Introduction to Mobile Robotics

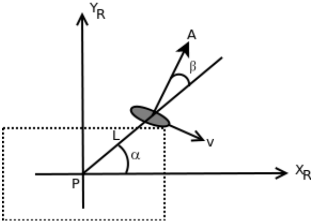
Jeff McGough

Department of Computer Science
South Dakota School of Mines and Technology
Rapid City, SD 57701, USA

October 1, 2012

Kinematic models

Approach the kinematic problem more formally to address different wheel types.



Let P be the chassis center, L is the distance from P to the wheel contact point. α is the angle of the wheel off of X_R and β is the angle of the wheel axis A from the line from P to the wheel contact point.

$$A = \langle \cos(\alpha+\beta), \sin(\alpha+\beta) \rangle$$

$$v = \langle \sin(\alpha+\beta), -\cos(\alpha+\beta) \rangle$$

Kinematic models

Assume that we have no slip (wheel spin) and no slide (horizontal motion) conditions. What does this mean?

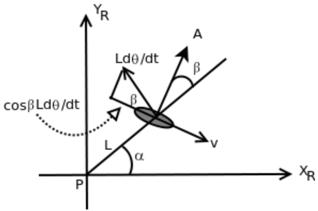
Begin with *no slip*, and project all the motion onto the wheel. Recall that the linear motion in the direction of the wheel is $r\dot{\phi}$. Thus the motion of P is a result of linear motion plus rotational motion

$$\langle \dot{x}_I, \dot{y}_I, 0 \rangle + \left\langle 0, 0, \dot{\theta} \right\rangle = \left\langle \dot{x}_I, \dot{y}_I, \dot{\theta} \right\rangle = \dot{\xi}_I$$

This motion needs to be rotated into the local coordinates: $\dot{\xi}_R = R(\theta)\dot{\xi}_I$. Then projected onto v:

$$\langle \sin(\alpha + \beta), -\cos(\alpha + \beta), 0 \rangle \cdot R(\theta) \langle \dot{x}_I, \dot{y}_I, 0 \rangle = P_v(\dot{\theta})$$

No Slip Condition



$$P_v[R(\theta)\dot{\xi}_I] = \langle \sin(\alpha+\beta), -\cos(\alpha+\beta), -L\cos(\beta)\rangle \cdot R(\theta)\left\langle \dot{x}_I, \dot{y}_I, \dot{\theta}\right\rangle$$

$$\Rightarrow P_v[R(\theta)\dot{\xi}_I] = r\dot{\phi}$$

No Slip and No Slide Conditions

For *No Slip* we have:

$$\Rightarrow \underbrace{\langle \sin(\alpha+\beta), -\cos(\alpha+\beta), -L\cos(\beta)\rangle}_{J_{1f}} R(\theta)\dot{\xi}_I = r\dot{\phi}$$

For *No Slide*, we want the projection in the direction of A to be zero:

$$P_A R(\theta)\dot{\xi}_I = 0$$

$$\Rightarrow \underbrace{\langle \cos(\alpha+\beta), \sin(\alpha+\beta), L\sin(\beta)\rangle}_{C_{1f}} \cdot R(\theta)\dot{\xi}_I = 0$$

Steered Wheel

The only difference for steered wheels is that the angle β varies over time. This does not have an effect on the form of the equations at an instantaneous time, but will integrated over time.

For *No Slip*:

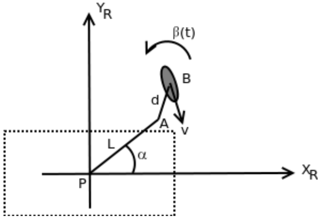
$$\Rightarrow \langle \sin(\alpha + \beta(t)), -\cos(\alpha + \beta(t)), -L\cos(\beta(t)) \rangle \, R(\theta)\dot{\xi}_I = r\dot{\phi}$$

For *No Slide*:

$$\Rightarrow \langle \cos(\alpha + \beta(t)), \sin(\alpha + \beta(t)), L\sin(\beta(t)) \rangle \cdot R(\theta)\dot{\xi}_I = 0$$

Castor Wheel

For the castor wheel, the no slip condition is the same (as the castor offset, d, plays no role in the motion in the direction of the wheel).
The offset, d, does change the equations in the no slide aspect.



For *No Slip*:

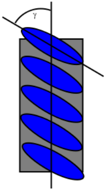
$$\langle \sin(\alpha+\beta(t)), -\cos(\alpha+\beta(t)), -L\cos(\beta(t))\rangle\, R(\theta)\dot{\xi}_I = r\dot{\phi}$$

For *No Slide*:

$$\langle \cos(\alpha+\beta(t)), \sin(\alpha+\beta(t)), d + L\sin(\beta(t))\rangle \cdot R(\theta)\dot{\xi}_I + d\dot{\beta} = 0$$

Swedish Wheel

Let γ be the angle between the roller axis and wheel plane (plane orthogonal to the wheel axis)



For *No Slip*:

$$\langle \sin(\alpha+\beta+\gamma), -\cos(\alpha+\beta+\gamma), -L\cos(\beta+\gamma)\rangle\, R(\theta)\dot{\xi}_I = r\dot{\phi}\cos(\gamma)$$

For *No Slide*:

$$\langle \cos(\alpha+\beta+\gamma), \sin(\alpha+\beta+\gamma), L\sin(\beta+\gamma)\rangle \cdot R(\theta)\dot{\xi}_I = r\dot{\phi}\sin(\gamma) + r_{sw}\dot{\phi}_{sw}$$

Note that since ϕ_{sw} is free (to spin), the no slide condition is not a constraint in the same manner as the fixed or steered wheels.

Multiple Wheel Model

- Let N denote the total number of wheels
- Let N_f denote the number of fixed wheels
- Let N_s denote the number of steerable wheels
- Let $\phi_f(t)$ and β_f be the fixed wheel angular velocity and wheel position.
- Let $\phi_s(t)$ and $\beta_s(t)$ be the steerable wheel angular velocity and wheel position.
- Bundle the values in a vector:

$$\phi(t) = (\phi_{f,1}(t), \phi_{f,2}(t), \phi_{f,3}(t), ..., \phi_{s,1}(t), \phi_{s,2}(t), ...)$$

$$\beta(t) = (\beta_{f,1}(t), \beta_{f,2}(t), \beta_{f,3}(t), ..., \beta_{s,1}(t), \beta_{s,2}(t), ...)$$

Matrix formulation

Collect the no slip constraints and place them in a matrix:

$$J_1 R(\theta)\dot{\xi}_I = \begin{bmatrix} J_{1f} \\ J_{1s} \end{bmatrix} R(\theta)\dot{\xi}_I = J_2 \dot{\phi}$$

where J_1 is the matrix with rows made up of the rolling constraints and J_2 is a diagonal matrix made from wheel diameters. In a similar manner we can bundle up the no slide constraints (fixed and steered):

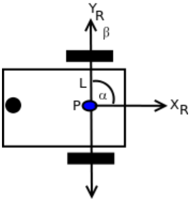
$$C_1 R(\theta)\dot{\xi}_I = \begin{bmatrix} C_{1f} \\ C_{1s} \end{bmatrix} R(\theta)\dot{\xi}_I = 0.$$

This is matrix shorthand to address the kinematic models for a variety of systems.

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

Example: differential drive

Rederive the equations for the differential drive robot.



From the figure we have:
Left wheel: $\alpha = \pi/2$, $\beta = 0$;
Right wheel: $\alpha = -\pi/2$, $\beta = \pi$ (to be consistent with previous model).

Example: differential drive

Left wheel rolling constraint

$$\langle \sin(\alpha+\beta), -\cos(\alpha+\beta), -L\cos(\beta)\rangle = \langle 1, 0, -L\rangle$$

Right wheel rolling constraint

$$\langle \sin(\alpha+\beta), -\cos(\alpha+\beta), -L\cos(\beta)\rangle = \langle 1, 0, L\rangle$$

Then

$$J_1 = \begin{bmatrix} 1 & 0 & -L \\ 1 & 0 & L \end{bmatrix}$$

Example: differential drive

Left wheel sliding constraint:

$$\langle \cos(\alpha + \beta), \sin(\alpha + \beta), L \sin(\beta) \rangle = \langle 0, 1, 0 \rangle$$

Right wheel sliding constraint:

$$\langle \cos(\alpha + \beta), \sin(\alpha + \beta), L \sin(\beta) \rangle = \langle 0, 1, 0 \rangle$$

Then

$$C_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Since the two rows are linearly dependent, we only need to keep one row.

Example: differential drive

The motion model is

$$\begin{bmatrix} 1 & 0 & -L \\ 1 & 0 & L \\ 0 & 1 & 0 \end{bmatrix} R(\theta)\dot{\xi}_I = \begin{bmatrix} r & 0 \\ 0 & r \\ 0 & 0 \end{bmatrix} \dot{\phi}$$

Expanding

$$\begin{bmatrix} 1 & 0 & -L \\ 1 & 0 & L \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \dot{\xi}_I = \begin{bmatrix} r & 0 \\ 0 & r \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi}_2 \\ \dot{\phi}_1 \end{bmatrix}$$

To be consistent with the previous example, we had the left wheel as (2) and the right wheel as (1) - hence the reverse ordering on the ϕ terms. This is the system to solve. Invert the left hand array first, then invert the rotation matrix.

Example: differential drive

Working out the details:

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \dot{\xi}_I = \begin{bmatrix} 1 & 0 & -L \\ 1 & 0 & L \\ 0 & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} r & 0 \\ 0 & r \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi}_2 \\ \dot{\phi}_1 \end{bmatrix}$$

$$\dot{\xi}_I = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 & -L \\ 1 & 0 & L \\ 0 & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} r & 0 \\ 0 & r \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi}_2 \\ \dot{\phi}_1 \end{bmatrix}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1/2 & 1/2 & 0 \\ 0 & 0 & 1 \\ -1/(2L) & 1/(2L) & 0 \end{bmatrix} \begin{bmatrix} r\dot{\phi}_2 \\ r\dot{\phi}_1 \\ 0 \end{bmatrix}$$

Example: differential drive

and finally ....

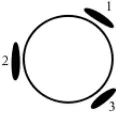
$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{r}{2}\dot{\phi}_1 + \frac{r}{2}\dot{\phi}_2 \\ 0 \\ -\frac{r}{2L}\dot{\phi}_2 + \frac{r}{2L}\dot{\phi}_1 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{r}{2}\left(\dot{\phi}_1 + \dot{\phi}_2\right)\cos\theta \\ \frac{r}{2}\left(\dot{\phi}_1 + \dot{\phi}_2\right)\sin\theta \\ \frac{r}{2L}\left(\dot{\phi}_1 - \dot{\phi}_2\right) \end{bmatrix}$$

(and you didn't think this was going to work out, eh?)
You may apply this machinery to other systems as well.

Example: omni-wheels

For this example we look at a three Swedish wheel robot.



We use an unsteered 90° Swedish wheel, so $\beta_i = 0$ and $\gamma_i = 0$ for all i. Going counterclockwise in the figure, we have $\alpha_1 = \pi/3$, $\alpha_2 = \pi$ and $\alpha_3 = -\pi/3$. You will note that the C_1 matrix is of zero rank and so the sliding constraint does not contribute to (nor is needed for) the model. The equations for motion then are

$$\dot{\xi}_I = R(\theta)^{-1} J_{1f}^{-1} J_2 \dot{\phi}$$

where

$$J_{1f} = \begin{bmatrix} \sqrt{3}/2 & -1/2 & -L \\ 0 & 1 & -L \\ -\sqrt{3}/2 & -1/2 & -L \end{bmatrix}, \quad J_2 = \begin{bmatrix} r & 0 & 0 \\ 0 & r & 0 \\ 0 & 0 & r \end{bmatrix}$$

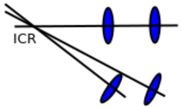
Maneuverability

Recall that

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

which means that $R(\theta)\dot{\xi}_I$ is in the nullspace[1] of the array $C_1 = \begin{bmatrix} C_{1f} \\ C_{1s} \end{bmatrix}$.

ICR - Instantaneous Center of Rotation.
Each sliding constraint generates a zero motion line (orthogonal to the wheel plane). The intersection of the zero motion lines is the ICR.



In other words, each wheel is traveling on a circle whose center must be on the zero motion line.

[1] Nullspace of the matrix A is the collection of vectors v such that $Av = 0$.

Maneuverability

The kinematics are a function of independent constraints. The rank [2] of C_1 is the number of independent constraints. The greater the rank, the more constrained the vehicle. Clearly

$$0 \leq \text{rank}(C_1) \leq 3.$$

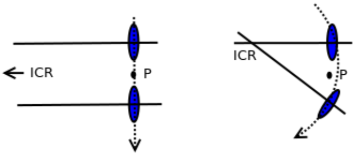
For the differential drive: $\alpha_1 = \pi/2$, $\beta_1 = 0$, $\alpha_2 = -\pi/2$, $\beta_2 = 0$

$$C_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix}, \quad \text{rank}(C_1) = 1.$$

[2] number of independent rows

Maneuverability

Example: fixed (not steerable) wheel bike.
We have $L_1 = L_2 = L$, $\beta_1 = \beta_2 = \pi/2$, $\alpha_1 = 0$, $\alpha_2 = \pi$



$$C_1 = \begin{bmatrix} 0 & 1 & L \\ 0 & -1 & L \end{bmatrix}, \quad \text{rank}(C_1) = 2.$$

In general, if the rank of C_1 is greater than one then the vehicle at best can only travel a line or a circle.
Rank $= 3$ means no motion at all.

Terms

Degree of mobility $= \delta_m$, also known as DDOF - differential degrees of freedom,

$$\delta_m \equiv \dim \mathcal{N}(C_1) = 3 - \text{rank}(C_1)$$

Differential drive: $\delta_m = 2$

Degree of steerability $= \delta_s$

$$\delta_s \equiv \text{rank}(C_{1,s})$$

Note that increasing this rank increases steerability, but since C_1 contains $C_{1,s}$, it will decrease mobility. DOF - degrees of freedom is based on the workspace which is three.

Example: auto

We have $N_f = 2$ and $N_s = 2$.

$$\text{rank}(C_{1f}) = 1$$

(since they share an axle).
Since all axle lines must intersect in a point for the vehicle to move,

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

So:

$$\text{rank}\begin{bmatrix} C_{1f} \\ C_{1s} \end{bmatrix} = 2$$

Thus $\delta_m = 1$ and $\delta_s = 1$.

Maneuverability

Degree of Maneuverability: δ_M

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

Equivalent to control degrees of freedom!

A *holonomic* robot is a robot with ZERO nonholonomic constraints. A holonomic kinematic constraint can be expressed as an explicit function of position variables alone.

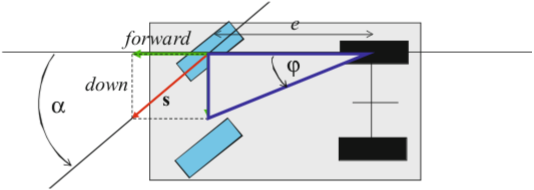
A robot is holonomic if and only if DDOF = DOF.

A robot is said to be ominidirectional if it is holonomic and DDOF = 3.

Maneuver and Orient

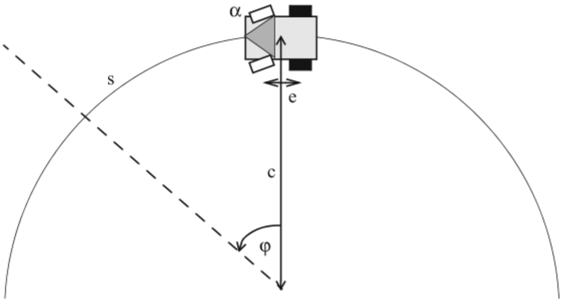
Ackermann Steering

forward = $s \cdot \cos\alpha$
down = $s \cdot \sin\alpha$



Ackermann Steering

α	steering angle,
e	distance between front and back wheels,
s_{front}	distance driven, measured at front wheels,
$\dot{\theta}$	driving wheel speed in revolutions per second,
s	total driven distance along arc,
φ	total vehicle rotation angle



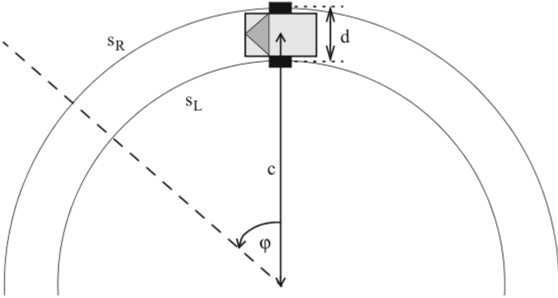
Ackermann Kinematics

Converting into the class notation ($e \rightarrow L_2$, $\omega \rightarrow \dot{\phi}$):

$$\begin{bmatrix} v \\ \dot{\theta} \end{bmatrix} = 2\pi r \dot{\phi} \begin{bmatrix} 1 \\ \frac{\sin \alpha}{L_2} \end{bmatrix}$$

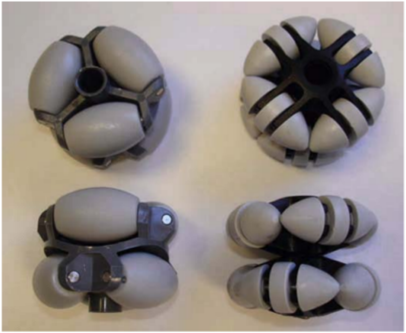
Problem with traditional design:

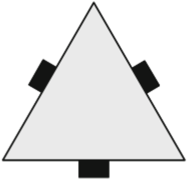
- Wheel paths of different lengths
- Rear wheels must skid if single axle
- Front wheels may skid

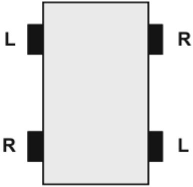


Omnidirectionality

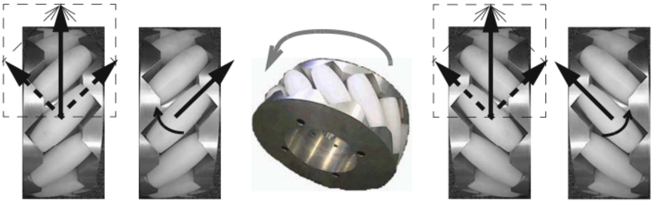
Recall the Swedish Wheel:





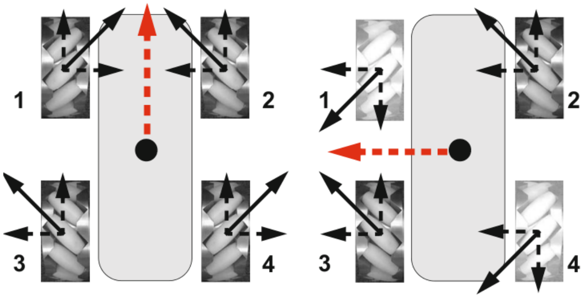


Swedish Wheel Driving

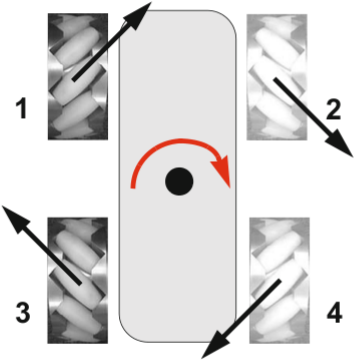


left-hand wheel
seen from below

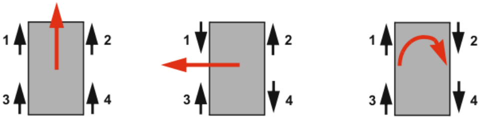
right-hand wheel
seen from below



Swedish Wheel Turning



Swedish Wheel Summary



- Driving forward: all four wheels forward
- Driving backward: all four wheels backward
- Driving left: 1,4 backwards; 2,3 forward
- Driving right: 1,4 forward; 2,3 backward
- Turning clockwise: 1,3 forward; 2,4 backward
- Turning counterclockwise: 1,3 backward; 2,4 forward

Swedish Wheel Kinematics

Let:

- r - wheel radius
- L_1 - distance between left and right wheel pairs
- L_2 - distance between front and rear wheel pairs
- $\dot{x}$, $\dot{y}$ - robot velocity in x and y
- $\dot{\theta}$ - robot angular velocity
- $\dot{\phi}_{FL}, \dot{\phi}_{FR}, \dot{\phi}_{BL}, \dot{\phi}_{BR}$ - front left, front right, back left, back right, wheel angular velocities.

Swedish Wheel Kinematics

Forward kinematics:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = 2\pi r \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ -\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{2(L_1+L_2)} & \frac{1}{2(L_1+L_2)} & -\frac{1}{2(L_1+L_2)} & \frac{1}{2(L_1+L_2)} \end{bmatrix} \begin{bmatrix} \dot{\phi}_{FL} \\ \dot{\phi}_{FR} \\ \dot{\phi}_{BL} \\ \dot{\phi}_{BR} \end{bmatrix}$$

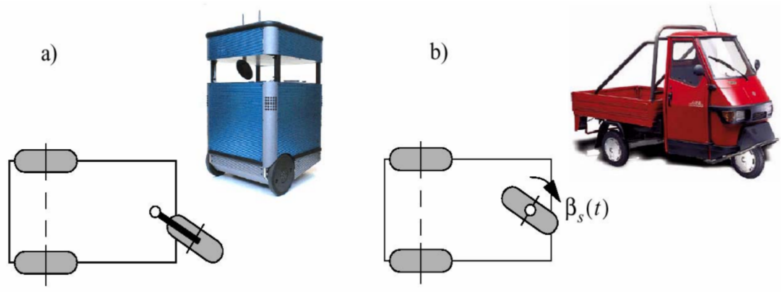
Inverse kinematics:

$$\begin{bmatrix} \dot{\phi}_{FL} \\ \dot{\phi}_{FR} \\ \dot{\phi}_{BL} \\ \dot{\phi}_{BR} \end{bmatrix} = \frac{1}{2\pi r} \begin{bmatrix} 1 & -1 & -(L_1+L_2)/2 \\ 1 & 1 & (L_1+L_2)/2 \\ 1 & 1 & -(L_1+L_2)/2 \\ 1 & -1 & (L_1+L_2)/2 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}$$

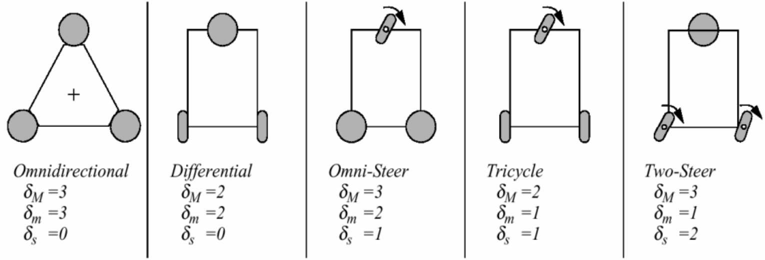
Parameter summary

- Differential Drive

Tricycle



Parameter summary



Legs

We move over to a more biological approach. What appears trivial in the natural world is not so easy for robotics. We will examine a few systems based on articulated arms.

Walking - issues

Stability

- Number of contact points
- Center of gravity
- Static/Dynamic stabilization
- Terrain

Contact

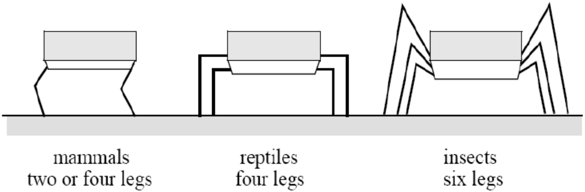
- Contact area
- Angle of contact
- Friction

Environment

- Structure
- Medium

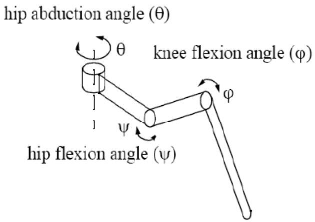
Legs

- Number of legs
  - Three legs for static stability
  - More legs - more coordination
  - Fewer legs - more complicated motion
- Legs must be lifted
  - Possible loss of stability
  - Shifting center of gravity
  - Additional energy cost
  - Complexity of positioning system
  - Articulator path planning
- Static stability and walking requires 6 legs.



Leg Joints

- Two DOF is required:
  lift and swing
- Three DOF is needed in most cases:
  lift, swing and position
- Fourth DOF is needed for stability:
  ankle joint - improves balance and walking



Motion Events

How many distinct gaits can be constructed?

- Motion forward (lifted or swinging): F
- Motion backwards (down or released): B

We need at least one transition $B \to F$ or $F \to B$.

Biped Motion - Todd

- With two legs (biped) one can have four different states
  - 1) Both legs down
  - 2) Right leg down, left leg up
  - 3) Right leg up, left leg down
  - 4) Both leg up
- A distinct event sequence can be considered as a change from one state to another and back.
- So we have the following $N=(2k-1)!=6$ distinct event sequences (change of states) for a biped:

● *Leg down*
○ *Leg up*

Motion Events

According to D.J. Todd, the number of events N with k legs

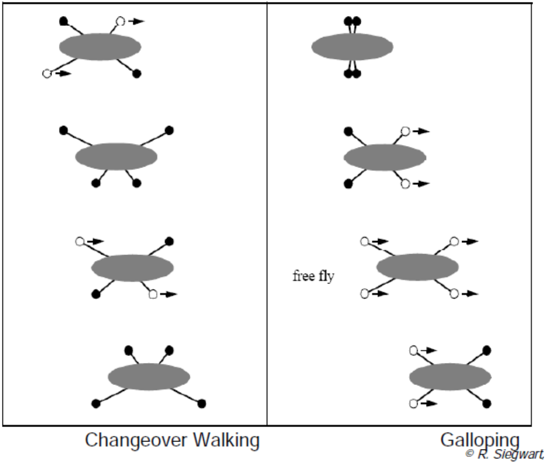
- $N = (2k-1)!$
- For $k = 2$ then $N = 6$
- For $k = 4$ then $N = 5,040$
- For $k = 6$ then $N = 39,916,800$
- For $k = 8$ then $N = 1,307,674,368,000$

Depends on how you classify and what you call distinct. A very simple argument gives you

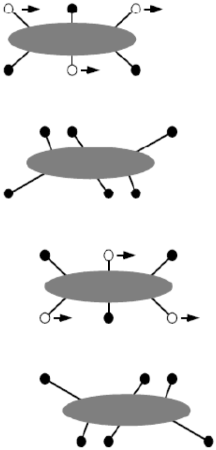
$$2^{3(N-1)}$$

- For $k = 2$ then $N = 8$
- For $k = 4$ then $N = 512$
- For $k = 6$ then $N = 32,768$
- For $k = 8$ then $N = 2,097,152$

Quadruped Motion

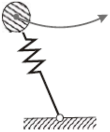


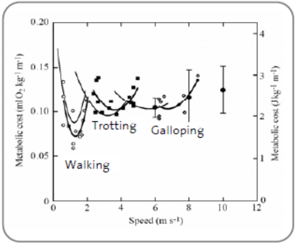
Hexapod Motion

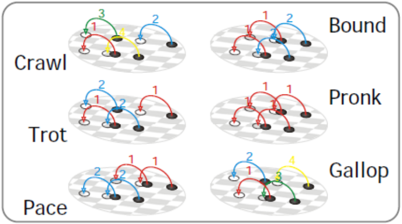


Optimal Gaits

- Nature optimizes its gaits
- Storage of "elastic" energy
- To allow locomotion at varying frequencies and speeds, different gaits have to utilize these elements differently



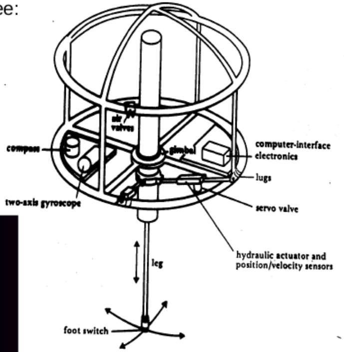




- The energetically most economic gait is a function of desired speed. (Figure [Minetti et al. 2002])

Examples

- No industrial applications up to date, but a popular research field
- For an excellent overview please see:
  http://www.uwe.ac.uk/clawar/

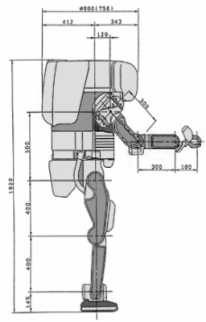


The Hopping Machine at MIT

Examples

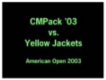
- P2 from Honda, Japan
  - Maximum Speed: 2 km/h
  - Autonomy: 15 min
  - Weight: 210 kg
  - Height: 1.82 m
  - Leg DOF: 2x6
  - Arm DOF: 2x7

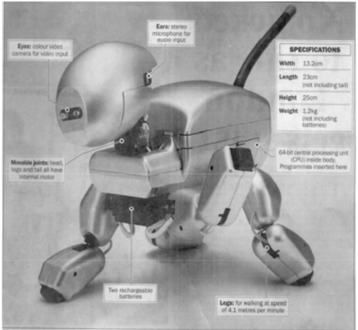
Honda P3



Examples

- Artificial Dog Aibo from Sony, Japan





Sony's new computer canine will chase a ball or sit when asked, but cleaning up will be a thing of the past

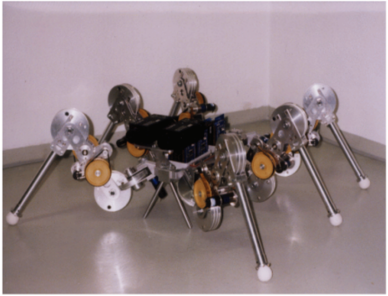
Examples

We can't forget about Boston Dynamics ...

Examples

- Most popular because static stable walking possible
- The human guided hexapod of Ohio State University
  - Maximum Speed: 2.3 m/s
  - Weight: 3.2 t
  - Height: 3 m
  - Length: 5.2 m
  - No. of legs: 6
  - DOF in total: 6*3

Examples



- Lauron II, University of Karlsruhe
  - Maximum Speed: 0.5 m/s
  - Weight: 6 kg
  - Height: 0.3 m
  - Length: 0.7 m
  - No. of legs: 6
  - DOF in total: 6*3
  - Power Consumption: 10 W

Bipeds

Consider a humanoid robot - what are the issues

- Need 5 DOF per leg
- Two legs
- Need 5 DOF per arm
- Two arms
- Camera pan and tilt - 2 DOF
- Gaits ($k = 2$ or 4?)

This adds to 22 DOF to control and control over numerous gaits.

Robot must also balance itself...

- Static balance
- Dynamic balance

Dynamic Balance

Balancing robots - inverted pendulum problem.

- Gyroscopes
- Accelerometers
- Compass
- IMUs
- Camera

Balancing robots - inverted pendulum problem.

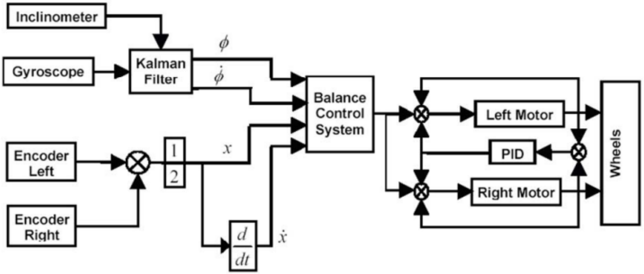
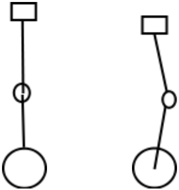


Figure 10.5: Kalman-based control system [Ooi 2003]

Dynamic Balance for walking

Double inverted pendulum problem.



Classic problems in controls. You will see more in the ECE controls courses.

Equations of motion for manipulators

Assume we have the forward kinematics map: $\xi = \phi(q)$. Motion is found via the time derivative.

$$\dot{\xi} = D_t \phi(q) = J_\phi(q)\dot{q}$$

where q are the joint angles and J is the Jacobian.

Note that the Jacobian need not be square or of full rank. Thus an inverse need not exist. Given $\phi^{-1} = \psi$, $q = \psi(\xi)$, one can in principle do the same thing

$$\dot{q} = D_t \psi(\xi) = J_\psi(\xi)\dot{\xi}$$

Chain Rule

Recall that if $w_k = f_k(x, y, z)$ and x, y, z are functions of t, then the chain rule states

$$\dot{w}_k = \frac{\partial f_k}{\partial x}\frac{dx}{dt} + \frac{\partial f_k}{\partial y}\frac{dy}{dt} + \frac{\partial f_k}{\partial z}\frac{dz}{dt}$$

where $k = 1, 2, \ldots n$.

For $n = 3$

$$\begin{bmatrix} \dot{w}_1 \\ \dot{w}_2 \\ \dot{w}_3 \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial z} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial z} \\ \frac{\partial f_3}{\partial x} & \frac{\partial f_3}{\partial y} & \frac{\partial f_3}{\partial z} \end{bmatrix} \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \\ \frac{dz}{dt} \end{bmatrix}$$

Equations of motion revisted

Assume we have the forward velocity model:

$$\dot{\xi} = J_\phi(q)\dot{q}$$

where q are state variables and J is the Jacobian of the forward map.

Determine a velocity (speed and direction) $\dot{\xi}$, and then solve $\dot{\xi} = J_\phi(q)\dot{q}$ for $\dot{q}$.

Thus we have the iterative process

- Define $\Delta\xi_k = \xi_k - \xi_{k-1}$
- Solve $\Delta\xi_k = J_\phi(q_{k-1})h$
- Set $q_k = q_{k-1} + h$

Least Squares

Let $A = J_\phi(q)$ and $x = h$ and $b = \Delta\xi$ then

$$J_\phi(q)h = \Delta\xi \quad \Rightarrow \quad Ax = b$$

What can be said when A is rank deficient?

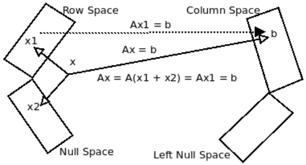
Linear Systems

- Column Space
  All vectors, y that can be reached by A: all $y = Ax$.
- Nullspace
  All vectors, v in the domain which are mapped to zero: $Av = 0$.
- Row Space
  All vectors in the domain space which are orthogonal to the Nullspace: $v \cdot x = 0$.
- Left Nullspace (nullspace of A^T)
  All vectors, w in the range space which are orthogonal to the Column Space: $w \cdot y = 0$

Least Squares

Assume that b is in the range of A, solve $Ax = b$.

Let $x = x_1 + x_2$ where x_2 is in the null space, $Ax_2 = 0$.

Domain  Range

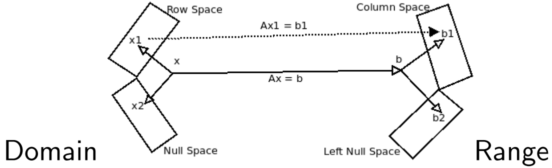
We can solve for $x_1 = x_p$ (and call $x_2 = x_H$) and the general solution is

$$x = x_p + c\, x_H$$

Least Squares

If b is not in the range of A.

Then we need to project b into the range of A. Again, let $x = x_1 + x_2$ where x_2 is in the null space, $Ax_2 = 0$ and $b = b_1 + b_2$ where b_2 is in the left nullspace $A^T b_2 = 0$.



We can solve the projected problem (call $x_1 = x_p$, $x_2 = x_H$) and the general solution is

$$x_{LS} = x_p + c\, x_H$$

Least Squares

Assume that $x_p \cdot x_H = 0$ and note that

$$b = Ax = A(x_p + x_H) = Ax_p$$

and

$$A^T b = A^T (b_1 + b_2) = A^T b_1 \in \text{Range}(A)$$

Least squares

$$Ax = b$$

Normal Equations (the residual error in left nullspace)

$$A^T A x = A^T b$$

Pseudoinverse

$$\hat{x} = (A^T A)^{-1} A^T b$$

Least Squares Motion Equations

Invert:

$$J_\phi h = \Delta\xi_k$$

Normal Equations

$$J_\phi^T J_\phi h = J_\phi^T \Delta\xi_k$$

Pseudoinverse

$$\hat{h} = (J_\phi^T J_\phi)^{-1} J_\phi^T \Delta\xi_k$$

Thus the iterative process is

- Define $\Delta\xi_k = \xi_k - \xi_{k-1}$
- $\hat{h} = (J_\phi^T J_\phi)^{-1} J_\phi^T \Delta\xi_k$
- Set $q_k = q_{k-1} + \hat{h}$

When $J_\phi^T J_\phi$ is not full rank, more information must be added for this term to be invertable.