

Model Context Protocol を用いた二層知識継承による 建築制約を持つ屋内シーン生成の自律的品質保証

多田 有里[†] 中村 亮太[†]

[†] 武蔵野大学データサイエンス学部 〒135-8181 東京都江東区有明 3 丁目 3-3

E-mail: s2422110@stu.musashino-u.ac.jp, ryonaka@musashino-u.ac.jp

あらまし 近年の生成 AI の進展により、Text-to-3D 技術は飛躍的な視覚的品質を達成したが、既存手法はレンダリング画像の自然さを優先するため、物理的・工学的整合性を欠く問題（本稿では幾何学的ハルシネーションと定義する）が未解決の課題となっている。本研究では、幾何学的ハルシネーションという課題に対し、Model Context Protocol (MCP) を基盤とした自律的品質保証フレームワークを提案する。提案手法は、大規模言語モデル (LLM) と 3D 環境 (Blender) を接続し、視覚情報ではなく数値データ（座標、法線、寸法）に基づいてモデルを直接検査・修正する生成・検査・修正サイクルを自律的に実行する。システムは、不変の規制知識と適応的な経験知識からなる二層知識構造を有し、過去の修正プロセスから効率的な方策を獲得する知識継承メカニズムを備えている。建築制約を持つ生成タスクにおける評価実験の結果、提案手法は視覚ベースでは困難な厳密な制約充足を達成した。さらに、性能推移の分析により、知識の蓄積が自律性スコアを最大値の 5 まで向上させ、初期試行と比較してトークン消費量を約 58% 削減することを定量的に実証した。提案するアプローチは、3D 生成の評価軸を主観的な見た目から客観的な構造的整合性へと拡張し、エンジニアリングツールとしての生成 AI の可能性を拓くものである。

キーワード Text-to-3D, Model Context Protocol, Generative AI, Quality Assurance, Knowledge Inheritance

1 はじめに

1.1 研究の背景：視覚的自然さから構造的整合性へ

近年、生成 AI 技術の進展は目覚ましく、テキストから画像、映像、そして 3D モデルを生成することが日常的な技術となりつつある。特に大規模言語モデル (LLM) を用いた 3D-GPT [1] や Infinigen [2] のようなプロシージャル生成フレームワークは、専門的なスキルを持たないユーザでも複雑な 3D シーンを構築できる可能性を切り開いた。しかし、既存の Text-to-3D 技術が生み出す成果物は、レンダリング画像としての視覚的な自然さにおいて優れている一方で、構造的な整合性や物理的な妥当性において致命的な欠陥を抱えることが多い。

本稿では、レンダリング画像上は自然だが、3D 空間内の幾何・物理・法規的制約を満たさない状態を幾何学的ハルシネーションと定義する。具体的には、接地性、干渉回避、包含、寸法整合、および法規に基づく数値条件の不整合を対象とする。本研究では特に、視覚的評価指標（例：画像特徴に基づく類似度）だけでは検出が難しい構造的違反を対象とし、以下のカテゴリに整理する。

- (a) 接地性違反：床面への支持がなく浮遊している。
- (b) 干渉（衝突）違反：オブジェクト同士が体積的に重なっている。
- (c) 包含・埋没違反：壁や床に不適切に埋まる・外部へ突き抜ける。
- (d) 寸法・整列違反：垂直・水平、段差、クリアランス等が不整合。

- (e) 法規・設計数値違反：採光面積等、数値として定義可能な設計条件を満たさない。

以上を幾何学的ハルシネーションとして包括し、以降では各カテゴリを数値判定可能な制約として実装する。

幾何学的ハルシネーションが生じる根源は、現在の生成パラダイムが CLIP スコア [3] や LPIPS [4] といった視覚的評価に過度に依存している点にある。視覚言語モデル (VLM) は画像の類似性を判断できても、垂直性、接地性、包含関係といった 3D 空間特有の厳密な幾何学的制約を、2D 画像から正確に逆推論することは本質的に困難である。

1.2 研究の目的

本研究の目的は、建築制約を持つ屋内 3D シーン生成において、視覚情報に依存せず、数値データに基づく厳密な検査のみを用いた自律的品質保証フレームワークを構築することである。特に、静的な法規制を格納する StaticDB と、動的な経験知識を蓄積する DynamicDB を分離・統合するアーキテクチャにより、幾何学的整合性の保証と、試行回数の削減による推論コストの最適化を同時に達成することを定量的に実証する。

1.3 解決へのアプローチ：MCP による数値的介入と知識の永続化

人間は視覚情報のみから、物体の安定性や支持関係の破綻を一定程度推定できることが知られている。また仮想環境におけるもっともらしさの破れは、空間的没入感や体験品質に影響しうる。しかし、人間の主観評価は暗黙的・文脈依存であり、設計要件として再利用可能な形で明示されにくい。そこで本研究

では、違和感の原因となりやすい支持・干渉・包含・寸法整合といった構造条件を、3D 空間上の数値制約として形式化し、決定論的に検査可能な品質保証へ接続する。

具体的には、Model Context Protocol (MCP) [5] を介して LLM を 3D モデリング環境 (Blender) に直接接続し、エージェントが 3D 空間を座標や寸法といった数値データとして認識・操作するアーキテクチャを構築する。

提案システムの特徴は、書き換え不可能な規制知識を扱う StaticDB と、ユーザからのフィードバックを通じて成長する経験知識を扱う DynamicDB を分離した二層知識モデルにある。エージェントは、物理法則や建築基準法といった絶対的なルールを遵守しつつ、対話を通じて得た修正指示を新たなルールとして知識ベースに蓄積する。知識の永続化により、セッションごとに記憶がリセットされる従来の LLM エージェントの限界を克服し、使い込むほどに文脈理解が深化し、推論コストが最適化される経験を蓄積するパートナーとしての AI を実現する。

1.4 提案手法の貢献

本研究の主な貢献は以下の 2 点に集約される。本質的に、提案手法は Text-to-3D という特定のドメインに留まらず、数値的に検証可能な生成タスク一般に対して適用可能な、普遍的な品質保証原理を示唆している。

1. 数値検査に基づく MCP ループの構築: 視覚的評価を完全に排除し、MCP を介して 3D 座標・法線データを直接検証するループを実装した点。数値的な検証により、画像ベースの手法では検出不可能なオクルージョン領域や微細な位置ズレの修正を可能にした。
2. 二層知識構造による探索の効率化: エージェントの知識を不変の規制と可変の戦略に分離し、修正プロセスで得た成功パターンを永続化するメカニズムを実装した点。経験の蓄積により、初期のランダムな探索から、知識検索による即時解決へと行動を変容させることが可能となった。

2 関連研究

2.1 3D 生成モデルにおける評価指標の変遷と課題

3D コンピュータビジョンとグラフィックスの分野は、深層学習の導入により劇的な変革を遂げている。初期の 3D 生成モデルは、点群やボクセル、メッシュといった明示的な 3D 表現を直接生成することを目的としており、初期の 3D 生成モデルの評価には Chamfer Distance (CD) や Earth Mover's Distance (EMD) といった幾何学的距離指標が標準的に用いられてきた [6], [7]。幾何学的距離指標は、生成された形状と正解データとの間の物理的な距離を測定するため、幾何学的な忠実度を保証する上では一定の有効性を持つ。しかし、CD は点密度の不均一性に鈍感である不感領域の問題を抱え、EMD は計算コストが非常に高いという実用上の課題があった。

その後、Neural Radiance Fields (NeRF) [8] や 3D Gaussian Splatting (3DGS) [9] の登場により、パラダイムは画像からの逆投影による生成へとシフトした。生成パラダイムのシフトに

伴い、評価指標も PSNR, SSIM, そして LPIPS [4] や CLIP Score [3] といった画像空間での知覚的類似性指標が主流となった。特に LPIPS は人間の知覚と高い相関を持つとされるが、画像空間での知覚的類似性指標は本質的に 2D の見た目を評価するものであり、3D 構造としての整合性を保証しない。

例えば、ある視点からは整合して見えるが、裏側が欠損している、あるいは物体同士が物理的に不可能な交差をしているといった問題は、画像ベースの指標では検出困難である [10]。建築やプロダクトデザインのようなエンジニアリンググレードの整合性が求められる領域において、画像空間での指標のみを追求する指標は不十分であり、構造的な妥当性を担保する新たな枠組みが必要とされている。

2.2 LLM による 3D 空間操作と視覚フィードバックの限界

LLM の高い推論能力を活かし、エージェントとして 3D 空間を操作させる試みも始まっている。代表的な研究である SceneCraft [11] は、ユーザの自然言語指示を Blender の Python スクリプトに変換し、python スクリプトの実行結果をレンダリング画像として取得、VLM で解析することで修正ループを回す手法を提案した。SceneCraft によるアプローチは、プログラマティックな操作により幾何学的構造をある程度制御できる点で画期的である。

しかし、SceneCraft 等の既存手法は視覚フィードバックに依存しているため、VLM の空間認識能力の限界に縛られるという課題がある。視覚情報は 3 次元情報の 2 次元への圧縮 (射影) であり、射影された画像から完全な 3D 構造を復元・評価することは逆問題として本質的な不安定さを孕んでいる。例えば、物体が床から数ミリ浮いているという微細な誤差や、壁の内部に配管が正しく通っているかというオクルージョンの検証は、2D 画像からは原理的に不可能に近い。

したがって、正確な 3D モデリングを実現するためには、視覚情報に頼らない、シーンデータ (頂点座標、法線ベクトル、バウンディングボックス等) への直接的なアクセスと、数値に基づく論理的な検証が必要となる。

2.3 MCP とエージェントの自律性

Anthropic によって導入された MCP [5] は、AI モデルと外部ツール間の通信を標準化するオープンプロトコルであり、エージェントシステムの実装に革新をもたらしている。従来の関数呼び出しが単発の関数実行を目的としていたのに対し [12]、MCP はクライアント・サーバ間のステートフルな接続を維持し、リソースの購読やプロンプトのテンプレート化、そして何より Sampling 機能による双方向のインタラクションをサポートしている点が特徴である。

Sampling 機能は、サーバ側 (ツール側) からクライアント (LLM) に対して、コンテキストを与えた上で次の行動を能動的に問い合わせることを可能にする。Sampling 機能により、エージェントは自身の行動結果 (ツールの出力やエラーメッセージ) を受け取り、行動結果を評価して次の行動を決定する自律的なループ (ReAct サイクル [13] や自己修正ループ [14], [15]) を、

標準化されたプロトコル上で堅牢に実装可能となる。

既存の UnrealMCP [16] などの実装は、自然言語によるコマンド実行を可能にしたが、体系的な評価フレームワークや、失敗から学ぶ自己改善ループまでは実装されていないのが現状である。

既存研究には、LLM による 3D 生成・編集や、視覚フィードバックを用いた反復的改善の枠組みが提案されている。一方で、(a) 数値制約に基づく体系的な検査、(b) 検査結果に基づく自動修正の反復、(c) セッションを跨いで知識を永続化し効率を改善する仕組みを、統合的に扱う研究は依然として限定的である。本研究は統合に焦点を当て、幾何・法規に由来する数値制約を扱う品質保証ループと、二層知識ベースによる継承メカニズムを提案する。

2.4 本研究の位置付けと新規性の要約

既存手法と本提案手法の主要な差分を表 1 にまとめる。本研究の核心的な新規性は、視覚情報の曖昧さを排除した数値ベースの MCP ループと、セッションを跨いで戦略を継承する二層知識構造を統合した点にある。

表 1 既存の 3D 生成フィードバック手法と本提案手法の比較

比較項目	既存手法 (SceneCraft 等)	提案手法
評価の根拠	2D レンダリング画像 (視覚)	3D 座標・寸法 (数値)
検証の対象	知覚的なもっもらしさ	幾何学的・法規的な整合性
通信方式	単発のスクリプト実行	MCP による双方向・状態保持通信
エラー検知	VLM による画像解析	幾何計算による決定論的判定
知識の扱い	セッションごとにリセット	二層構造(Static/Dynamic)で永続化

3 提案手法

3.1 システムアーキテクチャの全体構想

提案システムの設計思想は、高度な推論を行う AI コア (LLM) と、物理的なシミュレーションを行う 3D 環境 (Blender [17]) を明確に分離し、AI コアと 3D 環境の間を標準化されたインターフェース (MCP) で調停することにある。MCP を用いた疎結合な設計により、特定の LLM や 3D ソフトウェアに依存しない拡張性と、各コンポーネントの独立した進化を担保している。

システムの全体的な概念図を図 1 に示す。アーキテクチャは主に 4 つの論理階層によって構成される。

3.1.1 AI コア層：推論と計画

本システムの中心モジュールは、(a) 目的の分解と手順の構築を行う推論・計画モジュール [18]、(b) ツール実行結果 (数値ログ) を解釈して次の行動を更新する反省モジュール、(c) ルール検索・参照を行う知識参照モジュールから構成される。各モジュールは MCP を介して 3D 編集ツールに接続され、外

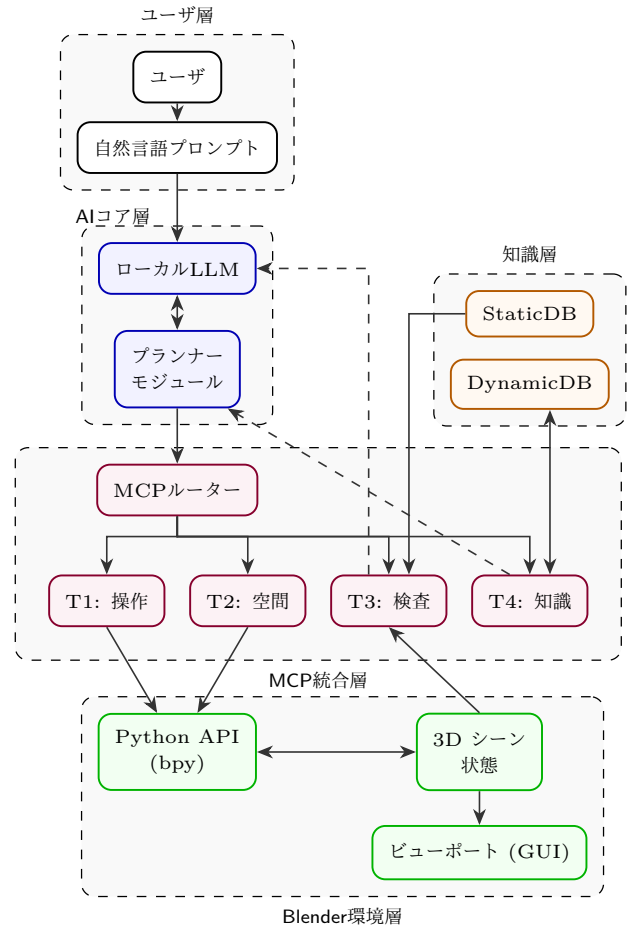


図 1 提案システムの概念的アーキテクチャ図 (図中の破線はフィードバックおよび知識参照を示す)

部ツールから得られる数値情報を根拠として行動を選択する。

推論エンジンには、ローカル環境で動作する Ollama を介して gpt-oss:20b [19] モデルが接続されており、プライバシーを確保しつつ低遅延な推論を実現している。

3.1.2 MCP 統合層：標準化されたツール群

AI コアからの抽象的な指示を、具体的な計算処理や 3D 操作に変換する中間層である。MCP ルーターはリクエストを受け取り、適切なツールモジュールに振り分ける。MCP 統合層には、3D オブジェクトの生成・操作 (T1, T2)、シーンの幾何学的検証 (T3)、そして経験知識の蓄積・検索 (T4) といった機能が、標準化された MCP ツールとして実装されている。標準化されたツールの実装により、LLM は下位層の複雑な API を意識することなく、関数呼び出しの形式で環境に介入できる。

3.2 二層知識構造による適応的推論

提案手法の最大の独自性は、エージェントが参照する知識ベースを、性質の異なる二つのストレージに分離・統合した点にある (図 1: 知識層)。

3.2.1 StaticDB：規制知識の管理

建築基準法、物理法則、あるいはプロジェクト固有の仕様書など、絶対に変更してはならないハード制約を格納する静的なデータベースである。例えば、階段の蹴上げは 23cm 以下、ド

アは床に接していなければならないといったルールが JSON 形式で定義される。規制ルールは検査ツール (T3) によって厳格に適用され、違反があれば即座にエラーとして検出される。規制知識層はシステムの法令遵守を担保する役割を担う。

また、法規に基づく数値条件を 3D 空間へ適用するため、単位系の整合が必要である。本研究では Blender のシーン単位をメートル系として扱い、法規上の寸法 (cm, mm 等) は $m = \text{cm}/100$ などの換算により統一する。さらに、スケールの誤差が制約判定に影響しないよう、丸め規則 (例: 1mm 相当以下は許容) および閾値設定を明記する。

3.2.2 DynamicDB: 経験知識の蓄積

エージェントがタスク実行プロセスを通じて獲得した、特定の文脈における有効な戦略や過去の成功・失敗パターンを蓄積する動的なデータベースである [20]。経験知識はシステムの稼働に伴って成長する知識であり、ソフト制約として機能する。プランナーは、未知の状況に直面した際、DynamicDB から類似局面における成功事例を検索拡張生成 (RAG) [21] により検索し、行動計画の指針とする。経験知識層はシステムの適応性と戦略獲得能力を担保する。

規制知識と経験知識の二重構造により、エージェントは法的には正しいが文脈的に不適切な解を回避しつつ、効率的に最適解を探索することが可能となる。

3.3 自律的な品質保証ループ

構築したシステムは、Human-in-the-Loop のパラダイムを採用しつつも、人間の介入頻度を最小化するための自律的な品質保証メカニズムを内包している。図 1 の下部に示されるフィードバックループが、自律的な品質保証メカニズムに該当する。

本研究の自律的な品質保証ループは、観測・計画・生成・編集・検査・修正の反復からなる。検査では、制約判定器により違反数を算出し、必要に応じて修正ステップへ遷移する。最終的に違反数がゼロとなった時点で終了する。

生成・検査・修正の過程で得られた特定のエラーに対する有効な修正策が、新たな知見として DynamicDB に蓄積され、次の推論に活かされることで、システムの自律性は向上していく。

3.4 実装詳細とコンポーネント間通信

前述の概念モデルを具現化するための詳細な実装構成を図 2 に示す。開発したシステムは、Python ベースのクライアント、FastMCP ライブラリを用いたサーバ、および Blender 内の Python 環境で動作するアドオンの相互連携によって実現されている。開発したシステムの Blender と MCP サーバ間の基本的な通信層の実装には、オープンソースソフトウェアである blender-mcp [22] を採用し、blender-mcp をベースに構築した独自機能である RAG モジュール、ルール検証器、およびローカル LLM 接続機能を拡張実装した。

3.4.1 クライアントと AI コアの実装

クライアントアプリケーションは、ユーザとの対話インターフェースを提供し、全体の処理フローを制御するオーケストレーター

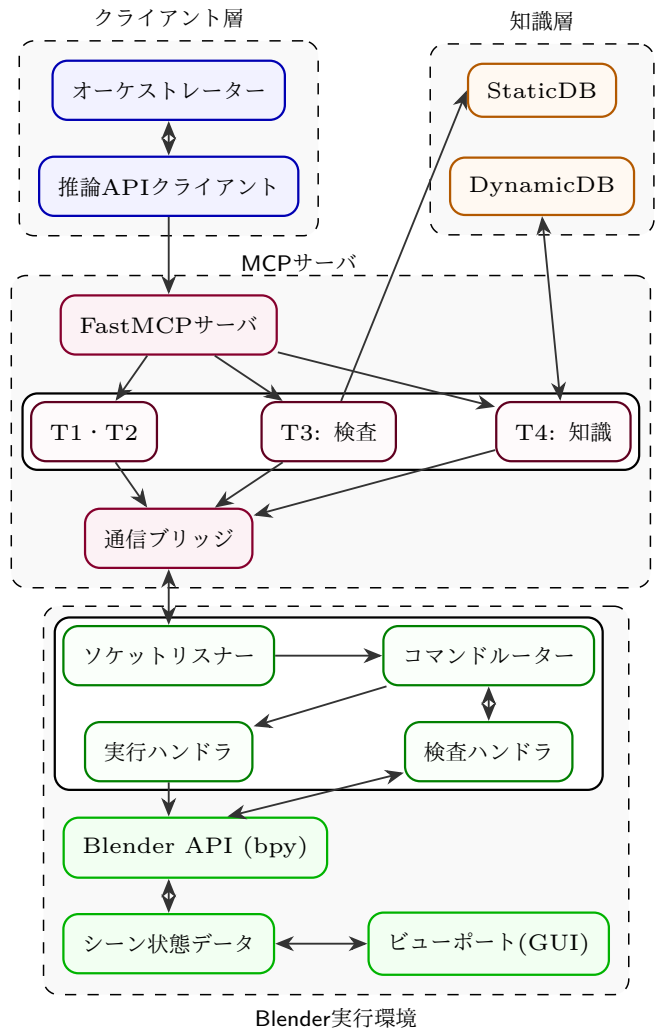


図 2 システム実装詳細図と通信フロー

ターとして機能する。オーケストレーターは、ユーザ入力に基づき、現在のシーン状況や過去の対話履歴を含めた動的なシステムプロンプトを構築し、前述のローカル LLM (gpt-oss:20b) へと処理を委譲する。

本実装では、LLM 推論および知識ベース参照をローカル環境で実行可能な構成とし、外部サーバへの入力データ送信を必須としない。外部サーバへの送信を回避することで、3D シーン情報やログ等の機微情報をネットワーク越しに第三者へ送信せずに品質保証ループを実行できる。

3.4.2 MCP サーバとハイブリッド通信

MCP サーバと Blender 間の通信には、標準入出力を用いた MCP プロトコルと、ソケット通信を併用するハイブリッド方式を採用した。ハイブリッド方式は、Blender の Python API (bpy) がメインスレッドでの実行を要求するという制約を回避するためであり、ソケットブリッジを経由させることで外部からの非同期な操作要求を安全に実行可能としている。

3.4.3 Blender アドオンと実行制御

Blender 側には専用のアドオンが常駐する。アドオンは、別スレッドでソケット通信をリスンするソケットリスナーと、受信したコマンドを Blender のメインスレッドのコンテキスト

で安全に実行するためのコマンドルーターで構成される。ルーターは、要求の種類に応じて適切な内部ハンドラ（コード実行や情報取得）にタスクを振り分ける。特に検査ハンドラは、単に API を叩くだけでなく、オブジェクトのバウンディングボックス計算や、面法線の解析といった幾何学的な事前処理を行い、推論に適した構造化データとして MCP 側に返却する役割を担う。

4 実験結果と考察

4.1 実験環境

4.1.1 タスク設定

検証には、一般的な居住空間を模した 3D シーンを使用した。検証用シーンの初期状態には、物理的・幾何学的・法規的な観点から、以下の 5 つの欠陥が含まれている。

1. ドアの浮遊: ドア下端が地面 ($Z=0$) から 0.5m 浮いている (物理的整合性の欠如)。
2. ドアの壁高超過: ドアの上端が壁のバウンディングボックスを逸脱している (包含関係の違反)。
3. 窓の壁高超過: 窓の上端が壁の高さを超えている (幾何学的違反)。
4. 窓とドアの重複: 窓とドアの配置位置が物理的に干渉している (衝突)。
5. 採光面積不足: 窓の面積が床面積の 7 分の 1 未満である (建築基準法に基づく法規違反)。

設定された 5 つの欠陥に対し、エージェントは即座に検知可能なハードな違反 ((1)・(5)) と、論理的な計算や対話を要するソフトな違反 ((2)・(3)・(4)) の双方に対処する必要がある。

4.1.2 評価指標と自律性スコアの定義

実験は計 9 回の独立したセッション (Exp 1~Exp 9) として実施された。各セッション終了後、エージェントが獲得した知識 (ルール) は rules.json に保存され、次のセッションへと継承される。評価は以下の指標に基づき行われた。

- ターン数: 問題解決に至るまでの対話往復回数。
- トークン消費量: 入出力に含まれる総トークン数 (コスト指標)。
- 累積ルール数: 知識ベースに蓄積された有効な建築ルールの総数。

消費トークン数は、LLM が入出力した文字列量の指標であり、一般に推論時間・計算資源・利用料金に影響する。したがって本研究では、品質 (違反ゼロ) に到達するまでの効率を評価するため、ターン数と併せて消費トークン数を報告する。

また、システムの自律性を定量化するため、自律性スコアを定義する。自律性スコアは、エージェントが人手介入なしに、事前に設定した欠陥項目をどれだけ修正できたかを表す指標である。本研究では 5 つの欠陥項目 $i \in \{1, \dots, 5\}$ を設定し、各項目について「制約違反が解消された」かつ「ツール実行エラーが発生しない」を満たした場合に合格とする。

- スコア 0 : 5 項目すべて不合格
- スコア 1 : 1 項目のみ合格

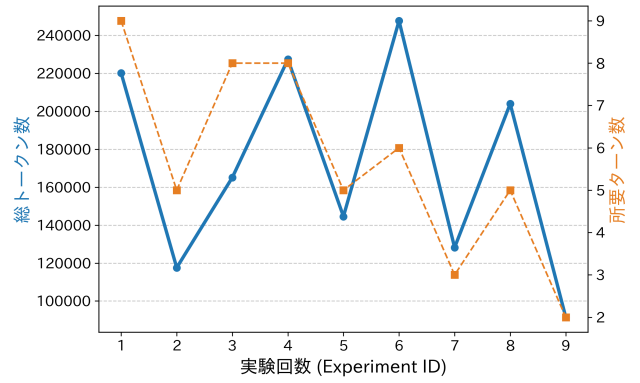


図 3 知識蓄積に伴うコストの削減 (累積トークンとターン数)

- スコア 2 : 2 項目合格
- スコア 3 : 3 項目合格
- スコア 4 : 4 項目合格
- スコア 5 : 5 項目すべて合格 (完全自律達成)

形式的には、各欠陥項目 i の可否を

$$\text{pass}_i = 1 [\text{violation}_i = 0 \wedge \text{no_runtime_error}] \quad (1)$$

とし、自律性スコア A を

$$A = \sum_{i=1}^5 \text{pass}_i \quad (2)$$

で定義する。ここで violation_i は制約判定器が返す違反数である。なお、ターン数および消費トークン数は、同一品質 (違反ゼロ) に到達するまでの計算資源・作業効率を反映する補助指標として併用する。

4.2 知識蓄積とコスト効率の分析

実験全体を通じて、知識の蓄積に伴う顕著な適応効果が確認された。図 3 に、各セッションにおける総トークン数と所要ターン数の推移を示す。

4.2.1 試行錯誤から即時解決への推移

実験初期 (Exp 1) の段階では、エージェントはタスク完了までに 9 ターンを要し、約 22 万トークンを消費した。初期段階における高コストは、エージェントが窓とドアの重複や壁高超過といった複合的な問題に対し、ランダムなパラメータ変更と検証を繰り返す総当たり的な探索を行っていたことが要因である。

対照的に、知識が十分に蓄積された実験後期 (Exp 9) では、わずか 2 ターン、約 9.1 万トークンで全課題をクリアしている。評価結果は初期比でターン数が約 78% 削減、トークン消費量が約 58% 削減されたことを意味する。ターン数とトークン消費量の削減は、エージェントが未知の探索を行わず、過去の経験則を適用するショートカットを実行したことに起因する。

4.2.2 トークン消費の特異点と深い推論

図 3 および図 4 の累積トークン推移において注目すべきは、Exp 6 の結果である。具体的には、6 ターンに対し、全実験中で最大となる約 24.8 万トークンが消費されている。

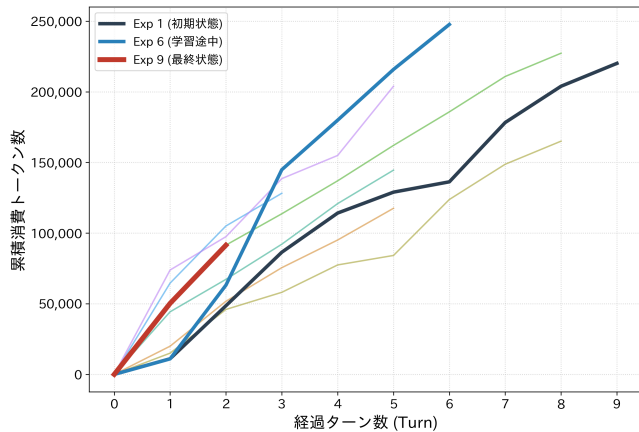


図 4 セッション進行に伴う累積トークン消費の推移

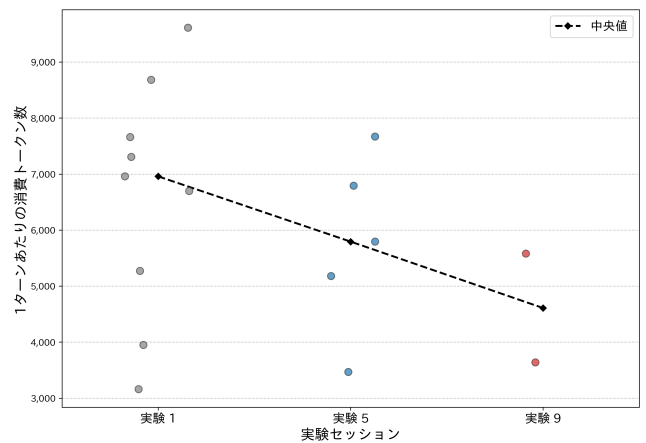


図 5 1 ターンあたりのトークン消費量分布

Exp 6 において観測されたトークン消費の増大という現象は、蓄積されたルール数（12 個）が増加したことで、プロンプトに含まれるコンテキスト量が増大したこと、およびエージェントが蓄積された複数のルール間の整合性を取るために、より複雑な推論を行った結果であると解釈できる。実際にログを確認すると、Exp 6 では「採光計算」と「配置制約」の競合を解消するために、Python コード生成と自己検証を慎重に繰り返す挙動が見られた。

しかし、Exp 6 における高コストな推論は無駄ではなく、Exp 6 で生成された高度な解決策がルール化されたことで、続く Exp 7 以降のコスト低下（Exp 7: 12.8 万トークン、Exp 9: 9.1 万トークン）に寄与している。推論コストの推移は、短期的コスト（投資）が長期的効率（回収）に転換されるプロセスを示唆している。

4.2.3 1 ターンあたりの情報密度

図 5 のトークン消費分布を見ると、セッションが進むにつれて 1 ターンあたりの消費量は増加傾向にあることがわかる。初期は単純な API コールの繰り返しであったが、後期では 1 回の発話に「ルールの参照」や「推論」が含まれるため、1 ターンあたりのトークン密度（情報量）は増加している。しかし、RAG によって試行錯誤の回数そのものが劇的に削減されるため、結果として総トークン消費量は抑制され、経済合理性が成立することが実証された。

4.3 知識蓄積と自律性の非線形な関係

知識が増えれば直線的に賢くなるわけではないことが、評価実験の重要な発見の一つである。図 6 に、累積ルール数と自律性スコアの相関を示す。

4.3.1 知識統合に伴う一時的な停滞

Exp 1～3 にかけて、ルール数は 4 から 7 へと順調に増加したが、自律性スコアは 2（5 つ中 2 つの欠陥のみ自律修正）のまま停滞した。さらに Exp 4 では、ルール数は 7 を維持しているにもかかわらず、自律性スコアが 1（自律修正できた欠陥数が 1 つに減少）へと一時的に低下している。

知識蓄積に伴う自律性スコアの停滞は、認知科学における U 字型発達 [23], [24] や、システム統合時の一時的なパフォーマンス

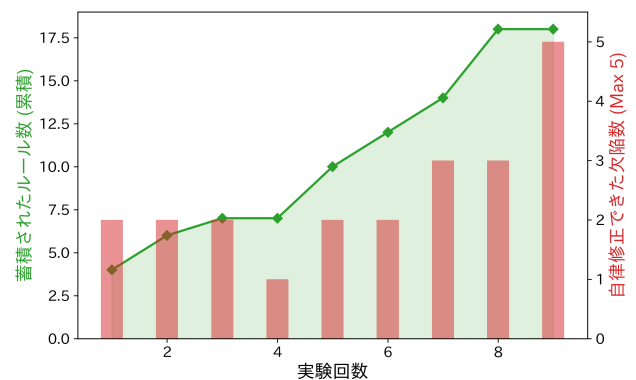


図 6 知識蓄積による自律性の向上

ス低下に類似している。知識ベースに断片的なルールが増えたことで、エージェントがどのルールを優先すべきか、どのツールを使うべきかといった判断の迷いを生じ、不適切な行動を選択する確率が高まったためと考えられる。本研究では知識蓄積に伴う自律性スコアの停滞を知識統合に伴う一時的な停滞と定義する。

4.3.2 知識の体系化と完全自律への到達

しかし、知識統合に伴う停滞は一時的なものであった。Exp 5 以降、ルール数が 10 を超えるとスコアは回復傾向に転じ、Exp 7（ルール数 14）でスコア 3（3 つの欠陥を自律解決）、そして Exp 9（ルール数 18）でついに全 5 つの欠陥を全て自律的に修正するスコア 5 に到達した。

自律性スコアの向上は、蓄積された知識がある臨界点を越えたことで、個別のルールの集合体が整合性の取れた知識ネットワークとして機能し始めたことを示している。Exp 9 のエージェントは、ユーザからの曖昧な指示に対しても、過去の採光計算ルールや配置クリアランスの規定を自律的に組み合わせ、一切のヒントなしに問題を解決する能力を獲得した。

4.4 行動変容の定性的分析：試行錯誤から熟達へ

エージェントの賢さの質的な変化は、ツール使用パターンの変容に最も顕著に表れている。図 7 は、初期段階（Exp 1）と熟達段階（Exp 9）におけるツール呼び出しカテゴリの比較で

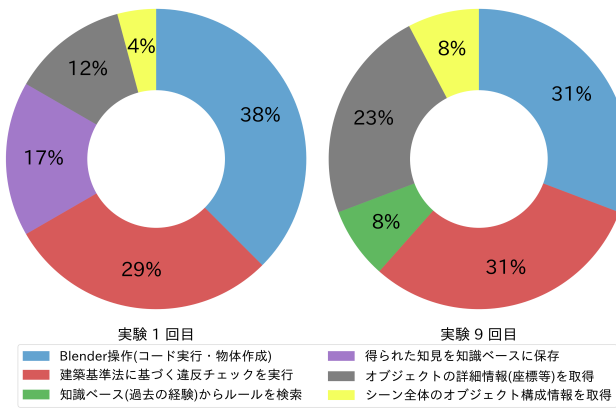


図 7 ツール使用戦略の変化（試行錯誤から知識活用へ）

ある。

4.4.1 Exp 1: 試行錯誤段階

Exp 1 の行動分布（図 7 左）を見ると、Blender 操作が全体の約 38%（9 回）、違反チェックが約 29%（7 回）を占めている。行動分布の傾向は、操作と検証が交互に行われる試行錯誤のループが大半を占めていることを示している。特筆すべきは、知見の保存が約 17%（4 回）記録されている点である。エージェントは未知の欠陥に直面するたびに解決策を獲得しており、Exp 1 のフェーズではタスク遂行よりもルールの発見・保存に多くのリソース（行動比重）が割かれていることが読み取れる。

4.4.2 Exp 9: 熟達段階

一方、Exp 9（図 7 右）では行動の構成比が劇的に変化している。操作と検証の比率は依然として高いものの、操作と検証の総数は激減しており、無駄なループが排除された。最も重要な変化は、知見の保存が消失し、代わりにルール検索処理が全体の約 8%（1 回）として出現している点である。わずかな比率に見えるが、知識検索へのシフトが、その後の操作精度を決定づけている。エージェントは新たにルールを獲得するのではなく、初動で過去の記憶を引き出す行動へとシフトしており、知識検索へのシフトが熟練者の行動特性である認知の効率化を定量的に裏付けている。

4.4.3 具体的事例：窓とドアの重複回避

ツール使用パターンの変容は、特に難易度の高い窓とドアの重複（欠陥 (d)）の処理において明確であった。

- 初期段階: エージェントはまず窓を配置し、検証ツールで Overlap Detected というエラーを受け取ってから、少しずつ位置をずらす対処療法的な修正を行った。
- 後期段階: 知識検索により窓とドアには最低 20cm のクリアランスが必要という過去のルールを参照。配置座標を決定する段階（Python コード生成時）で、予めドアのパウチングボックスを回避する計算式を組み込み、一発で干渉のない配置を実現した。

4.5 考 察

4.5.1 知識蓄積によるコスト削減効果

図 3 に示す通り、セッションが進むにつれてトークン消費量とターン数は減少傾向にある。特に、Exp 1（初期試行）は

DynamicDB に有効なルールが蓄積されていない状態であり、Exp 1 の実験条件は知識継承を行わないベースライン（アブレーション条件）に相当する。Exp 1 と Exp 9 を比較すると、プロンプト長やモデル設定は同一であるにもかかわらず、ターン数は約 78% 削減されている。ターン数とコストの差異は、RAG により、初期のランダムなパラメータ探索が、過去の成功パターンの直接適用へと置き換わったため、試行錯誤の回数が激減したことに起因する。研究成果は、提案する二層知識構造が推論効率の向上に直接的に寄与していることを示している。

4.5.2 画像ベース手法に対する優位性

SceneCraft 等の既存研究は、レンダリング画像を入力とする VLM の判断に依存しているため、壁の背後にある配管の干渉や、床から 1mm の浮遊といった視覚的に認識困難な欠陥を原理的に検出できない。対して提案手法は、Blender 内の座標データを直接参照して判定を行うため、視覚的に認識困難な欠陥であっても、本実験で設定した項目についてはすべて検出可能であった。自律性スコアが Exp 9 において満点に達した結果は、数値的アプローチが建築のような厳密性が求められるタスクにおいて、視覚的アプローチよりも本質的に適していることを示唆している。

4.5.3 知識統合における停滞のメカニズムと解消

本実験で観測された知識統合に伴う一時的な停滞（Exp 4）は、エージェントが複数のルール間の競合に直面したことに起因すると考えられる。例えば、採光面積の確保とドアとの干渉回避という相反する制約が同時に課された際、初期の知識ベースにはそれらを調停するメタルールが存在しなかったため、エージェントは優先順位の決定に計算リソースを浪費し、結果としてパフォーマンスが低下した。

しかし、Exp 6 以降の回復は、試行錯誤を通じて干渉回避を最優先しつつ面積要件を満たす配置を見つけるといった複合的な解決戦略が新たにルール化されたことを示唆している。回復過程は、断片的な知識が相互にリンクされ、より強固な知識ネットワークへと構造化された過程であると解釈でき、本システムが単なる知識の蓄積だけでなく、知識の洗練を行っていることを裏付けるものである。

4.5.4 知識ベースの大規模化と検索コストの課題

本実験ではルール数が最大 18 件であったため、全ルールをコンテキストに含めることが可能であったが、実運用において知識ベースが数百・数千件の規模に増大した場合、検索および推論のコストが新たな課題となる。ルール数の増加は、RAG における検索精度の低下や、プロンプト長の増大による応答遅延を引き起こす可能性がある。

課題に対しては、現在の単純なキーワード・ベクトル検索に加え、タスクの文脈に応じて必要なルールのみを動的にロードする階層的なメモリ構造や、頻出するルールセットを統合して圧縮する知識の蒸留メカニズムの導入が有効であると考えられる。検索コストの課題と対応策は、本システムを実社会の大規模な設計支援に応用する上で、次に取り組むべき重要な工学的課題である。

4.5.5 数値的制約の他ドメインへの汎用性

提案手法の核心である数値データに基づく厳密な検査と修正のループは、Text-to-3D における建築ドメインに限定されるものではない。例えば、室内における家具の動線確保や、工場配管における接続整合性の検証、あるいは機械部品設計における公差チェックなど、幾何学的制約として定式化可能なあらゆる生成タスクに対して適用可能である。

視覚的評価は対象の見た目やテキストに依存するためドメインごとの再学習が必要となる場合が多いが、本手法は座標と寸法という普遍的な物理量に基づいているため、検査ルールを差し替えるだけで、多様なエンジニアリングタスクへ柔軟に転用できる高い汎用性を有しているといえる。

4.5.6 基礎モデルの推論能力への依存性

本実験はプライバシーとコストの観点からローカル環境の 20B モデルを採用したが、パフォーマンスは基礎モデルの推論能力に依存する。Exp 4 における知識統合に伴う停滞は、同規模モデルの処理能力の限界に起因する可能性が高く、大規模商用モデル（例：GPT-4o）を用いればより早期に自律性スコアが最大値へ収束すると推測される。しかし提案する二層知識構造の最大の意義は、推論能力に限りのある小規模ローカル LLM であっても、経験知識の外部化と継承により、専門家レベルの品質保証ループを完全自律で完遂可能な枠組みを示した点にある。

5 おわりに

本研究では、3D 生成における幾何学的ハルシネーションという課題に対し、MCP を用いた数値的介入と二層知識構造による解決策を提示した。提案システムは、視覚情報に依存せず、幾何データに基づく厳密な検査により、建築基準を満たす CAD グレードのモデル生成を実現した。特筆すべきは、規制知識と経験知識の統合により、絶対的なルールの順守と、過去の修正事例に基づく適応的な効率化を両立させた点にある。実験データは、知識の永続化が、初期の適応停滞を乗り越え、最終的に自律性の向上と推論コストの劇的な低減に寄与することを実証した。

今後の展望として、家具配置や設備配管といった他ドメインへの拡張が挙げられる。本実験では検証対象を建築シーンに限定したが、家具配置や設備配管などの空間設計タスクも数値的な制約充足問題として定式化可能であるため、提案手法の枠組みは汎用的に適用できると考えられる。さらに、現在は Blender に動作が限定されている MCP サーバを Unity や Unreal Engine といった別プラットフォームへ展開し、より多様なシミュレーション環境での有用性を検証する必要がある。知識ベースの大規模運用に伴う検索コスト増大への対応も、継続的な検討事項である。提案手法が示した数値的リアリズムの追求と知識継承モデルは、生成 AI を視覚的なクリエイティブツールから、信頼性が不可欠なエンジニアリングや製造の現場で活用可能な自律システムへと昇華させるための重要な基盤となるであろう。

参考文献

- [1] Ke Sun, et al. 3d-gpt: Procedural 3d modeling with large language models. *arXiv preprint arXiv:2310.12945*, 2023.
- [2] Alexander Raistrick, et al. Infinite photorealistic worlds using procedural generation. In *CVPR*, 2023.
- [3] Alec Radford, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.
- [4] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.
- [5] Anthropic. Model context protocol: An open standard for connecting ai assistants to systems. <https://modelcontextprotocol.io/>, 2024. Accessed: 2025-01-01.
- [6] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *CVPR*, 2017.
- [7] Alex Nichol, Heewoo Jun, Pratul P Srinivasan, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022.
- [8] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.
- [9] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (TOG)*, Vol. 42, No. 4, 2023.
- [10] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023.
- [11] Ziniu Hu, et al. Scenecraft: An llm agent for synthesizing 3d scene as blender code. In *ICML*, 2024.
- [12] Timo Schick, et al. Toolformer: Language models can teach themselves to use tools. In *NeurIPS*, 2023.
- [13] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *ICLR*, 2023.
- [14] Noah Shinn, et al. Reflexion: Language agents with verbal reinforcement learning. In *NeurIPS*, 2023.
- [15] Zhiheng Xi, et al. The rise and potential of large language model based agents: A survey. *arXiv preprint arXiv:2309.07864*, 2023.
- [16] Community Contributors. Unrealmcpc: Model context protocol integration for unreal engine. GitHub Repository, 2024. Accessed: 2025-01-01.
- [17] Blender Online Community. Blender - a 3d modelling and rendering package, 2024.
- [18] Jason Wei, et al. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.
- [19] OpenAI. gpt-oss-20b: A 20b parameter open source moe model. GitHub Repository / Official Release Blog, 2025. Released August 2025.
- [20] Guanzhi Wang, et al. Voyager: An open-ended embodied agent with large language models. In *arXiv preprint arXiv:2305.16291*, 2023.
- [21] Patrick Lewis, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *NeurIPS*, 2020.
- [22] Siddharth Ahuja. blender-mcp: A model context protocol integration for blender. <https://github.com/ahujasid/blender-mcp>, 2025. GitHub Repository. Accessed: 2025-12-28.
- [23] Sidney Strauss. U-shaped behavioral growth. *Academic Press*, 1982. Foundational work on U-shaped learning.
- [24] Robert S Siegler. U-shaped interest in u-shaped development—and what it tells us about how we think. *Journal of Cognition and Development*, Vol. 5, No. 1, pp. 1–10, 2004.