SYM-AM-23-105

Excerpt from the Proceedings of the Twentieth Annual Acquisition Research Symposium

Acquisition Research: Creating Synergy for Informed Change

May 10–11, 2023

Published: April 30, 2023

Prepared for the Naval Postgraduate School, Monterey, CA 93943.

Acquisition Research Program
Department of Defense Management
Naval Postgraduate School

The research presented in this report was supported by the Acquisition Research Program at the Naval Postgraduate School.

To request defense acquisition research, to become a research sponsor, or to print additional copies of reports, please contact any of the staff listed on the Acquisition Research Program website (www.acquisitionresearch.net).

Acquisition Research Program
Department of defense management
Naval Postgraduate School

Using Digital Twins to Tame the Testing of AI/ML Systems

David G. Zurn—received his Bachelor and Master of Science degrees in Electrical Engineering from the Georgia Institute of Technology in 1985 and 1990 respectively. Since joining GTRI's Electronic Systems Laboratory (ELSYS) in 2003, he has worked on a variety of EW-related research efforts including Radar Warning Receiver hardware and software development and test, Missile Warning System hardware and software test, and development of Hardware in the Loop (HITL) test solutions tailored to EW applications. Zurn is currently serving as the Division Chief of the Test Engineering Division within ELSYS. Zurn is a lecturer for the RWR Design short course offered through GT's Professional Education program. His recent research interest areas are Cognitive EW T&E and Space EW T&E. [David.zurn@gtri.gatech.edu]

Dr. Craig Arndt—currently serves as a principal research engineer on the research faculty of the George Tech Research Institute (GTRI) in the System Engineering Research division of the Electronic Systems Lab. Dr. Arndt is a licensed Professional Engineer (PE), and has over 40 years of professional engineering and leadership experience. Dr. Arndt holds engineering degrees in electrical engineering, systems engineering, and human factors engineering and a Master of Arts in strategic studies form the U.S. Naval War college. He served as Professor and Chair of the engineering department at the Defense Acquisition University, and as technical director of the Homeland Security FFRDC at the MITRE Corporation. In industry he has been an engineering manager, director, vice president, and CTO of several major defense companies. He is also a retired naval officer. [Craig.Arndt@gtri.gatech.edu]

Jeremy Werner—Ph.D., ST was appointed DOT&E's Chief Scientist in December 2021 after initially starting at DOT&E as an Action Officer for Naval Warfare in August 2021. Before then, Jeremy was at Johns Hopkins University Applied Physics Laboratory (JHU/APL), where he founded a data science-oriented military operations research team that transformed the analytics of an ongoing military mission. Jeremy previously served as a Research Staff Member at the Institute for Defense Analyses where he supported DOT&E in the rigorous assessment of a variety of systems/platforms. Jeremy received a PhD in physics from Princeton University where he was an integral contributor to the Compact Muon Solenoid collaboration in the experimental discovery of the Higgs boson at the Large Hadron Collider at CERN, the European Organization for Nuclear Research in Geneva, Switzerland. Jeremy is a native Californian and received a bachelor's degree in physics from the University of California, Los Angeles, where he was the recipient of the E. Lee Kinsey Prize (most outstanding graduating senior in physics). [jeremy.s.werner.civ@mail.mil]

Abstract

Program test managers and test engineers should carefully consider Digital Twinning approaches for addressing training and testing challenges for Artificial Intelligence/Machine Learning (AI/ML) systems. A hybrid Hardware in the Loop (HITL) and Digital Twin (DT) architecture is discussed for a notional Cognitive EW system. This architecture may provide effective training and testing for complex AI/ML systems that incorporate extensive Cyber-Physical interactions. Considerations for generating realistic RF test environments for Cognitive EW systems are also considered.

Keywords: Digital Twin, AI/ML, Cognitive EW, HITL

Executive Summary

This research investigates the challenges associated with testing and training of AI/ML systems in the Electronic Warfare (EW) domain and how these challenges can be addressed using Digital Twins. The specific AI/ML testing and training challenges were identified during a Cognitive EW T&E working group conducted by GTRI while under contract to DOT&E. Several key DT capabilities are identified for addressing AI/ML training and testing challenges –

1. Simulation of the system and its operational environment with sufficient realism
2. Ability of the DT to create training and testing data
3. Ability to efficiently virtualize hardware models, system firmware, and software components into the Digital Twin, allowing for efficient Continuous Integration/Continuous Delivery (CI/CD)

To better understand whether a DT can provide these capabilities, a specific detailed Cognitive EW receiver use case is developed. A high-level hybrid HITL DT architecture for this use case is discussed along with specific functional use cases, such as training and testing data set generation and validation, AI/ML component training and DT validation. Using lessons learned from the Cognitive EW Receiver use case, considerations and limitations for using DT for the Cognitive EW Receiver are discussed.

Background

Weapons systems augmented with Artificial Intelligence/Machine Learning (AI/ML) capabilities are a new reality and driven by several trends. The modern battlefield is becoming dependent on connected kill-webs and the Joint All Domain Operations (JADO) environment, which is driving the emergence of AI/ML weapons systems on the Blue and Red side (NASEM, 2021). Indeed, strategic competitors, such as China and Russia, are making significant investments in AI for national security purposes (GAO, 2022a). The rapid explosion of AI/ML in the commercial sector is also enabling the adoption of AI/ML in weapons systems (USAF Chief of Staff, 2020).

According to the GAO, AI/ML is expected to transform all sectors of society, including, according to the Department of Defense (DoD), the very character of war. The failure to adopt and effectively integrate AI technology could hinder national security. As a result, the DoD is investing billions of dollars and making organizational changes to integrate AI into their warfighting plans. A total of almost 700 separate AI/ML programs were identified across the services either funded through R&D or procurement. This does not include classified programs or programs funded through O&M, which would inflate that total (GAO, 2022b). According to a recent National Defense Strategy, “The Department will invest broadly in military application of autonomy, artificial intelligence, and machine learning, including rapid application of commercial breakthroughs, to gain competitive military advantages” (DOD, 2018).

Historically, one of the more significant areas of DOD investment in AI/ML has been in the EW domain. GTRI has been involved in multiple efforts to develop, evaluate, and implement AI/ML algorithms on multiple RF EW systems. EW systems sample the RF environment and benefit from AI/ML capabilities designed to infer the behavior and intent of threat Radar waveforms in adversarial conditions. The remainder of this paper will consider AI/ML efforts specifically in that arena.

AI/ML EW T&E Challenges

GTRI, under contract to DOT&E Test and Evaluation Threat Resource Activity (TETRA), conducted a five session Cognitive EW T&E Working group in 2020–2021 to explore AI/ML T&E challenges for Cognitive EW systems. A variety of stakeholders from the AI/ML research community, the DOD T&E community, and acquisition and sustainment community gathered to identify Cognitive EW T&E challenges, gaps, and potential solutions. The working group findings relating to T&E challenges are summarized in Figure 1.

AI/ML systems present a unique set of test challenges. The massive coverage space and wide range of potential behaviors are difficult to address via legacy test methods. Major AI/ML T&E challenges are summarized as follows.

A. **Massive Coverage Space** - Extensive analysis has been done for the autonomous driving use case, specifically looking at testing for AI/ML techniques such as Deep Neural Nets (DNN). For complex systems these DNNs can be very high order non-linear functions. Common test issues arising from these functions are massive, multi-dimensional input–output coverage spaces. This creates issues such as how to optimize/efficiently explore these spaces during test, how to efficiently create test data, and whether it is possible to create a test oracle to determine whether the test has passed or failed (Tian, 2018).

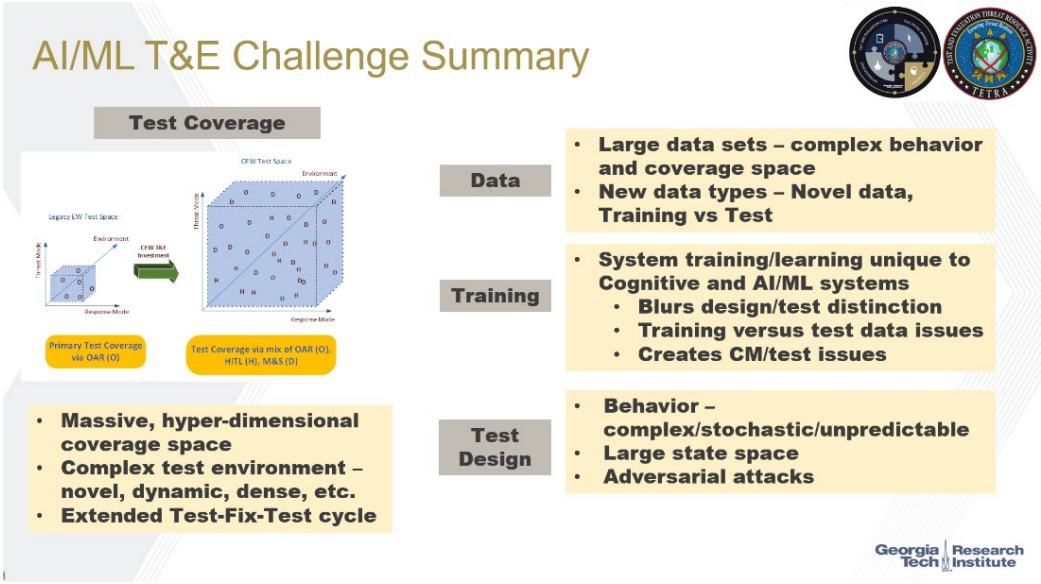

Figure 1 - AI/ML T&E Challenge Summary

B. **Unique quality parameters** - AI/ML systems, particularly AI/ML software, present new quality parameters or measures of performance associated with learning such as correctness, accuracy, explainability, system stability, timeliness, and robustness that are not typically considered. Rigorous definitions and processes relevant to DoD test systems need to be developed to address these new parameters (Chuanqi Tao, 2019).

C. **Adversarial exploits** - AI/ML systems require adversarial testing approaches to ensure that during operations adversarial manipulation of data does not affect the system in unpredictable ways (Prokhorov, 2019). Extensive research is underway to generate adversarial exploit data for improved system testing (Anthony Ortiz, 2018).

D. **Assurance case testing** - For AI/ML enabled autonomous systems, testing to assure safe operation can become an issue. According to a RAND study, the current state of T&E for AI technologies cannot ensure the performance and safety of AI systems, especially those that are safety-critical. Assurance case testing is required for these types of systems (RAND, 2021). An assurance case is “a structured argument that the system is sufficiently dependable to permit fielding in a specific operational context” (Tate, 2019). AI/ML systems exhibiting autonomy

require assurance cases because they are unpredictable due to the following attributes:

1. State space explosion
2. Non-smooth or fractal response
3. Lack of transparency
4. Changing system behaviors over time
5. Emergent behaviors

E. **Continuous test** - Another unique challenge relates to the fact that AI/ML systems require train-test cycles throughout the system life cycle. This is due to the need to continuously train AI/ML components to cope with environmental and threat changes. Indeed, this is a feature - the system can learn from changes in the environment, but learning must be followed by testing as part of a continuous cycle. These cycles are short in duration and potentially continue through the fielded life cycle of the system. The legacy waterfall and distinct separation of coding/testing/fielding phases are not adequate for AI/ML systems. According to the Defense Science Board (DSB) Summer Autonomy study, to address the train-test cycle challenge, the DoD should look to commercial practices like Agile for developing autonomous, AI/ML based systems. Agile and DevSecOps development practices provide an incremental development approach enabling tight train-test cycles (DSB, 2016).

F. **Data Generation** - Data generation for training and testing of AI/ML algorithms presents a significant T&E challenge. It's been estimated that 80% of the effort required to implement AI/ML systems is involved in data generation, tagging, and curation (Antonio Nieto-Rodriguez, 2023). The difficulty of procuring data depends on the AI/ML application area. For EW-related AI/ML applications, which this paper addresses, data is a significant challenge: collected and recorded raw high-fidelity data is often not tagged and cannot always be correlated with Blue (U.S.), Red (Adversary), and Gray (Commercial) RF sources. Synthetic data can be generated but replicating real-world environmental and propagation effects can be difficult.

These T&E challenges are exacerbated for AI/ML systems involving extensive interaction with the physical world (Autonomous vehicles, Industrial systems, RF systems). The Cyber–Physical interaction via sensors and effectors and the system interaction with the environment are often difficult or impractical to create in the real-world for test purposes. Testing in a real-world operational environment is ideal from a fidelity perspective, but testers face significant challenges generating sufficiently wide test coverage, creating edge cases and assuring the repeatability of complex test scenarios. Synthetic digital environments and DTs are often created to mitigate these challenges.

Digital Twin Overview

According to the Digital Twin Consortium, "A Digital Twin is a virtual representation of a real-world system. A digital twin is synchronized with the physical twin at a specific fidelity and frequency" (Digital Twin Consortium, 2020). The National Institute of Standards and Technology (NIST) definition is "A digital twin is the electronic representation—the digital representation—of a real-world entity, concept, or notion, either physical or perceived' (NIST, 2021). The application and usage of the DT concept varies widely across commercial industry and the DoD. A DOT&E memo assessing the usage of DT in DoD testing shows some progress in the adoption of DT, but it also sharply illustrates how far the DoD has to go:

- Approximately 7% of programs under DOT&E oversight have built or are planning to build a DT.
- Most of the programs that report usage of DTs are applying them for contractor-level testing in support of Engineering Manufacturing Development (EMD) and none have been used DT for operational testing (DOT&E, 2022).

The DoD recognizes the need to accelerate the adoption of DTs. The increasing use of AI/ML introduces "never-before-seen capabilities and vulnerabilities that change at never-before-seen dynamic rates." The DOT&E 2022 strategy defines five strategic pillars to transform T&E, two of which support the use of DTs for testing AI systems – **Accelerate the delivery of weapons** by embracing digital technologies as a key action and **Pioneer the T&E of weapons systems built to change over time** where enabling adequate assessment of AI-enabled weapons systems is one of the desired end states (Sandra Hobson, 2022).

Researchers in the Advanced Driver Assistance Systems (ADASs), Autonomous Vehicles (AVs), and other industries are taking up the usage of DTs and have explored the use of Digital Twins to address AI/ML training and test challenges. Recognizing the challenges of the complex Cyber-Physical interactions involved in these systems, the use of Hybrid DT systems has been considered (Jörn Thieling, 2021; Kirill Semenkov, 2020).

In the AI/ML training and testing context, a Hybrid DT might consist of (1) a real hardware/software system design instantiated in a HITL testbed, (2) a set of digital models and virtualized firmware and software representing that system and the system's operating environment and (3) a method for validating the digital model versus observed system behavior.

The Hybrid DT concept may be able to address some of the difficult AI/ML training and test challenges such as Massive Coverage Space, Continuous Test, and Data challenges outlined above. The following DT capabilities are required to address these challenges:

1. Simulation of the system and its operational environment with enough realism to support AI/ML training and testing to assure performance as expected in a real operational environment
2. Ability of the Hybrid DT operational environment simulation to create trusted training and testing data suitable for the system's AI/ML components
3. Ability to efficiently virtualize system digital models, system firmware, and software components into the Digital Twin, allowing for efficient Continuous Integration/Continuous Delivery (CI/CD)

Next, we'll explore a specific Cognitive EW Receiver use case to evaluate the applicability of the Hybrid DT approach.

AI/ML Training and Test Use case - Cognitive EW Receiver

First, consider the notional EW receiver in Figure 2. The receiver system is segmented into RF input, receiver system, and Pilot Vehicle and Federated systems interfaces.

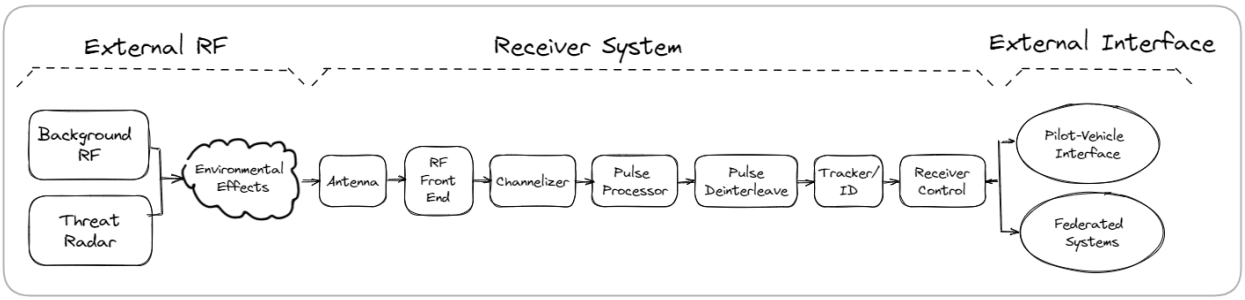

Figure 2 – Notional EW Receiver

The notional EW receiver typically processes and identifies threats using fixed lookup tables and relatively simple signal processing algorithms. This design performs well in simple threat environments where the number of threats is small and the threats produce known, predictable RF waveforms. As the number of threats increase and produce unexpected, unknown RF waveforms, the receiver performance degrades. To counter this problem, receiver designers add AI/ML algorithms to key processing components to improve their overall performance. A notional Cognitive EW Receiver with some of these cognitive components highlighted is shown in Figure 3.

For example, the Pulse Processor Component can be supplemented with a DNN based waveform discriminator. Traditional waveform discriminators measure waveform parameters such as frequency and phase modulation, then determine waveform type using a look-up table or simple heuristic. If these waveform properties are modified by the threat radar in a way that cannot be measured accurately, or measurements fall outside of the bounds of the lookup table, the traditional discriminator will not perform well. The DNN based waveform discriminator performs similarly to a DNN used for image recognition. The DNN ingests waveforms of different modulation types and attempts classification based on observable and latent waveform features. During training, DNN weights are iteratively adjusted to minimize classification error. The DNN can potentially outperform the traditional discriminator because the DNN extends beyond general classification and is able to handle waveforms with parameters that may not match pre-programmed receiver boundaries/features.

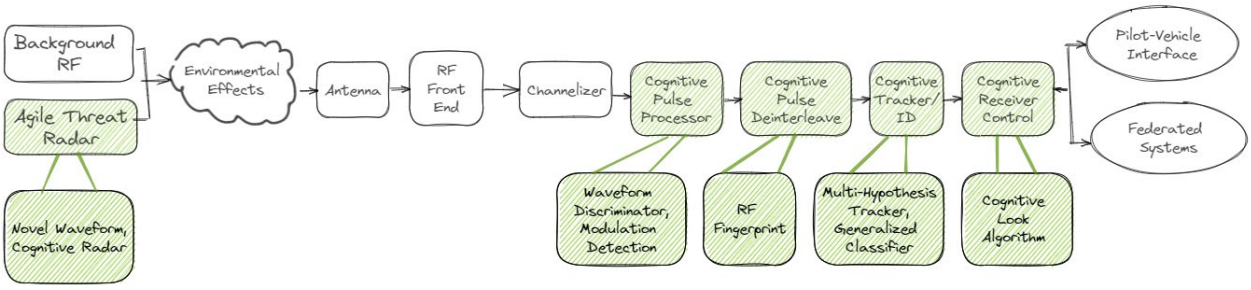

Figure 3 – Notional Cognitive Receiver

Typical receiver components and their AI/ML enhancements are listed in Table 1. This is a notional list—receiver designers may create many other AI/ML enhancements depending on receiver requirements.

These AI/ML components have great potential for increasing performance but come at a price. As discussed above, each component requires training and adds an extra test burden during system development and sustainment.

Table 1 – Cognitive EW Receiver Components

EW Receiver Component	AI/ML Enhancements	Comment
Pulse Processor	Waveform Discriminator	DNN classification based on observable and latent waveform features
Pulse Deinterleaver	RF Fingerprinting	Deinterleave pulse trains using RF features unique to a given RF threat
Tracker/ID	Multi-Hypothesis Tracker (MHT), Generalized classifier	MHT uses Bayesian inference to more accurately establish and maintain tracks; generalized classifier uses DNN or mode-intent algorithms instead of lookup table to identify threat based on class
Receiver Control	Cognitive Look Algorithm	Receiver uses inference engine to optimize receiver frequency look schedule in a dense threat environment

Cognitive Receiver Digital Twin

A Cognitive Receiver DT may be able to address these training and test issues. The DT provides a framework for training and testing individual AI/ML components efficiently. Without the DT, each component must be trained in a stand-alone hardware instantiation or in-situ in the receiver system. This may not seem like a problem, as the receiver developer typically implements stand-alone subcomponents for unit test. However, this is typically done only once during system development. AI/ML training is a continuous activity that needs to be done many times throughout the system's life cycle. It is required during systems development, integration testing, developmental testing, operational testing and system sustainment. It is not practical to create and maintain stand-alone component training setups like this for the entire system life cycle. In-situ training is also impractical. Training requires the introduction of a very large set of inputs to the AI/ML component and adjustment based on component output. Generating this set of inputs through the entire system processing chain is difficult and time-consuming. Moreover, training using real hardware either stand-alone or in-situ can only be done at real-time system operation speed which could be very time-consuming for large datasets. Frequently AI/ML systems are virtualized to enable Faster than Real-Time (FTRT) training.

The DT depicted in Figure 4 is implemented by digitally instantiating each system component using either digitally hosted hardware models or through virtualization of firmware and software. Note that the DT incorporates the complete Cognitive Receiver System and External RF and External Interface elements. Considerations with the Cognitive Receiver System visualization will be discussed, followed by External elements.

The Cognitive Receiver system consists of the antenna, RF front end, the chain of processing elements and the Receiver control block. The antenna and RF front end are modeled using RF modeling tools. Depending on complexity the antenna could be an engineering model based on frequency and polarization dependent azimuth and elevation lookup tables. The RF front end is more problematic as it typically consists of a chain of complex linear and non-linear RF components – limiters, amplifiers, filters and mixers and A/D converters, that can be difficult to accurately model. These components must be

accurately modeled to create a useful DT. Crude engineering-based models will not re-create the RF front end effects found in a real receiver system. These effects, such as noise, harmonics, distortion, ringing and filtering all impact overall receiver performance. If they aren't modeled with sufficient fidelity, the DT may not accurately predict real performance. The digital subcomponents are more straightforward. The discrete logic and Field Programmable Gate Array (FPGA) firmware can be more accurately virtualized in a digital environment. The Operational Flight Program (OFP) can be rehosted on a virtual processor. Salient challenges in firmware and OFP implementation include synchronizing multiple clock domains, replicating propagation delays and accurately virtualizing embedded processors, that need to be addressed however.

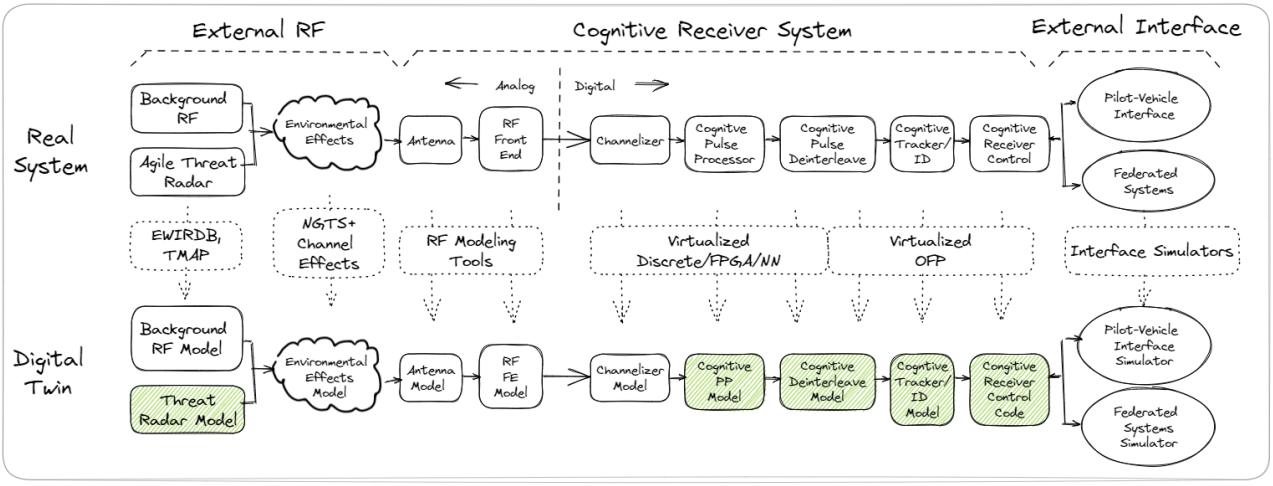

Figure 4 - Cognitive EW Receiver Digital Twin

The External RF element, though not part of the Cognitive receiver system, is critical to implementation of the DT. An accurate replication of the RF environment must be generated to feed the receiver model with realistic inputs. Threat Radar models and Background RF models create realistic waveforms that are modified by an RF Environmental effects model that introduces the doppler, gain, delay and other effects (multi-path and other topographical effects) the waveforms will be subjected to when propagating from RF source to the receiver. The Threat Radar models typically create Waveform Descriptor Words (WDW) or Pulse Descriptor Words (PDW). Higher fidelity models may output Digital I/Q waveforms. There are a range of techniques for creating RF environmental effects from high fidelity Complex Electromagnetics (CEM) to engineering models using simpler RF propagation formulas. The fidelity of the Threat models and RF Environmental models should be matched in the DT. For the notional DT, we are assuming Digital I/Q for the Threat and Background RF models and RF propagation formulas for environmental effects.

The External Interface shown connected to the Receiver Control components represents the receiver connection to the platform Pilot Vehicle Interface (PVI), which consists of the display and control used to operate the system. The Federated Systems are the avionics and other EW systems the receiver may be connected to. An ideal DT requires the modeling of these devices, with accurate interfaces connecting them to the DT Cognitive receiver control block. Note that for simplicity, we have omitted these interfaces from the discussion that follows.

Cognitive Receiver Digital Twin Use Cases

The DT is a valuable tool for complex systems development, training, test and sustainment. Following is a partial list of potential DT use cases in the Development and Sustainment life cycle of the Cognitive Receiver:

- System Development
    - Early algorithm development, 1st order AI/ML training
    - Verifying initial hardware design, unit test
    - Refined AI/ML Training/Testing
    - System/Integrated Test
- Formal testing - Developmental Test/Operational Test
- Sustainment
    - AI/ML training, firmware/software updates
    - Regression testing

The following discussion will focus on uses of the DT for AI/ML training/testing for development and sustainment functions for the Cognitive Receiver and will discuss the Hybrid DT concept in detail for a Cognitive Receiver.

DT applied to AI/ML Training and Testing

As discussed earlier, a DT is a valuable tool for addressing AI/ML training and the unique challenges associated with AI/ML testing. Specific DT benefits for the Cognitive Receiver use case are:

- It is very difficult to create the needed complex RF training and test environment efficiently either in the lab or on the Open-Air Range. The DT has the potential for doing this for the RF Threat, RF Background, and Systems interfaces needed.
- The DT provides test scalability to traverse the training and testing coverage space more quickly for regression training/testing.
    - FTRT training and testing is likely needed, which may be possible in a DT.
    - The DT can virtualize multiple instantiations of AI/ML algorithms to provide accelerated training.
- The use of a DT enables early Modeling and Simulation (M&S) for the design cycle, which is critical for AI/ML systems.
    - The DT supports AI/ML Algorithm development/design/training.

However, there are practical limits to the realism that can be achieved simulating complex systems in complex environments. Specifically, the Cognitive receiver RF and analog components may be difficult to model accurately. A Hybrid DT combining real and simulated components may be able to address this limitation.

Hybrid DT Architecture

The basic Cognitive Receiver hardware and its DT in the Hybrid DT architecture is redrawn in Figure 5. The Hybrid DT architecture supports RF stimulus (a primary component of the training/testing dataset) from one of three sources: Recorded RF, HITL RF and Digital RF. The selected RF stimulus feeds an Environmental effects generator to provide realistic

RF that changes throughout a dynamic scenario. The Environmental Effects block can feed either real Cognitive receiver hardware in a HITL setup or a DT implementation of the Cognitive receiver. For simplicity, the receiver processing chain blocks have been broken into Pre-Processing, AI/ML Component, and Post-Processing blocks. The AI/ML Component could be any of the AI/ML component blocks included in the Notional Cognitive EW receiver shown in Figure 3.

In summary, the Hybrid DT setup provides stimulus from recorded, real, and digital RF sources, feeding real or DT hardware. This flexibility is useful for conducting both AI/ML training and testing.

We'll discuss three major aspects of the Hybrid DT architecture – AI/ML training and testing dataset generation, AI/ML component isolation testing, and a specific process for using the Hybrid DT testbed during training and testing.

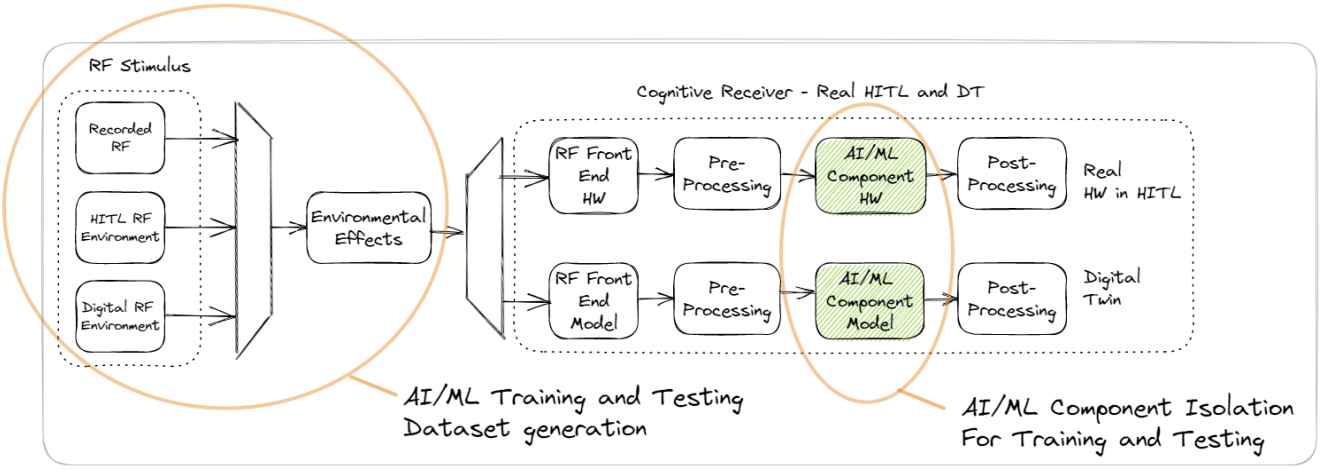

Figure 5 – Hybrid DT Architecture

AI/ML System Training and Testing Dataset Generation

A Cognitive receiver training/testing dataset is required to train and test the system. It consists of either RF or digitized I/Q data generated by the Digital RF Environment and Environmental Effects blocks shown in Figure 6. Data elements are tagged so they can be correlated with RF emitter activity, RF band, and environmental effects used to create it.

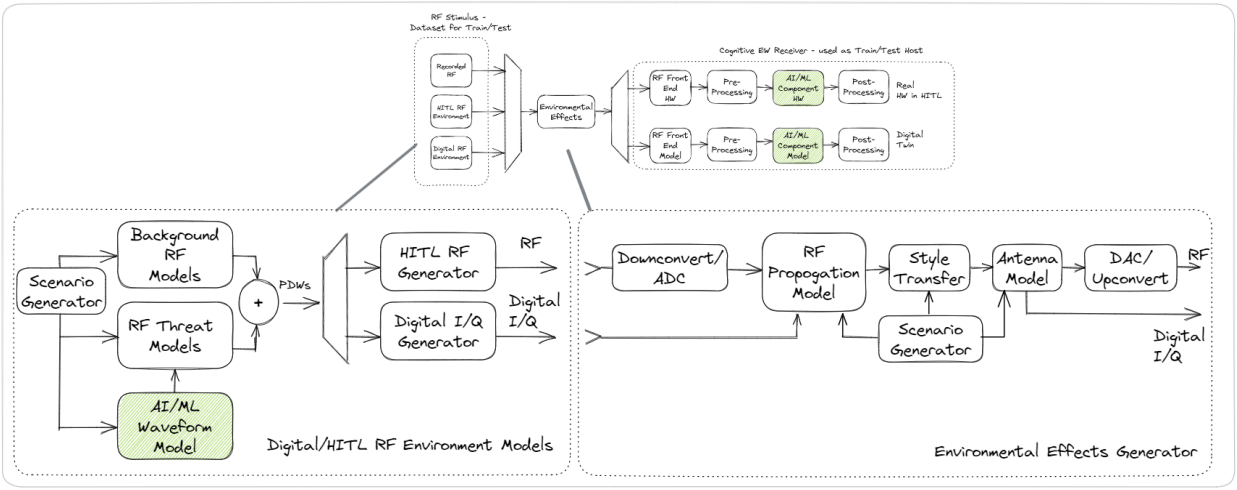

Figure 6 – Generating Training/Testing Dataset

The Scenario Generator (SG) simulates the path of the Cognitive receiver platform through an operational RF threat environment. It is scripted with the expected RF threat laydown, background RF sources, PVI and Federated system interaction and simulated flight path for the Cognitive receiver platform. It executes the script, generating messages that feed the Background RF model, RF threat model, Environmental Effects model and External interface models (not shown in diagram), allowing them to generate synchronized, scenario-representative interactions with the receiver. Note that the scenario script, SG and RF and Environmental Effects generator all work together to create the training and test data spaces for the receiver. The test designer needs a complete understanding of how these elements work together to create these spaces. Given the complexity of the problem a test coverage tool should be created to assess how well scenario scripts are covering the overall space.

Based on SG inputs, the background RF model and RF threat models continuously generate PDWs that define discrete RF pulses. Note that the RF threat model is also driven by an AI/ML waveform model that generates “Novel” waveforms outside of the parameter space of known RF threats. This model is intended to enable generalized classification training for relevant AI/ML receiver components.

The PDWs feed a digital I/Q generator that streams wide-band digital I/Q, providing realistic threat data for the receiver system. The I/Q data feeds an RF Propagation model that adds doppler, gain, delay, clutter and multipath effects that would be induced on the RF as the scenario executes.

The I/Q data is then fed through a Style Transfer block where additional effects can be applied. These effects might be additional RF threat or environmental effects that are added for realism.

Note that the receiver antenna model block is incorporated into the Environmental effects block, assuming that the HITL version of the receiver will not incorporate an antenna.

The system training/testing dataset, generated via the Digital I/Q generator and Environmental Effects generator, needs to be validated prior to usage for training and test. Validation should consist of comparison of individual model performance with real data and validation of end-to-end performance versus real data. Note that validation in this context does not refer to the formal, rigorous model validation required for operational test. Several end-to-end validation methods are briefly considered below. The most straightforward end-

to-end validation is done by generating an equivalent dataset using the HITL RF generator to generate real RF, then comparing the HITL and digital I/Q generated datasets. Recorded RF data could also be injected into the Environmental effects generator and compared to digital I/Q data as a further validation step. In this case the digital I/Q data would need to be driven with a scenario script matching the real scenario used when recording the RF data.

Recall that efficient data generation is a significant issue for training and test of AI/ML systems. This data generation method should mitigate the issue to some degree. It's important to note that real data is still required to verify the synthetic data.

AI/ML Component Isolation

The Hybrid DT is designed to provide individual AI/ML isolation, which allows for direct injection of inputs and direct access to outputs of a given component. This feature is required for training components in a complex system. Direct injection of component inputs allows for efficient input data generation and a higher degree of control for taking the AI/ML component input data through the complete coverage space. Direct access to outputs provides greater transparency when doing early training and testing – the tester can directly observe whether a component is performing as expected.

Component isolation can be done readily with a digital environment, but is more of a challenge with real hardware in a HITL environment. Isolation is enabled by ensuring that algorithms implemented in software and hardware conform to interface standards that are transparent to an ecosystem of potential algorithm developers. This minimizes the level of effort required to insert these algorithms into program of record (POR) systems.

Note that component isolation is very similar to AI/ML component training/testing in a standalone environment. The difference is that initial standalone development of AI/ML components provides a first order approximation of real inputs, meaning that the AI/ML algorithm at that point will not be adequate for usage in a real system. The next step for training should be done using AI/ML component isolation.

Hybrid DT Training and Testing process

To illustrate how this setup for AI/ML component training/testing can be used, the process is broken down into major process steps in Figure 7.

The first process step is generating the AI/ML system training and testing dataset, which was discussed in detail above. The next step consists of a loop of train-test cycles performed for each isolated AI/ML component. Each loop is comprised of the following major processes –

A. Generate isolated component training/testing Dataset
B. Train and test Isolated component on DT
C. Train and test Isolated component on HITL
D. Test, train and test component on full system DT and HITL

Once all components are trained and tested, the entire system is regression tested using the DT and HITL HW. Each of the major process steps is discussed below.

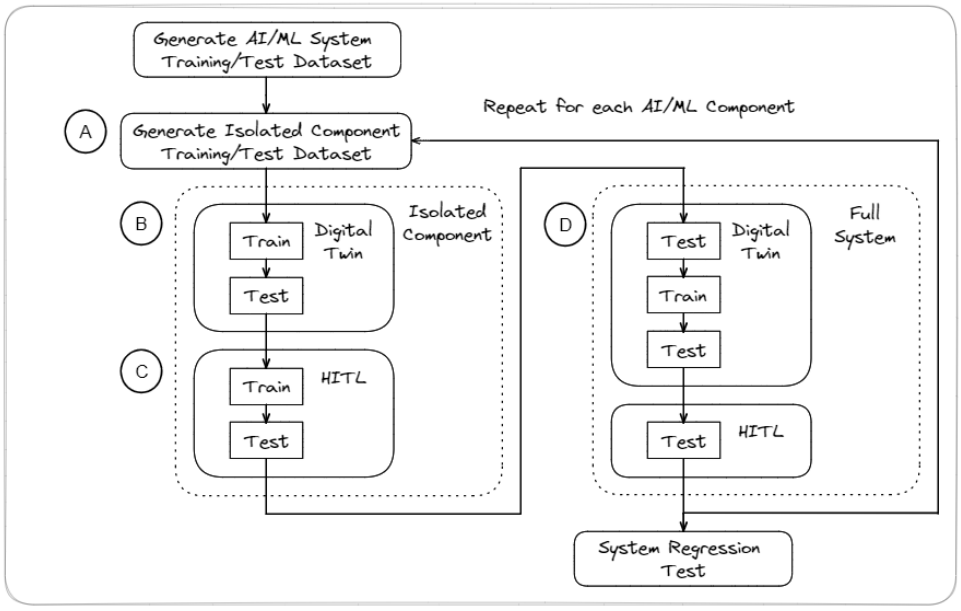

Figure 7 - AI/ML Training/Testing Process

Generate Isolated Component Training/Testing dataset

A dataset that is injected into an isolated component can be generated by stimulating the system with the system training and test dataset and recording inputs from components that feed the isolated component, as shown in Figure 8. These inputs are the direct injection data that will be used to train and test that component. The direct injection inputs are correlated with RF stimulus and Environmental effects that produced the data, along with event tags. The component designer will need to associate injected inputs with expected outputs for the AI/ML component model. These associations will form the truth data used for training and testing.

It would be beneficial to perform a coverage space analysis to determine how much of the AI/ML component model input space is actually covered by the generated dataset. If large parts of the direct inject dataset are uncovered, the designer may need to determine if there are issues with the SG scripting for the RF Stimulus or Environmental Effects blocks, or issues with the way these blocks are functioning.

There are challenges associated with this approach. Note in Figure 8 that Digital I/Q data is directly fed into the digital pre-processing model of the Cognitive Receiver, bypassing the RF front end model. This is done to simplify the creation of the DT. It may not be feasible to create a high-fidelity RF front end model. Additionally, feeding digitized RF into the front end may not be feasible either. The penalty paid by bypassing the RF front end model, is that the digital I/Q will have the RF front end effects absent, which could affect performance of downstream AI/ML processing in the real system. This may be mitigated by introducing RF front end effects in the Style Transfer Block (refer to Figure 6) during system training and testing dataset generation.

Care needs to be taken with the order of isolated AI/ML components for which direct inject data is generated. AI/ML component blocks that precede the chosen component must be trained before a given downstream component is addressed. If multiple components feed a given component, or there is component data feedback, then this process could become difficult and iterative training with multiple components may need to be done. This process

works well for the Cognitive receiver example due to its straightforward signal processing pipeline. It may not work as well for more complex systems; indeed, the feasibility and potential success of this approach heavily depend on the specific system architecture.

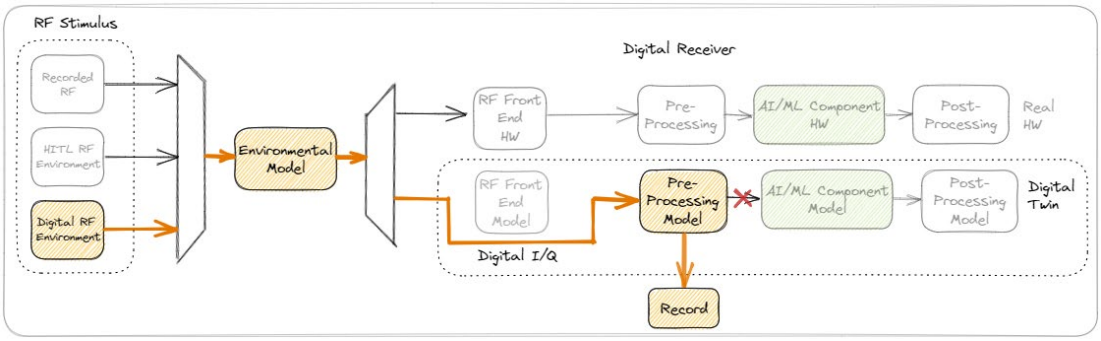

Figure 8 - Generating Isolated Component Dataset

Train and test Isolated component on DT

In this step, the Direct Inject dataset generated for the specific component is played back into the AI/ML component in the DT to train the component as shown in Figure 9. Conducting the training may involve very large data sets and require multiple training cycles. Doing the training on the DT using the recorded dataset allows the training to potentially be conducted faster than real time. This is useful in a system like a Cognitive Receiver that likely requires continuous training in sustainment to adapt to a changing RF and threat environment.

Note that it is crucial that the digital training data used in this step be as realistic as possible, reflecting real threat, environment and system front end effects. If not, the AI/ML algorithm probably won't handle these effects properly in an operational environment. It was noted above that the RF front end would likely need to be bypassed with Digital I/Q data generation for the DT. This may be mitigated through the use of Style Transfer in the Environmental Effects generator, but it is anticipated that this will present its own set of challenges. Generally, training is conducted in training-validation-test cycles. This architecture should support these functions.

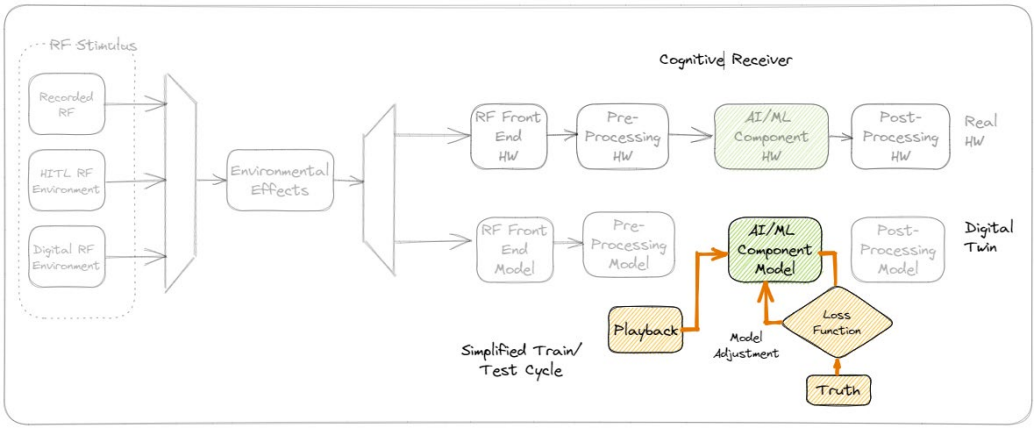

Figure 9 – Training and Testing Isolated Component on DT

Train and test Isolated component on HITL

In this step, training and testing is conducted on isolated AI/ML components on the HITL (refer to Figure 10). The HITL will provide a higher level of realism; it uses real RF sources and incorporates the RF front end signal path. The downside is that it must be run in real time, limiting extensive training cycles. It may be feasible to do fine tuning of AI/ML algorithms if the real time limitation does not create unacceptably long training cycles. It should be feasible, however, to use the HITL path for regression testing, which could be critical for verifying AI/ML performance after training on the DT.

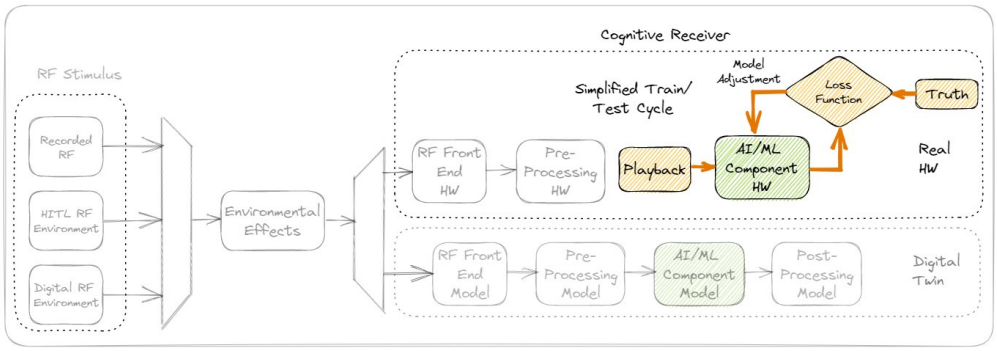

Figure 10 – Training and Testing Isolated Component on HITL

Test, Train and test component on Full System DT and HITL

After each component is individually trained and tested, it must be evaluated in the context of the overall system (refer to Figure 11). There may be interactions and dependencies that impact performance of the component that would only be seen in the full system environment.

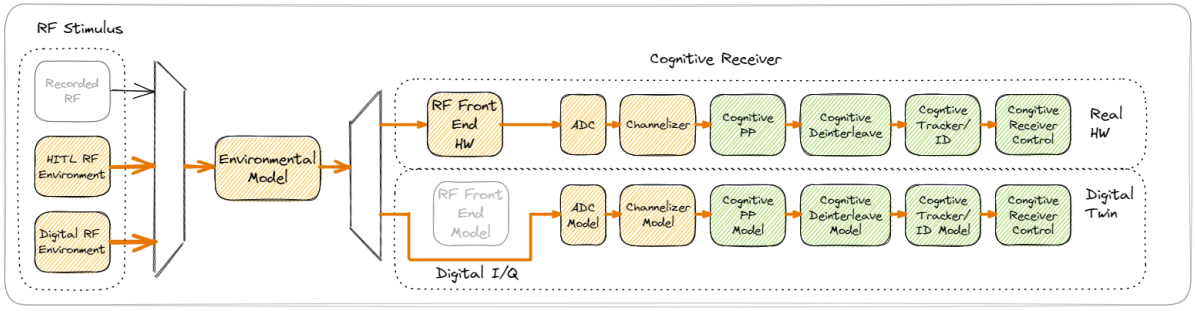

Figure 11 – Training and Testing Full System on HITL and DT

The recommended approach is to initially test the component on the full system using the DT to determine whether the component has an acceptable level of performance. The intent is early identification of component performance issues and root cause analysis. If causes are related to training data variances, then it may be necessary to adjust the isolated training dataset and re-train and re-test the component in the isolated environment. If the component performs acceptably, then a train–test cycle is initiated to further refine training.

Next the component is tested on the HITL setup. If performance is acceptable, the full process is repeated for the next AI/ML component. Root cause analysis is conducted if

the component fails, which may result in adjustment of training data and re-training and re-testing in the isolated component mode.

Depending on system complexity, there may be confounding interdependencies among the AI/ML components that prevent complete training and testing of a given component. For example, it may not be feasible to completely train/test component A, then completely train/test component B, etc., given component interdependencies. An iterative capability approach will likely be required: train/test component A with initial capability, train/test component B with initial capability, etc., iterating through the process repeated times, layering on additional capabilities for components in the chain.

A full system regression test will be run on the DT and HITL once component training and testing is completed.

Hybrid DT Model Validation

Initial Hybrid DT validation is required as soon as real system operational data can be collected. Prior to or during Developmental Test and Evaluation (DT&E) and Operational Test and Evaluation (OT&E), it may be feasible to collect data using an Installed System Test Facility (ISTF) like an anechoic chamber. During DT&E and OT&E Open Air Range (OAR) testing, real operational data will also be available for validation. Validation will also continue over the life of the system, if data can be collected during operational usage.

The type of data collected during these events will dictate how it is used for Hybrid DT validation. Ideally, a data recorder would collect RF data at the receiver faceplate in an operational environment with ground and airborne threat simulators. Truth data such as aircraft Time, Space, Position Information (TSPI), threat state data and range RF instrumentation for other emitters would also be required. The RF data would be played back as indicated in Figure 12. Using collected truth data, the Hybrid DT performance can be verified against actual Cognitive receiver performance.

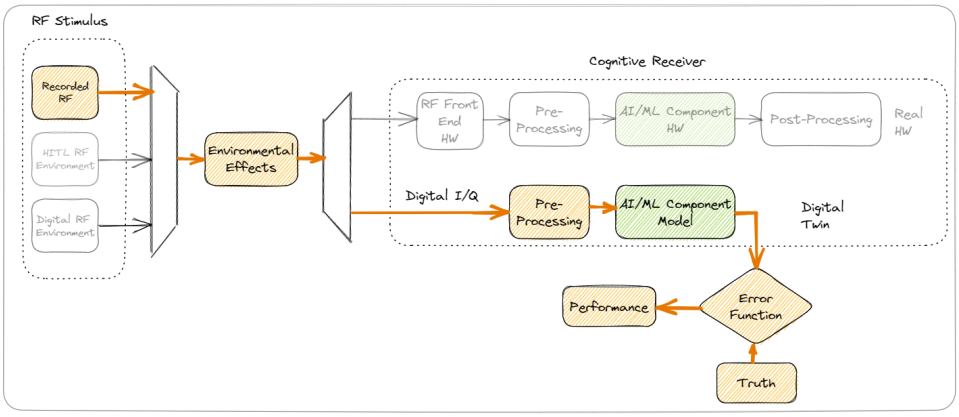

Figure 12 – Hybrid DT Model Validation - Component

It would also be useful to record and playback internal Cognitive receiver instrumentation data such as digital I/Q, PDW buffers, or emitter track file buffers to verify individual receiver component performance.

During Hybrid DT validation, it is likely that there will be some variance between the real system performance and the Hybrid DT performance. This should be viewed as an opportunity to further refine the Hybrid DT by verifying recorded playback, RF Stimulus

block, Environmental effects block, RF front end assumptions, and correct virtualization of the AI/ML components and OFP, or other potential sources of variance. As sources of variance are found and fixed, further confidence will be established for the Hybrid DT.

Considerations

The general advantages of using DTs is clearly understood and was stated above. For this discussion we'll consider the advantages of a Hybrid DT versus a purely digital DT. The primary advantage of the Hybrid DT architecture is improved realism. Systems with complex Cyber-Physical interaction and heavy sensor-dependencies are difficult to implement with purely digital DT. Implementing high-fidelity digital models for sensors and complex RF and analog signal processing can be a significant challenge. The use of hardware in the Hybrid DT to replicate these behaviors, serving as an adjunct to the DT can improve realism for those elements.

There is a basic trade-off of simulation realism versus simulation time that can be balanced with the Hybrid DT. The hardware components have increased realism but also have increased simulation time (they cannot be run faster than real time). The digital components will have less realism but can potentially be run faster than real time. The Hybrid DT uses hardware implementation only for the components that can't be simulated on a digital environment with sufficient realism. Fortunately, in the case of the Cognitive receiver, the AI/ML and software components can be virtualized with reasonably high fidelity because they are already in the digital domain.

However, the Hybrid DT will require significant effort to develop and maintain. Some in the EW T&E community argue that resources should be dedicated to improve operational testing of systems instead of DT and Digital M&S. Certainly, digital M&S has been overhyped in the past, leading to false perceptions of feasibility, accuracy, and utility. Several of the anticipated challenges implementing the Hybrid DT are as follows:

- There is no one-size-fits-all solution. The specific implementation and training/test process will vary depending on the system. The Cognitive receiver example is essentially an open loop system. More complex systems such as RF jammers will present additional difficulties.
- It is essential to verify the Hybrid DT with real OAR data collected during DT&E/OT&E and to continue validation over the system life cycle.
- For Cognitive EW applications, a critical part of the DT is the RF and threat environment. Great care needs to be taken to ensure that this environment is accurately replicated. Other AI/ML applications such as autonomous driving have similar challenges simulating realistic environments.

Conclusion

The Hybrid DT approach demonstrated above is a promising approach for providing the improved training and test capabilities required for complex AI/ML systems.

Through targeted usage of real hardware, coupled with digital simulations, the Hybrid DT should be able to simulate the system and its operational environment with sufficient realism. If the RF operational environmental simulation is built with scenario generation capability and environmental effects simulators, much of the required training and test data may be able to be generated. Finally, the Continuous Integration/Continuous Deployment (CI/CD) process required for AI/ML systems can be supported if the system is constructed with a development pipeline that supports efficient virtualization of AI/ML components and firmware/software components.

Program test managers should carefully consider Digital Twinning and Digital model approaches, and adapt test constructs that are best suited for their system, considering system Cyber–Physical interactions and system complexities. This is particularly true for AI/ML based systems, like Cognitive EW systems. Test constructs should also be chosen in the context of the complete system life cycle, including design, implementation, DT&E/OT&E and sustainment.

References

Anthony Ortiz, O. F. (2018). On the defense against adversarial examples beyond the visible spectrum. *Milcom 2018 Track 5 - Big Data and Machine Learning*, (pp. 553–558). Retrieved from https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8599763

Antonio Nieto-Rodriguez, R. V. (2023). How AI will transform project management. *Harvard Business Review*.

Chuanqi Tao, J. G. (2019, August 23). Testing and quality validation for AI software–Perspectives, issues, and practices. *IEEEAccess*, 120164–120175. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8811507

Collins, C. (2023). *Test and evaluation as a continuum.* OUSD (R&E).

Digital Twin Consortium. (2020, December 3). *Definition of a digital twin*. https://www.digitaltwinconsortium.org/initiatives/the-definition-of-a-digital-twin/

DoD. (2018). *National defense strategy.* https://nssarchive.us/wp-content/uploads/2020/04/2018_NDS.pdf

DOT&E. (2022). *Digital twin assessment, Agile verification processes, and visualization technology.*

DSB. (2016). *Summer study on autonomy.*

GAO. (2022a). *How artificial intelligence is transforming national security.* https://www.gao.gov/blog/how-artificial-intelligence-transforming-national-security

GAO. (2022b). *AI - Status of developing and acquiring capabilities for weapons systems.* https://www.gao.gov/products/gao-22-104765

Jörn Thieling, J. R. (2021). Scalable sensor models and simulation methods for seamless transitions within system development: From first digital prototype to final real system. *IEEE SYSTEMS JOURNAL*, 3273–3282.

Kirill Semenkov, V. P. (2020). Verification of large scale control systems with hybrid digital models and digital twins. *2020 International Russian Automation Conference (RusAutoCon)*, 325–329.

NASEM. (2021). *Necessary DoD range capabilities to ensure operational superiority of U.S. defense systems: Testing for the future fight.* The National Academies Press.

NIST. (2021). *Considerations for digital twin technology and emerging standards.* https://nvlpubs.nist.gov/nistpubs/ir/2021/NIST.IR.8356-draft.pdf

Prokhorov, D. (2019). Toward next generation of autonomous systems with AI. *IJCNN 2019. International Joint Conference on Neural Networks*, 1–5. Budapest, Hungary. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8851867

RAND. (2021). *The Department of Defense's posture for artificial intelligence.* https://www.rand.org/pubs/research_briefs/RB10145.html

Sandra Hobson, P. L. (2022). DOT&E strategy update 2022: Transforming T&E to enable delivery of the world's most advanced warfighting capabilities at the speed of need. *Aircraft Survivability Program - JASP Online*.

Tate, D. (2019). *What counts as progress in the T&E of autonomy.* Institute For Defense Analyses.

Tian, Y. (2018). DeepTest: Automated testing of deep-neural-network-driven autonomous cars. *2018 ACM/IEEE 40th International Conference on Software Engineering*, 303–314. https://doi.org/10.1145/3180155.3180220

USAF Chief of Staff. (2020). *Accelerate change or lose.* https://www.af.mil/Portals/1/documents/csaf/CSAF_22/CSAF_22_Strategic_Approach_Accelerate_Change_or_Lose_31_Aug_2020.pdf

Acquisition Research Program
Department of Defense Management
Naval Postgraduate School
555 Dyer Road, Ingersoll Hall
Monterey, CA 93943

www.acquisitionresearch.net