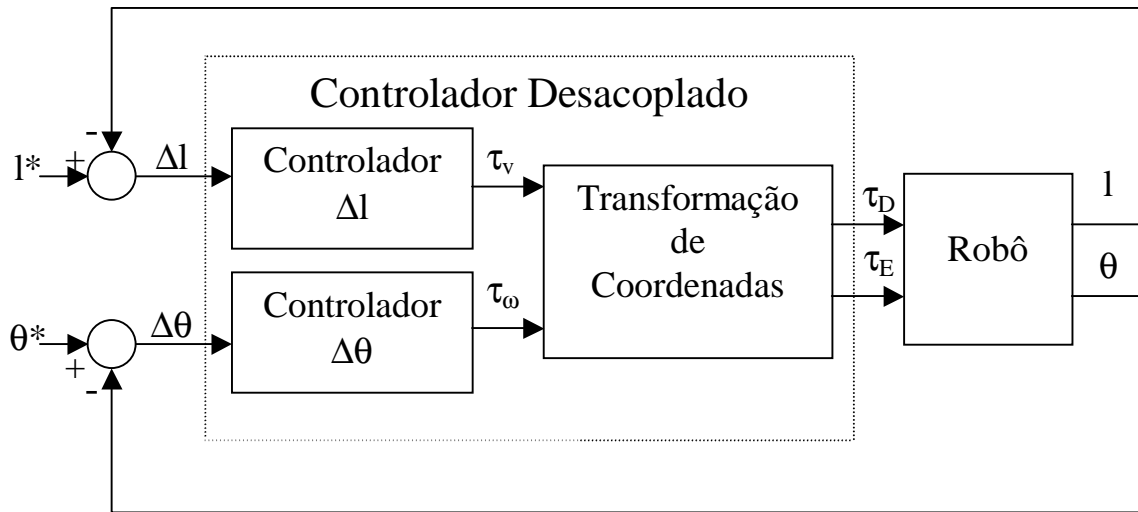
- Referências variáveis de posição e ângulo permitem desacoplar o controle de $l(t)$ e $\theta(t)$.
- Referências variáveis de posição e ângulo permitem atingir a posição e orientação final, mesmo sendo o sistema subatuado.
- Controladores lineares desacoplados em $l(t)$ e $\theta(t)$ atuam em paralelo de modo a garantir a posição e orientação final.
- Controladores lineares desacoplados em $l(t)$ e $\theta(t)$ (para deslocamento e orientação) são projetados a partir do modelo linear obtido aplicando a transformada de Laplace ao modelo dinâmico em \mathbf{V} :

$$\boldsymbol{\tau}_V = \mathbf{M}_V.\mathbf{V}' + \mathbf{B}_V.\mathbf{V} \quad \Rightarrow \quad \boldsymbol{\tau}_V(s) = s.\mathbf{M}_V.\mathbf{V}(s) + \mathbf{B}_V.\mathbf{V}(s)$$

ou, substituindo $\mathbf{V}(s)$ por $s.\mathbf{L}(s)$:

$$\boldsymbol{\tau}_V(s) = s.\mathbf{M}_V.s.\mathbf{L}(s) + \mathbf{B}_V.s.\mathbf{L}(s) = s.[s.\mathbf{M}_V + \mathbf{B}_V].\mathbf{L}(s)$$

- Controladores Desacoplados para $l(t)$ e $\theta(t)$:



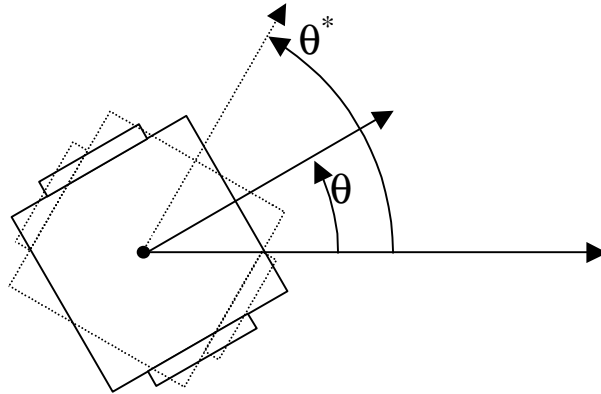
τ_v = entrada virtual, que controla $l(t)$, (ou $v(t)$).

τ_ω = entrada virtual, que controla $\theta(t)$, (ou $\omega(t)$).

- A transformação de coordenadas transforma as entradas de controle virtuais em entradas de controle em referencial de atuador (τ_D e τ_E ou, diretamente, e_D e e_E).
- Para obter esta transformação é necessário diagonalizar a matriz de transferência $s.[s.\mathbf{M}_V + \mathbf{B}_V]$, de modo a desacoplar o controle de $l(t)$ e $\theta(t)$.
- Problema: como formular o erro $\Delta l(t)$, uma vez que o mesmo não é mensurável.
- Solução: utilizar referência variável, de modo a reduzir o problema às situações triviais.
 - a) Quando na posição desejada, girar (sem sair da posição atual) para atingir a orientação desejada.
 - b) Quando na orientação desejada, andar em frente para atingir a posição desejada na direção da orientação atual.

- Caso trivial a). Erro de orientação a ser realimentado para o controlador de orientação quando o robô já está na posição final desejada:

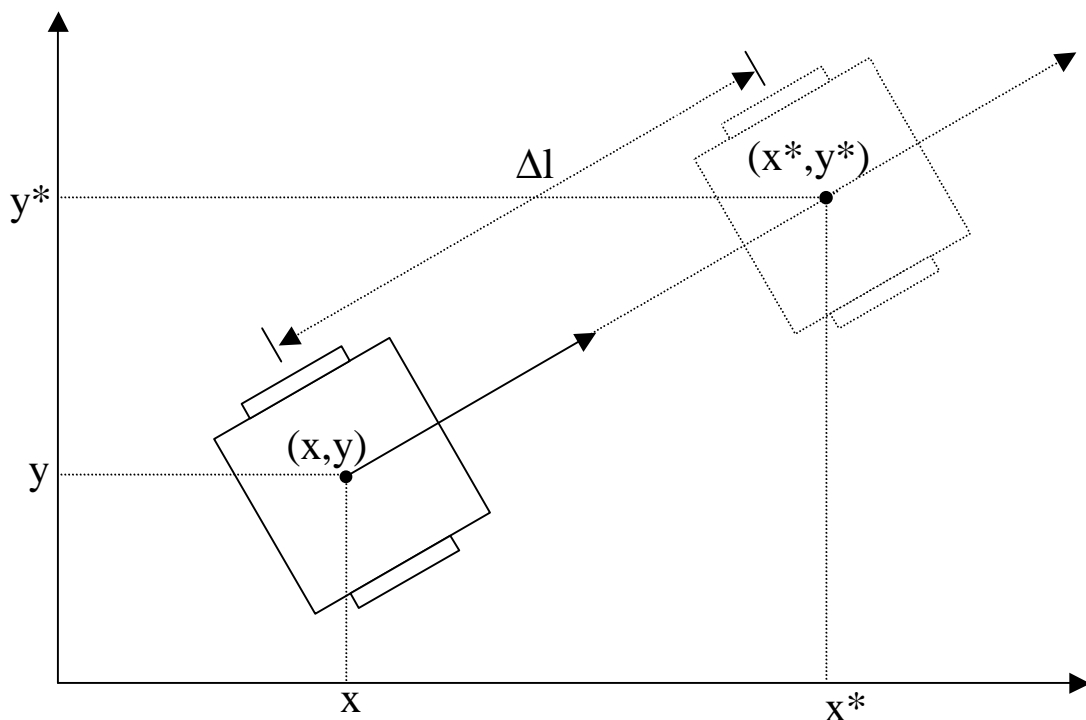
$$\Delta\theta = \theta^* - \theta$$



- Caso trivial b). Erro de posição a ser realimentado para o controlador de deslocamento quando o robô já está orientado em direção à posição final desejada:

$$\Delta l = [(\Delta x)^2 + (\Delta y)^2]^{1/2}$$

onde: $\Delta x = (x^* - x)$ $\Delta y = (y^* - y)$



- Referência Variável para controle de Posição:

- Deseja-se atingir a posição final (x^*,y^*) , não importando com que orientação se chegue à mesma (θ^* é escolhida variável, de modo a apontar sempre para o alvo).
- Define-se a referência variável de posição (x_r^*,y_r^*) , localizada na projeção da posição final (x^*,y^*) sobre a reta sobre a qual o robô está orientado.
- Os erros de orientação $\Delta\theta$ e de deslocamento Δl a serem fornecidos a os controladores são obtidos a partir de:

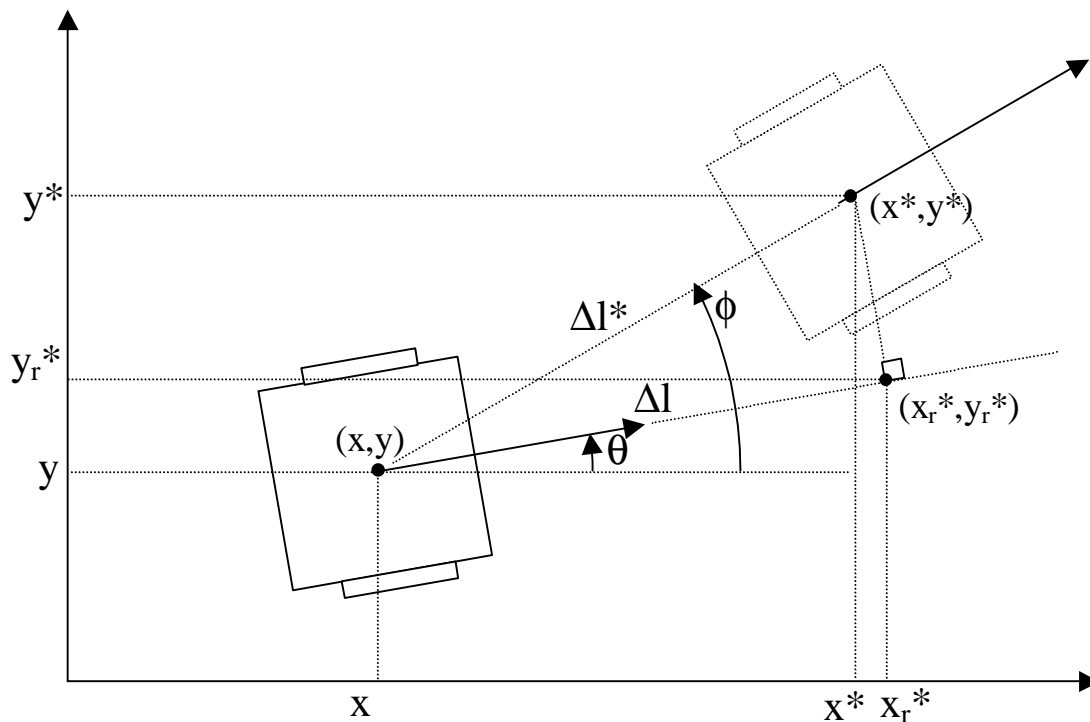
$$\Delta x = x^* - x \qquad \Delta y = y^* - y$$

$$\theta^* = \phi = \tan^{-1}(\Delta y / \Delta x)$$

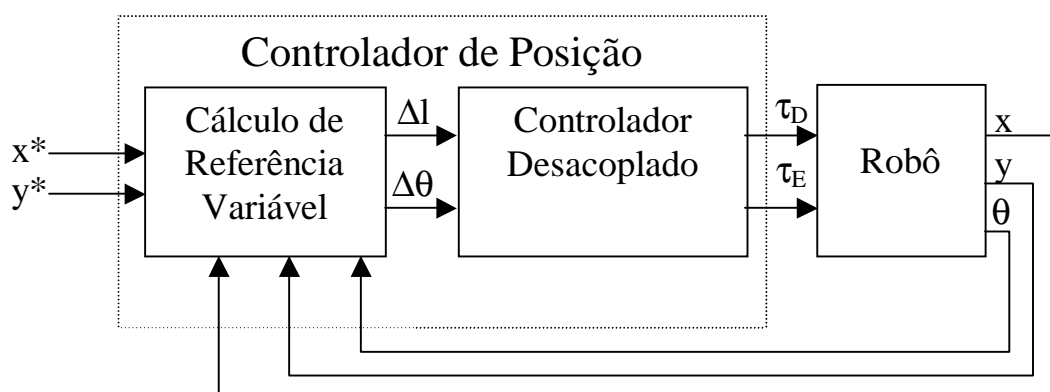
$$\Delta l^* = [(\Delta x)^2 + (\Delta y)^2]^{1/2}$$

$$\Delta\theta = \theta^* - \theta$$

$$\Delta l = \Delta l^* \cdot \cos(\Delta\theta)$$



- O erro de realimentação Δl fornecido ao controlador de deslocamento é calculado em relação à posição de referência variável (x_r^*, y_r^*) mais próxima da posição desejada (x^*, y^*) e localizada sobre a reta que o robô percorreria com a orientação atual θ . A posição (x_r^*, y_r^*) varia de acordo com a mudança de orientação do robô.
- Como (x_r^*, y_r^*) está na frente do robô, neste (e somente neste caso), pode-se saber exatamente qual é o erro de deslocamento Δl : a distância euclidiana entre (x, y) e (x_r^*, y_r^*) .
- O controlador de deslocamento guia o robô ao ponto (x_r^*, y_r^*) , que é o mais próximo ao alvo na direção da orientação atual do robô.
- O controlador de orientação orienta o robô em direção ao alvo. É responsável por diminuir o erro de realimentação $\Delta\theta$ de modo a alinhar o robô com a reta que aponta para a posição final desejada (x^*, y^*) . A orientação desejada θ^* é variável e sempre alinhada com esta reta.
- Se o controlador de orientação for mais rápido do que o controlador de deslocamento e for capaz de garantir que $\Delta\theta \rightarrow 0$ e se o controlador de deslocamento garantir que $\Delta l \rightarrow 0$, então, $\cos(\Delta\theta) \rightarrow +1$ e, conseqüentemente, devido a que $\Delta l^* = \Delta l / \cos(\Delta\theta)$, temos que $\Delta l^* \rightarrow 0$, ou seja, o robô atinge a posição final.



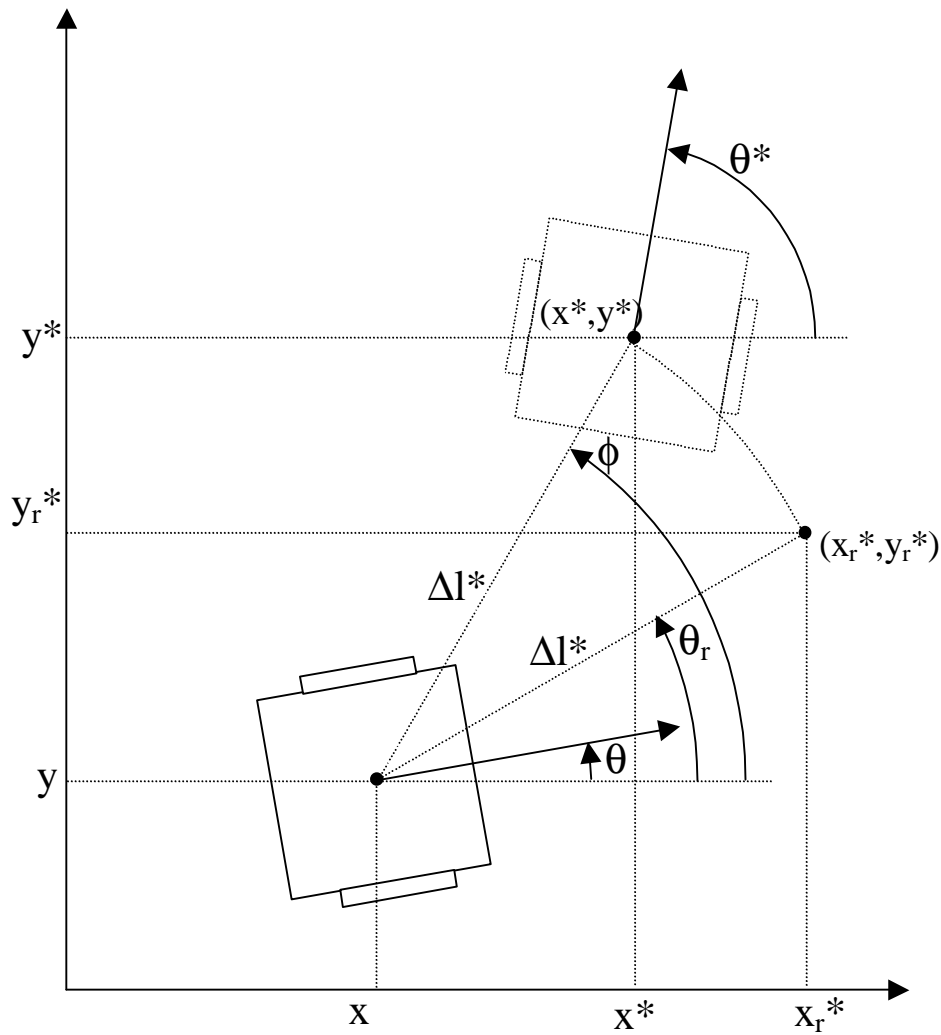
- Referência Móvel para controle de Posição e Orientação:
 - Deseja-se atingir a posição final (x^*, y^*) , com uma orientação θ^* especificada.
 - Solução: usar o controlador de posição anterior usando como referência uma posição móvel (x_r^*, y_r^*) .
 - Esta referência móvel deve fazer com que, inicialmente, o robô seja atraído para a reta de aproximação, (reta que passa pelo alvo (x^*, y^*) com a orientação θ^* final desejada).
 - À medida em que o robô chegar perto da reta de aproximação, a referência (x_r^*, y_r^*) deve ser progressivamente movida em direção ao alvo (x^*, y^*) .
 - Para alcançar este objetivo, o erro angular a ser minimizado pelo controlador é definido como:

$$\Delta\theta = \theta_r - \theta = (\phi - \theta^*) + (\phi - \theta)$$

onde $\phi = \tan^{-1}(\Delta y / \Delta x) = \tan^{-1}((y^* - y) / (x^* - x))$. O ângulo de referência móvel é dado por:

$$\theta_r = 2 \cdot \phi - \theta^*$$

- O erro $\Delta\theta$ possui duas parcelas:
 - a) Erro angular $(\phi - \theta^*)$ entre a reta que liga o robô ao alvo e a reta de aproximação. Quando o robô alcança a reta de aproximação, este erro é nulo e o problema recai no caso anterior.
 - b) Erro $(\phi - \theta)$, que guia o robô para apontar para o alvo.

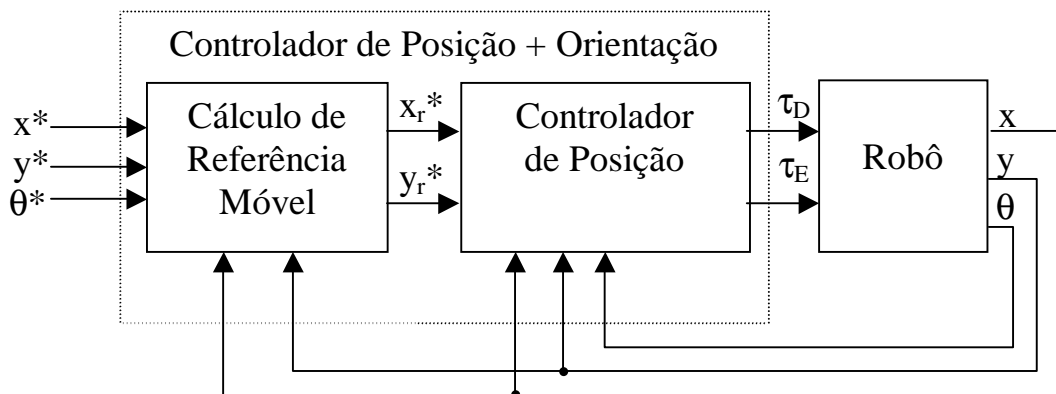


- A referência móvel a ser fornecida ao controlador de posição pode ser vista como uma rotação do alvo em torno da posição atual do robô. O valor desta rotação é de $(\theta_r - \phi) = (\phi - \theta^*)$ radianos.

- Assim, a referência móvel é dada por:

$$x_r^* = x + \Delta l^* \cdot \cos(\theta_r) \quad y_r^* = y + \Delta l^* \cdot \sin(\theta_r)$$

$$\text{onde } \Delta l^* = [(\Delta x)^2 + (\Delta y)^2]^{1/2}$$



Observações:

- O projeto dos controladores deve ser tal que o controlador de orientação seja mais rápido que o de deslocamento, uma vez que erros de orientação rapidamente amplificam erros de deslocamento.
- Quando a entrada é saturada, (tensão $\|\mathbf{E}\| \leq \|\mathbf{E}\|_{\text{Max}}$, ou torque $\|\boldsymbol{\tau}\| \leq \|\boldsymbol{\tau}\|_{\text{Max}}$), pelo mesmo motivo anterior, deve-se priorizar a entrada responsável pelo controle de orientação, τ_{ω} , em relação à entrada responsável pelo controle de deslocamento, τ_v .
- Assim, deve-se primeiro verificar se o valor de τ_{ω} requerido pode ser fornecido pela fonte de energia.
 - Em caso negativo (não há energia suficiente), satura-se τ_{ω} no valor máximo disponível e impõe-se τ_v nulo.
 - Em caso afirmativo (há energia suficiente para τ_{ω}), τ_{ω} assume o valor calculado e verifica-se se o que resta de capacidade disponível da fonte é suficiente para alimentar τ_v .
 - Em caso negativo (não há energia suficiente para τ_v), satura-se τ_v no valor do resto de entrada disponível.
 - Em caso afirmativo (há energia suficiente também para τ_v), τ_v assume o valor calculado.

- **Gerção de Trajetória:**

- Controladores de trajetória requerem que uma trajetória de referência seja especificada.
- A trajetória pode ser caracterizada como uma curva geométrica contínua entre a configuração inicial e final, (denominada caminho), à qual se associam restrições temporais.
- Em muitos casos, a trajetória de referência deve ser derivável ou duplamente derivável.
- Para robôs não holonômicos, a trajetória gerada deve respeitar as restrições não holonômicas.
- Outras restrições podem ter que ser levadas em conta ao especificar uma trajetória, por exemplo: velocidades e/ou acelerações máximas admissíveis ao robô, comprimento mínimo, etc.

Interpolação por Polinômios Cúbicos

Dado o parâmetro λ , com $0 \leq \lambda \leq 1$, a trajetória do robô pode ser especificada através de polinômios cúbicos nas componentes x e y da posição do robô, expressos em função do parâmetro λ :

$$x(\lambda) = a_0 + a_1 \cdot \lambda + a_2 \cdot \lambda^2 + a_3 \cdot \lambda^3$$

$$y(\lambda) = b_0 + b_1 \cdot \lambda + b_2 \cdot \lambda^2 + b_3 \cdot \lambda^3$$

onde $a_0, a_1, a_2, a_3, b_0, b_1, b_2$ e b_3 são parâmetros a determinar. Considera-se que $\lambda = 0$ quando o robô está na sua posição inicial $(x(0), y(0)) = (x_i, y_i)$, assim como $\lambda = 1$ quando o robô está na sua posição final $(x(1), y(1)) = (x_f, y_f)$.

O ângulo de orientação θ do robô deve satisfazer as restrições não holonômicas ao longo de toda a trajetória:

$$\theta(\lambda) = \tan^{-1}(dy/dx) = \tan^{-1}((dy/d\lambda)/(dx/d\lambda))$$

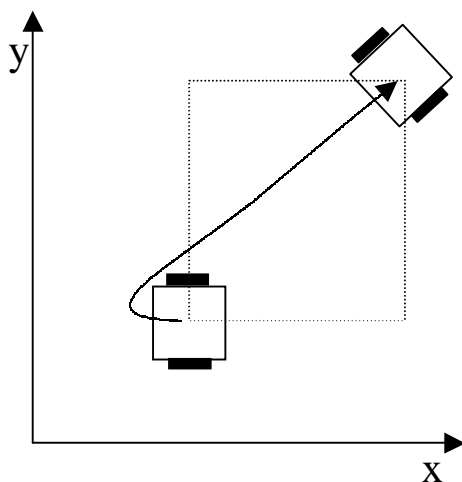
Definindo $d(\lambda) = (b_1 + 2.b_2.\lambda + 3.b_3.\lambda^2)/(a_1 + 2.a_2.\lambda + 3.a_3.\lambda^2)$

$$\Rightarrow \theta(\lambda) = \tan^{-1}(d(\lambda)) \quad \Rightarrow \quad \alpha(\lambda) = \tan(\theta(\lambda))$$

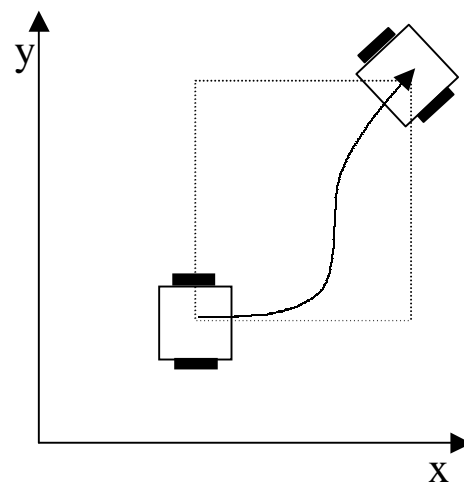
Aplicando as seis condições de contorno de modo a impor que a trajetória inicie na configuração $q_i = (x_i, y_i, \theta_i)$ e termine na configuração $q_f = (x_f, y_f, \theta_f)$, obtemos um sistema de seis equações e oito incógnitas (os coeficientes $a_0, a_1, a_2, a_3, b_0, b_1, b_2, b_3$).

Como o número de coeficientes é maior do que o número de equações, podemos escolher dois coeficientes livres de modo a obter soluções mais aprimoradas e que intuitivamente sejam “mais inteligentes”.

O critério utilizado para aprimorar a solução é fazer com que, na medida do possível, o robô não execute movimentos afastando-se do alvo (para trás). Matematicamente, isto é alcançado através de soluções para as quais o polinômio interpolador não apresente máximos nas direções x e/ou y.



a) Sem aprimoramento.



b) Trajetória aprimorada.

Dependendo das condições de contorno, para evitar divisões por zero na solução, torna-se necessário adotar diferentes parâmetros livres. Seja $\Delta x = x_f - x_i$ e $\Delta y = y_f - y_i$, definindo δ como um pequeno intervalo angular, (para efeitos práticos, $\delta \cong 1^\circ$), temos o seguinte procedimento para geração da trajetória:

i. **Se** $\theta_i \in [(\pi/2 - \delta), (\pi/2 + \delta)]$ e $\theta_f \in [(\pi/2 - \delta), (\pi/2 + \delta)]$

$$b_1 = \Delta y \quad (\text{coeficiente livre})$$

$$b_2 = 0 \quad (\text{coeficiente livre})$$

$$a_0 = x_i$$

$$a_1 = 0$$

$$a_2 = 3 \cdot \Delta x$$

$$a_3 = -2 \cdot \Delta x$$

$$b_0 = y_i$$

$$b_3 = \Delta y - b_1 - b_2$$

ii. **Senão, se** $\theta_i \in [(\pi/2 - \delta), (\pi/2 + \delta)]$

$$a_3 = -\Delta x/2 \quad (\text{coeficiente livre})$$

$$b_3 = \text{qualquer valor} \quad (\text{coeficiente livre})$$

$$a_0 = x_i$$

$$a_1 = 0$$

$$a_2 = \Delta x - a_3$$

$$b_0 = y_i$$

$$b_1 = 2(\Delta y - \alpha_f \cdot \Delta x) - \alpha_f \cdot a_3 + b_3$$

$$b_3 = (2 \cdot \alpha_f \cdot \Delta x - \Delta y) + \alpha_f \cdot a_3 - 2 \cdot b_3$$

iii. **Senão, se** $\theta_f \in [(\pi/2 - \delta), (\pi/2 + \delta)]$

$$a_1 = 3 \cdot \Delta x/2 \quad (\text{coeficiente livre})$$

$$b_2 = \text{qualquer valor} \quad (\text{coeficiente livre})$$

$$a_0 = x_i$$

$$a_2 = 3 \cdot \Delta x - 2 \cdot a_1$$

$$a_3 = a_1 - 2 \cdot \Delta x$$

$$b_0 = y_i$$

$$b_1 = \alpha_i \cdot a_1$$

$$b_3 = \Delta y - \alpha_i \cdot a_1 - b_2$$

iv. **Senão**

$$a_1 = \Delta x \quad (\text{coeficiente livre})$$

$$a_2 = 0 \quad (\text{coeficiente livre})$$

$$a_0 = x_i$$

$$a_3 = \Delta x - a_1 - a_2$$

$$b_0 = y_i$$

$$b_1 = \alpha_i \cdot a_1$$

$$b_2 = 3(\Delta y - \alpha_f \cdot \Delta x) + 2 \cdot (\alpha_f - \alpha_i) \cdot a_1 + \alpha_f \cdot a_2$$

$$b_3 = 3 \cdot \alpha_f \cdot \Delta x - 2 \cdot \Delta y - (2 \cdot \alpha_f - \alpha_i) \cdot a_1 - \alpha_f \cdot a_2$$

v. **Fazer λ variar de 0 a 1**

vi. **Calcular**

$$x(\lambda) = a_0 + a_1 \cdot \lambda + a_2 \cdot \lambda^2 + a_3 \cdot \lambda^3$$

$$y(\lambda) = b_0 + b_1 \cdot \lambda + b_2 \cdot \lambda^2 + b_3 \cdot \lambda^3$$

$$\theta(\lambda) = \tan^{-1}((b_1 + 2 \cdot b_2 \cdot \lambda + 3 \cdot b_3 \cdot \lambda^2) / (a_1 + 2 \cdot a_2 \cdot \lambda + 3 \cdot a_3 \cdot \lambda^2))$$

Observação:

- Os passos i, ii e iii correspondem a situações singulares (θ_i e/ou θ_f iguais a $\pm 90^\circ$).
- O passo iv corresponde ao caso geral (caso não singular).
- Os passos v e vi correspondem à geração de trajetória propriamente dita.

Reparametrização da Trajetória

Como parâmetro λ é adimensional, ($0 \leq \lambda \leq 1$), torna-se necessário reparametrizar a trajetória, de modo a associar o parâmetro λ com o tempo, de acordo com algum perfil de velocidade especificado para a curva geométrica $(x(\lambda), y(\lambda))$. Seja:

$$x(\lambda) = a_0 + a_1 \cdot \lambda + a_2 \cdot \lambda^2 + a_3 \cdot \lambda^3$$

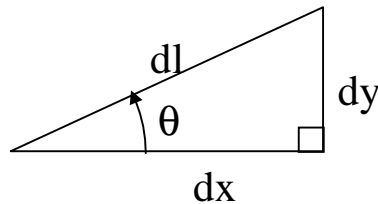
$$y(\lambda) = b_0 + b_1 \cdot \lambda + b_2 \cdot \lambda^2 + b_3 \cdot \lambda^3$$

Definindo o operador $D(.) = d./d\lambda$, temos:

$$Dx = dx(\lambda)/d\lambda = a_1 + 2 \cdot a_2 \cdot \lambda + 3 \cdot a_3 \cdot \lambda^2$$

$$Dy = dy(\lambda)/d\lambda = b_1 + 2 \cdot b_2 \cdot \lambda + 3 \cdot b_3 \cdot \lambda^2$$

Assim, para um deslocamento infinitesimal, dl , do robô, temos:



$$dl = [(dx)^2 + (dy)^2]^{1/2} = [(Dx)^2 + (Dy)^2]^{1/2} \cdot d\lambda$$

$$\Rightarrow d\lambda/dl = 1/[(Dx)^2 + (Dy)^2]^{1/2}$$

$$\Rightarrow \cos(\theta) = (dx/dl) = (dx/d\lambda) \cdot (d\lambda/dl) = Dx/[(Dx)^2 + (Dy)^2]^{1/2}$$

$$\Rightarrow \text{sen}(\theta) = (dy/dl) = (dy/d\lambda) \cdot (d\lambda/dl) = Dy/[(Dx)^2 + (Dy)^2]^{1/2}$$

$$\Rightarrow \tan(\theta) = Dy/Dx$$

A velocidade linear é dada por:

$$v(t) = dl/dt = (dl/d\lambda).(d\lambda/dt) = [(Dx)^2 + (Dy)^2]^{1/2}.d\lambda/dt$$

$$\Rightarrow d\lambda/dt = v(t)/[(Dx)^2 + (Dy)^2]^{1/2}$$

Assim, dado o perfil de velocidade $v(t)$, a expressão acima pode ser integrada de modo a reparametrizar a trajetória, (dado t e o perfil $v(t)$, obter o valor correspondente de λ). O Comprimento percorrido, l , é dado por:

$$l = \int v(t).dt = \int (dl/dt).dt = \int [(Dx)^2 + (Dy)^2]^{1/2}.d\lambda$$

Onde o intervalo de integração para t é $[0, t_{\max}]$, que corresponde ao intervalo $[0, 1]$ de variação de λ , sendo t_{\max} a duração especificada para a trajetória.

A velocidade angular do robô para um dado λ , pode ser obtida a partir da sua velocidade linear, através da derivada:

$$d(\tan(\theta))/dt = (1/\cos^2(\theta)).d\theta/dt = (1/\cos^2(\theta)).\omega$$

$$\Rightarrow \omega = \cos^2(\theta).[d(\tan(\theta))/d\lambda].d\lambda/dt$$

mas, $\cos^2(\theta) = (Dx)^2/[(Dx)^2 + (Dy)^2]$, $\tan(\theta) = Dy/Dx$. Assim:

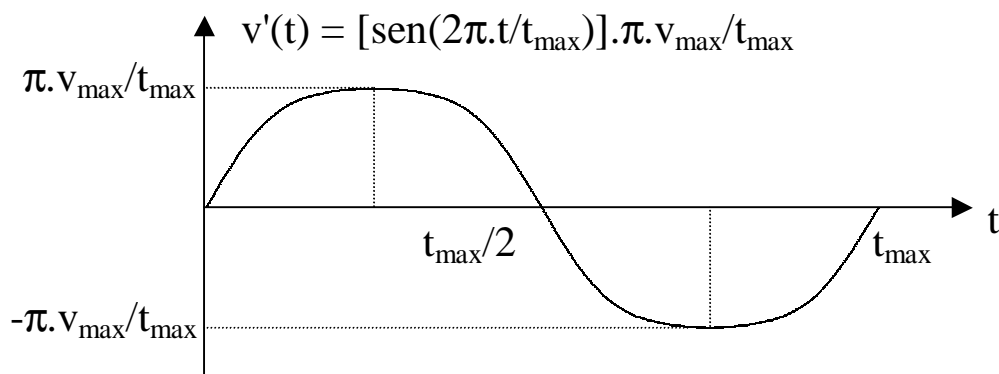
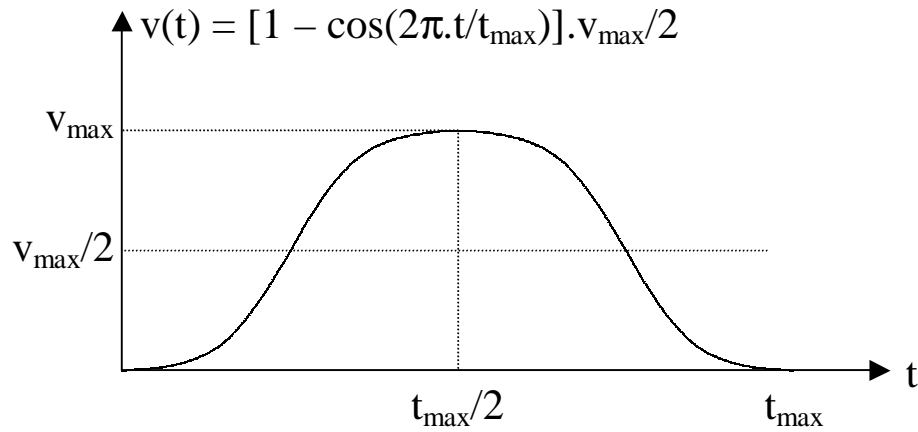
$$\omega(t) = v(t).[D^2y.Dx - D^2x.Dy]/[(Dx)^2 + (Dy)^2]^{3/2}$$

onde: $D^2x = D(Dx) = 2.a_2 + 6.a_3.\lambda$
 $D^2y = D(Dy) = 2.b_2 + 6.b_3.\lambda$

Então, para um dado λ , o raio de giro $r(t) = v(t)/\omega(t)$ é dado por:

$$r(\lambda) = [(Dx)^2 + (Dy)^2]^{3/2}/[D^2y.Dx - D^2x.Dy]$$

Perfil de Velocidade Cossenoidal:



O valor máximo da velocidade, v_{\max} , pode ser obtido em função do comprimento percorrido, l :

$$l = \int v(t).dt = \int [1 - \cos(2\pi.t/t_{\max})].v_{\max}/2.dt$$

Integrando no intervalo $[0, t_{\max}]$, obtemos:

$$l = v_{\max}.t_{\max}/2$$

Assim:

$$v_{\max} = 2.l/t_{\max}$$