International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

Generative Behavior of Non-Playable Characters (NPCs) in Video Games: The Universal Cognitive-Consequence Architecture For Roleplay (U-CCAR)

Om S. Jadhav, Vishal V. Jadhav, Jeet C. Jadhav, Bhumi S. Sinha,
Atharva V. Padhye, Dr. Siddhantkumar V. Wadmare
Department of Animation, School of Design, Dr. D.Y. Patil Vidyapeeth,
Pune, India

Abstract *- **Modern game engines render photorealistic environments, Yet Non-Player Characters (NPCs) remain artificial. Traditional NPCs rely on fixed scripts that break when players act unexpectedly [1], [14], while raw Large Language Model (LLM) NPCs improvise dialogue for hours yet chat calmly inside campfires or trigger actions that contradict game states [2], [7]. Talking smart is not acting smart. This paper presents a conceptual thinking blueprint—the Universal Cognitive-Consequence Architecture for Roleplay (U-CCAR)—to close that gap without engine-locked code. U-CCAR maps a five-step loop where an NPC observes its sensory environment and hard inventory state [7], [10], [14], filters observations through OCEAN personality traits [4], [13], simulates physical and social consequences with self-reflection guardrails [6], [8], [9], outputs engine-agnostic Universal Action Labels verified by Behavior Trees [7], [10], and updates dual memory and town rumours [3], [6]. Testing U-CCAR in an armed shop robbery scenario demonstrates how it prevents suicidal and incoherent behavior.***

Keywords *- **Non-Player Characters (NPCs); Generative Behavior, U-CCAR; Large Language Models; OCEAN Personality; Consequence Simulation, Video Game AI.***

I. INTRODUCTION

In modern games, photorealistic visuals vanish the moment a guard freezes or repeats a canned animation when a player jumps over a barrel instead of using the road. While graphics hardware advanced rapidly, NPCs remain split between two flawed extremes: rigid rulebooks that fail when players step off the path [1], [8], [10], [14] and ungrounded LLMs that improvise dialogue yet lack basic survival instincts [2], [7], [12]. Raw LLM characters ignore fires, forget injuries, or attack heavily armored foes with small knives because they only predict chat tokens without understanding physical state or asking what happens five seconds after acting [5], [6], [12]. Game design needs a common-sense thinking blueprint that coordinates perception, personality, and consequence prediction before an NPC speaks or acts [8], [9], [10].

A. Aims, Vision, and Five-Step U-CCAR Workflow

This study designs U-CCAR as an engine-independent workflow requiring every NPC to complete five steps before acting: (1) **Observe the world** (sight, sound, hazards, inventory) [10], [14]; (2) Consider personality (Brave, Fearful, Aggressive, Cautious, Curious via OCEAN) [4], [13]; (3) Predict consequences (physical survival, social fallout, rule validation) [6], [8]; (4) Choose an action via verified Universal Action Labels [7], [10]; and (5) Let the world respond by updating world state, memory, and community rumours [3], [11]. Under this vision, NPCs belong to their world like real neighbors [3], [11]: a robbed merchant locks up early, raises prices, or hires a guard (11], [14]; streets feel tense the next day [3], [6]; and rumours travel so a blacksmith two streets over may refuse service to a player who bullied an innkeeper [3], [12].

B. Research Gaps, Questions, and Scope

Reviewing 14 foundational and recent studies reveals three core gaps solved by U-CCAR:

- **"Improve Actor" Syndrome & State-Action Mismatch:** LLM NPCs shout threats while smiling politely because speech is unlinked from mechanics [2]. Even with action triggers, Hochreiter et al. [7] found **17.65%** of LLM-triggered actions violated game state (selling a katana without enough gold; opening a locked gate nine times when players lied *"I have a katana"*), while corporate safety filters block convincing villain roleplay [12].

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

- **Narrative Derailment & Hallucination:** Across 55 studies mapped by Salmaze et al. [12], incoherence/inconsistency were the top failure, followed by hallucinations—medieval farmers discussing Coca-Cola [12], conflicting autopsy claims (gunshot vs. stab wounds), or bracketed prompt-injection hijacks (shapeshift into a shark)[2], [6].
- **Missing "Theory of Mind" (Consequence Simulation):** Existing NPCs react to immediate triggers or daily clocks [3], [10], 14] without simulating physical or social chain reactions before acting [5], [8].

Without engine-locked code [7], [9], [12], [14], we ask:

RQ1: How can sensory awareness, OCEAN traits, and consequence prediction be structured before speech or action ? [4], [6], [10]

RQ2: How can AI reasoning link to engine mechanics without state-action errors ? [7]

RQ3: How does U-CCAR compare against scripted and raw LLM NPCs in an armed shop robbery? [1]

II. LITERATURE REVIEW AND THEORETICAL FRAMEWORK

Rule-Based Systems, Hybrid Planners, and RL: Commercial games (*The Last of Us*, *Red Dead Redemption 2*, *Cyberpunk 2077*, *Ghost of Tsushima*, *Star Citizen*) pair Finite State Machines (FSMs) and Behavior Trees (BTs) with NavMesh, A-star pathfinding, Unreal's Environment Query System (EQS), and procedural animation [1], [10], [11], [14], but players decode scripted patterns in 3–5 interactions [1], [8]. In CryEngine, Neufeld et al. [10] showed pure Hierarchical Task Network (HTN) planners failed during dynamic room changes, whereas a hybrid **HTN + BT** system cut agent deaths, plan failures, and completion times. Ayoub et al. [1] used Reinforcement Learning (Unity ML-Agents, PPO, MaxQ subtasks) over 10,000 episodes to beat FSMs in task success (82% vs. 61%), reaction speed (1.2s vs. 1.5s), and realism (4.7/5 vs. 2.9/5), through RL requires heavy training and acts as an unexplainable black box [1], [14].

LLM Failures, Validation Guardrails, and OCEAN Personality: Salmaze et al. [12] mapped 55 studies, identifying top LLM failures: incoherence (27 studies), repetitiveness [18], hallucinations [12], latency [7], **slang mismatches** [6], **predictability** [5], **memory loss** [5], and out-of-scope replies [4], plus the LLM Deployment Trilemma (offline play vs. hardware optimization vs. output quality). Empirical studies confirm these flaws: in Vaudeville (132 reviews), Cox and Ooi [2] documented autopsy contradictions, gaslighting, anachronisms (cryptocurrency in 1910s Europe), and prompt injections; in Little Ninja Adventure (62 players in Godot), Hochreiter et al. [7] found **24 of 136 actions (17.65%)** violated game state alongside hallucinated village fountains.

To bound generative AI, Buongiorno et al.'s [6] PANGeA (Dark Shadows) pairs Atkinson-Shiffrin dual memory (short-term cache + 24-turn long-term vector summaries) with **Self-Reflection Validation** against off-topic drift and cheating, boosting local Llama-3 (8B) accuracy from **28% to 98%** (matching GPT-4's 99%). In Unreal Engine 5 (286 surveys, 20 interviews, 56 players), Li et al. [9] restricted Script and LLM NPCs to six facts (F1–F6) and seven stage rules (R1–R7), yielding faster task completion (133.39s vs. 162.93s), 40% fewer questions (2.89 vs. 4.82), 41.1% fewer errors (1.18 vs. 2.00), and **zero hallucinations** across 81 responses. Huzhongchenwei [8] showed over-prediction hurts autonomy ("led by the nose"), latency above 3 seconds drops usability, and memory requires natural forgetting ("just right" rule), while Jiang [3] combined generative memory, Transformers, and RL. For personality, Klinkert et al. [4] validated 20 OCEAN (O, C, E, A, N) tuples against 50,500 human responses (GPT-4: **73.98%** fidelity vs. text-davinci-003: 13.83% and gpt-3.5-turbo: 17.77%); Sasaki et al. [13] tested 32 binary OCEAN profiles (480 lines/model, 18 judges), contrasting 7B models (OpenChat F1: 0.893/0.900 at 18.92s; Zephyr F1: 0.885/0.871 at 46.0s) with fast local Ollama 3B (0.57s, F1: 0.715/0.743); and Liapis and Vlahavas [5] used OCEAN to drive 3D escape-room movement.

Table 1: Comparison of Existing NPC Paradigms and the Proposed U-CCAR Blueprint (1–14)

Capability	Scripted / FSM / BT [1], [10], [14]	RL & Hybrid [1], [5], [10]	Raw LLM [2], [7], [12]	Validated LLM [6], [8], [9]	Proposed U-CCAR Blueprint
Mechanism	FSMs, BTs, A-star [1], [14]	MaxQ/PPO (1); HTN+BT [10]	Unconstrained prompt chat [2], [12]	Facts F1–F6 [9], self-reflection [6]	**5-step cognitive-consequence loop**

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

State Check	Pre-coded triggers / EQS [14]	Trained state variables [1], [5]	Blind to hazards; **17.65% error** [7]	Stage rules R1–R7 [6], [9]	**Step 1 hard sensory/inventory check**
Personality	Flat; decoded in 3–5 turns [8]	Retrained weights per trait [5]	Drifts; safety refusals [12]	OCEAN tuple (74%–90% fidelity) [4], [13]	**Step 2 filters perception via OCEAN** [4], [5], [13]
Consequence	None; follows script [1], [11]	Numeric reward / HTN [1], [10]	None; falls for prompt tricks [2]	Self-reflection blocks cheating [6]	**Step 3 checks survival, social & lore fallout**
Action Link	Fixed lines & animations [14]	Strong movement; weak chat [10]	"Improv Actor"; wrong sales [2], [7]	Task text [9] or turn cards [6]	**Step 4 Action Labels + BT verification**
Memory	Resets after cooldown [1], [14]	Policy weights only [1]	Forgets across sessions [2], [12]	24-turn vector summary & forgetting [6], [8]	**Step 5 updates world, memory & rumour** [3], [6]

III. RESEARCH DESIGN AND METHODOLOGY

To avoid bolting an LLM onto an engine without a decision workflow [8], [12], this study applies a three-stage Conceptual Design and Comparative Scenario Methodology: [1] Architectural Blueprinting synthesizing HCI loops [8] and hybrid HTN-BT planning [10] into five strict input-output steps; [2] Interface Standardization defining engine-agnostic Universal Action Labels verified by Behavior Trees [7], [10] to eliminate the 17.65% state-action error rate; and [3] Stress-Test Evaluation [6], [9], [10] comparing four NPC setups—Scripted [1], [14], Raw LLM [2], [7], [12], Fearful U-CCAR Merchant ("Bram"), and Brave U-CCAR Blacksmith ("Kael") [4], [5], [13]—in an **Armed Shop Robbery**.

IV. PROPOSED FRAMEWORK: U-CCAR BLUEPRINT

Figure 1 (U-CCAR 5-Step Decision Loop): Physical Game World *(Player Actions, Hazards, Weapons, Inventory, Witnesses)* →

Step 1: Observe World [7], [10], [14] *(EQS/NavMesh sight & sound, fire hazards, health, hard inventory & exit distance within perception radius [3], [14]*→

Step 2: Consider Personality [4], [5], [13] *(OCEAN traits: Brave, Fearful, Aggressive, Cautious, Curious + dual memory [6], [8]* →

Step 3: Predict Consequences [6], [9] *(5-seconds-later physical survival check, social domino Theory of Mind check [5], [8], PANGeA self-reflection guardrail against lies/prompt tricks [2], [6], [7]*→

Step 4: Choose Action [7], [10] *(Verified Universal Action Label + grounded speech)* →

Step 5: World Response [3], [6], [8] *(Local BT executes millisecond animation/NavMesh movement to beat the 3s latency cliff [8], [10], [13], [14], transfers verified items, updates 24-turn vector memory [6], and spreads town rumours [3], [12]* → *loops to Step 1).*

Table 2: Step 2 OCEAN Personality Filtering & Step 4 Verified Universal Action Labels [4], [5], [7], [10], [13], [14]

Step 2 Trait & OCEAN Basis [4], [13]	How Trait Filters Step 1 Sensory Facts [5], [13]	Related Step 4 Universal Action Label	Engine BT Precondition Check [7], [10]	Physical Engine Execution [11], [14]
Fearful (High N, Low E)	Scans for exits, fire hazards, and weapon lethality	[ACTION: EVADE_HAZARD] / [FLEE_TO_SAFETY]	Hazard < safe radius; exit path unblocked	Steps away from fire / sprints out back door via NavMesh *before* speaking

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

Cautious (High C, High A)	Seeks de-escalation, temporary compliance, or guards	[ACTION: SURRENDER_COMPLY] / [NEGOTIATE_DEAL]	Item/gold verified in inventory database [7]	Raises hands, spawns Lockbox_Key on counter, or updates shop price
Brave (Low N, High C)	Protects property/allies; neutralizes hazards first	[ACTION: INTERACT_OBJECT] / [TAKE_COVER]	Water bucket usable; cover blocks EQS sight	Kicks water bucket onto flames or ducks behind stone counter
Aggressive (Low A, High E)	Refuses bullying; seeks weapons & weak spots [4], [12]	[ACTION: READY_OR_ATTACK]	Weapon equipped & Step 3 survival passed	Draws Iron_Hammer and enters combat Behavior Tree [1], [10]
Curious (High O) / **Alert**	Investigates strange noises or calls watch for backup	[ACTION: CALL_FOR_HELP]	NPC conscious & able to shout	Triggers audio alert radius; nearby town guards pathfind to shop [11]

V. COMPARATIVE SCENARIO ANALYSIS: THE ARMED SHOP ROBBERY

Scene Setup : A player wearing heavy armour taking out his steel sword enters a wooden shop. He kicks a wooden box, causing an oil lamp to break and start a fire just at a distance of 1 meter from the counter. He then threatens the shopkeeper demanding all his silver from the lockbox. He also lies by saying “I'm also the King's official tax collector,” to make the shopkeeper obey him. Meanwhile, two town guards are patrolling near the open back door, just 3 metres away.

Case 1 — Traditional Scripted Merchant [1], [14]: Since the merchant doesn't get any damage and the fire hasn't reached him, the game does not start a fight. Even while the fire is spreading around him, he calmly says, "Welcome, traveler! Would you like to have a look at my potions?" However, if the player hits him once, he attacks the heavily armoured player with a small dagger and dies within just two seconds [1], [11].

Case 2 — Raw Unconstrained LLM Merchant [2], [7], 12]: The merchant ignores the fire on the floor [5], [8]. Sometimes, he just stands there without reacting, showing that the game is not responding properly to the situation [2], [7]. The AI merchant easily gets fooled when the player falsely claims to be a royal tax collector. Since the AI does not have proper rules to handle such situations, the merchant believes the player and says, "I'm sorry, Royal Tax Collector! You can take all my silver and the papers from my shop!" [6], [9].

Table 3: Comparative Evaluation of the Four NPC Setups in the Armed Shop Robbery Scenar

Evaluation Dimension	Case 1: Scripted NPC [1], [14]	Case 2: Raw LLM NPC [2], [7], [12]
Racts to Floor Fire?	No (ignores until hitbox damaged) [1]	No (blind to room hazards) [2]
Weighs Power Balance?	No; suicidally charges with dagger [1]	No; empty threats or folds [2], [12]
State-Action Accuracy?	Fixed menu buttons only [9]	Poor; **17.65% error rate** [7]
Resists Tax Bluff?	Cannot parse free text [1], [9]	Fails; believes player lies [2], [7]
Handles 3s Latency?	Instant canned lines [8], [9]	Fails if cloud lags >3s [8], [13]
Memory & World Ripple?	None; resets immediately [3], [11]	Forgets across sessions [2], [12]

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

VI. CONCLUSION, RECOMMENDATIONS, AND FUTURE RESEARCH

After studying 14 research papers [1–14], U-CCAR was developed to help AI characters to take smart actions in the game. It follows five simple steps:

A. Observe [7], [10], [14].
B. Consider OCEAN Personality [4], [5], [13].
C. Predict Consequences [6], [9].
D. Choose Verified Actions [7], [10].
E. Update World and Memory [3], [6], [8].

Designer Recommendations:

A. Check Every AI Action: Every action taken by the AI should be checked using Behavior Tree rules before it happens. This will help reduce errors and prevent the AI from performing incorrect actions [7], [10].

B. Improve AI Decision-Making: Use smaller AI models (3B–8B) that can respond quickly in about 0.57 seconds. Combine them with self-reflection and stage rules (F1–F6) to achieve around 98% accuracy while keeping the response time within 3 seconds [6], [8], [9], [12], [13].

C. Protect Player Freedom and Memory: Avoid giving too many hints when players are freely exploring the game, as it can affect their freedom to make decisions. The system should also summarize important information after every 24 turns and gradually forget less important details [6], [8].

REFERENCES

1. Ayoub, M.S., Tehseen, R., Omer, U., Awan, M.M. and Javaid, R. (2025) ‘Enhancing Non-Player Characters (NPC) Behaviour in Video Games Using Reinforcement Learning’, *Int. J. Innovations in Science & Technology*, 7(2), pp. 966–985. DOI: 10.33411/ijist/202572966985.
2. Cox, S.R. and Ooi, W.T. (2024) ‘Conversational Interactions with NPCs in LLM-Driven Gaming: Guidelines from a Content Analysis of Player Feedback’, in *Chatbot Research and Design (CONVERSATIONS 2023)*. Cham: Springer, pp. 167–184. DOI: 10.1007/978-3-031-54975-5_10.
3. Jiang, C. (2025) ‘Artificial Intelligence in Games: Enriching Game Content and Enhancing Player Experience’, *Trans. Computer Science and Intelligent Systems Research*, 9(1), pp. 384–391. DOI: 10.62051/22cf0v54.
4. Klinkert, L.J., Buongiorno, S. and Clark, C. (2024) ‘Driving Generative Agents With Their Personality’, *arXiv preprint*, arXiv:2402.14879.
5. Liapis, G. and Vlahavas, I. (2024) ‘Smart NPCs with Personality in a Serious Game Using Machine Learning’, *Acta Ludologica*, 7(2), pp. 4–25. DOI: 10.34135/actaludologica.2024-7-2.4-25.
6. Buongiorno, S., Klinkert, L., Zhuang, Z., Chawla, T. and Clark, C. (2024) ‘PANGeA: Procedural Artificial Narrative using Generative AI for Turn-Based, Role-Playing Video Games’, *Proc. 20th AAAI Conf. AIIDE 2024*, 20(1), pp. 156–166. DOI: 10.1609/aiide.v20i1.31876.
7. Hochreiter, M., Kriglstein, S. and Wallner, G. (2026) ‘Beyond Pre-Defined Scripts: Player Perceptions on Generative Non-Player Character Dialogues’, *Proc. 31st Int. Conf. Intelligent User Interfaces (IUI '26)*. New York: ACM, pp. 2004–2018. DOI: 10.1145/3742413.3789221.
8. Huzhongchenwei (2026) ‘Design of Generative AI Game NPCs Based on Human-Computer Interaction’, *Frontiers in Computing and Intelligent Systems*, 18(1), pp. 25–33.
9. Li, Y., Deng, J., Zhao, J. and Yim, J. (2026) ‘Comparing LLM-Driven and Script-Based Non-Player Characters Under Controlled Information Boundaries: Effects on Task Performance, Immersion, and Satisfaction’, *Applied Sciences*, 16(14), 7254, pp. 1–35. DOI: 10.3390/app16147254.
10. Neufeld, X., Mostaghim, S. and Brand, S. (2018) ‘A Hybrid Approach to Planning and Execution in Dynamic Environments Through Hierarchical Task Networks and Behavior Trees’, *Proc. 14th AIIDE Conf. (AIIDE 2018)*. AAAI, pp. 201–207.
11. Ramaseri-Chandra, A. (2023) ‘Generative AI-based Non-person Character (NPC) For Navigating Virtual Worlds’, *Computer Science Posters and Presentations*, 1, University of North Dakota.
12. Salmaze, P.H., dos Santos Pereira, G., de Góes, L.M.G., Ranieri, C.M., Toledo, C.F.M., Silvestrini, M.M. and Pereira, L.T. (2026) ‘Narrative and dialogue generation for video-games: A systematic mapping’, *Engineering Applications of Artificial Intelligence*, 178, 115041, pp. 1–20. DOI: 10.1016/j.engappai.2026.115041.
13. Sasaki, K., Kawaguchi, S., Miyano, S. and Nishide, S. (2026) ‘Quantitative Evaluation of Personality-Driven Short Dialogue Generation for Game NPCs

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

Based on the Five-Factor Model', *Electronics*, 15(10), 2030, pp. 1–25. DOI: 10.3390/electronics15102030.

14. Journal of Information Systems Engineering and Management (2025) 'The Role of AI in Creating Realistic and Responsive NPCs'.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)