Univerza v Ljubljani
Fakulteta za elektrotehniko

LABORATORIJ ZA MODELIRANJE, SIMULACIJO IN VODENJE

LABORATORIJ ZA AVTONOMNE MOBILNE SISTEME

Avtonomni mobilni sistemi

Izr. prof. dr. Gregor Klančar

gregor.klancar@fe.uni-lj.si

Vodenje kolesnih mobilnih sistemov

2013/2014

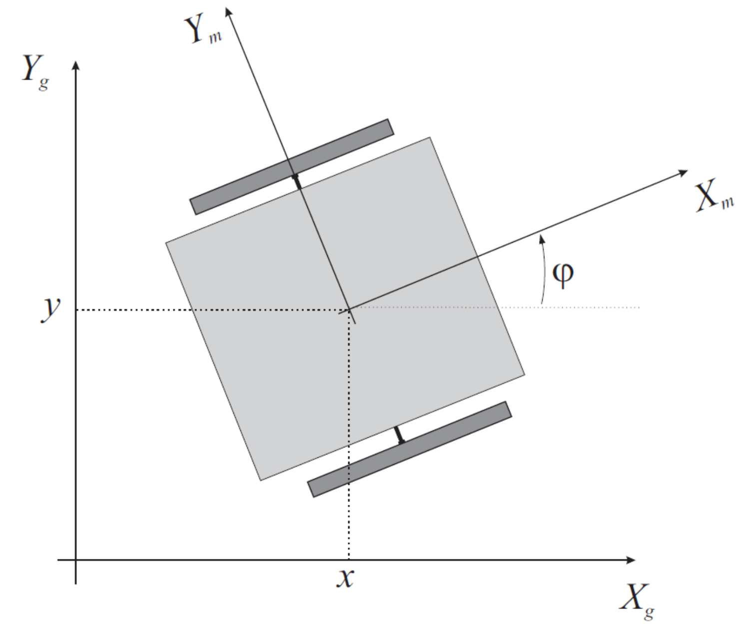
Uvod

- Mobilni sistem se giblje v okolici pri opravljanju določene naloge, kar zahteva vodenje, npr.
  - vožnja do cilja,
  - sledenje poti,
  - izogibanje ovir,…
- Vodenje
  - Klasično sledilno vodenje: v referenčno lego, vmesni potek stanj ni predpisan
  - Sledenje referenčni poti: do ciljne lege je predpisana referenčna pot (potek stanj)
  - Sledenje referenčni trajektoriji: do cilja je predpisana referenčna trajektorija odvisna od časa (potek stanj glede na čas x(t), y(t))
- Neholonomni sistemi (večina kolesnih MS) – bolj prikladno vodenje po ref. trajektoriji (neholonomične omejitve)
- Pogosto je vodenje s predkrmiljenjem (vodenje+krmiljenje)

Vodenje vozila z dif. pogonom

- Kinematika

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

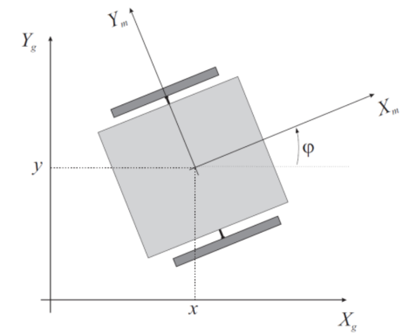


- Na vožnjo vplivamo z *v* in *w* (izhoda regulatorja)
- Osnovni pristop vodenja ločeno obravnava
  - Vodenje po orientaciji (izračun *w*)
  - Vodenje po poziciji (izračun *v*)
- Predstavljeni pristopi uporabni tudi za druge kinematike

Vodenje po orientaciji

- predpisana želena smer $\varphi_{ref}(t)$ (proti cilju, tangenta na trajektorijo,…)
- določitev $\omega(t)$, da bo pogrešek

$$e_{\varphi}(t) = \varphi_{ref}(t) - \varphi(t) \text{ ->0}$$



- Proporcionalni regulator

$$\omega(t) = K(\varphi_{ref}(t) - \varphi(t))$$

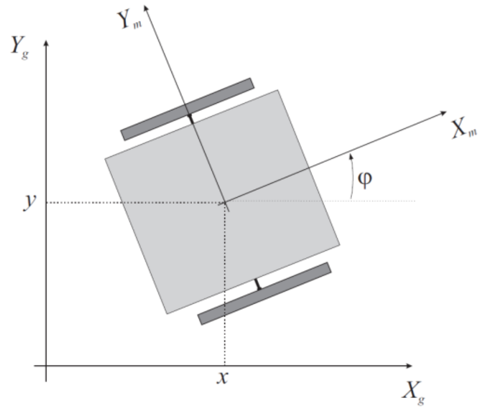
- Regulator PI

$$\dot{\omega}(t) = K_1(\varphi_{ref}(t) - \varphi(t)) + K_2\dot{\varphi}(t)$$

Vodenje po poziciji

- poleg orientacije (*w*), rabimo tudi regulator za določitev *v*
- za ref. pozicijo $(x_{ref}(t), y_{ref}(t))$ definiramo pogrešek

$$e_d = \sqrt{(x_{ref}(t) - x(t))^2 + (y_{ref}(t) - y(t))^2}$$



- Proporcionalni regulator

$$v(t) = K\sqrt{(x_{ref}(t) - x(t))^2 + (y_{ref}(t) - y(t))^2}$$

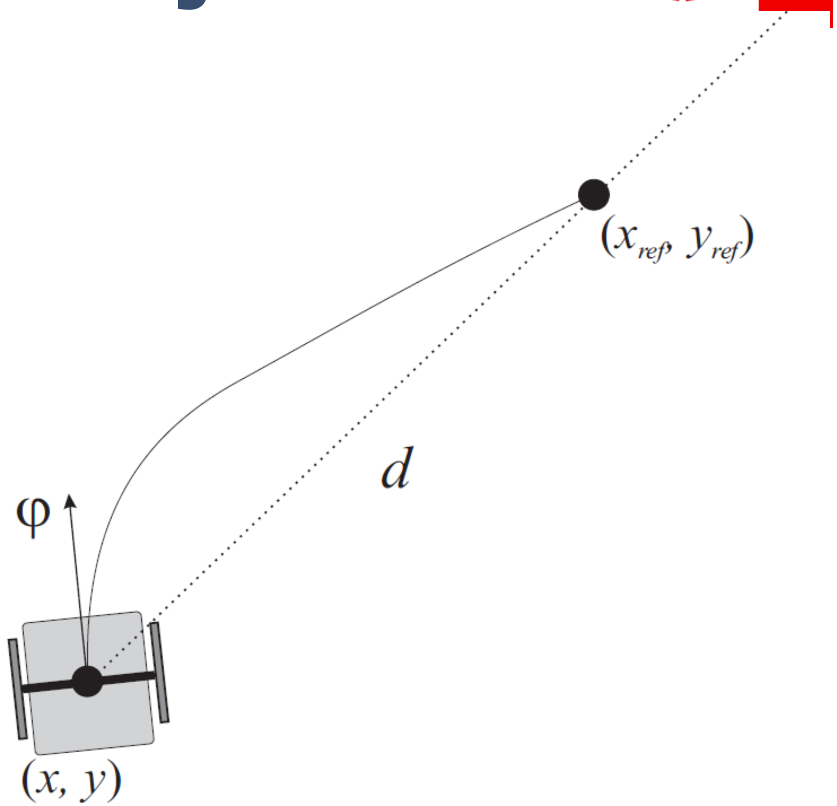
Primeri enostavnega vodenja

- Vodenje v referenčno pozicijo
- Vodenje v referenčno lego z vpeljavo vmesne točke
- Vodenje v referenčno lego z vpeljavo vmesne točke
- Vodenje po odsekoma zvezni poti (premica in krog)

Vodenje v referenčno pozicijo

- V vsakem času vzorčenja določimo

$$\varphi_{ref}(t) = \arctan \frac{y_{ref} - y(t)}{x_{ref} - x(t)}$$



$$\omega(t) = K(\varphi_{ref}(t) - \varphi(t))$$

$$v(t) = K\sqrt{(x_{ref}(t) - x(t))^2 + (y_{ref}(t) - y(t))^2}$$

- Hitrost lahko omejimo z dovoljenim pospeškom
- Prikaz primera (Matlab)

Vodenje v referenčno lego z vpeljavo vmesne točke

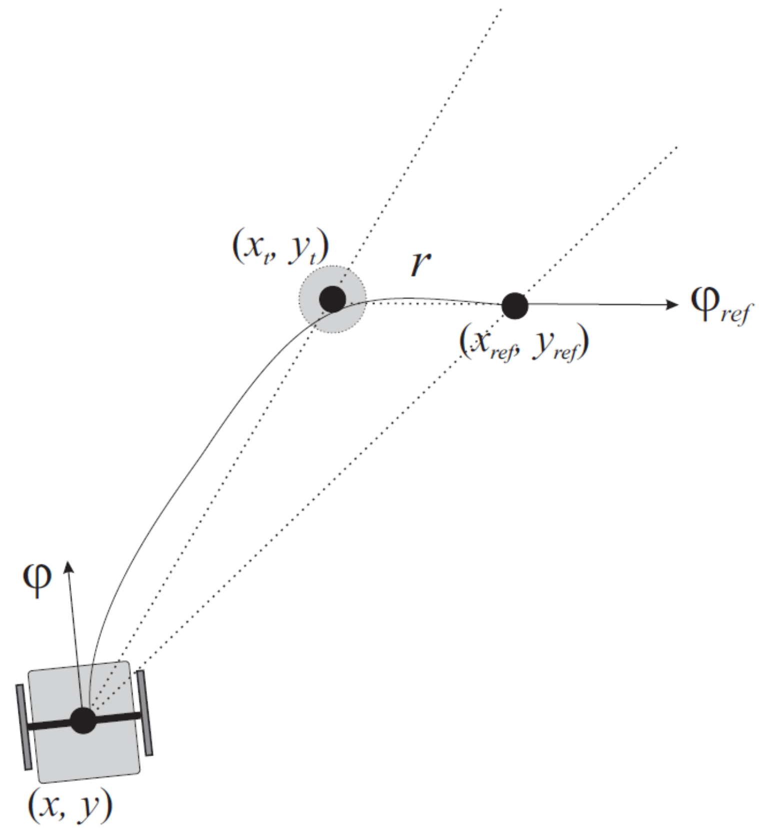
- Želeno orientacijo v končni poziciji dosežemo z vmesno točko:
- Najprej vodimo robota v vmesno točko

$$x_t = x_{ref} - r\cos\varphi_{ref}$$
$$y_t = y_{ref} - r\sin\varphi_{ref}$$

- Ko pridemo v bližino vmesne točke, preklopimo na ciljno točko.

$$\sqrt{(x - x_t)^2 + (y - y_t)^2} < d_{tol}$$

- Lastnosti:
  - Enostavni pristop,
  - Prisotna manjša napaka končne orientacije ,
  - Vmesna pot ni predpisana,
  - Možna vpeljava več vmesnih točk,
  - Potrebna nastavitev dveh dodatnih parametrov r in d_{tol}



Vodenje v referenčno lego z vpeljavo vmesne usmeritve

- Želeno orientacijo v končni poziciji dosežemo z vmesno smerjo

$$\alpha(t) = \varphi_t(t) - \varphi_{ref}$$

$$\varphi_t(t) = \arctan \frac{y_{ref} - y(t)}{x_{ref} - x(t)}$$

- Izračunamo pogrešek smeri kot

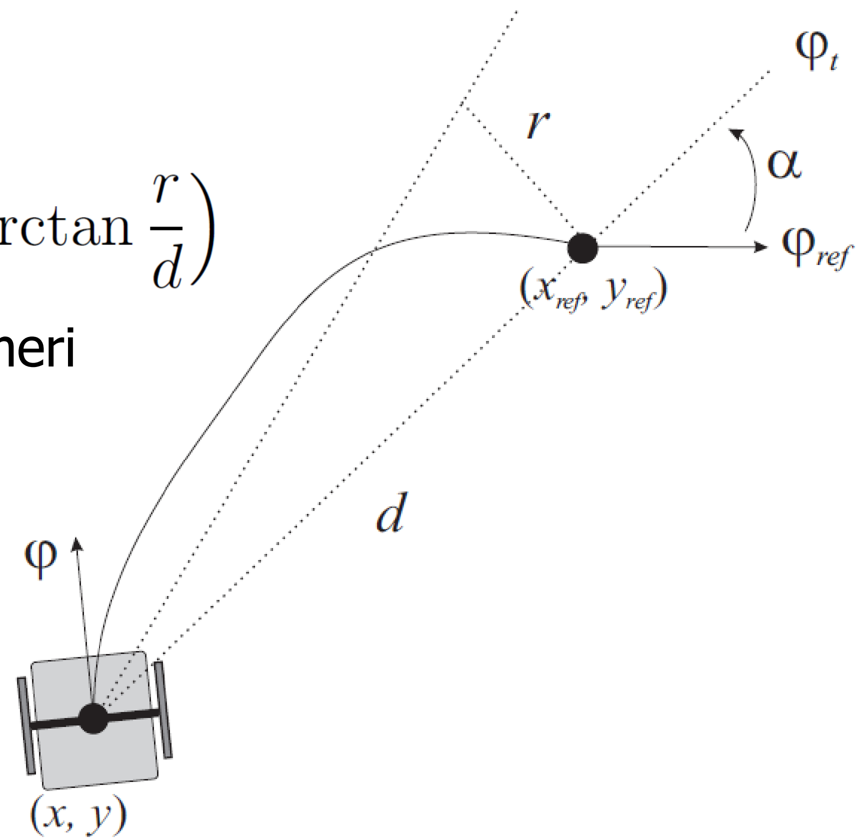
$$\varphi_{err}(t) = \varphi_t(t) - \varphi(t) + \min\left(\alpha(t), \arctan \frac{r}{d}\right)$$

- Algoritem ima dva dela, v prvem se usmeri nekoliko za cilj

$$\varphi_{err}(t) = \varphi_t(t) - \varphi(t) + \arctan \frac{r}{d}$$

- V drugem pa začne krožiti, dokler se ne poravna z ref. smerjo

$$\varphi_{err}(t) = \varphi_t(t) - \varphi(t) + \alpha(t)$$



Lastnosti: mehek preklop med režimoma, parameter *r*, točna končna orientacija

Vodenje po odsekoma zvezni poti (premica in krog)

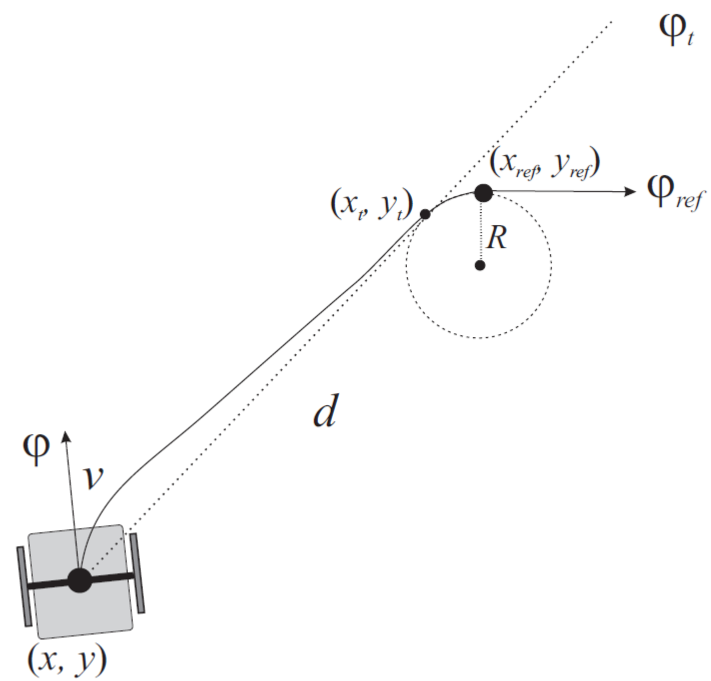
- Določitev poti do cilja iz daljice in krožnega izseka
- Najprej sledi premici do vmesne točke (x_t, y_t)

$$\omega(t) = K(\varphi_t(t) - \varphi(t)) \qquad \varphi_t(t) = \arctan \frac{y_t - y(t)}{x_t - x(t)}$$

- Nato sledi krožnici

$$\omega(t) = \frac{v(t)}{R} + K(\varphi_{ref}(t) - \varphi(t))$$

- *v* se določi glede na želeno hitrost vožnje
- Lastnosti:
  - Potrebna določitev vmesne poti
  - Prehod premica krog pomeni nezveznost za *w*
  - Kratka pot
  - Ref. pot se lahko določi v vsaki iteraciji (večjo robustnost)
  - dodatni parametri



Vodenje po referenčni trajektoriji

- Trajektorija $x_{ref}(t), y_{ref}(t)$ podana parametrično s časom *t*

- Krmiljenje (prekrmiljenje/feedforward) iz trajektorije lahko določimo potrebne vhode, da sistem sledi referenci

$$v_{ff}(t) = \sqrt{(\dot{x}_{ref}(t))^2 + (\dot{y}_{ref}(t))^2}$$

$$\omega_{ff}(t) = \frac{d}{dt}\left[\arctan\left(\frac{\dot{y}_{ref}(t)}{\dot{x}_{ref}(t)}\right)\right] = \frac{\dot{x}_{ref}(t)\ddot{y}_{ref}(t) - \dot{y}_{ref}(t)\ddot{x}_{ref}(t)}{\dot{x}_{ref}(t)^2 + (\dot{y}_{ref}(t)^2}$$

- Dodamo vodenje za odpravo: motenj, negotovosti, zač. pogreškov,…

Linearni regulator za sledenje trajektoriji

- Izpeljava linearnega regulatorja za lineariziran sistem
- Pogrešek definiran lokalno

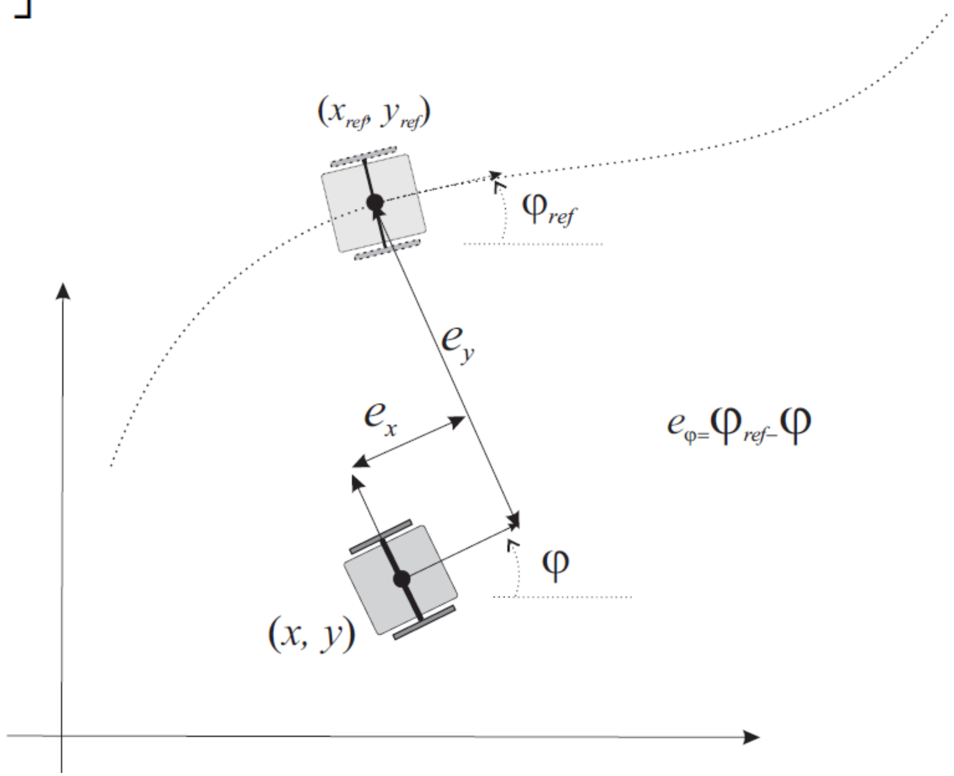
$$\begin{bmatrix} e_x(t) \\ e_y(t) \\ e_\varphi(t) \end{bmatrix} = \begin{bmatrix} \cos(\varphi(t)) & \sin(\varphi(t)) & 0 \\ -\sin(\varphi(t)) & \cos(\varphi(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} (\mathbf{q}_{ref}(t) - \mathbf{q}(t))$$

- Določimo kinematiko pogreška

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\varphi \end{bmatrix} = \begin{bmatrix} \cos e_\varphi & 0 \\ \sin e_\varphi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_{ff} \\ \omega_{ff} \end{bmatrix} + \begin{bmatrix} -1 & e_y \\ 0 & -e_x \\ 0 & -1 \end{bmatrix} \mathbf{u}$$

- Definiramo vhod

$$\mathbf{u} = \begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} v_{ff} \cos e_\varphi + v_{fb} \\ \omega_{ff} + \omega_{fb} \end{bmatrix}$$



Linearni regulator za sledenje trajektoriji…

- Izpeljemo nelinearni model

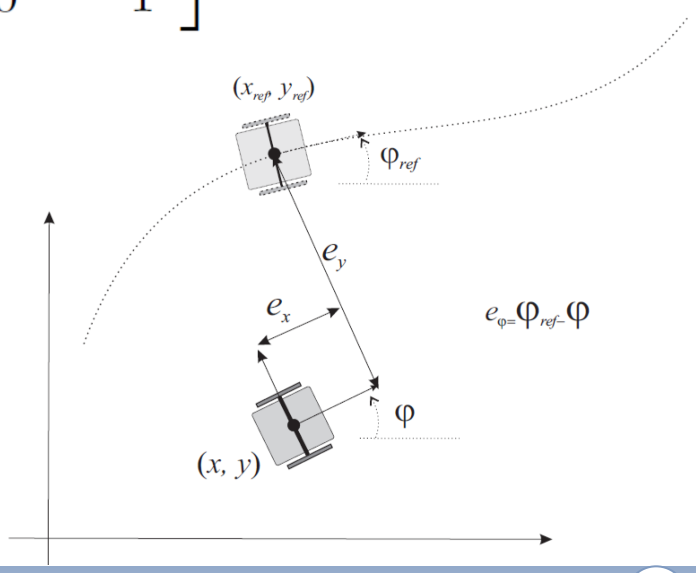
$$\begin{aligned}
\dot{e}_x &= \omega_{ff} e_y - v_{fb} + e_y \omega_{fb} \\
\dot{e}_y &= -\omega_{ff} e_x + v_{ff} \sin e_\varphi - e_x \omega_{fb} \\
\dot{e}_\varphi &= -\omega_{fb}
\end{aligned}$$

- Lineariziramo v okolici ref. trajektorije (ni LTI mode !)

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\varphi \end{bmatrix} = \begin{bmatrix} 0 & \omega_{ff} & 0 \\ -\omega_{ff} & 0 & v_{ff} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} e_x \\ e_y \\ e_\varphi \end{bmatrix} + \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & -1 \end{bmatrix} \mathbf{u}_{fb}$$

- Določimo regulator stanj *K* za lin. model

$$\mathbf{u}_{fb} = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & k_\varphi \end{bmatrix} \begin{bmatrix} e_x \\ e_y \\ e_\varphi \end{bmatrix}$$



Linearni regulator za sledenje trajektoriji...

- Matriko **K** ocenimo glede na želeno dinamiko (dušenje, lastno frekvenco) z Ackerman metodo ali primerjavo karakterističnih polinomov...

$$\mathbf{u}_{fb} = \begin{bmatrix} k_x & 0 & 0 \\ 0 & k_y & k_\varphi \end{bmatrix} \begin{bmatrix} e_x \\ e_y \\ e_\varphi \end{bmatrix}$$

- Želeni karakteristični polinom

$$(s+s_1)(s+s_2)(s+s_3) = 0$$

- Karakteristični polinom

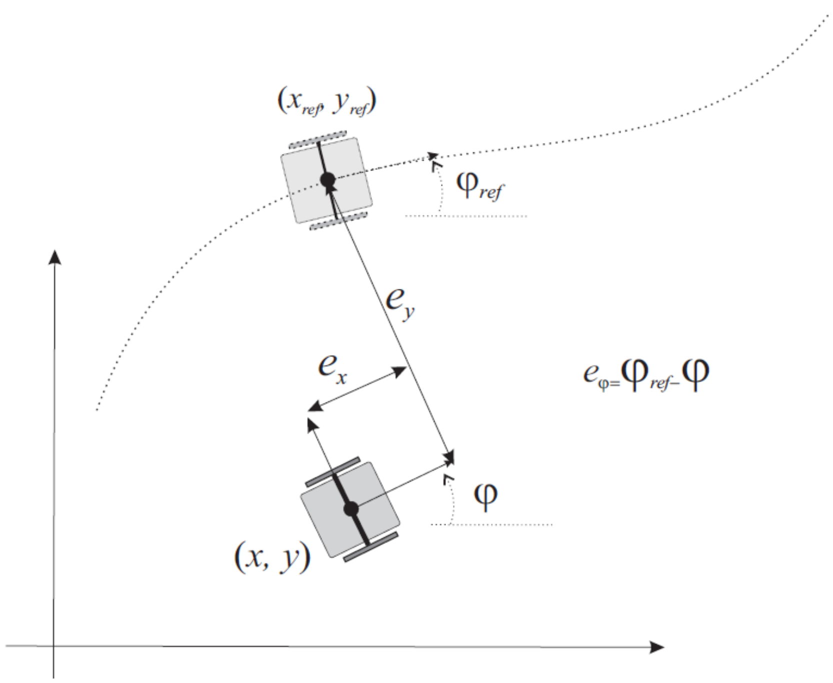
$$\det(s\mathbf{I} - \mathbf{A} + \mathbf{BK}) = 0$$

- Dobimo regulator (singularnost *vff*)

$$k_x = k_\varphi = 2\zeta\omega_n$$
$$k_y = \frac{\omega_n^2 - \omega_{ff}^2}{|v_{ff}|}$$

- Oziroma (dodaten parameter *g*>0)

$$k_x = k_\varphi = 2\zeta\sqrt{\omega_{ff}^2 + g v_{ff}^2}$$
$$k_y = g|v_{ff}|$$



Linearni regulator za sledenje trajektoriji...

- Sistem $\dot{\mathbf{e}} = \mathbf{A}\mathbf{e} + \mathbf{B}\mathbf{u}_{fb}$!= LTI (linear time invariant)

- Vodljivostna matrika ni zadosten pogoj za test vodljivosti

$$\begin{bmatrix} \mathbf{B} & \mathbf{A}\mathbf{B} & \mathbf{A}^2\mathbf{B} \end{bmatrix}$$

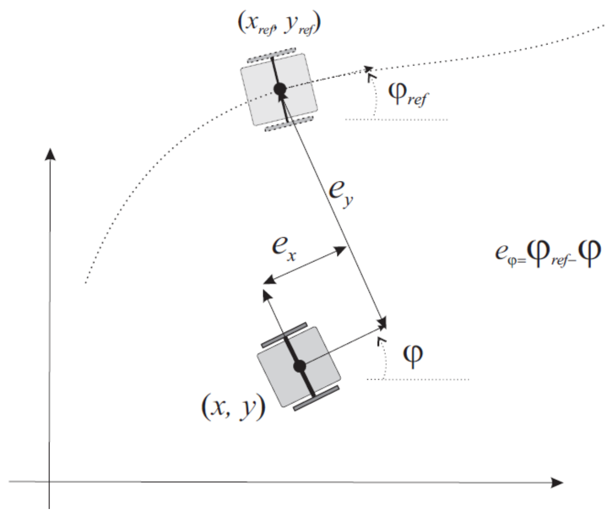
- Popolnoma neholonomičen sistem je vodljiv (Chow teorem) -> diff. pogon je vodljiv

- Če sistem != LTI, potem mora regulator:
  - != LTI ,da je zaprtozančni sistem lahko (globalno) asimptotično stabilen (poljubni začetni pogrešek e(0)->0 ko t->∞)
  - če je = LTI, potem asimptotična stabilnost z gladkim vodenjem (zvezni regulator) ni možna
  - če je = LTI, potem mora biti vodenje nezvezno (preklopi smeri) za asimptotično stabilnost

Nelinearni regulator

- Pogost v praksi in literaturi

$$v_{fb} = k_x e_x$$
$$\omega_{fb} = k_y v_{ff} \frac{\sin e_\varphi}{e_\varphi} e_y + k_\varphi e_\varphi$$



- ni LTI (nelineren in časovno spremenljiv)
- zvezno (gladko) vodenje
- je globalno asimptotično stabilen

(dokaz z Lyapunovo analizo stabilnosti je podan v učbeniku)

Potrebna primerna izbira ojačenj, da je glob. asim. stabilen, npr.:

$$k_x = k_\varphi = 2\zeta\sqrt{\omega_{ff}^2 + g v_{ff}^2}$$ ... pozitivni, zvezni in odvedljivi funkciji

$$k_y = g|v_{ff}|$$ ... pozitivna konstanta

Celoten vhod vsebuje predkrmiljenje

$$v = v_{ff} \cos e_\varphi + k_x e_x$$
$$\omega = \omega_{ff} + k_y v_{ff} \frac{\sin e_\varphi}{e_\varphi} e_y + k_\varphi e_\varphi$$

Nelinearni regulator, analiza stabilnosti

- Lyapunova analiza, sistem je:
  - **asimptotično stabilen**, če obstaja pozitivno definitna funkcija in je njen odvod negativno definiten

$$V(\mathbf{e}) \quad \text{->} \quad \dot{V}(\mathbf{e}) \qquad \text{-> za } \mathbf{e} \neq 0 \text{ je } \dot{V}(\mathbf{e}) < 0$$

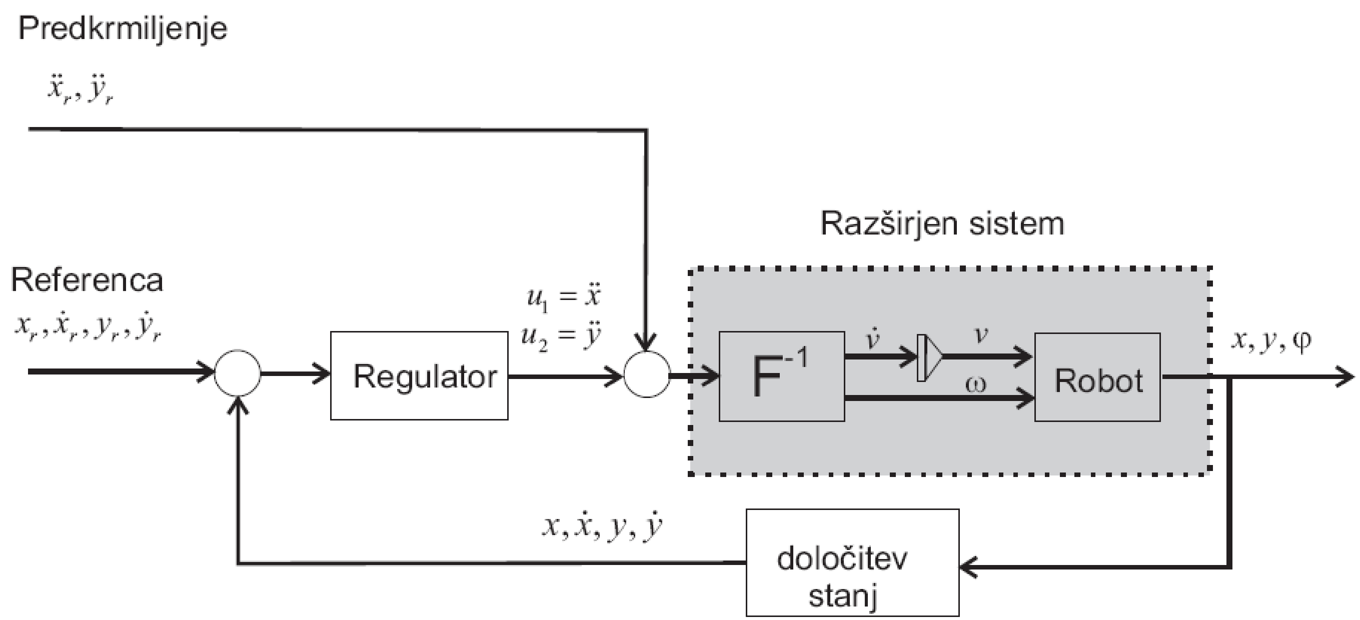
  - **stabilen**, če je odvod $V(e)$ semi-negativno definiten

$$V(\mathbf{e}) \quad \text{->} \quad \dot{V}(\mathbf{e}) \qquad \text{-> za določene } \mathbf{e} \neq 0 \text{ je } \dot{V}(\mathbf{e}) = 0$$

  - **globalno asimptotično stabilen**, če je asimptotično stabilen in če je $V(e)$ zvezno odvedljiva in omejena funkcija
- Izberemo
- $V(\mathbf{e}) = \frac{k_y}{2}(e_x^2 + e_y^2) + \frac{1}{2}e_\varphi^2$ ...izpeljava... $\dot{V}(\mathbf{e}) = -k_x k_y e_x^2 - k_\varphi e_\varphi^2$

Povratnozančna linearizacija, dif. pogon

- Regulacijo izvajamo v zunanjih (globalnih koordinatah), tam definiramo pogrešek za nek nov nabor spremenljivk ***z***.
- Možno za **diferencilno ploske sisteme**, za katere lahko določimo nabor spremenljivk ***z*** -**ploski izhodi sistema** s katerimi lahko izrazimo vsa stanja sistema ***q*** in vse vhode ***u*.**
- Določimo nelinearnim kompenzator $\mathbf{F^{-1}}$ in z njim lineariziramo sistem. Nato določimo linearni regulator.



Povratnozančna linearizacija, dif. pogon...

- Ploska izhoda sta *x* in *y*, dvakrat zapored ju odvajamo, da sta odvoda odvisna od obeh vhodov

$$\begin{aligned}\dot{x} &= v\cos\varphi \\ \dot{y} &= v\sin\varphi\end{aligned} \;\Rightarrow\; \begin{aligned}\ddot{x} &= \dot{v}\cos\varphi - v\sin\varphi\dot{\varphi} \\ \ddot{y} &= \dot{v}\sin\varphi + v\cos\varphi\dot{\varphi}\end{aligned} \;\Rightarrow\; \begin{bmatrix}\ddot{x}\\ \ddot{y}\end{bmatrix} = \begin{bmatrix}\cos\varphi & -v\sin\varphi \\ \sin\varphi & v\cos\varphi\end{bmatrix}\begin{bmatrix}\dot{v}\\ \omega\end{bmatrix} = \mathbf{F}\begin{bmatrix}\dot{v}\\ \omega\end{bmatrix}$$

- Določimo kompenzator in z njim lineariziramo sistem (razširjeni sistem)

$$\begin{bmatrix}\dot{v}\\ \omega\end{bmatrix} = \mathbf{F}^{-1}\begin{bmatrix}\ddot{x}\\ \ddot{y}\end{bmatrix} = \begin{bmatrix}\cos\varphi & \sin\varphi \\ -\frac{\sin\varphi}{v} & \frac{\cos\varphi}{v}\end{bmatrix}\begin{bmatrix}\ddot{x}\\ \ddot{y}\end{bmatrix}$$

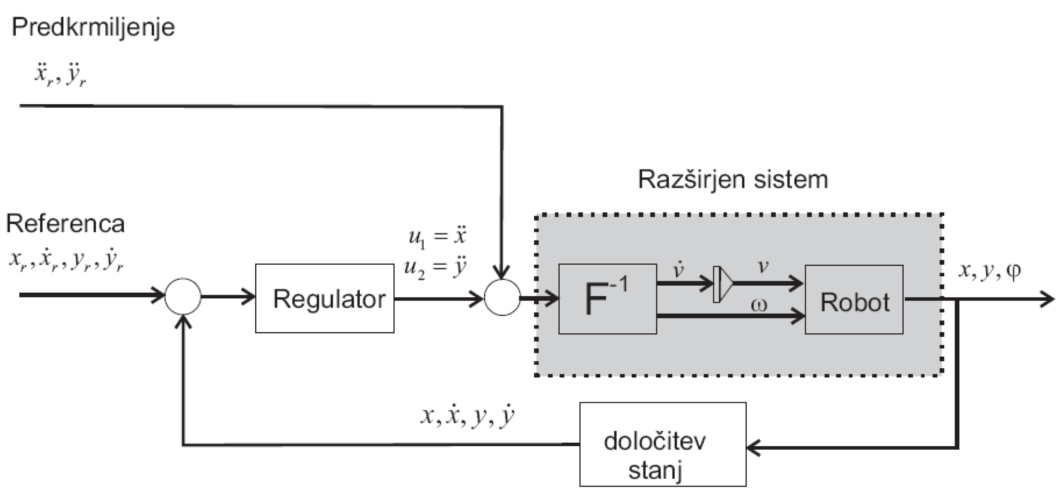
- Regulacijo izvajamo za nova stanja ***z***, regulator načrtamo za razširjeni linearni sistem (regulator PD ali regulator stanj)

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{y}\\ \ddot{y}\end{bmatrix} = \begin{bmatrix}0&1&0&0\\0&0&0&0\\0&0&0&1\\0&0&0&0\end{bmatrix}\begin{bmatrix}x\\ \dot{x}\\ y\\ \dot{y}\end{bmatrix} + \begin{bmatrix}0&0\\1&0\\0&0\\0&1\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\end{bmatrix} = \begin{bmatrix}0&1\\0&0\end{bmatrix}\begin{bmatrix}x\\ \dot{x}\end{bmatrix} + \begin{bmatrix}0\\1\end{bmatrix}u_1$$

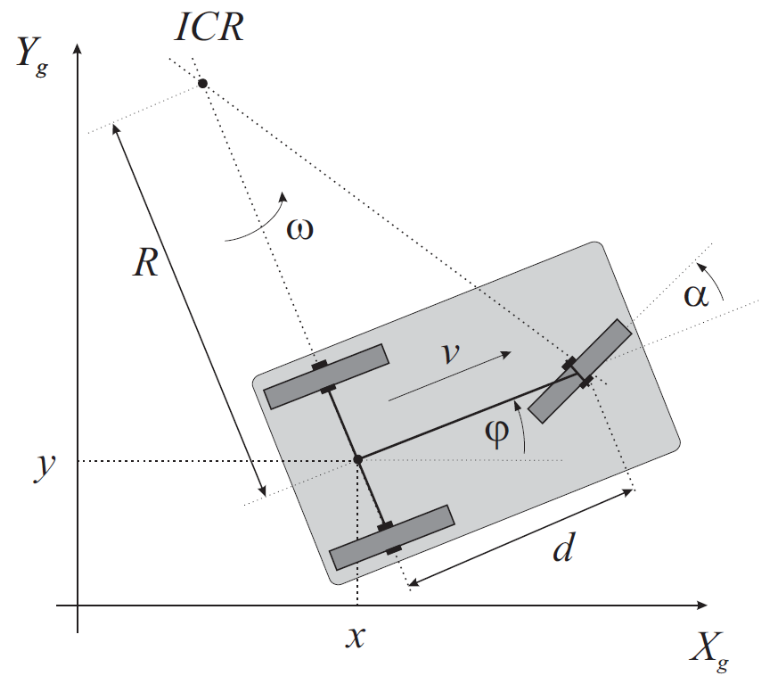
$$\begin{bmatrix}\dot{y}\\ \ddot{y}\end{bmatrix} = \begin{bmatrix}0&1\\0&0\end{bmatrix}\begin{bmatrix}y\\ \dot{y}\end{bmatrix} + \begin{bmatrix}0\\1\end{bmatrix}u_2$$



Vodenje gibanja akermanovega pogona

- Kinematika

$$\dot{x} = v_r(t)\cos(\varphi(t))$$
$$\dot{y} = v_r(t)\sin(\varphi(t))$$
$$\dot{\varphi} = \frac{v_r(t)}{d}\tan(\alpha(t))$$

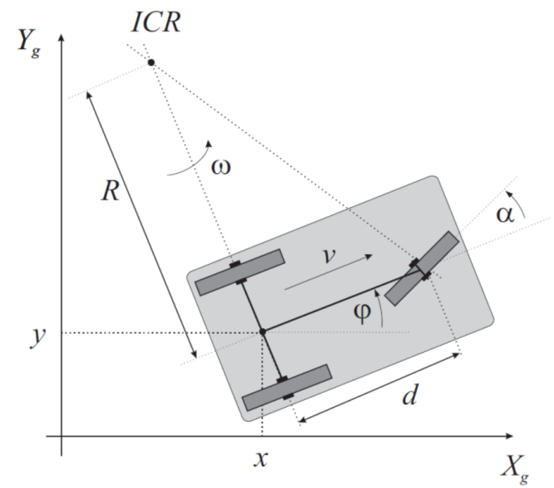


- Enostaven pristop, ločen regulator za smer in hitrost

Vodenje smeri, Akermanov pogon

- Kot krmiljenja proporcionalen smernemu pogrešku

$$\alpha(t) = K(\varphi_{ref}(t) - \varphi(t))$$



- Linearizacija kota

$$\dot{\varphi}(t) = \frac{v_r(t)}{d}\tan(\alpha(t)) \approx \frac{v_r(t)}{d}\alpha(t)$$

- Integrirni proces -> sledenje brez pogreška v ustaljenem stanju
- Zaprtozančna prenosna funkcija

$$G_z(s) = \frac{\phi(s)}{\phi_{ref}(s)} = \frac{1}{\frac{d}{VK}s + 1}$$

- Za hitrejši regulator lahko določimo

$$\dot{\alpha}(t) = K_1(\varphi_{ref}(t) - \varphi(t)) - K_2\dot{\varphi}(t)$$

$$G_z(s) = \frac{\phi(s)}{\phi_{ref}(s)} = \frac{K_1\frac{V}{d}}{s^2 + K_2\frac{V}{d}s + K_1\frac{V}{d}}$$

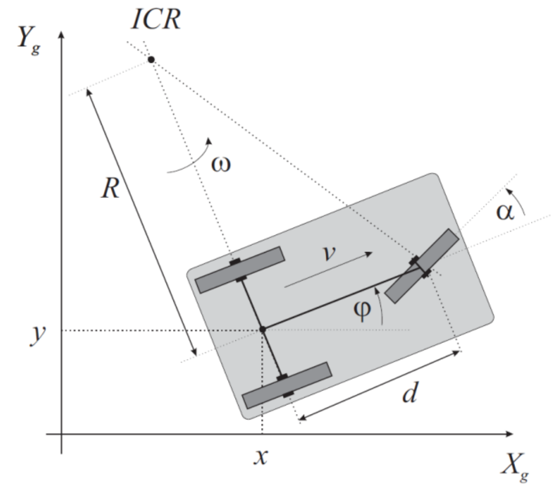
$$\omega_n = \sqrt{K_1\frac{V}{d}}$$

$$\zeta = \frac{K_2}{2}\sqrt{\frac{V}{dK_1}}$$

Vodenje hitrosti, Akermanov pogon

- Hkrati z vodenjem smeri, vodimo tudi hitrost
- za ref. pozicijo $(x_{ref}(t), y_{ref}(t))$ definiramo pogrešek

$$e_d = \sqrt{(x_{ref}(t) - x(t))^2 + (y_{ref}(t) - y(t))^2}$$



- Proporcionalni regulator

$$v(t) = K\sqrt{(x_{ref}(t) - x(t))^2 + (y_{ref}(t) - y(t))^2}$$

Vodenje v referenčno pozicijo (Akerman)

- Hkrati izvajamo vodenje po smeri in hitrosti
- Referenca za kot je določena z

$$\varphi_{ref}(t) = \arctan \frac{y_{ref} - y(t)}{x_{ref} - x(t)}$$

$$\alpha(t) = K(\varphi_{ref}(t) - \varphi(t))$$

$$v(t) = K\sqrt{(x_{ref}(t) - x(t))^2 + (y_{ref}(t) - y(t))^2}$$

- Hitrost lahko omejimo z dovoljenim pospeškom
- Prikaz primera (Matlab)

Vodenje po referenčni trajektoriji (Akerman)

- Referenčna točka se spreminja, podana je s trajektorijo
  $x_{ref}(t),\ y_{ref}(t)$
- Določimo referenčno smer za vsak trenutek t

$$\varphi_{ref}(t) = \arctan\frac{y_{ref}-y(t)}{x_{ref}-x(t)}$$

Uporabimo regulator za smer in regulator za hitrost

$$\alpha(t) = K(\varphi_{ref}(t) - \varphi(t))$$

$$v(t) = K\sqrt{(x_{ref}(t) - x(t))^2 + (y_{ref}(t) - y(t))^2}$$

- Dodamo predkrmiljenje

$$v_{ff}(t) = \sqrt{(\dot{x}_{ref}(t))^2 + (\dot{y}_{ref}(t))^2}$$

$$\omega_{ff}(t) = \frac{d}{dt}\left[\arctan\left(\frac{\dot{y}_{ref}(t)}{\dot{x}_{ref}(t)}\right)\right] = \frac{\dot{x}_{ref}(t)\ddot{y}_{ref}(t) - \dot{y}_{ref}(t)\ddot{x}_{ref}(t)}{\dot{x}_{ref}(t)^2 + (\dot{y}_{ref}(t))^2}$$

$$\alpha_{ff}(t) = \arctan\frac{d\,\omega_{ff}(t)}{v_{ff}(t)}$$

Vodenje gibanja– je to bistvo MS?

- Je pomembno, ni pa vse
- Pred izvedbo vodenja moramo:
  - Dobiti potrebno **senzorsko info. za vodenje**, ki ponavadi ni neposredno dostopna. Zato se ocenjuje iz drugih meritev (razdalj do ovir, meritev hitrosti, slike,...), modela senzorja, modela robota in okolice. Orodja za to so observatorji oz. pri MS bolj pogosto filtri (npr. Kalmanov filter, filter delcev)
  - Določitev **ustrezne poti**, ki vodi do cilja. Pot večinoma ni podana, treba jo je določiti:
    - Kako doseči cilj?, Po kakšni poti?,
    - Izogibanje ovir,
    - Optimalnost poti (čas, dolžina, energija,...)
- Vodenje gibanja je lahko le del neke obsežnejše akcije (opravila), ki jo sistem izvaja