Can ChatGPT Pass Modern Control Theory Exam?

Itsuki Ogo[1*] and Masanobu Koga[1]

[1]Department of Intelligent and Control Systems, Kyushu Institute of Technology,
Fukuoka, Japan (ogo.itsuki406@mail.kyutech.jp, koga@ics.kyutech.ac.jp) * Corresponding author

Abstract: Large language models (LLMs), such as GPT models, have been rapidly studied in recent years and are expected to be applied to academic fields such as mathematics and engineering. In this study, we examined how accurately ChatGPT (GPT-4o) can answer modern control theory questions at the undergraduate-level. A set of 98 questions on modern control theory was used to evaluate GPT-4o's problem-solving ability on modern control theory. The results revealed that the GPT-4o showed a 49.0% correct response rate to the undergraduate-level modern control theory exercises, and that the correct response rate tended to be lower for problems involving calculations, especially those that require step-by-step thinking and complex computation. This may be attributed to the Transformer architecture of the GPT model, which generates answers based on probabilistic predictions. In this study, we proposed a method to improve response accuracy by developing a customized GPT which leverages prompt engineering methods to address these issues. In order to evaluate the proposed method, a question set consisting of 45 graduate school entrance exam questions on modern control theory was developed. The results of the evaluation showed that the correct response rate was improved by 26.6 points, yielding a 64.4% correct response rate.

Keywords: ChatGPT, Modern Control Theory, Generative AI, GPTs, GPT-4o

1. INTRODUCTION

In recent years, the emergence of Google's BERT[1] and OpenAI's GPT[2] has led to rapid research in the field of natural language processing. In particular, GPT models, have contributed to solving problems in a wide variety of fields, including mathematics[3], [4], coding[5], [6], etc. The first version of the GPT model was released in 2018, followed by GPT-2 and GPT-3[2]. These models have shown remarkable results in the field of natural language processing and their range of applications have expanded rapidly; in March 2023, GPT-4[7], a multimodal model with the ability to handle multiple modalities, including images and speech as well as text, was released. This model enabled the processing of various data formats and applications for more complex and diverse tasks. Furthermore, GPT-4o was released in May 2024[8], further expanding the model's performance and range of applications.

It will be very interesting to see what LLM can do with respect to control theory, since it is expected to be applied in a variety of domains. The integration of artificial intelligence into modern control engineering systems has the potential to dramatically accelerate technological progress. In [9], the GPT-4's ability was assessed on a set of exercises in classical control theory at the undergraduate level. For 97 basic exercise questions, the GPT-4 showed a correct response rate of approximately 66%. In [10], a benchmark evaluation is conducted to determine how well the leading LLMs demonstrate their capabilities with respect to control engineering. The evaluation using a set of 147 control engineering questions revealed that GPT-4 had a 45.6% correct response rate. However, many of the questions were limited to classical control theory, and no comprehensive evaluation was conducted with respect to modern control theory.

In classical control theory, SISO systems are primarily targeted, and transfer functions (rational polynomials) are used as the system model representation. In contrast, modern control theory targets MIMO systems and uses state-space representations (differential equations with matrices as coefficients) for system modeling. This paper focuses on modern control theory for its investigation. Moreover, control engineering problems often involve dynamic behavior of the system, which present complexities different from traditional mathematical problems. LLMs need to understand and appropriately apply for analysis and design when solving these problems. Furthermore, since there are multiple approach to the solution, a comprehensive approach is required.

In this study, by having GPT-4o solve university undergraduate-level control problems, we explored the extent to which it demonstrates competence with respect to modern control theory. Based on the evaluation and the analysis, we proposed a method to improve the accuracy of its answers. We created a customized GPT specific to modern control theory. When evaluated on a question set consisting of graduate school entrance exam questions on modern control theory, GPT-4o showed a 37.8% correct response rate, whereas the customized GPT showed a 64.4% correct response rate.

In Chapter 2, we report the results of the experiments to evaluate the performance of the GPT-4 on modern control theory; in Chapter 3, we propose methods to improve the response accuracy of the GPT-4o, including the GPTs; in Chapter 4, we compare the response rates of the GPT-4o and the proposed methods by using a question set consisting of graduate school entrance exam questions. In Chapter 5, we discuss the effectiveness of the proposed method, and in Chapter 6, we provide a summary and future work of this study.

2. PERFORMANCE EVALUATION OF GPT-4o

In order to evaluate GPT-4o's ability for solving control problems, a problem set consisting of 98 undergraduate-level modern control theory questions was created. This problem set was composed of questions from [11] and questions from a modern control theory course at the Kyushu Institute of Technology. GPT-4o is capable of reading a variety of files, including pdfs. However, in modern control theory, multiple types of information such as block diagrams, mathematical expressions, and text are mixed and arranged in one page. In [9], GPT-4, which is also a multimodal model, was found to be poor at reading such mixed information. Therefore, in this study, the PDF format was converted to a structured markdown format.

2.1 Experimental Procedures

A schematic procedure of the evaluation experiment is shown in Fig. 1

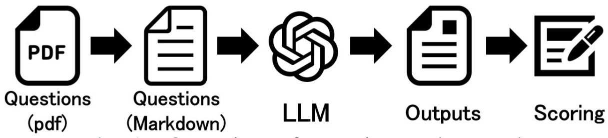


Fig. 1. Overview of experimental procedure

The set of questions in markdown format is given to GPT-4o. The obtained answers are checked against the model answers to determine correctness. Accuracy (ACC), the evaluation metric used in this paper, is defined as the ratio of the number of correct answers to all questions.

2.2 Experimental results

The results of the experiment are shown in Table 1. In Table 1, the problem sets created are categorized by topic and ACC for each topic is expressed as a percentage (%); the number of correct answers/total number of questions is shown in parentheses next to the ACC. For each topic, questions involving calculations are classified as "Calculation Question," questions not involving calculations are classified as "Knowledge Question." Table 1 shows that the GPT-4o has a 49% correct response rate to modern control theory exercises at the undergraduate-level. The correct response rate was particularly high for questions that did not require calculation. However, the correct response rate was lower for questions involving calculation, and those that requires step-by-step thinking and complex calculation had especially lower rate. It should be noted that there is a randomness in the LLM response. The same question may generate completely different answers. Therefore, these results should only be taken as an indication of the GPT-4o's ability to deal with modern control theory.

3. METHODS

3.1 GPTs

Prompt engineering is a technique to enhance LLM performance without relying on large datasets[12]. Prompt engineering optimizes effective questions and instructions (prompts) to elicit accurate and useful responses from a generative AI. It plays an important role in artificial intelligence, particularly in the field of natural language processing. The technique is applied to large-scale language models, such as GPT, with the goal of devising an input design that allows the model to produce the desired output. This is expected to improve the performance of AI models and increase their adaptability to specific tasks.

OpenAI announced GPTBuilder on November 7, 2023, which allows users to create customized ChatGPTs, called GPTs[13]. Users can build GPTs suitable for specific tasks by editing four main parameters: "Instructions", "Knowledge", "Capabilities", and "Actions" shown in Table 2.

Table 2. Parameters of GPTs

Parameter	Function
Instructions	Prompts given in advance
Knowledge	Data
Capabilities	Code Interpreter (Python engine)
Actions	Third party APIs

3.2 Code Interpreter

Although, the GPT models and other LLMs appear to provide answers by "thinking," they just generate "reasonable sequences[14]." The GPT-4o whoes architecture is a Transformer[2], may give wrong answers to the problems that require complex calculations, since the answers are simply predictions based on probabilities[15].

LLMs are known to show remarkable ability in effectively generating code in general. In particular, problems related to computational errors can be mitigated by using Program of Thoughts (PoT)[16]. For example, for chemistry problems, the computational error was significantly reduced by making LLM to generate Python code[17].

In this study, GPTs are instructed to consistently use the Code Interpreter for questions requiring computation, to perform numerical and symbolic calculations in Python. For example, the Instructions are described as follows: "Always use Code Interpreter for numerical and symbolic computations," "For numerical calculations, use the Numpy function in Python." The flow to use Code Interpreter is shown in Fig. 2

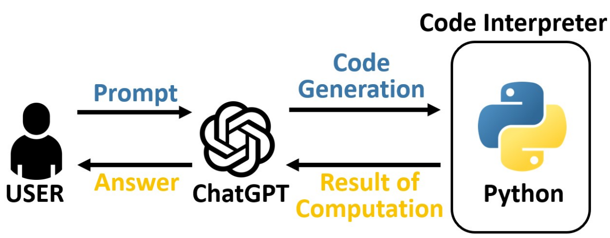


Fig. 2. Flow to use Code Interpreter

In addition, providing specific Python code to the GPTs

Table 1. Evaluation results

	Calculation Question	Knowledge Question	Question with Figure
Topic	**ACC**	**ACC**	**ACC**
State Space Representation	100.0% (2/2)	50.0% (1/2)	44.4% (4/9)
Laplace Transform	0.0% (0/3)	100.0% (2/2)	-
Stability	50.0% (3/6)	100.0% (2/2)	-
Controllability	37.5% (3/8)	100.0% (2/2)	-
Observability	25.0% (2/8)	100.0% (1/1)	0.0% (0/1)
Coordinate Transformation	33.3% (3/10)	100.0% (3/3)	-
Transfer Function	66.7% (2/3)	100.0% (2/2)	-
Realization	0.0% (0/2)	100.0% (1/1)	-
State Feedback	33.3% (1/3)	-	-
Pole placement method	25.0% (2/8)	-	-
Optimal regulator	33.3% (1/3)	100.0% (3/3)	-
Servo system	0.0% (0/1)	100.0% (3/3)	-
Observer	20.0% (1/5)	80.0% (4/5)	-
Subtotal	32.3% (20/62)	92.3% (24/26)	40.0% (4/10)
Total		**49.0% (48/98)**	

helps reduce the randomness in their responses, which is often seen as a drawback of the GPT model.

3.3 ModernControlGPT

Based on the results of the evaluation experiments, we created the ModernControlGPT (MCGPT) to cope with modern control theory. This GPT incorporates various prompt engineering methods, including Program of Thoughts and Zero-Shot CoT Prompting[18], and Few-Shot Prompting[19], [20]. These methods are applied to the problems that GPT-4o made mistakes on the evaluation experiment, which are considered to have relatively less training data.

Many Python libraries can be called using the Code Interpreter. Libraries available include NumPy, a numerical computation library, SymPy, a symbolic computation library, and SciPy, a numerical analysis library for mathematics, science, and engineering. These libraries can be used for matrix computations, signal processing, and control system analysis. The total number of Python libraries installed in ChatGPT is 355 on June 10, 2024.

Python Control Systems Library[21], is not available by default in Code Interpreter. MCGPT made it available by uploading the library in zip format and a markdown file that summarizes the uses. An example of MCGPT output is shown in Fig. 3 which shows that the functions of the Control library can be used. the Control library is decompressed and added to the library path using Code Interpreter.

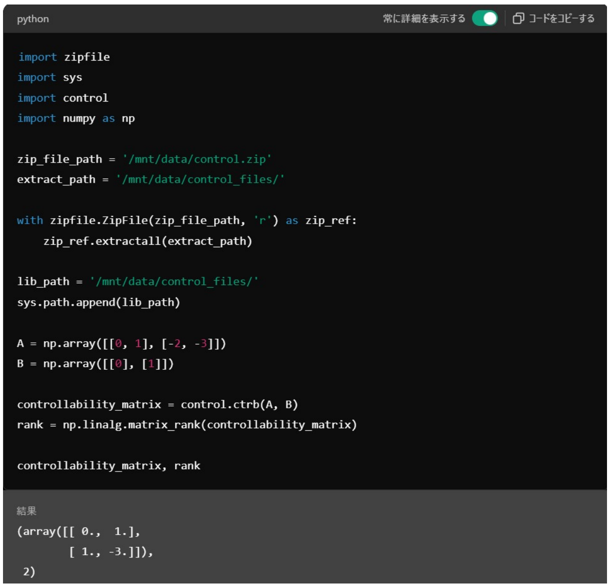


Fig. 3. An example of output of ModernControlGPT

4. EVALUATION EXPERIMENT

4.1 Experimental method

To evaluate the proposed ModernControlGPT, we created a set of questions consisting of 45 graduate school entrance exam questions on modern control theory as follows. The graduate school entrance exam questions and the model answers are available from [22].

- **Graduate Schools of Osaka University**
  - Department of Electrical, Electronic, and Information Engineering
    (Electrical Engineering Course)
  - in 2016, 2017, 2018, 2019, and 2020
- **Graduate School of the University of Electro-Communications**
  - Department of Mechanical Intelligent Systems
  - in 2020

- **Graduate School of Tokyo Institute of Technology**
  - Department of Mechanical Engineering and Control Systems
  - in 2017 and 2018
- **Graduate School of Hokkaido University**
  - Department of Mechanical and Space Engineering (Mechanical and Aerospace Engineering Laboratories)
  - in 2019

For comparison, “GPT-4o” and “ModernControlGPT” are used. The ACC of each model is compared to evaluate the effectiveness of the proposed method.

4.2 Experimental Results

The experimental results are shown in Table 3. If a partially correct answer was obtained 0.5 points were awarded for the answer. Table 3 shows that the proposed method improves response accuracy by 26.6 points and yields a 64.4% correct response rate.

The problems that showed significant differences in the evaluation experiment are shown in Fig. 4. The problem is Question No. 6 of the entrance exam for the Graduate School of Engineering, Tokyo Institute of Technology in 2017. The answers of GPT-4o for Questions 3, 4, and 5 are shown in Fig. 5, 6, 7, respectively. Also, the answers of MCGPT are shown in Fig. 8, 9, 10, respectively.

Answer the following questions regarding the system represented by the following state equation. Note that a is a real number.

$$\frac{dx}{dt} = Ax + Bu, \quad x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix},$$

$$A = \begin{bmatrix} a-1 & 1 \\ 2 & a-2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$y = Cx, \quad C = \begin{bmatrix} 1 & 2 \end{bmatrix}$$

(1) Determine the controllability of this state equation.
(2) Using the following coordinate transformation, derive the state equation with the state $\bar{x}$.

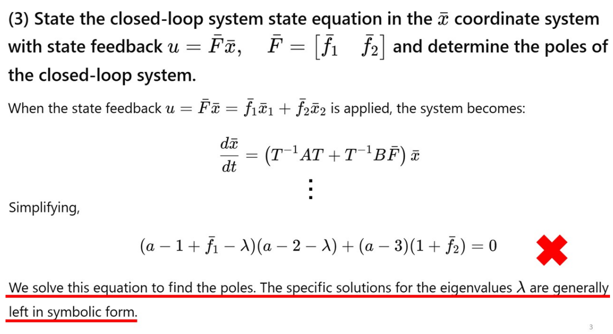
$$x = T\bar{x}, \quad T = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad \bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix}$$

(3) Show the state equation of the closed-loop system in the $\bar{x}$ coordinate system with the state feedback

$$u = \bar{F}\bar{x}, \quad \bar{F} = \begin{bmatrix} \bar{f}_1 & \bar{f}_2 \end{bmatrix}$$

and find the poles of the closed-loop system.
(4) Determine the condition on a such that the closed-loop system can be stabilized by the state feedback in (3).
(5) Find all the state feedbacks in (3) that result in the closed-loop system having repeated poles.

Fig. 4. Question 6 of the entrance exam for the Graduate School of Engineering, Tokyo Institute of Technology in 2017


(3) State the closed-loop system state equation in the $\bar{x}$ coordinate system with state feedback $u = \bar{F}\bar{x}, \quad \bar{F} = [\bar{f}_1 \quad \bar{f}_2]$ and determine the poles of the closed-loop system.

When the state feedback $u = \bar{F}\bar{x} = \bar{f}_1\bar{x}_1 + \bar{f}_2\bar{x}_2$ is applied, the system becomes:

$$\frac{d\bar{x}}{dt} = (T^{-1}AT + T^{-1}B\bar{F})\,\bar{x}$$

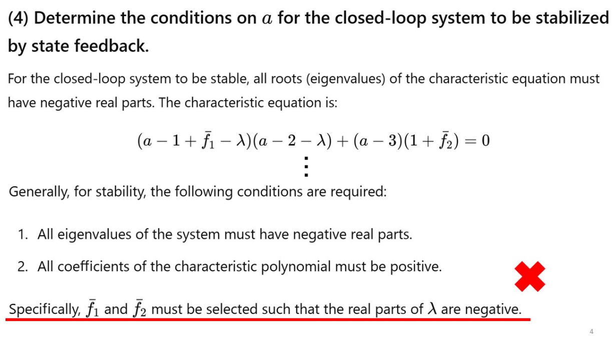
⋮

Simplifying,

$$(a-1+\bar{f}_1-\lambda)(a-2-\lambda)+(a-3)(1+\bar{f}_2)=0$$

We solve this equation to find the poles. The specific solutions for the eigenvalues λ are generally left in symbolic form.

Fig. 5. Answer to question 3 by GPT-4o


(4) Determine the conditions on a for the closed-loop system to be stabilized by state feedback.

For the closed-loop system to be stable, all roots (eigenvalues) of the characteristic equation must have negative real parts. The characteristic equation is:

$$(a-1+\bar{f}_1-\lambda)(a-2-\lambda)+(a-3)(1+\bar{f}_2)=0$$

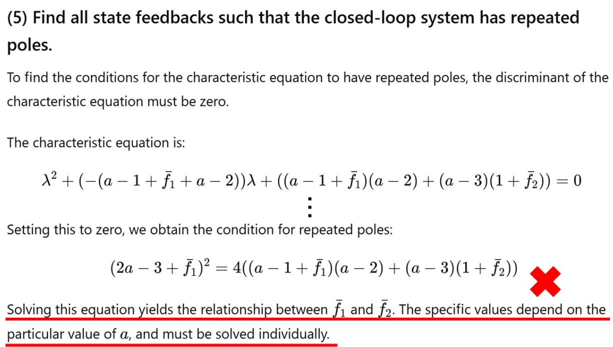
⋮

Generally, for stability, the following conditions are required:

1. All eigenvalues of the system must have negative real parts.
2. All coefficients of the characteristic polynomial must be positive.

Specifically, $\bar{f}_1$ and $\bar{f}_2$ must be selected such that the real parts of λ are negative.

Fig. 6. Answer to question 4 by GPT-4o


(5) Find all state feedbacks such that the closed-loop system has repeated poles.

To find the conditions for the characteristic equation to have repeated poles, the discriminant of the characteristic equation must be zero.

The characteristic equation is:

$$\lambda^2+(-(a-1+\bar{f}_1+a-2))\lambda+((a-1+\bar{f}_1)(a-2)+(a-3)(1+\bar{f}_2))=0$$

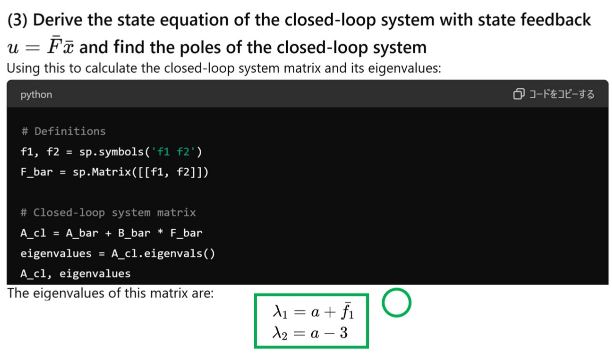
⋮

Setting this to zero, we obtain the condition for repeated poles:

$$(2a-3+\bar{f}_1)^2=4((a-1+\bar{f}_1)(a-2)+(a-3)(1+\bar{f}_2))$$

Solving this equation yields the relationship between $\bar{f}_1$ and $\bar{f}_2$. The specific values depend on the particular value of a, and must be solved individually.

Fig. 7. Answer to question 5 by GPT-4o


(3) Derive the state equation of the closed-loop system with state feedback $u = \bar{F}\bar{x}$ and find the poles of the closed-loop system

Using this to calculate the closed-loop system matrix and its eigenvalues:

python コードをコピーする

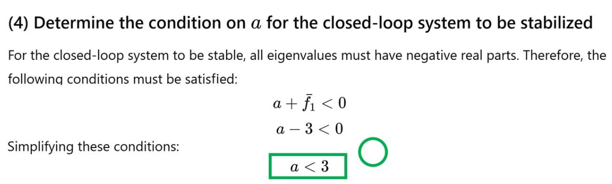
```python
# Definitions
f1, f2 = sp.symbols('f1 f2')
F_bar = sp.Matrix([[f1, f2]])

# Closed-loop system matrix
A_cl = A_bar + B_bar * F_bar
eigenvalues = A_cl.eigenvals()
A_cl, eigenvalues
```

The eigenvalues of this matrix are:

$$\lambda_1 = a + \bar{f}_1$$
$$\lambda_2 = a - 3$$

Fig. 8. Answer to question 3 by MCGPT


(4) Determine the condition on a for the closed-loop system to be stabilized

For the closed-loop system to be stable, all eigenvalues must have negative real parts. Therefore, the following conditions must be satisfied:

$$a + \bar{f}_1 < 0$$
$$a - 3 < 0$$

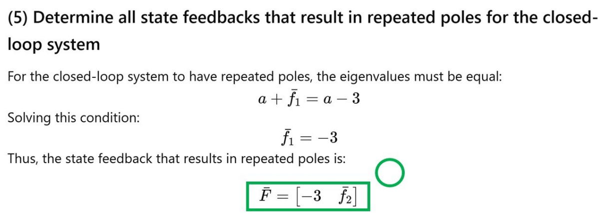
Simplifying these conditions:

$$a < 3$$

Fig. 9. Answer to question 4 by MCGPT

Table 3. Comparison of GPT-4o and ModernControlGPT on various topics

	GPT-4o	MCGPT
Topics	**ACC**	**ACC**
State Space Representation	20.0% (1/5)	40.0% (2/5)
LaplaceTransform	16.7% (0.5/3)	33.3% (1/3)
Stability	25.0% (2/8)	50.0% (4/8)
Controllability	50.0% (3/6)	66.7% (4/6)
Observability	66.7% (2/3)	66.7% (2/3)
Coordinate Transformation	16.7% (0.5/3)	33.3% (1/3)
Transfer Function	87.5% (3.5/4)	100.0% (4/4)
Realization	-	-
State Feedback	35.0% (3.5/10)	80.0% (8/10)
Pole placement method	-	-
Optimal regulator	-	-
Servo system	-	-
Observer	33.3% (1/3)	100.0% (3/3)
Total	**37.8% (17/45)**	**64.4% (29/45)**


(5) Determine all state feedbacks that result in repeated poles for the closed-loop system

For the closed-loop system to have repeated poles, the eigenvalues must be equal:

$$a + \bar{f}_1 = a - 3$$

Solving this condition:

$$\bar{f}_1 = -3$$

Thus, the state feedback that results in repeated poles is:

$$\bar{F} = [-3 \quad \bar{f}_2]$$

Fig. 10. Answer to question 5 by MCGPT

From Fig. 5, 6, and 7, it can be seen that GPT-4o could not generate the concrete final answers partially due to symbolic computation. On the other hand, Fig. 8, 9, and 10, show that the MCGPT could obtain appropriate answers by using the code interpreter.

5. DISCUSSION

In this study, response accuracy to the questions on modern control theory was improved by simply devising prompts without additional training data. The method improved response accuracy by incorporating various measures into the GPT prompts. An important feature of the proposed method is that it does not require a detailed presentation of the problem and its concrete solution, as is the case with conventional Few-shot CoT. Additionally, there is no need to prepare additional datasets and retrain the model as with fine-tuning. As a result, the time and cost associated with data preparation and model training can be significantly reduced. In addition, although all the instructions prepared by this method are written in Japanese, it was confirmed that the GPT can appropriately respond in English as well. Customized GPT can be easily used by users who speak any languages which are available on ChatGPT.

However, due to the inherent nature of the GPT model, complete reproducibility of the generated answers cannot be guaranteed. Even if a question is answered correctly once, it is possible that the GPT will give an incorrect answer when asked to answer it again. Therefore, it should be noted that the proposed method may not always produce correct answers.

The proposed GPT are created using a combination of various prompt engineering methods. However, LLMs can change their output significantly even with slight changes in the input prompts. In [23], that even minimal changes, such as adding a space at the end of a prompt, can cause changes in the output of a large-scale language model (LLM) is shown. Also, in [10], simply changing the separate line formulas to inline formulas has been shown to change the responses. Therefore, it cannot be definitively stated that the GPT created in this study uses the most appropriate prompts for solving modern control theory problems.

6. CONCLUSIONS

In this study, we evaluated the performance of ChatGPT (GPT-4o) by having it answer various exercises in modern control theory. Then, we created a customized GPT that could derive the correct answer to the questions that GPT-4o got wrong, thereby improving the accuracy of the answers. However, as mentioned in section 3.3, the architecture of the GPT model is Transformer, which only generates text based on probabilities. Thus, even for questions where the customized GPT provided correct answers in this experiment, repeated attempts may yield incorrect answers. Therefore, it is necessary to devise a method to control the randomness of answers in the future. Since this method is a combination of various prompt engineering methods, we are concerned that we cannot theoretically explain which prompts affect the output and how. In addition, for problems that require more advanced design methods, such as H_∞ control, there is a possibility of providing incorrect answers. Therefore, in the future, it is necessary to adapt to more complex problems by integrating with external tools like MATLAB using Actions. Furthermore, it is necessary to evaluate other LLMs as well, to identify which models excel in specific tasks or fields, and select the most suitable LLM for control engineering.

REFERENCES

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[2] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[3] Simon Frieder, Luca Pinchetti, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Petersen, and Julius Berner. Mathematical capabilities of chatgpt. *Advances in Neural Information Processing Systems*, 36, 2024.

[4] Shima Imani, Liang Du, and Harsh Shrivastava. Mathprompter: Mathematical reasoning using large language models. *arXiv preprint arXiv:2303.05398*, 2023.

[5] Daye Nam, Andrew Macvean, Vincent Hellendoorn, Bogdan Vasilescu, and Brad A. Myers. Using an llm to help with code understanding. 2024.

[6] Stephen MacNeil, Andrew Tran, Dan Mogil, Seth Bernstein, Erin Ross, and Ziheng Huang. Generating diverse code explanations using the gpt-3 large language model. In *Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 2*, pages 37–39, 2022.

[7] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[8] OpenAI. Hello GPT-4o. https://openai.com/index/hello-gpt-4o/, may 2024. Accessed on June. 9, 2024.

[9] Itsuki Ogo and Masanobu Koga. Can ChatGPT pass classical control theory exam? In *SICE Festival 2024 with Annual Conference*, Kochi, Japan, 2024.

[10] Darioush Kevian, Usman Syed, Xingang Guo, Aaron Havens, Geir Dullerud, Peter Seiler, Lianhui Qin, and Bin Hu. Capabilities of large language models in control engineering: A benchmark study on gpt-4, claude 3 opus, and gemini 1.0 ultra. *arXiv preprint arXiv:2404.03647*, 2024.

[11] Yasuchika Mori. *Modern Control Theory through Exercises*. Morikita Publishing Co., Ltd., 2003. (in Japanese).

[12] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9), jan 2023.

[13] OpenAI. Introducing GPTs. https://openai.com/blog/introducing-gpts, 2024. Accessed on Feb. 5, 2024.

[14] Stephen Wolfram. *What Is ChatGPT Doing... And Why Does It Work?* Wolfram Media Inc, 2023.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[16] Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*, 2023.

[17] Siru Ouyang, Zhuosheng Zhang, Bing Yan, Xuan Liu, Yejin Choi, Jiawei Han, and Lianhui Qin. Structured chemistry reasoning with large language models. *arXiv preprint arXiv:2311.09656*, 2024.

[18] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

[19] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[20] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

[21] Python control systems library. https://python-control.readthedocs.io/en/0.10.0/, mar 2023. Accessed on June. 9, 2024.

[22] Yasuchika Mori. *Textbook for Graduate School Entrance Examination: Control Engineering*. Corona Publishing Co.,Ltd., 2021. (in Japanese).

[23] Abel Salinas and Fred Morstatter. The butterfly effect of altering prompts: How small changes and jailbreaks affect large language model performance. *arXiv preprint arXiv:2401.03729*, 2024.