Methods of Technical Risk Assessment in a Regional Context

- Wolfgang Kröger, Professor and Head of former Laboratory for Safety Analysis (www.lsa.ethz.ch)
  - Founding Rector of International Risk Governance Council Geneva (www.irgc.org)
  - Executive Director, ETH Risk Center (www.riskcenter.ethz.ch)
Advanced Methods for Complex Systems Modeling and Simulation (II)

Object-oriented modeling for the reliability analysis of infrastructure systems

Part I: Short Introduction to Object-oriented Modeling
- Stochastic simulation – basic Monte Carlo methods for reliability analysis
- Object-oriented modeling approach – framework
- How to build an object-oriented model

Part II: Application to Complex Engineering Systems:
Reliability analysis of large-scale electric power systems
- The IEEE Reliability Test System 1996 and its implementation
- Results
Part I: Short Introduction to Object-oriented Modeling
Stochastic simulation – basic Monte Carlo methods for reliability analysis (I)

**Simulation:** an abstraction of a real system by a computer program in order to mimic and analyze its behavior

**Monte Carlo technique:** stochastic simulation using algorithmically generated random numbers

A simple example for estimating the unavailability $Q$ of a system:

Assume a system consisting of $N$ components, where:

- $s_i$: state of the $ith$ component (boolean)
- $Q_i$: failure probability of the $ith$ component
- $R_i$: random number for the $ith$ component; $R_i \sim \text{uniform}[0,1]$

then, assuming independent failures:

$$
\begin{cases}
0 \quad \text{(success)} & \text{if} \quad R_i > Q_i \\
1 \quad \text{(failed)} & \text{if} \quad 0 \leq R_i \leq Q_i
\end{cases}
$$
Stochastic simulation – basic Monte Carlo methods for reliability analysis (II)

1. sample the states of all components („throw the dices“) to get the system state \( s \):

\[ S = \{ s_1, \ldots, s_i, \ldots, s_N \} \]

2. Perform system analysis to judge whether \( s \) is a failure state or not:

\[ x_j = 0 \quad \text{if the system is in the up state} \]
\[ x_j = 1 \quad \text{if the system is in the down state} \]

3. Performing \( k \) system state samples, the unbiased estimate of the system unavailability then is given by:

\[ \bar{Q} = \frac{1}{k} \sum_{j=1}^{k} x_j \]

with variance:

\[ V(\bar{Q}) = \frac{1}{k} V(x) = \frac{1}{k(k - 1)} \sum_{j=1}^{k} (x_j - \bar{Q})^2 \]
Stochastic simulation – sequential Monte-Carlo simulation

Component model

\[
T_i^{\text{in}} = -\frac{1}{\lambda_i} \ln R_i^{\text{in}}
\]

\[
T_i^{\text{out}} = -\frac{1}{\mu_i} \ln R_i^{\text{out}}
\]

Time sequence

a) Component 1

b) Component 2

c) System
Combining Monte Carlo Techniques with Object-oriented modeling

Advantages for reliability analysis:
• Monte Carlo simulation helps to overcome the problem of the **state space explosion**:
  Consider a system of $N=20$ components with two states (e.g. up state and down state). A “state enumeration approach”, such as a “complete” fault tree, or a markovian chain would have to consider $2^N = 2^{20} = 10^6$ system states!

• Object-oriented modeling helps to explicitly consider **time-dependent interactions** between the components and to integrate feedback loops, which is not possible in “static approaches” such as fault tree analysis.

Disadvantages for reliability analysis:
• The simulation primarily aims at calculating mean values. **Some critical scenarios might be missed.**
• Depending on the analyzed system, the **validation** of the simulated system behavior might be a **difficult task**, due to lack of operational experience regarding low-probability-high-impact scenarios.
Object-oriented modeling approach – framework

- Modeling the behaviour of the **components** (objects) and their interaction with the environment
- Stochastic simulation (Monte Carlo methods) of all components to investigate the **macro-behaviour** of the whole system
- In contrary to established methods for risk analysis (ETA, FTA) the observed scenarios and system states are not predefined, but they emerge during the simulation (**emergence**)
- Frequency and consequence of events are determined “**experimentally**”
An object…

- **Has different states** (Finite State Machine, FSM)
- Is capable of interaction with its environment (e.g. other objects)
- has „receptors“ and „effectors“ for specific („messages“) and non-specific (environmental variables) signals
- Can act randomly
- May have a memory (learning)
- Can strive for a goal
Simulation of N objects

- One single object does not tell us much about the behaviour of its macro-system
- Therefore every component of a system has to be modelled separately by an object
- By the computational simulation of all objects, the global system behaviour and the system states s emerge
What can be represented by objects?

- Humans (e.g. operators)
- Components (e.g. turbine)
- Machines (e.g. power station)
- Whole systems (e.g. energy systems)
How to build a simplified object-oriented model for reliability analysis

1. Identify the components of the system
2. Determine the states of each component by making use of FSM, eg:

   ![State Machine Diagram]

   - Up state
   - Down state

3. Determine the transitions between the states and their triggers (e.g. lapse of time or signal from outside)

4. Establish the communication among the objects:

   ![Communication Diagram]

5. Simulate your model to generate the system states s and estimate $\bar{Q}$
Part II: Application to Complex Engineering Systems

Reliability Analysis of Electric Power Systems
Modeling the Electric Power System –
Two-layers approach:
The IEEE Reliability Test System 1996 (RTS `96)

- **Basic system layout**
  - 72 busbars
  - 107 transmission lines
  - 99 generators
  - 51 loads
  - Voltage levels: 230/138 kV
  - Installed capacity: 10‘215 MW (CH: ~12‘000 MW)
  - Peak load: 8‘550 MW (CH: 9‘650 MW)

- **Available data**
  - physical component data: branch reactances, operational thresholds etc.
  - load curves (hourly, daily, weekly)
  - reliability data: component outage and repair rates, min. down times, etc.
The RTS `96 - Implementation in AnyLogic

Objects:
- Lines
- Protection devices
- Busbars
- Generators
- Loads
- Operators (3)
Objects: generators
Objects: system operators
Results:
A) Expected Frequencies of Blackouts

Complementary cumulative blackout frequencies for four different system loading levels $L = 1.0, 1.1, 1.2$ and $1.37$ (circles, stars, triangles and diamonds, respectively) without operator intervention. The error bars indicate the 90% confidence interval.
Results:

B) Blackout causes

Impact of increasing the system loading from $L=1.0$ (dashed line) to $L=1.37$ (continuous line) on the absolute frequencies of blackouts caused by generation inadequacy (left) and system splitting (right).
Results:
C) Influence of the operator response time on the system reliability

Influence of the operator response time on the EENS due to generation inadequacy (left, black bar), operator action (middle, dark-grey bar) and system splitting (right, light-grey bar) for $L=1.37$.

EENS: Expected Energy Not Supplied