MATHEMATICAL FRAMEWORK FOR THE ANALYSIS OF DYNAMC STOCHASTIC SYSTEMS WITH THE RAVEN CODE

C. Rabiti, D. Mandelli, A. Alfonsi, J. Cogliati, R. Kinoshita
Idaho National Laboratory
2525 North Fremont Street, Idaho Falls (ID)
{cristian.rabiti, diego.mandelli, andrea.alfonsi, joshua.cogliati, robert.kinoshita}*@inl.gov*

ABSTRACT

RAVEN (Reactor Analysis and Virtual control Environment) is a software code under development at Idaho National Laboratory aimed at performing probabilistic risk assessment and uncertainty quantification using RELAP-7, for which it acts also as a simulation controller. In this paper we will present the equations characterizing a dynamic stochastic system and we will then discuss the behavior of each stochastic term and how it is accounted for in the RAVEN software design. Moreover we will present preliminary results of the implementation.

Key Words: probabilistic risk assessment, dynamic stochastic systems, uncertainty quantification

1. INTRODUCTION

The Light Water Reactor Sustainability (LWRS) [1] is a DOE campaign aimed to assess the safety of extending the operational lifetime of the Light Water Reactors beyond 60 years. Within this framework there are two major efforts of interest to the RAVEN (Reactor Analysis and Virtual control ENvironment) project [2] presented here: RELAP-7 [3] and RISMC [4] (Risk Informed Safety Margin Characterization).
RELAP-7 is the next generation of system analysis codes currently developed at INL for the safety analysis of Nuclear Power Plants (NPPs). RISMC is an effort to build a coherent risk informed framework to determine the risk associated with power uprates and life extension of NPPs. While RELAP-7 is used to provide a more accurate deterministic plant simulation, the task to perform the statistical part of the analysis is performed by the RAVEN code.
This paper focuses on the mathematical framework from which the software structure of RAVEN is derived. Raven performs a statistical analysis of the NPP behavior that accounts for all the phenomena that are sources of uncertainty in the dynamics of accident scenarios.
One of the main goals of this paper is to analyze how the time dependent behavior of the aleatory components of systems impacts the software requirements. In particular, we discovered that two fundamental characterizations of the system to be modeled have a deep impact on the software design: the Markovian [5] property and the size of the phase space.
A non-Markovian system is difficult to model when the system is as complex as a NPP because of the huge amount of information that needs to be calculated at each time step. Therefore, we have carefully evaluated under which assumptions is possible to cast a generic model in a Markovian equivalent system. Of course, this transformation requires the introduction of additional variables into the phase space that needs to be properly characterized. The approach that we present in this paper consists of analyzing individually the contributions of the additional

variables to the stochasticity of the system, evaluating their impacts on the phase space, and determining under which approximation they could be modeled.

2. MATHEMATICAL CHARACTERIZATION OF STOCHASTIC DYNAMIC SYSTEMS

2.1. General System Equation and Variable Classification

The scope of this section is to present the mathematical framework underlying RAVEN's development to model dynamic stochastic systems accounting also for uncertainties. By dynamic stochastic systems we refer to systems whose dynamics contain random (i.e., not predictable a priori) elements.
Possible examples of random variables/parameters are:

- Random variability of parameters (e.g., uncertainty associated physical parameters)

- Presence of noise (background noise due to intrinsically stochastic behaviors or lack of detail in the simulation)

- Uncertainty in the initial and boundary conditions

- Random failure of components

- Aging effect

Random behaviors, although present in nature, are often artificially introduced into models to account for the inability to produce simulations that explicitly treat all the underlying physics.
The distinction between variables that are artificially considered aleatory and the ones intrinsically aleatory corresponds with the classical definition of epistemic (artificial) and aleatory (intrinsic) uncertainties. From a system simulation point of view it is more relevant how these variables, the sources of aleatory behavior, change in time.

Before introducing the mathematical models for uncertainty, let's indicate the NPP status in the phase space through the vector $\bar{\theta}(t)$ and the NPP trajectory in the phase space through the following equation:

$$\frac{\partial \bar{\theta}(t)}{\partial t} = \bar{\mathcal{H}}(\bar{\theta}(t), t) \tag{1}$$

In Eq. (1) we have implicitly assumed a first order time differentiability of the trajectory equation representing the system in the phase space; typically, this is not correct and not generally required but it is used here for compactness of the notation.
By accounting also for the initial conditions we obtain:

$$\begin{cases} \dfrac{\partial \bar{\theta}(t)}{\partial t} = \bar{\mathcal{H}}(\bar{\theta}(t), t) \\ \bar{\theta}(t_0) = \bar{\theta}_0 \end{cases} \tag{2}$$

At this point, each possible source of uncertainty or stochastic behavior is considered and progressively added in Eq. (2).

For our scope, it is helpful to split the space phase between continuous (e.g., temperature and pressure) and discrete variables (e.g., status of components including both operating and failure states) as follows:

- $\bar{\theta}^c \in \Phi \subseteq \mathbb{R}^C$ the set of continuous variables,

- $\bar{\theta}^d \in \Psi \subseteq \mathbb{N}^D$ the set of discrete variables, and,

- $\bar{\theta}(t) = \bar{\theta}^d \oplus \bar{\theta}^c$

Equation (2) now takes the following form:

$$\begin{cases} \dfrac{\partial \bar{\theta}^c(t)}{\partial t} = f(\bar{\theta}^c, \bar{\theta}^d, t) \\ \dfrac{\partial \bar{\theta}^d(t)}{\partial t} = g(\bar{\theta}^d, \bar{\theta}^c, t) \\ \bar{\theta}^c(t_0) = \bar{\theta}^c_0 \\ \bar{\theta}^d(t_0) = \bar{\theta}^d_0 \end{cases} \tag{3}$$

As mentioned above, for simplicity of notation we also use the time derivative for discrete variables.

2.2. Probabilistic Nature of the Parameters Characterizing the Equation

The first stochastic behaviors to be introduced are the uncertainties associated with the:
- initial conditions (i.e., $\bar{\theta}^c$ and $\bar{\theta}^d$ at time t_0), and,
- parameters characteristic of $f(\bar{\theta}^c, \bar{\theta}^d, t)$ and $g(\bar{\theta}^d, \bar{\theta}^c, t)$

as shown in Eq. (4).

$$\begin{cases} \dfrac{\partial \bar{\theta}^c(t)}{\partial t} = f(\bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, t) \\ \dfrac{\partial \bar{\theta}^d(t)}{\partial t} = g(\bar{\theta}^d, \bar{\theta}^c, \bar{\alpha}_{Staz}, t) \\ \Pi(\bar{\theta}^c, t_0) \sim pdf(\bar{\theta}^c_0|, \sigma_c^2) \\ \Pi(\bar{\theta}^d, t_0) \sim pdf(\bar{\theta}^d_0|, \sigma_d^2) \\ \bar{\alpha}_{Staz}(t) = \bar{\alpha}_{Staz}(t_0) \sim pdf(\alpha^0_{Staz}|, \sigma^2_{Staz}) \end{cases} \tag{4}$$

In Eq. (4), $\Pi(\bar{\theta}^c, t_0)$ indicates the probability distribution of $\bar{\theta}^c$ at the initial time $t = t_0$ while $pdf(m|, \sigma^2)$ indicates a generic probability distribution function having mean m and sigma σ. The term $\bar{\alpha}_{Staz}$ is the vector of parameters affected by uncertainty but not varying over time.

Up to now, we have considered uncertainties whose values do not change during the simulation. This set of uncertainties accounts for most of the common sources of stochastic behaviors. Examples of this kind of uncertainties are:

- *Uncertainty associated with the heat conduction coefficient.* This value is known (but uncertain) and has no physical reason to change during the simulation

- *Uncertainty of failure temperature for a pipe*. This value is usually characterized by a probability distribution function but once the value has been set (like through random sampling) it will not change during the simulation.

From a software point of view, all the probabilistic behaviors connected to $\Pi(\bar{\theta}^c, t_0)$, $\Pi(\bar{\theta}^d, t_0)$ and $\bar{\alpha}_{Staz}(t)$ can be modeled without changing the dimensionality of the phase space (hence, no alteration of the solution algorithm is required) and they can be implemented by simply performing a sampling of the input space.

2.3. Variables Subject to Random Motion

The next aleatory component to be accounted for is the variable $\bar{\alpha}_{Brown}$ that continuously and randomly changes over time. To make an easy parallel we refer to these parameters as if they behave like a Brownian motion. Two examples of these randomly varying variables are:

- *Cumulative damage growth in materials.* Experimental data and models representing this phenomenon show large uncertainties. There is also an intrinsic natural stochasticity driving the accumulation of the damage (natural Brownian motion.)
- *Heat conductivity in the fuel gap during heating of fuel.* During some transients there are situations when the fuel is contact with the clad while in others where there is the presence of a gap. While in nature this is a discontinuous transition, it is not usually possible to model in such a way, especially if vibrations of the fuel lead to high frequency oscillations. In this case, it would be helpful to introduce directly into the simulation a random noise characterizing the thermal conductivity when these transitions occur (artificial Brownian motion.)

The system (4) should now be rewritten in the following form:

$$\begin{cases} \dfrac{\partial \bar{\theta}^c(t)}{\partial t} = f(\bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) \\[2mm] \dfrac{\partial \bar{\theta}^d(t)}{\partial t} = g(\bar{\theta}^d, \bar{\theta}^c, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) \\[2mm] \dfrac{\partial \bar{\alpha}(t)_{Brown}}{\partial t} = b(\bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t)\Gamma(t) \\[2mm] \Pi(\bar{\theta}^c, t_0) \sim pdf(\bar{\theta}_0^c |, \sigma_c^2) \\[2mm] \Pi(\bar{\theta}^d, t_0) \sim pdf(\bar{\theta}_0^d |, \sigma_d^2) \\[2mm] \bar{\alpha}_{Staz}(t) = \bar{\alpha}_{Staz}(t_0) \sim pdf(\bar{\alpha}_{Staz}^0 |, \sigma_{Staz}^2) \\[2mm] \bar{\alpha}_{Brown}(t_0) \sim \bar{\alpha}_{Brown}^0 \Gamma(t_0) \end{cases} \qquad (5)$$

where $\Gamma(t)$ is 0-mean random noise.

Clearly the equation referring to the time change of the parameters subject to the *Brownian* motion should be interpreted in the Ito sense [5].

From a software development point of view, parameters like $\bar{\alpha}_{Brown}$, subject to a Brownian motion behavior need to be added to the phase space and, therefore, evaluated at each time step. In fact the phase space is so composed $\bar{\theta}(t) = \bar{\theta}^d \oplus \bar{\theta}^c \oplus \bar{\alpha}_{Brown}$,

2.4. Discontinuously and Stochastically Varying Variables

The last and probably most difficult step is the introduction of parameters that are neither constant during the simulation nor continuously vary over time. As an example, consider a valve that, provided set of operating conditions, opens or closes. If this set of conditions is reached n times during the simulation, the probability of the valve correctly operating should be sampled n times. It is also foreseeable that the history of failure/success of the valve will impact future probability of failure/success. In this case the time evolution of such parameters, discontinuously stochastic changing parameters, $\bar{\alpha}_{DS}$, is governed by the following equation:

$$\frac{\partial \bar{\alpha}_{DS}}{\partial t} = \bar{\delta}(\bar{\alpha}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) \cdot \bar{V}(\bar{\alpha}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) \cdot$$
$$\cdot \bar{p}\left(\int_{t_0}^{t} dt' S(\bar{\theta}(t'), t') \right) \tag{6}$$

where:
- The function $\bar{\delta}(\bar{\alpha}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brow}n, t)$ is the delta of Dirac of the instant on which the transition need to be evaluated (control logic signaling to the valve to open/close)
- The term $\bar{p}\left(\int_{t_0}^{t} dt \bar{\alpha}_{DS}, \bar{\alpha}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t \right)$ represents the transition probability between different states (in case of the valve: open/close). Note that the time integral of the parameter history accounts for the memory of the component via the kernel $S(\bar{\theta}(t'), t')$.
- The term $\bar{V}(\bar{\alpha}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t)$ is the rate of change of $\bar{\alpha}_{DS}$. For a discrete parameter, it is defined as the value of the instantaneous $\bar{\alpha}_{DS}$ change

To deal with complexity of $\bar{p}$ we can introduce a set of new dimensions in the phase space: the time at which the parameters changed status and their correspondent value $\{(\bar{\alpha}_{DS}, t)_i\} = \{\bar{\alpha}_{DS_i}, t_i\} = \bar{\bar{\alpha}}_{DS}, \bar{t}$ (for $i = 1, \dots, n$)
Equation (6) takes now the form:

$$\frac{\partial \bar{\alpha}_{DS}}{\partial t} = \bar{\delta}(\bar{\alpha}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) \cdot \bar{V}(\bar{\alpha}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) \cdot$$
$$\cdot \bar{p}(\bar{\bar{\alpha}}_{DS}, \bar{t}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) \ for \ t \geq t_n \tag{7}$$

Note that this formulation introduces a phase space continuously growing over time ($n \rightarrow \infty$). In this respect, it is useful to introduce to discuss possible assumptions:
1. The memory of the past is not affected by the time distance; in this case:

$$\bar{p}(\bar{\bar{\alpha}}_{DS}, \bar{t}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) = \bar{p}(\bar{\bar{\alpha}}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brown}, t) \tag{8}$$

The dimensionality of the phase space is still growing during the simulation since more and more sampling is performed, but the time integral is removed from the transition probability. A simple example of this situation is a component activated on demand in which failure is a function of all previous sampling, but not of when the component was sampled or in which sequence the outcome occurred.

2. The number of samples is determined before the simulation itself takes place (e.g. n times.) In this case the different $\bar{\alpha}_{DS_i}$ could be treated explicitly as $\bar{\alpha}_{Staz}$ while $\bar{t}$ would still remain a variable to be added to the phase space (if simplification 1 is not valid) but of fixed dimension. The transition probability becomes: $\bar{p}\left(\int_{t_0}^{t} dtS(\bar{t}), \bar{\alpha}_{DS}, \bar{\theta}^c, \bar{\theta}^d, \bar{\alpha}_{Staz}, \bar{\alpha}_{Brow}, t\right)$.

This is the case, for example, of a component that is sampled a fixed number of times for a given simulation while the contribution of the history to the transition probability might decay exponentially over time. This approximation might eliminate the memory from the system by adding n variables to the phase space t_i (for $i = 1, \dots, n$) thus restoring the Markovian characteristics.

3. Another possible approximation alternative to the previous one is that the memory of the system is limited only to a fixed number of steps back in the past. In this case n is always bounded. Therefore adding $\{\bar{\alpha}_{DS_i}, t_i\}$ (for $i = 1, \dots, n$) would possibly preserve the system Markovian properties of the system. This approximation allows for eliminating the memory from the system by expanding the phase space 2n variables. From a software implementation point of view, this is the most complex situation since without any simplification we would have to deal with a system that is never reducible to a Markovian one and therefore forced to use the whole history of the system to forecast its evolution at each time step.

Assumption 1 limits this cost by restraining it to the set of values assumed by the variable but would still lead to very difficult to deal with situation. Assumption 2 would require an expansion of phase space to introduce the time at which the transitions happens but the value that the parameter will assume at each sampling could be treated as initial condition. Assumption 3 would instead require the expansion of the phase space for both the time and the values of the transitioning variables.

3. SOFTWARE INFRASTRUCTURE

3.1. Requirements

Table 1 summarizes the different sources of aleatory behaviors described in Section 2 and their impacts on the definition of the problem.

Assuming for the moment that the probabilistic analysis is performed using Monte-Carlo sampling and that discontinuously and stochastically varying variables always fall under assumptions 2 or 3 (see Section 2.4), to accommodate the necessary features, the software needs to:

1. Sample the input space parameters
2. Expand the phase space to be traced (introduction of auxiliary variables)
3. Evaluate transition probabilities at each time iteration

Table 1: Classification of Stochastic Sources

Type	*Impact on the Markovian property*	*Increase in the phase space*	*Impact in the initial condition space to be sampled*
Uncertainty in equation parameters and initial conditions	*None*	*None*	*All parameters need to be sampled at the beginning of the simulation*
Brownian-like variables	*None*	*One additional dimension is needed for each variable*	*Only their initial conditions*
Discontinuously and Stochastically Varying Variables	*In the general case the system might not be Markovian anymore*	*To recreate a Markovian system would require a continuously growing number of variables*	*None*
Discontinuously and Stochastically Varying Variables (Assumption 1)	*In the general case the system might not be Markovian anymore*	*To recreate a Markovian system would require a continuously growing number of variables*	*None*
Discontinuously and Stochastically Varying Variables (Assumption 2)	*Markovian property might be preserved*	*The transition instants need to be added to the phase space*	*The values must be added to the initial condition to be sampled*
Discontinuously and Stochastically Varying Variables (Assumption 3)	*Markovian property might be preserved*	*The transition instants and the corresponding values need to be added to the phase space*	*None*

3.2. Implementation

RAVEN currently interfaces with RELAP-7 but its implementation is such that it can easily interface with any MOOSE [6] based application. In fact, RAVEN does not interact directly with RELAP-7 but with the underlying solver (i.e., MOOSE) in order to collect the required information and implement control of the stochastic driven events.

The sequence of events is as follows:

1. Add to the input space $\bar{\alpha}_{Staz}$ the discontinuously and stochastically varying variables under assumption 2
2. Expand the phase space to accommodate $\bar{\alpha}_{Brown}$ and the variables represented under assumption 2 and 3
3. Sample the input space
4. Initiate the simulation
5. At each time step evaluate the transition probability for the variable represented under assumption 2 and 3 and the ones subject to Brownian-like behavior
6. Change the set of equations to be solved accordingly
7. Repeat points 5 and 6 until the simulation ending condition is reached
8. Return to point 1 until the prescribed number of samples is reached

To ensure proper non-correlation of the sampling, one seed is provided for all sampling occurring during one single simulation and the same random number generator is recursively interrogated. The following simulations are then seeded differently. Figure 1 summarizes this scheme.

The continuous feedback between the simulation (i.e., RELAP-7) and the evaluation of stochastic events (i.e., RAVEN) requires a high connection between the two codes, which is realized by the common underlying software framework (i.e., MOOSE).

In reality, RELAP-7 provides only the construction of the equations that are solved by the MOOSE engine. RAVEN identifies the location of the variables of interest from RELAP-7 but it interacts with MOOSE for feeding back the changes to the system due to stochastic events and to collect the information from the phase space necessary to evaluate the probability transitions. Such a scheme is represented in Figure 2.
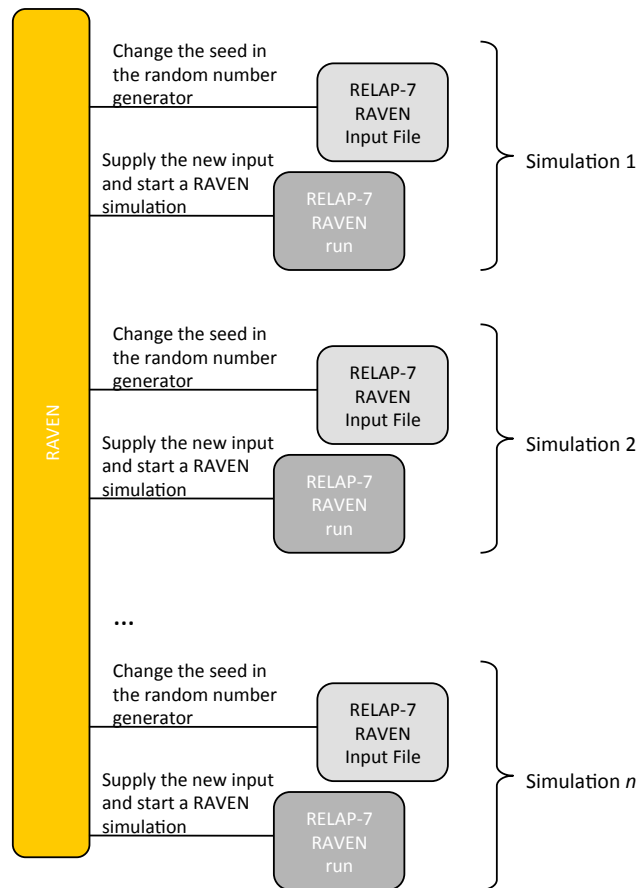
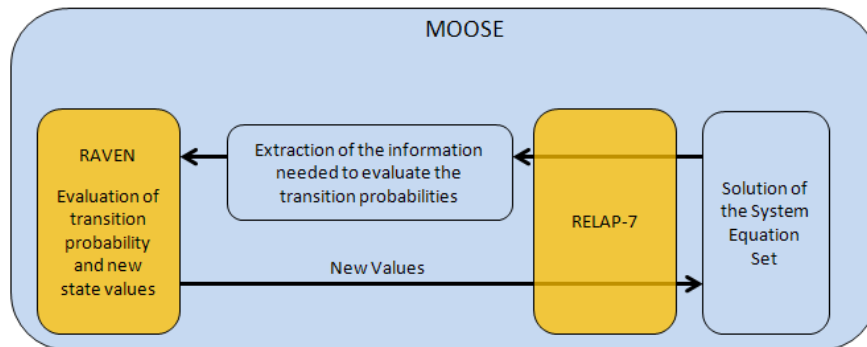**Figure 1: Sampling scheme of the input space for iterate simulations.**

Figure 2: Information generation and exchange for evaluating stochastic driven system changes.

4. EXAMPLES

4.1. Comparison Between Brownian and Constant Uncertainty Propagation

The scope of this section is to show a comparative analysis of the impact of a parameter subject to random distribution but not changing over time (α_{Staz}) or a parameter randomly changing as in Brownian motion like behavior (α_{Brown}). The system considered is a simplified model of material damage accumulation N. The starting equations are:

$$\begin{cases} \dfrac{dN}{dt} = \alpha \\ N(0) = N_0 \end{cases} \tag{10}$$

Since α will take a stochastic behavior it is more mathematically correct to use the integral from:

$$N(t) = \int_0^t \alpha \, dt + N_0 \tag{11}$$

Two different cases are now considered for the behavior of α:
1. $N = N_{Brown}$ having $\alpha = \alpha_{Brown}(t) = \mathbb{N}(\mu, \sigma)$
2. $N = N_{Staz}$ having $\alpha = \alpha_{Staz}(t) = \alpha_{Staz}(t = 0) = \mathbb{N}(\mu, \sigma)$

where $\mathbb{N}(\mu, \sigma)$ indicates a normal distribution having a mean μ and sigma σ.
The probability distribution function r for N_{Brown} and N_{Staz}, respectively the solution for $\alpha = \alpha_{Brow}$ or $\alpha = \alpha_{Staz}$, are provided by the solution of:

$$\begin{cases} \dfrac{\partial r(N_{Brown}, t)}{\partial t} = -\alpha_{Brow} \dfrac{\partial r(N_{Brown}, t)}{\partial N} + \dfrac{\sigma}{2} \dfrac{\partial^2 r(N_{Brown}, t)}{\partial N^2} \\ r(N, 0) = \delta(N_{Brown} - N_0) \end{cases} \tag{12}$$

$$\begin{cases} \dfrac{dN_{Staz}}{dt} = \alpha_{Staz} \\ N(0) = N_0 \end{cases} \tag{13}$$

The analytical solutions of Eq.s 13 and 14 are respectively:

$$r(N_{Brown}, t) = \frac{1}{\sqrt{2\pi\sigma^2 t}} exp\left[-\frac{(N - \mu t)^2}{2\sigma^2 t}\right] \tag{14}$$

$$r(N_{Staz}, t) = \frac{1}{\sqrt{2\pi(\sigma t)^2}} exp\left[-\frac{(N - \mu t)^2}{2(\sigma t)^2}\right] \tag{15}$$

In Figure 3 the probability density function of the two different solutions are plotted (color scale) versus time and damage. Surprising the uncertainty is lower (lower dispersion) when modeling the speed of damage as a Brownian variable.
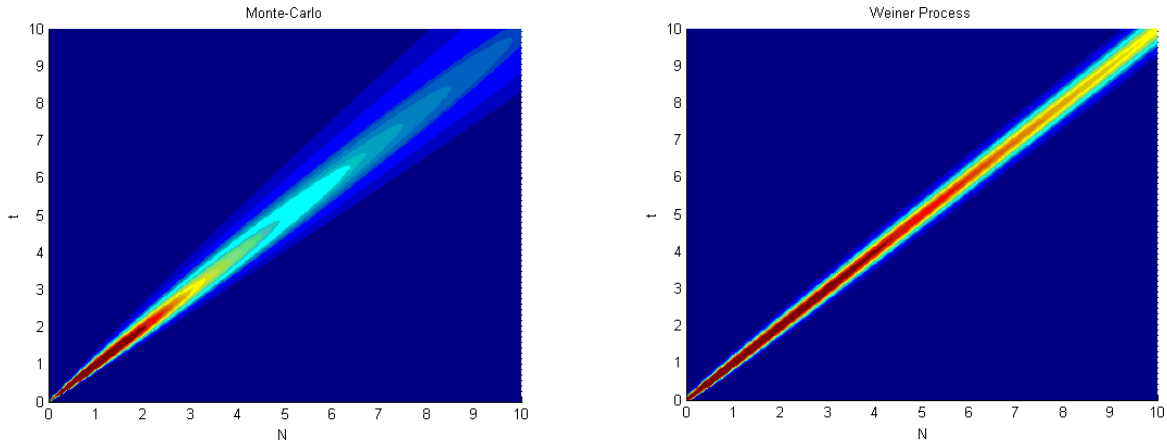
Figure 3: Uncertanty in cumulative damage with $\dot{N}$ costant (left) and continuosly changing (right)

4.2. PWR Demo Application

In this section we consider a simplified PWR system with two primary loops with pumps, heat exchangers and 3 representative core channels. We also consider the following sources of aleatory behavior:

- The thermal conductivity of the fuel in channel 1 is affected by a 1% sigma normal distribution around its nominal value ($\bar{\alpha}_{Staz}$).
- Almost immediately ($\sim 10^{-3}$s) the pump on the leg A is subject to a random noise with normal distribution -1, 0.01- around initial mass flow rate value (α_{Brown}).
- The starting time of the noise is affected by an uncertainty described by a normal distribution (10^{-3}s, 0.1) ($\bar{\alpha}_{Staz}$)
- After 0.1s +/-10% from the beginning of the noise pump A starts to slow down while Pump B ramps up ($\bar{\alpha}_{DS}$ under approximation 2)
- The mass flow rate of pump A during the slowing down is affected by a 1% sigma normal distribution around an exponential decay (α_{Brown}).
- Pump A stabilizes at 80% and Pump B at 120% of the respective initial mass flow rates

According to the scheme described in Section 3.2, the following steps need to be undertaken:
1. The input space is expanded to account for:
   - The time of start of the noise in pump A
   - The starting time delay with respect the beginning of the noise of pump A slowing down (and pump B ramp up)
2. The phase space is expanded to account for:
   - Noise in pump A head before slowing down
   - Noise in pump A head during slowing down
3. The following inputs are sampled:
   - Thermal conductivity of the fuel in channel 1
   - Starting time of the noise in pump A
   - The starting time delay with respect to the beginning of the noise of pump A slowing down (and pump B ramping up)

4. The simulation begins:
   - At each time step the noise is computed
   - At each time step the control logic controls the status of the pumps given the time values previously sampled
5. Simulation ends
6. The input space (point 3) is sampled again and a new simulation started

Figure 4 shows the temporal behavior of the heads of pumps A and B are shown for a single run.
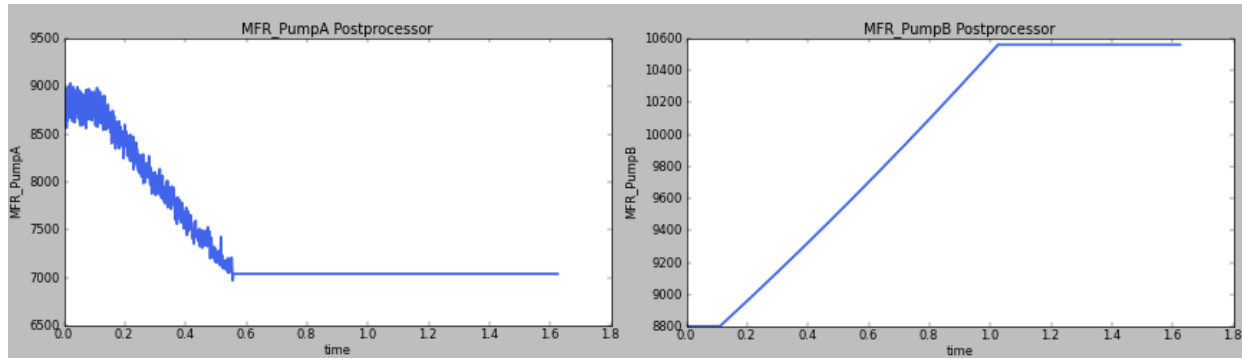

Figure 4: Pumps A (left) and B (right) head temporal behavior

5. CONLCUSIONS

After a careful evaluation of the possible behavior of interest in the analysis of dynamical stochastic system, here presented, we have performed the design and the implementation of the RAVEN software. So far it seems that we are able to cover all the behaviors of interest to perform active modeling of such behavior. As a consequence the twin codes RELAP-7 and RAVEN will be capable in the near future to perform dynamic Probabilistic Safety Analysis on NPPs, which is one of the fundamental steps in the assessment of safety of NPPs life extension. RAVEN is also acting as the plant control logic for RELAP-7. The simple sampling strategy currently implemented will be also soon replaced by more sophisticate adaptive sampling strategies and it parallel implementation on large cluster is already ongoing.

REFERENCES

1. Idaho National Laboratory, "Light Water Reactor Sustainability Research and Development Program Plan (Fiscal Years 2009-2013)", INL/MIS- 08-14918, Rev. 1 (2009).
2. C. Rabiti, A. Alfonsi, J. Cogliati, D. Mandelli, and R. Kinoshita, "Reactor analysis and virtual control environment (RAVEN) FY12 report," Tech. Rep. INL/EXT-12-27351, Idaho National Laboratory (INL) (2012).

3. D. Anders, R. Berry, D. Gaston, R. Martineau, J. Peterson, H. Zhang, H. Zhao, L. Zou, "RELAP-7 Level 2 Milestone Report: Demonstration of a Steady State Single Phase PWR Simulation with RELAP-7," Tech. Rep. INL/EXT-12-25924 (2012).
4. C. Smith, C. Rabiti and R. Martineau, "Risk Informed Safety Margins Characterization (RISMC) Pathway Technical Program Plan," INL/EXT-11-22977, November 2012
5. C. Gardiner, *Stochastic Methods*, Springer (2009).
6. D. Gaston, C. Newman, G. Hansen, D. Lebrun-Grandié, "MOOSE: A parallel computational framework for coupled systems of nonlinear equations," *Nuclear Engineering and Design*, **239**, Issue 10, pp. 1768-1778 (2009).