Robot Kinematics

Henrik I Christensen

Centre for Autonomous Systems

Kungl Tekniska Högskolan

hic@kth.se

March 27, 2006

Outline

Motion Control

- Requirements
  - Kinematic / dynamic model of the robot
  - Model of ground/wheel interaction
  - Definition of required motion $\rightarrow$ velocity / position control
  - Design of control law to satisfy constriants

Mobile Robot Kinematics

- Model of mechanical behaviour of robot for design and control
- Models can be used both for mobile systems and manipulators
- Manipulators allow "direct" estimation of position, which is not always true for mobile systems
- Position to be derived from integration over time
- Motion is not free. There are constraints to be considered in the design and control generation.

Outline

Coordinate Systems

- Points in space can be described by their position $\vec{p}$

$$\vec{p} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

- For structures the orientation is also of interest (α, ϕ, θ)
- Coordinate transformations are essential to modelling of robots
- The basis for the coordinate system decides on the simplicity of the model and control.

Transformations

- Translation is easy: $\vec{p_r} = \vec{p_0} + \vec{p_t}$
- Rotation can modelled through a rotation matrix

$$\vec{p_1} = \mathbf{R}\vec{p_0}$$

- Rotation α around X axis

$$\mathbf{R}_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

- and similarly for Y and Z rotations
- Rotations are commutative $\mathbf{R}_{xy} = \mathbf{R}_x\mathbf{R}_y$

Homogeneous Coordinates

- It would be useful to integrate scaling, translation and rotation into a single representation!
- Use of format

$$\vec{p} = \begin{bmatrix} wX \\ wY \\ wZ \\ w \end{bmatrix}$$

- Now

$$x = \frac{wX}{w} \quad y = \frac{wY}{w} \quad z = \frac{wZ}{w}$$

Transformations

- Now a transformation is a 4×4 matrix with a structure of

$$\mathbf{T} = \left[\begin{array}{ccc} \mathbf{R}_{3\times3} & | & \vec{p}_{3\times1} \\ -- & & -- \\ \vec{f}_{1\times3} & | & s_{1\times1} \end{array} \right] = \left[\begin{array}{ccc} \text{Rotation} & | & \text{translation} \\ \text{matrix} & | & \text{vector} \\ -- & & -- \\ \text{perspective} & | & \text{scale} \\ \text{transf} & | & \text{factor} \end{array} \right]$$

Homogenous Transformations

- Rotation around X

$$\mathbf{T}_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- And translation

$$\mathbf{T}_{trans} = \begin{bmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Offer a unified model frequently used.

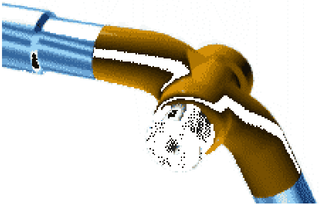
Types of Joints

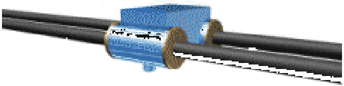
- Revolute – pure rotation
- Planar – motion in the plane
- Cylindrical – motion along a fixed axis
- Prismatic – translation wo rotation along axis
- Spherical – 3D rotation
- Screw – motion along threaded axis

Kinematics

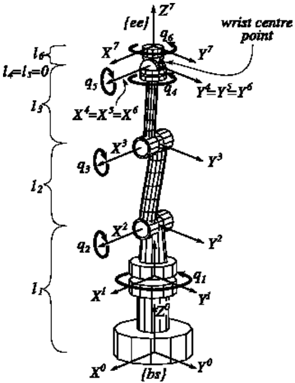
H.I. Christensen

Joint examples



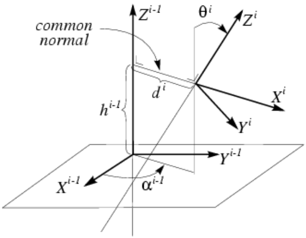


Reference frames



- Description of motion across the difference coordinate systems
- Control is in each axis. What is the overall motion

Denavit-Hartenberg Representation



- Relations are described by 4 parameters
- h_{i-1} shift along Z_{i-1} axis
- α_i is rotation X axis
- θ_i is the rotation around the X axis
- d_i is the translation along the X_i axis

The resulting transformation

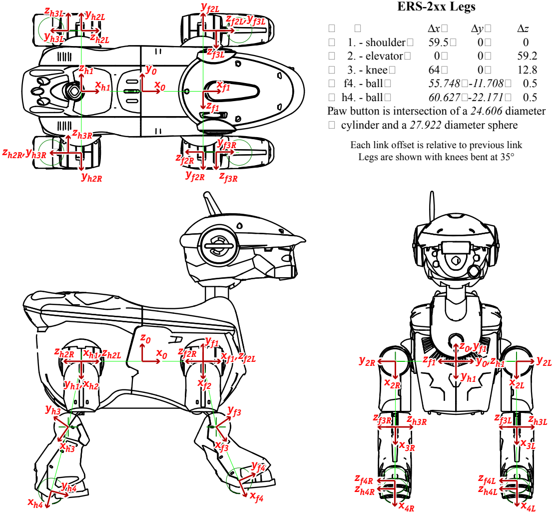
$$
{}_{i-1}^{\;\;i}\boldsymbol{T} = \boldsymbol{R}(Z,\alpha^{i-1})\,\boldsymbol{Tr}(Z,h^{i-1})\,\boldsymbol{Tr}(X,d^{i})\,\boldsymbol{R}(X,\theta^{i})
$$

$$
= \begin{pmatrix} c_\alpha^{i-1} & -s_\alpha^{i-1} & 0 & 0 \\ s_\alpha^{i-1} & c_\alpha^{i-1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & h^{i-1} \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & d^{i} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & c_\theta^{i} & -s_\theta^{i} & 0 \\ 0 & s_\theta^{i} & c_\theta^{i} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},
$$

Reference frames

- The DH representation is widely used.
- There are many reference frames for articulated systems
- The estimation and control is critically dependent on consideration of reference frames. Keep this is mind.

Sony AIBO ERS-220 reference frames



ERS-2xx Legs

	Δx	Δy	Δz
1. - shoulder	59.5	0	0
2. - elevator	0	0	59.2
3. - knee	64	0	12.8
f4. - ball	*55.748*	*-11.708*	0.5
h4. - ball	*60.627*	*-22.171*	0.5

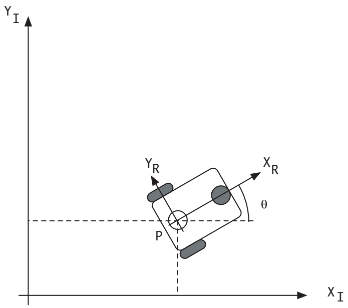
Paw button is intersection of a *24.606* diameter cylinder and a *27.922* diameter sphere

Each link offset is relative to previous link
Legs are shown with knees bent at 35°

Simplified reference frames

- Inertial reference frame (I)
- Robot references frame (R)
- Robot pose

$$\xi_I = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$$



Transformation between reference frames

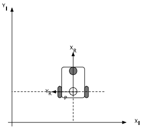
- The relation between the references frame is through the standard orthogonal rotation transformation:

$$R(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Enable handling of motion between frames

$$\dot{\xi}_R = R(\theta)\dot{\xi}_I$$

Simple 90° rotation



- Now with $\dot{\xi}_I = R(\theta)\dot{\xi}_R$
- $\dot{\xi}_I = R(\frac{\pi}{2})\dot{\xi}_R$

$$R(\frac{\pi}{2}) = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Implies

$$\dot{\xi}_I = R(\frac{\pi}{2})\dot{\xi}_R = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{y} \\ -\dot{x} \\ \dot{\theta} \end{bmatrix}$$

Outline

Kinematic modelling

- Goal:
  - Determine the robot speed $\dot{\xi} = \left[\begin{array}{ccc} \dot{x} & \dot{y} & \dot{\theta} \end{array} \right]^T$ as a function of wheel speed $\dot{\varphi}$, steering angle β, steering speed $\dot{\beta}$ and the geometric parameters of the robot.
  - Forward kinematics

$$\dot{\xi} = \left[\begin{array}{c} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{array} \right] = f(\dot{\varphi}_1, \ldots, \dot{\varphi}_n, \beta_1, \ldots, \beta_m, \dot{\beta}_1, \ldots, \dot{\beta}_m)$$

  - Inverse kinematics

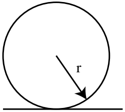
$$\left[\begin{array}{ccccccccc} \dot{\varphi}_1 & \ldots & \dot{\varphi}_n & \beta_1 & \ldots & \beta_m & \dot{\beta}_1 & \ldots & \dot{\beta}_m \end{array} \right]^T = f(\dot{x}, \dot{y}, \dot{\theta})$$

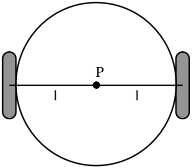
  - Why not

$$\left[\begin{array}{c} x \\ y \\ \theta \end{array} \right] = f(\varphi_1, \ldots, \varphi_n, \beta_1, \ldots, \beta_m)$$

the relation is not straight forward. See later.

Forward kinematic model – differential drive





- Assume a set up with two drive wheels. Wheels have radius r, and a places at a distance l from the center.
- Wheels rotate at speeds $\dot{\varphi}_1$ and $\dot{\varphi}_2$
- Prediction of the motion of the robot motion in the global frame

$$\dot{\xi}_I = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = f(l, r, \theta, \dot{\varphi}_1, \dot{\varphi}_2)$$

Differential drive model

- Given $\dot{\xi}_I = R(\theta)^{-1}\dot{\xi}_R$
- Speed of each wheel is $r\dot{\varphi}_i$, the translational speed is the average velocity

$$\dot{x}_R = r\frac{\dot{\varphi}_1 + \dot{\varphi}_2}{2}$$

- The instantaneous rotation of P for one wheel is

$$\omega_1 = \frac{r\dot{\varphi}_1}{2l}$$

- The total rotation is then

$$\dot{\theta} = \frac{r}{2l}(\dot{\varphi}_1 - \dot{\varphi}_2)$$

Differential drive model

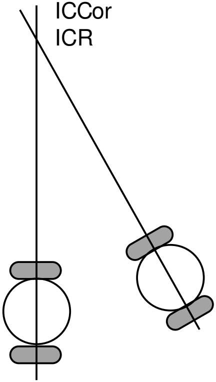
- Given $\dot{\xi}_I = R(\theta)^{-1}\dot{\xi}_R$
- The full model is then:

$$\dot{\xi}_I = R(\theta)^{-1}\frac{r}{2}\begin{bmatrix} \dot{\varphi}_1 + \dot{\varphi}_2 \\ 0 \\ \frac{\dot{\varphi}_1 - \dot{\varphi}_2}{l} \end{bmatrix}$$

- The rotation matrix is trivial to invert

$$R(\theta)^{-1} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Differential motion



- The Instantaneous Center of Rotation (ICR) or Instantaneous Center of Curvature is relevant. Characterised by rotation ω and radius R:

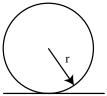
$$R = l\frac{\dot{\varphi}_1 + \dot{\varphi}_2}{\dot{\varphi}_1 - \dot{\varphi}_2}$$

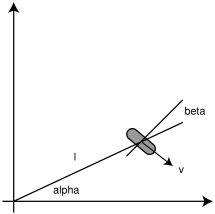
- With $\dot{\varphi}_1 = \dot{\varphi}_2 \Rightarrow R = \infty$

Kinematic constraints

- Assumptions
  - Plane of wheel is always vertical
  - Single point of contact with surface
  - Motion is purely by rolling (no slippage)
  - Rotation of wheel is around the vertical axis

Fixed wheel constraint





- The motion of the wheel must be in the plane of the wheel
- Speed of wheel $v = r\dot{\varphi}$
- The motion must then be (rolling constraint):

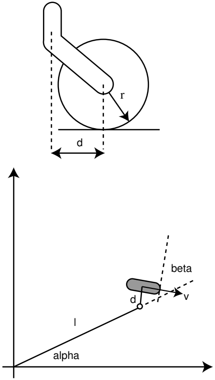
$$\begin{bmatrix} \sin(\alpha+\beta) & -\cos(\alpha+\beta) & (-l)\cos(\beta) \end{bmatrix} R(\theta)\dot{\xi}_I - r\dot{\varphi} = 0$$

- Motion in the orthogonal plane must be zero (sliding constraint), i.e.

$$\begin{bmatrix} \cos(\alpha+\beta) & \sin(\alpha+\beta) & l\sin(\beta) \end{bmatrix} R(\theta)\dot{\xi}_I = 0$$

- Wheels cannot slide sideways!
- Similar models can be developed for Swedish and Spherical wheels (see the book!)

Steering wheel constraint



- The motion of the wheel must be in the plane of the wheel
- Speed of wheel $v = r\dot{\varphi}$, Rotation speed is $\dot{\beta}$
- The motion must then be (rolling constraint):

$$\left[\sin(\alpha+\beta) \quad -\cos(\alpha+\beta) \quad (-l)\cos(\beta) \right] R(\theta)\dot{\xi}_I - r\dot{\varphi} = 0$$

- Motion in the orthogonal plane must be zero (sliding constraint), i.e.

$$\left[\cos(\alpha+\beta) \quad \sin(\alpha+\beta) \quad (d+l)\sin(\beta) \right] R(\theta)\dot{\xi}_I + \dot{\beta} = 0$$

- Rolling constraint similar to fixed wheel, but the sliding constraint is wrt the wheel-contact point

Robot Kinematic Constraints

- Combination of the wheel constraints imposes the overall constraints for the vehicle
- Differential between fixed and steerable wheels
- Assume N wheels divided into N_f fixed and N_s steerable wheels
- β_f is orientation of fixed wheels
- $\beta_s(t)$ is the steering angle of the controllable wheels
- Define motion of wheels as:

$$\varphi(t) = \begin{bmatrix} \varphi_f(t) \\ \varphi_s(t) \end{bmatrix}$$

Kinematic constraints

- The rolling constraints can be collected:

$$J_1(\beta_s)R(\theta)\dot{\xi}_I - J_2\dot{\varphi} = 0$$

- where

$$J_1(\beta_s) = \begin{bmatrix} J_{1f} \\ J_{1s}(\beta_s) \end{bmatrix}$$

- J_{1f} is an $N_f \times 3$ constant matrix and J_{1s} is an $N_s \times 3$ matrix of constraints.
- J_2 is a diagonal matrix with wheel radii r_i
- A similar set of constraints can be defined for sliding (C)

Differential drive constraints – example

- Fusing rolling and sliding constraints we obtain:

$$\begin{bmatrix} J_1(\beta_s) \\ C_1(\beta_s) \end{bmatrix} R(\theta)\dot{\xi}_I = \begin{bmatrix} J_2\varphi \\ 0 \end{bmatrix}$$

- Assume robot axis along $+X_R$, then $\alpha = -\frac{\pi}{2}$, and $\beta = \pi$ for right wheel and $\alpha = \frac{\pi}{2}$ and $\beta = 0$ for left wheel

Differential drive contraint

- Resulting in

$$\begin{bmatrix} \begin{bmatrix} 1 & 0 & l \\ 1 & 0 & -l \end{bmatrix} \\ \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \end{bmatrix} R(\theta)\dot{\xi}_I = \begin{bmatrix} J_2\varphi \\ 0 \end{bmatrix}$$

- Inverting the equation results in:

$$\dot{\xi}_I = R(\theta)^{-1} \begin{bmatrix} 1 & 0 & l \\ 1 & 0 & -l \\ 0 & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} J_2\varphi \\ 0 \end{bmatrix} = \frac{1}{2} R(\theta)^{-1} \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 2 \\ \frac{1}{l} & -\frac{1}{l} & 0 \end{bmatrix} \begin{bmatrix} J_2\varphi \\ 0 \end{bmatrix}$$

Outline

Degree of mobility

- The sliding constraints can be separated between fixed and steerable wheels:

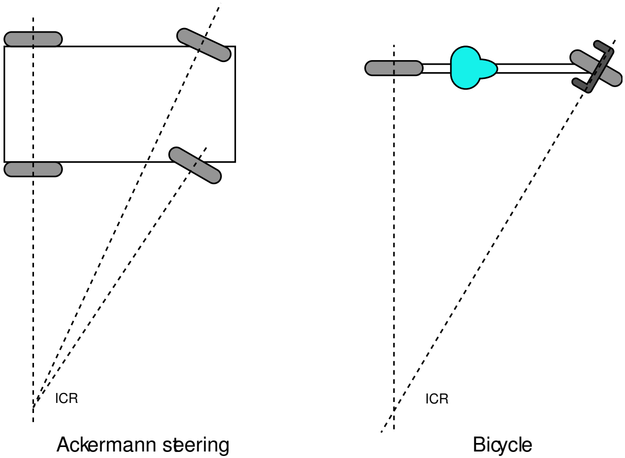
$$
\begin{aligned}
C_{1f} R(\theta)\dot{\xi}_I &= 0 \\
C_{1s}(\beta_s) R(\theta)\dot{\xi}_I &= 0
\end{aligned}
$$

- Motion of the robot must belong to the null space of C_1, i.e.

$$C_1(\beta_s)\vec{m} = 0, \quad \vec{m} \in null(C_1)$$

- Constraint is also shown by the instantaneous centre of rotation (ICR) mentioned earlier

Instantaneous centre of rotation



Mobility constraints

- The $rank(C_1)$ defines the number of independent constraints
- The degree of mobility is defined by the dimensionality of the null space of C_1 which for a mobile platform is equal to:

$$\delta_m = dim(null(C_1)) = 3 - rank(C_1)$$

- Examples:

Robot	δ_m
Differential drive	2
Bicycle	1

Degree of steerability

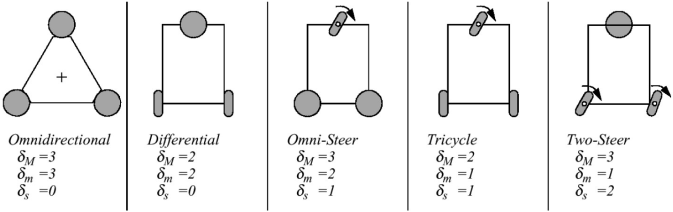
- Steerability is the number of independent DOF that can be controlled

$$\delta_s = rank(C_{1s})$$

- Similarly the degree of maneuverability is defined as

$$\delta_M = \delta_m + \delta_s$$

Maneuverability examples



Outline

The Workspace / Degree of Freedom

- Maneuverability:controllable degree of freedom
- The workspace: the space of possible configurations
- The velocity space: independent degree of motion that can be controlled. Sometimes referred to as the Differentiable degree of freedom (DDOF). DDOF $= \delta_m$.

$$DDOF \leq \delta_M \leq DOF$$

Holonomic Systems

- Holonomy is frequently used in robotics
- Holonomic kinematic constraints: defined by position / pose variables
- Non-holonomic kinematic constraints: defined by differential variables. The pose cannot be recovered by integration.
- Non-holonomic systems are also referred to as non-integrable systems.

Holonomic systems

- Example, fixed wheel sliding constraint:

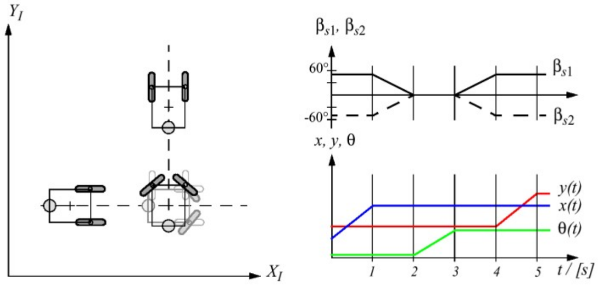
$$\begin{bmatrix} \cos(\alpha+\beta) & \sin(\alpha+\beta) & l\sin(\beta) \end{bmatrix} R(\theta)\dot{\xi}_I = 0$$

  the constraint is in terms of $\dot{\xi}$ rather the ξ as it constraint the motion not the final configuration.
- A systems is only holonomic iff DDOF = DOF.

Path/trajectory considerations

- The constraints only define what can be achieved and the limitations.
- For systems there is a a need to consider how to achieve different configuration.
- Trajectory planning: partly covered in final lecture
- Trajectory control/tracking: given a specification how can the robot be moved to achieve the specified trajectory

Example of trajectory control



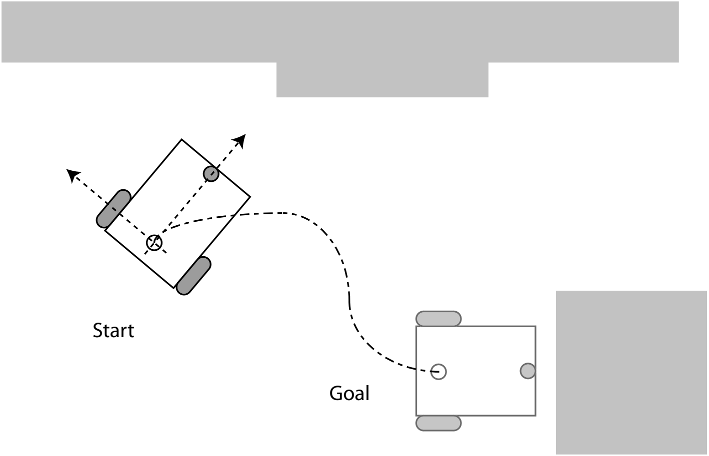
Outline

Beyond Basic Kinematics

- The above analysis has been performed under a strong set of assumptions: only rolling motion with no sliding
- Tanks, rigid vehicles, etc use skid steering in which sliding motion is utilized to ensure control of the vehicle.
- Requires consideration of dynamic models beyond pure kinematics
- The friction model for interaction between surface and wheel must be considered.
- High speed motion also requires explicit modelling of system dynamics for the control.
- Dynamics is a separate field of research in robotics. Beyond the scope of this course.

Outline

Kinematic Control



Trajectory following

- Kinematic control is following of a pre-specified trajectory described in terms of positions and velocities
- Often the trajectory is divided into trajectory segments
- Simple controllers use a combination of arcs and line segments (as done on American roads). Others use clothoids in which the curvature changes linearly with time, as done on European roads
- An entire field of robotics is devoted to path planning See `http://msl.cs.uiuc.edu/planning` for comprehensive / free book on the topic.

Feedback control

- A more appropriate strategy is a trajectory feedback controller that uses the path specification as “control” points to drive the robot system

Problem statement

- In the robot reference frame the error is

$$e = [x, y, \theta]_R^T$$

- The task is now to design a control matrix K

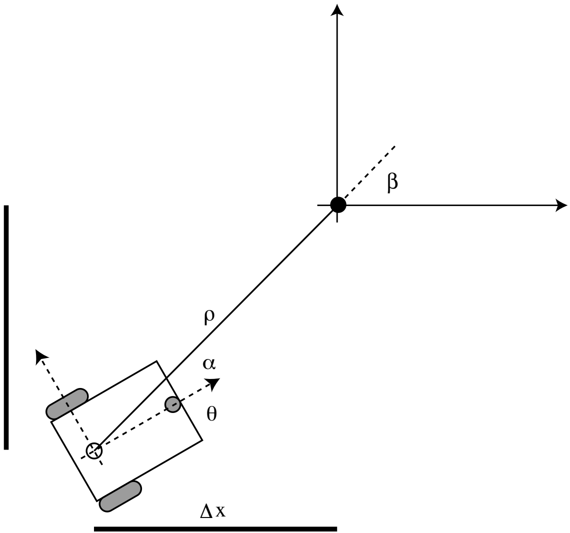
$$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \end{bmatrix} \quad k_{ij} = k(t, e)$$

- Such that

$$\begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix} = Ke = K \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}_R$$

drives the error to zero $\lim_{t \rightarrow \infty} e(t) = 0$

The basic setup for control



Kinematic model

- Consider a differential drive robot in the inertial frame

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

- In polar coordinates the error is now

$$\begin{aligned} \rho &= \sqrt{\Delta x^2 + \Delta y^2} \\ \alpha &= -\theta + atan2(\Delta y, \Delta x) \\ \beta &= -\theta - \alpha \end{aligned}$$

Kinematic control

- Rephrased in polar coordinates:

$$\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & 0 \\ \frac{\sin(\alpha)}{\rho} & -1 \\ -\frac{\sin(\alpha)}{\rho} & 0 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

The control law

- Control the linear control law:

$$
\begin{aligned}
v &= k_\rho \rho \\
\omega &= k_\alpha \alpha + k_\beta \beta
\end{aligned}
$$

- Which generates a closed loop system of:

$$
\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} -k_\rho \rho \cos(\alpha) \\ k_\rho \sin(\alpha) - k_\alpha \alpha - k_\beta \ \beta \\ -k_\rho \sin(\alpha) \end{bmatrix}
$$

Stability requirement

- It can be shown that the system is exponentially stable if:

$$\begin{aligned} k_\rho &> 0 \\ k_\beta &< 0 \\ k_\alpha - k_\rho &> 0 \end{aligned}$$

- Sketch of proof ($\cos x = 1$, $\sin x = x$):

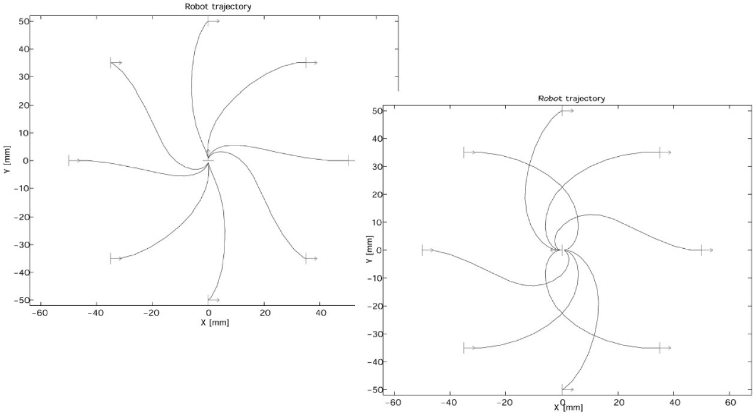
$$\begin{bmatrix} \dot\rho \\ \dot\alpha \\ \dot\beta \end{bmatrix} = \begin{bmatrix} -k_\rho & 0 & 0 \\ 0 & -(k_\alpha - k_\rho) & k_\beta \\ 0 & -k_\rho & 0 \end{bmatrix} \begin{bmatrix} \rho \\ \alpha \\ \beta \end{bmatrix}$$

Sketch of stability requirement

- if A has all eigenvalues where the real part is negative it is exponentially stable
- Characteristic polynomial:

$$(\lambda + k_\rho)(\lambda^2 + \lambda(k_\alpha - k_\rho) - k_\rho k_\beta)$$

Control examples



Outline

WRAP-UP

- Brief introduction to kinematic modelling of mobile systems
- Presentation of constraints and its use in models
- Example models for robots
- Brief example of kinematic control