

## 7 Regelaars

### DOELSTELLINGEN

- Aan de hand van een aan/uit regeling het begrip hysteresis toelichten.
- Het werkingsprincipe van de P-regelaar aan de hand van een voorbeeld toelichten.
- Het verband kennen tussen de versterkingsfactor van de P-regelaar en de breedte van de proportionele band.
- Het werkingsprincipe van de I-regelaar aan de hand van een voorbeeld kunnen toelichten.
- De begrippen integratietijd, nasteltijd en reset-rate verklaren.
- De eigenschappen van een PI-regelaar opgeven en grafisch interpreteren.
- Het werkingsprincipe van een D-regelaar aan de hand van een voorbeeld toelichten.
- De begrippen differentiatietijd en voorsteltijd kunnen verklaren.
- De eigenschappen van een PID-regelaar grafisch interpreteren.
- Aan de hand van een praktisch voorbeeld het werkingsprincipe van Fuzzy logic kunnen uitleggen.

### 7.1 Waarom regelen

We situeren nogmaals de regelaar in de regelkring en gebruiken vanaf nu terug het vereenvoudigd schema van de regelkring.

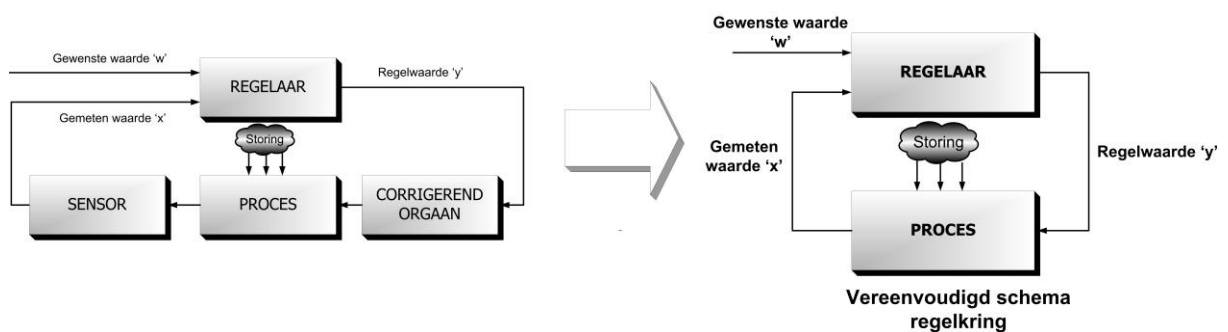


Fig. 7.1 Vereenvoudigd schema regelkring

De regelaar moet de gemeten waarde 'x' gelijk houden aan de gewenste waarde 'w'. De regelaar doet dit door met regelwaarde 'y' het proces in de juiste richting bij te regelen indien nodig.

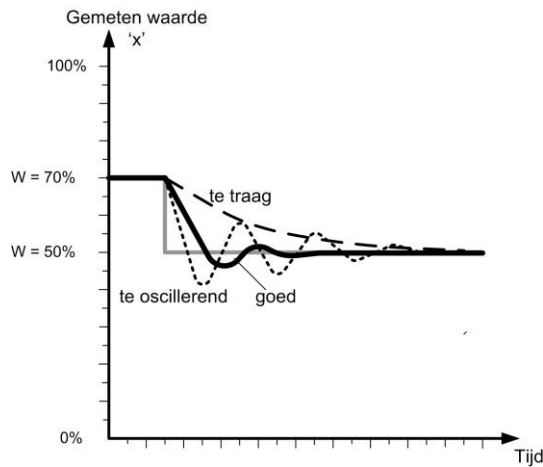
Er zijn verschillende types regelaars omdat niet alle processen op een even gemakkelijke wijze stabiel te houden zijn. Het proces bepaald hoofdzakelijk welk type regelaar nodig is en welke kwaliteiten deze regelaar moet bezitten.

Eisen, gesteld aan een regelaar:

- Snel ingrijpen
- Instellen van gewenste waarde moet mogelijk zijn
- De gemeten waarde zo stabiel mogelijk vasthouden op de gewenste waarde, onafhankelijk van belastingsveranderingen en storingen
- Een keuzeknop voor omschakeling van automatisch naar handbediening
- Een eenvoudige bediening van de regelaar

Er zijn 3 situaties waar de regelaar moet ingrijpen op de proceswaarde:

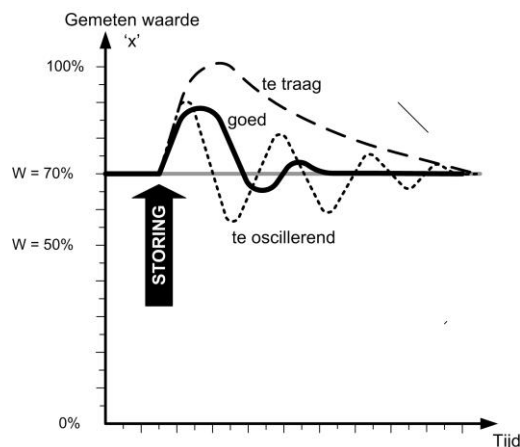
- Bij een verandering van de gewenste waarde (W of SP 'setpoint').



Bijvoorbeeld wanneer je de temperatuur van je huis enkele graden kouder wil instellen omdat je een feestje hebt en er veel volk in een kleine ruimte gaat zitten. Een goed afgestelde regelaar regelt deze bijstelling snel weg. Minder goed afgestelde regelaars regelen te traag, of te snel en dan krijgen we oscillaties.

Fig. 7.2 verandering setpoint

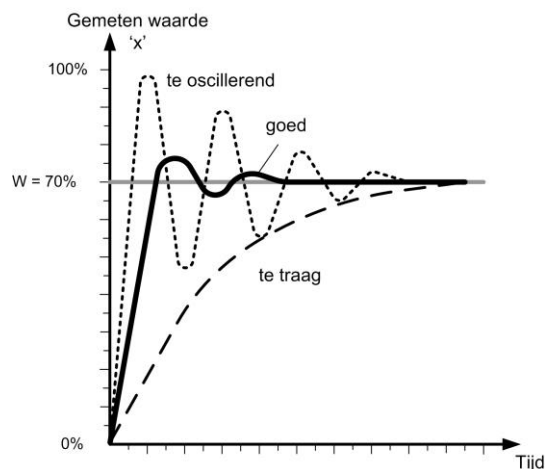
- Bij een fout, veroorzaakt door een storing.



Fouten kunnen door vele factoren veroorzaakt worden. Een goed afgestelde regelaar heeft een passend antwoord klaar om deze fouten snel terug weg te regelen. Minder goed afgestelde regelaars reageren ook weer te traag of te oscillerend.

Fig. 7.3 fout door storing

- Bij opstarten van een regelkring (proces en regelaar)



Opstarten van een regelkring moet vooral snel gebeuren en zonder dat we de gewenste waarde te veel overschrijden (overshoot). Minder goed afgestelde regelaars starten te traag op, of te snel, met het gevolg van oscillaties en te grote overshoot.

Fig. 7.4 opstarten

Te traag regelen:

- Het duurt te lang voordat de gemeten waarde gelijk is aan de gewenste waarde.
- Draaiende processen zullen tijdens deze tijd mogelijk slechtere producten afleveren en/of meer uitval hebben.
- Het voordeel is wel dat er hier absoluut geen overshoot kan zijn, en dat is voor sommige processen positief (liever wat trager, maar dan zonder het proces naar gevaarlijke waarden te drijven)

Te snel regelen:

- Dikwijls veel 'overshoot' (te ver boven de gewenste waarde komen). Dit is voor sommige processen, die niet ver boven de gewenste waarde mogen gaan, echt te mijden.
- Neiging tot oscillaties. De gemeten waarde komt meermaals boven en terug onder de gewenste waarde voordat het proces stabiel wordt.
- Het duurt te lang voordat het proces stabiel is en in de tussentijd worden er producten van mindere kwaliteit afgeleverd. In erge gevallen blijft de proceswaarde zelfs oscilleren.

## 7.2 Indeling regelaars

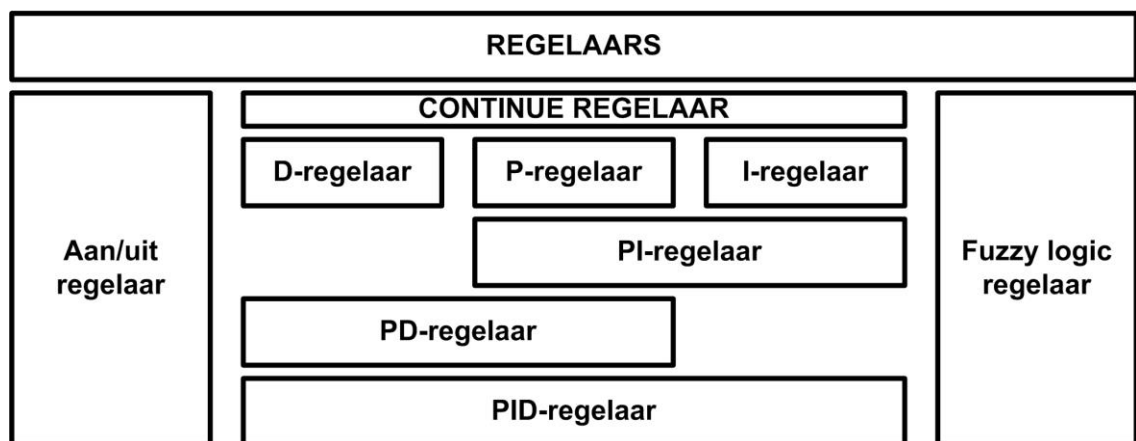


Fig. 7.5 indeling regelaars

Zoals je kan zien zijn er hier 3 grote blokken:

- De Aan/uit regelaars
- De continue regelaars met
  - P-regelaar
  - PI-regelaar
  - PD-regelaar
  - PID regelaar!!! (I en D zijn steeds vergezeld van een P-regelaar)
- De Fuzzy-logic regelaar

Soms zijn regelaars zelfs uitgerust met zowel aan/uit, continue als Fuzzy keuzemogelijkheden. Moderne regelaars zijn eigenlijk kleine computers en een regel-functie meer betekent gewoon dat het programma in de regelaar wat vergroot en dat er een keuzemenu bijkomt.

Binnen de continue, de zogenegde PID regelaars wordt er, meestal onbewust, ook nog een onderscheid gemaakt tussen seriële en parallelle regelaars. Aan de buitenkant van de regelaar kan je dit verschil enkel zien

aan de benamingen die bij de regelknoppen van de I- en van de D-regelaar staan.

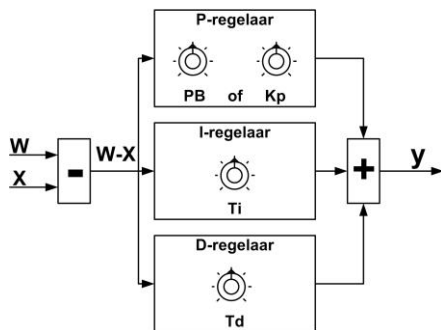


Fig. 7.6 parallelle PID regelaar

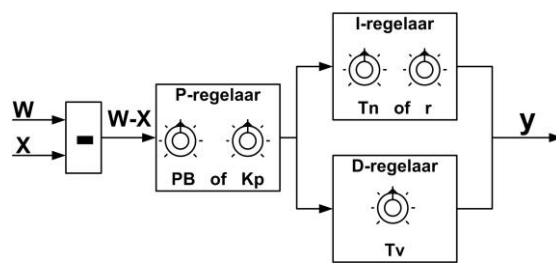


Fig 7.7 seriële PID regelaar

	De parallelle opstelling	De seriële opstelling
I-knop	Ti 'integratietijd'	Tn 'nasteltijd' of r 'reset-rate'
D-knop	Td 'differentiatietijd'	Tv 'voorsteltijd'

Bij de behandeling over de I en de D regelaar zal dit verder verduidelijkt worden.

### 7.3 De Aan/uit regelaar

De Aan/uit regelaar is de meest robuuste, goedkoopste, eenvoudigste en meest toegepaste regelaar. Hij wordt in tal van alledaagse producten gebruikt (koelkast, fritouse, koffiezet, waterkoker, wafelijzer, verwarming, elektrisch vuur,...)

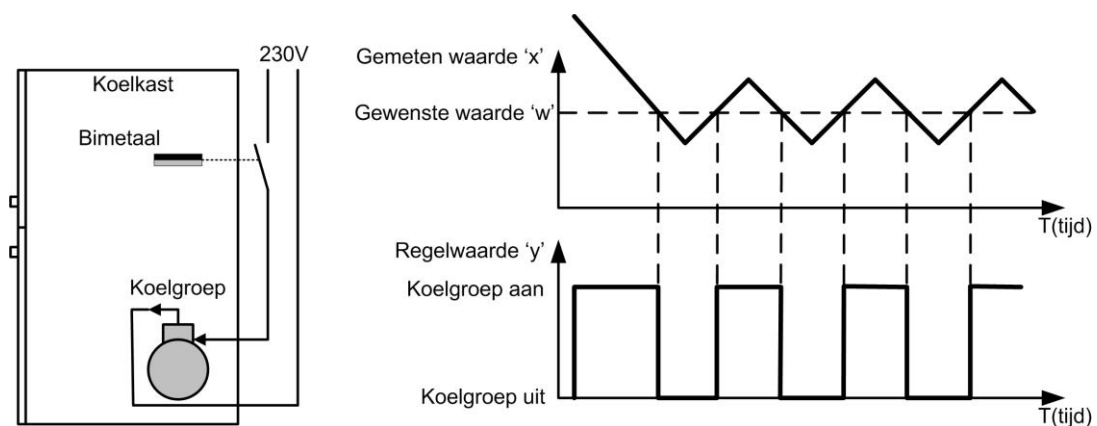


Fig. 7.8 Aan-uit regelaar koelkast

Als de temperatuur boven de gewenste waarde komt, schakelt de koelgroep van de koelkast aan en als de temperatuur onder de gewenste waarde daalt, schakelt de koelgroep uit.

Een aan/uit regelaar kan enkel worden toegepast bij trage / logge processen (ook processen met een grote capaciteit genoemd) omdat de gemeten waarde na de reactie van de aan/uit regelaar niet meteen terug onder en achteraf weer meteen terug boven de gewenste waarde mag komen. Dit zou een veel te zenuwachtige regeling zijn.

Voor processen met een kleine capaciteit kunnen we beter overschakelen op de continue regelaars.

Voor trage processen, echter met een niet echt stabiele proceswaarde, kiezen we best voor een aan/uit regelaar met hysteresis.

#### Aan/uit regelaar met of zonder hysteresis

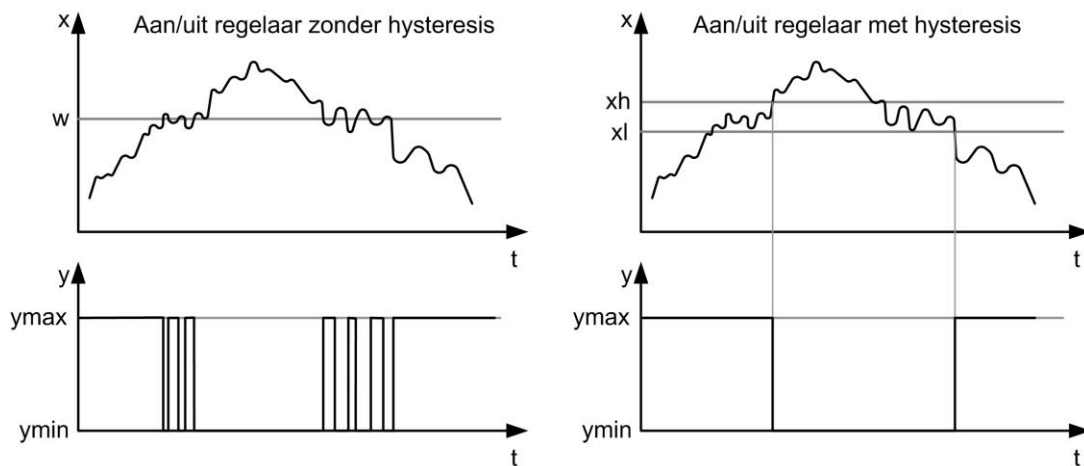


Fig. 7.9 Aan-uit zonder en met hysteresis

Je ziet bij de aan-uit regelaar zonder hysteresis duidelijk het zenuwachtig reageren op de waarden die rond de gewenste waarde liggen.

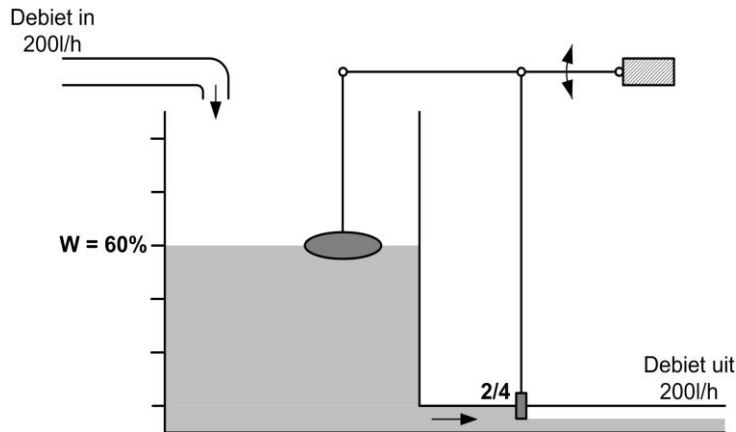
Bouwen we in de aan/uit regelaar nu wat hysteresis in, dan gaat de uitgang van de regelaar veranderen naar  $y_{\min}$  van het moment de gemeten waarde  $x_h$  bereikt heeft en zal de uitgang pas terug omslaan van het moment de gemeten waarde terug onder  $x_l$  gekomen is. Er zijn hier 2 omschakel niveaus en dit resulteert in een veel stabielere en rustigere omschakeling.

## 7.4 De P-regelaar

Definitie Proportionele regelaar:

**Een P-regelaar regelt afhankelijk van de grootte van de fout.**

De P-regelaar of proportionele regelaar is van de continue regelaars de enige die alleen kan werken.



Deze figuur is een mooi voorbeeld van een zelfwerkende P-regelaar. Zelfwerkend omdat deze regelaar al zijn energie uit het proces zelf haalt.

De bijstelling van de klep gebeurt evenredig of 'proportioneel' met de fout (= verschil tussen gemeten en gewenste waarde)

Fig. 7.10 zelfwerkende P-regelaar

Stijgt het niveau in de tank, dan zal de vlotter er via de hefboom voor zorgen dat de klep verder open komt staan zodat het niveau terug stabiliseert. Het nadeel van de P-regelaar is dat het niveau zich na een belastingsverandering (instroom bijvoorbeeld van 200l/h naar 300l/h) niet terug stabiliseert op de gewenste waarde, maar in dit geval een stukje erboven. Dit verschil noemen we statische offsetfout en dit zullen we verder nog behandelen.

De P-regelaar regelt ook niet persé over het volledige meetgebied (niveau van 0% tot 100%). Het gebied waarbinnen de P-regelaar werkt noemen we de Proportionele band 'PB'. Buiten dit gebied gedraagt de P-regelaar zich als gewone aan/uit regelaar.

Bij een P-regelaar stellen we het verband tussen de regelwaarde 'y' (klepstand) en de gemeten waarde 'x' (niveau tank) voor met de kenlijn.

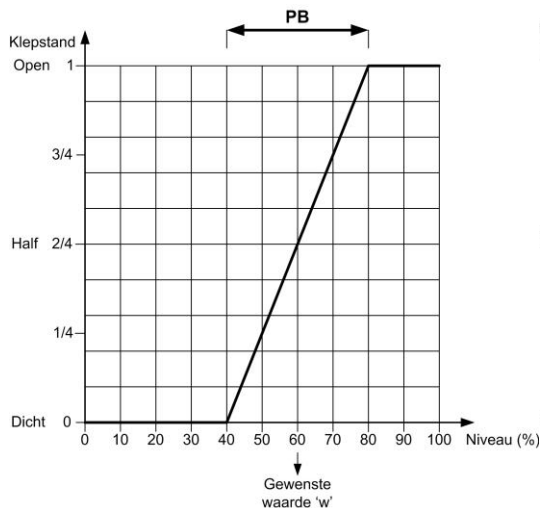


Fig. 7.11 kenlijn P-regelaar a

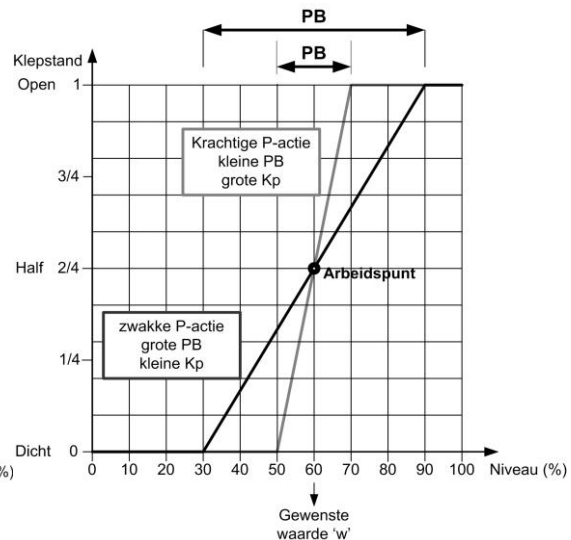


Fig. 7.12 kenlijn P-regelaar b

Op deze kenlijn is duidelijk te zien dat binnen de proportionele band de regelklep geleidelijk overgaat van dicht naar open. Hierbinnen werkt de P-regelaar als continue regelaar. Buiten deze proportionele band gedraagt de P-regelaar zich duidelijk als een Aan/uit regelaar.

- Als we de PB vergroten, dan zal de P-actie verzwakken
- Als we de PB verkleinen, dan zal dat de P-actie krachtiger maken.

Het instelpunt ligt hier voor een klepstand van 2/4 op de gewenste niveauwaarde van 60%. Dit is een goede instelling.

Op een P-regelaar staat één knop om de P-actie krachtiger of zwakker te maken. Deze knop kan twee verschillende opschriften hebben: Kp of PB

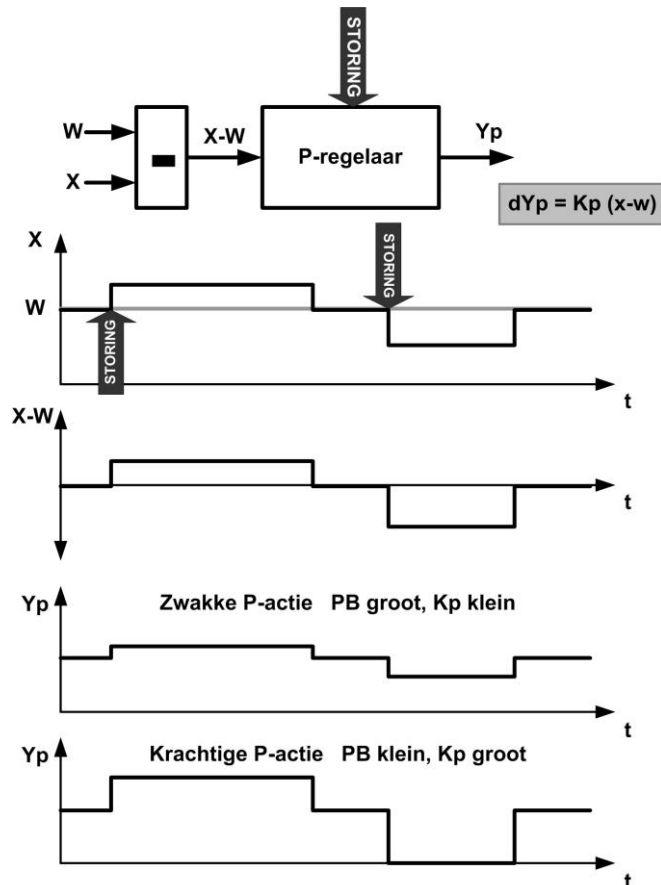
- PB staat voor proportionele band
- Kp staat voor versterkingsfactor van de P-regelaar

Het verband tussen Kp en PB is  **$PB = 1/Kp$**

Een PB van 40% resulteert dus in een Kp van 2,5

- Krachtige P-actie = kleine PB = grote Kp
- Zwakke P-actie = grote PB = kleine Kp

Als we de P-regelaar blokschematisch voorstellen, dan bekommen we onderstaande tekening



We leggen 2 storingen aan de ingang van de P-regelaar en meten aan de uitgang van de P-regelaar hoe deze daarop reageert. We nemen hier ook weer een stapantwoord op, zoals we ook reeds gezien hebben bij de processen.

Deze storingen resulteren in een fout 'x-w'. De P-regelaar reageert altijd proportioneel of 'in verhouding' met deze fout 'x-w'. De reactie van de P-regelaar heeft altijd dezelfde vorm als de fout 'x-w', maar niet noodzakelijk dezelfde grootte en teken.

Een koelkast moet namelijk opspringen als de temperatuur te hoog wordt, terwijl een verwarming dan juist moet uitspringen. Dit noemen we tegengesteld werkende regelaars.

Fig. 7.13 stapantwoord van P-regelaar

De P-regelaar wiskundig  **$dYp = Kp (x-w)$**

De formule van de P-regelaar ' $Yp = Kp (x-w)$ ' is heel gemakkelijk te verklaren:

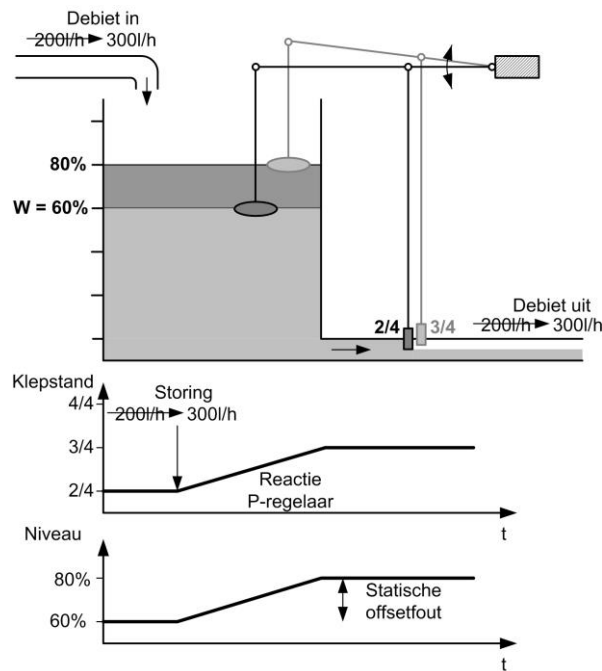
- (x-w) is de fout, veroorzaakt door een storing of verandering van SP.
- Kp is de versterkingsfactor van de P-regelaar, en mag dus vervangen worden door 1/PB
- Yp is de uitgangswaarde van de P-regelaar

De uitgangswaarde varieert dus evenredig of proportioneel met de fout (x-w) en met een kracht gelijk aan Kp.

Statische offsetfout

Het nadeel van de P-regelaar is de statische offsetfout. Bij belastingsveranderingen (bijvoorbeeld wanneer de instroom in een vat plots vergroot van 200l/h naar 300l/h) stabiliseert de P-regelaar de proceswaarde niet terug op de gewenste waarde, maar een stukje erboven.





Een plotse belastingsverandering gaat het niveau in het vat doen stijgen tot er terug evenveel water uit het vat stroomt, als er in het vat stroomt.

Het niveau is in deze tijd gestegen van 60% naar 80% en zal daar nu stabiel blijven.

Het verschil tussen dit nieuwe niveau en de gewenste waarde noemen we de statische offsetfout.

Fig. 7.14 Statische offsetfout

Om deze fout terug weg te krijgen zouden we bijvoorbeeld iemand de klep even 100% open kunnen laten zetten, tot het niveau terug de gewenste waarde bereikt heeft.

De actie die deze persoon doet is eigenlijk een I-actie. Deze actie zou perfect vervangen kunnen worden door een I-regelaar. Een I-regelaar regelt namelijk zo lang bij, tot de fout volledig weg is. Een PI-regelaar wordt daarom ook een proportionele regelaar met herstelactie genoemd.

Besluiten:

- Een P-regelaar is een krachtige regelaar, maar heeft een blijvende statische offsetfout.
- Deze statische offsetfout kan vermindert worden door PB klein te maken.
- Oppassen, een te kleine PB kan tot oscillaties leiden.
- Krachtige P-actie = Grote  $K_p$  = Kleine PB
- Zwakke P-actie = kleine  $K_p$  = Grote PB
- Willen we bij veranderende belastingen, toch een stabiel proces krijgen zonder statische offsetfout, dan moeten we overstappen naar een PI-regelaar.
- Als het arbeidspunt gelijk is aan de gewenste waarde, en er zijn geen belastingsveranderingen, dan zal er ook geen offsetfout zijn.

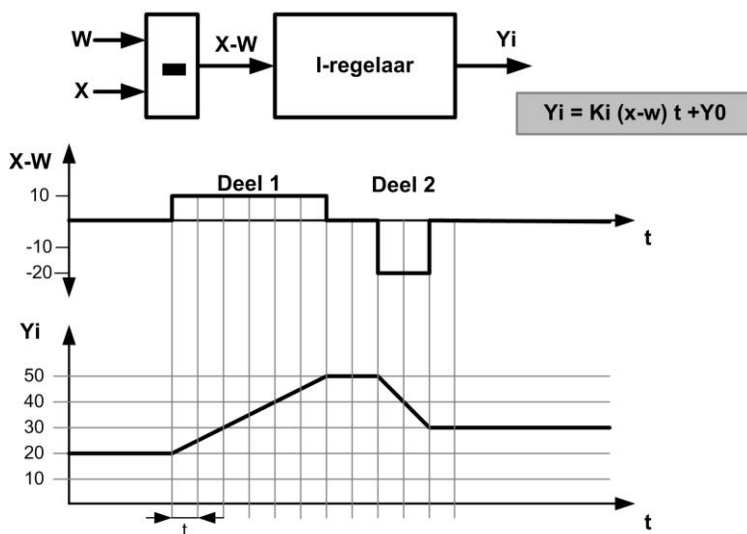
## 7.5 De I-regelaar in open kring

Definitie Integrerende regelaar:

**De I-regelaar regelt afhankelijk van de grootte en de duur van de fout.**

Om de I-regelaar te kunnen begrijpen moeten we eerst weten wat 'integreren' is.

'Integreren' of 'een integraal nemen' is eigenlijk een oppervlakteberekening doen op een bepaald signaal. We zouden dit met complexe wiskunde kunnen benaderen, maar waarom zouden we moeilijk doen als het makkelijk ook kan.



Als we de eenvoudige formule van de I-regelaar toe gaan passen op de fout  $x-w$ , dan bekomen we de getekende  $Y_i$ -waarde als reactie van de I-regelaar op deze fout.

Fig. 7.15 Wat is integreren

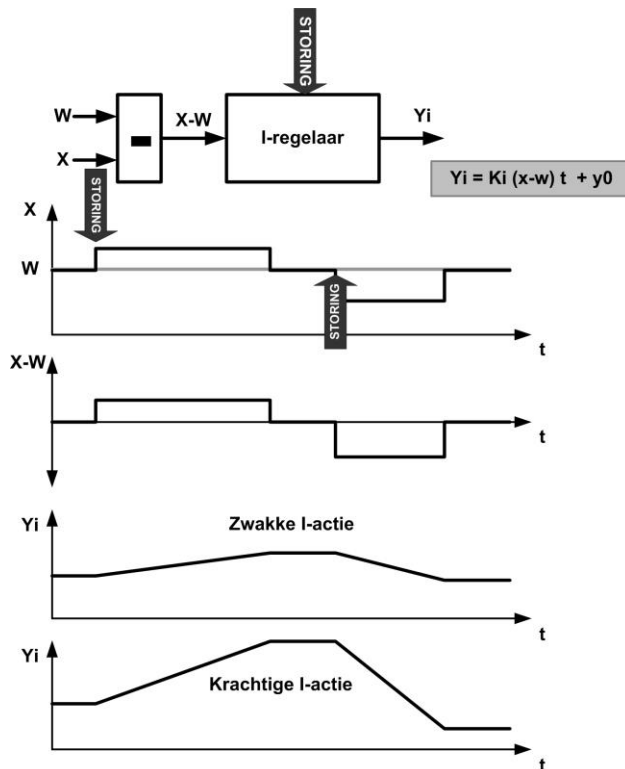
Formule:

$$Y_i = K_i (x-w)t + Y_0$$

Waarin :  $Y_i$  = uitgangswaarde I-regelaar  
 $K_i$  = instelling van de kracht van de I-regelaar (praktisch instelbaar met knop  $T_i$ ,  $T_n$  of  $r$ )  
 $(x-w)$  = fout, veroorzaakt door proces  
 $t$  = tijd, waarover wordt gemeten  
 $Y_0$  = vorige  $Y_i$  waarde van de I-regelaar

Eenvoudige berekening: (We nemen  $t=1$  en  $K_i=0,5$ )

1. $Y_i = 0,5 (0)x1+20$		=	20
2. $Y_i = 0,5 (10)x1+20$	Deel 1	=	25
3. $Y_i = 0,5 (10)x1+25$		=	30
4. $Y_i = 0,5 (10)x1+30$		=	35
5. $Y_i = 0,5 (10)x1+35$		=	40
6. $Y_i = 0,5 (10)x1+40$		=	45
7. $Y_i = 0,5 (10)x1+45$		=	50
8. $Y_i = 0,5 (0)x1+50$		=	50
9. $Y_i = 0,5 (0)x1+50$		=	50
10. $Y_i = 0,5 (-20)x1+50$	Deel 2	=	40
11. $Y_i = 0,5 (-20)x1+40$		=	30
12. $Y_i = 0,5 (0)x1+30$		=	30
13. $Y_i = 0,5 (0)x1+30$		=	30



Stapantwoord van een I-regelaar in open kring

U ziet op de tekening hoe een I-regelaar reageert op een fout (x-w).

De tekening maakt ook duidelijk wat het verschil is tussen een zwakke en een krachtige I-actie.

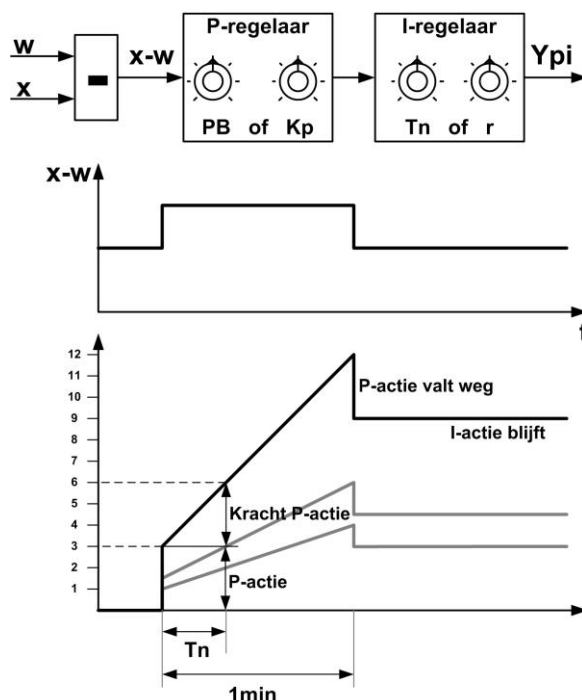
Zoals je kan zien stopt de I-regelaar pas met bijregelen als er geen fout (x-w) meer is.

Fig. 7.16 I-regelaar in open kring.

## 7.6 De PI-regelaar in open kring

Vermits de I-regelaar op zichzelf veel te traag is, en dus niet binnen een aanvaardbare tijd het proces terug stabiel kan krijgen, is een I-regelaar altijd vergezeld van een P-regelaar. De P-regelaar geeft een sterke tegenreactie op een fout terwijl de I-regelaar stilaan de overblijvende fout volledig weg regelt. Van een offsetfout is hier nu geen sprake meer, vermits de I-regelaar blijft regelen tot de gemeten waarde 'x' terug gelijk is aan de gewenste waarde 'w'.

### 7.6.1 Stapantwoord van een seriële PI-regelaar



De uitgangswaarde van de P-regelaar 'Yp' is hier meteen de ingangswaarde van de I-regelaar. Dit noemen we dus de seriële opstelling.

De P-regelaar heeft 2 mogelijke regelknoppen: PB of Kp

De I-regelaar in seriële opstelling heeft ook 2 mogelijke regelknoppen: Tn of r

Tn : nasteltijd  
r : reset-rate

Fig. 7.17 Stapantwoord seriële PI-regelaar.

De seriële opstelling verduidelijkt ook dat wanneer er aan de P-actie iets verandert, de I-actie ook mee verandert, in tegenstelling tot bij de parallelle opstelling.

#### **Tn: nasteltijd**

**De tijd die de I-actie nodig heeft om de kracht van de P-actie te evenaren.**

#### **r : reset-rate**

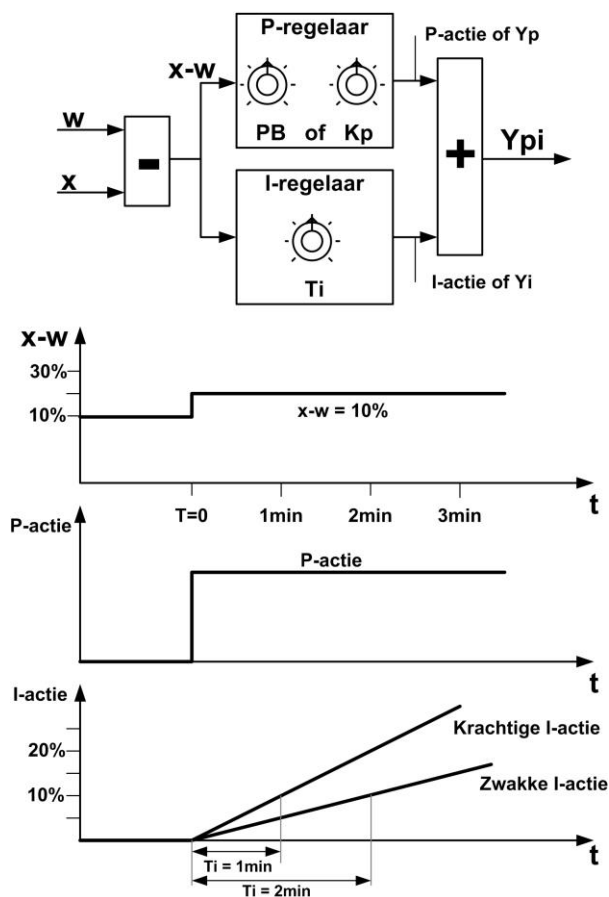
**Het aantal keer per minuut dat de I-actie de kracht van de P-actie kan evenaren**

Op elke seriële I-regelaar komt altijd slechts 1 van deze knoppen voor. Ofwel Tn, ofwel r.

### 7.6.2 Stapantwoord van een parallelle PI-regelaar

#### **Ti: integratietijd**

**De tijd die de I-regelaar nodig heeft om de grootte van de fout (x-w) te evenaren.**



De regelknop op de I-regelaar noemt bij de parallelle opstelling  $T_i$ , van integratietijd.

De parallelle PI-regelaar is intern iets anders opgebouwd. Hierbij is de I-regelaar wel onafhankelijk van de instelling van de P-regelaar.

Aan de buitenkant kan je dit enkel zien aan de aanduiding  $T_i$  (integratietijd) op de instelknop van de I-regelaar, in plaats van  $T_n$  of  $r$ .

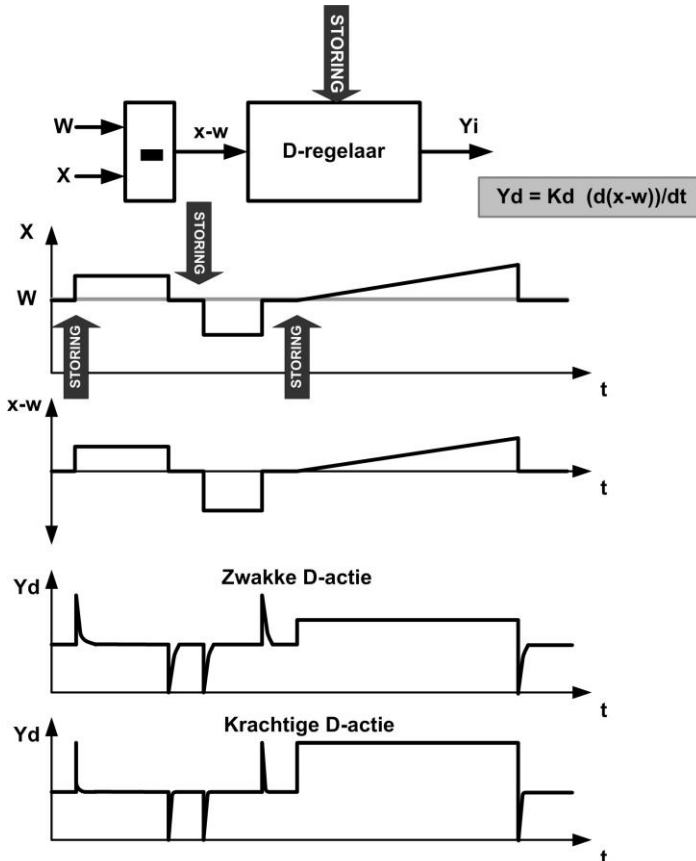
Een verandering van  $K_p$  of  $PB$  zal hier dus geen invloed hebben op de I-actie.

Fig. 7.18 Stapantwoord parallelle PI-regelaar.

## 7.7 De D-regelaar in open kring

Definitie van de D-regelaar:

**De D-regelaar regelt afhankelijk van de snelheid waarmee de fout zich voordoet.**



Als we een stapantwoord van een D-regelaar opnemen, dan krijgen we dit resultaat.

Fouten die **stapvormig** veranderen, veranderen oneindig snel. Vermits de D-regelaar enkel reageert op de snelheid van de fout, zal de D-regelaar tijdens de stijgende en dalende flanken een oneindig grote tegenactie geven aan zijn uitgang (praktisch zo groot als kan). Een fout die stabiel blijft, heeft geen snelheid, en veroorzaakt dus ook geen D-actie.

Fig. 7.19 stapantwoord D-regelaar

Een fout die **lineair** stijgt, zoals ook te zien is, heeft een stabiele snelheid over heel dit stuk. De D-actie zal dus ook over heel dit stuk constant zijn.

De D-regelaar wiskundig.

Ook hier zouden we enorme complexe wiskundige berekeningen kunnen gebruiken om iets uit te leggen dat ook heel eenvoudig kan.

Formule:  **$Y_d = K_d \times d(x-w)/dt$**

Waarin :	$Y_d$	= uitgangswaarde D-regelaar
	$K_d$	= instelling van de kracht van de D-regelaar (praktisch instelbaar met knop $T_v$ of $T_d$ )
	$d(x-w)$	= fout, veroorzaakt door proces in de tijd 'td'
	$dt$	= tijd, waarover wordt gemeten

We lezen  $d(x-w)/dt$  als fout per tijdseenheid.

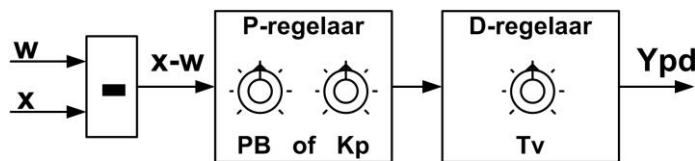
Dit verklaart ook waarom de D-actie reageert op de snelheid van de fout. Snelheid is immers afstand  $d(x-w)$  per tijdseenheid ( $dt$ )

## 7.8 De PD regelaar in open kring

Bij de D-regelaar is het heel duidelijk dat deze regelaar niet alleen een kring kan stabiliseren. Hier moet minimaal een P-regelaar bij gezet worden om de krachtige acties voor zich te nemen.

Ook hier wordt er een onderscheid gemaakt tussen de seriële en de parallelle opstelling van de D en de P regelaar. Het verschil is ook weer enkel te zien aan de benaming die bij de regelknop op de D-actie staat.

### 7.8.1 Stapantwoord van een seriële PD-regelaar



We nemen een stapantwoord op van een PD-regelaar in een seriële opstelling. De regelknop van de D-actie in deze opstelling noemt nu  $T_v$ .  $T_v$  staat voor voorsteltijd, of voorinsteltijd.

In de  $Y_{pd}$  grafiek zie je mooi de optelling van de D en de P-actie.

De D-actie is de snelste regelactie en is sneller dan de P-actie.

**$T_v$ : De tijd die de P-actie nodig heeft om de kracht van de D-actie te kunnen evenaren.**

Let wel op: Bij de seriële PI regelaar was dit duidelijker, maar ook hier heeft een verandering van de instelling van de kracht van de P-regelaar wel een invloed op de D-actie, maar de kracht van de D-actie heeft nooit enige invloed op de P-actie.

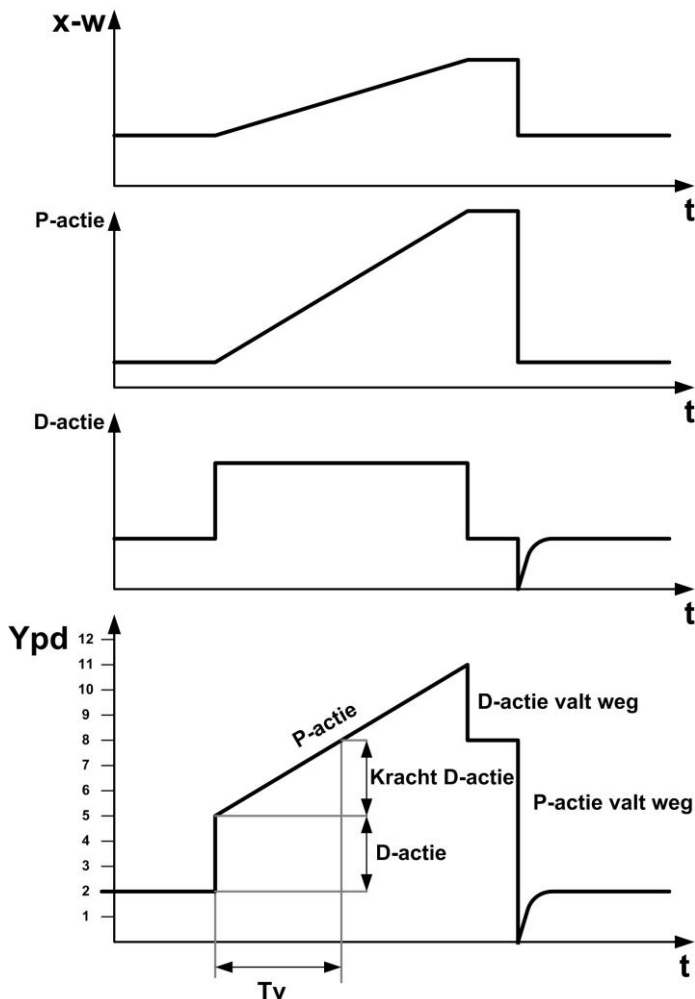
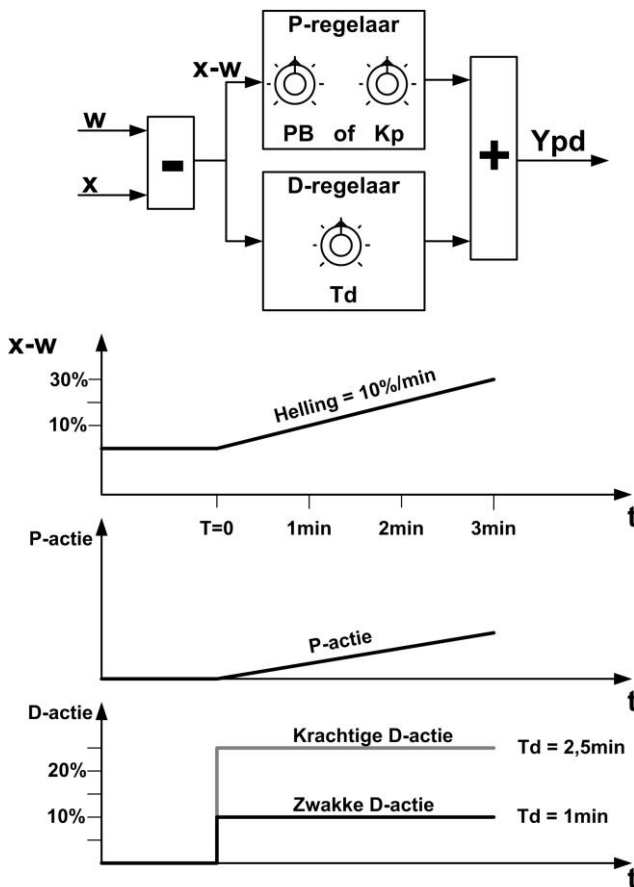


Fig. 7.20 stapantwoord seriële PD-regelaar

## 7.8.2 stapantwoord van een parallelle PD-regelaar

Bij de parallelle opstelling van de PD regelaar zijn P en D actie wel volledig onafhankelijk van elkaar.



De parallelle opstelling van de PD-regelaar is enkel te herkennen aan het opschrift dat bij de D-regelaar staat. Hier staat nu  $T_d$  (differentiatietijd) in plaats van  $T_v$ .

**$T_d$  : differentiatietijd : Bij een constant toenemende afwijking zal, na een tijd  $T_d$ , de fout ( $x-w$ ) even sterk zijn als de D-actie.**

$$\text{D-actie} = (10\%/ \text{min}) \times T_d$$

Fig. 7.21 Stapantwoord parallelle PD-regelaar

### Besluiten D-regelaar

- De D-actie wordt dikwijls gebruikt om de oscillaties, veroorzaakt door de I-regelaar te dempen.
- De PD regelaar heeft een statische offsetfout, enkel een I-regelaar kan deze statische fout weggeregelen.
- Te sterke D-actie veroorzaakt oscillaties met een kleine periode, terwijl een I-actie oscillaties veroorzaakt met een grote periode. (grote periode = lage frequentie)
- Ruis op het systeem heeft gewoonlijk een kleine amplitude, maar een hoge frequentie. Hoge frequentie betekent hoge snelheid en vermits de D-regelaar enkel reageert op snelheid, gaat de D-regelaar hier enorm hard op reageren. Daarom moeten we oppassen met de toepassing van een D-actie als er ruis op het systeem zit.
- Sterke D-actie = grote  $T_v$  = grote  $T_d$
- Zwakke D-actie = kleine  $T_v$  = kleine  $T_d$

## 7.9 PID regelaar

De PID regelaar is een combinatie van de P-, de I- en de D-regelaar. Meestal zitten deze 3 regelacties samen in één regelaar, maar worden ze niet alle 3 gebruikt.

De oefening hieronder zal je sterk vooruithelpen bij het leren van de typische reacties van de verschillende regelaars in open kring.

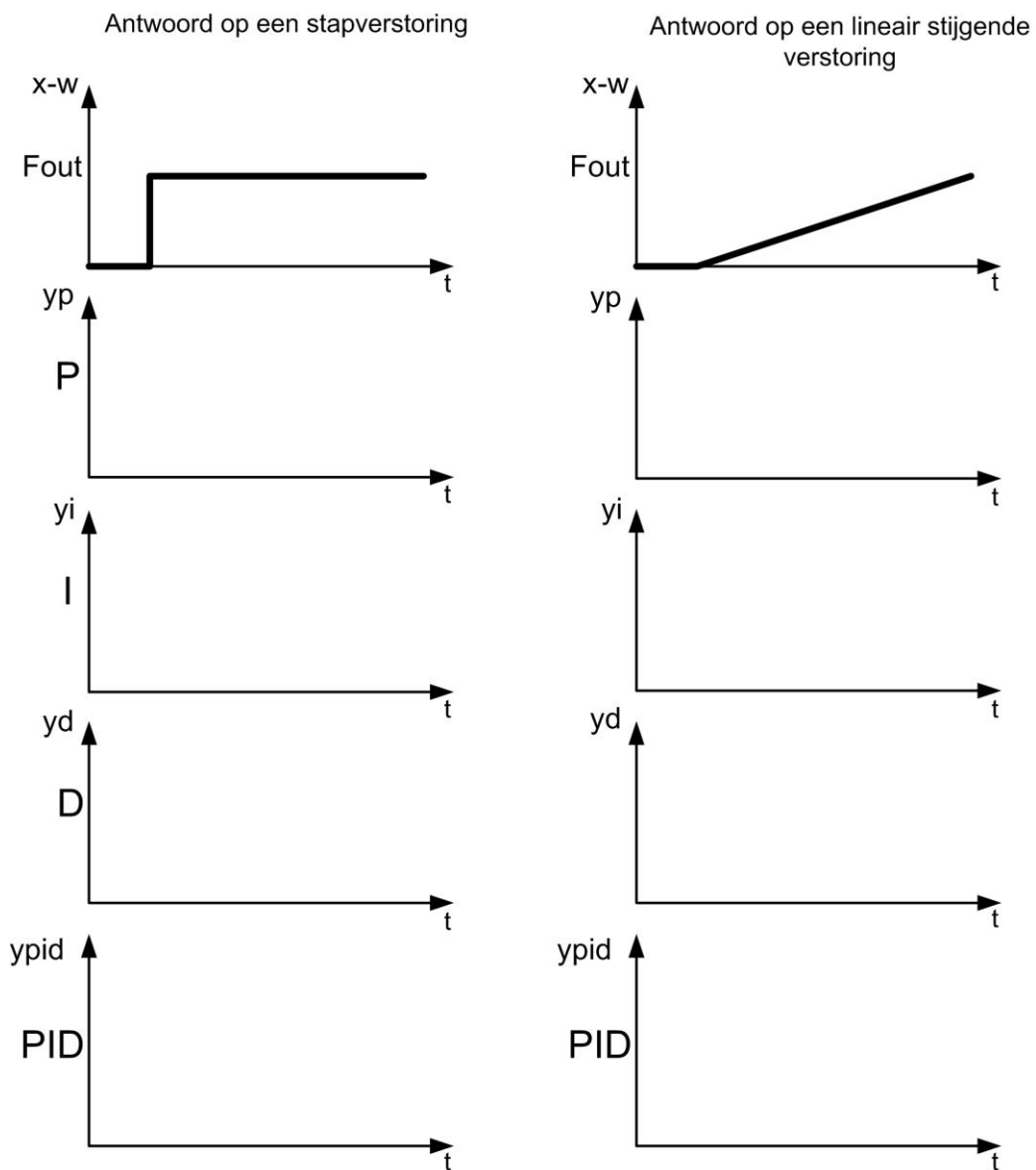


Fig. 7.22 oefening PID acties

Het hangt volledig af van de procesparameters ( $T$  en  $T_d$  en ORDE) of het nodig is een P-, een PI-, een PD- of een PID- regelaar in te schakelen. Ook de instelling van de verschillende regelaars afzonderlijk ( $P_b$  of  $K_p$ )( $T_n$ ,  $r$  of  $T_i$ )( $T_v$  of  $T_d$ ) hangt af van deze procesparameters. Hoe we deze informatie uit deze procesparameters halen, zien we in een volgend hoofdstuk.

Het is logisch dat een PID regelaar afregelen een flink stuk complexer is als het afregelen van een P-regelaar.



## 7.10 De Fuzzy-logic regelaar

De Fuzzy-logic regelaar is de tegenhanger van de klassieke PID regelaar. Verschillende fabrikanten van regelaars steken beide technologieën in hun regelaars en geven zo hun klanten de keuze tussen fuzzy en PID.

PID regelaars in combinatie met complexe processen vereisen nogal wat kennis van de mensen die deze regelaars moeten afstellen. Dikwijls wordt er hiervoor dan teruggegrepen naar hogere wiskunde en ingewikkelde stabiliteitscriteria. De Fuzzy logic theorie is, zoals u zal zien, veel eenvoudiger te begrijpen.

Beide technologieën, zowel PID als Fuzzy, hebben hun voordelen en de situatie zal uitwijzen welke technologie de beste is.

De theorie van de 'fuzzy logic' of vrij vertaald de 'vage logica' is pas in 1965 gepubliceerd door professor 'Lofti A Zadeh'. Alle mensen zijn eigenlijk Fuzzy regelaars. Dit wordt ook wel eens de ervaringsgerichte regelaar genoemd.

Voorbeeld:

Je wil een bad nemen en laat het bad alvast vol lopen.

Na enkele minuten ga je terug naar het bad en je neemt 2 zaken waar:

- Je voelt even met je vinger hoe warm het water is.
- Je kijkt hoe veel water er al in het bad staat

De temperatuur druk je voor jezelf niet uit in °C, maar in een aantal uitdrukkingen: veel te koud, te koud, goed, te warm of veel te warm.

Het niveau druk je voor jezelf natuurlijk ook niet uit in aantal cm, maar ook weer in een aantal uitdrukking: veel te weinig, te weinig, bijna goed, goed, te veel.

Hoogstwaarschijnlijk besef je dit niet, maar je hebt voor al deze verschillende combinaties een hoop regels in je hoofd zitten hoe je hierop zou moeten reageren.

VB1 T=veel te warm N=bijna goed  
Reactie= enkel nog koud water toevoegen

VB2 T=veel te warm N=te weinig  
Reactie= kraan warm water iets meer dicht zetten.

De mens is dus een Fuzzy regelaar: Je hebt voor jezelf een aantal 'vage' gebieden afgebakend waarin je je waarnemingen ordent en die bepalen aan de hand van jou ervaringen hoe je reageert op deze waarnemingen.

Om de regelactie, die hier door een mens gedaan wordt, te automatiseren, moeten we de ervaringen van deze mens in een elektronische regelaar programmeren. Vermits deze vage begrippen als te koud, of veel te warm voor een computer onbegrijpelijk zijn, is de fuzzy logic ontwikkeld. Deze

fuzzy logic is een systeem om de vage, ervaringsgerichte begrippen te vertalen naar computerbegrippen zodat de computer kan begrijpen wat te koud en wat veel te koud is.

We definiëren, aan de hand van onze ervaringen, gebieden die voor ons voldoen aan onze uitdrukkingen als 'veel te koud' of 'goed'. We gaan deze op een temperatuur-as uitzetten. Deze gebieden overlappen elkaar omdat wij zelf niet zo'n nauwkeurige meetinstrumenten hebben, en omdat deze gebieden voor onszelf ook niet zo heel nauwkeurig zijn afgebakend.

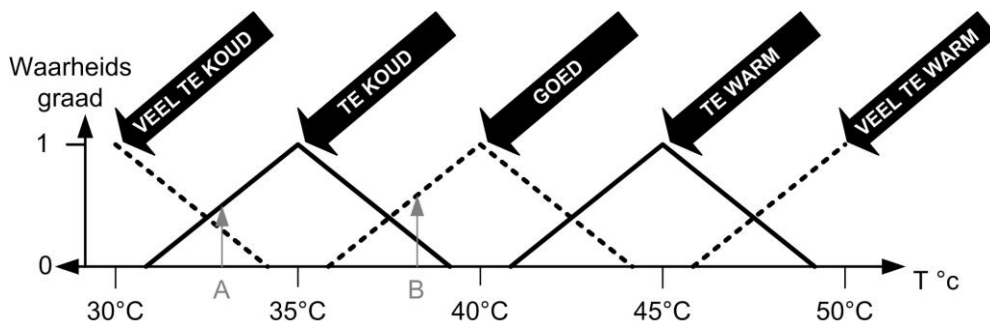


Fig. 7.23 waarheidskrommen temperatuur

Je leest deze grafiek als volgt:

In punt A heeft de uitspraak 'te koud' een hogere waarheidgraad als de uitspraak 'veel te koud'. Bij punt A hoort dus de uitspraak 'te koud'.

Dit is een bewerking die een computer wel kan doen. Via deze grafiek vertalen we onze vage ervaringen naar exacte computer-gegevens.

Ook voor het niveau kunnen we deze waarheidsgrafiek opstellen.

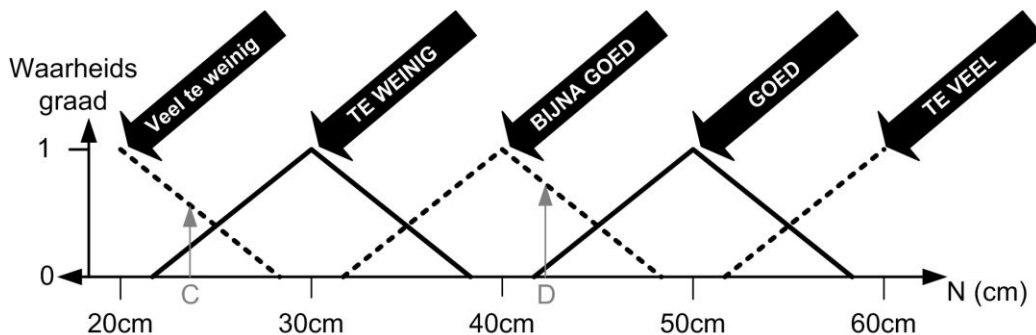


Fig. 7.24 waarheidskrommen niveau

Het niveau dat hier bij punt D hoort, en dat door een niveaumeting exact te bepalen is, is in onze ervaring een 'bijna goed' niveau.

Deze verschillende driehoeken noemt men lidmaatschapskrommen.

Van het moment de computer heeft uitgemaakt tot welke lidmaatschapskrommen de ingangsvariabelen (temperatuur en niveau) behoren (dit noemt men fuzzyficiering), gaat de computer in de regelaar kijken wat hij moet doen als deze combinatie zich voordoet.

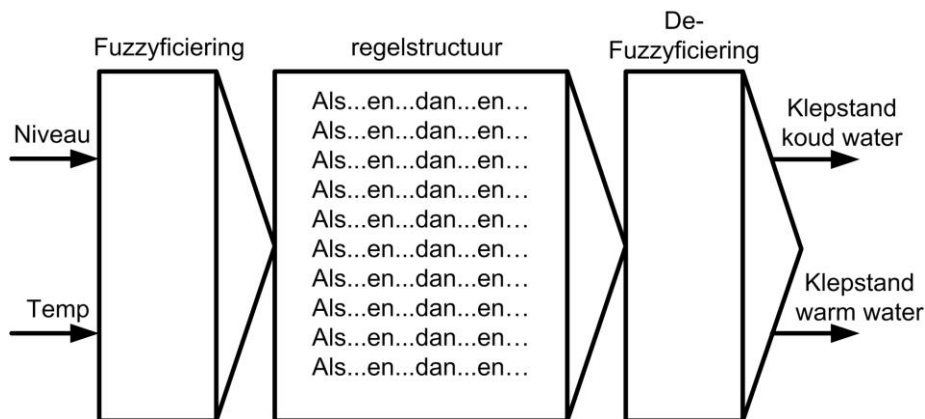


Fig. 7.25 Blokschema Fuzzy-regelaar

Via de regelstructuur programmeer je de Fuzzy logic regelaar. Jij bepaalt in het programma wat de regelaar exact moet doen als die bepaalde combinatie van ingangsvariabelen zich voordoet. Voor elke mogelijke combinatie moet je een bepaald actie programmeren.

Aan de hand van de geprogrammeerde actie zal de regelaar de uitgangsvariabelen bijsturen. Ook dit gebeurt aan de hand van een lidmaatschapskromme. Dit noemt men de-fuzzyficiering. In dit geval zijn er 2 krommen, één voor de klepstand van het koude water en één voor de klepstand van het warme water.

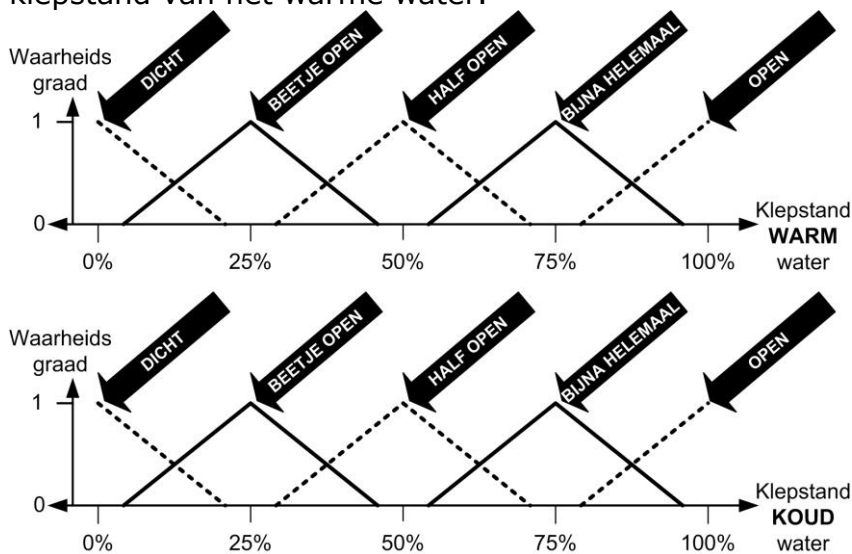


Fig. 7.26 klepstand warm en klepstand koud

De westerse wereld stond en staat nog steeds sceptisch tegenover Fuzzy logic. In het westen vertrouwen ze liever op complexe wiskundige criteria terwijl men in het oosten en dan vooral in Japan wel mee op de boot van de Fuzzy-logic gesprongen is.

De Fuzzy-logic regelaar lijkt veel eenvoudiger in te stellen als de PID regelaar, toch zijn er ook nog vele situaties waar de PID regelaar een betere keuze is. In de procesindustrie (petrochemie) worden Fuzzy regelaars nagenoeg niet toegepast. De PID regelaars draaien al veel langer mee en hebben hun sporen in deze tak van industrie al verdient. In de verwarmings- koel- en klimatisatie installaties kom je wel Fuzzy regelaars tegen.

## 7.11 Oefeningen

1. Omschrijf in je eigen woorden wat jij verstaat onder het nut van een regelaar in een regelkring.

---

---

---

---

2. Noem de 5 eisen die gesteld worden aan een regelaar.

---

---

---

---

---

3. Noem de 3 situaties waarin de regelaar moet ingrijpen op de regelwaarde 'y'.

---

---

---

4. Wat versta jij onder een oscillerend systeem? Verduidelijk d.m.v. een tekening.

---

---

---

5. Noem 2 grote nadelen van een oscillerend systeem.

---

---

---

6. Benoem de 3 grote blokken van verschillende soorten regelaars.

---

---

---

7. Teken hieronder een parallelle PID regelaar en een seriële PID regelaar.  
Waaraan kan je langs buiten zien met wat voor regelaar je te maken hebt.

---

---

---

8. Noem 3 voordelen van een aan/uit regelaar.

---

---

---

9. Voor welk type regelaar zou je kiezen als je een proces met een hele kleine capaciteit hebt.

---

---

10. Voor welk type regelaar zou jij kiezen als je een traag proces hebt, echter met een vrij onstabiele proceswaarde.

---

---

11. Wat is het voordeel van een aan/uit regelaar met hysteresis?

---

---

12. Geef hieronder de definities van de P, de I en de D regelaar.

---

---

---

---

---

---

---

---

---

---

13. Geef hieronder de formules voor de P, de I en de D regelaar.

---

---

---

14. Wat versta jij onder een zelfwerkende regelaar?

---

---

15. Invuloefening:

De P-regelaar regelt maar over een beperkt meetgebied, dit gebied noemen we de ..... . Buiten dit gebied gedraagt de P-regelaar zich als ..... . Om de P-actie te verzwakken, moeten we de PB ..... . Om de P-actie te verzwakken moeten we de  $K_p$  ..... . Om de P-actie te versterken moeten we de PB ..... . Om de P-actie te versterken moeten we de  $K_p$  ..... . De reactie van de P-regelaar heeft altijd dezelfde ..... als de fout 'x-w', maar niet noodzakelijk dezelfde ..... Een te ..... PB kan tot oscillaties leiden. De statische offsetfout kan enkel worden weggewerkt door het toevoegen van een .....-actie aan de P-actie. De .....-regelaar blijft regelen zolang er een fout is. Als  $(x-w) = 0$ , dan zal de I-actie ..... (terugvallen naar 0 / de laatste waarde blijven aanhouden). Omdat een I-regelaar te ..... is moet deze steeds vergezeld zijn van een .....-regelaar. Er kan bij de P-regelaar geen statische offsetfout ontstaan als er ..... zijn en het ..... goed gekozen is. Een seriële PI-regelaar herken je aan de .....-knop of de .....-knop, een parallelle PI-regelaar herken je aan de .....-knop. Een seriële PD-regelaar herken je aan de .....-knop, een parallelle PD-regelaar herken je aan de .....-knop. Grote  $T_v = \dots\dots\dots$   $T_d = \dots\dots\dots$  D-actie. kleine  $T_d = \dots\dots\dots$   $T_v = \dots\dots\dots$  D-actie.

16. Wat versta jij onder een statische offsetfout?

---

---

---

17. Noteer mooi onder elkaar de definities van  $T_n$ ,  $r$  en  $T_i$ . Verduidelijk dit d.m.v 2 tekeningen.

---

---

---

---

---

---

---

---

---

---

18. Hoe verklaar jij dat de reactie van een D-regelaar op een stapsprong zo piekvormig is? Kan je dit ook bewijzen met behulp van de formule van de D-regelaar?

---

---

---

---

---

---

---

---

---

19. Waarom gebruiken we, als we de D-regelaar willen uitleggen, liever een lineair stijgende functie in plaats van een stapsprong? Verduidelijk met een tekening.

---

---

---

---

20. Noteer mooi onder elkaar de definities van  $T_v$  en  $T_d$ . Verduidelijk dit d.m.v. 2 tekeningen.

---

---

---

---

---

---

---

---

---

21. Waarom spreken ze bij een PI-regelaar over nasteltijd en bij een PD-regelaar over voorsteltijd? Verduidelijk ook met een tekening.  
Tekening:

---

---

---

---

22. Heeft een PD-regelaar ook een statische offsetfout? (Ja/neen)

23. Waarom moeten we bij de toepassing van een D-actie opletten voor de aanwezigheid van ruis op het systeem?

---

---

---

---

---

---

24. Wat is de letterlijke vertaling van 'Fuzzy'?

---

---

---



25.oefening PID regelaar. Teken de reacties van de respectievelijke regelaars.  
Enkel de vorm is van belang, niet de grootte.

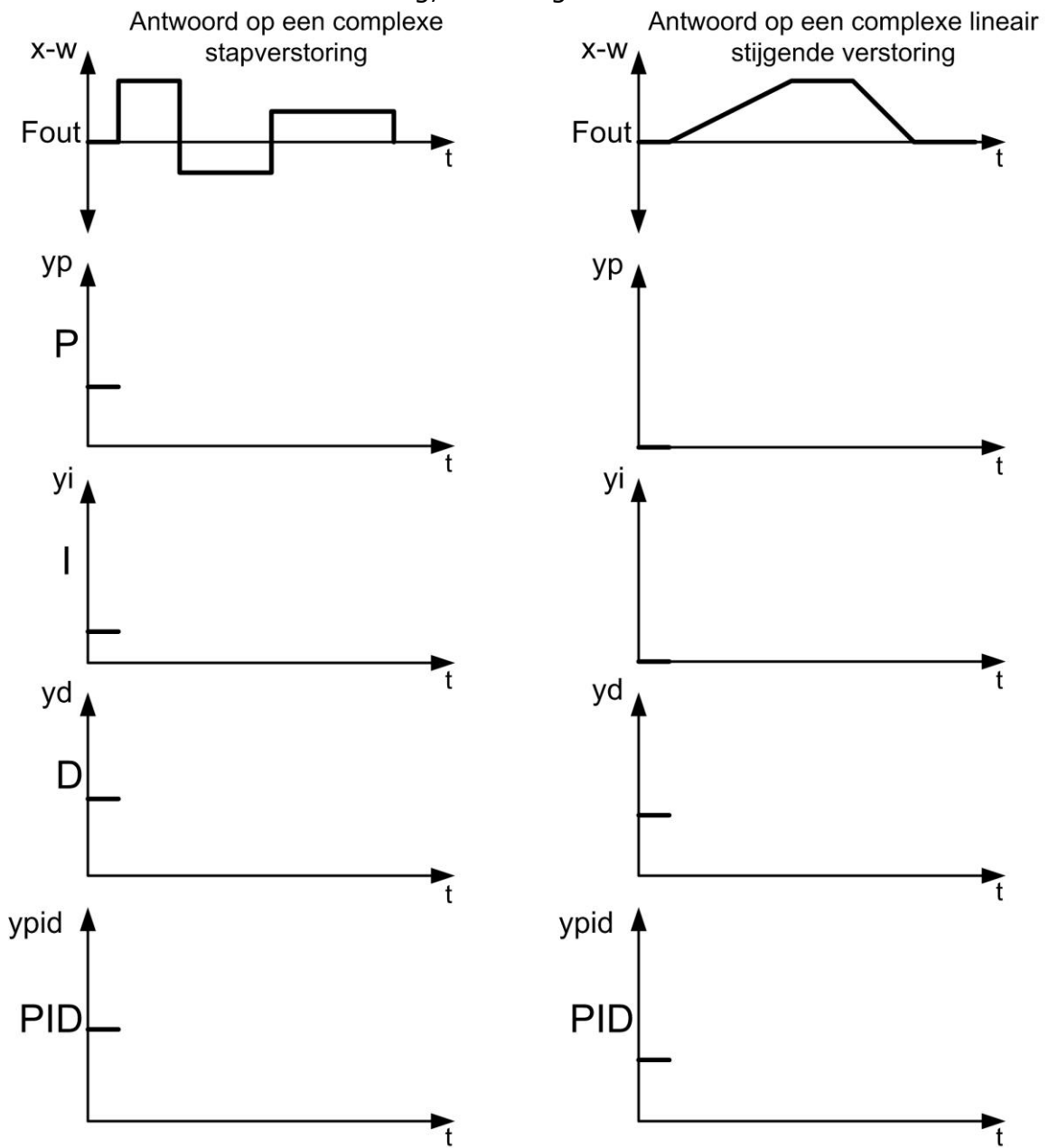


Fig. 7.27 oefening PID regelacties

26. Wat is de letterlijke vertaling van 'Fuzzy'?

---



---

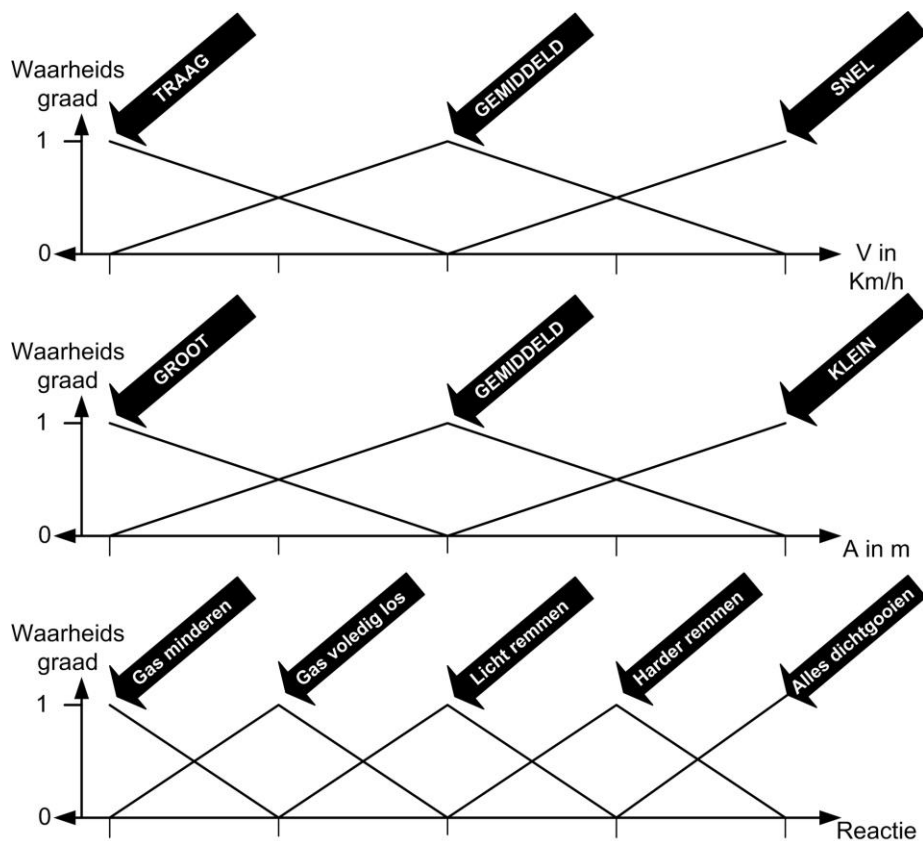


---

Diagram illustrating the variables for the problem:

- $v$ : Snelheid 'v' in km/h (Speed of truck A)
- $A$ : Afstand tussen vrachtwagens 'A' in (m) (Distance between trucks A and B)

De lidmaatschapskrommen van zowel in als uitgang zijn reeds opgesteld. Je moet hier nog wel exacte getallen bijvoegen zodat de computer weet binnen welke grenzen hij een gepaste actie moet uitvoeren.



Stel nu zelf alle regels op zoals jij denkt dat het goed is.

[illegible]