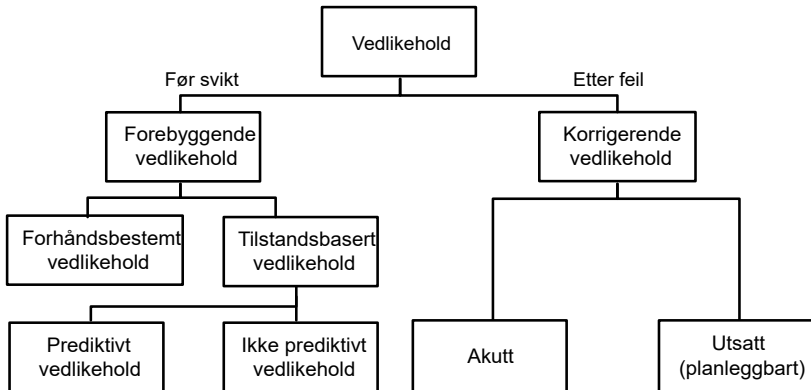


FLYT1102 - FLYVEDLIKEHOLD

Jørn Vatn/Februar 2025

Vedlikehold



Tilgjengelighet

Tilgjengelighet defineres som *andelen av tiden en enhet er i stand til å yte krevd funksjon.*

$$A = \frac{\text{MTTF}}{\text{MTTF} + \text{MDT}}$$

Utilgjengelighet ble defineres som *andelen av tiden en enhet ikke er i stand til å yte krevd funksjon.*

$$U = 1 - A = \frac{\text{MDT}}{\text{MTTF} + \text{MDT}}$$

Funksjon, svikt og feil

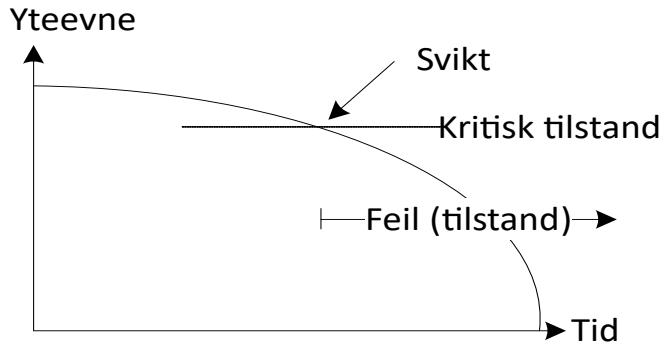
Alle komponenter vi betrakter i dette kurset er designet for å utføre en eller flere funksjoner.

- ▶ Pumpe vann
- ▶ Stenge væskestrøm (ventil)
- ▶ Gi moment (motor)

Svikt defineres som *tidspunktet* hvor en enhet ikke lenger er i stand til å utføre krevd funksjon. Begrepet feil(tilstand) benyttes om en *tilstand* hvor en enhet ikke er i stand til å utføre krevd funksjon.

- ▶ Failure = svikt (dvs tidspunkt/hendelse)
- ▶ Fault = feil(tilstand)

Svikttidspunkt og feiltilstand



Feilmode

Enhver komponent er konstruert for å utføre en eller flere funksjoner. Fravær av en slik funksjon kalles feilmode. Feilmoden angir hvordan fravær av funksjon arter seg når vi betrakter enheten sett utenfra:

- ▶ Total tap av funksjon (pumper ikke i det hele tatt)
- ▶ Redusert funksjon (pumper for lite)
- ▶ Feil funksjon (pumper i feil retning)

Feilmekanisme

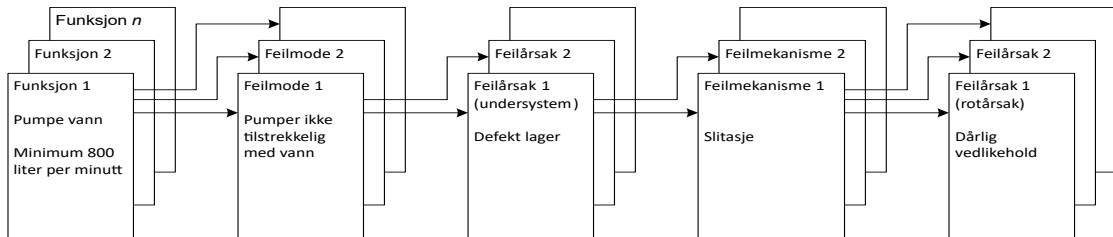
Feilmekanismer relaterer seg til fysiske, kjemiske eller andre prosesser som forringer enheten, og leder til svikt

- ▶ Slitasje
- ▶ Korrosjon
- ▶ Kavitasjon
- ▶ Utmattning

Feilårsak benyttes på to måter:

- ▶ Svikt på et lavere nivå, f eks defekt lager i ei pumpe
- ▶ Rotårsak, f eks dårlig vedlikehold, uhensiktsmessig design osv

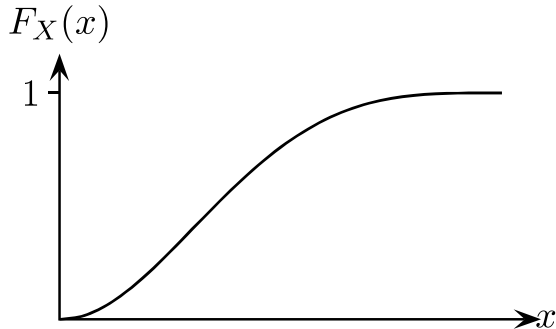
Sammenheng mellom funksjon, feilmode, feilårsak og feilmekanisme



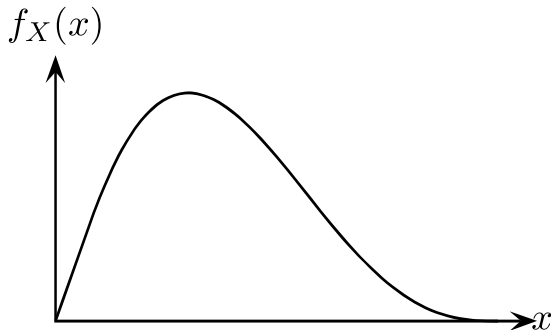
Sannsynlighetsregning

- ▶ Tilfeldige størrelser, X , T (STORE BOKSTAVER)
- ▶ Fordelingsfunksjon
- ▶ Sannsynlighetstetthet
- ▶ Overlevelsessannsynlighet
- ▶ Sviktintensitet

Kumulativ fordelingsfunksjon, $F_X(x) = \Pr(X \leq x)$



Sannsynlighetstetthet, $f_X(x) = dF_X(x)/dx$



Forventningsverdi

$$E[X] = \int_{-\infty}^{\infty} x f_X(x) dx$$

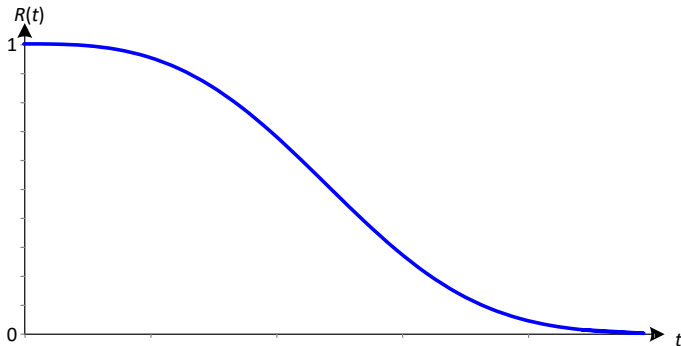
MTTF angir midlere tid til svikt (eng: Mean Time To Failure), og er for T (tid-til-svikt) gitt ved:

$$\text{MTTF} = E[T] = \int_0^{\infty} R(t) dt$$

Overlevelsessannsynlighet

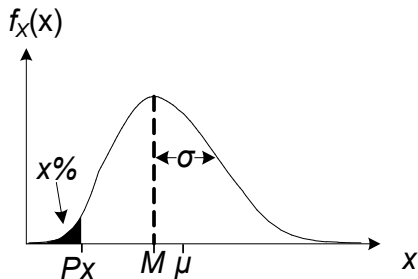
Overlevelsessannsynligheten angir sannsynligheten for tid-til-svikt T er større enn en verdi t :

$$R(t) = \Pr(T > t) = 1 - F_T(t)$$



Parametre

- ▶ Mest sannsynlige verdi (M)
- ▶ Forventet verdi (expected value, mean) ($\mu = E[X]$)
- ▶ Standardavvik ($\sigma = SD(X)$)
- ▶ Varians ($\sigma^2 = \text{Var}(X)$).



Sviktintensiteten, $z(t)$

Sviktintensiteten uttrykker den betingede sannsynligheten for at en enhet som fortsatt fungerer svikter i et lite tidsintervall.



Betinget overlevelsessannsynlighet

Overlevelsessannsynligheten $R(t)$ angir sannsynligheten for at en ny enhet overlever t tidsenheter. Dersom vi vet at enheten har overlevd t tidsenheter og vi ønsker å finne ut sannsynligheten for at den vil overleve x nye tidsenheter fra tidspunktet t kan vi beregne:

$$\begin{aligned} R(x|t) &= \Pr(T > t+x | T > t) = \Pr(T > t+x \cap T > t) / \Pr(T > t) \\ &= \Pr(T > t+x) / \Pr(T > t) = R(t+x) / R(t) \end{aligned}$$

Ekspensialfordelingen

Ekspensialfordelingen benyttes ofte for å beskrive levetider for komponenter som ikke er utsatt for aldring. Sannsynlighetstettheten til T gitt ved:

$$f_T(t) = \lambda e^{-(\lambda t)}$$

Overlevelsessannsynligheten til T er gitt ved:

$$R(t) = e^{-\lambda t}$$

hvor λ er den konstante sviktintensiteten i fordelingen. Forventning er gitt ved:

$$E[T] = \frac{1}{\lambda}$$

Ekspensialfordelingen har konstant sviktintensitet, dvs $z(t) = \lambda = 1/\text{MTTF}$

Weibullfordeling

Weibullfordelingen benyttes ofte for å beskrive levetider for komponenter som aldres. Sannsynlighetstetthet:

$$f_T(t) = \alpha \lambda (\lambda t)^{\alpha-1} e^{-(\lambda t)^\alpha}$$

Overlevelsessannsynlighet::

$$R(t) = e^{-(\lambda t)^\alpha}$$

Forventningsverdi:

$$E[T] = \frac{1}{\lambda} \Gamma\left(\frac{1}{\alpha} + 1\right)$$

Sviktintensitet i Weibullfordeling

Weibullfordelingen har sviktintensitet

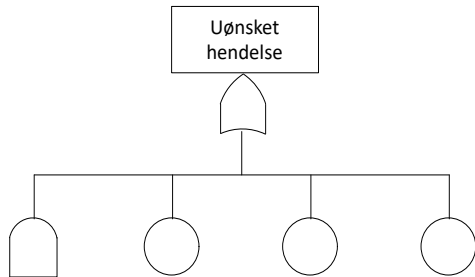
$$z(t) = \alpha \lambda^\alpha t^{\alpha-1}$$

Sviktintensiteten er:

- ▶ Økende for $\alpha > 1$ (IFR=increasing failure rate)
- ▶ Avtagende for $\alpha < 1$ (DFR=decreasing failure rate)
- ▶ Konstant for $\alpha = 1$

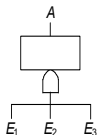
Feiltreanalyse

Et feiltre er et logisk diagram som illustrerer sammenhengen mellom en uønsket hendelse i et system og årsakene til denne hendelsen. Feiltreanalysen er en systematisk analyse for å etablere slike sammenhenger. Vi bruker ofte forkortelsen FTA = Fault Tree Analysis.

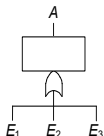


Symboler

OG-porten indikerer at utgangshendelsen A inntreffer hvis alle inngangshendelsene E_i inntreffer.

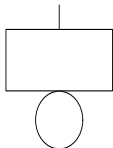


ELLER-porten indikerer at utgangshendelsen A inntreffer hvis minst en av inngangshendelsene E_i inntreffer.



Symboler, forts

En *basishendelse* er et symbol for komponent i primær feiltilstand, oppstått under normal drift. Vi bruker også en basishendelse for å angi menneskelig feilhandling samt “normale” hendelser.



Definisjon av den uønskede hendelsen (TOPP-hendelsen)

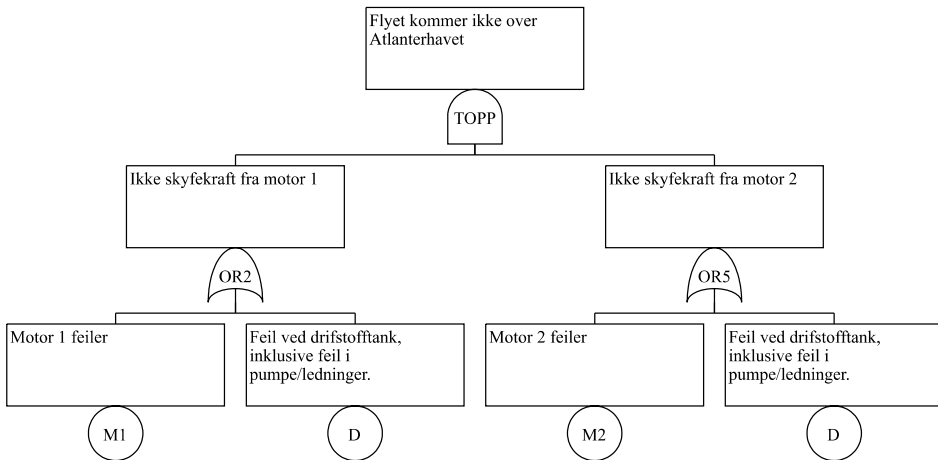
- ▶ *Hva:* F eks brann
- ▶ *Hvor:* I kontrollrom
- ▶ *Når:* Under normal drift

Konstruksjon av feiltreet

Å tegne selve feiltreet følger en systematisk tilnærming:

- ▶ Start med "TOPP"-hendelsen
- ▶ Spør hvilke hendelser som er de direkte årsaker til "TOPP"-hendelsen
- ▶ Kople disse sammen med en OG/ELLER-port
- ▶ Hver av hendelsene behandles nå på samme måte, og man arbeider seg suksessivt ned til basishendelsene

Konstruksjon av feiltreet, eksempel



Bestemmelse av minimale kuttmengder

- ▶ En kuttmengde i et feiltre er en mengde av basishendelser som ved å inntreffe sikrer at TOPP-hendelsen inntreffer
- ▶ En kuttmengde sies å være minimal hvis den ikke kan reduseres uten å miste status som kuttmengde

For å finne minimale kuttmengder kan man:

- ▶ Bruke MOCUS (Method for Obtaining Cut Sets) algoritmen (omfattende)
- ▶ Inspisere feiltreet for direkte å finne kuttmengdene (små trær)
- ▶ Kjøpe et feiltreprogram
- ▶ Håpe at kuttmengdene blir oppgitt på eksamen

Kvantitativ analyse av feiltrær

Formålet med den kvantitative analysen er å beregne ulike systemmål knyttet til den uønskede hendelsen. De viktigste systemmålene er:

- ▶ $Q_0(t)$ = Sannsynligheten for at TOPP-hendelsen er inntruffet ved tid t (utilgjengelighet)
- ▶ $F_0(t)$ = Frekvens av TOPP-hendelsen ved tid t
- ▶ $R(t)$ = Sannsynligheten for at TOPP-hendelsen ikke har inntruffet i $[0, t >$

Vi har kun sett på $Q_0(t)$ for reparerbare systemer og $R(t)$ for ikke-reparerbare systemer.

Beregning av q_i -ene

For en ikke-reparerbar enhet er sannsynligheten for at den er i en feiltilstand på tidspunktet t lik $1 - R(t)$:

$$q_i(t) = F(t) = 1 - e^{-\lambda_i t}$$

der λ_i er den konstante feilraten til komponenten og vi antar eksponensialfordeling.

Dersom enheten har økende sviktintensitet, og enheten har levd i t_0 tidsenheter, vil sannsynligheten for at den ikke overlever t nye tidsenheter være:

$$q_i(t) = 1 - R(t + t_0)/R(t_0)$$

Dersom $t \ll t_0$ kan vi benytte tilnærmingen:

$$q_i(t) \approx z(t_0)t$$

Beregning av q_i -ene, forts

Reparerbar enhet

$$q_i \approx \lambda_i \text{MDT}_i$$

Enheter som funksjonstestes har skjult funksjon. Dette betyr at dersom funksjonsevnen opphører, vil dette ikke være åpenbart for operativt personell. For å avdekke skjulte feil foretas funksjonstest. Intervallet mellom funksjonstester betegnes funksjonstestintervallet (τ):

$$q_i \approx \lambda_i \tau_i / 2$$

Beregning av $Q_0(t)$

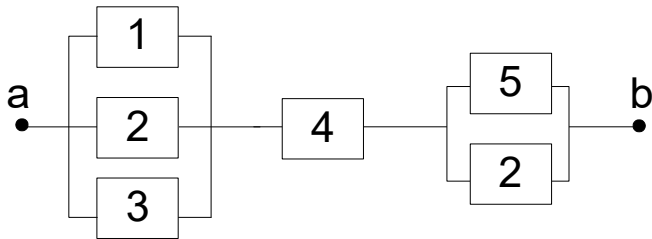
Sannsynligheten for at kuttmengd K_j har inntruffet:

$$\check{Q}_j = \prod_{i \in K_j} q_i$$

Sannsynligheten for at TOPP-hendelsen inntreffer finnes ved å summere bidragene fra hver kuttmengde:

$$Q_0 = \sum_{j=1}^k \check{Q}_j$$

Beregning av $Q_0(t)$, illustrasjon



$$\check{Q}_1 = q_1 \cdot q_2 \cdot q_3$$

$$\check{Q}_2 = q_4$$

$$\check{Q}_3 = q_5 \cdot q_2$$

$$Q_0 = \check{Q}_1 + \check{Q}_2 + \check{Q}_3$$

RCM = Pålitelighetsstyrt vedlikehold RCM = Reliability Centred Maintenance

- ▶ systematisk analyse av systemfunksjoner
- ▶ hvordan disse kan feile, for å kunne:
- ▶ etablere et forebyggende vedlikeholdsprogram
- ▶ hvor vi spesielt har sett på:
 - ▶ Funksjonsfeilanalyse
 - ▶ FMECA = Feilmode og effekt analyse
 - ▶ RCM beslutningslogikk for å bestemme type vedlikehold
 - ▶ Intervallfastsettelse

Funksjonsfeilanalyse - Sykkel

System: Bremsesystem

Performed by: Jørn Vatn

Page: 1 of: 1

Operational mode	Function	Function requirements	Functional failure	Criticality			
				S	E	A	C
Stille	Holde sykkel i ro	Tilstrekkelig med enten for- eller bakkbrems	Ikke tilstrekkelig bremskraft	L			
Bevegelse	Redusere hastighet	Variabel bremskraft	Ingen bremskraft	H			
			Redusert bremskraft	M			
			Utisliktet atkivering	M		M	M
			Ujevn bremskraft	L		L	L

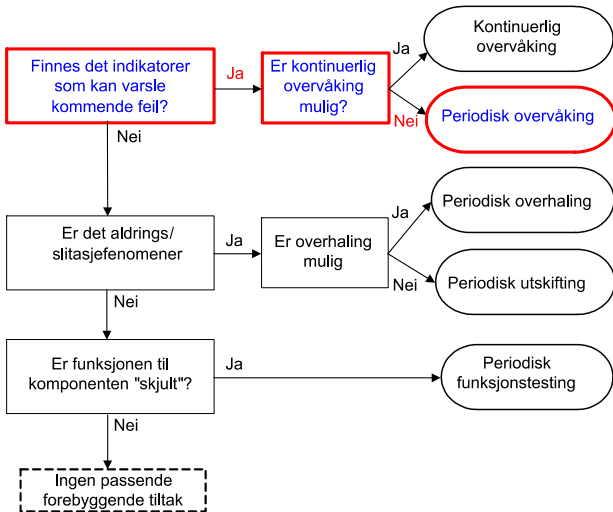
MSIer: Brems håndtak, bremsewire, bremsesklosser, feste for bremsesklosser



FMECA - Sykkel

MSI	Op.tilst.	Funksjon	Feilmode	Systemeffekt	Feilårsak/ Feilmekanisme	Feilutvikling
Bremsekloss	Passiv	Sørge for fri gange (ikke berøre felg)	Berører felg	Økt energiforbruk	Feil montert, ikke til-strammet skruer	Kan observere om klossene løsner, og berører felg
	Aktiv	Gi friksjonskraft	Begrenset friksjonskraft	Fare for kollisjon	Slitasje	Observerbar
					Oljefilm	Ingen
					Feil gummi-kvalitet	Ingen
			Gir ikke friksjonskraft	Fare for kollisjon	Slitasje	Observerbar

RCM-logikk: Bremsekloss, slitasje



MSG-3

- ▶ MSG står for “Maintenance Steering Group” og er kanskje et litt missvisende navn på en metode for å etablere et forebyggende vedlikeholdsprogram
- ▶ Navnet har en historisk bakgrunn ved at det på 60-tallet ble satt ned en arbeidsgruppe som skulle jobbe med å etablere mer hensiktsmessige vedlikeholdsprogram for fly
- ▶ MSG-konseptet har blitt utviklet over tid, og er i dag kjent som MSG-1, MSG-2 og MSG-3.

Historie-3

- ▶ MSG-1 ble først publisert i 1968 og brukt til å utvikle forebyggende vedlikehold for Boieng 747
- ▶ Semere kom MSG-2 som var prosessorientert og brukte en nedenfra-og-opp-tilnærming ("bottm-up"). MSG-2 introduserte også konseptet "tilstandsovervåket vedlikehold"
- ▶ Basert på erfaringene og identifiserte svakheter i MSG-2 ble MSG-3, og første gang publisert i 1980. Siden da har det kun vært oppdateringer av MSG-3. Metoden endret da karatker til en ovenfra-og-ned-tilnærming ("top-down") ved å fokusere på *konsekvenser av feil*.

MSG-3 vs RCM

Det er verdt å merke seg at RCM-metoden ble utviklet i parallel med MSG. Mens MSG-3 har blitt en metode som kun benyttes innen luftfart, har RCM et lang bredere anvendelsesområde. Og dersom man ser bort fra flyindustrien, er det kun RCM man snakker om når det gjelder metodisk tilnærming slik MSG la opp til. Første RCM publikasjon var i 1978.

MSG-3 og ansvar

- ▶ MSG-3 er et dokument utviklet av Airlines For America (tidligere Air Transport Association eller ATA). Dokumentet presenterer en metodikk for å utvikle et forebyggende vedlikeholdsprogram med planlagte vedlikeholdsoppgaver og intervaller som vil være akseptable for reguleringsmyndighetene, operatørene og produsentene.
- ▶ Hovedideen bak dette konseptet er å ta utgangspunkt i den innebygde påliteligheten til systemer og komponenter, unngå unødvendige vedlikeholdsoppgaver, og øke effektivitetene.

MSG-3 antagelser

- ▶ Vedlikehold er bare effektivt hvis vedlikeholdsoppgavene har en mening
- ▶ Man oppnår ingen forbedring av påliteligheten ved overdrevt vedlikehold
- ▶ Unødvendige oppgaver kan også medføre menneskelige feil
- ▶ For komplekse systemer er det sjeldent at slitasje er den dominerende feilårsaken
- ▶ Generelt er overvåking mer effektivt enn overhaling/utskifting ved faste intervaller, dvs bruk av tilstandsabsert vedlikehold
- ▶ Pålitelighet kan kun bedres ved modifikasjon
- ▶ Vedlikehold er kanskje ikke nødvendig dersom det er billigere å kjøre til svikt

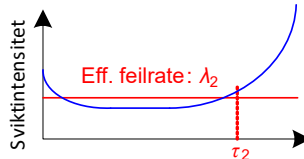
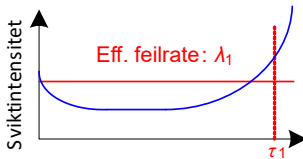
MSG-3 er mye brukt til å utvikle et innledende vedlikeholdsprogram for moderne kommersielle fly. Dette programmet utvikles som en del av prosessen med typesertifisering.

MSG-3 vs RCM

- ▶ Innholdsmessig er MSG-3 dokumentet langt mindre konkret enn dokumenter som beskriver RCM
- ▶ Mens RCM legger stor vekt på en FMECA, har MSG-3 ingen slik systematisk gjennomgang av feilmøder, feilmekanismer og feilårsaker
- ▶ Det er også verdt å merke seg at terminologi i MSG-3 er annerledes enn vanlig terminologi som brukes innenfor RAMS-området
 - ▶ Spesielt kan man merke seg bruken av begrepet “functional failure”. Dette begrepet svarer nok mer til begrepet “feilmode” slik vi bruker det i FMECA-analysen, hvor feilmode beskriver hvordan funksjon ikke kan oppnås, f eks FTO = Fail To Open = Åpner ikke (ventil).

Intervalloptimalisering

- ▶ Vi trenger å vite sammenheng mellom vedlikeholdsintervall, τ og
- ▶ effektiv feilrate = $\lambda_E(\tau)$



Kostnadsligning

Vi setter opp kostnadsligningen, dvs forventet kostnad per tidsenhet:

$$C(\tau) = C_{PM}/\tau + \lambda_E(\tau)[C_{CM} + C_{EP} + C_{ES} + C_{EM}]$$

hvor

- ▶ C_{PM} er kostnad for forebyggende vedlikehold (preventive maintenance)
- ▶ C_{CM} er kostnad for korrigerende vedlikehold (corrective maintenance)
- ▶ C_{EP} er forventet produksjonstap (evt punktlighetstap) ved en svikt
- ▶ C_{ES} er forventet sikkerhetstap ved en svikt
- ▶ C_{EM} er forventet materielle tap ved en svikt

Produksjonstap / forsinkelsestap

$$C(\tau) = C_{PM}/\tau + \lambda_E(\tau) [C_{CM} + C_{EP} + C_{ES} + C_{EM}]$$

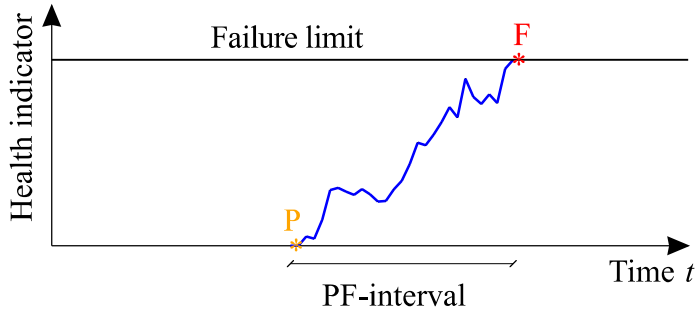
$$C_{EP} = \Pr(P) \cdot (C_P \text{MDT} + C_T)$$

- ▶ hvor $\Pr(P)$ er sannsynligheten for at en komponentsvikt leder til produksjonstap (punktlighestap osv)
- ▶ C_P = Tap per tidsenhet
- ▶ MDT = Midlere nedetid, påvirkes av logistikk, beredskap mm
- ▶ C_T = Direktekostnad ved svikt

Model for “badekarskurven”

- ▶ $\lambda_E(\tau) = \left(\frac{\Gamma(1+1/\alpha)}{\text{MTTF}} \right)^\alpha \tau^{\alpha-1}$
- ▶ Innsatt i kostnadsligningen, så deriviert mht τ , og satt lik null gir:
$$\tau = \frac{\text{MTTF}}{\Gamma(1+1/\alpha)} \sqrt[\alpha]{\frac{C_{\text{PM}}}{(\alpha-1)(C_{\text{CM}}+C_{\text{EP}}+C_{\text{ES}}+C_{\text{EM}})}}$$
- ▶ Modellen gjelder når vi i RCM-logikken ender opp med “overhaling” eller “utskifting”
- ▶ For “tilstandskontroll” og “funksjonstest” er det andre ligninger som gjelder, men dette dekkes ikke i FLYT1102

PF-modellen



Thank you for your attention

