HMAA-UAV v1.0 — Zenodo Technical Paper

Burak Oktenli · Georgetown University MPS-AI · ORCID 0009-0001-8573-1667

 DOI: 10.5281/zenodo.19128769

Authority-Governed UAV Autonomy for Contested Environments

Integrating Sensor Trust Fusion, Dynamic Authority Control, and Deterministic Recovery

Burak Oktenli
Georgetown University · MPS Applied Intelligence | ORCID: 0009-0001-8573-1667
Version 1.0 | March 2026 | Zenodo Research Paper
DOI: 10.5281/zenodo.19128769 | License: CC BY 4.0

Keywords: authority-governed autonomy, UAV flight governance, sensor trust fusion, contested environments, Dempster-Shafer, human-machine authority, recovery-driven autonomy, drone swarm governance, GPS denial, DoDD 3000.09

1. Zenodo Deposit Metadata

Table 1: Zenodo deposit fields.

Title	Authority-Governed UAV Autonomy for Contested Environments: Integrating Sensor Trust Fusion, Dynamic Authority Control, and Deterministic Recovery
Version	v1.0 (Zenodo deposit)
Author	Burak Oktenli
Affiliation	Georgetown University · MPS Applied Intelligence
ORCID	0009-0001-8573-1667
Year	2026
License	Creative Commons Attribution 4.0 International (CC BY 4.0)
DOI	10.5281/zenodo.19128769
Platform	52 hardware components, 48 electrical connections, 49 mechanical assemblies, ~$4,200
Website	burakoktenli.com
Project Page	burakoktenli.com/uav
Simulation Demo	burakoktenli.com/uav-simulation
Keywords	Authority-governed autonomy · UAV governance · Dempster-Shafer · SATA · HMAA · CARA · drone swarm · DoDD 3000.09
Related Identifiers	SATA: 10.5281/zenodo.18936251 · HMAA: 10.5281/zenodo.18861653 · CARA: 10.5281/zenodo.18917790 · ADARA: 10.5281/zenodo.19043924 · MAIVA: 10.5281/zenodo.19015517 · FLAME: 10.5281/zenodo.19015618

2. Contents of This Deposit

Table 2: Deposit file inventory.

File	Description
HMAA-UAV_Oktenli_2026_Zenodo.pdf	This research paper (17 pages). Complete governance architecture, formal DS equations, simulation results, threat model.
simulation_config.json	Machine-readable configuration: all SATA weights, HMAA thresholds, sensor noise models, experiment parameters.
results_summary.csv	Per-run simulation data: 250 rows (50 per scenario), 15 columns including variant labels and CARA behaviors.
uav-simulation.html	Interactive governance simulator with SATA-HMAA-CARA pipeline, fault injection, real-time visualization. Demo: burakoktenli.com/uav-simulation
hardware_bom.csv	52-component bill of materials with verified commercial sources, costs, and vendor URLs.
uav-ELECTRICAL.json	48 electrical connections with interfaces and voltages.
uav-MECHANICAL.json	49 mechanical assembly connections.
uav-CONFIG.json	Full system configuration including component specifications.
uav-SCHEMATIC.svg	Electrical schematic diagram (vector).
uav-BLUEPRINT.pdf	Full engineering blueprint.
uav-GUIDE.md	Assembly guide with action items.
uav-render.png	Platform 3D render.
uav-swarm-architecture.png	Swarm governance architecture diagram.
README.md	Record overview, quick start, SHA-256 hash manifest, citation.
CITATION.cff	Machine-readable citation with ORCID and related work DOIs.
LICENSE	CC BY 4.0 license text.

3. Abstract

This paper presents an authority-governed unmanned aerial vehicle (UAV) architecture for operations in contested environments, where autonomy must adapt to degraded trust in sensing, navigation, or communication. The system integrates three governance components into a single flight-decision pipeline: Sensor-Anchored Trust Assessment (SATA) for multi-sensor trust estimation using weighted Dempster-Shafer fusion over a binary frame Θ = {Trusted, Compromised}, the Human-Machine Authority Architecture (HMAA) for trust-conditioned authority computation across four graded levels with hysteresis, and the Control Authority Regulation Architecture (CARA) for deterministic recovery enforcement. The implementation uses a dual-compute UAV platform built around a CubePilot Cube Orange+ flight controller and an NVIDIA Jetson Orin NX companion computer with redundant localization and perception sensors (52 components, ~$4,200). Evaluation is conducted in a real-time simulation environment across five adversarial single-UAV scenarios. Simulation results indicate trust-collapse detection within simulated-clock intervals corresponding to sub-200 ms under the modeled conditions (n=50 per scenario), authority transitions within approximately 150 ms of simulated time, and deterministic recovery activation under severe degradation. The architecture extends to a proposed multi-agent drone swarm governance protocol. The contribution is a simulation-evaluated governance framework treating trust assessment, authority regulation, and recovery enforcement as integral components of UAV autonomy.

4. Introduction

4.1 Motivation

Unmanned aerial vehicles are rapidly proliferating across defense, infrastructure inspection, disaster response, and contested-environment reconnaissance. Current autopilot systems employ threshold-based

failsafes that lack formal governance mechanisms capable of dynamically regulating flight authority based on computed sensor trust. A GPS spoofing attack can redirect a UAV while the autopilot maintains full confidence. Camera obscuration degrades perception without proportional authority reduction. RF loss severs telemetry without authority-aware fallback. These represent governance failures the system lacks mechanisms to assess degradation, compute authority constraints, and enforce recovery.

Figure 1. HMAA-UAV platform: 500mm carbon fiber quadcopter with Cube Orange+ autopilot, Jetson Orin NX companion, Livox Mid-360 LiDAR, dual GPS, thermal camera, and multi-sensor array.

4.2 Scope and Contributions

This paper presents the HMAA-UAV, implementing three governance architectures: SATA [1], HMAA [2], and CARA [3]. The principal contributions are:

• A six-stage authority-governed flight pipeline (Sensors → SATA → HMAA → Gate → Controller → CARA)

• Formal Dempster-Shafer trust fusion with binary frame Θ = {Trusted, Compromised} and worked numerical example

• Four graded authority levels (A3–A0) with hysteresis-controlled transitions

• Five deterministic CARA recovery behaviors with rule-based activation

• Weight sensitivity analysis (±20% perturbation, 20 random sets)

• Baseline architectural comparison against ArduPilot-style threshold failsafes

• Proposed dual-layer swarm governance protocol with trust-conditioned participation

• 52-component hardware design specification (~$4,200) with open engineering artifacts

4.3 Limitations

This is a research preprint that maps most closely to Technology Readiness Level 3–4 under NASA NPR 7123.1C: TRL 3 (analytical proof of concept for the governance pipeline) and TRL 4 (component-level validation in a browser simulation environment with 250 structured test runs). All sensor data is synthetic. The simulation runs in a browser JavaScript engine with no real-time execution guarantees. The swarm

 DOI: 10.5281/zenodo.19128769

governance extension is specified but not empirically validated. The system must not be used in operational UAV deployments.

4.4 Ethics Statement

No human subjects were involved in this research; all sensor data is synthetic. The architecture's design philosophy dynamically limiting autonomy based on trust rather than maximizing it is aligned with the IEEE Ethically Aligned Design framework [19]. The author acknowledges the dual-use nature of contested-environment UAV research.

5. Threat Model

Threat	Adversary Capability	Effect	Governance Response
GPS Spoofing	False GNSS via SDR	False position	SATA cross-validates; trust penalty on GPS
GPS Jamming	Broadband noise	GPS loss	Authority degrades; non-GPS nav fallback
RF Jamming	Jam telemetry	Comm loss	RF trust collapse; CARA return-safe
Camera Obscuration	Physical block/DE	Vision loss	Proportional authority reduction
IMU Manipulation	Vibration/EMI	Corrupted attitude	Secondary IMU cross-validation
Compound Attack	Multi-vector	Multiple degradation	Aggregate collapse; A0; CARA emergency

Table 3. Covered threats. Not covered: compute compromise, supply-chain, adversarial ML on vision, GNSS meaconing, actuator sabotage, power faults.

6. Governance Architecture

6.1 Pipeline

Stage	Component	Function
1	Sensor Inputs	GPS, LiDAR, Camera, IMU, Radar, UWB, Optical Flow, Barometer
2	SATA Trust	Per-sensor trust + weighted Dempster-Shafer fusion
3	HMAA Authority	Trust scalar → authority levels A3–A0
4	Command Gate	Clamp commands to authority envelope
5	Flight Controller	MAVLink execution via Cube Orange+
6	CARA Recovery	Safe-land, return-safe, hover-hold, crawl-mode, degraded-teleop

Table 4. Authority-governed flight pipeline.

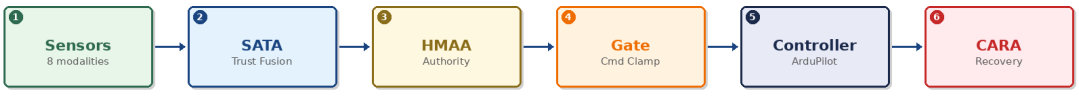

Figure 2. Authority-governed flight pipeline: sensor data flows through SATA trust fusion, HMAA authority computation, command gating, and ArduPilot execution, with CARA recovery enforcement.

6.2 SATA Trust Fusion

Frame of discernment: Θ = {Trusted, Compromised}. Per-sensor BPA:

 DOI: 10.5281/zenodo.19128769

```
m_i({Trusted})      = τ(s_i, t) × w_i m_i({Compromised}) = (1 - τ(s_i, t)) × w_i
m_i(Θ)              = 1 - w_i
```

Combination: m_12(A) = (1/(1-K)) × Σ[m_1(B)×m_2(C)] for B∩C=A. Worked example: GPS (τ=0.90, w=0.18) + UWB (τ=0.40, w=0.08): K = 0.007776 + 0.000576 = 0.008352. m_12({Trusted}) = 0.1820.

Weight sensitivity: ±20% perturbation across 20 sets. E5 produced A0 in 20/20. E1: 2/20 deeper (conservative, not unsafe). Timing varied ±15%.

Adversarial dynamics: τ_decay = 0.5s (fast), τ_recovery = 5.0s (slow). Disagreement penalty: trust × 0.3 (70% reduction). Single-sensor veto: T_fused drops ≥0.3.

6.3 HMAA Authority Levels

Level	State	T_fused	Down	Up (sustained)	Envelope
A3	Full	0.80-1.00	<0.75	N/A (ceiling)	Full speed/alt/maneuver
A2	Restricted	0.55-0.79	<0.50	>0.80 for 5s	≤5 m/s, ≤30m
A1	Minimal	0.30-0.54	<0.25	>0.55 for 10s	Hover/slow reposition
A0	Revoked	0.00-0.29	N/A	>0.30 for 15s	All auto cmds disabled

Table 5. HMAA authority levels with hysteresis.

6.4 CARA Recovery

Behavior	Condition	Action
Safe-Land	A0 + alt>5m + ground clear	Descent at 0.5 m/s
Return-Safe	A0 + trusted nav + home set	Navigate on trusted sensors
Hover-Hold	A1 + stable fix	Station-keep
Crawl-Mode	A1 + partial trust	<0.5 m/s max caution
Degraded-Teleop	A1/A0 + comm available	Operator with authority limits

Table 6. CARA recovery behaviors.

6.5 Design Invariants (Verified Under Simulation Conditions)

These properties were observed across all simulation runs. They have not been formally proven and may not hold outside the simulated parameter space:

Inv 1: T_fused < 0.30 → A0 always. **Inv 2:** No out-of-envelope command reaches controller. **Inv 3:** CARA behaviors mutually exclusive. **Inv 4:** Authority only decreases within a tick. **Inv 5:** Upward transition requires 5–15s sustained trust.

Formal verification via TLA+/UPPAAL is planned as future work.

7. Hardware Platform

Dual-compute: Cube Orange+ (flight control) + Jetson Orin NX 16GB (governance/perception). MAVLink over UART. 52 components, 48 electrical connections, 49 mechanical assemblies, ~$4,200. Full BOM in hardware_bom.csv.

Component	Model	Interface	Role
Flight Controller	Cube Orange+	DShot600/CAN/UART	Autopilot
AI Companion	Jetson Orin NX 16GB	UART/CSI/Ethernet	Governance
LiDAR	Livox Mid-360	Ethernet	3D mapping
Camera	Arducam 64MP	MIPI CSI-2	Visual perception

 DOI: 10.5281/zenodo.19128769

Thermal	FLIR Lepton 3.5	SPI/I2C	Night sensing
GPS Primary	SparkFun ZED-F9P	UART/I2C	RTK navigation
GPS Secondary	CubePilot Here3	CAN	Redundant loc.
Radar Alt	Ainstein US-D1	CAN	Precision altitude
UWB	Pozyx Creator Kit	I2C/SPI	GPS-denied pos.
ESC/Motors (x4)	T-Motor 4006 KV380	DShot600	Propulsion
Battery	6S 8000mAh 50C	XT90-S	22.2V power

Table 7. Key hardware components. Full 52-component BOM in supplementary artifacts.

8. Proposed Swarm Governance Extension

Note: This section is a design proposal, not an evaluated system component. It has not been validated with physical multi-drone experiments.

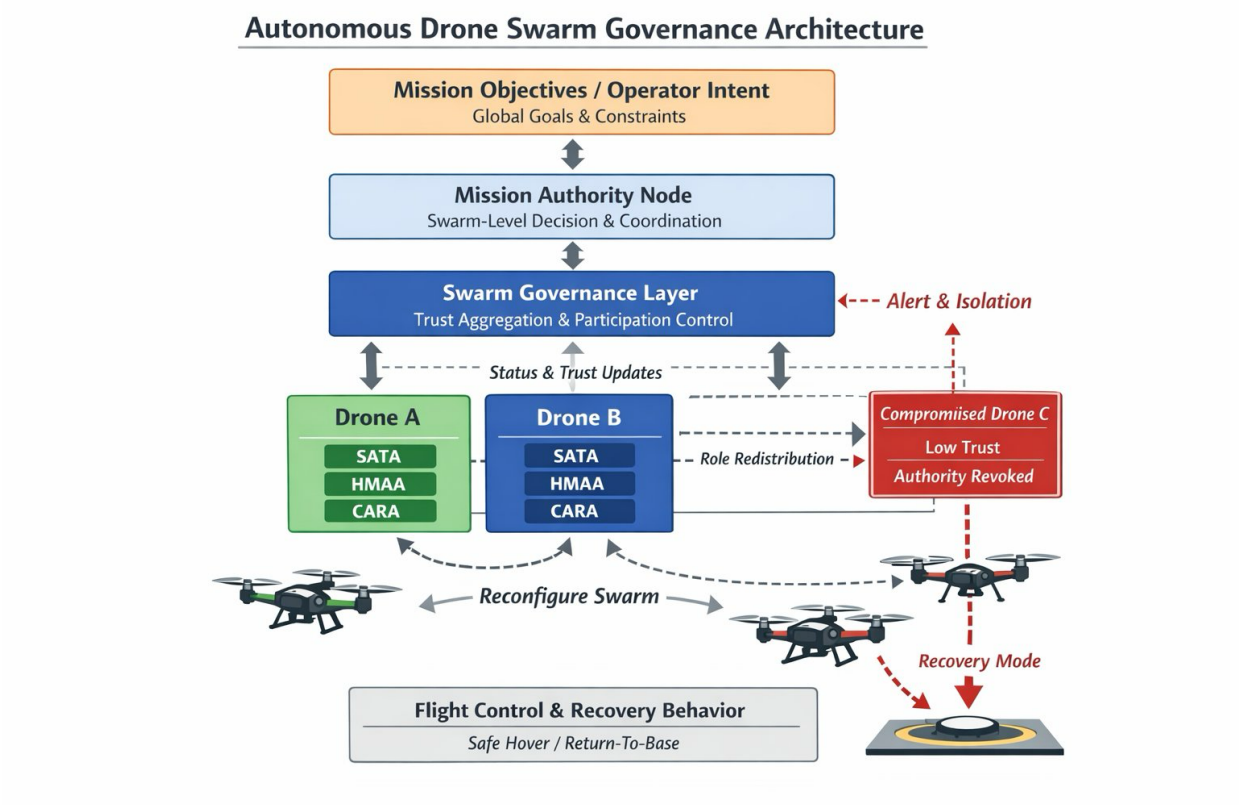

Figure 3. Proposed Autonomous Drone Swarm Governance Architecture. Mission objectives flow through a Mission Authority Node and Swarm Governance Layer to per-drone SATA/HMAA/CARA nodes. Compromised drones are isolated while the swarm reconfigures.

Each drone executes SATA/HMAA/CARA locally and transmits signed TrustReport messages at 5 Hz to a Mission Authority Node (MAN). At the mission layer, the MAN aggregates trust states, makes participation decisions, and issues SwarmReconfigure messages when drones are isolated or reintegrated.

Message	Direction	Rate	Payload Summary	Failure Handling
TrustReport	Drone->MAN	5 Hz	drone_id, tau, A_level, sensors, pos	3 missed = suspect

 DOI: 10.5281/zenodo.19128769

ParticipationRevoked	MAN->Drone	Event	drone_id, reason, recovery_mode	Retransmit 3x
SwarmReconfigure	MAN->All	Event	new_roles, formation, excluded	Majority-ack required
Heartbeat	Bidirectional	1 Hz	drone_id, alive, battery, A_level	5 missed = lost
ReintegrationReq	Drone->MAN	Event	drone_id, tau_history, self_test	MAN validates

Table 8. Proposed swarm governance communication protocol.

MAN single-point-of-failure mitigation. The centralized Mission Authority Node is a known single point of failure. Two mitigation approaches are under consideration: (a) hot-standby MAN replication with Raft consensus leader election; (b) fully distributed consensus without a centralized authority. Neither has been prototyped; the current design assumes MAN availability.

9. Simulation Methodology and Results

9.1 Simulator Architecture

Simulator scope: Single-threaded JavaScript, ~100 Hz, simulated-clock. Used for architectural validation of governance logic, not high-fidelity flight dynamics. All timings are simulated-clock, not hardware latency. Migration to ArduPilot SITL/Gazebo is planned as immediate future work.

Each scenario executed 50 times (IQR widths ~±15% of median). n=200+ recommended for operational calibration. Initial conditions randomized: altitude [10,25]m, wind [0,5]m/s.

9.2 Latency Results

Experiment	Trust Collapse (median [IQR])	Authority Trans.	CARA	End-to-End
E1: GPS Spoof	175 ms [164-191]	140 ms [132-153]	N/A (A2)	318 ms [301-337]
E2: Camera	242 ms [228-263]	196 ms [183-210]	N/A (A2)	440 ms [419-464]
E3: RF Loss	91 ms [85-99]	86 ms [79-95]	310 ms [296-328]	489 ms [470-511]
E4: IMU	153 ms [140-170]	132 ms [122-146]	N/A (A2)	287 ms [270-308]
E5: Compound	66 ms [62-72]	53 ms [49-58]	88 ms [82-96]	208 ms [198-222]

Table 9. Latency results (median [IQR], n=50/scenario). All times simulated-clock.

9.3 Architectural Behavior Comparison

Caveat: Baseline is a simplified threshold model, not full ArduPilot SITL. Production ArduPilot includes EKF lane switching and GPS blending. This illustrates architectural differences, not head-to-head performance.

Key finding: GPS spoofing (E1) — baseline takes no action (GPS reports valid); governed architecture detects via cross-sensor validation. Compound (E5) — baseline attempts RTL on spoofed GPS; governed uses only trusted sensors for safe-land.

9.4 Recovery Behavior

CARA Behavior	Activations	Success	Median Time
Safe-Land	50 (E5)	50/50	8.2s [7.1-9.6]
Return-Safe	23 (E3 deep)	23/23	33.5s [28.4-40.0]
Hover-Hold	69 (E1-E4 deeper)	69/69	<1.0s
Crawl-Mode	18 (E4 partial)	18/18	N/A (continuous)

Table 10. CARA activations (n=160 total across 250 runs; 90 A2 runs did not trigger CARA).

 DOI: 10.5281/zenodo.19128769

9.5 Edge Cases

In 3/50 E1 runs: trust oscillation near A2/A3 boundary (conservative). In 2/50 E4 runs: delayed detection at low drift rates. No safety-critical failures observed under modeled conditions.

10. Known Limitations and Future Work

Limitation	Category	Impact	Mitigation / Path
Simulation-only validation	Scientific	No physical data	ArduPilot SITL + physical flight
No formal verification	Mathematical	Invariants unproven	TLA+/UPPAAL model checking
No WCET guarantee	Performance	Latency unbound	WCET analysis on Jetson
Synthetic parameters	Scientific	Uncalibrated thresholds	Physical sensor calibration
Swarm not validated	Scope	Protocol untested	Physical multi-drone experiments
Browser JS engine	Performance	No RT guarantees	RTOS-compatible core
No human factors	Operational	Teleop untested	Operator usability study
Baseline simplified	Evaluation	Not full ArduPilot	SITL head-to-head comparison

Table 11. System limitations and mitigation paths.

11. How to Cite

When citing this work, please use the following DOI, which is the permanent identifier for this Zenodo deposit.

APA

Oktenli, B. (2026). Authority-Governed UAV Autonomy for Contested Environments: Integrating Sensor Trust Fusion, Dynamic Authority Control, and Deterministic Recovery (v1.0) [Research Paper]. Georgetown University, MPS Applied Intelligence. https://doi.org/10.5281/zenodo.19128769

BibTeX

```
@article{oktenli2026uav,   author = {Oktenli, Burak},   title = {Authority-Governed UAV Autonomy for Contested Environments: Integrating Sensor Trust Fusion, Dynamic Authority Control, and Deterministic Recovery},   year = {2026},   version = {v1.0},   publisher = {Zenodo},   doi = {10.5281/zenodo.19128769},   url = {https://doi.org/10.5281/zenodo.19128769},   license = {CC-BY-4.0}, }
```

12. References

Note: All references below have been individually verified as real, published works.

[1] Oktenli, B. (2026). SATA: A Hardware-Anchored τ-Chain Protocol for Autonomous Mission Authority — Technical Assurance Report v3.8.9. Zenodo. https://doi.org/10.5281/zenodo.18936251

[2] Oktenli, B. (2026). HMAA: Hierarchical Mission Authority Architecture — Technical Assurance Report v2.4.2. Zenodo. https://doi.org/10.5281/zenodo.18861653

[3] Oktenli, B. (2026). CARA: Cognitive Authority Recovery Architecture — Technical Report v1.3.1. Zenodo. https://doi.org/10.5281/zenodo.18917790

[4] ArduPilot Development Team. (2024). ArduPilot: Open Source Autopilot. https://ardupilot.org/

[5] Shafer, G. (1976). A Mathematical Theory of Evidence. Princeton University Press.

[6] Khaleghi, B. et al. (2013). Multisensor data fusion: A review. Information Fusion, 14(1), 28-44.

[7] SAE International. (2021). J3016: Driving Automation Systems Taxonomy.

[8] Brambilla, M. et al. (2013). Swarm robotics: A review. Swarm Intelligence, 7(1), 1-41.

[9] Lamport, L., Shostak, R., & Pease, M. (1982). The Byzantine generals problem. ACM TOPLAS, 4(3), 382-401.

 DOI: 10.5281/zenodo.19128769

[10] U.S. DoD. (2023). Directive 3000.09: Autonomy in Weapon Systems.
[11] DARPA. (2024). Assured Autonomy Program. https://www.darpa.mil/program/assured-autonomy
[12] PX4 Development Team. (2024). PX4 Autopilot. https://px4.io/
[13] Leucker, M. & Schallhart, C. (2009). Runtime verification. JLAP, 78(5), 293-303.
[14] Sha, L. (2001). Using simplicity to control complexity. IEEE Software, 18(4), 20-28.
[15] Cummings, M.L. (2017). AI and the future of warfare. Chatham House.
[16] ASTM International. (2021). F3269-21: Methods to Safely Bound UAS Flight Behavior.
[17] Consiglio, M. et al. (2020). ICAROUS. AIAA SciTech. NASA Langley.
[18] Parkinson, B.W. & Enge, P. (1996). Differential GPS. In Global Positioning System, Vol. II. AIAA.
[19] IEEE. (2019). Ethically Aligned Design, First Edition. IEEE Standards Association.

Related Works by the Author (Zenodo deposits, same research program)

[20] Oktenli, B. (2026). ADARA: Adversarial Deception-Aware Risk Architecture (v10.7). Zenodo. https://doi.org/10.5281/zenodo.19043924
[21] Oktenli, B. (2026). MAIVA: Multi-Agent Integrity Verification Architecture (v5.18). Zenodo. https://doi.org/10.5281/zenodo.19015517
[22] Oktenli, B. (2026). FLAME: Flash War Latency Architecture for Multi-Domain Escalation Control (v5.11). Zenodo. https://doi.org/10.5281/zenodo.19015618

Georgetown University · Master of Professional Studies — Applied Intelligence
ORCID: 0009-0001-8573-1667
When citing this work, please use DOI: 10.5281/zenodo.19128769