On Dependability Assessment of Fault Tolerant Systems by Means of Statistical Model Checking

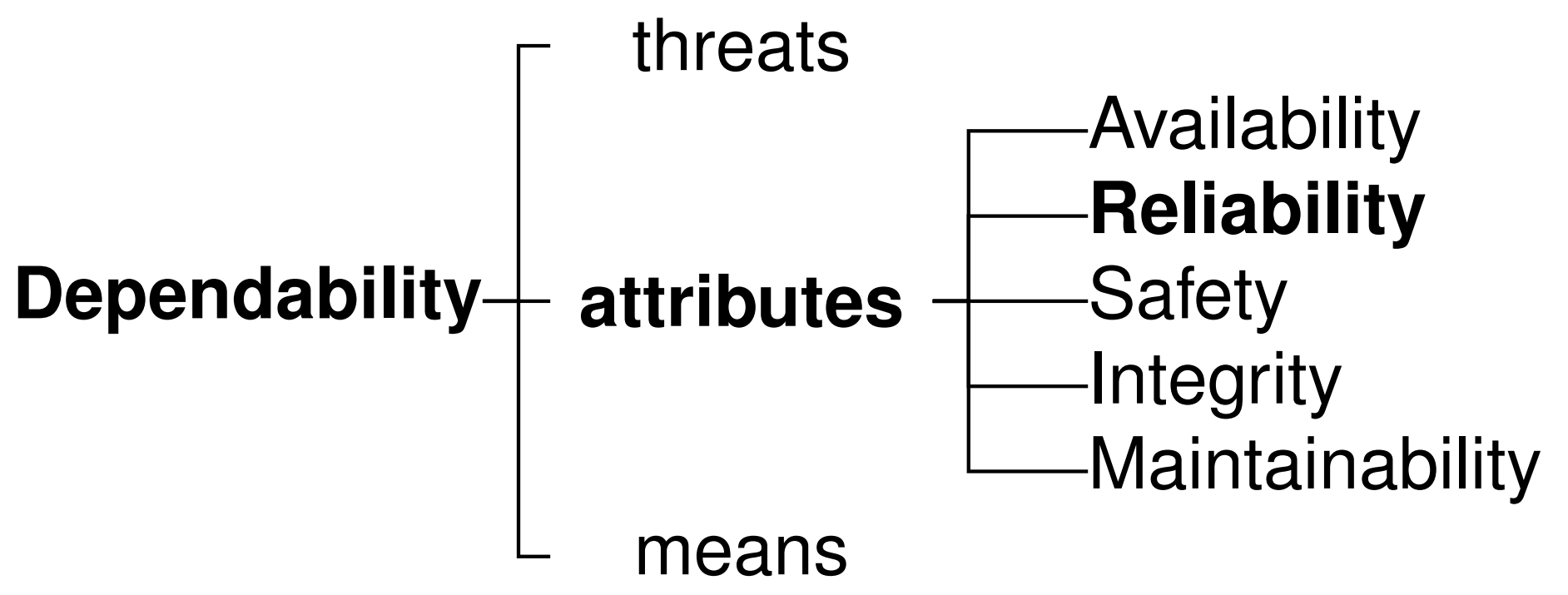
Josef Strnadel

Faculty of Information Technology, Brno University of Technology, Czech Republic

strnadel@fit.vutbr.cz, http://www.fit.vutbr.cz/∼strnadel

Introduction to Dependability

- The ability of a system to provide a required service and to perform it for a specified period of time within specified conditions is denoted as **dependability**.
- It can be meant in a **qualitative** or a **quantitative** manner [1]. **Qualitatively**, it can be seen as "the ability to deliver a service that can be justifiably trusted" [1] or, as a property such that "reliance can be justifiably placed on the services delivered by the system" [2].
- Since dependability is a complex feature composed of many attributes, the (overall) dependability cannot be simply quantified by a single value. Instead, the **attributes are quantified to form a complex image about dependability**. As the time of occurrence of a fault, error or failure cannot be specified certainly, the attributes are typically described by means of the probability theory based on which attributes such as reliability, maintainability or availability can be quantified.



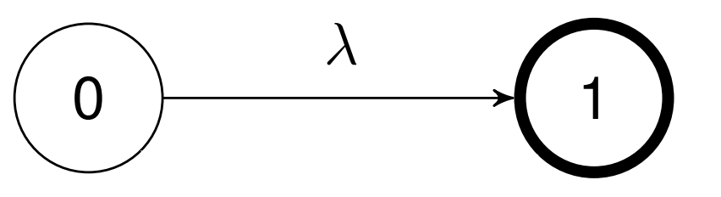
Dependability Assessment

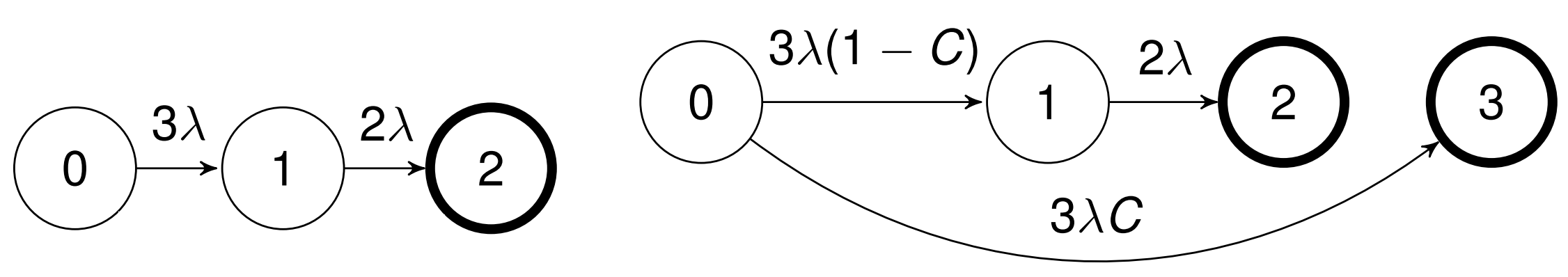
- X_{TTF} …continuous random variable representing the **time to failure** (**TTF**)
- $f(t)$ …**probability density function** (**PDF**) of X_{TTF} representing the probability that a system fails in t
- $F(t)$ …prob. that a failure occurs before or at t; i.e., **cumulative distribution function** (**CDF**) of X_{TTF}; $F(t) \overset{def}{=} \int_{-\infty}^{t} f(x)\, dx$
- $R(t)$ …**reliability function** (**reliability**): prob. that a failure occurs after t; $R(t) \overset{def}{=} 1 - F(t) = \int_{t}^{\infty} f(x)\, dx$
- MTTF (**Mean Time To Failure**)
- $h(t)$ …**hazard** (**rate**) **function**: prob. that a failure occurs in $[t, t + dt]$ given that no has occurred prior to t; $h(t) \overset{def}{=} \frac{dF(t)}{dt} \times \frac{1}{R(t)} = \frac{f(t)}{R(t)}$

But, the assessment is complicated by real facts such as fault dependencies, dynamic behavior of faults, state-dependent behavior, faults being introduced into the reconfiguration/recovery process, shared load/repair facilities, multiplicity of faults and failure modes etc.

Common Reliability Models

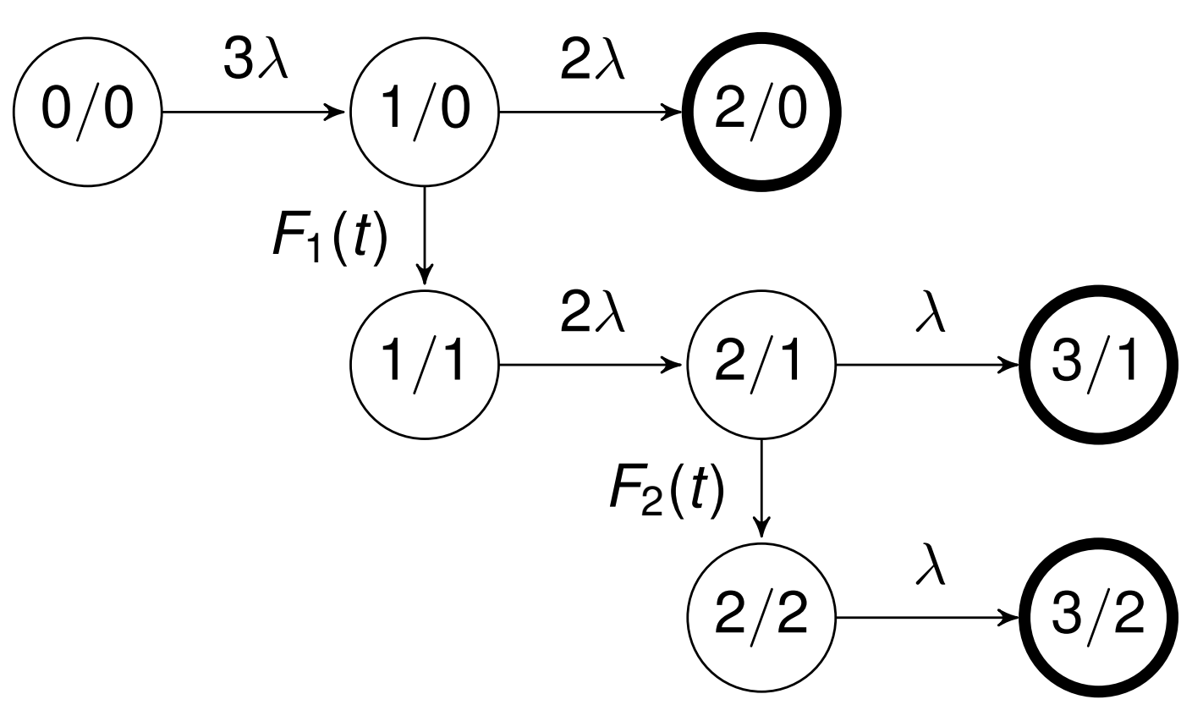
a) **Simplex** (SPX)

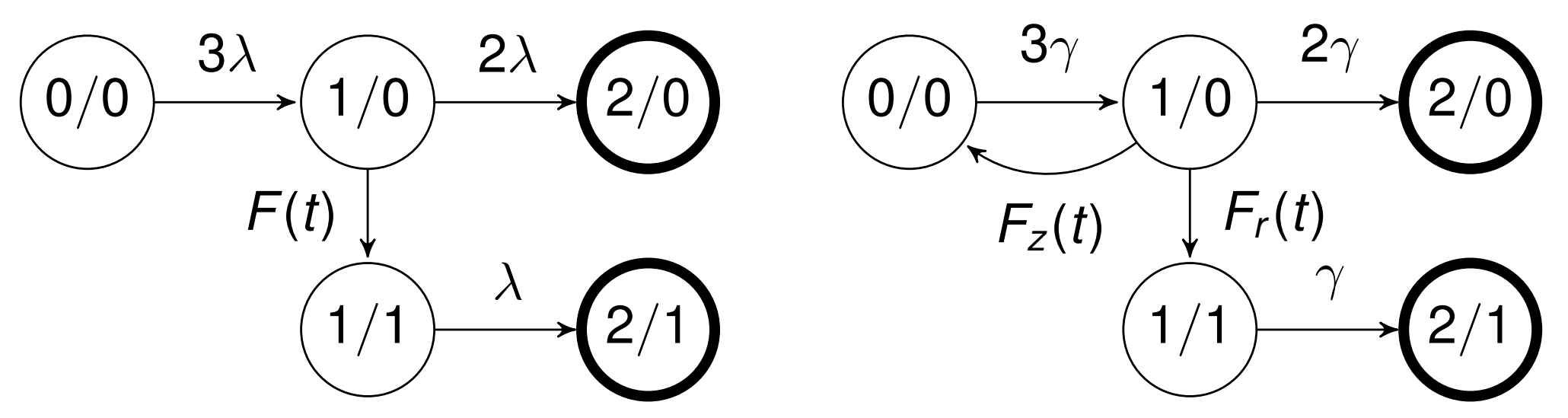




b) **Triple Modular Redundancy** (TMR) w.o. resp. with **Single Point of Failure** (SPF) on the left (TMR_{NSF}) resp. right (TMR_{1SF})

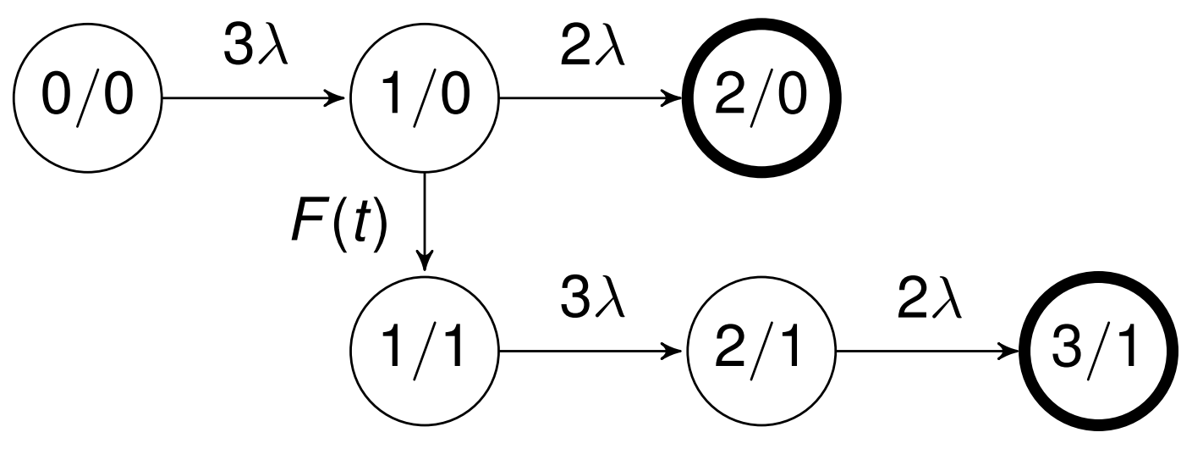
c) **Triplex with successive degradation** (TSD)



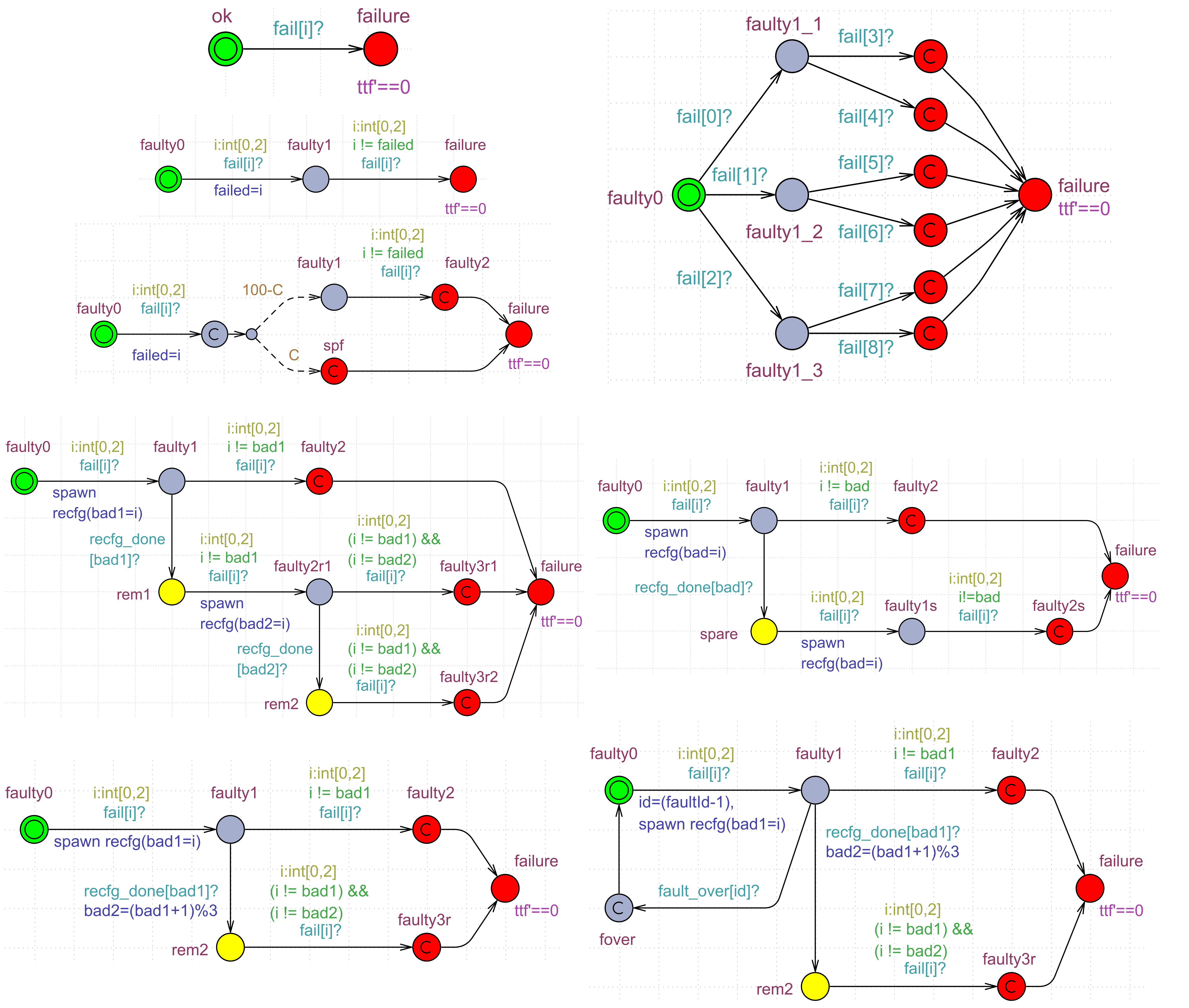


d) **Triplex to simplex degradation** (TTS) subject to permanent resp. transient faults on the left (TTS_P) resp. right (TTS_T) of the figure

e) **Triplex with one spare** (T1S)



STA Reliability Models



Utilized STA Fault Models

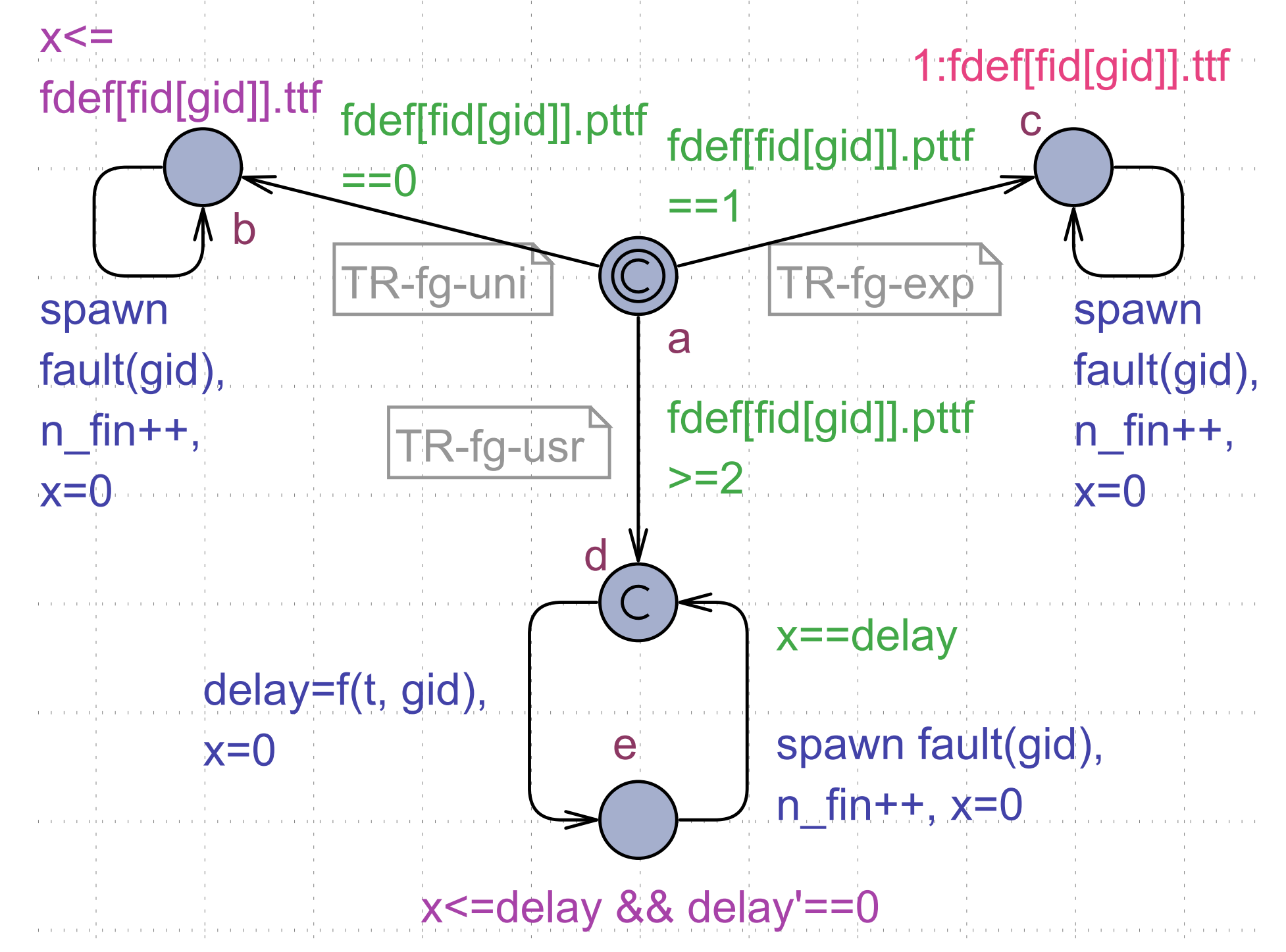


Fig. 1: Fault generator model [4]

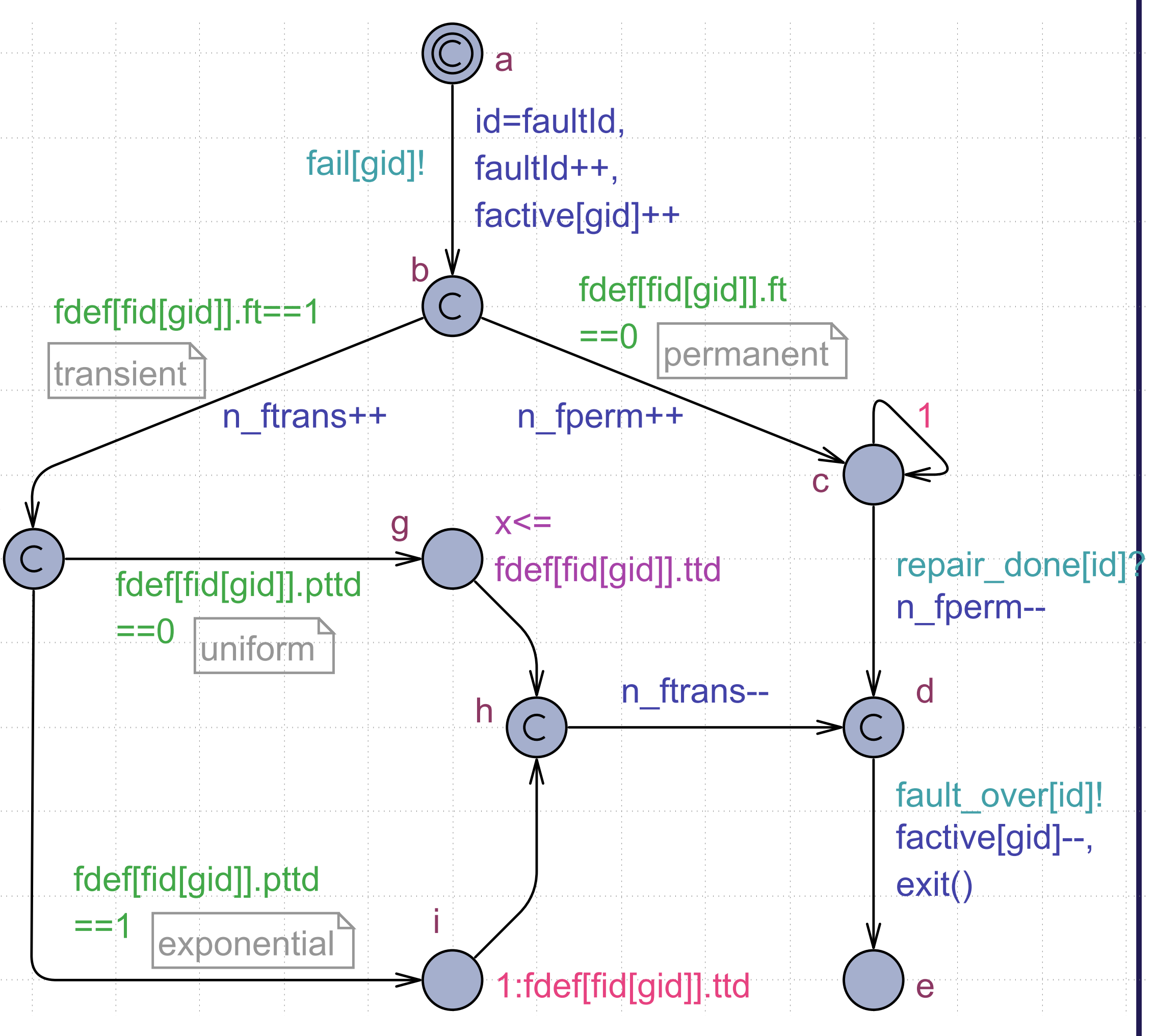


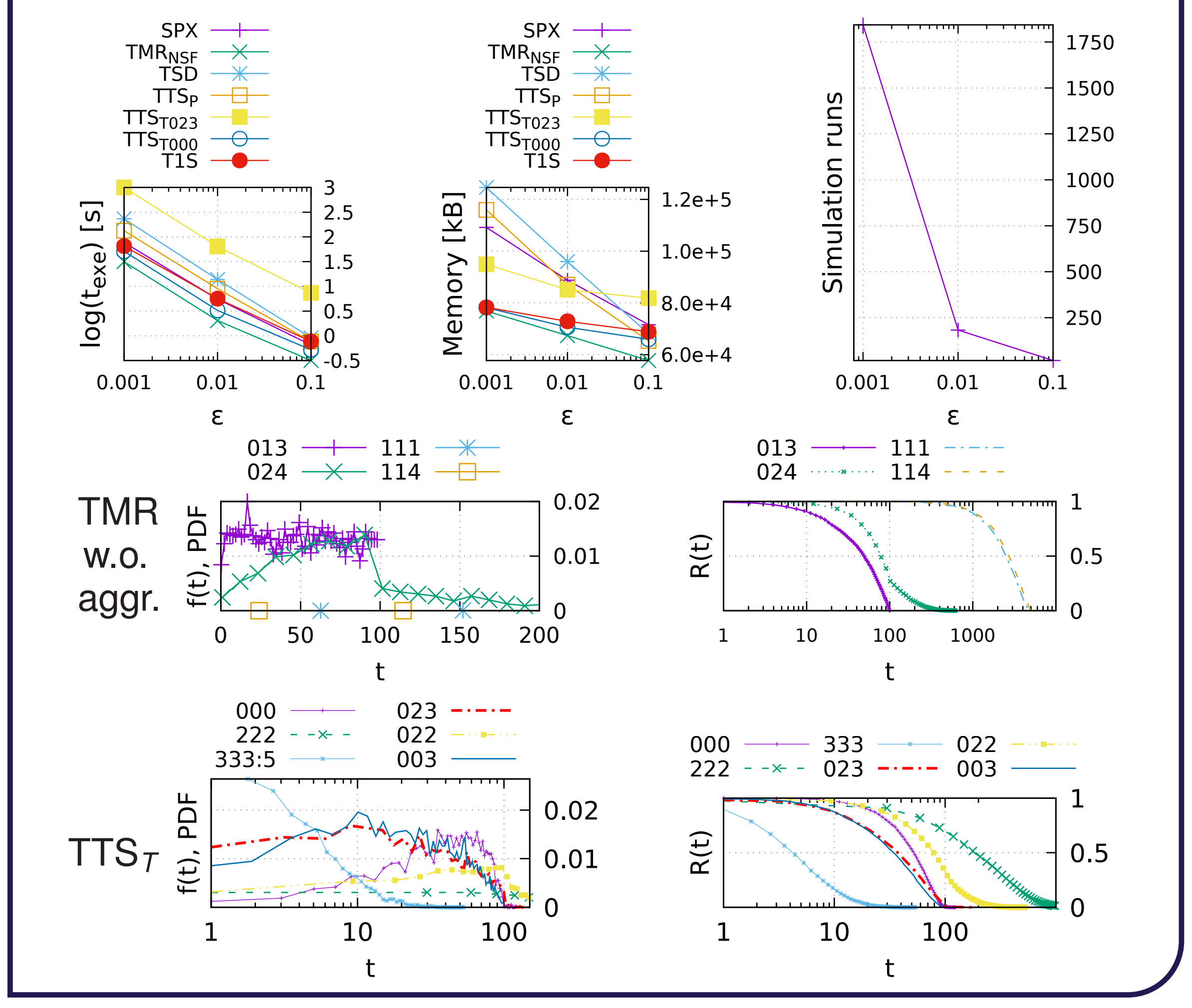
Fig. 2: Fault behavior model [4]

SMC Query Example

Probability estimation using "`Pr[bound](`ϕ`)`"

`Pr[<= 100000](<> STA.failure)`

Representative Results



References

[1] A. Avižienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic Concepts and Taxonomy of Dependable and Secure Computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 1, pp. 11–33, 2004. DOI 10.1109/TDSC.2004.2.

[2] J.-C. Geffroy and G. Motet, *Design of Dependable Computing Systems*. Hingham, MA, USA: Kluwer Academic Publishers, 2002.

[3] A. David, K. Larsen, A. Legay, M. Mikucionis, and D. Poulsen, "Uppaal SMC Tutorial," *International Journal on Software Tools for Technology Transfer*, vol. 17, no. 4, pp. 397–415, 2015. DOI 10.1007/s10009-014-0361-y.

[4] J. Strnadel, *On Creation and Analysis of Reliability Models by Means of Stochastic Timed Automata and Statistical Model Checking: Principle*. In: Proc. of 7th International Symposium on Leveraging Applications of Formal Methods, Verification and Validation: Foundational Techniques (ISoLA), Part I. Cham: Springer International Publishing, 2016, pp. 166–181. [Online]. Available: `http://dx.doi.org/10.1007/978-3-319-47166-2\_11`

Acknowledgement

This work was supported by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project "IT4Innovations excellence in science – LQ1602" and the project Advanced Parallel and Embedded Computer Systems (FIT-S-17-3994).

To be presented at the 2017 Euromicro Conference on Digital System Design (DSD 2017) held in Vienna, Austria on 30 August – 1 September, 2017