### Kinematik mobiler Roboter

- Radbetriebene Roboter
- Kinematik:
   Roboter-Pose, Trajektorien und Momentanpol
- Zweiradfahrzeug mit Differentialantrieb
- Ackermann-Antrieb
- Mecanum-Antrieb
- Kinematische Grundfertigkeiten

#### Radbetriebene Roboter

- Weit verbreitet
- Robust und vergleichsweise einfach zu steuern (im Vergleich zu Lauf-Robotern)
- Statische Stabilität einfach zu erreichen (durch wenigstens 3 Räder)
- Wichtige Antriebssysteme:
  - Differential-Antrieb (2 Antriebsräder mit Stützrad; Indoor-Anwendungen)
  - Ackermann-Antrieb (Automobile, autonomes Fahren)
  - Mecanum-Antrieb (omni-direktionaler Antrieb, Transportaufgaben im industriellem Umfeld)



Differential-Antrieb; Pioneer 3DX



Ackermann-Antrieb; Modellauto



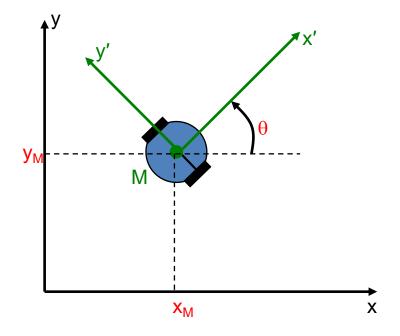
Mecanum-Antrieb; Kuka YouBot

### Koordinatensysteme und Roboterpose

- Mit dem Roboter ist ein lokales Koordinatensytem verbunden, wobei der Ursprung üblicherweise in der Mitte M der Antriebsachse liegt und die x-Achse in Richtung des Roboterfrontteils zeigt.
- Die Pose p des Roboters wird festgelegt durch die Koordinaten von M im globalen Koordinatensystem und durch den Winkel θ zwischen der lokalen x-Achse und der globalen x-Achse.

$$p = (x_M, y_M, \theta)^T$$

 Die Position des Roboters ist dann die Pose ohne Orientierung θ.



### Koordinatentransformation

Punkt P im lokalen Koordinatensystem L

$$p^L = (x_L, y_L)^T$$

Punkt P im globalen Koordinatensystem O

$$p^G = (x_G, y_G)^T$$

 Transformation von p<sup>L</sup> nach p<sup>G</sup> mit m = (x<sub>M</sub>, y<sub>M</sub>)<sup>T</sup>:

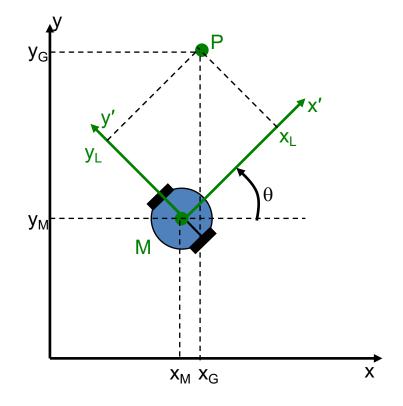
$$p^G = \mathbf{R}(\theta)p^L + m$$

 Dabei ist **R**(θ) die sogenannte Rotationsmatrix:

$$\mathbf{R}(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$$

Transformation von p<sup>G</sup> nach p<sup>L</sup>:

$$p^L = \mathbf{R}(\theta)^{-1}(p^G-m) = \mathbf{R}(-\theta)(p^G-m)$$



#### Polar- und kartesische Koordinaten

Polarkoordinaten von Punkt P:

Kartesische Koordinaten von Punkt P:

Umrechnung von Polar- in kartesische Koordinaten:

$$x_P = d^* cos(\alpha)$$

$$y_P = d*sin(\alpha)$$



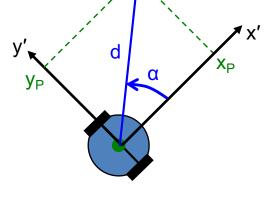
Umrechnung von kartesische in Polarkoordinaten:

$$\alpha = atan2(y_P, x_P)$$

$$d = \sqrt{x_P^2 + y_P^2}$$

atan2(y,x) berechnet (im Gegensatz zu atan(y/x)) Winkel für den Quadranten, in dem P liegt. Üblich:

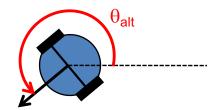
atan2(y,x) 
$$\in [-\pi, +\pi]$$

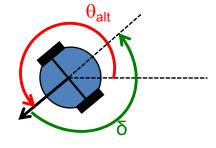


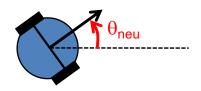
### Einschub: Orientierung

- Die Orientierung eines Roboters wird durch einen Winkel θ aus dem Intervall [0,2π) definiert.
- Ändert sich die Orientierung um einen Winkel δ, so muss immer modolo 2π gerechnet werden.
- Beispiel:

$$θalt = 1.25π$$
 $δ = π$ 
 $θneu = θalt + δ mod 2π$ 
 $= 0.25π$ 







### Einschub: Winkeldifferenz

- Die Differenz diff( $\theta_1, \theta_2$ ) zwischen zwei Winkel  $\theta_1$  und  $\theta_2$  wird so festgelegt, dass diff( $\theta_1, \theta_2$ ) im Intervall [-π,+π) liegt.
- Formel (für Bogenmaß):

$$diff(\theta_1, \theta_2) = (\theta_1 - \theta_2 + \pi) \mod 2\pi - \pi$$

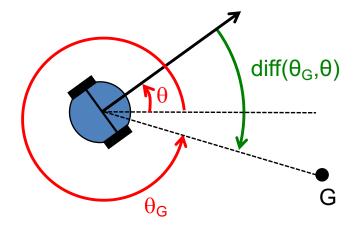
Beispiel (in Grad gerechnet):

Orientierung des Zielpunkts G:

$$\theta_G = 350^{\circ}$$

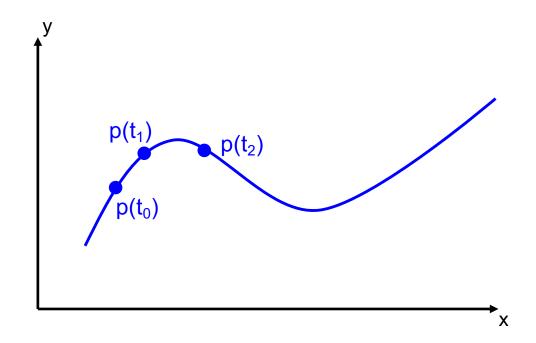
Roboterorientierung:  $\theta = 20^{\circ}$ 

Winkeldifferenz diff( $\theta_G$ , $\theta$ ) = -30°



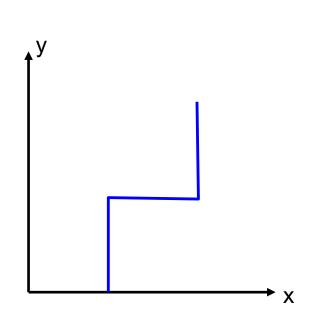
### Trajektorie und Pfad

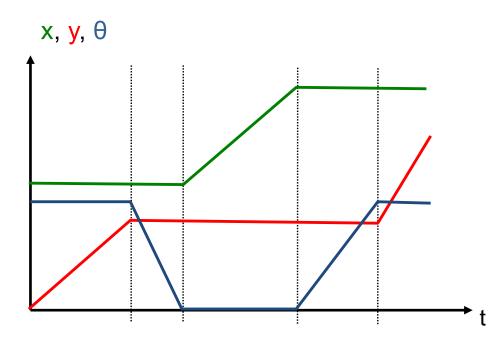
- Eine Trajektorie ist eine Kurve in der Ebene (Raum) parameterisiert über die Zeit.
- Die einzelnen Punkte der Kurve stellen Positionen zu bestimmten Zeitpunkten dar.
- Eine Trajektorie ohne Zeitinformationen wird auch Pfad genannt.



### Trajektorie und Posen

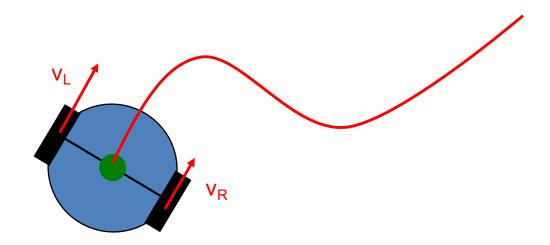
- Manchmal ist auch der zeitliche Verlauf von Posen (Position und Orientierung) gewünscht.
- Bei einer glatten (Positions)Trajektorie kann implizit die Orientierung als Tangente an den jeweiligen Punkten gewählt werden (siehe vorhergehende Folie).
- Bei einer nicht-glatten Trajektorie kann der Verlauf der Orientierung separat dargestellt werden (siehe unten).





#### Kinematik

- Kinematik = Lehre von den Bewegungen (keine Berücksichtigung von Kräften und Drehmomenten)
- In Vergleich dazu berücksichtigt die Kinetik Kräfte und Drehmomente.
- Grundlegende Fragestellung in der Roboterkinematik:
   Zusammenhang zwischen Einstellung der beweglichen Teile des Roboters (Räder, Drehgelenke) und Pose des Roboters.



#### Kinematisches Gesetz: Kreisbewegung um Momentanpol

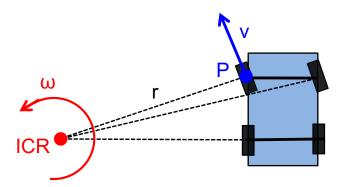
#### Momentanpol

Die Bewegung eines starren Körpers in der Ebene lässt sich in jedem Zeitpunkt als reine Drehbewegung um einen momentanen Drehpunkt auffassen (ICR = instantaneous center of rotation, Momentanpol)

Rotiert der Körper mit der Winkelgeschwindigkeit ω um den ICR auf einem Kreis mit Radius r, dann gilt für die Geschwindigkeit v in einem Punkt P:

$$\omega = \frac{v}{r}$$

- Der Geschwindigkeitsvektor v steht dabei senkrecht auf dem Radius r.
- Der Radius r kann unendlich gross werden.
   Dann wird die Winkelgeschwindigkeit ω = 0 (Geradeausfahrt).



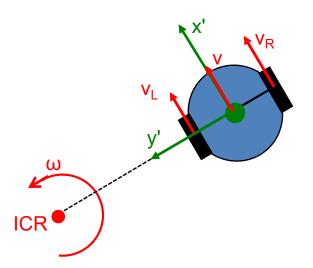
### Kinematik mobiler Roboter

- Radbetriebene Roboter
- Kinematik:
   Roboter-Pose, Trajektorien und Momentanpol
- Zweiradfahrzeug mit Differentialantrieb
- Ackermann-Antrieb
- Mecanum-Antrieb
- Kinematische Grundfertigkeiten

### Differentialantrieb (1)

- Kinematisches Modell: Roboter wird von zwei unabhängigen Rädern angetrieben. Zusätzlich ist ein Stützrad angebracht.
- Geschwindigkeit des linken Rads  $v_L$  und des rechten Rads  $v_R$  werden eingestellt. Steuerbefehl  $u(t) = (v_L, v_R)$
- Nach dem kinematischen Grundgesetz bewegt sich der Roboter um ICR mit Winkelgeschwindigkeit ω und Geschwindigkeit v in lokaler x-Richtung.





# Differentialantrieb (2)

Es gelten folgende kinematischen Zusammenhänge:

$$v_{L} = \omega r$$

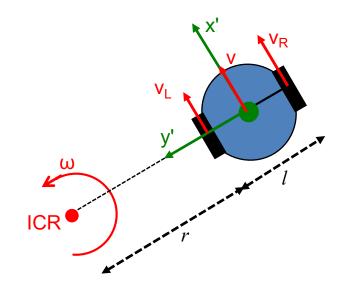
$$v = \omega (r + l/2)$$

$$v_{R} = \omega (r + l)$$

Daraus ergibt sich:

$$v = \frac{v_R + v_L}{2}$$

$$\omega = \frac{v_R - v_L}{l}$$



- Also lassen sich v und ω unmittelbar aus v<sub>L</sub>, v<sub>R</sub> und der Achslänge *l* ermitteln.
- Ebenso einfach lässt sich umgekehrt  $v_L$  und  $v_R$  aus v und  $\omega$  bestimmen.

#### Vorwärts- und Rückwärtskinematik



#### Anmerkungen:

- Die hier dargestellten Kinematiken sind einfache lineare Zusammenhänge (s. Seite 3-14)
- Wesentlicher komplizierter ist der Zusammenhang zwischen v(t) und ω(t) und der Pose oder Trajektorie des Roboters.
- Die Berechnung der Pose oder Trajektorie aufgrund von v(t) und ω(t) wird im Abschnitt Lokalisierung – Koppelnavigation behandelt.
- Für eine gewünschte Pose oder Trajektorie eine Folge von Steuerbefehlen v(t) und ω(t) zu berechnen, wird im Abschnitt Pfadplanung behandelt.

### Ackermann-Antrieb (1)

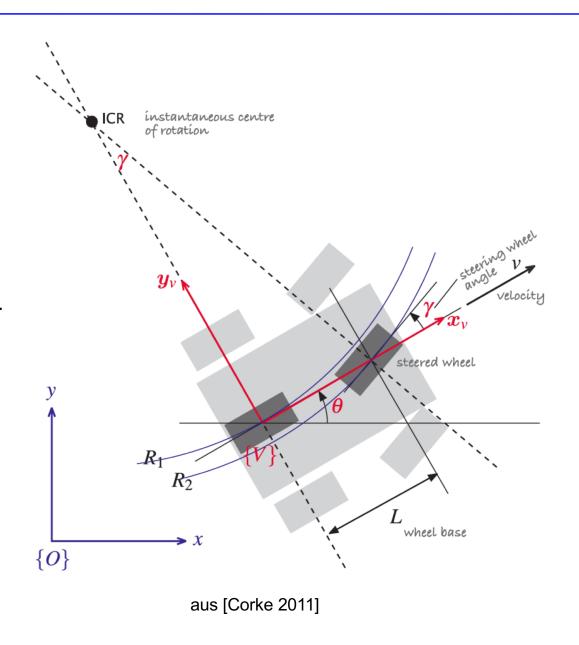




- v(t) = Geschwindigkeit der Antriebsräder
- $\gamma(t) = Lenkwinkel$

### Ackermann-Antrieb (2)

- Mit dem Automobil ist ein lokales Koordinatensystem {V} (Vehicle) fixiert.
- Die Steuerung des Fahrzeugs geschieht durch den Lenkwinkel γ und die Geschwindigkeit v der Hinterachse in Richtung der lokalen x-Achse.
- Das Fahrzeug dreht sich um den ICR.
- Die 4 R\u00e4der bewegen sich auf unterschiedlichen Radien.
   Dar\u00fcber hinaus muss bei der dargestellten Linkskurve der Lenkwinkel vom linkem Rad gr\u00f6\u00dfer als vom rechten Rad sein. Das wird durch die Ackermannsteuerung gew\u00e4hrleistet.
- Zweckmäßigerweise wird das Automobil durch ein Fahrradmodell approximiert.



### Ackermann-Antrieb (3)

 Es gelten folgende kinematische Beziehungen:

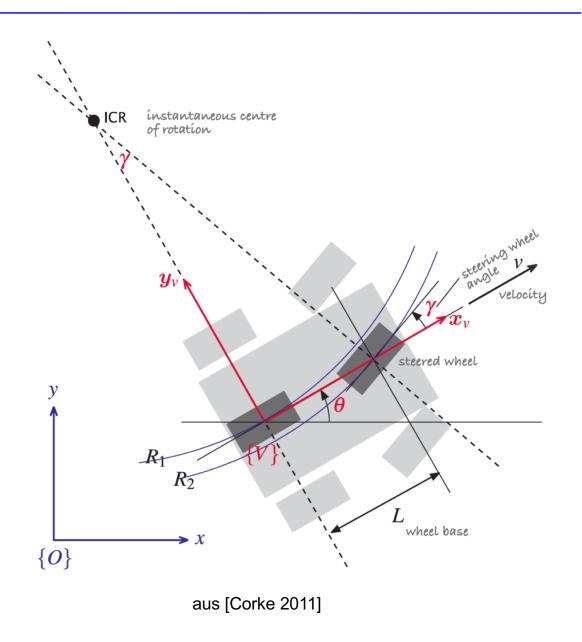
$$\tan \gamma = \frac{L}{R_1}$$

$$v = \omega R_1$$

Daraus ergibt sich:

$$\omega = \frac{v}{L} \tan \gamma$$

- Also lässt sich aus dem Lenkwinkel γ und der Geschwindigkeit v der Hinterachse die Winkelgeschwindigkeit ω direkt ermitteln (Vorwärtskinematik).
- Rückwärtskinematik durch Auflösung nach γ.



#### Mecanum-Antrieb

- Wurde 1973 von Bengt Ilon bei der schwedischen Firma Mecanum erfunden.
- Antrieb gestattet Drehbewegung und Translations-Bewegung in allen Richtungen: omnidirektional
- Dazu werden (wenigstens) 4 Räder (fast) unabhängig voneinander bewegt.



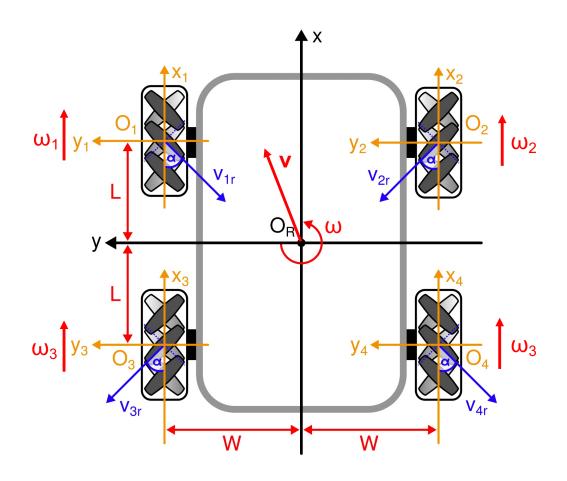
Kuka YouBot



Kuka FTS in der Airbus-Produktion

#### Mecanum-Antrieb – schematischer Aufbau

- x-y-KS im Zentrum des Roboters.
- Roboter bewegt sich mit Geschwindigkeit v = (v<sub>x</sub>,v<sub>y</sub>)<sup>T</sup> (in beliebiger Richtung!) und Winkelgeschwindigkeit ω.
- Die 4 Räder mit Radius R sind mit dem Winkel α = 45° ausgerichtet.
- Räder haben eine X-Anordnung (von oben gesehen).
- Die Räder können mit unabhängigen Winkelgeschwindigkeiten gedreht werden: ω<sub>1</sub>, ω<sub>2</sub>, ω<sub>3</sub>, ω<sub>4</sub>



Roboter von oben gesehen. [aus Woltjen, Bachelorarbeit HTWG, 2017]

### Rückwärtskinematik für Mecanum-Antrieb

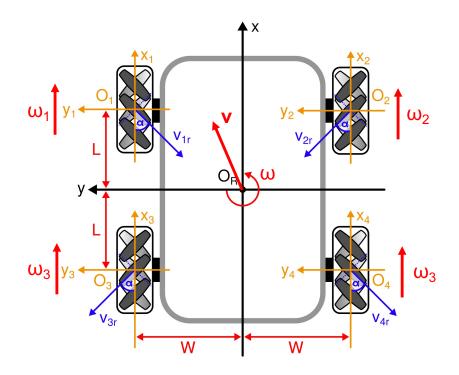
 Durch kinematische Betrachtungen an den Radmittelpunkten O<sub>i</sub> ergibt sich ein einfacher linearer Zusammenhang:

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = \frac{1}{R} \begin{pmatrix} 1 & -1 & -G \\ 1 & 1 & G \\ 1 & 1 & -G \\ 1 & -1 & G \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix}$$

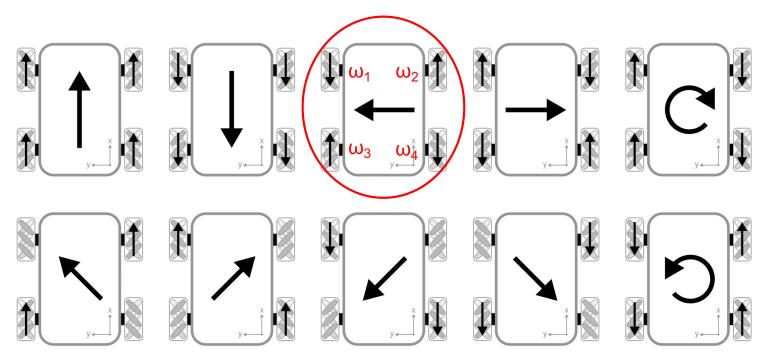
R = Radradius

G = W+L

Geschwindigkeit 
$$\mathbf{v} = \begin{pmatrix} v_x \\ v_y \end{pmatrix}$$



### Rückwärts-Kinematik an Beispielen



Ansicht von oben; [aus Woltjen, Bachelorarbeit HTWG, 2017]

• Beispielsweise ergibt sich für die umkreiste Konstellation mit  $v_x = 0$ ,  $v_y = 1$  und  $\omega = 0$  und der Formel für Rückwärtskinematik:

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = \frac{1}{R} \begin{pmatrix} 1 & -1 & -G \\ 1 & 1 & G \\ 1 & -1 & G \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \frac{1}{R} \begin{pmatrix} -1 \\ 1 \\ 1 \\ -1 \end{pmatrix}$$

#### Vorwärtskinematik für Mecanum-Antrieb

- Auflösung der Gleichung von Seite 3-21 nach Geschwindigkeit v<sub>x</sub>, v<sub>y</sub> und Winkelgeschwindigkeit ω führt zu einem überbestimmten Gleichungssystem.
- Durch eine Least-Square-Approximation erhält man:

$$\begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix} = \frac{R}{4G} \begin{pmatrix} G & G & G & G \\ -G & G & G & -G \\ -1 & 1 & -1 & 1 \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix}$$

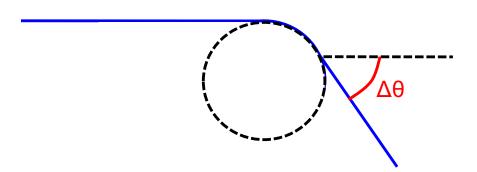
### Kinematik mobiler Roboter

- Radbetriebene Roboter
- Kinematik:
   Roboter-Pose, Trajektorien und Momentanpol
- Zweiradfahrzeug mit Differentialantrieb
- Ackermann-Antrieb
- Mecanum-Antrieb
- Kinematische Grundfertigkeiten

### Kinematische Grundfertigkeiten

- Richtungsänderung
- Fahrspurwechsel
- Auf Punkt zufahren
- Linie verfolgen
- PID-Regler
- Bahn verfolgen.

# Richtungsänderung



- Gewünschte Richtungsänderung  $\Delta\theta$ .
- Setze  $\omega(t) = \omega_0$  über eine Zeitperiode von

$$\Delta t = \frac{\Delta \theta}{\omega_0}$$

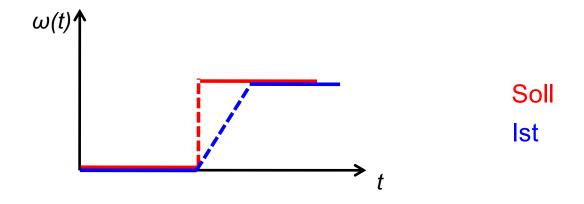
• Gefahrener Kurvenradius r bei einer Geschwindigkeit  $v(t) = v_0$  ist dabei

$$r = \frac{v_0}{\omega_0}$$

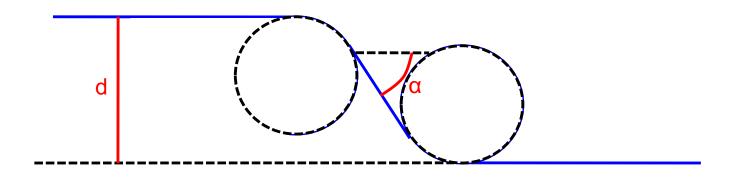
 Beachte: bei einer Rechtskurve (negative Winkeländerung) ist die Winkelgeschwindigkeit negativ. Entsprechend ist bei einer Linkskurve die Winkelgeschwindigkeit positiv.

### Bemerkung

- Bei einem realen Roboter stellt sich die gewünschte Winkelgeschwindigkeit nicht sofort ein, sondern erst mit einer gewissen Verzögerung, die durch die maximal mögliche Winkelbeschleunigung bestimmt ist.
- Analoges gilt für die Geschwindigkeit.



### Spurwechsel

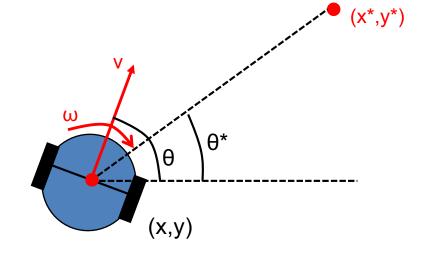


- Führe zwei entgegengesetzte Richtungsänderungen mit gleichem Betrag durch.
- Die Schräge des Spurwechsels α und die Spurbreite d lassen sich aus den gewählten Geschwindigkeiten und Zeitdauer berechnen

### Auf Punkt zufahren (1)

- Bewege Roboter auf Zielpunkt (x\*,y\*)
- Wähle Geschwindigkeit:

$$v = \min(v_{max}, K_v \sqrt{(x - x^*)^2 + (y - y^*)^2})$$



Zielrichtung:

$$\theta^* = \operatorname{atan} 2(y^* - y, x^* - x)$$

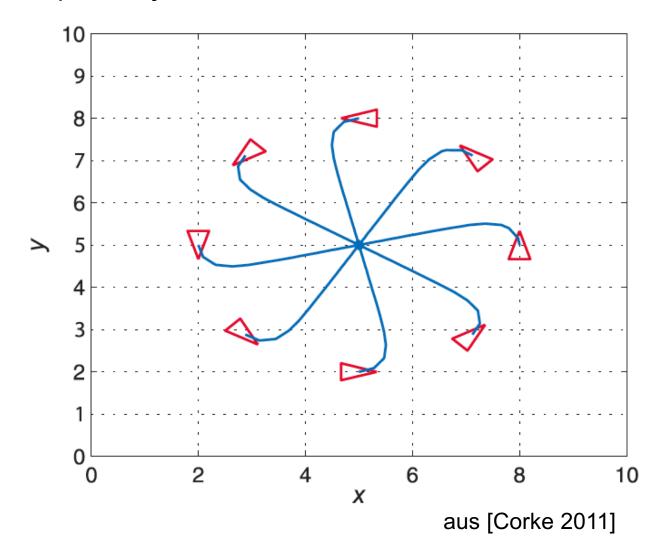
Winkelgeschwindigkeit:

$$\omega = \min(\omega_{max}, K_{\omega} \operatorname{diff}(\theta^*, \theta))$$

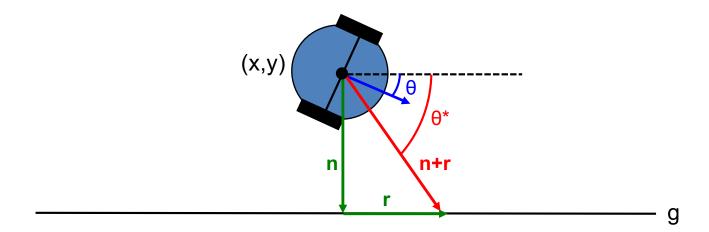
dabei ist  $diff(\theta^*,\theta)$  die Winkeldifferenz aus dem Intervall [- $\pi$  + $\pi$ ).

# Auf Punkt zufahren (2)

Beispiel-Trajektorien:



### Linienverfolger



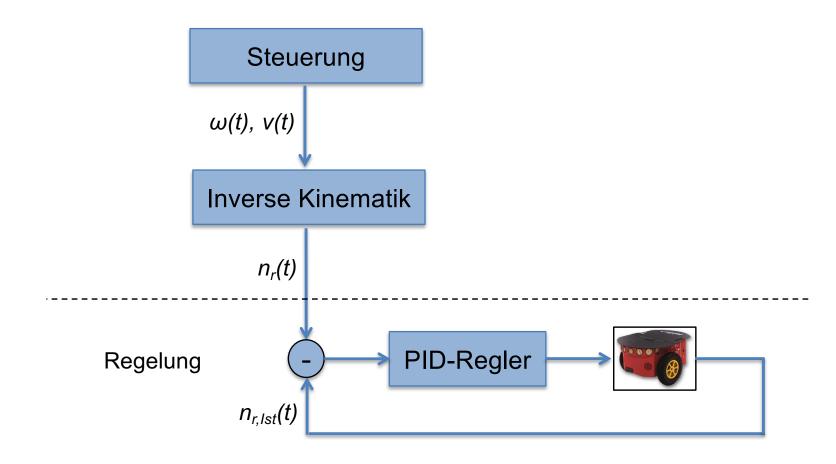
- Verfolge Linie (Gerade) g in Richtung r (|r| = 1).
- Bestimme Abstandsvektor n (orthogonal zu g mit |n| = Abstand zu (x,y))
- Berechne Wunschrichtung θ\* aus n + r
- Winkelgeschwindigkeit:

$$\omega = \min(\omega_{max}, K_{\omega} \operatorname{diff}(\theta^*, \theta))$$

 Hinweis: Aus Hessesche Normalform einer Gerade g lassen sich sowohl Abstand eines Punktes zu g als auch eine Geradennormale berechnen. (https://de.wikipedia.org/wiki/Hessesche\_Normalform)

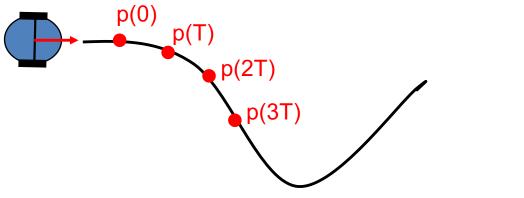
### Einschub: PID-Regler

- Aus  $\omega(t)$  und v(t) wird mittels inverser Kinematik eine Drehzahl  $n_r(t)$  für jedes Rad r bestimmt.
- Die Drehzahl  $n_r(t)$  wird mit einem PID-Regler für jedes Rad einzeln geregelt.



### Bahn verfolgen

- Ist die vorgegebene Bahn als glatte, kinematisch befahrbare Trajektorie vorgebenen, dann kann mit einem Linienverfolger die Trajektorie abgefahren werden.
- Eine Alternative ist das Carrot-Donkey-Verfahren
- Dabei bewegt sich ein Zielpunkt p(t) = (x\*(t),y\*(t)) über die gewünschte Trajektorie.
- Ein PID-Regler für die Geschwindigkeit v sorgt dafür, dass der Abstand zu p(t) = (x\*(t),y\*(t)) einen konstanten Wert d\* behält.
- Ein zweiter PID-Regler sorgt dafür, dass der Roboter in Richtung Zielpunkt  $p(t) = (x^*(t), y^*(t))$  ausgerichtet wird (wie bei Regler, der auf einen Punkt zufährt.)





### Polylinie verfolgen

#### Einfacher Ansatz:

fahre ersten Eckpunkte an; sobald Eckpunkt mit einer gewissen Toleranz erreicht ist, drehe Roboter in die Richtung des nächsten Eckpunkts und fahre entsprechend fort. Versuche Geschwindigkeit möglichst konstant zu halten.

- Die Polylinie kann zu einer kinematisch befahrbaren Kurve geglättet werden (z.B. mit Bezierkurven)
- Die Polylinie kann auch mit einem Carrot-Donkey-Verfahren abgefahren werden.

