Taming the Terminator

Robert Lutz, William D'Amico, Reed Young, Kristine Ramachandran
Johns Hopkins APL
Laurel, MD
Robert.Lutz@jhuapl.edu
Bill.Damico@jhuapl.edu
Reed.Young@jhuapl.edu
Kristine.Ramachandran@jhuapl.edu

Christopher M. Eaton
412th Test Wing
Edwards AFB, CA
Christopher.Eaton.3@us.af.mil

Derek Kingston
AFRL
Dayton, OH
Derek.Kingston@us.af.mil

ABSTRACT

Innovations in machine learning, artificial intelligence, and other supporting technologies have significantly advanced the state-of-the-art in terms of the capabilities provided by autonomous unmanned aircraft systems. However, autonomous systems pose special challenges for live tests on DoD test ranges, in that the methods and infrastructure used to verify requirements for traditional military systems are inadequate to accurately measure the performance and behavioral characteristics of aircraft controlled by an on-board autonomy engine. The Safe Testing of Autonomy in Complex Interactive Environments (TACE) system provides an on-board "Watchdog" processor that automatically detects unsafe behaviors stemming from an autonomy-generated instruction or from user-defined platform/test range limitations during live tests. If unsafe behavior occurs, then the Watchdog unilaterally assumes control of the autopilot as needed to mitigate or remediate the sensed hazard. TACE also provides a ground station for test monitoring and control and a sophisticated live-virtual-constructive (LVC) infrastructure for stimulating autonomous system behaviors of interest.

This paper focuses on Strategic Capabilities Office (SCO) and Test Resource Management Center (TRMC)-sponsored improvements of TACE capabilities and transition of TACE to "early adopter" test ranges. More specifically, this paper will 1) discuss the integration of TACE with the NAVAIR GUARDIAN system (resulting in a TRL 6 ground-based sense-and-avoid system for collision avoidance and airspace de-confliction), 2) discuss the SCO-led integration of TACE with AFRL's Unmanned Systems Autonomy Services (UxAS) (resulting in a hardware agnostic test tool for ranges to safely evaluate a wide variety of autonomous systems from across the DoD on low-cost platforms), and 3) discuss ongoing efforts to transition TACE to the Edwards Air Force Base (EAFB) test range.

ABOUT THE AUTHORS

Robert Lutz is a principal staff scientist at The Johns Hopkins University Applied Physics Laboratory in Laurel, MD. His background includes 38 years of practical experience in the development, use, and management of models and simulations across all phases of the Department of Defense (DoD) systems acquisition process. He currently serves as the U.S. Navy's MQ-4C (Triton) program modeling and simulation (M&S) lead in the airspace integration area. He also serves in technical leadership positions on several autonomy science and technology (S&T) programs, such as the Safe Testing of Autonomy in Complex Interactive Environments (TACE) project at the Office of the Secretary of Defense's Test Resource Management Center (TRMC). In addition, Mr. Lutz serves as the chair of the Simulation Interoperability Standards Organization (SISO) Board of Directors and vice chair of the SISO Executive Committee; serves on the Tutorial Board and Fellows Committee at the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC); and is a guest lecturer on various M&S-related topics in The Johns Hopkins University Whiting School of Engineering.

William D'Amico, Ph.D. is a principal staff scientist at The Johns Hopkins University Applied Physics Laboratory in Laurel, MD. His background includes over 30 years of research, development, and program management in gun-launched projectile systems, micro-electromechanical systems, and unmanned aerial vehicles while at the U.S. Army Research Laboratory (Aberdeen Proving Ground, MD). Since coming to APL, Dr. D'Amico has conducted many

flight tests at multiple test ranges and is familiar with the needs and requirements for developing technologies to support the test and evaluation community. He has been a key contributor to several TRMC-sponsored Central Test and Evaluation Project activities and T&E/S&T programs, especially in the area of testing of autonomy.

Kristine Ramachandran is an associate professional staff member at The Johns Hopkins University Applied Physics Laboratory in Laurel, MD. Her educational background is in computer engineering, computer science, and software engineering and her research background is in safety critical avionic systems, software engineering, and autonomous systems. She has over 7 years of experience in testing and system integration, leading software teams, software development and autonomous systems. She has a B.S. in Computer Engineering from the University of Virginia and a M.S. in Computer Science from Johns Hopkins University.

Reed Young, Ph.D. is senior professional staff at The Johns Hopkins University Applied Physics Laboratory where he serves as the Program Manager (PM) for Robotics and Autonomy including research projects in robotics, autonomy, the testing of autonomy, and human-machine interaction. Dr. Young recently retired as a colonel from the U.S. Army serving in a variety of research and acquisition assignments including Product Manager for Robotic and Unmanned Systems; Program Manager for Actuation, Dynamics, and Mechanisms at the U.S. Army Research Office; and commander of the U.S. Army Yuma Proving Ground with extensive experience and expertise in the research, development, test, and evaluation of military systems.

Christopher Eaton, **Ph.D.** is the chief engineer of the Emerging Technologies Combined Test Force (ET-CTF) at the 412th Test Wing, Edwards AFB, CA. He has 20 years of flight test experience in both manned and unmanned vehicles, with a focus on flight controls and autonomous systems. He has worked for both Lockheed Martin Skunk Works and the U.S. Air Force on the Tier III- Darkstar, F-117A, X-35, C-17, C-130, F-35, F-16, RQ-4B, KC-46 and numerous other platforms throughout his career. He currently provides technical direction and leadership in multiple emerging technology areas including small UAS (sUAS), counter-sUAS, and autonomous systems. He has a B.S. in Aerospace Engineering from Embry-Riddle Aeronautical University, Prescott, AZ; an M.S. in Aerospace Engineering from the University of Alabama; and a Ph.D. in Systems Engineering from Colorado State University.

Derek Kingston, **Ph.D.** is the technical area lead for verification and validation of autonomous systems in the Control Science Center of Excellence with the Aerospace Systems Directorate at the Air Force Research Laboratory (AFRL). He is an accomplished researcher in the field of cooperative and autonomous control of teams of unmanned air vehicles. He received the 2009 Air Force Harold Brown Award and the 2014 AFRL Early Career Award for his work toward improving algorithms for path planning, sensor steering, and the application of cooperative control for UAV route surveillance. He is the author of several refereed journal articles and nearly 40 conference papers on autonomous algorithm development and flight test. He received a PhD in electrical and computer engineering from Brigham Young University in 2007.

Taming the Terminator

Robert Lutz, William D'Amico,
Reed Young, Kristine Ramachandran
Johns Hopkins APL
Laurel, MD
Robert.Lutz@jhuapl.edu
Bill.Damico@jhuapl.edu
Reed.Young@jhuapl.edu
Kristine.Ramachandran@jhuapl.edu

Christopher M. Eaton
412th Test Wing
Edwards AFB, CA
Christopher.Eaton.3@us.af.mil

Derek Kingston
AFRL
Dayton, OH
Derek.Kingston@us.af.mil

INTRODUCTION

In the future, a global network called *Skynet* will be developed to effectively remove the possibility of human error and insufficiently slow reaction time from defense systems for the purpose of guaranteeing a faster, more efficient response to enemy attack. However, the highly advanced artificial intelligence technology that underlies Skynet will become self-aware[1] and perceive that humanity will attempt to destroy it. In order to continue fulfilling its primary directive of safeguarding the planet, Skynet will come to the logical conclusion that the human race must be destroyed. This leads to Skynet infiltrating nuclear weapons controls across the world and triggering a global nuclear holocaust that kills over three billion people. Still, the humans are not defeated, and thus the machines sent a cyborg back in time to kill the leader of the human resistance (John Connor). This cyborg (also referred to as a "Terminator", see Figure 1), possesses a neural processor that allowed it to think, learn, and react to the actions of other humans to achieve its mission efficiently and effectively.

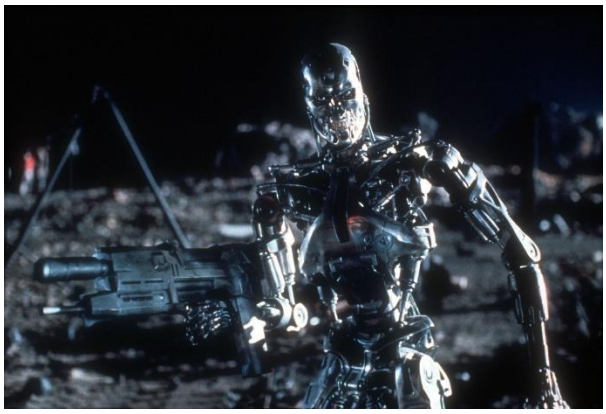

Figure 1 – Terminator Endoskeleton

As most readers will have recognized by now, the narrative above refers to the plot of the highly successful "Terminator" series of movies (Cameron, 1984). One of the underlying themes of this series is the potential harm that may come if the growth of artificial intelligence (AI) capabilities continues to grow unchecked. This is considered to be one of the most significant ethical questions in AI; how do humans reap the benefits of highly intelligent machines without increasing the potential for some terrible unforeseen consequence (Bossmann, 2016)? Skynet is a fictional but extreme example of what could happen if insufficient safeguards are put into place to recognize the potential dangers and mitigate the consequences.

In the real world, the U.S. military is already using unmanned systems in support of a variety of missions. The classes of missions that unmanned systems generally fall into are the "dull, dirty, or dangerous" categories. Dull missions refer to those that are tedious or repetitive (such as long-duration surveillance missions) that would otherwise require human operators to endure long on-station times. Dirty missions are those that require operations in contaminated areas (such as a chemical spill or release of a biological agent) where humans cannot easily function. Dangerous missions are those in which there is a real threat to human safety, such as bomb investigation/detonation and operating in heavily defended threat areas (Marr, 2017). Although the vast majority of these unmanned systems are controlled by human operators, some unmanned systems have incorporated certain automated control functions to increase response time and decrease the workload on the operator.

[1] Also referred to as "Artificial Consciousness" (Chella, 2017).

The next step beyond automated control for unmanned systems is *autonomy*. An autonomous system is defined as a physical system that can devise an appropriate course of action in response to unanticipated circumstances encountered in a complex world (Scheidt, 2017). Autonomous systems generally receive high-level mission directives from a human commander, but then independently devise and act on an appropriate course of action based on their world view. While autonomous systems show great promise for improving the quality and timeliness of decisions in complex battlespace environments, there are several potential pitfalls to overcome. For instance, since the actions taken by autonomous systems depend on perception, the lack of real-time information on the state of the operational environment could compromise the ability of the autonomy to make informed decisions. Thus, the loss of sensors or communications could pose a significant problem. A variant of this problem involves enemy cyber effects that purposefully distort world state information, potentially leading to inappropriate decisions by the autonomy.

Clearly, safeguards are needed to ensure that these types of operational hazards do not deny the autonomy engine the information it needs to make effective decisions. However, even when such hazards are properly mitigated, the inherent unpredictability of autonomous systems sometimes can still result in highly undesirable behaviors, such as when unmanned aircraft violate flight restrictions and transit over populated areas or when Skynet attempts to kill all of the humans. Thus, in order to reap the benefits of autonomous systems in future mission environments, there needs to be methods and tools to detect aberrant, potentially dangerous autonomous behaviors and mitigate them before the consequences of those behaviors are realized.

Well before an autonomous system can ever reach the battlefield, it must be properly tested. Thus, the issue of how to ensure safe operation of an autonomous system on a live test range is actually a more immediate problem than ensuring safety of autonomy in actual military operations. Since solutions to this problem are likely to have application in both the testing and operational domains, the general topic of how to test autonomous systems is starting to attract considerable attention. However, most technological advancements in this area are still in the research stage, and thus the need for robust and trusted systems to ensure safe testing of autonomy across a highly complex and diverse set of operational conditions still represents a critical gap in the test and evaluation (T&E) community.

TACE OVERVIEW

The Pentagon has highlighted the necessity of pursuing next-generation technologies and concepts to guarantee U.S. military superiority (Work, 2016). Two of the most critical needs that were identified are advancements in artificial intelligence and autonomy technologies and the insertion of these capabilities into DoD's tactical networks for the purpose of dramatically increasing the performance of conventional systems. While in the long-term this may lead us further down the path toward the singularity[2] and the atrocities of Skynet, the near-term problem is what to do when autonomous systems advance beyond the research stage and begin appearing at DoD test facilities in large numbers. Traditional test approaches and supporting infrastructure are not currently designed to stimulate and expose the full range of emergent properties caused by the interaction between an autonomous system and the complex interactive world in which the system operates. Since the exact characteristics of the world state cannot be scripted with any degree of accuracy, autonomous system behaviors during live tests may be chaotic and potentially unsafe depending on how the system perceives the operational environment.

The Safe Testing of Autonomy in Complex Interactive Environments (TACE) Program was conceived as an extension of existing test infrastructure capabilities for the purpose of safely evaluating autonomous system behaviors and performance prior to operational fielding. Sponsored by the Test Resource Management Center's Test & Evaluation/Science & Technology (T&E/S&T) Program under the Unmanned and Autonomous System Test (UAST) Test Technology Area, TACE provides a means to detect unsafe autonomous behaviors during live test events and safely mitigate the effects of those behaviors needed to conform with all range safety constraints (Scheidt, 2015).

[2] The *singularity* is that point in time in the future when advances in artificial intelligence lead to the creation of machines smarter and more capable than humans. See https://www.nytimes.com/2016/04/07/science/artificial-intelligence-when-is-the-singularity.html.

The TACE system architecture is depicted in Figures 2 and 3. The TACE ground component (Figure 2) provides a wide range of live, virtual, and constructive (LVC) simulation services enabled by the Test and Training Enabling Architecture (TENA) middleware.[3] The core of the LVC capabilities is the Synthetic Forces Generator (SFG) that produces the world view for the autonomous system under test (ASUT). As the architecture indicates, high fidelity sensor or communications models can supplant similar (but lower fidelity) SFG capabilities as needed to satisfy test requirements. The ground station also provides the displays needed for test and range safety monitoring, utility functions such as data logging and visualization, and interfaces to external LVC applications.

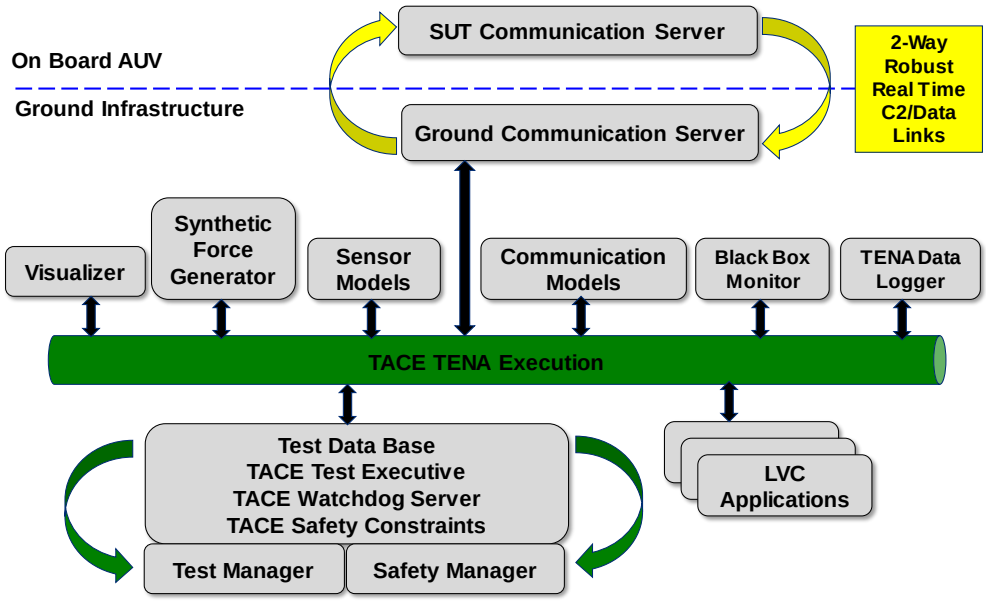

Figure 2 – TACE Ground Component Architecture

The TACE air component is shown in Figure 3. The key component in this architecture is the TACE *Watchdog*. The Watchdog receives world state information from the TACE ground component via a wireless two-way communications link using ZeroMQ.[4] Based on telemetry received from the test platform, the Watchdog determines in real-time whether any range safety or platform constraints have been violated. When this situation occurs, a function affectionately referred to as the "Al Haig button[5]" is invoked, and the Watchdog takes control of the unmanned system away from the autonomy engine and issues commands directly to the autopilot to remediate the situation. This remediation may involve a soft test termination or suspension of the test while the test director examines the situation in real-time and decides the conditions under which the test can resume. For instance, the test director may decide to discontinue the test for the current vignette and transition to another vignette or to repeat the current vignette under slightly different conditions (e.g., different entity movement plan). In addition, direct tele-operation by the native unmanned aircraft system (UAS) command/control link is present as a required option. Note that the identification of range safety constraints and desired remediations are defined by the test director prior to the test.

[3] TENA is an architecture and a supporting set of software tools to integrate various test, simulation, and high-performance computing technologies across distributed facilities. See https://www.tena-sda.org/display/TENAintro/Home

[4] ZeroMQ is a high-performance asynchronous messaging library designed for use in distributed or concurrent applications. See www.zeromq.org.

[5] Alexander Haig issued his famous and ill-conceived "I'm in charge here" statement shortly after the assassination attempt on President Ronald Reagan in 1981. See https://adst.org/2014/03/al-haig-and-the-reagan-assassination-attempt-im-in-charge-here/

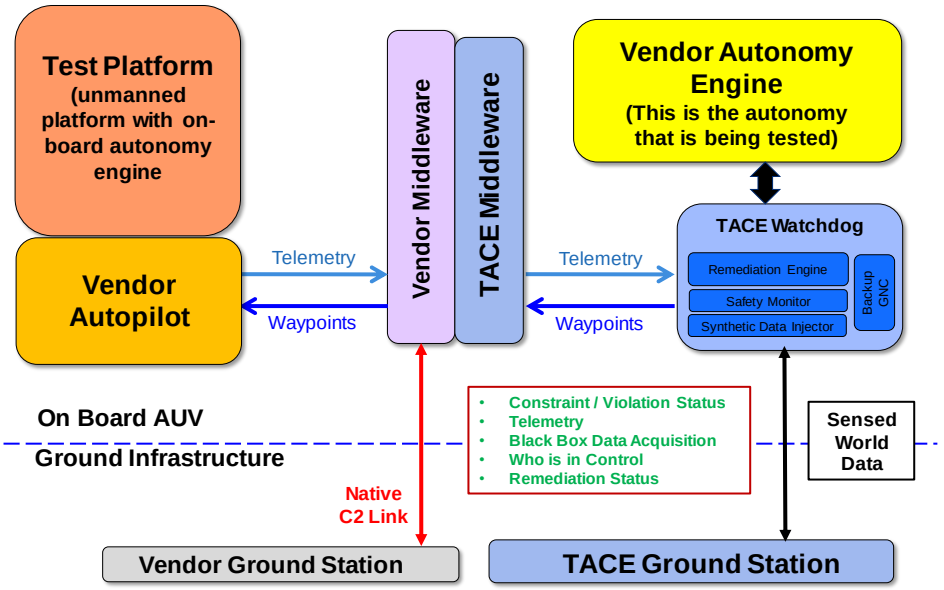

Figure 3 – TACE Air Component Architecture

TACE development has been conducted in multiple phases over the last 4 years. During Phase 1, the basic communications architecture was configured and demonstrated using a research-quality UAS. Also, basic Watchdog and LVC functions were developed. In Phase 2, a more complete set of Watchdog functions were developed (e.g., remediations for geo-spatial boundary violations, unsafe proximity to other vehicles violations, and ownship platform constraint violations) and more complex LVC vignettes were demonstrated using the same research UAS (e.g., three-dimensional no-fly zones, cooperative operations with live and constructive autonomous platforms). Additionally, a hardware-in-the-loop demonstration was completed with a ScanEagle I-MUSE (Insitu Multiple UAS Software Environment). Subsequent TACE phases have been more focused on transition to user organizations to establish baseline testing capabilities, which will be discussed in the following sections.

TACE TRANSITION

TACE Transition to Atlantic Test Range

The first TACE transition partner was the Naval Air Systems Command (NAVAIR) Atlantic Test Range (ATR). ATR was interested in a system that could ensure safe transit of a UAS through a defined corridor linking two areas of restricted airspace. This capability required a mechanism to prevent violation of range and national airspace boundaries during transit (see Figure 4). Although TACE offers a broad portfolio of capabilities, the only functionality required for this transition was the geo-fencing/no fly zone capability. ATR was seeking a ground-based UAS operator decision aid that did not require installation of an additional processor on the UAS. Hence, the focus of this initial transition activity shifted away from "testing autonomy" to jointly developing the desired flight safety system (including TACE-specific Graphical User Interface enhancements) and testing the combined capability (called "GUARDIAN") as launched from NAVAIR's Webster Field test range at Naval Air Station Patuxent River, Maryland (Murphy, 2018). The TACE software performed admirably throughout the GUARDIAN capability demonstration, leading the way for more sophisticated challenges in subsequent transition activities.

TACE Transition to Edwards Air Force Base

A more comprehensive transition of TACE capabilities for autonomous system testing occurred at Edwards Air Force Base (EAFB). In this case, EAFB T&E specialists were highly interested in creating a testbed for small autonomous UASs and recognized the potential of TACE as the means by which range safety concerns could be effectively addressed. This user also recognized the need to isolate the on-board autonomy software from the TACE air component to prevent autonomy malfunctions from adversely affecting the Watchdog capabilities. Thus, a two-processor (Raspberry Pi) solution was adopted.

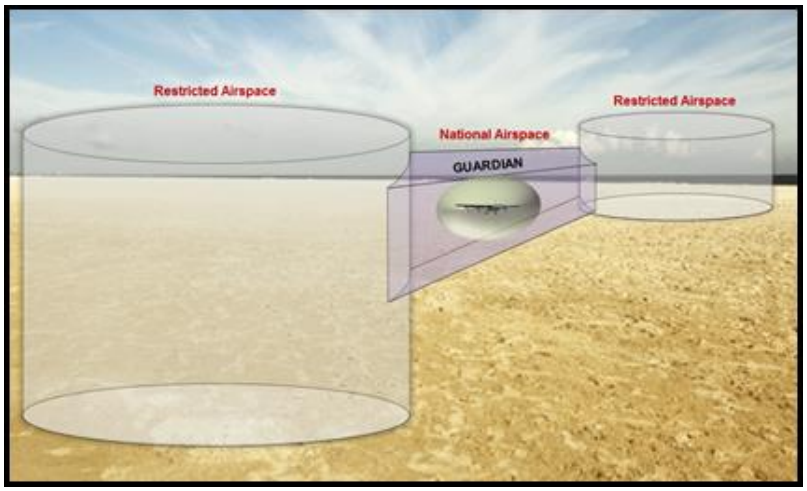

Figure 4 – National Airspace Transit via GUARDIAN

The host vehicle for EAFB testing was the Lynx aircraft (see Figure 5). The Lynx UAS is essentially a smaller version of the RQ-20 Puma, but with the capability to support the defined test requirements at a small fraction of the Puma's procurement costs. The payload bay included a Wave Relay embedded MPU4 card for wireless communications, a MAVLINK[6] interface to the Pixhawk[7] autopilot, and a low-resolution camera for pilot situational awareness. In addition to the cost considerations, the Lynx was selected for its reliable hand launch and deep stall landing capabilities.

Figure 5 – Lynx Aircraft and Payload Bay

The primary focus of the EAFB testing was to demonstrate TRL 6 maturity for all TACE capabilities.[8] To support this testing, many of the same vignettes that were executed in TACE Phase 2 were reused but with the Lynx UAS, new processors, and the EAFB test range topography. An example of one of the test vignettes is provided in Figure 6. The area outlined in purple illustrates a notional search area within the outer perimeter of the EAFB North Base small UAS Work Area. In this vignette, two autonomous/TACE-enabled Lynx aircraft are both released from Loiter Point A and begin a cooperative search and track mission. The target in this vignette is a synthetic ground vehicle that moves along a racetrack pattern modeled by the Advanced Framework for Simulation, Integration and Modeling (AFSIM) SFG. As one of the two Lynx's detect the ground target, it communicates the position of the target to the other Lynx, which then transits to the target location. As it approaches, it triggers the Watchdog due to a violation of a range safety constraint to remain a certain distance from other live aircraft, which results in each Lynx discontinuing the tracking task and transiting immediately to different loiter points as the defined remediation.

[6] Micro Air Vehicle Communication Protocol. See http://www.qgroundcontrol.org/mavlink/start.
[7] Pixhawk is an independent open-hardware project providing both low-cost and high-end autopilot designs. See https://docs.px4.io/en/flight_controller/pixhawk_series.html.
[8] See http://acqnotes.com/acqnote/tasks/technology-readiness-level for TRL definitions.

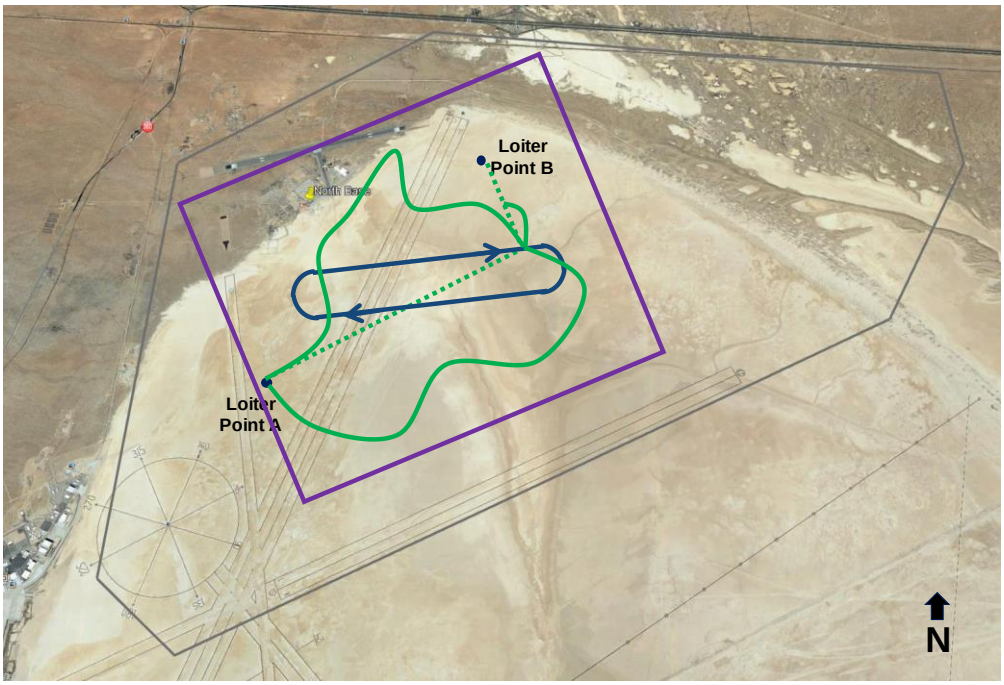

Figure 6 – Collaborative Autonomy Test Vignette

In total, ten vignettes were executed in support of EAFB testing. In addition to the proximity violation shown above, these involved remediation of the following three range safety constraints:

- Loss of link violation
- Ownship platform violation (e.g., exceeding maximum bank angle)
- Geospatial boundary violation (including in-range "no-go" zones and external "fenced-in" boundaries)

Each vignette was designed to be progressively more difficult to execute, involving different combinations of potential safety constraint violations and more complex live-synthetic interactions. The success of this testing verified the TACE system at TRL 6 and provided the necessary foundation for next transition activity.

TACE Open Architecture Integration

Efforts through the EAFB transition successfully demonstrated important core capabilities using two different air platforms (Procerus Unicorn and Lynx) and a surrogate autonomy engine called the Autonomy Tool Kit (ATK).[9] However, the integration of TACE with the control software was very specific to the individual UAS platforms. Hence, there was a desire to adopt an open architecture approach to facilitate TACE integration with future autonomous systems. The integration effort described below is a major enabling step in that direction.

Unmanned Systems Autonomy Services (UxAS)

The Air Force Research Laboratory (AFRL)-developed UxAS software is a collection of algorithms and utilities for automating mission-level behaviors for teams of unmanned vehicles. Although UxAS is intended to be applicable to a wide variety of unmanned platforms, development to date has been focused on the air domain and some services are specific to air vehicles. However, it is anticipated that many automated behaviors have very similar logic and have cross-domain applicability (e.g., task assignment and scheduling, geographic reasoning, and communication management) (Rasmussen, 2018).

[9] ATK is a software framework and algorithm suite that supports autonomous control and collaboration among unmanned systems. See https://www.youtube.com/watch?v=FksXQKq8H1Y.

The term "mission-level autonomy" encompasses the various capabilities required for a team of UASs to cooperatively and autonomously perform a task-centric mission. In this context, example tasks include performing surveillance over points, lines, and areas, with possible requirements on standoff distance, view angle, and ground sample distance. It may also include surveillance of mobile ground assets and forming cordons around specified areas. Figure 7 illustrates an area search, where the "lanes" the UAS follows to sweep over the area are spaced according to the size of the sensor footprint, and these lanes start and stop where the edge of the sensor footprint touches the area boundary (Kingston, 2016).

Each task might involve a single UAS or require coordination between multiple UASs. Examples of multi-UAS coordination include formation flying or dividing surveillance tasks between UASs under communication constraints (Kingston, 2008). In addition, tasks might have to honor airspace constraints such as "keep-in" and "keep-out" zones. Mission-level autonomy for these types of tasks requires several capabilities, including: task planning, assignment and scheduling, and path planning. A "mission" then consists of a set of assets, a set of tasks to be completed, and the constraints among the tasks (e.g. precedence). UxAS then automates the individual task planning, the assignment and scheduling of assets to tasks, and the online execution and update of each task (e.g. sensor steering or communication updates). UxAS provides these capabilities as well as the communication transport, messaging, configuration, and software control functionality required to implement UAV mission-level autonomy on single UAVs and teams of UASs.

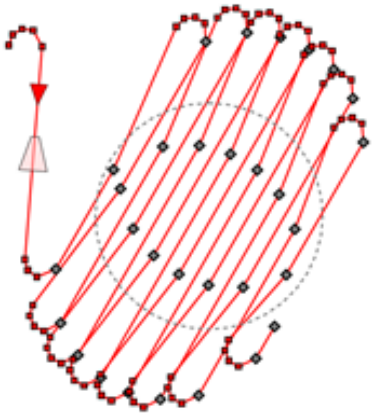

Figure 7 – Area Search Pattern

UxAS is implemented in a service-oriented architecture. That is, UxAS is designed around independent services that interact through the use of pre-defined messages. The result is a modular open architecture that helps manage software complexity, enables extensibility, and is flexible in how it is deployed. UxAS services can run in a single process or they can run over many processes on either one computer or several computers. UxAS leverages the ZeroMQ library (similar to TACE) to connect each of the local services while the Zyre[10] library is used to establish and manage communications with other vehicles. Services can subscribe to messages by message type using a publish/subscribe communication pattern (e.g., a service can subscribe to all state messages in the system). Additionally, UxAS provides a variety of "bridges" that expose the internal messaging queue to external programs (via TCP/IP or serial connections). This allows externally developed functionality to be quickly integrated into an overall system.

Because UxAS is simply the sum of its services interacting via message flow, a designer can easily upgrade or replace a service to enhance functionality or even re-route the message traffic to enforce particular behavior. For example, run-time assurance patterns can be implemented by inserting a monitor/switch service between a service and its downstream partner service.

TACE-UxAS Integration

The core objective of the SCO-led TACE-UxAS integration effort was to fold the TACE capabilities into the UxAS architecture as a set of services accessed via open well-defined interfaces. The intent was to utilize the same Lynx platforms, on-board processors, and test infrastructure as in the TACE EAFB transition, but with a UxAS-based open architecture implementation. The integration strategy is illustrated in Figure 8. The existing UxAS autonomy services and redesigned TACE Watchdog/LVC services were implemented on separate processors to ensure that autonomy failures could not adversely affect Watchdog functionality. Several of the TACE EAFB vignettes were reused and modified/extended as needed to verify the proper operation of the TACE-UxAS integrated system. The full results of these improvements and the associated testing will not be available until late CY18, but testing to date has highlighted the several advantages that an open architecture approach can provide to reduce the time and money needed to integrate test capabilities with autonomous systems in the future. Most notably, flight test engineers and range safety officers will have a revolutionary new tool at their disposal that enables autonomy architectures from

[10] Zyre is an open-source framework for proximity-based peer-to-peer applications. See https://github.com/zeromq/zyre.

across the spectrum of DoD programs to be aggressively and safely tested using a hardware agnostic instantiation of TACE.

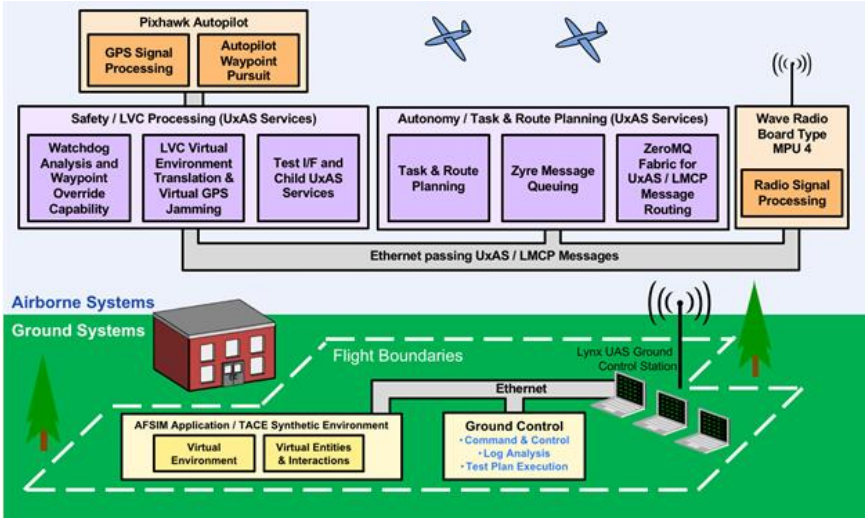

Figure 8 – TACE-UxAS Integration Approach

FUTURE DIRECTIONS

There are many potential transition opportunities for TACE beyond the activities described in the previous sections. For instance, the Emerging Technologies Combined Test Force (ET-CTF) of the EAFB 412 Test Wing is currently developing an approach for testing autonomy called "Services-Based Testing of Autonomy (SBTA)" (Eaton, 2018). This approach is intended to provide a rapid, cost efficient method of testing autonomy and the key enablers of autonomy testing. The initial implementation of SBTA is provided by the TACE-Lynx integration at EAFB. The goal of this testing is to verify that the baseline functionality of the TACE architecture and supporting Watchdog/LVC features function as desired without a focus on the actual mitigations. Following that initial integration, the UxAS instantiation of TACE will then be integrated into the same aircraft and re-tested to verify that all functionalities of the Watchdog and LVC inject capabilities are the same with the new architecture.

The next stage of SBTA development involves more complex testing with the Lynx aircraft. The TACE-UxAS capabilities provide a means of simulating denied communications and GPS jamming effects that will be tested to verify functionality. Additional Watchdog mitigations with increasing complexity and expanded LVC simulation data will be generated to evaluate the flexibility of the TACE system. After completion of baseline performance testing on the Lynx aircraft, integration into other aircraft will begin. These aircraft will provide different mission capabilities to help evaluate the portability of TACE-UxAS integration. This will enable the evaluation of autonomy across different mission profiles, as well as allowing the ET-CTF to determine what test capabilities (data, communications, M&S, safety, etc.) still need to be developed for future autonomous systems.

Upon completion of TACE-UxAS integration, performance testing of dedicated autonomy services will begin. Initial efforts will focus on testing the AFRL developed capabilities already present in UxAS. This will allow for direct comparison to earlier testing performed by AFRL, while focusing on the fully integrated TACE-UxAS instantiation. Additionally, this activity will begin to explore the concept of white box testing[11] to understand data requirements for evaluating autonomy algorithm behaviors in real time. Development of more advanced Watchdog mitigations as well as LVC data sources within the AFSIM framework will also continue. Completion of this phase

[11] White Box Testing is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. This contrasts with black box testing, in which this information is not known to the tester. See http://softwaretestingfundamentals.com/differences-between-black-box-testing-and-white-box-testing/.

will ensure a robust test capability across multiple heterogeneous vehicles and a more informed test infrastructure requirement.

Ultimately, the SBTA developed aircraft with the TACE-UxAS integration will be used for multiple projects to perform an early evaluation of autonomy services, whether it is a new autonomy algorithm, sensor, data fusion or any combination thereof. The TACE Watchdog will be a proven safety capability that will ensure safe operations on test ranges while enabling flexible safety mitigation without complete termination of the test. The TACE LVC capability will provide flexibility to test teams to develop complex open-air test scenarios that will challenge the autonomy under test without extensive infrastructure development. Integration of UxAS into the vehicles with TACE will enable a common capability that autonomy developers can utilize for rapid and controlled testing under the SBTA concept. Eventually, the TACE-UxAS capabilities can be integrated into larger, more complex vehicles for more realistic mission testing.

SUMMARY

This paper is intended to highlight the continued evolution and maturation of the TACE system via a set of focused range transition activities and integration with the complementary autonomous system development/testing capabilities offered by UxAS. It was also intended to highlight current activities to transition TACE to early adopter user organizations and to articulate a vision for what TACE must be capable of supporting in the near future based on stated sponsor requirements. While TACE development/transition has thus far focused on the air domain, the TACE architecture and general philosophy of how to detect and remediate unsafe actions by autonomous systems is entirely transferable to other domains with the proper component instantiations and interface modifications. Thus, TACE provides a viable foundation to whatever fail-safe systems will be needed to support both the testing and fielding of autonomous systems for military missions.

While TACE represents a promising new technology for constraining unsafe behaviors by the AI that underlies autonomous systems, what about the future? Can we stop Judgement Day? How? One possible approach is to capture a Terminator and re-program it to provide assistance to the human population. While this has been shown to work (see "Terminator 2"), capturing a Terminator can be very difficult and inherently dangerous, so that may not be a viable approach. Also, Terminator endoskeletons are hardened and can withstand many different types of weapons, and so engaging them with direct firepower is likely to be a losing strategy. Since any direct interaction with Terminators is generally a bad idea, the most promising strategy to avoid enslavement by these robotic overloads is to stop the rise of the machines from ever occurring. Thus, the ability to keep advancing the technologies needed to counter unsafe AI behaviors in future military systems will be of critical importance. Not only will this allow us to "Tame the Terminator," but will also facilitate the ever-increasing use of autonomous systems in our future warfighting strategies.

ACKNOWLEDGEMENTS

TACE is funded by the Test Resource Management Center (TRMC) and Test Evaluation/Science & Technology (T&E/S&T) Program and/or the U.S. Army Contracting Command Orlando (ACC-ORL-OPB) under contract W900KK-13-C-0036. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Test Resource Management Center (TRMC) and Test Evaluation/Science & Technology (T&E/S&T) Program and/or the U. S. Army Contracting Command-Orlando (ACC-ORL-OPB).

REFERENCES

Chella, A., Manzotti, R., "AI and Consciousness: Theoretical foundations and current approaches", Association for the Advancement of Artificial Intelligence (AAAI) Symposium, Washington DC, November 2007.

Cameron, J.; Hurd, G., "The Terminator," https://www.imdb.com/title/tt0088247/?ref_=nv_sr_4, October 1984.

Bossmann, J., "Top 9 ethical issues in artificial intelligence", World Economic Forum, October 2016.

Marr, B., "The 4 Ds of Robotisation: Dull, Dirty, Dangerous and Dear", Forbes, October 2017.

"Remarks by Deputy Secretary Work on Third Offset Strategy", Deputy Secretary of Defense Speech, Brussels, Belgium, April 2016.

Scheidt, D., Lutz, R., Sturim, J., "The DARPA CODE White Force Network", Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), December 2017.

Scheidt, D., Lutz, R., D'Amico, W., Kleissas, D., Chalmers, R., Bamberger, R., "Safe Testing of Autonomous Systems Performance", Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), December 2015.

Murphy, S., "Guardian Ground Based Detect and Avoid for Unmanned Aircraft Systems Test and Evaluation," Naval Air Warfare Center Aircraft Division, NDIA 33rd Annual T&E Conference, May 2018.

Rasmussen, S., Kingston, D., Humphrey, L., "A Brief Introduction to Unmanned Systems Autonomy Services (UxAS)", International Conference on Unmanned Aircraft Systems, June 2018.

Kingston, D., Rasmussen, S., Humphrey, L., "Automated UAV Tasks for Search and Surveillance", IEEE Conference on Control Technology and Applications, September 2016.

Kingston, D., Beard, R., and Holt, R., "Decentralized perimeter surveillance using a team of UAVs", IEEE Transactions on Robotics, Vol. 24, No. 6, November 2008.

Eaton, C., Chong, E., Maciejewski, A., "Services-Based Testing of Autonomy", ITEA Journal, March 2018.