

UNIVERSITÀ DI TRENTO



FIG:

the Finite Improbability Generator v1.3

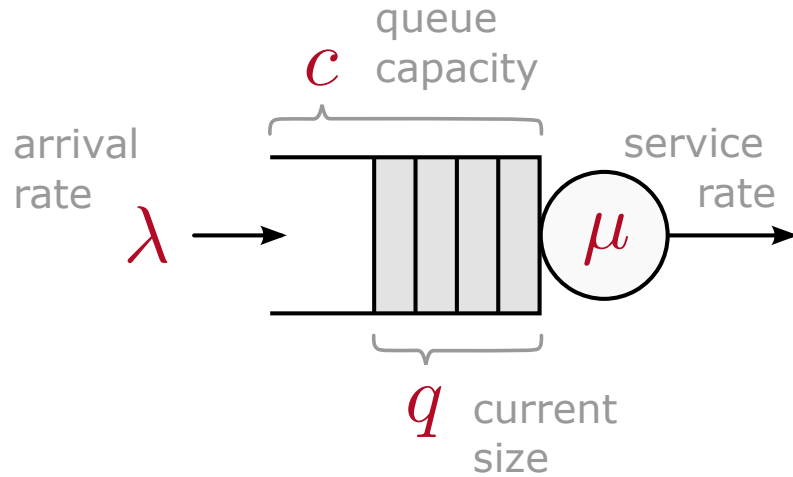
Carlos E. Budde

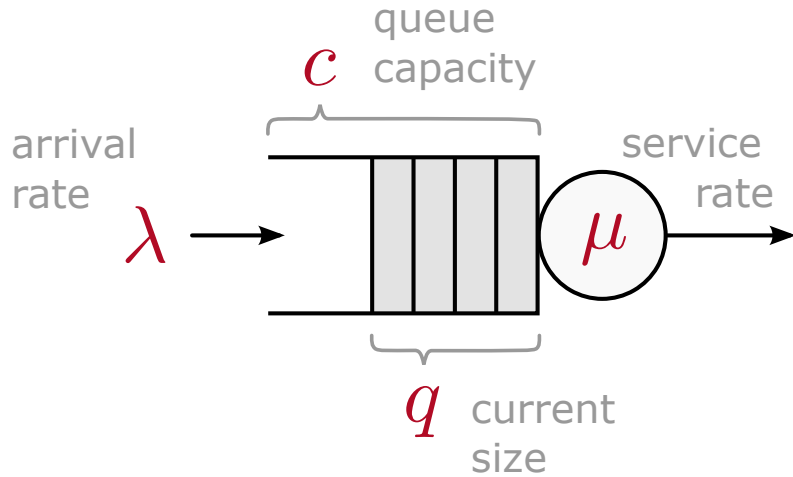
Security group @ Dipartimento di Ingegneria e Scienza dell'Informazione
Funded by EU grants 830929 ([CyberSec4Europe](#)) and 952647 ([AssureMOSS](#))

carlosesteban.budde@unitn.it

TOSME — Nov 12, 2021

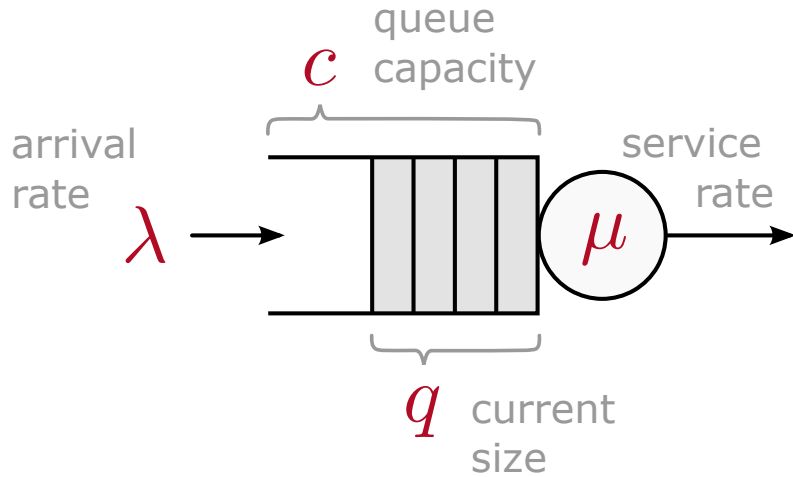
Discrete event simulation on stochastic systems





IOSA

```
1 const int c = 12;
2 const int lambda = 3;
3 const int mu = 8;
4
5 module Arrivals
6     clk0: clock; // External arrivals ~ Exponential(lambda)
7     [arr!] @ clk0 -> (clk0' = exponential(lambda));
8 endmodule
9
10 module Queue
11     q: [0..c] init 1;
12     clk: clock; // Queue service ~ Exponential(mu)
13     // Packet arrival
14     [arr?] q == 0 -> (q' = q+1) & (clk' = exponential(mu));
15     [arr?] q > 0 & q < c -> (q' = q+1);
16     [arr?] q == c -> ;
17     // Packet service
18     [ser!] q == 1 @ clk -> (q' = q-1);
19     [ser!] q > 1 @ clk -> (q' = q-1) & (clk' = exponential(mu));
20 endmodule
21
22 properties
23     S( q == c ) // steady-state
24     P( q > 0 U q == c ) // transient
25 endproperties
```



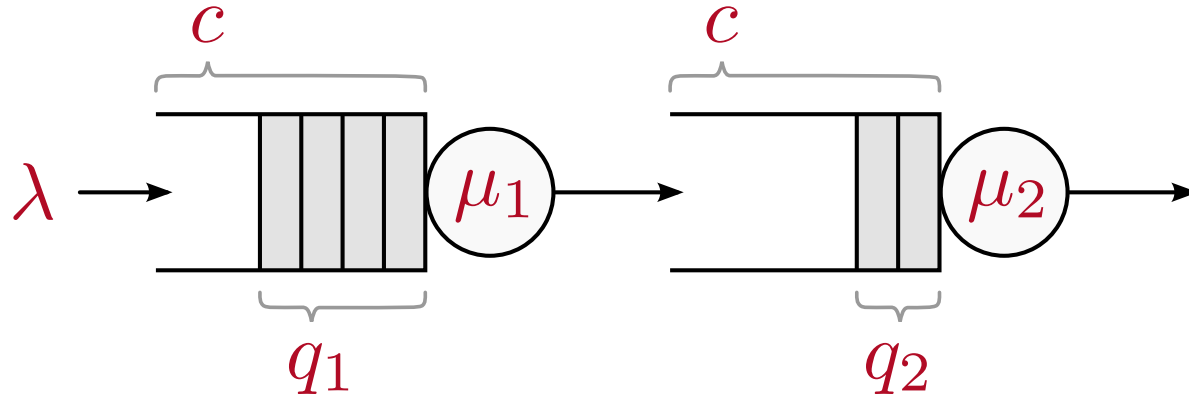
IOSA

```
1 const int c = 12;
2 const int lambda = 3;
3 const int mu = 8;
4
5 module Arrivals
6     clk0: clock; // External arrivals ~ Exponential(lambda)
7     [arr!] @ clk0 -> (clk0' = exponential(lambda));
8 endmodule
9
10 module Queue
11     q: [0..c] init 1;
12     clk: clock; // Queue service ~ Exponential(mu)
13     // Packet arrival
14     [arr?] q == 0 -> (q' = q+1) & (clk' = exponential(mu));
15     [arr?] q > 0 & q < c -> (q' = q+1);
16     [arr?] q == c -> ;
17     // Packet service
18     [ser!] q == 1 @ clk -> (q' = q-1);
19     [ser!] q > 1 @ clk -> (q' = q-1) & (clk' = exponential(mu));
20 endmodule
21
22 properties
23     S( q == c ) // steady-state
24     P( q > 0 U q == c ) // transient
25 endproperties
```

Transient: $Prob(\textit{condition} \cup \textit{goal})$

Steady-state: $Prob(\lim_{t \rightarrow \infty} \textit{goal}_t)$

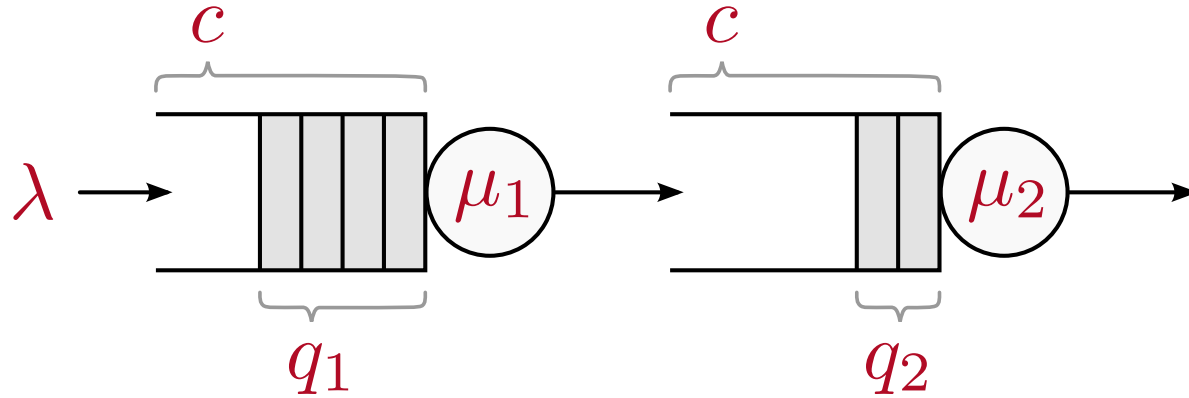
Tandem queue



Transient: $Prob(q_2 > 0 \cup q_2 \geq c)$

Steady-state: $Prob(\lim_{t \rightarrow \infty} q_2 \geq c)$

Tandem queue

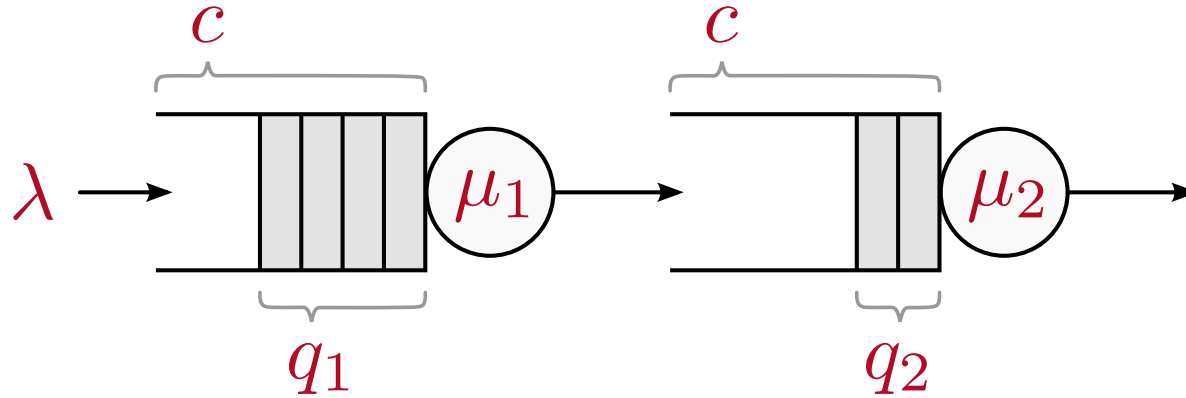


Transient: $Prob(q_2 > 0 \cup q_2 \geq c)$

Steady-state: $Prob(\lim_{t \rightarrow \infty} q_2 \geq c)$

$$CI_{\alpha} \doteq [l, u]_{\alpha} = \hat{p} \pm z_{\frac{\alpha}{2}} \sigma_{\hat{p}} \\ \cap \\ [0, 1]$$

Tandem queue



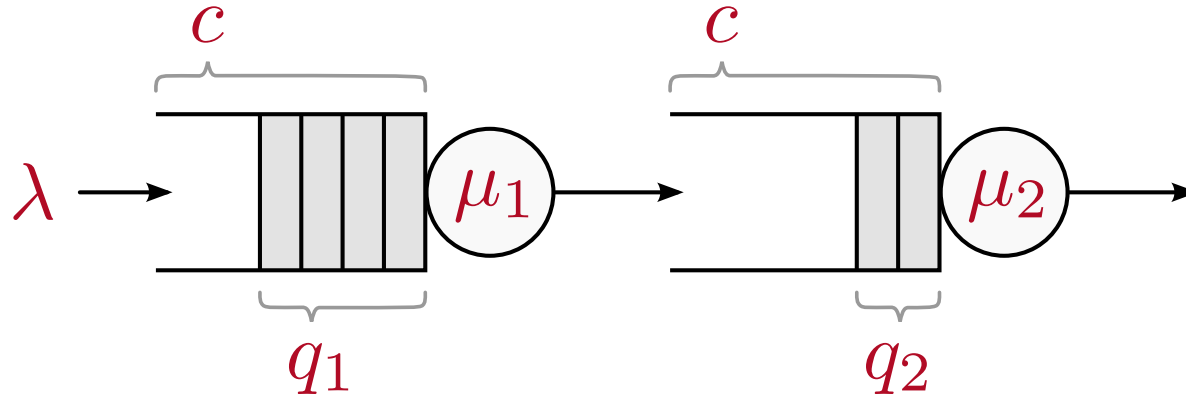
Transient: $Prob(q_2 > 0 \cup q_2 \geq c)$

Steady-state: $Prob(\lim_{t \rightarrow \infty} q_2 \geq c)$

$$CI_{\alpha} \doteq [l, u]_{\alpha} = \hat{p} \pm \underbrace{z_{\frac{\alpha}{2}} \sigma_{\hat{p}}}_{\text{precision (CI width)}}$$

\downarrow
confidence \cap [0, 1]

Tandem queue



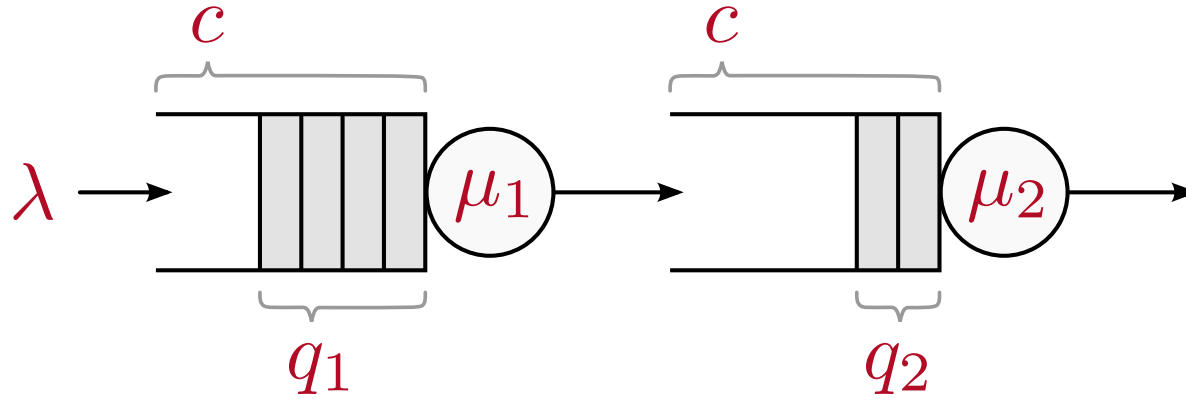
If $\mu_2 > \mu_1$
 then $q_2 \geq c$ is rare

Transient: $Prob(q_2 > 0 \cup q_2 \geq c)$

Steady-state: $Prob(\lim_{t \rightarrow \infty} q_2 \geq c)$

$$\begin{array}{ccc}
 CI_{\alpha} \doteq [l, u]_{\alpha} = \hat{p} \pm \underbrace{z_{\frac{\alpha}{2}} \sigma_{\hat{p}}}_{\text{precision (CI width)}} \\
 \downarrow \quad \quad \quad \cap \\
 \text{confidence} \quad [0, 1]
 \end{array}$$

Tandem queue



If $\mu_2 > \mu_1$
 then $q_2 \geq c$ is rare



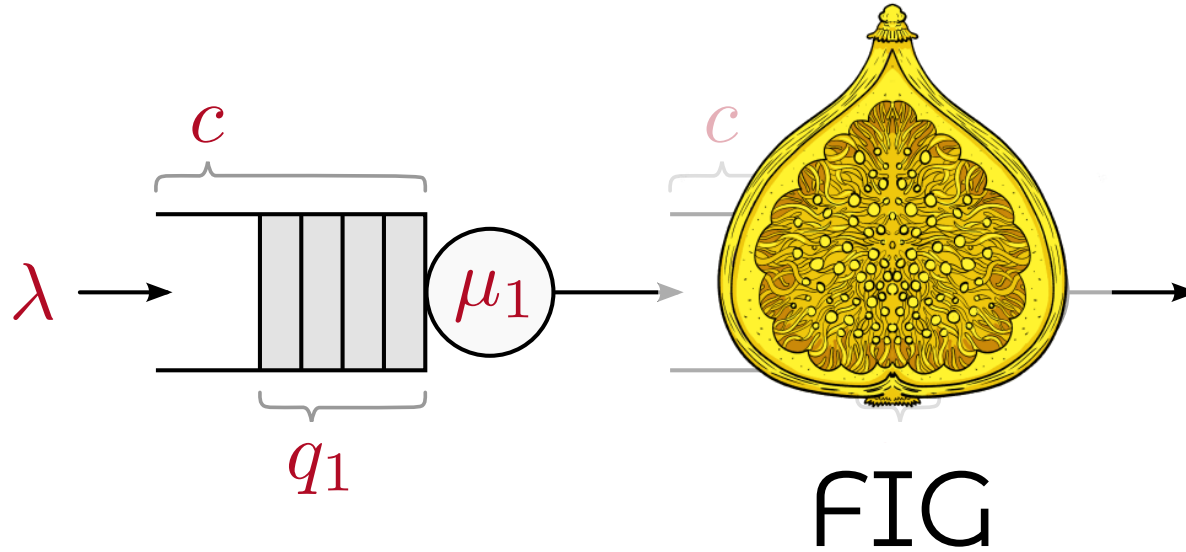
Transient: $Prob(q_2 > 0 \cup q_2 \geq c)$

Steady-state: $Prob(\lim_{t \rightarrow \infty} q_2 \geq c)$

$$CI_{\alpha} \doteq [l, u]_{\alpha} = \hat{p} \pm \underbrace{z_{\frac{\alpha}{2}} \sigma_{\hat{p}}}_{\text{precision (CI width)}}$$

\downarrow
 confidence \cap $[0, 1]$

Tandem queue



If $\mu_2 > \mu_1$
then $q_2 \geq c$ is rare



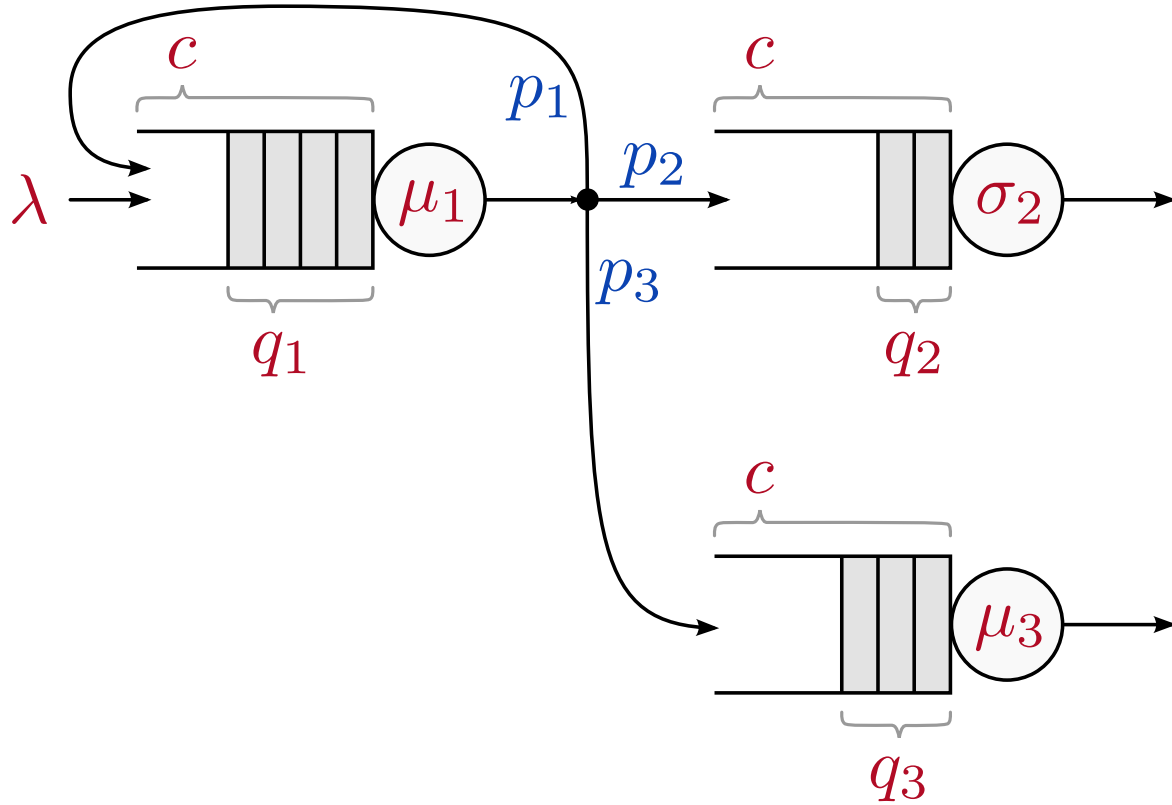
Transient: $Prob(q_2 > 0 \cup q_2 \geq c)$

Steady-state: $Prob(\lim_{t \rightarrow \infty} q_2 \geq c)$

$$CI_{\alpha} \doteq [l, u]_{\alpha} = \hat{p} \pm \underbrace{z_{\frac{\alpha}{2}} \sigma_{\hat{p}}}_{\text{precision (CI width)}}$$

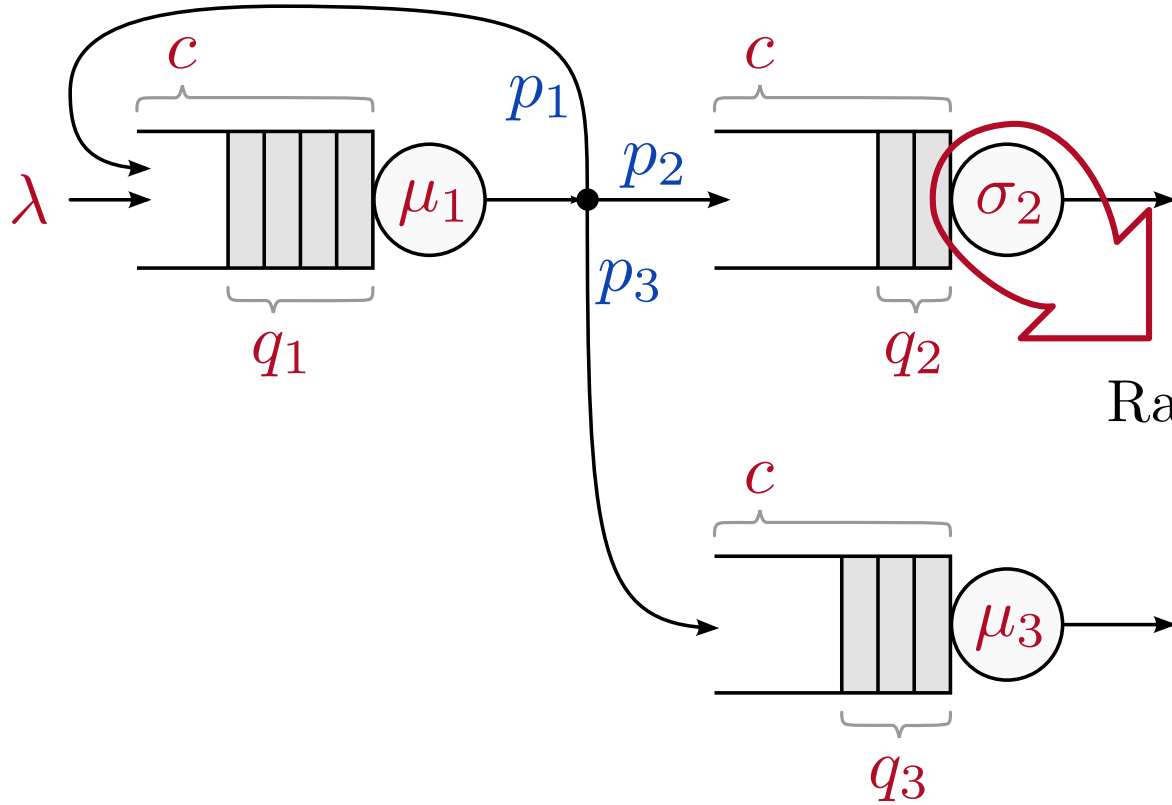
\downarrow confidence \cap $[0, 1]$

Forked-cycle tandem queue



$$S(q_2 \geq 20)$$

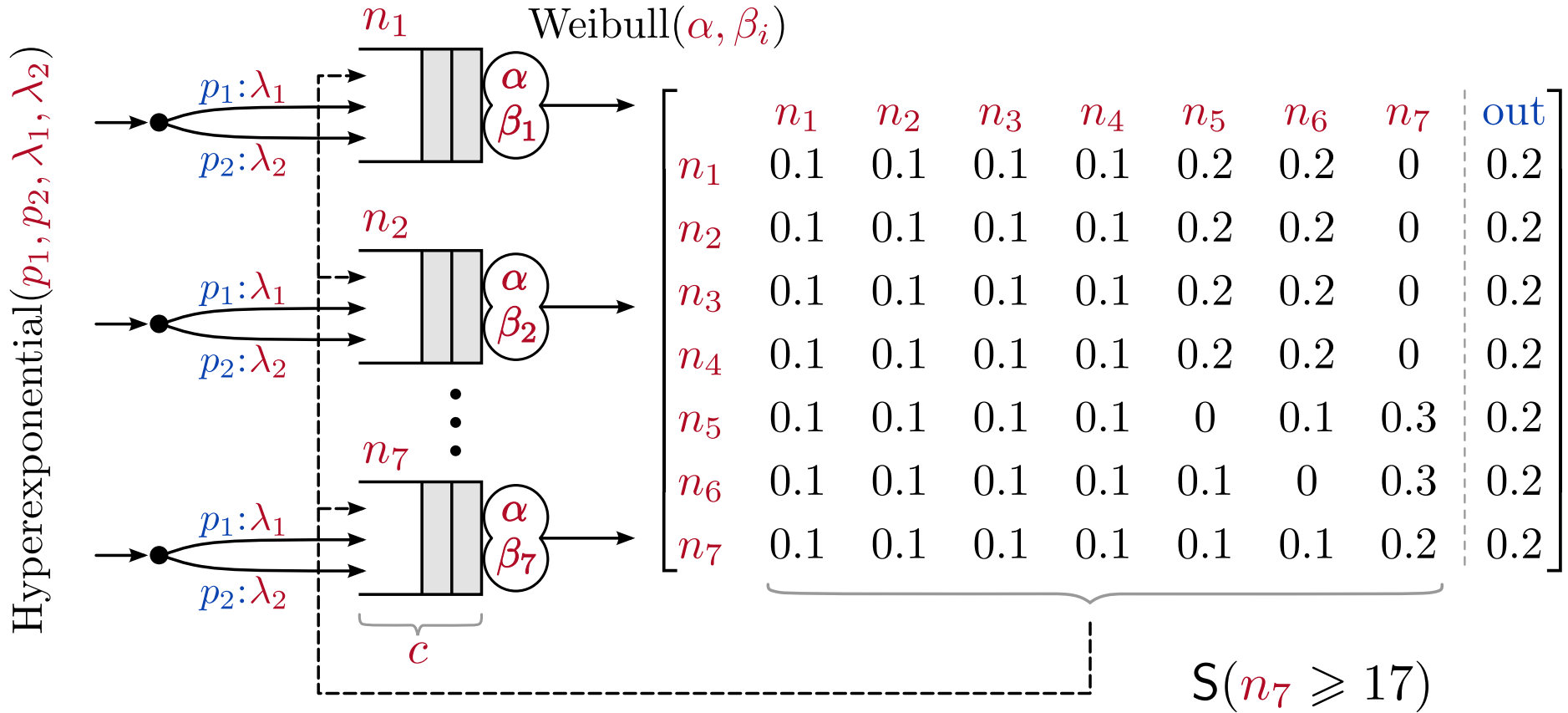
Forked-cycle tandem queue

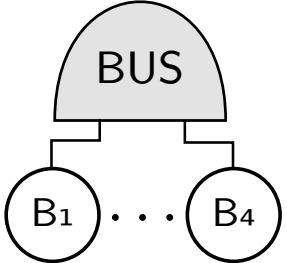
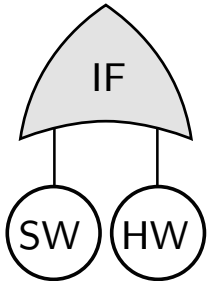


$$S(q_2 \geq 20)$$

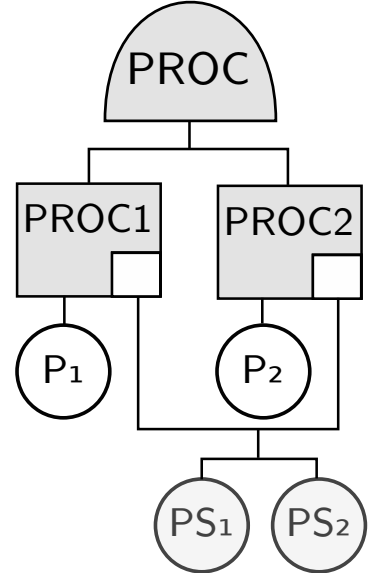
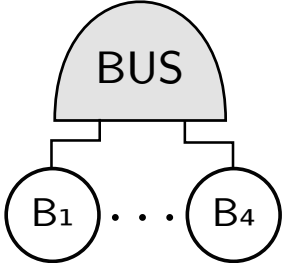
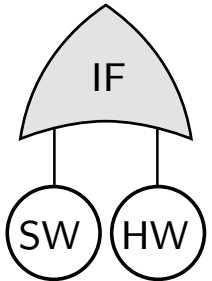
$$\text{Rayleigh}(\sigma_2) \sim \text{Weibull}(2, \sqrt{2} \sigma_2)$$

7-nodes non-Jackson queueing network

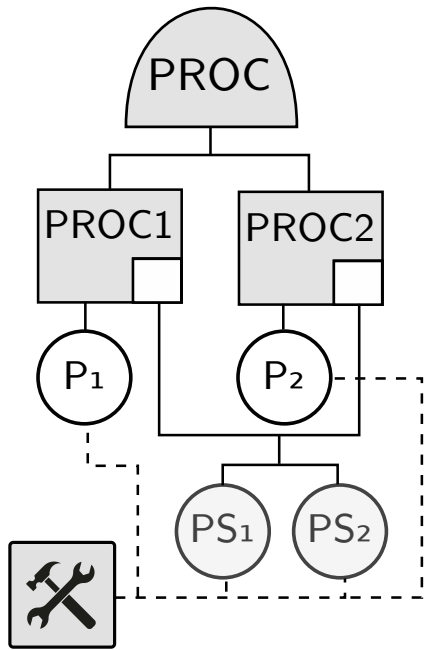
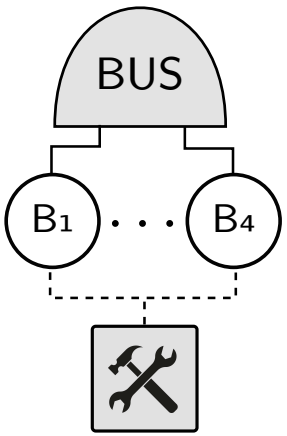
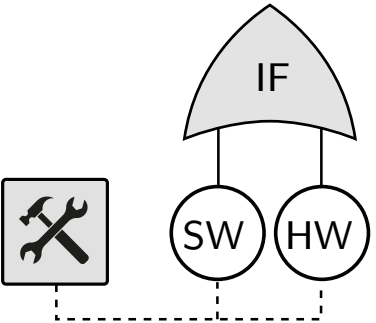




Fault Trees

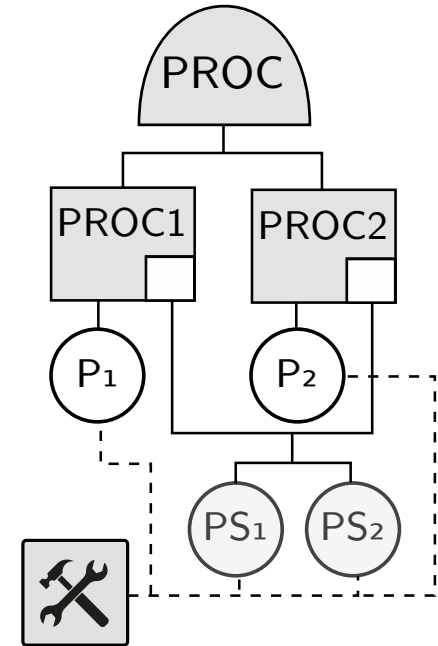
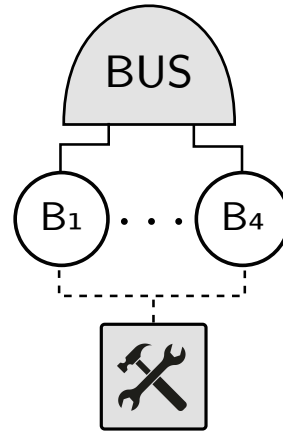
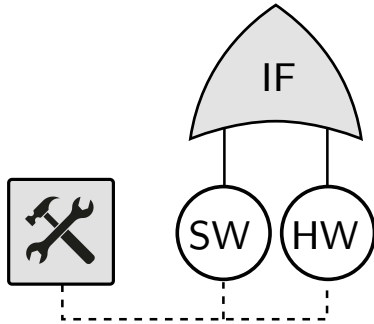


Fault Trees

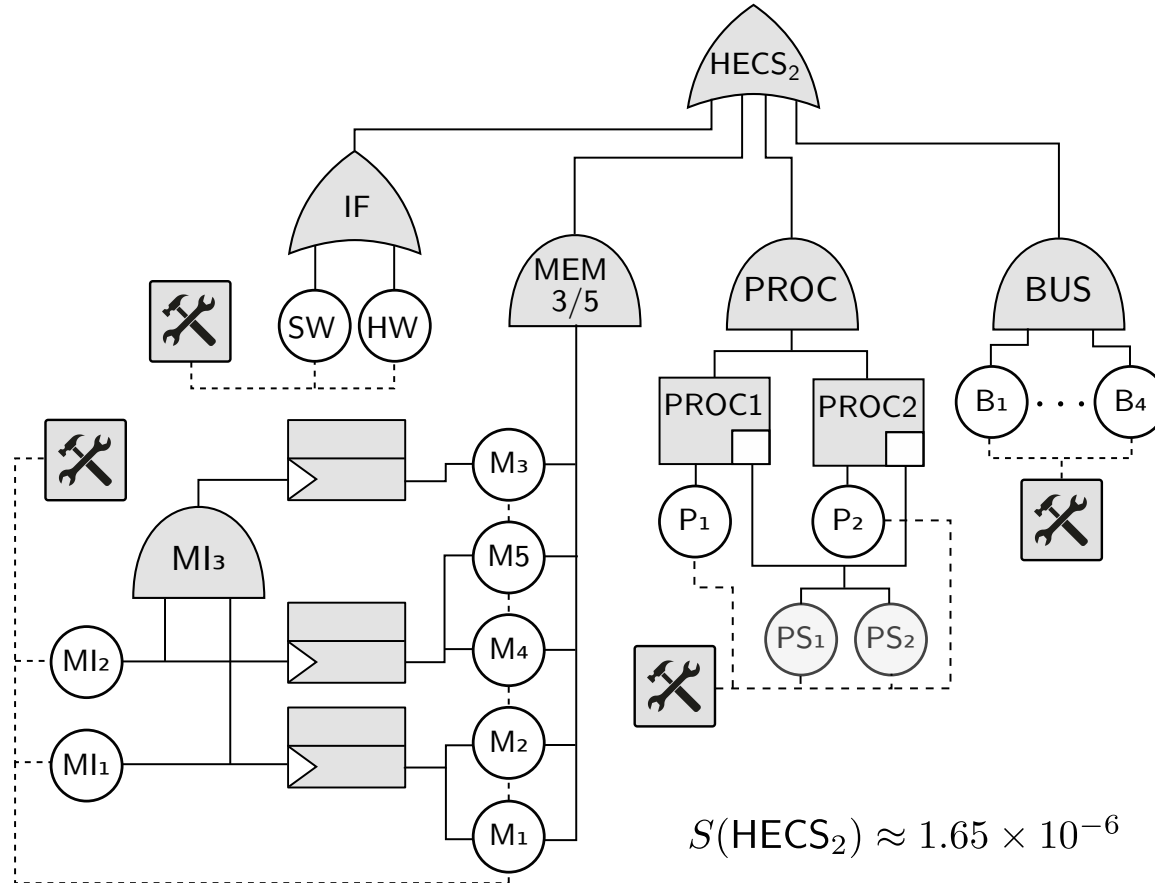




Dynamic + Repairable + Non-Markovian



HECS: Hypothetical Example Computer System



UNIVERSITÀ DI TRENTO

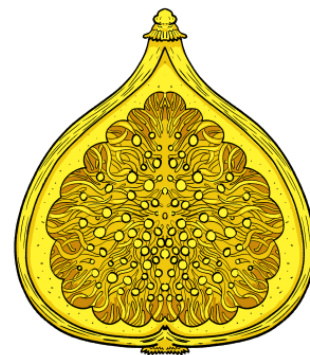


FIG:

the Finite Improbability Generator v1.3

Carlos E. Budde

Security group @ Dipartimento di Ingegneria e Scienza dell'Informazione
Funded by EU grants 830929 ([CyberSec4Europe](#)) and 952647 ([AssureMOSS](#))

carlosesteban.budde@unitn.it

TOSME — Nov 12, 2021