# Mitigating Global Failure Regimes in Large Distributed Systems (*Ongoing Research*)

C. Dabrowski, J. Filliben, K. Mills, S. Ressler and B. Rust from NIST

***No effective methods exist to predict failure regimes in large distributed systems—the search space is large and causality is difficult to establish. Our research goals are to: (1) develop design-time methods that system engineers can use to detect existence and causes of costly failure regimes prior to system deployment and (2) develop run-time methods that system managers can use to detect onset of costly failure regimes in deployed systems, prior to collapse.***

## *First Hard Problem*

Complex information systems encompass an infeasible search space.

$$y_1, \ldots, y_m = f(\, x_{1|[1,\ldots,k]}, \ldots, x_{n|[1,\ldots,k]}\,)$$

System Response Space — System Parameter Space

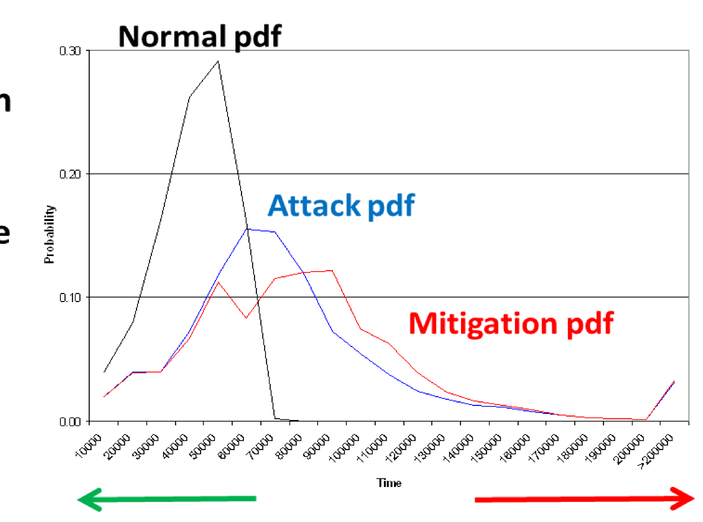
System Parameter Space = $k^x$, e.g., for $k$ of $2^{32}$ and $x$ of 1000, = $(2^{32})^{1000}$ = $O(10^{9633})$

Atoms in the visible universe = about $10^{80}$

## *Second Hard Problem*

Determining causality is difficult given only patterns of global system behavior.

For example, unexpected collapse in the mitigation probability density function of job completion times in a computing grid was unexplainable without more detailed data and analysis.




## *Approach One: Combine Markov Models, Graph Analysis and Perturbation Analysis*

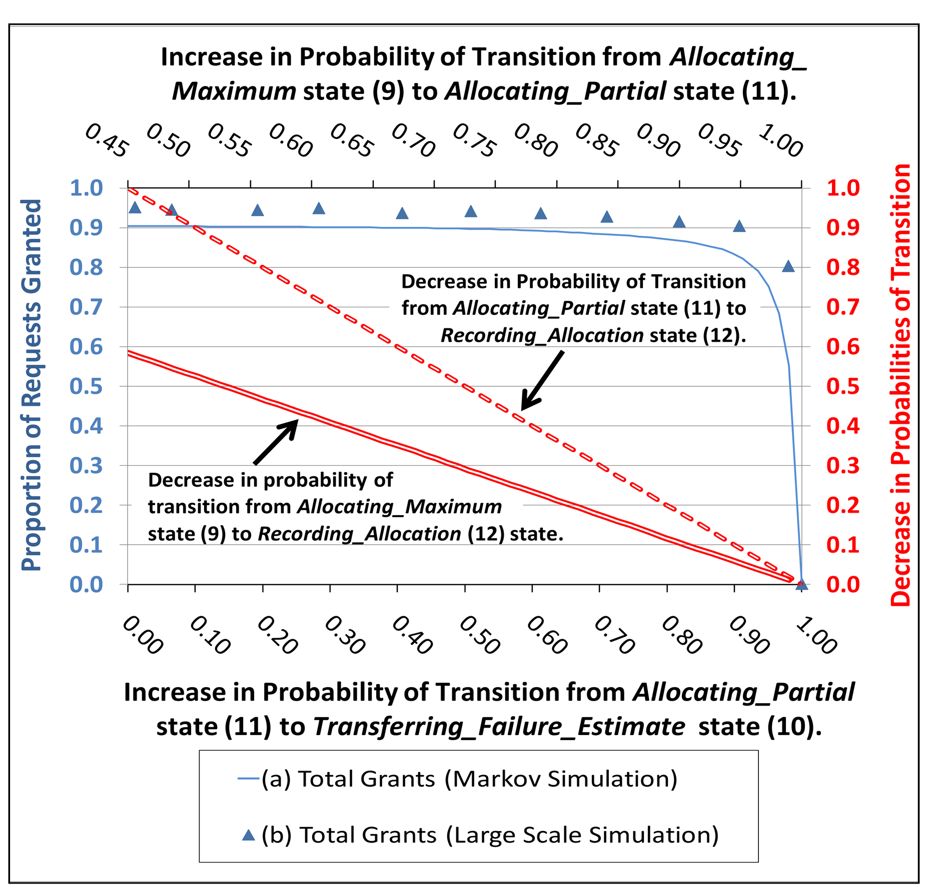
*Innovations in Measurement Science*

**Using simulated failure scenarios in a Markov chain model to predict failures in a Cloud**

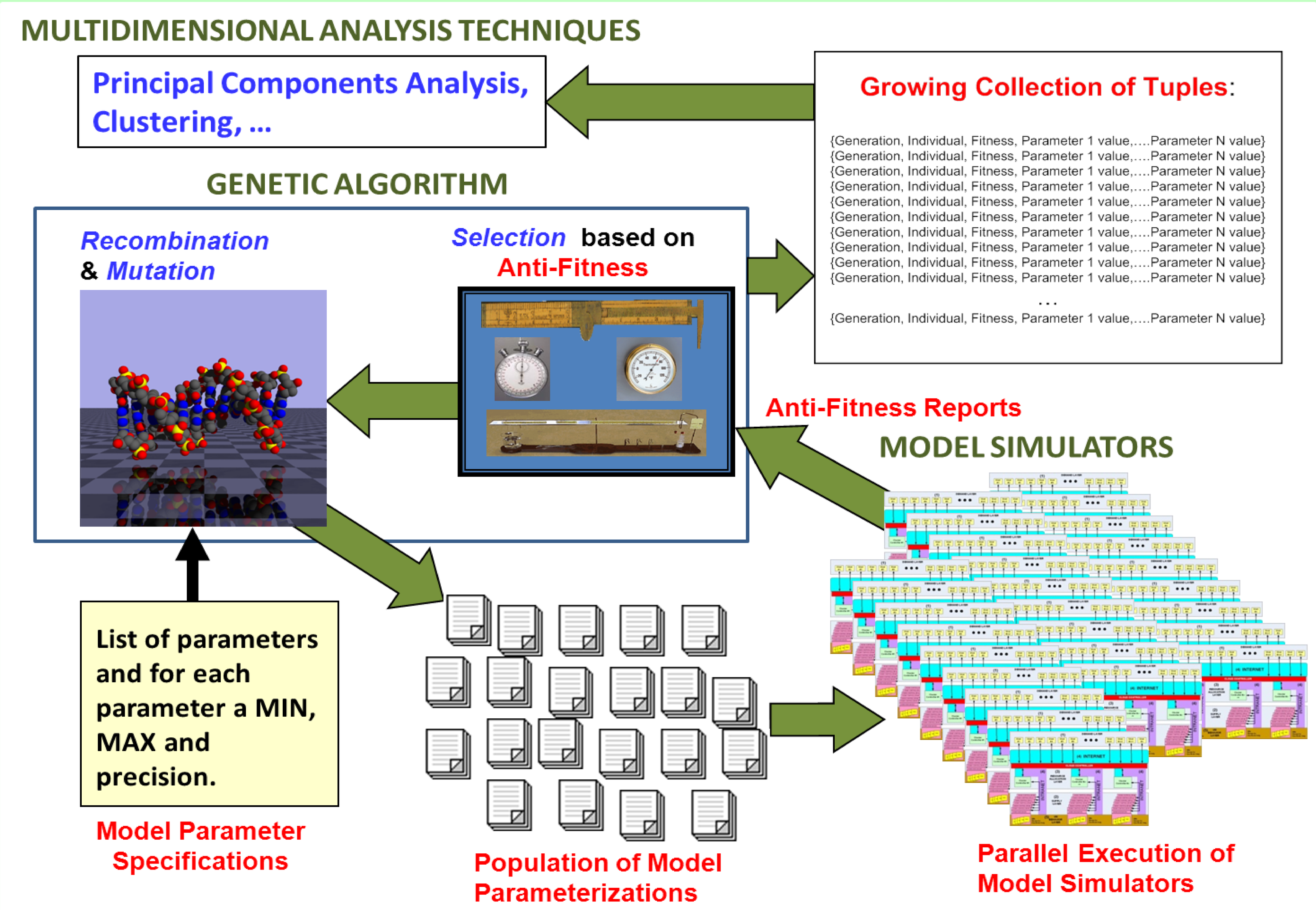
Example: Markov simulation and perturbation of ***a minimal s-t cut set*** of a Markov chain graph:

- Corresponds to software failure scenario involving multiple faults/attacks.
- Simulation identifies threshold beyond which increased failure incidence causes drastic performance collapse

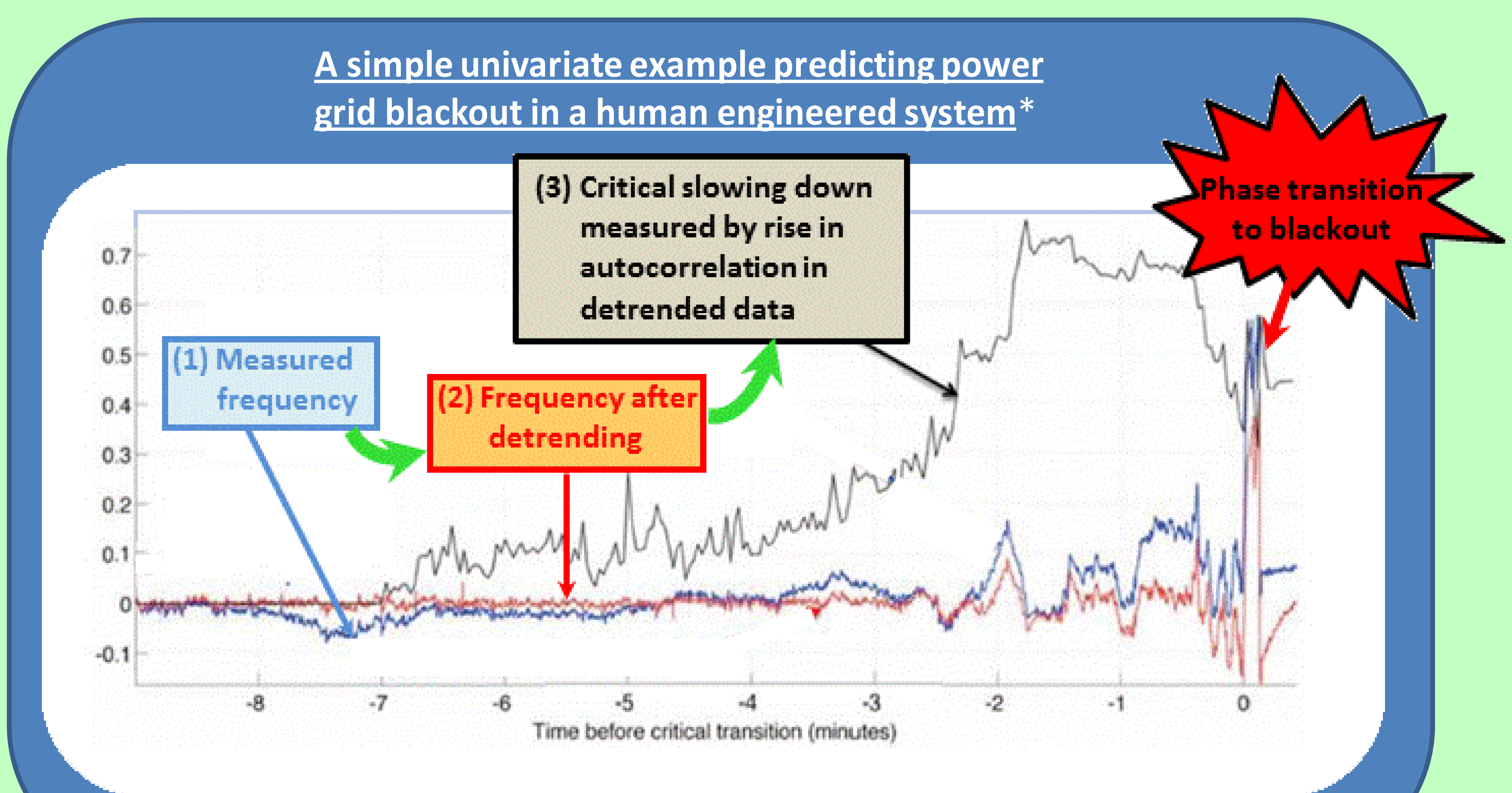
→ ***Verified in target system being modeled (i.e., Koala, a large-scale simulation of a Cloud)***




## *Approach Two: Combine Anti-Optimization and Genetic Algorithms*




## *Approach Three: Measuring Key System Properties such as Critical Slowing Down*




*From P. Hines, E. Cotilla-Sanchez, and S. Blumsack. Topological Models and Critical Slowing Down: Two Approaches to Power System Risk Analysis. Proceedings of the 44th Hawaii Conference on System Sciences. IEEE Computer Society, Washington, DC, USA, pp. 1-10.

## *How might your organization benefit from collaborating with us?*

- **IF your organization designs and deploys Clouds and other large distributed systems AND**
  - **You wish to improve the reliability of your system AND**
  - **You have a model of your system OR**
  - **You are willing to share sufficient information for us to construct a model AND you are willing to help us ensure our model suitability represents your system**
- **THEN working together we could help you improve the reliability of your system (or specific aspects of your system) by:**
  - **Applying our design-time methods to search the design space for potential collapse scenarios (and iterating on any proposed design revisions you create to mitigate collapse scenarios) AND/OR**
  - **Exploring run-time monitoring and measurement approaches that could signal incipient onset of collapse scenarios that were not detected using our design-time methods**

**WIN-WIN:** ***we would gain additional evaluation and refinement of our methods and you could gain a transfer of our technology to enhance your design process.***

***http://www.nist.gov/itl/antd/emergent_behavior.cfm***