

Identifiering av fysiska parametrar för en AGV, (autonomous guided vehicle)

Examensarbete i matematik, utfört på uppdrag av
Danaher Motion, Särö
av

Helena Johansson,
Industrimatematik

och

Mirsada Hromic,
Tillämpad matematik

vid
Göteborgs Universitet

Handledare: **Thomas Ericsson**, Matematiska Institutionen, Chalmers/GU
 Ulf Larsson, Danaher Motion, Särö

Examinator: **Thomas Ericsson**

14 september 2004

Sammanfattning

Danaher Motion utvecklar styrsystem till förarlösa truckar. Dessa program använder sig av data som erhålls från de hjul som styr och driver trucken, samt från en laserscanner monterad på trucken. Laserscannern mäter vinklar till reflektorer med kända positioner i omgivningen. Med hjälp av matematiska relationer beräknar programmet truckens position under tiden den kör. I dessa matematiska relationer ingår termer som beror på hjulens och laserscannerns monteringspositioner på trucken, dessa kallas fysiska parametrar. Eftersom dessa positioner är svåra att bestämma exakt, kan det leda till att de av styrsystemen beräknade positionerna avviker från de verkliga. Detta vill man naturligtvis förhindra genom att skatta de fysiska parametrarna för hjulen och laserscannern så bra som möjligt. Det är också önskvärt att denna skattning kan ske automatiskt.

Idag finns redan ett program som automatiskt skattar de fysiska parametrarna för truckar med ett styr- och drivhjul. I detta examensarbete studeras truckar som har två styr- och drivhjul. Dessa två hjul kan styras oberoende av varandra och till skillnad från truckar med bara ett styr- och drivhjul tillåter en sådan hjulkonfiguration rörelser i alla riktningar, t.ex. i sidled. Styrning för denna typ av truck modelleras i MATLAB. Med hjälp av modellen undersöks om det är möjligt att skatta de fysiska parametrarna. En målfunktion bildas genom att jämföra de av laserscanner uppmätta vinklarna med de från modellen beräknade. Målfunktionen optimeras för att undersöka hur bra parametrarna går att skatta. Arbetet resulterar i utarbetandet av en metod för hur dessa parametrar bestäms på ett godtagbart sätt innan trucken tas i bruk. Metoden verifieras genom testkörningar, vilka ger ett bra resultat. Metoden ställer rimliga krav på hur mycket monteringen av hjul och laserscanner får avvika från ritningarna och den är relativt enkel att utföra.

Abstract

Danaher Motion develops control system for autonomous guided vehicles. These programmes use data from the wheels that steers and drives the vehicle, together with data from a laser scanner mounted on the vehicle. The laser scanner measures angles to reflectors, with known positions, in its surroundings. Using mathematical relations, the programmes updates the position of the vehicle while it is moving. These mathematical relations include some terms, which depend on the mounting positions of the wheels and the laser scanner, these are called physical parameters. Since these positions are hard to determine exactly, the calculated positions from the control systems could differ from the true positions. This is of course something one wants to avoid by calibrating the physical parameters as well as possible. It is also desirable that the calibration could be done automatically.

Today there already exist a program for automatic calibration of the physical parameters for a vehicle with one steer and drive wheel. In this thesis work a type of vehicle with two steer and drive wheels is studied. These two wheels could be steered independent of each other and in contrast to vehicles with only one steering wheel, this wheel configuration allows movements in all directions, for example laterally. The navigation for this type of vehicle is modelled in MATLAB. Using this model the possibility of estimating the physical parameters is examined. By taking the difference between the measured angles from the scanner and the ones calculated using the model, an objective function is created. The objective function is optimised to examine how accurate the calibration of the parameters could become. The work results in a relative simple method that describes how to calibrate the physical parameters before beginning using the vehicle. The method is verified by some tests, which gives good results. The method has reasonable demands on the allowed divergence for the mounted positions for the wheels and the laser scanner, from the positions on the drawings.

Förord

Detta examensarbete har utförts på uppdrag av Danaher Motion, Särö och är ett slutresultat av våra utbildningar vid Göteborgs Universitet. Helena har följt det Industrimatematiska programmet och Mirsada har inriktat sig på tillämpad matematik. Arbetet har utförts under tiden mars till augusti 2004, med största delen förlagd på Danaher Motion i Särö under regelbunden kontakt med Universitetet. Stort tack till vår handledare, Ulf Larsson, på Danaher Motion samt till Thomas Ericsson, vår handledare vid Göteborgs Universitet. Mirsada vill ge ett jättetack till sin man som varit ett stort stöd, samt till föräldrarna för all hjälp de gett under utbildningstiden. Helena riktar ett enormt tack till sina föräldrar som ställt upp på alla sätt, vilket har gjort det möjligt att genomföra utbildningen.

Göteborgs Universitet, September 2004

Helena Johansson & Mirsada Hromic

Symbollista

Beteckning	Variabel
X	laserscannerns globala x-koordinat
Y	laserscannerns globala y-koordinat
θ	laserscannerns, (och vagnens), riktning relativt den globala x-axeln
X_b	x-koordinat för reflektor b, (global)
Y_b	y-koordinat för reflektor b, (global)
β_b	uppmätt vinkel av laserscannern till reflektor b
γ_b	beräknad vinkel till reflektor b
X_F	laserscannerns faktiska globala x-koordinat
Y_F	laserscannerns faktiska globala y-koordinat
θ_F	laserscannerns faktiska riktning relativt global x-axel
x_s	laserscannerns lokala x-koordinat på vagnen
y_s	laserscannerns lokala y-koordinat på vagnen
θ_s	vinkel mellan laserscannerns nolläge och den lokala x-axeln
v_1	hastighet från encoder för hjul 1
v_2	hastighet från encoder för hjul 2
v_x	laserscannerns hastighet i lokalt x-led relativt globalt koord.system
v_{x_1}	laserscannerns hastighetskomponent i x-led, beräknad m.a.p encodervärden från hjul 1
v_{x_2}	laserscannerns hastighetskomponent i x-led, beräknad m.a.p encodervärden från hjul 2
v_y	laserscannerns hastighet i lokalt y-led relativt globalt koord.system
v_{y_1}	laserscannerns hastighetskomponent i y-led, beräknad m.a.p encodervärden från hjul 1
v_{y_2}	laserscannerns hastighetskomponent i y-led, beräknad m.a.p encodervärden från hjul 2
ω	laserscannerns, (och vagnens), rotationshastighet i det globala koordinatsystemet
v_X	laserscannerns hastighetskomponent i globalt x-led
v_Y	laserscannerns hastighetskomponent i globalt y-led
u_1	styrvinkel för hjul 1
u_2	styrvinkel för hjul 2
d_1	skalfel för v_1
d_2	skalfel för v_2

Beteckning	Variabel
α_1	vinkelfel för u_1
α_2	vinkelfel för u_2
L	avstånd mellan hjulen
p	parametervektor, $p = [\alpha_1, \alpha_2, d_1, d_2, L, x_s, y_s, \theta_s]$
$r(p)$	residualfunktionen, $(\beta - \gamma(p))$, vektorvärd
φ	vridningsvinkel för vagnen och det lokala koordinatsystemet
\mathbf{X}	tillståndsvektorn, $[X, Y, \theta]^T$
\mathbf{e}	vektor med encodervärden, $[v_1, u_1, v_2, u_2]^T$
\mathbf{w}	vektor med brus p.g.a modelleringsfel i rörelsemallen
$\bar{\mathbf{X}}$	linjäriseringspunkt för rörelsemallen
$\delta \mathbf{X}$	avvikelsen från $\bar{\mathbf{X}}$
ν	vitt brus p.g.a fel i mätmodellen
$\hat{\mathbf{X}}$	skattad tillståndsvektor
$\hat{\gamma}$	skattad vinkel
G	matris som modellerar fel i dödräkningsmodellen
F	Jacobianen till rörelsemallen
H	gradient till mätmodellen
r_x	avstånd mellan X_b och X
r_y	avstånd mellan Y_b och Y
K	Kalmanfiltrets återkopplingsfaktorn
P	kovariansmatrisen för felet
R	kovariansen för mätbruset
Q	modell av kovariansmatrisen för modellbruset

Innehåll

1 Inledning	1
1.1 Bakgrund och syfte	1
1.2 Problemformulering	4
1.3 Målformulering	5
2 Matematisk modell	7
2.1 Rörelsemmodell	7
2.2 Mätmodell	10
3 Residualfunktion	12
3.1 Residualens utseende	12
3.2 Påverkan av vinkeloffset för styrvinklarna, α_1 respektive α_2 , på residualfunktionen	14
3.3 Påverkan av skalfel på hastigheterna, d_1 respektive d_2 , på residualfunktionen	17
3.4 Övriga parametrars påverkan på residualfunktionen	19
3.5 Sammanfattning av parametrarnas påverkan	22
4 Optimering	23
4.1 Optimering av den raka banan	23
4.2 Optimering av den kurviga banan	25
4.3 Utvärderingsprogram	26
5 Kalmanfilter	30
5.1 Linjärisering av rörelsemmodell	30
5.2 Linjärisering av mätmodellen	31
5.3 Filterekvationerna	31
5.4 Residualfunktionen utökad med filterekvationerna	32
5.5 Optimering av residualfunktionen med Kalmanfilter	34
6 Störningskänslighet med avseende på de olika parametrarna	36
6.1 Orsaker till vinkelparametrarnas injusteringar vid störning	37
7 Olika försök att komma ifrån injusteringen av vinkelparametrarna	39
7.1 Ändring av origo för det lokala koordinatsystemet	39
7.2 Ny avståndspараметer	39
7.3 Olika körmönster	41

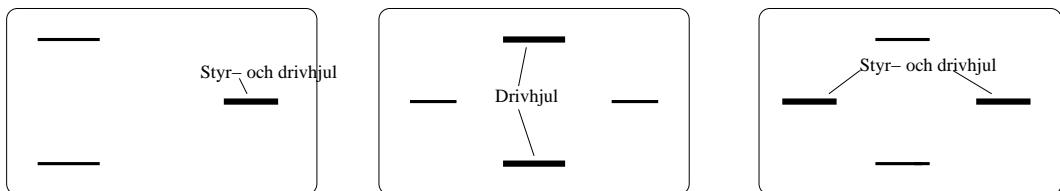
7.4	Lösning av en vinkeloffsetparameter	41
7.5	Efterjustering av parametrarna	42
7.6	Testkörningar	43
7.7	Störda parametrar i verkligheten	44
8	Resultat och Slutsatser	46
8.1	När är parametrarna observerbara och vilken påverkan har de	46
8.2	Optimering på olika sätt	47
8.3	Vilka parametrar går att skatta	49
8.4	Korrigering av parametrar	49
8.5	Testkörning	50
8.6	Metod för identifiering av parametrar	50

1 Inledning

1.1 Bakgrund och syfte

Danaher Motion, Särö är ett företag som bland annat utvecklar mjukvara som används för navigering av förarlösa truckar, vilka benämns AGV, (autonomous guided vehicle). Under utvecklingen används specialbyggda testvagnar för att undersöka hur navigeringen fungerar. En sådan testvagn används även i detta arbete, därför kommer ordet vagn häданefter användas i rapporten i stället för truck.

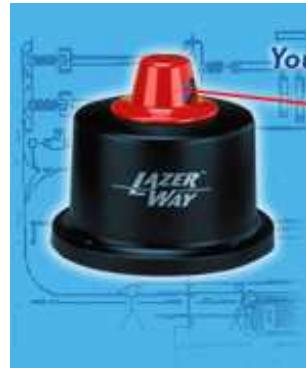
Det finns olika utföranden av dessa vagnar med avseende på antal hjul, hjulens placering på vagnen samt hjulens vridbarhet. Dels finns trehjuliga vagnar som har ett hjul i mitten fram och två hjul bak vid var kant. Här är det framhjulet som styr och driver vagnen och de två bakre hjulen fungerar som stödhjul. Sedan finns det fyrrhjuliga vagnar som har två hjul vid var kant på mitten av vagnen och ett hjul mitt fram samt ett hjul mitt bak. För en variant av dessa fyrrhjulingar är det de två hjulen på mitten av vagnen som är de drivande hjulen och en vridning av vagnen fås genom att låta dessa två hjul ha olika hastighet. D.v.s hjulen är inte vridbara relativt vagnen. Denna sort av fyrrhjuling kallas diff-vagn.



Figur 1. De olika vagnstyperna, t.v trehjuling, mitten diff-vagn, t.h quad-vagn

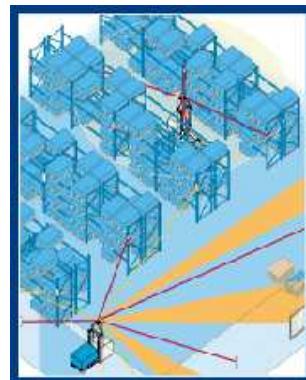
Den typ av vagn som detta arbete avser är en annan modell av fyrrhuling och denna typ kallas quad-vagn. Här är det hjulet mitt fram och det mitt bak som är de styrande och drivande hjulen, vagnen svänger alltså när hjulen vrider sig, vid sidorna mitt på vagnen sitter nu stödhjul.

Varje styr- och drivhjul, (oavsett vagnstyp), har en givare monterad som kallas encoder. Dessa encodrar ger med jämna mellanrum, (varje 50 ms), data angående tidpunkt och hjulens hastigheter samt styrvinklar vid denna tidpunkt. För varje testkörning kan dessa data sparas i en logfil. Vagnen har en känd startposition och använder de värden encodrarna ger ifrån sig för att beräkna och uppdatera sin position vid varje tidpunkt, vilket alltså sker 20 gånger per sekund.



Figur 2. Laserscanner.

Som hjälp för navigeringen används också en laserscanner som oftast är placerad överst på vagnen så att den har 360° synfält. Efter det att scannern har startats roterar den med en konstant hastighet vilken är satt till 38 rad/s, detta motsvarar ungefär 6 varv/s. Runt om i den miljö vagnen rör sig är reflektorer placerade, de liknar vanliga reflexer och kan vara platta eller runda. Miljön är indelad i ett koordinatsystem och reflektorernas position m.a.p detta system är kända. Under tiden vagnen rör sig mäter laserscannern vinklar till de reflektorer den träffar. Dessa vinkelmätningar kan vara av olika kvalitet bl.a beroende på om reflektorn är delvis dold eller beroende på med vilken infallsvinkel laserstrålen träffar reflektorn. Ibland kan det vara något annat föremål som reflekterar laserstrålen vilket då ger en felaktig mätning.



Figur 3. Laserscannern mäter till olika reflektorer i sin omgivning.

De data scannern ger kan, på samma sätt som för encodrarna, sparas i en logfil. Denna innehåller: tidpunkter då mätningarna gjordes, vilken vinkel laserscannern mätt vid respektive tidpunkt, samt vilket avstånd som är uppmätt till reflektorn som vinkelmätningen avser. Dessutom erhålls ett mått på kvaliteten för respektive mätning, samt med vilken infallsvinkel som laserscannern träffade aktuell reflektor.

Ytterligare data som kan sparas i en log-fil då vagnen kör är koordinater relativt omgivningen för en referenspunkt på vagnen. Denna punkt är, för en quad-vagn, definierad till att vara mitt emellan fram- och bakhjulet. Även tidpunkten då dessa koordinater gäller samt vilken riktning vagnen då har fås som data. Med vagnens riktning avses hur vagnen är vriden i förhållande till x-axeln i det koordinatsystem man indelat omgivningen i. Dessa tidpunkter som sparas för en körning är de samma som de tidpunkter avläsning av encodrarna sker.

Eftersom scannern roterar med konstant hastighet erhålls olika antal vinkelmätningar för varje intervall som encodrarna ger ifrån sig data, men endast en vinkelmätning används vid varje positionsuppdatering,(var 50 ms). Speciella associeringsalgoritmer avgör till vilken reflektor vinkelmätningen är gjord eller om den inte kommer från någon reflektor alls. Om ingen vinkel kunnat associeras för en tidpunkt används endast encodervärdena för att beräkna och uppdatera positionen, detta kallas då dödräkning.

Data från laserscannern och encodrarna innehåller vissa fel, detta beror bl.a på svårighet att exakt bestämma placering av hjul och scanner på vagnen. På grund av denna osäkerhet kan uppdateringen av positionen stämma sämre och sämre överens med verkligheten.

Fysiska parametrar som påverkar positionsbestämningen som sker enligt ovan är: laserscannerns placering på vagnen, hjulens montering d.v.s hur parallella är de med vagnen då hjulen inte svänger, hjulavstånd samt hastighetsskalningen. Detta examensarbete går ut på att:

- undersöka vilka av de fysiska parametrarna eller kombinationer av dessa som låter sig skattas på ett tillförlitligt sätt
- undersöka vilka vagnsrörelser som ska ingå i en testbana för att parametrarna ska vara observerbara
- avgöra vilka parametrar som kan skattas automatiskt och vilka en användare behöver hjälpa till med att skatta
- föreslå en metod för denna skattning, så att vissa villkor uppfylls.

De villkor som avses är dels att associeringsalgoritmen ska kunna associera tillräckligt många vinklar till reflektorerna, annars stannar vagnen, och dels att vagnen stannar i rätt position relativt t.ex en laststation.

Liknande arbeten har tidigare gjorts för trehjuliga AGV, se t.ex [4] och [12], där man nu använder ett automatiskt intrimningsprogram för att bestämma de parametrar som är relevanta för denna typ av vagn.

1.2 Problemformulering

Man vet att montering av laserscanner och hjul på vagnen inte kan göras exakt enligt ritningar. Detta gör att de matematiska relationer som används för vagnens navigering inte stämmer helt med verkligheten. Relationerna innehåller termer beroende på scannerns och hjulens positioner på vagnen samt indata från laserscannern och encodrarna. Om de antagna värdena för dessa positioner på vagnen, (dessa används vid uppdatering av vagnens position i den miljö den befinner sig), avviker för mycket mot de verkliga värdena leder detta till att vagnens verkliga position relativt omgivningen avviker mot den beräknade. Om denna avvikelse i sin tur blir för stor kan inga reflektorer associeras till vinkelmätningarna och vagnen kommer då av säkerhetsskäl att stanna.

Ett sätt att angripa problemet är att låta vagnen köra olika testbanor och logga data som fås från laserscannern, hjulen och referenspunkten. Efteråt kan man då modellera hur vagnen rört sig m.h.a de relationer som gäller mellan hjulens hastigheter och styrvinklar. Detta ger en diskret modell eftersom indata som används för uppdatering av positionen fås med jämma mellanrum. Detta är det sätt som används här och modelleringen görs i MATLAB, [1], vilket är det program som huvudsakligen kommer att användas för modellering och beräkning. Efter det att man har en modell för hur vagnen rört sig samt koordinater för reflektorerna kan man m.h.a relationer mellan vinklar och avstånd modellera och beräkna vilken vinkel laserscannern borde mäta till de olika reflektorerna vid varje tidpunkt.

Då man har koordinater för referenspunkten samt riktning på vagnen i varje tidpunkt, kan man ganska enkelt bestämma till vilken reflektor laserscannern har mätt den vinkel som ligger närmast i tid. På detta sätt kan den vinkel som mäts av laserscannern jämföras med den som beräknas utifrån vagnens modellerade rörelse vid varje tidpunkt. Skillnaderna mellan dessa vinklar sparas i en vektor där varje element är skillnaden vid denna tidpunkt.

Funktionen för att beräkna denna vektor kallas residualfunktion, (ibland bara residual). På grund av osäkerheten i bl.a montering av hjul och scanner beror denna funktion på parametrar som anger vinkeloffset för hjulen, d.v.s hur mycket hjulen är vridna då vridningsvinkel antas vara noll, hastighetsskalningen för hjulen, hjulavstånd, laserscannerns placering lokalt på vagnen samt vinkeln mellan laserns nolläge för avläsning av vinklar och den vridning vagnen har relativt sitt omgivande koordinatsystem. Denna residualfunktion är den som i detta arbete kommer att undersökas med avseende på de ingående parametrarna.

Genom att logga data från testkörningar med olika körmönster kan residualfunktionen undersökas m.a.p var och en av parametrarna. Dessa un-

dersökningar kan då ge information om hur en viss parameter påverkar funktionen för de olika körmönstren och hur stor denna påverkan är. Olika körmönster kan t.ex vara en rak körning utan svängar, enbart cirklar, eller svängar och raksträckor blandat. Om två, (eller flera), parametrar har likartad påverkan på residualfunktionen benämns detta att det är ett beroende mellan dessa parametrar. Med likartad påverkan avses t.ex. att samma påverkan på funktionen kan fås genom att lägga till ett litet värde till den ena parametern, eller genom att dra bort samma värde från den andra parametern.

Utifrån dessa undersökningar ska det anges vilka parametrar som går att bestämma samt vilket körmönster vagnen behöver köra för att parametrarna ska vara observerbara. Parametrarna ska sedan bestämmas så att residualfunktionen minimeras m.a.p dessa, d.v.s de beräknade vinkelskillnaderna i residualen ska vara så små som möjligt.

Olika metoder att bestämma parametrarna kan vara att först köra en slags bana och optimera residualfunktionen m.a.p en delmängd av parametrarna som kan observeras för detta körmönster. Därefter körs efter ett annat mönster med de redan bestämda parametrarna som konstanter och residualfunktionen optimeras m.a.p resten av parametrarna. Denna metod kan göra så att vissa beroenden isoleras och inte påverkar optimeringen, men samtidigt förloras information då man optimerar i olika etapper. En annan metod är att låsa någon av de parametrar det eventuellt finns ett beroende mellan och köra en bana där alla övriga parametrar är observerbara och därefter optimera residualfunktionen m.a.p dessa. Fördelen är att man inte delar upp optimeringen. Nackdelen däremot är att ett eventuellt fel i den låsta parametern kan påverka det optimum som fås. Om man t.ex låser en av parametrarna för styrvinklarna är det troligt att nolläget för styrvinkelns avviker från det sanna nolläget. Vet man då hur detta fel skulle kunna påverka övriga parametrars värden kan kanske en efterjustering av dessa värden få bort detta fel.

Idag görs en intrimning av en quad-vagn manuellt enligt vissa regler, [10]. Önskvärt är att ta fram en metod för automatisk intrimning liknande de som redan finns för trehjuliga AGV, t.ex. [15].

1.3 Målformulering

Målet är att ta fram en metod, (helst helautomatisk), för intrimning av quad-vagnar innan de tas i bruk, så att vissa villkor är uppfyllda. Ett första villkor är att elementen i vektorn som ges av residualfunktionen ska vara så små som möjligt. Ett annat villkor är att associeringen av uppmätta vinklar till kända reflektorer lyckas och ytterligare ett är att vagnen stannar i rätt po-

sition relativt t.ex en lastningsstation. För att jämföra olika optimeringar av residualfunktionen för de olika körmönstren beräknas standardavvikelsen. Denna vill man då ha så liten som möjligt.

En bra intrimmad vagn följer den banan den är programmerad att åka enligt, den går rakt, d.v.s hjulen är inte vridna nämnvärt relativt vagnen då den inte svänger, och vagnen ska stanna i rätt position.

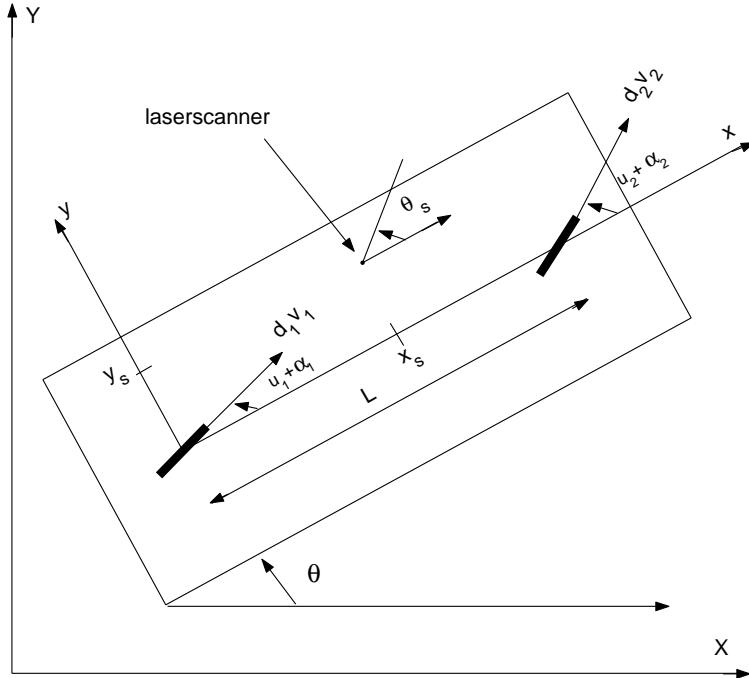
2 Matematisk modell

Den matematiska modellen består av två delar, rörelsemallen som beskriver vagnens rörelse samt mätmodellen som beskriver hur lasern mäter vinklarna.

Ett globalt och ett lokalt koordinatsystem införs, med det globala koordinatsystemet avses det plan med fasta koordinater som golvet utgör med origo bestämt i något hörn. Det lokala koordinatsystemet gäller, som ordet antyder, lokalt på vagnen med origo placerat i bakhjulets mitt och x-axeln i vagnsriktningen, se figur 4. Detta lokala koordinatsystem roterar alltså relativt det globala när vagnen svänger. Koordinater i det globala system betecknas med versaler och med gemener avses koordinater i det lokala systemet.

2.1 Rörelsemall

Med hjälp av rörelsemallen vill man beräkna var laserscannern befinner sig globalt och även vilken riktning scannern har då den är i sitt nolläge, detta läge antas följa den lokala x-axelns riktning, se figur 4. Scannerns riktning är då den samma som vagnens riktning relativt det globala koordinatsystemet. Positionen anges med koordinaterna (X, Y) relativt det plan vagnen rör sig på. Riktningen eller som det också kallas orienteringen på laserscannern anges av vinkeln θ relativt den globala X-axeln. Detta kommer häданefter betecknas av trippeln (X, Y, θ) och benämns laserscannerns position.



Figur 4. Samband mellan laserscannerns hastigheter.

Vagnen rör sig över golvet och har laserscannern monterad i punkten med de lokala koordinaterna (x_s, y_s) . För varje styr- och drivhjul fåras från encodrar data som beskriver hjulens hastigheter, v_1 och v_2 . Encodrarna ger också riktningar för varje styr- och drivhjul, u_1 och u_2 , som data, dessa benämns även styrvinklar. Alla värden från encodrarna är relativt det lokala koordinatsystemet och dessa värden fåras med jämna tidsintervall. Varje tidpunkt numreras med k som går från 1 till m , där 1 står för den första tidpunkten som data sparas för en körning och m står för den sista tidpunkten. Alla encodervärden beror alltså på tidpunkten k men detta k utelämnas häданefter så att $v_1 = v_1(k)$ och p.s.s för övriga data. Hastigheterna antas ha ett skalfel d_i och styrvinklarna ett vinkeloffset α_i , dessa är skalärer och förändras ej med tiden.

Laserscannerns hastigheter som beräknas med avseende på encodervärden är v_x och v_y . v_x betecknar hastigheten i vagnens x-riktning relativt det globala koordinatsystemet och v_y är hastigheten i vagnens y-riktning relativt det globala koordinatsystemet. Scannerns hastighetskomponenter v_x och v_y kan beräknas med hjälp av indata från båda styr- och drivhjulen och därfor betecknas v_x som beror på bakhjulet med v_{x_1} och v_x som beror på framhjulet betecknas v_{x_2} . På samma sättas indexeras hastighetskomponenten i y-led. Då vagnen rör sig på en relativt plan yta och hjulen inte slirar beskrivs ett idealt

samband mellan indata och laserscannerns hastighet av följande ekvationer:

$$\begin{aligned} v_{x_1} &= d_1 v_1 \cos(u_1 + \alpha_1) - y_s \omega \\ v_{y_1} &= d_1 v_1 \sin(u_1 + \alpha_1) + x_s \omega \\ v_{x_2} &= d_2 v_2 \cos(u_2 + \alpha_2) - y_s \omega \\ v_{y_2} &= d_2 v_2 \sin(u_2 + \alpha_2) + (x_s - L) \omega \end{aligned} \quad (1)$$

Eftersom vagnen är en stel kropp, d.v.s inte deformbar måste villkoren $v_{x_1} = v_{x_2}$ och $v_{y_1} = v_{y_2}$ uppfyllas.

Detta, (1), är ett överbestämt ekvationssystem som man vill lösa för v_x , v_y och ω . ω är laserscannerns, och vagnens, rotationshastighet, d.v.s förändringen av scannerns orientering relativt det globala koordinatsystemet m.a.p tiden. För att inte indata från ett hjul ska få större inflytande än från det andra, väljs att använda medelhastigheten av dessa hastighetsekvationer, (1), i beräkningarna av laserscannerns hastigheter i vagnens x- och y-led, dvs:

$$\begin{aligned} v_x &= \frac{v_{x_1} + v_{x_2}}{2} \\ v_y &= \frac{v_{y_1} + v_{y_2}}{2} \end{aligned} \quad (2)$$

Det finns även andra sätt att beräkna dessa hastigheter på, t.ex som ett minstakvadratproblem, men det prövas inte här.

ω får från villkoret $v_{y_1} - v_{y_2} = 0$ vilket ger:

$$\omega = \frac{d_2 v_2}{L} \sin(u_2 + \alpha_2) - \frac{d_1 v_1}{L} \sin(u_1 + \alpha_1) \quad (3)$$

Dessa hastigheter, (v_x, v_y) , transformeras till globala hastigheter, (v_X, v_Y) , d.v.s hastigheter i globalt x- och y-led. De globala hastigheterna för laser-scannern blir:

$$\begin{aligned} v_X &= v_x \cos \theta - v_y \sin \theta \\ v_Y &= v_x \sin \theta + v_y \cos \theta \end{aligned} \quad (4)$$

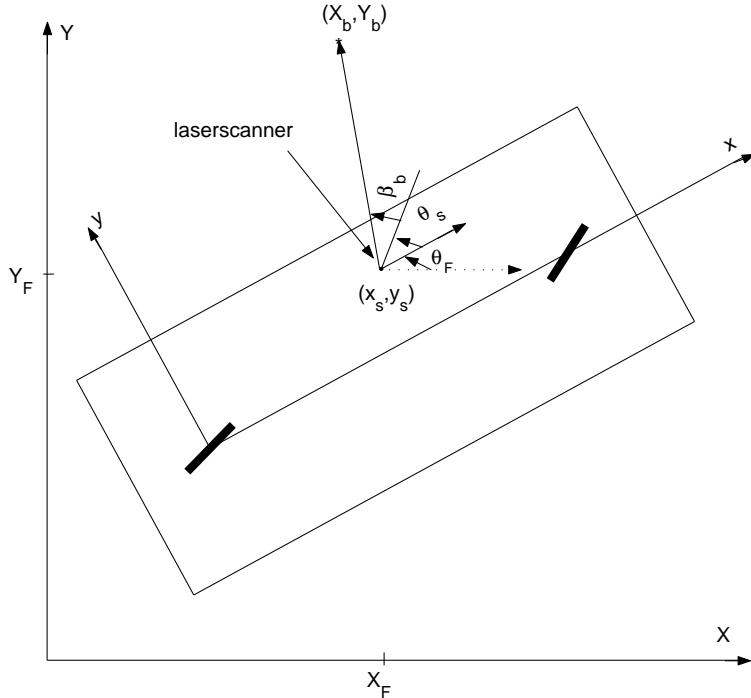
där θ ges från formel (5), nedan. Laserscannerns hastigheter, (v_X, v_Y, ω) , integreras enligt Eulers metod för att få laserscannerns globala position i rummet (X, Y, θ) . Då man antar att hastigheter och styrvinkelar som får som indata är konstanta under tidsintervallet $\delta_T = t_{k+1} - t_k$, samt att förändring i riktning är liten under samma intervall får följande diskreta ickelinjära rörelsemodell:

$$\begin{aligned} X(k+1) &= X(k) + \delta_T (v_x \cos \theta(k) - v_y \sin \theta(k)) \\ Y(k+1) &= Y(k) + \delta_T (v_x \sin \theta(k) + v_y \cos \theta(k)) \\ \theta(k+1) &= \theta(k) + \delta_T \omega \end{aligned} \quad (5)$$

$X(k)$ betecknar värdet av X vid tidpunkt k och på samma sätt för $Y(k)$ och $\theta(k)$. Hädanefter kommer k ofta utelämnas för att texten ska bli mer lättläst och formlerna inte ska bli för långa.

2.2 Mätmodell

Ett antal reflektorer, (n st), finns i rummet med kända positioner (X_b, Y_b) , $b = 1, \dots, n$. Laserscannern är placerad på vagnen med de lokala koordinaterna (x_s, y_s) och vinkeloffset θ_s i förhållande till den lokala x-axeln, (jmf. med figur 4). Vinkeloffset är den vinkel som scannerns nolläge avviker med från det antagna nolläget. Detta är enligt definition parallell med den lokala x-axeln, vilket illustreras i figur 5, där den streckade linjen är parallell med den globala x-axeln.



Figur 5. Laserscannerns mätningar till kända reflektorer.

Laserscannern roterar med konstant hastighet och mäter vinklar β , till de givna reflektorerna. Vinklarna mäts från laserns faktiska globala position, (X_F, Y_F, θ_F) :

$$\beta = \arctan \left(\frac{Y_b - Y_F}{X_b - X_F} \right) - (\theta_F + \theta_s) \quad (6)$$

Genom beräkningar enligt rörelsemodellen approximeras nu laserscannerns globala position (X, Y, θ) för varje tidpunkt k . Eftersom modellen, (5),

inte är exakt på grund av bl.a osäkerhet i indata, ger detta ett fel i positionsbestämningen. Detta kan ses om man jämför den beräknade positionen (X, Y, θ) med den faktiska positionen (X_F, Y_F, θ_F) .

Efter att de globala koordinaterna (X, Y, θ) approximerats, beräknas vilken vinkel γ , som laserscannern mäter till reflektorerna utifrån dessa koordinater. Detta ger följande mätmodell:

$$\gamma = \arctan\left(\frac{Y_b - Y}{X_b - X}\right) - (\theta + \theta_s) \quad (7)$$

Dessa två modeller, (5) och (7), används för att beräkna laserscannerns rörelse mellan reflektormätningarna samt vilken vinkel scannern mäter till en reflektor med känd position. Modellerna innehåller alltså ett antal okända parametrar och dessa är:

α_1 och α_2 , vinkelfel för styrvinklarna,

d_1 och d_2 , skalfel på hastigheterna,

L , avstånd mellan hjulen,

x_s, y_s , lasercannerns position lokalt på vagnen och

θ_s , scannerns avvikelse relativt den lokala x axeln.

Dessa parametrar samlas i en parametervektor som betecknas p , där: $p_1 = \alpha_1$, $p_2 = \alpha_2$, $p_3 = d_1$, $p_4 = d_2$, $p_5 = L$, $p_6 = x_s$, $p_7 = y_s$ och $p_8 = \theta_s$

3 Residualfunktion

Residualfunktionen, r , (ofta kallad residualen), ger skillnaden mellan de uppmätta vinklarna, β , (6), och de beräknade vinklarna, γ , (7), för varje tidpunkt k så att: $(r(p))(k) = (\beta - \gamma(p))(k)$. De uppmätta vinklarna till de olika reflektorerna är data från laserscannern som roterar med konstant hastighet. För varje tidsintervall som data returneras från encodrarna, erhålls ett varierande antal vinkelmätningar från laserscannern. Genom beräkningar matchas vinkelmätningarna till reflektorerna, d.v.s man avgör vilken reflektor respektive vinkel borde vara uppmätt till.

Den vinkel som används för beräkning av residualfunktionen i tidpunkt k är den som en reflektor kunnat matchats till och som ligger närmast i tid med indata från encodrarna. Element k i residualen blir då vinkelskillnaden av den uppmätta vinkeln och den beräknade vinkeln till aktuell reflektor i tidpunkt k . Indata från encodrarna kommer alltså med jämna tidsintervall och för varje tidpunkt görs en positionsberäkning samt en vinkelberäkning och residualfunktionen växer då med ett element. Antal element i residualfunktionen kommer därför att bero på hur länge vagnen körs.

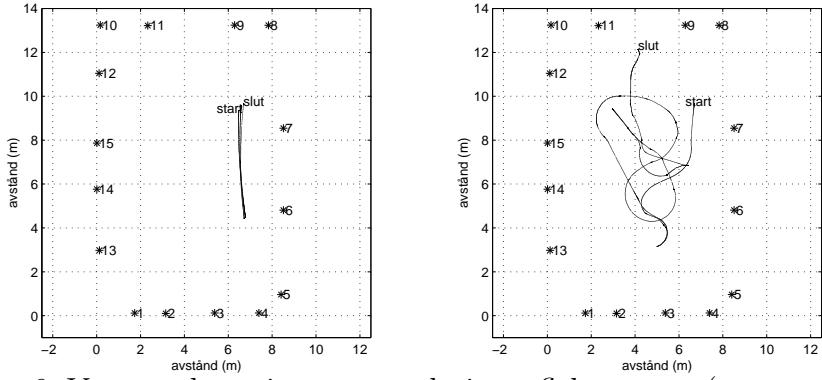
Kvadratsumman av denna residual är målfunktionen som väljs att minimera m.a.p de okända parametrarna.

$$\min_p \sum_{k=1}^n ((\beta - \gamma(p))(k))^2 \quad (8)$$

Att undersöka residualfunktionen istället för målfunktionen m.a.p de olika parametrarna kan ge mer information om hur de olika parametrarna påverkar var och en. Därför är det residualfunktionen som undersöks grundligt och inte målfunktionen direkt.

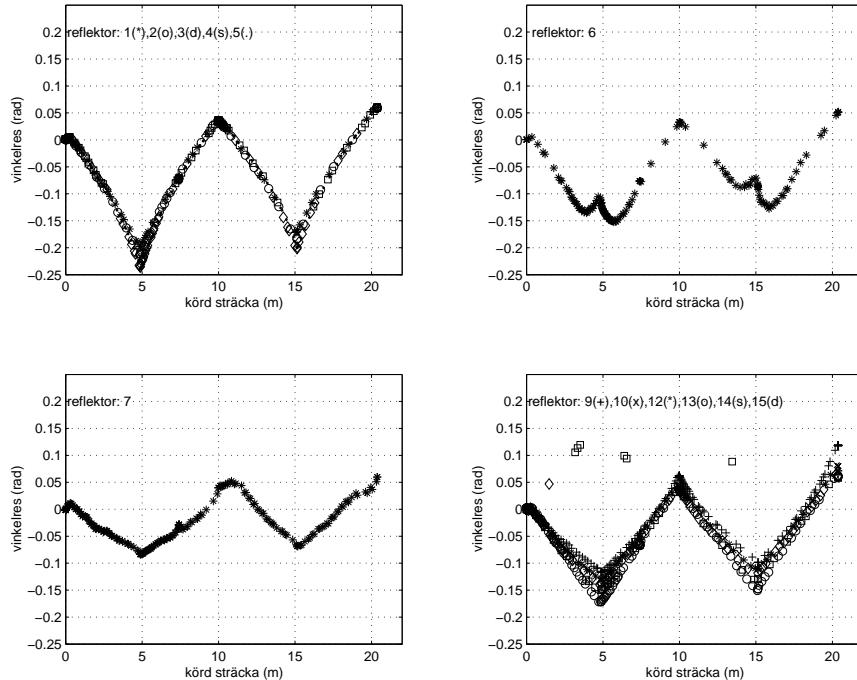
3.1 Residualens utseende

Residualfunktionen beräknas från början för två olika testkörningar, en rak bana, fall a, och en kurvig bana, fall b.

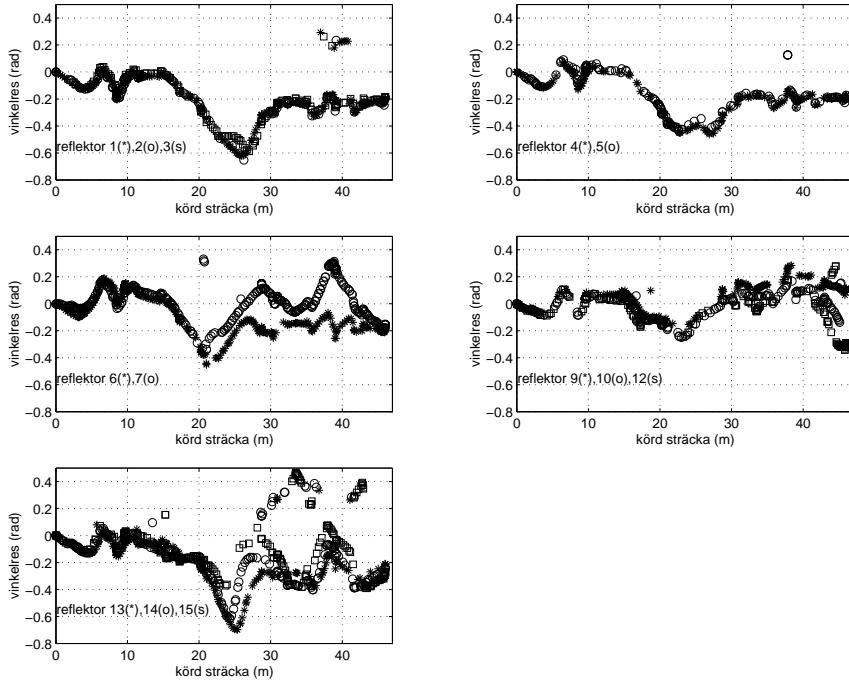


Figur 6. Vagnens bana i rummet relativt reflektorerna, (numrerade), t.v fall a, t.h fall b.

Om residualen plottas för varje reflektor som en funktion av tillryggalagd vägsträcka, kan olika mönster hos residualen för olika grupper av reflektorar ses. I beräkningarna används följande värden för parametrarna: $p = [0, 0, 1, 1, 0.690, 0.585, 0, 1.5758]$. Den gruppindelning som görs verkar ha med reflektorernas placering relativt vagnen att göra. Detta är särskilt tydligt hos residualen för den raka banan där varje spets motsvaras av en ändring av körriktningen hos vagnen.



Figur 7. Gruppindelade residualkurvor för de olika reflektorerna i fall a.



Figur 8. Gruppindelade residualkurvor för de olika reflektorerna i fall b.

Som tidigare nämnts är indelningen för fall b inte lika självklar. Notera att skalningen för de olika fallen inte är densamma.

För att få en förståelse för hur residualfunktionen ändrar sig/uppför sig m.a.p de olika parametrarna, studeras deras olika inverkan på funktionen. Då en parameters påverkan eller en kombination av vissa parametrars påverkan på residualfunktionen undersöks antas de övriga parametrarna vara konstanta och residualfunktionen betecknas $r(p_i)$, där p_i nu betecknar aktuell parameter. En metod för dessa undersökningar är att variera värdet för den parameter som undersöks och se vad som händer med residualfunktionen. Denna plottas på olika sätt för att få en bild av parameterns påverkan. En annan metod är att approximera derivatan för residualfunktionen m.a.p aktuell parameter. Då denna plottas fås också en bild av en eventuell påverkan och ett mått på hur stor denna påverkan är.

3.2 Påverkan av vinkeloffset för styrvinklarna, α_1 respektive α_2 , på residulfunktionen

Det första paret av parametrar som undersöks är vinkeloffset för styrvinklarna, α_1 och α_2 , med resten av parametrarna konstanta. Residualfunktionen, $r(\alpha_1, \alpha_2)$ plottas i MATLAB m.a.p en liten förändring i α_1 respektive α_2 .

Residualen för de olika reflektatorgrupperna, i fall a, påverkas ungefär lika mycket för varje förändring. Bilaga 1 visar residualens förändring m.a.p olika störningar, δ_{α_1} och i Bilaga 2 ses det samma m.a.p δ_{α_2} . Det verkar finnas ett beroende mellan dessa parametrar på så sätt att $r(\alpha_1, 0) \approx r(0, -\alpha_1)$, vilket skulle innebära att en ändring på α_1 ger ungefär samma residualvektor som en ändring på α_2 om $\alpha_2 = -\alpha_1$. Detta undersöks närmare bl.a. genom beräkningar i Mathematica, [2], och kan nedan ses numerisk då skillnaderna för residualfunktionen m.a.p α_1 respektive α_2 minskar då α_1 går mot noll.

Man har att α_1 och α_2 påverkar beräkningen av residualfunktionen fram till och med det att laserscannerns globala position, (X, Y, θ) , beräknas. Därför räcker det att linjärisera dessa koordinater m.a.p α_1 respektive α_2 och beräkna skillnaden. Skillnaden mellan residualfunktionen m.a.p var och en av dessa parametrar beror på skillnaden mellan de globala positionerna. Beräkningarna görs endast för ett tidssteg då uttrycken därefter blir mer komplexa, men det verkar rimligt att resultaten även gäller för kommande tidssteg.

Den relativna skillnaden för X_1 , Y_1 och θ_1 m.a.p α_1 :

$$\begin{aligned} \frac{X_1(\alpha_1, 0) - X_1(0, -\alpha_1)}{\alpha_1} &= \frac{-2\delta_T(Lv_1 + (v_2 - v_1)x_s) \sin(\theta_0) \cos(u_2)}{2L} \\ &+ \frac{\delta_T(v_1 - v_2) \cos(\theta_0)(2y_s \cos(u_2) + L \sin(u_2))}{2L} \\ \frac{Y_1(\alpha_1, 0) - Y_1(0, -\alpha_1)}{\alpha_1} &= \frac{2\delta_T(Lv_1 + (v_2 - v_1)x_s) \cos(\theta_0) \cos(u_2)}{2L} \quad (9) \\ &+ \frac{\delta_T(v_1 - v_2) \sin(\theta_0)(2y_s \cos(u_2) + L \sin(u_2))}{2L} \\ \frac{\theta_1(\alpha_1, 0) - \theta_1(0, -\alpha_1)}{\alpha_1} &= \frac{\delta_T(v_2 - v_1) \cos(u_2)}{L} \end{aligned}$$

Undersökningarna görs då styrvinklarna är relaterade så att $u_1 = -u_2$, $d_1 = d_2$ antas vara 1 samt $\delta_T = 0.05$ s, $|v_1 - v_2| \leq 0.05$ m/s och $v_i < 0.7$ m/s. Sätts dessa värden in i (9), blir produkterna $\delta_T(v_1 - v_2)$ och $\delta_T(v_2 - v_1)$ så små att termer beroende på dessa är försumbara. Kvar blir då

$$\begin{aligned} \frac{X_1(\alpha_1, 0) - X_1(0, -\alpha_1)}{\alpha_1} &\approx -0.05v_1 \sin(\theta_0) \cos(u_2) \\ \frac{Y_1(\alpha_1, 0) - Y_1(0, -\alpha_1)}{\alpha_1} &\approx 0.05v_1 \cos(\theta_0) \cos(u_2) \quad (10) \\ \frac{\theta_1(\alpha_1, 0) - \theta_1(0, -\alpha_1)}{\alpha_1} &\approx 0 \end{aligned}$$

där $|v_1 \sin(\theta_0) \cos(u_2)| \leq 1$ och $|v_1 \cos(\theta_0) \cos(u_2)| \leq 1$. Detta kan förklara

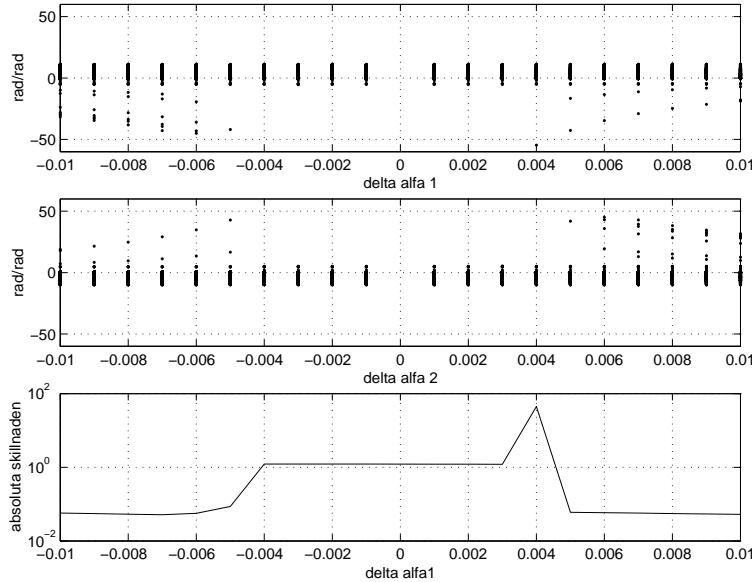
varför plottarna har den likhet de har och man kan dra slutsatsen att hypotesen $r(\alpha_1, 0) \approx r(0, -\alpha_1)$ stämmer då α_1 är liten.

I Bilaga 7 ses hela uträkningen och där finns även jämförelser för styrvinkelrelationerna $u_1 = u_2$ samt $u_1 = u_2 = \pi/2$. Då den senare relationen gäller fås en vridning av vagnen genom att låta hjulen rotera med olika hastighet, vagnen benämns då diff-vagn.

För att få en jämförelse mellan parametrarnas olika påverkan på residualfunktionen approximeras även derivatorna m.a.p dessa enligt differensmetoden:

$$\begin{aligned}\frac{\partial r(\alpha_1, \alpha_2)}{\partial \alpha_1} &\approx \frac{r(\alpha_1 + \delta\alpha_1, \alpha_2) - r(\alpha_1, \alpha_2)}{\delta\alpha_1} \\ \frac{\partial r(\alpha_1, \alpha_2)}{\partial \alpha_2} &\approx \frac{r(\alpha_1, \alpha_2 + \delta\alpha_2) - r(\alpha_1, \alpha_2)}{\delta\alpha_2}\end{aligned}\quad (11)$$

Då derivatorna m.a.p dessa två parametrar plottas i MATLAB kan man konstatera att det även här finns ett beroende på sådant sätt att $\frac{\partial r(\alpha_1, 0)}{\partial \alpha_1} \approx -\frac{\partial r(0, -\alpha_2)}{\partial \alpha_2}$ då $\alpha_2 = \alpha_1$. Vilket är en naturlig konsekvens av slutsatsen att sambandet $r(\alpha_1, 0) \approx r(0, -\alpha_1)$ gäller.



Figur 9. Likhet mellan derivatorna m.a.p α_1 . Överst $\frac{\partial r(\alpha_1, \alpha_2)}{\partial \alpha_1}$ m.a.p olika $\delta\alpha_1$, mitten $\frac{\partial r(\alpha_1, \alpha_2)}{\partial \alpha_2}$ m.a.p olika $\delta\alpha_2$ och nederst $\|\frac{\partial r(\alpha_1, 0)}{\partial \alpha_1}\|_2 - \|\frac{\partial r(0, -\alpha_2)}{\partial \alpha_2}\|_2$.

Residualfunktionen har som tidigare nämnts ett element för varje tidpunkt. Därför fås en vektor vid varje beräkning av residualfunktionen m.a.p ett nytt α_i . Detta är förklaringen till att det är staplar i de två översta

plottarna i figur 9 för varje $\delta\alpha_i$, det är hela vektorn som plottas. I den nedsta plotten är det den euklidiska vektornormen som absoluta skillnaden beräknas för. Detta gör att en skillnad på ett inte är så stor då vissa element i residualen har t.ex. absoluta värdet 25.

3.3 Påverkan av skalfel på hastigheterna, d_1 respektive d_2 , på residualfunktionen

På samma sätt som för $r(\alpha_1, \alpha_2)$ plottas residualfunktionen $r(d_1, d_2)$ i MATLAB m.a.p en liten förändring i d_1 respektive d_2 . Nu är alla parametrar förutom d_1 och d_2 konstanta. I Bilaga 3 och 4 ses att residualen för reflektor 6 och 7 påverkas mycket mer av en störning, δ_{d_i} , på d_1 respektive d_2 än vad residualen för övriga reflektorggrupper gör för den rak banan. Residualfunktionerna m.a.p de olika parametrarna, d_1 och d_2 , verkar även här följa varandra. På grund av detta undersöks om det kan finnas ett beroende även mellan dessa parametrar så att $r(d_1, 1) \approx r(1, d_1)$, d.v.s en liten ändring på d_1 ger ungefär samma residualvektor som samma ändring av d_2 .

Laserscannerns globala position (X, Y, θ) linjäriseras denna gång m.a.p d_1 respektive d_2 och den relativ skillnaden beräknas även här i Mathematica, resultatet blir

$$\begin{aligned} \frac{X_1(d_1, 1) - X_1(1, d_1)}{d_1} &= (d_1 - 1)\delta_T \left(\frac{2(Lv_1 + (v_2 - v_1)x_s) \sin(\theta_0) \sin(u_2)}{2L} \right. \\ &\quad \left. + \frac{(v_1 - v_2) \cos(\theta_0)(L \cos(u_2) - 2y_s \sin(u_2))}{2L} \right) \quad (12) \\ \frac{Y_1(d_1, 1) - Y_1(1, d_1)}{d_1} &= (d_1 - 1)\delta_T \left(\frac{-2(Lv_1 + (v_2 - v_1)x_s) \cos(\theta_0) \sin(u_2)}{2L} \right. \\ &\quad \left. + \frac{(v_1 - v_2) \sin(\theta_0)(L \cos(u_2) - 2y_s \sin(u_2))}{2L} \right) \\ \frac{\theta_1(d_1, 1) - \theta_1(1, d_1)}{d_1} &= \frac{(d_1 - 1)\delta_T(v_1 - v_2) \sin(u_2)}{L} \end{aligned}$$

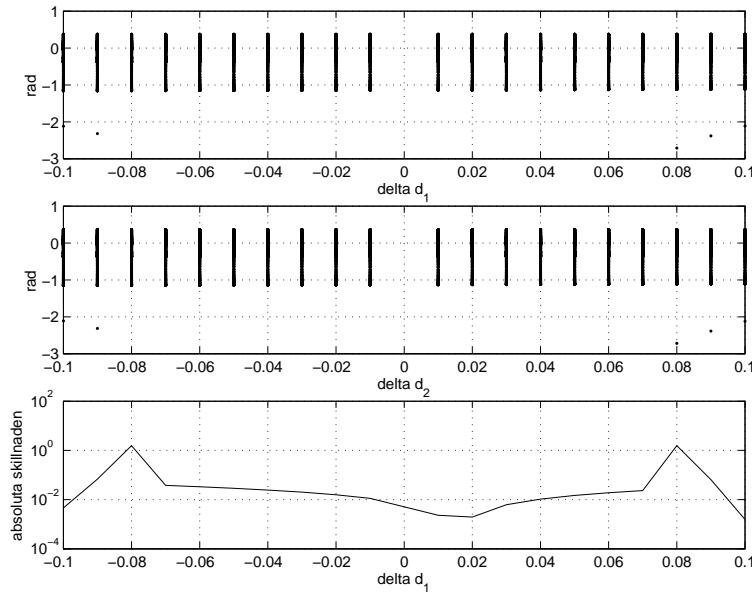
Beräkningarna görs denna gång under följande villkor: $u_1 = -u_2$, $\alpha_1 = \alpha_2$ antas vara 0, $\delta_T = 0.05$ s och $|v_1 - v_2| \leq 0.05$ m/s. Då dessa värden sätts in i (12) blir termer beroende av produkterna $\delta_T(v_2 - v_1)$ och $\delta_T(v_1 - v_2)$ försumbara och kvar blir då:

$$\begin{aligned}
\frac{X_1(d_1, 1) - X_1(1, d_1)}{d_1} &\approx (d_1 - 1)0.05v_1 \sin(\theta_0) \sin(u_2) \\
\frac{Y_1(d_1, 1) - Y_1(1, d_1)}{d_1} &\approx -(d_1 - 1)0.05v_1 \cos(\theta_0) \sin(u_2) \\
\frac{\theta_1(d_1, 1) - \theta_1(1, d_1)}{d_1} &\approx 0
\end{aligned} \tag{13}$$

Låter man nu $d_1 \rightarrow 1$ ses på samma sätt som för α_1 att skillnaden $\rightarrow 0$, vilket förklarar plottarnas likheter. Precis som för α är även beräkningar gjorda för de andra styrvinkelrelationerna, hela uträkningen finns i Bilaga 8. Derivatan för residualfunktionen approximeras nu m.a.p d_1 respektive d_2 :

$$\begin{aligned}
\frac{\partial r(d_1, d_2)}{\partial d_1} &\approx \frac{r(d_1 + \delta d_1, d_2) - r(d_1, d_2)}{\delta d_1} \\
\frac{\partial r(d_1, d_2)}{\partial d_2} &\approx \frac{r(d_1, d_2 + \delta d_2) - r(d_1, d_2)}{\delta d_2}
\end{aligned} \tag{14}$$

Dessa approximerade derivator plottas i MATLAB och man kan konstatera att även de följer varandra på sådant sätt att $\frac{\partial r(d_1, 1)}{\partial d_1} \approx \frac{\partial r(1, d_2)}{\partial d_2}$ då $d_2 = d_1$. Vilket även här är en naturlig konsekvens av det beroende som konstaterats finns.

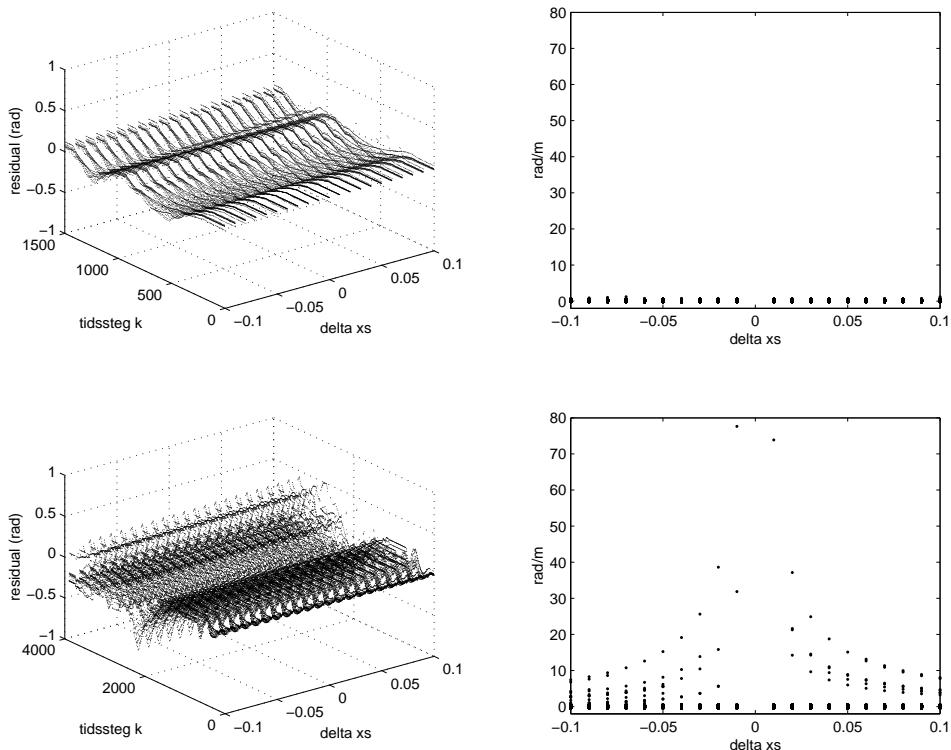


Figur 10. Likhet mellan derivatorna m.a.p d_1 , överst $\frac{\partial r(d_1, d_2)}{\partial d_1}$ m.a.p olika δd_1 , mitten $\frac{\partial r(d_1, d_2)}{\partial d_2}$ m.a.p olika δd_2 och nederst $||\frac{\partial r(d_1, 1)}{\partial d_1} - \frac{\partial r(1, d_2)}{\partial d_2}||_2$.

Precis som för α_i beräknas hela residualvektorn för varje d_i och plottas som en stapel i övre och mittersta plotten i figur 10. Nedersta plotten visar också här absoluta skillnaden av den euklidiska vektornormen av de approximerade derivatorna.

3.4 Övriga parametrars påverkan på residualfunktionen

Då residualfunktionen ritas ut som en funktion av både tiden och aktuell parameter fås en 3D-plott med de olika parametervärdena på x-axeln, tidssteg på y-axeln och residualfunktionens värde i z-led. Detta kan man säga ger en yta av residualfunktioner, kallas nedan residualyta. Ytans förändring m.a.p olika parametervärden är då en illustration av hur parametern påverkar residualfunktionen. Sådana residualytor plottas m.a.p olika x_s , scannerns lokala x-koordinat, både för fall a och b, se figur 11.



Figur 11. T.v Residualytor m.a.p x_s och t.h approximerade derivator m.a.p x_s

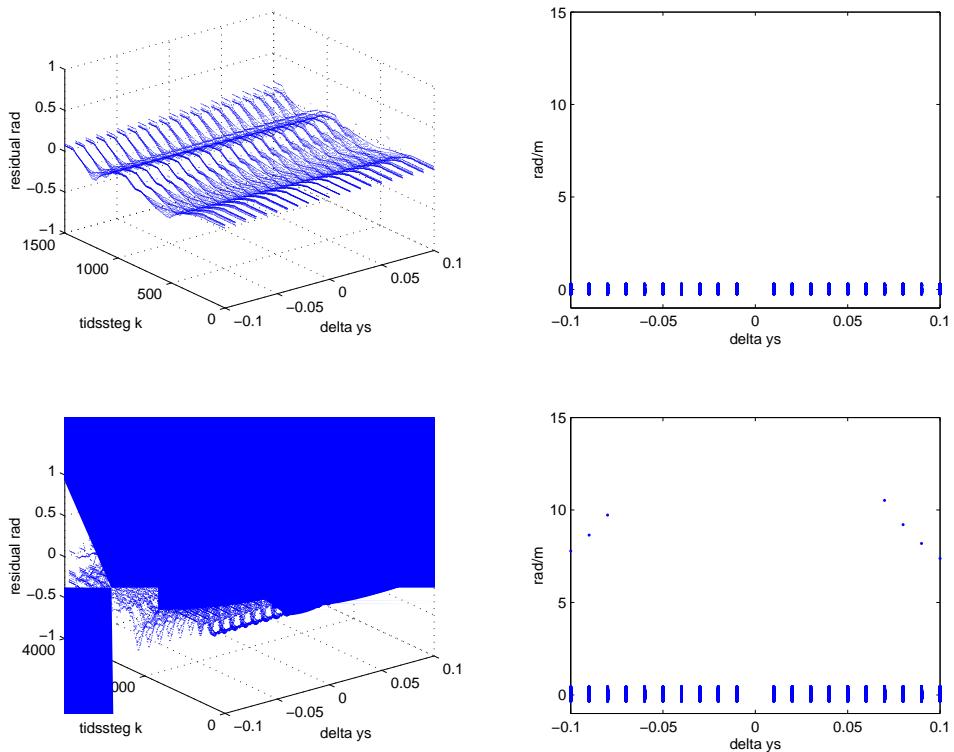
Övre bilden t.v är residualytan för fall a och nedre bilden t.v är för fall b. I båda fallen verkar ytan relativt plan och detta kan då tolkas som att x_s inte

har någon större påverkan på residualfunktionen oavsett om vagnen rör sig rakt, d.v.s utan vridning på hjulen, eller om den svänger. Residualfunktionens derivata approximeras m.a.p x_s :

$$\frac{dr(x_s)}{dx_s} \approx \frac{r(x_s + \delta x_s) - r(x_s)}{\delta x_s} \quad (15)$$

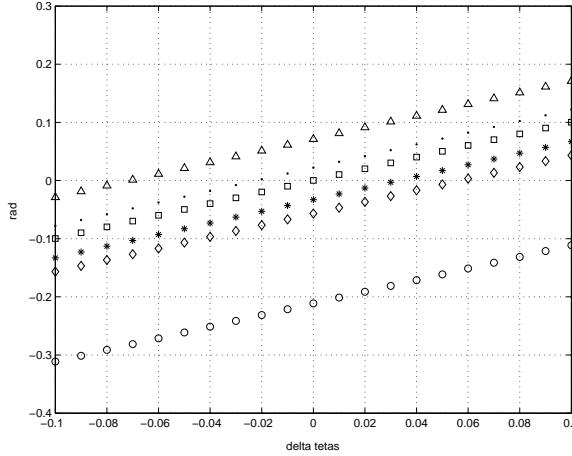
och plottas i MATLAB både för fall a och b, se figur 11, där övre bilden t.h är derivatan i fall a och nedre bilden t.h i fall b. Precis som tidigare plottas hela residualvektorn som en stapel för varje värde på x_s . Denna stapel innehåller lika många punkter för varje parametervärde men skiljer i antal punkter för de olika testfallen eftersom de körda banorna är olika långa. Man ser att den approximerade derivatan för den raka banan, fall a, är liten vilket förklarar att residualytan inte påverkats speciellt mycket i detta fall. För den kurviga banan, fall b, ändå är derivatan större för vissa punkter i residualstapeln. De flesta punkter verkar dock ha en liten derivata vilket kan förklara att residualytan ser relativt plan, men man får ändå anta att x_s har en viss påverkan på residualfunktionen då vagnen svänger.

På samma sätt plottas residualytor m.a.p tiden och olika värden av parametern y_s , scannerns lokala y-koordinat. Även dessa ytor ser plana ut, se figur 12, där övre plotten t.v är för fall a och nedre t.v för fall b. Detta antyder att förändringar av y_s borde ha en liten påverkan på residualfunktionen. Derivatan approximeras enligt differensmetoden, p.s.s som för parametern x_s ovan, och plottarna av dessa för fall a ses i figur 12 överst t.h och för fall b nederst t.h. För fall a är den approximerade derivatan väldigt liten och för fall b är det några punkter m.a.p olika y_s som har en högre derivata. Slutsatsen blir även här att förändringar på y_s inte påverkar residualen då vagnen rör sig rakt, utan att svänga, men att viss påverkan finns då vagnen svänger.



Figur 12. T.v Residualytör m.a.p y_s och t.h approximerade derivator m.a.p y_s

Undersökningar utförs nu av parametern θ_s , laserscannerns avvikelse från lokala x-axeln. Utseendet på plottytan för fall a, se Bilaga 5 övre plott t.v, ger intryck av att θ_s påverkar residualen linjärt i detta fall. Ytan för fall b, Bilaga 5 nedre t.v, är inte lika tydligt linjär, men man ser att den förändras m.a.p olika θ_s . De approximerade derivatorna för residualfunktionen m.a.p θ_s visar att påverkan av parametern på funktionen finns i båda fallen, men den ser inte ut att vara konstant för alla punkter i residualstaplarna för fall a, Bilaga 5 övre plott t.h. Det borde vara så om förändringen av residualfunktionen m.a.p olika θ_s är linjär. Förändringen för vissa, slumpmässigt valda, tidpunkter i residualen plottas m.a.p olika θ_s , se figur 13. Här syns att residualens värde i de olika tidpunkterna ökar linjärt med θ_s , så trots att den approximerade derivatan inte är konstant kan man anta att residualfunktionen förändras linjärt m.a.p θ_s . Med slumpmässigt valda tidpunkter menas att sex stycken tidpunkter valts ut med ungefär samma mellanrum, ca 300 tidssteg mellan varje.



Figur 13. Residualens värde m.a.p olika θ_s för vissa tidpunkter k . Trekanter $k = 1475$, punkter $k = 913$, fyrkanter $k = 3$, stjärnor $k = 310$, ruter $k = 1200$, ringar $k = 602$.

Nu undersöks parametern L på liknande sätt, se Bilaga 6. Här ser det ut som att residualytan för fall a, övre plott t.v, är nästan plan, precis som för x_s och y_s . För fall b, nedre t.v, märks däremot en påverkan av L på residualfunktionen. De approximerade derivatorna för residualfunktionen i fall a och b m.a.p L visar att man kan anta att påverkan i fall a, övre t.h, är näst intill obefintlig då derivatan är konstant nära noll, däremot i fall b, nedre t.h, är påverkan betydligt större.

3.5 Sammanfattning av parametrarnas påverkan

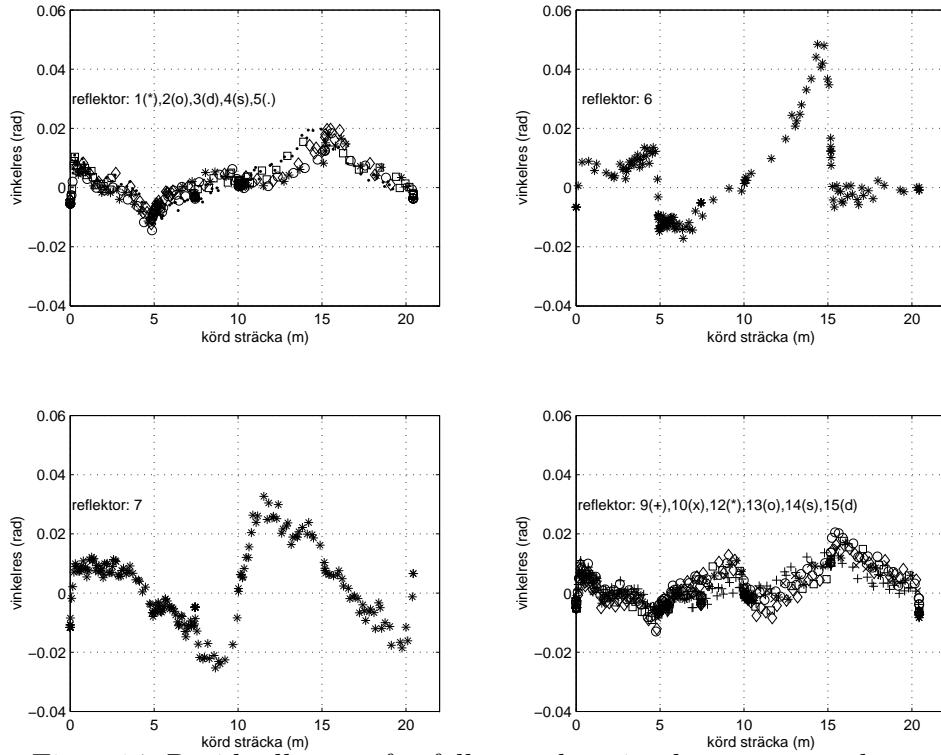
Ovan gjorda undersökningar visar att det finns ett beroende mellan parametrarna α_1 och α_2 på sådant sätt att $r(\alpha_1, 0) \approx r(0, -\alpha_2)$ och $\frac{\partial r(\alpha_1, 0)}{\partial \alpha_1} \approx -\frac{\partial r(0, -\alpha_2)}{\partial \alpha_2}$ då $\alpha_2 = \alpha_1$. De visar även att vinkeloffsetparametrarna har en ganska stor påverkan på residualfunktionen både då vagnen kör rakt, d.v.s inte svänger, samt när den svänger. Ett beroende mellan parametrarna d_1 och d_2 kan också ses på sådant sätt att $r(d_1, 1) \approx r(1, d_2)$ och $\frac{\partial r(d_1, 1)}{\partial d_1} \approx \frac{\partial r(1, d_2)}{\partial d_2}$ då $d_2 = d_1$. Dessa skalfelsparametrar har också en påverkan på residualfunktionen både då vagnen inte svänger, samt när den gör det. Parametrarna L , x_s och y_s verkar däremot inte ha någon större påverkan på residualen då vagnen bara kör fram och tillbaka utan att svänga. För att någon förändring m.a.p dessa parametrar ska märkas måste vagnen svänga, men inte heller då är påverkan av x_s eller y_s så stor. Påverkan av parametern θ_s är märkbar för båda körsätten men större då vagnen svänger. Då vagnen kör rakt verkar påverkan vara linjär.

4 Optimering

Den optimeringsrutin som används är MATLAB's lsqnonlin, vilket är en lösare för ickelinjära minstakvadratproblem. Denna rutin kan ta en vektorvärd funktion som argument, (Bilaga 9), och minimerar sedan kvadratsumman av funktionen. En annan metod som också prövas är fminsearch, också den en MATLAB-rutin. Denna metod kan bara ta skalära funktioner som argument, vilket gör att målfunktionen som ska minimeras måste definieras. På så vis är detta en generellare metod eftersom målfunktionen inte behöver vara kvadratsumman av någon funktion. Optimering med fminsearch kan alltså inte ske direkt på residualfunktionen, utan kvadratsumman av denna beräknas först. Då de två metoderna används visar det sig att det tar betydligt längre tid för fminsearch att konvergera än det tar för lsqnonlin. Resultaten blir dock lika och valet blir alltså att använda lsqnonlin. Nedan kommer den parametervektor som optimeringsrutinen konvergerar mot kallas optimal, detta är kanske inte helt sant eftersom det kan vara ett lokalt minimum som rutinen hittar. För utförligare teori angående icke-linjär optimering se t.ex [7] eller [9].

4.1 Optimering av den raka banan

Optimering görs för hela residualen m.a.p alla okända parametrar. Som startvektor används det p som definierats tidigare, se sid. 13 avsnitt 3.1. Optimeringsrutinen konvergerar mot en optimal parametervektor enligt satta avbrottsskriterier i MATLAB, se Bilaga 9. Residualen beräknas med denna optimala parametervektor och plottas i MATLAB för de olika reflektorggrupperna enligt tidigare uppdelning, figur 14. Det ses då att residualen minskat med ungefär en faktor 10. Avvikande är residualkurvorna för reflektorer 6 och 7, som även med p har annorlunda form än residualen för övriga reflektorer fast då inte högre värden vilket nu är fallet, jmf. med figur 7, (observera att skalningen är olika).



Figur 14. Residualkurvor för fall a med optimal parametervektor.

En övre acceptabel gräns för residualen är att det absoluta värdet av varje element är ≤ 0.03 rad, men med ett bra val av parametrar är detta värde ≤ 0.005 rad. Som ses från figur 14 uppfylls inte dessa villkor för reflektor 6 eller 7, därför utförs försök med en generellare modell för att se om residualkurvorna för dessa reflektorer kan påverkas mer. Parametrarnas olika ingående i beräknanget av residualfunktionen förändras också, t.ex. får de indata som påverkas relativt av en parameter även påverkas absolut av ytterligare en parameter. Relativ påverkan har t.ex hastighetsskalningen och absolut påverkan har vinkeloffset. Följden av att förändra ett absolut ingående av parametern till att vara relativt undersöks också. På samma sätt förändras även ingåendet för de parametrar som ingår relativt. Även tillägg av hur fler parametrar i modellen påverkar residualen undersöks. Inget av dessa gjorda försök ger dock någon förbättring.

Med tanke på residualens ursprungliga utseende antas att ett bättre resultat kan uppnås om residualen optimeras i delsekvenser, då med hänsyn till vagnens rörelse. Det skulle kanske behövas en uppsättning parametrar då vagnen kör framåt och en annan då den backar. Vilket skulle kunna vara en följd av mekaniska orsaker, t.ex att hjulen slirar som då kan resultera i att styrvinkeloffset förändras då vagnen kör framåt jämfört med när den

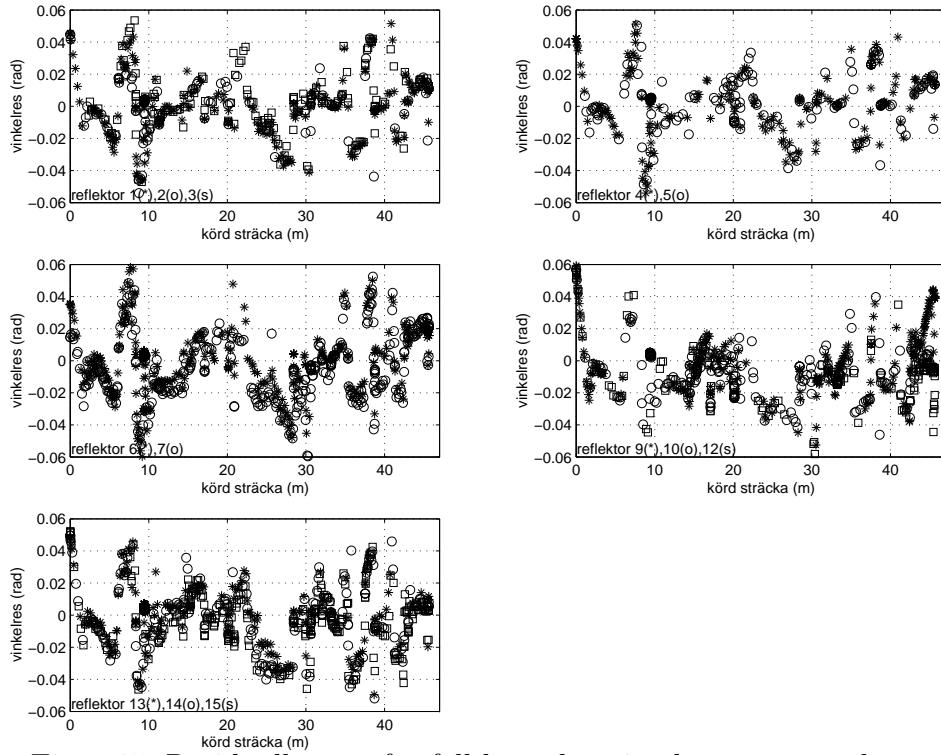
backar. Optimering av delsekvenser prövas och ger faktiskt ett bättre resultat med absoluta beloppet för varje element i residualvektorn ≤ 0.03 rad, men utseendet på residualen för reflektor 6 och 7 är fortfarande avvikande.

Om man delar upp banan i ännu mindre delsekvenser och optimerar blir beloppet för residualens element ännu lägre men nu är inte längre kopplingen till vagnens rörelsen lika tydlig. För att uppnå detta resultat i praktiken skulle en kontinuerlig uppdatering av parametrarna behöva göras under tiden då vagnen körs. Detta är inget alternativ i nuläget utan parametrarna ska bestämmas innan vagnen tas i bruk.

Residualfunktionen optimeras med en störning på en parameter i startvektorn p i taget för att undersöka om parametervärdena går mot samma optimala värden som utan störningen. Detta för att det kan finnas olika lokala minimum som optimeringsrutinen kan konvergera mot då man startar från olika startvektorer. För var och en av vinkeloffsetparametrarna, α_1 och α_2 , läggs på en störning med 0.09 rad $\approx 5^\circ$. För varje skalfelsparameter, d_1 och d_2 , ändras startvärdet med 10% och för parametrarna L , x_s och y_s ändras det med 10 cm var. θ_s 's startvärde får ett tillägg på 0.09 rad, precis som för α_1 och α_2 . För alla nya startvektorer blir parametrarnas optimala värden desamma och av detta kan man dra slutsatsen att optimeringen sannolikt konvergerar mot ett optimum. Tyvärr ger den optimala parametervektorn inte tillräckligt liten residual och skalningsfaktorerna, d_1 och d_2 , för hastigheterna antar underliga värden, $d_1 = 1.9382$ och $d_2 = 0.0633$, mot vad som är rimligt, 1+något litet. Detta beror förmödligen på det beroende som antagits finns mellan dessa parametrar, (se avsnitt 3.3). d_1 och d_2 kan förmödligen inte bestämmas enskilt utan endast någon kombination av dessa.

4.2 Optimering av den kurviga banan

Residualfunktionen optimeras igen m.a.p alla okända parametrar, denna gång på den kurviga banan, med p från sid 13. Som bäst lyckas residualens värde minskas med en faktor 10, i början av residualkurvorna är minskningen ungefär hälften, (se figur 15). Flera värden är fortfarande alldeles för höga enligt gränsvärdet för hur mycket residualen får avvika. Kurvan har heller ingen slumpmässig spridning kring noll, vilket är ett önskvärt resultat vid en bra optimering, ursprungsformen ser ut att finnas kvar. Jämför med figur 8, (observera även här skillnaden i skalning).



Figur 15. Residualkurvor för fall b med optimal parametervektor.

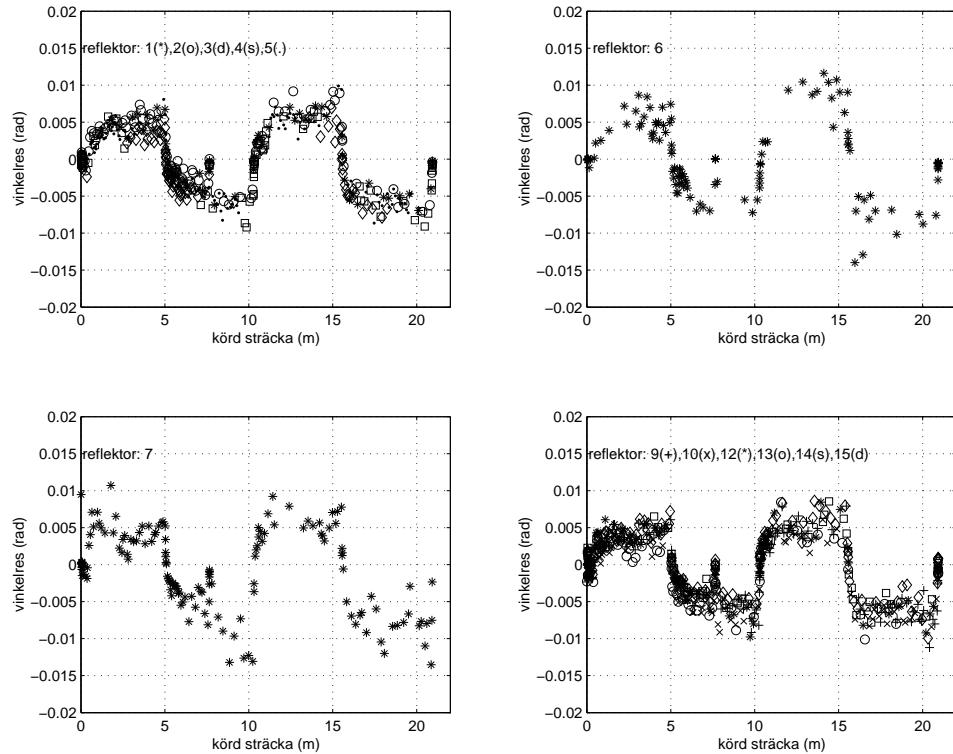
Precis som för optimering av residualfunktionen för fall a undersöks parametrarnas störningskänslighet. Detta genom att lägga till förändringar för en parameter i taget i p som residualen sedan optimeras m.a.p. Förändringarna är de samma som för fall a. Det visar sig att alla parametrar konvergerar mot samma optimum m.a.p alla nya startvektorer. I detta fall är även de optimala värdena för d_1 och d_2 rimliga, d.v.s 1+något litet.

4.3 Utvärderingsprogram

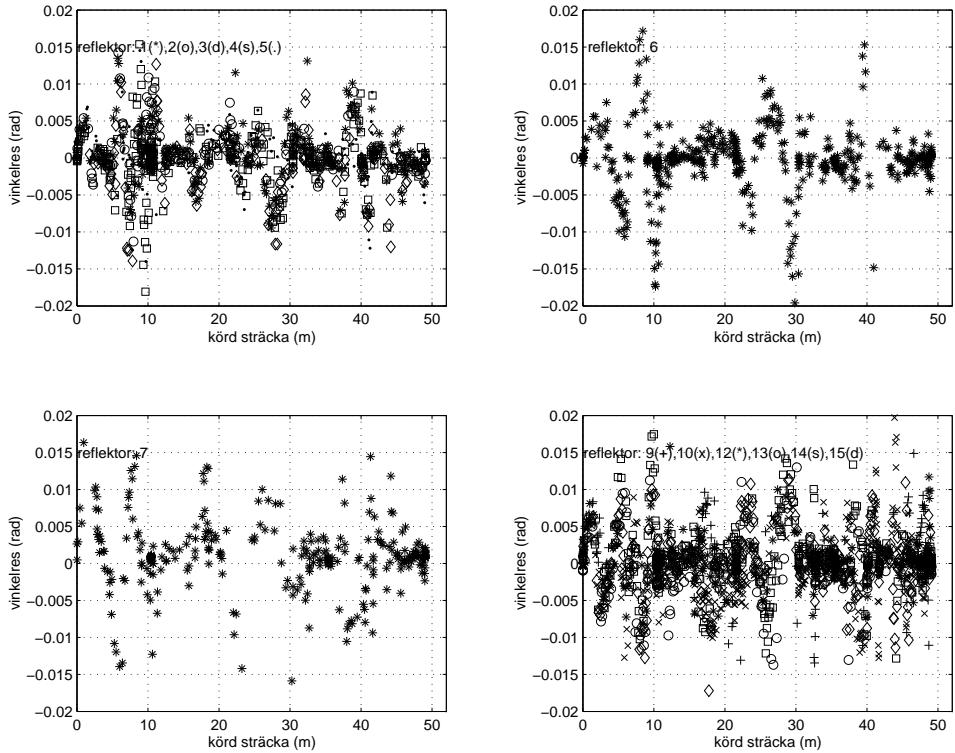
I praktiken uppdateras vagnens position med hjälp av ett Kalmanfilter, se kapitel 5. För att se residualens verkliga värden testas de optimala och ursprungliga parametervärdena i ett utvärderingsprogram skrivet av uppdragsgivaren. Detta program består av det navigeringsprogram som används för körning av vagnarna. Utvärderingsprogrammet gör så att navigeringen kan ske med redan loggad data från en körning. Man kan säga att utvärderingsprogrammet kör "off-line" och beräknar residualen för körningen med hänsyn till ett Kalmanfilter.

Det visar sig att residualens utseende förändras för de olika testfallen, se figur 16 och 17, och redan med startvektorn p , sid 13, är residualens värden

betydligt lägre. Nästan alla element i residualvektorn för den raka banan har absolutbeloppet lägre än 0.01 rad och för den kurviga banan är beloppet av elementen mindre än 0.03 rad, men i många fall bättre.



Figur 16. Residualkurvor för fall a, uppdelade i reflektorggrupper p.s.s som tidigare, här beräknade i utvärderingsprogrammet med p .



Figur 17. Residualkurvor för fall b beräknade i utvärderingsprogrammet med p .

Utvärderingsprogrammet testas på den raka banan med de optimala parametrarna för residualfunktionen för denna körning. Detta ger en klar försämring. En orsak till detta kan vara att i beräkningen av residualfunktionen används medelhastigheten, (2), men i utvärderingsprogrammet tas bara hänsyn till hastigheten från framhjulet. Beroendet mellan skalningsfaktorerna, sidan 17, ställer då till det och vad som är optimalt för ett sätt att beräkna residualen är det inte då beräkningssättet ändras.

Vad som känns naturligt är att en residualfunktion som tar hänsyn till Kalmanfiltrering vid uppdatering av laserscanners position behövs. Optimering av parametrarna bör ske på den funktion som används i praktiken. Eftersom målfunktionen blir en annan då filtret används i modellen borde detta påverka optimum. Då Kalmamfiltret ingår i modellen ger detta en rörelse för vagnen som stämmer bättre överens med navigeringen. Detta ger då naturligtvis lägre residualvärdet.

Försök med att först optimera residualen för fall a m.a.p vissa parametrar och sedan låsa dessa då residualen för fall b optimeras m.a.p resten av parametrarna görs enligt nedan. Detta på grund av parametrarnas olika påverkan på residualfunktionen m.a.p olika körmönster, se avsnitt 3.5, sid 22.

Encodervärden från det hjul som ej används i utvärderingsprogrammet ignoreras, d.v.s både hastighet och styrvinkel. Residualfunktionen för testfall a, den raka banan, optimeras m.a.p parametrarna α_2 , d_2 och θ_s , detta ger ett bra resultat för detta fall. Dessa optimala parametrar används i beräkningarna av residualen för testfall b, den kurviga banan, som optimeras m.a.p resterande parametrar α_1 , L , x_s och y_s . d_1 används som skalfaktor så att villkoret $v_{x_1} = v_{x_2}$ uppfylls.

Samtliga optimala parametrar används nu för att beräkna residualen för fall a respektive b. I fall a märks en förbättring, residualens värde minskas, men för fall b syns ingen förändring av residualens utseende.

Att dela upp optimeringen i flera steg är egentligen inget bra alternativ eftersom viss information går förlorad. Bättre vore att hitta en annan modell så att man fångar beroendet mellan parametrarna och på så sätt få bort det.

5 Kalmanfilter

Då enbart rörelsemodellen används för att uppdatera laserscannerns position ger detta större fel ju längre tid vagnen kör. För att undvika detta används de uppmätta vinklarna av Kalmanfiltret så att den beräknade positionen i aktuell tidpunkt korrigeras. I detta fall med en ickelinjär modell, utförs en linjäriseringsprocess för att få filterekvationerna som används för att bestämma vagnens position. Kalmanfilter som fås på detta sätt kallas utökat Kalmanfilter, (EKF- extended Kalman filter), se t.ex. [6]

5.1 Linjärisering av rörelsemodell

Den ickelinjära rörelsemodellen, (5), kan skrivas som

$$\mathbf{X}(k+1) = f(\mathbf{X}(k), \mathbf{e}(k)) + G(k)\mathbf{w}(k) \quad (16)$$

då $\mathbf{X}(k)$ är tillståndsvektorn $[X(k), Y(k), \theta(k)]^T$, $\mathbf{e}(k) = [v_1(k), u_1(k), v_2(k), u_2(k)]^T$ är styrsignalerna som fås in från encodrarna och $\mathbf{w}(k)$ är brussignaler som innehåller modelleringsfel p.g.a osäkerhet i encodervärdena. G är matrisen som modellerar fel i dödräkningsmodellen. Dödräkning är, vad som nämnts tidigare, då vagnens position enbart uppdateras m.a.p indata från encodrarna. Ingen hänsyn tas till vinkelmätningar.

Denna ickelinjära modell, (16), linjäriseras runt en punkt $\bar{\mathbf{X}}(k)$. $\delta\mathbf{X}(k) = \mathbf{X}(k) - \bar{\mathbf{X}}(k)$ är avvikelsen från den punkt som modellen linjäriseras kring och $\delta\mathbf{e}(k) = \mathbf{e}(k) - \bar{\mathbf{e}}(k)$ är styrsignalernas avvikelse från de värden som används i linjäriseringen. Eftersom linjäriseringen görs runt de uppmätta värdena på hastigheter och styrvinkelar, är $\delta\mathbf{e}(k) = 0$ och första ordningens Taylorutveckling blir:

$$f(\mathbf{X}(k), \mathbf{e}(k)) = f(\bar{\mathbf{X}}(k), \bar{\mathbf{e}}(k)) + F(k)\delta\mathbf{X}(k) \quad (17)$$

där:

$$F(k) = \left[\frac{\partial f(\mathbf{X})}{\partial (\mathbf{X})} \right]_{\mathbf{X}=\bar{\mathbf{X}}(k)} = \begin{bmatrix} 1 & 0 & f_{13} \\ 0 & 1 & f_{23} \\ 0 & 0 & 1 \end{bmatrix} \quad (18)$$

är Jacobianen till rörelsemodellen med

$$\begin{aligned} f_{13} &= -\delta_T(v_x \sin \theta(k) + v_y \cos \theta(k)) \\ f_{23} &= \delta_T(v_x \cos \theta(k) - v_y \sin \theta(k)) \end{aligned} \quad (19)$$

i detta fall.

5.2 Linjärisering av mätmodellen

Den ickelinjära mätmodellen, (7), kan skrivas som

$$\gamma(k) = h(\mathbf{X}(k)) + \nu(k) \quad (20)$$

där \mathbf{X} är tillståndsvektorn enligt ovan. Alla fel i mätningarna samlas ihop i termen $\nu(k)$ som antas vara vitt brus med medelvärde noll och kovariansen R . $\nu(k)$ antas också vara okorrelerad med processbruset. Denna modell linjäriseras nu också runt punkten $\bar{\mathbf{X}}(k)$ och en första ordningens Taylorutveckling kring denna punkt blir:

$$\gamma(k) = h(\bar{\mathbf{X}}(k)) + H(k)(\mathbf{X}(k) - \bar{\mathbf{X}}(k)) + \nu(k) \quad (21)$$

där:

$$H(k) = \left[\frac{\partial h(\mathbf{X})}{\partial (\mathbf{X})} \right]_{\mathbf{X}=\bar{\mathbf{X}}(k)} = \left[\frac{dh}{dx}, \frac{dh}{dy}, \frac{dh}{d\theta} \right] \quad (22)$$

är gradienten till mätmodellen med:

$$\begin{aligned} \frac{dh}{dx} &= \frac{r_y}{r_x^2 + r_y^2} \\ \frac{dh}{dy} &= \frac{-r_x}{r_x^2 + r_y^2} \\ \frac{dh}{d\theta} &= -1 \end{aligned} \quad (23)$$

i detta fall, och

$$\begin{aligned} r_x &= X_b - X(k+1) \\ r_y &= Y_b - Y(k+1) \end{aligned} \quad (24)$$

är avståndet mellan reflektor b och laserscannern.

5.3 Filterekvationerna

Utförlig teori och härledning finns bl.a. i [6], [8] och [14].

För att minska felen i den rekursiva uppdateringen av positionen tas hänsyn till uppmätta vinklar av laserscannern enligt:

$$\hat{\mathbf{X}}(k+1|k+1) = \hat{\mathbf{X}}(k+1|k) + K(k+1)(\gamma(k+1) - \hat{\gamma}(k+1|k)) \quad (25)$$

där $\hat{\gamma}(k+1|k) = h(\hat{\mathbf{X}}(k+1|k))$ är den förväntade vinkeln i nästa tidpunkt givet vinkeln i aktuell tidpunkt. För mer sannolikhetsteori se t.ex. [3] eller [6]. $K(k)$ är Kalmanfiltrets återkopplingsfaktorn och denna väljs så att:

$$K(k+1) = P(k+1|k)H(k+1)^T[H(k+1)P(k+1|k)H(k+1)^T + R]^{-1} \quad (26)$$

med R som enligt ovan är kovariansen för mätbruset. P är kovariansmatrisen för felet och H är gradienten för mätmodellen definierad ovan. Kovariansen är ett mått på hur osäker den skattade positionen är. Ju större element i kovariansmatrisen man får desto osäkrare anses positionen vara. Kovariansmatrisen för felet definieras av väntevärdet av produkten av $\delta \mathbf{X}(k)$, som är avvikelsen från den punkt modellen linjäriserades runt:

$$P(k) = E(\delta \mathbf{X}(k)\delta \mathbf{X}(k)^T) \quad (27)$$

och den förväntade kovariansmatrisen i mätpunkten $k + 1$ blir då:

$$P(k+1|k) = E(\delta \mathbf{X}(k+1|k)\mathbf{X}(k+1|k)^T) \quad (28)$$

Den rekursiva formeln för felkovariansmatrisen blir enligt [14]:

$$P(k+1|k) = F(k)P(k|k)F(k)^T + G(k)Q(k)G(k)^T \quad (29)$$

där

$$Q(k) = \begin{bmatrix} q_x & 0 & 0 \\ 0 & q_y & 0 \\ 0 & 0 & q_\theta \end{bmatrix} \quad (30)$$

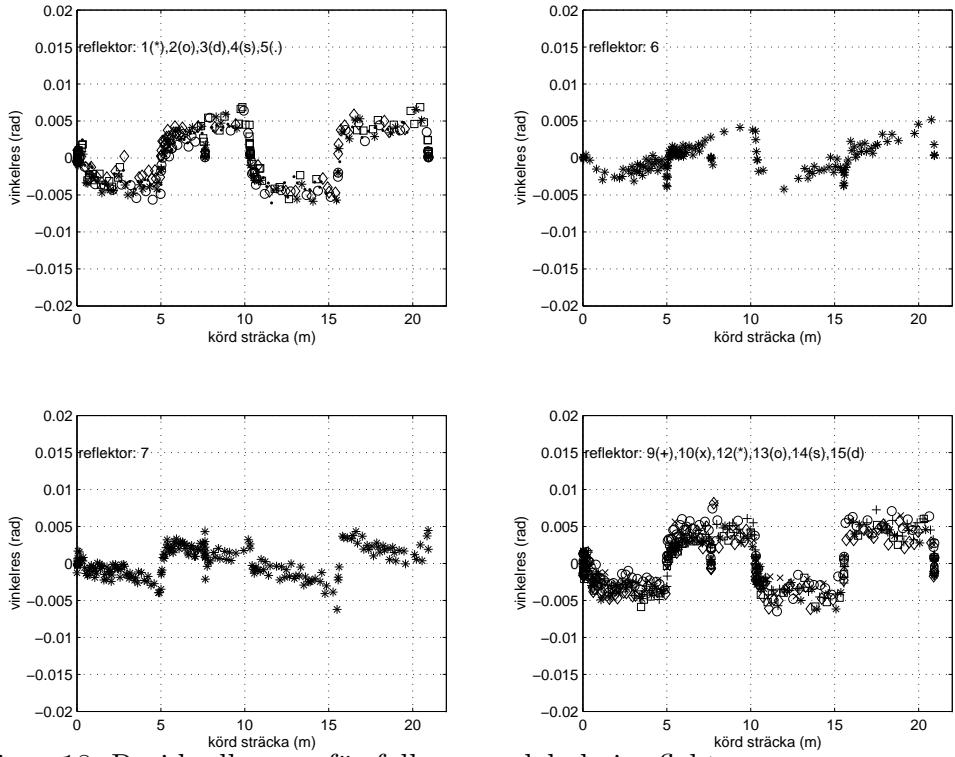
är en modell av kovariansen för modellbruset med

$$\begin{aligned} q_x &= (|v_x|10^{-4} + (|v_y| + 0.1)(|\omega| + 0.01)10^{-3})\delta_T \\ q_y &= (|v_y|10^{-4} + (|v_x| + 0.1)(|\omega| + 0.01)10^{-3}\delta_T \\ q_\theta &= (|\omega| * 2 * 10^{-4} + (|v_x| + |v_y|)10^{-6})\delta_T \end{aligned} \quad (31)$$

i detta fall.

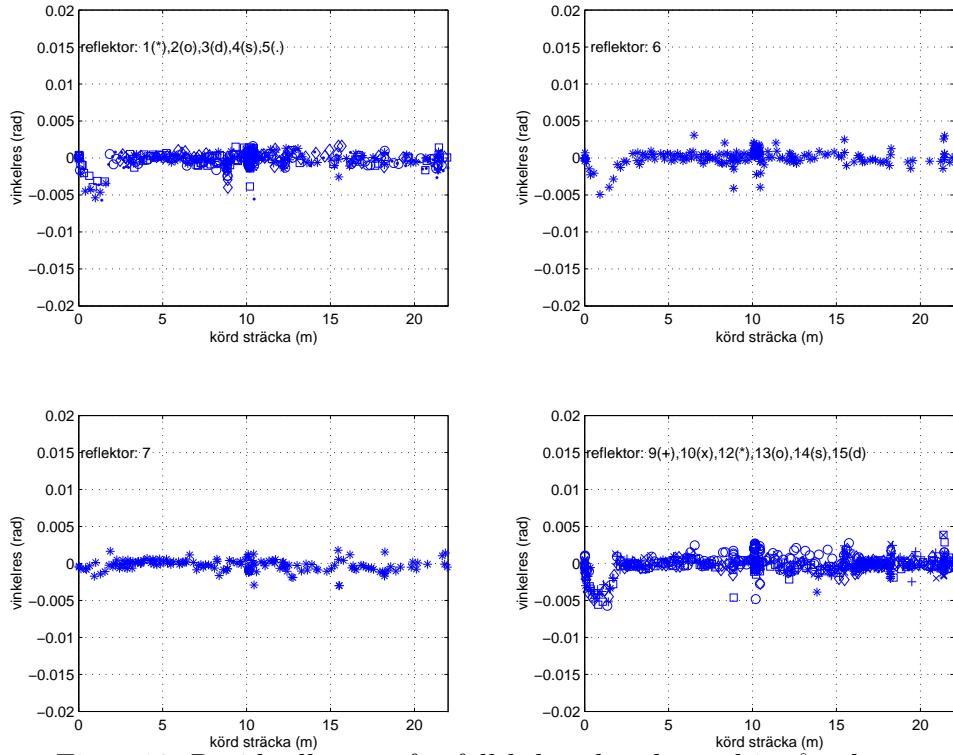
5.4 Residualfunktionen utökad med filterekvationerna

Filterekvationerna läggs nu till vid beräkningarna av residualfunktionen, detta innebär att även målfunktionen förändras. Endast en vinkel används vid varje tidpunkt k , då uppdatering sker. Residualen får mindre värden än utvärderingsprogrammet enbart genom att Kalmanfiltret läggs till i ursprungsmodellen. Residualfilterfunktionen plottas både för den raka och den kurviga banan, (figur 18 och 19), med $p = [0, 0, 1, 1, 0.690, 0.585, 0, 1.5758]$ som tidigare. Elementen i residualvektorn för den raka banan får alla absolutbeloppen mindre än 0.01 rad, de flesta är mindre än 0.005 rad.



Figur 18. Residualkurvor för fall a, uppdelade i reflektorggrupper p.s.s som tidigare, beräknade med ingående av Kalmanfilter.

Residualen har fortfarande ett speciellt mönster men liknar nu mer en stegfunktion istället för kurvor med toppar, se figur 7, precis som residualkurvorna för utvärderingsprogrammet, figur 16. Residualvektorns element för den kurviga banan har i princip alla absolutbeloppet mindre än 0.005 rad, vilket är mindre värden än med utvärderingsprogrammet. Bättre resultat kan nästan inte förväntas även med ett bra val av parametrar.



Figur 19. Residualkurvor för fall b beräknade med ingående av Kalmanfilter.

5.5 Optimering av residualfunktionen med Kalmanfilter

Optimering av residualen för den raka banan, fall a, m.a.p alla parametrar ger väldigt bra resultat, mönstret försätter och värdena sprids slumpräktigt runt noll. Standardavvikelsen för residualen är 0.8 mrad. För att undersöka hur de optimala parametrarna påverkar residualfunktionen då vagnen kör lite mer allmänt beräknas residualen för den kurviga banan med dessa. Detta ger en liten försämring men residualen är fortfarande inom godkända gränser.

På samma sätt som ovan optimeras residualen för den kurviga banan med avseende på alla parametrarna, detta ger en lite förbättring av den redan fina residualkurvan, standardavvikelsen är 0.98 mrad. Residualen beräknas för den raka banan med dessa optimala parametrar. Detta ger ingen förändring av residualens utseende eller värden den hade med p . Värdena var redan tidigare inom godkända gränser och är så fortfarande. Den parameter som är mest förändrad efter optimering av residualen för de två olika testfallen är α_2 som har en skillnad på 19.9 mrad, vilket är ganska mycket.

Olika försök att utöka modellen med ytterligare parametrar görs för att fånga denna skillnad. Då skulle det kanske vara möjligt att hitta ett optimum som förbättrar residualen för båda testfallen samtidigt.

De försök som görs är bl.a att lägga till en extra parameter i cosinustermen för hastigheterna, samt försök med en extra parameter som kommer in tillsammans med en extra cosinusterm för vinklarna. Detta eftersom sinustermen av vinklarna i modellen blir nästan noll då vagnen inte svänger och detta misstänks kunna vara en orsak till att parametrarnas värden skiljer sig mot då vagnen kör kurvigt. De olika utökade modellerna optimeras, men ingen ger någon förbättring mot tidigare resultat.

Försök görs nu med att optimera residualen för de båda testfunktionerna samtidigt, detta ger då en förbättring för båda testfallen om än liten för fall b. Dessa optimala parametrar används i utvärderingsprogrammet som igen ger en felaktig bild eftersom hastigheterna beräknas på olika sätt. Trots detta kan man troligtvis utgå ifrån att optimeringen är tillräckligt bra. Värdena av residualfunktionens element är inom godkända gränser och standardavvikelsen är liten, ca 1.0 mrad.

Häданefter används filtret i residualfunktionen och denna residualfunktion med filterekvationerna kommer fortsättningsvis kallas residualfunktionen.

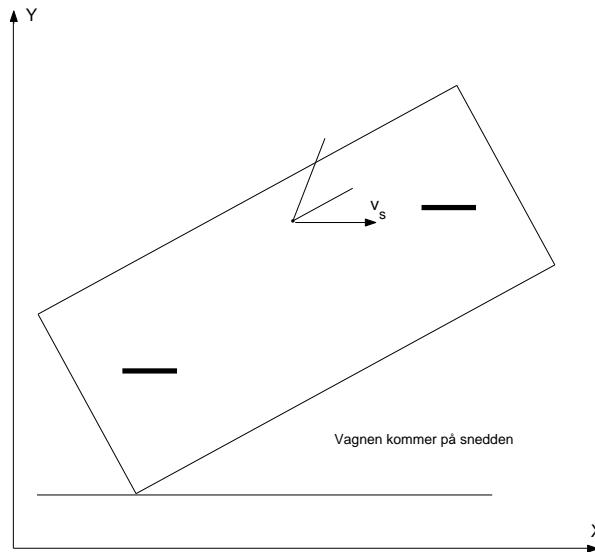
6 Störningskänslighet med avseende på de olika parametrarna

Residualfunktionen undersöks nu för att se hur störningar för de olika parametrarnas startvärden påverkar optimeringen. Residualfunktionen optimeras först med p och sedan med en störning på en parameter i taget för att se om optimum konvergerar mot samma värden.

Störningar som undersöks är tillägg för parametrarna α_1 , α_2 och θ_s med $5^\circ \approx 0.09$ rad, 10 cm fel i avståndsbedömning för L , x_s , y_s och 10% fel i hastighetsskalningen d_1 och d_2 .

Vid optimering av en störd parameter i taget visar det sig att standardavvikelsen hela tiden får samma låga värde, men funktionen klarar inte av en störning på α_1 eller α_2 , d.v.s optimeringsrutinen hittar ett annat optimum. För övriga störda parametrar konvergerar optimeringen mot samma optimala värden som fatts med p . En störning på α_1 resp. α_2 resulterar alltså i att optimalt parametervärde för denna parameter blir ett annat, närmare störningen, samt att den andra vinkeloffseten förändras ungefär lika mycket och även θ_s justerar in sig efter detta värde.

Geometriskt betyder detta att hjulen och laserscannern justerar in sig och följer en rät bana fast vagnen går snett. Detta kan ställa till problem i praktiken vid t.ex en laststation som vagnen ska docka till.



Figur 20. Laserscannern och hjulen rör sig efter en rät bana, men vagnen åker snett.

6.1 Orsaker till vinkelparametrarnas injusteringar vid störning

Att förändra värdet för α_1 eller α_2 i startvektorn för optimeringen leder till att optimeringen konvergerar mot ett annat optimalt värde än tidigare för både α_1 , α_2 och θ_s . Förändringen av värdena är ungefär lika stor för alla tre parametrar, vilket, som beskrivits ovan, kan tolkas som att systemet m.a.p dessa tre vinklar har vrudit sig med en viss vinkel.

Antag att det lokala koordinatsystemets origo är placerat i laserscannern istället för i bakhjulet, då är (x_1, y_1) koordinater för bakhjulet och (x_2, y_2) koordinater för framhjulet. Laserscannerns hastighet, (v_x, v_y, ω) , kan då beräknas för varje tidpunkt m.a.p de encodervärden, $[v_1, u_1, v_2, u_2]$, som kommer från hjulen. Har nu följande hastighetsekvationerna:

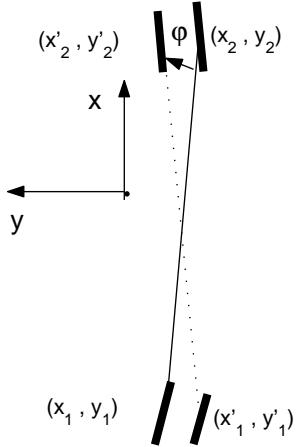
$$\begin{aligned} v_{x_1} &= d_1 v_1 \cos(u_1 + \alpha_1) - y_1 \omega \\ v_{y_1} &= d_1 v_1 \sin(u_1 + \alpha_1) + x_1 \omega \\ v_{x_2} &= d_2 v_2 \cos(u_2 + \alpha_2) - y_2 \omega \\ v_{y_2} &= d_2 v_2 \sin(u_2 + \alpha_2) - x_2 \omega \end{aligned} \quad (32)$$

och låter:

$$\begin{aligned} v_x &= \frac{v_{x_1} + v_{x_2}}{2} \\ v_y &= \frac{v_{y_1} + v_{y_2}}{2} \\ \omega &= \frac{d_2 v_2}{x_1 + x_2} \sin(u_2 + \alpha_2) - \frac{d_1 v_1}{x_1 + x_2} \sin(u_1 + \alpha_1) \end{aligned} \quad (33)$$

Det är dessa hastigheter, (v_x, v_y) , som transformeras till (v_X, v_Y) . Enligt rörelsemodellen integreras (v_X, v_Y, ω) för att få scannerns globala position. Utifrån denna position beräknas sedan vinklar till kända reflektorer enligt mätmodellen. När dessa jämförs med de vinklar laserscannern mäter fås residualen. På grund av detta är det enbart sådant som påverkar hastigheterna, (v_x, v_y, ω) , för laserscannern som påverkar residualfunktionen. Det som påverkar hastigheterna är encodervärdena och parametrarna.

Genom att visa att hastighetsskillnaden är väldigt liten mellan hastigheten som beräknas av indata och den hastighet som beräknas med de indata som borde fås då koordinatsystemet är roterat en liten vinkel, φ . Då kan man anta att en liten förändring i vinkeloffset, α_1 och α_2 , (ungefär lika för båda), inte påverkar laserscannerns hastighet. På så sätt påverkas inte heller den globala positionen och därfor konvergerar optimeringen mot dessa andra värden för α_1 och α_2 när startvektorn ändras m.a.p en av dessa. Även θ_s justerar in sig efter denna förändring.



Figur 21. Lokalt koordinatsystem med origo i laserscannern, samt hjulens positioner då systemet är roterat en liten vinkel φ .

Antag nu att alla parametervärden är konstanter så att: $\alpha_1 = \alpha_2 = 0$, $d_1 = d_2 = 1$, (x_1, y_1) och (x_2, y_2) har de koordinater man antar de är monterade vid. Om man nu har dessa beräknade hastigheter för en viss körning och sedan låter hjulen roteras en liten vinkel, φ , se figur 21, kan man ur ekvationssystemet:

$$\begin{aligned} v_{x_1} &= v'_1 \cos(u'_1) - y'_1 \omega \\ v_{y_1} &= v'_1 \sin(u'_1) + x'_1 \omega \\ v_{x_2} &= v'_2 \cos(u'_2) - y'_2 \omega \\ v_{y_2} &= v'_2 \sin(u'_2) - x'_2 \omega \end{aligned} \quad (34)$$

lösa ut vilka encodervärden, $[v'_1, u'_1, v'_2, u'_2]$, hjulen borde ge i sina nya positioner:

$$\begin{aligned} x'_1 &= x_1 \cos \varphi \\ y'_1 &= y_1 \sin \varphi \\ x'_2 &= x_2 \cos \varphi \\ y'_2 &= y_2 \sin \varphi \end{aligned} \quad (35)$$

Dessa nya encodervärden används nu för att beräkna laserscannerns hastighet, (v'_x, v'_y, ω') , m.a.p de gamla hjulpositionerna (x_1, y_1) och (x_2, y_2) . Detta görs för några av testkörningarna och då dessa nya hastigheter sedan jämförs med de ursprungliga hastigheterna, (v_x, v_y, ω) , ses att skillnaderna är små. Av detta kan man dra slutsatsen att en lika stor ändring av vinkeloffset för styrvinklarna inte påverkar laserscannerns globala hastighet. På grund av detta kan det vara svårt, (omöjligt?), att skatta båda vinkelparametrarna samtidigt.

7 Olika försök att komma ifrån injusteringen av vinkelparametrarna

7.1 Ändring av origo för det lokala koordinatsystemet

Placeringen av origo ändras för vagnens lokala koordinatsystem, istället för i bakhjulets centrum är det nu placerat i scannerns position. Lokala x-axeln pekar i laserns riktning/nolläge, d.v.s θ_s ingår inte längre som fri parameter, så gör ej heller x_s eller y_s som nu är noll. Nu skattas istället positionen för bak och framhjul, (x_1, y_1) resp. (x_2, y_2) , samt α_1 , α_2 , d_1 och d_2 som tidigare. Laserscannerns hastigheter, (v_x, v_y, ω) , i modellen beräknas nu från följande relationer:

$$\begin{aligned} v_{x_1} &= d_1 v_1 \cos(u_1 + \alpha_1) \cos(\theta_s) - y_1 \omega \\ v_{y_1} &= d_1 v_1 \sin(u_1 + \alpha_1) \cos(\theta_s) + x_1 \omega \\ v_{x_2} &= d_2 v_2 \cos(u_2 + \alpha_2) \cos(\theta_s) - y_2 \omega \\ v_{y_2} &= d_2 v_2 \sin(u_2 + \alpha_2) \cos(\theta_s) - x_2 \omega \end{aligned} \quad (36)$$

där:

$$\omega = \frac{d_2 v_2}{x_1 + x_2} \sin(u_2 + \alpha_2) \cos(\theta_s) - \frac{d_1 v_1}{x_1 + x_2} \sin(u_1 + \alpha_1) \cos(\theta_s) \quad (37)$$

På samma sätt som i den tidigare modellen används medelhastigheten, (2), av hastighetsekvationerna ovan för att beräkna laserscannerns x- och y-hastigheter.

Residualfunktionen med det ingående Kalmanfiltret optimeras för denna ändrade modell. Ingen konvergering sker med de satta avbrottsskriterierna, (Bilaga 9). Först vill rutinen att MaxFunEvals ska utökas, detta villkor sätts då till att vara 6000 evalueringar. Inte heller detta ger någon konvergens utan rutinen då vill att MaxIter utökas, vilket görs till 4000 iterationer. Efter väldigt lång tid utan att optimeringen konvergerar väljs att avbryta. Slutsatsen blir att denna modell inte är lämplig att optimera.

7.2 Ny avståndspараметer

Efter återgång till det gamla koordinatsystemet undersöks först hur villkoret $v_{x_1}=v_{x_2}$, sid. 9, uppfylls för de olika fallen a och b med denna modell. Först används p , ursprungsparametrarna och sedan de optimala parametrarna. Villkoret $v_{y_1} = v_{y_2}$ är automatiskt uppfyllt p.g.a valet av ω , (3). Undersökningarna visar att villkoret för v_x inte uppfylls, maximala skillnaden

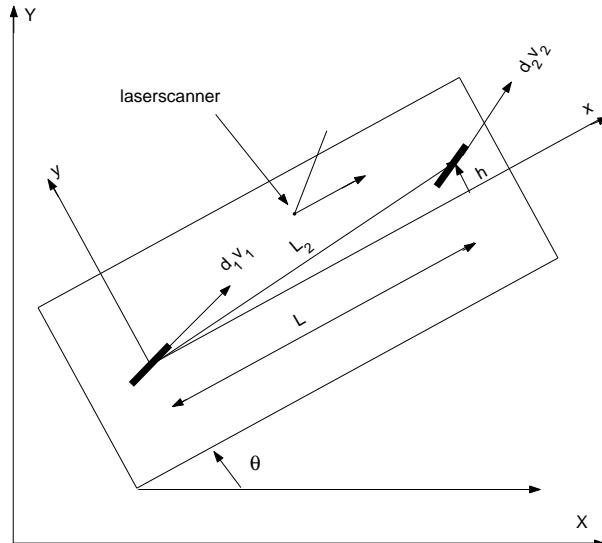
är 0.0570 m/s i fall a och 0.0480 m/s i fall b. Med optimala parametrar för fall a blir samma skillnad 0.1150 m/s och detsamma för fall b ger maximala skillnad 0.1314 m/s.

För att villkoret ska vara uppfyllt används d_2 som skalfaktor, detta innebär att en parameter/frihetsgrad försvinner.

$$d_2 = \frac{d_1 v_1 \cos(u_1 + \alpha_1)}{v_2 \cos(u_2 + \alpha_2)} \quad (38)$$

Nu blir istället max skillnaden i v_x 0.0030 m/s för fall a och 0.0087 m/s för fall b med p , med optimala parametrar blir detsamma 0.0030 m/s för fall a och 0.0098 m/s för fall b.

En ny parameter, h , införs som betecknar avståndet mellan den lokala x-axeln och framhjulets placering. Detta kommer då att påverka ω , (3), på så sätt att L ersätts med L_2 , se figur 22.



Figur 22. Modell med ny parameter h , där h är framhjulets avvikelse från lokala x-axeln.

För att h ska kunna skattas måste α_1 låsas, ty annars finns oändligt antal lösningar. Att låsa α_1 ställer en del krav på bakhjulets första injustering. Under antagandet att denna injustering är tillräckligt bra sätts $\alpha_1 = 0$ och h 's påverkan på residualfunktionen undersöks. Först approximeras derivatan enl. differensmetoden p.s.s som tidigare gjorts för övriga parametrar. Denna approximerade derivatan är väldigt liten.

Olika värden för h ansätts nu och residualfunktionen optimeras m.a.p parametrarna $\alpha_2, d_1, L, x_s, y_s$ och θ_s för dessa olika värdena på h . I alla försök går parametrarna mot samma optimala värde de hade då $h = 0$. Det visar

sig att h kan vara upp till 50 cm utan att residualen påverkas något, detta är då inget problem eftersom man i praktiken kan mäta in avståndet med högst 10 cm osäkerhet.

Slutsats som man kan dra är att hänsyn inte behöver tas till eventuell avvikelse av framhjulet från lokala x-axel.

7.3 Olika körmönster

Några nya testkörningar utförs, eftersom vagnsdatorn är utbytt sedan de förra, (fall a och b), är startvärdena för parametrarna något förändrade. Syftet med dessa nya körningar är att se om det finns något speciellt körmönster så att beroendet mellan vinklarna kan fångas. Först prövas att köra i en cirkel sedan i spiral, d.v.s cirkelmönster med ökande radie, och därefter i en rektangel. Residualen för rektangelkörningen liknar den som fästs för det tidigare fall b, kan alltså ses som ett allmänt körmönster där de flesta manövreringar finns med. Även parametrarnas störningskänslighet uppför sig på samma sätt för denna rektangelkörning.

Parametrarnas, $\alpha_1, \alpha_2, d_1, L, x_s, y_s$ och θ_s , uppförande på residualen för spiralbanorna, både höger och vänstervarv undersöks. Först optimeras residualfunktionen med startvektorn $p = [0, 0, 1, 0.690, 0.545, 0, 4.7124]$, och sedan läggs en störning på var och en av parametrarna. p är nu ett element mindre eftersom d_2 är borttaget som fri parameter. Jämförelserna visar att det fortfarande finns ett beroende mellan α_1 och α_2 , de justerar fortfarande in sig efter varandra då en av parametrarna störs. θ_s påverkas inte av denna injustering då vagnen körs i cirklar, detta till skillnad mot föregående testkörningar, (fall a och b). Däremot L , hjulavståndet, blir störningskänslig vid detta körmönster. En förändring i L påverkar både optimalt värde för L samt α_1 och α_2 . Slutsatsen blir att parametrarnas beroende inte kan hittas/slås isär med hjälp av något speciellt körmönster.

7.4 Låsning av en vinkeloffsetparameter

För att injusteringen av vinklarna inte ska ske läses en av vinkeloffseten i taget och residualfunktionens beteende undersöks utifrån detta.

Först sätts $\alpha_2 = 0$ och residualfunktionen för fall a optimeras med p , där α_2 är borttaget och sedan med en störning på α_1 , (0.09 rad). De optimala värdena för alla fria parametrar går då mot samma optimum som utan störningen, skillnaden i α_1 och θ_s är ca. 4 mrad. Därefter optimeras residualen för fall b p.s.s som ovan, störningen ger även här parametrar som går mot samma optimala värden som utan störning, skillnaden i α_1 är 1.4 mrad och för θ_s är det ingen skillnad.

α_2 får nu vara fri och istället sätts $\alpha_1=0$. Samma undersökning som gjordes ovan görs nu för denna situation. För fall a blir skillnaden i optimalt värde för α_2 utan och med störning 64 mrad och för θ_s 20 mrad. För fall b blir samma skillnad ca 15 mrad för α_2 och 12 mrad för θ_s . Störningar i θ_s har, som tidigare försök visat, ingen påverkan på optimala värden.

Varför är α_2 mer störningskänslig än α_1 ? Som ses i formel (1), ingår de inte på exakt samma sätt i modellen även om första tanken är att de gör så. Detta skulle kunna vara orsaken till att de inte uppför sig exakt lika.

För att komma ifrån vinkelparametrarnas injustering väljs nu att låsa vinkeloffset för framhjulet, α_2 . På så sätt har ytterligare en parametrar som ska skattas reducerats. De fria parametrarna som man vill skatta är nu $\alpha_1, d_1, L, x_s, y_s$ och θ_s .

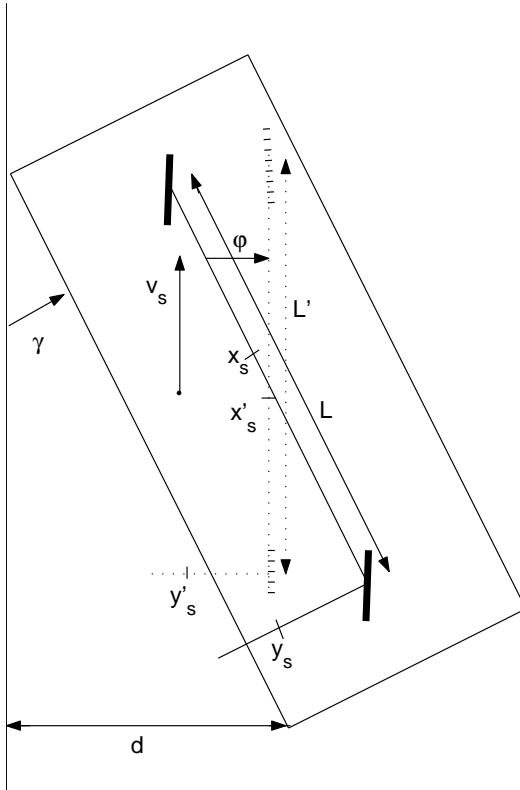
7.5 Efterjustering av parametrarna

Att låsa α_2 , d.v.s anta att $\alpha_2=0$, innebär att man läser en eventuell felinställning som finns för framhjulet från början. P.g.a vad som framkommit i undersökningarna av parametrarnas uppträdande kan man anta att α_1 och θ_s justerar in sig efter detta begynnelsefel. Detta kan då i sin tur ge upphov till att vagnen rör sig snett, se avsnitt 6, sid. 36. För att rätta till detta, d.v.s få vagnen att gå rakt, behövs en efterjustering av parametrarna göras. Med uttrycket att få vagnen att gå rakt menas att vagnens lokala x-axel är parallell till den räta linje vagnen kör efter.

Vagnen körs nu längs en rät linje och avståndet mellan vagnens framrespektive bakkant vid en viss punkt mäts. På detta sätt fås en s.k vridningsvinkel, φ , som visar hur snett vagnen rör sig när lasern följer en rät linje. d betecknar avståndet mellan fram- och bakkant vid en viss punkt och D står för längden mellan fram- och bakkant, se figur 23.

$$\varphi = \arcsin \left(\frac{d}{D} \right) \quad (39)$$

Man kan anta att x_s, y_s och L har justerat in sig efter ett koordinatsystem vridet denna vinkel φ kring mittpunkten mellan hjulen, vilket skulle vara som att vagnen gick rakt.



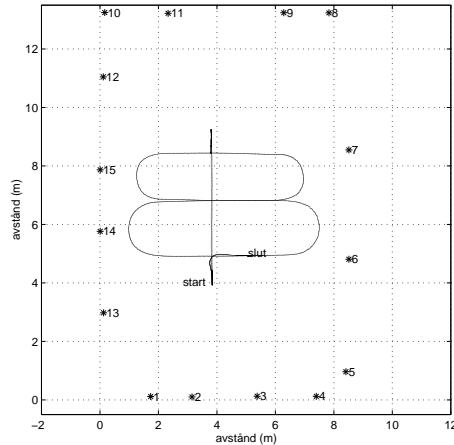
Figur 23. Vagnens vridning φ . Streckade linjer avser det vridna koordinatsystemet som x_s, y_s och L antagits justerat in sig efter.

För att få vagnen att gå rakt bör följande efterjusteringar göras:

$$\begin{aligned}
 \alpha'_1 &= \alpha_1 - \varphi \\
 \alpha'_2 &= -\varphi \\
 L' &= \frac{L}{\cos \varphi} \\
 x'_s &= x_s \cos \varphi - y_s \sin \varphi \\
 y'_s &= x_s \sin \varphi + y_s \cos \varphi \\
 \theta'_s &= \theta_s - \varphi
 \end{aligned} \tag{40}$$

7.6 Testkörningar

En layout på en allmän bana, figur 24, som ska täcka in de flesta tänkta manövreringar görs, denna körs med samma testvagn som använts tidigare.



Figur 24. Layout för allmän bana. Kör först fram och tillbaka och sedan i en åtta.

Residualen för denna bana beräknas och optimeras därefter m.a.p α_1 , d_1 , L , x_s , y_s och θ_s . De optimala värden för dessa parametrar skickas nu till vagnsdatorn, därefter körs vagnen längs en rät linje. Markeringar för framrespektive bakkant på vagnen görs vid en viss punkt i golvet och efterjustering av parametrarna beräknas enligt det sätt som beskrivits ovan. Detta visar sig fungera bra och vagnen 'rätar' på sig, vilket ses då den körs längs den räta banan än en gång.

Av detta kan slutsats dras att metoden verkar fungera bra.

7.7 Störda parametrar i verkligheten

Den vagn som används vid testkörningarna är redan någorlunda bra intrimmad. För att se vad som händer om den inte vore intrimmad ändras värdet på varje fri parameter ett i taget i vagnsdatorn och vagnen körs samma bana som ovan, figur 24. För varje körning beräknas först vilken reflektor som ska jämföras med i varje tidpunkt om parametervärdena varit de antagna, $p = [0, 1, 0.690, 0.545, 0, 4.7124]$, där nu $p = [\alpha_1, d_1, L, x_s, y_s, \theta_s]$. Därefter beräknas och optimeras residualfunktionen. De optimala parametervärdenen jämförs med de som fåtts tidigare för att se om värden blir ungefärliga.

Förändring av en parameters värde i vagnsdatorn ger fel i utdata från encodrar och laserscanner, detta motsvarar situationerna då hjul och scanner avviker från den perfekta monteringen. En sådan montering innebär t.ex. att hjulen är parallella med vagnen, pekar i lokala x-axelns riktning, att hjulavstånd är exakt uppmätt och lasersens lokala position är exakt bestämd. Detta är något som i praktiken är omöjligt att utföra.

För α_1 prövas först att lägga till 0.09 rad $\approx 5^\circ$ och sedan 0.05 rad $\approx 3^\circ$. En optimering av första fallet ger en betydligt större standardavvikelse än

tidigare och speciellt θ_s ' optimala värde avviker mycket från det ursprungliga. Däremot det andra fallet konvergerar mot samma optimala värde för parametrarna och standaravvikelsen blir låg.

d_1 förändras med 10% och efter att banan körts och residualfunktionen optimerats fås konvergens mot samma optimum.

Värdet för parametrarna L, x_s och y_s ändras alla med 10 cm och samma beräkningar som ovan ger för varje optimering ungefär samma optimala parametervärden som tidigare.

θ_s förändras precis som α_1 med att först lägga till 0.09 rad $\approx 5^\circ$ och sedan 0.05 rad $\approx 3^\circ$. I första fallet kan man här inte hitta tillräckligt många reflektorer som skulle användas för att beräkna residualen om θ_s inte vore störd, så inga beräkningar enligt det här sättet går att utföra. Med en störning på 3° , går det däremot bra och optimeringen ger liknande optimala värden som ursprungligen fästs.

Slutsatsen är att optimering enligt det sätt som används ovan fungerar om hjulens montering inte avviker mer än 3° från lokala x-axeln, att hastighetsskalningen inte avviker med mer än 10%, att uppmätt hjulavstånd och scannerns lokala position inte skiljer på mer än 10 cm och till sist att monteringen av startläget för laserscannern inte avviker med mer än 3° från antagit startläge.

Detta är i praktiken inga problem utan kan uppfyllas relativt enkelt.

8 Resultat och Slutsatser

Syftet med detta arbete är att undersöka vilka fysiska parametrar för en typ av fyrfjädrad AGV som går att bestämma på ett tillförlitligt sätt. Dessa parametrar påverkar vagnens navigering under körning. Utifrån dessa undersökningar ska sedan en metod tas fram så att parametrarnas värde bestäms på ett optimalt sätt. Metoden måste uppfylla villkoret att navigeringen lyckas, d.v.s associeringsalgoriterna måste kunna associera tillräckligt många vinkelmätningar. Dessutom ska vagnen gå rakt, d.v.s den lokala x-axeln på vagnen måste vara parallell med den räta bana vagnen följer. Detta för att det inte ska bli problem vid eventuella arbetsstationer vagnen ska stanna vid.

De parametrar som avses är: α_1 , α_2 , d_1 , d_2 , L , x_s , y_s och θ_s , (se sid 11) och dessa samlas i en vektor p . Navigeringen utförs utifrån indata som fås från hjulencodrar och laserscanner beträffande styrvinkelar och hastighet för hjulen samt vinkelmätningar till reflektorer med kända positioner.

En modell av vagnens navigering m.a.p indata från testkörningar görs i MATLAB, denna första variant tar bara hänsyn till encodervärden vid uppdatering av positionen. På detta sätt kan de vinklar, $\gamma(p)$, laserscannern borde mäta till reflektor b beräknas för varje tidpunkt, k , som indata från encodrarna fås. Dessa beräknade vinklar jämförs med de vinklar, β , som fås som indata från laserscannern, detta ger en vektorvärd residualfunktion, r , som beror av p där varje element är vinkelskillnaden, $(\beta - \gamma(p))$, vid tidpunkt k .

8.1 När är parametrarna observerbara och vilken påverkan har de

Undersökningar av hur residualfunktionen påverkas av respektive parameter utförs för olika testkörningar, bl.a en rak bana utan svängar och en kurvig bana. Genom dessa undersökningar kan man då få en uppfattning om vilket körmönster vagnen måste ha för att en parameter ska vara observerbar. Även hur stor en eventuell påverkan på residualfunktionen är samt om flera parametrar har liknande påverkan.

Metoder för dessa undersökningar är t.ex att för en körning ändra ett parametervärde i taget och plotta de olika residualvektorerna som fås. Plotterna kan sedan jämföras och därigenom ge en bild av hur residualfunktionen förändras m.a.p aktuell parameter. Man kan också se om plotterna liknar varandra m.a.p olika parametrar och på det sättet se om det finns ett beroende mellan parametrar. En annan metod är att approximera derivatorna för residualfunktionen m.a.p de olika parametrarna. Dessa derivator kan

då också plottas och ge en bild av parametrarnas påverkan på residualfunktionen.

Undersökningarna enligt dessa metoder leder till resultaten: α_1 , α_2 , d_1 , d_2 och θ_s är observerbara för alla körsätt som prövas. För att det ska vara möjligt att skatta L , x_s och y_s måste vagnen längre omvänt svänga. Påverkan av x_s och y_s på residualfunktionen är inte speciellt stor även då vagnen svänger, för L längre omvänt ser den ut att vara större. Undersökningarna leder också till iakttagelsen att det finns ett beroende mellan parametrarna α_1 och α_2 . Genom att förändra den ena parametern får man samma påverkan på residualfunktionen som om man gör samma förändring med ombytt tecken på den andra parametern. Ett beroende mellan d_1 och d_2 visar sig också finnas, förändras d_1 har det samma påverkan som att förändra d_2 med samma värde. θ_s ser ut att ha en linjär påverkan på residualfunktionen då vagnen kör rakt.

8.2 Optimering på olika sätt

Den målfunktion som minimeras är $\sum_{k=1}^n ((\beta - \gamma(p))(k))^2$. Då residualfunktionen plottas med de parametervärden som optimeringen konvergerar mot ses att värdena för elementen i vektorn är större än vad som är acceptabelt, (sid 24). Försök utförs genom att dela upp den körlägena i delsekvenser m.a.p om hjulen åker framåt eller bakåt och optimera dessa delsträckor. Detta ger ett bättre resultat vilket skulle kunna bero på att det är olika parameteruppsättningar, (styrvinkeloffset), ifall vagnen kör framåt eller bakåt. Det är något som skulle kunna undersökas grundligare, men inget som hänvisar till detta arbete.

Optimering prövas också genom uppdelning i olika etapper på så sätt att först använda en rak körning och optimera residualfunktionen m.a.p de parametrar som är observerbara för denna. Sedan används dessa som konstanter då en kurvig bana optimeras m.a.p resterande parametrar. Detta ger inga förbättrade resultat och att dela upp optimeringen kan leda till att viss information förbises. Så det anses inte vara något alternativ.

Parametrarna som fästs ifrån optimeringen av den hela banan används i ett utvärderingsprogram som ska efterlikna verkligheten. I detta program används gjorda vinkelmätningar från laserscannern av ett Kalmanfilter för att uppdatera positionen. Detta ger betydligt mindre värden för residualfunktionen och valet är att utöka modellen för navigering med detta filter. På detta sätt förändras beräkningen av $\gamma(p)$ och då även residualfunktionen samt i slutändan målfunktionen som optimeras.

Optimering av denna nya målfunktion konvergerar mot parametrar som ger mycket bra värden för elementen i residualfunktionen. Undersökningar utförs för att se hur stabil optimeringen är, detta genom att förändra startvekt

torn m.a.p en parameter i taget. Resultatet är att förändringar i startvärdet för d_1 , d_2 , L , x_s , y_s och θ_s ger konvergens mot samma värden som med ursprunglig startvektor. Föändras dock α_1 eller α_2 leder detta till att optimeringen konvergerar mot andra värden för α_1 , α_2 och θ_s . Detta samtidigt som standardavvikelsen antar samma låga värde som för optimering utan störning. Skillnaderna mellan dessa nya parametervärden och de som fästs med den ursprungliga startvektorn är ungefär lika stora för alla vinkelparametrar. Detta kan då tolkas som att hela koordinatsystemet har vridit sig en viss vinkel φ , som då skulle vara denna skillnad. Resultatet av detta blir att vagnen är vriden då den följer en rät bana, vilket kan ge problem vid dockning.

En förklaring till att det finns olika optimum beroende på vilken startvektor som används m.a.p α_1 eller α_2 , är att en liten vridning av koordinatsystemet inte förändrar hastigheten för laserscannern och på detta sätt inte påverkar navigeringen. Beräkningar och jämförelser av hastigheterna m.a.p en sådan liten vridning görs för några olika testkörningar och resulterar i en liten skillnad mellan hastigheterna. Men om detta är hela sanningen är osäkert, eftersom då konvergens för den förra målfunktionen, (utan filter), prövades visade den sig konvergera mot samma värden för alla förändrade startvektorer. Detta skulle vara intressant att undersöka vidare.

Olika undersökningar utförs för att se om denna konvergens, för målfunktionen med ingående Kalmanfilter, mot olika optimum för olika startvektorer kan förhindras. T.ex prövas att köra olika speciella körmönster, bl.a cirklar, spiraler och rektanglar. Därefter optimeras de erhållna residualfunktionerna för dessa körningar, först med ursprunglig startvektor och sedan med störd startvektor. Tendenserna är fortfarande likartade och slutsatsen är att konvergensen mot olika optimum inte kan förhindras på detta sätt. Ett annat sätt som undersöks är att ändra origo för det lokala koordinatsystemet. Istället för att vara placerad i bakhjulets mitt är origo i laserscannerns position. Då residualfunktionen, m.a.p denna modell, optimeras fås ingen konvergens under väldigt lång tid. Detta resulterar i att optimeringen avbryts och slutsatsen blir att denna modell inte är lämpad för optimering.

Om istället optimering sker med en av parametrarna α_1 eller α_2 låsta, och konvergens undersöks m.a.p olika startvärden blir resultatet att optimeringen konvergerar mot liknande värden då α_1 är förändrad, men inte lika bra då α_2 förändras. Denna skillnad i parametrarnas uppförande här kan bero på att parametrarna kommer in på lite olika sätt i hastighetsekvationerna, (1), som används för att beräkna laserscannerns globala hastighet. Vad som skulle kunna undersökas mer är denna konvergens m.a.p andra sätt att beräkna hastigheten istället för medelhastigheten som används här.

8.3 Vilka parametrar går att skatta

Ytterligare en parameter, h , som får beteckna framhjulets eventuella avstånd från lokala x-axeln införs. För att denna parameter ska kunna skattas måste α_1 vara låst, annars finns oändligt antal lösningar. Antag att α_1 är låst trots att det är ett sämre alternativ enligt ovan resultat. Nu undersöks h 's påverkan på residualfunktionen på liknande sätt som de övriga parametrarnas. Resultatet blir att ett sådant avstånd kan vara ganska stor, upp till 50 cm, utan att residualfunktionen påverkas. Slutsatsen är att en sådan parameter inte behöver tas med.

Ovan resultatet visar att inte båda parametrarna α_1 och α_2 kan skattas samtidigt på ett bra sätt. Därför väljs α_2 till att vara låst, och på detta sätt har antalet parametrar som ska skattas reducerats.

Det villkor som finns för hastigheterna, (sid 9), uppfylls inte automatiskt vid optimering och för att det ska bli uppfyllt används d_2 som skalfaktor och beräknas enligt $d_2 = \frac{d_1 v_1 \cos(u_1 + \alpha_1)}{v_2 \cos(u_2 + \alpha_2)}$. Nu har ytterligare en parameter försvunnit från de som ska skattas.

Kvar att skatta är parametrarna: α_1 , d_1 , L , x_s , y_s och θ_s . För att dessa ska vara observerbara måste man köra en bana som innehåller svängar. För att optimeringen ska bli så bra som möjligt bör banan innehålla både raksträckor samt höger- och vänstersvängar. Körfördelningen m.a.p svängar och raksträckor bör vara ungefär lika.

Genom att förändra värden i vagnsdatorn för en parameter i taget och köra en bana fås annan indata från encodrar och laserscanner. Därefter optimeras residualfunktionen beräknad med dessa värden. På så sätt undersöks hur stort fel i hjulen och laserscannerns placering relativt den antagna placeringen optimeringen klarar av. Man vill att optimeringsrutinen ska konvergera mot samma optimum som fås då vagnen redan är relativt bra intrimmad. Resultatet blir följande villkor för att parametrarna ska gå att skatta: hjulens montering relativt den lokala x-axeln får inte avviken med mer än 3° och hjulavståndet ska vara uppmätt inom 10 cm noggrannhet. Hastighetskalningen ska vara bestämd med 10% säkerhet. Laserscannerns placering på vagnen ska vara inmätt med 10 cm noggrannhet och vinkeln θ_s' antagna värde relativt lokala x-axeln får ej avvika med mer än 3° . Dessa villkor kan relativt enkelt uppfyllas.

8.4 Korrigering av parametrar

En lösning av α_2 leder till att ett eventuellt vinkelfel i monteringen relativt lokala x-axlen följer med genom optimeringen. Detta eftersom vinklarna justerar in sig efter varandra på det sätt som beskrivs ovan då konvergens

undersöks. De parametervärden som fås i optimeringen måste därför efterjusteras med avseende på detta fel. Kalla detta vinkelfel φ .

Resultatet är då att parametrarnas värden konvergerar mot optimala värden relativt ett koordinatsystem som är vridet med vinkelns φ relativt det man har. Detta ger följande efterjustering som behövs göras: $\alpha'_1 = \alpha_1 - \varphi$, $\alpha'_2 = -\varphi$, $L' = \frac{L}{\cos \varphi}$, $x'_s = x_s \cos \varphi - y_s \sin \varphi$, $y'_s = x_s \sin \varphi + y_s \cos \varphi$ och $\theta'_s = \theta_s - \varphi$. För att få detta vinkelfel, φ , mäts avstånd, d , mellan fram och bakkant till en rät linje och på sätt lösas φ ut. $\varphi = \arcsin\left(\frac{d}{D}\right)$, D är längden mellan fram- och bakkant.

8.5 Testkörning

En bana som innehåller en raksträcka samt går i en åtta körs med en testvagn, data från encodrar och laserscanner loggas. Med dessa data beräknas sedan residualfunktionen och optimeras m.a.p $\alpha_1, d_1, L, x_s, y_s$ och θ_s . Detta ger en residualfunktion med mycket små värden för respektive komponent. Efter att parametervärdena i vagnsdatorn ändras till de som fåtts i optimeringen körs en raksträcka. Vid en punkt i golvet markeras position för fram- respektive bakkant av vagnen och vinklen φ beräknas. Efterjusteringar av parametrarnas värden i vagnsdatorn sker enligt ovan och en raksträcka körs igen. Denna gång är det i princip ingen avvikelse mellan positionerna för fram- och bakkant på vagnen.

Slutsatsen är att parametrarna $\alpha_1, d_1, L, x_s, y_s$ och θ_s går att bestämma på ett tillförlitligt sätt. Om optimering görs på det sätt som beskrivits ovan konvergerar dessa mot parametervärden som ger väldigt små värden för residualfunktionen då denna beräknas. Detta kan då tolkas som att navigeringen fungerar väldigt bra. Dessutom vet man att vagnen rör sig rakt då den följer en rak bana.

8.6 Metod för identifiering av parametrar

1. Var säker på att villkoren som gäller hjulen och scannerens montering, beskrivna ovan, är uppfyllda.
2. Beordra vagnen att köra från en punkt till en annan, mät den sträcka som vagnen faktiskt körde och jämför med avståndet mellan punkterna. På detta sätt kan hastighetsskalningen bestämmas inom ovan angivna gräns, (10%).
3. Kör en allmän bana som innehåller de flesta tänkbara manövreringar, liknande den som används vid testkörningarna på sid.43. Beräkna

därefter residualen och optimera denna m.a.p parametrarna α_1 , d_1 , L , x_s , y_s och θ_s .

4. Sätt dessa parametrars värden i vagnsdatorn till de optimala värden som fästs vid optimeringen. Kör efter en rät linje. Vid en punkt i golvet, som vagnen passerar, markeras positionen för framkant respektive bakkant.
5. Beräkna vilken vridningsvinkel, φ , vagnen har m.h.a avståndet mellan markeringar gjorda i punkt 4. Gör efterjusteringar av parametrarna enligt formel (40). Nu är vagnen bra intrimmad.

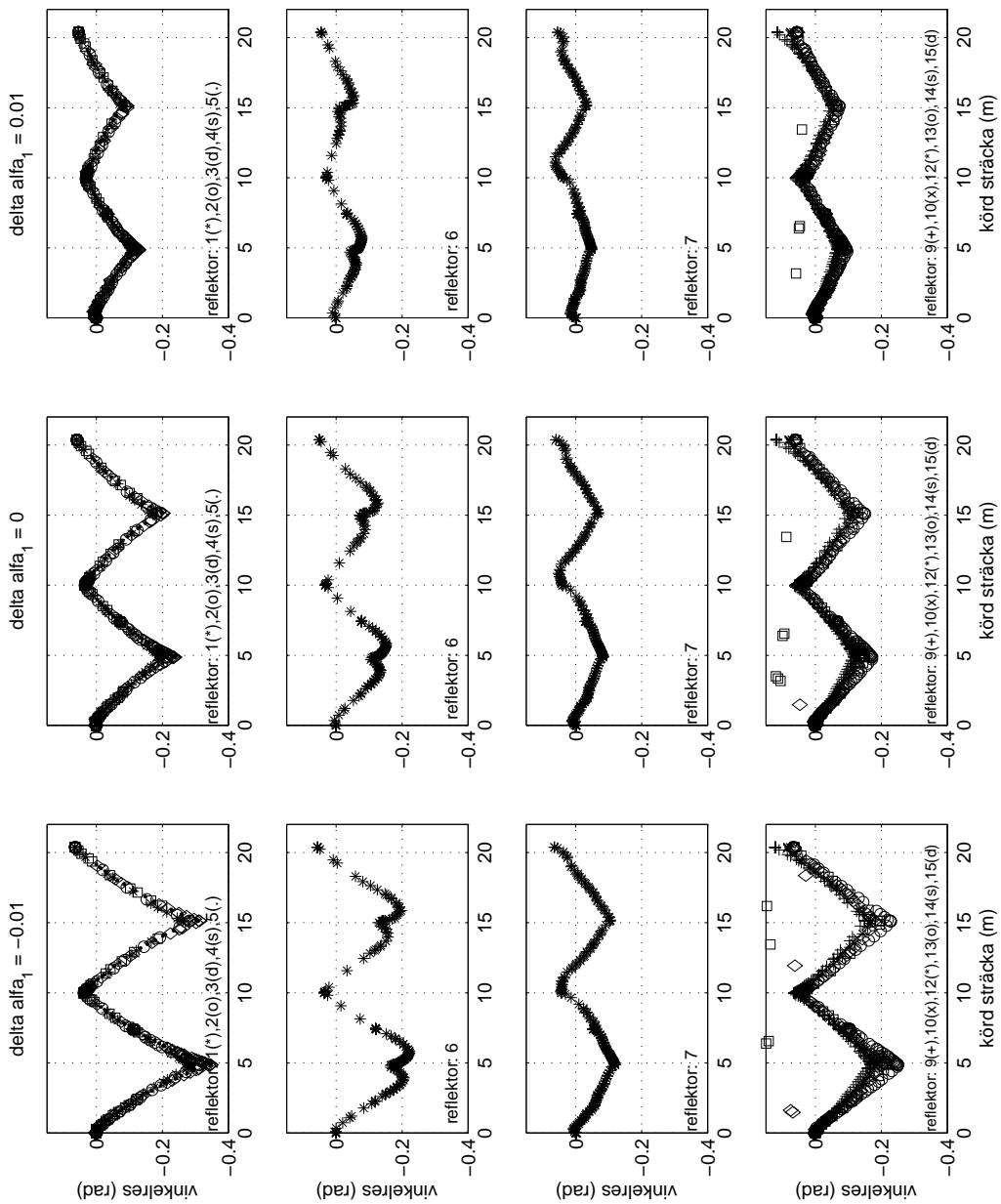
Referenser

- [1] <http://www.mathworks.com/>: Hemsida med information m.m. angående MATLAB
- [2] <http://www.wolfram.com/>: Hemsida med information m.m. angående Mathematica
- [3] Gunnar Blom: *Sannolikhetsteori och statistikteori med tillämpningar*. Studentlitteratur, Lund 1989.
- [4] Michael Bohman: *Identifiering av vagnsparametrar för AGV*. Projekt #7174 Netzler & Dahlgren Co AB.
- [5] Robert Grover Brown, Patrick Y.C. Hwang: *Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions -3rd edition*. Library of Congress Cataloging in Publication Data, USA 1997.
- [6] C.K.Chui, G.Chen: *Kalman Filtering with Real-Time Applications*. Springer-Verlag, Berlin Heidelberg 1987.
- [7] Michael T. Heath: *Scientific Computing: An Introductory Survey*. McGraw-Hill Companies, Inc. Singapore 1997.
- [8] Miller Leskiw: *An Introduction to Kalman Filtering with applications*. Robert E. Krieger Publishing Company Malabar, Florida 1987.
- [9] Stephen G. Nash, Ariela Sofer: *Linear and Nonlinear Programming*. McGraw-Hill Companies, Inc. Singapore 1996.
- [10] Robert Olsson: *Vehicle Start-Up, Method Description*. Netzler & Dahlgren Co AB , Särö 2000.
- [11] Magnus Oskarsson: *One-dimensional retina vision with applications in structure and motion estimation*. Centre for Mathematical Sciences Lund University, Sweden 2000.
- [12] Henrik Stewenius: *Automatic calibration of vehicle parameters from visual data*. Centre for Mathematical Sciences Lund University, Sweden 2003.
- [13] Henrik Stewenius: *Studies in mobile camera calibration*. Centre for Mathematical Sciences Lund University, Sweden 2003.

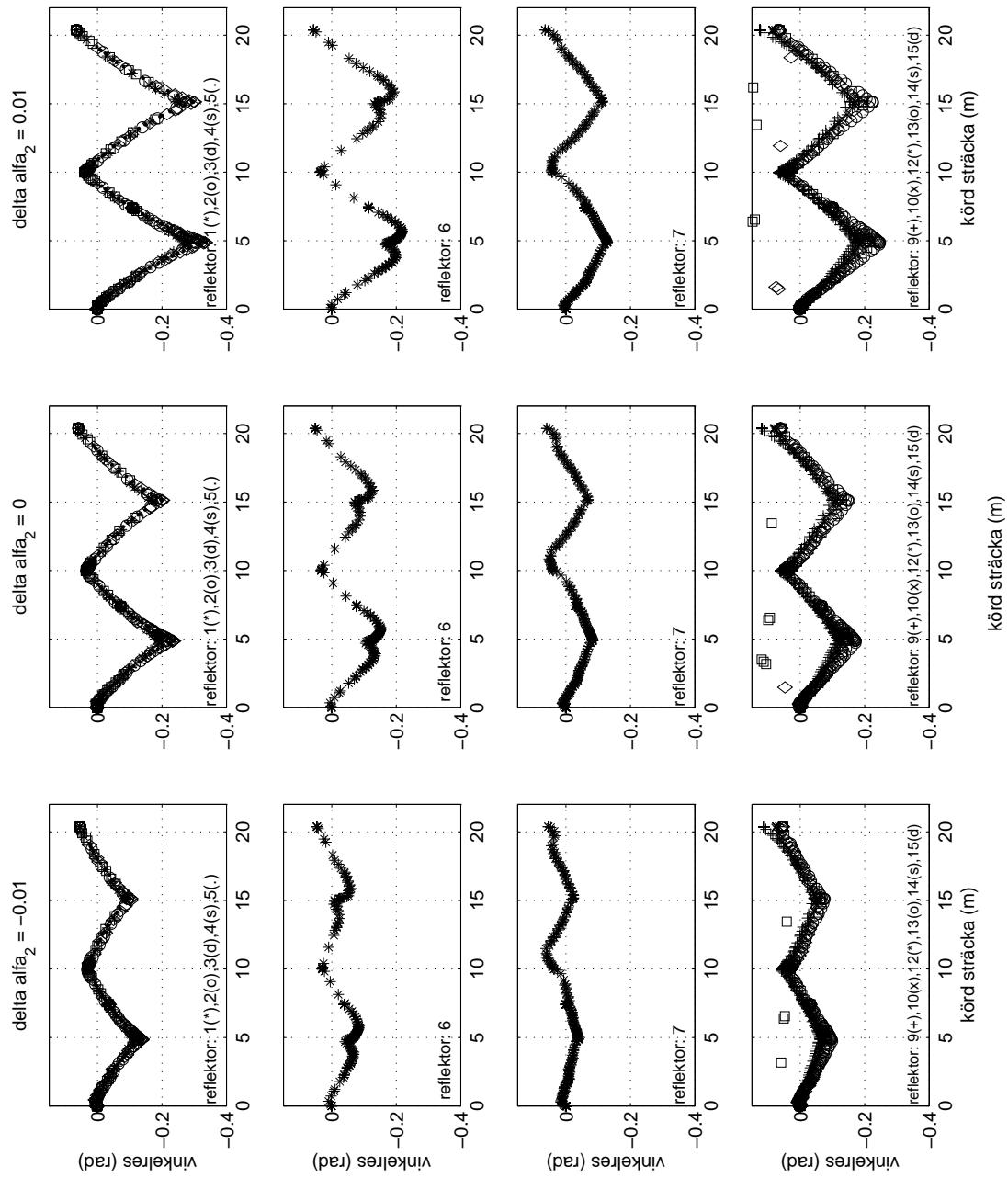
- [14] Urban Wiklund: *Algorithms for Navigation of Autonomous Guided Vehicles based on measurements of directions to identical beacons, PARTH II, Recursive estimation of the trajectory.*
Division of Industrial Electronics and Division of Automatic Control
Luleå University of Technology S-951 87 Luleå, Sweden.
- [15] Vehicle Maintenance Program,(VMP): Användarmanual för automatisk kallibrering av trehjuling, NDC AB 1998

Bilagor

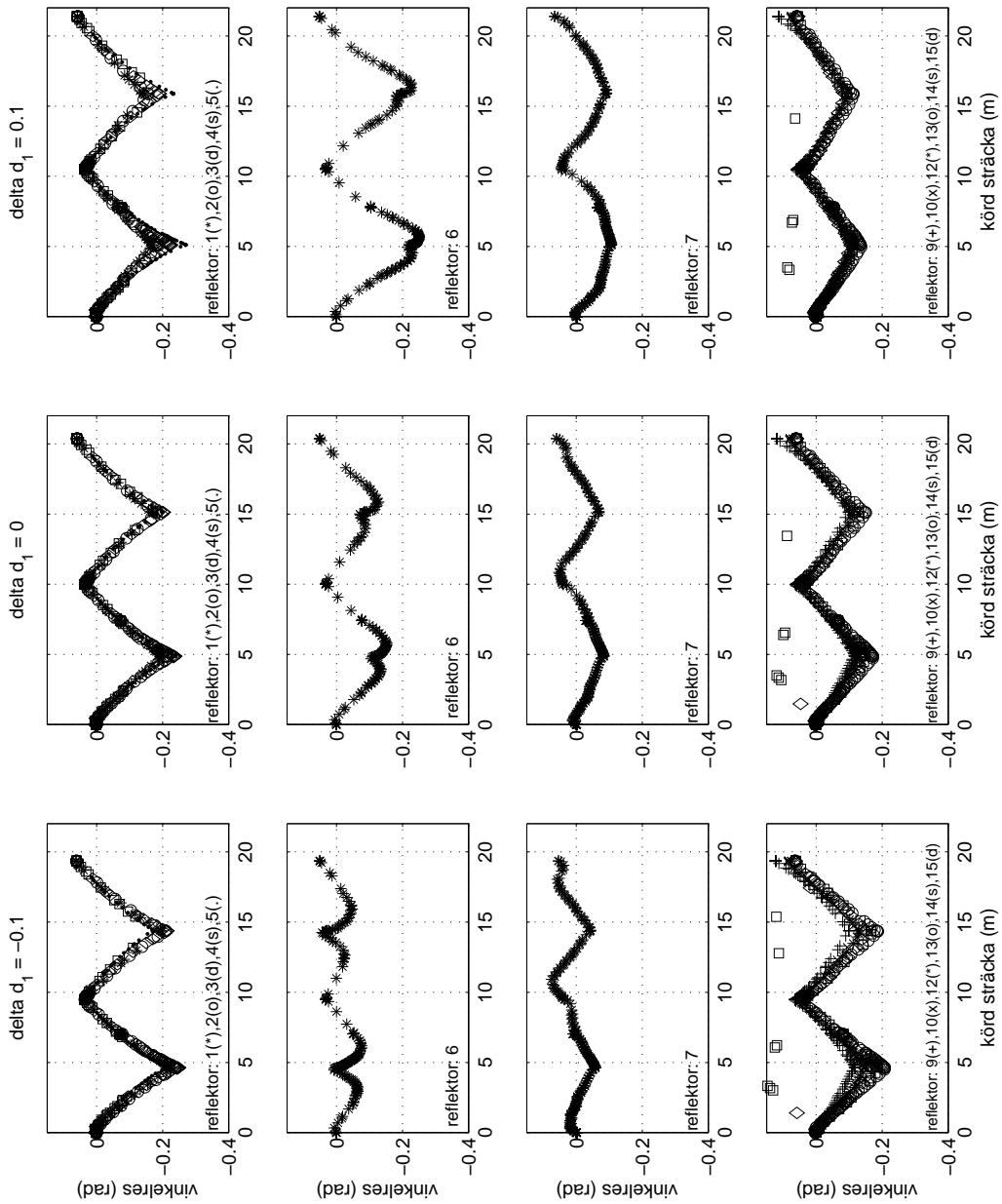
Bilaga 1. Residualfunktionen plottad m.a.p olika värden
på α_1 , övriga parametrar är konstanta



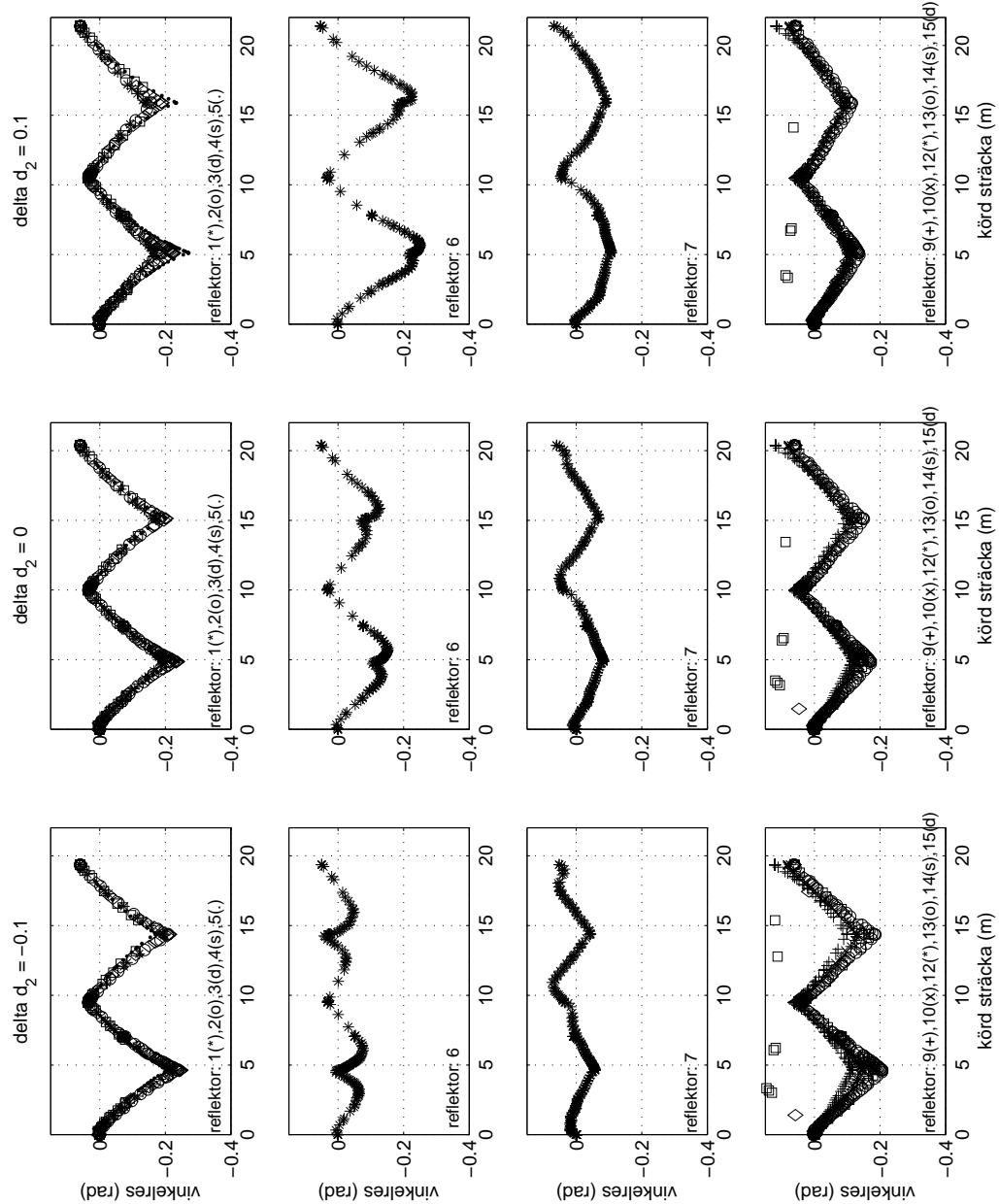
Bilaga 2. Residualfunktionen plottad m.a.p olika värden på α_2 , övriga parametrar är konstanta



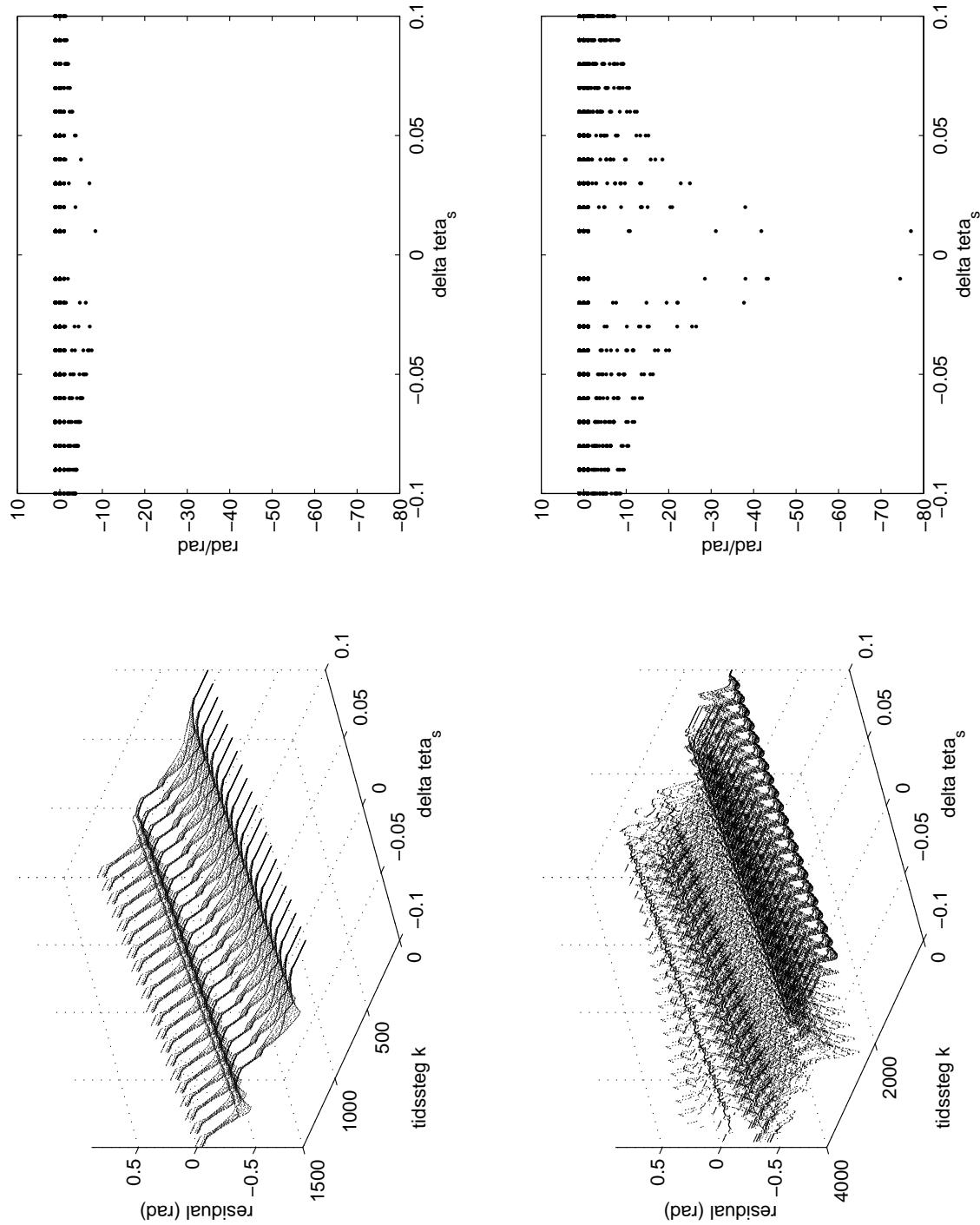
**Bilaga 3. Residualfunktionen plottad m.a.p olika värden
på d_1 , övriga parametrar är konstanta**



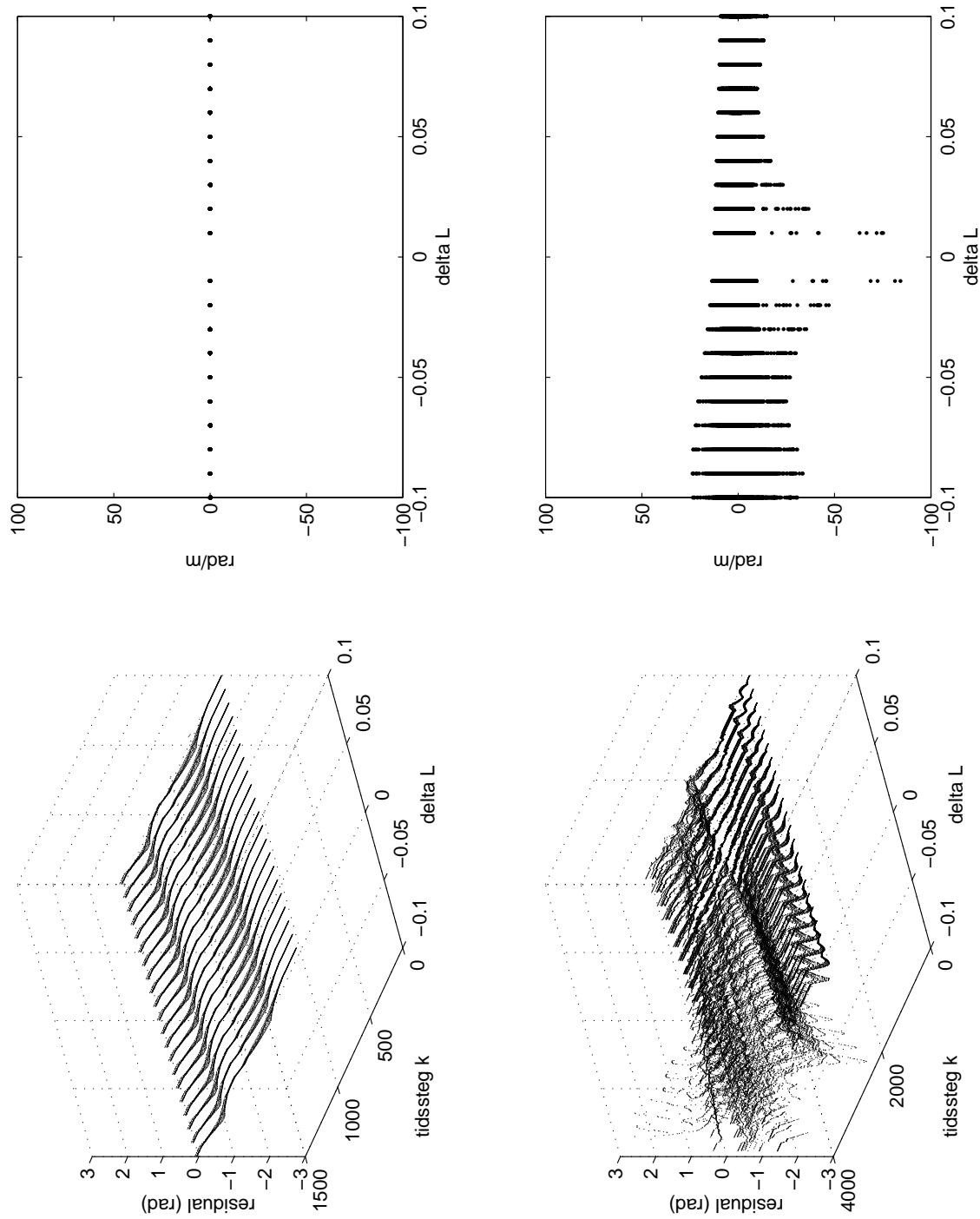
**Bilaga 4. Residualfunktionen plottad m.a.p olika värden
på d_2 , övriga parametrar är konstanta**



**Bilaga 5. Residualytör och approximerade derivator m.a.p
olika värden på θ_s , övriga parametrar är konstanta**



**Bilaga 6. Residualytör och approximerade derivator m.a.p
olika värden på L , övriga parametrar är konstanta**



Bilaga 7. Beräkningar i Mathematica för beroende mellan α_1 och α_2

Bilaga 8. Beräkningar i Mathematica för beroende mellan d_1 och d_2

Bilaga 9. Funktionen lsqnonlin i MATLAB och dess avbrottsskriterier

LSQNONLIN Solves non-linear least squares problems.

LSQNONLIN solves problems of the form:

$\min_x \sum \{FUN(X)^2\}$ where X and the values returned by FUN can be vectors or matrices.

X=LSQNONLIN(FUN,X0) starts at the matrix X0 and finds a minimum X to the sum of squares of the functions in FUN. FUN accepts input X and returns a vector (or matrix) of function values F evaluated at X. NOTE: FUN should return FUN(X) and not the sum-of-squares sum(FUN(X).^2)). (FUN(X) is summed and squared implicitly in the algorithm.)

X=LSQNONLIN(FUN,X0,LB,UB) defines a set of lower and upper bounds on the design variables, X, so that the solution is in the range LB <= X <= UB. Use empty matrices for LB and UB if no bounds exist. Set LB(i) = -Inf if X(i) is unbounded below; set UB(i) = Inf if X(i) is unbounded above.

X=LSQNONLIN(FUN,X0,LB,UB,OPTIONS) minimizes with the default optimization parameters replaced by values in the structure OPTIONS, an argument created with the OPTIMSET function. See OPTIMSET for details. Used options are Display, TolX, TolFun, DerivativeCheck, Diagnostics, Jacobian, JacobMult, JacobPattern, LineSearchType, LevenbergMarquardt, MaxFunEvals, MaxIter, DiffMinChange and DiffMaxChange, LargeScale, MaxPCGIter, PrecondBandWidth, TolPCG, TypicalX. Use the Jacobian option to specify that FUN also returns a second output argument J that is the Jacobian matrix at the point X. If FUN returns a vector F of m components when X has length n, then J is an m-by-n matrix where J(i,j) is the partial derivative of F(i) with respect to x(j). (Note that the Jacobian J is the transpose of the gradient of F.)

X=LSQNONLIN(FUN,X0,LB,UB,OPTIONS,P1,P2,...) passes the problem-dependent parameters P1,P2,... directly to the functions FUN: FUN(X,P1,P2,...). Pass an empty matrix for OPTIONS to use the default values.

[X,RESNORM]=LSQNONLIN(FUN,X0,...) returns the value of the squared 2-norm of the residual at X: sum(FUN(X).^2).

[X,RESNORM,RESIDUAL]=LSQNONLIN(FUN,X0,...) returns the value of the

```
residual at the solution X: RESIDUAL = FUN(X).
```

[X,RESNORM,RESIDUAL,EXITFLAG]=LSQNONLIN(FUN,X0,...) returns a string EXITFLAG that describes the exit condition of LSQNONLIN.

If EXITFLAG is:

- > 0 then LSQNONLIN converged to a solution X.
- 0 then the maximum number of function evaluations was reached.
- < 0 then LSQNONLIN did not converge to a solution.

[X,RESNORM,RESIDUAL,EXITFLAG,OUTPUT]=LSQNONLIN(FUN,X0,...) returns a structure OUTPUT with the number of iterations taken in OUTPUT.iterations, the number of function evaluations in OUTPUT.funcCount, the algorithm used in OUTPUT.algorithm, the number of CG iterations (if used) in OUTPUT.cgiterations, and the first-order optimality (if used) in OUTPUT.firstorderopt.

[X,RESNORM,RESIDUAL,EXITFLAG,OUTPUT,LAMBDA]=LSQNONLIN(FUN,X0,...) returns the set of Lagrangian multipliers, LAMBDA, at the solution: LAMBDA.lower for LB and LAMBDA.upper for UB.

[X,RESNORM,RESIDUAL,EXITFLAG,OUTPUT,LAMBDA,JACOBIAN]=LSQNONLIN(FUN,X0,...) returns the Jacobian of FUN at X.

Examples

FUN can be specified using @:

```
x = lsqnonlin(@myfun,[2 3 4])
```

where MYFUN is a MATLAB function such as:

```
function F = myfun(x)
F = sin(x);
```

FUN can also be an inline object:

```
fun = inline('sin(3*x)')
x = lsqnonlin(fun,[1 4]);
```

Avbrottsskriterier m.m:

```
ActiveConstrTol: []
DerivativeCheck: 'off'
    Diagnostics: 'off'
DiffMaxChange: 0.1000
DiffMinChange: 1.0000e-008
```

```
    Display: 'final'
GoalsExactAchieve: []
GradConstr: []
    GradObj: []
    Hessian: []
    HessMult: []
HessPattern: []
HessUpdate: []
    Jacobian: 'off'
    JacobMult: []
JacobPattern: 'sparse(ones(jrows,jcols))'
LargeScale: 'on'
LevenbergMarquardt: 'on'
LineSearchType: 'quadcubic'
MaxFunEvals: '100*numberofvariables'
    MaxIter: 400
MaxPCGIter: 'max(1,floor(numberofvariables/2))'
MaxSQPIter: []
MeritFunction: []
    MinAbsMax: []
Preconditioner: []
PrecondBandWidth: 0
ShowStatusWindow: []
    TolCon: []
    TolFun: 1.0000e-006
    TolPCG: 0.1000
    TolX: 1.0000e-006
TypicalX: 'ones(numberofvariables,1)'
```

Bilaga 10. MATLAB-kod för residualfunktionen med Kalmanfilter

```
function res = vinkelresidual_filter(v)
%funktion för att beräkna skillnad mellan uppmätt och beräknad
% vinkel till matchad reflektor för de olika tidpunkterna.

global laser      % indata från laserscanner
global map        % positioner för reflektorerna
global enc         % indata från encodrarna
global state       % indata från referenspunkt
global reflektor  % vilken reflektor mätningen är
                   % associerade till för varje tidpunkt

%parametrar
alfa_1 = v(1);    % rad
alfa_2 = v(2);    % rad
d1 = v(3);
d2 = v(4);
L = v(5);         % m
xs = v(6);        % m
ys = v(7);        % m
teta_s = v(8);   % rad

res = zeros(size(enc,1),1);
gamma_diff = zeros(size(enc,1),1);

%lokala koordinater state-punkt
x = 345/1000;    % m
y = 0/1000;       % m

teta_ = zeros(size(enc,1),1);
X = zeros(size(enc,1),1);
Y = zeros(size(enc,1),1);

%startpositionen, state globalt
teta_0 = state(1,4)/1000;    % rad
X_0 = state(1,2)/1000;       % m
Y_0 = state(1,3)/1000;       % m

%startposition, lasern globalt
teta = teta_0;                % rad
alfa = atan((ys-y)/(xs-x));   % vinkel från lokal x-axel till r, enhet rad
```

```

r = sqrt((xs-x)^2 + (ys-y)^2); % avstand laser-state, enhet m
X = X_0 + r*cos(alfa + teta); % m
Y = Y_0 + r*sin(alfa + teta); % m

%Initiering av fel-kovariansmatris
P = eye(3,3)*0.01;

%integrering med Eulers metod, ger laserscannerns position globalt map
%hjulens hastighet och styrvinkel
for k = 1:(size(enc,1)-1)
    %värden från encodrar
    v1 = enc(k,4)/1000; % m/s
    v2 = enc(k,2)/1000; % m/s
    u1 = enc(k,5)/1000; % rad/s
    u2 = enc(k,3)/1000; % rad/s

    t1 = enc(k,1)/1000;
    t2 = enc(k+1,1)/1000;
    delta_t = t2 - t1;

    %rotationshastighet
    w = (d2*v2)/L * sin(u2 + alfa_2)-(d1*v1)/L * sin(u1 + alfa_1); % rad/s

    %hastigheter m.a.p encodervärden
    vx1 = d1*v1*cos(u1+alfa_1) - ys*w; % m/s
    vy1 = d1*v1*sin(u1+alfa_1) + xs*w; % m/s
    vx2 = d2*v2*cos(u2+alfa_2) - ys*w;
    vy2 = d2*v2*sin(u2+alfa_2) + (xs-L)*w;

    vx = (vx1+vx2)/2;
    vy = (vy1+vy2)/2;

    %globala hastigheter
    vX = vx*cos(teta) - vy*sin(teta);
    vY = vx*sin(teta) + vy*cos(teta);

    %den reflektor vi ska matcha med i nuvarande punkt
    i_1 = reflektor_a(k,1);
    if i_1 >0
        beta_i = reflektor(k,3); %avläst vinkel av lasern
        X_i_1 = map(i_1,1);
        Y_i_1 = map(i_1,2);
        g = gama(X,Y,X_i_1,Y_i_1);
    end
end

```

```

if g < (teta + teta_s)
    fi_i = 2*pi - (teta + teta_s - g);
else
    fi_i = g -(teta_(k) + teta_s);
end
if (beta_i-fi_i)<= pi & (fi_i-beta_i)<=pi
    res(k) = beta_i-fi_i;
elseif (beta_i-fi_i)> pi
    res(k) = fi_i + 2*pi-beta_i;
else
    res(k) = beta_i + 2*pi-fi_i;
end
else
    res(k) = 0;
end

%Jacobianen till rörelsemodellen
f_13 = -1*(vx*sin(teta)+vy*cos(teta)*delta_t;
f_23 = 1*(vx*cos(teta)-vy*sin(teta)*delta_t;
F = [1 0 f_13; 0 1 f_23; 0 0 1];

%(brus)
G = [cos(teta) -sin(teta) 0; sin(teta) cos(teta) 0; 0 0 1];

%Variansmatris (brus)
qx = (abs(vx)*10^(-4)+(abs(vy)+0.1)*(abs(w)+0.01)*10^(-3))*delta_t;
qy = (abs(vy)*10^(-4)+(abs(vx)+0.1)*(abs(w)+0.01)*10^(-3))*delta_t;
qt = (abs(w)*2*10^(-4)+(abs(vx)+abs(vy))*10^(-6))*delta_t;

Q = [qx 0 0; 0 qy 0; 0 0 qt];

%Uppdatering av kovariansmatrisen
P = F*P*F' + G*Q*G';

%Positionsbestämning enligt rörelsemodell
X = X + delta_t*vX;
Y = Y + delta_t*vY;
teta = teta + delta_t*w;

%den reflektor vi ska matcha med i nästa punkt, hjälpa till vid uppdat.
%av position
i_2 = reflektor_a(k+1,1);

```

```

if i_2 > 0
    X_i_2 = map(i_2,1);
    Y_i_2 = map(i_2,2);

    rx = X_i_2 - X;
    ry = Y_i_2 - Y;

    %Gradienten till mätekvationen
    dh_dx = - ry*(-1.0)/(rx^2+ry^2);
    dh_dy = rx*(-1.0)/(rx^2+ry^2);
    dh_dt = -1.0;

    H = [dh_dx dh_dy dh_dt];

    %Kalman återkopplingsfaktor
    R = 10^(-6);
    K = (P*H')* inv(H*P*H' + R);

    I = eye(3,3);
    %Uppdatering av kovariansmatrisen då vi har associerat en
    %reflektor
    P = (I-K*H) * P * (I-K*H)' + K*R*K';

    beta_i = reflektor_a(k+1,3);           %avläst vinkel av lasern
    g = gama(X,Y,X_i_2,Y_i_2);
    if g < (teta + teta_s)
        fi_i = 2*pi - (teta_ + teta_s - g);
    else
        fi_i = g -(teta + teta_s);
    end
    if (beta_i-fi_i)<= pi & (fi_i-beta_i)<=pi
        gamma_diff(k+1) = beta_i-fi_i;
    elseif (beta_i-fi_i)> pi
        gamma_diff(k+1) = fi_i + 2*pi-beta_i;
    else
        gamma_diff(k+1) = beta_i + 2*pi-fi_i;
    end

    %positions skattning med hänsyn till vinkelmätning
    K_g = K*gamma_diff(k+1);

    X = X + K_g(1);
    Y = Y + K_g(2);

```

```

    teta = teta + K_g(3);
end
end

%för att få med vinkel-skillnaden vid den sista tidpunkten
i = reflektor(k+1,1); %den reflektor vi ska matcha med
if i > 0
    beta_i = reflektor(k+1,3); %avläst vinkel av lasern
    X_i = map(i,1);
    Y_i = map(i,2);
    g = gama(X,Y,X_i,Y_i);
    if g < (teta + teta_s)
        fi_i = 2*pi - (teta + teta_s - g);
    else
        fi_i = g -(teta + teta_s);
    end
    if (beta_i-fi_i)<= pi & (fi_i-beta_i)<=pi
        res(k+1) = beta_i-fi_i;
    elseif (beta_i-fi_i)> pi
        res(k+1) = fi_i + 2*pi-beta_i;
    else
        res(k+1) = beta_i + 2*pi-fi_i;
    end
else
    res(k+1) = 0;
end

```