From Text to Map; Why Not Vice Versa, and Beyond? Investigating the Application of LLMs in Interpreting SD Models' Output Graphs

SeyedAmirreza Eslami Pouya[a], Shayan Firouzian H.[b,] *, Mohammad Hossein Foroughi[c]

[a] *School of Industrial and Systems Engineering, Amirkabir University of Technology, Tehran, Iran*

[b] *Department of Socio-economic Systems, Faculty of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran*

[c] *School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran*

Abstract

Rapid advances in Large Language Models (LLMs) have opened new possibilities for interpreting System Dynamics (SD) model outputs. This paper investigates whether LLMs can analyze SD output graphs—without access to underlying model structures—and generate coherent textual interpretations, exploring how LLMs may reveal causal relationships and feedback loops embedded in output graphs, and use it in combination with previous Artificial Intelligence (AI)-based tools that derived SD diagrams from textual data; streamlining an automated cicular modeling process. Two benchmark SD models were constructed in Vensim to produce output graphs. Three state-of-the-art LLMs were provided with a standardized prompt instructing them to interpret the graphs. Their responses were then evaluated by SD experts and analyzed via the TOPSIS method. LLMs demonstrated a promising ability to interpret SD output graphs. The generated interpretations successfully identified key system feedback loops and suggested causal relationships, enabling a reverse-engineered depiction of the original model's structure. However, the approach remains exploratory, with current outputs reflecting both potential and limitations. Future research should refine prompting strategies and evaluation methods and explore integrating LLM-driven analysis directly into SD software to enhance practical analytical workflows.

Keywords: System Dynamics (SD), Artificial Intelligence (AI), Large Language Model (LLM), Causal Loop Diagram (CLD), Stock-Flow Diagram (SFD), Vensim

* Corresponding author:
Email addresses: shayanfirouzian@gmail.com (Shayan Firouzian Haji).

1. Introduction

The rapid advancements in artificial intelligence, particularly in the development of large language models (LLMs) and their reasoning capabilities, have opened new avenues for cross-disciplinary applications. This paper explores the promising prospect of leveraging reasoning variations of LLMs to analyze and interpret output graphs from System Dynamics (SD) models. Such an approach is driven by the potential to automate complex analytical tasks traditionally performed by expert system dynamicists, fostering educational use and offering a means to cross-validate human analysis.

SD models, often implemented in simulation software like Vensim, are pivotal for understanding the intricate behavior of complex systems. These models typically generate output graphs—such as time-series representations of populations or resource flows—that encapsulate the dynamic interplay of system variables. While experts are skilled in deciphering these graphs, early career practitioners and students frequently encounter challenges grasping the underlying dynamics. The ability to automate the analysis of such graphs not only enhances learning but also provides an efficient tool for preliminary assessments in research and practice.

In this study, we focus on a unique experimental setup where the underlying structure of the SD models, such as the Stock-Flow Diagrams (SFDs) used in Vensim, remains hidden from the LLMs. Instead, only the resulting output graphs (e.g., population trends over time) are provided as input. This design tests the models' ability to independently interpret the graphs, draw conclusions about the system's behavior, and potentially reverse-engineer the model structure—inferring causalities and feedback loops typical of Causal Loop Diagrams (CLDs) or SFDs.

The primary research objective is to evaluate the capabilities of LLMs in interpreting these visual outputs. The investigation centers on answering whether and how effectively these models can analyze graph-based data and whether they can reconstruct or explain the dynamics that generated them.

Comparing different responses is crucial for identifying the most effective version of an LLM's output. This evaluation process encompasses various forms of content, including textual and visual elements, among other types (Pagano *et al.*, 2025). An experimental methodology was implemented to assess performance, involving designing and distributing a detailed questionnaire to experts in SD. Their evaluations were then used to score the LLMs based on four key criteria: accuracy, clarity and readability, comprehensiveness, and creativity. The collected data were subsequently analyzed using a Multi-Criteria Decision Making (MCDM) approach, specifically the TOPSIS method, to provide a robust, comparative assessment of the models.

This paper addresses these research questions and contributes to the emerging dialogue on the intersection of AI and SD. It highlights the potential for advanced LLMs to be utilized to develop AI assistants that can understand, analyze, interpret, and explain SD models of complex dynamic systems and their outputs and serve as educational aids and validation tools in professional practice.

On the other hand, while Veldhuis et al. (2024) have shown how natural language processing (NLP) can help with SD modeling by identifying causal relationships in text, a more recent study by Hosseinichimeh et al. (2024) directly investigated the potentials of LLMs to automate the creation of causal loop diagrams by developing a framework which receives narrative textual data and after synthesizing it for causal relations, draws the CLD of the hypothetical SD model which that input text was supposedly its description. Following these efforts, alongside ever-increasing applications of AI and particularly LLMs, the reverse of the same path (*vice versa*), which can serve as the missing link in the cycle of the modeling process and its automation (*and beyond*), sounds very appealing, yet unexplored.

The final part of this study will assess the practicality and viability of such attempts by developing a framework to streamline the modeling process. The findings of this study are expected to pave the way for future research on integrating AI-driven analytical methods in various domains where dynamic modeling plays a critical role.

2. Background

Wang et al. (2024) explored the use of LLMs in generating expressive robot behavior in conversations, showcasing the integration of LLMs into social robots to enhance dynamic and expressive interactions. This application of LLMs in generating robot responses with personality congruence highlights the potential for utilizing LLMs in interpreting outputs of system dynamic models for enhanced communication and interaction.

Zheng et al. (2022) investigated quantized guaranteed cost output feedback control for nonlinear networked control systems, emphasizing the use of quantized control inputs for improved performance. Liu and Keith (2024) introduce and test a method for automating the translation of dynamic hypotheses into CLDs using LLMs combined with designing prompting strategies, while Hu (2025) presents a method to integrate system dynamics models into ChatGPT-4, enabling users to interact with and simulate complex models through natural language.

3. Methods and Material

3.1. Experiment's Design

In the initial phase of this study, two SD models were developed based on their benchmark status and recognition within the field. The first is a predator-prey system[1], while the second represents a SEIRD[2] model. Both models were constructed using Vensim software (Figures 1-2). This step served as the foundation of the methodology, as the output diagrams (Figures 3-4) generated in this phase were essential for the subsequent steps.

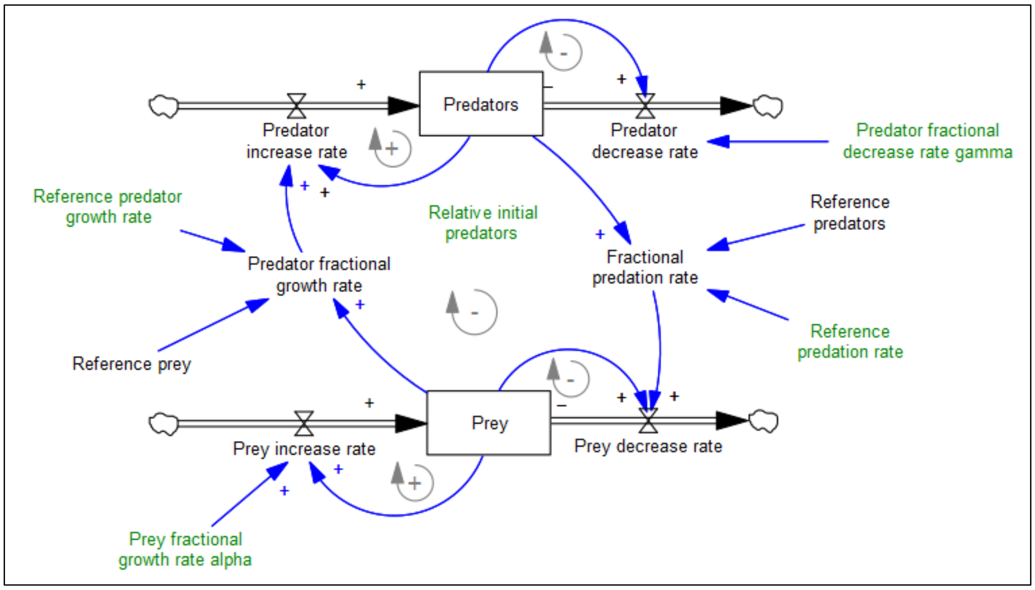
Predators
Predator increase rate
Predator decrease rate
Predator fractional decrease rate gamma
Reference predator growth rate
Relative initial predators
Reference predators
Predator fractional growth rate
Fractional predation rate
Reference prey
Reference predation rate
Prey
Prey increase rate
Prey decrease rate
Prey fractional growth rate alpha
+
-

Figure 1. SFD of the Predator-prey model (model 1)

[1] Based on Lotka–Volterra equations
[2] Acronym for susceptible, exposed, infectious, recovered, and dead

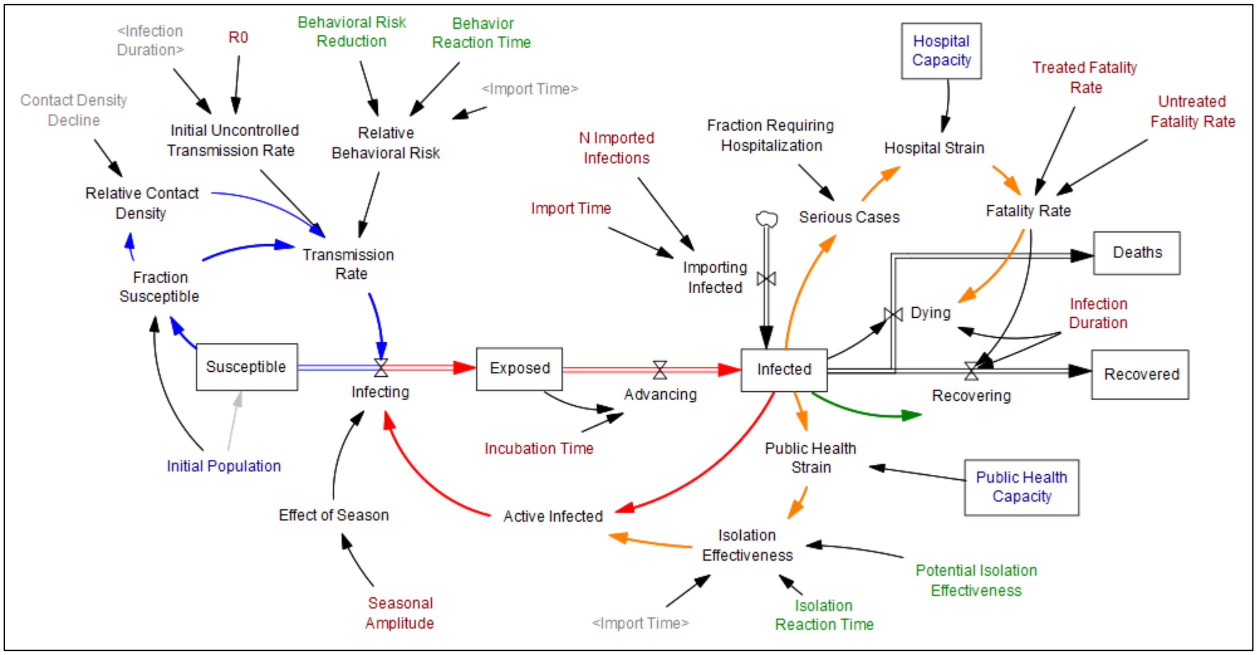

Figure 2. *SFD of SEIRD model (model 2)*

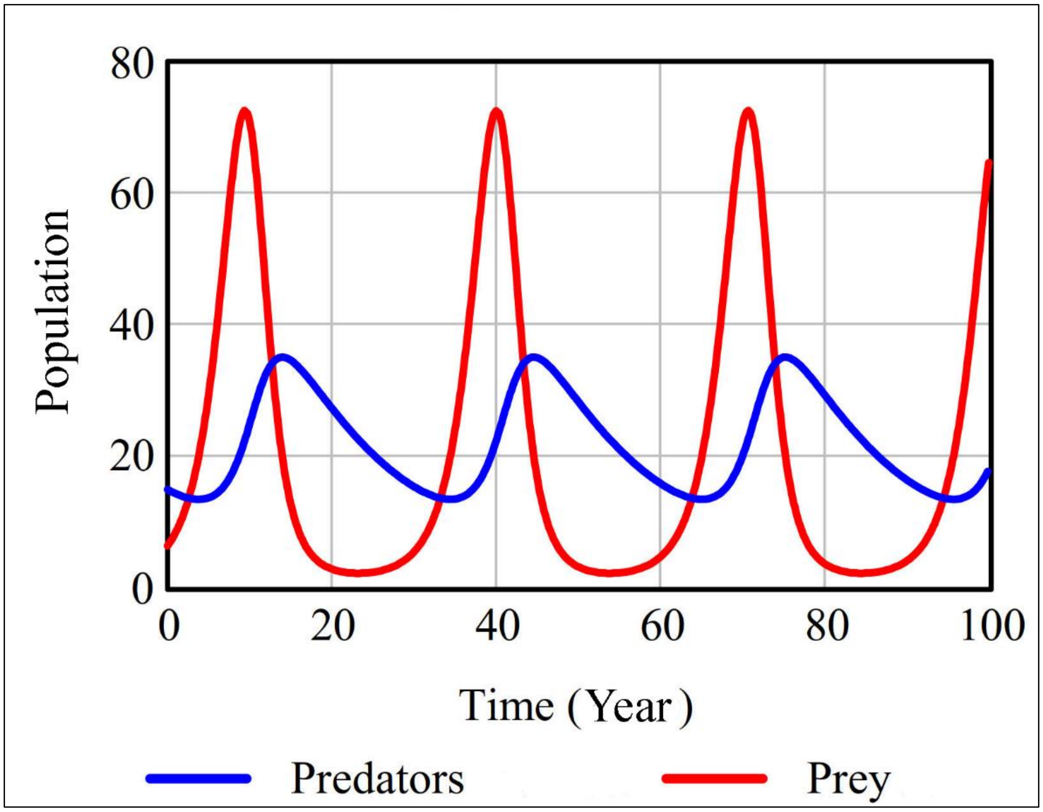

Figure 3. *Output graph of the predator-prey model (graph 1)*

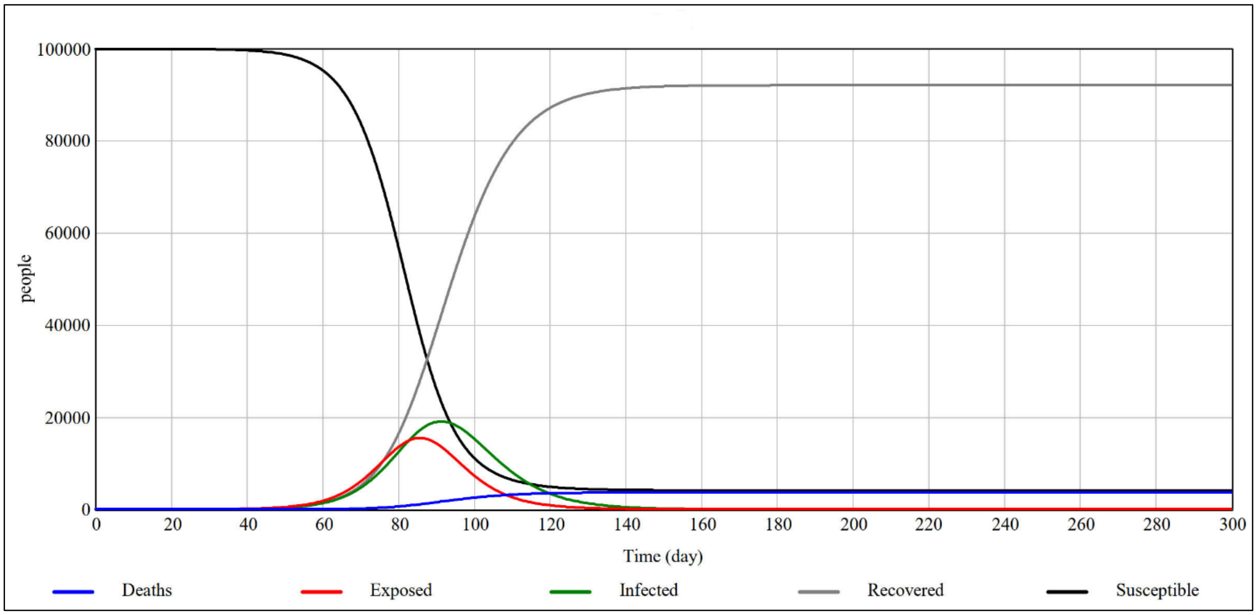

Figure 4. *Output graph of the SEIRD model (graph 2)*

In the next phase, three state-of-the-art artificial intelligence models were selected based on their advanced reasoning capabilities and ability to process and analyze images. An important selection criterion was the ability to interpret visual data. The AI models employed in this research were OpenAI's *o1*, Anthropic's *Claude 3.7 Sonnet*, and Google's *Gemini 2 Flash Thinking (Exp),* which, for convenience, from now on will be called o1, Claude, and Gemini, respectively.

Following this selection, a Python-based implementation was developed to integrate these AI models and prepare them for processing images and generating textual analyses (for technical details, see Appendix 1). A standardized instruction was provided to all models as a prompt to ensure consistent and precise responses:

```
You are a System Dynamics expert, and your task is to interpret System
Dynamic models' output graphs and draw conclusions from them. Describe the
graph and explain its dynamics, causalities, and loops.
```

This prompt was crafted to guide the AI models in analyzing the diagrams effectively, extracting key trends, and generating coherent and structured textual responses. The results can be seen in Appendix 2.

3.2. Data Acquisition

The questionnaire consisted of twelve declarative statements categorized under four evaluation criteria, each broken down into three sub-criteria. These statements were arranged in rows, while the columns represented the outputs of the three AI models under evaluation. This structure created a response framework where participants could provide their assessments using a discrete numerical scale with regard to four main evaluation criteria: ‘accuracy,’ ‘clarity and readability,’ ‘comprehensiveness,’ and ‘creativity’ (for more details on the questionnaire, its questions, and its reliability and validity, see Appendix 3).

The questionnaire was then distributed among a population of SD experts ranging from graduate students and practitioners to professors. Each participant received a brief explanation of the study and evaluation framework; however, the names of the AI models were deliberately omitted to prevent potential bias in responses.

Once all responses were collected, the completed questionnaires were compiled into a final dataset, which served as the basis for evaluating both the AI models and the predefined criteria. Here, the study advanced to the evaluation of AI models using Multi-Criteria Decision-Making (MCDM) methods. This phase enabled a structured assessment of the AI-generated textual analyses, ensuring a rigorous comparative evaluation of their performance in interpreting SD models (Sajjadian *et al.*, 2025).

Following the initial analysis of the collected survey data, it was observed that approximately 64% of participants assigned the highest average score to Claude. Additionally, 22% of respondents believed that o1 outperformed the other two AI models, while 14% considered Gemini to be the best-performing model.

3.3. Evaluating LLMs

The TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) method is a widely used Multi-Attribute Decision-Making (MADM) technique that ranks alternatives based on their proximity to an ideal solution and distance from an anti-ideal solution (Hwang and Yoon, 1981). This method is particularly suitable for ranking Large Language Models (LLMs) according to specified criteria, as it provides a systematic and expert-driven approach to decision-making. Below, we outline the steps involved in applying the TOPSIS method for ranking LLMs.

The first step involves aggregating all the survey matrices obtained from experts into a single decision matrix. Suppose we have a decision matrix of size $m \times n$, where m represents the number of LLMs (alternatives), and n represents the number of evaluation criteria. For example, a 3×4 matrix could represent 3 LLMs evaluated against four criteria.

The decision matrix is normalized using the Euclidean norm to eliminate the influence of different scales among criteria. The normalized value r_{ij} for each element in the matrix is calculated as:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}}$$

The normalized matrix is then weighted by multiplying each normalized value by the corresponding weight of the criterion. The weights are determined using the entropy method

(Shannon, 1948), which is used to calculate the weights of the criteria based on the degree of uncertainty or information content in the data. It is carried out in four steps.

I) Normalization of the decision matrix:

The decision matrix is normalized using the simple normalization method:

$$p_{ij} = \frac{x_{ij}}{\sum_{i=1}^{m} x_{ij}}$$

II) Calculation of the constant:

The constant k, which depends on the number of alternatives m, which in this case is equal to 3, is calculated as:

$$k = \frac{1}{\ln(m)}$$

III) Calculation of entropy E_j each criterion:

$$E_j = -k \sum_{i=1}^{m} p_{ij} \ln(p_{ij})$$

IV) Calculation of weights w_j for each criterion (demonstrated in Table 1):

$$w_j = \frac{d_j}{\sum_{j=1}^{n} d_j}$$

where $d_j = 1 - E_j$ and represents the degree of certainty associated with the j criterion.

	Accuracy	**Clarity and Readability**	**Completeness**	**Creativity and Insightfulness**
$\boldsymbol{w_j}$	0.3041	0.3429	0.0724	0.2804
Rank	2	1	4	3

Table 1. *Weights of criteria*

The weighted normalized matrix is constructed by multiplying the normalized values r_{ij} by their corresponding weights w_j:

$$v_{ij} = w_j \cdot r_{ij}$$

The ideal solution A_j^+ (maximum value for benefit criteria and the minimum value for cost criteria) and the anti-ideal solution A_j^- (minimum value for benefit criteria and the maximum value for cost criteria) are used in calculating the separation measures s_i^+ and s_i^- for each alternative are calculated using the Euclidean distance:

$$s_i^+ = \sqrt{\sum_{j=1}^{n} (v_{ij} - A_j^+)^2}$$

$$s_i^- = \sqrt{\sum_{j=1}^{n} (v_{ij} - A_j^-)^2}$$

The relative closeness C_i of each alternative to the ideal solution is calculated as:

$$C_i = \frac{S_i^+}{S_i^+ + S_i^-}$$

The alternatives (LLMs) are ranked based on the values of C_i (Table 2). The LLM with the highest C_i value is considered the best.

LLMs	**Rank**
Claude	1
o1	2
Gemini	3

Table 2. *LLMs' ranks and weights*

The primary objective of this ranking process was to determine which AI model most closely resembles human-like analytical capabilities when interpreting SD models. For details and technical implementation, see Appendix 4.

3.4. Closing the Loop

Now that the capability of LLMs in explaining and describing SD models' outputs have been assured, attempts to close the loop and streamline the automated modeling process can be made. Utilizing the bot designed by Hosseinichimeh et al. (2024) and considering its scope, which is only to produce CLD, it is first necessary to derive the CLD of one of the models that were implemented in the initial phase. To this end, and to keep the evaluation simple, the first model's (predator-pray) CLD was derived from it, as can be seen in Figure 5.

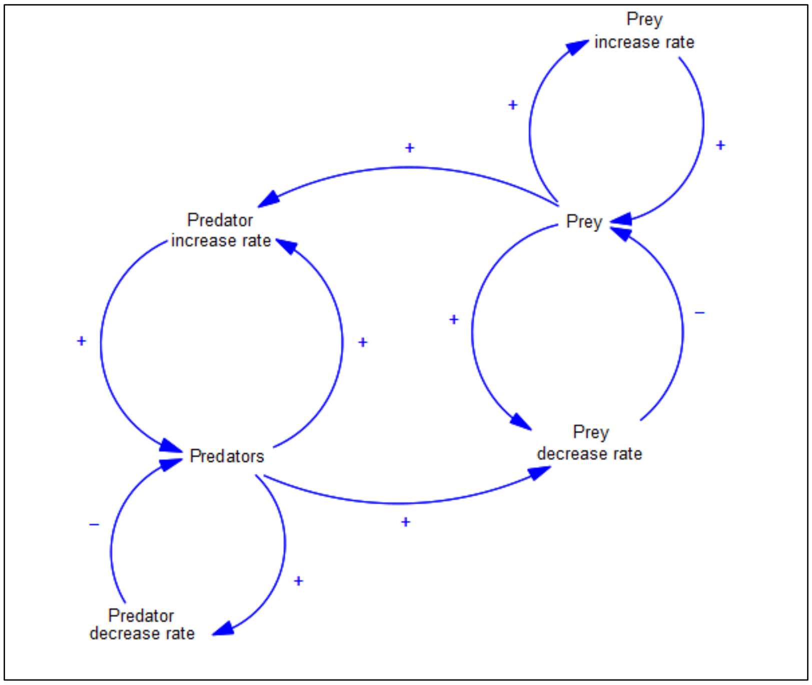

Figure 5. *Predator-prey model's CLD*

Next, another prompt should be engineered, aiming to get a fully covering explanation of the model from the LLM:

```
The graph is the output of a System Dynamic model. Based on this output, explain the hypothetical structure of the initial model by describing its Causal Loop Diagram.
Your explanation should include all variables, their causal relations, and their polarities so that anyone could recreate the Causal Loop Diagram that produced the output graph.
Make sure that your response contains a list of all hypothetical variables, existing causal relation between each pair of variables, and the polarity of the links.
```

Continuing with Claude as the best LLM for the purpose of analyzing SD outputs and feeding it with the new prompt (but the same graph as the initial phase, i.e., Figure 3), a comprehensive report was generated (see Appendix 5) that was directly passed to Hosseinichimeh et al. (2024)'s bot (slightly modified to utilize the more recent version, GPT-4.5 Preview), the result can be seen in Figure 6 (after redrawing in Vensim due to low quality of PyGraphviz's outputs).

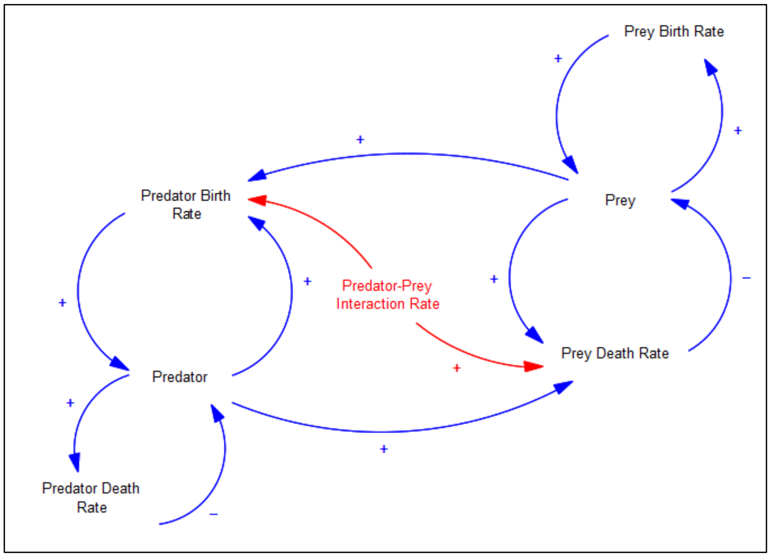

Figure 6. *CLD generated by Hosseinichimeh et al. (2024)'s bot (red color indicates elements that did not exist in the reference graph)*

Apart from remarkable visual similarity and success in reverse engineering, the main model structure and polarities completely match; an objective evaluation of the seamless automated process can be seen in Table 3.

	Links	Loops	Variables
Reference	10	5	6
Bot's output	12	5	7

Table 3. *Comparing the performance of the automated process against the reference model*

4. Results and Conclusion

The findings suggest that certain AI models can effectively analyze SD models at a level comparable to domain experts, providing detailed insights that could facilitate the reconstruction of SD models based on textual interpretations.

Another key insight derived from the survey data was obtained using the Entropy method in MCDM, which assigned a weight to each evaluation criterion. The results indicated that clarity and readability were the most important factors, followed by accuracy, with respective weights of

0.342 and 0.304. Notably, Claude demonstrated the highest performance across all evaluation criteria. Moreover, with an overall average rating of 3.73 out of 5, this study demonstrates that AI-driven textual interpretations of SD models can serve as a valuable complement to SD research.

Most importantly, it was shown that the application of AI and LLM is two-way and can operate in each direction, either from textual description to CLD and SD model structure or *vice versa*, from visual outputs of the SD model to textual interpretation and CLD description, thus, going *beyond* the defined limitations of human-involved SD modeling.

For future research, additional evaluation criteria and a broader range of AI models could be considered. Furthermore, integrating AI-powered analysis directly into SD modeling software (e.g., Vensim) could enhance its functionality. This integration would allow users to receive real-time AI-generated interpretations immediately after executing an SD model, thereby improving analytical workflows in the field.

5. References

Hosseinichimeh N, Majumdar A, Williams R, Ghaffarzadegan N. 2024. From text to map: a system dynamics bot for constructing causal loop diagrams. *System Dynamics Review* **40**(3). https://doi.org/10.1002/sdr.1782

Hu B. 2025. ChatPySD: Embedding and Simulating System Dynamics Models in <scp>ChatGPT</scp> -4. *System Dynamics Review* **41**(1). https://doi.org/10.1002/sdr.1797

Hwang C-L, Yoon K. 1981. *Multiple Attribute Decision Making*. Springer Berlin Heidelberg, Berlin, Heidelberg.

Liu N-YG, Keith D. 2024. Leveraging Large Language Models for Automated Causal Loop Diagram Generation: Enhancing System Dynamics Modeling through Curated Prompting Techniques.

Pagano S, Strumolo L, Michalk K, Schiegl J, Pulido LC, Reinhard J, Maderbacher G, Renkawitz T, Schuster M. 2025. Evaluating ChatGPT, Gemini and other Large Language Models (LLMs) in orthopaedic diagnostics: A prospective clinical study. *Computational and Structural Biotechnology Journal* **28**: 9–15. https://doi.org/10.1016/j.csbj.2024.12.013

Sajjadian M, Firozjaee TT, Kootenaei FG, Firouzian Haji S. 2025. Application of the Analytic Hierarchy Process for the Selection of Appropriate Target of Municipal Wastewater Reuse in Shiraz City. *Water Conservation Science and Engineering* **10**(1): 26. https://doi.org/10.1007/s41101-025-00351-6

Shannon CE. 1948. A Mathematical Theory of Communication. *Bell System Technical Journal* **27**(3): 379–423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x

Veldhuis GA, Blok D, de Boer MHT, Kalkman GJ, Bakker RM, van Waas RPM. 2024. From text to model: Leveraging natural language processing for system dynamics model development. *System Dynamics Review* **40**(3). https://doi.org/10.1002/sdr.1780

Wang Z, Reisert P, Nichols E, Gomez R. 2024. Ain't Misbehavin' - Using LLMs to Generate Expressive Robot Behavior in Conversations with the Tabletop Robot Haru. In *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, New York, NY, USA, 1105–1109.

Zheng Q, Xu S, Du B. 2022. Quantized Guaranteed Cost Output Feedback Control for Nonlinear Networked Control Systems and Its Applications. *IEEE Transactions on Fuzzy Systems* **30**(7): 2402–2411. https://doi.org/10.1109/TFUZZ.2021.3082691

Acknowledgment

We extend our gratitude to Dr. Zarinbal for her meaningful guidance and contribution to the data, which greatly facilitated the research process.