Machine Learning Models to Enhance the Science of Cognitive Autonomy

Ganapathy Mani[*], Bharat Bhargava[†], Pelin Angin[‡],
Miguel Villarreal-Vasquez[§], Denis Ulybyshev[¶]
Department of Computer Science & CERIAS
Purdue University
West Lafayette, USA
manig@purdue.edu[*], *bbshail@purdue.edu*[†], *pangin@purdue.edu*[‡],
mvillar@purdue.edu[§], *dulybysh@purdue.edu*[¶]

Donald Steiner[*], Jason Kobes[†]
NGC Research Consortium
Northrop Grumman Corporation
McLean, USA
Donald.Steiner@ngc.com[†], *Jason.Kobes@ngc.com*[†]

***Abstract*—Intelligent Autonomous Systems (IAS) are highly cognitive, reflective, multitask-able, and effective in knowledge discovery. Examples of IAS include software systems that are capable of automatic reconfiguration, autonomous vehicles, network of sensors with reconfigurable sensory platforms, and an unmanned aerial vehicle (UAV) respecting privacy by deciding to turn off its camera when pointing inside a private residence. Research is needed to build systems that can monitor their environment and interactions, learn their capability as well limitations, and adapt to meet the mission objectives with limited or no human intervention. The systems should be fail-safe and should allow for graceful degradations while continuing to meet the mission objectives. In this paper, we propose new methodologies and workflows, and survey the existing approaches and new ones that can advance the science of autonomy in smart systems through enhancements in real-time control, auto-reconfigurability, monitoring, adaptability, and trust.**

***Keywords*-cognitive autonomy; autonomous systems; trust; reinforcement learning; deep learning; trusted computing; blockchain; clustering;**

I. Introduction

Systems with smart autonomy should be capable of exhibiting high-level understanding of the system beyond their primary actions and their limitations and capacity [1]. They should predict possible errors, initiate backup plans, and adapt accordingly. They should be able to multitask: collaborating with their neighboring entities and human counterparts, communicating, and executing actions and progress processes in parallel. A smart system is also required to monitor its interactions with the environment, identify current and potential problems, optimize, reconfigure, and fix those problems autonomously, while improving its operations overtime. A comprehensive IAS should be rich in discovered knowledge on which it can reason with that knowledge at various levels of abstraction using several quantitative and qualitative models: semantic, probabilistic, ontological, symbolic, and even, commonsense. Hence, an IAS is contingent on its cognizance of its operational boundaries, operating environment, and interactions with clients and other services. An IAS should demonstrate reflexivity implying that it continuously adjusts its behavior and adapts to uncertain situations. It should have reasoning where it can introspect about its own reasoning limitations and capacity.

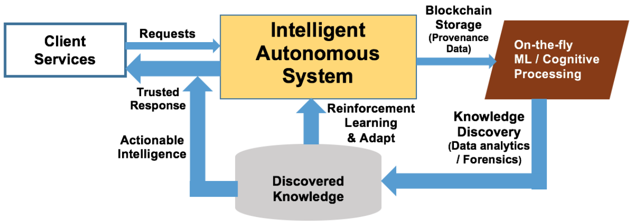
Client Services
Requests
Trusted Response
Intelligent Autonomous System
Blockchain Storage (Provenance Data)
On-the-fly ML / Cognitive Processing
Reinforcement Learning & Adapt
Actionable Intelligence
Knowledge Discovery (Data analytics / Forensics)
Discovered Knowledge

Figure 1. Conceptualization of Comprehensive Intelligent Autonomous Systems (IAS)

These characteristics lead to the following research questions:

- how to enhance the cognizance of IAS using novel cognitive processing approaches that enable the system to be aware of the underlying operating and client context where the data is being generated.
- how to conduct distributed processing of streaming data on-the-fly (and in parallel) in order to apply advanced analytics techniques and machine learning models for knowledge discovery.
- how to model new analytics for finding underlying patterns and anomalies in new and unknown data, thus increasing the value of the gathered data
- how to facilitate learning from data to improve the adaptability (reflexivity) of the IAS
- how to apply blockchain technology in order to provide trust and verifiability to IAS through data provenance
- how to contribute to representation and reasoning approaches based on both qualitative and quantitative models—probabilistic, ontological, semantic, and commonsense—to discover new knowledge, and finally
- how to advance science of learning algorithms to enable autonomy in self-optimization, self-healing, self-awareness, and self-protection, and to reason about making decisions under uncertainties.

In this paper, we propose several methodologies and workflows to enhance cognitive autonomy. Figure 1 conceptualizes comprehensive IAS.

1) We employ deep learning techniques in our model on sensor and provenance data to learn and understand the underlying patterns of interaction, conduct forensics to detect anomalies, and provide assistance in decision making by on-the-fly semantic and probabilistic reasoning.
2) We apply advanced data analytics techniques (such as aggregated analytics over untrusted distributed environment) to incomplete and hidden raw system data (provenance data, error logs, etc.,) to discover new knowledge that contributes to the success of the IAS mission.
3) We enhance the autonomous systems self-awareness, self-protection, self-healing, and self-optimization by learning from the knowledge discovered in real-time.
4) We utilize blockchain technology for storing provenance data for providing monitoring, trust, and verification.

II. System Description

We propose a novel approach that performs on-the-fly analytics on data streams gathered from sensors/monitors of autonomous systems to discover valuable knowledge, learn from the systems interactions with the runtime environment and adapt its actions in a way to maximize its benefits over time for enhanced self-awareness and auto-configuration capability, and track the provenance of the data gathered/generated by the system to provide increased trust in the actions of the system. By integrating components for streaming data analytics, cognitive computing with deep reinforcement learning and knowledge discovery through unsupervised/supervised learning on streamed data, the proposed model aims to provide a unified architecture for smart autonomy, applicable to various systems. The overall architecture of the proposed model is demonstrated in Figure 2.

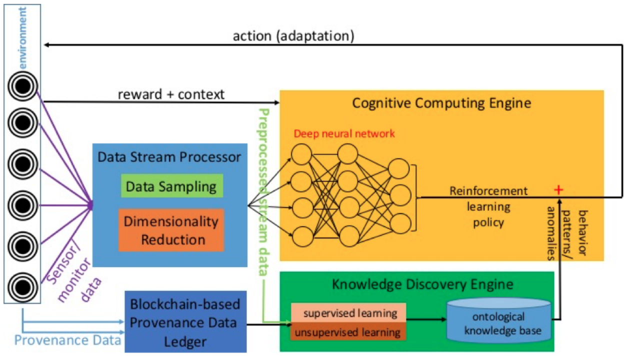

Figure 2. Conceptualization of Comprehensive Intelligent Autonomous Systems (IAS)

General characteristics of the proposed solution are as follows:

- Data obtained through the sensors or monitors of the autonomous system are fed into data stream processor, which contains modules for pre-processing of the data to prepare it for analytics to derive valuable knowledge. The dimensionality of the data is reduced and data is sampled to allow for real-time processing.
- The pre-processed data is fed into the data analytics module (knowledge discovery engine), which applies unsupervised deep learning algorithms to detect deviations from the normal behavior of the system. The gathered data is used to build a model of the systems environment and actions by storing it in a knowledge discovery module, which is consulted repeatedly through the lifetime of the system, acting like the memory of a human-being to decide which actions to perform under different contexts.
- The provenance of the data gathered by the sensors or monitors of the system is logged in an immutable private ledger based on the blockchain technology. This provides verifiability of the data which is used in the knowledge discovery process. It helps in building and measuring the level of trust of an IAS.
- The data pre-processed by the data stream processor and the provenance data are fed into the cognitive computing engine, forming the observations for reinforcement learning in the system, so that the system gains self-awareness over time through a reward-based process. The reward can be based on the type of the system; for a UAV, it could be based on the quality of image processing, while for a missile defense system it could be accuracy and time needed to mitigate an attack. The reinforcement learning process utilizes deep neural networks to build a model of the big data gathered, rather than utilize a trial-and-error learning approach. This enables the system to gain increased self-awareness in time, and gain auto-configuration or self-healing abilities. The system acts upon its environment based on the outcomes of the reinforcement learning and knowledge discovery processes, keeping it in an action-value loop as long as it functions.

III. Related Work

A concept generation system for cognitive robotic entities is implemented by Algorithm of Machine Concept Elicitation (AMCE) [2]. AMCE enables autonomous concept generation based on collective intention of attributes and attributes elicited from formal and informal definitions in dictionaries. In [3], a bio-inspired autonomous robot with spiking neural network (SNN) is built with a capability of implementing the same SNN with five variations through conditional learning techniques: classical conditioning (CC) and operant conditioning with reinforcement or punishment and positive or negative conditioning. A wide-band autonomous cognitive radio (WACR) has been designed and implemented for anti-jamming in [4]. The system has the collected data on spectrum acquisition as well as the location of the sweeping jammer. This information and reinforcement learning is used to learn the perfect communication mode to avoid the jammer. Here, the system is self-aware about the current context. To conduct data analytics on-the-fly and change the analytics techniques automatically, an instrumented sandbox and machine learning classification for mobiles is implemented in [5]. The analysis is conducted, adjusted, and readjusted based on the information of mobile applications submitted by the subscribers. There are well-known knowledge discovery mechanisms that can be applied on raw data to discover patterns. In [6], the authors outline scalable optimization algorithms and architectures encompassing advanced versions of analytics techniques such as principle component analysis (PCA), dictionary learning (DL), and compressive sampling (CS).

A lightweight framework for deep reinforcement learning is presented in [7]. The learning algorithm uses the asynchronous gradient descent for optimization of deep neural networks. In [8], the authors introduce an agent that maximizes the reward function by continuous reinforcement learning with an unsupervised auxiliary task. Reinforcement learning is one of the major machine learning methods that is used primarily on automated cyber physical systems such as autonomous vehicles [9] and unmanned aerial vehicles (UAVs) [10]. Defender-and-attacker game, a game theoretic approach, is employed in general learning models of security as well. When the attacker information is very limited and attacker persistently makes her moves (in the game) to affect the system, the defender needs to constantly adapt to the attackers novel strategies. So the defender constantly reinforces her beliefs based on the attacker moves and creates a robust defense strategy for future attacks [11]. Reinforcement learning algorithms to enhance automated decision making and dynamic reconfiguration capabilities to increase the reflexivity of the system.

Data provenance is used in forensics and security for providing robust support for the underlying systems, sometimes autonomous, through valuable meta-information about the system and its interactions [12]. Data provenance has been modeled for and used in autonomous systems in service-oriented architecture [13] and autonomous information systems [14]. Further investigation is needed to model the use of provenance in enabling autonomy. The Database-Aware Provenance (DAP) architecture [15] provides a workflow that detects the addition of any new autonomous unit of work for fielding any service request and tracks its activities to extract the relevant operational semantics. Provenance data is also used to enhance trust and security in autonomous systems. Trust in information flow can be maintained and verified by provenance data [16], where trust of autonomous entities can be quantified by data provenance and internal values of the data items. Piercing perimeter defenses in autonomous systems can be resolved by provenance-aware applications and architectures [17]. To enable autonomy, systems must be able to reason about and represent provenance data at multiple levels of abstraction. Quantitative and qualitative reasoning can enable semantic knowledge discovery and predictable events. Semantic ontologies are widely used in autonomous cyber-physical systems (CPS) [18]. Ontology-like reasoning over several intelligence representations of new entities can enable the autonomous system to reason about unexpected entities present in their environment [19]. A recent study [20] shows that trust and immutability are provided through provenance on blockchain technology, where smart contracts can be created. This increases trust, provides consensus, and reduces the need for third party intervention: creating a decentralized autonomous setting. Provchaina blockchain-based data provenance architecture is proposed in [21] to provide enhanced availability and privacy in cloud environments. Blockchain provides integrity to provenance data through its immutable property.

IV. Components of IAS

The quality and trustworthiness of data in an IAS is of prime importance for achieving the above mentioned goals. The following data storage/sharing technologies and data sources when modeling the system.

Active Bundle (AB) prototype system: Data are stored in the Active Bundle [22], which is a selfprotected structure that contains encrypted data items, access control policies, and a policy enforcement engine. It assists in privacy preserving data dissemination. This system can be used to deal with all data generated and monitored in IAS and its interactions with outside entities.

Provenance data: In the Active Bundle scheme, provenance meta data is generated, attached to an AB and sent to a central monitor each time a service accesses data. Provenance metadata contains information on when data was accessed, where, by whom, as well as several execution environment parameters, such as OS version, Java version, libraries, CPU model at data recipient's side. Using provenance as a basis for decision making largely depends upon the trustworthiness of provenance. We can deploy Active Bundle as used in AB and blockchain storage for provenance data in order to provide trust and integrity to IAS.

Monitoring Data: Log files are one of the most numerous data collection methods to record activities, user-and-system generated errors, notifications, transactions, interaction with third parties, etc., . Employing advanced data analytics techniques can provide us with rich knowledge of patterns and anomalies. We intend to use the log files of the AB system. Analytics on numerical data from sensors/monitors of autonomous systems can be used to verify the convergence of reinforcement algorithms.

The individual components of the proposed smart autonomy model are described in the subsections below.

A. Cognitive Autonomy

An IAS in a distributed environment should be aware of its three major system, software, and interaction layers (Figure 3):

1) its own state of the system and software as well as operational parameters
2) state of its neighboring systems
3) client or third party services and their interactions with the system

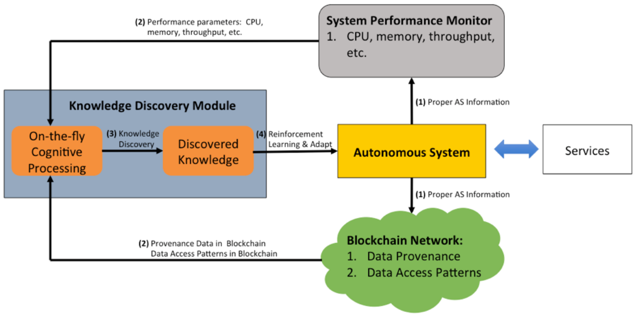
(2) Performance parameters: CPU, memory, throughput, etc.
System Performance Monitor
1. CPU, memory, throughput, etc.
(1) Proper AS Information
Knowledge Discovery Module
On-the-fly Cognitive Processing
(3) Knowledge Discovery
Discovered Knowledge
(4) Reinforcement Learning & Adapt
Autonomous System
Services
(1) Proper AS Information
(2) Provenance Data in Blockchain Data Access Patterns in Blockchain
Blockchain Network:
1. Data Provenance
2. Data Access Patterns

Figure 3. Conceptualization of Comprehensive Intelligent Autonomous Systems (IAS)

Here we propose a novel approach that uses Artificial Intelligence (AI) techniques to monitor and learn the state of autonomous systems to automatically adapt to meet mission objectives with limited or no human intervention. The main idea of the proposed research is actively monitoring the system to provide those results as inputs to decision-making machine learning algorithms that determine the new configuration of the system based on the resulting outputs. Knowledge discovery module focuses on the analysis of two types of data:

1) Performance parameters, such as response time, CPU usage, memory usage, etc., and sensor data particular to the system
2) Data access patterns stored as data provenance in blockchain for misbehavior detection. By integrating system performance and either benign or malicious behavior data in making decisions from past experience the proposed model aims to provide a unified and comprehensive architecture for self-healing IAS.

Deep reinforcement learning is utilized as the primary machine learning technique for cognitive computing in the system to achieve adaptability to different environments, learn from previous vulnerabilities and maximize the security. As stated by Mnih et al. [23], reinforcement learning provides a way to model human behavior in terms of optimizing control of an environment of the agent, through an action-value feedback loop. Reinforcement learning is a difficult task due to the complexity of representing an environment with high-dimensional sensory data. Nevertheless, recent advancements in deep learning allow for building more abstract representations of data from sensors through utilizing multiple levels of nodes, which can be used as the model to optimize the action-value function in the reinforcement learning process. Deep reinforcement learning has recently been successfully applied for tasks like playing Atari games [7].

The deep neural network (DNN) component of the cognitive computing engine can be used to approximate the optimal action-value function for the reinforcement learning model. Deep neural networks also solve the problems of adversarial search and Markov decision processes. The Markov property is nothing but the probability of the current event (E_i) depending on the probability of the previous event (E_{i-1}). With DNNs, we can store and build more memory in the previous state. Through this increased memory, we can build effective Higher-order Markov models, which recollect more data history, enhancing more predictive capability of the system. We can represent the Markov decision process as follows: in the nth Markov model,

$$Pr(E_i|E_{i-1,i-2,\ldots,1}) = Pr(E_i|E_{i-1,i-2,\ldots,i-n})$$

IAS can employ higher-order Markov Decision Processes (MDP) to create novel reinforcement algorithms. For example, consider a smart system executing functionalities in a cloud environment. In the Markov model, there are states before (past, present, and future states) but currently the future states of the system are not only affected by the past state but also affected by the current actions of the client services and the system. There will be a reward function for the autonomous system, and in the transaction, the system must maximize the rewards. Given time (t), actions (A_t), rewards (R_t), and states (S_t), a reinforcement learning model is represented in Figure 4,

$$S_t \xrightarrow[A_t]{R_{t+1}} S_{t+1} \qquad \xrightarrow[A_{t+1}]{R_{t+2}} S_{t+2} \qquad \xrightarrow[A_{t+2}]{R_{t+3}} S_{t+3} \quad \ldots$$

Figure 4. Reinforcement Learning Model

Each state is combined with the actions and maximized reward function, so the system learns which actions to perform to gain more rewards and which actions to reduce the loss. The cognitive computing engine in the proposed research takes as input the data preprocessed by the data stream processor as well as provenance data, which represent the state/observations of the autonomous system for reinforcement learning. The task of the engine is to enable the system to make the best decision for the next action given the context of interaction, the current states of the various system parameters and the knowledge discovered through performing on-the-fly analytics on the streamed data. The overall goal is to select actions in a way to maximize the cumulative QoS parameters that include security and trust, performance, real-time response, and degradation.

B. Knowledge Discovery

The knowledge discovery component of an IAS employs methodologies from pattern recognition, machine learning, and statistics to extract knowledge from raw, and sometimes unknown data. Knowledge discovery is an important element in supporting cognitive autonomy since new knowledge discovered can trigger changes in smart systems to adapt to the new parameters, thus enabling autonomy. Discovered knowledge constitutes the representation of unknown data, its form, and its degree of certainty. The generic process of knowledge discovery is shown in Figure 5 below.

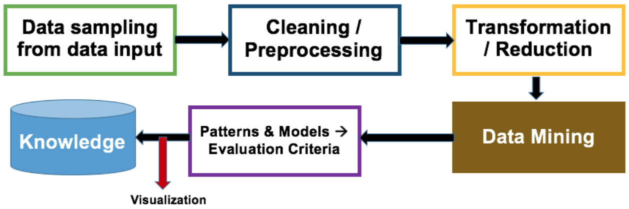
Data sampling from data input
Cleaning / Preprocessing
Transformation / Reduction
Data Mining
Patterns & Models → Evaluation Criteria
Knowledge
Visualization

Figure 5. Knowledge Discovery in Autonomous Systems

Knowledge discovery on large data, in particular streaming data, needs efficient data processing. Distributed data processing on streaming data becomes a necessity for faster classification and storage of data [52]. A parallel processing of data items that can classify and categorize the streaming data considerably fast is embedded in to stream processing unit. The classification and clustering techniques must be capable of on-the-fly processing of data streams: distributed data processing can accommodate simultaneous processing of sequential/parallel data streams: the key idea behind the parallel processing is to host distributed data processing units (DDPU) that can (a) read (R) to load the data, (b) Analyze (A) to process and classify the data, and (c) toggle (T) to shift to/from read or analyze.

DDPU 1 :	**R item 1**	**T→**	**R item 2**	**T→**	**A item 2**
DDPU 2 :	**A item 1**	**T→**	**R item 3**	**T→**	**A item 4**
DDPU 3 :	**R item 4**	**T→**	**…**		
	Cycle 1		**Cycle 2**		**Cycle 3**

Figure 6. Read, Analyze, and Toggle Processing Model

The representation above shows a fundamental distributed data processing technique—*RAT*—to processes data on-the-fly, which is scalable to process Big Data streams. Depending on the priority and availability of the data items each processing unit prioritizes the RAT operation for each data item. In this way, instead of relying on static rules and heuristics to determine prioritization of data processing, we can compute the value of the data on-the-fly based on datas quantitative/qualitative system metrics such as sensitivity, dependence, and importance of the data, and process the data items accordingly.

The processed data can contain both categorized (easy to label) such as data origin, time of creating, and modification, etc., and uncategorized data such as error logs (text). Hence, both customizedcombination of multi-level decision trees and Bayesian probabilistic methodsclassification and regression algorithms [24], and advanced clustering techniques can be applied to achieve high dimensionality and to label the data, and prepare it for analysis. Bayesian statistics can be used to estimate the reliability of the autonomous system and quantify the unknown due to lack of data (missing data). Bayes theorem states that, given two data items D_1 and D_2,

$$Pr(D_1|D_2) = [Pr(D_2|D_1)/Pr(D_2)] * Pr(D_1)...$$

The reliability of the autonomous system can be measured using Bayesian statistical methods with conditional probability and prior distribution of the autonomous systems states. New knowledge can be discovered through reliability analysis of autonomous system, which will contribute to the self-awareness of the system, enabling smart autonomy.

Quantitative and qualitative reasoning can enable semantic knowledge discovery and predictable events. Semantic ontologies are widely used in autonomous cyber-physical systems. Ontologies are applied to generate semantic reasoning over the provenance data. For example, semantic ontology reasoning is used to extract attributes of provider-client interaction such as: platform, data requested, update, and access. Applying semantic reasoning models to the log files of provenance data will help the system discover new knowledge about the client. This will be stored and used to make decision and contribute to autonomy. Of particular interest to the knowledge discovery process in the proposed system are the following methods are integrated into the knowledge discovery engine:

Association Rule Mining: Association rule mining discovers patterns of the form if X then Y, where X and Y are item sets. This allows us to find frequent patterns of co-occurrence in large datasets. Typical algorithms for association rule mining include the Apriori algorithm, sampling algorithm, frequent pattern tree and partition algorithm. For IAS, the mentioned association rule mining algorithms to discover system events that co-occur frequently under normal and anomalous circumstances can be utilized (e.g. CPU and memory usage spiking up together). This will allow the system to have increased awareness of what environment and system conditions to expect when a certain event occurs and adapt itself accordingly.

Clustering: Clustering allows us to partition data without having a training sample, which is useful in situations where the system has just started functioning and we need to discover groups of events/data similar to each other in terms of certain parameters, representing different states of the system. Finding clusters of IAS data along various dimensions will allow for detection of anomalies when incoming data does not belong to any of the previously built clusters. This is also useful for discovering cases like zero-day attacks, which have no known attack signature through detecting deviations from the normal behavior of the system.

Sequential/Temporal Pattern Mining: Sequential/temporal pattern mining discovers patterns in a dataset that occur frequently in a particular sequence. The gold standard for time series analysis is Hidden Markov Models (HMM), therefore HMM is used to build a representation of IAS behavior through observation of the system states and state transitions over time. Based on HMM, the system can be in one of the N possible states $\{S_1,\ S_2,\ ,\ S_N\}$, and undergoes a transition from one state to another at particular times. The state transition probabilities of the system depends on the immediate past, i.e.

$$P(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k) = P(q_t = S_j | q_{t-1} = S_i)$$

Additionally, the observations (data gathered through sensors/monitors) are a probabilistic function of each state, i.e.

$$P(o_t = v_k | q_t = S_j)$$

where o_t is the data observed at time t and v_k is a distinct observation in the set of possible observations for the system. Using HMM, knowledge discovery module contains a probabilistic model of the system from a sequential set of observations/data, which best explains the behavior of the system in terms of transitioning between different states and the data resulting from the transitions. For example, a low CPU usage observation can be associated with a malfunctioning module state with high probability, while an extremely high CPU usage observation can be associated with a system under attack state. Based on the knowledge discovered over time with HMM, the IAS will be able to predict current and next states more accurately, and take adaptability actions accordingly. Critical node analysis in higher order Markov models can lead to identifying critical steps in complex attack strategies of adversaries, reducing resource usage for target analysis. Once the pattern is discovered, the systems can reinforce its understanding and adapt to the new set up.

In addition to the above mentioned techniques, various models for detection of outliers in different types of data have been devised by the machine learning community. While supervised and unsupervised learning models have been applied with success to a variety of domains, robust models for detecting anomalies and failures in IAS operation are still lacking. The main shortcoming of supervised anomaly detection models are that they require a large amount of training data and can only provide accurate results on anomalies that were previously observed in the system. This makes such models unable to capture threats/anomalies that are completely new, which is essential in an environment of ever-growing security vulnerabilities and attacks. A significant advantage of unsupervised models is that the training data required is gathered from the behavior of services operating under normal conditions (possibly in an isolated environment/private cloud); i.e. no attack data is required to train these models.

C. Reflexivity of the System

The goals of IAS in the proposed approach are (1) replacing anomalous / under-performing modules with reliable versions or adapting to a new mechanism to avoid anomalies, (2) reconfiguring system parameters to respond to anomalous system behavior, (3) swiftly self-adapting to changes in context, (4) enforcing proactive and reactive response policies to achieve performance and security goals, and (5) achieving continuous availability even under attacks and failures. Providing adaptability in order to achieve increased autonomy in IAS relies on two main elements:

1. Being cognitive and determining action: Monitoring of systems is of utmost importance in achieving high self-awareness, as systems in environments with highly dynamic contexts may exhibit frequent changes in many QoS parameters. We measure the assurance level, (integrity/accuracy/trust) of the system from the performance parameters such as response time, throughput, packet loss, delays, consistency, acceptance test success, etc. Compliance with all the requirements of IAS is hard to achieve in such dynamic environments, making monitoring a must for accurate decision-making. The tasks involved in effective monitoring and analysis of the obtained data include the following: (a) identification of QoS metrics, such as response time, CPU usage, memory usage, etc., to determine the performance and behavior of IAS; (b) development of models for identifying deviations from performance (e.g., achieving the total response time below a specific threshold) and security goals (e.g., having trust levels above a certain threshold).

2. Autonomous system reconfiguration based on changes in context: Changes in the context of IAS can affect system behavior, requiring autonomous reconfiguration. While changes in user context can result in updated priorities such as trading accuracy for lower response time in an emergency, changes in system context can result in failures requiring the restart of a component of the IAS. Dynamic reconfiguration of system modules based on the updated constraints and contexts enables success of mission objectives.

Adaptability allows dynamic configuration of software and execution to meet the changing demands of autonomous systems for performance, reliability, security, and resource utilization. Adaptable systems provide graceful degradation and can respond to the timing, duration, type, extent, severity of failures and attacks. Adaptation must satisfy the consistency and integrity constraints. The granularity of formally defined classes of algorithms will determine the overhead and benefits of adaptation. One of them is based on graceful degradation. The main idea is having primary and alternate modules and using an acceptance test to validate their operation. Initially a primary module is used and constantly tested. In case of failure there are two alternatives: (1) weaken the acceptance test or (2) replace the primary module with the alternate/replica that can pass the acceptance test. Figure 7 illustrates the concept. In the case that an alternate module replaces the primary module of the IAS not able to pass an acceptance test, the composition of a process in the IAS can change as shown in the lower part of the figure (Note that here the system has two module alternatives for a process A, which invokes a process B with three module alternatives, that further invokes process M having three module alternatives chosen based on the acceptance test process in the upper part of the figure).

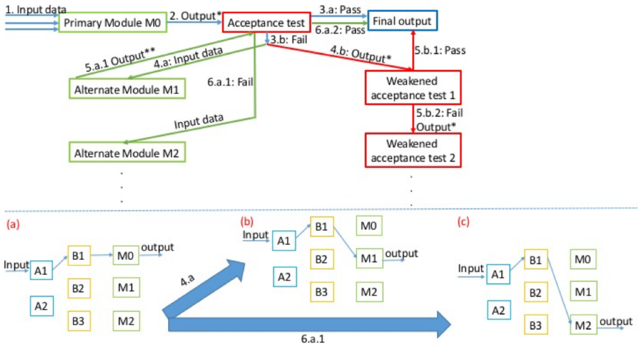

Figure 7. Dynamic Adaptation based on Recovery Block Scheme

Adaptable autonomous systems should be able change their system configuration to guarantee mission critical operations at the cost of sacrificing performance. Because some services may continue their effort to compromise these systems there exists a need for more adaptable solutions to protect systems. Our proposed Moving Target Defense (MTD)-type is a defensive strategy that aims to reduce the need to continuously fight against attacks by decreasing the gain-loss balance perception of attackers. The framework narrows the exposure window of a node (module) to such attacks, which increases the cost of attacks on a system and lowers the likelihood of success and the perceived benefit of compromising it. The achieved reduction in the vulnerability window makes this strategy optimum for adaptable autonomous systems. The proposed framework introduces reflexivity and adaptability to systems. Reflexivity has two main components: (1) continuing operation and (2) adapting to counter anomalies. The MTD-style approach takes into consideration these components since it transforms systems to be able to adapt and self-heal when ongoing anomalies are detected, which guarantees operation continuity.

D. *Trust in Autonomous Systems*

Self-protection (automatic identification and protection from security threats) and self-healing (automatic fault discovery and correction) are important properties of an IAS. We propose an approach for data provenance with blockchain-based mechanisms to build trustworthiness of the data and ensure identities of network participants. Integrity of data will be guaranteed by blockchain technology. Data can be used for threat detection. Provenance record contains information on what data type has been accessed / updated, by whom (by which service), when and who sent the Active Bundle to the service.

Challenges of blockchain technology deployment: **Performance:** Blockchain is replicated to all the network participants and this imposes a performance overhead. **Access Control (Read):** In case of access revocation or subjects role change, access to data must be revoked immediately within an information system when authorization is no longer valid. However, revoked access to data on a blockchain can be bypassed in the following ways: (a)by replaying old blocks against an empty blockchain and stopping before the revocation block is appended and (b) an attacker holding a copy of a blockchain could use a modified client to just ignore the revocation block. Even if read access to local blockchain requires an off-chain token handshake with a centralized authority for authorization; then that token would continue to work forever in the future. The requirement to revoke previously granted access can be bypassed by rolling the local clock back and restoring unauthorized access to blockchain data.

V. Conclusion

We propose a comprehensive approach to attain cognitive autonomy in smart cyber systems.The proposed workflow operates in stages where each module solves specific problems and provide the output to the next module, which creates a streamlined and accurate learning and prediction model. Stream processing module handles continuous data flow with *RAT* processing, knowledge discovery module identifies patterns in processed data, recognized patterns are utilized by cognitive computing engine to learn and predict future operational contexts and inputs, and based on the predictions IAS reflect to adapt to the operational contexts by retraining and testing. The model can also adapt to situations where it faced with unknown raw data that was not present in training or testing datasets. Through deep learning the model can autonomously retrain and recognize the new data. The framework provides integrity and verifiability to IAS by utilizing blockchain mechanism. This theoretical framework can be expand to use Internet of Things (IoT) as a methodological concept to enable cognitive autonomy. Diverse extensions are possible with this model to various types of setups in autonomous systems. As a future work, we will be implementing cognitive computing engine with reflexivity property for IAS.

Acknowledgment

This research is funded by NGC Research Consortium.

References

[1] NSF, “Smart and Autonomous Systems (S&AS),” 2017, Program Solicitation NSF 16-608. [Online]. Available: https://www.nsf.gov/pubs/2016/nsf16608/nsf16608.pdf

[2] O. A. Zatarain and Y. Wang, “Experiments on the supervised learning algorithm for formal concept elicitation by cognitive robots,” in *Cognitive Informatics & Cognitive Computing (ICCI* CC), 2016 IEEE 15th International Conference on*. IEEE, 2016, pp. 86–96.

[3] E. Dumesnil, P.-O. Beaulieu, and M. Boukadoum, “Single snn architecture for classical and operant conditioning using reinforcement learning,” *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, vol. 11, no. 2, pp. 1–24, 2017.

[4] S. Machuzak and S. K. Jayaweera, “Reinforcement learning based anti-jamming with wideband autonomous cognitive radios,” in *Communications in China (ICCC), 2016 IEEE/CIC International Conference on*. IEEE, 2016, pp. 1–5.

[5] T. H. Titonis, N. R. Manohar-Alers, and C. J. Wysopal, “Automated behavioral and static analysis using an instrumented sandbox and machine learning classification for mobile security,” Jun. 6 2017, uS Patent 9,672,355.

[6] K. Slavakis, G. B. Giannakis, and G. Mateos, “Modeling and optimization for big data analytics:(statistical) learning tools for our era of data deluge,” *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 18–31, 2014.

[7] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, “Asynchronous methods for deep reinforcement learning,” in *International Conference on Machine Learning*, 2016, pp. 1928–1937.

[8] M. Jaderberg, V. Mnih, W. M. Czarnecki, T. Schaul, J. Z. Leibo, D. Silver, and K. Kavukcuoglu, “Reinforcement learning with unsupervised auxiliary tasks,” *arXiv preprint arXiv:1611.05397*, 2016.

[9] D. Meyer, J. Feldmaier, and H. Shen, “Reinforcement learning in conflicting environments for autonomous vehicles,” *arXiv preprint arXiv:1610.07089*, 2016.

[10] H. Rastgoftar and E. M. Atkins, “Unmanned vehicle mission planning given limited sensory information,” in *American Control Conference (ACC), 2017*. IEEE, 2017, pp. 4473–4479.

[11] Z. Hu, M. Zhu, P. Chen, and P. Liu, “On convergence rates of robust adaptive game theoretic learning algorithms,” *arXiv preprint arXiv:1612.04724*, 2016.

[12] B. Glavic, “Big data provenance: Challenges and implications for benchmarking,” in *Specifying big data benchmarks*. Springer, 2014, pp. 72–80.

[13] S. Miles, S. Munroe, M. Luck, and L. Moreau, “Modelling the provenance of data in autonomous systems,” in *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*. ACM, 2007, p. 50.

[14] T. Malik, L. Nistor, and A. Gehani, “Tracking and sketching distributed data provenance,” in *e-Science (e-Science), 2010 IEEE Sixth International Conference on*. IEEE, 2010, pp. 190–197.

[15] A. Bates, W. U. Hassan, K. Butler, A. Dobra, B. Reaves, P. Cable, T. Moyer, and N. Schear, “Transparent web service auditing via network provenance functions,” in *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017, pp. 887–895.

[16] E. Bertino, “Data trustworthinessapproaches and research challenges,” in *Data privacy management, autonomous spontaneous security, and security assurance*. Springer, 2015, pp. 17–25.

[17] T. Moyer, K. Chadha, R. Cunningham, N. Schear, W. Smith, A. Bates, K. Butler, F. Capobianco, T. Jaeger, and P. Cable, “Leveraging data provenance to enhance cyber resilience,” in *Cybersecurity Development (SecDev), IEEE*. IEEE, 2016, pp. 107–114.

[18] S. Amrouch, S. Mostefai, and M. Fahad, “Decision trees in automatic ontology matching,” *International Journal of Metadata, Semantics and Ontologies*, vol. 11, no. 3, pp. 180–190, 2016.

[19] L. Zhao, R. Ichise, S. Mita, and Y. Sasaki, “An ontology-based intelligent speed adaptation system for autonomous cars,” in *Joint International Semantic Technology Conference*. Springer, 2014, pp. 397–413.

[20] G. Gordon, “Provenance and authentication of oracle sensor data with block chain lightweight wireless network authentication scheme for constrained oracle sensors,” 2017.

[21] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla, “Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability,” in *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. IEEE Press, 2017, pp. 468–477.

[22] L. Lilien and B. Bhargava, “A scheme for privacy-preserving data dissemination,” *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 36, no. 3, pp. 503–506, 2006.

[23] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, p. 529, 2015.

[24] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.