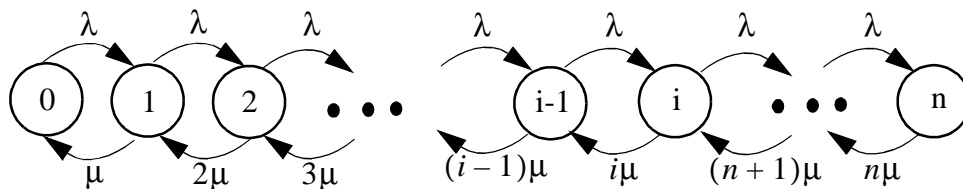


Oppgave 1 Ytelse

- a) Definér antall belagte talekanaler i som tilstandsvariable. Skissér et representativt tilstandsdiagram for antall belagte talekanaler. Hva kalles dette systemet?

Tilstandsdiagram, i =antall belagte talekanaler. Systemet er et Erlangs tapssystem.



Figur 1.1 Tilstandsdiagram for Erlangs tapssystem

- b) Hva er tidssperr? Vis hvordan du beregner tidssperr ved hjelp av tilstandssannsynlighetene for antall belagte kanaler. Hva er anropssperr? Vis hvordan de samme tilstandssannsynlighetene kan brukes for å beregne anropssperr.

Anropssperr = “den andelen anrop som avvises (møter fullt system)” Beregnes ved:

$$\lambda p(n) / \sum_{i=1}^n \lambda p(i) = \lambda p(n) / (\lambda \sum_{i=1}^n p(i)) = p(n).$$

Tidssperr = “den andelen av tiden hvor alle ressurser er belagt” = sannsynligheten for at Alle kanalene er belagt = $p(n)$. Dermed er tidssperr = anropssperr i dette tilfellet.

- c) La $\lambda = 36$ og $\mu = 2$. Bruk kurven til å bestemme antall organer n man minst må ha for at tidssperrssannsynligheten $E_n(A)$ skal være mindre enn 0.20.

- beregner først trafikken: $A = \lambda / \mu = 36 / 2 = 18$

- avles deretter på figuren $\Rightarrow n \geq 18$

- d) Jackson kønett, antakelser

- eksterne ankomster til hvert knutepunkt (node) fra Poissonprosess

- betjeningsstider er n.e.d. [NB! vi har ikke antatt at kandidatene vet om at dette for pakke-trafikk betyr at pakker som går fra node til node får ny størrelse hvert sted]

- stokastisk ruting

- køene har ∞ kapasitet

- e) Totalt ankomstintensitet

$$\Gamma_1 = \alpha_t + 0,3\alpha_d$$

$$\Gamma_2 = 0,7\alpha_d$$

$$\Gamma_3 = \Gamma_1 + 0,4\Gamma_2 = \alpha_t + 0,58\alpha_d$$

$$\Gamma_4 = 0,2\Gamma_3 + 0,6\Gamma_2 = 0,2\alpha_t + 0,536\alpha_d$$

(1.1)

Innsatt $\alpha_t = 0,5$ og $\alpha_d = 0,3$

$$\begin{aligned}
 \Gamma_1 &= 0,59 \\
 \Gamma_2 &= 0,21 \\
 \Gamma_3 &= 0,674 \\
 \Gamma_4 &= 0,2608
 \end{aligned}
 \tag{1.2}$$

f) Ventetid i hvert knutepunkt

- husker at for M/M/1 så er forventet ventetid, $E(W) = \lambda / ((\mu - \lambda)\mu)$

$$\begin{aligned}
 E(W_1) &= 1,439 \\
 E(W_2) &= 0,266 \\
 E(W_3) &= 2,067 \\
 E(W_4) &= 0,353
 \end{aligned}
 \tag{1.3}$$

g) Oppholdstid i systemet

- husker at for M/M/1 så er forventet oppholdstid, $E(S) = 1 / (\mu - \lambda)$

- finner først oppholdstiden i hver node

$$\begin{aligned}
 E(S_1) &= 2,439 \\
 E(S_2) &= 1,266 \\
 E(S_3) &= 3,067 \\
 E(S_4) &= 1,353
 \end{aligned}
 \tag{1.4}$$

- total oppholdstid \bar{T} finnes ved hjelp av Little.

$$(\alpha_t + \alpha_d) \bar{T} = \sum_{i=1}^4 \Gamma_i E(S_i)
 \tag{1.5}$$

- løses med hensyn på $\bar{T} = 5,156$

(a) Oppholdstid, telefonibruker

- sannsynlighetene for at telefonibruker besøker node i , $p_t(i)$:

$$\begin{aligned}
 p_t(1) &= 1 \\
 p_t(2) &= 0 \\
 p_t(3) &= 1 \\
 p_t(4) &= 0,2
 \end{aligned}
 \tag{1.6}$$

- oppholdstiden bli

$$E(S|t) = \sum_{i=1}^4 E(S_i)p_t(i) = 5,777 \quad (1.7)$$

(b) Oppholdstid, databruker

- sannsynlighetene for at databruker besøker node i , $p_d(i)$:

$$\begin{aligned} p_d(1) &= 0,3 \\ p_d(2) &= 0,7 \\ p_d(3) &= p_d(1) + 0,4p_d(2) \\ p_d(4) &= 0,6p_d(2) + 0,2p_d(3) \end{aligned} \quad (1.8)$$

- oppholdstiden bli

$$E(S|d) = \sum_{i=1}^4 E(S_i)p_d(i) = 4,122 \quad (1.9)$$

Kontroll av resultat, forventet oppholdstid for begge type brukere:

$$E(S) = E(S|d)p(d) + E(S|t)p(t) \quad (1.10)$$

hvor $p(d) = \alpha_d/(\alpha_d + \alpha_t)$ og $p(t) = \alpha_t/(\alpha_d + \alpha_t)$ og dermed

$$E(S) = E(S|d)(\alpha_d/(\alpha_d + \alpha_t)) + E(S|t)(\alpha_t/(\alpha_d + \alpha_t)) = 5,156 \quad (1.11)$$

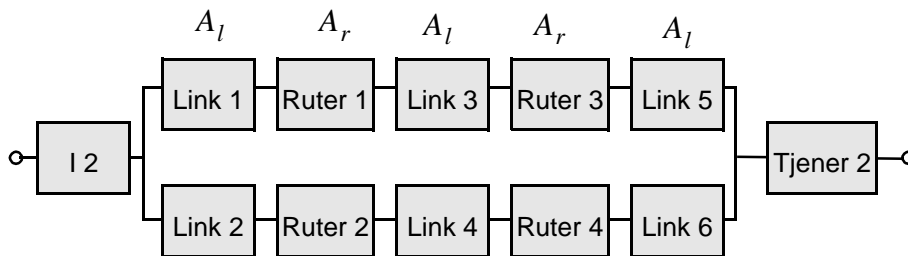
Oppgave 2 Pålitelighet

- a) Gitt: MTBF=75 timer

$$\text{MDT} = 0,95\left(\frac{3}{60}\right) + 0,05(3) \text{ timer} = 0,1975$$

$$A = 1 - U = 1 - \text{MDT}/\text{MTBF} = 1 - \frac{0,1975}{75} = 0,99736667$$

- b)



Bruken av pålitelighetsblokkskjema “krever” at alle de modellerte systemelementene feiler uavhengig av hverandre og blir reparert uavhengig av hverandre.

- c) Seriestrukturen i en parallellgren har tilgjengeligheten $A_{\text{gren}} = A_l^3 A_r^2$

Tilgjengeligheten til de to parallellgrenene er $1 - (1 - A_{\text{gren}})^2$ som danner en seriestruktur med de to gjenværende elementene. Følgelig blir tilgjengeligheten til tjenester fra tjener 2

$$A = A_{I2} A_{T2} (1 - (1 - A_l^3 A_r^2)^2)$$

- d) La Xi betene den Boolske som angir at element nr. i av type X fungerer, f.eks. L3 angir at link tre virker.

Ser at vi har fire minimale stisett definert av hvordan vi benytter de midterste lenkene i nettet:

I2 L1 R1 L3 R3 L5 T2

I2 L2 R2 L4 R4 L6 T2

I2 L1 R1 L8 R4 L6 T2

I2 L2 R2 L7 R3 L5 T2

Strukturfunksjonen er sann dersom det er minst en sti “gjennom systemet”, følgelig

$$\Phi = I2 L1 R1 L3 R3 L5 T2 + I2 L2 R2 L4 R4 L6 T2 + I2 L1 R1 L8 R4 L6 T2 + I2 L2 R2 L7 R3 L5 T2$$

- e) **Minimalt kuttsett:** En mengde av nettelement som medfører at systemet feiler når og bare når samtlige elementer i settet feiler, gitt at systemelementene som ikke er inkludert i settet er arbeidende.

Vi kan finne de minimale kuttsett ved å transformere Strukturfunksjonen fra din minimale produktform gitt i punkt d) til en minimal sum form, dvs. på formen $S = \{L1+L3\} \{L1+R3\} \{L3+R1\} \dots$

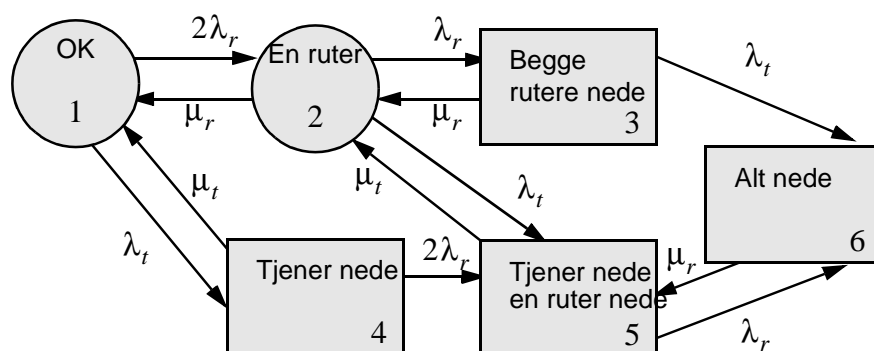
Samtlige kuttsett blir (kun fire av disse kreves for fullstendig besvarelse):

$\{\{L1,L2\}, \{L1,R2\}, \{L2,R1\}, \{L5,L6\}, \{L5,R4\}, \{L6,R3\}, \{R1,R2\}, \{R3,R4\}, \{L1,L4,L5\}, \{L1,L4,L7\}, \{L1,L4,R3\}, \{L1,L6,L7\}, \{L1,L7,R4\}, \{L3,L2,L6\}, \{L3,L$

$\{2, L8\}, \{L3, L2, R4\}, \{L3, L6, L7\}, \{L3, L6, R2\}, \{L3, L7, R4\}, \{L3, L8, R2\}, \{L3, R2, R4\}, \{L2, L5, L8\}, \{L2, L8, R3\}, \{L4, L5, L8\}, \{L4, L5, R1\}, \{L4, L7, R1\}, \{L4, L8, R3\}, \{L4, R1, R3\}, \{L5, L8, R2\}, \{L6, L7, R1\}, \{L7, R1, R4\}, \{L8, R3, R2\}, \{L3, L4, L7, L8\}, I2, T2\}$

- f) Vi har en begrenset reparasjonsressurs som introduserer avhengigheter mellom enhetene i domenet. Dette bryter forutsetningene for å kunne benytte blokkskjema.

Tilstandsdiagram nedenfor. Merk kun en reparasjons transisjon ut av hver tilstand.



I tilstand 5 repareres tjeneren først fordi det vil gi minst nedetid (kun en rep nødvendig for å få opp systemet.) Med dette valget vil reparasjons-sekvensene 6-5-2 og 6-3-2 være likeverdige.

- g) Stasjonærsannsynligheten for å være i tilstand i betegnes p_i

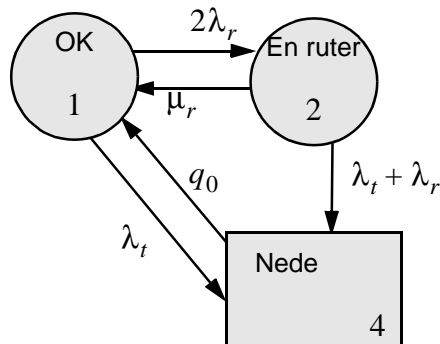
Den etterspurte tilgjengeligheten er: $A = p_1 + p_2$

Introduserer tilstandsvektoren $\underline{p} = \{p_i\}_{i=1, \dots, 6}$

Etablerer et likningsett på matrisefor som seks balanseligningene (inn og ut av alle tilstander) på matriseform. Erstatte en av disse med normeringsbetingelsen.

$$\underline{p} \cdot \begin{bmatrix} -(2\lambda_r + \lambda_t) & 2\lambda_r & 0 & \lambda_t & 0 & 1 \\ \mu_r & -(\mu_r + \lambda_r + \lambda_t) & \lambda_r & 0 & \lambda_t & 1 \\ 0 & \mu_r & -(\mu_r + \lambda_t) & 0 & 0 & 1 \\ \mu_t & 0 & 0 & -(2\lambda_r + \mu_t) & 2\lambda_r & 1 \\ 0 & \mu_t & 0 & 0 & -(\mu_t + \lambda_r) & 1 \\ 0 & 0 & 0 & 0 & \mu_r & 1 \end{bmatrix} = [0, 0, 0, 0, 0, 1]$$

- h) Når MTFF skal finnes kan vi slå sammen alle feiltilstandene til en, for hvilken feiltilstans som inntreffer ikke er interessant, og feilutvikling etter første feil har ikke betydning. Får da nedenstående diagram.



Introdusere den kunstige reparasjonsintensiteten q_0 fra feiltilstanden tilbake til alt OK tilstanden. Har nå et ekvivalent system hvor MUT er lik MTFF i det originale.

Gjennomføres prosedyren beskrevet under punkt g) finnes (for detaljer se kompendiet):

```
In[8]:= Transpose[M]//MatrixForm
```

Out[8]//MatrixForm=

$$\begin{pmatrix} !2 \text{ " }_r ! \text{ " }_t & 2 \text{ " }_r & \text{ " }_t \\ \#_r & ! \text{ " }_r ! \text{ " }_t ! \#_r & \text{ " }_r \$ \text{ " }_t \\ q_0 & 0 & !q_0 \end{pmatrix}$$

$$In[9] := !p_1, p_2, p_F" = \text{ProbStationary}\#M\$$$
[illegible]

$$MTFF = MUT = A * MTBF =$$

```
In[10]:= %1 - p_F & ' %p_F * q_0 & ' ' Simplify
```

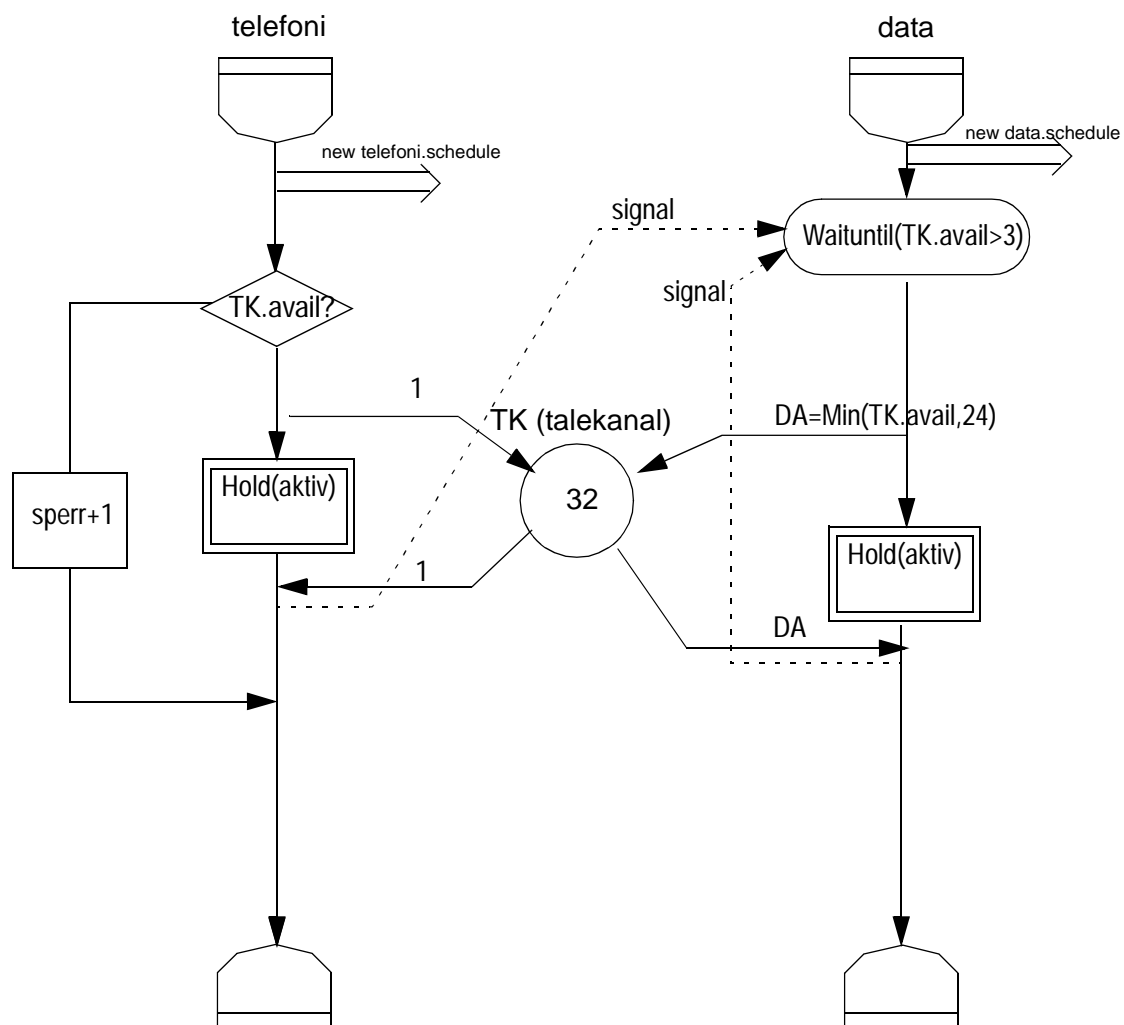
Out[10]=

Oppgave 3 Simulering

- a) Hva velger du som entiteter og som ressurser i modellen?

Link mellom node 1 og 3 er flaskehals, den eneste som deles av tele- og databrukere. Diskretiserer kapasiteten i 32 kanaler, modellerer denne ved hjelp av RES. Tele- og databrukere blir entiteter.

- b) Lag en enkel simuleringmodell av dette systemet hvor vi skal bestemme hvor lang tid det tar å laste ned en fil, og hvor mange samtaler som blir blokkert. Beskriv modellen ved hjelp av aktivitetsdiagrammer eller liknende.



Kildekode

```

BEGIN
  EXTERNAL CLASS demos="/home/fag/pyse/cim/demos.atr";
  demos BEGIN
    INTEGER MaxHop = 3;
    INTEGER tale=1;
    INTEGER data=2;
  
```

```

INTEGER Cap = 32;

REF(rdist) ARRAY active(tale:data);
REF(rdist) ARRAY interarr(tale:data);
REF(condq) kanalQ;
REF(res) kanal;
REF(count) block;
REF(count) calls;
REF(tally) kanalused;
integer i;

entity CLASS talegen;
BEGIN
    NEW talegen("TaleGen").schedule(interarr(tale).sample);
    IF kanal.avail > 0 THEN
        BEGIN
            kanal.acquire(1);
            Hold(active(tale).sample);
            kanal.release(1);
            kanalQ.signal;
        END
    ELSE
        block.update(1);
        calls.update(1);
    END *** talegen ***;

entity CLASS datagen;
BEGIN
    INTEGER numkanal;
    NEW datagen("DataGen").schedule(interarr(data).sample);
    kanalQ.waituntil(kanal.avail > Entier(0.10*Cap));
    BEGIN
        numkanal := Min(kanal.avail, Entier(0.75*Cap));
        kanal.acquire(numkanal);
        Hold(active(data).sample/numkanal);
        kanalused.update(numkanal);
        kanal.release(numkanal);
        kanalQ.signal;
    END;
END *** talegen ***;

! initiate the random generators;
interarr(tale) :- NEW NegExp("Tale, inter",8.0);
interarr(data) :- NEW NegExp("Data, inter",4.0);
active(tale)    :- NEW NegExp("Tale, aktiv",1.0/2);
active(data)    :- NEW NegExp("Data, aktiv",10);

! res = 32 channels;
kanal :- NEW res("Kanal", Cap);

! statistics;

```



```

kanalused :- NEW tally("Data, in use");
block :- NEW count("Blocked");
calls :- NEW count("Calls");

! conditional queue;
kanalQ :- NEW condq("Wait kanal");

! start the packet generators;
NEW talegen("TaleGen").schedule(0.0);
NEW datagen("DataGen").schedule(0.0);

! start simulation, hold for 2000 time units;
!trace;
hold(2000.0);
END *** demos ***;
END;

```

Resultater

D I S T R I B U T I O N S

TITLE	/	(RE)SET/	OBS/TYPE	/	A/	B/	SEED
Tale, inter		0.000	15999 NEGEXP		8.000		33427485
Data, inter		0.000	7967 NEGEXP		4.000		22276755
Tale, aktiv		0.000	15615 NEGEXP		0.500		46847980
Data, aktiv		0.000	7967 NEGEXP		10.000		43859043

C O U N T S

TITLE	/	(RE)SET/	OBS
Blocked		0.000	384
Calls		0.000	15978

T A L L I E S

TITLE	/	(RE)SET/	OBS/	AVERAGE/EST.ST.DV/	STD.ERR./	MINIMUM/	MAXIMUM
Data, in use		0.000	7967	16.234	3.8834.350&-002	4.000	24.000

R E S O U R C E S

TITLE	/	(RE)SET/	OBS/	LIM/	MIN/	NOW/	% USAGE/	AV. WAIT/QMAX
Kanal		0.000	23561	32	0	11	50.241	0.000 1

C O N D I T I O N Q U E U E S

```
TITLE      /   (RE)SET/   OBS/ QMAX/ QNOW/ Q AVERAGE/ZEROS/   AV. WAIT
Wait kanal      0.000   7967      4      0 2.755&-003   7726 6.915&-004
```

19 garbage collection(s) in 0.5 seconds.

- c) Hvilke primitiver og metoder i DEMOS vil du bruke? Hvordan får dere til å prioritere tale over data pakker?

Vil bruke RES for å modellere kanalene, CONDQ for å sjekke tilgjengelighet for data-brukere (som skal vente hvis ikke til strekkelig kapasitet er tilgjengelig)

Ikke-avbrytende prioritet ved tilordning av ressurser som står og venter kan gjøres ved å sette PRIORITY atributten i demos klassen til ulike verdier. {Men: denne ha liten effekt ettersom alle som venter har samme prioritet (data) mens høyprioritet (tale) vil aldri vente. En mekansime med bruk av INTERRUPT er derfor nødvendig for å få til avbrytende prioritet}.

- d) Beskriv spesielt hvordan du kan samle inn statistikk om tid for nedlasting og hvordan DEMOS kan støtte innsamling av data om fordeling av nedlastingstidene.

Legg inn HISTOGRAM i datagen entiteten.

- e) Estimer gjennomsnittelig filestørrelse \bar{X} og variansen S^2 ?

$$\bar{X} = 18000/100 = 180 \text{ [kbytes]} \quad (3.1)$$

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n X_i^2 - \frac{n}{n-1} \hat{X}^2 = 7676 \quad (3.2)$$

- f) Estimer hvor stor andel av kapasiteten som er utnyttet til overføring av filer.

$$(\sum X_i)/(TC) = \frac{18000 \cdot 8}{2 \cdot 3600 \cdot 2048} = 0.0098 \quad (3.3)$$

- g) Vis hvordan du på en enkel måte kan utvide modellen til å ta hensyn til at rutere, linjer og server kan feile?

Legg til ekstern entiet som “spiser alle kanalene”

```
entity CLASS kanaldown;
BEGIN
    priority:=down;
loop:
```

```
kanalup := TRUE;
Hold(interarr(down).sample);
kanalup := FALSE;
kanal.acquire(Cap);
Hold(active(down).sample);
kanal.release(Cap);
kanalQ.signal;
repeat;
END *** kanaldown ***;
```