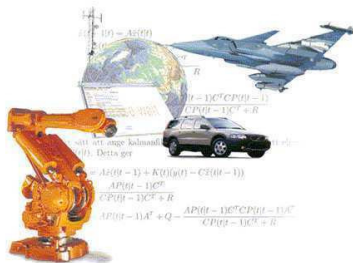


# Licentiatseminarium

## *On Sensor Fusion Applied to Industrial Manipulators*



Patrik Axelsson

Division of Automatic Control  
Department of Electrical Engineering  
Linköping University



1. Introduktion
2. Modelling
3. Resultat
4. Sammanfattning och framtida arbete



## 1. Introduktion

- Reglerteknik
- Industrirobotik
- Problemformulering
- Tillståndsskattning

## 2. Modellering

## 3. Resultat

## 4. Sammanfattning och framtida arbete



**Reglerteknik** handlar om att få ett system (bil, robot, pappersmaskin, mobiltelefon, . . . ) att göra det man vill.



**Reglerteknik** handlar om att få ett system (bil, robot, pappersmaskin, mobiltelefon, ...) att göra det man vill.

- Reglerteknik underlättar applikationer som är
  - farliga
  - svåra
  - repetitiva
  - tråkiga
  - ...



**Reglerteknik** handlar om att få ett system (bil, robot, pappersmaskin, mobiltelefon, ...) att göra det man vill.

■ Reglerteknik underlättar applikationer som är

- farliga
- svåra
- repetitiva
- tråkiga
- ...

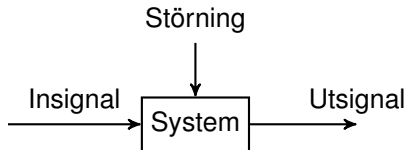
■ Men även de som kräver

- hög noggrannhet
- hög produktionstakt
- ...



System med:

- Insignal (det man kan ändra/påverka),
- Utsignal (det man vill styra/kan mäta),
- Störning (det man inte kan påverka).

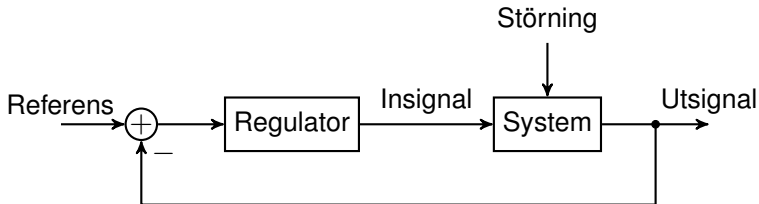


System med:

- Insignal (det man kan ändra/påverka),
- Utsignal (det man vill styra/kan mäta),
- Störning (det man inte kan påverka).

Återkoppling

- Återkopplar det man kan mäta.
- Jämför med en referens.
- Regulator som beräknar en insignal.





Första ...

- ... patentansökan 1954. Beviljades 1961.
- ... roboten Unimate lanserades 1959.
- ... robotinstallationen 1961 för General Motors.
- ... mikroprocessorstyrda roboten IRB6 lanserades av ASEA (nuvarande ABB) 1973.



## Första ...

- ... patentansökan 1954. Beviljades 1961.
- ... roboten Unimate lanserades 1959.
- ... robotinstallationen 1961 för General Motors.
- ... mikroprocessorstyrda roboten IRB6 lanserades av ASEA (nuvarande ABB) 1973.

## Stora tillverkare:

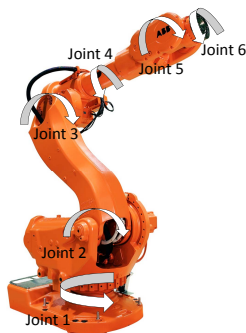
- ABB (Sverige/Schweiz),
- KUKA (Tyskland),
- Motoman (Japan),
- FANUC Robotics (USA).



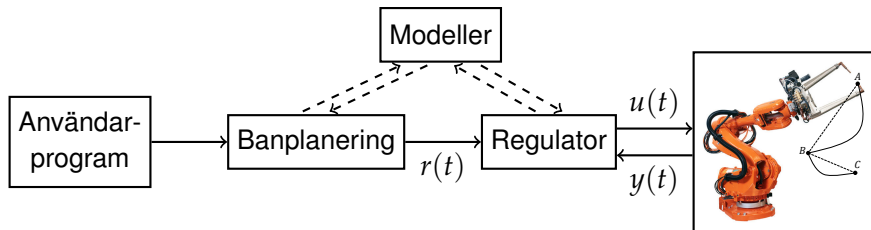
En seriell robot har

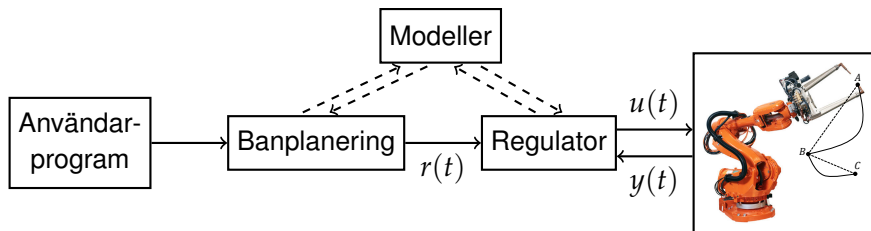
- $n$  leder vilket innebär  $n$  frihetsgrader (DOF) (Ofta 6 st),
- $n + 1$  länkar,
- Aktuatorer: Elektriska motorer,
- Sensorer: Enkoder/resolver som mäter motorvinklarna  $q_m$ .

Verktygets position (TCP) beskrivs av ledvinklarna  $q_a$  (armvinklar).



För en ideal växellåda gäller  $q_m = \eta q_a$ . I praktiken är det en vek koppling  $\Rightarrow$  innehåller dynamik.





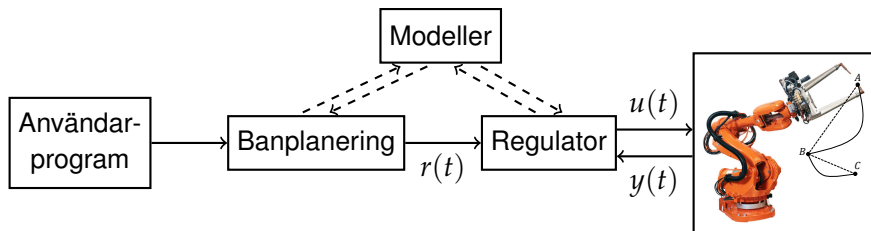
## Modeller

- Kinematiska modeller. Robotens rörelse utan påverkan av en yttre kraft.

$$\text{Framåt: } X = Y(q_a), \dot{X} = \frac{\partial Y(q_a)}{\partial q_a} \dot{q}_a = \mathcal{J}(q_a) \dot{q}_a,$$

$$\ddot{X} = \mathcal{J}(q_a) \ddot{q}_a + \left( \frac{d}{dt} \mathcal{J}(q_a) \right) \dot{q}_a.$$



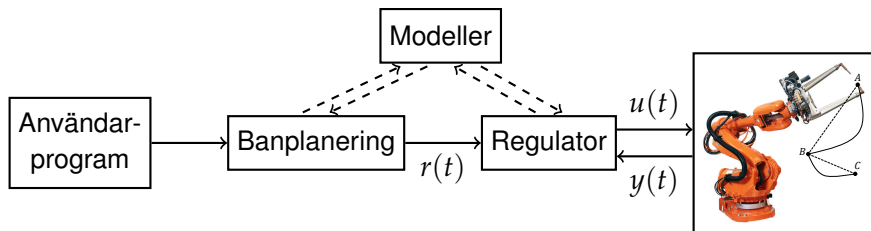


## Modeller

- Kinematiska modeller. Robotens rörelse utan påverkan av en yttre kraft.

$$\text{Invers: } q_a = Y^{-1}(X), \quad \dot{q}_a = \mathcal{J}^{-1}(q_a)\dot{X},$$

$$\ddot{q}_a = \mathcal{J}^{-1}(q_a) \left( \ddot{X} - \frac{d}{dt} (\mathcal{J}(q_a)) \dot{q}_a \right).$$

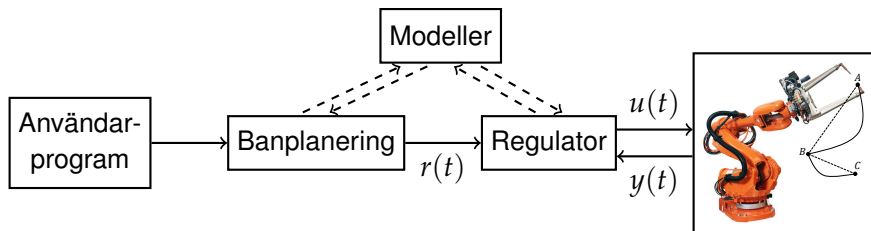


## Modeller

- Dynamiska modeller. Robotens rörelse när en yttre kraft påverkar.

$$q = (q_a^T \quad q_m^T)^T,$$

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) + T(q) + D(q) + F(q) = u.$$

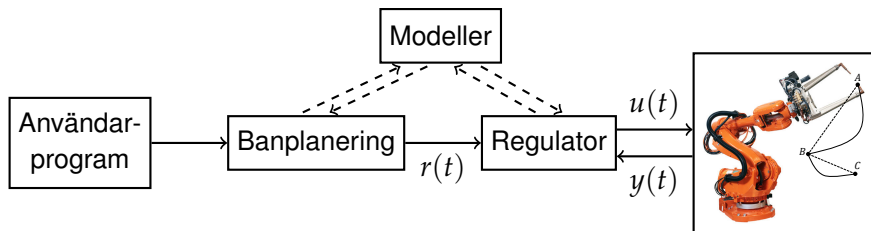


## Banplanering

- Tar kommandona från användarprogrammet och skapar en geometrisk bana för TCP.
- Gör om den geometriska banan för TCP till hur axlarna ska röra sig.
- Läger på hastighets- och accelerationskraven.
- Även fysiska krav på roboten tillkommer, t.ex. max moment för motorer/växellådor.

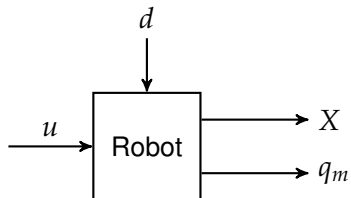


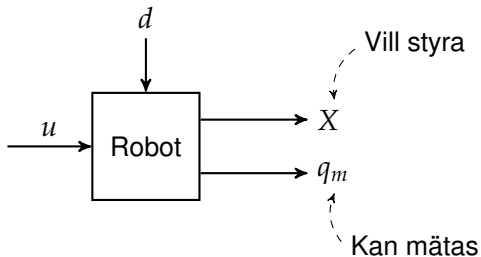




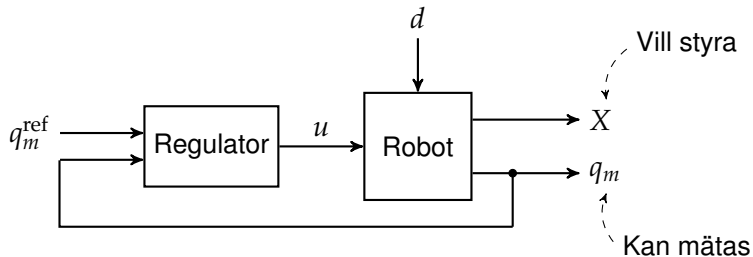
## Regulator

- **Framkoppling.**
  - Beräknar det moment som krävs för att klara av gravitationen.
  - Beräknar motorreferenser från armreferenserna.
- **Återkoppling.** Mäter motorvinklarna och jämför med referensen. Ofta räcker det med en diagonal PID-regulator.

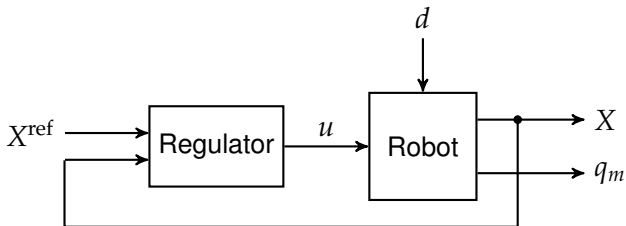




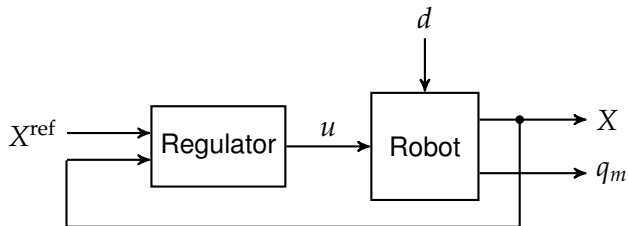
- Vill styra TCP men kan endast mäta motorvinklarna  $q_m$ .



- Vill styra TCP men kan endast mäta motorvinklarna  $q_m$ .
- Återkoppling av  $q_m$  fungerar ej tillräckligt bra p.g.a. flexibla leder.

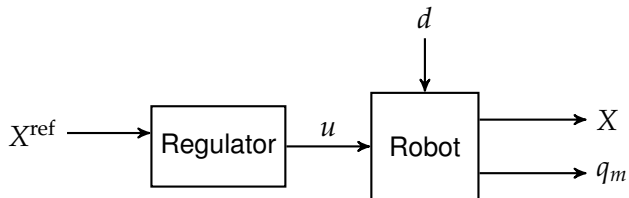


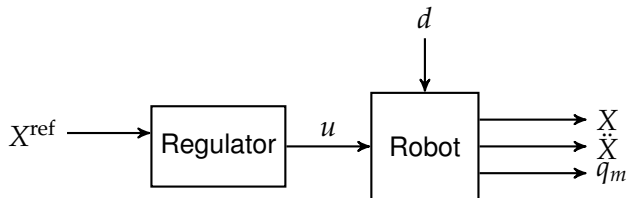
- Vill styra TCP men kan endast mäta motorvinklarna  $q_m$ .
- Återkoppling av  $q_m$  fungerar ej tillräckligt bra p.g.a. flexibla leder.
- Om TCP kan mätas är det naturligt att återkoppla detta.



- Vill styra TCP men kan endast mäta motorvinklarna  $q_m$ .
- Återkoppling av  $q_m$  fungerar ej tillräckligt bra p.g.a. flexibla leder.
- Om TCP kan mätas är det naturligt att återkoppla detta.

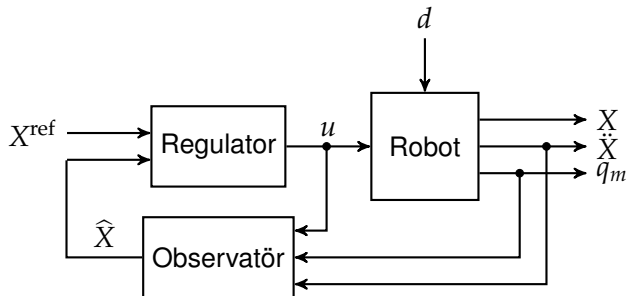
Hur ska man göra istället för att mäta TCP?



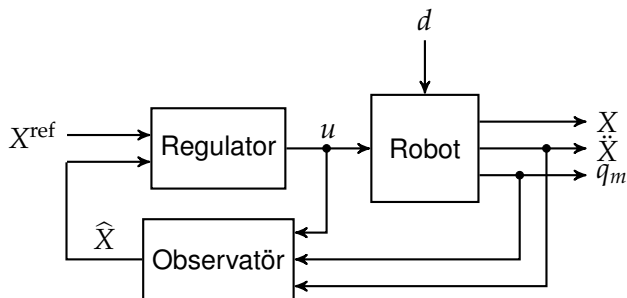


- Inför en mätning av TCP i form av accelerationen.





- Inför en mätning av TCP i form av accelerationen.
- Skapa en observatör för att skatta TCP.



- Inför en mätning av TCP i form av accelerationen.
- Skapa en observatör för att skatta TCP.

Hur ska man skatta TCP?

Givet en modell för systemet

$$\begin{aligned}x_{k+1} &= f(x_k, u_k, v_k) \sim p(x_{k+1}|x_k) \\ y_k &= h(x_k, u_k, e_k) \sim p(y_k|x_k)\end{aligned}$$

samt  $y_{1:k}$  och  $u_{1:k}$ .



Givet en modell för systemet

$$\begin{aligned}x_{k+1} &= f(x_k, u_k, v_k) \sim p(x_{k+1}|x_k) \\ y_k &= h(x_k, u_k, e_k) \sim p(y_k|x_k)\end{aligned}$$

samt  $y_{1:k}$  och  $u_{1:k}$ . Då kan tillstånden  $x_k$  beräknas från

$$p(x_k|y_{1:k}) = \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})},$$

$$p(x_{k+1}|y_{1:k}) = \int_{\mathbb{R}^{n_x}} p(x_{k+1}|x_k)p(x_k|y_{1:k}) \, dx_k,$$

$$p(y_k|y_{1:k-1}) = \int_{\mathbb{R}^{n_x}} p(y_k|x_k)p(x_k|y_{1:k-1}) \, dx_k,$$

där  $k = 1, 2, \dots, N$ .



- En analytisk lösning till dessa ekvationer existerar endast i det linjära fallet. Mer känt som Kalman filtret.
- Approximativa lösningar måste användas i det olinjära fallet.



- En analytisk lösning till dessa ekvationer existerar endast i det linjära fallet. Mer känt som Kalman filtret.
- Approximativa lösningar måste användas i det olinjära fallet.
- Här används
  - Extended Kalman filter (EKF) (Extended Kalman smoother (EKS))
    - Approximerar systemet med en linjärisering av de olinjära ekvationerna.
    - Antar additivt Gaussiskt brus.



- En analytisk lösning till dessa ekvationer existerar endast i det linjära fallet. Mer känt som Kalman filtret.
- Approximativa lösningar måste användas i det olinjära fallet.
- Här används
  - Extended Kalman filter (EKF) (Extended Kalman smoother (EKS))
    - Approximerar systemet med en linjärisering av de olinjära ekvationerna.
    - Antar additivt Gaussiskt brus.
  - Particle filter (PF)
    - Approximerar “posterior” fördelningen med ett stort antal partiklar med tillhörande vikter.
    - Den optimala “proposal” fördelningen approximeras med ett EKF.

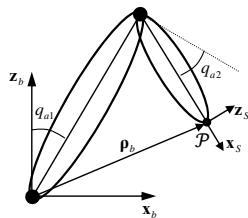


1. Introduktion
2. Modellering
  - Robotmodell
  - Accelerometermodell
  - Skattningsmodeller
3. Resultat
4. Sammanfattning och framtida arbete

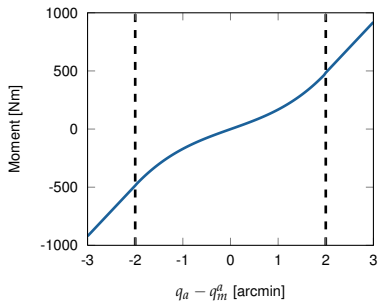




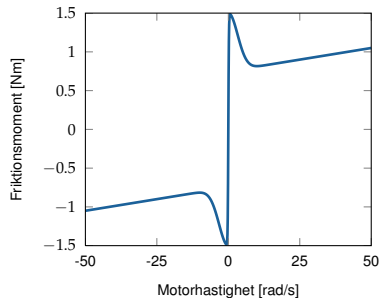
- Seriell robot med 2 DOF.



- Seriell robot med 2 DOF.
- Olinjära vektorer.



- Seriell robot med 2 DOF.
- Olinjära vektorer.
- Olinjär friktion.

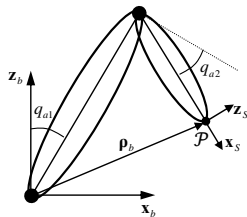


- Seriell robot med 2 DOF.
- Olinjära vektorer.
- Olinjär friktion.

$$q_a = (q_{a1} \quad q_{a2})^T,$$

$$q_m^a = (q_{m1}/\eta_1 \quad q_{m2}/\eta_2)^T,$$

$$\tau_m^a = (\tau_{m1}\eta_1 \quad \tau_{m2}\eta_2)^T,$$

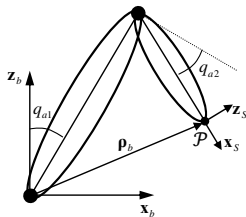


- Seriell robot med 2 DOF.
- Olinjära vektorer.
- Olinjär friktion.

$$q_a = (q_{a1} \quad q_{a2})^T,$$

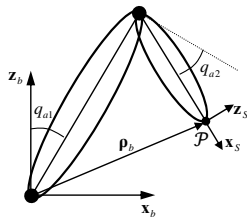
$$q_m^a = (q_{m1}/\eta_1 \quad q_{m2}/\eta_2)^T,$$

$$\tau_m^a = (\tau_{m1}\eta_1 \quad \tau_{m2}\eta_2)^T,$$



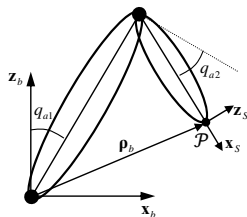
$$M_a(q_a)\ddot{q}_a + C(q_a, \dot{q}_a) + G(q_a) + T(q_a - q_m^a) + D(\dot{q}_a - \dot{q}_m^a) = 0,$$

$$M_m\ddot{q}_m^a + F(\dot{q}_m^a) - T(q_a - q_m^a) - D(\dot{q}_a - \dot{q}_m^a) = \tau_m^a.$$



$$\ddot{\mathbf{p}}_s(q_a) =$$

- Accelerationen från rörelsen.

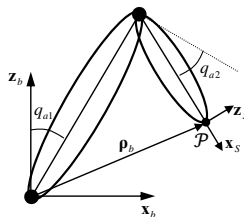


$$\ddot{\rho}_s(q_a) = \ddot{\rho}_b(q_a)$$

$$\rho_b(q_a) = Y_{\text{ACC}}(q_a),$$

$$\ddot{\rho}_b(q_a) = \frac{d^2}{dt^2} Y_{\text{ACC}}(q_a) = \mathcal{J}_{\text{ACC}}(q_a) \ddot{q}_a + \left( \frac{d}{dt} \mathcal{J}_{\text{ACC}}(q_a) \right) \dot{q}_a$$

- Accelerationen från rörelsen.
- Accelerationen från gravitationen.

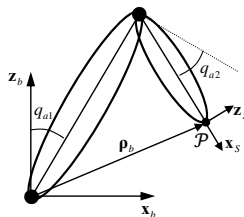


$$\ddot{\rho}_s(q_a) = \ddot{\rho}_b(q_a) + G_b$$

$$G_b = \begin{pmatrix} 0 \\ g \end{pmatrix}$$



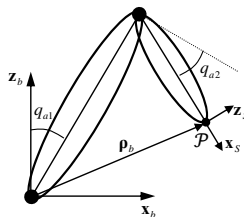
- Accelerationen från rörelsen.
- Accelerationen från gravitationen.
- Rotationsmatris från  $Ox_bz_b$  till  $Ox_sz_s$ .



$$\ddot{p}_s(q_a) = \mathcal{R}_{b/s}(q_a) (\ddot{p}_b(q_a) + G_b)$$

$$\mathcal{R}_{b/s}(q_a) = \begin{pmatrix} \cos(q_{a1} + q_{a2}) & -\sin(q_{a1} + q_{a2}) \\ \sin(q_{a1} + q_{a2}) & \cos(q_{a1} + q_{a2}) \end{pmatrix}$$

- Accelerationen från rörelsen.
- Accelerationen från gravitationen.
- Rotationsmatris från  $Ox_bz_b$  till  $Ox_sz_s$ .
- Biasparameter.



$$\ddot{\rho}_s(q_a) = \mathcal{R}_{b/s}(q_a) (\ddot{\rho}_b(q_a) + G_b) + \mathbf{b}^{\text{ACC}}$$

$$\mathbf{b}^{\text{ACC}} = \begin{pmatrix} \mathbf{b}^{\text{ACC},x} \\ \mathbf{b}^{\text{ACC},z} \end{pmatrix}$$

## 1. Olinjär modell

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$



## 1. Olinjär modell

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Fysikaliska modellen ger kontinuerlig tillståndsmodell.



## 1. Olinjär modell

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Fysikaliska modellen ger kontinuerlig tillståndsmodell.
- Diskretisering med Euler framåt ger

$$x_{k+1} = f(x_k, u_k) + g(x_k)v_k.$$



## 1. Olinjär modell

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Fysikaliska modellen ger kontinuerlig tillståndsmodell.
- Diskretisering med Euler framåt ger

$$x_{k+1} = f(x_k, u_k) + g(x_k)v_k.$$

- Mätetekvation:

$$y_k = \left( \mathcal{R}_{b/s}(x_{1,k}) \left( J_{\text{ACC}}(x_{1,k}) \dot{q}_{a,k} + \left( \frac{d}{dt} J_{\text{ACC}}(x_{1,k}) \right) x_{3,k} + G_b \right) \right) + e_k$$



## 2. Linjär dynamikmodell

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T)^T = (q_a^T \quad \dot{q}_a^T \quad \ddot{q}_a^T)^T$$



## 2. Linjär dynamikmodell

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T)^T = (q_a^T \quad \dot{q}_a^T \quad \ddot{q}_a^T)^T$$

- Linjär dynamikmodell (Dubbelintegrator i diskret tid):

$$x_{k+1} = \mathcal{F}x_k + \mathcal{G}_u u_k + \mathcal{G}_v v_k$$





## 2. Linjär dynamikmodell

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T)^T = (q_a^T \quad \dot{q}_a^T \quad \ddot{q}_a^T)^T$$

- Linjär dynamikmodell (Dubbelintegrator i diskret tid):

$$x_{k+1} = \mathcal{F}x_k + \mathcal{G}_u u_k + \mathcal{G}_v v_k$$

- Insignalen är tredjederivatatan av armvinkelreferensen.



## 2. Linjär dynamikmodell

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T)^T = (q_a^T \quad \dot{q}_a^T \quad \ddot{q}_a^T)^T$$

- Linjär dynamikmodell (Dubbelintegrator i diskret tid):

$$x_{k+1} = \mathcal{F}x_k + \mathcal{G}_u u_k + \mathcal{G}_v v_k$$

- Insignalen är tredjederivatatan av armvinkelreferensen.
- Mätekvation:

$$y_k = \begin{pmatrix} x_{1,k} + K^{-1} (M_a(x_{1,k})x_{3,k} + C(x_{1,k}, x_{2,k}) + G(x_{1,k})) \\ \mathcal{R}_{b/s}(x_{1,k}) \left( J_{\text{ACC}}(x_{1,k})x_{3,k} + \left( \frac{d}{dt} J_{\text{ACC}}(x_{1,k}) \right) x_{2,k} + G_b \right) \end{pmatrix} + e_k$$



### 3. Olinjär modell med accelerationen som insignal

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$



### 3. Olinjär modell med accelerationen som insignal

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Löser ut armvinkelaccelerationen ur accelerometersignalen.



### 3. Olinjär modell med accelerationen som insignal

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Löser ut armvinkelaccelerationen ur accelerometersignalen.
- Fysikaliska modellen + Euler framåt ger

$$x_{k+1} = \begin{pmatrix} x_{1,k} + T_s x_{3,k} \\ x_{2,k} + T_s x_{4,k} \\ x_{3,k} + T_s \ddot{q}_{a,k}^{\text{IN}} \\ x_{4,k} + T_s M_m^{-1} (u_k - F(x_{4,k}) + T(x_{1,k} - x_{2,k}) + D(x_{3,k} - x_{4,k})) \end{pmatrix} + g(x_k) v_k.$$



### 3. Olinjär modell med accelerationen som insignal

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Löser ut armvinkelaccelerationen ur accelerometersignalen.
- Fysikaliska modellen + Euler framåt ger

$$x_{k+1} = \begin{pmatrix} x_{1,k} + T_s x_{3,k} \\ x_{2,k} + T_s x_{4,k} \\ x_{3,k} + T_s \ddot{q}_{a,k}^{IN} \\ x_{4,k} + T_s M_m^{-1} (u_k - F(x_{4,k}) + T(x_{1,k} - x_{2,k}) + D(x_{3,k} - x_{4,k})) \end{pmatrix} + g(x_k) v_k.$$

- Mättekvation:

$$y_k = \begin{pmatrix} 0 & \mathbf{I} & 0 & 0 \end{pmatrix} x_k + e_k$$



#### 4. Linjär modell med accelerationen som insignal

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Löser ut armvinkelaccelerationen ur accelerometersignalen.



#### 4. Linjär modell med accelerationen som insignal

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Löser ut armvinkelaccelerationen ur accelerometersignalen.
- Fysikaliska modellen + linjär fjäder, dämpare och friktion ger

$$\dot{x} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ M_m^{-1}K & -M_m^{-1}K & M_m^{-1}D & -M_m^{-1}(D + F_d) \end{pmatrix} x + \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \\ \mathbf{0} & M_m^{-1} \end{pmatrix} \begin{pmatrix} \ddot{q}_a^{IN} \\ u \end{pmatrix}.$$





#### 4. Linjär modell med accelerationen som insignal

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Löser ut armvinkelaccelerationen ur accelerometersignalen.
- Fysikaliska modellen + linjär fjäder, dämpare och friktion ger

$$\dot{x} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ M_m^{-1}K & -M_m^{-1}K & M_m^{-1}D & -M_m^{-1}(D + F_d) \end{pmatrix} x + \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \\ \mathbf{0} & M_m^{-1} \end{pmatrix} \begin{pmatrix} \ddot{q}_a^{IN} \\ u \end{pmatrix}.$$

- Diskretiseras med ZOH.



#### 4. Linjär modell med accelerationen som insignal

- Tillståndsvektor:

$$x = (x_1^T \quad x_2^T \quad x_3^T \quad x_4^T)^T = (q_a^T \quad q_m^{a,T} \quad \dot{q}_a^T \quad \dot{q}_m^{a,T})^T$$

- Löser ut armvinkelaccelerationen ur accelerometersignalen.
- Fysikaliska modellen + linjär fjäder, dämpare och friktion ger

$$\dot{x} = \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ M_m^{-1}K & -M_m^{-1}K & M_m^{-1}D & -M_m^{-1}(D + F_d) \end{pmatrix} x + \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \\ \mathbf{0} & M_m^{-1} \end{pmatrix} \begin{pmatrix} \ddot{q}_a^{IN} \\ u \end{pmatrix}.$$

- Diskretiseras med ZOH.
- Mättekvation:

$$y_k = (\mathbf{0} \quad \mathbf{I} \quad \mathbf{0} \quad \mathbf{0}) x_k + e_k$$



1. Introduktion
2. Modellering
3. Resultat
  - Simuleringsresultat
  - Skattning av kovariansmatriser
  - Beräkning av accelerometers position och orientering
  - Experimentella resultat
4. Sammanfattning och framtida arbete



- Olinjära skattningsmodellen och EKF med och utan bias-kompensering.



- Olinjära skattningsmodellen och EKF med och utan bias-kompensering.
- Simulering ...
  - SIM1: ...utan modellfel,
  - SIM2: ...med modellfel i dynamik- och accelerometermodellen samt bias,
  - SIM3: ...med endast modellfel i accelerometermodellen samt bias.



- Olinjära skattningsmodellen och EKF med och utan bias-kompensering.
- Simulering ...
  - SIM1: ... utan modellfel,
  - SIM2: ... med modellfel i dynamik- och accelerometermodellen samt bias,
  - SIM3: ... med endast modellfel i accelerometermodellen samt bias.
- Mätbrusets kovariansmatrix  $R$  från mätdata.
- Processbrusets kovariansmatrix  $Q$  från ett optimeringsproblem.

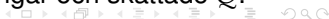
$$e_k = \sqrt{(x_k - \hat{x}_k)^2 + (z_k - \hat{z}_k)^2}$$



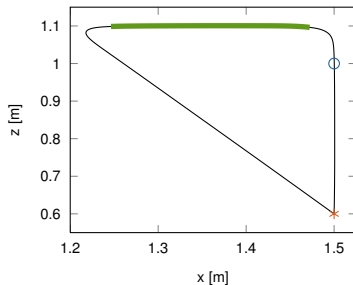
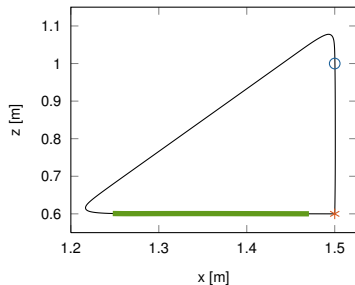
- Olinjära skattningsmodellen och EKF med och utan bias-kompensering.
- Simulering ...
  - SIM1: ... utan modellfel,
  - SIM2: ... med modellfel i dynamik- och accelerometermodellen samt bias,
  - SIM3: ... med endast modellfel i accelerometermodellen samt bias.
- Mätbrusets kovariansmatrix  $R$  från mätdata.
- Processbrusets kovariansmatrix  $Q$  från ett optimeringsproblem.

$$e_k = \sqrt{(x_k - \hat{x}_k)^2 + (z_k - \hat{z}_k)^2}$$

- Optimerar  $Q$  för alla 3 simuleringar med och utan bias-kompensering.
- Utvärdera olika kombinationer av simuleringar och skattade  $Q$ .

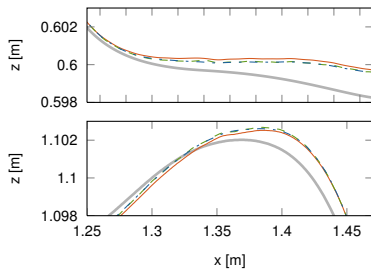


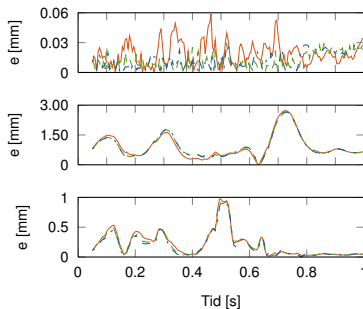
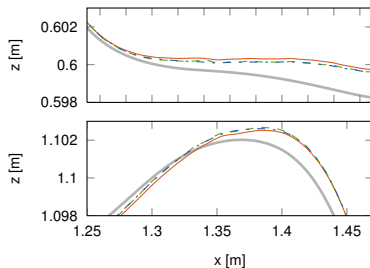
## Banor



- Grön del: Används för utvärdering.
- Röd stjärna: Startpunkten (medsols rörels).
- Blå cirkel: TCP då  $q_a = (0 \ 0)^T$ .

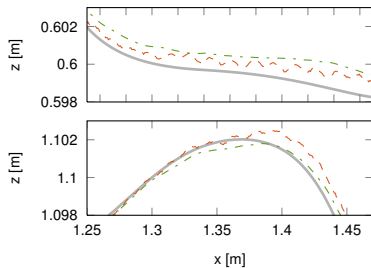


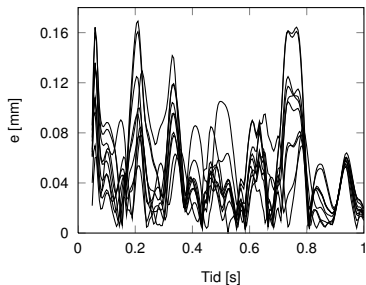
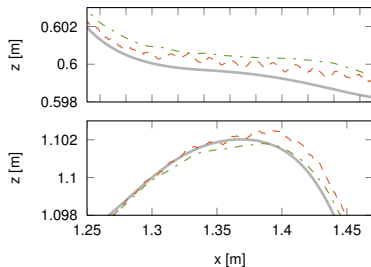




	COV1	COV2	COV3	COV4	COV5
SIM1	0.0221	0.0143	0.0143	—	—
SIM2	1.0967	1.0816	1.0913	1.1636	1.2755
SIM3	0.3053	0.2904	0.2767	0.1681	0.1267







För att skattningen ska bli bra krävs det

- små modellfel i den dynamiska modellen,
- bra val av processbrusets kovariansmatris,
- god information om position och orientering av accelerometern.



- Skatta  $Q$  med Expectation Maximisation (EM) algoritmen.

1. Välj ett startvärde  $Q_0 \in \mathcal{S}_+^{n_v}$  och sätt  $l = 0$ .



■ Skatta  $Q$  med Expectation Maximisation (EM) algoritmen.

1. Välj ett startvärde  $Q_0 \in \mathbb{S}_+^{n_v}$  och sätt  $l = 0$ .
2. Beräkna

$$M_i = (-J_{1,i-1} \quad I) P_{i|N}^{\xi,s} (-J_{1,i-1} \quad I)^T + \left( \hat{x}_{i|N}^s - F_1(\hat{x}_{i-1|N}^s) \right) \left( \hat{x}_{i|N}^s - F_1(\hat{x}_{i-1|N}^s) \right)^T$$

med hjälp av  $Q_l$ .



■ Skatta  $Q$  med Expectation Maximisation (EM) algoritmen.

1. Välj ett startvärde  $Q_0 \in \mathbb{S}_+^{n_v}$  och sätt  $l = 0$ .
2. Beräkna

$$M_i = (-J_{1,i-1} \quad I) P_{i|N}^{\xi,s} (-J_{1,i-1} \quad I)^T + \left( \hat{x}_{i|N}^s - F_1(\hat{x}_{i-1|N}^s) \right) \left( \hat{x}_{i|N}^s - F_1(\hat{x}_{i-1|N}^s) \right)^T$$

med hjälp av  $Q_l$ .

3. Beräkna uppdateringen av  $Q$  enligt

$$Q_{l+1} = \frac{1}{N-1} \sum_{i=2}^N F_2^+(\hat{x}_{i-1|N}^s) M_i \left( F_2^+(\hat{x}_{i-1|N}^s) \right)^T.$$





■ Skatta  $Q$  med Expectation Maximisation (EM) algoritmen.

1. Välj ett startvärde  $Q_0 \in \mathbb{S}_+^{n_v}$  och sätt  $l = 0$ .
2. Beräkna

$$M_i = (-J_{1,i-1} \quad I) P_{i|N}^{\xi,s} (-J_{1,i-1} \quad I)^T + (\hat{x}_{i|N}^s - F_1(\hat{x}_{i-1|N}^s)) (\hat{x}_{i|N}^s - F_1(\hat{x}_{i-1|N}^s))^T$$

med hjälp av  $Q_l$ .

3. Beräkna uppdateringen av  $Q$  enligt

$$Q_{l+1} = \frac{1}{N-1} \sum_{i=2}^N F_2^+(\hat{x}_{i-1|N}^s) M_i (F_2^+(\hat{x}_{i-1|N}^s))^T.$$

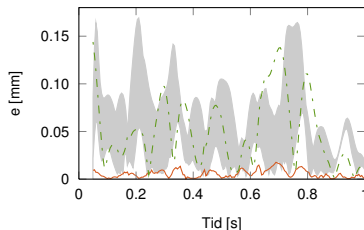
4. Stanna om  $|L_{Q_l}(y_{1:N}) - L_{Q_{l-m}}(y_{1:N})| \leq \gamma$ , annars sätt  $l = l + 1$  och gå till steg 2.

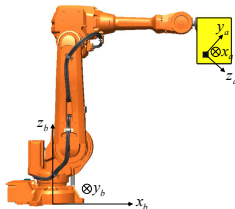


- Jämför med föregående metod (Alt. 1) samt en metod som beräknar  $Q$  direkt från modellen och glättade tillstånden (Alt. 2).
- Monte Carlo simuleringar med olika startvärden.



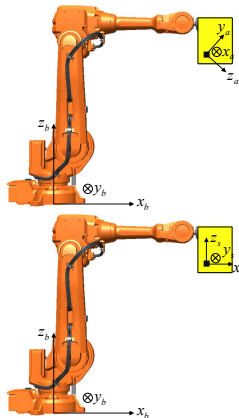
- Jämför med föregående metod (Alt. 1) samt en metod som beräknar  $Q$  direkt från modellen och glättade tillstånden (Alt. 2).
- Monte Carlo simuleringar med olika startvärden.
- Resultat:
  - EM algoritmen konvergerar till "samma" lösning för alla startvärden. Det samma gäller för Alt. 2.
  - EM algoritmen ger lägst skattningsfel.





- Söker en transformation

$$\rho_s = \kappa \mathcal{R}_{a/s} \rho_a + \rho_0.$$

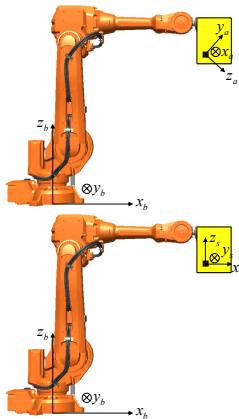


- Söker en transformation

$$\rho_s = \kappa \mathcal{R}_{a/s} \rho_a + \rho_0.$$

- Definiera residualerna

$$e_k = \rho_{s,k} - \kappa \mathcal{R}_{a/s} \rho_{a,k} - \rho_0.$$



- Söker en transformation

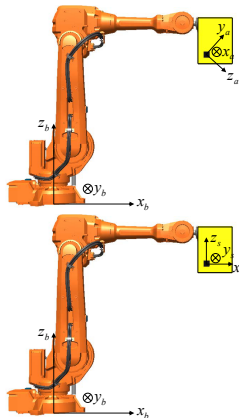
$$\rho_s = \kappa \mathcal{R}_{a/s} \rho_a + \rho_0.$$

- Definiera residualerna

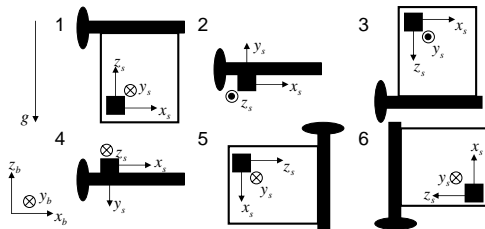
$$e_k = \rho_{s,k} - \kappa \mathcal{R}_{a/s} \rho_{a,k} - \rho_0.$$

- Optimeringsproblem:

$$\begin{aligned} &\text{minimise} && \sum_{k=1}^N \|e_k\|^2 \\ &\text{subject to} && \det(\mathcal{R}_{a/s}) = 1 \\ &&& \mathcal{R}_{a/s}^T = \mathcal{R}_{a/s}^{-1} \end{aligned}$$

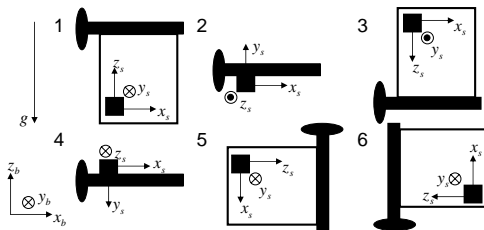


- Mät gravitationen i olika orienteringar.





- Mät gravitationen i olika orienteringar.



- Vet att  $\rho_s$  ska anta följande värden:

$$\begin{aligned} \rho_s^1 &= (0 \ 0 \ g)^T, & \rho_s^2 &= (0 \ g \ 0)^T, & \rho_s^3 &= (0 \ 0 \ -g)^T, \\ \rho_s^4 &= (0 \ -g \ 0)^T, & \rho_s^5 &= (-g \ 0 \ 0)^T, & \rho_s^6 &= (g \ 0 \ 0)^T. \end{aligned}$$



- Roterar axel 1 med konstant hastighet.
- Innebär att accelerationen pekar in mot rotationscentrum i ett plan ortogonalt mot gravitationen.

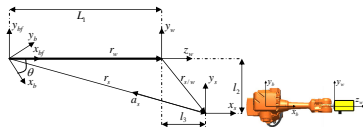
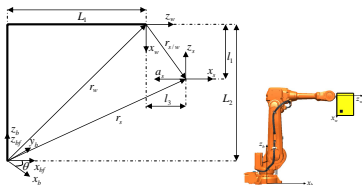


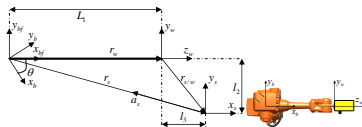
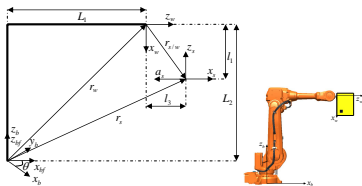
- Roter axel 1 med konstant hastighet.
- Innebär att accelerationen pekar in mot rotationscentrum i ett plan ortogonalt mot gravitationen.
- Beräkna ett analytiskt uttryck för accelerationen och jämför med mätningen.
- Det analytiska uttrycket beror på de okända positionsparametrarna.



- Roter axel 1 med konstant hastighet.
- Innebär att accelerationen pekar in mot rotationscentrum i ett plan ortogonalt mot gravitationen.
- Beräkna ett analytiskt uttryck för accelerationen och jämför med mätningen.
- Det analytiska uttrycket beror på de okända positionsparametrarna.
- Accelerometers orientering anses vara känd, d.v.s. transformtionen från  $Ox_a y_a z_a$  till  $Ox_s y_s z_s$  är känd.
- Orientera accelerometern så att gravitationen är riktad längs en av axlarna i  $Ox_s y_s z_s$ .

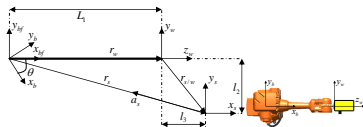
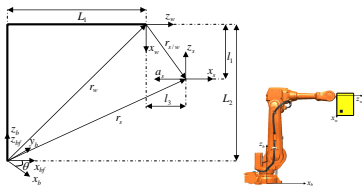






■ Accelerometers position ges av  $r_s$

$$[r_s]_b = [Q_{bf/b}]_b \left( [r_w]_{bf} + [r_{s/w}]_{bf} \right).$$



- Accelerometers position ges av  $r_s$

$$[r_s]_b = [Q_{bf/b}]_b \left( [r_w]_{bf} + [r_{s/w}]_{bf} \right).$$

- Derivering två gånger ger accelerationen

$$[a_s]_b = S(\omega)S(\omega) [Q_{bf/b}]_b \left( [r_w]_{bf} + [r_{s/w}]_{bf} \right).$$

- Jämförelse med mätningen från accelerometern ger

$$\begin{pmatrix} 0 & -\dot{\theta}^2 \\ \dot{\theta}^2 & 0 \end{pmatrix} \begin{pmatrix} l_2 \\ l_3 \end{pmatrix} = \begin{pmatrix} a_{s,x}^M + \dot{\theta}^2 L_1 \\ a_{s,y}^M \end{pmatrix}.$$





- Jämförelse med mätningen från accelerometern ger

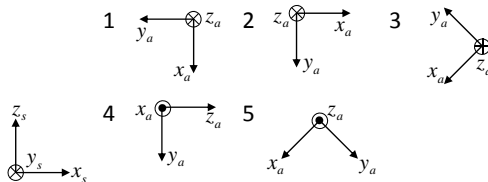
$$\begin{pmatrix} 0 & -\dot{\theta}^2 \\ \dot{\theta}^2 & 0 \end{pmatrix} \begin{pmatrix} l_2 \\ l_3 \end{pmatrix} = \begin{pmatrix} a_{s,x}^M + \dot{\theta}^2 L_1 \\ a_{s,y}^M \end{pmatrix}.$$

- Ytterligare två robotkonfigurationer ger

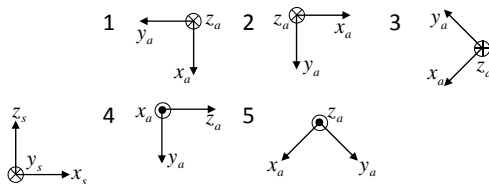
$$\underbrace{\begin{pmatrix} 0 & 0 & -\dot{\theta}_{c1}^2 \\ 0 & \dot{\theta}_{c1}^2 & 0 \\ \dot{\theta}_{c2}^2 & 0 & 0 \\ 0 & \dot{\theta}_{c2}^2 & 0 \\ 0 & 0 & -\dot{\theta}_{c3}^2 \\ \dot{\theta}_{c3}^2 & 0 & 0 \end{pmatrix}}_A \underbrace{\begin{pmatrix} l_1 \\ l_2 \\ l_3 \end{pmatrix}}_l = \underbrace{\begin{pmatrix} a_{s,x,c1}^M + \dot{\theta}_{c1}^2 L_1 \\ a_{s,y,c1}^M \\ a_{s,z,c2}^M + \dot{\theta}_{c2}^2 L_3 \\ a_{s,y,c2}^M \\ a_{s,x,c3}^M + \dot{\theta}_{c3}^2 L_1 \\ a_{s,z,c3}^M \end{pmatrix}}_b.$$



- Accelerometern monteras i 5 olika positioner och orienteringar.



- Accelerometern monteras i 5 olika positioner och orienteringar.



- Rotationsmatriserna ska likna ( $a, b, c, d \approx \cos(45^\circ) \approx 0.7071$ )

$$\mathcal{R}_{a/s}^1 = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{pmatrix}, \mathcal{R}_{a/s}^2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix}, \mathcal{R}_{a/s}^3 = \begin{pmatrix} -a^3 & -b^3 & 0 \\ 0 & 0 & 1 \\ -c^3 & d^3 & 0 \end{pmatrix}$$

$$\mathcal{R}_{a/s}^4 = \begin{pmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix}, \mathcal{R}_{a/s}^5 = \begin{pmatrix} -a^5 & b^5 & 0 \\ 0 & 0 & -1 \\ -c^5 & -d^5 & 0 \end{pmatrix}.$$



- M.h.a. kvarternioner kan rotationsskillnaden mellan “sann” och skattad rotationsmatris beräknas.

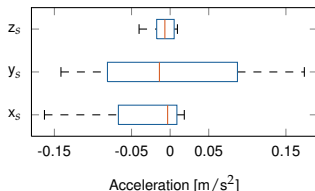
Test	1	2	3	4	5
$\vartheta$	$1.4^\circ$	$1.8^\circ$	$5.8^\circ$	$2.4^\circ$	$6.0^\circ$



- M.h.a. kvarternioner kan rotationsskillnaden mellan “sann” och skattad rotationsmatris beräknas.

Test	1	2	3	4	5
$\vartheta$	$1.4^\circ$	$1.8^\circ$	$5.8^\circ$	$2.4^\circ$	$6.0^\circ$

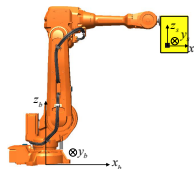
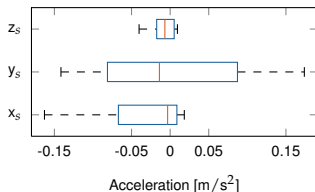
- Transformera mätningarna och jämför med vad man vill ha.



- M.h.a. kvarternioner kan rotationsskillnaden mellan “sann” och skattad rotationsmatris beräknas.

Test	1	2	3	4	5
$\vartheta$	$1.4^\circ$	$1.8^\circ$	$5.8^\circ$	$2.4^\circ$	$6.0^\circ$

- Transformera mätningarna och jämför med vad man vill ha.



■ Skattad position.

Test	Est. pos. ( $\hat{l}$ ) [cm]	$\Delta = \hat{l} - l^M$ [cm]	Std. for $\hat{l}$ [cm]
1	$(35.2 \quad 6.3 \quad 15.5)^T$	$(0.2 \quad 2.3 \quad -1.0)^T$	$(0.4 \quad 0.5 \quad 0.5)^T$
2	$(14.2 \quad 5.8 \quad 16.9)^T$	$(-0.3 \quad -1.2 \quad 1.8)^T$	$(0.3 \quad 0.3 \quad 0.3)^T$
3	$(36.3 \quad 6.3 \quad 21.4)^T$	$(-1.7 \quad 2.3 \quad -1.6)^T$	$(0.5 \quad 0.6 \quad 0.6)^T$
4	$(29.2 \quad 1.6 \quad 5.9)^T$	$(2.2 \quad 1.6 \quad 0.4)^T$	$(0.4 \quad 0.4 \quad 0.4)^T$
5	$(34.8 \quad -3.9 \quad 16.5)^T$	$(-0.7 \quad 0.1 \quad 1.0)^T$	$(0.5 \quad 0.6 \quad 0.6)^T$



- Experiment utförda på ABB IRB4600.
- Endast axel 2 och 3 samt påverkan från handleden minimerad.

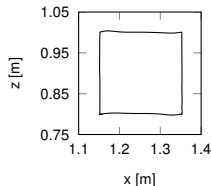




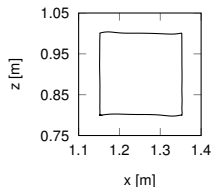
- Experiment utförda på ABB IRB4600.
- Endast axel 2 och 3 samt påverkan från handleden minimerad.
- Sann bana uppmätt med ett lasersystem från Leica Geosystems



- Experiment utförda på ABB IRB4600.
- Endast axel 2 och 3 samt påverkan från handleden minimerad.
- Sann bana uppmätt med ett lasersystem från Leica Geosystems



- Experiment utförda på ABB IRB4600.
- Endast axel 2 och 3 samt påverkan från handleden minimerad.
- Sann bana uppmätt med ett lasersystem från Leica Geosystems
- Den sanna banan ej synkroniserad med de andra mätningarna  
⇒ Fel kan införas vid valideringen.
- Osäkerhet i programmerad TCP ⇒ Skattad och mätt position justeras för att minimera statistiska fel.



## ■ 6 observatörer

OBS1: EKF med den olinjära modellen.

OBS2: EKS med den olinjära modellen.



## ■ 6 observatörer

OBS1: EKF med den olinjära modellen.

OBS2: EKS med den olinjära modellen.

OBS3: EKF med linjär dynamikmodell.

OBS4: PF med linjär dynamikmodell.



## ■ 6 observatörer

OBS1: EKF med den olinjära modellen.

OBS2: EKS med den olinjära modellen.

OBS3: EKF med linjär dynamikmodell.

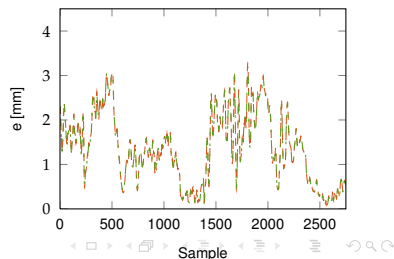
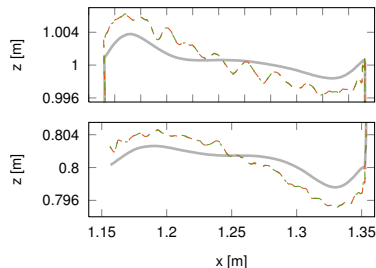
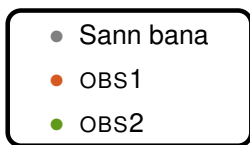
OBS4: PF med linjär dynamikmodell.

OBS5: EKF med olinjär modell där accelerationen används som insignal.

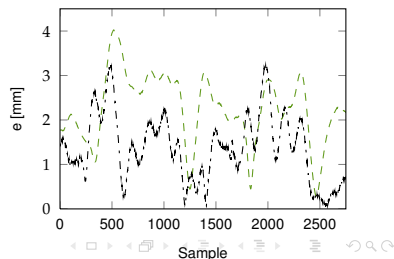
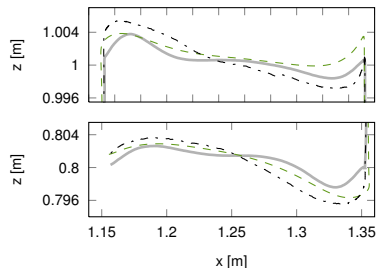
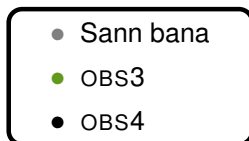
OBS6: Linjär dynamisk observatör m.h.a. polplacering med linjär modell där accelerationen används som insignal.



- Ger samma skattning.
- Svänger med högre frekvens än den sanna banan.
- Biastillstånden påverkar inte.
- Nuvarande MATLAB-implementering ej i realtid för någon av observatörerna.

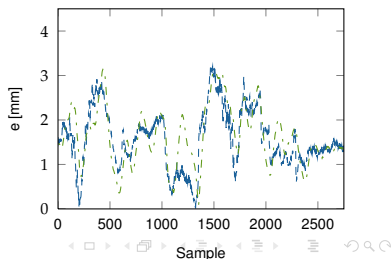
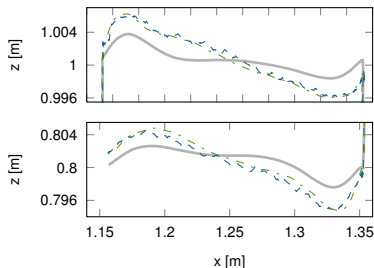
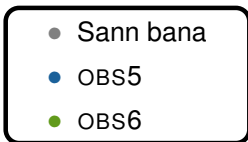


- Båda följer den sanna banan.
- OBS3 går förbi i hörnen.
- Bias-tillstånd för både motor- och accelerometer-mätningen krävs för att få bra resultat.
- Nuvarande MATLAB-implementering ej i realtid för någon av observatörerna.





- Observatören med linjära modellen fungerar bra.
- Observatören med olinjära modellen svänger med högre frekvens än den sanna banan.
- Ingen kompensering för bias.
- Nuvarande MATLAB-implementering i realtid för båda observatörerna.



1. Introduktion
2. Modelling
3. Resultat
4. Sammanfattning och framtida arbete



- Skattning av robotens TCP m.h.a. Bayesianska metoder och en accelerometer.
- Metod för att skatta orienteringen och positionen för accelerometern på roboten har presenterats.
- Skattning av processbrusets kovariansmatris m.h.a. EM-algoritmen.
- Experimentell utvärdering av 6 olika observatörer på en ABB IRB4600.



- Diskretisering av kontinuerlig tillståndsmodell.
- Känslighetsanalys av modellparametrar.
- Tidsberoende kovariansmatriser för att klara av kraftiga ändringar i banan.
- Utöka observatörerna till 6 DOF. Fler sensorer?
- Reglering från skattad position.



Tack för att ni lyssnade!

