1

Using the Unscented Transform to Assess Systems Reliability

José Edil G. de Medeiros
Gilmar Silva Beserra
Sandro A. P. Haddad

Universidade de Brasília

2

Outline

1. Problem
2. The Unscented Transform
3. Monte Carlo x UT
4. Case study on reliability assessment
5. Future work

2

Outline

3

Context

- Safety is a critical constraint on some applications (avionics, nuclear power plants, implantable medical devices, etc).
  - Design fault-tolerant systems is imperative.
  - Fault-tolerance is in most cases achieved by means of redundancy.
  - Adds costs and complexity.
- Reliability: the probability of a given system to be working on time t given it was initially working correctly.
  - Measures the continuous delivery of correct service by a system.

3

Safety is a critical constraint in some applications where people's lives depend on the correct functioning of systems such as in avionics, nuclear power plants, implantable medical devices and others. They shall deliver correct service even in the presence of failing parts. Fault-tolerance is, in this way, a design constraint for such systems.
Reliability is a measured of dependability, defined by the probability of a given system to be working correctly on time t given it was initially working correctly.

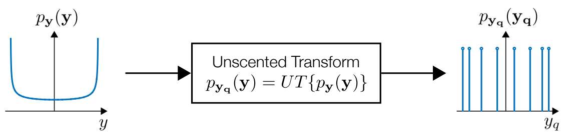
4

How to **efficiently** include reliability evaluation on system design methodology?

So, in this work our problem is how to efficiently include reliability evaluation on the design methodology?

5

The Unscented Transform



5

The Unscented Transform is a mathematical framework that models a continuous probability function into a discrete one. It was first proposed on the context of the control theory by Uhlmann to overcome the linearization issues in the implementation of Extended Kalman Filter. It has being used on many different applications, including the study of Failure Prognosis (Leão, 2016 - Embraer) for aircraft mechanical parts and for representing uncertainty on aircraft route evaluation (Erlandsson, 2014 - Saab).

6

UT formulation

$$E\{X\} = \int_{-\infty}^{\infty} x^k p_x(x)\, dx = \sum_{i=1}^{m} S_i^k w_i = E\{X_q\}$$

6

The UT formulation is based on preserving the statistical moments of the continuous distribution. The goal is to calculate a list of Sigma points with their associated probabilities. On the numerical side, we do this by looking at the UT as a gaussian quadrature scheme and we use recurrence relations for orthogonal polynomials. We can show that these sigma points can be used to study the statistical behavior of a nonlinear mapping g(x) by means of the moments.

Monte Carlo x UT

$$I = \int_0^1 g(t)dt.$$

7

7

We can show how to use the UT as an alternative to Monte Carlo by the problem of calculating the value of the integral of a nonlinear function.

Monte Carlo x UT

$$E\{g(X)\} = \int_{-\infty}^{\infty} g(x)p_x(x)\,dx = \int_0^1 g(x)\,dx$$

8

8

If g(x) is too complex or if we don't know an analytical expression for it, as is the case of complex algorithms, we can do so by defining an uniform distributed random variable X.

Them, we form a new random variable y = g(x). The first moment of Y will be given by the following equation, which is essentially the value we want to get. Thus, the initial problem can be changed to the estimation of the first moment of a random variable.

Monte Carlo x UT

$$E\{g(X)\} = \int_{-\infty}^{\infty} g(x)p_x(x)\,dx = \int_0^1 g(x)\,dx$$

Monte Carlo

$$I = E\{g(x)\} \approx \frac{1}{n}\sum_{n} g(x_i).$$

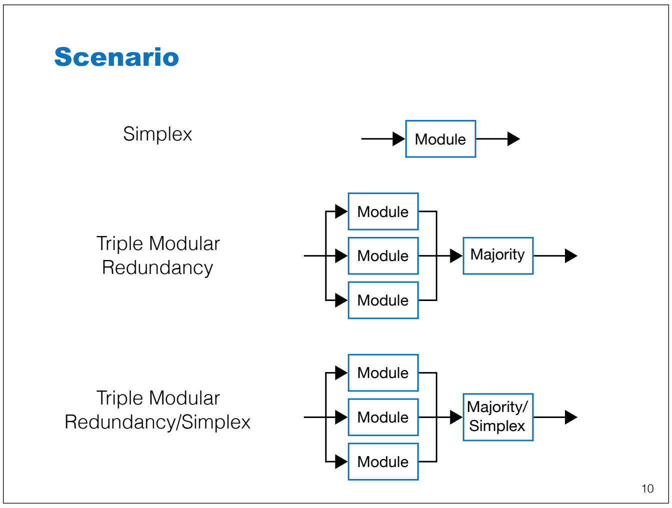
Unscented Transform

$$I = E\{g(x)\} \approx \sum_{i=1}^{m} w_i g(S_i).$$

9

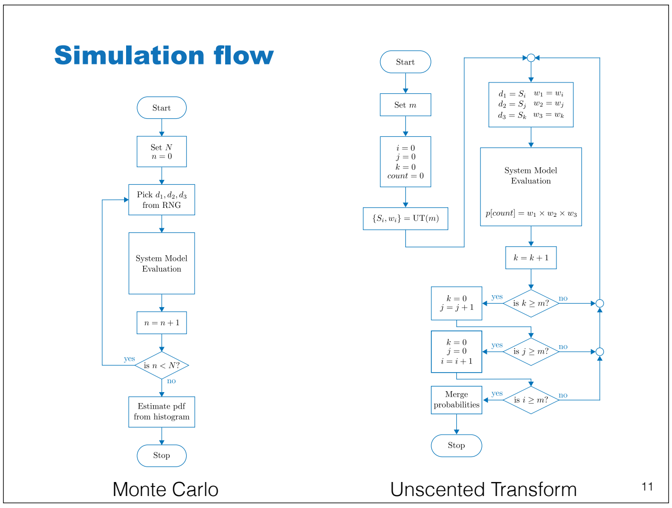
9

On the Monte Carlo approach, we pick random samples, evaluate the nonlinear mapping on each of these samples and use an estimator to get the desired result. On the other hand, with the UT, we evaluate the nonlinear mapping on each of the sigma points and calculate the first moment of this new discrete distribution. The advantages of the UT is a faster convergence for the estimator and the use of deterministic points. Also, in Monte Carlo, you have to guess a probability distribution for the random variables. In the UT framework, we can rely on empirical data to get the sigma points.



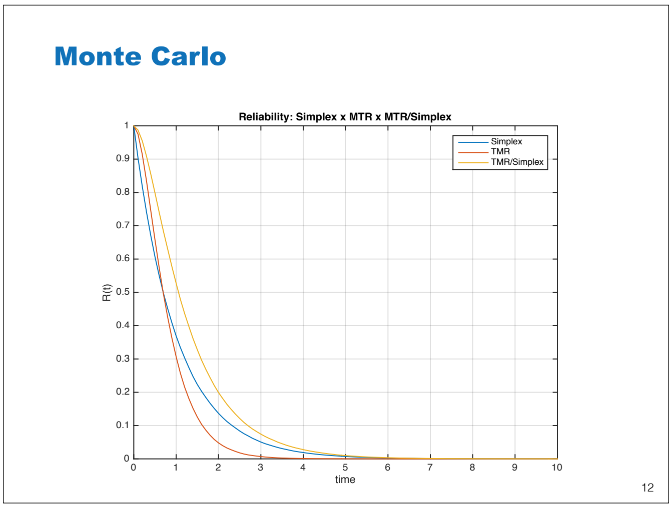
10

To show how it applies to fault-tolerant systems context, lets consider three kinds of fault tolerance schemes: simplex, TMR and TMR/Simplex. On Simplex, we have no fault-tolerance at all. It is simple and cheap on design resources. On TMR we have replicated modules and the results of the module are taken by a majority voter. For the TMR/ Simplex, when a module fails, we put a good module aside.



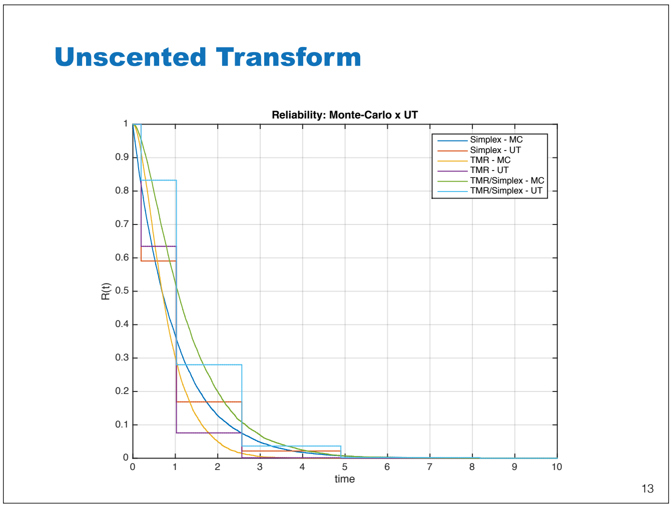
11

These are flux diagrams detailing the simulations. Note that the difference between these and the MC schemes resides on how we choose our simulation points and on how we process the results later. The system model remains the same so it can be used with any existing modeling and simulation tool.



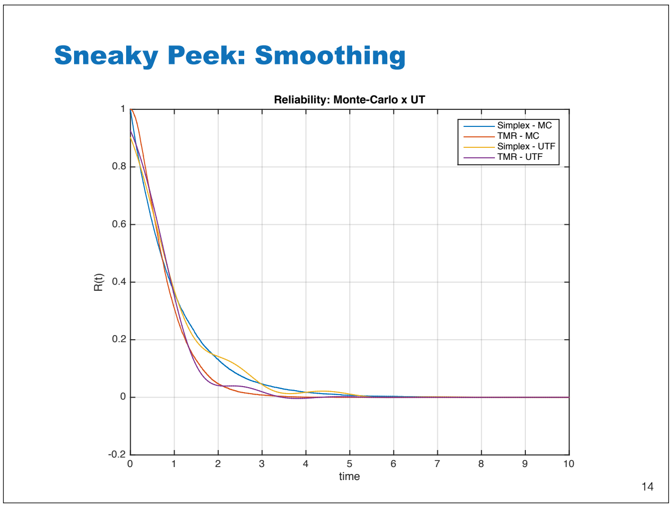
12

To give a feel on what we get, this the kind of result taken from the Monte Carlo with 1e4 points. Note that TMR is more reliable than the Simplex scheme only for mission times shorter than 0.7 of MTTF.



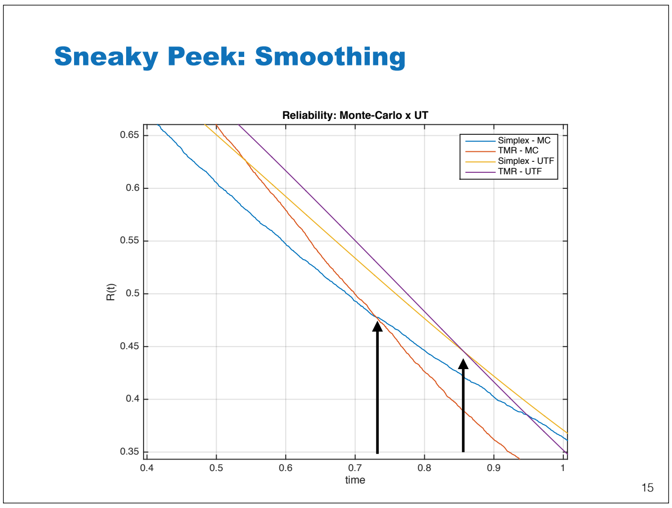
13

This is what we get with the UT with 7 simulation points for each module. We get these staircase patterns for the probabilities due to the discrete nature of the transform. We can get range estimations for important features just like the mission time.



14

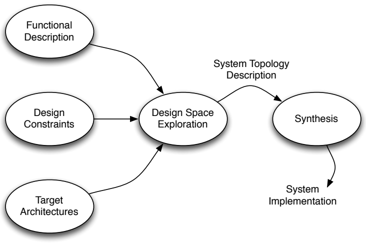
This is a sneaky peek on what we are trying to do in Brazil. Using signal processing techniques, we can smooth out those probability curves, getting more accurate estimates for important features.



15

Such as mission time for different FT techniques.

Next steps



16

I have just moved to Sweden to work for one year in KTH. They have being working for a long time on ForSyDe, a formal methodology to design heterogeneous embedded and cyber-physical systems. I have already a colleague working on reliability evaluation using ForSyDe models but I believe we can go further and use the UT also on the design space exploration phase of the methodology where we can consider design constraints such as safety and include platform reliability details into the design tools. In this way, we may expand the universe of possible implementations for given applications.

Universidade de Brasília

j.edil@ene.unb.br

17