Towards guaranteed reliability and resilience computation for complex networks

Leonardo Dueñas-Osorio
Associate Professor
Department of Civil and Environmental Engineering
Rice University
Houston, Texas

Center for Informatics and Computational Science
University of Notre Dame

Notre Dame, Indiana
November 14th, 2018
Motivation (1/3)

- Safety and reliability of complex engineered systems

  Human spaceflight

  Nuclear power

  Lifeline systems
Motivation (2/3)

- Reliability and resilience of socio-technical systems

Smart cities

Deep space habitats
Motivation (3/3)

- Recent national movements towards *community resilience*
Presentation Outline

1. Computational system safety today
2. Methods for reliability-based analysis and design
3. Methods for resilience-based analysis and design
4. New directions for performance assessment
5. Concluding remarks
Main features of modern critical infrastructure systems:

- Large scale and highly exposed systems
- Undergoing aging and deterioration phase of their life cycle
- More interdependent for optimized operation
- Evolving into human-AI-aided interventions
1. System Safety Today (2/8)

- Relevant performance metrics (intrinsic to systems)

- Reliability
- Redundancy
- Robustness
- Special Reliability
  - Class #P
  - Class P
1. System Safety Today (3/8)

- Relevant performance metrics (extrinsic to systems)

   - Reliability
   - Redundancy
   - Robustness
   - Resilience
   - Life-Cycle Cost
   - Sustainability
   - Risk
   - Possibly classes PSPACE or EXP

Decision-Driven Systems
1. System Safety Today (4/8)

- **Qualitative/Quantitative approaches:** Failure modes and event criticality assessment—FMECA
1. System Safety Today (5/8)

- Qualitative/quantitative approaches: Fault Trees (FT)

\[ g(x):= \text{Limit state function} \]

\[ x_1, \ldots, x_8 := \text{Random variables} \]
1. System Safety Today (6/8)

- **Numerical methods**: Monte Carlo simulation (MCS)

\[ g(\mathbf{x}) := \text{Limit state function} \]

\[ x_1, x_2 := \text{Random variables} \]
1. System Safety Today (7/8)

- Numerical/Data-driven methods: MCS and machine learning
1. System Safety Today (8/8)

- **Analytical methods:** First and second order reliability (FORM and SORM)

\[ g(x) := \text{Limit state function} \]

\[ x_1, x_2 := \text{Random variables} \]
Presentation Outline

1. Computational system safety today

2. Methods for reliability-based analysis and design

3. Methods for resilience-based analysis and design

4. New directions for performance assessment

5. Concluding remarks
2. Computational Reliability (1/5)

- Combinatorial (Boolean) approaches

\[ K = \begin{pmatrix} C_1 & \overline{C}_1 & C_1 & \overline{C}_1 \\ C_2 & \overline{C}_2 & C_2 & \overline{C}_2 \end{pmatrix} \]
2. Computational Reliability (1/5)

- Combinatorial (Boolean) approaches

- Any failure configurations

- Complete failure configuration

\[ K = \begin{bmatrix} C_1 & C_1 \bar{C}_1 & C_1 \bar{C}_1 & C_1 \bar{C}_1 \\ C_2 & C_2 \bar{C}_2 & C_2 \bar{C}_2 & C_2 \bar{C}_2 \end{bmatrix} \]
2. Computational Reliability (2/5)

- Combinatorial approaches: Example wind turbine (WT)

$$K_{3x2^3} = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$
2. Computational Reliability (2/5)

- **Combinatorial approaches:** Example wind turbine (WT)

![Wind Turbine Diagram]

$$K_{3 \times 2^3} = \begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 1
\end{bmatrix}$$

$$P(\text{NC} = 2) = \sum_{k^* \in K_2 \subset K} \prod_{i=1}^{3} P_i^{k^*_i} (1 - P_i)^{1-k^*_i}$$

**NC:** Number of failed components

**\( P_i \):** \( i \)th component failure probability

$$k^*_i = \begin{cases} 
1 & \text{if component } i \text{ fails} \\
0 & \text{otherwise}
\end{cases}$$
2. Computational Reliability (3/5)

- Combinatorial approaches: WT failure probability mass function (PMF)

\[
K_{3 \times 3} = \begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 1
\end{bmatrix}
\]

\[
P(\text{NC} = s) = \sum_{k^* \in K_s \subset K} \prod_{i=1}^{N} P_i^{k^*} (1 - P_i)^{1-k^*}
\]

\[O(2^N) \forall s / N\]
2. Computational Reliability (4/5)

- **Combinatorial approaches:** Recursive strategies

\[
\binom{N}{s} = \binom{N}{s-1} + \binom{N-1}{s-2}
\]

Known recursion for binomial coefficient
2. Computational Reliability (4/5)

- **Combinatorial approaches:** Recursive strategies

\[
\binom{N}{s} = \binom{N}{s-1} + \binom{N-1}{s-2}
\]

Known recursion for binomial coefficient

Generalize binomial recursion to sets:

\[
K = \begin{cases}
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\end{cases}
\quad K_1 \\
\begin{cases}
1 & 1 & 0 & 1 \\
1 & 0 & 1 & 1 \\
0 & 1 & 1 & 1 \\
\end{cases}
\quad K_2
\]

\[
R(2, K_2) = \{R(2, K_2(1)) + \alpha_1 R(1, K_1(1))\}
\]

\[O(N^2) \forall s\]
2. Computational Reliability (5/5)

- **Combinatorial approaches:** Computational complexity
Presentation Outline

1. Computational system safety today
2. Methods for reliability-based analysis and design
3. Methods for resilience-based analysis and design
4. New directions for performance assessment
5. Concluding remarks
3. Resilience-Driven Methods (1/9)

- Minimize the cost of interdependent system restoration
3. Resilience-Driven Methods (2/9)

- Minimize the cost of interdependent system restoration

Minimize

\[
\sum_{t \in T | t > 0} \left( \sum_{s \in S} g_{st} \Delta z_{st} \right) + \sum_{k \in \mathcal{K}} \left( \sum_{(i,j) \in \mathcal{A}'} f_{ijkt} \Delta y_{ijkt} \right) + \sum_{i \in \mathcal{N}_k'} q_{iklt} \Delta w_{iklt} \\
+ \sum_{t \in T} \sum_{k \in \mathcal{K}} \left( \sum_{l \in \mathcal{L}_k} \sum_{i \in \mathcal{N}_k} \left( M_{iklt}^+ \delta_{iklt}^+ + M_{iklt}^- \delta_{iklt}^- \right) + \sum_{(i,j) \in \mathcal{A}_k} c_{ijkt} x_{ijkt} \right)
\]

Interdependent network design problem (INDP)
3. Resilience-Driven Methods (3/9)

- **Solution strategies for i-INDP:** Simulation and decompositions

**COUPLING CONSTRAINTS**
- Shared resources
- Interdependence
- Co-location

**DECOUPLED CONSTRAINTS**
For each network:
- Adjacency
- Flow balance
3. Resilience-Driven Methods (4/9)

- **Solution strategies for td-INDP**: Simulation and decompositions

<table>
<thead>
<tr>
<th>Recovery variables</th>
<th>Element functionality, flow of commodities, and over/under supply</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recovery variables</td>
<td>Element functionality, flow of commodities, and over/under supply</td>
</tr>
</tbody>
</table>

- Simulation
- Decompositions

- **Coupling constraints**:
  - Time

- **Graphical representation**

  - $t = 0$
  - $t = 1$
  - $t = 2$
  - $t = N$
3. Resilience-Driven Methods (5/9)

- **Solution strategies for s-INDP**: Simulation and decompositions

td-INDP

sINDP
3. Resilience-Driven Methods (6/9)

- **Example application:** Earthquakes in Memphis, TN

Shelby County, TN
- Power network
- Gas network
- Water network

Subject to earthquakes (New Madrid Seismic Zone)
- Epicenter 35.4 N – 90.3 W (33km from Memphis)
- Analyzed magnitudes from $M_w = 6$ to $M_w = 9$
3. Resilience-Driven Methods (7/9)

- **Example application:** Networks in Memphis, TN

![Networks in Memphis, TN](image)
3. Resilience-Driven Methods (8/9)

- **s-INDP results**: Performance recovery for $M_w = 6.5$

![Graphs](image)

(a) Balanced system  
(Supply surplus = 0%)

(b) Excess of supply capacity  
(Supply surplus = 10%)
3. Resilience-Driven Methods (9/9)

- **s-INDP results**: Value of stochastic information

(a) Using only the expected demands (Supply surplus = 10%)

(b) Assuming perfect information (Supply surplus = 10%)
1. Computational system safety today

2. Methods for reliability-based analysis and design

3. Methods for resilience-based analysis and design

4. New directions for performance assessment

5. Concluding remarks
4. New Computational Paths (1/10)

- The NP and \#P revolutions: Oracles in practice
4. New Computational Paths (2/10)

- A satisfiability (#SAT) revolution: CNF’s

\[ f(x)_{EC} = (x_1 \lor x_2 \lor x_6) \land (x_1 \lor x_3 \lor x_4) \land (x_2 \lor x_3 \lor x_5) \land (x_4 \lor x_5 \lor x_6) \]
4. New Computational Paths (2/10)

- A satisfiability (#SAT) revolution: CNF’s

\[ f(x)_{EC} = (x_1 \lor x_2 \lor x_6) \land (x_1 \lor x_3 \lor x_4) \land (x_2 \lor x_3 \lor x_5) \land (x_4 \lor x_5 \lor x_6) \]
4. New Computational Paths (3/10)

- A satisfiability (#SAT) revolution: Existential CNF’s

\[
f(x)_{UNREL} = (\exists S)_{S \subseteq C} \left[ \left( \bigvee_{j \in C} c_j \right) \land \left( \bigvee_{k \in C} \neg c_k \right) \land \left( \bigvee_{x \in E} L_{x_i} \right) \right]
\]

\[
L_{x_i} = (c_u \land x_i \rightarrow c_v) \land (c_v \land x_i \rightarrow c_u), \forall x_i \in E
\]
4. New Computational Paths (4/10)

- A satisfiability (#SAT) revolution: Results

Size $N \times N$ and edge failure probabilities $p = 1/2^i$. 

![Diagram showing the relationship between size parameter $N$ and CPU time](chart)
4. New Computational Paths (5/10)

- A satisfiability (#SAT) revolution: Results

![Graphs showing CPU Time vs. Size parameter N for RelNet and Karger methods with different δo values and εo values for various i values.]
4. New Computational Paths (6/10)

- **Quantum Computation**: Quantum Boolean tensor networks
4. New Computational Paths (7/10)

- **Quantum Computation:** Quantum Boolean tensor networks

\[ \psi_f = \sum_x |x\rangle \langle f(x) | 1 \alpha \rangle = \sum_x f(x) |x\rangle \]

Quantum Boolean state superposition
4. New Computational Paths (8/10)

- Quantum simulation results: Bounds to reliability as a counting problem

![Graph showing the relationship between time (seconds) and N for different counting problems.](image)
4. New Computational Paths (9/10)

- **IBM-Q:** Experiments with Hamiltonian dynamics
  - SAT formula
  
  \[ f(x) = (x_1 \lor x_2) \land (x_2 \lor x_3) \land (x_3 \lor x_1) \]
  
  - Hamiltonians
  
  \[ \hat{H}_X = -\sum_{i=1}^{3} \sigma_i^x \]
  
  \[ \hat{H}_Z = \frac{1 - \sigma_1^z}{2} \frac{1 - \sigma_2^z}{2} + \frac{1 - \sigma_2^z}{2} \frac{1 - \sigma_3^z}{2} + \frac{1 - \sigma_3^z}{2} \]

  - Annealing
  
  \[ \hat{H}(t) = \lambda_X(t)\hat{H}_X + \lambda_Z(t)\hat{H}_Z \]
4. New Computational Paths (10/10)

- IBM-Q: Experiments with 12 q-bits
5. Conclusions

- Engineering reliability and resilience are PAC computable!

- Algorithmic methods with guarantees of optimality or quality are essential to inform AI decision making

- Hardware-based reliability and resilience quantification is attainable, given breakthroughs in SAT and quantum computation
Thank you!

Questions?