THE OHIO STATE UNIVERSITY

# The Backtracking Process Algorithm: A Dynamic Probabilistic Risk Assessment Method for Autonomous Vehicle Control Systems

**Mohammad Hejase**

**Arda Kurt · Tunc Aldemir · Umit Ozguner**

# Motivation

- Autonomous vehicle operations are set to expand in the National Airspace and National Highway System.
- Vehicles need to be equipped with controllers that have the capability of safely completing tasks and reacting to sub-nominal situations.

https://c8.alamy.com/comp/JJMFB4/smart-car-hud-and-autonomous-self-driving-mode-vehicle-on-metro-city-JJMFB4.jpg

https://www.extremetech.com/wp-content/uploads/2014/09/self-driving-head-640x353.jpg

- Such situations can lead to a degraded performance, causing the occurrence of hazards or accidents.
- Safety and Risk Assessment techniques need to be developed to identify risk-significant event sequences leading to hazardous situations.

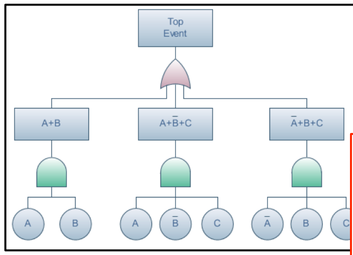
- Physical testing of all emerging autonomous technologies is too expensive and time-consuming.
- Alternatives to physical testing need to be developed to ensure safe operation of autonomous functions.
- Dynamic Probabilistic Risk Assessment techniques can provide safety assurance by utilizing model-based designs of autonomous systems.

https://financialtribune.com/sites/default/files/field/image/shahrivar1/11_NM_Auto%20quality%20-%20500.jpg

# Risk Analysis Techniques

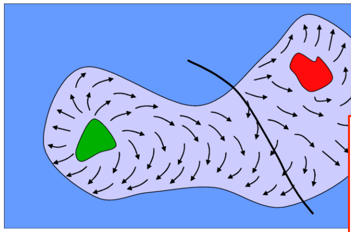
- Quantitative analysis (QA) methods are typically used for estimating likelihoods of violating safety goals under certain system failures.

- Some common QA tools in industry:
  - Fault Tree Analysis
  - Event Tree Analysis
  - Failure Mode and Effects Analysis
  - Reliability Block Diagrams



*Have challenges in expressing events that include changes to system dynamics*

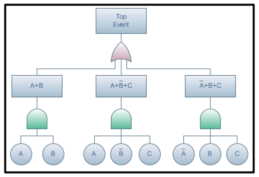
- In control systems literature, reachability analysis is a commonly used assurance method.



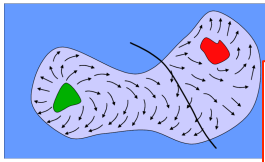
Colbaugh, Rich et al. "Some intelligence analysis problems and their graph formulations." J. Intelligence Community Research and Development (2010).

*Has challenges when incorporating system component changes, or high fidelity system models that cannot be expressed in analytical form*

- Dynamic Probabilistic Risk Assessment (DPRA) approaches are capable of providing frameworks that allow considering epistemic and aleatory uncertainties in physical processes and system safety responses.



+



*Allows expressing changes in system dynamics and system configuration*

- The Markov Cell to Cell Mapping technique is one such DPRA method

# Markov/CCMT History

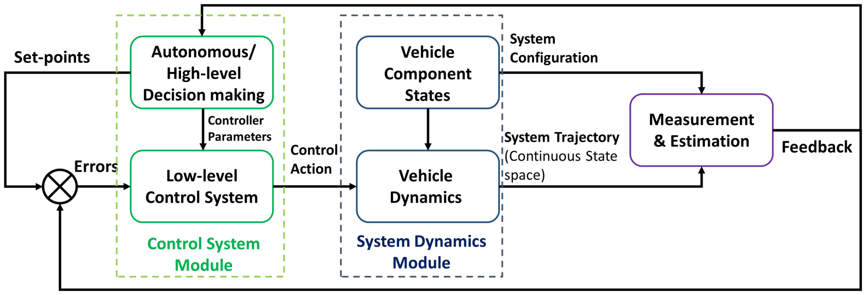
- **1980** – Theory of CCMT introduced by Hsu.
- **1987** – A Markovian interpretation of CCMT proposed by Aldemir
- **1990** – Failure analysis performed on closed loop control system of a process plant by utilizing databases to represent system dynamics
- **1996** – Continuous Markov/CCMT developed
- **2006** – Markov/CCMT utilized for PRA of control systems of nuclear reactors.
- **2016** – BPA proposed by Yang and Aldemir and demonstrated on a level control system.

**Funded by NASA Ames (SSAT) / NSF CPS**

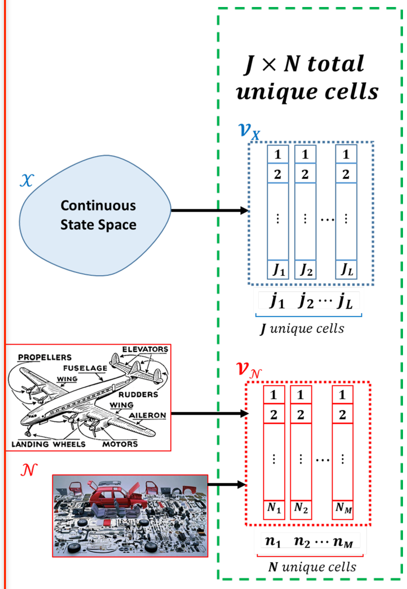
- **2017** – BPA was used to identify risk-significant scenarios leading to hazardous events for unmanned aircraft control systems.
- **2018** – BPA was used to identify risk-significant scenarios leading to hazardous events for unmanned ground vehicle control systems.
- **2018** – Concept of S-BPA introduced for risk-analysis of multi-phased control systems equipped with contingency actions.

# General System-Level Description of AVs

- An autonomous vehicle (AV) is one that can guide itself without human conduction.
- This requires design and implementation of high-level decision making techniques, low-level control systems, estimation algorithms, etc.
- The system is composed of a continuous state space (position, velocity, rates, angles, etc.) and a discrete state space (component fault modes, operation modes, etc.).



## Cell Space Creation



Continuous $L$ dimensional state space

$\mathcal{X} \triangleq \mathbb{R}^L$

↓

Partition Variables

↓

$x_l \ \rightarrow \{j_l \mid j_l = 1,2,\cdots,J_l\}$
$l = 1,\cdots L$

$\mathcal{V}_\mathcal{X}$

Discrete M system components

$\mathcal{N} \triangleq \mathbb{Z}^M$

↓

Define System Config. Modes

↓

$n_m \ \rightarrow \{n_m \mid n_m = 1,2,\cdots,N_m\}$
$m = 1,\cdots M$

$\mathcal{V}_\mathcal{N}$

$\mathcal{V} \triangleq \mathbb{Z}^{L+M}$

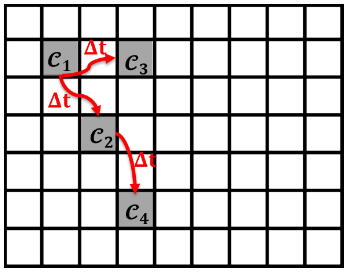
Cell space $\mathcal{V}$ is composed of $J \times N$ unique cells

**Each cell in the cell space is represented by an $(L + M)$ dimensional vector**

$[\boldsymbol{j}\ \boldsymbol{n}] \equiv [\boldsymbol{j_1}, \dots, \boldsymbol{j_l}, \cdots, \boldsymbol{j_L}, \boldsymbol{n_1}, \dots, \boldsymbol{n_m}, \cdots, \boldsymbol{n_M}]$

System evolution is represented by discrete-time transitions over $\Delta t$.

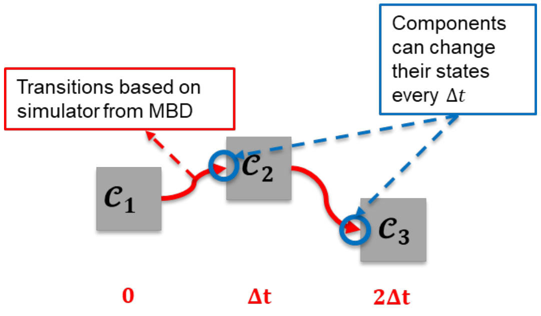
Transitions are determined from system dynamics, controller behavior, and fault behavior.



**Assumptions**:

**1)** System component configurations are fixed over $[t, t + \Delta t)$.

**2)** System is autonomous (if not, augment time to state-space).


Transitions based on simulator from MBD

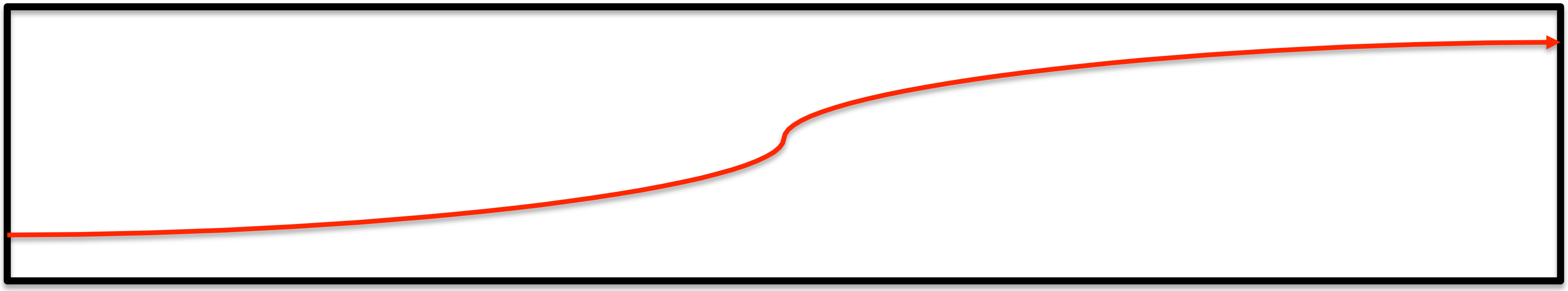
Components can change their states every $\Delta t$

# An Illustrative Explanation of Markov/CCMT

## Monte Carlo

Initial Conditions

System State Evolution under fixed component conditions

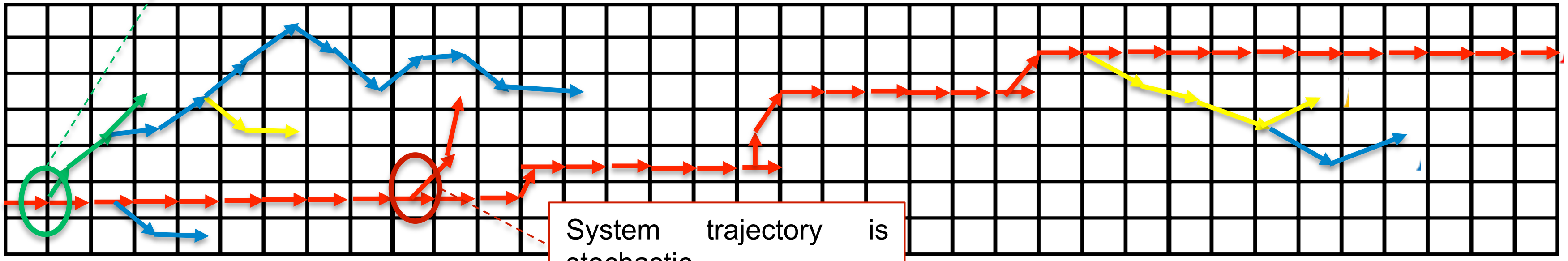


## Markov/CCMT

Initial Conditions

Components can change states along the way. (e.g.: component failure, or gear change, etc.)

System State Evolution under possibly changing component conditions with a path probability



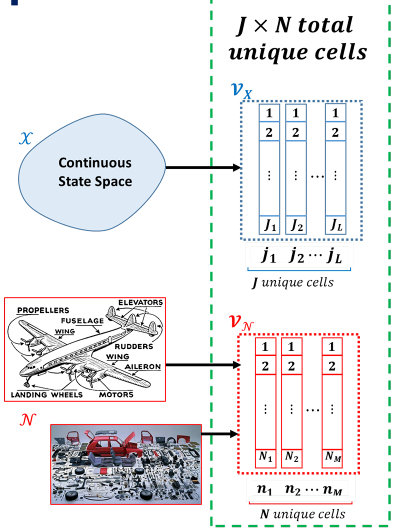
System trajectory is stochastic

# The Deductive Markov Cell to Cell Mapping Technique

System continuous states and discrete component states are partitioned into finite cells.

A cell space is then constructed from the partitioned variables.

1

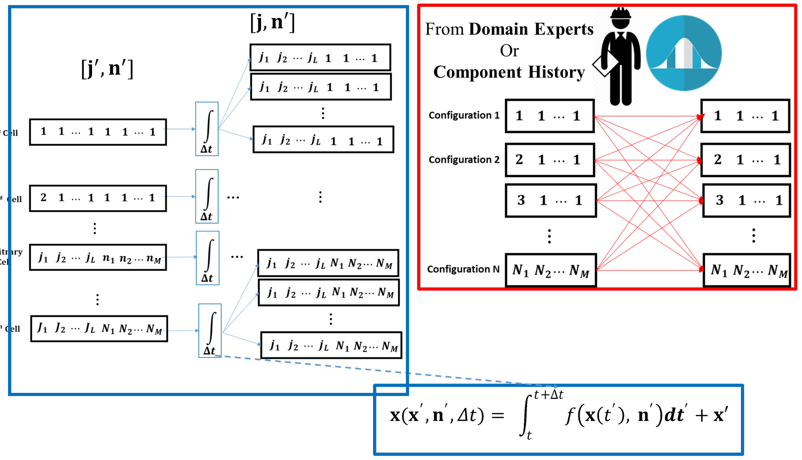


2

Transition probability from cell $\mathbf{j}'$ to $\mathbf{j}$ over $\Delta t$ under configuration $\mathbf{n}'$

$$q(\boldsymbol{j}, \mathbf{n} | \, \mathbf{j}', \mathbf{n}', \Delta \mathrm{t}) = g(\mathbf{j}|\mathbf{j}', \mathbf{n}', \Delta t) \times h(\mathbf{n}|\mathbf{n}', \mathbf{j}' \rightarrow \mathbf{j}, \Delta t)$$

System configuration transition probabilities over $\Delta t$


$[\mathbf{j}', \mathbf{n}']$
$[\mathbf{j}, \mathbf{n}']$
1st Cell
2nd Cell
arbitrary Cell
JxNth Cell
$1 \; 1 \cdots 1 \; 1 \; 1 \cdots 1$
$2 \; 1 \cdots 1 \; 1 \; 1 \cdots 1$
$j_1 \; j_2 \cdots j_L \; n_1 \; n_2 \cdots n_M$
$J_1 \; J_2 \cdots J_L \; N_1 \; N_2 \cdots N_M$
$\int_{\Delta t}$
$j_1 \; j_2 \cdots j_L \; 1 \; 1 \cdots 1$
$j_1 \; j_2 \cdots j_L \; N_1 \; N_2 \cdots N_M$
From **Domain Experts** Or **Component History**
Configuration 1
Configuration 2
Configuration N
$1 \; 1 \cdots 1$
$2 \; 1 \cdots 1$
$3 \; 1 \cdots 1$
$N_1 \; N_2 \cdots N_M$

$$\mathbf{x}(\mathbf{x}', \mathbf{n}', \Delta t) = \int_t^{t+\Delta t} f(\mathbf{x}(t'), \mathbf{n}')dt' + \mathbf{x}'$$

System evolution for each cell in the cell space is obtained over $\Delta t$.

System component transition probabilities over $\Delta t$ are obtained from domain experts or component data (history)

A cell to cell mapping is constructed for the whole cell space.

Such a map can be used to construct possible system trajectories

3

System evolution in forward time:

$$\boldsymbol{P}^{K+1} = \boldsymbol{Q}\boldsymbol{P}^{K}$$

$\boldsymbol{P}^k$: **Probability of system being at** $[\mathbf{j} \; \mathbf{n}]$ at time *t=kΔt*

$\boldsymbol{P}^{k+1}$: **Probability of system being at** $[\mathbf{j}' \; \mathbf{n}']$ at time *t=(k+1)Δt*

$$\mathbf{Q} \quad \begin{pmatrix} q(\mathbf{1},\mathbf{1}|\,\mathbf{1},\mathbf{1},\Delta\mathrm{t}) & q(\mathbf{1},\mathbf{1}|\,\mathbf{2},\mathbf{1},\Delta\mathrm{t}) & \cdots & q(\mathbf{1},\mathbf{1}|\,\mathbf{J},\mathbf{N},\Delta\mathrm{t}) \\ q(\mathbf{2},\mathbf{1}|\,\mathbf{1},\mathbf{1},\Delta\mathrm{t}) & & & \\ q(\mathbf{3},\mathbf{1}|\,\mathbf{1},\mathbf{1},\Delta\mathrm{t}) & \ddots & & \\ \vdots & & & \\ q(\boldsymbol{J},\mathbf{1}|\,\mathbf{1},\mathbf{1},\Delta\mathrm{t}) & & & \\ q(1,\mathbf{2}|\,\mathbf{1},\mathbf{1},\Delta\mathrm{t}) & & \ddots & \vdots \\ \vdots & & & \\ q(\mathrm{J},\mathbf{2}|\,\mathbf{1},\mathbf{1},\Delta\mathrm{t}) & & & \\ \vdots & & & \ddots \\ q(\mathrm{J},\mathbf{N}|\,\mathbf{1},\mathbf{1},\Delta\mathrm{t}) & & \cdots & q(\mathbf{J},\mathbf{N}|\,\mathbf{J},\mathbf{N},\Delta\mathrm{t}) \end{pmatrix}$$

4

A deductive search for paths leading to a Top Event of interest can be accomplished through:

$$\mathbf{P^k} = [\mathbf{Q^T Q}]^{-1}\mathbf{Q^T P^{k+1}}$$

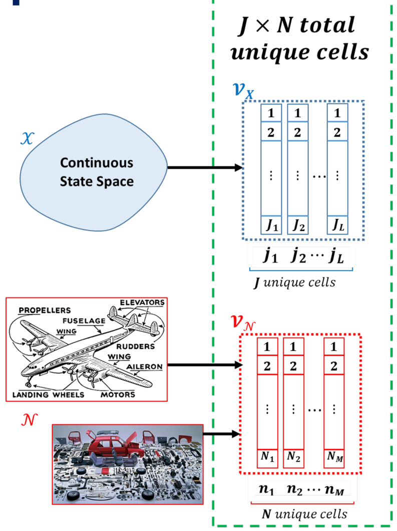
The cell to cell map can be used to deductively identify paths that lead to a Top Event (represented by a collection of cells)

# The Deductive Markov Cell to Cell Mapping Technique

System continuous states and discrete component states are partitioned into finite cells.

A cell space is then constructed from the partitioned variables.

1

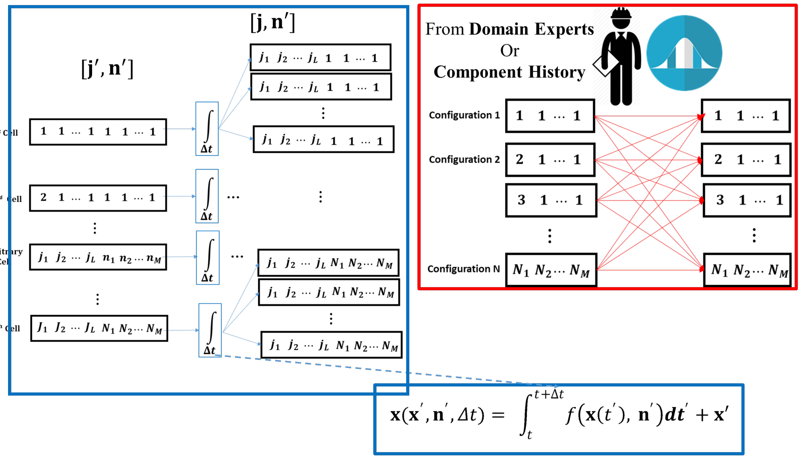


**Challenge 1:** Forward integration of the system cells is generally not a trivial task for the case of autonomous vehicles.

2 Transition probability from cell $\mathbf{j}'$ to $\mathbf{j}$ over $\Delta t$ under configuration $\mathbf{n}'$

$$q(\boldsymbol{j},\mathbf{n}|\,\mathbf{j}',\mathbf{n}',\Delta t)= g(\mathbf{j}|\mathbf{j}',\mathbf{n}',\Delta t)\ \times\ h(\mathbf{n}|\mathbf{n}',\mathbf{j}'\rightarrow\mathbf{j},\Delta t)$$

System configuration transition probabilities over $\Delta t$



$[\mathbf{j}',\mathbf{n}']$
$[\mathbf{j},\mathbf{n}']$
1st Cell
2nd Cell
arbitrary Cell
JxNth Cell
1 1 … 1 1 1 … 1
2 1 … 1 1 1 … 1
$j_1\ j_2 \cdots j_L\ n_1\ n_2 \cdots n_M$
$J_1\ J_2 \cdots J_L\ N_1\ N_2 \cdots N_M$
$\int_{\Delta t}$
$j_1\ j_2 \cdots j_L\ 1\ 1 \cdots 1$
$j_1\ j_2 \cdots j_L\ N_1\ N_2 \cdots N_M$
From Domain Experts
Or
Component History
Configuration 1
Configuration 2
Configuration N
1 1 … 1
2 1 … 1
3 1 … 1
$N_1\ N_2 \cdots N_M$

$$\mathbf{x}(\mathbf{x}',\mathbf{n}',\Delta t)=\int_{t}^{t+\Delta t} f(\mathbf{x}(t'),\,\mathbf{n}')\boldsymbol{dt}'+\mathbf{x}'$$

System evolution for each cell in the cell space is obtained over $\Delta t$.

System component transition probabilities over $\Delta t$ are obtained from domain experts or component data (history)

A cell to cell mapping is constructed for the whole cell space.

Such a map can be used to construct possible system trajectories

3 System evolution in forward time:

$$\boldsymbol{P}^{\boldsymbol{K}+\mathbf{1}}=\boldsymbol{QP}^{\boldsymbol{K}}$$

$\boldsymbol{P}^{k}$: **Probability of system being at** $[\mathbf{j}\ \mathbf{n}]$ at time *t=kΔt*

$\boldsymbol{P}^{k+1}$: **Probability of system being at** $[\mathbf{j}'\ \mathbf{n}']$ at time *t=(k+1)Δt*

$$\mathbf{Q}\quad\begin{pmatrix} q(\mathbf{1},\mathbf{1}|\,\mathbf{1},\mathbf{1},\Delta t) & q(\mathbf{1},\mathbf{1}|\,\mathbf{2},\mathbf{1},\Delta t) & \cdots & q(\mathbf{1},\mathbf{1}|\,\mathbf{J},\mathbf{N},\Delta t)\\ q(\mathbf{2},\mathbf{1}|\,\mathbf{1},\mathbf{1},\Delta t) & & & \\ q(\mathbf{3},\mathbf{1}|\,\mathbf{1},\mathbf{1},\Delta t) & \ddots & & \\ \vdots & & & \\ q(\boldsymbol{J},\mathbf{1}|\,\mathbf{1},\mathbf{1},\Delta t) & & & \\ q(1,\mathbf{2}|\,\mathbf{1},\mathbf{1},\Delta t) & & & \vdots\\ \vdots & & \ddots & \\ q(\mathrm{J},\mathbf{2}|\,\mathbf{1},\mathbf{1},\Delta t) & & & \\ \vdots & & \ddots & \\ q(\mathrm{J},\mathbf{N}|\,\mathbf{1},\mathbf{1},\Delta t) & & \cdots & q(\mathbf{J},\mathbf{N}|\,\mathbf{J},\mathbf{N},\Delta t) \end{pmatrix}$$

**Challenge 2:** Q matrix is very large and expensive to store

4 A deductive search for paths leading to a Top Event of interest can be accomplished through:

$$\mathbf{P}^{\mathbf{k}}=[\mathbf{Q}^{\mathbf{T}}\mathbf{Q}]^{-\mathbf{1}}\mathbf{Q}^{\mathbf{T}}\mathbf{P}^{\mathbf{k}+\mathbf{1}}$$

The cell to cell map can be used to deductively identify paths that lead to a Top Event (represented by a collection of cells)

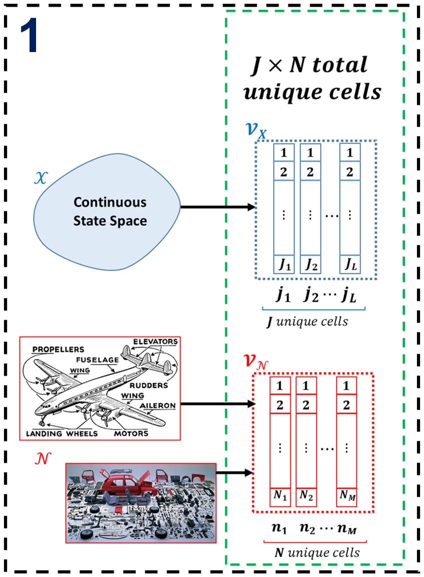
**Challenge 3:** $Q^{T}Q$ **may not be invertible, meaning that backtracking is not always possible**

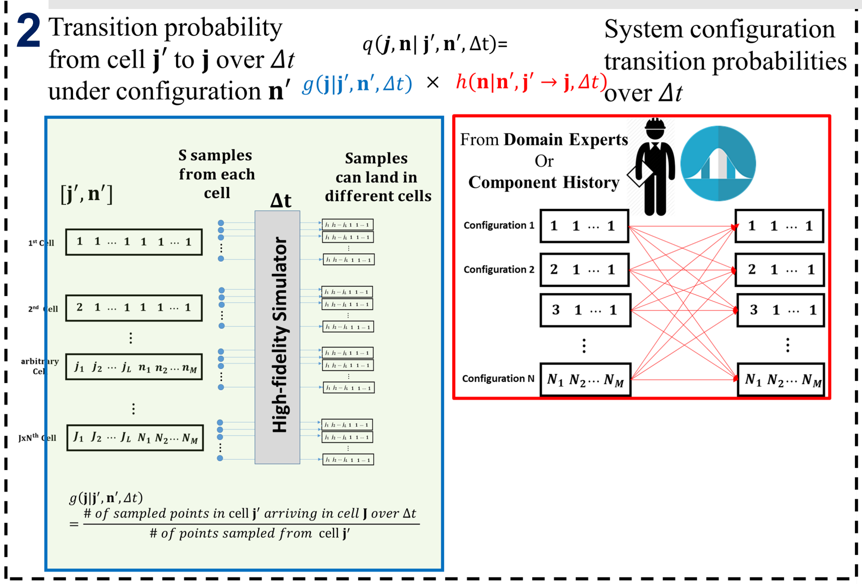
THE OHIO STATE UNIVERSITY

# The Backtracking Process Algorithm

**Solution 1:** **An equal weight quadrature scheme can be used to sample multiple points from each cell, and run these samples in forward simulation over a single time-step to represent cell-to-cell mappings.**

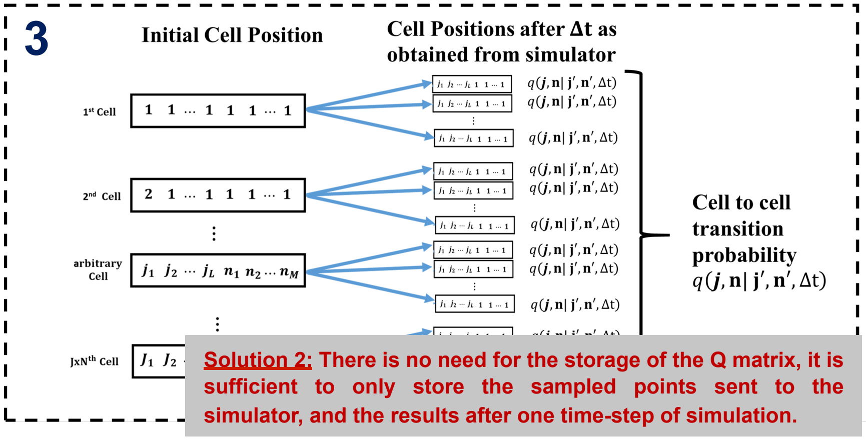
- BPA was designed to overcome the aforementioned challenges, which are associated with deductive implementations of Markov/CCMT.
- BPA is a deductive and memory efficient implementation of Markov/CCMT.
- The algorithm includes the addition of three main components to Markov/CCMT. Each component addresses a challenge associated with the deductive implementation of Markov/CCMT.



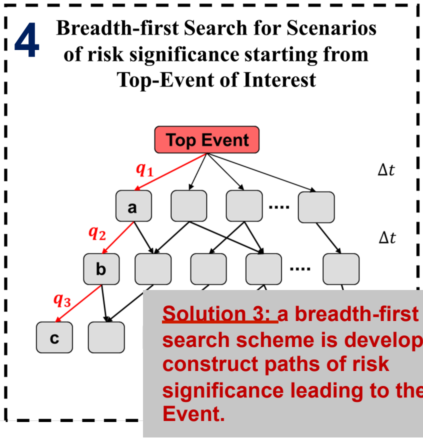

2
Transition probability from cell $\mathbf{j}'$ to $\mathbf{j}$ over $\Delta t$ under configuration $\mathbf{n}'$
System configuration transition probabilities over $\Delta t$
$[\mathbf{j}', \mathbf{n}']$
S samples from each cell
Δt
Samples can land in different cells
1st Cell
2nd Cell
arbitrary Cell
JxNth Cell
$1 \; 1 \cdots 1 \; 1 \; 1 \cdots 1$
$2 \; 1 \cdots 1 \; 1 \; 1 \cdots 1$
$j_1 \; j_2 \cdots j_L \; n_1 \; n_2 \cdots n_M$
$J_1 \; J_2 \cdots J_L \; N_1 \; N_2 \cdots N_M$
High-fidelity Simulator
$j_1 \; j_2 \cdots j_L \; 1 \; 1 \cdots 1$
From Domain Experts
Or
Component History
Configuration 1
Configuration 2
Configuration N
$1 \; 1 \cdots 1$
$2 \; 1 \cdots 1$
$3 \; 1 \cdots 1$
$N_1 \; N_2 \cdots N_M$

$$q(\boldsymbol{j}, \mathbf{n} | \, \mathbf{j}', \mathbf{n}', \Delta t) = g(\mathbf{j}|\mathbf{j}', \mathbf{n}', \Delta t) \; \times \; h(\mathbf{n}|\mathbf{n}', \mathbf{j}' \rightarrow \mathbf{j}, \Delta t)$$

$$g(\mathbf{j}|\mathbf{j}', \mathbf{n}', \Delta t) = \frac{\#\ of\ sampled\ points\ in\ \text{cell}\ \mathbf{j}'\ arriving\ in\ cell\ \mathbf{J}\ over\ \Delta t}{\#\ of\ points\ sampled\ from\ \text{cell}\ \mathbf{j}'}$$

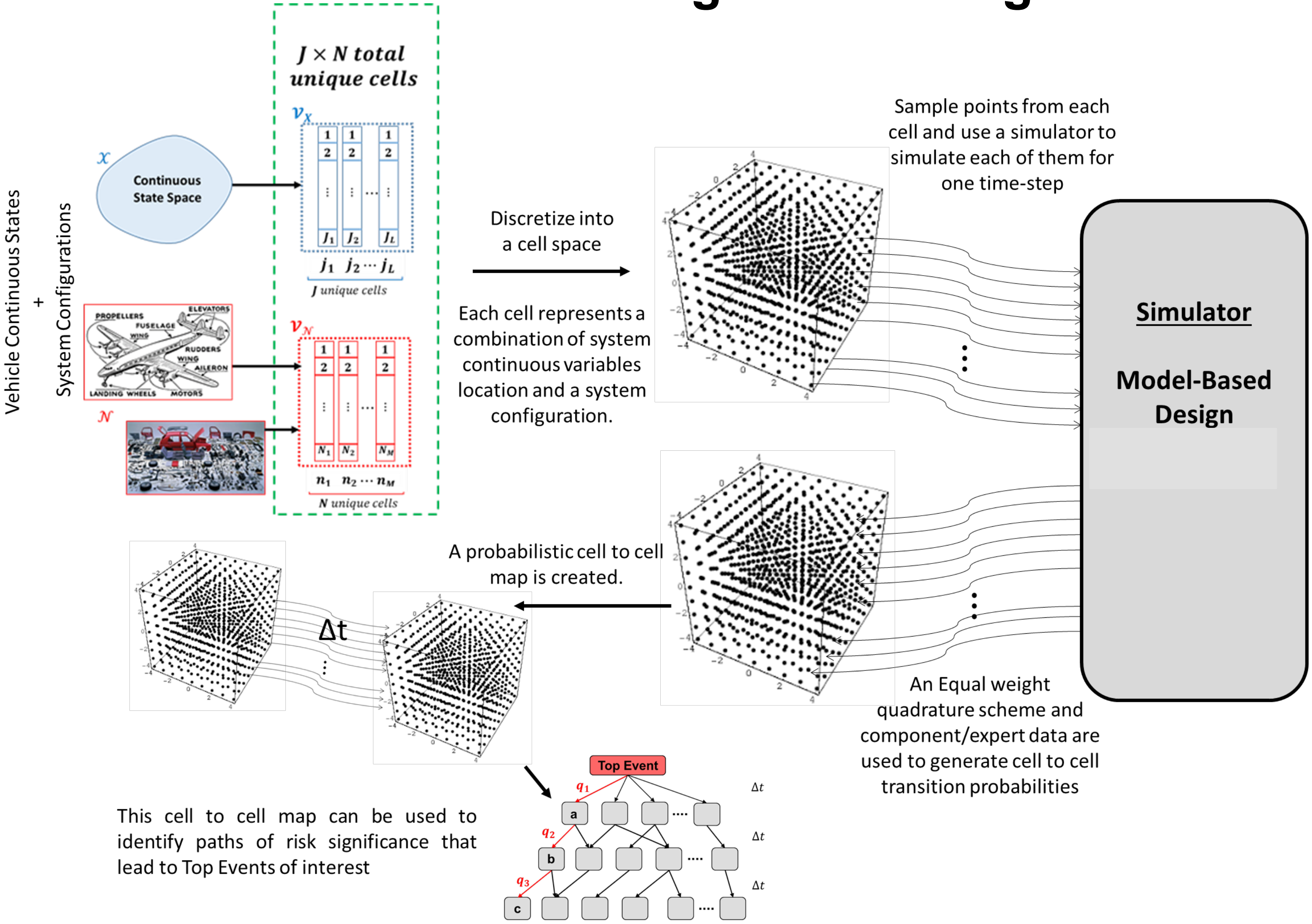

3
Initial Cell Position
Cell Positions after Δt as obtained from simulator
1st Cell
2nd Cell
arbitrary Cell
JxNth Cell
$1 \; 1 \cdots 1 \; 1 \; 1 \cdots 1$
$2 \; 1 \cdots 1 \; 1 \; 1 \cdots 1$
$j_1 \; j_2 \cdots j_L \; n_1 \; n_2 \cdots n_M$
$J_1 \; J_2$
$j_1 \; j_2 \cdots j_L \; 1 \; 1 \cdots 1$
$q(\boldsymbol{j}, \mathbf{n} | \, \mathbf{j}', \mathbf{n}', \Delta t)$
Cell to cell transition probability $q(\boldsymbol{j}, \mathbf{n} | \, \mathbf{j}', \mathbf{n}', \Delta t)$

**Solution 2:** **There is no need for the storage of the Q matrix, it is sufficient to only store the sampled points sent to the simulator, and the results after one time-step of simulation.**


4
Breadth-first Search for Scenarios of risk significance starting from Top-Event of Interest
Top Event
$q_1$
$q_2$
$q_3$
a
b
c
....
Δt
Δt

**Solution 3:** **a breadth-first search scheme is developed to construct paths of risk significance leading to the Top Event.**

# An Overview of the Backtracking Process Algorithm



Discretize into a cell space

Each cell represents a combination of system continuous variables location and a system configuration.

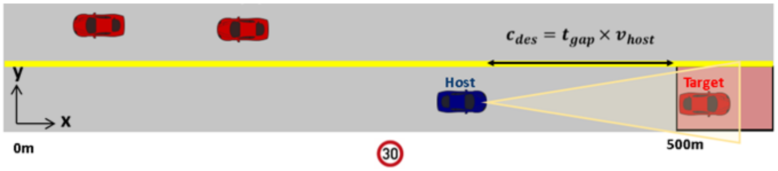
Sample points from each cell and use a simulator to simulate each of them for one time-step

An Equal weight quadrature scheme and component/expert data are used to generate cell to cell transition probabilities

A probabilistic cell to cell map is created.

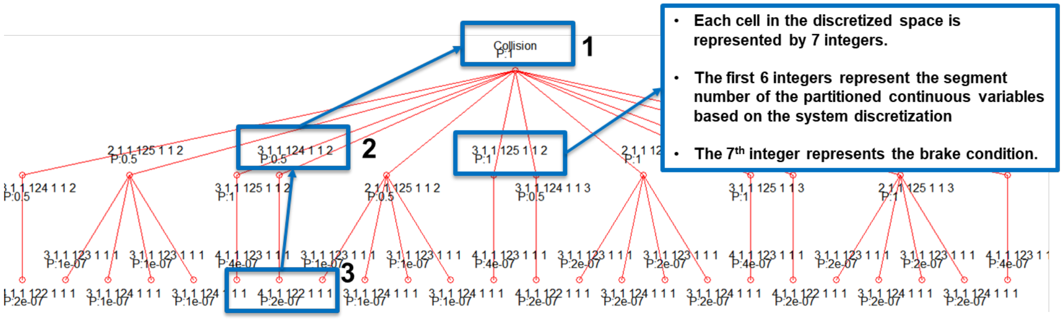
This cell to cell map can be used to identify paths of risk significance that lead to Top Events of interest

# BPA Example – A Simple Case Study



- BPA was used to provide an assurance case for a simple case study of a controller for an autonomous car approaching a stationary vehicle or object under possible Brake malfunctions
- Emergency and contingency actions can be modified based on the identified scenarios from BPA.
- This process can then be iteratively used to modify contingency actions, until results ensure that scenarios only lead to the violation of a safety goal within acceptably low probabilities

# Sample Results



- Each cell in the discretized space is represented by 7 integers.
- The first 6 integers represent the segment number of the partitioned continuous variables based on the system discretization
- The 7th integer represents the brake condition.

- BPA Results are presented in a search tree format.
- The top node represents the Top Event, all subsequent nodes represent the system state at previous time-steps, with probabilities associated to transitions in the tree.
- Once the nodes are converted back to the continuous domain representation, each branch would represent a sequence of events that led to the Top Event under current controller actions.

1. [4 1 1 122 1 1 **1**]– The AGV initially has a forward velocity of 16 to 20 m/s, a sideward velocity of -0.5 to 0.5m/s, a yaw rate of -0.05 to 0.05 rad/s, an x-position of 484-488, an y-Position of -2 to 2m, and a Yaw angle of –3° to 3°. The brake state was Normal.

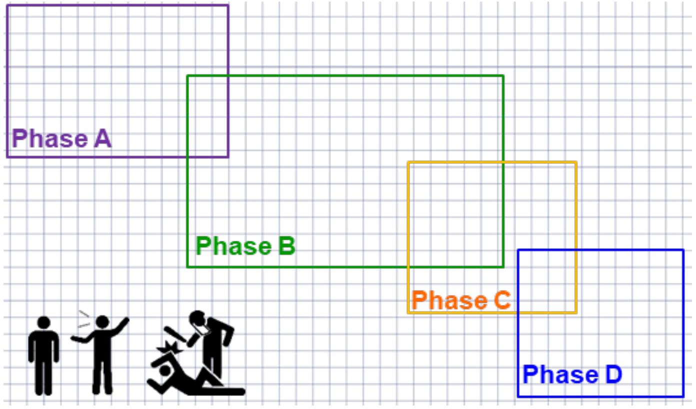
2. [3 1 1 124 1 1 **2**]– One time step later (.66s), the AGV had a forward velocity of 12 to 16m/s, a sideward velocity of -0.5 to 0.5m/s, a yaw rate of -0.05 to 0.05 rad/s, an x-position of 492 – 496m, an y-Position of -2 to 2m, and a Yaw angle of –3° to 3°. The vehicle experienced a Minor Brake Fault.

3. **Collision** – One time step later the AGV collides with the stationary vehicle located at an x-position that is greater than 500m, leading to a violation of the safety goal.

# Challenges of BPA and Proposed Solutions

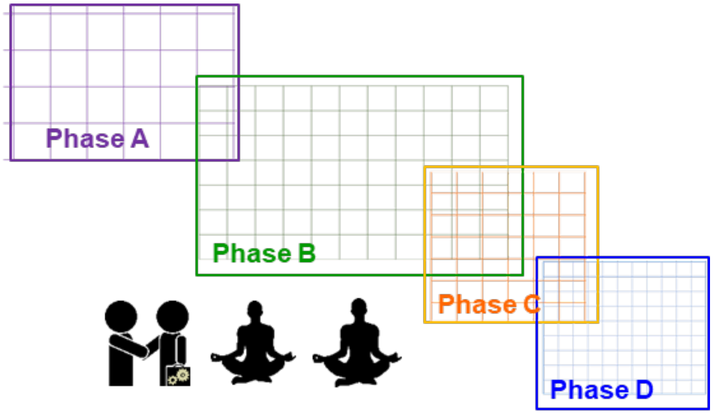
- While BPA can alleviate challenges associated with Markov/CCMT, use of a BPA breadth-first search scheme has two main limitations:

1. Large scale control systems that involve high levels of autonomy and numerous hardware are generally hard to accurately set up and initialize using single BPA implementations.
   - For large scale systems, using a single partitioning scheme to create a cell space can prove to be very computationally expensive.
   - Domain experts need a high level of coordination for a single BPA implementation across all mission phases.
2. For autonomous systems with large state spaces such as platoons or formations of vehicles, combinatorial and computational issues are prone to appear.



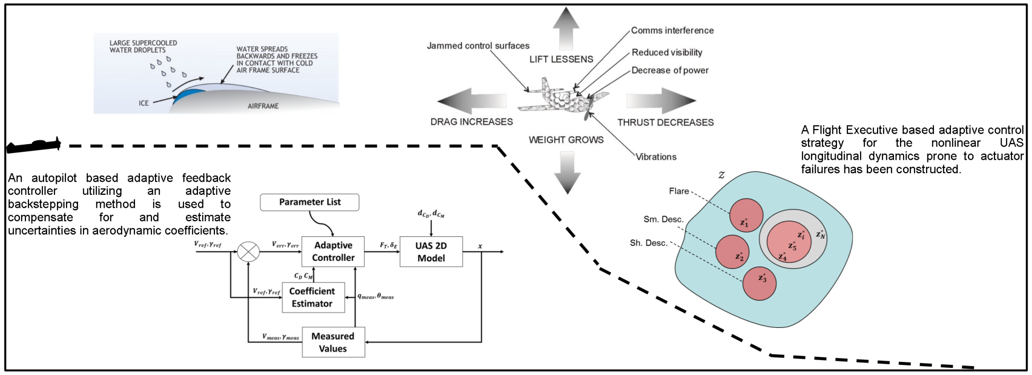
- Two solutions that have been explored to address the challenges are the following:

1. Use of phase-specific BPA implementations, and the integration of analysis results that are obtained from runs over multiple phases.
2. Reduction of the system cell-to-cell map by use of a coarser partitioning scheme, and compensating for the coarser scheme by increasing the number of samples taken from each cell in the quadrature scheme.



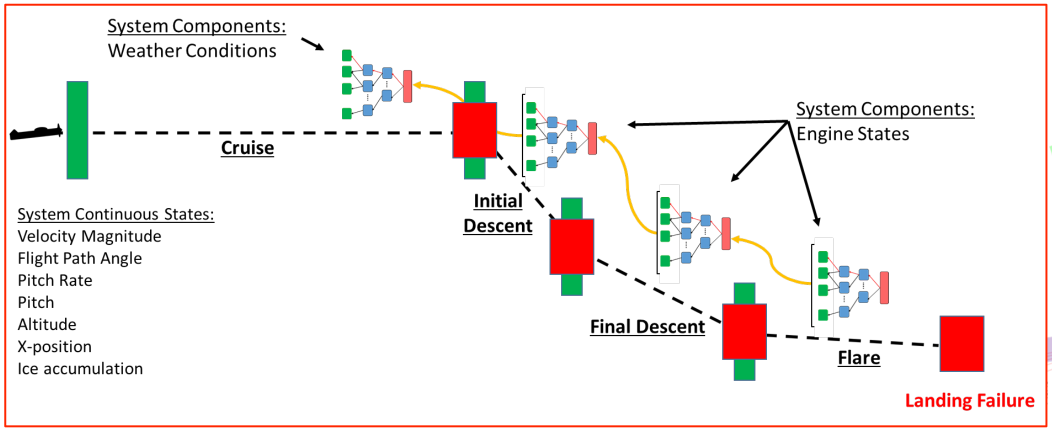
# The Phased BPA Implementation: A Case Study

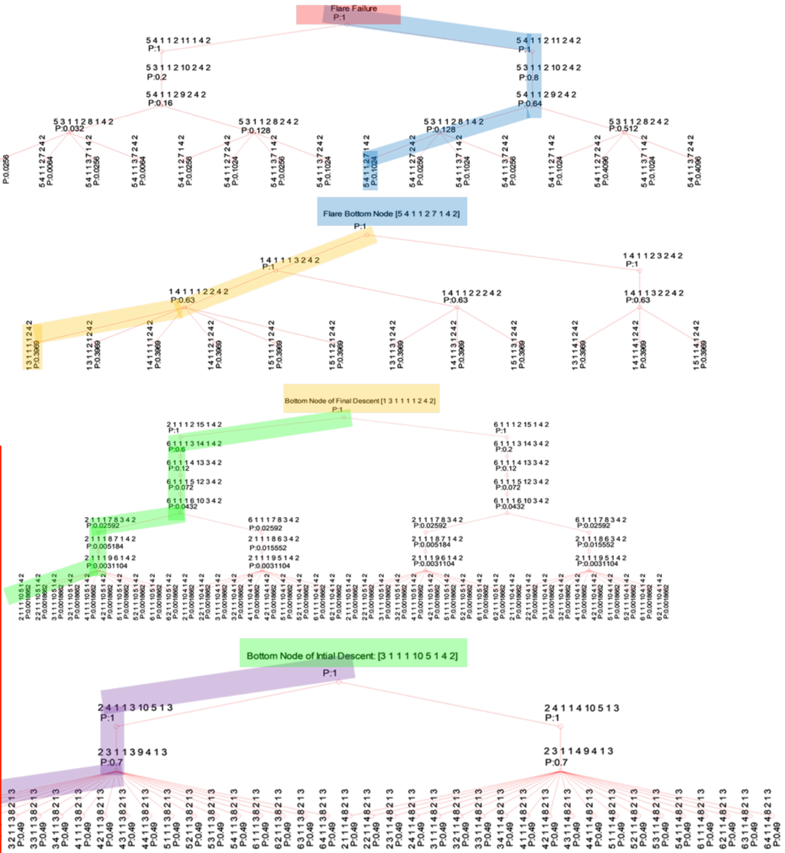
- A scenario is constructed for a UAS cruising at an altitude that makes it prone to icing. The UAS then performs a  land, or a sub-nominal land maneuver with an initial descent, final descent, and flare.



An autopilot based adaptive feedback controller utilizing an adaptive backstepping method is used to compensate for and estimate uncertainties in aerodynamic coefficients.

A Flight Executive based adaptive control strategy for the nonlinear UAS longitudinal dynamics prone to actuator failures has been constructed.





| Phase | Cell representation of Process Variables & System Configuration | Time-step | Elapsed Time | Cell Description<br>*only the process variables and configurations relevant to the system evolution are described |
|---|---|---|---|---|
| Cr | $[2\ 3\ 1\ 1\ 3\ 8\ 2\ \mathbf{1}\ \mathbf{3}]_{C}$ | 100s | 0s | UAS level at [1425,1475) m, a velocity of [40, 45) m/s, [20,40) % icing, under severe weather conditions. |
| Cr | $[2\ 3\ 1\ 1\ 3\ 9\ 4\ \mathbf{1}\ \mathbf{3}]_{C}$ | 100s | 100s | UAS level at [1425,1475) m, a velocity of [40, 45) m/s, [40,60) % icing, under severe weather conditions. |
| Cr | $[2\ 3\ 1\ 1\ 3\ 10\ 5\ \mathbf{1}\ \mathbf{3}]_{C}$ | 100s | 200s | UAS level at [1425,1475) m, a velocity of [40, 45) m/s, [80,100) % icing, under severe weather conditions. |
| ID | $[3\ 1\ 1\ 1\ 10\ 5\ \mathbf{1}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 300s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [40, 45) m/s, altitude of [1350, 1500) m, full icing, and engine power loss. |
| ID | $[2\ 1\ 1\ 1\ 9\ 6\ \mathbf{1}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 360s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [35,40) m/s, altitude of [1200, 1350) m, full icing, and engine power loss. |
| ID | $[2\ 1\ 1\ 1\ 8\ 7\ \mathbf{1}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 420s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [35,40) m/s, altitude of [1050, 1200) m, full icing, and engine power loss. |
| ID | $[2\ 1\ 1\ 1\ 7\ 8\ \mathbf{3}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 480s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [35,40) m/s, altitude of [900, 1050) m, full icing, and engine power loss. Set-point mode changes from 1 to 3 (higher velocity) |
| ID | $[6\ 1\ 1\ 1\ 6\ 10\ \mathbf{3}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 540s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [55,60) m/s, altitude of [750, 900) m, full icing, and engine power loss. |
| ID | $[6\ 1\ 1\ 1\ 5\ 12\ \mathbf{3}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 600s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [55,60) m/s, altitude of [600, 750) m, full icing, and engine power loss. |
| ID | $[6\ 1\ 1\ 1\ 4\ 13\ \mathbf{3}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 660s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [55,60) m/s, altitude of [450, 600) m, full icing, and engine power loss. |
| ID | $[6\ 1\ 1\ 1\ 3\ 14\ \mathbf{1}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 720s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [55,60) m/s, altitude of [300, 450) m, full icing, and engine power loss. Set-point mode changes from 3 to 1 (adjusting FPA for flare entry). |
| ID | $[2\ 1\ 1\ 1\ 2\ 15\ \mathbf{1}\ \mathbf{4}\ \mathbf{2}]_{ID}$ | 60s | 780s | UAS descending at $\gamma \in [-6°, -3°)$, a velocity of [35,40) m/s, altitude of [150, 300) m, full icing, and engine power loss. |
| FD | $[1\ 3\ 1\ 1\ 1\ 1\ \mathbf{2}\ \mathbf{4}\ \mathbf{2}]_{FD}$ | 12s | 840s | UAS descending at $\gamma \in [-4°, -3°)$, a velocity of [34,36) m/s, altitude of [0,40) m, full icing, and engine power loss. |
| FD | $[1\ 4\ 1\ 1\ 1\ 2\ \mathbf{2}\ \mathbf{4}\ \mathbf{2}]_{FD}$ | 12s | 852s | UAS descending at $\gamma \in [-3°, -2°)$, a velocity of [34,36) m/s, altitude of [0,40) m, full icing, and engine power loss. |
| FD | $[1\ 4\ 1\ 1\ 1\ 3\ \mathbf{2}\ \mathbf{4}\ \mathbf{2}]_{FD}$ | 12s | 864s | UAS descending at $\gamma \in [-3°, -2°)$, a velocity of [34,36) m/s, altitude of [0,40) m, full icing, and engine power loss. |
| Fl | $[5\ 4\ 1\ 1\ 2\ 7\ \mathbf{1}\ \mathbf{4}\ \mathbf{2}\ ]_{Fl}$ | 5s | 876s | UAS descending at $\gamma \in [-2.5°, -2°)$, a velocity of [34,36) m/s, altitude of [13,26.7) m, full icing, and engine power loss. |
| Fl | $[5\ 3\ 1\ 1\ 2\ 8\ \mathbf{1}\ \mathbf{4}\ \mathbf{2}]_{Fl}$ | 5s | 881s | UAS descending at $\gamma \in [-3°, -2.5°)$, a velocity of [34,36) m/s, altitude of [13,26.7) m, full icing, and engine power loss. |
| Fl | $[5\ 4\ 1\ 1\ 2\ 9\ \mathbf{2}\ \mathbf{4}\ \mathbf{2}]_{Fl}$ | 5s | 886s | UAS descending at $\gamma \in [-2.5°, -2°)$, a velocity of [34,36) m/s, altitude of [13,26.7) m, full icing, and engine power loss. |
| Fl | $[5\ 3\ 1\ 1\ 2\ 10\ \mathbf{2}\ \mathbf{4}\ \mathbf{2}]_{Fl}$ | 5s | 891s | UAS descending at $\gamma \in [-3°, -2.5°)$, a velocity of [34,36) m/s, altitude of [13,26.7) m, full icing, and engine power loss. |
| Fl | $[5\ 4\ 1\ 1\ 2\ 11\ \mathbf{2}\ \mathbf{4}\ \mathbf{2}]_{Fl}$ | 5s | 896s | UAS descending at $\gamma \in [-2.5°, -2°)$, a velocity of [34,36) m/s, altitude of [13,26.7) m, full icing, and engine power loss. SP Mode is changed from 2 to 1 in preparation for touchdown |
| **TE** | **[Flare Failure]** |  | 901s | UAS arrives at touchdown point under undesirable conditions. |

# Conclusion & Insights

- Dynamic PRA methods can overcome some of the challenges posed by the traditional risk assessment methods and reachability analysis techniques.

- The BPA is ideal for use in emerging autonomous technologies that are utilizing model-based design procedures.

- The development of a set of procedures and methods for autonomous vehicle control system assurance is vital for the certification and safe deployment of such systems in civilian applications.

- The capability of BPA to consider the uncertain and stochastic nature of autonomous vehicles is especially advantageous when considering their insufficient operating experience.

- BPA is capable of identifying sequences of actions that involve changes in the system dynamics and component states that might not be directly obvious or easy to capture.

- The algorithm is easy to implement and can be used for practical problems in a mechanized manner. However, speed of analysis is largely dependent on the system size and the simulator representing the system.

# Future Work

- Theoretical foundation of the Sequential BPA is being developed for handling phased missions of large scale systems.

- BPA will be used to test controllers for multi-agent systems such as platoons of vehicles, or formations of aircraft.

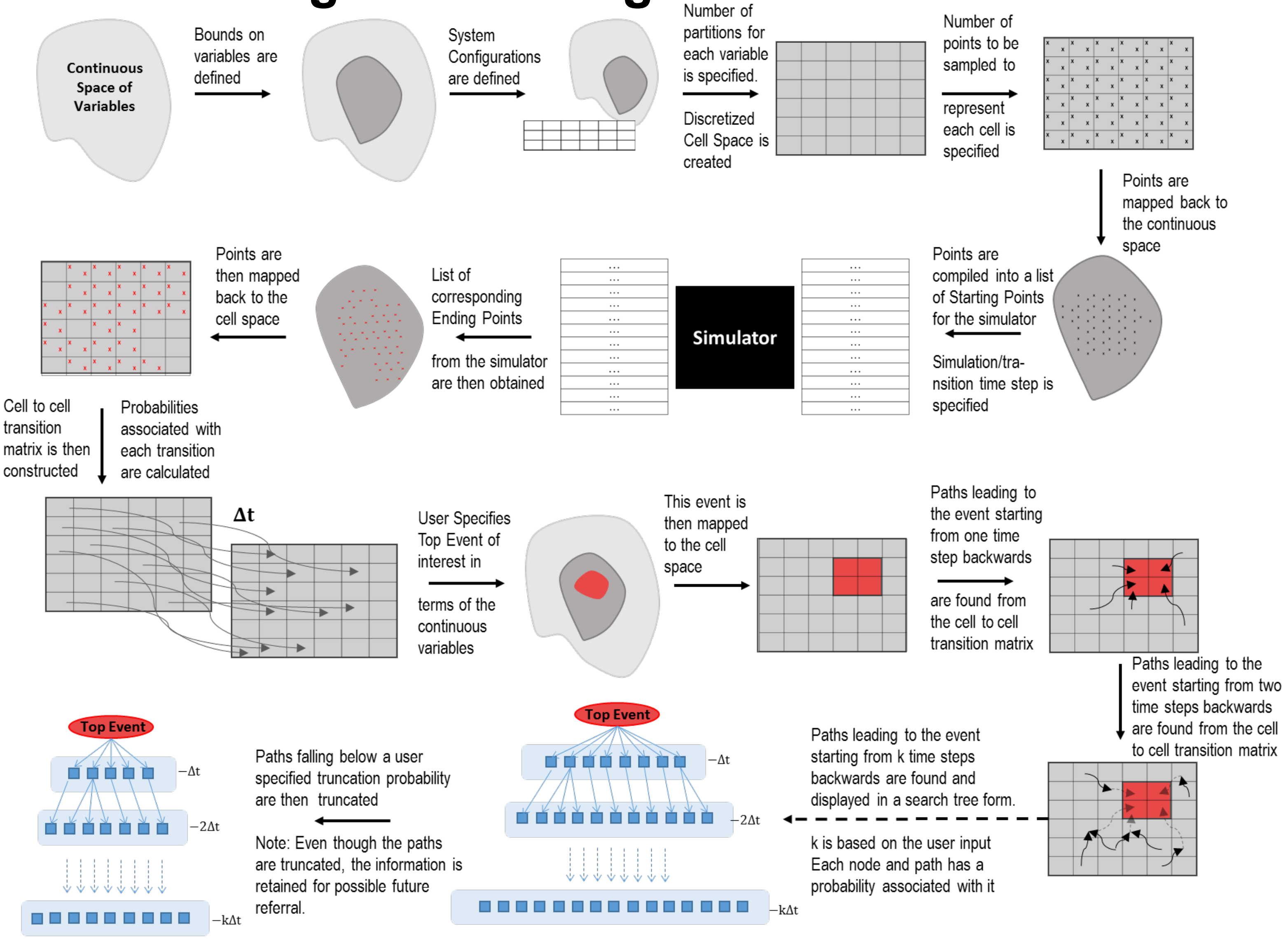
- The role BPA can play in early system Validation and Verification is being investigated.

# Thank You

**Contact:** hejase.1@osu.edu

# Appendix

# The Backtracking Process Algorithm Flowchart



Bounds on variables are defined

System Configurations are defined

Number of partitions for each variable is specified.

Discretized Cell Space is created

Number of points to be sampled to represent each cell is specified

Points are mapped back to the continuous space

Points are compiled into a list of Starting Points for the simulator

Simulation/transition time step is specified

List of corresponding Ending Points from the simulator are then obtained

Points are then mapped back to the cell space

Cell to cell transition matrix is then constructed

Probabilities associated with each transition are calculated

User Specifies Top Event of interest in terms of the continuous variables

This event is then mapped to the cell space

Paths leading to the event starting from one time step backwards are found from the cell to cell transition matrix

Paths leading to the event starting from two time steps backwards are found from the cell to cell transition matrix

Paths leading to the event starting from k time steps backwards are found and displayed in a search tree form.

k is based on the user input
Each node and path has a probability associated with it

Paths falling below a user specified truncation probability are then truncated

Note: Even though the paths are truncated, the information is retained for possible future referral.

# Publications (1/2)

Journal Publication

1. Hejase, M., Kurt, A., Aldemir, T., Ozguner, U., Guarro, S. B., Yau, M. K., & Knudson, M. D. (2017). **Quantitative and Risk-Based Framework for Unmanned Aircraft Control System Assurance.** *Journal of Aerospace Information Systems*, 1-15.

Conference publications

1. Hejase, M., Oguz, A. E., Kurt, A., Ozguner, U., & Redmill, K. (2016). **A Hierarchical Hybrid State System Based Controller Design Approach for an Autonomous UAS Mission**. In *16th AIAA Aviation Technology, Integration, and Operations Conference* (p. 3294).

2. Guarro, Sergio, Umit Ozguner, Tunc Aldemir, Matt Knudson, Arda Kurt, Michael Yau, Mohammad Hejase, and Steve Kwon. **Formal Validation and Verification Framework for Model-Based and Adaptive Control Systems.** In *NASA Formal Methods Symposium*, pp. 227-233. Springer, Cham, 2016.

3. Guarro, S., Yau, M. K., Ozguner, U., Aldemir, T., Kurt, A., Hejase, M., & Knudson, M. (2017). **Risk Informed Safety Case Framework for Unmanned Aircraft System Flight Software Certification**. In *AIAA Information Systems-AIAA Infotech@ Aerospace* (p. 0910).

4. Guarro, S., Yau, M. K., Ozguner, U., Aldemir, T., Kurt, A., Hejase, M., & Knudson, M. (2017). **Formal Framework and Models for Validation and Verification of Software-Intensive Aerospace Systems**. In *AIAA Information Systems-AIAA Infotech@ Aerospace* (p. 0418).

5. Hejase, M., Kurt, A., Aldemir, T., Ozguner, U., Guarro, S., Yau, M. K., & Knudson, M. (2017). **A quantitative and risk based framework for UAS control system assurance.** In *AIAA Information Systems-AIAA Infotech@ Aerospace* (p. 0882).

# Publications (2/2)

Conference Publications (continued)

6. Hejase, M., Kurt, A., Aldemir, T., Ozguner, U., Guarro, S., Yau, M. K., & Knudson, M. (2018). **Dynamic Probabilistic Risk Assessment of Unmanned Aircraft Adaptive Flight Control Systems.** In *2018 AIAA Information Systems-AIAA Infotech@ Aerospace* (p. 1982).

7. Hejase, M., Kurt, A., Aldemir, T., Ozguner, U. (2018). **The Backtracking Process Algorithm: A Dynamic Probabilistic Risk Assessment Method for Autonomous Vehicle Control Systems.** Submitted to the *14th Probabilistic Safety Assessment and Management (PSAM14)* conference.

Workshop publication

1. Hejase, M., Kurt, A., Aldemir, T., & Ozguner, U. (2018). **Identification of Risk Significant Automotive Scenarios Under Hardware Failures**. *arXiv preprint arXiv:1804.04348*.