Assured Autonomy: Bringing Guarantees to Large-Scale Autonomous Operations

Saurabh Bagchi
Purdue University

Supported by:
Army Research Lab

Joint work with:
David Inouye, Mung Chiang, Somali Chaterji;
Prateek Mittal (Princeton)

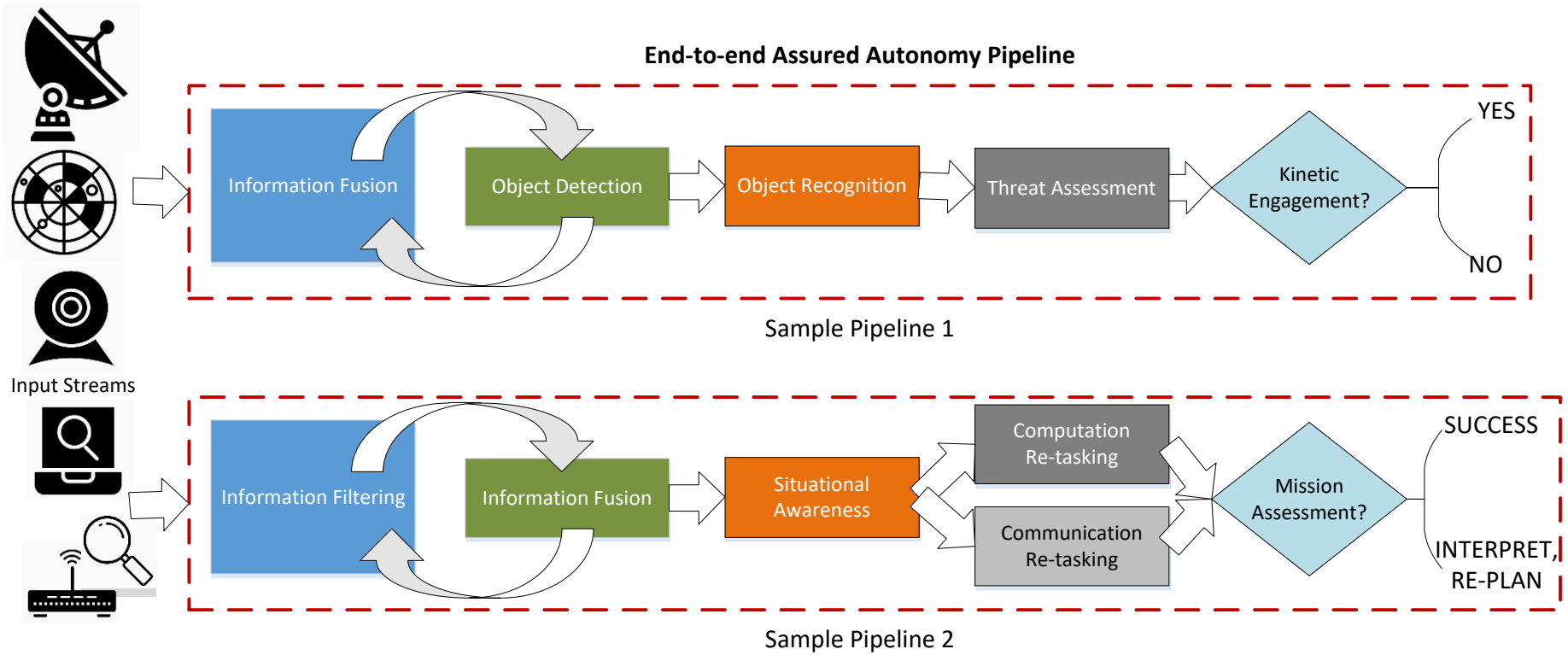
January 2021

PURDUE UNIVERSITY

Problem Context

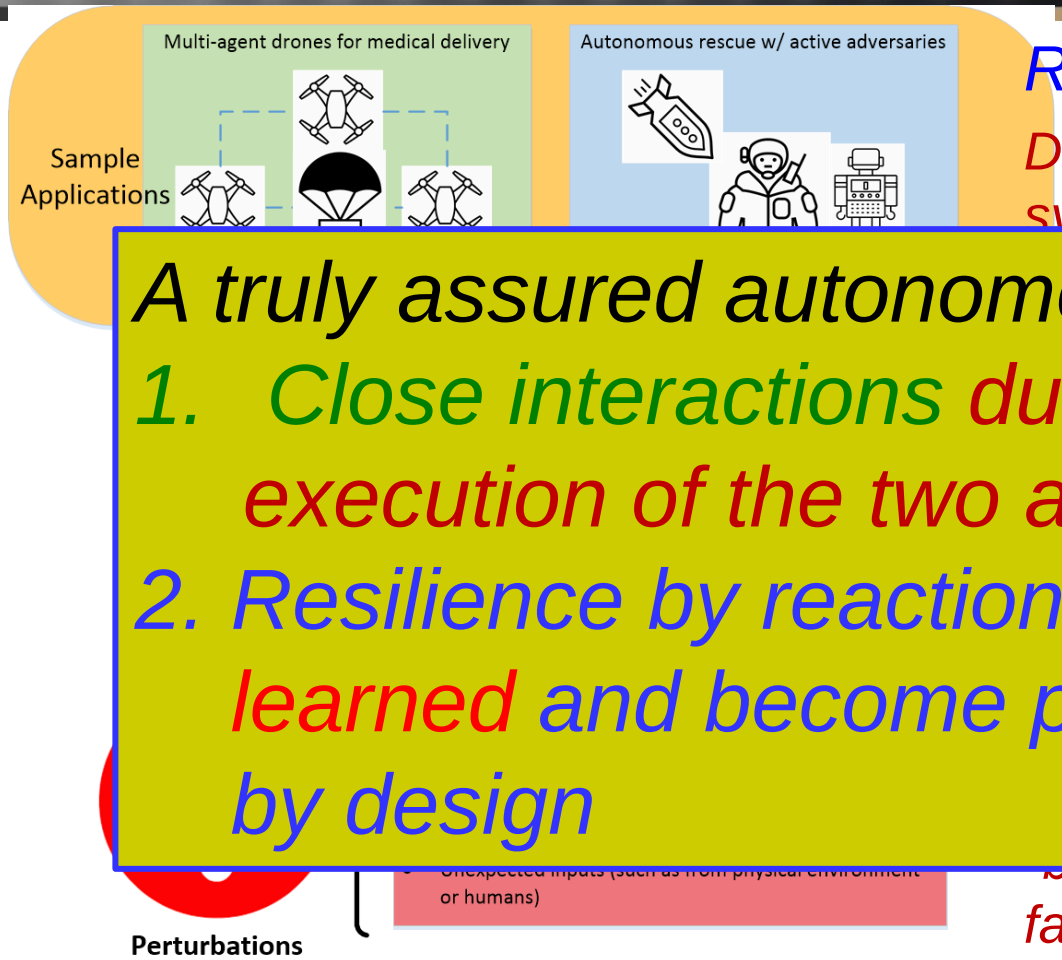
- Multi-domain operations of the DOD will involve **autonomous operations** among multiple cyber, physical, and kinetic assets, together with interactions with humans
- Such autonomous operation will rely on:
  - A **pipeline** of machine learning (ML) algorithms
  - Executing in **real-time** on
  - A **distributed** set of **heterogeneous platforms**
- Conditions will be adversarial and operation must be guaranteed to be secure while **maintaining timeliness guarantees**
  - Security guarantees need to be carefully analyzed and proven, under well-quantified adversary models
  - Move away from one-off solution for specific attack type
- Different degrees of autonomy
  - Some require humans in the loop; in such cases, the cognitive load of any software solution must be analyzed for feasibility
  - Some require humans on the loop
  - Some are fully autonomous agents

Sample Autonomy Pipelines



- Multiple ML algorithms in a pipeline
  - Different resource requirements, different input-output patterns
  - Can be required to execute on vastly different platforms

Aspects of Resilience of Autonomous Systems



Resilience by design:

Designs & develops autonomous systems so that they are resilient

A truly assured autonomous system has:

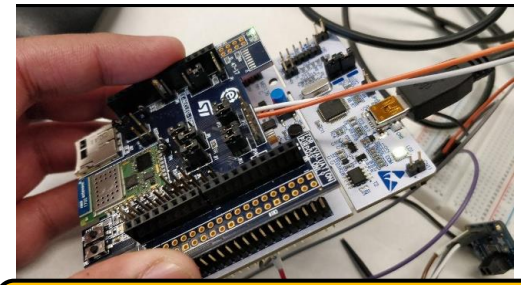
1. *Close interactions during design and execution of the two aspects*
2. *Resilience by reaction principles are learned and become part of resilience by design*

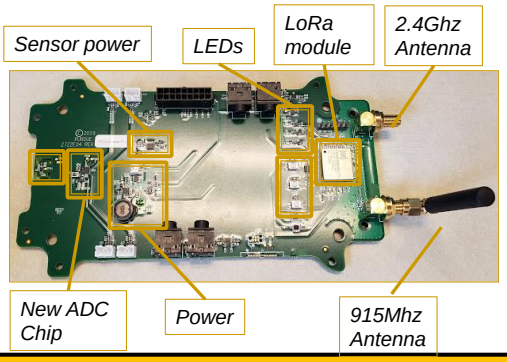
failure triggered by a perturbation

S. Bagchi, V. Aggarwal, S. Chaterji, F. Douglis, A. Gamal, J. Han, B.J. Henz, H. Hoffmann, S. Jana, M. Kulkarni, F.X. Lin, K. Marais, P. Mittal, S. Mou, X. Qiu, and G. Scutari, "Vision Paper: Grand Challenges in Resilience: Autonomous System Resilience through Design and Runtime Measures," in IEEE Open Journal of the Computer Society (OJCS), pp. 1-15, 2020.

Testbed for Evaluation of Assured Autonomy Protocols

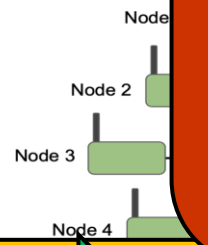
PURDUE UNIVERSITY Discovery Park





Low-end embedded devices

The integrated testbed is network accessible and programmable, ready for controlled experimentation of assured autonomy algorithms



Sensor → Edge → Cloud

Purdue campus-wide deployment of mesh network

PURDUE UNIVERSITY

Requirements for Assured Autonomy

PURDUE UNIVERSITY Discovery Park

1. ML algorithms must be capable of
   - Executing on **a distributed set of execution platforms** (mobile nodes, ground-based or aerial sensors, edge computing nodes, private cloud nodes, etc.)
   - Trained **both offline** and **in the field**
   - Tolerating **varying amounts of noise** either due to naturally occurring causes or due to maliciously injected errors
2. Autonomous algorithms must interface well with humans who may need to act on their decisions
   - Interpretable and explainable at **the tactical level** in real time
   - Interpretable and explainable at **the strategic level** so that a leader may make modifications for future missions

PURDUE UNIVERSITY

Requirements

3. Probabilistic guarantees
   - On **accuracy** *and* **latency**
   - Guarantees must hold under adversarial actions
   - Guarantees must hold under batch mode *and* incremental training
4. Algorithms must be able to ingest heterogeneous sources of data
   - Data sources will vary in their **fidelity**, **rate**, and **characteristic**
   - These data sources will be **intermixed** coming from white, blue, and red networks
   - In the process of inferencing, the algorithms also **tag data sources with their trust level**, so that future decision making becomes more accurate

Sample Project: Anomaly Detection in Multi-variate Sensor Streams

- Anomaly detection ----» Distribution shift detection and localization
- **Complex shift:** The means marginal distributions are equal, but the joint distributions are different

Our solution:

1. *Detect if a distribution shift has occurred in a time series*
2. *Detect if the shift is due to conditional distribution change*
3. *Perform this through a test statistic based on the density model score function (i.e., gradient with respect to the input)*
4. *Perform this efficiently where test statistics for all dimensions is calculated in a single forward and backward pass*
5. *Perform localization to determine which sensor(s) are compromised*

S. Kulinski, S. Bagchi, D. Inouye, "Feature Shift Detection: Localizing Which Features Have Shifted via Conditional Distribution Tests," in NeurIPS, pp. 1–21, November 2020.

Next Steps

PURDUE UNIVERSITY Discovery Park

- *Start of a 5-year Army Research Lab Assured Autonomy Institute (2020-25)*

- *Three Thrusts of Inquiry*
  1. *Robust adversarial algorithms*
  2. *Interpretable algorithms aiding the trust of the user on the results of the autonomous algorithms*
  3. *Secure, distributed execution of the autonomy pipeline among multiple platforms*

PURDUE UNIVERSITY