

Steering Complex Systems using a Dynamic, Data-Driven Modeling Approach

Stacy Patterson and Carlos A. Varela

Department of Computer Science, Rensselaer Polytechnic Institute

1 Motivation

Dynamic data-driven applications and systems (DDDAS) use models to control systems using a feedback loop, however, they go beyond traditional control-theoretic approaches in that the data acquisition process itself may be controlled by steering the system resources. Thus, there is the opportunity to strategically collect data and modify the models dynamically to incorporate recent observations. This improved modeling capability can then be used in feedback to steer the system towards a desired trajectory or outcome. Such an adaptive, data-driven approach has the potential to significantly improve the modeling capability of complex systems that include streaming data and opens the door to new applications, including smarter, safer vehicles and environments.

Examples of such applications include:

Online failure modeling and control in aircraft. Suppose there is a failure model of an aircraft for the situation in which it loses all engines. The model simulates the aircraft's gliding capabilities and can help to determine the best flight plan to land in a nearby airport. Other models may consider failures where only one of the multiple engines has failed. However, should a bird strike cause engines to have partial power, as in the US Airways Flight 1549 in 2009 that was ditched over the Hudson river, no offline model may be able to predict the aircraft capabilities after the bird strike. In cases such as these, a DDDAS approach has the potential to more accurately model aircraft capabilities and therefore enable better decision support systems. Such DDDAS approaches may perform certain maneuvers in the aircraft (e.g., small changes in power, pitch, roll) to help more accurately determine the aircraft's damage conditions. In the case of the US Airways Flight 1549, we would like to be able to simulate, faster than real-time, landing the aircraft at Teterboro airport, back at La Guardia, and at the Hudson river, and associate a probability of success for each plan to inform the pilot, within at most 30 seconds of the bird strike. Furthermore, as the aircraft moves in space, flight plans need to be efficiently updated.

Fire hazard monitoring and management. Municipalities in areas prone to disasters, such as fires, have pre-designed plans and models to address the most common risk scenarios and locations, including fire spread, firefighting efforts, resource allocation, and evacuation routes. These models provide guidelines for actions in the event of a fire, but effective management of fire containment and public safety requires that these models be updated in real-time using data collected on the ground and from the air. With the advent of the Internet of Things and the Internet of Robots comes the widespread deployment of fixed sensor networks and autonomous information-gathering ground and air drones. These sensors and drones can generate streams of data about fire location, wind conditions, traffic along the evacuation routes, etc. With a DDDAS approach, these data streams may be used to adapt and refine models and plans. Further, within this plan adaptation process, it may be possible to identify where and when to gather additional sensor data and to steer the drones accordingly.

Internet of Planes Preferred Instrument Flight Rules (IFR) routes do not consider weather. It is the task of human air traffic controllers and Pilots in Command to change routes dynamically according to weather conditions, which may include hazardous icing or thunderstorm areas, typically moving and changing in size and severity. In extreme cases, planes are grounded, or flights are cancelled or postponed, with significant economic impact. Dynamic Data Driven Air Traffic Control can actively use weather information from planes in the air, to feed back into computer weather simulations, improve forecast accuracy based on real-time weather data, and use the new results dynamically to re-optimize routing of airplanes in such a way

as to optimize fuel consumption, and passenger safety. There are approximately 87,000 commercial and private flights per day in the U.S. that could benefit. With the advent of the Next Generation Transportation system (all airplanes will have ADS-B technology installed by 2020), there is an unprecedented opportunity for decentralized autonomous air traffic control [4, 3].

2 Data Error Tolerant Stream Processing in Avionics

Using a data-driven feedback loop, DDDAS-based avionics continuously analyze spatio-temporal data streams from airplane sensors, identify potential failure modes, and correct erroneous data. The result is a new layer of logical redundancy in addition to existing physical redundancy for safer flight systems.

We have created two new mathematical concepts: *error signatures* and *mode likelihood vectors*, to analyze redundant spatio-temporal data streams. An error signature is a set of constrained mathematical functions that is used to capture the characteristic pattern of an error function, a.k.a. residual, $e(t)$ under a specific failure condition. A mode likelihood vector assigns a likelihood $l_i \in [0, 1]$ to each of m modes, including a *normal* mode, and $m - 1$ known failure modes, based on the distance between measured data stream values over a time window ω and corresponding mode's theoretical signatures [6, 7].

We have also developed a domain-specific programming language: ProgrammIng Language for spatio-Temporal data Streaming applications (PILOTS) to enable a declarative (high-level) specification of DDDAS data streaming application models (input-output relationships between data streams), error signatures, and error correction functions. PILOTS software detects specific (e.g., failure-induced) data errors based on signatures and corrects data before processing streams according to the application model [5].

We have shown our DDDAS approach to be able to detect and correct for the air speed failure in the Air France Flight 447 (AF447) that crashed in the Atlantic Ocean in June 2009. Airspeed sensor failure of the AF447 flight is successfully detected and corrected after 5 seconds from beginning of the failure. Overall error mode detection accuracy reaches 96.31% [2]. We have also applied our mathematical techniques and software to analyzing the Tuninter 1153 accident in August 2005. The underweight condition due to the installation of an incorrect fuel sensor is successfully detected with 100% accuracy during the cruise phase of flight [1].

3 Research Challenges

To support steering of complex systems in real-time requires sophisticated online models of the cyber and physical components of the system. Due to their size and complexity, these online models must be derived, at least in part, from pre-computed offline models and plans. The models must then be continuously adapted based on real-time streaming data gathered by the system.

Research challenges for steering complex systems using a DDDAS approach include:

1. How to formalize models in such a way that some of the computation can be performed offline and yet, online data streams from sensors can be used to adapt the models' parameters to dynamic data. To be relevant for steering, in many cases, the adaptation method must run in real-time, and thus we must offload as much computation as possible to a pre-computation step. Techniques must also be devised to incrementally update models without needing to re-compute them from scratch upon gathering new data.
2. How to seamlessly partition and replicate the model and computation between different components of the system so as to maximize accuracy of prediction and minimize response times. To meet the real-time guarantees for model adaptation may require aggregating computational and data resources from different system components (for example, multiple drones) as well as incorporating additional external resources such as cloud computing. Therefore, there is a need to reconcile and aggregate multiple models in a meaningful way. Further, this aggregation must be fast, scalable, and robust to faults.

3. How to quantify the uncertainty and likelihood of success of given plans and predictions, adapt this uncertainty based on the data streams, and identify data points that can decrease this uncertainty. Quantification of uncertainty will aid decision-makers in formulating actions based on the models. By learning not only the model, but also how best to increase the accuracy of the model through additional data collection, the system operators (e.g., pilot) and participants (e.g., drones) can steer the vehicles to achieve a desired goal when possible and gather more information when needed.

Furthermore, it is imperative to have models with different accuracy that are evaluated concurrently. For example, an aircraft's computing resources can be used to quickly evaluate an approximate (reduced order) model and create an initial plan. Cloud computing resources off-the-aircraft can concurrently evaluate more accurate (higher-dimensional) models for better flight plans to propose to pilots only if such simulations can be computed under the real-time constraints imposed by the emergency condition. Thus, individual system components can function autonomously and obtain higher-accuracy plans or models from remote resources either when needed to reduce uncertainty or when possible given the time constraints of the situation.

4. How to create programming models and frameworks that support analytics and steering over streaming data. Such models and frameworks will make it easier for developers to create new analytics and decision support applications for streaming systems. Programming languages derived from these models should support both compiler (static) and middleware (dynamic) optimizations.

Acknowledgments This research is partially supported by the DDDAS program of the Air Force Office of Scientific Research, Grant No. FA9550-15-1-0214 and NSF Awards No. 1553340 and 1462342.

References

- [1] S. Imai, A. Galli, and C. A. Varela. Dynamic data-driven avionics systems: Inferring failure modes from data streams. In *Dynamic Data-Driven Application Systems (DDDAS 2015)*, Reykjavik, Iceland, June 2015.
- [2] S. Imai, R. Klockowski, and C. A. Varela. Self-healing spatio-temporal data streams using error signatures. In *2nd International Conference on Big Data Science and Engineering (BDSE 2013)*, Sydney, Australia, December 2013.
- [3] S. Imai, S. Patterson, and C. A. Varela. Cost-efficient high-performance internet-scale data analytics over multi-cloud environments. In *15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2015)*, Shenzhen, China, May 2015.
- [4] S. Imai, S. Patterson, and C. A. Varela. Elastic virtual machine scheduling for continuous air traffic optimization. In *16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2016)*, Cartagena, Columbia, May 2016.
- [5] S. Imai and C. A. Varela. A programming model for spatio-temporal data streaming applications. In *Dynamic Data-Driven Application Systems (DDDAS 2012)*, pages 1139–1148, Omaha, Nebraska, June 2012.
- [6] S. Imai and C. A. Varela. Programming spatio-temporal data streaming applications with high-level specifications. In *3rd ACM SIGSPATIAL International Workshop on Querying and Mining Uncertain Spatio-Temporal Data (QeST) 2012*, Redondo Beach, California, USA, November 2012.
- [7] R. S. Klockowski, S. Imai, C. Rice, and C. A. Varela. Autonomous data error detection and recovery in streaming applications. In *Proceedings of the International Conference on Computational Science (ICCS 2013). Dynamic Data-Driven Application Systems (DDDAS 2013) Workshop*, pages 2036–2045, May 2013.