STK1500: Notat 11
Pålitelighetsteori – del 1

I pålitelighetsanalyse prøver man å beregne sannsynligheten for at et gitt *system* funksjonerer eller ikke. Med *system* menes her en teknologisk enhet bestående av en endelig mengde *komponenter* som opererer sammen. I denne sammenheng begrenser vi oss til å studere binære systemer. Et *binært system* har bare to mulige tilstander: *funksjonerende* eller *feilet*. Det samme gjelder for komponentene i systemet. Et *binært monotont system* er et binært system som er slik at reparasjon av en komponent aldri kan resultere i at systemet svikter, og ødeleggelse av en komponent kan aldri resultere i at systemet starter å fungere igjen.

Binære systemer kan være mer eller mindre kompliserte, og utfordringen ligger i å kunne analysere disse mest mulig effektivt. Generelt kan det vises at det å beregne påliteligheten til et vilkårlig binært system er et såkalt NP-hardt problem. Dette betyr at det ikke eksisterer (og antagelig aldri vil eksistere) noen effektiv algoritme for å finne påliteligheten. For visse klasser av systemer finnes det likevel effektive algoritmer. Selvom det ikke finnes effektive algoritmer som kan løse ethvert pålitelighetsproblem, kan det likevel være mye å vinne på å velge en *best mulig* metode.

Det er vanligvis også mulig å estimere påliteligheten til et system ved å benytte Monte Carlo simulering. En utfordring med slike metoder er at de ofte ikke er tilstrekkelig nøyaktige. Dette gjelder spesielt ved analyse av systemer med høy pålitelighet. I situasjoner der det er mulig å beregne påliteligheten eksakt, er dette derfor å foretrekke.

1 Binære monotone systemer

Vi betrakter et binært monotont system med komponentmengde, C der $C = \{1, \ldots, n\}$, og introduserer komponenttilstandsvariabelen x_i som betegner *tilstanden* til komponent i, for alle $i \in C$:

$$x_i = \begin{cases} 1 & \text{hvis komponent } i \text{ funksjonerer} \\ 0 & \text{ellers.} \end{cases}$$

Videre, innfører vi variabelen ϕ som representerer *tilstanden* til systemet:

$$\phi = \begin{cases} 1 & \text{hvis systemet funksjonerer} \\ 0 & \text{ellers.} \end{cases}$$

Variablene x_i, $i = 1, \ldots, n$ og ϕ sies å være *binære* siden de bare har to mulige verdier, 0 og 1.

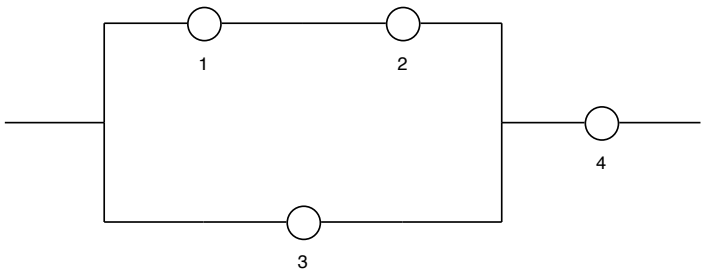
Tilstanden til systemet antas å være entydig bestemt av tilstandene til komponentene. Vi kan derfor uttrykke ϕ som:

$$\phi = \phi(\boldsymbol{x}) = \phi(x_1, x_2, \ldots, x_n)$$

Funksjonen $\phi : \{0,1\}^n \to \{0,1\}$ kalles *strukturfunksjonen* til systemet.

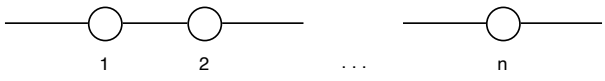
Definisjon 1.1 *Et binært monotont system er et ordnet par (C, ϕ), der $C = \{1, \ldots, n\}$ er komponentmengden og ϕ er strukturfunksjonen. Strukturfunksjonen ϕ er ikke-avtagende i hvert argument. Antall komponenter, n, kalles* ordenen *til systemet.*

I et *blokkdiagram* av et binært monotont system tegnes komponentene som sirkler som er forbundet med streker. Systemet funksjonerer hvis og bare hvis det er mulig å finne en vei gjennom diagrammet fra venstre mot høyre som kun passerer komponenter som funksjonerer. Et eksempel er vist i Figur **??**.

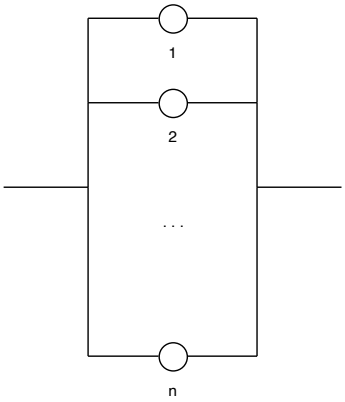


Figur 1.1: Blokkdiagram av et binært monotont system med 4 komponenter

Bemerk: For å kunne representere alle mulige binære monotone systemer, tillater vi at komponentene kan tegnes flere ulike steder i blokkdiagrammet. Diagrammet kan derfor ikke nødvendigvis fortolkes som en fysisk representasjon av systemet.



Figur 1.2: Blokkdiagram av et seriesystem.



Figur 1.3: Blokkdiagram av et parallellsystem.

1.1 Strukturfunksjonen til noen systemer

Av blokkdiagrammet vist i Figur 1.2 ser vi at dette seriesystemet funksjonerer hvis og bare hvis alle komponentene funksjonerer, dvs. hvis og bare hvis:

$$x_1 = 1 \text{ og } \cdots \text{ og } x_n = 1$$

Dette betyr at strukturfunksjonen til systemet kan uttrykkes som produktet av komponentenes tilstandsvariable:

$$\phi(\boldsymbol{x}) = x_1 \cdot x_2 \cdots x_n = \prod_{i=1}^{n} x_i$$

Alternativt kan strukturfunksjonen til seriesystemet skrives på følgende form:

$$\phi(\boldsymbol{x}) = \min\{x_1, \ldots, x_n\}.$$

Dette holder siden:

$$\min\{x_1, \ldots, x_n\} = 1 \;\Leftrightarrow\; x_1 = 1 \text{ og } \cdots \text{ og } x_n = 1$$

Før vi kan utlede strukturfunksjonen for et parallelsystem, trenger vi å innføre den såkalte *koprodukt*-operatoren. For enhver mengde av reelle tall $y_1, y_2, \ldots, y_n$ defineres *koprodukt*-operatoren, betegnet II, som følger:

$$y_1 \amalg y_2 = 1 - (1 - y_1)(1 - y_2),$$

$$\coprod_{i=1}^{n} y_i = 1 - \prod_{i=1}^{n} (1 - y_i).$$

Hvis $y_1, y_2, \ldots, y_n \in \{0, 1\}$, så kan koproduktet alternativt uttrykkes som:

$$y_1 \amalg y_2 = \max\{y_1, y_2\},$$

$$\coprod_{i=1}^{n} y_i = \max\{y_1, \ldots, y_n\}.$$

For å forlare dette nærmere, bemerker vi først at:

$$y_1 \amalg y_2 = 1 - (1 - y_1)(1 - y_2) = 0 \;\Leftrightarrow\; (1 - y_1)(1 - y_2) = 1$$

Herav ser vi at:

$$y_1 \amalg y_2 = 0 \;\Leftrightarrow\; y_1 = 0 \text{ og } y_2 = 0.$$

Dette er ekvivalent med at:

$$y_1 \amalg y_2 = 1 \;\Leftrightarrow\; y_1 = 1 \text{ eller } y_2 = 1.$$

Siden $y_1, y_2 \in \{0, 1\}$, vil:

$$y_1 = 1 \text{ eller } y_2 = 1 \;\Leftrightarrow\; \max\{y_1, y_2\} = 1$$

Dermed følger det at:

$$y_1 \amalg y_2 = \max\{y_1, y_2\}.$$

Tilsvarende har vi mer generelt at:

$$\coprod_{i=1}^{n} y_i = 1 - \prod_{i=1}^{n}(1 - y_i) = 0 \;\Leftrightarrow\; \prod_{i=1}^{n}(1 - y_i) = 1$$

Herav ser vi at:

$$\coprod_{i=1}^{n} y_i = 0 \;\Leftrightarrow\; y_1 = 0 \text{ og } \cdots \text{ og } y_n = 0.$$

Dette er ekvivalent med at:

$$\coprod_{i=1}^{n} y_i = 1 \;\Leftrightarrow\; y_1 = 1 \text{ eller } \cdots \text{ eller } y_n = 1.$$

Siden $y_1, \ldots, y_n \in \{0, 1\}$, vil:

$$y_1 = 1 \text{ eller } \cdots \text{ eller } y_n = 1 \;\Leftrightarrow\; \max\{y_1, \ldots, y_n\} = 1.$$

Dermed følger det at:

$$\coprod_{i=1}^{n} y_i = \max\{y_1, \ldots, y_n\}.$$

Vi kan nå utlede strukturfunksjonen for parallellsystemet illustrert i Figur 1.3. Av blokkdiagrammet ser vi at dette systemet funksjonerer hvis og bare hvis minst én av komponentene funksjonerer, dvs. hvis og bare hvis:

$$x_1 = 1 \textit{ eller } \cdots \textit{ eller } x_n = 1$$

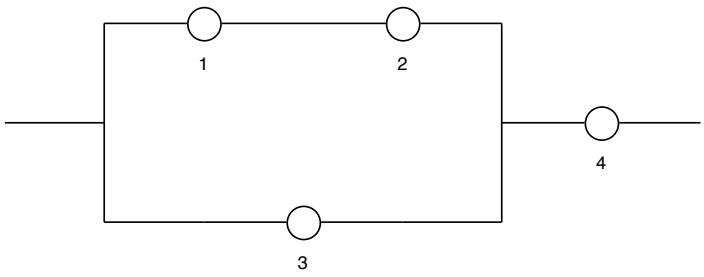
Herav følger det at strukturfunksjonen til systemet kan uttrykkes som koproduktet av komponentenes tilstandsvariable:

$$\phi(\boldsymbol{x}) = x_1 \amalg x_2 \cdots \amalg x_n = \coprod_{i=1}^{n} x_i$$

Alternativt kan strukturfunksjonen til parallellsystemet skrives på følgende form:

$$\phi(\boldsymbol{x}) = \max\{x_1, \ldots, x_n\}.$$

$$\max\{x_1, \ldots, x_n\} = 1 \;\Leftrightarrow\; x_1 = 1 \textit{ eller } \cdots \textit{ eller } x_n = 1$$



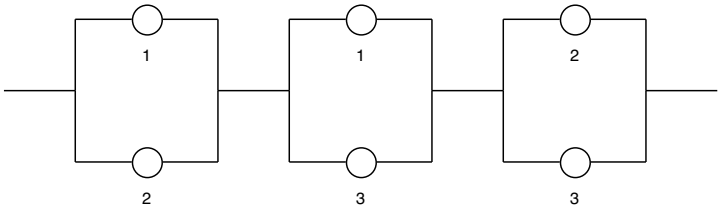
Figur 1.4: Blokkdiagram av et binært monotont system med 4 komponenter

Det neste systemet vi skal utlede et utrykk for strukturfunksjonen til, er systemet illustrert i Figur 1.4. Av blokkdiagrammet ser vi at komponent 1 og 2 er *i serie*. Disse er igjen *i parallell* med komponent 3. Til slutt er disse igjen *i serie* med komponent 4.

Ved å benytte dette, og kombinere produkt- og koprodukt-operatorene for henholdsvis serie- og parallellkoblinger, får vi at strukturfunksjonen for systemet blir:

$$\phi(\boldsymbol{x}) = [(x_1 \cdot x_2) \amalg x_3] \cdot x_4$$

Til sist i denne seksjonen skal vi se på 2-av-3-systemet vist i Figur 1.5. Dette systemet funksjonerer hvis og bare hvis to av de tre komponentene



Figur 1.5: Blokkdiagram av et 2-av-3 system.

funksjonerer. Av blokkdiagrammet ser vi at systemet består av tre parallellkoblinger: Komponent *1 og 2*, Komponent *1 og 3* og Komponent *2 og 3*. Disse tre parallellkoblingene er igjen koblet i serie.

Ved å benytte dette, og kombinere produkt- og koprodukt-operatorene for henholdsvis serie- og parallellkoblinger, får vi at strukturfunksjonen for systemet blir:

$$\phi(\boldsymbol{x}) = (x_1 \amalg x_2)(x_1 \amalg x_3)(x_2 \amalg x_3)$$

Binære monotone systemer

Et binært monotont system er et ordnet par (C, ϕ), der $C = \{1, \ldots, n\}$ er komponentmengden og ϕ er strukturfunksjonen. Strukturfunksjonen ϕ er ikke-avtagende i hvert argument. Antall komponenter, n, kalles *ordenen* til systemet.

La $y_1, y_2, \ldots, y_n \in \mathbb{R}$. *Koprodukt*-operatoren, betegnet $\amalg$, defineres da som følger:

$$y_1 \amalg y_2 = 1 - (1 - y_1)(1 - y_2),$$

$$\coprod_{i=1}^{n} y_i = 1 - \prod_{i=1}^{n} (1 - y_i).$$

Strukturfunksjonen til et seriesystem av n komponenter er:

$$\phi(\boldsymbol{x}) = x_1 \cdot x_2 \cdot \cdots \cdot x_n = \prod_{i=1}^{n} x_i = \min\{x_1, \ldots, x_n\}$$

Strukturfunksjonen til et parallellsystem av n komponenter er:

$$\phi(\boldsymbol{x}) = x_1 \amalg x_2 \cdots \amalg x_n = \coprod_{i=1}^{n} x_i = \max\{x_1, \ldots, x_n\}$$

2 Pålitelighet av binære monotone systemer

La (C, ϕ) være et binært monotont system. Vi innfører nå *stokastiske tilstandsvariable*, $X_1, \ldots, X_n$, og lar $\boldsymbol{X} = (X_1, \ldots, X_n)$. Vi lar så:

$$p_i = P(X_i = 1) = \text{Påliteligheten til komponent } i, \quad i \in C.$$

Siden tilstandsvariabelen X_i er *binær*, har vi for alle $i \in C$:

$$\mathbb{E}[X_i] = 0 \cdot P(X_i = 0) + 1 \cdot P(X_i = 1) = P(X_i = 1) = p_i$$

Dette betyr at *påliteligheten* til komponent i er lik *forventningsverdien* til komponentens tilstandsvariabel, X_i.

Tilsvarende innfører vi:

$$P(\phi(\boldsymbol{X}) = 1) = \text{Påliteligheten til systemet}$$

Siden tilstandsvariabelen til systemet ϕ er *binær*, har vi:

$$\mathbb{E}[\phi(\boldsymbol{X})] = 0 \cdot P(\phi(\boldsymbol{X}) = 0) + 1 \cdot P(\phi(\boldsymbol{X}) = 1) = P(\phi(\boldsymbol{X}) = 1) = h.$$

Dette betyr at påliteligheten til systemet er lik *forventningsverdien* til systemets tilstandsvariabel, $\phi = \phi(\boldsymbol{X})$.

Herav følger det at påliteligheten til systemet (i prinsippet) kan beregnes av følgende formel:

$$\mathbb{E}[\phi(\boldsymbol{X})] = \sum_{\boldsymbol{x} \in \{0,1\}^n} \phi(\boldsymbol{x}) P(\boldsymbol{X} = \boldsymbol{x}),$$

der $P(\boldsymbol{X} = \boldsymbol{x}) = P(X_1 = x_1 \cap \cdots \cap X_n = x_n)$ betegner simultanfordelingen til tilstandsvariablene.

Dersom komponentenes tilstandsvariable kan antas å være *uavhengige*, dvs. at simultanfordelingen for tilstandsvariablene $X_1, \ldots, X_n$ er lik produktet av marginalfordelingene for $X_1, \ldots, X_n$, får vi at:

$$P(\boldsymbol{X} = \boldsymbol{x}) = P(X_1 = x_1 \cap \cdots \cap X_n = x_n) = \prod_{i=1}^{n} P(X_i = x_i)$$

Vi innfører også vektoren av komponentpåliteligheter: $\boldsymbol{p} = (p_1, p_2, \ldots, p_n)$. For $i \in C$ har vi:

$$P(X_i = x_i) = \begin{cases} p_i & \text{if } x_i = 1, \\ 1 - p_i & \text{if } x_i = 0. \end{cases}$$

Siden x_i er enten 0 eller 1, kan $P(X_i = x_i)$ uttrykkes mer kompakt som følger:

$$P(X_i = x_i) = p_i^{x_i}(1 - p_i)^{1-x_i}.$$

At dette er riktig innsees ved å betrakte de to mulige tilfellene $x_i = 0$ og $x_i = 1$:

$$P(X_i = 1) = p_i^1(1 - p_i)^{1-1} = p_i$$
$$P(X_i = 0) = p_i^0(1 - p_i)^{1-0} = (1 - p_i)$$

Når komponentenes tilstandsvariable er *uavhengige*, så er, som nevnt, deres simultanfordeling lik produktet av marginalfordelingene. Dette betyr at simultanfordelingen kan skrives på følgende form:

$$P(\boldsymbol{X} = \boldsymbol{x}) = \prod_{i=1}^{n} P(X_i = x_i) = \prod_{i=1}^{n} p_i^{x_i}(1 - p_i)^{1-x_i}.$$

Setter vi denne simultanfordelingen inn i uttrykket for systempåliteligheten, får vi:

$$h(\boldsymbol{p}) = \mathbb{E}[\phi(\boldsymbol{X})] = \sum_{\boldsymbol{x} \in \{0,1\}^n} \phi(\boldsymbol{x}) \prod_{i=1}^{n} p_i^{x_i}(1 - p_i)^{1-x_i},$$

der funksjonen $h(\boldsymbol{p})$ kalles *pålitelighetsfunksjonen* til systemet.

Ulempen med formelen over, er at summen inneholder 2^n ledd. Hvis n er stor, vil antall ledd i summen fort bli så stort at beregning av systemets pålitelighet i praksis blir uhåndterlig. Heldigvis finnes det ofte langt mer effektive formler for $h(\boldsymbol{p})$. Vi skal nå vise noen eksempler på dette. I alle eksemplene antar vi at komponentenes tilstandsvariable er uavhengige.

Vi starter med å se på et *seriesystem* av orden n. Pålitelighetsfunksjonen til dette system er gitt ved:

$$h(\boldsymbol{p}) = \mathbb{E}[\phi(\boldsymbol{X})] = \mathbb{E}[\prod_{i=1}^{n} X_i] = \prod_{i=1}^{n} \mathbb{E}[X_i] = \prod_{i=1}^{n} p_i,$$

der den *tredje likheten* følger siden $X_1, X_2, \ldots, X_n$ er uavhengige.

Vi ser så på et *parallellsystem* av orden n. Pålitelighetsfunksjonen til dette system er gitt ved:

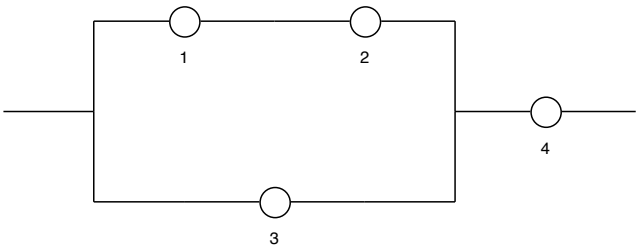
$$h(\boldsymbol{p}) = \mathbb{E}[\phi(\boldsymbol{X})] = \mathbb{E}[\coprod_{i=1}^{n} X_i] = \mathbb{E}[1 - \prod_{i=1}^{n}(1 - X_i)]$$
$$= 1 - \prod_{i=1}^{n}(1 - E[X_i]) = \coprod_{i=1}^{n} \mathbb{E}[X_i] = \coprod_{i=1}^{n} p_i,$$

der den *fjerde likheten* følger siden $X_1, X_2, \ldots, X_n$ er uavhengige.

Bemerk: Vi ser at *koproduktoperatoren* har følgende egenskap:

$$E[Z_1 \amalg \cdots \amalg Z_n] = E[Z_1] \amalg \cdots \amalg E[Z_n]$$

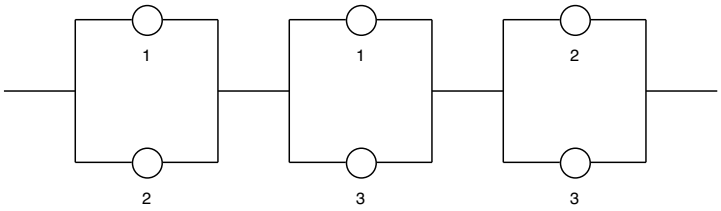
for *uavhengige* variabler $Z_1, \ldots, Z_n$. Den samme egenskapen gjelder naturligvis også for produktoperatoren.



Figur 2.1: Blokkdiagram av et binært monotont system med 4 komponenter

Det neste systemet er illustrert i blokkdiagrammet vist i Figur 2.1. For dette systemet er pålitelighetsfunksjonen gitt ved:

$$\begin{aligned} h(\boldsymbol{p}) &= \mathbb{E}[\phi(\boldsymbol{X})] = \mathbb{E}[[(X_1 \cdot X_2) \amalg X_3] \cdot X_4] \\ &= \mathbb{E}[(X_1 \cdot X_2) \amalg X_3] \cdot \mathbb{E}[X_4] \\ &= [\mathbb{E}[X_1 \cdot X_2] \amalg \mathbb{E}[X_3]] \cdot \mathbb{E}[X_4] \\ &= [(\mathbb{E}[X_1] \cdot \mathbb{E}[X_2]) \amalg \mathbb{E}[X_3]] \cdot \mathbb{E}[X_4] \\ &= [(p_1 \cdot p_2) \amalg p_3] \cdot p_4. \end{aligned}$$



Figur 2.2: Blokkdiagram av et 2-av-3 system.

Til sist i denne seksjonen skal vi se på 2-av-3-systemet vist i Figur 2.2. Vi har tidligere utledet at strukturfunksjonen for dette systemet er:

$$\phi(\boldsymbol{X}) = (X_1 \amalg X_2) \cdot (X_1 \amalg X_3) \cdot (X_2 \amalg X_3)$$

Som vanlig antar vi at tilstandsvariablene X_1, X_2, X_3 er uavhengige. På tross av dette har vi imidlertid at:

$$h(\boldsymbol{p}) = \mathbb{E}[\phi(\boldsymbol{X})] \quad \neq \quad (p_1 \amalg p_2) \cdot (p_1 \amalg p_3) \cdot (p_2 \amalg p_3)$$

Dette skyldes at koproduktene $(X_1 \amalg X_2)$, $(X_1 \amalg X_3)$ og $(X_2 \amalg X_3)$ *ikke* er uavhengige av hverandre siden samme komponent inngår ulike steder.

For å løse dette problemet kan vi finne et alternativt uttrykk for strukturfunksjonen ved å *løse opp koproduktene*, samt benytte at for binære variable X, så er $X^p = X$ for $p = 1, 2, 3, \ldots$:

$$\begin{aligned}
\phi(\boldsymbol{X}) &= (X_1 \amalg X_2) \cdot (X_1 \amalg X_3) \cdot (X_2 \amalg X_3) \\
&= (1 - (1 - X_1)(1 - X_2)) \cdot (1 - (1 - X_1)(1 - X_3)) \\
&\quad \cdot (1 - (1 - X_2)(1 - X_3)) \\
&= (X_1 + X_2 - X_1X_2) \cdot (X_1 + X_3 - X_1X_3) \cdot (X_2 + X_3 - X_2X_3) \\
&= 27 \text{ ledd} \qquad \textit{etter utmultiplisering} \\
&= X_1X_2 + X_1X_3 + X_2X_3 - 2X_1X_2X_3 \qquad \textit{etter sammentrekning}
\end{aligned}$$

Denne metoden kalles gjerne for *utmultipliseringsmetoden*. Ved å benytte dette uttrykket for ϕ blir det lett å finne pålitelighetsfunksjonen til systemet:

$$\begin{aligned}
h(\boldsymbol{p}) &= \mathbb{E}[\phi(\boldsymbol{X})] = \mathbb{E}[X_1X_2] + \mathbb{E}[X_1X_3] + \mathbb{E}[X_2X_3] - 2\mathbb{E}[X_1X_2X_3] \\
&= p_1p_2 + p_1p_3 + p_2p_3 - 2p_1p_2p_3
\end{aligned}$$

Anta spesielt at $p_1 = p_2 = p_3 = \frac{1}{2}$. Da blir:

$$h(\boldsymbol{p}) = 3 \cdot \tfrac{1}{2} \cdot \tfrac{1}{2} - 2 \cdot \tfrac{1}{2} \cdot \tfrac{1}{2} \cdot \tfrac{1}{2} = \tfrac{1}{2}$$

Vi kan sammenligne dette med resultatet vi hadde fått dersom vi *feilaktig* hadde regnet som om:

$$h(\boldsymbol{p}) = (p_1 \amalg p_2) \cdot (p_1 \amalg p_3) \cdot (p_2 \amalg p_3)$$

Innsatt at $p_1 = p_2 = p_3 = \frac{1}{2}$ blir:

$$(p_i \amalg p_j) = 1 - (1 - p_i)(1 - p_j) = 1 - \tfrac{1}{2} \cdot \tfrac{1}{2} = \tfrac{3}{4} \quad i \neq j$$

Totalt ville vi da ha endt opp med å *feilaktig* konkludere med at:

$$h(\boldsymbol{p}) = \left(\tfrac{3}{4}\right)^3 = \tfrac{27}{64} = \textit{0.421875}$$

Pålitelighet av binære monotone systemer

La (C, ϕ) være et binært monotont system av orden n, og anta at komponentenes tilstandsvariable, $X_1, \ldots, X_n$ er uavhengige med $P(X_i = 1) = p_i$ for alle $i \in C$. Vi innfører også $\boldsymbol{p} = (p_1, \ldots, p_n)$ og lar $h(\boldsymbol{p}) = P(\phi(\boldsymbol{X}) = 1)$ betegne pålitelighetsfunksjonen gitt ved:

$$h(\boldsymbol{p}) = \mathbb{E}[\phi(\boldsymbol{X})] = \sum_{\boldsymbol{x} \in \{0,1\}^n} \phi(\boldsymbol{x}) \prod_{i=1}^{n} p_i^{x_i} (1 - p_i)^{1 - x_i}$$

Hvis (C, ϕ) er et seriesystem, så er $h(\boldsymbol{p})$ gitt ved:

$$h(\boldsymbol{p}) = p_1 \cdot p_2 \cdot \cdots \cdot p_n = \prod_{i=1}^{n} p_i$$

Hvis (C, ϕ) er et parallellsystem, så er $h(\boldsymbol{p})$ gitt ved:

$$h(\boldsymbol{p}) = p_1 \amalg p_2 \cdots \amalg p_n = \coprod_{i=1}^{n} p_i$$

3 Pålitelighet av k-av-n-systemer

Et *k-av-n system* er et binært monotont system (C, ϕ) der $C = \{1, \ldots, n\}$ og som funksjonerer hvis og bare hvis minst k av de n komponentene funksjonerer. Strukturfunksjonen, ϕ, kan skrives veldig effektivt på følgende form:

$$\phi(\boldsymbol{x}) = \begin{cases} 1 & \text{hvis } \sum_{i=1}^{n} x_i \geq k \\ 0 & \text{ellers.} \end{cases}$$

For å beregne *påliteligheten* til et k-av-n system, innfører vi vektoren av de stokastiske tilstandsvariablene $\boldsymbol{X} = (X_1, \ldots, X_n)$. Det skal også vise seg at det er lurt å innføre den stokastiske variablen S gitt ved:

$$S = S(\boldsymbol{X}) = \sum_{i=1}^{n} X_i.$$

Systemet funksjonerer hvis og bare hvis $S \geq k$. Pålitelighetsfunksjonen til systemet kan derfor skrives som:

$$h(\boldsymbol{p}) = P(\phi(\boldsymbol{X}) = 1) = P(S \geq k) = \sum_{s=k}^{n} P(S = s).$$

Dersom komponentenes tilstandsvariable er *uavhengige*, og alle komponentene har *samme pålitelighet*, dvs.: $p_1 = \cdots = p_n = p$, så blir S *binomisk fordelt*. Sannsynlighetsfordelingen for S blir således gitt ved:

$$P(S = s) = \binom{n}{s} p^s (1-p)^{n-s}, \quad s = 0, 1, \ldots, n.$$

Systemets pålitelighetsfunksjon kan dermed enkelt finnes som følger:

$$h(\boldsymbol{p}) = P(S \geq k) = \sum_{s=k}^{n} \binom{n}{s} p^s (1-p)^{n-s}$$

Eksempel 3.1 *La (C, ϕ) være et 2-av-3 system der komponentenes tilstandsvariable er uavhengige, og der $p_1 = p_2 = p_3 = p$. Da har vi:*

$$P(S = 2) = \binom{3}{2} p^2 (1-p)^1 = 3p^2(1-p)$$

$$P(S = 3) = \binom{3}{3} p^3 (1-p)^0 = p^3.$$

Systemets pålitelighetsfunksjon blir dermed:

$$h(\boldsymbol{p}) = P(S \geq 2) = 3p^2(1-p) + p^3 = 3p^2 - 2p^3.$$

Eksempel 3.2 *La (C, ϕ) være et 3-av-4 system der komponentenes tilstandsvariable er uavhengige, og der $p_1 = p_2 = p_3 = p_4 = p$. Da har vi:*

$$P(S = 3) = \binom{4}{3} p^3 (1-p)^1 = 4p^3(1-p)$$

$$P(S = 4) = \binom{4}{4} p^4 (1-p)^0 = p^4.$$

Systemets pålitelighetsfunksjon blir dermed:

$$h(\boldsymbol{p}) = P(S \geq 3) = 4p^3(1-p) + p^4 = 4p^3 - 3p^4.$$

Hvis komponentene i et k-av-n *ikke* har samme pålitelighet, er det litt vanskeligere å finne pålitelighetsfunksjonen til systemet. Det finnes en svært effektiv algoritme som håndterer denne situasjonen også, men den skal vi ikke gå inn på her. I stedet ser vi på et par utvalgte eksempler.

Eksempel 3.3 *La (C, ϕ) være et 2-av-3 system der komponentenes tilstandsvariable er uavhengige, og der komponentenes påliteligheter er p_1, p_2, p_3. Vi har da:*

$$P(S = 2) = p_1 p_2 (1-p_3) + p_1 (1-p_2) p_3 + (1-p_1) p_2 p_3$$

$$P(S = 3) = p_1 p_2 p_3$$

Systemets pålitelighetsfunksjon blir dermed:

$$\begin{aligned} h(\boldsymbol{p}) &= p_1p_2(1-p_3) + p_1(1-p_2)p_3 + (1-p_1)p_2p_3 + p_1p_2p_3 \\ &= p_1p_2 + p_1p_3 + p_2p_3 - 2p_1p_2p_3. \end{aligned}$$

Bemerk: Vi ser at sammenlignet med *utmultipliseringsmetoden* som vi benyttet tidligere for dette systemet, så er denne metoden veldig mye raskere. Dette er typisk innen pålitelighetsteori. Valg av metode kan ofte avgjøre om et problem kan løses innen rimelig tid, eller om det i praksis ikke lar seg løse.

Eksempel 3.4 *La (C, ϕ) være et 3-av-4 system der komponentenes tilstandsvariable er uavhengige, og der komponentenes påliteligheter er p_1, p_2, p_3, p_4. Vi har da:*

$$\begin{aligned} P(S=3) &= p_1p_2p_3(1-p_4) + p_1p_2(1-p_3)p_4 \\ &\quad + p_1(1-p_2)p_3p_4 + (1-p_1)p_2p_3p_4 \\ P(S=4) &= p_1p_2p_3p_4 \end{aligned}$$

Systemets pålitelighetsfunksjon blir dermed:

$$\begin{aligned} h(\boldsymbol{p}) &= p_1p_2p_3(1-p_4) + p_1p_2(1-p_3)p_4 \\ &\quad + p_1(1-p_2)p_3p_4 + (1-p_1)p_2p_3p_4 \\ &\quad + p_1p_2p_3p_4 \\ &= p_1p_2p_3 + p_1p_2p_4 + p_1p_3p_4 + p_2p_3p_4 - 3p_1p_2p_3p_4 \end{aligned}$$

3.1 Monte Carlo simulering av k-av-n-systemer

I stedet for å beregne påliteligheten til et k-av-n-system analytisk, er det også mulig å estimere påliteligheten ved hjelp av Monte Carlo simulering. For å beskrive hvordan dette gjøres, betrakter vi et k-av-n-system (C, ϕ), der $C = \{1, \ldots, n\}$, og der strukturfunksjonen ϕ som før er gitt ved:

$$\phi(\boldsymbol{x}) = \begin{cases} 1 & \text{hvis } \sum_{i=1}^{n} x_i \geq k \\ 0 & \text{ellers.} \end{cases}$$

Videre lar vi $\boldsymbol{X} = (X_1, \ldots, X_n)$ være vektoren av stokastiske tilstandsvariable, der $P(X_i = 1) = p_i$, for alle $i \in C$. Vi antar også at tilstandsvariablene er uavhengige.

Ved å benytte Monte Carlo simulering genererer vi N observasjonssett:

$$\boldsymbol{X}_j = (X_{1j}, \ldots, X_{nj}), \quad j = 1, \ldots, N,$$

der X_{ij} er trukket fra fordelingen til X_i, dvs. $P(X_{ij} = 1) = p_i$, $i = 1, \ldots, n$. For hvert sample beregnes så den resulterende verdien av strukturfunksjonen, $\phi(\boldsymbol{X}_j)$. Basert på disse verdiene estimeres systemets pålitelighet ved:

$$\widehat{\mathbb{E}[\phi(\boldsymbol{X})]} = \frac{1}{N} \sum_{j=1}^{N} \phi(\boldsymbol{X}_j)$$

Vi ser at:

$$\mathbb{E}[\frac{1}{N} \sum_{j=1}^{N} \phi(\boldsymbol{X}_j)] = \frac{1}{N} \sum_{j=1}^{N} \mathbb{E}[\phi(\boldsymbol{X}_j)] = \frac{1}{N} \sum_{j=1}^{N} h(\boldsymbol{p}) = h(\boldsymbol{p})$$

Dette viser at estimatoren $\widehat{\mathbb{E}[\phi(\boldsymbol{X})]}$ er *forventningsrett*.

For å se hvordan estimatet for påliteligheten konvergerer, er det av interesse å plotte en *konvergenskurve* for estimatet. Dette gjøres ved å la:

$$R_m = \frac{1}{m} \sum_{j=1}^{m} \phi(\boldsymbol{X}_j), \quad m = 1, \ldots, N.$$

Her er R_m et estimat for systemets pålitelighet basert på de m første simuleringene. Ved å plotte m mot R_m får man frem en kurve som viser hvordan pålitelighetsestimatet stabiliserer seg gjennom simuleringen.

Et python-script for å utføre Monte Carlo simuleringen er gjengitt under. I dette scriptet estimeres påliteligheten til et 7-av-10-system der komponentenes påliteligheter er gitt ved:

$$p_1 = 0.65, \;\; p_2 = 0.33, \;\; p_3 = 0.71, \;\; p_4 = 0.58, \;\; p_5 = 0.82,$$
$$p_6 = 0.62, \;\; p_7 = 0.77, \;\; p_8 = 0.68, \;\; p_9 = 0.43, \;\; p_{10} = 0.31$$

Disse tallene lagres i scriptet i listen `p`. Antall simuleringer, `N`, er satt til 100000.

For å få en tilfredsstillende konvergenskurve er det typisk ikke nødvendig å beregne R_m for alle m. I scriptet har vi valgt å begrense oss til å beregne R_m for hver hundrede simulering. Dette er spesifisert via variabelen `sampleInterval` som i scriptet er satt til 100.

For å kunne plotte konvergenskurven, lagres R_m-verdiene i tabellen `R` og de tilhørende verdiene av m i tabellen `M`. Disse tabellene opprettes slik i koden:

```
N = 100000
sampleInterval = 100
R = np.zeros(int(N / sampleInterval))
M = np.linspace(1, N, int(N / sampleInterval))
```

Estimering av pålitelighet av et k-av-n-system

```
############################################################
#    Estimer paaliteligheten til et k-av-n-system          #
#    ved Monte Carlo simulering.                           #
############################################################

import matplotlib.pyplot as plt
import numpy as np
from random import random

# Component reliabilities
p = [0.65, 0.33, 0.71, 0.58, 0.82, 0.62, 0.77, 0.68, 0.43, 0.31]

k = 6
n = len(p)

N = 100000
sampleInterval = 100
R = np.zeros(int(N / sampleInterval))
M = np.linspace(1, N, int(N / sampleInterval))

def randomBinary(pp):
    u = random()
    if u <= pp:
        return 1
    else:
        return 0

phi_sum = 0
jj = 0

for j in range(N):
    S = 0
    for i in range(n):
        S += randomBinary(p[i])
    if S >= k:
        phi_sum += 1
    if j % sampleInterval == 0:
        R[jj] = phi_sum / (j+1)
        jj += 1

reliability = phi_sum / N
print(f"Reliability: {reliability:.3f}")

# Plot convergence curve
fig = plt.figure(figsize = (7, 4))
plt.plot(M, R, label='h')
plt.xlabel('Iteration')
plt.ylabel('Reliability')
plt.title("Reliability convergence curve")
plt.show()
```

Trekning av de binære tilstandsvariablene gjøres i funksjonen:

```
def randomBinary(pp):
    u = random()
    if u <= pp:
        return 1
    else:
        return 0
```

Denne funksjonen virker ved at man først trekker et tilfeldig tall `u` mellom 0 og 1 ved bruk av funksjonen `u = random()`. Hvis tallet er mindre enn eller lik inputverdien `pp`, returneres verdien 1. I motsatt fall returneres verdien 0. Siden tallet `u` er et tilfeldig tall mellom 0 og 1, vil:

$$P(\text{Funksjonen returnerer verdien 1}) = P(\texttt{u} \leq \texttt{pp}) = \texttt{pp}$$

I scriptet kalles funksjonen `randomBinary(pp)` med komponentenes påliteligheter, $p_1, \ldots, p_{10}$ som input. Dette sikrer at verdien av tilstandsvariablene blir trukket slik at $P(X_i = 1) = p_i$, for alle $i \in C$.

Selve simuleringen gjøres i kodesegmentet:

```
for j in range(N):
    S = 0
    for i in range(n):
        S += randomBinary(p[i])
    if S >= k:
        phi_sum += 1
    if j % sampleInterval == 0:
        R[jj] = phi_sum / (j+1)
        jj += 1
```

I hver iterasjon blir variabelen `S` beregnet som summen av alle tilstandsvariablene. Hvis `S` er større enn eller lik `k`, så betyr det at systemet funksjonerte i denne iterasjonen. I motsatt fall feilet systemet i denne iterasjonen. Antall iterasjoner der systemet funksjonerte telles opp i variabelen `phi_sum`.

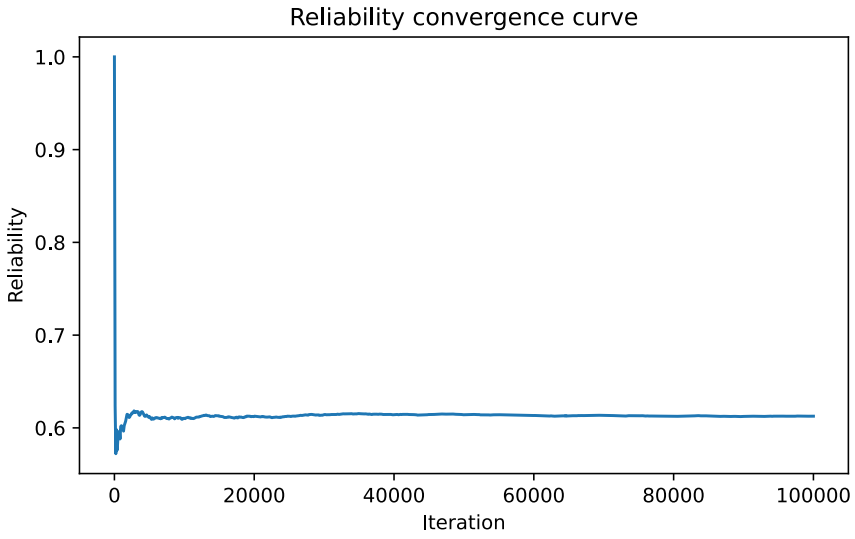
Når alle simuleringene er ferdige, estimeres systemets pålitelighet:

```
reliability = phi_sum / N
print(f"Reliability: {reliability:.3f}")
```

Til slutt plottes konvergenskurven. Et mulig resultat av å kjøre scriptet kan f.eks. være:

```
Reliability: 0.613
```

Den tilhørende konvergenskurven er vist i Figur 3.1. Av denne kurven ser vi at estimatet for systemets pålitelighet har stabilisert seg tydelig. Det kan forøvrig vises at den eksakte påliteligheten for dette systemet er 0.614, så estimatet var veldig nær det korrekte svaret.



Figur 3.1: Konvergenskurve for et 6-av-10 system.

Pålitelighet av k-av-n-systemer

Et *k-av-n system* er et binært monotont system (C, ϕ) der $C = \{1, \ldots, n\}$ som funksjonerer hvis og bare hvis minst k av de n komponentene funksjonerer. Strukturfunksjonen er gitt ved:

$$\phi(\boldsymbol{x}) = \begin{cases} 1 & \text{hvis } \sum_{i=1}^{n} x_i \geq k \\ 0 & \text{ellers.} \end{cases}$$

Hvis tilstandsvariablene er uavhengige, er pålitelighetsfunksjonen:

$$h(\boldsymbol{p}) = P(\phi(\boldsymbol{X}) = 1) = \sum_{s=k}^{n} P(S = s),$$

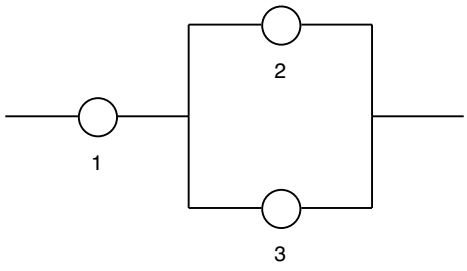
der $S = S(\boldsymbol{X}) = \sum_{i=1}^{n} X_i$. Hvis i tillegg $p_i = p$ for alle $i \in C$, er pålitelighetsfunksjonen:

$$h(\boldsymbol{p}) = \sum_{s=k}^{n} \binom{n}{s} p^s (1-p)^{n-s}$$

4 Oppgaver

I alle oppgavene under antar vi at komponentenes tilstandsvariable er uavhengige.

OPPGAVE 9. La (C, ϕ) være det binære monotone systemet vist i Figur 4.1.



Figur 4.1: Blokkdiagram av et binært monotont system.

a) Finn strukturfunksjonen til dette systemet, $\phi(\boldsymbol{x})$, der $\boldsymbol{x} = (x_1, x_2, x_3)$.

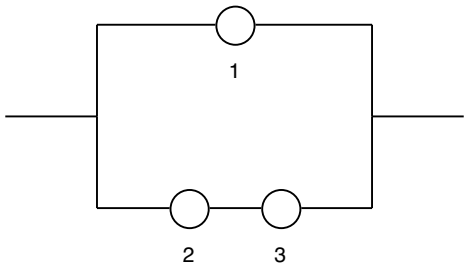
b) Finn pålitelighetsfunksjonen til dette systemet, $h(\boldsymbol{p})$, der $\boldsymbol{p} = (p_1, p_2, p_3)$.

Anta at:

$$p_1 = p_2 = p_3 = 0.5$$

c) Finn $P(\phi(\boldsymbol{X}) = 1)$ og $P(\phi(\boldsymbol{X}) = 0)$.

OPPGAVE 10. La (C, ϕ) være det binære monotone systemet vist i Figur 4.2.



Figur 4.2: Blokkdiagram av et binært monotont system.

a) Finn strukturfunksjonen til dette systemet, $\phi(\boldsymbol{x})$, der $\boldsymbol{x} = (x_1, x_2, x_3)$.

b) Finn pålitelighetsfunksjonen til dette systemet, $h(\boldsymbol{p})$, der $\boldsymbol{p} = (p_1, p_2, p_3)$.

Anta at:

$$p_1 = p_2 = p_3 = 0.5$$

c) Finn $P(\phi(\boldsymbol{X}) = 1)$ og $P(\phi(\boldsymbol{X}) = 0)$.

OPPGAVE 11. La (C, ϕ) være et seriesystem av 6 komponenter, der $P(X_i = 1) = p_i$, $i = 1, \ldots, 6$.

a) Finn pålitelighetsfunksjonen til systemet, $h(\boldsymbol{p})$.

Anta at komponentenes påliteligheter er:

$$p_1 = p_2 = 0.9, \quad p_3 = p_4 = 0.8, \quad p_5 = p_6 = 0.7$$

b) Beregn $P(\phi(\boldsymbol{X}) = 1)$ og $P(\phi(\boldsymbol{X}) = 0)$

OPPGAVE 12. La (C, ϕ) være et parallelsystem av 4 komponenter, der $P(X_i = 1) = p_i$, $i = 1, \ldots, 4$.

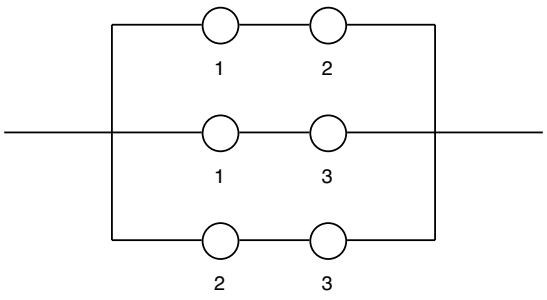
a) Finn pålitelighetsfunksjonen til systemet, $h(\boldsymbol{p})$.

Anta at komponentenes påliteligheter er:

$$p_1 = p_2 = p_3 = p_4 = 0.8$$

b) Beregn $P(\phi(\boldsymbol{X}) = 1)$ og $P(\phi(\boldsymbol{X}) = 0)$

OPPGAVE 13. La (C, ϕ) være det binære monotone systemet vist i Figur 4.3.



Figur 4.3: Alternativt blokkdiagram av et 2-av-3-system.

a) Forklar hvorfor dette systemet er et 2-av-3-system, og vis at:

$$\phi(\boldsymbol{x}) = (x_1 x_2) \amalg (x_1 x_3) \amalg (x_2 x_3)$$

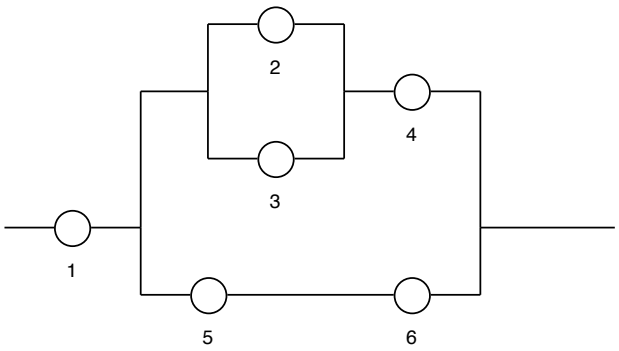
Anta at $p_1 = p_2 = p_3 = \frac{1}{2}$. Vi har tidligere vist at dette medfører at $h(\boldsymbol{p}) = \frac{1}{2}$.

b) Beregn i stedet verdien av:

$$(p_1 p_2) \amalg (p_1 p_3) \amalg (p_2 p_3)$$

og sammenlign resultatet med den korrekte pålideligheten.

OPPGAVE 14. La (C, ϕ) være det binære monotone systemet vist i Figur 4.4.



Figur 4.4: Blokkdiagram av et binært monotont system.

a) Finn strukturfunksjonen til dette systemet, $\phi(\boldsymbol{x})$, der $\boldsymbol{x} = (x_1, \ldots, x_6)$.

b) Finn pålitelighetsfunksjonen til dette systemet, $h(\boldsymbol{p})$, der $\boldsymbol{p} = (p_1, \ldots, p_6)$.

Anta at:

$$p_1 = 0.8, \quad p_2 = p_3 = 0.7, \quad p_4 = 0.6, \quad p_5 = p_6 = 0.5$$

c) Finn $P(\phi(\boldsymbol{x}) = 1)$ og $P(\phi(\boldsymbol{x}) = 0)$.

OPPGAVE 15. La (C, ϕ) være et 3-av-5-system der alle komponentene har samme pålitelighet p.

a) Finn pålitelighetsfunksjonen til dette systemet uttrykt ved p.

b) Beregn påliteligheten til systemet gitt at $p = 0.5$.

OPPGAVE 16. La (C, ϕ) være et 6-av-9-system der komponentenes pålitelighet er gitt ved:

$$p_1 = 0.55, \;\; p_2 = 0.49, \;\; p_3 = 0.82, \;\; p_4 = 0.72, \;\; p_5 = 0.86,$$
$$p_6 = 0.79, \;\; p_7 = 0.54, \;\; p_8 = 0.58, \;\; p_9 = 0.49$$

Ta utgangspunkt i python-scriptet *Estimering av pålitelighet av et k-av-n-system*, og modifiser dette slik at det estimerer påliteligheten for 6-av-9-systemet med de oppgitte komponentpålitelighetene, og plotter en konvergenskurve.