Resilient and Adaptive Autonomy Using Multi-Agent Reasoning

Josef Schaff
Johns Hopkins University Applied Physics Laboratory
josef.schaff@jhuapl.edu

Abstract

The ability to predict catastrophic system events is important for computers and autonomous systems overall, in order to employ mitigations or at least minimize damage. Predicting these "tipping points" using machine learning is one way to forecast when a system will destabilize and the time before its expected failure. Frequently, the response time between detection and implementing a mitigation or shutdown needs to be at machine speeds vs. human-response speeds. We created and tested an algorithmic "toolkit" for different scales of complex systems, ranging from a small nonlinear analog system to a large power grid system. All of the algorithms can run without GPU enhancements, with the most lightweight nonlinear algorithm exhibiting zero-shot learning. We provided a second set of experiments with collaborative agents, which may allow zero-shot algorithms to be used in critical components to extend resilience for complex autonomous systems.

Introduction

Complex autonomous systems are inherently nonlinear systems by nature. They consist of multiple heterogeneous system components, which require stability or at least resiliency of their most-critical components. There are varying dependencies on each subsystem, which can allow other System-of-System (SoS) components to dynamically substitute functionality for the critical components. These dependencies also scale to collaborative platforms needed for critical missions, where some platforms may choose to aggregate for a mission when key platforms are lost.

If critical components are lost through attrition and not replaced immediately, platforms and overall missions may fail. Two key attributes needed for both recovery and to maintain resiliency are:

- the ability to predict the time to the onset of failure, so that other components can be dynamically reconfigured to take over a component's functionality, and
- a collaborative agent-based approach to communicate the potential failures and respective substitutions.

This prediction capability for complex systems not only allows mitigations for secure resilient systems, but lays the foundation for augmenting the recovery of these systems at runtime, beyond their normal performance envelope. A corollary of this is the ability to anticipate adversary countermeasures, so as to create enduring capabilities and robustness in our critical systems.

We have achieved functional performance of these two attributes in separate experiments over the past two years. One class of machine learning (ML) algorithms constructs internal models of these nonlinear systems for specific tasks, and experimental verifications of the algorithms provides evidence at different scales of their utility. Additionally, we demonstrated agent-based collaboration for disaggregated sensors and platforms, that allowed a mission to dynamically reallocate based on changing platform-sensor needs.

Forecasting Tipping Points with the "Tip-it" Algorithms: Introduction

The ability to predict catastrophic system events is important for computers and autonomous systems overall, in order to employ mitigations or at least minimize damage. Predicting these "tipping points" using machine learning is one way to forecast when a system will destabilize and the time before its expected failure. Frequently, the response time between detection and implementing a mitigation or shutdown needs to be at machine speeds vs. human-response speeds.

Adversaries may initiate non-conventional attacks that include cyber, assumed by the attacker to be undetectable until it is too late. Similarly, complex production / processing systems may be on the verge of failure for multiple reasons without the obvious clues available in time for a human operator to take action. This goes beyond normal cyber mitigations and respective system predictive maintenance, and there is a clear need to forecast such failures. To solve this problem of proactive cyber, we generalize it to the forecast-

ing of system stability rather than looking at specific individual incidents. This gives us a broader application that minimizes the chance of our solution only working on specific incidents, and includes the support of complex systems.

Our internally funded research effort called "Tip-it" developed and tested specific classes of algorithms to forecast critical events that may be the precursor of cyber-attacks, or lead up to system failure. These algorithms "learn" the system by internally creating nonlinear mathematical models of the system with their respective environmental dependencies. This allows them to predict behaviors several time steps before they occur. The research has concluded with the following results:

- Successfully forecasted impending destabilizations for both small and large systems.
- Demonstrated scalability, ranging from a simple physical system to regional power grid-sized systems.
- Leveraged theoretical (nonlinear mathematical equations), real physical systems, actual static production plant data, and simulated power grid data.
- Demonstrated the ability to learn and successfully forecast simple physical systems at first exposure or run time, known as zero-shot learning; and refined forecasting (greater details) with 40 iterations total.

This research used several classes of ML algorithms selected for the appropriate problem space, which have been collected into an algorithm "toolkit."

The objective was to forecast serious anomalies that could result in systems shutdowns or overall failures. This is done by detecting "tipping points", using machine learning as a way to forecast when a system will destabilize as well as the time before expected failure. The end-objective for this, and follow-on work is twofold: a dynamically re-stabilizing system that is more resilient to attacks or failures, and a means of discretely de-stabilizing an adversary's system.

Tip-It Framework: Overview and Background

Tip-it is an autonomous mitigation framework for cyber-physical systems (CPS) as shown in Figure 1. Cyber-physical systems are a broad class of control systems that integrate computation and physical processes that allows the physical systems to be controlled in real-time. Examples of cyber-physical systems include critical infrastructure (power grids, water treatment facilities), autonomous mobile systems (uncrewed aerial systems (UAS), self-driving cars), autopilot and robotics. As shown on the left side of Figure 1, cyber-physical systems can be abstracted as a CPS loop where the physical system (i.e., the physical components of a power grid) are monitored by a set of sensors which continuously provide feedback to a controller. The controller then applies commands back to the physical system through actuators. In a traditional CPS, the controller selects actions with the goal of driving the physical system towards a desired state. For example, if the CPS is an airplane on autopilot the controller will update the inferred state of the system with the latest sensor readings and using well known equations of motion, select an action that ensures the airplane remains in stable flight and in the direction chosen by the pilot.

A key assumption in the design of traditional cyber-physical systems is that the controller has access to a well-defined *model of the physical system*. In this case, the model is a set of dynamical equations that define the evolution of the system over time. Formally, the model can be defined as $T(S_{t+1}|S_{0:t}, A)$ which represents the probability of transitioning from state S_t to state S_{t+1} after executing action A. When working with cyber-physical systems, the state S includes all information that is measured in the system including sensor readings and/or actuation signals. Traditionally control algorithms including proportional-integral-derivative (PID) and model-predictive controllers (MPC) then utilize the model to select a desired action at each timestep.

Data-Driven Modelling

A key challenge in modelling complex cyber-physical systems is formulating the equations to represent the model. For example, in the case of power-grids there are potentially hundreds of separate sub-systems that interact in highly nonlinear, complex ways. Additionally, in the case of a cyber-attack we cannot assume that all components will behave as expected rendering the physical equations ineffective. The same situation arises as components degrade over time. All of these examples share a common problem – the dynamical model of the system can change over time.

Therefore, the tip-it framework provides a set of general-purpose algorithms that can autonomously learn the model of a system purely through data. This has numerous benefits. First, data-driven models do not require domain-expert design. Not only does this save significant time when designing cyber-physical systems, but it allows a model to be easily learned and deployed across a variety of applications rapidly. In fact, we show that we can learn highly accurate models of complex dynamical systems purely from sensor readings without having access to the underlying schematic designs or consulting with domain experts. Second, data-driven models allow us to model highly dynamical systems with hundreds of components that would be intractable for a purely physics-based approach. Third, data-driven methods allow us to *effectively forecast over long time horizons* which is particularly useful for identifying destabilizing points as well as detecting anomalies. Lastly, unlike pre-defined physical equations, our methods can adapt rapidly to changing system dynamics online. This means that our models can autonomously learn new dynamics in the case of cyber-attacks or component degradation.

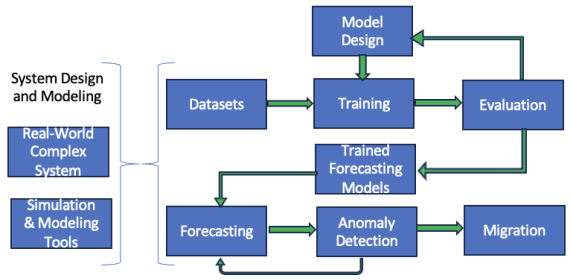

Figure 1: Forecasting methods on a set of real-world datasets as well as using simulation and modelling tools. We design our forecasting and anomaly detection algorithms to work both in offline settings where data is collected for training prior to deployment as well as in online settings where the algorithm is updated continuously at runtime.

One of the primary challenges in utilizing traditional ML methods for forecasting cyber-physical systems is that offline training assumes that the scenarios seen at runtime are from the same distribution as the data collected for training. In other words, offline training fails when runtime conditions are out-of-distribution. This means that data-driven methods perform poorly when the system is under-going a targeted cyber-attack or when a dynamical system switches phases – i.e., from normal to transient, or transient to destabilization (Kong et al. 2021). Online training, or adaptation, is the process by which the forecasting algorithms are continuously updated at runtime. Online adaptation is required to adapt forecasting algorithms to new behaviors or system dynamics not seen during training, and is one of the most challenging open problems in machine learning (Hoi et al. 2021).

To address these challenges, we proposed an algorithm (Algorithm 1) for online learning that jointly learns and forecasts at runtime as shown in Figure 2. The algorithm starts with the weights of a data-driven model W which as we will show in our experiments can be either randomly initialized and completely trained online (zero-shot learning). For each timestep, the algorithm alternates forecasting future timesteps with a model update step that updates the model weights to reflect the new information. This procedure keeps the model up-to-date with current operation conditions. Algorithm 1 is model-agnostic in that it does not specify what type of algorithm was used to represent the system model. However, as we showed in our experiments, reservoir computing is particularly effective within this framework and can even work well in *zero-shot learning* scenarios.

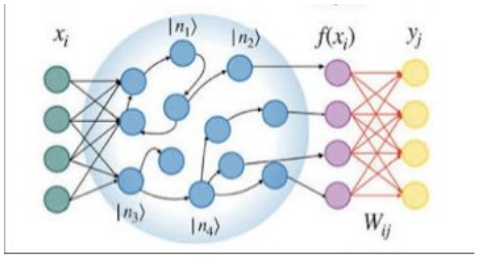

Figure 2: Overview of a Tip-It forecasting algorithm – Reservoir computing variant.

We explored multiple reservoir computing (RC) algorithms including echo state networks and Next-Generation Reservoir Computing (NGRC) (Gauthier et al. 2021). We also looked at applying it to Tipping Point detection (Li et al. 2023), and realized that we may need to augment NGRC (Ma, Prosperino, and Räth 2023) for our purposes. RC is a subset of recurrent neural networks (RNNs) that include a fixed-weight reservoir that projects the state history into a higher dimensional space and then utilizes a linear layer to forecast the state at the next timestep. These algorithms have shown to be highly effective at modelling known dynamical equations, however their usage in real-world cyber-physical systems has yet to be well studied. Our experiments showed that RC has numerous benefits including working well with sparse data in the context of online and zero-shot learning.

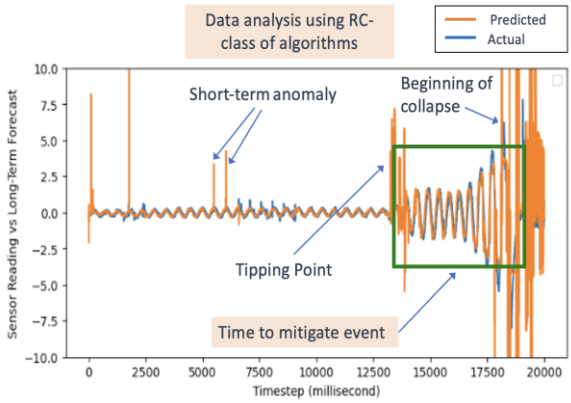

Figure 3: Orange is the predicted graph; Blue is actual data. The samples are taken in 100 millisecond increments. This was generated with data from the first run (zero-shot), and shows how well the RC learned the system "online", i.e., at runtime.

Tip-it Summary

The Tip-it toolkit successfully forecasted not only cyber anomalies, but chaotic attractors (Mukeshwar and Lai 1999) as well as overall system failing points in advance, allowing future mitigations prior to the failure's onset (Figure 3). By embedding this computationally lightweight ML algorithm class into critical components or even platforms, these components could successfully forecast their future "health", indicating whether they could complete their mission independently or need assistance.

Agent-Based Collaborating Disaggregated Sensors: Introduction

A simple demonstration was done on how secure Agent-To-Agent (A2A) protocol-enabled sensors and vehicles could autonomously accomplish a shared mission without the need for sensor fusion or large bandwidth. Heterogeneous sensors using A2A can selectively accomplish a mission. By using A2A and enabling most sensors and platforms to have differing agent-based capabilities, disaggregated sensors and platforms can selectively aggregate locally on demand to accomplish a mission. The aggregation could be done fully autonomously for sensors and uncrewed platforms. Varying degrees of autonomy can be allowed, depending upon both the capabilities and trustworthiness of the agents incorporated using the A2A protocol (Surapaneni, Jha, Vakoc, and Segal 2025).

Experiment Objectives

We proved that heterogeneous sensors and platforms can collaborate together to accomplish a mission that cannot be accomplished by any individual one, using the state-of-art A2A protocol. Each agent shares their specific A2A card with others. Sensors and systems are distributed arbitrarily, and have different abilities and protocols - e.g., ultrasonic short-range, IR comms, visual (camera) sensors, while some are fixed or moving - e.g., static sensors or localized robots with moving sensors. We showed that a more intelligent agent or agentic system can select from the A2A cards of sensors / platforms to use for the mission. This can be accomplished using simple Artificial Intelligence (AI) reasoning, with no need for computationally complex AI/ML. Agents can also seamlessly integrate advanced AI/ML algorithms for sensors as needed, including Large Language Models (LLMs).

Description

Using low-cost hardware and sensors, our software achieved a simple demonstration of how Agent-To Agent (A2A)-enabled sensors and vehicles could autonomously accomplish a shared mission without the need for sensor fusion or large bandwidth usage. This holds the potential for enabling most sensors and platforms to have an agent-based capability, and allow disaggregated sensors and platforms to selectively aggregate locally on demand to accomplish a mission. The aggregation could be done fully autonomously to achieve commander's intent, or with human assistance or intervention as needed.

Mission Details

The mission consisted of two parts, the first involving an intelligent camera that detected a target, and sent a robot to directly intercept it. The second part involved the same camera detector, but the simple agents in both sensor and robot determined that the target "fly-over" was beyond interception for the default robot. The collaborating agents included a remote robot with other sensors as well as a remote stationary Lidar. Since the agents had shared their A2A cards beforehand, a master node agent, in this case residing on the camera, selected the cards of agents that would be the remote segment of the mission. Once the fly-over was detected, the master A2A node requested the remote robot to launch in "search mode" once the Lidar observed the target in proximity. The remote robot activated, and a copy of its launch command activation was sent back for confirmation to the master node, and completion of the mission.

This mission demonstrated integration of commercial sensors relevant to DoW and DHS missions to show how agent-based coordination can be layered onto existing technology. This created a scalable, low-cost framework where heterogeneous sensors can use simple reasoning about mission objectives to rapidly form ad hoc networks and collaboratively achieve mission objectives, either autonomously or with human oversight. These steps paved the way for any sensor to become intelligent and mission-ready.

Conclusion and Summary

When critical components or platforms are lost through attrition or simple failures, the mission could still be accomplished by combining Tip-it forecasting with A2A-based agents. The critical elements of a mission would have embedded Tip-it forecasters and internal agents that could collaborate together to find resilient solutions to completing a mission despite attrition. This would be done by agents autonomously selecting alternative components or platforms through collaboration, in order to have mission success. Tip-it could also do zero-shot learning of the new collaboration-created system's performance, focusing on mission completion. With this approach, platforms could at times exceed their normal performance envelope, producing a far more resilient defense than expected by an adversary.

References

Gauthier, D. J.; Bollt, E.; Griffith, A.; and Barbosa, W.A. 2021. Next generation reservoir computing. *Nature communications* 12(1), 5564. doi.org/10.1038/s41467-021-25801-2.

Hoi, S. C.; Sahoo, D.; Lu, J.; and Zhao, P. 2021. Online learning: A comprehensive survey. *Neurocomputing* 459C: 249-289. doi.org/10.1016/j.neucom.2021.04.112.

Kong, L. W.; Fan, H. W.; Grebogi, C.; and Lai, Y. C. 2021. Machine learning prediction of critical transition and system collapse. *Physical Review Research* 3, 013090. doi.org/10.1103/PhysRevResearch.3.013090.

Li, X.; et al. 2023. Tipping Point Detection Using Reservoir Computing. *Research*(6) 0174. doi.org/10.34133/research.0174.

Ma, H.; Prosperino, D.; and Räth, C. 2023. A novel approach to minimal reservoir computing. *Scientific Reports* 13(1), 12970. doi.org/10.1038/s41598-023-39886-w.

Mukeshwar, D.; Lai, Y. C. 1999. Controlling transient chaos in deterministic flows with applications to electrical power systems and ecology. *Physical Review* E 59, 1646. doi.org/10.1103/PhysRevE.59.1646

Surapaneni, R.; Jha, M.; Vakoc, M.; and Segal, T. 2025. Announcing the Agent2Agent Protocol (A2A). https://developers.googleblog.com/en/search/?author=Rao+Surapaneni. Accessed: 2026-01-22.