

大規模言語モデルを用いた 1DCAE のモデリング作業の支援

○池田 孟^{*1}, 石橋 祐太郎^{*2}, 金丸 正寛^{*3}, 小林 孝^{*1}

Making research paper (About the use of the JSME specification template file)

Hajime IKEDA^{*1}, Yutaro ISHIBASHI^{*2}, Masahiro KANAMARU^{*3} and Takashi KOBAYASHI:w

^{*1}

^{*1} Japan Society of Mechanical Engineering
35Shinanomachi, Shinjuku-ku Tokyo 160-0016, Japan

^{*2,*3,*1} Department of Mechanical Engineering, Kikai University
1 Kogakumachi, Shinjuku-ku, Tokyo 160-0001, Japan

Received: XXXX; Revised: XXXX; Accepted: XXXX

Abstract

In complex system design, where mechanical, electrical, and control subsystems interact, 1D CAE is increasingly adopted to optimize products quickly by verifying requirements through early-stage physical simulations on virtual prototypes. The Modelica language has become the standard for modeling these systems due to its object-oriented, declarative nature and ability to handle multi-domain interactions. It supports hierarchical modeling using extensive libraries like the Modelica Standard Library, which includes components for thermal, electromagnetic, and control domains. Recent advancements in generative AI, particularly Large Language Models (LLMs), have enabled new approaches in design engineering, such as automating design system handling and managing design knowledge. This study presents a framework that uses Retrieval-Augmented Generation (RAG) to search closed corporate Modelica repositories and employs three collaborative agents—a planner, coder, and tester—to automatically assemble libraries. Experiments using OpenAI-4o demonstrated that this agent-based workflow could generate candidate code for tasks that RAG alone could not handle. The results also showed improved code generation performance when working with proprietary internal libraries, highlighting the potential of combining LLMs with structured agent collaboration in engineering design automation.

Keywords : Term1, Term2, Term3, Term4, ...(Show five to ten key words)

1. 緒 言

複雑なシステムの設計においては、機械、電気、制御の各サブシステムが相互に干渉する複雑な製品を短時間で最適化するため、設計初期から物理シミュレーションを活用して仮想プロトタイプ上で要求充足性を検証する 1DCAE が主流となりつつある(日本機械学会, 2020)。物理シミュレーションモデルの記述には、オブジェクト指向かつ宣言型で多領域連成を自然に扱える Modelica 言語が事実上の標準となっている。Modelica Standard Libraryをはじめ、熱流体、電磁気、制御系など多様なドメインのサブモデルがライブラリとして公開されており、ユーザはライブラリから既存コンポーネントをインポートして階層的にモデルを構築できる。設計工学では、生成 AI を用いて設計知識を管理し、設計システムの取り扱いを自動化する試みが行われてきた(藤田喜久雄, 2023)。ここ最近では、生成 AI の中でも大規模言語モデル Large Language Model (LLM)を中心とした生成 AI の性能は大きく向上し、さまざまな設計問題への応用が検討されてきている。本研究は、企業内でクローズされた 1DCAE のライブラリの作業支援・読解支援に LLM を適用した。企業内でクローズされた Modelica のライブラリを対象

[No.25-10] 日本機械学会 第 35 回設計工学・システム部門講演会講演論文集 [2025.9.3.-5 明治大学アカデミーコモン (東京都千代田区)]

^{*1}正員, 三菱電機株式会社設計開発技術センター (〒661-8661 兵庫県尼崎市塚口本町 8-1-1)

^{*2}, 三菱電機株式会社生産技術センター 〒661-8661 兵庫県尼崎市塚口本町 8-1-1

^{*3}三菱電機株式会社設計開発技術センター (〒661-8661 兵庫県尼崎市塚口本町 8-1-1)

E-mail of corresponding author: taro@jsme.or.jp

に RAG で関連ドキュメントを検索し、プランナー、コーダ、テスターの三つのエージェントが協調してライブラリを自動組立するフレームワークを構築した事例を報告する。OpenAI 4o-mini を用いた実験では RAG だけのシステムでは、生成できなかったコード作成タスクにおいて候補コードが生成できることを確認した。企業内のクローズなライブラリの操作においてもエージェントワークフローによってコード生成能力が向上することがわかった。

2. 先行研究

2.1 LLM の設計の展開事例

2024 年以降に報告された最近の設計問題の LLM を適用した事例を紹介する。現在では、設計知識の管理、作業の自動化、情報の共有といった設計問題に LLM が使われるようになってきている。Picard らは、概念設計から製造・検査に至る四つの設計フェーズを対象として、マルチモーダル Vision-Language Model (VLM) である GPT-4V と LLaVA 1.6 34B の有効性を体系的に評価した(Picard et al., 2024)。同研究は VLM が生成した回答根拠を設計図や仕様書にトレースする手法を提案し、設計者が結果を検証しやすい解釈性向上の方向性を示した。

トヨタ自動車は、熟練エンジニアの暗黙知を LLM にリンクさせて次世代車両の開発速度を高める目的で、マルチエージェント型の設計支援基盤を構築している(Onishi, 2024)。同社では、パワートレイン設計部門において LLM で構成されたエージェントが設計資料の探索を加速する事例が報告された。三菱電機は、設計・生産準備を含む ECM (Engineering Chain Management) 領域に生成 AI を組み込む社内プラットフォームを構築し、2017 年以降の AI 活用技術を集約したソリューション群を展開している(玉谷基亮, 2024)。同社は技術文書のレイアウト解析と差分照合を行う自然言語処理 AI、FMEA/DRBFM データを参照して新規設計変更リスクを自動抽出するリスク抽出 AI、CAE 解析結果を学習したサロゲートモデル AIなどを設計フローに導入し、設計パラメーター探索の高速化と知見の再利用を図った。さらに、デジタルツインと生成 AI エージェントを連携させ、IoT で収集した実機データと仮想空間のシミュレーション結果を横断的に解析しながら設計条件を最適化する将来像を提案している。企業内では、社内でクローズされた情報を検索し設計活動に用いる。先行研究のとおり、機密情報を RAG などの検索システムを構築し、設計知識管理や情報の共有に LLM が活用されている。

2.1.1 物理シミュレーションの支援

本研究では Modelica 言語によって作成されたライブラリによるシミュレーションのモデリング作業の支援を検討している。Modelica に限らず、物理シミュレーションは既存のライブラリを操作し、モデリングを行うことが多い。ここでは、LLM を物理シミュレーションのライブラリ操作に適用事例を記す。近年の大規模言語モデル LLM はコード生成力の向上により、Modelica のコンポーネント自動生成への応用が進んでいる。Xiang(Xiang et al., 2025)は、Modelica のコンポーネントを作成する LLM エージェント「ModiGen」を構築した。「ModiGen」は、構築したベンチマーク問題に対して、1 度の指示でコード完成をおこなうことができる指標で 0.33 以上を達成する事例が報告されている。複雑なモデル構造でも一定の妥当性を保持する段階に到達した。

数値流体シミュレーションでよく使われる OpenFOAM は、C++言語で作成されたライブラリである。Pandy (P et al.,)は、OpenFOAM を用いた CFD シミュレーションの自動化を目的として、GPT-4o および CoT 機能搭載モデルを基盤とするエージェント「OpenFOAMGPT」が提案している。本手法では、OpenFOAMチュートリアルケースから得た領域知識を検索拡張生成 (RAG) パイプラインに組み込み、入力ファイルの自動生成・実行・エラー訂正を反復的に行う構造を採用している。評価では、RAG を用いた少数ショットプロンプトにより、境界条件やメッシュ解像度の変更に対応できることが示された。一方、精度担保のための人手監視やモデル性能の時間変動が課題として指摘されている。Chen(Chen et al., 2024),(Chen et al., 2025)は、ソフトウェア開発エージェント MetaGPT を OpenFOAM に適用した「MetaOpenFOAM」は、Architect, InputWriter, Runner, Reviewer の四つのエージェントが連携し、ユーザーの自然言語要求から入力ファイル生成、シミュレーション実行、エラー分析・修正までを自動化する反復ループを構築する。

これらの先行研究では LLM による完全自動化を試行しているが、いまだに LLM によるシミュレーションの完全自動化は難しい。表 1 に完全自動化が困難な理由を列挙する。

Table 1 list of difficulties in applying LLM to simulation modeling

要因	備考
高度なドメイン知識の不足	LLM はインターネット上の広範囲のデータを用いて訓練されているが、Modelica の特定のライブラリや OpenFOAM のファイル構造やパラメータ設定や各種コマンドのなどの情報をインターネット上の汎用的な訓練データから学習することは、難しい。
信頼性と正確性	シミュレーション作業のセットアップは、一般的に厳密な正確性が要求される。モデリング作業は、複数のセットアップ作業を正確に実行する必要があるが、すべての作業を正確にじっこうすることは現在の LLM にとっては、難しい。
ライブラリ機密性	専門ライブラリは、機密性が高く、インターネットには公開されている活用例は少ない。LLM の訓練データにはいっさい含まれていないため、LLM に専門ライブラリの情報を与えるために、RAG によって検索結果を LLM に与えることが一般的である専門ライブラリの活用は、一企業内だけにとどまることがほとんどのため、ファインチューニングに必要な訓練データを用意することも難しい。
ベンチマーク問題の作成	LLM の回答精度を向上させるためには、システムプロンプトの工夫やパラメータのチューニングが必要である。専門ライブラリは、ユーザ数が限られており、専門性の高い知識が必要でありするため、LLM をチューニングするためのベンチマーク問題を用意することが難しい。

LLM によるシミュレーションの完全自動化は困難であるが、人間にとって難解な専門ライブラリの読解やコード開発支援は実施可能である。これまで専門家でしか扱えなかったライブラリを使った開発のハードルが大幅に下がり、有用である。

3. 本研究

本研究では、企業の設計活動に用いる 1DCAE のモデリング作業の支援に LLM の適用可能性を検討した。

3-1 設計活動における 1DCAE のモデルの役割

藤田(藤田喜久雄, 2023)に言及した設計における 1DCAE のモデルとライブラリの役割についてまとめる。

3-1-1 コンカレントエンジニアリングを実現するための 1DCAE の役割

コンカレントエンジニアリングは、メカ、電気、ソフトウェア（制御）といった異なる領域の設計をオーバーラップさせて並行的に進める手法である。近年の製品は、メカの挙動をソフトウェアによって細かく制御することが価値向上につながる機能であることが多く、複合領域の設計が重要になっている。各領域のオーバーラップ部分をシミュレーションモデルを使い効率的に開発するための技術体系は、モデルベース開発（MBD）と呼ばれている。モデルベース開発では、メカ領域は 3 次元 CAD をベースとした CAE によって製品の挙動や性質を分析する一方で、電気や制御は集中乗数系のシミュレーションで動的な挙動を計算することが多い。1DCAE とは、MBD におけるシミュレーション手法の一つである。3 次元のメカ領域を抽象化し、電気、ソフトなどのシミュレーションに合わせ、機械系モデルを含めた複合領域の統合シミュレーションを可能にする。1DCAE により複合領域のシミュレーションが可能になり、コンカレントエンジニアリングの実現を支援する。

3-1-2 設計カタログとしての 1DCAE ライブラリ

設計工学では、設計を効率的に進めるための物理法則の知識を設計知識と呼ぶ。これらの知識や設計の根拠を取りまとめたものを設計カタログと呼ぶ。設計カタログは、設計根拠や設計知識を体系化し、設計の再利用性を高める役割を持つ。

1DCAE は、制御シミュレーションや物理シミュレーションが含まれる。これらのシミュレーションのモデルは、ライブラリとして管理されている。ライブラリには、対象システムの動的挙動を再現するための理論と実験

データが集積されており、モデルの忠実度（Fidelity）を支える根拠として機能する。このライブラリは、シミュレーションの実務が繰り返されるたびに、拡張されていき、複雑度を増していく。完成度の高い 1DCAE のライブラリは、対象システムを物理法則やその根拠となるデータを備えており、設計カタログとしての性質を備える。1DCAE におけるモデルは、シミュレーションの手段ではなく、設計根拠のカタログとなっている。

3.2 1DCAE のライブラリを活用することの実務課題

このようなモデリング作業は、モデル対象への理解、プログラム言語と専門ライブラリの仕様、数値計算などの知識や経験など高度なドメイン知識と経験が必要になる。このようなモデリングとシミュレーションを実行可能なエンジニアの養成するためには、多くの時間とコストが必要になる。

物理シミュレーション言語 Modelica や数値流体計算ライブラリ OpenFOAM によるモデリング作業とシミュレーション実行は、研究開発でよく使われる専門ライブラリであるが、ライブラリの複雑さと、モデル対象の理解に専門性が必要なため、高度な専門知識をもつエンジニアや研究者を必要とする。専門性の狭い分野のため、一般書籍や Review 誌などのまとまった情報を入手することも容易ではなく、ライブラリの学習の妨げになっている。このようにシミュレーション手段と設計カタログの性質を兼ね備えるライブラリであるが、利用するのは高度なスキルが必要になる。

3.3 本研究で対象とした Modelica ライブラリ

本研究では、空調開発における 1DCAE ライブラリに生成 AI の支援を適用した例を報告する。当社（三菱電機）でクローズされたライブラリの活用 LLM を適用した。

本研究で対象としたライブラリについて説明する。

三菱電機では、冷凍サイクルの物理シミュレーションに海外研究所 Mitsubishi Electric Research Laboratory (MERL) で開発した Modelica ライブラリを使用している。本稿では、このライブラリを MERL HVAC Library と呼ぶことにする。このライブラリは、エアコンなどの空気調和機の冷凍サイクルを動的な挙動をシミュレーションすることができる。冷凍サイクルを構成する熱交換器、圧縮機、膨張弁などの内部を流れる冷媒の圧力、温度、比エンタルピーといったパラメータを計算することができる。

ユーザはそれらを組み合わせることで高い再利用性と保守性をもったモデルを構築できる。

冷媒の挙動は、非線形性が強く、動的なシミュレーションが困難である。MERL HVAC Library では、冷媒流れの差分化方法、計算スキームを選択できる (Laughman,)。

3.3.1 ライブラリの内部構造

MERL HVAC Library はシステム全体の熱的振る舞いをモデリング可能なコンポーネント群を備えており、建築熱負荷・冷凍サイクル・制御アルゴリズムに関するモデルが豊富に実装されている。2025 年 2 月現在では、MERL HVAC Library は 8 種類のパッケージで校正されている。現在は、パッケージの全ファイル数は 3,112 個ある。各ファイルに定義されたクラスとメソッドの継承関係が複雑であることと、冷凍サイクルの物理モデリング手法が多岐にわたることがライブラリの理解を難しくしている。

3.3.2 数値積分手法の理解

このライブラリには主に 3 種類の差分法が実装されている。モデルの物理スケールと時間スケールによって適切なモデルを選択する必要がある。シミュレーション実行時には、熱的・流体的計算の複雑性により収束性や計算時間が問題となる場合があるため、ステップサイズ・ソルバ・パラメータのスケールに留意する。

以上のように、複雑な Modelica ライブラリを活用するには、(i) 開発環境と依存関係の適切な管理、(ii) ライブラリ内部構造の理解、(iii) Modelica 言語の文法と設計思想の把握、(iv) 数値計算の安定性に関する知識、が不可欠である。

3.3.3 専門ライブラリを読み解くためのドキュメント

4. Modelica ライブラリの設計における役割

前節の MERL HVAC Library には、よるモデリングの一例としてルームエアコンのモデルを示す。一部のモデルはダイアグラム表示によってクラスの継承関係の可視化が可能であるが、ほとんどのモデルはダイアグラム表示の実装中であり、クラス継承関係はコードから読み解く必要がある。

企業内で構築されているライブラリについては、上記と同様に、高度な理論

4.1 シミュレーションの方法

MERL HVAC Library などの物理シミュレーションライブラリを用いてシミュレーションモデルを構築する際のエンジニアの作業プロセスを書く。ルームエアコンを例にとると、熱交換器、ファン、圧縮機、膨張弁などの該当するクラスを探索し、各クラスを継 w したモデルを接続すれば良いのであるが、

また、数値差分法のアルゴリズムを選択する。

4.1.1 モデルの理論の調査

公開情報は、博士論文や最新の論文などを調べる。社外秘密の情報

4.2 大規模言語モデルの応用

本稿で構築した 2 つのシステムについて説明する。

4.2.1 RAG システム

図 1 に示す RAG システムは 4 層構造で設計した。

- (1) データ：MERL HVC Library コード、ドキュメント、論文を Markdown へ変換し、Sentence-BERT で 768 次元ベクトル化して FAISS に格納する。
- (2) 検索層：ユーザ質問を同一埋め込み空間で検索し、上位 k 件を取得。複数チャンクを階層マージしてプロンプト長を最適化する。
- (3) 生成層：OpenAI GPT-4o-mini をベースとする LLM に対し、役割指示+検索結果+会話履歴から構成されるテンプレートを送信する。
- (4) 出力層：回答と同時に引用情報を提示し、ユーザはソースコード断片へ直接ジャンプできる。

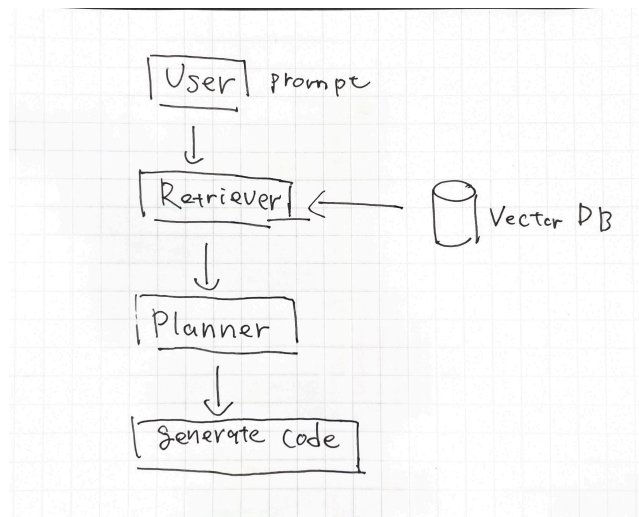


Fig. 1 Workflow chart of our agent.

4.2.2 Modelica コード生成エージェント

図 2 にエージェントのワークフローを示す。核心は、Plan-Execute-Critique ループにより「検索→提案→コーディング」を自律実行する点である。

- (1) RAG: 前節の RAG システムを用いて、ユーザからのプロンプトに関連するドキュメントを検索する。

- (2) プランナー：RAG システムが出力した検索結果からモデル構造の手順書（pseudo-plan）を生成する。
- (3) コーダー：手順書に従いコードスニペットを生成する。

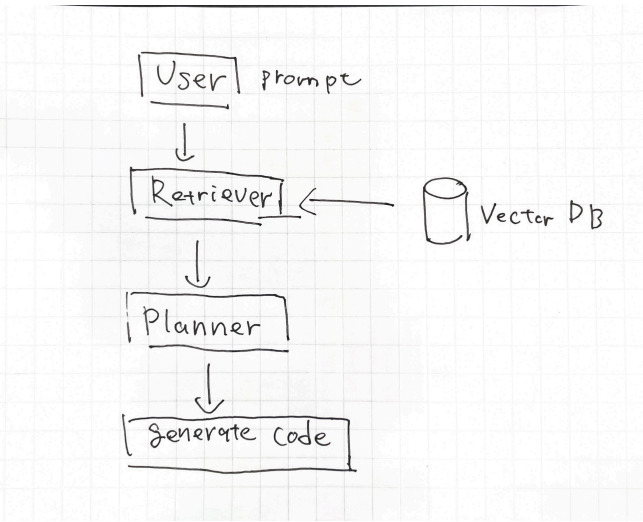


Fig. 2 Workflow chart of our agent.

5. 結果

5.1 単体システムの回答

一般知識で回答できる問題は、熱力学の知識に関する問題など、設計エンジニアにとって難易度の高い問題も回答することができる。ただし、

5.1.1 公知の情報で回答可能な可能な可能質問

5.2 RAG システムの回答

5.2.1 文書データベースの情報が回答に必要な質問

prompt

5.2.2 コード作成の指示

Table 2 Response of LLM

prompt	Construct room air conditioner model by MERL-HVAC Library.
Answer	I don't know. Source Chunks: <ul style="list-style-type: none">- VCComponents\Components\AirHandling\Testing\MultiportLumpedRoomTest.mo-0- VCComponents\Components\AirHandling\FanCurveModels\package.mo-0- VCComponents\Components\AirHandling\FanCurveModels\BaseClasses\package.mo-0- MEBuildings\USD0E\SmallOffice\FullMode\FourZoneOffice.mo0

5.3 Modelica コード生成エージェント

RAG システムでは、無回答であったコーディング作業にも

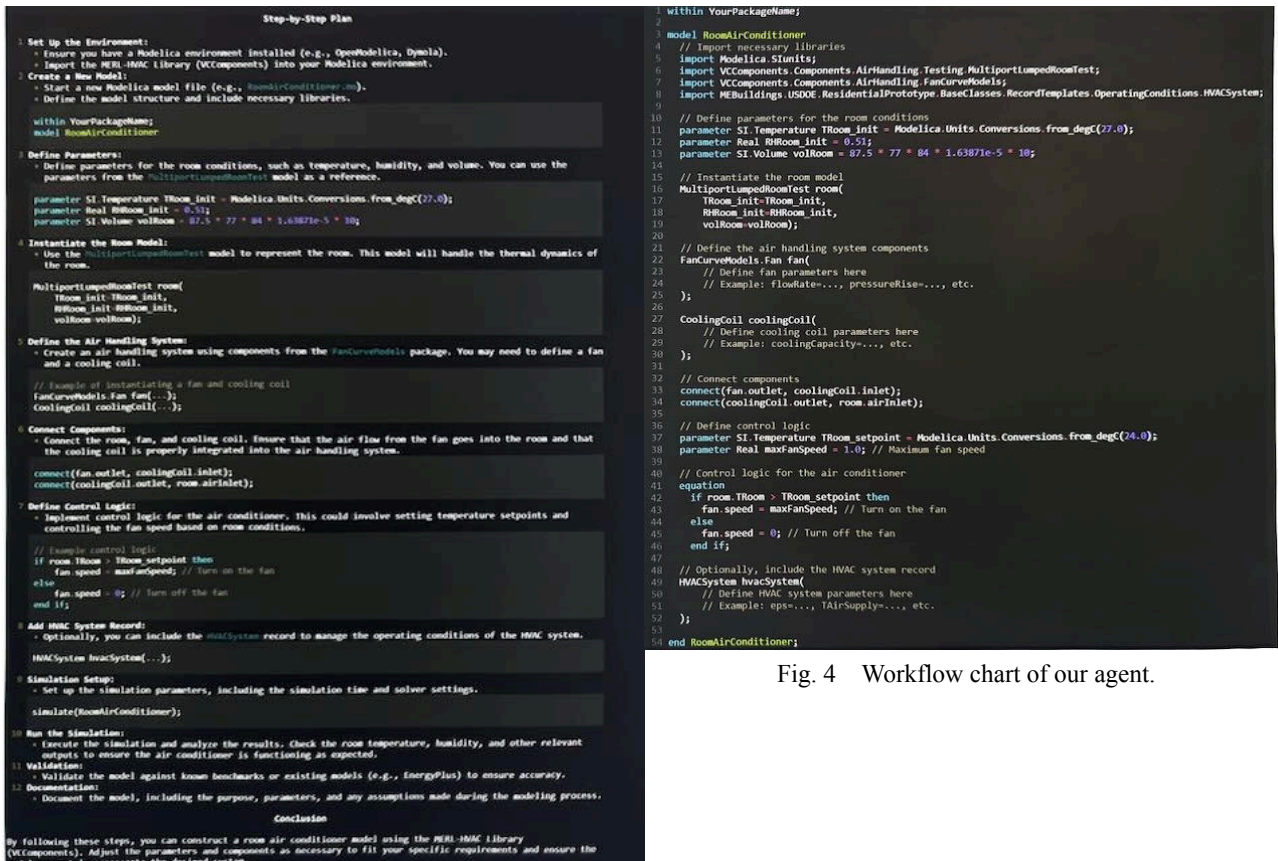


Fig. 3 Workflow chart of our agent.

Fig. 4 Workflow chart of our agent.

6. 結 言

本稿では、大規模言語モデルの物理シミュレーションへの作業支援について調べた。現状では、複雑なプロセスが必要な実用ライブラリのコーディング作業は完全には自動化できないが、知識探索やコードの作成方針を探索には実用できることを明らかにした。

References

- Chen, Y., Zhang, L., Zhu, X., Zhou, H. and Ren, Z., OptMetaOpenFOAM: Large Language Model Driven Chain of Thought for Sensitivity Analysis and Parameter Optimization based on CFD (2025).
- Chen, Y., Zhu, X., Zhou, H. and Ren, Z., MetaOpenFOAM: an LLM-based multi-agent framework for CFD (2024).
- Laughman, C. R., A Comparison of Transient Heat-Pump Cycle Simulations with Homogeneous and Heterogeneous Flow Models.
- Onishi, K., Lessons from toyota for building durable multi-agent copilots, Presented at Microsoft Ignite 2024 (2024), Session BRK117.
- P, Ey, S., Eep, Xu, R., Wang, W. and Chu, X., OpenFOAMGPT: a RAG-Augmented LLM Agent for OpenFOAM-Based Computational Fluid Dynamics, available from <<http://arxiv.org/abs/2501.06327>>.
- Picard, C., Edwards, K. M., Doris, A. C., Man, B., On, Giannone, G., Alam, M. F. and Ahmed, F., From Concept to Manufacturing: Evaluating Vision-Language Models for Engineering Design (2024).
- Xiang, J., Ye, T., Liu, P., Zhang, Y. and Wang, W., Modigen: a large language model-based workflow for multi-task modelica code generation (2025).

日本機械学会, 設計のための 1DCAE 概念と実現技術, 日本機械学会 (2020), p. 256, 初版発行日: 2020 年 11 月 2 日.

玉谷基亮, 三菱電機グループの持続的なものづくりを支える AI ソリューション群, [三菱電機技報](#), Vol. 98, No. 9 (2024), pp. 3–6.

藤田喜久雄, 設計論: 製品設計からシステムズイノベーションへ, コロナ社 (2023), p. 494, 初版第 1 刷発行日: 2023 年 7 月 7 日.