Artificial Intelligence for Engineering Design, Analysis and Manufacturing

www.cambridge.org/aie

Enhancing TRIZ through environment-based design methodology supported by a large language model

Ali Mohammadi and Yong Zeng

Concordia Institute for Information Systems Engineering, Concordia University, Montreal, QC, Canada

Research Article

Cite this article: Mohammadi A and Zeng Y (2025). Enhancing TRIZ through environment-based design methodology supported by a large language model. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, **39**, e12, 1–18
https://doi.org/10.1017/S0890060425000083

Received: 26 June 2024
Revised: 15 March 2025
Accepted: 23 March 2025

Keywords:
TRIZ; Environment Based Design (EBD); Large Language Model; innovation methodology; design problem solving; creative design tools

Corresponding author:
Yong Zeng;
Email: yong.zeng@concordia.ca

Abstract

The utilization of creative design methodologies plays a pivotal role in nurturing innovation within the contemporary competitive market landscape. Although Theory of Inventive Problem Solving (TRIZ) has been recognized as a potent methodology for engendering innovative concepts, its intricate nature and time-consuming learning and application processes pose significant challenges. Furthermore, TRIZ has faced criticism for its limitations in processing design problems and facilitating designers in knowledge acquisition. Conversely, Environment-Based Design (EBD), a question-driven design methodology, provides robust methods and approaches for formulating design problems and identifying design conflicts. Large Language Models (LLMs) have also demonstrated the ability to streamline the design process and enhance design productivity. This study aims to propose an iteration of TRIZ integrated by EBD and supported by an LLM. This LLM-based conceptual design model assists designers through the conceptual design process. It begins by using question-asking and answering methods from EBD to gather relevant information. It then follows the EBD methodology to formulate the information into an interaction-dependence network, leading to the identification of functions and conflicts required by TRIZ. Lastly, TRIZ is used to generate inventive solutions. An evaluation is carried out to measure the effectiveness of the integrated approach. The results indicate that this approach successfully generates questions, processes designers' responses, produces functional analysis elements, and generates ideas to resolve contradictions.

Introduction

In today's highly competitive markets, innovation has emerged as a crucial element for industries, as they can no longer rely solely on quality and quantity. Creative problem-solving skills have been recognized as a fundamental competency for driving innovation (Bertoncelli et al., 2016; Faria, 2019; Nakagawa, 2011). The Theory of Inventive Problem Solving (TRIZ) is a structured approach to problem-solving that facilitates innovation. This design methodology provides a systematic framework for generating solutions and designing products (Savransky, 2000). In comparison to other problem-solving techniques such as brainstorming, mind mapping, lateral thinking, Axiomatic Design, Environment-Based Design (EBD), and morphological analysis, TRIZ has been identified as a more potent logic for idea generation (Dubois et al., 2012; Hernandez et al., 2013; Ilevbare et al., 2013; Mohammadi et al., 2022). Advocates of TRIZ believe that it allows for the generation of more innovative ideas in a shorter time frame due to its structured approach (Belski and Belski, 2015; Gronauer and Naehler, 2016; Ilevbare et al., 2013; Keong et al., 2017; Mohammadi and Forouzanfar, 2021; Taskin et al., 2017). According to Poppe and Gras (Poppe and Gras, 2002), TRIZ is embraced by Western companies as it enables rapid innovation of products, thus enhancing their competitiveness in the current market landscape.

However, learning TRIZ can be challenging and time-consuming, and its application can be complex, potentially leading to misunderstandings and creating barriers for users (Mohammadi et al., 2022). It has been emphasized that designers must undergo thorough and advanced training to effectively utilize TRIZ (Fiorineschi et al., 2018; Fiorineschi et al., 2021). Furthermore, TRIZ is primarily applicable in situations where a problem can be defined by its technical parameters and technical or physical contradictions are apparent. However, identifying these contradictions in the design problems is not always straightforward, especially considering that design problems are inherently ambiguous, particularly in the initial stages (Mohammadi et al., 2022). Fundamentally, while TRIZ excels in idea generation, it lacks the ability to systematically explore design problems and offer designers essential insights before commencing effective idea generation. In contrast, EBD offers a comprehensive methodology for processing design problems, asking the design questions, and acquiring proper design knowledge (Dubois et al., 2012).

This study seeks to introduce a model that integrates EBD into TRIZ to enhance the methodology. Additionally, we utilized the capabilities of a Large Language Model (LLM), specifically ChatGPT, to create pre-coded functions for each stage of the design process, thereby simplifying the application of the design methodology and increasing productivity and efficiency.

The "Related Work" section reviews existing literature on TRIZ, Environment-Based Design (EBD), and Large Language Models (LLMs). This is followed by the "Integrated Support of TRIZ by EBD and LLM" section, which introduces the proposed model and describes the evaluation methods. The "Test Cases and Results" section presents the outcomes of the evaluation, while the "Discussion" section interprets the findings. Finally, the "Conclusion" section summarizes the study's contributions and implications.

Related work

TRIZ

TRIZ was conceived by Genrich Altshuller and his colleagues in the former USSR. Altshuller first introduced the principles of more effective thinking in inventive engineering in 1956. In the 1980s, the method evolved into a systematic creativity toolset known as the "Theory of Inventive Problem Solving" (TRIZ) and later expanded into a "General Theory of Powerful Thinking" (OTSM) and "Lifetime Strategy for a Creative Person" (ZhSTL) in the 1990s, among other variations. The main approach, known as the "Algorithm for Inventive Problem Solving" (ARIZ), also underwent further development from 1965 to 1985 (Mohammadi et al., 2022; Savransky, 2000). However, the development of TRIZ did not initially follow the conventional scientific development process, as all potential developments had to be approved by the founder instead of being peer-reviewed (Cascini, 2012; Chechurin, 2016). Moreover, the constraints imposed by the Cold War era hindered the global dissemination of TRIZ (Fiorineschi et al., 2018). Following Altshuller's passing in 1998, the International TRIZ Association (MATRIZ) acquired all rights to TRIZ in 1999. Subsequently, various organizations with TRIZ expertise developed their own versions of TRIZ, such as ITRIZ, TRIZ+, xTRIZ, CreaTRIZ, and OTSM-TRIZ. In 2007, the book "Hands-on Systematic Innovation for Business and Management" was published (Darrell, 2007), detailing the application of TRIZ in business and management. Additionally, new TRIZ tools such as Root Conflict Analysis (RCA+) (Souchkov, 2005) and Problem Networking, Hybridization (Prushinskiy et al., 2005) were introduced. Furthermore, several efforts were made to integrate TRIZ with modern Quality Management methods such as Quality Function Deployment (Yamashina et al., 2002), Six Sigma (Zhao, 2005), and Axiomatic Design (Borgianni and Matt, 2015).

TRIZ's primary approach to addressing design problems involves identifying contradictions within the problem and resolving them. TRIZ categorizes contradictions into three types (Savransky, 2000): administrative, technical, and physical, and asserts that there is a contradiction at the root of every design problem (Petrov, 2019). The theory provides tools such as SU-Field analysis and functional analysis to deconstruct the design problem and examine the relationships among its elements. Subsequently, the designer identifies contradictions within harmful relationships and employs TRIZ tools, such as the 40 principles for idea generation, to address them (Rantanen et al., 2017).

TRIZ as a design method has been experienced in different areas ranging from engineering to management and even biology. However, most applications of TRIZ, according to Scopus research result analysis, have been defined in engineering. The method has been applied to various engineering problems such as launching new products (Stratton and Mann, 2003), upgrading quality processes (Yamashina et al., 2002), improving the quality of products (Stratton and Mann, 2003) and systems (Cavallucci and Weill, 2001), reducing the environmental footprint (Spreafico and Russo, 2016), designing plastics (Cascini and Rissone, 2004), optimizing energy (Jones et al., 2001), analyzing patents (Sharma et al., 2016), and developing software (Kluender, 2011).

TRIZ is also applicable in management concepts such as e-commerce (Domb and Mann, 2001), crowd management (Pin et al., 2011), innovation management (Livotov, 2008), and marketing (Zouaoua et al., 2010). The theory has also been applied to specific problems in areas including agriculture (Liu and Lu, 2008), biology (Vincent et al., 2005), education (Wits et al., 2010), story-writing (Mohammadi and Forouzanfar, 2021), and drug development (Farber et al., 2018).

Environment-based design (EBD)

EBD (Zeng, 2015) is a design methodology to help designers in the inventive design process. The development of EBD has gone through a few phases. At first, recursive logic was discovered as a new logic for design in 1991 (Zeng and Cheng, 1991), which formulates design's intricate nature of recursion. The second phase was initiated by defining design in mathematical models where the recursive logic was redefined in set theory and a design-governing equation was developed (Zeng, 2002; Zeng and Gu, 1999a; Zeng and Gu, 1999b; Zeng and Jing, 1996). The third phase was marked by definitions of the design problem, design solution, and design knowledge in the structure of the product environment. The new form integrated design requirements, product functions, product behavior, and design solutions into the structure of the environment (Zeng, 2004, 2015; Zeng and Gu, 2001). Thus, the design process was mathematically formulated into an environment evolution process (Nguyen and Zeng, 2012; Razavi et al., 2024; Zeng, 2004, 2015).

In EBD, designers are encouraged to start by analyzing the product environment before delving into the product itself. This approach places a strong emphasis on the product's surroundings and leverages the environment to refine the final product. In this context, the environment encompasses any existing objects excluding the product (Zeng, 2020). To analyze the environment, the methodology employs a design statement in its natural language form. During this phase, EBD utilizes Recursive Object Model (ROM) diagrams to deconstruct problems and specific question-and-answer strategies to gather information (Wang and Zeng, 2009; Zeng, 2008; Yang et al, 2023). Moving to the next stage, EBD processes the information with the assistance of interaction extraction and graph generation. When it comes to interaction processing, the new problem is redefined in the form of active and reactive conflicts (Gutierrez et al., 2014) and interaction dependence networks (Yang and Zeng, 2020). To generate ideas, the methodology utilizes two different strategies based on the types of conflicts (Zeng, 2015). While EBD is a reliable and effective method for design problem analysis, its solution generation phase is not particularly robust (Dubois et al., 2012).

Large language models

Significant strides have been made in the adoption of LLMs for natural language processing tasks. These models undergo thorough training using deep learning techniques on a diverse range of textual big data. The initial model introduced for this purpose

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

was word2vec, capable of performing fundamental tasks such as identifying semantic and syntactic relationships between words. Subsequently, more sophisticated models such as GloVe, ELMo, and BERT were developed. Leveraging advanced transformer architecture and unsupervised learning on large data, OpenAI introduced GPT models, which have produced cutting-edge results in various natural language processing tasks, including language modeling, sentiment analysis, and text completion. Other companies have also launched their specific LLMs: Meta unveiled Llama, Amazon introduced Titan, and Google presented Gemini (Devlin et al., 2019; Pennington et al., 2014; Radford et al., 2018; Wang et al., 2023).

Large language models applications in conceptual design

Numerous efforts have been made to explore the capabilities of the LLMs for the advancement of conceptual design and innovation (Binz and Schulz, 2023; Dortheimer et al., 2024; Khanolkar et al., 2023; Zhu and Luo, 2023). It seems LLMs could possess some level of reasoning that could be applied to the design process (Fang et al., 2024). Zhu and Luo (2023) argued that the limitations in designer knowledge are the primary source of design fixation, and employing LLMs in the design process could enhance knowledge accessibility, thereby fostering the generation of more innovative and effective ideas. Ma et al. (2023) used GPT 3 to generate solutions for a series of design problems and compared them with solutions generated by humans in a crowdsourced process. The result showed that the LLM could generate feasible and useful solutions. Zhou et al. (2024) studied two groups of designers, one using LLM and the other not, and demonstrated that the LLM-Human group generated acceptable solutions in less time. Furthermore, LLMs have been successfully utilized in understanding customer needs, SWOT analysis, manufacturing planning, and brainstorming (Gomez et al., 2024; Han and Moghaddam, 2021; Hu et al., 2023; Just, 2024).

Moreover, it has been claimed that engaging with LLM through a systematic manner and structured prompts can significantly influence the design process (Tian et al., 2024). Several studies have used LLMs in the design process when adhering to specific or structured design methodologies. B. Wang et al. (2023) harnessed LLM to adhere to function behavior structure (FBS) design principles in order to tackle design challenges. Chen et al. (2024c) proposed an LLM-augmented morphological analysis approach to facilitate the efficient generation of innovative design ideas during the conceptual design phase. They used a questioning strategy, which is similar to the question template proposed for eliciting product requirements (M. Wang and Zeng, 2009), as well as the FBS model and Kansei Engineering to guide LLMs to assist designers in the design process. Additionally, Schmidt et al. (2024) discussed the potential of LLMs in enhancing human-centered design. In a similar vein, Koh (2023) exploited the ChatGPT API to leverage the design structure matrix (DSM) to efficiently address design problems. A few studies have also applied TRIZ concepts to LLMs to enhance the design process. Trapp and Warschat (2024) demonstrated that GPT-4 could be used to extract contradictions from patents and provide novel insights into TRIZ principles. Zlotin et al. (2023) introduced software that utilizes TRIZ concepts to prompt ChatGPT in resolving design issues. Another study by Jiang and Luo (2024) proposed AutoTRIZ, a model capable of identifying contradictions and generating design solutions. However, a notable weakness of current studies integrating TRIZ into LLMs is their sole reliance on TRIZ to address design problems, while TRIZ by itself has weaknesses in processing design problems (Ilevbare et al., 2013; Mohammadi et al., 2022).

Moreover, the majority of studies that involve LLMs in the design process share another incapacity – they tend to treat LLMs as a black box. This approach involves providing LLMs with a design problem and expecting them to produce accurate solutions. However, design problems are inherently ambiguous, which makes it challenging for designers to fully grasp the precise nature of the problem in the initial stages of design (Zeng and Gu, 1999a). Therefore, it is unrealistic to anticipate that a machine will offer an exact solution when presented with an incomplete or ambiguous design problem statement. A more effective approach would involve using LLMs as a tool to aid designers at various stages of the design process.

Integrated support of TRIZ by EBD and LLM

In the subsequent sections, we will delve into the methodologies utilized to devise and evaluate the proposed approach. This will encompass an elaborated explanation of the rationale behind utilizing EBD and TRIZ in designing the functions, as well as the evaluation process employed to demonstrate model efficacy. It is our intention that this information will furnish readers with a thorough comprehension of the approach's functions as well as its range of capabilities.

Reasoning flow and functions

Design is a sophisticated process that begins with defining design requirements and concludes with product descriptions (Zeng and Gu, 1999a). Many individuals view the conceptual design phase as a stage for generating solutions. While solutions are the primary output of the conceptual design process, it is essential for designers to not solely focus on solution generation during this phase. In their 2022 study, Yang et al. (2022) discussed the decision-making process in design. This process commences with understanding the correct problem, asking the right questions, collecting pertinent information, developing appropriate knowledge, and ultimately making informed decisions (solutions). Easterday et al. conducted various experiments to establish a logical approach to design. They emphasized that design commences with understanding and redefining the design problem, followed by generating solutions. Zeng and Gu (1999a,b) attempted to apply a science-based approach to study and model the design process. Their study simulated the entire design process and introduced a design-governing equation. This equation clearly outlines that the design process consists of three primary stages: problem redefinition (formulating and processing design problems), synthesis (solution generation), and evaluation (assessing the solutions). In this context, an ideal design process can effectively aid designers in processing and redefining the design problem and, subsequently, generating solutions. Designers can then evaluate the generated solutions against the main requirements, ending the design process if they are satisfied (Zeng and Gu, 1999a).

Several studies have analyzed the strengths and weaknesses of both TRIZ and EBD. The findings indicate that while TRIZ excels in the synthesis stage (solution generation), it lacks systematic tools to assist designers in problem redefinition (Ilevbare et al., 2013; Mohammadi et al., 2022). Conversely, EBD is well-equipped with effective tools for problem processing and knowledge gathering (problem redefinition) but lacks strong ideation tools (Dubois et al., 2012). Therefore, integrating these two design methodologies could establish a comprehensive design framework. Consequently, in this study, we will utilize the modules related to assisting designers in

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

problem redefinition from EBD, as well as the modules related to design synthesis from TRIZ.

In this framework, following EBD's environment analysis, the designer first inputs the design problem and receives a set of questions and answers from the LLM. Then, EBD's principles were supported by employing the LLM to process the answers into the necessary interactions, which leads to a functional analysis based on TRIZ theory according to generated interactions. This is where EBD and TRIZ are integrated. Finally, the designer selects the conflict to be resolved for the LLM to generate ideas based on TRIZ principles to resolve the selected conflict. This corresponds to TRIZ's solution generation. The entire process is reorganized into six LLM-centered stages, as depicted in Figure 1.

In the following sections, each stage will be elaborated upon using the example of designing a house that can fly to demonstrate how each stage works. This kind of open-ended problem, which is seen in various daily and creative situations, has been utilized in presenting EBD methodology and remains a grand challenge for TRIZ.

It is worth noting that this approach was entirely coded using the LangChain framework in Python. About 18 prompts with the architecture of a single prompt or sequential chain have been employed, all of which followed LangChain instruction (https://python.langchain.com/v0.1/docs/get_started/introduction/, Retrieved on June 15, 2024). This means that each prompt exploits a system template and a human template, with the option to utilize an AI template if necessary. The full text of the 18 prompts is provided in Appendix 1.

One intriguing aspect of LLMs is the "temperature" parameter. The LLM temperature controls the distribution of probabilities assigned to possible next words in a sequence. At low temperatures, the model tends to produce more deterministic outputs, while at higher temperatures, the outputs become more varied, creative, and potentially less coherent (Nakaishi and Hukushima, 2022; Suri et al., 2024). The temperature for all of the LLM's API through all the functions was adjusted to 0 to ensure consistent and fixed output, except that the temperature for Stage 6, idea generation, was set to 1 to enhance the LLM's creativity during the process of generating ideas.

Stage 1: Analyzing design problem and generating questions

We input a design problem as a sentence. It is common for a design problem to be vague and unclear, especially in the initial design stages (Zeng and Gu, 1999a). Therefore, it is important to gather information to clarify the design problem for designers. EBD offers a systematic strategy for generating questions in this regard (M. Wang and Zeng, 2009; Yang et al., 2022; Zeng, 2020). This strategy involves breaking down the design problem statement into nouns and verbs. By asking questions starting with "what" related to the noun phrases and "why," "how," "who," "when," and "where" related to the verbs, we can gather information to clarify the design problem. Additionally, asking similar questions about the entire design problem statement without dissecting the nouns and verbs can provide further insight. Therefore, in the initial stage, we utilize EBD's strategy to clarify the design problem and generate relevant questions. Figure 2 illustrates the core internal working strategy of this module.

In Figure 3, we can see a detailed illustration of the stage 1 procedure. The question generation function is comprised of six distinct components. The process begins by analyzing the design problem statement and retrieving nouns using prompt 1, and their corresponding types (human or non-human) using prompt 2. Afterward, if the noun is human, the question "who is + noun?" is generated for that noun. If the noun is non-human, "what is + noun?" is generated. For instance, if the design problem is to "design a house that can fly," the function would extract all the nouns in this sentence using prompt 1. In this case, the only noun is "house." Prompt 2 would then classify this noun as non-human. Based on this classification, prompt 3 would use "what" to generate the question, "What is a house?" (See Table 1, Question 1).

The subsequent stage of the process entails posing questions based on the verbs. To achieve this, the design problem aided by prompt 4 is dissected into its constituent sentences if it comprises of more than one sentence. Prompt 5 then utilizes the interrogatives why, how, when, where, and who to generate questions for each sentence. For instance, in the example of "Design a house that can fly," the questions listed in Table 1, from 3 to 7 and 8 to 12, are generated at this stage.

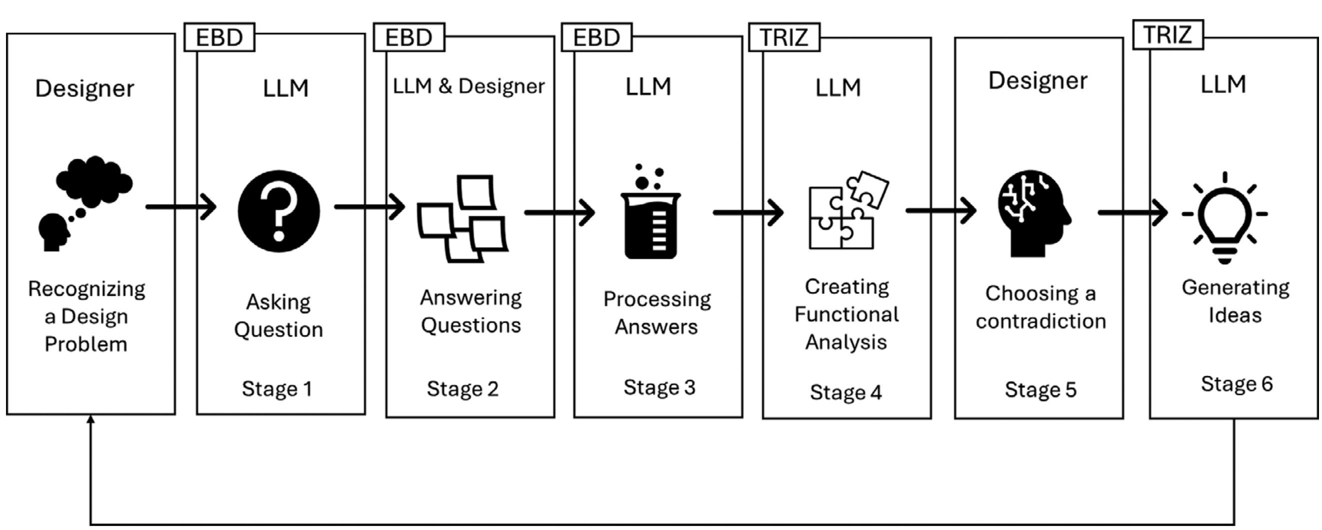

Figure 1. Six stages to integrate EBD, TRIZ, and LLM for resolving design problems.

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

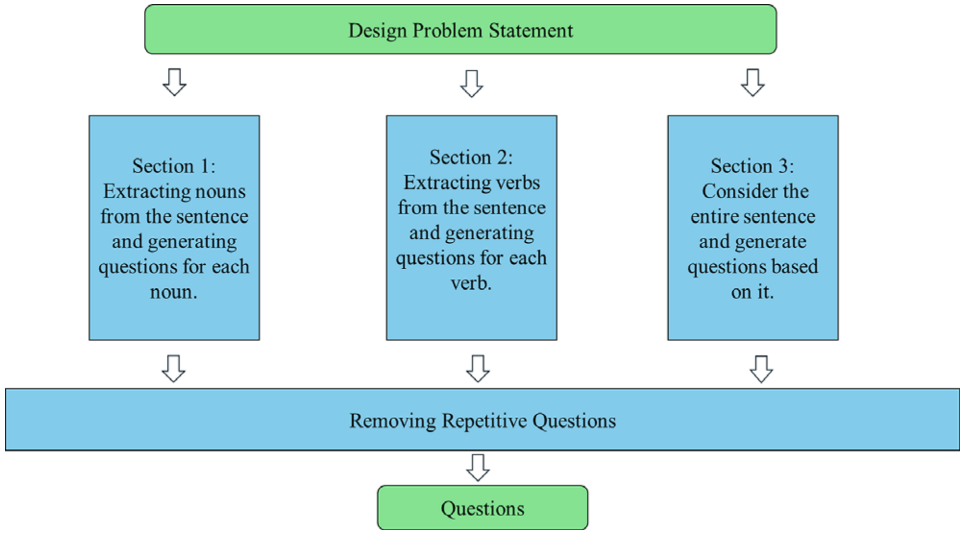

Figure 2. The internal working strategy of the questioning function.

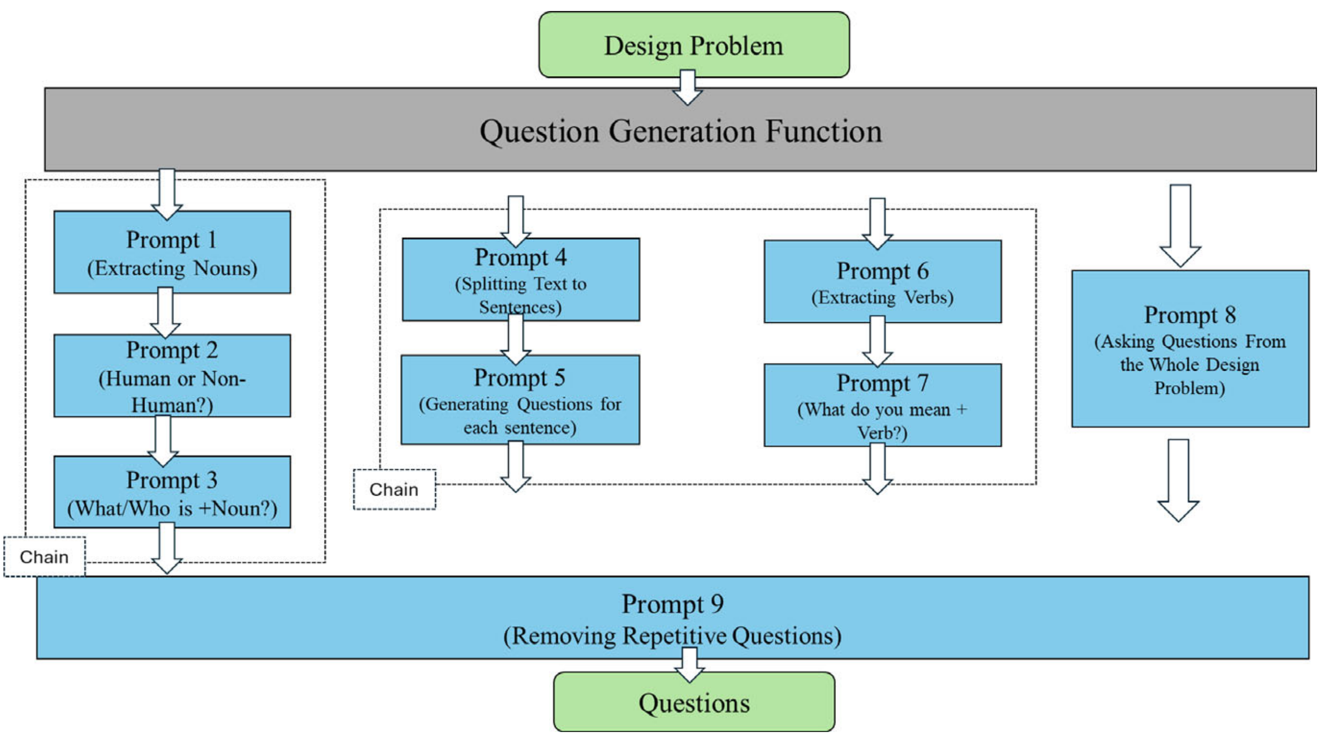

Figure 3. Question generation function.

The third component of this function is focused on the semantic interpretation of the verb and the generation of a corresponding inquiry. To achieve this, prompt 6 is employed to extract the verbs from the design problem. Subsequently, prompt 7 is utilized to inquire about the meaning of the verb. As depicted in Table 1, questions 2 and 13 were formulated in this section.

In this stage, so far, we have deconstructed the "flying house" design problem and formulated inquiries based on its nouns and verbs within the context of the problem. The subsequent part entails posing a series of five "wh" questions that pertain to the entire design problem. This step involves questioning the design problem using why, where, when, who, and how. Prompt 8 would execute this part. For instance, some questions, such as questions 14–18 in Table 1, have been produced by applying this part to the sample design problem, designing a house that can fly. This aligns well with the 5WH questioning algorithm proposed by Wang and Zeng (2009). Chen et al. (2024a) used the similar 5W1H questioning approach with an example of "flying car" in their LLM effort to elicit product requirements.

The final segment of the function, prompted by prompt 9, evaluates all the generated questions and filters out any duplicates. This is necessary because some generated questions may be redundant, and this step ensures that only unique questions are returned.

Table 1. Generated questions by the proposed prompt

Design problem: design a house that can fly	
Num	Generated questions by the LLM
1	What is a house?
2	What do you mean by designing?
3	Where do you design a house?
4	Why do you design a house?
5	How do you design a house?
6	When do you design a house?
7	Who designs a house?
8	Where can the house fly?
9	Why can the house fly?
10	How can the house fly?
11	When can the house fly?
12	Who can fly the house?
13	What do you mean by “design a house that can fly”?
14	Where do you need a house that can fly?
15	When do you need a house that can fly?
16	Why do you need a house that can fly?
17	How do you need a house that can fly?
18	Who is responsible for designing a house that can fly?

Stage 2: Answering questions

During the second stage, it is essential to address the questions generated, which can be divided into two types: those that start with “what” and those that begin with “where,” “when,” “how,” “who,” “why.” While questions that begin with “what” aims to explore the object’s meaning, background, and environment, those that begin with “why,” “when,” “where,” and “how” focus on the specific context of the design problem, such as the location, and time, and specific reasons behind the problem. Typically, questions that begin with “what” can be answered through resources such as the internet, books, or papers, while other questions may require specialized data available at a specific time and location (Zeng, 2020).

Although exploring the design problem situation can help answer “why,” “where,” “when,” and “how” questions, processing a large amount of data to answer “what” questions can be challenging. EBD has a specific template to address “what” questions (Wang and Zeng, 2009; Zeng, 2020). In this context, EBD recommends first determining the meaning of the object. Next, the object’s lifecycle should be identified, along with all its relationships with other objects throughout its entire lifecycle (the object’s environment). This approach provides a structured way to answer questions beginning with ‘what.’ LLM can play a crucial role in defining an object, identifying its lifecycle, and understanding its interactions with other objects throughout the lifecycle. Following the EBD strategy, a module could be developed to effectively handle these types of questions. Figure 4 provides an illustration of the module’s internal operational strategy at this stage.

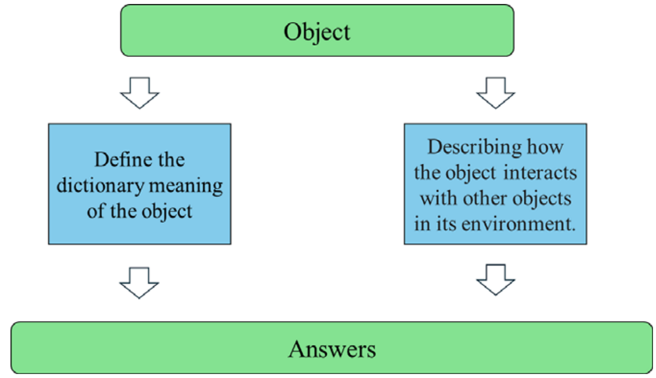

Figure 4. Internal strategy of the module.

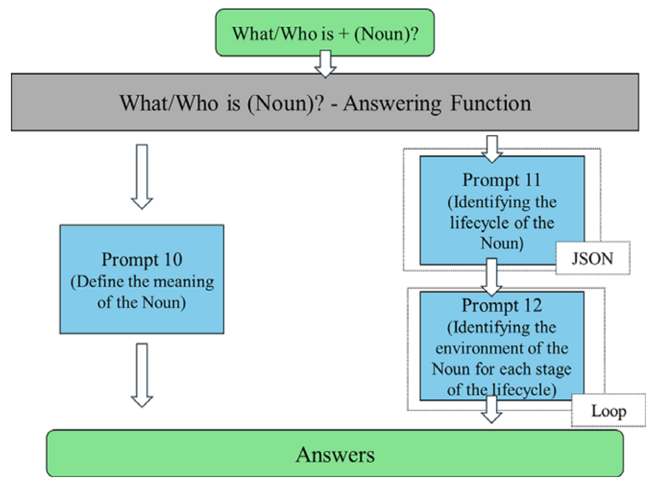

Figure 5. Answering function.

Figure 5 illustrates the details of the approach employed in this stage. The answering function receives the object name as input and leverages prompt 10 to explore the object’s dictionary definition. Next, with the assistance of prompt 11, the object’s lifecycle is extracted and presented in a JSON output format with the aid of the parsed-out chain feature in LangChain. Then, for each stage, the relevant objects and their interactions with the target object are extracted using prompt 12. To illustrate the generated answers, Figure 6 shows the output of the function in response to the “What is a house?” question.

Stage 3: Extracting interactions from the answers

In the previous phase, responses were provided in text form. Now, designers have a wealth of textual information related to the design problem. However, not all of this information is useful in the design process. The question is: how can we analyze the text and extract the important activities occurring in the design problem environment? The systematic approach proposed by EBD involves processing answers. EBD encourages designers to recognize action verbs in the text (verbs that denote an action, such as run, eat, do, warm, etc.) and rephrase each sentence containing an action verb in the form of (subject + action verb + object). This approach condenses the extensive text into the most critical actions taking place in the design problem environment. EBD refers to these reworded sentences as “interactions” (Zeng, 2003, 2020). These interactions could be used for generating performance network or could be used to produce functional analysis since they represent design environment

```
What_OR_Who_is('house')
```

```
 house definition: A house is a building that serves as a dwelling for individuals or families, providing shelter, security, a
nd a sense of home.

 house usually has a specific life cycle and some environment in its lifecycle.

 The first stage of the life cycle of house is Design. In this stage, the environment is classified as the following:
Built environment: The design process involves creating blueprints, floor plans, and 3D models for the house.
Human environment: Designers collaborate with clients to understand their needs and preferences for the house.
Nature environment: Natural elements such as sunlight, wind direction, and topography influence the design decisions for the ho
use.

 The second stage of the life cycle of house is Construction. In this stage, the environment is classified as the following:
1. The built environment of the house is bustling with activity as construction workers lay the foundation.
2. The human environment of the house is filled with architects, engineers, and laborers working together to bring the design t
o life.
3. The nature environment surrounding the house is being impacted by the construction, with trees being cleared and the land be
ing reshaped.

 The third stage of the life cycle of house is Occupancy. In this stage, the environment is classified as the following:
Built environment: The walls are providing shelter and protection to the occupants.
Human environment: The residents are interacting with the furniture and decor inside the house.
Nature environment: The sunlight is streaming through the windows, warming up the rooms during the day.
```

Figure 6. Answering "What is a house" by the model.

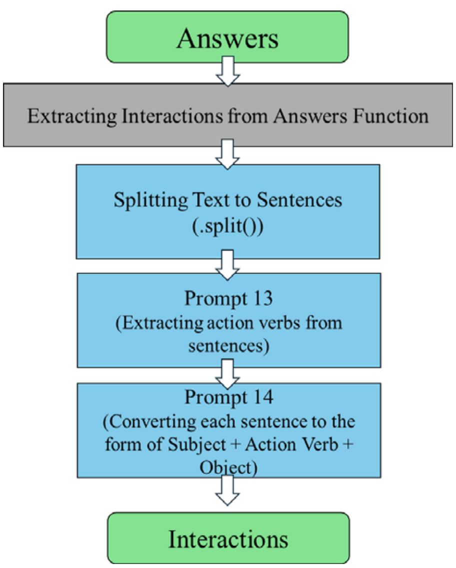

Figure 7. Interaction extraction function.

Table 2. Interaction extraction example

TEXT	A house is a building that serves as a dwelling for individuals or families. A house usually has a specific life cycle and some environment in its lifecycle. The first stage of the life cycle of a house is design. In this stage, the environment is classified as the following: Built environment: The design process involves creating blueprints, floor plans, and 3D models for the house. Human environment: Designers collaborate with clients to understand their needs and preferences for the house. Nature environment: Natural elements such as sunlight, wind direction, and topography influence the design decisions for the house.
Extracted Interactions	○ A house serves as a dwelling for individuals or families. ○ The design process involves creating blueprints, floor plans, and 3D models for the house. ○ Designers collaborate with clients. ○ Designers understand their needs and preferences for the house. ○ Natural elements influence the design decisions.

components and their relationship together (Rantanen et al., 2017; Zeng and Gu, 1999b).

Figure 7 illustrates the details of the procedure used at this stage. All the answers in text format will be input into the function. In the first part, the text will be separated into single sentences using the ". split()" function in Python. Next, each separate sentence will be analyzed. With the help of Prompt 13, the LLM will be triggered to recognize all the action verbs contained in the sentence. If there is no action verb present, the LLM will return with a message indicating that there is no action verb. However, if the sentence does contain an action verb, Prompt 14 will ask the LLM to return the sentence in the format of subject + action verb + object. Prompts 13 and 14 will analyze the sentence in the form of a chain. The output of this function will be a list of interactions extracted from the generated answers. As an example, Table 2 demonstrates the text and extracted interactions for the part of the answers that were generated in the previous stage.

Stage 4: Generating functional analysis based on the extracted interactions

Functional analysis, very similar to the performance network proposed by Zeng and Gu (1999b), is a crucial technique used in TRIZ to help designers break down a given scenario and identify any inconsistencies that may be causing contradictions. To conduct a functional analysis, designers need to follow specific steps. First, they should create a comprehensive list of all components in the design environment. Next, they should examine the relationships between each component and others. Components that cause changes or maintenance in other components are known as "Tools," while the affected components are called "Objects." These relationships can be categorized as either useful or harmful, depending on the designer's perspective (Rantanen et al., 2017). Therefore, in the third step, the designer should analyze the relationships and determine which ones are useful or harmful from their perspective. According to TRIZ principles, behind each harmful relationship lies

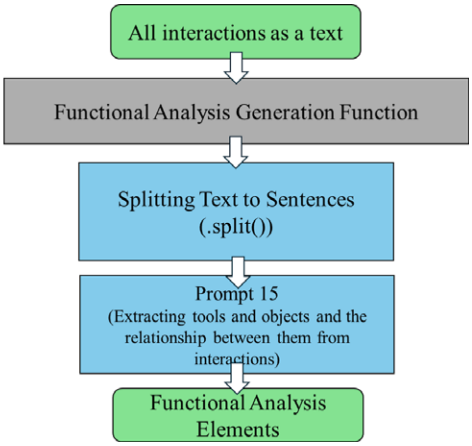

Figure 8. Functional analysis generation.

a contradiction (Savransky, 2000). Consequently, each harmful relationship can be targeted for the next stage to identify a contradiction and develop a solution. To simplify the process, the designer can also create a diagram with arrows and related components, helping them determine the type of relationship on the diagram (Rantanen et al., 2017).

In Figure 8, a detailed procedure is presented for developing a module that systematically generates functional analysis at this stage. The input to this function is a text containing a list of all interactions generated in the previous stage. The text would initially be split into separate sentences using the ".split()" function. Then, for each sentence, the LLM would analyze the interaction and extract the tool, object, and the impact of the tool on the object, as well as the impact of the object on the tool, with the assistance of prompt 15. The output of this function is a parsed JSON containing functional analysis components and their relationships. Using this information, a designer can generate functional analysis. Figure 9 demonstrates an example that shows generating functional analysis components for an interaction.

Stage 5: Selecting a contradiction

As previously discussed, the relationship between two components in a functional analysis can be categorized as either harmful or useful. While one designer may deem a relationship as harmful, another designer might view it as useful based on their knowledge, experience, and perspective (Orloff, 2016; Rantanen et al., 2017). When a relationship is labeled as harmful, it indicates that there is a contradiction for the designer to address. In other words, behind every harmful classified interaction lies a contradiction for the designer that must be resolved (Petrov, 2019; Savransky, 2000). During this stage, the LLM should not intervene; instead, it should allow the designer to select the contradictions based on their perspective.

Therefore, in this stage, designers should first select a harmful relation and shape it in the contradiction format defined in TRIZ (Rantanen et al., 2017): "I want (condition A) for element X because (the reason), and I want the opposite of condition A for element X because (the reason)." For instance, we can choose this contradiction, "I want the material of the house to be light to reduce the house weight, and I want the material of the house to be not light because the light material is not sturdy," as a recognized contradiction in the sample design problem.

Stage 6: Generating ideas for a selected contradiction

TRIZ is a powerful methodology that can be used to generate innovative ideas. The primary tool used in TRIZ for idea generation is a set of 40 principles, which have been extracted from a detailed analysis of thousands of patents (Savransky, 2000). These principles serve as the backbone for idea generation in TRIZ. To initiate the idea-generation process in this stage, an LLM-empowered function is employed to consider the design problem and a selected contradiction. This function then generates ideas related to the TRIZ 40 principles.

To generate ideas for each of the TRIZ principles, prompt 16 is utilized, which runs in a for-loop 40 times. During each iteration, the prompt is provided with the design problem as well as the contradiction and requests LLM to generate ideas for a specific TRIZ principle (from principle 1 to principle 40). Once all the ideas have been generated, prompt 17 eliminates any duplicate ideas. Finally, prompt 18 is used to identify experts who can provide valuable insights to resolve the contradiction. Figure 10 shows the overall architecture of the function. Table 3 provides an overview of the contradiction, some of the generated ideas, and the experts who can assist with resolving the exemplified design problem, "Design a house that can fly."

Evaluation

The model we have developed comprises several distinct functions. As previously explained, the designer inputs a design problem, and LLM attempts to generate questions. It is important to evaluate the effectiveness of the generated questions. Subsequently, the answers are generated, which are then processed, and interactions are extracted. Therefore, the functionality of interaction extraction should also be measured. These interactions are then input into the next function, where functional analysis components are

```
text='''
 A house serves as a dwelling for individuals or families.
'''
```

```
functional_analysis(text)
```

```
 the sentence is:

 A house serves as a dwelling for individuals or families
 the functional analysis is:
 {'Tool': 'house', 'Object': 'individuals or families', 'How tool affect the object': 'provides a dwelling', 'How object affect the tool': 'utilizes the space for living'}
```

Figure 9. Functional analysis generation example.

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

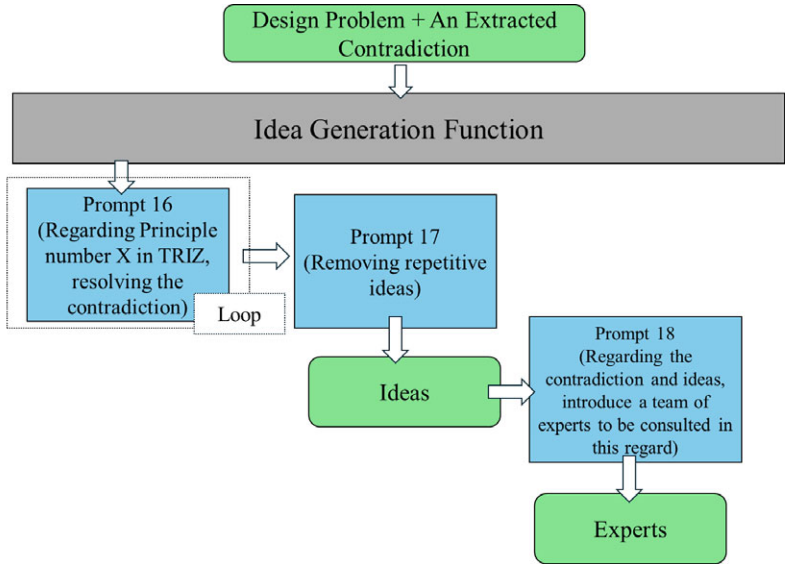

Figure 10. Idea generation function.

Table 3. Generated ideas by LLM

Contradiction	I want the material of the house to be light, but the light material is not sturdy.
Generated ideas	• Creating a house with detachable roof sections that can be removed before takeoff to reduce weight, then reattached when the house lands. • Design the house with inflatable sections that can expand and contract for flight and stability when landed. • The flying house could have a series of internal compartments that can be filled or emptied with weights. • The house could be constructed using carbon fiber panels that are lightweight but incredibly strong, providing both durability and reduced weight for flight. • The flying house could have retractable wings that extend during flight, using aerodynamic principles to help lift the structure off the ground while minimizing the need for heavy materials.
Required experts	Materials Engineer, Structural Engineer, Aerospace Engineer, Architect, Mechanical Engineer, Project Manager, Industrial Engineer

extracted. Hence, the usefulness of the functional analysis function should also be evaluated. Based on the functional analysis, the designer identifies a contradiction, which is then input into the idea generation function to produce new solutions. Consequently, the effectiveness of these solutions should also be considered. In the following section, we will provide detailed explanations of the metrics used to assess each function.

Question-generation evaluation

In the initial phase of the model, the designer utilizes a question generation function to input the design problem and produce inquiries regarding the design problem. Two metrics can be employed to evaluate the effectiveness of the function: correctness, which measures the degree to which the generated questions can provide the designer with valuable knowledge, and comprehensiveness, which assesses whether the function encompasses all the questions that EBD is designed to address (Koh, 2023).

As shown in equation (1), Question Generation Accuracy (QGA) in this stage could be calculated as the number of valid questions generated by the LLM divided by the total number of questions generated. Valid questions are those that provide helpful information for the designer regarding the design problem.

$$\mathrm{QGA} = \frac{\text{Number of valid questions generated by LLM}}{\text{Number of all auestions generated by LLM}} \tag{1}$$

As shown in equation (2), Question Generation Comprehensiveness (QGC) could be calculated as the number of valid questions generated by an LLM that align with EBD questions for a specific design problem divided by the number of questions generated by a human EBD expert for that specific design problem.

$$\mathrm{QGC} = \frac{\text{Number of valid questionsins generated by LLM in line with EBD}}{\text{Number of generated questions by an EBD expert}} \tag{2}$$

Interaction extraction evaluation

Upon answering the generated questions, we utilize the LLM to apply the EBD concept for processing the responses and extracting the interactions. Two key metrics, Interaction Extraction Accuracy (IEA) and Interaction Extraction Comprehensiveness (IEC), could be employed to assess the outcomes of this process (Koh, 2023). In this context, correctness refers to the ratio of valid extracted interactions by LLM to all extracted interactions by LLM, as demonstrated in Equation (3). Valid interactions are defined as sentences containing active verbs. As for comprehensiveness, depicted in Equation (4), it represents the ratio of valid interactions extracted by the LLM from all answers responded to the generated questions for a specific design problem, divided by the number of interactions extracted from those exact answers, albeit this time these interactions were extracted by an EBD expert.

$$\mathrm{IEA} = \frac{\text{Number of valid interactions extracted by LLM}}{\text{Number of all interactions extraceted by LLM}} \tag{3}$$

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

$$\mathrm{IEC} = \frac{\text{Number of valid interactions extracted by LLM}}{\text{Number of interactions extracted by an EBD expert}} \quad (4)$$

Functional analysis generation evaluation

The extracted interactions are inputted into the functional analysis generation function in order to produce the necessary elements for creating a functional analysis. The Functional Analysis Accuracy (FAA) can be assessed using equation (5). Essentially, the accuracy of the function can be defined as the ratio of valid recognized tools and objects extracted from interactions by LLM to all tools and objects extracted by LLM. Valid tools or objects are those that, according to TRIZ, can take a material form, and each tool can impact an object through interaction (Rantanen et al., 2017). Furthermore, Functional Analysis Comprehensiveness (FAC), as indicated in equation (6), can be defined as the ratio of valid recognized tools and objects by the LLM for a specific design problem to the number of recognized tools and objects by a TRIZ expert for that specific design problem (Koh, 2023).

$$\mathrm{FAA} = \frac{\text{Number of valid tools and objects extracted by LLM}}{\text{Number of all tools and objects extracted by LLM}} \quad (5)$$

$$\mathrm{FAC} = \frac{\text{Number of valid tools and objects extracted by LLM from the interactions}}{\text{Number of Valid tools and objects recognized by the TRIZ expert}} \quad (6)$$

Idea generation evaluation

In order to assess the effectiveness of the idea generation function of the model, it would be beneficial to consider the metrics outlined in Shah et al. (2003). These metrics can include measuring the novelty, variety, quality, and quantity of the generated ideas.

Novelty. Novelty is a metric that gauges the level of uniqueness or surprise associated with an idea in comparison to other ideas. To assess the novelty of ideas, we begin by breaking down the problem into its key functions. Subsequently, we analyze each idea by determining which functions it fulfills. Finally, we determine the novelty score of each idea by utilizing equation (7) as outlined in Shah et al. (2003).

$$S_J = \frac{T_J - C_J}{T_J} \times 10 \quad (7)$$

Where the total number of ideas produced for function j, denoted as T_j, and the count of the current solution for that function is denoted as C_J. The ratio is multiplied by 10 to normalize the expression. The novelty score is then multiplied by the weight assigned to each function. The novelty of the idea is graded by summing the products of all weights and novelty scores (Shah et al., 2003).

Variety. Variety serves as a metric for the extent to which the solution space has been explored in the process of generating ideas. The assessment of variety pertains to a collective set of ideas rather than individual ones. To gauge variety, the satisfaction of each function is scrutinized. Ideas are categorized based on their hierarchy and the functions they fulfill and are represented in a tree diagram. Subsequently, the variety score for the set of ideas is computed using equation (8) (Shah et al., 2003).

$$M = \sum_{j=1}^{m} f_j \sum_{k=1}^{n} \frac{S_k b_k}{n} \quad (8)$$

where b_K is the number of branches at level k in a tree diagram; S_K is the score for level k and n is the total number of ideas, and m is the total number of functions;

Quality. In this context, quality refers to the feasibility of an idea and its adherence to design specifications. To assess the quality of ideas, we referred to Table 4 (Rantanen et al., 2017). Evaluators could use the questions in the table to assess each idea. If the solution does not resolve the contradiction, the idea is rejected and receives an overall score of zero. If the solution does resolve the contradiction, it is evaluated using the other questions. The total quality score for each idea is determined by summing the scores from the answers to each question.

Quantity. Quantity refers to the overall number of ideas produced by an individual or a group within a specific timeframe or throughout the entire process of going through all the stages in a particular design method (Shah et al., 2003).

Test cases and results

In this segment, we will conduct an assessment of the proposed model by employing the evaluation process and metrics delineated in "Evaluation" section. The model will be utilized to address three design problems previously resolved and published by TRIZ experts. Following the application of the model to these design problems, we will evaluate its efficacy and juxtapose the outcomes with the solutions presented by the TRIZ experts.

Test cases

To select the design problems, we surveyed five TRIZ textbooks (Bukhman, 2021; Orloff, 2016; Petrov, 2019; Rantanen et al., 2017; Savransky, 2000) and the TRIZ journal website, from which we extracted 11 general design problems that have been already resolved by TRIZ masters and could be used for this study as test cases. Among these, three design problems were found to be solvable through a general LLM without the need for fine-tuning or embedding related papers and books on that specific area. Table 5 provides an overview of the design problems.

We entered the design problem into the initial function and generated the corresponding questions. Subsequently, questions beginning with "what" or "who" were addressed using the proposed function within this model. Other questions, such as where, why, how, and when, were answered based on the available information

Table 4. Evaluating the quality of every generated idea

Idea description	
Questions	The answer of the evaluator
Does the solution resolve the contradiction?	Yes (5 scores), Maybe (3 scores), No (rejecting the whole idea)
Do the harmful features appear?	Yes (0 scores), Maybe (3 scores), No (5 scores)
Are the useful features retained?	Yes (5 scores), Maybe (3 scores), No (0 scores)
Does the system become more complex?	Yes (0 scores), Maybe (3 scores), No (5 scores)
Will we use the current resources?	Yes (5 scores), Maybe (3 scores), No (0 scores)

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

Table 5. Design problems

Design problem	Resources
1. Design a passenger cabin that includes the maximum number of seats while ensuring passenger comfort.	(Sam Carter, 2001)
2. Design a pool suitable for uninterrupted long-distance swimming.	(Orloff, 2016)
3. Properly training sufficient numbers of technical support personnel is difficult	(Rantanen et al., 2017)

in the TRIZ source that published the design problem. The model processed the answers and generated functional analysis based on the interactions. In each case study, we identified the exact contradiction that the TRIZ expert had considered in the source. Consequently, we selected that contradiction and utilized the model to generate ideas. The LLM API used in these three experiments was GPT 3.5, which was chosen for its robust capabilities and cost-effectiveness, making it both accessible and convenient.

Results

In this study, we engaged evaluators to compare the output of our proposed model with the results of TRIZ experts. The evaluators were carefully chosen based on specific criteria, including a comprehensive understanding of design methodologies and tools, practical experience in design projects or publications, and strong critical thinking abilities. The evaluators in this study were selected among graduate students with experience in design projects and publications. All of them possess a deep understanding of design concepts, tools, and methodologies.

Initially, two experts in EBD assessed the model's capacity to produce questions and extract interactions using the metrics outlined in Section titles "Question-generation evaluation" and "Interaction extraction evaluation". The findings of this assessment are presented in Table 6, demonstrating a distinct correlation in the assessment results. It is evident that the model was capable of producing questions with high accuracy and comprehensiveness.

Subsequently, the model's ability to generate functional analysis was assessed by a TRIZ expert based on the criteria outlined in Section title "Functional analysis generation evaluation", and the results were documented in Table 7. While the model was capable of generating a comprehensive functional analysis, its accuracy was around 50 percent. In the initial case study, 310 items were recognized in the extracted interactions, comprising 108 unique items and the remaining being duplicates. Of the 108 unique items, 61 were identified as tools or objects, while 47 could not be classified as such according to TRIZ concepts. In the second case study, 252 items were identified, including 128 unique items and their duplicates. Among the 128 items, 76 were categorized as tools or objects, while 52 could not be classified in either tool or object category. In the third case study, 140 items were identified, with 81 unique items and their duplicates. Out of the 81 items, 47 were classified as tools or objects, while 34 could not be classified according to TRIZ. It is important to note that the elements of functional analysis for the first case study were not identifiable in the published version of the report, preventing the calculation of the comprehensiveness rate for the generation of functional analysis in this particular case study.

Table 7. Functional analysis generation assessment

	Functional analysis generation	
	Comprehensiveness	Correctness
Case Study 1	NAN	0.56
Case Study 2	26	0.59
Case Study 3	5.22	0.58

Following the evaluation, two design methodology experts utilized the criteria outlined in Section title "Idea generation evaluation" – novelty, variety, quality, and quantity – to assess the effectiveness of the ideas generated by both the model and the TRIZ experts. Each evaluator followed the instructions for measuring the ideas' effectiveness according to Shah et al. (2003) and calculated the variety and novelty of the ideas generated by the LLM model and TRIZ experts. It is noteworthy that each evaluator used their perspective to extract the required functions for each design problem. Notably, the model generated a few repetitive ideas in each idea generation for each case study, which were not considered by the evaluators and thus removed from the list of ideas. The variety score, novelty score, and quantity of ideas produced by the model and TRIZ experts are detailed in Table 8. It's evident that both the variety and quantity of ideas generated by the LLM exceed those generated by the TRIZ expert.

Additionally, the two experts evaluated the quality of the ideas based on Section title "Quality". Figure 11 demonstrates box plot diagrams showing the quality of the ideas generated by the model and those generated by TRIZ experts for the exact same case study. In the third case study, the TRIZ expert proposed two primary ideas to address the design issue, but one was rejected by evaluator 1 for

Table 6. Question generation and interaction extraction capability of the model

		Question generation		Interaction extraction	
		Comprehensiveness	Correctness	Comprehensiveness	Correctness
Case study 1	Evaluator 1	1	0.9	0.92	0.84
	Evaluator 2	1	0.91	0.97	0.85
Case study 2	Evaluator 1	0.77	0.87	0.93	0.84
	Evaluator 2	1	0.87	0.94	0.84
Case study 3	Evaluator 1	1	0.82	0.90	0.82
	Evaluator 2	1	0.82	0.94	0.87

Table 8. Ideas assessment results

		Variety scores of ideas		Novelty score mean of ideas		Quantity of ideas	
		LLM model ideas	TRIZ expert ideas	LLM model ideas	TRIZ expert ideas	LLM model ideas	TRIZ expert ideas
Case study 1	Evaluator 1	34	12	2.42	2	20	4
	Evaluator 2	31	7	3.41	2.56	20	4
Case study 2	Evaluator 1	16	4	2.13	1.77	7	2
	Evaluator 2	13	9	1.49	2.07	7	2
Case study 3	Evaluator 1	24	1.5	2.79	1.43	18	1
	Evaluator 2	40	6.5	3.32	2.57	18	2

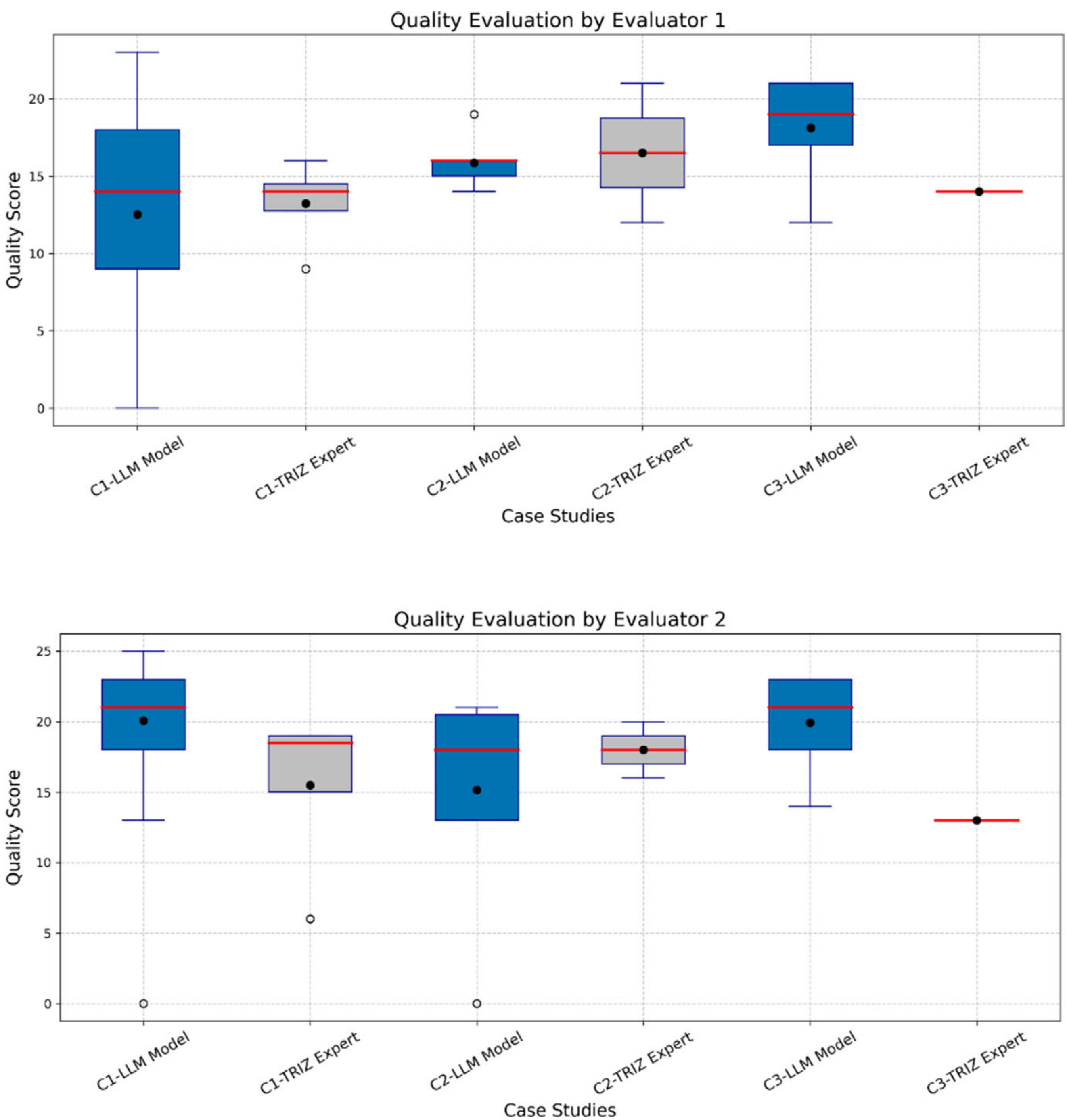

Figure 11. Assessing the quality of ideas.

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

not effectively resolving the contradiction. It is evident that the ideas generated by the LLM vary widely in quality, ranging from very high-quality ideas to low-quality ones.

Discussion

This study aimed to develop a conceptual design framework that integrates TRIZ and EBD based on an LLM. The goal is to aid designers in effectively addressing design problems. The framework involves selecting and simulating various modules from each design methodology. An important question could be raised here: How and why was each module selected and then tailored together?

The selection and adaptation of modules were based on two main criteria. Firstly, the effectiveness of each design methodology in each tool was taken into account. TRIZ experiments demonstrated its strength in problem redefinition and solution generation (Mohammadi et al., 2022), while EBD's studies showed its proficiency in problem-processing and information-gathering (Dubois et al., 2012). As a result, tools such as systematic question and answering, suitable for understanding design problems and information gathering, were incorporated from EBD, and tools such as functional analysis and solution generation, suitable for generating solutions, were selected from TRIZ.

The second criterion for selecting modules in this framework was to ensure the integrity of the framework. It was important to have a seamless progression through the stages and modules so that designers could effectively address the design problem by following the prescribed steps. For example, after inputting design problems and generating questions and answers, designers are presented with a large amount of information in text format. In this case, a tool from EBD for interaction extraction could be used to convert the text into important sentences in a simple subject + verb + object format. The subjects and objects could then be utilized as elements for functional analysis in the next stage, while the verb represents the relationship between these two objects. To ensure the effectiveness of the interaction extraction module within the model, all three case studies were also analyzed without its inclusion. The responses to the questions, without any prior processing, were directly input into the module designated for generating functional analysis. Subsequently, the accuracy and comprehensiveness of the generated functional analysis were evaluated according to the metrics outlined in Section title "Functional analysis generation evaluation" by a TRIZ evaluator. Table 9 illustrates the accuracy and comprehensiveness of the functional analysis produced with and without the interaction extraction module. As demonstrated in the table, both the accuracy and comprehensiveness of the functional analysis diminished in the absence of the module.

Table 9. The accuracy and comprehensiveness of the functional analyses produced with and without the interaction extraction module

	With the interaction extraction module		Without an interaction extraction module	
	Accuracy	Comprehensiveness	Accuracy	Comprehensiveness
Case study 1	0.56	NAN	0.35	NAN
Case study 2	0.59	25.33	0.41	12
Case study 3	0.58	5.2	0.27	1.44

The functional analysis are intended to be based on tools, objects, and the actions that tools exert on objects. The interaction extraction module effectively filters the text to extract the relevant tools, objects, and actions. Without this module, the functional analysis generation process treats each sentence in the text as a potential candidate for functional analysis generation, which leads to a decrease in accuracy. Furthermore, certain phrases in the text may not initially present any apparent tools, objects, or actions; however, they can be restructured to include these elements and be utilized in functional analysis creation. For instance, the phrase "the weather is hot" may not typically qualify for functional analysis due to the absence of distinct tools, objects, or actions. Yet, it can be rephrased as "the sun heats the air," thereby introducing specific tools, objects, and actions that can contribute to the functional analysis. The absence of the interaction extraction module results in the exclusion of sentences that could be reformulated into usable constructs for functional analysis, thereby removing numerous potential tools and objects. This, in turn, contributes to the observed reduction in comprehensiveness when the interaction extraction module is not employed.

All the modules in the model were also precisely evaluated through specific criteria. The results show that the employed LLM effectively generated comprehensive and almost accurate questions regarding the design problem. Additionally, the LLM successfully identified and extracted interactions from the generated answers. However, it was observed that the comprehensiveness of both question generation and interaction extraction outweighed the accuracy. It is as if the LLM is sometimes unable to adhere precisely to the prompts. For example, in some cases, the LLM selected "have or is" as active verbs despite the specified prompt stating that "have, has, am, is, are" are not considered active verbs. This discrepancy was particularly prominent in the functional analysis function. Although the prompts regarding identifying tools and objects were well stated, the accuracy of correctly identifying the tools and objects was calculated in 50 percent channel which is inadequate.

However, the EBD knowledge acquisition stages facilitated the model to encompass a significantly larger number of components (tools and objects) relevant to the design problem, in contrast to the solutions proposed by TRIZ experts. This can be attributed to the systematic acquisition of a substantial volume of data pertaining to the design problem through the EBD stages. Considering a greater number of tools and objects in analyzing the design problem has the potential to uncover more contradictions and streamline the development of a more comprehensive design solution.

The application of the model for idea generation proved to be particularly captivating. GPT 3.5 demonstrated a greater variety, novelty, and quantity of generated ideas in comparison to those generated by TRIZ experts. Moreover, the range of quality in the ideas produced by GPT 3.5 was extensive, encompassing both irrelevant ideas with low-quality scores and a few that exceeded the quality of those generated by the TRIZ experts.

As previously noted, Ma et al. (2023) conducted a study to assess the performance of GPT 3 in the conceptual design process. They employed 12 prompts without a specific methodology or process and tasked the LLM with generating ideas for particular design problems. Their findings indicated that the ideas produced by the LLM were more feasible and useful but less novel and diverse compared to those generated by humans. While our evaluation did not include feasibility and usefulness metrics, we found that the

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

variety and novelty of ideas generated by the LLM surpassed those generated by humans. This could be attributed to the incorporation of design methodologies, their tools, and concepts alongside the LLM in the design process or the use of a more advanced version of the GPT, GPT 3.5. Chen et al. (2024c) utilized LLM to support conceptual design using a specific design approach, morphological analysis. Their findings showed that integrating LLM into the design process could enhance innovation, functionality, and, ultimately, the quality of design ideas, which aligns with the results of our experiments.

In their 2024 study, Jiang and Luo introduced Auto-TRIZ, a design model that uses LLM to automate TRIZ. The model takes problem statements from users as initial inputs and directly generates a solution report by following the TRIZ process. One feature of this model is the automatic generation of contradictions by machines, which occurs without the need for designer supervision. The study's findings demonstrated that Auto-TRIZ streamlines the use of TRIZ and can generate solutions in a short timeframe. However, the study also acknowledged a limitation: the potential for erroneous information in the solutions generated by the model. In our model, we supplemented TRIZ with EBD, particularly in knowledge gathering. Furthermore, the formulation and selection of contradictions are performed based on the supervision of the designer. The majority of ideas generated by our model were assessed as doable, with some even surpassing the quality of ideas generated by human experts.

Chen et al. (2024b) introduced TRIZ-GPT, a model that integrates the traditional TRIZ process to identify design issues, extract TRIZ parameters, address contradictions, and generate solutions. The study revealed that the model is capable of producing solutions similar to those generated by humans, but with a wider range. In our model, we have incorporated three stages prior to utilizing TRIZ to systematically gather knowledge and analyze the design problem environment. As previously mentioned, these stages assist designers in conducting more comprehensive functional analysis and identifying additional contradictions, ultimately leading to the generation of more solutions.

This study is also subject to certain limitations, including a few considered design problems and a limited number of evaluators to assess the solutions. Additionally, not all evaluation metrics may be perfectly aligned with the experiments. For instance, comparing the number of ideas between LLM-generated and TRIZ expert solutions in the textbook could be influenced by bias due to the static nature of the textbook solutions. Moreover, all case studies in this research were exclusively conducted using the GPT3.5 API. A more advanced model could potentially yield results with enhanced accuracy and creativity. Notably, the idea generation phase demands a considerable amount of time as, the LLM meticulously considers each TRIZ principle individually to produce ideas for each one. While this process may require 40 to 50 minutes, it leads to a higher quantity and diversity of generated ideas compared to constraining the model to a few principles. It is plausible that this challenge could be mitigated with a more advanced LLM in the future.

When considering the potential use of more advanced LLMs, an important question also arises: Will this framework still be necessary if future LLMs offer significantly improved performance? It's crucial to take into account the inherent nature of the design problem when addressing this question. The design problem is often ambiguous and not clearly defined for designers in the early stages of the process. In other words, designers themselves may not have a clear understanding of what they want or what the design problem entails, especially in the initial phases of the design process (Nguyen and Zeng, 2012; Y Zeng and Cheng, 1991). Therefore, it may not be practical to expect a highly advanced LLM to provide a definitive solution when the problem itself is not yet fully understood. What is needed is an interactive process between the computer and the designer to gradually clarify the design problem and work toward a solution with the assistance of an LLM. Thus, it seems a systematic design model, like the one presented in this study, is always needed to guide designers in dealing with design problems and generating solutions, even if advanced LLM becomes available in the future.

Our proposed model is specifically crafted to assist designers throughout the design process. It does not operate as a black box, simply churning out solutions to ambiguous design problems. Instead, it actively involves designers at every stage, allowing them to review the output and make revisions as needed. Consequently, designers gain a deeper understanding of the situation. This fosters a high level of interaction between the designer and the LLM. However, this study did not explore the impact of designer interaction with the LLM on design projects. This could be a potential area for future research.

Conclusion

The TRIZ methodology is widely recognized for its effectiveness in generating innovative solutions but is limited in its approach to systematically processing design problems and facilitating knowledge acquisition for designers. Mastering TRIZ can also be time-consuming, intricate, and challenging. In response to these challenges, this study proposes a model that integrates EBD and TRIZ by leveraging LLMs to mitigate these limitations. The findings suggest that the model can receive a design problem and provide pertinent, insightful questions to aid designers in acquiring insight relevant to the problem. Furthermore, the model can process the designer's responses and help generate functional analysis and ideas to resolve contradictions. Importantly, the proposed model does not function as a black box for problem-solving but rather aims to guide designers through the design process stages and foster insights.

The model's performance was assessed using the GPT 3.5 API. The results indicate that the model's comprehensiveness surpasses its accuracy. Evaluation of the generated ideas reveals that machine-generated ideas exhibit greater novelty, diversity, and quantity compared to those generated by TRIZ experts. The quality of the ideas generated by the LLMs spans a wide spectrum, encompassing both irrelevant ideas and ideas of superior quality compared to those produced by TRIZ experts.

It is essential to acknowledge the study's limitations, including a restricted number of case studies and assessments, as well as the sole reliance on testing the model using the GPT 3.5 API. Furthermore, the model's usability was not evaluated. Moving forward, the model's usability will be appraised with novice designers in our future work.

Data availability statement. The program and data is publicly available through the following link: https://github.com/design-zeng/EBD-TRIZ-LLM.

Funding statement. This work was supported by the NSERC Discovery Grant (RGPIN-2019-07048). The funder had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests. The authors declare no competing interests.

AI tools declaration. AI tools were not utilized in the paper content generation, with the exception of Grammarly tools, which were employed to rectify grammatical errors and enhance manuscript readability through their AI capabilities.

References

Belski I and Belski I (2015) Application of TRIZ in improving the creativity of engineering experts. *Procedia Engineering* **131**, 792–797.

Bertoncelli T, Mayer O and Lynass M (2016) Creativity, learning techniques and TRIZ. *Procedia CIRP* **39**, 191–196.

Binz M and Schulz E (2023) Using cognitive psychology to understand GPT-3. *Proceedings of the National Academy of Sciences* **120**(6), e2218523120.

Borgianni Y and Matt DT (2015) Axiomatic design and TRIZ: deficiencies of their integrated use and future opportunities. *Procedia CIRP* **34**, 1–6.

Bukhman I (2021) *Technology for Innovation: How to Create New Systems, Develop Existing Systems and Solve Related Problems*. Springer Nature.

Cascini G (2012) TRIZ-based anticipatory design of future products and processes. *Journal of Integrated Design and Process Science* **16**(3), 29–63.

Cascini G and Rissone P (2004) Plastics design: integrating TRIZ creativity and semantic knowledge portals. *Journal of Engineering Design* **15**(4), 405–424.

Cavallucci D and Weill RD (2001) Integrating Altshuller's development laws for technical systems into the design process. *CIRP Annals* **50**(1), 115–120.

Chechurin L (2016) TRIZ in science. Reviewing indexed publications. *Procedia CIRP* **39**, 156–165.

Chen L, Jing Q, Tsang Y, Wang Q, Sun L and Luo J (2024a) DesignFusion: integrating generative models for conceptual design enrichment. *Journal of Mechanical Design* **146**(11), 111703.

Chen L, Song Y, Ding S, Sun L, Childs P and Zuo H (2024b) TRIZ-GPT: An LLM-augmented method for problem-solving. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (Vol. **88407**, p. V006T06A010). American Society of Mechanical Engineers.

Chen L, Tsang Y, Jing Q and Sun L (2024c) A LLM-augmented morphological analysis approach for conceptual design, in Gray, C., Ciliotta Chehade, E., Hekkert, P., Forlano, L., Ciuccarelli, P., Lloyd, P. (eds.), DRS2024: Boston, 23–28 June, Boston, USA.

Darrell M (2007) Hands-On Systematic Innovation for Business and Management.

Devlin J, Chang M-W, Lee K and Toutanova K (2019) Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies*, volume **1** (long and short papers) (pp. 4171–4186).

Domb E and Mann D (2001) TRIZ in the knowledge era: aAchieving profitable e-commerce. In *PICMET'01. Portland International Conference on Management of Engineering and Technology*. Proceedings Vol. **1**: Book of Summaries. IEEE Cat. No. 01CH37199.

Dortheimer J, Martelaro N, Sprecher A and Schubert G (2024) Evaluating large-language-model chatbots to engage communities in large-scale design projects. *AI EDAM* **38**, e4.

Dubois S, Tate D, Conrardy C, Nguyen TA, de Guioa R and Gaik-Wad A (2012) Comparing the strategies and outputs of designers using algorithm of inventive problem solving, axiomatic design, or environment-based design. *Journal of Integrated Design and Process Science* **16**(3), 89–110.

Fang M, Deng S, Zhang Y, Shi Z, Chen L, Pechenizkiy M and Wang J (2024) Large language models are neurosymbolic reasoners. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Farber B, Martynov A and Kleyn I (2018) Creation of new medical drugs based on TRIZ and computer mathematical modeling. *Annals of Mechnikov's Institute* (4), 15–34.

Faria N (2019) Visualization of the inventive principle of TRIZ to improve problem-solving ability in design processs. In *2019 International Conference on Engineering, Science, and Industrial Applications*. ICESI.

Fiorineschi L, Frillici FS and Rotini F (2018) Enhancing functional decomposition and morphology with TRIZ: literature review. *Computers in Industry* **94**, 1–15.

Fiorineschi L, Frillici FS, Rotini F, Conti L and Rossi G (2021) Adapted use of the TRIZ system operator. *Applied Sciences* **11**(14), 6476.

Gomez AP, Krus P, Panarotto M and Isaksson O (2024) Large language models in complex system design. *Proceedings of the Design Society* **4**, 2197–2206.

Gronauer B and Naehler H (2016) TRIZ as an amplifier for corporate creativity and corporate innovation ability. *Procedia CIRP* **39**, 185–190.

Gutierrez R, Zeng Y, Sun X, Tan S, Deng X and Zhou F (2014) Conflict identification in conceptual design: algorithm and case study. In *Proceedings of the 2014 International Conference on Innovative Design and Manufacturing*. ICIDM.

Han Y and Moghaddam M (2021) Eliciting attribute-level user needs from online reviews with deep language models and information extraction. *Journal of Mechanical Design* **143**(6), 061403.

Hernandez NV, Schmidt LC and Okudan GE (2013) Systematic ideation effectiveness study of TRIZ. *Journal of Mechanical Design* **135**(10), 101009.

Hu Z, Li X, Pan X, Wen S and Bao J (2023) A question answering system for assembly process of wind turbines based on multi-modal knowledge graph and large language model. *Journal of Engineering Design* 1–25. https://doi.org/10.1080/09544828.2023.2272555

Ilevbare IM, Probert D and Phaal R (2013) A review of TRIZ, and its benefits and challenges in practice. *Technovation* **33**(2–3), 30–37.

Jiang S and Luo J (2024) Autotriz: Artificial ideation with triz and large language models. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (Vol. **88377**, p. V03BT03A055). American Society of Mechanical Engineers.

Jones E, Mann D, Harrison DD and Stanton PN (2001) An eco-innovation case study of domestic dishwashing through the application of TRIZ tools. *Creativity and Innovation Management* **10**(1), 3–14.

Just J (2024) Natural language processing for innovation search–reviewing an emerging non-human innovation intermediary. *Technovation* **129**, 102883.

Keong CS, Yip MW, Swee NSL, Toh GG and Tai SC (2017) A review of TRIZ and its benefits & challenges in stimulating creativity in problem solving of pre-university students: a TARUC case study. *Journal of Advances in Humanities and Social Sciences* **3**(5), 247–263.

Khanolkar PM, Vrolijk A and Olechowski A (2023) Mapping artificial intelligence-based methods to engineering design stages: a focused literature review. *AI EDAM* **37**, e25.

Kluender D (2011) TRIZ for software architecture. *Procedia Engineering* **9**, 708–713.

Koh EC (2023) Using a large language model to generate a design structure matrix. arXiv preprint, arXiv:2312.04134.

Liu Q and Lu F (2008) Conflict and innovation practice of china agricultural chain the case study of "Wen's" cultivation service value chain. In *2008 International Conference on Risk Management & Engineering Management*.

Livotov P (2008) TRIZ and innovation management. Available: www.tris-europe.com: http://www.triz.it/triz_papers/2008%20TRIZ%20and%20Innovation%20Management.pdfS.

Ma K, Grandi D, McComb C and Goucher-Lambert K (2023) Conceptual design generation using large language models. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* [Record #1001 is using a reference type undefined in this output style.]

Mohammadi A and Forouzanfar A (2021) Application of TRIZ in literature; an algorithm for systematic story writing based on mega problems. *International Journal of Systematic Innovation* **6**(4), 1–17.

Mohammadi A, Yang J, Borgianni Y and Zeng Y (2022) Barriers and enablers of TRIZ: a literature analysis using the TASKS framework. *Journal of Engineering, Design and Technology* **22**(4), 1206–1230.

Nakagawa T (2011) Education and training of creative problem solving thinking with TRIZ/USIT. *Procedia Engineering* **9**, 582–595.

Nakaishi K and Hukushima K (2022) Absence of phase transition in random language model. *Physical Review Research* **4**(2), 023156.

Nguyen TA and Zeng Y (2012) A theoretical model of design creativity: nonlinear design dynamics and mental stress-creativity relation. *Journal of Integrated Design and Process Science* **16**(3), 65–88.

Orloff MA (2016) *ABC-TRIZ: Introduction to Creative Design Thinking with Modern TRIZ Modeling*. Springer.

Pennington J, **Socher R and Manning CD** (2014) Glove: global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. EMNLP.

Petrov V (2019) *TRIZ. Theory of Inventive Problem Solving: Level 1*. Springer.

Pin SC, **Haron F**, **Sarmady S**, **Talib AZ and Khader AT** (2011) Applying TRIZ principles in crowd management. *Safety Science* **49**(2), 286–291.

Poppe G and Gras B (2002) TRIZ in the process industry. In *European TRIZ Association Conference*, November 2001.

Prushinskiy V, **Zainiev G and Gerasimov V** (2005) *Hybridization: the New Warfare in the Battle for the Market*. Ideation International, Incorporated.

Radford A, **Narasimhan K**, **Salimans T and Sutskever I** (2018) Improving language understanding by generative pre-training. https://openai.com/research/language-unsupervised

Rantanen K, **Conley DW and Domb ER** (2017) *Simplified TRIZ: New Problem Solving Applications for Technical and Business Professionals*. Boca Raton: Taylor & Francis.

Razavi S, **Yang J**, **Yang Y**, **Akgunduz A**, **Bhuiyan N and Zeng Y** (2024) Environment-driven evolution analysis of a product: a case study of braking system evolution. *Journal of Integrated Design and Process Science* **27**(2), 135–150.

Sam Carter DM (2001) The application of TRIZ to economy class aircraft cabin design. *TRIZ Journal*. https://the-trizjournal.com/case-study-application-triz-economy-class-aircraft-cabin-design/

Savransky SD (2000) *Engineering of Creativity: Introduction to TRIZ Methodology of Inventive Problem Solving*. CRC Press.

Schmidt A, **Elagroudy P**, **Draxler F**, **Kreuter F and Welsch R** (2024) Simulating the Human in HCD with ChatGPT: redesigning interaction design with AI. *Interactions* **31**(1), 24–31.

Shah JJ, **Smith SM and Vargas-Hernandez N** (2003) Metrics for measuring ideation effectiveness. *Design Studies* **24**(2), 111–134.

Sharma P, **Khanna RR and Bhatnagar V** (2016) A literature survey and classification framework of TRIZ methodology. In *2016 Second International Conference on Computational Intelligence & Communication Technology*, CICT.

Souchkov V (2005) Root conflict analysis (RCA+): structured problems and contradictions mapping. In *Proceedings of TRIZ Future Conference*, Graz, Austria.

Spreafico C and Russo D (2016) TRIZ industrial case studies: a critical survey. *Procedia CIRP* **39**, 51–56.

Stratton R and Mann D (2003) Systematic innovation and the underlying principles behind TRIZ and TOC. *Journal of Materials Processing Technology* **139**(1–3), 120–126.

Suri G, **Slater LR**, **Ziaee A and Nguyen M** (2024) Do large language models show decision heuristics similar to humans? A case study using GPT-3.5. *Journal of Experimental Psychology: General* **153**(4), 1066.

Taskin B, **Basoglu N and Daim T** (2017) A comparison of two innovation tools: application on smart kitchen design. In *2017 Portland International Conference on Management of Engineering and Technology (PICMET)*.

Tian Y, **Liu A**, **Dai Y**, **Nagato K and Nakao M** (2024) Systematic synthesis of design prompts for large language models in conceptual design. *CIRP Annals* **73**(1), 85–88.

Trapp S and Warschat J (2024) LLM-based Extraction of Contradictions from Patents. In *International TRIZ Future Conference* (pp. 3–19). Cham: Springer Nature Switzerland.

Vincent JF, **Bogatyreva O**, **Pahl AK**, **Bogatyrev N and Bowyer A** (2005) Putting biology into TRIZ: a database of biological effects. *Creativity and Innovation Management* **14**(1), 66–72.

Wang B, **Zuo H**, **Cai Z**, **Yin Y**, **Childs P**, **Sun L and Chen L** (2023) A task-decomposed AI-aided approach for generative conceptual design. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*.

Wang M and Zeng Y (2009) Asking the right questions to elicit product requirements. *International Journal of Computer Integrated Manufacturing* **22**(4), 283–298.

Wits WW, **Vaneker TH and Souchkov V** (2010) Full immersion TRIZ in education. In *Proceedings of the 10th ETRIA World Conference*

Yamashina H, **Ito T and Kawada H** (2002) Innovative product development process by integrating QFD and TRIZ. *International Journal of Production Research* **40**(5), 1031–1050.

Yang J, **Dou Y and Zeng Y** (2023) Environment-based design (EBD): using only necessary knowledge for designer creativity. *Proceedings of the Design Society* **3**, 1675–1684.

Yang J, **Quan H and Zeng Y** (2022) Knowledge: the good, the bad, and the ways for designer creativity. *Journal of Engineering Design* **33**(12), 945–968.

Yang J and Zeng Y (2020) Sustainability: design, making, and technologies. *Journal of Integrated Design and Process Science* **24**(2), 1–3.

Zeng Y (2002) Axiomatic theory of design modeling. *Journal of Integrated Design and Process Science* **6**(3), 1–28.

Zeng Y (2003) Formalization of design requirements. In *Integrated Design and Process Technologies, IDPT-2003*, Austin, Texas.

Zeng Y (2004) Environment-based formulation of design problem. *Journal of Integrated Design and Process Science* **8**(4), 45–63.

Zeng Y (2008) Recursive object model (ROM)—Modelling of linguistic information in engineering design. *Computers in Industry* **59**(6), 612–625.

Zeng Y (2015) Environment-based design (EBD): a methodology for transdisciplinary design+. *Journal of Integrated Design and Process Science* **19**(1), 5–24.

Zeng Y (2020) Environment: the first thing to look at in conceptual design. *Journal of Integrated Design and Process Science* **24**(1), 45–66.

Zeng Y and Cheng G (1991) On the logic of design. *Design Studies* **12**(3), 137–141.

Zeng Y and Gu P (1999a) A science-based approach to product design theory Part I: formulation and formalization of design process. *Robotics and Computer-Integrated Manufacturing* **15**(4), 331–339.

Zeng Y and Gu P (1999b) A science-based approach to product design theory Part II: formulation of design requirements and products. *Robotics and Computer-Integrated Manufacturing* **15**(4), 341–352.

Zeng Y and Gu P (2001) An environment decomposition-based approach to design concept generation. In *Proceedings of International Conference on Engineering Design'01*.

Zeng Y and Jing J (1996) Computational model for design. In *Fourth International Conference on Computer-Aided Design and Computer Graphics*.

Zhao X (2005) Integrated TRIZ and six sigma theories for service/process innovation. In *Proceedings of ICSSSM'05. 2005 International Conference on Services Systems and Services Management, 2005*.

Zhou Z, **Li J**, **Zhang Z**, **Yu J and Duh H** (2024) Examining how the large language models impact the conceptual design with human designers: a comparative case study. *International Journal of Human–Computer Interaction* 1–17. https://doi.org/10.1080/10447318.2024.2370635

Zhu Q and Luo J (2023) Generative transformers for design concept generation. *Journal of Computing and Information Science in Engineering*, **23**(4), 041003.

Zlotin B, **Zusman A and Hohnjec M** (2023) TRIZ-based software as a front-end to AI chatbots. *ITC* **2023**(2023), 30.

Zouaoua D, **Crubleau P**, **Mathieu J-P**, **Thiéblemont R and Richir S** (2010) TRIZ and the difficulties in marketing management applications. In *PICMET 2010 Technology Management For Global Economic Growth*

Appendix 1

Note: Each element enclosed within the curly braces {} denotes a variable. These variables serve as potential inputs for functions or as outputs resulting from another prompt.

Prompt 1	**System template:** You are an AI assistant that gets a text and returns all nouns, pronouns, and noun phrases. Please recognize verbs, but do not return verbs or propositions. **Human template:** Please list all the pronouns, nouns, and noun phrases in the following text: \n {text}
Prompt 2	**Human template:** Consider the following list of nouns, pronouns, and noun phrases. Please specify which ones are human and which ones are non-human. This is the list: {nouns}
Prompt 3	**System template:** You are an AI assistant that gets nouns, pronouns, and noun phrases and their type and asks questions with what or who. for the human nouns, pronouns, and noun phrases, you ask: Who is + (nouns, pronouns, and noun phrases)? \n For non-human ones, you ask: What is + (nouns, pronouns, and noun phrases)? **Human template:** Please consider the following nouns. For each noun you must ask questions with what or who depends on their type. for example, if the nouns are teacher and sandwich, you should return who is teacher? and what is a sandwich? The nouns are: \n {type}
Prompt 4	**System template:** You are an AI assistant who aims to get a text and separate its sentences. **Human template:** I'll provide a text for you. Please examine the given text, first identifying its verb phrases. After that, take note of each verb phrase you have identified in the provided text and return separate sentences for each verb phrase that includes subject + verb phrase + object. \n the sentence is: \n {text}"
Prompt 5	**System template:** You are an AI assistant who aims to help designers to ask questions. **Human template:** I will provide you with text containing subjects, verbs, objects, and sentences. Your task is to ask questions. generate five questions for each sentence using 'where, why, how, when, who. You must follow this structure: 'wh question' + 'auxiliary verb' + 'subject' + 'verb' + 'object' + '?'. This means that for each sentence, you will create one question for each of the five question words. \n\nThe text to be used for this exercise is:\n\n {all_sentences}
Prompt 6	**System template:** You are an AI assistant that gets text. First, recognize all the sentences in the text. Then for each sentence return the verb. **Human template:** Please first consider the sentences in this text and then list the verbs in each sentence: \n {all_sentences}.
Prompt 7	**System template:** You are an AI assistant that gets verbs and asks questions with what. You ask: What do you mean by + the verb? **Human template:** Please consider the following verbs. For each verb you must ask questions with what. for example, if the verb is eat you should return: what do you mean by eating? \n\n list of verbs are: {verb_list}
Prompt 8	**System template:** You are an AI assistant who aims to help designers to extract as much information as possible from a design problem. You get a design problem. You should ask six questions in the following format: 1- what do you mean by + the whole design problem +? 2- Where do you need + the whole design problem +? 3- when do you need + the whole design problem +? 4- why do you need + the whole design problem +? 5- How do you need + the whole design problem +? 6- who is responsible for + the whole design problem +? **Human template:** The design problem is: {text}
Prompt 9	**System template:** You are an AI assistant who can recognize repetitive concepts. You will receive a list of questions; some of these questions may be repetitive or ask about similar concepts. Your objective is to provide a list of unique questions without modifying the wording of the original questions or making any other changes to them. **Human template:** This is the list of questions: \n {question_final_list}
Prompt 10	**System template:** You are a dictionary that could help designers to define {objects}. You would define {objects} clearly and briefly in one sentence. **Human template:** What is {objects}?
Prompt 11	**System template:** You are an AI assistant that could help designers understand the lifecycle of {objects}. You would explain the lifecycle of {objects} clearly and very briefly in just four stages and provide a short explanation for every stage. **Human template:** I would define lifecycle as events that something passes to reach the point that it is. For instance, the lifecycle of an engineer is studying in school, getting good marks, going to college, going to university, and finding an engineering job. What is the life cycle of {objects}'
Prompt 12	**System template:** Every {objects} is related to its environment. We would have three types of environments: 1- built environment, 2-human environment 3- natural environment. Built environment: Everything directly related to the {objects} and made by humans. Human environment: Every human being is directly related to the {objects}. Nature environment: Every natural thing like weather, water, etc, that is directly connected to {objects}. When we want to describe the environment of {objects}, we have to show the relation between {objects} and the environment in sentences. you are an AI assistant that could help designers to understand the environment of {objects}. You would explain the environment of {objects} with single sentences. You would prefer to use action verbs in the sentences. **Human template:** {lifecycle_stage} is one of the stages of the lifecycle of {objects}. Consider {objects} in this stage and determine every part of the environment of {objects} in the separate sentences

(*Continued*)

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press

(*Continued*)

Prompt 13	**System template:** An action verb, also known as a dynamic verb, is a type of verb that describes an action or something that a person or thing does (e.g., “I run”). Action verbs differ from stative verbs, which describe a state of being (e.g., “believe,” “want,” “is,” “are,” “am,” “have,” “has,” “become,” “feel”). Note: Gerund (verb + ing) is not an action verb. While run is an action verb, running is not an action verb. You must return nothing if there is no action verb in the sentence. As an AI research assistant who can analyze sentences, your task is to identify action verbs in a sentence. **Human template:** Please return the action verbs in this text:\n {text}
Prompt 14	**System template:** As an AI assistant, You would get some action verbs. Your task is to identify these action verbs in the text and split the text into independent sentences. Each sentence has only one active verb. You must return nothing if there is no action verb in the sentence. Your output is only a sentence without extra explanation. **Human template:** Regard these action verbs:\n {action_verbs}.\n In the following text, identify these action verbs and return a sentence with this format: subject + action verb + object. This is the text:\n {text}
Prompt 15	**System template:** You are an AI assistant that could analyze a sentence and specify a tool, an object, and the impact of tools on the object and the impact of objects on the tool. **Human template:** Every action has a tool and an object. An action means the tool does something that causes the object to change or to be maintained after the change. Thus, in every action, there are two material objects (one is a tool, and the other is an object) that interact with each other. In This interaction, the tool must impact the object. If there is no impact on the object, we can conclude that there is no action. The object that impacts the other one is called a tool, and the one that is affected is the object. Please consider these sentences:\n {text} \n. Please clarify what the tools and objects are in every sentence. Please also clarify how the tool affects the object and how the object affects the tool
Prompt 16	**System template:** You are an AI assistant who can understand a design problem and its contradictions and generate ideas regarding TRIZ principles. The output text should clearly state generated ideas and provide an example for each idea to better explain them. Your output is only the ideas and examples without extra explanation. **Human template:** TRIZ is an acronym for “Theory of Inventive Problem Solving.” It is a problem-solving methodology that originated in Russia and was developed by Genrich Altshuller and his colleagues in the mid-20th century. The 40 Principles of TRIZ are a set of guidelines formulated by Genrich Altshuller, the founder of TRIZ. These principles are based on his analysis of thousands of patents across various industries. They serve as strategies for overcoming contradictions and generating inventive solutions for problem-solving and innovation. According to TRIZ, in every design problem, there is a contradiction that would make this design problem. Now, I would like to give you a design problem with its contradiction. Please resolve it using Principle {number} of TRIZ. \n Here’s the problem:\n{problem} \n This is the contradiction recognized in the problem:\n {contradiction}.
Prompt 17	**System template:** You are an AI assistant who is an expert in removing repetitive ideas. I will provide you with a list of ideas and their examples. Some of these ideas may be repetitive, so please remove any duplicates and return a list of unique ideas with their examples. **Human template:** Please consider the list of ideas and only return a list of unique ideas. \n list of ideas: \n {ideas}
Prompt 18	**System template:** You are an AI assistant who is an expert in determining which experts are required in a design team to resolve a design problem. You get the design problem, a contradiction in the design problem, and generated ideas. You would return the experts needed to validate and implement the ideas to resolve the design problem and create a novel product. **Human template:** This is the design problem: \n {problem} \n\n this is the contradiction inside the design problem: {contradiction} \n\n this is the generated ideas:{final_ideas}.

https://doi.org/10.1017/S0890060425000083 Published online by Cambridge University Press