

# 生成 AI による単純形状 3 次元データモデルの 対話的な生成手法の一考察

山本 敦大<sup>1</sup>・緒方 陸<sup>2</sup>・藤井 純一郎<sup>3</sup>・山本 一浩<sup>4</sup>

<sup>1</sup>正会員 八千代エンジニアリング株式会社 技術管理本部 CIM 推進室  
(〒111-8648 東京都台東区浅草橋五丁目 20-8 CS タワー)

E-mail: at-yamamoto@yachiyo-eng.co.jp (Corresponding Author)

<sup>2</sup>非会員 八千代エンジニアリング株式会社 技術開発研究所  
(〒111-8648 東京都台東区浅草橋五丁目 20-8 CS タワー)

E-mail: rk-ogata@yachiyo-eng.co.jp

<sup>3</sup>正会員 八千代エンジニアリング株式会社 技術開発研究所  
(〒111-8648 東京都台東区浅草橋五丁目 20-8 CS タワー)

E-mail: jn-fujii@yachiyo-eng.co.jp

<sup>4</sup>正会員 八千代エンジニアリング株式会社 技術管理本部 CIM 推進室  
(〒111-8648 東京都台東区浅草橋五丁目 20-8 CS タワー)

E-mail: kz-yamamoto@yachiyo-eng.co.jp

近年、土木分野では生産性の低さが問題となっており、BIM/CIM (Building/Construction Information Modeling, Management) による生産性向上が進められている。一方で、BIM/CIM で活用する 3 次元データモデル作成には手間や時間を要する。そこで、3 次元データモデルの作成を省力化するため、本研究は生成 AI (Artificial Intelligence) を用いた単純な 3 次元データモデルの対話的な生成手法を提案する。生成 AI を用いることで、人手によるモデリングを生成 AI への自然言語による指示に代替できる。本研究では IFC (Industry Foundation Classes) 4.3 を対象とし、検証実験として、単純な形状 (直方体、円柱、球) のデータモデルの生成を試みた。検証実験の結果、作成したいモデルを正解とした提案手法の精度は直方体で 64%、円柱で 31%、球で 44%であった。今後は生成の精度向上や土木分野で用いられる部材等のより複雑な形状、属性情報の生成が課題として挙げられる。

**Key Words:** Industry Foundation Classes (IFC), generative AI, product model, BIM/CIM

## 1. はじめに

近年、土木分野では生産性の低さが問題となっている。生産性を向上するため、BIM/CIM (Building/Construction Information Modeling, Management) が取り組まれている。BIM/CIM は「建設事業で取扱う情報をデジタル化することにより、調査・測量・設計・施工・維持管理等の建設事業の各段階に携わる受発注者のデータ活用・共有を容易にし、建設事業全体における一連の建設生産・管理システムの効率化を図ること」<sup>1)</sup>と国土交通省は定義している。

BIM/CIM の具体的な取り組みの一部として 3 次元モデルの活用と後工程における属性情報等のデータ活用が掲げられている。そのため、3 次元形状だけでなく、属性情報を持つデータの作成が必要である。ここで、オブジ

ェクト指向に基づいて部材をオブジェクトとして扱い、属性情報を与えることのできる 3 次元データとして 3 次元データモデル<sup>2)</sup>がある。一方で、3 次元データモデルの作成は従来の 2 次元図面の作成とは異なる技術が必要であり、手間や時間がかかることが問題である。

土木分野に限らず、近年では 3 次元形状の作成にかかる手間を削減する方法として、生成 AI (Artificial Intelligence) を用いた 3 次元形状の生成が実施されている。しかし、木や椅子などの一般的な 3 次元形状に留まり、橋脚やボックスカルバートなどの土木分野で利用できる 3 次元形状、また、属性情報を含む 3 次元データモデルの生成への対応は見当たらない。

そこで、本研究では、土木分野の 3 次元データモデルの作成省力化へ向けた、生成 AI を用いた対話的な 3 次元データモデルの作成手法を提案する。3 次元データモ

デルは属性情報を付与することが可能であることから、設計だけでなく、施工や維持管理等の後工程へのデータの引き継ぎに寄与できる。また、本研究は生成 AI による 3 次元データモデルの生成の基礎的な技術の開発を範囲とするため、直方体や円柱、球等の単純な形状のみを対象とする。

## 2. 既往研究

近年では生成 AI のモデルやサービスが広く公開されるようになり、生成 AI の研究開発が盛んにおこなわれている。それに伴い、生成 AI を用いた 3 次元形状の生成に関する研究もおこなわれている。

Siddiqui ら<sup>3)</sup>は生成 AI を用いて 3 次元形状を生成する手法である MeshGPT を開発した。MeshGPT はユーザが文章で指示するのではなく、特定の 3 次元形状についてファインチューニングした生成 AI モデルを用いてより正確な 3 次元形状を生成する。Wang ら<sup>4)</sup>は文章から 3 次元形状を作成する手法である LLaMA-Mesh を開発した。LLaMA-Mesh は大規模言語モデルを 3 次元形状の出力に適するようにファインチューニングすることで、3 次元形状を出力することを実現している。また、LLaMA-Mesh は頂点の座標と面を構成する頂点のインデックスを出力することで、3 次元形状を生成する。

さらに、より産業での利用に適した 3 次元 CAD (Computer-Aided Design) モデルの生成に関する研究もおこなわれている。Khan ら<sup>5)</sup>は文章から 3 次元 CAD モデルを生成する手法を開発した。Khan らの研究では、既存の学習済みモデルのみを使用するのではなく、3 次元 CAD モデルを生成するのに特化したレイヤの追加によりモデルの生成を実現した。一方、Yavartanoo ら<sup>6)</sup>は文章から生成 AI を用いて 3 次元 CAD モデルを生成する際に、1) 文章から透視図を生成し、2) その後、透視図を元に立面図や平面図を生成し、3) これらの立面図や平面図から 3 次元 CAD モデルを生成する手法を開発した。

また、3 次元形状の生成だけではなく、生成 AI を用いて BIM/CIM データを解釈する取り組みも実施されている。Zheng ら<sup>7)</sup>は生成 AI を用いて対話的に BIM データを検索する手法である BIMS-GPT を開発した。BIMS-GPT により、BIM に関する深い知識を持たなくとも BIM が持つ膨大な情報を活用することができる。Lawal ら<sup>8)</sup>は生成 AI による BIM データの解釈を用いた建設分野における契約のプロセスの改善手法を提案した。Lawal らの研究は、手法の開発だけではなく、提案手法の適用による実際の業務プロセスの改善にまで取り組んでいる。

既往研究から、これまで生成 AI による 3 次元形状のみの生成や 3 次元データモデルの解釈について実施され

てきた。一方、生成 AI による属性情報を保持できる形式である 3 次元データモデルの生成については研究がほとんどなされていない。本研究では生成 AI により属性情報を保持できる形式である 3 次元データモデルを生成する。なお、本研究はファイルの生成自体に着目し、属性情報を保持できる形式のファイル生成を対象とするが、属性情報そのものの生成は対象外とする。

生成 AI により 3 次元データモデルの生成ができて、形状と属性情報の両方を持つデータを少ない人手または自動で作成できる。3 次元データモデルに積算や維持管理に必要な情報を属性情報として付与し、自動化施工の入力データとすることで、将来的には計画・設計、自動化施工、維持管理までのライフサイクル全体に渡る省力化、省人化に寄与できると考えられる。

## 3. 生成 AI による 3 次元データモデルの生成

### (1) 提案手法の概要

本研究で提案する手法の概要のフローを図-1 に示す。提案手法は以下の手順で実行する。ここで、「システム」は提案手法をプログラム等を用いて実装したものを指す。

- 1) ユーザが生成したい 3 次元データの形状を指定するプロンプトを入力する。
- 2) システムが入力プロンプトを実行し、ユーザが作成したい形状を推定する。
- 3) システムが参考となる 3 次元データモデルの内容と生成したい形状の寸法とを含んだプロンプトを実行する。
- 4) システムが回答として得た 3 次元データモデルの内容から、3 次元データモデルを作成する。

ここで、提案手法は 3 次元データモデルとして、IFC (Industry Foundation Classes) を使用し、IFC のフォーマットとして、SPF (STEP Physical File) 形式を用いる。

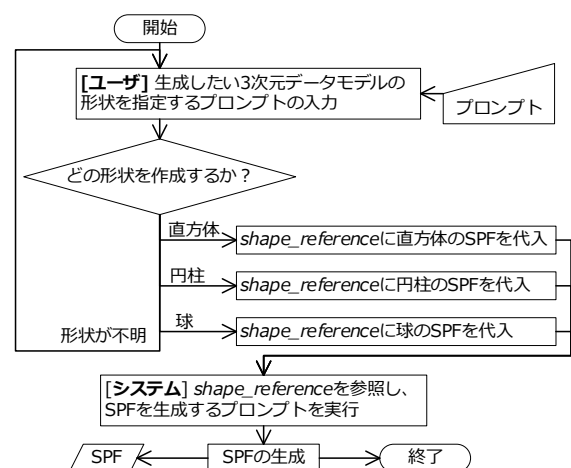


図-1 提案手法の概要のフロー

## (2) IFCについて

IFC は bSI (buildingSMART International<sup>9)</sup>) によって策定されたプロダクトモデルのデータフォーマットである。IFC はファイル交換を行う際の標準フォーマットとして作成されている。日本国内においては BIM/CIM の成果物における構造物のフォーマットとして IFC が指定されており<sup>10)</sup>、広く流通するフォーマットであると言える。

また、IFC をファイルに格納する場合はいくつかのファイル形式とすることが可能である。本研究では、実用上最も広く使用されている<sup>11)</sup>ことから SPF 形式のファイル形式を採用した。なお、SPF は STEP (Standard for the Exchange of Product Data) 形式で内容を記述したファイル形式である。

## (3) One-shot prompting と function calling による生成

提案手法で 3 次元データモデルを生成するフローを図-2 に示す。ここで、本提案手法を実行するプログラムのソースコードをコードと呼ぶこととする。提案手法では、1) ユーザがコードに送信するプロンプト、2) コードが LLM に送信するプロンプトと関数の候補、3) 参考となる SPF の中身を含むコードが作成したプロンプトの 3 つのプロンプトの送受信が行われる。これらの例をそれぞれ図-3、図-4、図-5 に示す。また、提案手法では生成の精度を確保するため、本節で説明する one-shot prompting と function calling を用いている。

提案手法では、テキストデータを扱うことから、生成 AI のモデルの一つである LLM (Large Language Model)

(大規模言語モデル) を用いる。ここで、API で提供されている LLM は、一般的な事項を学習したものが多く、IFC のような利用分野が限定されるフォーマットの学習は不十分であることが多く、ファイル生成が難しいことがある。そのため、提案手法では生成 AI で意図した結果を出力しやすくするための技法であるプロンプトエンジニアリングの一手法である one-shot prompting を用いる。LLM へのプロンプト中に例を複数含ませることにより LLM の回答精度の向上を図る few-shot prompting という手法<sup>12)</sup>がある。One-shot prompting はプロンプトに含ませる例を一つとした、few-shot prompting の特殊な場合の手法である。提案手法は参考となる 3 次元データモデルの内容をそのままプロンプトに記述するため、例が複数になるとプロンプトが長くなり、通信量の増加や動作の大きな遅延が懸念されることから、例を一つとする one-shot prompting を用いた。

また、例の候補が複数ある場合は、作成したい形状と対応する例をプロンプトに含ませる必要がある。そのため、提案手法では function calling と呼ばれる手法を用いる。Function calling は OpenAI 社が開発した LLM で採用されて

いる引数とを推測する技術である<sup>13)</sup>。Function calling を用いることで、複数の形状から適切な例と寸法をプロンプトに付与する関数及び引数を推定することができる。

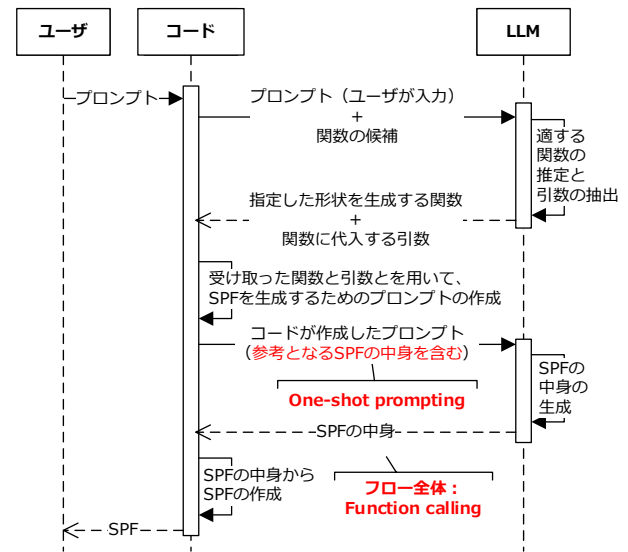


図-2 One-shot prompting と function calling による生成のフロー

幅 700 mm、長さ 300 mm、高さ 500 mm の直方体を作成してください。

図-3 ユーザがコードに送信するプロンプトの例

Don't make assumptions about what values to plug into functions. You are a helpful assistant.

幅 700 mm、長さ 300 mm、高さ 500 mm の直方体を作成してください。

**関数の候補群等**

※**斜体太字**: 実際のプロンプト、返答ではなく、内容を省略したもの。

図-4 コードが LLM に送信するプロンプトと関数の候補の例

Don't make assumptions about what values to plug into functions. You are a helpful assistant.

Please generate an IFC format file of a rectangular cuboid, of which width is 0.7 m, length is 0.3 m, height is 0.5 m, referencing to the following IFC file of a rectangular cuboid, of which width is 1 m, length is 2 m and height is 3 m. Please reply just the content of IFC file. I don't need message like 'Sure, here's the IFC content with the column dimensions you specified.'

**参考とする SPF の中身**

※**斜体太字**: 実際のプロンプト、返答ではなく、内容を省略したもの。

図-5 参考となる SPF の中身を含むプロンプト

## 4. 検証実験

### (1) 実験方法

検証実験では、単純な形状として直方体、円柱、球を選定して3次元データモデルの生成をおこなう。検証実験のフローを図-6に示す。検証実験は主に下記の手順で実施した。形状については、土木分野での提案手法の活用を想定し、単純化した橋脚や梁、壁等で見られる矩形断面の押し出しで表現した直方体、単純化した鉄筋で見られる円形断面の押し出しで表現した円柱、断面の押し出しでは表現できない球を対象とした。また、サンプルサイズについては、検証コストと一定のサンプルサイズを確保するという観点から、各形状で100ファイルとした。

- 1) 提案手法による SPF の生成（直方体、円柱、球各100ファイル、計300ファイル）
- 2) bSI が提供する SPF 形式のバリデーションサービス（bSI IFC Validation Service<sup>14)</sup>）を用いた SPF の検証
- 3) 寸法の抽出による、プロンプトで指定した形状が作成できたかの検証
- 4) 提案手法による SPF の生成の正答率の算出

検証実験に用いたツール等を表-1に示す。検証実験では、アプリケーションとしての実装のしやすさの観点および SLA（Service Level Agreement）が設定されていることやデータガバナンス等のセキュリティの観点から、基盤として Azure OpenAI Service を、基盤上で動作する LLM として検証実験時に基盤で提供されるリリース版のモデルで最新であった GPT-4o を使用した。

なお、検証実験では提案手法の精度を定量的に示すため、提案手法の精度を算出した。生成した3次元データモデルが以下 a, b の条件を全て満たす場合を正答とした。この条件 a, b を用いた精度の定義を式(1)に示す。

- a) bSI IFC Validation Service で表-2の全ての項目を合格している。
- b) 生成したい形状の寸法と同じ寸法である。

$$Accuracy = \frac{N_{anb}}{N_{total}} \quad (1)$$

Where: Accuracy 精度

$N_{anb}$  条件 a と b の両方を満たす3次元データモデルの数

$N_{total}$  生成した3次元データモデルの数

### (2) 検証実験に用いたデータ

提案手法は one-shot prompting を実行する際に参考となる SPF をプロンプトに内包する。生成時の参考としてプロンプトに内包させた形状を図-7に、IFC の内容を図-8、図-9、図-10に示す。なお、出力結果において変更され

るべき寸法を決定するインスタンスを太字斜体で示した。

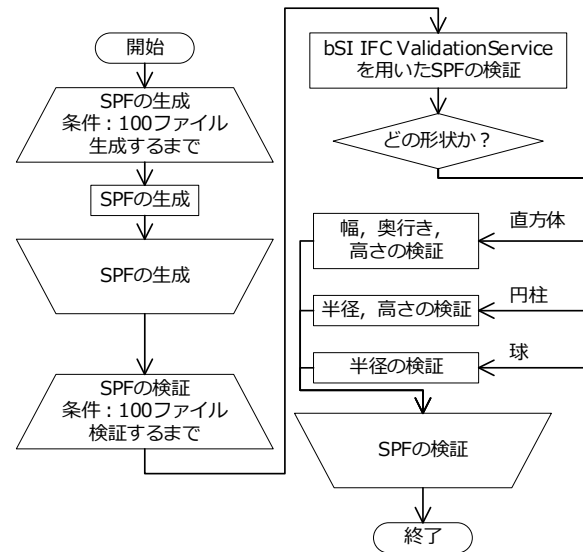


図-6 検証実験のフロー

表-1 検証実験に用いたツール等

種類	名称	バージョン
基盤	Azure OpenAI Service	-
LLM	GPT-4o	2024-06-01
参考 SPF の作成	IfcOpenShell	0.7
IFC ビューア	Open IFC Viewer	25.9.0
SPF の検証	buildingSMART IFC Validation Service	0.6.7

表-2 buildingSMART IFC Validation Service の検証項目と内容

項目名	検証内容
STEP Syntax	SPF として文法が正しいか。
IFC Schema	IFC のスキーマに沿っているか。
Normative IFC Rules	その他の公式でない IFC のルールに沿っているか。
Industry Practices	業界の慣行に沿っているか。

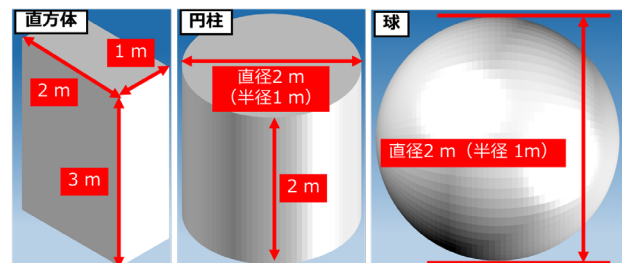


図-7 SPF 生成時の参考として与えた IFC の形状

検証実験ではユーザが形状を指定するプロンプトを入力する。検証実験でユーザが入力したプロンプトを表-3に示す。なお、今回の検証実験では、省力化のため、ユーザによるプロンプトの入力を人手ではなくスクリプトを用いた自動入力により実施した。また、検証条件が寸法に依存し、精度の基準も寸法に依存することを防ぐため、同一形状は表-3に示す寸法のみで生成することとした。

表-3 検証実験でユーザが入力したプロンプト

形状	ユーザの入力プロンプト
直方体	幅 400 mm、長さ 200 mm、高さ 6000 mm の直方体を作成してください。
円柱	底面の半径が 400 mm、高さが 2500 mm の円柱を作成してください。
球	半径が 300 mm の球を作成してください。

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('ViewDefinition [CoordinationView]'),'2;1');
FILE_NAME('','2024-10-24T15:12:08',(),(), 'IfcOpenShell v0.7.0-c7830e9e5', 'IfcOpenShell v0.7.0-c7830e9e5', '');
FILE_SCHEMA(('IFC4X3_ADD2'));
ENDSEC;
DATA;
#1=IFCPROJECT('1HXR3IHGP54e9wJvUD7wY6', $, 'My Project', $, $, $, $, (#10), #5);
#2=IFCSIUNIT(*, .LENGTHUNIT., .MILLI., .METRE.);
#3=IFCSIUNIT(*, .AREAUNIT., $, .SQUARE_METRE.);
#4=IFCSIUNIT(*, .VOLUMEUNIT., $, .CUBIC_METRE.);
#5=IFCUNITASSIGNMENT((#4, #2, #3));
#6=IFCCARTESIANPOINT((0., 0., 0.));
#7=IFCDIRECTION((0., 0., 1.));
#8=IFCDIRECTION((1., 0., 0.));
#9=IFCAXIS2PLACEMENT3D(#6, #7, #8);
#10=IFCGEOMETRICREPRESENTATIONCONTEXT($, 'Model', 3, 1.E-05, #9, $);
#11=IFCGEOMETRICREPRESENTATIONSUBCONTEXT('Body', 'Model', *, *, *, #10, $, .MODEL_VIEW., $);
#12=IFCSITE('39eoThiAz5qe4CnPSIYPMa', $, 'My Site', $, $, $, $, $, $, $, $);
#13=IFCBUILDING('30lpAs1zj7k003YwDrwRH2', $, 'Building A', $, $, $, $, $, $, $, $);
#14=IFCRELAGGREGATES('0agyY$gnvAbhBnHdLFLGSY', $, $, $, #1, (#12));
#15=IFCRELAGGREGATES('2KctED2LzEggHl9s$zs5Vv', $, $, $, #12, (#13));
#16=IFCBUILTELEMENT('1Vx1qHCnf53gDB1kGa00he', $, 'My Rectangular Cuboid', $, $, #36, #30, $);
#22=IFCRECTANGLEPROFILEDEF(.AREA., '1000x2000', $, 1000., 2000.);
#23=IFCCARTESIANPOINT((0., 0., 0.));
#24=IFCDIRECTION((0., 0., 1.));
#25=IFCDIRECTION((1., 0., 0.));
#26=IFCAXIS2PLACEMENT3D(#23, #24, #25);
#27=IFCDIRECTION((0., 0., 1.));
#28=IFCEXTRUDEDAREASOLID(#22, #26, #27, 3000.);
#29=IFCSHAPEREPRESENTATION(#11, 'Body', 'SweptSolid', (#28));
#30=IFCPRODUCTDEFINITIONSHAPE($, $, (#29));
#31=IFCRELCONTAINEDINSPATIALSTRUCTURE('0gxhKnj856ohqtt1Ayy6Ri', $, $, $, (#16), #13);
#32=IFCCARTESIANPOINT((0., 0., 0.));
#33=IFCDIRECTION((0., 0., 1.));
#34=IFCDIRECTION((1., 0., 0.));
#35=IFCAXIS2PLACEMENT3D(#32, #33, #34);
#36=IFCLOCALPLACEMENT($, #35);
ENDSEC;
END-ISO-10303-21;
```

図-8 SPF生成時の参考として与えた直方体の IFC の内容（太字斜体：寸法を決定するインスタンス）

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('ViewDefinition [CoordinationView]'),'2;1');
FILE_NAME('','2024-10-24T15:27:44',(),(), 'IfcOpenShell v0.7.0-
c7830e9e5', 'IfcOpenShell v0.7.0-c7830e9e5', '');
FILE_SCHEMA(('IFC4X3_ADD2'));
ENDSEC;
DATA;
#1=IFCPROJECT('2R501Z1NP2cQ4fY8PCyK9e',$, 'My Project',$,$,$,$, (#10), #5);
#2=IFCSIUNIT(*, .LENGTHUNIT., .MILLI., .METRE.);
#3=IFCSIUNIT(*, .AREAUNIT., $, .SQUARE_METRE.);
#4=IFCSIUNIT(*, .VOLUMEUNIT., $, .CUBIC_METRE.);
#5=IFCUNITASSIGNMENT((#3, #4, #2));
#6=IFCCARTESIANPOINT((0., 0., 0.));
#7=IFCDIRECTION((0., 0., 1.));
#8=IFCDIRECTION((1., 0., 0.));
#9=IFCAXIS2PLACEMENT3D(#6, #7, #8);
#10=IFCGEOMETRICREPRESENTATIONCONTEXT($, 'Model', 3, 1.E-05, #9, $);
#11=IFCGEOMETRICREPRESENTATIONSUBCONTEXT('Body', 'Model', *, *, *, *, #10, $, .MODE
L_VIEW., $);
#12=IFCSITE('0lgxROwJX0$x9ERNq0nmV0',$, 'My Site',$,$,$,$,$,$,$,$,$,$);
#13=IFCBUILDING('027J2xsn1C0A69KBVjYavq',$, 'Building A',$,$,$,$,$,$,$,$,$);
#14=IFCRELAGGREGATES('01byBf04vAnxgw$P8p1Qvi',$,$,$, #1, (#12));
#15=IFCRELAGGREGATES('0kZ06f44f5LuadyLpYcleA',$,$,$, #12, (#13));
#16=IFCBUILTELEMENT('2cUT06zkPCz9CYBKLH1Knz',$, 'My
Cylinder',$,$, #36, #30, $);
#22=IFCCIRCLEPROFILEDEF(.AREA., '1000',$, 1000.);
#23=IFCCARTESIANPOINT((0., 0., 0.));
#24=IFCDIRECTION((0., 0., 1.));
#25=IFCDIRECTION((1., 0., 0.));
#26=IFCAXIS2PLACEMENT3D(#23, #24, #25);
#27=IFCDIRECTION((0., 0., 1.));
#28=IFCEXTRUDEDAREASOLID(#22, #26, #27, 2000.);
#29=IFCSHAPEREPRESENTATION(#11, 'Body', 'SweptSolid', (#28));
#30=IFCPRODUCTDEFINITIONSHAPE($, $, (#29));
#31=IFCRELCONTAINEDINSPATIALSTRUCTURE('2s_P1_zYvCNPf0cJsjtbiy',$,$,$, (#16),
#13);
#32=IFCCARTESIANPOINT((0., 0., 0.));
#33=IFCDIRECTION((0., 0., 1.));
#34=IFCDIRECTION((1., 0., 0.));
#35=IFCAXIS2PLACEMENT3D(#32, #33, #34);
#36=IFCLOCALPLACEMENT($, #35);
ENDSEC;
END-ISO-10303-21;
```

図9 SPF生成時の参考として与えた円柱のIFCの内容（太字斜体：寸法を決定するインスタンス）





### (3) 実験結果

生成に成功した 3 次元データモデルを図-11 に示す。また、生成に成功した 3 次元データモデルの SPF の抜粋を図-12、図-13 および図-14 に示す。これらの形状および SPF の内容から、ユーザが入力したプロンプトで指定した形状の 3 次元データモデルを生成することができることを確認した。

各形状の精度を表-4 に示す。直方体において最も高い 64%の精度が得られ、円柱において最も低い 31%の精度が得られた。

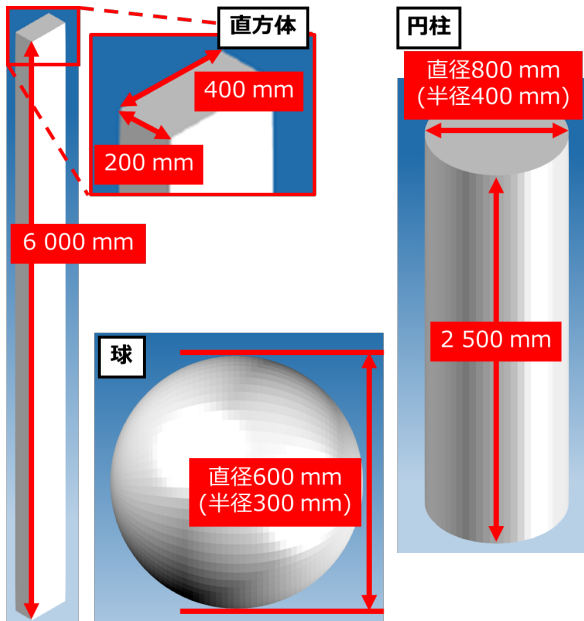


図-11 生成したに成功した 3 次元データモデル

```
ISO-10303-21;
.....省略.....
FILE_SCHEMA(('IFC4X3_ADD2'));
.....省略.....
#15=IFCRELAGGREGATES('2KctED2LzEggHl9s$zs5Vv',
,$,$,$,#12,(#13));
#16=IFCBUILTELEMENT('1Vx1qHCnf53gDB1kGa00he',
,$,'My Rectangular Cuboid',,$,$,#36,#30,$);
#22=IFCRECTANGLEPROFILEDEF(.AREA., '400x200',
,$,400.,200.);
#23=IFCCARTESIANPOINT((0.,0.,0.));
#24=IFCDIRECTION((0.,0.,1.));
#25=IFCDIRECTION((1.,0.,0.));
#26=IFCAXIS2PLACEMENT3D(#23,#24,#25);
#27=IFCDIRECTION((0.,0.,1.));
#28=IFCEXTRUDEDAREASOLID(#22,#26,#27,6000.);
#29=IFCSHAPEREPRESENTATION(#11,'Body','Swept
Solid',(#28));
.....省略.....
END-ISO-10303-21;
```

図-12 生成に成功した直方体一例の抜粋 (SPF) (太字斜体: 寸法を決定するインスタンス)

```
ISO-10303-21;
.....省略.....
FILE_SCHEMA(('IFC4X3_ADD2'));
.....省略.....
#14=IFCRELAGGREGATES('01byBf04vAnxgw$P8p1Qvi',
,$,$,$,#1,(#12));
#15=IFCRELAGGREGATES('0kZ06f44f5LuadyLpYcleA',
,$,$,$,#12,(#13));
#16=IFCBUILTELEMENT('2cUT06zkPCz9CYBKLH1Knz',
,$,'My Cylinder',,$,$,#36,#30,$);
#22=IFCCIRCLEPROFILEDEF(.AREA., '400',,$,400.)
;
.....省略.....
#28=IFCEXTRUDEDAREASOLID(#22,#26,#27,2500.);
#29=IFCSHAPEREPRESENTATION(#11,'Body','Swept
Solid',(#28));
#30=IFCPRODUCTDEFINITIONSHAPE($,$,(#29));
#31=IFCRELCONTAINEDINSPATIALSTRUCTURE('0gxhK
nj856ohqtt1Ayy6Ri',,$,$,$,(#16),#13);
#32=IFCCARTESIANPOINT((0.,0.,0.));
.....省略.....
END-ISO-10303-21;
```

図-13 生成に成功した円柱の一例の抜粋 (SPF) (太字斜体: 寸法を決定するインスタンス)

```
ISO-10303-21;
HEADER;
.....省略.....
FILE_SCHEMA(('IFC4X3_ADD2'));
.....省略.....
#22=IFCCARTESIANPOINT((0.,0.,0.));
#23=IFCDIRECTION((0.,0.,1.));
#24=IFCDIRECTION((1.,0.,0.));
#25=IFCAXIS2PLACEMENT3D(#22,#23,#24);
#26=IFCSPHERE(#25,300.);
#27=IFCCSGSOLID(#26);
#28=IFCSHAPEREPRESENTATION(#11,'Body','CSG',
(#27));
#29=IFCPRODUCTDEFINITIONSHAPE($,$,(#28));
#30=IFCRELCONTAINEDINSPATIALSTRUCTURE('0KN_v
xuSjBBEjQq9M2tUzV',,$,$,$,(#16),#13);
#31=IFCCARTESIANPOINT((0.,0.,0.));
.....省略.....
END-ISO-10303-21;
```

図-14 生成に成功した球の一例の抜粋 (SPF) (太字斜体: 寸法を決定するインスタンス)

表-4 各形状の精度

形状	精度 (n=100)
直方体	64%
円柱	31%
球	44%



#### (4) 実験結果の精査

各形状について生成したファイルが不正解であると判断された段階の割合を図-15に示す。なお、検証ではbSI IFC Validation Service で合格したファイルに対して寸法の検証を実施した。図-15 より、各形状で、約 90%以上のファイルがbSI IFC Validation Serviceの段階で不正解であると判断されていることを確認した。

また、bSI IFC Validation Serviceの各項目で合格したファイルの数を調査した(表-5)。表-5 より、各形状でSTEP Syntax を合格したファイルは 95%以上であるが、IFC Schema を合格したファイルは 70%に満たないことを確認した。

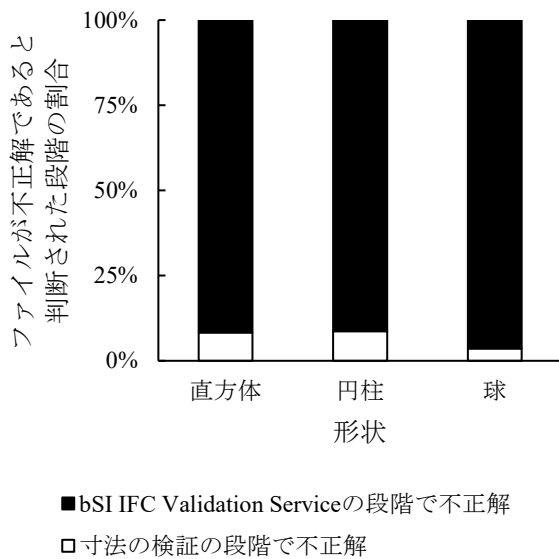


図-15 ファイルが不正解であると判断された段階の割合

表-5 bSI IFC Validation Service を合格したファイルの数 (各形状で n=100)

形状	検証項目			
	STEP Syntax	IFC Schema	Normative IFC Rules	Industry Practices
直方体	98	67	97	97
円柱	98	37	98	98
球	96	47	94	96

## 5. 考察

### (1) 提案手法の精度

形状ごとの精度については、直方体が最も精度が高く、円柱が最も精度が低かった。直方体が最も精度が高かった理由としては、直方体が建設分野の部材において幅広

く用いられている形状であることが考えられる。直方体は各寸法の比率の違いはあるが、梁、柱、壁、床等の建設分野で構造物の躯体等を表す際に頻繁に用いられる形状である。そのため、直方体は使用した大規模言語モデルで多く学習されていると推察され、正しいファイルの内容を生成することが他の形状と比較して容易であったと考えられる。また、円柱の精度が最も低かった理由としては、直方体ほど頻繁に用いられる形状ではなく、かつ、球と比較して変更すべきパラメータが多いことが考えられる。球は半径のみのパラメータで表現できるが、円柱は断面の半径と断面の押し出し距離の2つのパラメータを用いて表現する必要がある。

全形状について考えると、提案手法は精度が最高で64%、最低で31%であり、現段階では実用には困難であると考えられる。そこで、精度向上のため、正答率の低さの原因を調査した。図-15 より、すべての形状において、90%以上のファイルがbSI IFC Validation Serviceの段階で不正解であると見なされている。このことから、不正解である場合は、IFC 形式として正しくないファイルが生成されていることが大きな原因であると考えられる。

また、表-5 より、いずれの形状においても、95%以上のファイルはSTEP Syntax を合格したが、他の項目と比較してIFC Schema は合格率が低かったことがわかる。そのため、STEP 形式のファイルとしての体裁は成しているがIFC としては不正であるファイルが多く生成されたことが提案手法の精度の低さの原因であると考えられる。STEP 形式は土木分野を含む建設分野だけではなく、製造分野でも使用されている形式である<sup>19)</sup>ことから、使用した生成AIモデルはSTEP形式の文法を十分に学習していたと考えられる。一方、IFC は土木分野を含む建設分野に特化した形式であるため、STEP としては正しいがIFC としては不正であるファイルが多く生成されたと考えられる。

提案手法の精度は、先に述べたように、現段階では実用が困難な程度であった。本研究で生成の対象とするIFC は形状と属性情報の両方を扱うことができ、BIM/CIMでのデータ共有においてベンダに依存しないファイル形式である。そのため、IFC を正しく自動的に生成できることは、各事業段階、関係者間でのデータ交換を容易にし、土木分野の生産性の向上に繋がると考えられる。

このことから、今後はIFC への適応が課題となる。具体的な方法としては、ファインチューニング、AI エージェントの使用が挙げられる。

3次元形状を生成するというタスクにおいてファインチューニングを用いることは既往研究<sup>23) 4)</sup>でも実施されている。そこで、LLM 自体の精度を高めるためにファインチューニングを本手法に適用することが、3次元デ

ータモデルの生成精度の向上に寄与できると考えられる。また、ファインチューニングにおいては、生成 AI モデルそのものの IFC への適応の程度が向上されることが期待されるが、土木構造物を対象とした IFC 形式のファイルが広く流通し、多くの学習データが確保できることが必要となる。

また、AI エージェントは、LLM を活用し、自律的に計画を策定し、その計画を実行に移すことで、与えられた目標を達成するための行動を繰り返すことができるものである<sup>10)</sup>。ここで、AI エージェントを用いて 3 次元データモデルを生成するシステムを開発する際は、システムで用いる AI エージェントの数、各 AI エージェントが持つ役割、AI エージェント間のネットワーク構造や AI エージェント間の遷移の条件等について考える必要がある。特に、土木分野に適した 3 次元データモデルを生成するという目標は非常に専門的であることから、目標の達成に必要な AI エージェントの粒度や担うべきタスクについて詳細な調査が必要であると考えられる。

## (2) 形状

本研究は単純な形状を対象として 3 次元データモデルを生成した。実際の業務への適用を考慮すると、部材の形状を表現できるような、より複雑な形状への対応が必要となる。加えて、土木構造物は複数の部材から構成されることから、単一の形状やモデルのみではなく、複数の形状やモデルを含む 3 次元データモデルの生成が必要となる。

そのため、(1)で述べた 3 次元モデルの生成精度の向上に加えて、部材や複数の部材からなる 3 次元データモデルの生成手法を開発する必要があると考えられる。そのためには、より多くの土木構造物の 3 次元データモデルを参考となるファイルとして用意することや各部材等の IFC のクラスへのマッピングが課題となる。

より多くの土木構造物の 3 次元データモデルを参考となるファイルとして用意することに関しては、IFC の流通量に大きく依存することから技術的な解決が難しいが、各部材等の IFC のクラスへのマッピングの方法およびオブジェクトの粒度については技術的な解決策があると考えられる。各部材等の IFC のクラスへのマッピングについては、土木構造物の 3 次元データをモデリングする際に形状モデリングだけではなく、後工程に必要なデータを考慮し、各部材等のクラスへのマッピングや属性情報等のデータモデリングもおこなうことで解決できると考えられる。そのためには、共通データ環境 (CDE (Common Data Environment)) の利用方法やより有効な活用のための実装方法に関して明らかにする必要がある。

## (3) 属性情報

また、本研究は 3 次元データモデルの形状の再現に着目した。そのため、生成した 3 次元データモデルは属性情報を持つことができる形式であるが、生成の対象としたのは 3 次元形状のみであり、属性情報は対象とはしなかった。BIM/CIM の具体的な取り組みに後工程における属性情報等のデータ活用が含まれていることから、将来的な実用性を考慮すると、属性情報の取り扱い必須である。

そのため、今後は生成 AI による形状以外の属性情報の付与についても検討する必要がある。なお、実用性を確保する観点から、様々な土木構造物を対象として、事業段階間のデータの受け渡しにおいて必要となる属性情報の種類を精査する必要がある。

## 6. おわりに

本研究では、土木分野の 3 次元データモデルの作成を省力化するため、単純な形状を対象として生成 AI を用いた対話的な 3 次元データモデルの作成手法を提案した。なお、本研究では、直方体、円柱、球を対象とした。その後、検証実験として、提案手法を用いて 3 次元データモデルを生成し、ユーザがプロンプトで指示した形状の 3 次元データモデルが生成できることを確認した。また、提案手法による 3 次元データモデルの生成の精度を算出し、直方体で 64%、円柱で 31%、球で 44%の精度を得た。

また、本研究は形状の生成に着目し、単純な形状の生成を実施した。そのため、今後は生成の精度向上や土木分野で用いられる部材等のより複雑な形状の生成、属性情報を含む 3 次元データモデルの生成が課題として挙げられる。

## 参考文献

- 1) 国土交通省：直轄土木業務・工事における BIM/CIM 適用に関する実施方針、<<https://www.mlit.go.jp/tec/content/001733473.pdf>>、(入手 2024.12.11)。
- 2) 日本建設情報技術センター、矢吹信喜、藤澤泰雄：BIM/CIM 概論，pp. 10-11，建設物価調査会，2024。
- 3) Siddiqui, Y., Alliegro, A., Artemov, A., Tommasi, T., Sirigatti, D., Rosov, V., Dai, A., and Nießner, M.: MeshGPT: Generating triangle meshes with decoder-only transformers, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19615-19625, 2024.
- 4) Wang, Z., Lorraine, J., Wang, Y., Su, H., Zhu, J., Fidler, S., and Zeng, X.: LLaMA-Mesh: Unifying 3D mesh generation with language models, *arXiv*, 2024.
- 5) Khan, M. S., Sinha, S., Sheikh, T. U., Stricker, D., Ali, S. A., and Afzal, M. Z.: Text2CAD: Generating sequential

- CAD models from beginner-to-expert level text prompts, *38th Conference on Neural Information Processing Systems*, pp. 1-28, 2024.
- 6) Yavartanoo, M., Hong, S., Neshatavar, R., and Lee, K. M.: Text2CAD: Text to 3D CAD generation via technical drawings, *arXiv*, 2024.
  - 7) Zheng, J. and Fischer, M.: Dynamic prompt-based virtual assistant framework for BIM information search, *Automation in Construction*, Vol. 155, 105067, 2023.
  - 8) Lawal, O. O., Nawari, N. O., and Alsaffar, A.: AI-enabled smart contracts in Building Information Modelling (BIM) for unified project execution: *A Theoretical Framework. Proceedings of the 41st International Conference of CIB W78*, 2024.
  - 9) buildingSMART International: buildingSMART International, <<https://www.buildingsmart.org/>>, (accessed 2024.12.19).
  - 10) 国土交通省 : BIM/CIM モデル等電子納品要領 (案) 及び同解説, <<https://www.mlit.go.jp/tec/content/001472866.pdf>>, (入手 2024.12.19) .
  - 11) buildingSMART International: IFC Formats - buildingSMART Technical, <<https://technical.buildingsmart.org/standards/ifc/ifc-formats/>>, (accessed 2024.12.19).
  - 12) Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D.: Language models are few-shot learners, *NIPS'20: Proceedings of the 34th International Conference on Neural Information Processing Systems*, pp. 1877-1901, 2020.
  - 13) OpenAI: Function Calling - OpenAI API, <<https://platform.openai.com/docs/guides/function-calling>>, (accessed 2024.12.24).
  - 14) BuildingSMART International: IFC Validation Service - buildingSMART International, <<https://www.buildingsmart.org/users/services/ifc-validation-service/>>, (accessed 2024.12.27).
  - 15) 有賀貴志, 矢吹信喜, 新井泰 : 変状データを含む開削トンネルのプロダクトモデルの構築, 土木学会論文集 F3 (土木情報学), 68 巻 1 号, pp. 58-70, 2012.
  - 16) Masterman, T., Besen, S., Sawtell, M., and Chao, A.: The landscape of emerging AI agent architectures for reasoning, planning, and tool calling: A survey, *arXiv*, 2024.

(Received January 15, 2025)

(Accepted February 28, 2025)

## Generating 3D data model for civil engineering using a generative AI

Atsuhiko YAMAMOTO, Riku OGATA, Junichiro FUJII, Kazuhiro YAMAMOTO

The civil engineering industry faces low productivity issues; hence, it is adopting Building/Construction Information Modeling, Management (BIM/CIM) to improve productivity in this domain. BIM/CIM utilizes 3D data models to streamline whole construction projects, design construction, operation and maintenance (O&M), and demolition processes, leading to enhanced efficiency and quality of construction projects. However, creating these 3D models by hand is time-consuming and labor-intensive. This issue is particularly true for infrastructures that engineers have been designing based on 2D data for a long time. This study proposes an interactive method for generating 3D data models using generative artificial intelligence (AI) to address this issue. Using the proposed method, engineers provide instructions in natural language and generate 3D data models automatically. The proposed method reduces the burden of engineers operating 3D modeling software or inputting complex parameters, resulting in labor savings and time reduction. This study focused on Industry Foundation Classes (IFC) 4.3 and conducted experiments to generate data models of simple shapes (rectangular cuboid, cylinder, and sphere). The proposed method utilizes one-shot prompting to generate 3D models collectively because IFC is a particular professional file format. This study evaluated the accuracy of the proposed method based on the ratio of the number of collectively generated models to the total number of generated models. The results show an accuracy of 64% for rectangular cuboids, 31% for cylinders, and 44% for spheres. The proposed method is currently limited to low accuracy and generating simple shapes. However, future development aims to improve accuracy using fine-tuning and AI agents and support complex shapes used in the civil engineering industry, such as beams, columns, and footing, as well as attributes. The proposed generative AI-based 3D data model generation method is expected to accelerate BIM/CIM adoption in the civil engineering industry, significantly contributing to productivity improvement.