

SELVOPIMALISERENDE HIERARKISKE SYSTEMER – PROSESSREGULERING, NASJONALØKONOMI, HJERNE OG MARATONLØPING

foredrag på møte
19. januar 2017

av professor Sigurd Skogestad, Institutt for kjemisk prosessteknologi,
NTNU, Trondheim

Denne presentasjonen fokuserer på hvordan vi skal implementere beslutninger i en usikker verden. Hvordan får vi systemet til å oppføre seg som ønsket på tross av ukjente forstyrrelser? Hvordan kombinerer vi optimalitet med robusthet, enkelhet og rask respons?

Dette har vært et viktig forskningsområde for meg de siste 30 år, med utgangspunkt i hvordan man opererer og regulerer store prosessanlegg, som oljeraffinerier. Store prosessanlegg er sannsynligvis de systemer som har kommet lengst når det gjelder sentralisert beslutningstagning, fordi all informasjon fra anlegget samles i ett kontrollrom, og alle beslutninger tas der. Likevel finner man i praksis at beslutningssystemet er hierarkisk og desentralisert.

I teorien er et sentralisert beslutningssystem optimalt, men i praksis gir det lang responstid og ofte feil beslutninger. Et eksempel er en sentralisert kommunistisk sentraløkonomi, som i teorien er best siden man kan koordinere behov og ressurser på en optimal måte. I praksis har historien vist at det ikke virker.

Fokuset i denne teksten er på hierarkiske reguleringsystemer basert på tidsskalaseparasjon. Et viktig valg er hvilke beslutningsvariabler man har mellom lagene i hierarkiet. Det anbefales å søke etter selvoptimaliserende variabler som kan holdes på konstant settpunkt. Ideen illustreres med praktiske eksempler som maratonløping og nasjonaløkonomi.

1. Virkelige systemer

La oss ta utgangspunkt i hvordan virkelige systemer opereres. Det som går igjen, er et hierarkisk beslutningssystem basert på tidsskalaseparasjon og settpunktsstyring (figur 1), eller kommandostyring om vi skal bruke et litt mer belastet begrep. Et eksempel er menneskets beslutningssystem som har mange nivåer, fra den bevisste sentraliserte tenkningen, til reaksjoner i sentralnervesystemet og responsen i enkeltceller. En typisk del av et hierarkisk regulerings-system i dagliglivet kan være temperaturregulering, der oppgaven er å bruke en varmeovn til å holde temperaturen på et gitt settpunkt, for eksempel 23° Celsius. Settpunktet kan kanskje variere over tid, men dette er en beslutning som tas på en langsommere tidsskala lenger opp i beslutningspyramiden.

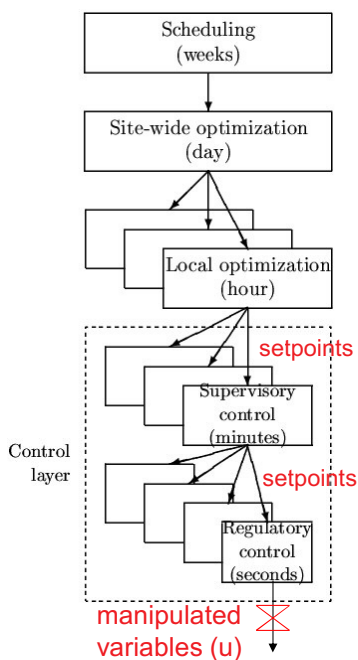
Hovedgrunnen til at et hierarkisk beslutningssystem fungerer så bra, er at hver del av hierarkiet er enkel. Hver del fungerer selvstendig (autonomt), i kombinasjon med en utstrakt bruk av lokal tilbakekobling (*feedback*). Dette gjør at responstiden kan være rask. Hver del (regulator) fungerer autonomt så lenge den greier å holde settpunktet, men hvis det er problemer med dette, for eksempel fordi varmeovnen (input eller pådrag) har nådd sitt maksimum,

så kan det sendes beskjed oppover i hierarkiet slik at en korleksjon kan gjøres, for eksempel ved at settpunktet endres.

En nøkkelbeslutning er *hvilke* variabler man velger å ha settpunkt på, og det er her ideen om selvoptimalisering kommer inn (Skogestad, 2000).

Selvoptimaliserende regulering er en strategi hvor man søker å finne variabler som gir nær-optimal oppførsel når de holdes på konstant settpunkt, på tross av uforutsette forstyrrelser.

Selvoptimaliserende variabler er variabler som når de holdes konstant, gir nær-optimal oppførsel på tross av for-



Figur 1. I praksis benyttes et hierarkisk desentralisert beslutningssystem (pyramide). Her vist for et prosessanlegg.

styrrelser, i alle fall over en kortere tidsperiode. Eksempler kan være hjertefrekvens for optimal maratonløping, eller rente fra nasjonalbanken for optimal nasjonaløkonomi. Det er også utviklet matematisk teori rundt denne ideen (se referanselisten).

Det menneskelige styringssystemet er utviklet gjennom evolusjon, så det er kanskje ikke så rart at det har blitt hierarkisk. Etter hvert som organismen har blitt mer komplisert, har man beholdt det meste i det underliggende systemet, for eksempel en enkeltcelle, og bygget på etter hvert med mer langsomme styringssystemer på høyere nivåer.

Det som er vel så interessant, er at også menneskeskapt beslutningssystemer er hierarkiske og fungerer på en lignende måte. Et typisk beslutningssystem fra et kjemisk prosessanlegg er vist i figur 1. Den fysiske implementeringen av beslutningen (den manipulerede variabelen u) er nederst i pyramiden. I et prosessanlegg er dette typisk en ventil som justeres manuelt av en operatør eller automatisk med en elektrisk motor.

Man kan merke seg at systemet i figur 1 er delt opp i blokker basert på tre prinsipper:

Prinsipp 1. Tidsskalaseparasjon (hierarkisk regulering)

Prinsipp 2. Romlig separasjon (desentralisert regulering)

Prinsipp 3. Utstrakt bruk av lokal tilbakekobling (negative feedback)

Den største fordelen ved tidsskalaseparasjon (prinsipp 1) er at beslutningsnivåene kan operere uavhengig av hverandre. Hver beslutning blir da mye enklere. Typisk bør responstiden i et nivå være minst 5–10 ganger raskere enn nivået over for å oppnå dette. Romlig separasjon (prinsipp 2) foregår på samme tidsskala, og er illustrert i figur 1 ved at det er flere beslutningsblokker på samme hierarkiske nivå, slik at vi får en beslutningspyramide. Også her er fordelen at man tar lokale beslutninger uavhengig av andre beslutninger på samme nivå.

Prinsipp 3 er ikke vist i figur 1, men hver beslutningsblokk i de to laveste nivåene (*control layer*) forutsettes å ha lokal tilbakekobling hvor man sammenligner settpunktverdien (som er vist i figur 1) med den målte verdien (som ikke er vist i figur 1), og justerer beslutningen (pilen ut av hver blokk i figur 1) inntil settpunkt og måling har samme verdi. For eksempel, hvis settpunktet er å holde temperaturen på 23° Celsius (settpunkt), så vil regulatorblokken (*controller*) justere varmemengden inntil målt temperatur faktisk er lik 23° Celsius.

2. Teori: Sentralisert beslutningssystem

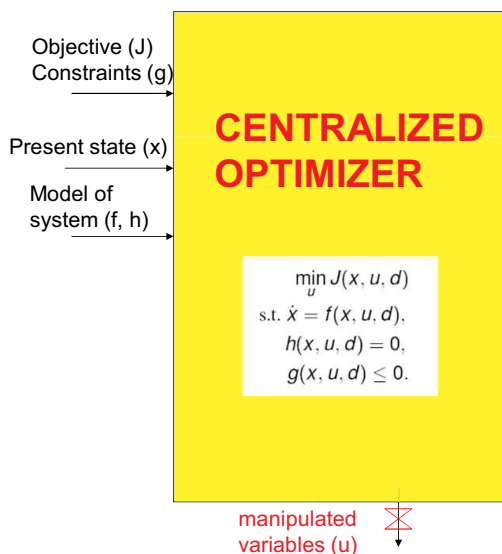
Separasjonen i blokker (romlig separasjon) i figur 1 virker logisk og den er enkel, men rent teoretisk (eller matematisk om man vil) er et sentralisert beslutningssystem det optimale. Dette skyldes at det hierarkiske systemet i figur 1 er et spesialtilfelle av det sentraliserte. Det optimale beslutningssystemet har derfor kun en blokk, se figur 2. Det meste av forskningen innen reguleringsteknikk de siste 50 år har fokusert på det sentraliserte beslutningssystemet. De viktigste elementene i et slikt system er

- Matematisk modell av systemet (for eksempel av et prosessanlegg eller av økonomien i et land) som oppdateres på bakgrunn av målinger (feedback).
- Sentralisert beslutningstagning basert på optimalisering av en definert objektfunksjon, typisk relatert til maksimering av profitt eller minimering av kostnad.

Et sentralisert beslutningssystem er i teorien optimalt, men det er dessverre kostbart å bygge opp og vedlikeholde, og krever ofte mye informasjon som viser seg å være lite relevant for den ”optimale” beslutningen som til slutt tas. Men et vel så stort problem, er at det ofte er langsomt og lite robust.

La oss se på kommunistisk sentral- og planøkonomi, som ofte var basert

på 5-årsplaner. Det er ganske opplagt at hvis det oppstår viktige forstyrrelser som har raskere frekvens enn 5-årsplanen, så vil dette fungere dårlig. En forbedring er en rullerende 5-årsplan som oppdateres hvert år. Dette er ideen bak det som i mitt fagområde kalles ”prediktiv regulering” eller ”receding horizon control”. Man lager da en langtidsplan basert på



Figur 2. I teorien er et sentralisert beslutningssystem det optimale.

optimalisering over en lang horisont, gjerne uendelig, men man implementer bare den første beslutningen i denne langtidsplanen. Langtidsplanen er nemlig løpende, og man starter umiddelbart med å lage en ny langtidsplan basert på oppdatert informasjon, for eksempel, om forbrukerne foretrekker store eller små biler. Dette høres ut som en god idé, og det er det, men det er fremdeles komplisert. Derfor brukes det bare i begrenset grad i praksis, selv i virksomheter som oljeraffinering, som i utgangspunktet egner seg perfekt for denne typen optimalisering.

Det er flere problemer, blant annet at det krever mye regnekraft, men det mest fundamentale er kanskje kostnaden med å samle inn informasjon og ha en oppdatert beskrivelse av systemet, i form av en matematisk modell, som man prøver å optimalisere. Det kan være vanskelig å bygge inn robusthet og unngå at reguleringen overreagerer og begynner å svinge.

3. Hvordan lager man et hierarkisk beslutningssystem?

Hvis man ser på prosessregulering, så har man tre hovednivåer i et hierarkisk beslutningssystem (se figur 1):

Toppnivå optimalisering (langsomt): Sentralisert *optimalisering* basert på økonomiske kriterier, ofte over en ganske lang tidshorisont (timer og mer). Det optimaliseres for å finne hva som er ”aktive begrensninger” (for eksempel hvis man skal varme opp hytta raskt, så er max varme en begrensning) og optimale settpunkter (for eksempel optimal temperatur for ølbrygging). Man antar at det underliggende reguleringsystemet implementer de nye settpunktsverdiene nesten umiddelbart.

Mellomnivå økonomisk regulering: Overordnet (supervisory) *regulering* som holder de økonomiske variablene på det gitte settpunkt ved å endre settpunktet til nivået under.

Laveste nivå stabiliserende regulering (raskt): Basis eller ”regulatory” *regulering* som endrer de fysiske pådragene (typisk ventilposisjoner i et prosessanlegg) som skal følge settpunkter fra mellomnivået, men som for øvrig har som hovedformål å stabilisere systemet og hindre at det drifter av gårde. Et eksempel er å holde nivået i en tank omtrent konstant.

Det kan være flere nivåer, men de tre gitt over har klart forskjellige oppgaver. Merk at i de to nederste reguleringsnivåene er hovedmålet å holde

utvalgte variabler på gitte settpunkter. Men hva er de beste ”utvalgte” variablene? Det er her vi bruker ideen om selvoptimaliserende regulering og tidsskalaseparasjon.

Innen ”economic MPC” som er veldig populært i akademiske miljøer for tiden, prøver man å kombinere alle nivåene i en sentralisert løsning. Men har dette noen hensikt? Neppe, med mindre tidsskalaene for nivåene overlapper.

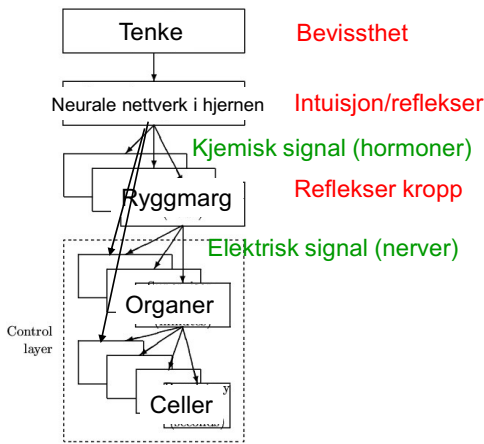
Jeg har utviklet en enkel oppskrift for hvordan man kan utforme et hierarkisk system for prosessregulering (Skogestad, 2004). Hovedtrinnene er:

1. *Problemdefinisjon*. Hva er objektfunksjonen J som skal optimaliseres? Hva er begrensninger i systemet? Hva er uavhengige variabler (pådrag, input)? Hva er viktige forstyrrelser (d) som gjør at vi må justere input underveis?
2. *Optimalisering*. Hvordan opereres systemet på best mulig måte, både nominelt og med forstyrrelser (d), dvs. hva er de optimale verdier for input u under ulike forhold? Dette er i prinsippet enkelt hvis man har en god matematisk beskrivelse / modell av systemet, men er ofte veldig tidkrevende.
3. *Implementering* av optimal drift (reguleringsstrategi). Hvordan kan man i praksis få til det man fant i punkt 2 på en enkel måte? Det viktigste er å velge *hvilke* variabler man skal holde på konstant settpunkt. Regelen er å regulere følgende CV’er (*Controlled variables*):
 1. Aktive begrensninger (som må holdes på sin grenseverdi)
 2. Selv-optimaliserende variabler (for de gjenværende frihetsgradene).

Et meget viktig element i et hierkisk beslutningssystem er at hvert nivå er autonomt over en kortere tidsskala, dvs. at hvert nivå tar beslutninger uten å vente på ny instruks (nytt settpunkt) fra nivået over. Dette gjør at systemet blir enkelt, raskt og robust. Som sagt er det ikke helt optimalt rent matematisk, men det kan være meget nært dersom man velger de rette CV’er og har tilstrekkelig tidsskalaseparasjon mellom nivåene.

Valg av de rette selvoptimaliserende variablene (CV’er) sikrer konsistens mellom nivåene. Ideen er at hvis det kommer en endring (forstyrrelse d), så skal det laveste nivået reagere raskt og flytte systemet i riktig retning. Det vil kanskje ikke være helt optimalt, men dette kan korrigeres på en lengre tidsskala.

Et eksempel der dette ikke fungerte, er politiets beslutningssystem under Utøya-tragedien. Her fikk det laveste beslutningsnivået, som var politistyrken som var rett ved Utøya, beskjed om å vente på ordre fra en sentral



Figur 3. Skisse av menneskets hierarkiske beslutningssystem.

myndighet. Det normale i en situasjon med skyting, er at de lokale styrker opererer autonomt og tar umiddelbar aksjon basert på den kunnskap og erfaring de har. Dette vil kanskje ikke være optimalt på alle måter, men det gir en mye raskere responstid og ville høyst sannsynlig

vært en mye bedre løsning i Utøya-tragedien.

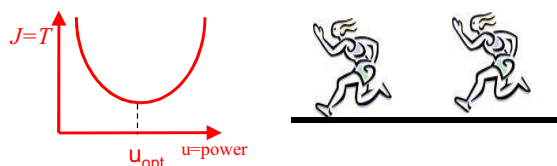
Det menneskelige beslutningssystem er bygget opp som et hierarkisk beslutningssystem med bevisstheten på toppen. En antydning er vist i figur 3 (slik tenker i alle fall jeg meg det, men jeg fikk vite under diskusjonen etter foredraget at dette er noe kontroversielt). Igjen er det viktig at det er en rimelig konsistens mellom de lavere ubevisste nivåene og det øverste bevisste nivået. Under alkoholpåvirket tilstand ser man kanskje hvordan systemet fungerer når man delvis kobler ut det høyere korrigerende bevissthetsnivået? Et annet eksempel er hypnose hvor man kan få en person til å gjøre merkelige ting. Det som er interessant er at når personen kommer ut av hypnosen, så prøver ofte bevisstheten, som jeg antar har vært delvis koblet ut, å gi en logisk forklaring på det rare som skjedde. Personen prøver kanskje å skape en konsistens mellom sine beslutningsnivåer.

Jeg har ikke mye kjennskap til hvordan hjernen fungerer, men ut fra min kunnskap om andre beslutningssystemer så mener jeg det må være en del elementer som må være tilstede. Det viktigste er utstrakt bruk av hierarkisk regulering på ulike tidsskalaer og utstrakt bruk av settpunktsstyring med lokal tilbakekobling (negative feedback).

Selvoptimaliserende variabler

For å forstå hva en selvoptimaliserende variabel er, la oss se på et enkelt eksempel, nemlig optimal strategi for en friidrettsløper (Skogestad, 2004b).

Målet er å bruke kortest mulig tid (T), så objektfunksjonen som vi skal minimere er $J=T$. Hva er den beste strategien? Vi har en pådragsvariabel (input u) som er hvor mye kraft vi bruker, dvs. hvor fort vi løper (se figur 4).



Figur 4. Ubegrenset optimum for maratonløper. Hva er beste løpsstrategi?

Tilfelle 1. Sprinter (100 m). For en sprinter er strategien enkel: Løp så raskt som du kan! Dette er ”active constraint control”, vi ligger på en begrensning. Siden begrensningen er på pådraget (input), så trenger vi her faktisk ikke regulering i det hele tatt!

Merk at det ikke alltid er slik at ”active constraint control” ikke krever regulering. Hvis målet er å kjøre raskest mulig med bil fra Trondheim til Oslo, så er fartsgrensen sannsynligvis den aktive begrensningen, dvs. $CV =$ fart. Så også her er strategien klar: Kjør så fort som det er tillatt! Men her trenger vi et reguleringssystem, og det som i dagligtale er kjent som ”cruise control”.

Tilfelle 2. Maratonløper (42 km). Dette er nøyaktig samme systemet som for en sprinter, men her virker opplagt ikke strategien om å løpe så raskt som du kan; da blir man veldig fort trett og slutt-tiden T vil bli stor. Dette er et eksempel på et ubegrenset (unconstrained) optimum, og da er det ikke så , så går selvsagt tiden T opp, og løper man for raskt, så vil man bli sliten og dermed ikke greie å holde farten, slik at også da går tiden T opp.

Er det noen ”lure” selvoptimaliserende variabler som vi kan holde på konstant settpunkt? Noen alternativer kan være:

- $CV_1 =$ avstand til den som leder
- $CV_2 =$ fart
- $CV_3 =$ puls (hjerterefrekvens)
- $CV_4 =$ melkesyre (”smerte”)

Den første strategien er veldig enkel, men den gir ikke nødvendigvis mini-

mum tid (T). Strategien kan også vise seg å være umulig hvis lederen er en bedre løper. Strategi 2 er også relativ enkel, men krever en måling av fart. Strategien kan også være umulig dersom løperen ikke greier å holde den ønskede farten. Dessuten er det opplagt at konstant fart ikke er optimalt hvis terrenget er ujevnt; det er ganske opplagt at det lønner seg å løpe litt langsommere i oppoverbakke. Fart er altså ikke en god selvoptimaliserende variabel fordi vi må justere settpunktet når vi har en forstyrrelse (d) i stigningen.

Strategi 3 og 4 er mer selvoptimaliserende; vi vil automatisk løpe litt saktere når vi kommer til en bakke dersom vi holder konstant puls eller konstant melkesyre. Måling av puls er dessuten veldig enkelt med pulsklokke. En løper vil også kjenne melkesyren godt, i alle fall etter en stund. En god strategi for å få beste tid i maraton vil kanskje være å løpe med konstant puls (CV_3) i begynnelsen av løpet og endre til konstant melkesyre (smerte) (CV_4) når det nærmer seg slutten.

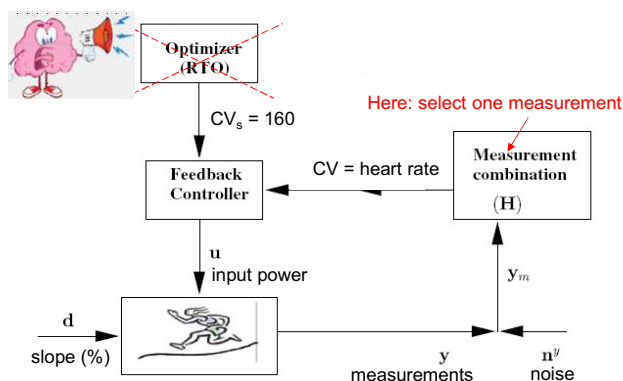
Eksempelet med løperen illustrerer på en god måte hvordan de fleste virkelige systemer reguleres. Hvis det er aktive begrensninger, trenger vi ikke å tenke så mye; det er bare å holde seg på grenseverdien. For ubegrensede systemer, som for maratonløperen, er det ikke så opplagt. Det kan til og med søkes patent hvis man finner en spesielt lur variabel å regulere.

Tabellen under viser mulige selvoptimaliserende variabler for en del systemer.

System	Variabel J som skal optimeres	Pådrag (u)	Selvoptimaliserende variabel (CV)	Typisk verdi settpunkt (CV_s)
Nasjonaløkonomi	Velferd	Lånerente	Inflasjon	2.5 %
Kakebaking	Smak	Varme	Ovnstemperatur	200 C
Postombæring	Profitt	Arbeidstimer	Leveringstid	2 dager
Investor	Kapital	Investering i bank/aksjer	Aksjeandel	60%
Maratonløper	Tid	Kraft	Puls	160
Forretningsvirksomhet	Profitt	Innsatsfaktorer	Key performance indicator (KPI)	Finnes fra «benchmarking»

Implementeringen med bruk av tilbakekobling skjer som illustrert for maratonløperen i figur 5.

Merk at i alle tilfellene i tabellen vil det være slik at det blir mindre optimalt om settpunktet er høyere eller lavere, dvs. vi har et ubegrenset optimum. For eksempel for postombæringen, så vil man bruke for mye ressurser dersom leveringstiden er kortere enn 2 dager. På den annen side vil man få



Figur 5. Maratonløper som bruker tilbakekobling til å holde pulsen konstant på tross av endringer i stigningen. Hvis det er nær optimalt å holde settpunktet konstant, for eksempel på 160 slag i minuttet, så kan man kutte optimaliseringen (treneren) i nivået over.

mye klager og sannsynligvis bli utkonkurrert om leveringstiden er lengre. Merk at inversoren som alltid setter 60 % av sin kapital i aksjer vil selge når kursen går opp og kjøpe når kursen går ned.

For en mer generell forretningsvirksomhet prøver man å holde KPI'er (Key performance indicators) på gitte verdier. KPI'er har funksjonen av selvoptimaliserende variabler for systemet, og den optimale verdien finnes ofte ved "benchmarking" med andre virksomheter. Eksempler på KPI'er i forretningslivet kan være:

- Responstid på ordre
- Energiforbruk pr. kg produsert vare
- Antall ansatte pr. kg produsert vare
- Forskningsinnsats pr. kg produsert vare

Biologiske systemer er robuste og basert på utstrakt bruk av tilbakekobling og hierarkisk regulering med settpunkter, for eksempel, for hormonnkonsentrasjoner, temperatur og blodtrykk. Dette kan også sees på som selvoptimaliserende variabler, og naturen har funnet fram til gode variabler ved evolusjon.

5. Hvordan finner man de selvoptimaliserende variablene?

Den ideelle selvoptimaliserende CV'en er gradienten, som skal holdes på null. Som et eksempel la oss anta at vi ønsker å gå langs en fjellrygg i tett tåke. Da vil vi holde oss langs ryggen dersom gradienten, som i dette tilfellet er stigningen sideveis, holdes på null.

Problemet er at vi sjelden har en måling av gradienten, så da gjelder det å finne en annen "lur" variabel, og dette vil være en selvoptimaliserende variabel. Det viktigste er å finne hvilken variabel som skal holdes på et konstant settpunkt. Merk at vi også må kjenne settpunktet, men dette kan finjusteres senere fra nivået over.

En god selvoptimaliserende variabel (CV) bør oppfylle følgende kriterier (Skogestad, 2000, 2004b)

1. Det optimale settpunktet (CV_{opt}) er *lite sensitivt* for forstyrrelser (d)
2. Variabelen (CV) er *sensitiv* for endringer i pådraget (u)
3. Variabelen (CV) er lett å måle og regulere.

Kriterium 2 er ikke helt opplagt, men det er meget viktig. Det er også kjent som "maximum gain rule".

La oss anvende de tre kriteriene på maratonløperen. Pulsen (CV_3) og melkesyren (CV_4) er gode kandidater som oppfyller alle tre kriteriene. Farten (CV_2) oppfyller ikke kriterium 1, fordi vi må senke farten i oppoverbakke.

De to første kriteriene kan bevises matematisk, og man kan ut fra dette finne optimale selv-optimaliserende variabler. Kriterium 1 er basis for nullromsmetoden (Alstad og Skogestad, 2007) og kombinasjonen av kriterium 1 og 2 er basis for det som kalles "exact local method" (Alstad et al, 2009). Disse metodene gir CV som en lineær kombinasjon av målingene y , $CV = H*y$. I referansene er det gitt en rekke praktiske eksempler fra prosessregulering, for eksempel for en kjølesyklus (Skogestad, 2000; Skogestad, 2004b).

For visse ulineære systemer kan man eliminere ukjente målinger og forstyrrelser fra gradienten og ut fra dette finne en optimal ulineær kombinasjon (Jäschke og Skogestad, 2012), men uttrykkene blir fort veldig kompliserte.

Konklusjon

Virkelige beslutningssystemer er hierarkisk oppdelt basert på tidsskala-separasjon og settpunktsstyring med utstrakt bruk av lokal tilbakekobling.

Et meget viktig valg i et slikt system er hvilke variabler man skal holde på settpunkt. Aktive begrensninger bør alltid holdes på settpunkt for å holde systemet optimalt. De gjenværende frihetsgradene brukes til å holde selv-optimaliserende variabler på gitte settpunkt. Målet er at disse settpunktene kan holdes konstant over en viss tidsperiode, på tross av forstyrrelser som krever endring i beslutningene.

Referanser

- Alstad, V. og Skogestad, S. (2007). Null Space Method for Selecting Optimal Measurement Combinations as Controlled Variables, *Ind.Eng. Chem.Res.*, 46, 846–853.
- Alstad, V., Skogestad, S. og Hori, E.S. (2009). Optimal measurement combinations as controlled variables, *Journal of Process Control*, 19, 138–148.
- Jäschke, J. og Skogestad, S. (2012). Optimal controlled variables for polynomial systems, *Journal of Process Control*, 22, 167–179 (2012).
- Skogestad, S. (2000). Plantwide control: the search for the self-optimizing control structure, *Journal of Process Control*, 10, 487–507.
- Skogestad, S. (2004a). Control structure design for complete chemical plants, *Computers and Chemical Engineering*, 28, 219–234.
- Skogestad, S. (2004b). Near-optimal operation by self-optimizing control: From process control to marathon running and business systems, *Computers and Chemical Engineering*, 29, 127–137.