RAMS

Reliability, Availability,
Maintainability, and Safety

TPK4190 - Produksjonsteknologi

Forelesningsnotater - Sikkerhet, pålitelighet og vedlikehold

Jørn Vatn

March 2023

Department of Mechanical and Industrial Engineering
Norwegian University of Science and Technology

Forord

Dette forelesningsnotatet er utviklet for emnet TPK4190 - Produksjonsteknologi. Notatet gir en kort innføring i to av teknikkene vi benytter innenfor området sikkerhet, pålitelighet og vedlikehold.

Vi trenger litt enkel sannsynlighetsregning for å benytte noen av metodene.

Trondheim, 2023-03-13

Jørn Vatn

Contents

Forord i

1 Introduksjon **2**

1.1 Bakgrunn 2

1.2 Formål 3

1.3 Motivasjon 3

2 FMECA **7**

2.1 Grunnleggende spørsmål 7

2.2 Når brukes FMECA 8

2.3 Hovedinnhold i analysen 8

2.4 FMECA skjema 8

2.4.1 Operasjonell tilstand 9

2.4.2 Funksjon 9

2.4.3 Feilmode 9

2.4.4 Eksempler på “mini”-FMECA 10

2.4.5 Feilårsak 10

2.4.6 Feilmekanisme 11

2.4.7 Kritikalitet / Risiko 11

2.5 Eksempel - Terrengbil 11

2.5.1 Funksjonsdiagram 11

2.5.2 Terrengbil - Komponenter 13

2.5.3 Funksjonsfeilanalyse - Traksjon 13

2.5.4 FMECA-skjema for terrengbil 14

3 Feiltreanalyse **17**

3.1 Bakgrunn 17

3.2 Byggeklosser - Feiltresymboler 18

3.2.1 Seriestruktur 18

3.2.2 Parallelstruktur 18

3.2.3 KooN-struktur 18
3.2.4 Basishendelse 19
3.3 Trinn i en feiltreanalyse 19
3.3.1 Definisjon av problem og randbetingelser 19
3.4 Konstruksjon av feiltreet 20
3.5 Bestemmelse av minimale kuttmengder 20
3.6 Kvalitativ analyse av feiltrær 21
3.7 Kvantitativ analyse av feiltrær 21
3.8 Cara Fault Tree 23

Kapittel 1

Introduksjon

1.1 Bakgrunn

En rekke store ulykker (Piper Alpha, Tsjernobyl, romfergen Columbia, m.fl.) er forårsaket av svikt i samspillet mellom menneske, teknologi og organisasjon (MTO). Felles for disse ulykkene er at de i ettertid har bidratt til økt sikkerhet og pålitelighet i sine respektive bransjer - offshore olje og gass, kjernekraft, luftfart og romfart - et resultat av lærdom tilegnet blant annet gjennom granskning av ulykker.

Fagområdet sikkerhet, pålitelighet og vedlikehold handler om viktige systemegenskaper. Tidligere ble ofte slik kunnskap ikke integrert i systemutviklingsprosessen, og kom dermed i bakgrunnen for kostnadsvurderinger og fokus på økonomi. I dag gjør tekniske nyvinninger, endrede rammebetingelser og økt fokus på allmenn sikkerhet at det stadig blir viktigere å ivareta sikkerhets- og pålitelighetsaspektet ved enkle så vel som komplekse systemer. Sikkerhet og pålitelighet er blitt et fagområde som bidrar med helt nødvendig kunnskap - ikke bare etter at ulykken har skjedd, men enda viktigere - i forkant.

Norge er i dag en ledende nasjon på sikkerhets-, pålitelighets-, og vedlikeholdsområdet, mye takket være kunnskap ervervet fra olje- og gassindustrien. Selv om grunnlaget for forskningen i stor grad er utviklet sammen med olje- og gassindustrien ser vi at metodene og modellene som er utviklet kan anvendes over alt. Dette har vi sett for eksempel i transportsektoren, i offentlig forvaltning, i kraftbransjen, og nå ser vi nytten av forskningen når vi skal utvikle grønt og blått hydrogen, og i satsningen på havvind.

Som student på studieprogrammet produktutvikling og produksjon kan du senere spesialisere deg i sikkerhet, pålitelighet og vedlikehold hvor du for eksempel kan lære om:

- redusere risiko knyttet til tekniske systemer for mennesker, miljø og materielle verdier
- redusere sårbarheten til tekniske systemer og informasjon

- bistå i utvikling av komponenter og systemer slik at de får lang levetid, få feil og liten nedetid
- verifisere at komponenter og systemer oppfyller sikkerhets- og pålitelighetskrav fra kunder og myndigheter
- analysere hvorfor og hvordan ulykker skjer
- utvikle konsepter og planer for effektivt vedlikehold av teknisk utstyr
- samle og analysere erfaringsdata fra drift og vedlikehold av teknisk utstyr
- optimalisere vedlikehold med hensyn på kostnader og driftsregularitet
- gjennomføre levetids-kostnadsanalyser (LCC analyser)

På engelsk betegnes fagområdet RAMS - Reliability, Availability, Maintainability and Safety.

1.2 Formål

Formålet med dette forelesningsnotatet er å gi studentene ved studieprogrammet produktutvikling og produksjon en kort innføring i fagområdet sikkerhet, pålitelighet og vedlikehold. Vi behandler kun to metoder ut av svært mange metoder som man senere kan fordype seg i. Vi vil benytte en terrengbil laget av Smørekoppen Engineering for noen år siden. I eksemplet ser vi på sikkerhet og pålitelighet for selve bilen, mens for emnet TPK4190 - Produksjonsteknologi, er nok fokuset egentlig større mht sikkerhet og pålitelighet av produksjonsprosessene.

1.3 Motivasjon

For noen år siden laget Smørekoppen Engineering en terrengbil i Institutt for maskinteknikk og produksjon sine lokaler. Figur 1.1 viser bilen som i dag er utstilt i Perleporten.

I emnet TPK4190 - Produksjonsteknologi er vi primært interessert i å se på *produksjonen* av deler til en slik terrengbil. Metodene og modellene vi benytter innenfor fagområdet sikkerhet, pålitelighet og vedlikehold er relevant for produksjonen av deler. Men for å gi eksempler som kanskje er litt lettere å forstå fokuserer vi i første omgang på de samme metodene og modellene når vi skal analysere sikkerhets-, pålitelighets- og vedlikeholdsegenskaper til terrengbilen.

Terrengbilen skal være sikker å kjøre. Fra bildet ser vi at det ikke er noen "veltebøyle", og bilen ville derfor neppe blitt godkjent for bruk. I tillegg til å risikoen knyttet til at bilen evt velter, vil vi også være opptatt va at den har gode bremser. Som vi ser har vi fire "hjul", og da må vi vurdere i hvilken grad vi trenger bremseeffekt for alle hjulene. Det vil avhenge av operasjonell tilstand,

Figur 1.1: Terrengbil

dvs dersom vi kjører sakte er det kanskje nok med bremsekraft på ett "hjul", mens dersom vi kjører i stor fart kreves trolig bremsekraft for alle "hjulene". I sikkerhetsfaget lærer vi hvordan vi kan analysere systemer med hensyn på for eksempel {Bilen gir ikke tilstrekkelig bremseeffekt}.

Terrengbilen skal også ha høy pålitelighet, dvs den skal virke når vi trenger den. Også her er det viktig å definere hva som menes med *virker*. Er det kun at vi skal kunne komme oss fra A til B, eller er det også krav om at vi skal kunne gjøre det så raskt som mulig. Avhengig av kravene, vil vi også ha ulike modeller når vi skal se på påliteligheten. Begrepet pålitelighet betyr i dagligtalen at vi kan stole på at enheten gjør det den skal.

Pålitelighet defineres som *en enhets evne til å utføre en påkrevd funksjon, under gitte miljø- og operasjonelle betingelser for en gitt tidsperiode*. Begrepet pålitelighet benyttes i ulike sammenhenger:

- Komponent- og systempålitelighet (fokus i dette emnet)
- Strukturpålitelighet
- Menneskelig pålitelighet
- Programvarepålitelighet

Bilen skal også kunne vedlikeholdes. Vedlikehold er viktig for å kunne opprettholde den innebygde påliteligheten til bilen, men er også viktig for å forhindre sikkerhetskritiske svikt.

Vedlikehold defineres som *en kombinasjon av alle tekniske og administrative aktiviteter, inkludert ledelsesaktiviteter som har til hensikt å opprettholde eller gjenvinne en tilstand som gjør en enhet i stand til å utføre en krevd funksjon.*

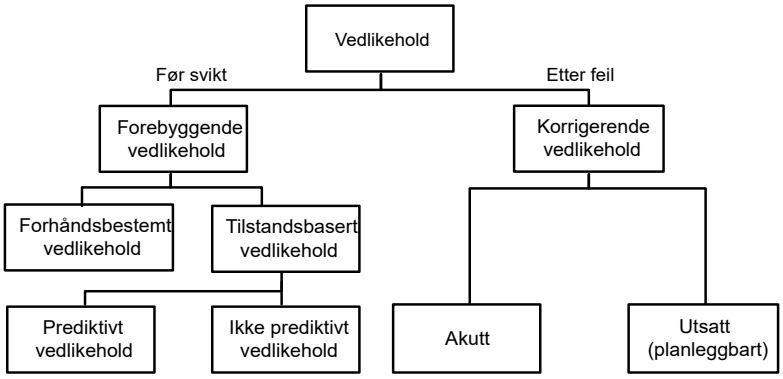
Vi skiller mellom:

- Forebyggende vedlikehold: Vedlikehold som utføres for å vurdere og/eller minske degraderingen og redusere sannsynligheten for svikt i en enhet
- Korrigerende vedlikehold: Vedlikehold som utføres etter at en feil er funnet, og som har som formål å gjenopprette en enhet til en tilstand der den kan oppfylle krevd funksjon

For forebyggende vedlikehold skiller vi mellom:

- Forhåndsbestemt vedlikehold: Forebyggende vedlikehold som utføres i henhold til etablerte tidsintervaller eller antall bruksenheter, men uten forutgående tilstandsundersøkelser
- Tilstandsbasert vedlikehold: Forebyggende vedlikehold som omfatter vurdering av fysisk tilstand, analyse og mulige påfølgende vedlikeholdstiltak, hvor vi igjen skiller mellom:
  - Prediktivt vedlikehold: Tilstandsbasert vedlikehold som utføres etter en prognose utledet av gjentatt analyse eller kjente egenskaper og evaluering av de vesentlige parameterne for degradering av enheten
  - Ikke prediktivt vedlikehold: Tilstandsbasert vedlikehold hvor det ikke finne en prognose for utvikling av degradering

Figur 1.2 viser strukturen i begrepene for vedlikehold.



Figur 1.2: Vedlikeholdsterminologi

For bilen ønsker vi ikke å gjøre for mye vedlikehold, det vil være kostbart, og gjør også at bilen ikke er tilgjengelig mens den er på verksted for vedlikehold. Vi ønsker også å gjøre vedlikeholdet på optimale tidspunkter. Vi kan for eksempel forsøke å optimalisere bytte av slitedeler. For noen slitedeler kan vi benytte prediktivt vedlikehold fordi det er mulig å måle tilstand, og ut fra målingene og annen kunnskap lage prognoser som kan forutsi tidspunkt for svikt, slik at vi kan iverksette et tiltak i rimelig tid før svikt inntreffer. For andre slitasjemekanismer kan det være

vanskelig å måle degradering, og vi må kanskje benytte alder som indikator for tidspunkt for svikt.

Anta nå at vi har funnet ut at det vanligvis er optimalt å skifte en kritisk komponent i terrengbilen ved et tidspunkt som er neste uke. Nå viser det seg imidlertid at neste uke er det en konkurranse hvor vi skal benytte bilen. Det vil si at vi har operasjonelle begrensninger i tidspunkt for vedlikehold. For å unngå suboptimalisering og "silotenking" mellom fagmiljøene for drift og vedlikehold, er det viktig å se vedlikehold i en driftsammenheng. Det betyr at modellene som vedlikeholdsavdelingen benytter må "spille på lag" med modellene som driftsavdelingen har, dvs de som skal benytte terrengbilen. Her blir det viktig å utvikle såkalte "digitale tvillinger", både for vedlikehold og drift. Digitale tvillinger behandles ikke i dette emnet, men vil bli behandlet dersom du senere velger å spesialisere deg i sikkerhet, pålitelighet og vedlikehold.

Kapittel 2

FMECA

Feilmode, effekt og kritikalitetsanalyse (FMECA) er en metode for å avdekke og analysere

- Alle potensielle feilmoder for ulike deler av et system
- Effekten disse kan ha på systemet
- Hvordan unngå feilmodene og/eller redusere konsekvensene disse har på systemet

Opprinnelig ble FMECA kalt FMEA, "C"-en i FMECA ble innført for å rangere alvorligheten av feileffektene. Vi vil i det etterfølgende benytte betegnelsen FMECA.

2.1 Grunnleggende spørsmål

En FMECA skal gi svar på følgende spørsmål:

- Hvilke feilmoder kan inntreffe for hver enkelt av komponentene i systemet?
- Hva er årsakene til disse feilmodene?
- Hvilken effekt har hver enkelt feilmode på resten av systemet?
- Hvordan oppdages feilmodene?
- Hvor ofte inntreffer hver feilmode?
- Hvor alvorlig er hver feilmode?
- Hva er risikoen knyttet til hver feilmode?
- Hvilke risikoreduserende tiltak kan være aktuelle?

2.2 Når brukes FMECA

FMECA kan brukes til:

- Gi bedre mulighet til å velge designalternativer med høy pålitelighet og sikkerhet i de tidlige designfaser
- Sikre at systemets mulige feilmoder og konsekvenser er vurdert
- Utvikle kriterier for planlegging av tester og krav til testutstyr
- Danne dokumentasjongrunnlag for fremtidig analyser av feil og designendringer
- Danne underlag for vedlikeholdsplanlegging
- Danne underlag for kvantitative pålitelighets- og tilgjengelighetsanalyser

Innefor flere bransjer stilles det krav om at FMECA skal inngå som en del av designprosessen, og at analyseresultatene skal være en del av systemdokumentasjonen.

2.3 Hovedinnhold i analysen

En FMECA-analyse starter med å gjøre nødvendige avgrensninger og valg av type skjema. Deretter utføres en systemanalyse:

- Del systemet inn i delsystemer som analyseres separat
- Etabler et funksjonsdiagram som viser hvordan delsystemene henger sammen
  - En funksjonell feilanalyse (FFA) kan være en nyttig tilnærming for å systematisere funksjoner og funksjonsfeil på et systemnivå
  - Under leksjonen RCM er funksjonell feilanalyse beskrevet
- For hvert delsystem identifiseres alle komponenter

Så følger hoveddelen av analysen som er utfylling av selve skjemaet med basis i komponentene

2.4 FMECA skjema

Det finnes ulike skjema for bruk til FMECA-analyser. Figur 2.1 viser et eksempel på et slikt skjema.

I det følgende diskuteres innholdet i noen av kolonnene.

System:									Performed by:			
Subsystem:									Date:			
Function:									Page			
description of unit			description of failure			effect of failure			failure rate	criticality	corrective action	remarks
identi-fication	operational mode	function	failure mode	failure cause/ mechanism	how to detect	local	system	operat. status				

Figur 2.1: FMECA skjema

2.4.1 Operasjonell tilstand

Ulike enheter kan ha ulike operasjonelle tilstander, f eks

- Pumpe: Aktiv ↔ Standby
- Ventil: Åpen ↔ Lukket

2.4.2 Funksjon

En enhet kan ha ulike funksjoner (for hver operasjonell tilstand). Vi skiller mellom typer av funksjon:

- Hovedfunksjon (pumpe vann)
- Hjelpefunksjoner (holde på vannet)
- Beskyttelsesfunksjoner (f eks sikkerhet; beskytte mot roterende deler)
- Informasjonsfunksjoner (gi statusinformasjon)

2.4.3 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)

2.4.4 Eksempler på "mini"-FMECA

Figur 2.2 viser utsnitt for en nødavstegningsventil:

Enhet	Op. tilst.	Funksjon	Feilmode
Nødav-stengn-ings ventil	Åpen	Sikre v .strøm	Lukker utilsikt.
		Lukke ved foresp.	Lukker ikke
			Lukker for sakte
	Lukket	Hindre v. strøm	Holder ikke tett
			Utilsiktet åpning
		Åpne ved foresp.	Åpner ikke

Figur 2.2: FMECA utsnitt for en nødavstegningsventil

Figur 2.3 viser utsnitt for en reservepumpe:

Enhet	Op. tilst .	Funksjon	Feilmode
Reserve pumpe	Standby	Starte ved behov	Starter ikke
	Aktiv	Pumpe 800 l pr min	Pumper ikke
			Redusert kapasitet
		Holde på oljen	Ekstern lekasje
		Gi status informasjon	Ingen informasjon
			Feil informasjon

Figur 2.3: FMECA utsnitt for en pumpe

2.4.5 Feilårsak

Mens en feilmode angir hvordan fravær av funksjon arter seg når vi betrakter enheten sett utenfra benyttes begrepet feilårsak for å si noe om hvorfor feilmoden inntreffer.

Begrepet feilårsak benyttes på to måter:

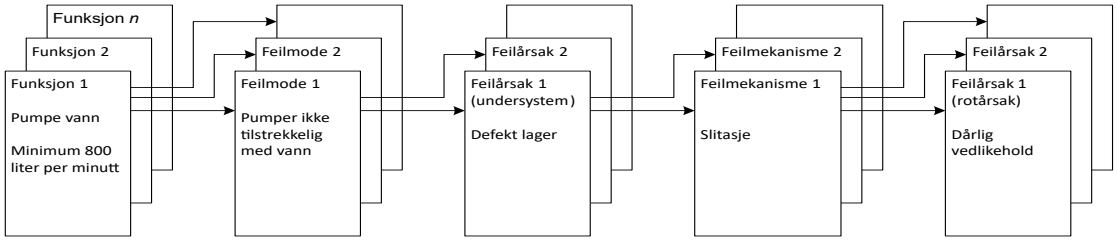
- Svikt på et lavere nivå, f eks defekt lager i ei pumpe
- Rotårsak, f eks dårlig vedlikehold, uhensiktsmessig design osv

2.4.6 Feilmekanisme

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

- Slitasje
- Korrosjon
- Kavitasjon
- Utmatting

Figur 2.4 viser sammenheng mellom funksjon, feilmode, feilårsak og feilmekanisme.



Figur 2.4: Sammenheng mellom funksjon, feilmode, feilårsak og feilmekanisme

2.4.7 Kritikalitet / Risiko

Ved å kombinere forekomst av en feilmode (feilrate) og konsekvens kan man etablere et risikomål (betegnes av og til for kritikalitet). Ofte synliggjør man i form av en risikomatrise. Figur 2.5 viser en slik matrise.

2.5 Eksempel - Terrengbil

Vi vil bruke terrengbilen i Figur 1.1 som grunnlag for å gjennomføre en FEMCA-analyse.

2.5.1 Funksjonsdiagram

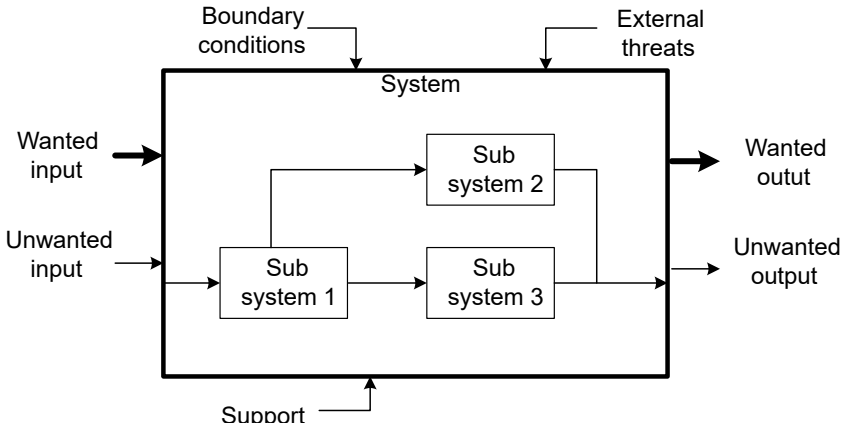
Et funksjonsdiagrammet kan brukes til å synliggjøre funksjonelle sammenhenger, randbetingelser osv. Figur 2.6 viser et generisk funksjonsdiagram:

Figur 2.7 viser funksjonsdiagrammet for terrengbilen. Her har vi følgende beskrivelser:

- Wanted input: En god sjåfør, oppladet batteri

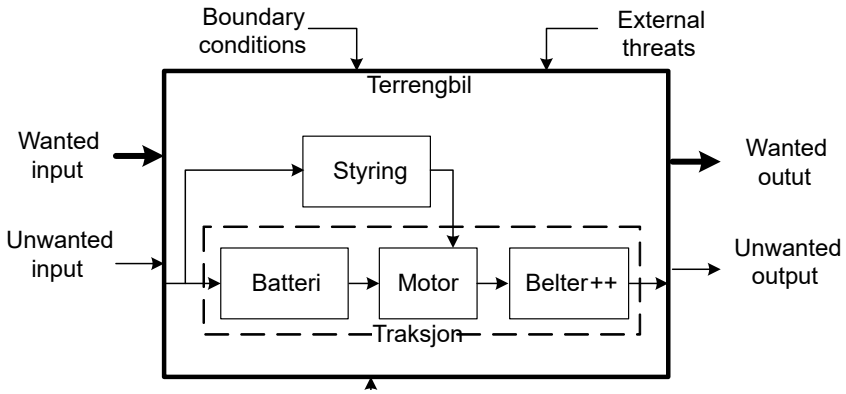
Frekvens / konsekvens	1 Ofte	2 Av og til	3 Mulig	4 Skjelden	5 Usannsynlig
1 Minimal					
2 Lav					
3 Middel					
4 Alvorlig					
5 Katastrofal					

Figur 2.5: Risikomatrise



Figur 2.6: Generisk funksjonsdiagram

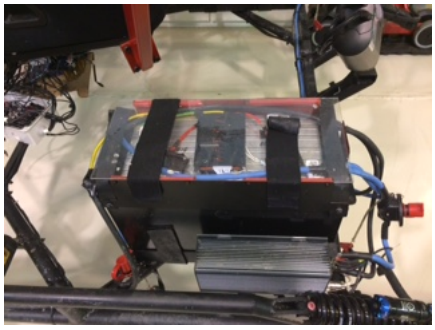
- Unwanted input: En aggressiv sjåfør
- Wanted outut: Kjøre fra A til B
- Unwanted output: Ulykke, komme for sent til B
- Support: Vedlikehold
- Boundary conditions: Kan kun kjøre i terreng, ikke lov å kjøre på offentlig veg
- External threats: Elg, tynn is



Figur 2.7: Funksjonsdiagram for terrengbil

2.5.2 Terrengbil - Komponenter

Nedenfor vises noen bilder av noen komponenter i terrengbilen. Disse komponentene er naturlige å ta videre inn i selve FMECA-skjemaet.



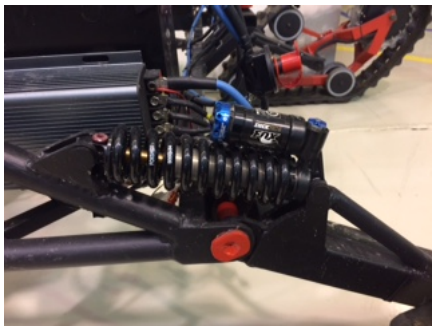
Figur 2.8: Batteri

2.5.3 Funksjonsfeilanalyse - Traksjon

En funksjonell feilanalyse gjennomføres på systemnivå å har som formål å få oversikt over systemet. Funksjonsfeilanalysen hjelper deg å få oversikt over funksjonene og kan tydeliggjøre innholdet i funksjonsdiagrammet i Figur 2.7. Spesielt viktig er:

- Identifisere systemet sine hovedfunksjoner
- Finne system-feilmodene (funksjonsfeil)
- Finne de kritiske feilmodene på systemnivå

Figur 2.9: Motor



Figur 2.10: Støtdemper/fjær

Figur 2.13 viser et et typisk skjema som benyttes for en funksjonsfeilanalyse. Merk her at bokstavene S, E, A og C står for henholdsvis sikkerhet, miljø, tilgjengelighet og kostand. Boktavene L, M og H står for henholdsvis lav, middels og høy.

2.5.4 FMECA-skjema for terrengbil

Den største jobben ved en FMECA-analyse er å fylle ut selve FMECA-skjemaet for alle komponentene. Figur 2.14 viser et utsnitt for batteriet.

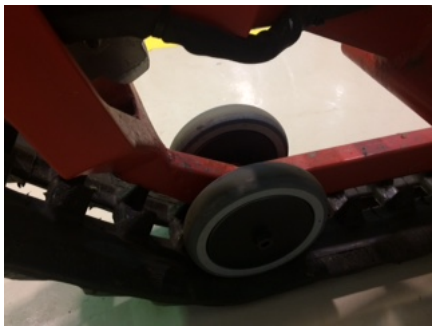
Øvingsoppgave 1

Velg ut en til to av komponentene vist i Figur 2.8 til Figur 2.12, eller velg en eller to adnre komponenter. Fyll inn relevant informasjon i FMECA-skjemaet. Du kan evt laste ned en Word fil som du finner på følgende lenke:

https://folk.ntnu.no/jvatn/eLearning/TPK4190/formFMECA.docx.

Diskuter spesielt

Figur 2.11: Belter



Figur 2.12: Støttehjul for belter

- Feilårsak
- Feilmekanisme
- Rotårsak

System: Traksjon		Performed by: Jørn Vatn		Page: 1 of: 1			
Operational mode	Function	Function requirements	Functional failure	Criticality			
				S	E	A	C
Stille	Holde bil i ro	Tilstrekkelig med brems på ett hjul	Ikke tilstrekkelig bremsekraft	H	-	L	M
Bevegelse	Fremdrift	Mulighet for variabel hastighet, styring ved ulik hastighet på belter	Ingen fremdrift	-	-	H	H
			Redusert fremdrift	L	-	M	M
			Ikke mulig med ulik hastighet på belter	M	L	M	M
			For stor fart	H	M	M	M
			Ujevn gange	-	L	H	M

Figur 2.13: Skjema for funksjonsfeilanalyse - Terrengbil

System: Terrengbil									Performed by: Jørn Vatn			
Subsystem: Traksjon									Date: 2017-03-07			
Function: Framdrift									Page 1			
description of unit			description of failure			effect of failure			failure rate	criticality	corrective action	remarks
identi-fication	operational mode	function	failure mode	failure mechanism	how to detect	local	system	operat. status				
Batteri	Lader	Ta opp strøm	Tar ikke opp strøm	Lednings-brudd	Måle motstand	Ikke strøm til motor	Ikke fram-drift	Bil står stille	1/10	Høy	Fixe ledning	Kvalitet i lodding
	I drift	Levere strøm	Leverer ikke strøm	Material-degradering	Måle ladetid	Ikke strøm til motor	Ikke fram-drift	Bil står stille	1/3	Høy	Nytt batteri	Riktig lading!

Figur 2.14: FMECA skjema - Terrengbil

Kapittel 3

Feiltreanalyse

3.1 Bakgrunn

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.

CARA FaultTree er et enkelt program for feiltreanalyse som er gratis for studenter ved NTNU.
Resultat fra en feiltreanalyse

- Liste over mulige kombinasjoner av feil og forhold som medfører den uønskede hendelsen
  - Miljøfaktorer
  - Menneskelige feilhandlinger
  - Normale hendelser
  - Komponentsvikt
- Sannsynlighet/frekvens av den uønskede hendelsen
- Liste over de mest betydningsfulle komponenter/hendelser

Trinn i en feiltreanalyse

1. Definisjon av problem og randbetingelser
2. Konstruksjon av feiltreet
3. Bestemmelse av minimale kutt- og stimengder
4. Kvalitativ analyse av feiltreet
5. Kvantitativ analyse av feiltreet

3.2 Byggeklosser - Feiltresymboler

Vi skiller mellom to typer symboler:

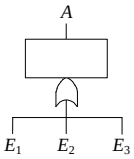
- Logiske porter som kobler sammen underliggende symboler
- Basishendelser (løvnoder i feiltreet)

Alle symboler i et feltre må få et navn (identifikator). Symboler med samme navn representerer den fysisk samme hendelsen. Identiske, men forskjellige hendelser må gis forskjellig navn. Identiske basishendelser vil ha samme pålitelighetsdata, f eks feilrate.

For å etablere et feiltre har vi noen symboler som representerer ulike type strukturer:

3.2.1 Seriestruktur

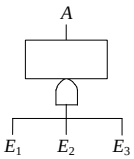
En seriestruktur er en struktur hvor alle komponentene i strukturen må fungere for at strukturen skal fungere. Det betyr også at dersom en eller flere av komponentene svikter, vil strukturen (systemet) svikte. I et feiltre representeres en seriestruktur ved en ELLER port. Rektanglet over porten er kun for å gi tilleggsinformasjon.



ELLER-port

3.2.2 Parallelstruktur

En parallellstruktur er en struktur hvor det er tilstrekkelig at en av komponentene i strukturen må fungere for at strukturen skal fungere. Det betyr også at alle komponentene må være i feiltilstand for at strukturen (systemet) skal være i feiltilstand. I et feiltre representeres en parallellstruktur ved en OG port. Rektanglet over porten er kun for å gi tilleggsinformasjon.



OG-port

3.2.3 KooN-struktur

En Koo$N$ (K-out-of-N) struktur er en struktur der minst K ut av N komponenter nå fungere dersom strukturen skal fungere. En slik struktur er da noe "imellom" en serie- og en parallellstruktur. Tilsvarende for en port i et feiltre, vil KooN-porten indikerer at utgangshendelsen A inntreffer hvis K eller flere av inngangshendelsene E_i inntreffer. Noen feiltreprogrammer har et eget symbol for Koo$N$-porten, hvor man da må angi K Merk at K-en blir forskjellig om vi fokuserer på at en struktur virker, kontra at den feiler. Derfor bruker vi ofte følgende notasjon for å tydeliggjøre:

- $k$oo$n$: G brukes når vi fokuserer på at strukturen fungerer (G=Good). strukturen fungerer dersom k eller flere av de n komponentene fungerer

- $k$oo$n : F$ brukes når vi fokuserer på at strukturen feiler (F=Fault state). Strukturen feiler, dvs porten eller TOPP-hendelsen inntreffer dersom k eller flere av n komponentene feiler

Følgende relasjon gjelder:

$$k\text{oo}n : G = (n–k+1)\text{oo}n : F$$
$$k\text{oo}n : F = (n–k+1)\text{oo}n : G$$

3.2.4 Basishendelse

Symbol for komponent i primær feiltilstand, oppstått under normal drift. Rektanglet ovenfor selve symbolet benyttes for å gi tilleggsinformasjon.

3.3 Trinn i en feiltreanalyse

3.3.1 Definisjon av problem og randbetingelser

Den såkalte TOPP-hendelsen er det øverste symbolet i analysen. Ofte betegnes dette den uønskede hendelsen, og eksempler kan være

- Ulykke
- Produksjonsstans
- Havari

For å definere TOPP-hendelsen benytter vi ofte følgende hjelpespørsmål

- Hva: Bilen bremser ikke
- Hvor: I nedoverbakke
- Når: Under normal drift

Randbetingelsene angir hva som er med i analysen, og hva som er utenfor. F eks kan vi se bort fra sabotasje, at vi ikke har ekstern strømforsyning for lading osv. Det er viktig å angi randbetingelsene, slik at man kan vurdere om det er behov for ytterligere analyser.

3.4 Konstruksjon av feiltreet

Følgende trinn inngår når man skal konstruere (tegne) et feiltre:

- 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

3.5 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, evt bruke CARA FaultTree som student (Se lenke til slutt i dette dokumentet)

I videregående emner (TPK4120) vil vi lære hvordan MOCUS algoritmen fungerer. Her ser vi på direkte inspeksjon av feiltreet:

- Start fra toppen
- Ved ELLER-port, ta med hendelser lenger ned fra hver gren inn til porten. For hver hendelse får vi en ny kuttmengde.
- Ved OG-port, ta med hendelser lenger ned fra hver gren inn til porten. Alle hendelsene vi tar med oss “opp” blir i en kuttmengde.
- Kan bli veldig stort dersom vi har mange OG porter
- Eliminer ikke-minimale kuttmengder

3.6 Kvalitativ analyse av feiltrær

Minimale kuttmengder med få basishendelser er de viktigste. Vi lister derfor opp kuttmengdene etter stigende orden. Kuttmengder av orden 1 representer potensielle enkeltfeilsituasjoner. I f eks jernbane ig olje- og gassnæringen er det myndighetskrav som krever at ingen enkeltfeil skal lede til død, alvorlig personskade eller alvorlige miljøutslipp.

3.7 Kvantitativ analyse av feiltrær

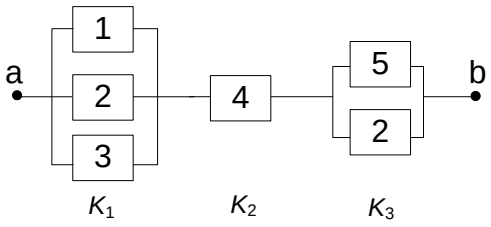
Formålet med den kvantitative analysen er å bergne ulike systemmål knyttet til den uønskede hendelsen. De viktigste systemmålene er:

- $Q_0(t)$ = Sannsynligheten for at TOPP-hendelsen er inntruffet ved tid t (utilgjengelighet)
- $F_0(t)$ = Frekvens av TOPP-hendelsen ved tid t
- $R_0(t)$ = Sannsynligheten for at TOPP-hendelsen ikke har inntruffet i $[0, t\rangle$
- $I^{\mathrm{B}}(i)$ = Birnbaum's mål for pålitelighetsmessig betydning av basishendelse i

For å beregne systemmålene kreves pålitelighetsdata for basishendelsene. I videregående emner ser vi på ulike situasjoner. Her forenkler vi, og antar at vi alltid kan finne q_i = sannsynligheten for at komponent i er i feiltilstand.

For beregninger vil vi kun demonstrere en tilnærming. Denne betegnes ofte for øvre skranketilnærmingen (Upper Bound Approximation) for Q_0

Anta at vi har funnet de minimale kuttmengdene for feiltreet, dvs K_j. Anta videre at de minimale kuttmengdene ikke inneholder felles basishendelser (komponenter), og at basishendelsene er stokastisk uavhengige av hverandre. Vi kan no stille opp kuttmengdene slik det er vist i Figur 3.1:



Figur 3.1: Eksempel på kuttmengdestruktur

La E_j betegne hendelsen at kuttmengde j inntreffer, dvs at alle komponentene i kuttmengden feiler. Sannsynligheten for at kuttmengde j inntreffer er:

$$\Pr(E_j) = \check{Q}_j(t) = \prod_{i \in K_j} q_i \tag{3.1}$$

Vi har nå at:

$$Q_0 = \Pr(\text{TOPP-hendelsen inntreffer}) = 1 - \Pr(\text{TOPP-hendelsen ikke inntreffer}) \tag{3.2}$$

$$= 1 - \Pr(\text{Ingen av kuttmengdene inntreffer}) \tag{3.3}$$

Siden kuttmengdene er uavhengig, og sannsynligheten for at kuttmengde j inntreffer er gitt ved $\check{Q}_j(t)$, har vi:

$$Q_0 = 1 - \prod_{j=1}^{k} (1 - \check{Q}_j) \tag{3.4}$$

where

$$\check{Q}_j = \prod_{i \in K_j} q_i \tag{3.5}$$

Generelt kan noen basishendelser inngå i flere kuttmengder, og da kan det vises at en øvre skranke er gitt ved::

$$Q_0 \leq 1 - \prod_{j=1}^{k} (1 - \check{Q}_j) \tag{3.6}$$

og vi benytter ofte:

$$Q_0 \approx 1 - \prod_{j=1}^{k} (1 - \check{Q}_j) \tag{3.7}$$

For de som ikke husker sannsynlighetsregning fra videregående skole: Kast en terning to ganger, og finn sannsynligheten for at vi får en 6-er i begge kast. Sannsynligheten for en sekser i første kast er 1/6, tilsvarende for andre kast, og sannsynligheten for en sekser i begge kast er 1/6×1/6= 1/36.

Øvingsoppgave 2

- Tegn et feiltre med TOPP-hendelse: {Ikke tilstrekkelig framdrift}

- Gjør analysen både om du trenger 3 motorer og 2 motorer for å ha tilstrekkelig framdrift
- Velg selv hvilke komponenter du vil ta med, men det er naturlig å ta med batteri, og reguleringssystem (gasspedal) og motorene

Finn Q_0 dersom feilsannsynlighet for hver motor er 10^{-2} og for batteriet 10^{-3}

3.8 Cara Fault Tree

Du kan laste ned et feiltreprogram her:

https://folk.ntnu.no/jvatn/eLearning/CaraFaultTree/Caftan.zip

Du må pakke ut .zip fila på en lokal disk. Programmet virker kun på PC. Programmet er skrevet for Windows 3.11, men virker for nyeste versjon av Windows. Du får en "registry" feil, men den kan ignoreres. Det er caftan.exe-fila som skal kjøres.