Application of Procedural Content Generation to Game NPC Dialogue Design: Research Progress and Challenges

Tingxuan Fan

The Pennsylvania State University, University Park, PA, USA
tingxuanfan0406@gmail.com

Abstract As open-world and role-playing games grow in scale, NPC dialogue increasingly supports characterization, narrative progression, and worldbuilding. Manually authored dialogue provides control but becomes costly across many characters and game states. This review examines procedural content generation (PCG) in NPC dialogue, tracing preset text, branching narratives, template and rule-based systems, and generative AI. It considers narrative design, human-computer interaction, and artificial intelligence. Evidence suggests that procedural techniques can reduce repetitive authoring, while some user studies associate open-ended dialogue with longer conversations and higher engagement in specific settings. These findings do not establish universal superiority. Large language models increase conversational flexibility but introduce narrative inconsistency, character drift, hallucinations, latency, and moderation challenges. The review proposes a risk-sensitive hybrid approach that preserves human-authored critical scenes while constraining AI-generated optional dialogue through game-state checks, character memory, and evaluation.

Keywords procedural content generation; NPC dialogue; interactive narrative; generative AI; large language models

1. Introduction

Non-player characters (NPCs) are central to role-playing, adventure, simulation, and open-world games. Their dialogue communicates quests, world knowledge, and character identity. As worlds expand and narrative states multiply, writing and testing every possible utterance becomes costly. Procedural content generation (PCG) involves computationally assisted creation of content including levels, items, quests, and narrative elements [1]. Applying PCG to NPC dialogue thus represents a shift toward collaboration between human authors and computational systems.

Unlike generic text generation, NPC dialogue must respect character identity, narrative state, and knowledge boundaries. Its central design tension is between player agency and authorial control [2]. Machine learning and large language models amplify this tension by producing lines that have not been individually prewritten [3, 4].

This review asks four questions: How has NPC dialogue generation evolved? Which methods does it use? What value does it offer production and player experience? What limitations and future directions remain? The review draws on Google Scholar, ACM Digital Library, IEEE Xplore, SpringerLink, and ACL Anthology, using search terms including "procedural content generation," "NPC dialogue," "interactive narrative," "generative agents," and "large language models." Relevant peer-reviewed and foundational work was prioritized, while material without a clear relationship to game dialogue or interactive agents was excluded.

2. Theoretical Background

2.1 . Procedural Content Generation

PCG uses algorithms to generate or organize game content automatically or semi-automatically. Shaker, Togelius, and Nelson describe constructive, search-based, constraint-based, and narrative methods [1]. This review uses a broad definition that includes rule-based selection, template filling, and runtime generation. Selecting a prewritten line, substituting variables, choosing a branch, and generating a novel response are distinct degrees of computational participation.

2.2 . NPC Dialogue Systems

NPC dialogue is both a narrative tool and an interaction interface. Unlike generic text generation, it must satisfy character identity, game-state, knowledge-boundary, and

player-action constraints. A character should neither reveal a locked secret nor forget an important completed event without justification. Evaluation must therefore consider linguistic quality alongside characterization, narrative coherence, and responsiveness.

2.3 . Theoretical Framework for NPC Dialogue Generation

From a narrative-design perspective, dialogue must balance player agency and story coherence. Adaptive systems can respond to player behavior while preserving intended dramatic effects [2]. Generation must therefore account for plot stages and consequences.

From a human-computer-interaction perspective, players form expectations about what an NPC knows, remembers, and can do. Contradictions, ignored events, and slow responses weaken credibility. Contextual continuity and timely feedback are essential design goals.

From an AI perspective, dialogue methods have moved from explicit rules toward learned generation. Procedural content generation via machine learning (PCGML) uses models trained on existing game content [3]. LLMs support open-ended input, but practical NPC systems still require world knowledge and narrative constraints [4, 5].

3. Literature Review

3.1 . Preset Text and Branching Narratives

Preset dialogue gives authors direct control over wording, plot, and characterization, but adding branches increases writing, state-management, and testing costs. The interactive drama Façade combined natural-language processing, character behavior, and drama management to allow relatively open player input [6]. As an academic example, it demonstrates that increased freedom still requires narrative coordination.

As an illustrative commercial example, certain quests and dialogue choices in Genshin Impact are subject to prerequisite tasks and state conditions [7]. This demonstrates conditional dialogue organization, not the use of generative AI. Genshin Impact and Façade thus illustrate different aspects of state control and interactive narrative.

3.2 . Rule-Driven and Template-Based Generation

Templates primarily improve the efficiency of text assembly through reusable patterns and variable substitution, although repetition remains noticeable. Rule-driven systems additionally control which lines are available under particular quest or relationship states. Their greater flexibility brings more complicated condition and test scripts [1, 8]. Siegel and Szafron identify filtering and action scripting as significant burdens in dynamic dialogue development [8].

Context-sensitive dialogue goes further by incorporating history. Rather than responding only to the current state, it records past player actions and retrieves relevant events later. The progression is therefore from template composition to conditional selection and then to historical context management.

3.3 . Generative AI and Natural Language Generation

Large language models (LLMs) extend NPC interaction beyond finite menus: players can express intentions in natural language, and the system can generate context-sensitive responses at runtime.

Park et al.'s Generative Agents use memory retrieval, reflection, and planning to shape agent behavior [4]. Their relevance to NPC research lies in persistent memory and behavioral continuity, not evidence of deployment in commercial RPGs. Shao et al.'s Character-LLM investigates role-playing agents trained on character information, experiences, and emotional states [5]. It offers lessons about persona consistency and knowledge boundaries. Neither study should be equated directly with measured outcomes in commercial NPC systems.

3.4 . Gaps in Existing Research

Research repeatedly reveals a trade-off between openness and control. Preset text is verifiable but limited; generative dialogue is flexible but can contradict lore, drift from a character's personality, or reveal information prematurely. Evaluation remains fragmented across naturalness, contextual consistency, narrative coherence, safety, agency, and immersion. Short demonstrations may hide long-term memory failures.

A second gap concerns the unit of evaluation. Many studies assess a dialogue system at the level of individual responses, but an NPC is experienced across a sequence of encounters. A line that is locally fluent can still become problematic if it conflicts with an earlier promise, repeats information the player has already learned, or changes the character's motivation without narrative justification. Generative-agent and role-playing work therefore suggests that evaluation should move from isolated utterances toward session-level and longitudinal consistency [4, 5]. This shift is important because memory, state tracking, and persona stability are precisely the areas in which open-ended systems differ most from preset dialogue.

A third gap is the limited comparability of existing evaluations. Studies use different games, dialogue tasks, player populations, latency settings, and outcome measures [11-14]. Engagement, immersion, naturalness, task completion, and narrative coherence are related but not interchangeable. A system may produce longer conversations without improving story understanding, and a system that players find entertaining in a short experiment may still be too costly or unpredictable for production use. Future comparisons therefore need clearer task definitions and common reporting of interaction length, failure cases, response time, state consistency, and player-level outcomes.

ISSN 3106-7018 (Print) ISSN 3106-5791 (Online)

4. Discussion and Synthesis

4.1 . Technological Evolution and Comparison

Unlike Section 3's chronological review, this section compares methods across five dimensions: controllability, openness, state dependence, development and testing costs, and appropriate use cases. Preset and branching dialogue are controllable but relatively closed; templates efficiently assemble repeated text; rules and contextual systems depend on state and history; LLMs offer greater linguistic openness but create additional validation and runtime costs. No single method is optimal for every NPC interaction.

Critical scenes, irreversible decisions, and secrets generally benefit from human-authored dialogue. Templates suit repeated structured messages, while rules support state-dependent interactions. Constrained LLMs may be more useful for optional open-ended conversation. The design choice should follow narrative risk and interaction requirements rather than a general preference for AI.

Baldur's Gate 3 provides a commercial illustration of how extensive authored branches and dialogue states can support player agency without real-time generative AI [9]. This example illustrates design possibilities but does not by itself establish a general academic conclusion.

The comparison also suggests that the methods should be treated as a continuum rather than as mutually exclusive generations of technology. Preset text can remain the authoritative source for plot-critical information, while rules determine when that information becomes available. Templates can vary low-risk wording, and generative models can be placed at the outer layer of the system for optional conversation. In this layered view, newer techniques do not replace earlier ones; instead, each method is assigned to the part of the dialogue problem that matches its level of risk and required openness. This interpretation is consistent with PCG's broader emphasis on combining authored constraints with computational generation [1, 3].

4.2 . Application Value

Procedural dialogue may reduce repetitive writing for background characters and state-dependent responses. By acknowledging quests and earlier actions, NPCs can make a world feel more responsive; variation may also distinguish repeat playthroughs. Generative systems allow players to express themselves without relying solely on predetermined menus.

Environmental storytelling research shows that characters, spaces, and objects can jointly communicate a game's world [10]. Low-risk background NPCs may therefore complement authored narrative milestones through rule-based or constrained generative variation.

Empirical studies offer qualified evidence. In a pilot RPG study with 64 participants, Ploug, Rimer, and Scirea reported longer conversations and greater engagement for open-ended LLM dialogue, particularly among casual players [11]. A VR study reported positive reactions to natural-language freedom and immersion alongside speech-recognition problems and inconsistent answers [12]. These are measured outcomes within particular settings; the broader inference is that open dialogue may improve some dimensions of interaction, not that longer conversations automatically imply a better overall game experience.

For development practice, the value of procedural dialogue can also be understood as a reallocation of authoring effort rather than simple automation. Human writers can concentrate on plot-critical scenes, character arcs, and high-impact choices, while procedural systems produce or select lower-risk variations. This division is most useful when the generated content is bounded by explicit state and knowledge constraints. It also changes testing priorities: authored dialogue is mainly checked for branch coverage and continuity, whereas generative dialogue additionally requires sampling across prompts, player phrasings, and conversational histories. The benefit therefore depends on whether reduced repetitive writing outweighs the new costs of model integration, validation, and moderation. Evidence from current user studies should be interpreted within this production trade-off rather than as proof that open-ended dialogue should replace authored systems [11, 12, 14].

4.3 . Main Problems and Limitations

At the content-quality level, narrative inconsistency, repetition, memory limitations, and hallucination threaten dialogue reliability [12, 14]. Character consistency also requires explicit persona information and constraints [5]. Linguistic fluency alone is not a sufficient quality measure.

At the system-performance level, response latency affects player experience. Roso et al. found that per-token delay negatively affected some ratings more than initial waiting time, and that character presentation also influenced evaluations [13]. Practical systems must optimize response timing and inference pipelines.

At the production-governance level, live AI dialogue involves input handling, retrieval, inference, content filtering, and potentially speech and animation. These increase deployment, testing, and moderation demands [13, 14]. Writers remain responsible for knowledge boundaries, narrative milestones, and deciding where generation is permitted.

A further limitation is evaluation and maintenance cost. A generated dialogue system creates a much larger effective output space than a finite dialogue tree, so exhaustive manual testing is impossible. Quality assurance therefore has to combine automated checks with sampled human review. Automated tests can verify game-state variables, forbidden knowledge, response length, and safety constraints, but they are less reliable for tone, dramatic pacing, and subtle character consistency. This means that

generative NPCs may reduce some authoring costs while shifting effort toward prompt design, retrieval curation, monitoring, regression testing, and post-release maintenance [14]. The production question is therefore not simply whether AI can generate acceptable lines, but whether the full pipeline can be validated at a cost compatible with the game's scale and risk profile.

4.4 . Future Directions

A feasible engineering direction is a risk-tiered hybrid architecture. Authors lock critical plot events, irreversible choices, and secrets; constrained models handle lower-risk optional dialogue. Game-state checks address narrative consistency; character profiles and relevant memories reduce persona drift; factual and safety filters mitigate hallucination and inappropriate output; caching and optimized inference reduce latency. Responses failing validation should fall back to safe authored lines rather than being shown to players [4, 14].

Long-term memory remains an important direction. Generative-agent research demonstrates retrieval of experiences for future behavior [4]. A game should distinguish world facts, events witnessed by an NPC, secondhand reports, and personal beliefs rather than giving every character omniscient knowledge. Future evaluation should measure character and plot consistency, memory accuracy, response latency, safety, and player experience.

Evaluation should also become risk-sensitive. Plot-critical dialogue should be tested primarily for factual and state consistency, whereas optional social dialogue may reasonably allow more linguistic variation. Player studies should distinguish novelty effects from durable value by including repeated sessions and delayed measures. For commercial deployment, useful reporting would include the frequency of invalid or contradictory responses, fallback rates, end-to-end latency, moderation interventions, and the amount of human revision required. These measures would make it easier to compare a generative system with authored or rule-based alternatives on production value rather than fluency alone.

Overall, the most promising direction is not unrestricted generation but controllable variation. A practical NPC system should expose clear boundaries between authored facts, state-dependent rules, retrieved memories, and generated wording. This separation makes failures easier to diagnose and lets designers decide which component should be corrected when an NPC contradicts the world state. It also supports incremental adoption: a studio can begin with generated optional dialogue, measure failure rates and player response, and expand generation only when reliability is demonstrated.

5. Conclusion

This review answers four questions. NPC dialogue has evolved from preset and branching scripts through templates, state-based rules, historical context, and generative AI. These methods differ in how they manage wording, state, memory, and open-ended language. User studies show potential engagement benefits in limited settings, but results are not universal. Key challenges concern content quality, system performance, and production governance. Generative AI expands conversational freedom without eliminating the need for human authorship. Risk-based layering and a hybrid of authored core narrative with constrained optional generation provide the principal design direction synthesized here.

References

[1] Shaker, N., Togelius, J., & Nelson, M. J. Procedural Content Generation in Games. Springer, 2016.

[2] Riedl, M. O., & Bulitko, V. Interactive Narrative: An Intelligent Systems Approach. AI Magazine, 34(1), 67-77, 2013.

[3] Summerville, A., Snodgrass, S., Guzdial, M., et al. Procedural Content Generation via Machine Learning (PCGML). IEEE Transactions on Games, 10(3), 257-270, 2018.

[4] Park, J. S., O'Brien, J., Cai, C., Morris, M. R., Liang, P., & Bernstein, M. S. Generative Agents: Interactive Simulacra of Human Behavior. Proceedings of UIST 2023, ACM, 2023.

[5] Shao, Y., Li, L., Dai, J., & Qiu, X. Character-LLM: A Trainable Agent for Role-Playing. Proceedings of EMNLP 2023, 13153-13187, 2023.

[6] Mateas, M., & Stern, A. Structuring Content in the Façade Interactive Drama Architecture. Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 2005.

[7] HoYoverse. How can I get a specific daily commission? HoYoverse Help Center, 2025.

[8] Siegel, J., & Szafron, D. Dialogue Patterns—A Visual Language for Dynamic Dialogue. Journal of Visual Languages & Computing, 20(3), 196-220, 2009.

[9] Larian Studios. Community Update #22: Power Of A Mind Flayer. 2023.

[10] Cyran, J. P., Myszor, D., & Cyran, K. A. Environmental Storytelling in Video Games: Crafting Narratives beyond Words. In From Pixels to Play—The Art and Science of Video Games. IntechOpen, 2025.

[11] Ploug, R., Rimer, E., & Scirea, M. Open-Ended NPC Dialogue Favors Casual Players: A Pilot Comparison of Three LLM-Driven Dialogue Systems. 2025 IEEE Conference on Games (CoG), 2025.

[12] Zargham, N., et al. Dialogs with GenAI NPCs: Exploring Player Interactions with Speech Agents in a VR Game. International Journal of Human-Computer Interaction, 2026.

[13] Roso, N., Srebot, S., Markus, N., & Suznjevic, M. When NPCs Take Their Time: Token Latency Effects in LLM-Driven Game Conversations. Entertainment Computing, 58, 101213, 2026.

[14] Salmaze, P. H., Pereira, G. S., de Góes, L. M. G., et al. Narrative and Dialogue Generation for Video-Games: A Systematic Mapping. Engineering Applications of Artificial Intelligence, 178, 115041, 2026.