

Optimalisatie gordeltrekbank B-style automotive

Citation for published version (APA):

Pesgens, M. F. M. (2000). *Optimalisatie gordeltrekbank B-style automotive*. (DCT rapporten; Vol. 2000.039). Technische Universiteit Eindhoven.

Document status and date:

Gepubliceerd: 01/01/2000

Document Version:

Uitgevers PDF, ook bekend als Version of Record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

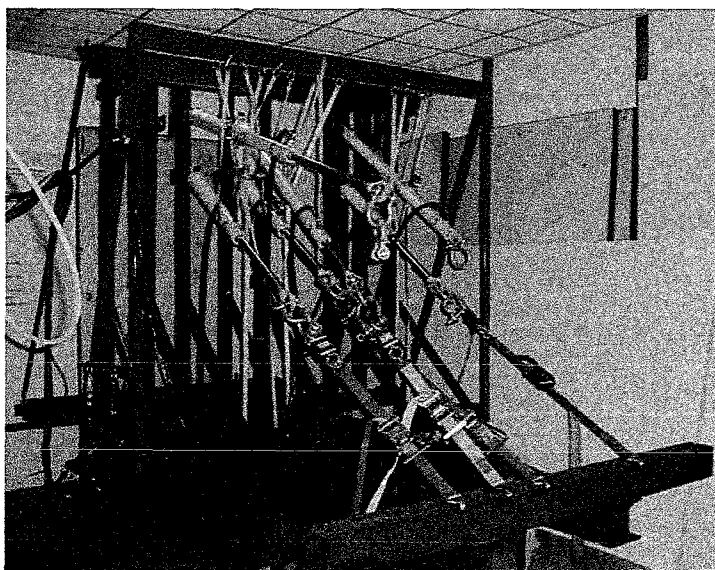
providing details and we will investigate your claim.

Optimalisatie gordeltrekbank B-style Automotive

M.F.M. Pesgens

4 December 2000
Stageverslag 1

Sectie Systems & Control
Rapportnummer 2000-39



Michiel Pesgens
Studentnummer 436027

Stagebegeleider B-style Automotive:
Jeroen Frankenmolen

Stagebegeleider TUE:
Jos Banens

Rapportnummer B-style: R.20.00.00.24/MP
Rapportnummer TUE: 2000-39

Samenvatting

B-style Automotive te Eindhoven houdt zich bezig met conversies van bestaande bedrijfsauto's voor speciale toepassingen, onder rolstoelvervoer en lage vloer bussen. Na deze ombouw moeten volgens de Rijksdienst voor het Wegverkeer (RDW) een aantal proeven worden doorstaan. Een onderdeel hiervan is een test voor de sterkte van de gordelophangpunten. Hierbij moet zo snel mogelijk een gestandaardiseerde kracht op heup- en schoudergordel aangebracht worden, en deze kracht moet 0,2 seconde vastgehouden worden. De constructie mag hierbij een beperkte vervorming ondergaan, om een goedkeuringscertificaat te verkrijgen.

B-style heeft voor deze tests een eigen opstelling. Hierop is een krachtregeling geïmplementeerd. De performance van dit geregelde systeem is echter niet naar wens. Er treedt teveel overshoot (zo'n 1700 N) op in de kracht (maximaal 50000 N) die geleverd wordt, en de krachtopbouw zou liefst sneller moeten gebeuren.

De performance-eisen die gesteld worden zijn:

- Maximaal 5% overshoot.
- Maximaal 200 N statische afwijking.
- Binnen 3 s alle SISO-systemen op hun setpoint, maar hoe sneller, hoe beter.
- Vloeiend (zonder schokken) aanbrengen van de kracht.
- Alle SISO-systemen tegelijkertijd op hun setpoint.
- Als alle SISO-systemen op hun setpoint zijn deze setpoints 0,2 s vasthouden.

De opstelling is opgebouwd uit 6 SISO-systemen, die onafhankelijk geregeld worden, maar die wel enige mate van interactie hebben, via de actuator en de te belasten constructie.

Het blijkt dat de oorzaak van de overshoot ligt in een te lage samplefrequentie (10 Hz) in combinatie met een niet constante sampletijd. Dit hangt samen met gebruikte besturingssysteem Windows NT.

Dan is een lineair tweede orde model van het systeem opgesteld, dat gebruikt is om een nieuwe regeling te destilleren. Er worden een aantal regelaanpakken belicht en indien nodig door middel van een experiment getest. Er wordt voornamelijk gekeken naar stabiliteit door middel van een worst-case aanpak en naar het volgedrag.

Er wordt gekeken naar:

- P-actie
- Feedforward
- PI-actie
- Initialisatie-routine

Uiteindelijk wordt een PI-regelaar gekozen in combinatie met een initialisatieroutine. Het blijkt verder dat trajectorieplanning nodig is om aan de performance-eisen te kunnen voldoen. Na een aantal alternatieven af te wegen wordt een implementatie gekozen onder MS-DOS. De samplefrequentie wordt 200 Hz gekozen.

Uit de eerste tests blijkt dat de ontworpen regeling een zodanig krachtverloop kan realiseren, dat de performance-eisen gehaald worden.

Inhoud

Symbolenlijst		4
Inleiding		5
1. Beschrijving van de opstelling en de huidige regeling		6
2. Analyse		7
2.1	<i>Performance eisen</i>	7
2.2	<i>Analyse van bestaande data</i>	7
2.3	<i>In- en uitgangs metingen met de eigen PC</i>	7
2.3.1	<i>Stapresponsie</i>	8
2.3.2	<i>Integrerende regelaar</i>	8
2.4	<i>In- en uitgangs metingen met externe meetapparatuur</i>	9
2.5	<i>Conclusies en verdere doelstellingen</i>	10
3. Modelvorming		11
3.1	<i>Statisch model</i>	11
3.2	<i>Dynamisch model</i>	11
3.3	<i>Tuning SISO-systemen</i>	12
4. Regelaarontwerp		13
4.1	<i>Proportionele regelaar</i>	13
4.1.1	<i>Stabiliteitsanalyse</i>	13
4.1.2	<i>Volgedrag</i>	14
4.2	<i>Feedforward</i>	15
4.2.1	<i>Volgedrag</i>	15
4.2.2	<i>Resultaten</i>	16
4.3	<i>Integrerende actie</i>	16
4.3.1	<i>Stabiliteit</i>	16
4.3.2	<i>Volgedrag</i>	16
4.3.3	<i>Resultaten</i>	17
4.4	<i>Initialisatieroutine</i>	18
4.5	<i>Schatter</i>	20
4.6	<i>Regelaarkeuze</i>	20
5. Krachtverloop		21
5.1	<i>Referentiekraft</i>	21
5.2	<i>Tolerantie</i>	21
5.3	<i>Totale krachtverloop</i>	22
5.4	<i>Resultaten</i>	22
6. Regelaarimplementatie		24
6.1	<i>Besturingssysteem</i>	24
6.2	<i>Programmatuur</i>	24
6.3	<i>Gebruikers-interface</i>	24
7. Conclusies en aanbevelingen		25
Bijlage 1	Schema opstelling trekbank B-Style Automotive	26
Bijlage 2	Meetplan Trekbank B-Style Automotive m.b.v. SigLab 28/09/2000	27
Bijlage 3	Resultaten metingen 28/9/2000	28
Bijlage 4	Data voor fitten eerste-orde model	32
Bijlage 5	Fit van 2^e orde-model	33
Bijlage 6	Meetplan B-style Automotive 19/10/2000	34
Bijlage 7	Worst-case systeem-identificatie experimenten	35
Bijlage 8	Worst-case systeem- en regelaar-parameters	37
Bijlage 9	C- regelaarimplementatie	38
Bijlage 10	Gebuurde programma's en bestanden	43
Bijlage 11	Procedures	45

Symbolenlijst

a_{\max}	=	Maximale dubbele afgeleide van de kracht [F/s^2]
f_{nyq}	=	Nyquist frequentie [$1/s$]
k	=	Versterkingsfactor [-]
p	=	Druk [$\text{Bar} = 10^5 \text{ Pa}$]
u	=	Ingangsspanning [V]
u_{ini}	=	Initialisatie-ingang [V]
u_v	=	du/dt [V/s]
v_{\max}	=	Maximale afgeleide van de kracht [F/s]
t	=	Tijd [s]
A	=	Systeemparameter [$1/s$]
B	=	Systeemparameter [$1/s^2$]
C	=	Systeemparameter [$1/Vs$]
D	=	Systeemparameter [$1/Vs^2$]
F	=	Kracht [N]
F_{offset}	=	Systeemparameter [N]
F_{sp}	=	Setpoint [N]
F_r	=	Referentie [N]
F_{tol}	=	Tolerantie referentie [N]
F_v	=	Krachtsignaal, uitgedrukt als spanning [V]
F_{vs}	=	Voorspankracht [N]
I	=	Integrerende versterkingsfactor [V/Ns]
P	=	Proportionele versterkingsfactor [V/N]
T	=	Sampletijd [s]
T_{eff}	=	Effectieve sampletijd [s]
T_{wait}	=	Wachttijd [s]
γ	=	Modelfouten en ruis [N]
φ	=	Systeemparameter [$1/s$]
ξ	=	(Dimensieloze) dempingsfactor [-]
ω_0	=	(Ongedempte) eigenfrequentie [$1/s$]
τ	=	Dode tijd (systeemparameter) [s]
τ_i	=	Tijdsconstante integrator [$1/s$]

Inleiding

B-style Automotive te Eindhoven houdt zich bezig met conversies van bestaande bedrijfsauto's voor speciale toepassingen, onder andere rolstoelvervoer en lage vloer bussen. Na deze ombouw moeten volgens de Rijksdienst voor het Wegverkeer (RDW) een aantal proeven worden doorstaan. Een onderdeel hiervan is een test voor de sterkte van de gordelbevestigingspunten.

Hierbij moet zo snel mogelijk een gestandaardiseerde kracht op heup- en schoudergordel aangebracht worden, en deze kracht moet 0,2 seconde vastgehouden worden. De constructie mag hierbij een beperkte vervorming ondergaan, om een goedkeuringscertificaat te verkrijgen.

B-style heeft voor deze tests een eigen opstelling. Hierop is een krachtregeling geïmplementeerd.

De performance van dit geregelde systeem is echter niet geheel naar wens. Er treedt teveel overshoot op in de kracht die geleverd wordt, en de krachtopbouw zou liefst sneller moeten gebeuren.

In hoofdstuk 1 zal geanalyseerd worden hoe de huidige opstelling is opgebouwd.

In hoofdstuk 2 zal de reeds bestaande data worden bekeken en onderzocht worden wat de oorzaken zijn van de overshoot.

Dan zal in hoofdstuk 3 een model van het systeem worden opgesteld, dat in hoofdstuk 4 gebruikt zal worden om een nieuwe regeling te destilleren.

In hoofdstuk 5 zal de trajectorieplanning beschreven worden en in hoofdstuk 6 zal de uiteindelijke (software-) implementatie volgen.

In hoofdstuk 7 volgen conclusies en aanbevelingen.

1. Beschrijving van de opstelling en de huidige regeling

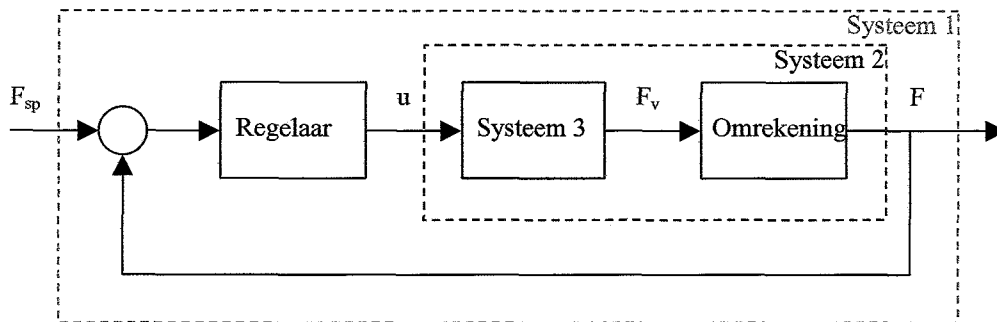
De gordeltrekbank bestaat uit 6 hydraulische cylinders, die door een oliepomp gevoed worden. Tussen de oliepomp en de cylinders zit per cylinder 1 ventiel, die door proportionele drukregelaars op druk geregeld worden. Deingangsspanning u [V] van deze drukregelaars wordt door een D/A kaart van de PC gerealiseerd. De gerealiseerde kracht wordt door 6 krachtdozen gemeten (uitgevoerd met rekstrookjes) die na 6 signaalversterkers een krachtsignaal F_v aan de A/D-kaart van de PC aanbieden. Op de PC is een regeling geïmplementeerd die u uit F_v berekent.

In feite kan de zojuist beschreven opstelling gezien worden als 6 SISO-systemen, die alleen een afhankelijkheid van elkaar hebben doordat ze door dezelfde oliepomp gevoed worden en via de te belasten constructie. Vanaf nu wordt steeds slechts 1 SISO-systeem meegenomen in de analyse.

De SISO-systemen zullen worden aangeduid als SISO-1 t/m SISO-6.

De opbouw van de gordeltrekbank in componenten staat weergegeven in Bijlage 1.

Dit wordt hieronder vereenvoudigd in een blokschema weergegeven (figuur 1.1).



Figuur 1.1: Blokschema

Hierin is:

F	:	Kracht [N]
F_{sp}	:	Te bereiken kracht (setpoint) [N]
F_v	:	Krachtsignaal als spanning [V]
u	:	Ingang [V]

Systeem 1 is het totale geregelde systeem met ingang F_{sp} [N] (het te bereiken setpoint) en uitgang F.

Systeem 2 in figuur 1.1 heeft de spanning u [V] als ingang en kracht F [N] als uitgang. Dit is in feite het ongeregelde systeem.

Systeem 3 heeft u als ingang en F_v als uitgang. Systeem 3 is Systeem 2 zonder een omrekening per SISO-systeem. Dit is een lineaire sensorschaling.

2. Probleemidentificatie

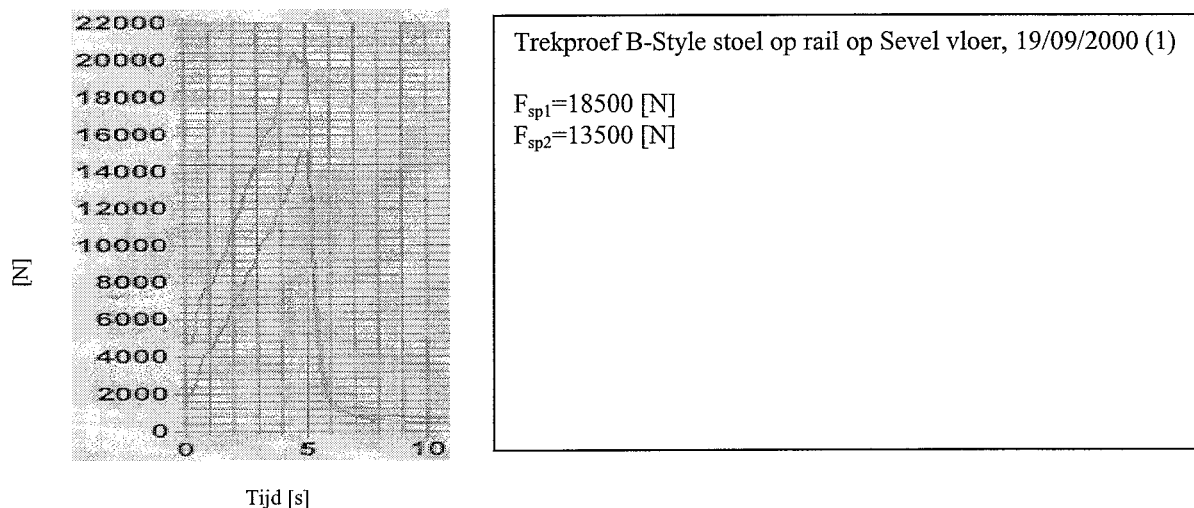
2.1 Performance eisen

De eisen die aan het systeem gesteld worden zijn:

- Maximaal 5% overshoot.
- Maximaal 200 [N] statische afwijking.
- Binnen 3 [s] alle SISO-systemen op hun setpoint, maar hoe sneller, hoe beter.
- Vloeiend (zonder schokken) aanbrengen van de kracht.
- Alle SISO-systemen tegelijkertijd op hun setpoint.
- Als alle SISO-systemen op hun setpoint zijn, deze setpoints 0,2 [s] vasthouden.

2.1 Analyse van bestaande data

Eerst zal er gekeken worden wat er aan informatie over meetresultaten reeds voorhanden is. In figuur 2.1 staat een karakteristieke meting (aan Systeem 2). In dit experiment worden twee SISO systemen gebruikt, die we dus onafhankelijk van elkaar kunnen beschouwen. In figuur 6 worden twee krachten tegelijkertijd opgevoerd tot een gewenste kracht F_{sp1} en F_{sp2} .



Figuur 2.1: Meetdata B-style

De regelwet die gebruikt wordt is:

$$u = I \cdot \int_0^t \text{sign}(F_{sp} - F) d\tau$$

Dit is dus een integrerende regelaar. Aangenomen dat het systeem stabiel, lineair en (na uitdemping van inschakelverschijnselen) quasi-statisch is, zou dit een rechte lijn als grafiek moeten opleveren. Er zijn echter golf-verschijnselen zichtbaar in F , en niet alleen als (uitdempend) inschakelverschijnsel.

De waarde voor I is experimenteel bepaald, er is niet theoretisch geverifieerd of de stabiliteit gegarandeerd is. De performance van de huidige opstelling (Systeem 1) voldoet niet qua snelheid en overshoot (zie figuur 3). Er is een overshoot van 1900 [N] respectievelijk 1700 [N], wat zo'n 10 % respectievelijk 12,5 % van het setpoint is. Verder blijkt dat de overshoot toeneemt als de kracht sneller wordt opgebouwd. Kijkende naar deze en andere grafieken valt op dat de maximale waarde van de overshoot 0,6 [s] na het passeren van het setpoint wordt bereikt.

2.3 In- en uitgang metingen met de eigen PC

Om Systeem 2 nader te kunnen analyseren is het nodig zowel de ingang u als de uitgang F te kunnen meten en u te kunnen controleren. Daarom is het huidige regelprogramma (behorende bij Systeem 1) hergeprogrammeerd. Nu is een willekeurige ingang u te realiseren (al dan niet met terugkoppeling meting F). De data van zowel u als

F wordt opgeslagen. Tevens wordt het aantal software-bewerkingen in de loop (bij terugkoppeling als systeem 1 in figuur 2) verminderd om de doorlooptijd van het systeem te minimaliseren.

Nu worden een aantal experimenten uitgevoerd.

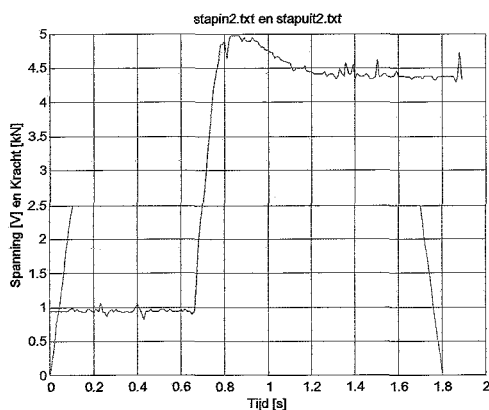
2.3.1 Stapresponsie

Er is een stapresponsie gemeten door een stap in de ingangsspanning op Systeem 2 te zetten (zie figuur 2.2). De regelkring wordt nu dus niet gesloten. Er vindt sturing plaats.

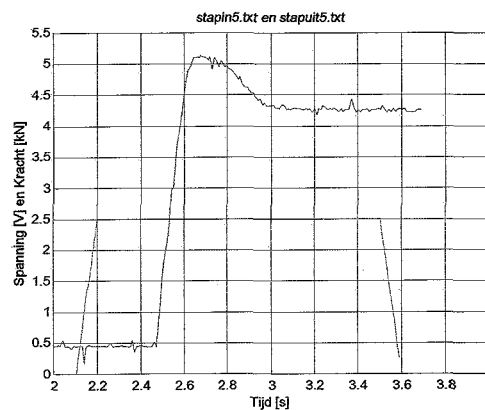
De volgende instellingen zijn gebruikt:

$$u=2,5 \text{ [V]} \quad (\text{SISO-2})$$

Wat direct opvalt is de dode tijd τ van 0,6 [s]. Het is niet duidelijk of de stap op $t=0$ [s] aangegrepen heeft of op $t=0.1$ [s]. Verder is er een overshoot waar te nemen. De constante kracht alvorens de stap aangrijpt wordt veroorzaakt door de voorspanningskracht (F_{vs}) die handmatig met spanbanden op het systeem wordt gezet. Om nu te onderzoeken of de dode tijd een inschakelverschijnsel van de software is, omdat het programma bij dit experiment tegelijk gestart is met de stap in u , wordt er nu een volgend experiment uitgevoerd. In dit experiment worden de berekeningen door de software gestart, en de stap wordt later aangeboden (zie figuur 2.3).



Figuur 2.2: Stapresponsie



Figuur 2.3: Stapresponsie met vertraging

De dode tijd τ is nu 0,3 [s] geworden. De eindwaarde van de responsie blijft toch steeds rond de 4,3 [kN], de piekwaarde rond de 5 [kN]. Bij variëren van het tijdstip waarop de stap aangrijpt, komen steeds andere waarden voor de dode tijd (τ) naar voren, met 0,6 [s] als maximale waarde. De overshoot in figuur 2.1 zou door dezelfde dode tijd veroorzaakt kunnen zijn.

2.3.2. Integreerend regelaar

Nu wordt er een integreerend regelaar op Systeem 2 gezet. Dit is dus weer Systeem 1.

De regelwet wordt dus:

$$u = I \cdot \int_0^t (F_{sp} - F) dt \quad (2.1)$$

met

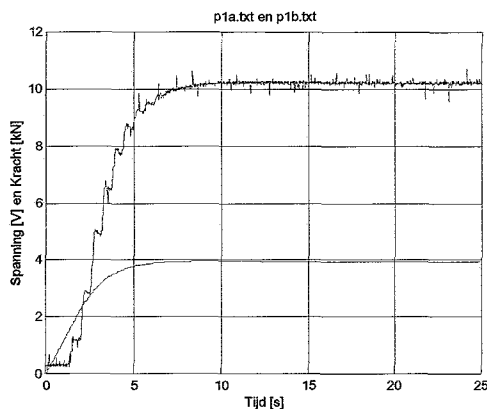
F	:	Kracht [N]
F_{sp}	:	Setpoint [N]
u	:	Spanning [V]
I	:	Regelversterking [Vs/N]

De volgende instellingen zijn gebruikt (SISO-2):

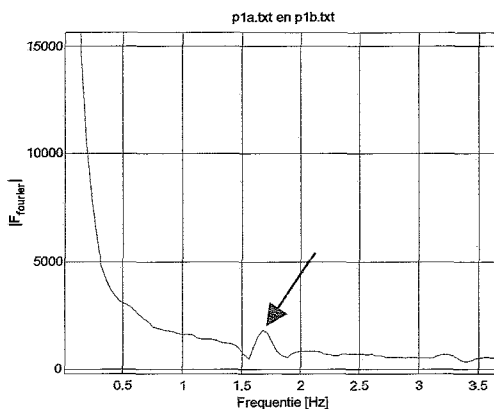
$$F_{sp} = 10200 \text{ [N]}$$

$$I = 1 \cdot 10^{-5} \text{ [Vs/N]}$$

De resultaten van dit experiment staan in figuur 2.4 weergegeven.



Figuur 2.4: Responsie proportionele regelaar



Figuur 2.5: Fourier-getransformeerde krachtsignaal

De dode tijd is nog steeds aanwezig. Hetingangssignaal u (rood) verloopt exponentieel, wat te verwachten valt met de gekozen regeling. Er treden echter weer schokachtige verschijnselen in F (blauw) op die niet in u terug te vinden zijn. Dit is vreemd, want dit ingangssignaal wordt weer uit de krachtmeting berekend (met formule 2.1). De schokverschijnselen waren reeds aanwezig als golfjes in de resultaten van B-style (zie figuur 2.1). Dit duidt erop dat er bemonsterd wordt op de volgende tijdstippen:

$$t = t_0 + kT_{eff} \quad (2.2)$$

met:

$$k = 1, 2, 3, \dots$$

$$t_0 \quad : \quad \text{Begintijdstip [s]}$$

$$T_{eff} \quad : \quad \text{Effectieve sampletijd [s] > T}$$

Er worden echter wel punten getekend voor tussenliggende tijdstippen. De schokverschijnselen zijn visueel duidelijk waar te nemen op de opstelling zelf.

Wat verder opvalt is dat de vorm van de schokverschijnselen erg lijken op de stapresponsies die reeds eerder gemeten zijn (figuur 2.2 en 2.3). Dit versterkt het vermoeden dat we te maken hebben met een effectieve sampling als in formule 2.2. Formule 2.2 heeft als gevolg dat u verandert met frequentie $1/T_{eff}$. Deze frequentie is door Fourier-analyse van het krachtsignaal te nemen, af te lezen (figuur 2.5). Deze frequentie komt overeen met een T_{eff} van 0,6 [s] (bij de pijl in figuur 2.5), wat tevens de waarde van de maximale dode tijd bij de stapresponsies was (figuur 2.2).

2.4 In- en uitgangsmetingen met externe meetapparatuur

De herkomst van deze dode tijd en de schokverschijnselen zullen nu verklaard moeten worden en de oorzaak van deze problemen gelokaliseerd. Daarom zullen er nu een aantal metingen worden gedaan met een onafhankelijk meetapparaat.

Hierbij zullen u en F_v van systeem 3 worden gemeten en de waarden voor u zullen achtereenvolgens door de pc en het meetapparaat worden gegenereerd.

Het meetplan staat in bijlage 2, de grafieken van alle metingen staan in bijlage 3.

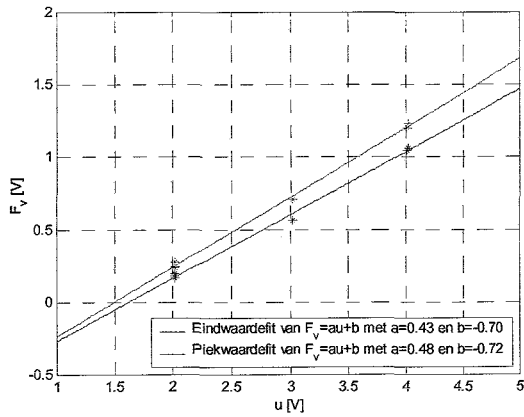
Er is bij deze metingen alleen gekeken naar SISO-2 van de opstelling.

Bij de stapresponsie-metingen van Systeem 3 (figuren B3.1 t/m B3.12) valt direct op dat de dode tijd zeer sterk gereduceerd is ($\tau=0.05$ [s]) ten opzichte van de waarden in paragraaf 2.3 (0.6 [s]), zie figuur 7 en 8. Dus het grootste deel van de dode tijd wordt niet in Systeem 3 veroorzaakt, en dus ergens anders in Systeem 1 (in de PC, zie bijlage 1).

Er is duidelijk een offset in u nodig om tot een verandering in F te komen.

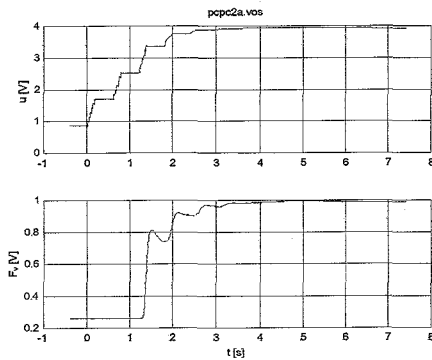
Ook nu blijft de overshoot bij een stapresponsie aanwezig.

Om de lineariteit van het systeem te analyseren nemen we de eindwaarden en de piekwaarden van de stapresponsies (figuren B3.1 t/m B3.12), en fitten hier met de kleinste kwadraten methode een rechte lijn door. De data staat in bijlage 3, figuur 2.6 geeft de fit door de meetpunten weer. De maximale fout in F_v is 0,07[V].



Figuur 2.6: Fit op stapresponsie-data

Verder blijkt formule 2.2 niet helemaal te gelden. er wordt klaarblijkelijk niet bemonsterd met een sampletijd T van $0,6$ [s], maar om de $0,6$ [s] komt een serie van nieuwe waarden voor u (zie figuur 2.7). Dit betekent dat de sampletijden niet constant zijn. Dit is zeer vervelend in een closed-loop situatie aangezien de maximale waarde voor de sampletijd T een extra faseachterstand veroorzaakt. Dit is nadelig voor de stabiliteit. Verder zijn de schokken niet wenselijk, zie de performance-eisen in paragraaf 2.1.



Figuur 2.7: Responsie op I-regelaar

2.5 Conclusies en verdere doelstellingen

De huidige regelaar is ingesteld zonder te kijken naar robuustheid qua stabiliteit. Stabiliteit is de belangrijkste voorwaarde waaraan voldaan moet worden want zonder stabiliteit wordt geen enkele performance-eis gehaald. Daarom zal eerst een model opgesteld worden, waarop een regelaar gebaseerd zal worden. Deze zal ten eerste qua stabiliteit robuustheid ingebouwd moeten krijgen, en tevens een goed volgedrag hebben (met paragraaf 2.1 in het achterhoofd).

De eis dat alle SISO-systemen gelijk op hun setpoint moeten komen, is te realiseren met een trajectorie in plaats van een vast setpoint.

Het bemonsterings-gedrag is niet naar wens (niet-constante, en te grote T). Daarom zal er een andere regelaarimplementatie gerealiseerd moeten worden.

3. Modelvorming

Er zal nu een model afgeleid worden dat gebruikt kan worden bij het ontwerp van een regelaar.

3.1 Statisch model

De eindwaarde van figuur 2.6 is een statisch model van Systeem 3 (SISO-2):

$$F_v = 0.43u - 0.70 \quad (3.1)$$

met :

u : Setpoint [V]
 F_v : Krachtsignaal als spanning [V]

De eindwaarde van het uitgangssignaal is proportioneel met het ingangssignaal plus een offset.

Dit model gecombineerd met formule 2.2 geeft de volgende formule voor de statische relatie tussen u en F (Systeem 2, SISO-2):

$$F = 4.1 \cdot 10^3 u - 7.0 \cdot 10^3 \quad (3.2)$$

De maximale fout tussen model en meting is 290 [N] in statische situatie. Om het dynamische gedrag van Systeem 2 te beschrijven is dit model niet geschikt. De fout kan dan oplopen tot 100% van full-scale (bij een stapresponsie).

3.2 Dynamisch model

Er zal nu een dynamisch model van Systeem 2 gemaakt worden om een regeling op te kunnen baseren.

Uit de resultaten in bijlage 3 (traagheidsverschijnselen, overshoot, damping) blijkt dat er een dynamisch model nodig is om het gedrag van Systeem 2 (de relatie tussen F en u) goed te kunnen beschrijven.

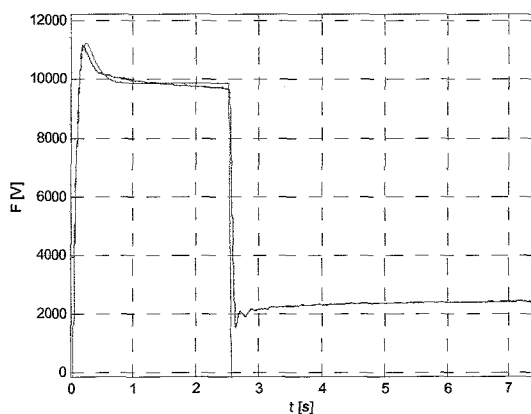
Kijkende naar de stapresponsies in bijlage 3 (figuren B3.1 t/m B3.12) is een model met een vergelijkbare responsie:

$$Y(s) = \frac{(Cs + D)e^{-s\tau}}{s^2 + As + B} U(s) - F_0 + \gamma(s) \quad (3.3)$$

ofwel:

$$\ddot{F}(t) + A\dot{F}(t) + B(F(t) + F_0 - \gamma(t)) = C\dot{u}(t - \tau) + Du(t - \tau) \quad (3.4)$$

De fitresultaten zijn grafisch uitgedrukt in figuur 3.1 en bijlage 5. Hierin staat tevens de gebruikte data voor het fitten van de parameter τ . De bij dit model behorende parameters (voor SISO-2) staan in bijlage 5. Bij het fitten hiervan is τ niet meegenomen. De waarde van τ zal gebruikt worden om de stabiliteitsgrenzen te bepalen bij het ontwerp van een regelaar.



Figuur 3.1: Tweede-orde fit op stapresponsie

3.3 Tuning SISO-systemen

Er zijn 6 SISO-systemen, die elk een ander kwantitatief dynamisch gedrag vertonen, maar dat kwalitatief erg op elkaar lijkt (zie de figuren van bijlage 3, 5 en 7).

De componenten van deze 6 SISO-systemen zijn op de cylinders na gelijk. Er zijn twee soorten cylinders, dus kunnen we de SISO-systemen indelen in twee groepen: 1, 3, 5 en 2, 4, 6. Door nu voor elk van deze twee groepen 1 SISO-systeem uit te kiezen als referentie, kunnen de andere 2 systemen door instellen van de klepversterkers (zie bijlage 1) in principe hetzelfde dynamische gedrag gaan vertonen.

deze aanpak is vooral handig, omdat er geen nieuwe modelvorming en regelaar-instellingen berekend hoeft te worden als er een component in het systeem vervangen moet worden, waardoor het gedrag van dat SISO-systeem verandert. De "service-ability" verbetert hierdoor.

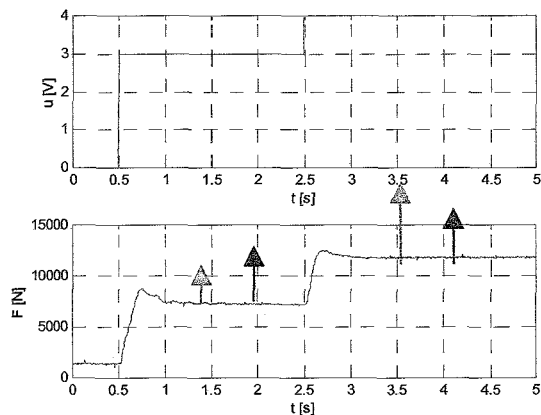
Aan de klepversterkers kunnen twee parameters ingesteld worden, namelijk de gain en de offset.

Het gedrag van het systeem is te vergelijken door een ingangssignaal te kiezen waar de invloed van deze parameters duidelijk naar voren komt.

Er is gekozen voor een signaal, opgebouwd uit 2 achtereenvolgende stapresponsies (zie figuur 3.2).

Als de gain omhooggeschroefd wordt, wordt de grafiek opgeblazen (groen), bij verandering van de offset wordt de grafiek alleen getransleerd (rood) dit alles in verticale richting.

De praktische handelingen bij het tunen van een SISO-systeem staan uitgelegd in bijlage 11.



Figuur 3.2: Ingangssignaal voor tuning SISO-systemen

4. Regelaarontwerp

4.1 Proportionele regelaar

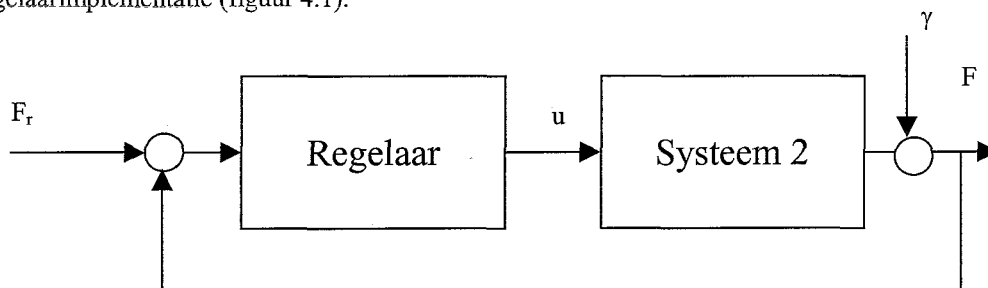
Een eerste voorstel van een regeling is een P-regeling op de fout in de kracht.

$$u = P(F_r - F) \quad (4.1)$$

met:

F_r : De te volgen referentietrajectorie [N]
 P : De proportionele regelversterking P [V/N]

Systeem 4 is het geregelde Systeem 2 wat verschilt met Systeem 1 qua regelaarontwerp en regelaarimplementatie (figuur 4.1).



Figuur 4.1: Systeem 4 in detail

4.1.1 Stabiliteitsanalyse

Stabiel systeem is de eerste vereiste.

Model van paragraaf 3.2:

$$\ddot{F}(t) + A\dot{F}(t) + B(F(t) + F_0 - \gamma(t)) = C\dot{u}(t - \tau) + Du(t - \tau) \quad (3.4)$$

Invullen van formule 4.1 geeft de closed loop differentiaalvergelijking

$$\ddot{F}(t) + A\dot{F}(t) + B(F(t) + F_{offset} - \gamma(t)) = CP(\dot{F}_r(t) - \dot{F}(t - \tau)) + DP(F_r(t) - F(t - \tau)) \quad (4.2)$$

Als overdrachtsfunctie (na Laplace-transformatie) is het voorafgaande als volgt te schrijven:

Model en regelwet Laplace-transformeren:

$$\frac{F(s)}{U(s)} = \frac{(As + B)e^{-s\tau}}{s^2 + As + B}, \quad \frac{U(s)}{F(s)} = P \quad (4.3)$$

dus de open loop overdracht is:

$$H_{ol} = \frac{P(As + B)e^{-s\tau}}{s^2 + As + B} \quad (4.4)$$

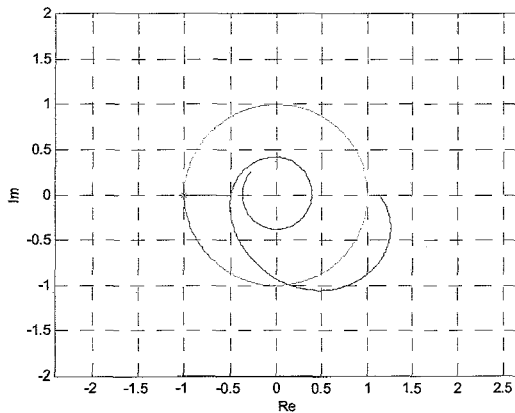
Door nu met de criteria van Nyquist (amplitudemarge 0.5 en fasemarge 45°) te kijken wat qua stabiliteit een geschikte waarde voor P is, kunnen we deze regelaar implementeren. Echter het model van het Systeem 2 is niet constant, dat wil zeggen dat dit model afhangt van een de fysieke opstelling van het systeem, en dan ook nog eens verschillend per SISO-systeem. Verder hebben deze SISO-systemen ook nog eens invloed op elkaar. Er zijn een aantal proeven gedaan om te onderzoeken of er een worst-case systeem is qua stabiliteit.

Er is gekeken naar een aantal stapresponsies met verschillende fysieke opstellingen die in de praktijk voor kunnen komen. Dit is gedaan met SISO-systemen 3 en 4. Het meetplan hiervan staat in bijlage 6. De resultaten staan in bijlage 8. Uit deze resultaten blijkt dat de opstelling met de snelste responsie (die met de kortst mogelijke trekband, 1 SISO-systeem tegelijk) het worst-case model voor Systeem 2 qua stabiliteit is. Dit systeemmodel wordt nu verder voor de stabiliteitsanalyse gebruikt.

De aanname is gedaan dat alle SISO-systemen kwalitatief hetzelfde gedrag vertonen. Dit is te rechtvaardigen omdat de responsies van deze SISO-systemem qua vorm erg op elkaar lijken. Voor de overige SISO-systemen behoeft nu alleen een experiment met deze worst-case systeem-opstelling gedaan worden om een systeemmodel te fitten.

De regelaarinstellingen staan in bijlage 8.

Zie figuur 4.2 voor de bijbehorende Nyquist-figuur.



Figuur 4.2: Nyquist plot voor H_o , Systeem 2 met P-regelaar, Cyl 3

4.1.2 Volgedrag

Een tweede eis is dat zo snel mogelijk een setpoint F_{sp} bereikt moet worden zonder overshoot.

Om te kijken of dit setpoint bereikt zal worden met deze proportionele regelaar is een stationair model gebruikt.

Stationair model (alle afgeleiden nul) voor Systeem 2:

$$B(F + F_{offset}) = Du + \gamma \quad (4.5)$$

met:

γ : constante systeem- en model-fouten

Regelwet (zie 4.1):

$$u = P(F_{sp} - F) \quad (4.6)$$

met:

F_{sp} : het gewenste setpoint (De referentie F_r op $t \rightarrow \infty$)

Ingevuld:

$$F = \frac{DP}{B+DP} F_{sp} - \frac{B}{B+DP} F_{offset} + \frac{B}{B+DP} \gamma \quad (4.7)$$

Voor $P \rightarrow \infty$ wordt F gelijk aan F_{sp} voor $t \rightarrow \infty$ bij een stabiele regelkring van Systeem 4.

De proportionele regelversterking P kan in de praktijk niet te ver opgevoerd worden in verband met de stabiliteits-eis. De dode tijd τ is de belangrijkste beperking in verband met de fasedraaiing die hierdoor veroorzaakt wordt. De vraag is nu of P groot genoeg kan worden gemaakt om met het kwantitatieve volgedrag binnen de toleranties te vallen.

Uit de stabiliteits analyse volgt een maximale waarde van $P=5 \cdot 10^{-4}$ (Cyl 3).

Het worst-case systeemmodel voor Systeem 2 (SISO-3) genomen, geeft dit:

$$F = \frac{1,8 \cdot 10^5 \cdot 5,0 \cdot 10^{-4}}{79,5 + 1,8 \cdot 10^5 \cdot 5,0 \cdot 10^{-4}} F_{sp} - \frac{79,5}{79,5 + 1,8 \cdot 10^5 \cdot 5,0 \cdot 10^{-4}} 2,17 \cdot 10^3 = 0,53 \cdot F_{sp} - 1 \cdot 10^3 \quad (4.7a)$$

P kan niet groot genoeg worden gemaakt om een stationaire versterkingsfactor tussen F_{sp} en F te krijgen van 1 ($\pm 5\%$). Hiervoor kan een computed torque gebruikt worden, oftewel een feedforward regelactie die ervoor zorgt dat de statische versterkingsfactor gelijk aan 1 wordt. Een voorwaarde voor iedere feedforward is een goed (genoeg) systeemmodel zodat γ klein is.

Verder blijkt dat F_{offset} te groot is om een goed volgedrag te krijgen. Deze veroorzaakt dus een statische afwijking. Verder moet γ klein genoeg zijn.

Het is dus nodig dat de constante term F_{offset} gecompenseerd wordt is om een goed volgedrag te krijgen.

Dit kan op de volgende manieren gerealiseerd worden:

1. F_{offset} is als constante te bepalen en deze dan als feedforward gebruiken.
2. Alle constante termen door een I-actie laten integreren.
3. Een initialisatieroutine initieel de constante termen te laten integreren.
4. F_{offset} als online schatter mee laten lopen.

4.2 Feedforward

4.2.1 Volgedrag

Gewenst als stationaire overdracht voor $t \rightarrow \infty$ (stabiel systeem) is:

$$F = 1 \cdot F_{sp} \quad (4.8)$$

De stationaire overdracht van Systeem 2:

$$F = \frac{D}{B} u + F_{offset} + \gamma \quad (4.9)$$

met:

$$u = \frac{B}{D} u_2 \quad (4.10)$$

wordt dit:

$$F = \frac{D}{B} F_{offset} + u_2 + \gamma \quad (4.11)$$

Zoals in de vorige paragraaf reeds afgeleid, voldoet enkel een P-regelaar niet qua setpoint-realiserie.

$$u_2 = P(F_{sp} - F) + u_3 \quad (4.12)$$

ingevuld in formule 4.11:

$$F = \frac{DP}{B + DP} F_{sp} + \frac{D}{B + DP} u_2 + \frac{B}{B + DP} \gamma \quad (4.13)$$

met:

$$u_3 = \frac{B}{D} F_{sp} \quad (4.14)$$

wordt dit:

$$F = \frac{DP}{B + DP} F_{sp} + \frac{B}{B + DP} F_{sp} + \frac{B}{B + DP} \gamma = F_{sp} + \frac{B}{B + DP} \gamma \quad (4.15)$$

Wat gewenst is.

Alleen γ is nog aanwezig als verstoring, maar deze wordt toch gedeeltelijk weggeregeld door de P-actie. Deze fout is verder te verkleinen door een beter model te gebruiken voor de feed-forward actie.

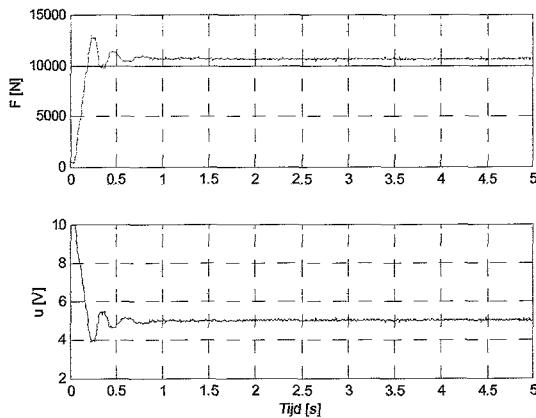
De totale (feedforward en feedback) regelactie wordt nu de combinatie van formules 4.10, 4.12 en 4.14:

$$u = \frac{B}{D} F_{offset} + \frac{B}{D} F_{sp} + P(F_{sp} - F) \quad (4.16)$$

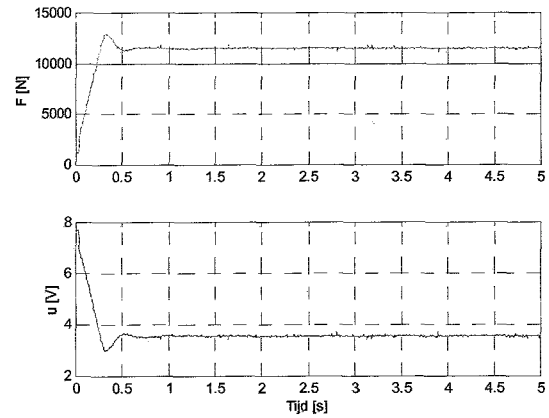
De gefitte worst-case systeemparemeters staan in bijlage 8.

Het is nu de vraag of de systeemparemeters die gebruikt worden om deze feedforward te berekenen nauwkeurig genoeg bekend zijn en over het gehele werkgebied constant genoeg zijn om een nauwkeurig genoeg volgedrag te krijgen.

4.2.2 Resultaten



Figuur 4.3: Responsie P-regelaar met feed-forward, Cyl 3



Figuur 4.4: Responsie P-regelaar met feed-forward, Cyl 4

De regelwet (formule 4.16) is geïmplementeerd op het geregelde Systeem 4 (figuur 4.1). De resultaten voor SISO-3 staan in figuur 4.3 en de resultaten voor SISO-4 staan in figuur 4.4. De eis wat betreft setpoint-realiseratie wordt niet gehaald aangezien er bij beide SISO-systemen een flinke offset in F overblijft. De slingeringen en de overshoot in de responsies zijn wel te vermijden/beperken door een PD-regelaar te ontwerpen of door een (tragere) realiseerbare trajectorie voor F_1 te gebruiken. Dit is wel het snelste haalbare systeemgedrag met de huidige configuratie.

4.3 Integrerende actie

Aan de proportionele regelactie is een integrerende actie toegevoegd:

$$u = P(F_{sp} - F) + \frac{P}{\tau_i} \int_0^t (F_{sp} - F) \tau \quad (4.17)$$

met:

τ_i : De tijdsconstante van de integrerende actie

4.3.1 Stabiliteit

Deze integrerende actie heeft consequenties voor de stabiliteit.

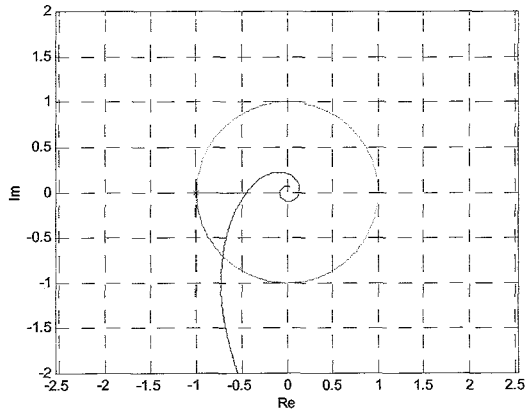
Het blijkt dat de worst-case situatie qua systeemgedrag dezelfde is gebleven als in 4.1.1. De snelste responsie, dus de opstelling met de korst mogelijke trekband is de worst-case situatie.

Deze worst-case situatie is als volgt bepaald: Bij ieder van de responsies in Bijlage 7 is een model gefit, waarbij met de criteria van Nyquist (amplitudemarge 0.5 en fasemarge 45°) maximale waarden voor P en $I (=P/\tau_i)$ bepaald zijn. Het blijkt dat beide versterkingsfactoren P en I het minst ver opgevoerd kunnen worden bij het worst-case model.

De worst-case regelparemeters staan in bijlage 8.

De waarde van P kan nu echter niet zover opgevoerd worden als enkel met een proportionele regelaar, wat een tragere responsie oplevert.

De nyquist-plot wordt nu (figuur 4.5):



Figuur 4.5: Nyquist plot voor H_{ol} , Systeem 2 met PI-regelaar, Cyl 3

4.3.2 Volgedrag

Nu wordt F gelijk aan F_{sp} voor $t \rightarrow \infty$ zonder de eis $P \rightarrow \infty$:

$$B(F + F_{offset}) = D \left(P(F_{sp} - F) + \frac{P}{\tau_i} \int_0^t (F_{sp} - F) dt \right) + \gamma \quad (4.18a)$$

$$F = \frac{DP}{B + DP} F_{sp} - \frac{B}{B + DP} F_{offset} + \frac{P}{\tau_i(B + DP)} \int_0^t (F_{sp} - F) dt + \gamma \quad (4.18b)$$

$$F = F_{sp} \quad (4.18c)$$

De integraal compenseert dus de constante termen (ook van γ) voor $t \rightarrow \infty$ zodanig dat $F = F_{sp}$:

$$\frac{P}{\tau_i(B + DP)} \int_0^t (F_{sp} - F) dt = \frac{B}{B + DP} F_{sp} + \frac{B}{B + DP} F_{offset} - \gamma \quad (4.19)$$

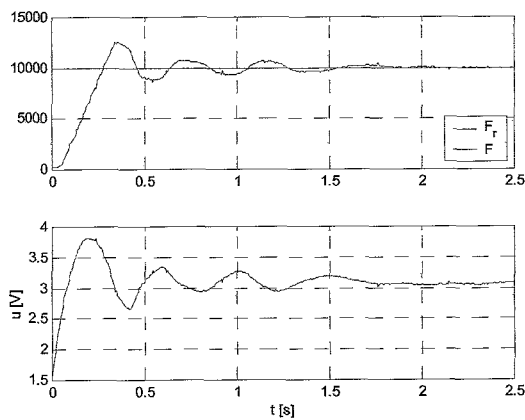
De totale regelactie wordt nu:

$$u = P(F_r - F) + \frac{P}{\tau_i} \int_0^t (F_r - F) dt \quad (4.20)$$

De fout $F_r - F$ moet binnen de toleranties liggen binnen de gestelde tijd (3 [s]).

4.3.3 Resultaten

In figuur 4.6 staat een experiment weergegeven met een PI-regelaar.



Figuur 4.6: Krachtverloop met PI-regelaar, SISO-4

In dit experiment wordt een stapvormige referentie gevolgd, dus dit is de snelste responsie die te halen valt met deze PI-regelaar.

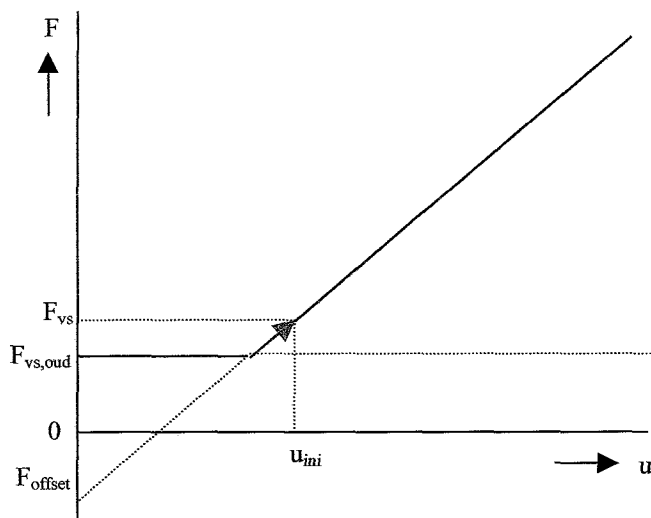
In het eerste deel van de responsie is een rechte lijn in de grafiek van F te zien. Deze komt door beperking van de actuator (oliepomp). Een snellere krachtopbouw in dit deel van de responsie moet gezocht worden in verbetering van de actuator, bijvoorbeeld door een grotere oliepomp of toepassing van accumulatoren. Omdat in het begin van het experiment veel rek in de constructie is, moet de pomp relatief veel olie naar de cylinders pompen. Omdat dit een constante debiet-pomp is, ontstaat het rechthoekige krachtverloop. Verder is hier de aanname gedaan dat de wet van Hooke voor de belaste constructie geldt. Dit is ook te zien in figuren 4.3 en 4.4 bij de P-regeling.

Daarna is er een flinke overshoot van zo'n 25 % van F_{sp} , wat teveel is. Dit is echter op te lossen door F op te laten lopen volgens een (tragere) referentie. De uiteindelijke fout gaat naar nul.

De PI-regelaar vertoont een gedrag wat een factor 0,7 trager is dan de P-regelaar (figuren 4.3 en 4.4). De doorschot is gelijk, alleen dempen slingeren minder snel uit.

4.4 Initialisatieroutine

Alvorens de kracht F op te laten lopen tot het setpoint, eerst een integrerende actie te laten werken die de voorspanningskracht langzaam ophooft tot een door de gebruiker gekozen waarde F_{vs} (zie figuur 4.7).



Figuur 4.7: Stationaire verband tussen u en F

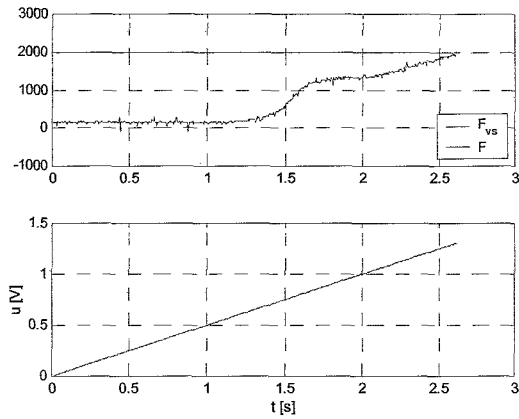
Op deze manier wordt het systeem eerst in het lineaire gebied (-) gebracht, alvorens de echte regeling op te starten. In formulevorm wordt de regelwet voor de initialisatieroutine:

$$u_{ini} = u_v \cdot T \int_0^t \text{sign}(F_{vs} - F) d\tau \quad (4.21)$$

met:

$$u_v \quad : \quad du/dt \text{ [V/s]}$$

Dit is vooral nodig bij een regelaar met I-actie, als het in het eerste deel van het stationaire verband tussen u en F , F niet reageert op veranderingen van u (rood). Een integrerende actie zal zich "opladen" en als het lineaire gebied binnengetreden wordt zal de integraal reeds een waarde hebben, waardoor een schok ontstaat. Dit gebeurt bij de initialisatie nog steeds, maar met een lagere snelheid, waardoor de schok verkleind wordt. (figuur 4.8). Het is duidelijk dat het volgedrag tot 2 [s] niet-lineair is.



Figuur 4.8: Krachtverloop tot F_{sp} met initialiseroutine

Ook voor een P-regeling heeft deze initialisatie nut. Door de verkregen eindwaarde voor u_{int} op $t=t_2$ (figuur 4.9) te gebruiken als feedforward, is de constante F_{offset} reeds geschat, mits γ (o.a. niet-lineaire verschijnselen) klein genoeg is. Er is dan echter nog steeds een feedforward-term nodig om de stationaire overdracht tussen F en F_{sp} gelijk aan 1 te krijgen. Dit eist echter nog steeds een goed (stationair) model van het systeem om goed volgedrag te krijgen (B/D moet geschat worden). Het verschil met de feedforward van paragraaf 4.2 ligt erin dat de constante term met F_{offset} nu ieder experiment opnieuw geschat wordt. Alleen in het onderste stuk van het werkgebied wordt geschat, maar wel ieder experiment opnieuw. De kwaliteit van het volgedrag is nu robuust voor langzame veranderingen tussen de experimenten.

De regelwet na initialisatie wordt bij een P-regelaar:

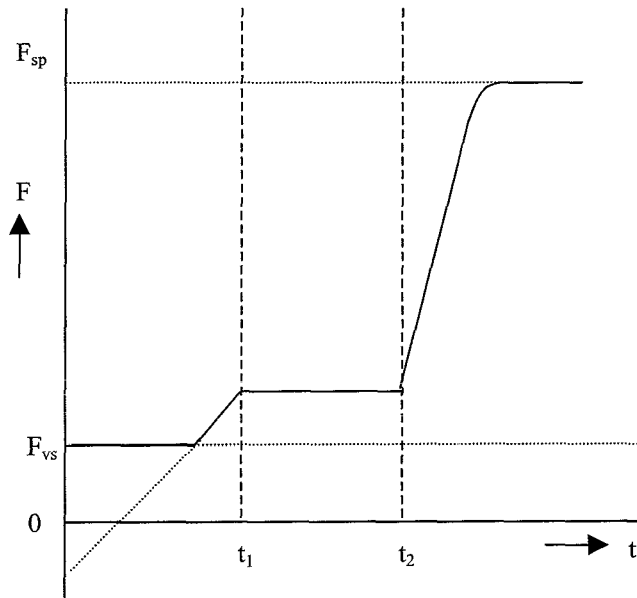
$$u = u_{ini} + \frac{B}{D} F_{sp} + P(F_{sp} - F) \quad (4.22)$$

Bij een feedforward van de eindwaarde van u_{int} op $t=t_2$ samen met een PI-regelaar wordt de volgende regelwet verkregen:

$$u = u_{ini} + P(F_{sp} - F) + \frac{P}{\tau_i} \int_0^t (F_r - F) d\tau \quad (4.23)$$

De kracht zal nu dus oplopen in de tijd als volgt (zie figuur 4.9):

1. $t_0 \leq t < t_1$ Initialiseroutine tot gewenste voorspanning (F_{vs})
2. $t_1 \leq t < t_2$ Opslaan voorspannings-parameters en opstarten P(I)-regelkring
3. $t_2 \leq t < t_3$ Regelloop blijven doorlopen tot het setpoint bereikt wordt, dan gedurende T_{wait} het setpoint vasthouden.



Figuur 4.9: Voorspelde krachtverloop met initiële integrerende actie

Het voordeel van deze aanpak is dat bij een voldoende goed model, de integrerende actie in het tijdsdeel na t_2 weggelaten kan worden. De constante constante afwijkingen die optreden bij een P-regeling reeds geschat zijn door de integrerende regelacties voor t_1 . Er moet dan echter nog steeds een truc (bijvoorbeeld feedforward) gebruikt worden om de statische overdracht tussen F_{sp} en F gelijk aan 1 te krijgen.

4.5 Schatter

Het is verder mogelijk om een online schatter mee te laten lopen tijdens het eerste deel van het experiment, die de constante termen compenseert tijdens het tweede deel van de krachtopbouw.

4.5 Regelaarkeuze

De keuze voor een PI-regelaar wordt gemaakt omdat het op dit moment beschikbare model niet goed genoeg is om een modelgebaseerde feedforward actie te gebruiken. Dit is noodzakelijk bij een P-regelaar en daarom valt deze dus af als mogelijkheid. Dit op basis van de resultaten uit paragraaf 4.2.2. Daarom zal er nu een initialiseringsroutine zoals vermeld in paragraaf initieel de constante termen te schatten en het systeem in het lineaire gebied te brengen, zoals vermeld in paragraaf 4.3.4. Daarna zal de PI-regelaar ingeschakeld worden, die het setpoint F_{sp} realiseert.

5. Krachtverloop

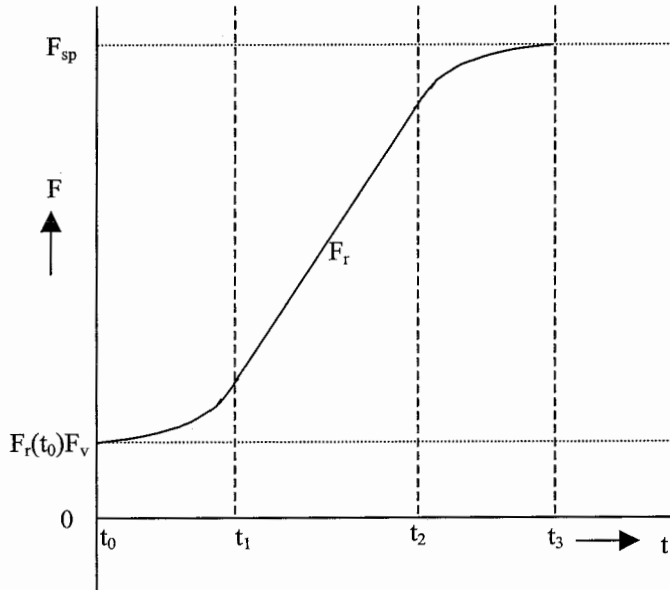
5.1 Referentiekracht

Om aan de performance-eisen te kunnen voldoen, is het nodig aan te kunnen geven op welk moment alle SISO-systemen op hun setpoint aankomen en hoe hun krachtopbouw zal verlopen.

Daarom zal er aan trajectorieplanning gedaan worden. Het is namelijk de bedoeling dat de gebruikte SISO-systemen vrijwel gelijk op het setpoint arriveren om de constructie zo kort mogelijk op maximale kracht F te belasten.

Er is gekozen voor een realiseerbaar traject om intree-effecten te minimaliseren. Verder zal de I-actie niet instantaan opladen door een initiële fout van $F_{sp} - F_{vs}$.

De referentiekracht F_r ziet er als volgt uit (Figuur 5.1):



Figuur 5.1: Referentietrajectorie F_r

De referentie is kwadratisch in de tijd, voldoet aan:

$$F_r = F_0 + v_{\max} * t + \frac{1}{2} a_{\max} * t^2 \quad (5.1)$$

v_{\max} : Maximale snelheid waarmee de kracht opgevoerd wordt [F/s]
 a_{\max} : Maximale acceleratie waarmee de kracht opgevoerd wordt [F/s²]

De referentie F_r is een aaneenkoppeling van drie tijdstukken:

1. Maximale gewenste acceleratie

$$t_0 \leq t < t_1 \quad F_r(t_0 \rightarrow t_1) = F_r(t_0) + \frac{1}{2} a_{\max} t^2 \quad (5.2a)$$

2. Maximale gewenste snelheid

$$t_1 \leq t < t_2 \quad F_r(t_1 \rightarrow t_2) = F_r(t_1) + v_{\max} (t - t_1) \quad (5.2b)$$

3. Maximale gewenste acceleratie

$$t_2 \leq t < t_3 \quad F_r(t_2 \rightarrow t_3) = F_r(t_2) + v_{\max} t_2 - \frac{1}{2} a_{\max} (t - t_2)^2 \quad (5.2c)$$

5.2 Tolerantie

Omdat F (voor alle SISO-systeem tegelijkertijd) een bepaalde wachttijd T_{wait} op of boven het geeiste setpoint F_{sp} moet zijn, alvorens het experiment geslaagd is, zal de $F_r(t_3)$ een waarde F_{tol} boven F_{sp} worden gepland.

In formulevorm:

$$F_r(t_3) = F_{sp} + F_{\text{tol}} \quad (5.3)$$

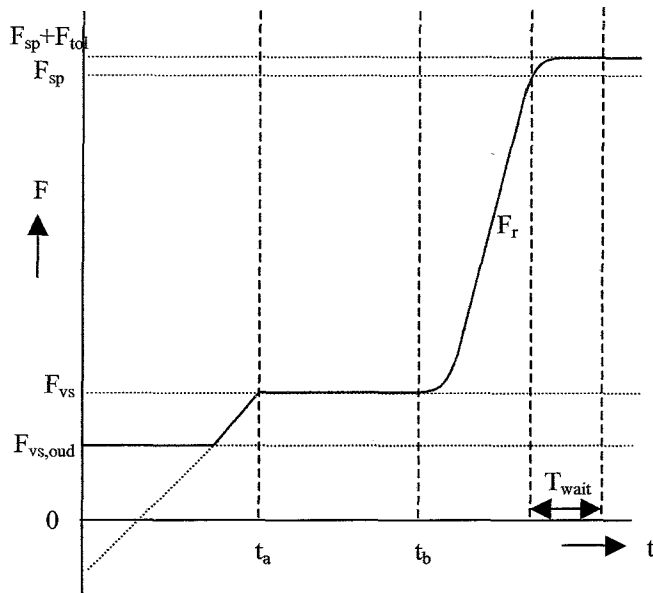
F_{tol} wordt nu gekozen als minimaal 2 keer de standaarddeviatie van de meetruis plus een factor voor andere kleine fluctuaties in de kracht die voor kunnen komen (experimenteel te bepalen).

De wachttijd T_{wait} wordt als volgt geïmplementeerd.

Er wordt ieder sample gecontroleerd of alle SISO-systemen aan $F > F_{sp}$ voldoen. Er wordt gecontroleerd of er over een periode T_{wait} geen enkel SISO-systeem onder F_{sp} uit komt. Mocht dit het geval zijn wordt de telling opnieuw begonnen.

5.3 Totale krachtverloop

Het totale (voorspelde) krachtverloop komt er nu als volgt uit te zien (figuur 5.2):



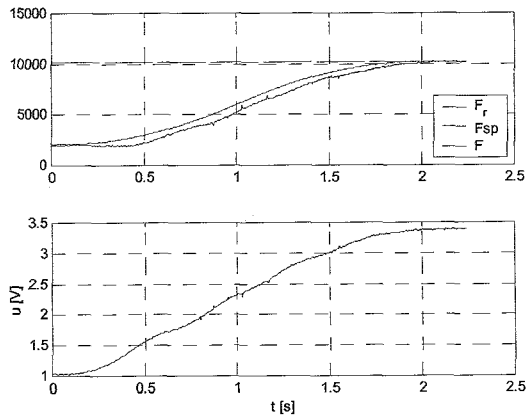
Figuur 5.2: Voorspelde krachtverloop met initiële integrerende actie en referentie

De referentietrajectorie F_r voor het SISO-systeem dat het grootste setpoint F_{sp} moet realiseren wordt op bovenstaande manier berekend. De referenties voor de overige SISO-systemen worden berekend door de referentie van het snelste SISO-systeem zodanig te schalen dat t_b voor alle SISO-systemen gelijk blijft en dat F_r loopt van F_{vs} tot F_{sp} .

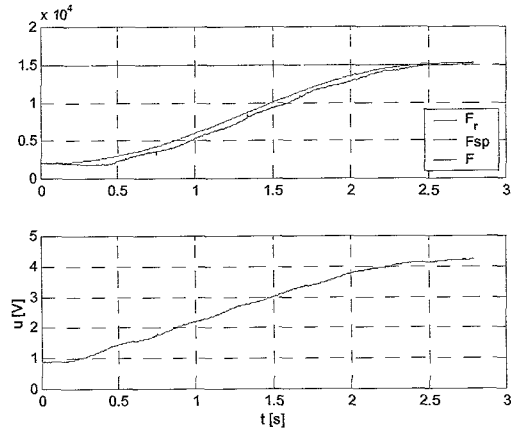
5.4 Resultaten

De experimentele resultaten van een PI-regelaar vanaf tijdstip t_2 (figuur 4.9) staan in figuur 4.10. Er zijn vrijwel geen ongewenste inschakelverschijnselen zoals schokken waar te nemen. Verder is er geen ongewenste overshoot. Omdat de regelaar pas begint te werken wanneer er een fout $F_{sp} - F$ optreedt, is er een achterstand op de referentie. Deze wordt door de I-actie vervolgens weer geëlimineerd.

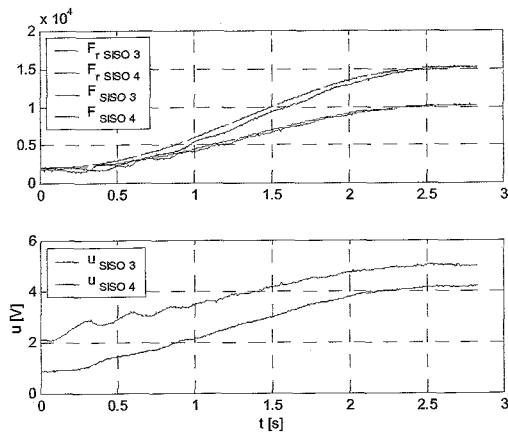
De snelheid van de referentietrajectories is beperkt. Ten dele om overshoot te vermijden, veroorzaakt in de closed loop van klepversterkers met regelventielen.



Figuur 4.9: Krachtverloop PI-regelaar na initialisatie, SISO-4



Figuur 4.10: Krachtverloop PI-regelaar na initialisatie, SISO-4



Figuur 4.11: Krachtverloop PI-regelaar na initialisatie, SISO-3 en SISO-4

Het huidige geregelde systeem voldoet in deze eerste tests aan de gestelde performance eisen:

- Geen overshoot
- Een statische afwijking van maximaal 200 N om robuustheid in te bouwen zodat de kracht 0,2 seconden boven het geeiste setpoint gehouden kan worden.
- Maximaal binnen 0,5 [s] (zie figuur 4.6) op kracht, maar er wordt een referentie gevolgd die ervoor zorgt dat de SISO-systemen binnen 2,5 s op kracht zijn, om niet over de beperkingen van de actuator (oliepomp) heen te gaan bij een grote vraag aan olieflow (rek in de te testen constructie).
- Vloeiend (zonder schokken) aanbrengen van de kracht.
- Alle SISO-systemen komen vrijwel gelijk op hun setpoint en dit wordt 0,3 [s] aangehouden.

6. Regeling-implementatie

6.1 Besturingssysteem

Enkele mogelijkheden met betrekking tot de regelaarimplementatie en voor-/nadelen die onmiddellijk duidelijk zijn geworden:

- D-Space systeem in samenwerking met MATLAB onder Windows
Voordelen: Hoge samplefrequenties mogelijk, gebruiksvriendelijk, snelle implementatie mogelijk.
Nadelen: Kosten van apparatuur en licenties
- Windows-programma onder Microsoft Windows NT
Voordelen: Gebruiksvriendelijke interface
Nadelen: Geen constante en hoge sample-frequentie mogelijk zonder speciale device-drivers te schrijven die dit mogelijk maken. Dit is niet haalbaar binnen deze stage.
- MS-DOS-programma in een dos-venster onder Microsoft Windows NT
Voordelen: Geen nieuw besturingssysteem nodig
Nadelen: Geen constante en hoge samplefrequentie mogelijk, dus onbruikbaar
- MS-DOS-programma onder MS-DOS
Voordelen: Sample-frequenties hoog te kiezen.
Nadelen: Gebruikersinterface niet gebruiksvriendelijk, wisselen van besturingssysteem voor rapportage naar Windows NT niet gebruikersvriendelijk (opnieuw opstarten met opstartdiskette)

De regelaar wordt op de pc geïmplementeerd onder het besturingssysteem MS-DOS. Hiervoor wordt gekozen omdat onder dit besturingssysteem totale controle over de hardware en regelacties op specifieke tijdstippen kunnen worden afgedwongen.

6.2 Programmatuur

Er is een programma in de programmeertaal C geschreven, waarmee metingen binnengehaald kunnen worden en nieuwe regelacties berekend kunnen worden, die weer naar de actuator uitgestuurd kunnen worden.

Er wordt een samplefrequentie van 200 [Hz] gekozen. De maximale snelheid die gehaald wordt in de stapresponsies in bijlage 3 is $36 \cdot 10^3$ [N/s]. Bij een setpoint van 10000 [N] en een oneidig grote versnelling (worst-case), duurt de opzetbeweging 0.28 [s]. Om dit waar te kunnen nemen geeft het Nyquist-criterium:

$$f_{nyq} = \frac{2}{0.28} = 7 \text{ Hz}$$

Een samplefrequentie van 200 Hz zit hier dus een factor 30 boven en dit is dus ruim voldoende.

Het deel van het C-programma dat de regeling uitvoert, is weergegeven in bijlage 9.

Verder zijn er nog een aantal programma's geschreven voor calibratie-doeleinden en dergelijke.

Een opsomming van de gebruikte programma's en bestanden staat in bijlage 10.

6.3 Gebruikers-interface

Voordat een experiment wordt uitgevoerd wordt de PC opgestart onder MS-DOS, waarna het rekenprogramma MATLAB wordt opgestart. De te gebruiken sensoren worden hysteresevrij neergelegd, waarna een calibratieprogramma wordt opgestart. Nu worden de nul-offsets van de sensoren gemeten, die worden opgeslagen.

Nu worden de instellingen voor het experiment ingegeven en opgeslagen.

Nu wordt het regelprogramma gestart (zie bijlage 9), dat aangeeft dat trajectories toont en na aanzetten van de oliepomp kan het experiment gestart worden door de druk op een toets.

Na een experiment worden de resultaten gevisualiseerd met MATLAB, onder MS-DOS.

Na een reeks experimenten wordt de pc opnieuw opgestart met Windows NT, waarin de testrapporten kunnen worden gegenereerd met een script dat draait onder het programma Testpoint.

Een uitgebreide beschrijving van de door de gebruiker in acht te nemen procedures staan in bijlage 11.

Het zou wellicht gebruiksvriendelijker zijn om in plaats van de PC steeds opnieuw op te starten om van besturingssysteem te wisselen, een tweede pc te gebruiken om alleen te meten, dataverwerking en wellicht instellingen door te geven aan de regel-PC.

7. Conclusies en aanbevelingen

Het is gebleken dat de performance-problemen van de oorspronkelijke configuratie samen hingen met te lange en niet constante sampletijden, wat veroorzaakt werd door het besturingssysteem Windows NT. Daarom is de overstap gemaakt naar het besturingssysteem MS-DOS, waar met 200 [Hz] gesample wordt.

Er is een regulaar ontworpen op basis van een worst-case situatie qua stabiliteit, wat inhoudt dat er een robuustheids-marge is ingebouwd voor onbekende verschijnselen.

Het huidige geregelde systeem voldoet aan de gestelde performance eisen:

- Geen overshoot
- Een statische afwijking van maximaal 200 N om robuustheid in te bouwen zodat de kracht 0,2 [s] boven het geeiste setpoint gehouden kan worden.
- Maximaal binnen 0,5 [s] op kracht, maar er wordt een referentie gevolgd die ervoor zorgt dat de SISO-systemen binnen 2,5 [s] op kracht zijn, om overshoot te vermijden.
- Vloeiend (zonder schokken) aanbrengen van de kracht.
- Alle SISO-systemen komen vrijwel gelijk op hun setpoint, dat 0,3 [s] wordt aangehouden.

De huidige beschikbare modellen zijn niet nauwkeurig genoeg om feed-forward toe te passen wat nodig is bij een proportionele regeling. Aangezien dit wel de snelst mogelijke regeling is, zou deze tot een verdere kwantitatieve verbetering kunnen bijdragen. Er wordt dan wel een stuk robuustheid met betrekking tot verandering van systeemparameters opgeofferd.

Als er flow wordt gevraagd van de hydraulische oliepomp, zakt hiervan de druk snel in (bij rek van de belastte constructie). Een snelle drukopbouw is hierdoor niet mogelijk, omdat er achter de drukventielen geen hogere druk kan heersen dan de toevoerdruk (bij de oliepomp). Pas als de cylinders minder flow nodig hebben komt de druk van de oliepomp weer op peil, en kan de krachtopbouw verder worden gerealiseerd.

Mogelijke verbeteringen kunnen een oliepomp met grotere capaciteit, en/of accumulatoren liefst per SISO-systeem één, om tevens de interactie tussen de SISO-systemen te verminderen.

Deze accumulatoren kunnen een kortstondige vraag aan flow opvangen, met minder drukval.

Een manier om de vraag aan flow ten dele te verminderen is het vervangen van de synthetische trekbanden door stalen kettingen, die minder rek hebben.

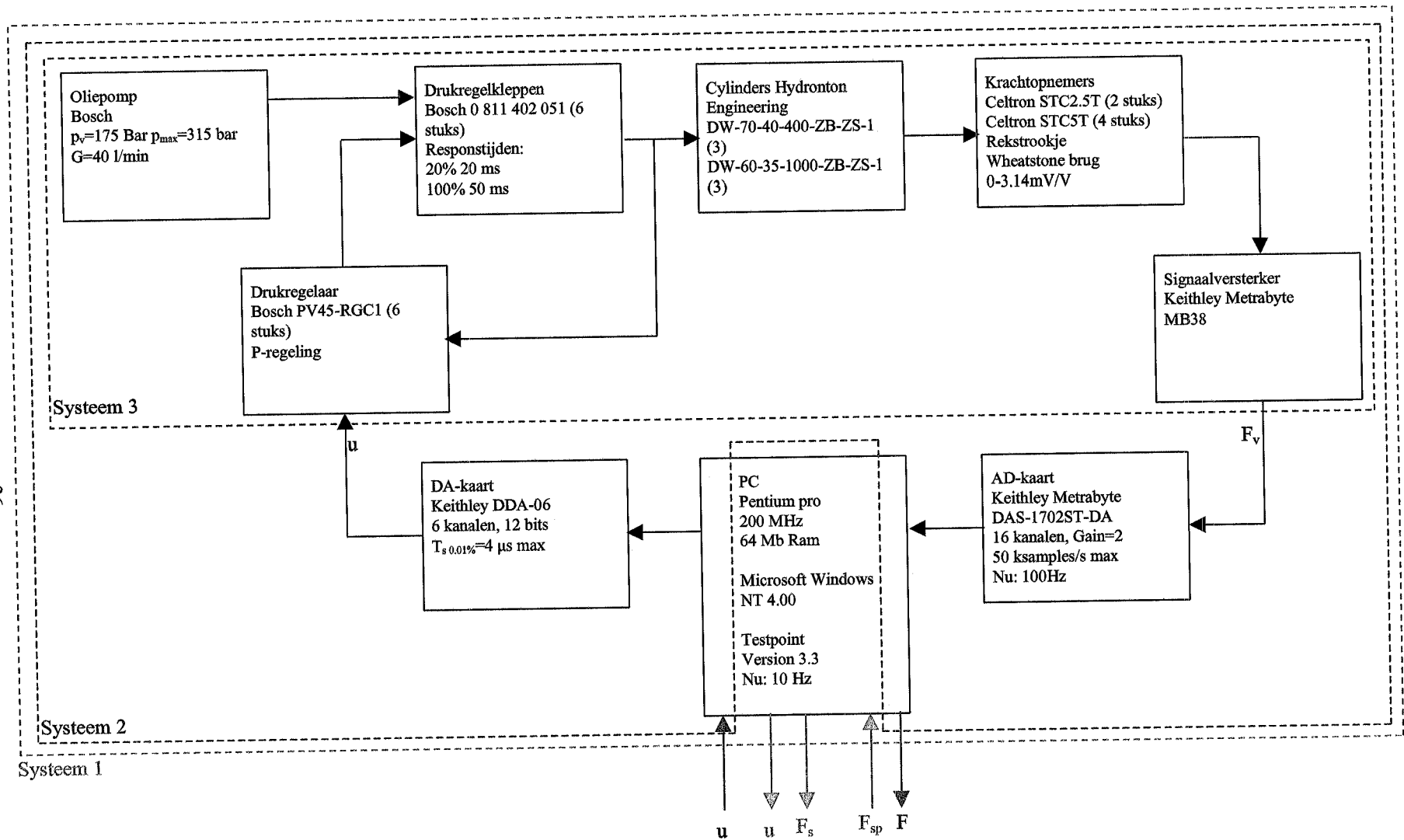
In de huidige configuratie is er een tweede closed-loop met klepversterker en regelventiel aanwezig. Om volle controle over het systeemgedrag te krijgen, zouden de klepversterkers uitgeschakeld kunnen worden, zodat de kleppen rechtstreeks aangestuurd kunnen worden. De snelheid van de referentietrajectories moet nu nog beperkt worden om overshoot te vermijden, veroorzaakt in deze tweede closed loop.

Om de gebruiksvriendelijkheid te verhogen zou het wellicht beter zijn om een tweede PC te gebruiken voor enkel de instellingen, metingen en dataverwerking.

In verband met de veiligheid is het raadzaam om een onafhankelijke veiligheidsvoorziening te treffen in de vorm van een noodstop, die de oliepomp uit kan schakelen. Dit moet kunnen zonder achter het veiligheidsscherm te hoeven komen. Verder zou het aan de veiligheid bijdragen als er buiten de testruimte een aanduiding is, die aangeeft dat de trekbank in bedrijf is.

Bijlage 1 Schema opstelling trebank B-Style Automotive

Hieronder staat schematisch de opbouw van de oorspronkelijke opstelling. Er is 1 van de 6 SISO systemen getekend. De totale opstelling bestaat uit 6 maal System 3, met een gezamenlijke oliepomp.



Bijlage 2 Meetplan Trekbank B-Style Automotive m.b.v. SigLab 28/09/2000

Als meetapparaat wordt een DSP-SigLab gebruikt.

Eerst zullen er een aantal stapresponsies gegenereerd worden door de PC. Dit zal gebeuren bij verschillende waarden van u om te kunnen valideren in hoeverre de stationaire relatie tussen u en F_v van Systeem 3 lineair is. Hier kan verder uit achterhaald worden welk deel van de dode tijd in Systeem 3 veroorzaakt wordt. De I-regelaar zal op de Systeem 1 gezet worden om te kunnen valideren in hoeverre de door de eigen PC getoonde u ook werkelijk geproduceerd wordt. Er wordt dan aan Systeem 3 u en F_v gemeten.

Er zullen een aantal langzaam variërende signalen voor u gebruikt worden om de minimaal benodigde waarde voor u te vinden, waarbij een verandering in F plaatvindt.

Om het dynamische gedrag van het systeem te kunnen karakteriseren zullen nog een aantal snellere signalen op u (Systeem 3) gezet worden door middel van de SigLab.

Stap door PC					
Pompdruk 100 bar	Cylinder #2	Samplefreq. 512 Hz	Meettijd 8 s		
Meting #	Staphoogte [V]	Testpoint bestand	Databestand	Opmerkingen	
1	1	Stap1.tst	Stappc01a.vos		
2	2	Stap2.tst	Stappc02a.vos		
3	3	Stap3.tst	Stappc03a.vos	Trekband misschien geslipt	
4	4	Stap4.tst	Stappc04a.vos	Trekband heeft geslipt	
5	1	Stap1.tst	Stappc01b.vos.		
6	2	Stap2.tst	Stappc02b.vos		
7	3	Stap3.tst	Stappc03b.vos		
8	4	Stap4.tst	Stappc04b.vos	Trekband slipt, cylinder tegen aanslag	

Trekband beter opgerold, 2 metingen met een stap van 4 [V], databestanden: Stappc04b1.vos en Stappc04b2.vos

P-controller door PC					
Pompdruk 100 bar	Cylinder #2	Samplefreq. 512 Hz	Meettijd 8 s		
Meting #	Setpoint [N]	Testpoint bestand	Databestand	Opmerkingen	
9	5000	PC1.tst	Pcpc1a.vos	Trigger misschien wat laat	
10	10000	PC2.tst	Pcpc2a.vos		

Extra meting stap 2 [V], databestand Stappc02e.vos, dus hogere voorspanning als initieel.

Stap door Siglab					
Pompdruk 100 bar	Cylinder #2	Samplefreq. 512 Hz	Meettijd 8 s		
Meting #	Staphoogte [V]	Siglab bestand	Databestand	Opmerkingen	
11	1		Stapsl01a.vos	Niet uitgevoerd, stap PC goed genoeg.	
12	2		Stapsl02a.vos	Niet uitgevoerd, stap PC goed genoeg.	
13	3		Stapsl03a.vos	Niet uitgevoerd, stap PC goed genoeg.	
14	4		Stapsl04a.vos	Niet uitgevoerd, stap PC goed genoeg.	
15	1		Stapsl01b.vos.	Niet uitgevoerd, stap PC goed genoeg.	
16	2		Stapsl02b.vos	Niet uitgevoerd, stap PC goed genoeg.	
17	3		Stapsl03b.vos	Niet uitgevoerd, stap PC goed genoeg.	
18	4		Stapsl04b.vos	Niet uitgevoerd, stap PC goed genoeg.	

Stap door Siglab, hogere druk					
Pompdruk 175 bar	Cylinder #2	Samplefreq. 512 Hz	Meettijd 8 s		
Meting #	Staphoogte [V]	Siglab bestand	Databestand	Opmerkingen	
19	2		Stapsl02c.vos	Niet veel verschil met 100 bar, als allereerste meting uitgevoerd	

Extra meting sinus, Sinsl01.vos
 Extra meting sinus, Sinsl02.vos, hogere frequentie, hogere offset
 Extra meting offset aanzetten, Nulsl01.vos

Triangle door Siglab						
Pompdruk 100 bar		Cylinder #2	Samplefreq. 512 Hz	Meettijd 8 s		
Meting #	Hoogte [V]	Freq [Hz]	Siglab bestand	Databestand	Opmerkingen	
19	4	0.125		Trisl04.vos	Andere frequentie gepakt, andere amplitude.	

Extra meting triangle, Trisl02.vos, hogere frequentie.

Het uitgangssignaal van SISO-2 is met de volgende formule om te rekenen naar de bijbehorende kracht:
 Sensorschaling:

$$F = 9547 \cdot F_v - 360$$

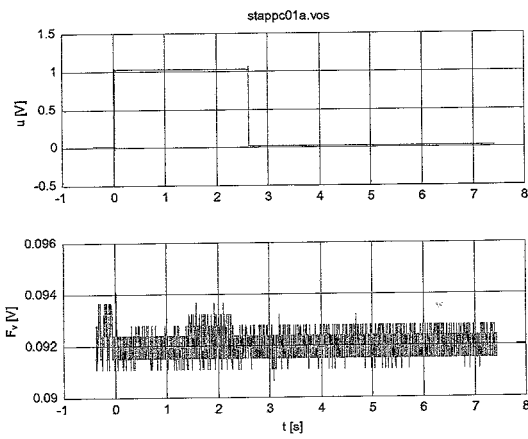
(B2.1)

met:

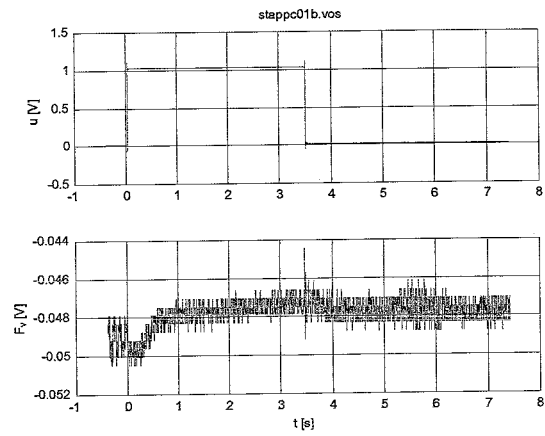
F=Kracht [F]

F_v=Krachtinformatie als spanning [V]

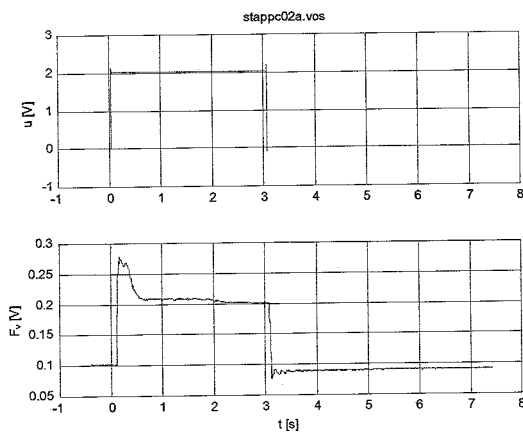
Bijlage 3 Resultaten metingen 28/9/2000



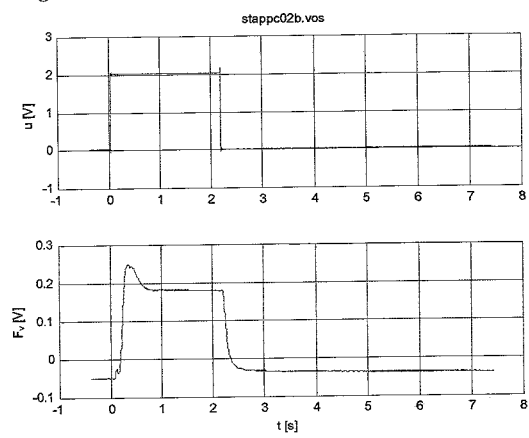
Figuur B3.1



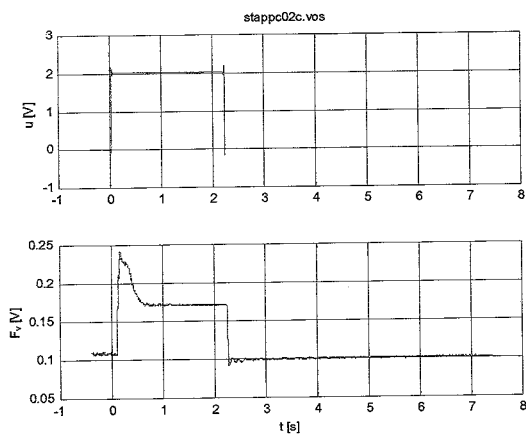
Figuur B3.2



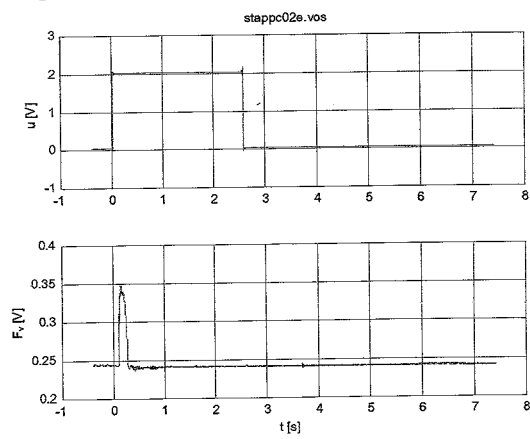
Figuur B3.3



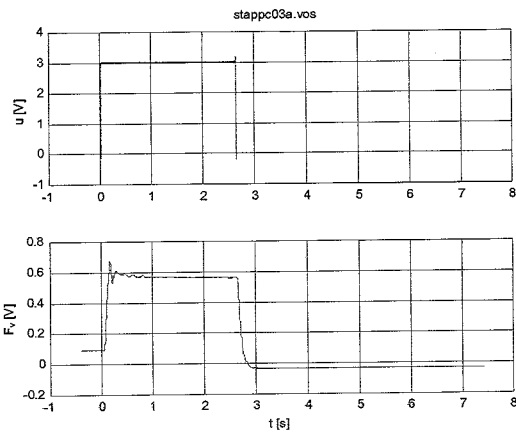
Figuur B3.4



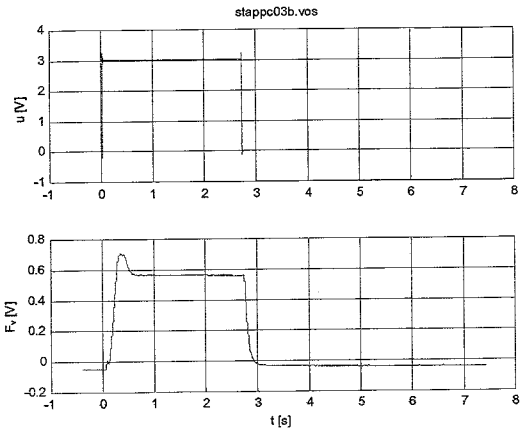
Figuur B3.5



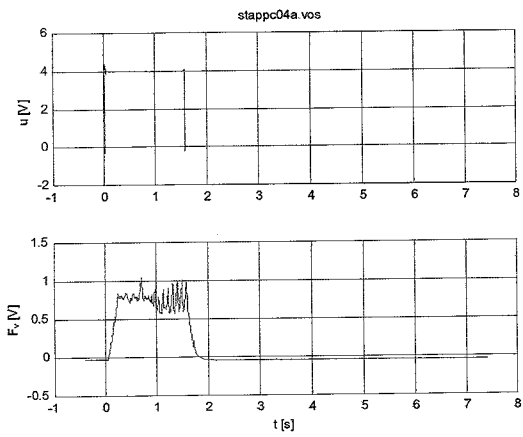
Figuur B3.6



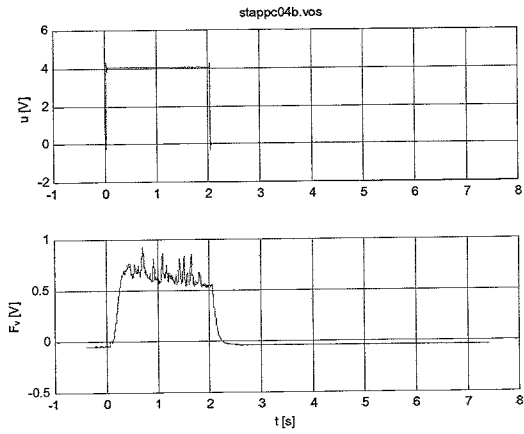
Figuur B3.7



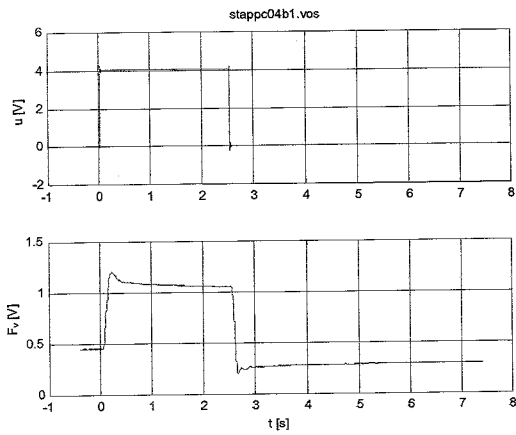
Figuur B3.8



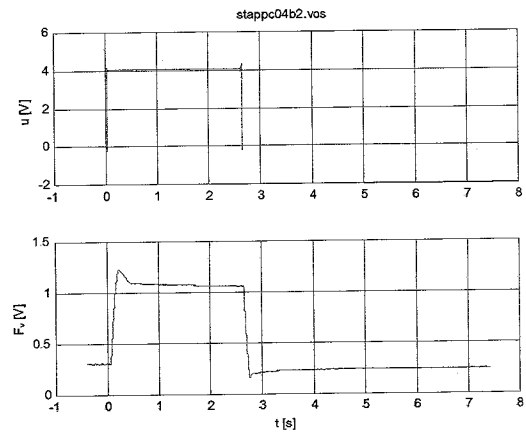
Figuur B3.9



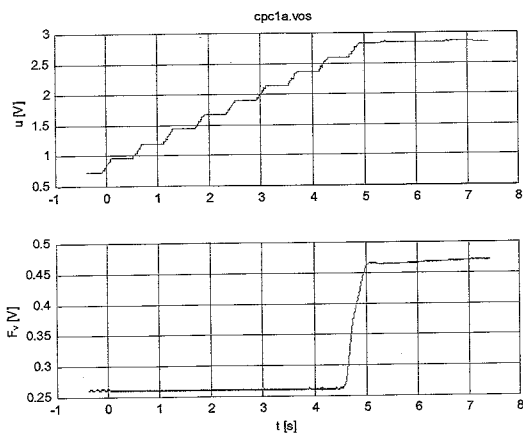
Figuur B3.10



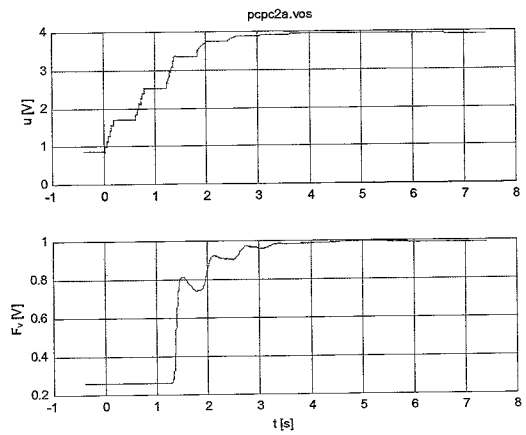
Figuur B3.11



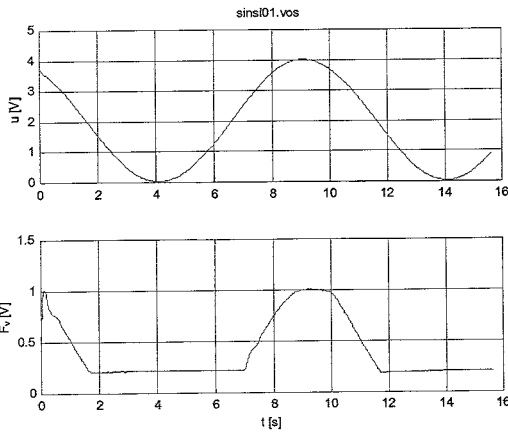
Figuur B3.12



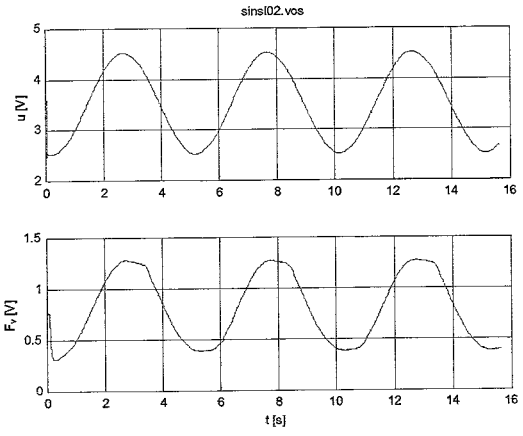
Figuur B3.13



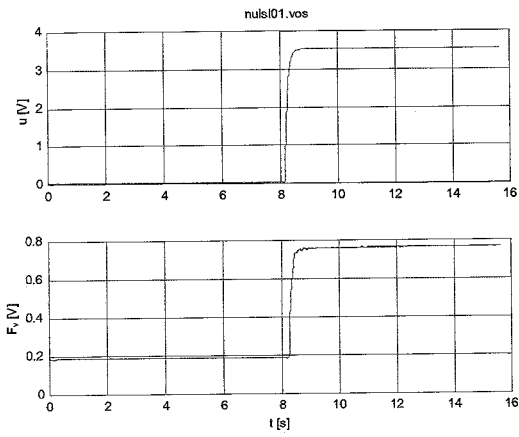
Figuur B3.14



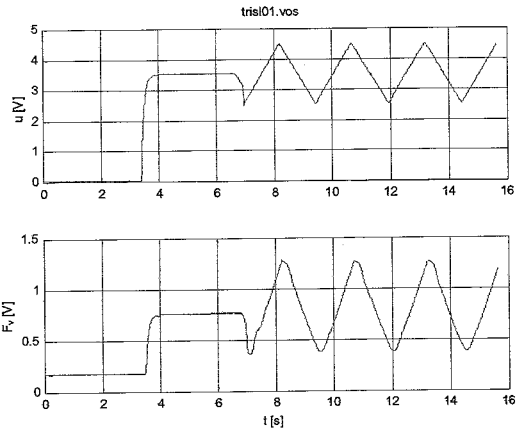
Figuur B3.15



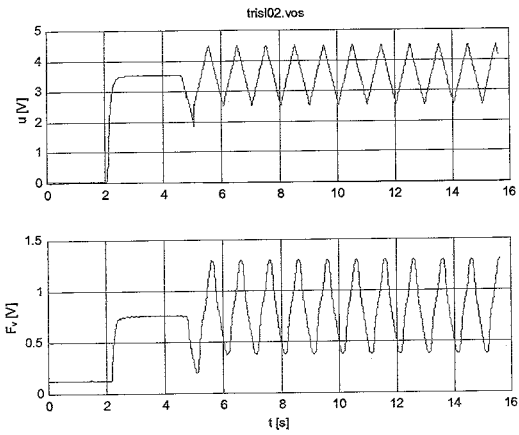
Figuur B3.16



Figuur B3.17



Figuur B3.18



Figuur B3.19

Bijlage 4 Data voor fitten eerste-orde model

Staphoogte Ingang [V]	Eindwaarde Uitgang [V]	Piekwaarde Uitgang [V]	Databestand
2.02	0.20	0.28	Stap02a.vos
2.02	0.18	0.25	Stap02b.vos
2.02	0.17	0.24	Stap02c.vos
3.02	0.56	0.71	Stap03b.vos
4.01	1.05	1.20	Stap04b1.vos
4.02	1.06	1.23	Stap04b2.vos

Hierbij zijn de volgende data niet meegenomen:

Databestand	Reden
Stap01a.vos	Voorspanning hoger dan opgewekte krachtsignaal
Stap01b.vos	Voorspanning hoger dan opgewekte krachtsignaal
Stap02e.vos	Te hoge voorspanning
Stap03a.vos	Slippende riem bij meting
Stap04a.vos	Slippende riem bij meting
Stap04b.vos	Slippende riem bij meting

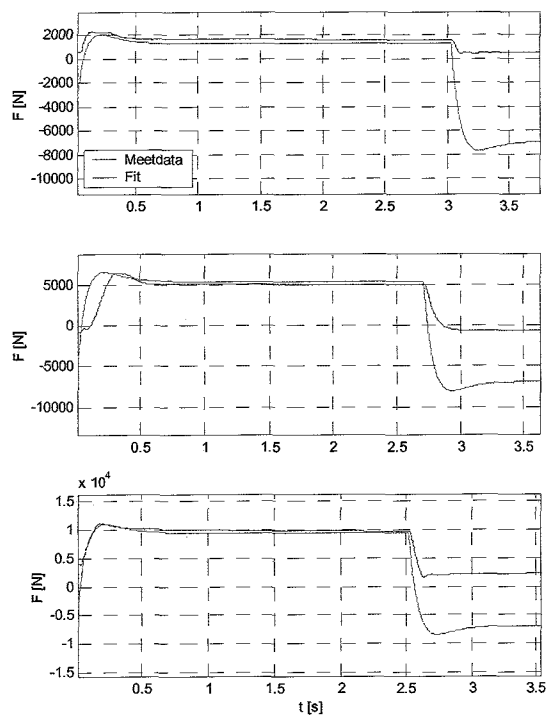
Bijlage 5 Fit van 2^e orde-model

Dode tijd-data

Data-bestand	τ [s]
stappc02a.vos	0.068
stappc03a.vos	0.050
stappc04b1.vos	0.043

Resultaten

Stap02a.vos, Stap03b.vos, Stap04b1.vos.



Figuur B5.1

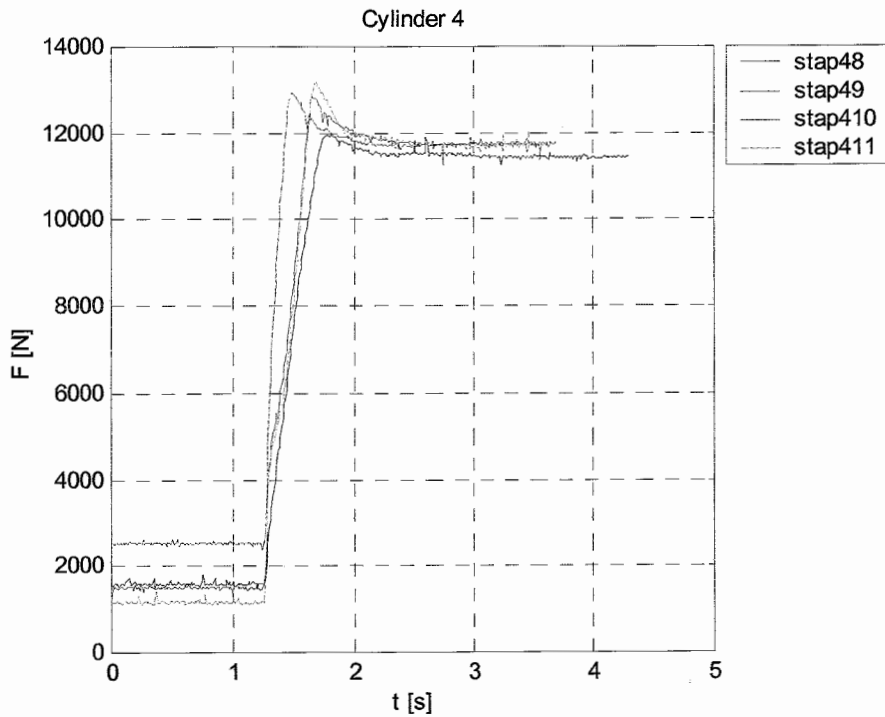
Modelparameters voor SISO-2:

A	= 20	[1/s]
B	= 100	[1/s ²]
C	= $75 \cdot 10^3$	[N/Vs]
D	= $4,1 \cdot 10^5$	[N/Vs ²]
F_0	= 7000	[N]
τ	= 0.05	[s]

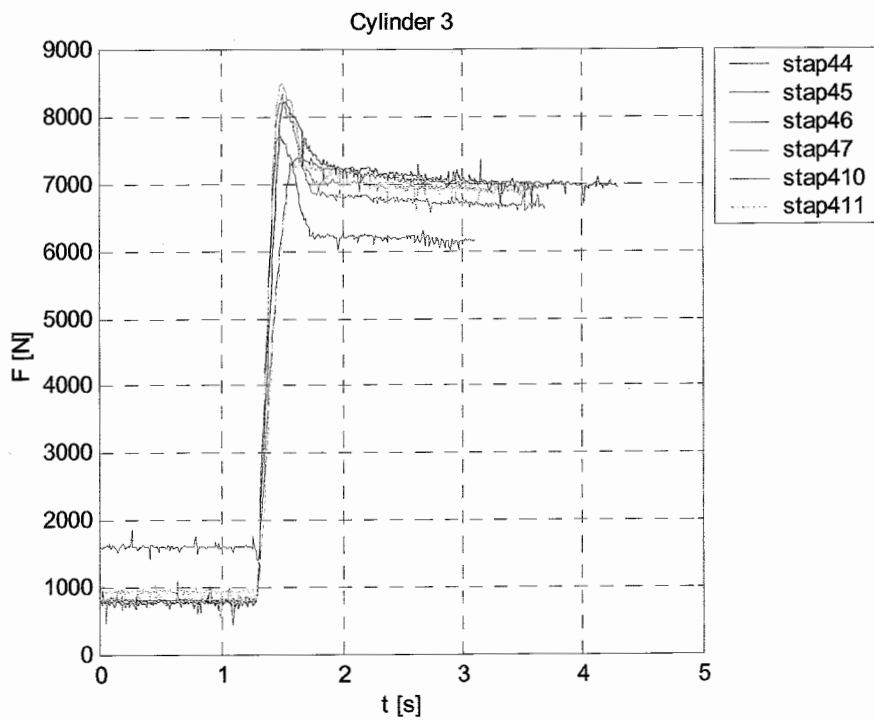
Bijlage 6 Meetplan Trekbank B-Style Automotive 19/10/2000

Stap door PC					
Samplefreq. in 10Hz	Samplefreq. uit 100Hz	Data 5			
Meting #	Staphoogte [V]	Cylinder	Databestand	Opmerkingen	
1	3-4	2	Stap41a.tst Stap41b.tst	Korte trekband 100 bar	
2	2-5	2	Stap52a.tst Stap52b.tst	Korte trekband 175 bar	
3	0	Allemaal	Stap03a Stap03b	Nultest	
4	4	3	Stap44a Stap44b	Lange trekband, 110 bar	
5	4	3	Stap45a Stap45b	Terugschroeven gain klepversterker, 0.5 slag, 110 bar	
6	4	3	Stap46a Stap46b	Terugschroeven gain klepversterker, 1 slag (1.5 totaal), 110 bar	
7	4	3	Stap47a Stap47b	Gain klepversterker weer terug naar beginstand, 110 bar	
8	4	4	Stap48a Stap48b	Lange trekband, 100 bar	
9	4	4	Stap49a Stap49b	Korte trekband, 100 bar	
10	4	3 en 4	Stap410a Stap410b	2 cylinders tegelijk, trekband 3 lang, 4 kort, 100 bar	
11	4	3 en 4	Stap411a Stap411b	2 cylinders tegelijk, trekband 3 lang, 4 kort, 175 bar	

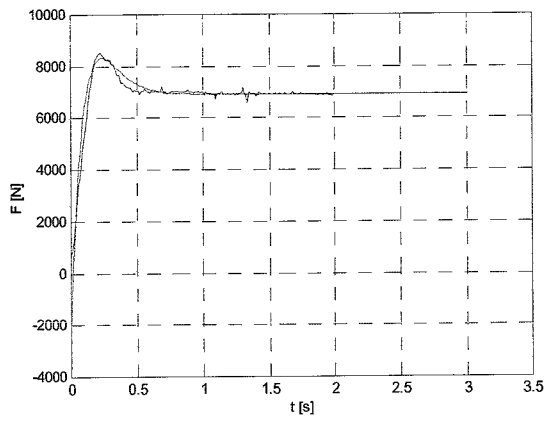
Bijlage 7 Worst-case systeem-identificatie experimenten



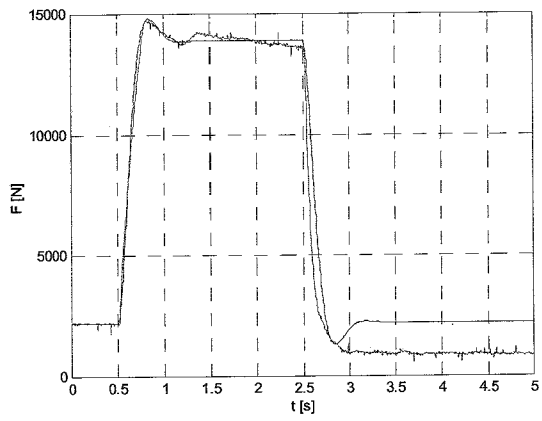
Figuur B7.1 Worst case experimenten cylinder 4



Figuur B7.2 Worst case experimenten cylinder 3



Figuur B7.3 Worst case fit cylinder 3



Figuur B7.3 Worst case fit cylinder 4

Bijlage 8 Worst-case systeem- en regelaar-parameters

Cylinder 3

Systeemparemeters:

$$A=16.1$$

$$B=79.5$$

$$C=3.66 \cdot 10^4$$

$$D=1.81 \cdot 10^5$$

$$F_{\text{offset}}=2.17 \cdot 10^3$$

Regelaarparameters:

Proportioneel

$$P=5 \cdot 10^{-4}$$

Proportioneel en integrerend

$$P=3.5 \cdot 10^{-4}$$

$$\tau_i=0.07$$

Cylinder 4

Systeemparemeters:

$$A=15$$

$$B=140$$

$$C=0$$

$$D=4.1 \cdot 10^5$$

$$F_{\text{offset}}=2.2 \cdot 10^3$$

Regelaarparameters:

Proportioneel

$$P=5 \cdot 10^{-4}$$

Proportioneel en integrerend

$$P=1.6 \cdot 10^{-4}$$

$$\tau_i=0.07$$

Bijlage 9 C- regelaarimplementatie

```

/*      regelvs.c
*/

#include <tce.h>      /* voor ml_ en ts_-functies      */

#include "exp.c"      /* Onze experimenteer omgeving in de      */
                    /* vorm van de functies exp_get, */
                    /* exp_init en exp_put.      */

                    /* Regelparameters hier (als globale      */
                    /* variabelen) omdat ze zowel in */
                    /* regelwet als in main gebruikt worden.*/

double dt,          /* sample tijd      */
t;                /* huidige tijdstip      */
double P[NZ];      /* Proportionele versterking      */
double tau[NZ];    /* Referentie signalen op een tijdstip      */
double ref[NZ];
double power[NA];
double klaar[NZ];
double klaartot;
double time;
double z[NZ];      /* Metingen op een tijdstip, NZ uit exp.c*/
double u[NA];      /* gerealiseerd commando naar de actuator*/
double uc[NA];
double u_comp[NA];
double Twait;
double Fsp[NZ];
int good_count;
double z_cum[NZ];
double z_gem[NZ];
double Fvs[NA];
double dudt;
double uvs[NA];
int break_reason;

void regelwet(void) {
/*
Functie, die een (nieuwe) waarde voor de actuator berekent
en retourneert.
*/

u[0] = uc[0] = power[0]*(u_comp[0]+uvs[0]+P[0]*( (ref[0]-exp_z[0]) + 1/taui[0]*exp_int_e[0]));
u[1] = uc[1] = power[1]*(u_comp[1]+uvs[1]+P[1]*( (ref[1]-exp_z[1]) + 1/taui[1]*exp_int_e[1]));
u[2] = uc[2] = power[2]*(u_comp[2]+uvs[2]+P[2]*( (ref[2]-exp_z[2]) + 1/taui[2]*exp_int_e[2]));
u[3] = uc[3] = power[3]*(u_comp[3]+uvs[3]+P[3]*( (ref[3]-exp_z[3]) + 1/taui[3]*exp_int_e[3]));
u[4] = uc[4] = power[4]*(u_comp[4]+uvs[4]+P[4]*( (ref[4]-exp_z[4]) + 1/taui[4]*exp_int_e[4]));
u[5] = uc[5] = power[5]*(u_comp[5]+uvs[5]+P[5]*( (ref[5]-exp_z[5]) + 1/taui[5]*exp_int_e[5]));
}

int checkvsv (void) {
int ncheck,i,j;
ncheck = 10;

for (i=0; i<NZ; i++) {
z_cum[i] = 0;
}
exp_init(dt);
for (j=0; j<ncheck; j++) {
exp_put(u);
exp_get();
for (i=0; i<NZ; i++) {
z_cum[i] = z_cum[i] + exp_z[i];
}
}
for (i=0; i<NZ; i++) {

```

```

        z_gem[i]=z_cum[i]/ncheck;
    }
    printf("Voorspanningskracht checked...\n");
    printf("Fvs=%6.0lf,%6.0lf,%6.0lf,%6.0lf,%6.0lf,%6.0lf\n\n",z_gem[0],z_gem[1],z_gem[2],z_gem[3],z_gem[4],z_gem[5]);
}

int      voersp(void) {
double   timevs;
double   klaarvs[NZ];
double   klaarvstot;
int      klaarvstime;
int      i;

    for (i=0; i<NA; i++) {
        if ( power[i]==0 ) {
            klaarvs[i]=1;
        }
        if ( power[i]==1 ) {
            klaarvs[i]=0;
        }
    }
    /* als klaar is 1 -> klaar */
    /* 0 betekent nog niet klaar */

    /* printf("Klaar? (0=nee, 1=ja) 1:%6.2lf, 2:%6.2lf, 3:%6.2lf, 4:%6.2lf, 5:%6.2lf,
6:%6.2lf\n\n",klaarvs[0],klaarvs[1],klaarvs[2],klaarvs[3],klaarvs[4],klaarvs[5]);*/

    break_reason=0;
    klaarvstot=0;
    timevs=0;
    klaarvstime=0;
    while ( 1 ) {
        exp_put(u);
        /*printf("Merkteken!!\n\n");*/
        exp_get();
        for (i=0; i<NA; i++) {
            if( klaarvs[i]<0.5 ) {
                if ( exp_z[i]>Fvs[i]) {
                    klaarvs[i] = 1;
                    /*printf("klaar\n");*/
                }
                else {
                    u[i]=uc[i]=uc[i]+dudt*dt;      /*0.25V per seconde */
                    klaarvs[i] = 0;
                    /*printf("plus\n");*/
                }
            }
        }
        klaarvstot=0;
        for (i=0; i<NA; i++) {
            klaarvstot=klaarvstot+klaarvs[i];
        }

        if ( klaarvstot>(NA-0.5) ){
            klaarvstime=klaarvstime+1;
        }
        else klaarvstime=0;

        if ( klaarvstime>10 ) {
            break_reason=1;
            break;
        }
        timevs=timevs+dt;
        if ( timevs>10 ) {
            break_reason=2;
            break;
        }
        for (i=0; i<NA; i++) {
            uvs[i]=uc[i];
        }
    }

    if ( break_reason==2 ) {

```



```

        printf("Voorspanning: Timeout!\n"); /*20 seconden wachten, dan timeout */
    }
    if ( break_reason==1 ) {
        printf("Voorspanning aangebracht...\n");
    }
}

printf("Klaar? (0=nee, 1=ja) 1:%2.2lf, 2:%2.2lf, 3:%2.2lf, 4:%2.2lf, 5:%2.2lf, 6:%2.2lf\n, Aantal klaar:
%2d\n\n",klaarvs[0],klaarvs[1],klaarvs[2],klaarvs[3],klaarvs[4],klaarvs[5],klaarvstot);
printf("u=%6.2lf,%6.2lf,%6.2lf,%6.2lf,%6.2lf,%6.2lf\n\n",uc[0],uc[1],uc[2],uc[3],uc[4],uc[5]);
}

int main(void) {
/*
Hoofdprogramma. Verklaart zichzelf (?).
*/
    int i,j,n;

    P[0]=0;
    P[1]=0;
    P[2]=0;
    P[3]=0;
    P[4]=0;
    P[5]=0;

    tau[0]=1e9;
    tau[1]=1e9;
    tau[2]=1e9;
    tau[3]=1e9;
    tau[4]=1e9;
    tau[5]=1e9;

    ml_open("cl",0); /* Van MATLAB: */
    ml_get_vec(P,NA,"P");
    ml_get_vec(tau,NA,"tau");
    dt = ml_get_scalar("dt",0.005); /* sample tijd dt */
    n = (int)ml_get_scalar("n",1000); /* aantal samples n */
    dudt = ml_get_scalar("dudt",0.25);

    ml_open("regelin",0); /* Van MATLAB: */
    ts_load0(ref,NA,"ref");
    ml_get_vec(power,NA,"power");
    ml_get_vec(Fsp,NA,"Fsp");
    Twait = ml_get_scalar("Twait",0.2); /* sample tijd dt */
    ml_get_vec(Fvs,NA,"Fvs");

/*
printf("Parameter load done...\n\n");*/

    for ( i=0; i<NA; i++) u[i]=uc[i]=0;

    exp_init(dt); /* met metingen voor t=0 */
    voorsp();
    checkvs();
    for (i=0; i<NA; i++) {
        u_comp[i]=-P[i]*(Fvs[i]-z_gem[i]); /*initiele fout corrigeren (zie regelwet)*/
    }

/* Maak tijdreeksen aan voor t, u en z */
    ts_init0(&t,1,n,"t"); /* array &t, lengte 1, inhoud t.*/
    ts_init0(u,NA,n,"u"); /* analoog .. */
    ts_init0(uc,NA,n,"uc"); /* analoog .. */
    ts_init0(exp_z,NZ,n,"z"); /* array z, lengte NZ */

/*printf("Time Series created...\n\n");*/

    for (i=0; i<NA; i++) {
        if ( power[i]==0 ) {
            klaar[i]=1;
        }
        if ( power[i]==1 ) {
            klaar[i]=0;
        }
    }
}

```

```

    }
}
/* 0 betekent nog niet klaar */

/*printf("Klaar? (0=nee, 1=ja) 1: %6.2lf, 2:%6.2lf, 3:%6.2lf, 4:%6.2lf, 5:%6.2lf,
6:%6.2lf\n\n",klaar[0],klaar[1],klaar[2],klaar[3],klaar[4],klaar[5]);*/

printf("Loop started, wait for %4.1lf",n*dt);
printf(" seconds...\n\n");

t = 0;
ts_get_all(); /* Referentie signalen voor t=0 */

good_count=0;
break_reason=0;
klaartot=0;
time=0;
while ( 1 ) {
    for ( j=0; j<NZ; j++ ) {
        exp_r[j] = ref[j]; /* Referentiewaarden aan exp.c doorgeven */
    }
    regelwet(); /* Berekend commando uc */
    for ( i=0; i<NA; i++ ) {
        if( klaar[i]<0.9 ) {
            if ( exp_z[i]>Fsp[i] ) {
                klaar[i] = 1;
                printf("Klaar, Cyl %6d = %6.0lf [N] \n",i,exp_z[i]);
            }
            else {
                klaar[i] = 0;
                /*printf("niet klaar\n");*/
            }
        }
    }

    exp_put(u); /* Gerealiseerd commando u */

    ts_put_all(); /* Alle tijdreeksen bijwerken */
    t = t+dt;
    ts_get_all(); /* Volgende referentie signalen */
    exp_get(); /* Metingen op volgende sample */
    /* moment */

    klaartot=0;
    for ( i=0; i<NA; i++ ) {
        klaartot=klaartot+klaar[i];
    }
    if ( klaartot>(NA-0.5) ) {
        good_count=good_count+1;
    }
    else good_count=0;

    if ( good_count>Twait/dt ) {
        break_reason=1;
        break;
    }

time=time+dt;
    if ( time>10 ) {
        break_reason=2;
        break;
    }
}

/*
for ( i=0; i<Twait/dt; i++ ) {
    exp_put(u);
    ts_put_all();
    t=t+dt;
    ts_get_all();
    exp_get();
}
*/

uc[1]=0;
uc[2]=0;
uc[3]=0;
uc[4]=0;
uc[5]=0;

```

```

uc[6]=0;
exp_put(uc); /* Actuator uit ... */
exp_get();

if ( break_reason==2 ) {
    printf("Timeout!\n\n"); /*20 seconden wachten, dan timeout */
}
if ( break_reason==1 ) {
    printf("Test terminated successfully!\n\n");
}
printf("klaar=%6.2f,%6.2f,%6.2f,%6.2f,%6.2f,%6.2f,klaartot=%6.2f\n\n",klaar[0],klaar[1],klaar[2],klaar[3],klaar[4],klaar[5],klaartot);

ml_open("regeluit",1); /* Naar MATLAB: */
ts_save_all(1); /* alle tijdreeks data */

printf("Data saved.\n\n");

ts_fini();
return 0;
}

/* end of regelvs.c */

```

Bijlage 10 Gebruikte programma's en bestanden

Bestanden die aangemerkt zijals (oud), zijn bestanden die niet meer gebruikt worden, deze zijn tussenstappen om te komen tot de huidige implementatie.

Bestand	Gebruikt	Genereert	Beschrijving
checkvs.c meet.exe	-	checkvs.mat	Voorspanning controleren, gemiddelde van 100 metingen.
exp.c	-	-	Gebruikt bij compilatie van alle andere c-bestanden.
meet.c meet.exe			(oud)
meet2.c meet2.exe			(oud)
regel.c regel.exe	regel_c.mat	regel_r.mat	Regeling PI-regelaar naar referentie
regelm.c regelm.exe			Regeling PI-regelaar naar referentie, wachttijd na $F_{sp} < F_{ref}$ (oud)
regelp.c regelp.exe	regel_c.mat	regel_r.mat	Proportionele regeling, alleen cylinder 3 en 4 (oud)
regelvs.c regelvs.exe	regelin.mat	regeluit.mat	Uiteindelijke regelprogramma, met initialisatie, voorspanningscheck, referentie, setpoint, wachttijd.
stuur.c stuur.exe	stuur.mat	stuur.mat	Sturing-programma, om Cylinders met u [V] aan te sturen en F [N] te meten. (Bijvoorbeeld stapresponsies)
test.c test.exe	-	-	(oud)
voorsp.c voorsp.exe	voorsp.mat	voorsp.mat	Initialisatie-programma (oud)

uit deze C-bestanden kan met het commando "c" een "exe" bestand gecompileerd worden. Voorbeeld bestand.c: "c bestand", dus zonder extensie geeft: bestand.exe

Bestand	Gebruikt	Genereert	Beschrijving
analyse.m	regeluit.mat	Plaatjes	Tekent grafieken van trek.m nogmaals
cal.m	cal.mat	cal.dat cal.mat	Calibratie-bestand, met de hand of automatisch
checkvs.m	checkvs.exe	-	Voorspanningsmeting
cl.m	-	cl.mat	Regelaarinstellingen (niet aanpassen)
cyl1.m	stuur.exe cyl3tune.mat	cyl1.m	Tunen SISO-systeem 1
cyl2.m	stuur.exe cyl4tune.mat	cyl2.m	Tunen SISO-systeem 2
cyl3.m	stuur.exe cyl3tune.mat	cyl3.m cyl3tune (opt.)	Tunen SISO-systeem 3
cyl4.m	stuur.exe cyl4tune.mat	cyl4.m cyl4tune (opt.)	Tunen SISO-systeem 4
cyl5.m	stuur.exe cyl3tune.mat	cyl5.m	Tunen SISO-systeem 5
cyl6.m	stuur.exe cyl4tune.mat	cyl6.m	Tunen SISO-systeem 6
makedat.m	regeluit.mat	ingang.dat uitgang.dat	Maakt DAT-files voor in Testpoint, staan in map "c:\trekpr\result".
makeref.m	-	-	Berekent referentie F_{ref}
meet.m	meet.exe	meet.mat	Metingen binnenhalen (oud)
regel.m	regel.exe	regel_c.mat	(oud)
regelm.m	regelm.exe	regel_c.mat	(oud)
regelp.m	regelp.exe	regel_c.mat	proportionele regelaar-aansturing(oud)
regelvs.m	regelvs.exe	regelin.mat	regelvs.exe vanuit 1 bestand(oud)

set.m	settings.m	settings.m	Opent settings.m
settings.m	-	-	Instellingen trekproef
stuur.m	stuur.exe	stuur.mat	Stuur-experiment uitvoeren
trek.m	settings.m cl.mat	-	Trekproef uitvoeren
voersp.m	voersp.exe	-	Testprogramma voor voorspanningsophoging (oud)

De volgende programma's zijn aanwezig:

Norton editor

ne.exe	ne.ini *.txt *.m *.c	-	ASCII-editor
ne.ini	-	-	

Program file manager

pfm.com	*.*	-	Copieren e.d.
---------	-----	---	---------------

Matlab

ml.exe	matlab.hlp *.m *.mat	-	Berekeningen
matlab.hlp	-	-	

C-compiler

c.bat	*.c	-	C-compiler
c\	-	-	

Bijlage 11 Procedures

Opstarten in MS-DOS

Opstartdiskette in drive a: en

- De computer aanzetten.
- of:
- vanuit Windows: Start-Afsluiten-De computer opnieuw starten.

Terug naar Windows

Opstartdiskette uit drive a: en Ctrl-Alt-Del.

MATLAB opstarten

c:\trekpr\ml

MS-DOS-commando's vanuit MATLAB

!

Voorbeelden

```
>>!ne settings.m  
>>!dir
```

Calibratie (sensor-offset bepalen vector B uit $F=AF_v+B$)

- Alle sensoren hysteresevrij neerleggen
- Vanuit MATLAB:
- Automatisch de B-vector (en x_offset) in cal.mat updaten
>>cal

Handmatige Calibratie

Vanuit MATLAB:

- cal.m wijzigen
>>!ne cal.m
- A-vector (of indien gewenst de B-vector wijzigen), berekening als bij Testpoint

Betekenis x_sens en x_offset in cal.m:

Omrekenen van bytes (-4096 tot 4096) naar Newtons (-50000N tot +50000N).

Voor iedere grootte x :

$F(\text{bytes})_{\text{meting}} = x_{\text{offset}} + x_{\text{sens}} * F(N)$

of andersom :

$F(N) = (F(\text{bytes}) - x_{\text{offset}}) / x_{\text{sens}}$

- Automatisch de A- en B-vector in cal.mat updaten
>>cal

Trekproef uitvoeren:

Vanuit MATLAB:

- settings.m openen en aanpassen
>>set *of*
>>!ne settings.m
- Trekproef uitvoeren
>>trek
- Referentiesignaal wordt getoond
- Oliepomp aanzetten

- Toets drukken om te starten met experiment
- Grafieken tonen resultaten
- Oliepomp uit
- Grafieken beoordelen
- data-file voor Testpoint maken
 >>makedat

- Optioneel: data opnieuw laten zien:
 >>analyse
- Optioneel: data copieren naar andere naam (bij sequentie van experimenten)
 >>!copy c:\trekpr\result\ingang.dat c:\trekpr\result\naam1.dat
 >>!copy c:\trekpr\result\uitgang.dat c:\trekpr\result\naam2.dat

Trekproef-resultaten verwerken met Testpoint

- Testpoint opstarten en verwerkprogramma openen
 c:\trekpr\Verwerkdata1.tst openen

Tunen SISO-systeem

Vanuit MATLAB:

- Testprogramma cyl1.m, cyl2.m, cyl3.m, cyl4.m, cyl5.m of cyl6.m starten
 bijv:
 >>cyl2
- Ingangssignaal wordt getoond
- Oliepomp aanzetten
- Toets drukken om te starten met experiment
- Grafieken tonen resultaten
- Oliepomp uit
- Grafieken beoordelen, rood=geteste SISO-systeem, blauw=referentiesysteem (3 of 4)
- Nieuwe instelling klepregelaar op basis van grafieken (zie paragraaf 3.3).
- Opnieuw testen totdat de grafieken over elkaar liggen.

MATLAB-commando's

Bewerking	MATLAB-Commando
Opstarten	c:\trekpr\ml
Afsluiten	>>quit
Aanwezige variabelen?	>>who
Help	>>help
Laden (bestand.mat)	>>load bestand
Grafiek tekenen (x of x tegen t)	>>plot(x) of >>plot(t,x)