UNIVERSITÀ DI TRENTO

Department of Information Engineering and Computer Science

Master's Degree in
Artificial Intelligence Systems

Final Dissertation

PhysIQ: A Benchmark for Physical Reasoning in VLMs

Design, Implementation, and Evaluation of Interactive Tasks for Testing AI's Understanding of Simplified Physical Interactions

Supervisor
Jacopo Staiano

Student
Massimo Stefan

Academic year 2023/2024

Contents

Abstract

Vision–Language Models (VLMs) have rapidly progressed in tasks such as object recognition and captioning but continue to struggle with deeper forms of physical reasoning. This thesis introduces *PhysIQ*, a novel benchmark designed to rigorously test these models' abilities to understand and predict the outcomes of physically grounded scenarios in a two-dimensional puzzle environment. Building upon the PHYRE framework, PhysIQ provides a structured dataset comprising puzzles of varying complexities, each requiring the model to reason about forces, collisions, and object placements in order to make two target objects maintain contact for a specified duration.

We propose five complementary evaluation tasks—ranging from simple sanity checks to iterative, interactive puzzle-solving—to diagnose both the breadth and depth of a model's physical understanding. We systematically generated valid and invalid "proposals" (i.e., inserted balls or other objects) for each puzzle, enabling fine-grained assessment of a model's precision and capacity for nuanced reasoning. Our experiments with state-of-the-art VLMs (e.g., GPT-4o, Claude 3.5 Sonnet, Qwen2.5, Gemini 2.0 Flash) reveal pronounced limitations: although some models perform modestly well in static binary classification (especially when given relevant examples), most fail to adaptively refine solutions through iterative feedback or to accurately rank proposals requiring subtle comparative judgments.

These findings underscore the gap between superficial pattern-matching and genuinely "grounded" understanding of physical principles. By pinpointing where and why models falter, PhysIQ illuminates key avenues for future improvement—such as integrating physics-based inductive biases or hybridizing symbolic simulators with data-driven neural networks. Overall, our benchmark offers a rigorous, puzzle-focused lens on the elusive challenge of teaching AI to reason about the physical world.

1 Introduction

In recent years, Vision Language Models (VLMs) have demonstrated remarkable progress in understanding and reasoning about visual content. These models can recognize objects, describe scenes, answer questions about images, and even engage in complex conversations grounded in visual information. As these systems continue to evolve and find applications in various domains—from healthcare and autonomous vehicles to education and assistive technologies—it becomes increasingly important to rigorously evaluate their capabilities and limitations. Among the various cognitive skills required for real-world deployment, physical reasoning stands out as particularly critical yet insufficiently benchmarked in current VLMs.

Physical reasoning encompasses the ability to understand, interpret, and predict how objects behave according to the laws of physics. Humans develop an intuitive understanding of physics from an early age—we can anticipate how objects will fall, collide, balance, or interact with their environment. This intuitive physics forms a cornerstone of our daily decision-making, enabling us to navigate the physical world safely and efficiently. For artificial intelligence systems intended to operate in or reason about the physical world, comparable capabilities are essential. A robot manipulating objects, an AI assistant interpreting DIY instructions, or a system analyzing sports videos all require robust physical reasoning to be truly effective.

Despite the importance of physical reasoning, existing benchmarks for VLMs often fail to comprehensively evaluate this particular aspect of intelligence. While benchmarks exist for general visual understanding, common sense reasoning, and specialized domains like mathematics or coding, physical reasoning has remained relatively underexplored. The benchmarks that do touch on aspects of physical reasoning typically do so in a limited manner, focusing on narrow subsets of physical phenomena or evaluating them through indirect proxy tasks that may not fully capture the breadth and depth of physical reasoning capabilities.

This thesis introduces PhysIQ, a novel benchmark specifically designed to evaluate physical reasoning capabilities in Vision Language Models. PhysIQ addresses the aforementioned gap by providing a comprehensive, systematic, and fine-grained assessment framework that targets various dimensions of physical reasoning through visually-grounded tasks. The benchmark consists of carefully curated image-based problems that require models to demonstrate understanding of concepts such as gravity, momentum, stability, support, containment, object permanence, and causal physical relationships.

For PhysIQ, we adapted puzzles from the PHYRE (PHYsics REasoning) dataset, rather than creating new templates from scratch. We specifically chose PHYRE due to its inherent granularity in difficulty levels, the systematic approach used for template puzzle differentiation, and the simplicity of its environment. PHYRE provides a collection of physics puzzles represented as 2D simulations with clearly defined goals and constraints, making it an ideal foundation for our benchmark.

Our methodical approach focused on adapting these existing puzzles within a framework that ensures both breadth and depth in the evaluation of physical reasoning. We leveraged PHYRE's predefined dimensions of physical reasoning which are grounded in cognitive science literature and work on intuitive physics. The varying complexity levels within PHYRE's puzzle templates were particularly valuable, as they allowed PhysIQ to differentiate between basic and more sophisticated reasoning capabilities. All visual elements in our benchmark are derived from PHYRE's simulation environment, maintaining consistency throughout the evaluation process.

To validate PhysIQ, we conducted extensive experiments with a diverse set of state-of-the-art Vision Language Models, including but not limited to GPT-4o, Claude 3.5 Sonnet, Gemini 2.0 lite and Qwen 2.5-vl-72b. Our experimental methodology included not only quantitative performance assessment but also qualitative analysis of model responses to gain deeper insights into their reasoning patterns, failure modes, and potential biases. We developed rigorous evaluation metrics that consider both the correctness of predictions and the quality of explanations provided by the models, recognizing that physical reasoning often involves explaining the "why" behind physical phenomena.

Our findings reveal several interesting patterns in the physical reasoning capabilities of current VLMs. While these models demonstrate a moderate performance on certain types of physical reasoning tasks, particularly those involving static scenes and common everyday scenarios, they struggle significantly with dynamic predictions and scenarios that require multi-step physical inference. We observed substantial variability across different models, with some exhibiting particular strengths in specific dimensions of physical reasoning while showing weaknesses in others. Notably, our results indicate that performance on general vision-language benchmarks does not necessarily correlate with performance on physical reasoning tasks, suggesting that physical reasoning represents a distinct capability that requires dedicated evaluation.

Beyond benchmarking existing models, PhysIQ offers valuable insights for the future development of VLMs with enhanced physical reasoning capabilities. Our analysis identifies specific areas where current models fall short and suggests potential directions for improvement, such as incorporating physics-based inductive biases, augmenting training data with physically-grounded scenarios, or developing specialized architectural components for physical reasoning. These insights could guide researchers and developers in creating the next generation of VLMs that can better understand and reason about the physical world.

The contributions of this thesis are threefold. First, we provide a comprehensive benchmark for evaluating physical reasoning in VLMs, filling an important gap in the landscape of AI evaluation. Second, we present a detailed analysis of the physical reasoning capabilities of current state-of-the-art models, offering insights into their strengths and limitations. Third, we identify promising directions for future research to enhance physical reasoning in vision-language systems. Through these contributions, we aim to advance the development of AI systems that can better understand and interact with the physical world.

In the subsequent chapters, we delve deeper into the background of VLMs and physical reasoning (chapter 2), review related work in this domain (chapter 3), describe our methodology for creating and validating the PhysIQ benchmark (chapter 4), present our experimental results (chapter 5), discuss their implications (chapter 6), and conclude with reflections on the future of physical reasoning in AI systems (chapter 7). The complete resource, including datasets, evaluation code, and detailed results, is available in our public GitHub repository at `https://github.com/Itakello/PhysIQ`.

2 Background

2.1 Foundations of Physical Reasoning in AI

Physical reasoning is broadly defined as the ability to predict, interpret, or plan interactions in a world governed by physical laws. In artificial intelligence (AI), these interactions typically concern how objects move, collide, support or obstruct each other, and whether certain configurations are stable or likely to fail. Historically, researchers have approached physical reasoning through two primary routes: *classical physics simulation* and *learned approximations*. Classical simulation engines model real-world physics through explicit, hand-engineered equations, whereas learned approximations rely on data-driven representations that map observed scenarios directly to predictions. Both strategies remain instrumental in applications such as robotics (e.g., for grasping and navigation), virtual assistants (e.g., for scene understanding or object placement suggestions), and any AI system that engages with an environment where physical causality underpins behavior.

2.1.1 Classical Simulations vs. Learned Approximations

Classical Simulations

Classical physics simulations are computational frameworks that implement established physical laws and principles through explicit mathematical formulations. These engines—such as Box2D, Bullet, MuJoCo, and Pymunk—encode Newtonian mechanics, rigid body dynamics, collision detection, and response algorithms directly into their architecture. They operate by numerically integrating equations of motion, typically using methods like Euler integration or more sophisticated Runge-Kutta schemes, to calculate how objects move and interact over discrete time steps.

The fundamental strength of classical simulators lies in their *high fidelity* to physical principles. When properly calibrated, these systems can produce remarkably accurate predictions for a wide range of scenarios, from simple projectile motion to complex multi-body interactions. Their outputs are also highly *interpretable*—each state transition can be traced back to specific physical laws, making it straightforward to explain why an object followed a particular trajectory or why a structure collapsed in a certain way. This interpretability proves invaluable for debugging, verification, and building trust in safety-critical applications.

Furthermore, classical simulators exhibit *robustness* across diverse scenarios. Unlike data-driven approaches, they do not suffer from distribution shifts or out-of-distribution failures, provided the scenario remains within the scope of the implemented physics. A ball rolling down an inclined plane follows the same principles whether the plane is wooden, metal, or glass—the simulator simply adjusts parameters like friction coefficients accordingly.

However, these advantages come with significant trade-offs. The *computational cost* of high-fidelity physics simulation can be prohibitive, especially for real-time applications or complex environments with many interacting objects. Detailed simulations of cloth, fluids, or granular materials may require minutes or hours of computation for just seconds of simulated time. Additionally, classical simulators have a *limited scope* defined by their implemented equations. Phenomena like complex fluid dynamics, material deformations, or quantum effects often require specialized simulators or significant approximations. Even seemingly simple real-world effects—such as the way a coffee cup slides differently on a dry versus slightly wet surface—can be challenging to model accurately without extensive parameter tuning.

Learned Approximations

In contrast to classical simulations, learned approximations leverage machine learning techniques to infer physical dynamics directly from data, without explicitly encoding physical laws. These approaches typically employ neural network architectures—ranging from convolutional networks for image-based predictions to graph neural networks for object-centric reasoning—trained on datasets of physical interactions, either from real-world observations or from classical simulators themselves.

The *Interaction Network* approach mentioned earlier exemplifies this paradigm, where objects and their relations are modeled as nodes and edges in a graph, with learned functions approximating the dynamics between them. Other examples include video prediction models that forecast future frames based on observed sequences, or dynamics models that predict the next state of a system given its current configuration and applied actions.

The primary advantage of learned approximations is their *computational efficiency.* Once trained, these models can generate predictions orders of magnitude faster than running a full physics simulation, making them suitable for real-time applications like robotic control or interactive virtual environments. They also demonstrate remarkable *adaptability* to noisy, partial, or complex real-world data. A neural network trained on real-world video sequences naturally learns to handle occlusions, sensor noise, and other practical challenges that might require explicit modeling in a classical simulator.

Furthermore, learned models can capture *flexible* and nuanced physical behaviors that might be difficult to formulate mathematically. For instance, they might implicitly learn the complex ways in which a piece of cloth folds and unfolds, or how granular materials like sand flow and settle—phenomena that classical simulators often struggle to model efficiently without significant domain expertise.

However, these benefits come with their own limitations. Learned approximations exhibit strong *reliance on training data*, often failing dramatically when confronted with scenarios that differ substantially from their training distribution. A model trained exclusively on rigid body collisions might produce nonsensical predictions when asked about fluid dynamics or elastic deformations. Their failure modes are also typically more *opaque* than those of classical simulators—when a prediction goes wrong, it can be challenging to diagnose why or to guarantee that a simple fix will resolve the issue. This lack of interpretability poses challenges for safety-critical applications where understanding failure cases is essential.

Furthermore, learned models typically provide a lower degree of guaranteed accuracy compared to classical simulators. Although they are effective at capturing the average behavior of physical systems, they often struggle to deliver precise and deterministic predictions over extended forecasting horizons, as minor errors can accumulate and lead to significant deviations over time.

Complementary Approaches

In practice, classical simulations and learned approximations often serve complementary roles in physical reasoning systems. Classical simulators provide reliable ground truth for training data generation and benchmark evaluation, while learned models offer efficiency and adaptability for real-time applications. Hybrid approaches are increasingly common, where learned components augment classical simulators (e.g., by predicting contact points to accelerate collision detection) or where classical physics principles are incorporated as inductive biases in neural network architectures (e.g., through physics-informed neural networks or differentiable physics engines).

As the field advances, the boundary between these approaches continues to blur, with researchers exploring ways to combine the strengths of both paradigms—the physical grounding and interpretability of classical simulations with the flexibility and efficiency of learned approximations—to build more robust, generalizable physical reasoning systems.

2.1.2 Causal Reasoning and Mental Simulations

A longstanding hypothesis in cognitive science posits that humans rely on approximate ”mental simulations” to reason about physical events, rather than using purely symbolic or analytical methods. This perspective is captured by the *Intuitive Physics Engine* (IPE) framework introduced in [9]. There, human judgments about everyday scenarios—whether a pile of blocks will topple, a ball will roll off a table, or a container will support its contents—are explained more accurately by sample-based forward simulations than by static heuristics. According to this view, people mentally run multiple ”what-if” scenarios, adjusting parameters like object mass or initial velocity, then aggregating likely outcomes to form robust predictions.

Over the last decade, AI systems have attempted to replicate this mental simulation ability, especially in simplified 2D or 3D environments. Early research indicated that purely deterministic or purely rule-based approaches often failed to capture the inherent uncertainties of real-world physics, such as friction or slight changes in object positioning. Incorporating a *probabilistic* element into simulated rollouts enabled closer alignment with human-like reasoning [9]. For instance, small perturbations in an initial state (e.g., object location or momentum) can affect whether a block tower collapses or remains stable, mirroring how people anticipate real-world noise and hidden variables.

2.1.3 Insights from Human Cognitive Development

Studies from developmental psychology reinforce the notion that humans grasp basic physical principles—like object permanence, solidity, and continuity—very early in life. Research on infant cognition demonstrates that children attend more strongly to events that violate *intuitive physics*, such as objects disappearing behind screens or moving through solid barriers [26, 23]. These "violation-of-expectation" paradigms show that infants as young as a few months old exhibit surprise when witnessing physically impossible scenarios, suggesting innate or rapidly learned "core knowledge" about object behavior.

Recent work has attempted to translate these infant paradigms into AI benchmarks, proposing that a system's ability to detect and explain physical impossibilities is a proxy for genuine causal understanding [27]. Intuitively, if a model only memorizes correlations in the training data, it may fail to detect subtle physical violations in new, out-of-distribution settings. An AI agent that more faithfully replicates infant-like responses to physical anomalies might instead encode deeper object-centric and temporal reasoning.

2.1.4 Learned Approximations of Physical Dynamics

Over the past few years, neural-network-based approaches to physical reasoning have proliferated, often reflecting the principle that objects and their relations can be modeled explicitly. A representative example is the *Interaction Network (IN)* approach [10], in which an agent first recognizes individual objects (e.g., shapes, positions, velocities) and then processes pairwise "interaction edges" through a relational module. This structure-based approach encodes collision forces, gravitational pull, or joint constraints as message-passing between objects in a graph. Such models have demonstrated impressive abilities to generalize across different numbers of objects, varied shapes, or new arrangements unseen during training, as long as the fundamental physics remains consistent.

Nevertheless, these learned approximations are often constrained by the coverage of their training data. For instance, if the neural model has not seen enough examples of stable towers on slopes, it may struggle to extrapolate friction or torque in angled scenarios. Similarly, ignoring certain physical parameters (e.g., object mass, friction coefficients) can lead to mispredictions. The gap between these purely data-driven networks and an ideal physics engine—capable of simulating any physically plausible arrangement—remains substantial. Yet learned systems bring unique benefits, including efficiency in certain tasks, adaptability to non-ideal or partially observable conditions, and integration with high-dimensional sensory inputs such as images or point clouds.

2.1.5 From Simple 2D Scenes to Complex Reasoning

A major theme in modern AI is bridging the gulf between toy physical puzzles and the messy, continuous world. Benchmarks like PHYRE [8], CLEVRER [31], and many other puzzle-based or simulation-based testbeds show that even in relatively simple 2D spaces, the combinatorial explosion of object interactions is nontrivial for a learned model to master. Realistic 3D domains pose even greater challenges, involving friction, deformable objects, fluid dynamics, or partial occlusions. This thesis's approach—building on 2D puzzle tasks but emphasizing iterative attempts and more sophisticated evaluation criteria—contributes one step toward bridging that gulf, addressing the open problem of robust, generalizable physical reasoning.

2.1.6 Applications and Broader Impact

Physical reasoning undergirds numerous applications. In robotics, for example, grasp planning and manipulation rely heavily on accurate predictions of how objects move upon contact, or whether they can be lifted without slipping. Virtual assistants may benefit from physical reasoning when explaining everyday phenomena, suggesting arrangement of furniture, or guiding user interactions with real-world devices. Computer vision systems that recognize collisions or mechanical failures can assist in hazard detection, automated warehouse logistics, or augmented reality. More broadly, understanding physical causality is often viewed as a cornerstone of common-sense AI, where systems can explain and predict outcomes that deviate from mere pattern-matching. As the field evolves, combining classical simulation methods with flexible, learned approximations will remain central to building AI agents that can operate reliably in real-world environments and reason about them at a deep, causal level.

2.2 Vision-Language Models (VLMs)

Vision-Language Models (VLMs) aim to integrate visual and textual representations, enabling them to process images or video frames and produce relevant textual outputs (e.g., captions, answers to queries) or vice versa. Most commonly, these models map an image into a set of learned embeddings, then fuse these embeddings with contextual linguistic tokens so that text-based attention mechanisms can reference visual elements. Typical tasks include *image captioning*, *visual question answering (VQA)*, and *visual classification*.

2.2.1 Combining Visual and Textual Embeddings

Broadly, VLM architectures vary in how they merge image and language features. Some rely on a two-stream approach, where an image encoder (e.g., a CNN or ViT) and a language encoder (e.g., a Transformer) run in parallel, later aligning representations through cross-modal attention (aka two-stream architecture), like ViLBERT or CLIP [21, 28]. Others use a single, unified Transformer that processes image patches alongside word tokens (aka one-stream architecture), like the models used in this study [24, 4, 1, 2, 7, 18, 3]. Regardless of the architecture, a key design challenge lies in ensuring the model captures *both* low-level perceptual details (shapes, colors, textures) and high-level semantic cues (actions, relationships, events). This synergy often supports tasks like describing a complex scene or answering queries about object attributes and their spatial arrangements.

Despite the success of such systems in capturing visual semantics, they often exhibit *limited causal or physical reasoning*. Indeed, many VLMs appear to rely on learned correlations: if an image shows a ball near the edge of a table, they might guess "the ball will fall" purely because such scenarios are common in training data, without a robust internal model of gravity, collisions, or momentum. As a result, these models can fail systematically when asked about outcomes requiring deeper inference, such as "Will the ball continue rolling and knock another object off the table?" or "Which object exerts the greater force during a collision?"

2.2.2 Limitations in Causal Reasoning

The inability of standard VLMs to reason explicitly about physical cause and effect has motivated new lines of research. For instance, chain-of-thought approaches [25] encourage the model to generate a step-by-step reasoning trace, aiming to connect visual observations to textual inferences more systematically. While such chain-of-thought expansions yield more transparent intermediate reasoning, they do not inherently supply a physics engine or robust causal simulator. Thus, a model might recite plausible reasoning steps but still lack a genuine understanding of friction or momentum transfer.

Deeper, physics-centric frameworks, such as *ESPRIT* [29], propose structured pipelines that first detect pivotal physical events (e.g., collisions) and then generate textual explanations referencing the underlying causes. Similarly, certain interactive learning paradigms incorporate forward simulation to refine the chain-of-thought in an online manner [13], albeit primarily for agent-based tasks where an LLM can trial different actions and observe outcomes. Though these methods still rely on large pre-trained language models, they attempt to endow them with a more causal or physically grounded approach than conventional VLM pipelines.

2.2.3 Tasks Addressed by Vision-Language Models

Beyond physical reasoning, Vision-Language Models excel in several interconnected tasks. They generate concise textual descriptions of images through image captioning—such as describing a scene with 'a cat sleeping on a windowsill'—and tackle visual question answering by responding to both open-ended queries and multiple-choice questions regarding image content. Moreover, these models are adept at crafting detailed narratives that capture contextual events in scene descriptions and summarizations, while also effectively handling referring expressions and dialogue by identifying objects based on linguistic cues, for example, instructing to 'pick up the red cup next to the book'.

2.2.4 Motivating Deeper Evaluation Methods

The limitations of VLMs in physical domains underscore the motivation for more specialized testbeds and protocols. Merely verifying whether a model can caption an image or classify its salient objects reveals little about its ability to *simulate* or *explain* dynamic events. As a result, researchers have proposed several strategies to advance evaluation methods in this area. For instance, physics-focused benchmarks utilize datasets that emphasize collisions, stability, gravitational forces, and other mechanical phenomena, typically featuring multi-frame or short video clips [29]. In addition, interactive evaluation setups enable models to generate predictions, observe outcomes from physics engines, and iteratively refine their guesses [13]. Moreover, chain-of-thought or structured reasoning prompts engage models in articulating the intermediate steps behind their predictions, thereby reducing reliance on mere data frequency-driven guesswork [25]. Taken together, these efforts seek to push VLMs beyond superficial pattern-matching and toward robust physical understanding.

2.2.5 Implications for This Thesis

The present work pursues precisely this goal: we argue that standard VLM tasks fail to probe deeper reasoning about object interactions, impetus, collisions, or multi-step cause-and-effect. Hence, the subsequent chapters propose a puzzle-based framework derived from real puzzle templates, combining 2D simulation with iterative attempts. The fundamental ambition is to investigate whether widely-used VLMs can accurately determine whether two objects will come into sustained contact, or if they can iteratively propose actions to achieve that contact. This focus on causality and simulation stands in contrast to purely descriptive tasks (like attempted in [29] with the same dataset) and aims to illuminate genuine physical reasoning capabilities (or lack thereof) in state-of-the-art Vision-Language Models.

2.3 Evaluating Physical Understanding in AI

Artificial Intelligence (AI) research in physical reasoning has benefited greatly from benchmarks that test whether a model's apparent "intuitive physics" goes beyond mere pattern recognition. As discussed, success on standard tasks may result from memorizing data distributions rather than truly internalizing how collisions, gravity, friction, or object shapes affect physical outcomes. Consequently, designing fair, robust evaluations is essential for distinguishing *causal reasoning* from *pattern-based reasoning*, examining an AI's ability to generalize in previously unseen scenarios, and probing the underlying representations (whether purely data-driven or endowed with more mechanistic insight).

2.3.1 Causal vs. Pattern-Based Reasoning

A central debate in physical reasoning research concerns whether a given model genuinely captures *cause-and-effect mechanisms* or merely reproduces correlations learned from training data. Broadly, we can distinguish between two approaches: causal reasoning and pattern-based reasoning. In the case of causal reasoning, a system identifies why events occur by approximating or encoding physical laws; such systems are capable of robustly handling counterfactual or hypothetical scenarios—for example, considering what might happen if a block were lighter—while generating mechanistic explanations (e.g., noting that a ball transfers momentum upon collision) and detecting violations of established laws such as reversed gravity. Conversely, pattern-based reasoning involves learning statistical regularities from large datasets without a deep understanding of the underlying physics. Although these models might perform well in familiar scenarios, such as those in which a ball rolling off a table is frequently observed, they can fail or behave inconsistently when faced with out-of-distribution conditions like frictionless surfaces or novel shapes, as their knowledge is anchored in superficial correlations. In practice, many AI models operate somewhere between these extremes, blending learned correlations with partial inductive biases regarding forces or collisions; however, bridging the gap between purely pattern-driven approaches and those demonstrating robust causal or mechanistic understanding remains a significant research challenge.

2.3.2 Challenges in Distinguishing True Physical Reasoning

Evaluating whether a model has *merely memorized* typical outcomes or has *internalized* physical laws poses several difficulties. One challenge is the issue of **Distribution shifts**, in which a model trained on common scenarios (e.g., typical collisions) may rely on correlations that break in novel or extreme cases; benchmarks like PHYRE explicitly test generalization to new templates [8], ensuring that memorized solutions do not trivially apply across tasks. Furthermore, the complications of **Partial occlusion and multi-step interactions** emerge as many real-world physical interactions are visually complex, featuring overlapping objects or prolonged contact, and handling such complexities requires robust modeling of hidden states (*e.g.*, an object's velocity behind an occluder), with CLEVRER [31] capturing collisions and sequences of events that demand multi-step causal reasoning, rendering superficial pattern-matching insufficient. In addition, issues of **Ambiguities and illusions** surface, since even humans can fall prey to illusions (*e.g.*, when an object's mass is misjudged by appearance), necessitating that an AI capable of causal reasoning correct such ambiguous cues by tracking physical consistency, as suggested by the *Intuitive Physics Engine* perspective [9]. Finally, **Violation-of-expectation (VoE) tasks**—inspired by infant cognition studies and demonstrating possible versus impossible events—challenge systems to detect anomalies based on underlying physical laws rather than on memorized frames [27, 26].

Designing evaluations that *separate* correlation-based success from genuine causal reasoning calls for test environments that provide *Systematic coverage of core physics concepts* (gravity, stability, collisions, support), incorporate *New or varied object configurations* outside the training distribution, and employ *Temporal queries, counterfactuals, and partial observability,* thereby encouraging the model to infer hidden states and reason about alternatives or hypothetical changes. In short, robust physical reasoning benchmarks strive to ensure that a model *cannot* rely on shallow pattern matching; instead, it must infer deeper, mechanistic relationships to succeed, especially under novel conditions.

2.3.3 Representative Benchmarks for Physical Reasoning

A variety of specialized benchmarks target different facets of physical understanding:

PHYRE [8] Focuses on 2D puzzle scenarios where the goal is to make designated objects contact each other. Agents insert one or two balls to trigger chain reactions. Evaluation in PHYRE is typically measured by how many tries a model needs to find a successful action ($AUCCESS$). This setup provides a direct gauge of an agent's ability to reason about collisions, support, and object trajectories without memorizing puzzle solutions.

CLEVRER [31] Centers on short video clips of colliding objects and poses descriptive, predictive, explanatory, and counterfactual questions. By probing higher-level queries (e.g., *"Which object caused the collision?"* or *"What if we removed object A?"*), CLEVRER tests whether models can extract the causal chain from multi-object interactions in an event-based manner.

VoE-Based Datasets [27, 26] Inspired by infant experiments, these datasets show possible and impossible events—such as an object disappearing behind a screen or passing through walls—and ask whether the model recognizes the violation. Distinguishing physically plausible from implausible scenarios can highlight whether the model encodes object persistence or continuity.

In many of these benchmarks, naive pattern-based learners achieve some success but fail systematically on *out-of-distribution* tasks or those requiring counterfactual thinking. Thus, they serve as crucial testbeds for investigating how far an AI system has progressed toward true causal modeling of everyday physics.

2.3.4 Implications for This Thesis

The present work contributes to this ongoing effort by developing a puzzle-based framework where proposals (e.g., inserted balls) must physically trigger or maintain contact between target objects. Building upon PHYRE [8] and related puzzle-based evaluations, we adopt a similar strategy, wherein we *enforce distribution shifts* by testing the model on new puzzle iterations or templates not explicitly seen during training to see if it generalizes, and through *iterative attempts* by observing how the model reacts to feedback and refines subsequent proposals, thereby differentiating short-term memorization from more robust simulation-based reasoning.

By combining puzzle tasks, iterative trials, and deeper queries about outcomes, we aim to discern whether modern Vision-Language Models (VLMs) rely purely on pattern-based heuristics or exhibit a *causal* understanding akin to simplified ”mental simulations” of physical laws.

2.4 Mental Simulations in Humans

While the previous sections have examined how AI systems might approximate or emulate physical reasoning, extensive evidence from *human cognition* reveals that people seemingly maintain a robust, flexible *mental simulation* capacity [9]. This ”intuitive physics engine” handles uncertain inputs and incomplete information through approximate forward modeling—akin to running a simplified physics simulator in the mind, replete with uncertain parameters like friction, mass, or object shape.

2.4.1 Probabilistic Nature and Noise Handling

Human mental simulations are not purely deterministic. Instead, individuals incorporate *probabilistic* elements, sampling possible outcomes and weighting them by likelihood [9]. For instance, when evaluating whether a block tower will topple, a person might internally consider small perturbations—does the top block shift slightly left or right? This sample-based approach is well-aligned with real-world conditions, where friction, angles, or shape imperfections can alter an outcome. Humans thus demonstrate surprising agility when confronted with mild distribution shifts (*e.g.*, unusual block shapes), as long as the underlying physics remains familiar.

2.4.2 Violation-of-Expectation in Infancy

Support for an early-emerging mental simulation capacity comes from infant studies on *violation-of-expectation (VoE)*. Infants as young as a few months old look longer at events defying intuitive physical principles, such as objects passing through walls or disappearing behind occluders [26, 23]. These behaviors suggest an innate (or rapidly learned) blueprint for physical reasoning—infants appear to track object continuity even if it is temporarily hidden. The stable individual differences documented in [26] imply that some infants consistently display stronger surprise responses to anomalies, hinting that domain-general or personality-based factors (like curiosity) shape their budding physical cognition.

2.4.3 Approximate Simulations vs. Rules

In conceptualizing how humans reason about everyday physics, *rule-based* or *heuristic* theories (e.g., ”If an object is not supported from below, it falls”) might seem plausible. However, empirical studies frequently show that no single fixed rule set covers the wide range of everyday physical phenomena. Instead, mental simulation accounts propose that the brain uses partial, approximate dynamics that can handle complexities like friction, collisions, or multi-step interactions in a more unified manner. Notably, [9] observed that people's judgments about block stability align more closely with sample-based forward simulation than with a simple rule counting how many blocks overhang the edge.

2.4.4 Relevance for AI: Bridging the Gap

Human mental simulations and infant cognition paradigms inspire AI in several ways. Notably, **Probabilistic rollouts** involve introducing noise into initial conditions to help AI models capture real-world uncertainty, an approach found in some *stochastic* or *sample-based* physics engines as well as in learned approximations that explicitly reason about parameter uncertainty. In addition, **Violation-of-expectation tests** repurpose infant-based VoE tasks as benchmarks for AI, requiring detection of physically impossible outcomes [27, 26]. Moreover, the concept of **Hierarchical or dynamic structure** is illustrated by humans seemingly maintaining stable object representations over time, with transformations (like collisions or re-orientations) computed on demand, which encourages AI architectures that are *object-centric* with flexible relational updates [10]. Furthermore, developmental psychology highlights how *minimal exposure* can foster early physical reasoning as infants require only a few months of observing everyday object interactions to exhibit stable expectations [23]. The disparity with data-hungry AI models—often needing millions of samples—emphasizes a possible shortfall: many AI systems remain heavily *pattern-based*, whereas infants can extrapolate from relatively limited exposure, suggesting more *causal* or *structured* representation building.

2.4.5 Implications for Evaluation and Model Design

If human mental simulations can handle unstructured, partially observable conditions from a young age, then AI models may benefit from *object-centric inductive biases,* reflecting how infants track discrete objects and expect them to persist even when occluded [23]; *probabilistic or sample-based strategies,* capturing the robust yet approximate quality of human physical intuitions [9]; *VoE-inspired tasks,* testing whether a model flags improbable collisions, unexpected stability, or magical transformations; and *exploratory or curiosity-driven learning,* akin to how some infants consistently show heightened surprise at anomalies [26], which might map to improved data-driven but mechanistically oriented AI approaches.

Despite the complexities of human cognition, the broad lesson remains: *people do not simply memorize event frequencies; they appear to approximate a causal simulator that can quickly adapt to new contexts.* By adopting a similar perspective in AI research—incorporating simulation-like components, object-centric reasoning, and robust evaluation protocols—models may move closer to the flexible physical understanding that characterizes humans from infancy onward.

The heuristic of mentally simulating physical interactions not only underpins human perception but also motivates the design of our confidence evaluation in the experimental framework. Specifically, by requiring models to provide a subjective probability estimate of success, we aim to mirror the human ability to internally assess the viability of a proposed action. For further details on this approach, see subsection 4.4.4.

3 Related Works

Broadly speaking, previous work on physical reasoning in artificial intelligence (AI) can be divided into two main thrusts: (i) the design of **benchmarks** and **datasets** that systematically evaluate an agent's ability to predict, explain, or plan physical interactions; and (ii) the development of **modeling techniques** and **evaluation protocols**, including interactive frameworks, that address the inherent challenges of dynamic, causal reasoning. This chapter first surveys representative benchmarks dedicated to physical reasoning and then highlights interactive evaluation methodologies. We then describe some of the primary approaches to enhancing model capabilities in physical domains, focusing on structured representations, object-centric architectures, and domain-specific training curricula. Finally, we situate the current thesis within these broader efforts, emphasizing how it leverages insights from existing benchmarks while introducing puzzle-based evaluation methods that incorporate iterative feedback.

3.1 Physical Reasoning Benchmarks

Over the last decade, the research community has devoted considerable effort to constructing benchmarks that target an AI system's capacity for *intuitive physics*, *causal reasoning*, and *object-interaction* understanding. Unlike standard vision or language tasks, physical reasoning benchmarks typically stress dynamic changes (collisions, toppling, stability, or contact) governed by approximate Newtonian mechanics.

3.1.1 From Static Scenes to Dynamic Interactions

Early attempts at evaluating physical reasoning often relied on static images containing mild cues about gravity or support. However, as the field advanced, researchers recognized the need for richer *temporal* contexts, where the system must track object motion over time. This evolution is evident in multiple 2D or 3D synthetic benchmarks that provide short videos or interactive puzzle environments.

3.1.2 PHYRE Benchmark

The *PHYRE* benchmark [8] is one of the cornerstones in 2D puzzle-based evaluation. It offers a collection of tasks in which an agent inserts red balls to achieve a simple goal, such as causing two specific objects to make contact for at least three seconds. Scenes include static and dynamic objects, simple shapes, and frictional interactions. PHYRE is split into two primary tiers:

1. **Single-Ball Tasks (PHYRE-B):** Agents place exactly one ball within a bounded 2D canvas.
2. **Two-Ball Tasks (PHYRE-2B):** Agents can place two balls at once, increasing combinatorial complexity.

A key metric is **AUCCESS** (Area Under the n-Curve of Cumulative successes), emphasizing *sample efficiency*: the fewer attempts needed to solve a puzzle template, the better the agent. In cross-template evaluation (i.e., generalizing to new puzzle templates), even advanced deep reinforcement learning strategies exhibit difficulty, underscoring the complexity of generalizable 2D physical reasoning.

3.1.3 CLEVRER Benchmark

Another influential dataset is *CLEVRER* [31]. Constructed with the Bullet physics engine and rendered to produce short synthetic videos of colliding objects, CLEVRER extends beyond descriptive queries by incorporating:

- *Predictive* questions: "What will happen next after the current frame?"
- *Explanatory/causal* questions: "Which object caused the collision?"

- *Counterfactual* queries: ”If we removed object A, would the collision still happen?”

Whereas PHYRE focuses on puzzle-solving and continuous scene dynamics, CLEVRER emphasizes *compositional reasoning*, *multi-step collisions*, and *counterfactual thinking.* Even advanced models tend to struggle with explanatory and counterfactual queries, reflecting a gap between simple correlation-based solutions and robust causal understanding.

3.1.4 PhysGame

Recently, *PhysGame* [12] examined how AI systems detect physical *glitches* or *violations* in real gameplay videos (e.g., an object hovering in midair in an otherwise realistic 3D environment). Each short clip is accompanied by a multiple-choice question, focusing on whether a system can identify the improbable event that defies known physical laws. Although the environment is arguably more complex and less controlled than PHYRE or CLEVRER, PhysGame underscores the real-world variety of ”broken physics” scenarios. The authors further propose specialized instruction-tuning sets (PhysInstruct, PhysDPO) to help large vision-language models detect anomalies. However, the approach lacks an explicit puzzle-solving dimension and leans toward single-pass question answering about short gameplay segments.

3.1.5 Additional Benchmarks: IntPhys, Craft, ShapeStacks, and Others

Beyond PHYRE and CLEVRER, a range of other physical reasoning testbeds have emerged:

IntPhys. The *IntPhys* framework and benchmark [30] focus on *violation-of-expectation* scenarios reminiscent of infant studies, using realistic 3D rendering. Models classify whether a short video depicts a ”possible” or ”impossible” event, highlighting occlusion-based illusions, object permanence, and shape constancy. Baselines struggle particularly with occluded or disjointed motion, demonstrating the challenge of multi-step reasoning under partial observability.

CRAFT. In *CRAFT* [6], synthetic 2D videos are used to test explicit *causal relationships* grounded in ”force dynamics”—which objects ”cause,” ”enable,” or ”prevent” certain outcomes. The tasks revolve around collisions, rolling, or containment events and feature descriptive, predictive, and counterfactual queries. The deliberate design ensures that purely correlation-based pattern matchers tend to fail in deeper causal questions.

ShapeStacks. *ShapeStacks* [19] addresses stability judgments in 3D scenes of stacked geometric objects. Each configuration is labeled as stable or unstable, along with precise mechanical points of failure. Such tasks highlight how small changes in object shapes or contact points can drastically alter structural integrity—an essential test for whether AI can parse the physics of supporting structures.

ComPhy. *ComPhy* [14] focuses on hidden object properties (e.g., mass, charge) and compositional reasoning across multiple videos of the same objects. A model must deduce underlying attributes from partial references before explaining or predicting outcomes in a longer ”target” video. This notion of *latent physical attributes* resonates with the real world, where appearance alone may not reveal mass or friction but must be inferred from interactions.

NovPhy. Another direction is *NovPhy* [17], addressing *open-world novelty* in a 2D puzzle environment akin to Angry Birds. Agents experience normal tasks followed by novel tasks that violate or shift some known dynamics. Success depends on detecting and adapting to unexpected physical changes, echoing how robust reasoning systems must handle real-world unpredictability.

SPACE Simulator. The *SPACE* dataset [16] contributes 3D scenes focusing on *containment*, *stability*, and *contact.* This environment produces multi-modal data (RGB frames, depth maps, object masks), facilitating training of physically grounded representations. Models pre-trained on SPACE can transfer to real-world tasks or other 3D physical benchmarks.

Overall, these benchmarks collectively underscore the varied facets of ”intuitive physics”: from object permanence and stability, to collisions and multi-step planning, to open-world novelty. They reveal that, while AI systems can master simpler or carefully constrained tasks, robust and generalizable physical reasoning in complex scenarios remains challenging.

3.2 Interactive Frameworks for Evaluation

While static benchmarks (like CLEVRER or IntPhys) provide valuable insights into a model’s ability to interpret or predict an outcome from a fixed scene or short video, *interactive* frameworks push further by allowing iterative queries, multiple attempts, or real-time modifications of the environment. Interactive evaluation can uncover deeper reasoning skills—especially how a system learns from feedback and adapts strategies over repeated tries.

3.2.1 Motivation for Interactive Evaluations

Many physical tasks (e.g., manipulations in robotics, puzzle solving) involve *trial and error*, hypothesizing an action, observing the outcome, and refining the approach. Simple single-pass question answering might miss this adaptive dimension. Researchers thus propose interactive protocols where a model can propose an action or partial solution, see the result (success, fail, or partial progress), and refine subsequent moves.

3.2.2 Clembench

A generic framework for multi-turn dialogue or ”game-based” interactions, *clembench* [11] aims to evaluate LLM-based agents across a suite of tasks. Each task—referred to as a ”dialogue game”—can incorporate rules and dynamic states, forcing the agent to adhere to formatting constraints and solve an interactive puzzle. Although *clembench* is not exclusively about physical reasoning, it illustrates how *dynamic benchmarks* can reduce memorization and better measure an agent’s iterative problem-solving capabilities. By generating new puzzle instances on the fly, clembench helps mitigate data contamination and repeated memorized solutions.

3.2.3 GLAM and Online Reinforcement Learning

Another approach, *GLAM* [13], integrates large language models (LLMs) with online reinforcement learning in text-based worlds. Although not always directly about Newtonian physics, the principle is analogous: an LLM-based policy receives partial observations, attempts an action, and obtains a reward. Over repeated episodes, it refines internal representations of how the environment behaves. This perspective can be adapted to puzzle-based physical tasks, where each trial refines the model’s understanding of collisions or gravity.

3.2.4 BabySit a Language Model, INTERACT, and Step-by-Step Feedback

Recent work also emphasizes how interactive feedback can shape or accelerate a model’s ”acquisition” of domain knowledge:

- *BabySit a Language Model from Scratch* [22] shows that a small student LLM can be taught by a teacher LLM through corrective demonstrations and iterative trials. Though primarily focusing on language tasks, the method illustrates the broader principle of *iterative teacher–student loops* where mistakes are clarified in real time.

- *INTERACT* [20] frames an LLM’s knowledge acquisition as question-driven dialogues with a teacher model. The student can ask follow-up questions or clarifications, thereby improving performance on subsequent queries. For physical reasoning, such an approach might allow a system to request clarifications like ”Is friction coefficient relevant here?” or ”How heavy is the object behind the occluder?” if the environment permits such an exchange.

While these frameworks are not all dedicated exclusively to physical puzzles, they underscore how repeated attempts, dynamic scene updates, and clarifying queries can reveal or improve an agent’s deeper causal understanding. By allowing multi-step trials, an agent must engage in mental simulation or learned heuristics more actively than in single-shot prediction tasks.

3.3 Approaches to Enhancing Model Capabilities

Physical reasoning is often conceptualized as a synergy of *structured representation* (e.g., object-centric, graph-based) and *learned dynamics* from data. Over time, researchers have proposed various ways to incorporate domain priors—like collisions or force constraints—into neural networks.

3.3.1 Relational Models and Object-Centric Reasoning

A foundational example is *Interaction Networks* [10], which treat each object as a node in a graph and each pairwise interaction (e.g., potential collision or gravitational force) as an edge. A learned function processes edges to update node states, capturing how local interactions affect global dynamics. This approach has shown remarkable generalization in n-body gravitational systems, rigid-body collisions, and mass-spring assemblies. The success of such methods highlights how representing objects and relationships explicitly (rather than flattening the entire scene into a single hidden vector) yields robust, generalizable physical predictions.

3.3.2 Hierarchical Reinforcement Learning and Simulation-based Tools

Several works combine object-centric reasoning with *hierarchical reinforcement learning* or *simulation-based planning*, especially for tasks involving tool use or puzzle solving:

- *Virtual Tools Benchmark* [5] reveals how humans rapidly learn novel tool placements to achieve a goal (like bridging a gap). A sim-based model that approximates human iterative solution strategies can match key aspects of rapid adaptation. By contrast, purely model-free RL approaches can flounder unless given extremely large training corpora.

- *SPACE* [16] pre-trains models to reason about containment and contact in realistic 3D scenes, after which these models transfer to real-world tasks requiring object manipulation. Such domain-specific pre-training can accelerate tool usage or puzzle-solving performance by giving networks ”core knowledge” about collisions, gravity, and friction.

In parallel, some lines of work adopt *curriculum learning*, starting from simpler physical tasks (e.g., single-object collisions) and gradually adding complexity (multi-object collisions, more friction variations), forcing a model to build structured knowledge incrementally.

3.3.3 Multimodal Reasoning and Instruction Tuning

Modern *vision-language models (VLMs)* increasingly integrate multimodal signals, leading to new opportunities for physically grounded language reasoning. However, many pretrained VLMs rely on vast, uncurated web data, capturing *statistical associations* rather than explicit physical laws. Hence, specialized instruction tuning or domain-specific data can inject deeper physical context:

- *LLF-Bench* [15] and *GLAM* [13] propose interactive or online approaches to refine large models, effectively bridging the gap between passive, correlation-driven training and active, environment-driven learning.

- Targeted fine-tuning sets (like PhysInstruct [12]) highlight physically anomalous or glitchy scenarios, teaching the model to pay attention to friction or support constraints.

Still, the performance of VLMs in these tasks remains significantly behind strong object-centric or simulator-based methods when exact physical predictions are required, confirming that bridging text and pixel-level cues is nontrivial.

3.3.4 Bringing Explanation and Transparency to Physical Reasoning

In addition to raw predictive power, some methods aim for interpretability:

- *ESPRIT* [29] identifies pivotal collisions in 2D puzzle tasks (drawn from PHYRE) and translates them into textual explanations referencing friction or gravity. This approach suggests a pipeline: (1) detect relevant collisions or events, (2) generate coherent, human-readable narratives.

- The intuitive physics engine hypothesis [9] similarly supports a mental *sample-based* approach to capturing collisions, stability, or tool usage, potentially improving explanation if a chain-of-thought or event-based module is available.

Such interpretability frameworks can be crucial in debugging or verifying physically grounded tasks, especially in safety-critical domains.

3.4 Positioning of This Thesis

In light of the prior work above, the present thesis situates itself at the intersection of *puzzle-based evaluation* and *interactive, iterative attempts* for physical reasoning—particularly within the domain of *Vision-Language Models (VLMs)*:

Building on the *PHYRE* dataset [8], our approach harnesses a large suite of puzzle templates, each requiring an agent to insert one or two balls, thereby incorporating **Puzzle-Based Incremental Difficulty**. By systematically varying puzzle configurations, we evoke an incremental difficulty reminiscent of the transitions from easy single-object interactions to more complex multi-object collisions or precarious arrangements. Moreover, rather than simply posing static classification or yes/no queries, we adopt an *iterative framework* (analogous to [11, 13, 22, 20]), wherein models can propose a solution (a ball insertion), observe partial outcomes or failures, and refine subsequent attempts, effectively bridging the gap between single-step question answering and repeated problem-solving scenarios.

Furthermore, our thesis integrates ideas from *ESPRIT* [29]—which employs puzzle scenarios to highlight collisions and friction—to assess broader ”mental simulation” capabilities of VLMs, reflecting the **Integration of Vision-Linguistic and Causal Physical Reasoning**. By requiring stable contact between certain objects, the puzzle tasks target essential aspects of intuitive physics—namely, collisions, support, friction, and partial occlusion—and resonate with the *simulation-as-engine* viewpoint [9] that evaluates whether modern large multimodal models can approximate a mental forward simulation or simply recall patterns from training data. In addition, by systematically varying puzzle arrangements and presenting out-of-distribution configurations, our approach examines whether VLMs rely on memorized heuristics or exhibit genuine *causal* and *mechanistic* reasoning, a question echoed by benchmarks like CLEVRER [31] that hamper superficial pattern matching by requiring predictions and explanations in new contexts. In doing so, the thesis addresses several open challenges by providing an expanded puzzle-based benchmark that continues the PHYRE tradition with explicit integration of *Vision-Language* queries and multi-step attempts, adapting interactive evaluation protocols to highlight iterative refinement—an ability many standard benchmarks fail to measure—and focusing on success/failure outcomes and collisions in each attempt to clarify whether a model is capable of ”mental simulation” or simply relies on static data correlations.

Consequently, this thesis not only leverages existing insights on puzzle-based physics tasks but also extends them to a more *interactive* and *multimodal* frontier, thereby offering a novel lens through which to assess and improve the physical reasoning capabilities of modern AI models.

4 Methodology

4.1 Overview of the Problem and Approach

The main objective of this study is to evaluate whether a given 2D environment, populated with simple objects, presents a scenario in which two specific objects—distinguished by their colors (for example, green and blue)—are predicted to make contact for a required duration within a given time frame. This evaluation not only centers on a straightforward prediction task, but it also delves into a broader examination of the physical reasoning abilities of SOTA Vision Language Models (VLMs). All code used in this chapter is available in our repository[1], where it is well-documented, modularly structured, and accessible for researchers interested in extending the framework or reproducing our evaluation results.

4.1.1 Terminology

To ensure clarity throughout this chapter, the following key terms are defined:

- **Template:** A unique physical scenario or puzzle concept from the PHYRE benchmark. Each template represents a distinct physical reasoning challenge with specific dynamics and goals. The PhysIQ benchmark uses 50 templates.
- **Puzzle/Iteration:** A specific instance of a template with varied object positions and sizes, while maintaining the core challenge. For each template, 100 different puzzles or iterations are present.
- **Proposal:** One (or more) object(s) (one or two red ball(s)) inserted into a puzzle with the goal of solving it by making the target objects come into contact for a required duration. For 20 of the 100 puzzles of each template, one correct proposal and three incorrect proposals of varying difficulty levels have been identified.
- **Simulation:** The process of applying physics rules to the puzzle environment over time to determine whether a proposal successfully solves the puzzle. The simulation captures the dynamic evolution of objects, generates visual representations at different time points, and determines outcomes based on three possible termination conditions: goal achievement (target objects maintain contact for the required duration), static termination (objects settle into a stable configuration without achieving the goal), or frame limit reached (simulation exceeds the maximum allowed frames).
- **Goal State:** Achieved when the two target objects (green and blue/purple) maintain contact for a required duration.

4.2 Dataset Construction

4.2.1 Parameters extraction from PHYRE

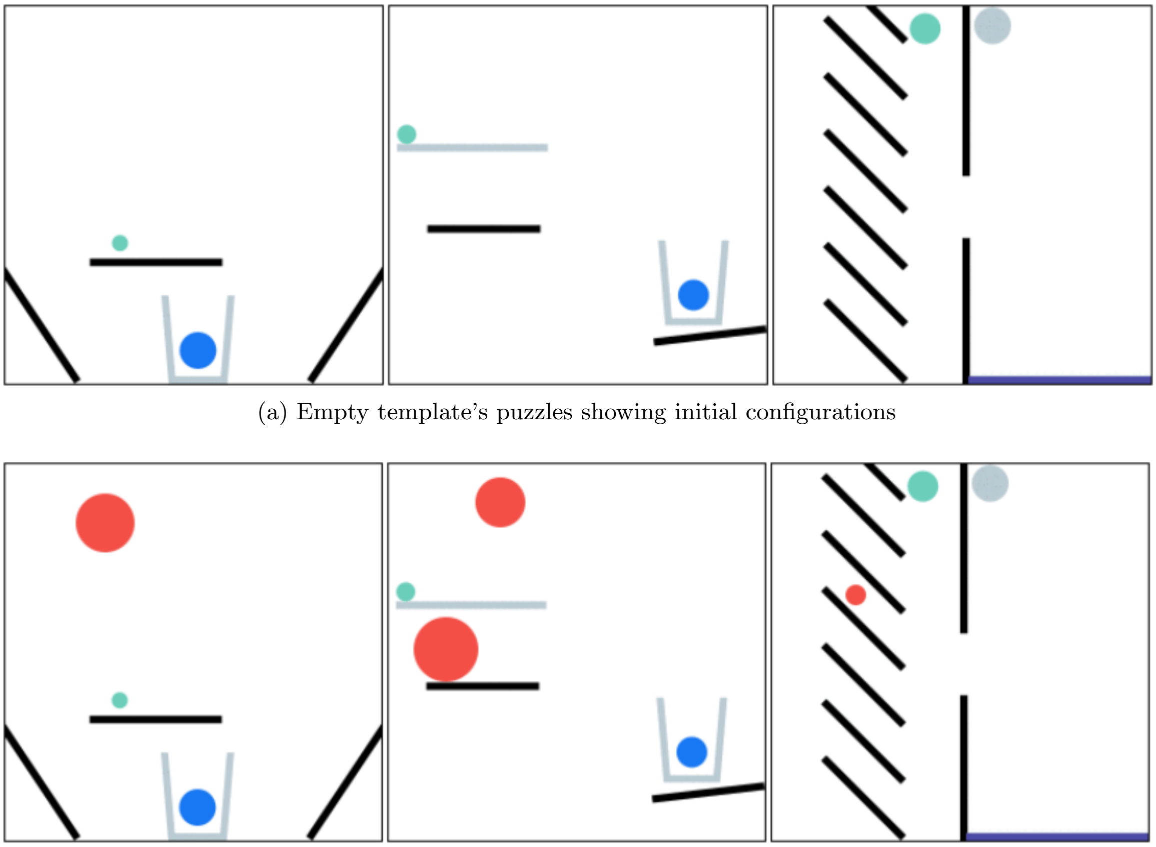
The foundation of the PhysIQ benchmark is built upon a carefully selected subset of templates from the PHYRE benchmark [8]. PHYRE (PHYsical REasoning) provides a diverse collection of physics puzzles designed to evaluate an agent's ability to reason about physical interactions. The decision to build upon the PHYRE dataset rather than creating puzzles from scratch was deliberate and strategic. This approach allowed us to avoid the time-consuming process of manually designing templates from the ground up, enabling us to focus our efforts on developing the broader evaluation framework and assessment methodologies.

[1] Available at https://github.com/Itakello/PhysIQ

PHYRE offers several key advantages that made it particularly suitable for our study. It represents the largest, most complete, and most diverse collection of physics puzzles available, providing a granular spectrum of challenges with varying complexity levels. The simplicity of its design, focusing on a minimal number of variables without extraneous noise, makes it ideal for isolating and testing specific aspects of physical reasoning. Templates increase in difficulty based on factors such as the number of bodies in the scene, the use of leverage mechanics, and reducing the number of possible proposals that lead to reaching the goal. More difficult templates also often require the insertion of two balls instead of one to solve the puzzle.

Despite these advantages, PHYRE does present certain limitations that should be acknowledged. The benchmark is constrained to a single type of goal—objects touching for a fixed duration—which limits the diversity of physical reasoning tasks that can be evaluated. Additionally, there is relatively limited variance in object shapes, with most puzzles featuring basic geometric forms. Furthermore, the puzzles tend to be crowded with objects, with only a small subset featuring a more minimal design of 3-6 objects. These limitations, while not undermining the value of PHYRE as a foundation, do suggest potential directions for future extensions of the PhysIQ benchmark.

For our study, we focused on extracting and adapting 50 distinct templates from PHYRE, comprising 25 templates requiring single ball insertion and 25 requiring two ball insertions. Each template in the dataset represents a unique physical scenario with specific object configurations and interaction goals. As illustrated in Figure 4.1, these scenarios involve various geometric objects such as circles, rectangles, and polygons, positioned in arrangements that present distinct physical reasoning challenges.



(a) Empty template's puzzles showing initial configurations

(b) The same template's puzzles with the red ball proposals inserted (the middle one should be attempted with two balls)

Figure 4.1: Examples of template's puzzles from the PhysIQ benchmark. Each puzzle presents a unique physical reasoning challenge where the goal is to make the green and blue/purple objects come into contact. The top row shows the initial configurations, while the bottom row shows the same puzzles with proposed solutions (red balls) inserted.

An important characteristic of the PHYRE dataset is that each template comes with 100 pre-generated iterations, where each iteration maintains the same fundamental puzzle concept but varies in the specific positioning and sizing of objects. This feature of PHYRE ensures that our benchmark captures a wide range of physical configurations while preserving the core reasoning challenge of each template.

The extraction process from the PHYRE library was implemented through a two-stage pipeline. First, using the `phyre.loader` module, we extracted the raw puzzle data including scene dimensions, body properties, and relationship definitions. This data was then converted to a standardized JSON format. Further details about the JSON structure and puzzle representation can be found in subsection 4.2.2.

In the second stage, these JSON representations were processed and stored in a MongoDB database to facilitate efficient retrieval and manipulation during the benchmark evaluation. Each puzzle entry in the database contains comprehensive information about the bodies (including position, shape, size, and physical properties), the relationship indices defining the goal condition (which bodies need to make contact), and metadata describing the puzzle type and characteristics.

Starting with this established benchmark provided a solid foundation for our work, although recreating the simulations with precise fidelity to the original PHYRE implementation presented significant technical challenges that required careful attention to physical parameters and simulation details.

This structured approach to template acquisition and adaptation ensures that our benchmark provides a robust foundation for evaluating the physical reasoning capabilities of Vision Language Models across a diverse range of scenarios.

4.2.2 Puzzle Representation

The PhysIQ benchmark represents puzzles as structured data objects that encapsulate all necessary information for physics simulation. Each puzzle is extracted from the PHYRE benchmark and converted into a standardized JSON format that captures the geometric, physical, and relational properties of all objects in the scene. This representation serves as the foundation for both simulation and evaluation tasks.

Core Components of Puzzle Representation

Each puzzle in the PhysIQ benchmark consists of four primary components:

The first component, **Scene Dimensions**, ensures that each puzzle operates within a fixed 2D space of 256 × 256 pixels, providing a consistent environment for all simulations. In addition, the **Bodies** component represents the collection of physical objects that populate the scene, each with specific properties that determine its behavior during simulation.

Furthermore, the **Relationship** component defines the goal condition by specifying which bodies must make contact for the puzzle to be considered solved, while the **Metadata** component provides additional information about the puzzle, including its tier (single ball or two balls) and a textual description.

Body Properties and Shape Types

Each body in a puzzle is defined by a comprehensive set of properties that determine both its appearance and physical behavior:

1. **Position:** The (x, y) coordinates of the body's center within the scene.
2. **Body Type:** Determines the physics behavior of the object:
   - Static (0): Fixed objects that do not move during simulation.
   - Dynamic (1): Objects that respond to forces and collisions.
   - Kinematic (2): Objects that can be moved programmatically but are not affected by physics.[2]

[2] The Kinematic body type was not used in any of the puzzles in the PHYRE/PhysIQ benchmark.

3. **Angle:** The rotation of the body in radians.

4. **Color:** An integer index that maps to a specific color. The colors used are Red (0), Black (1), Green (2), Azure/Blue (3), Purple (4), and Grey (5).

5. **Shape Type:** Defines the geometric form of the body:

   - Polygon (0, 2): Defined by a set of vertices.
   - Circle (1): Defined by a center point and radius.
   - Compound (3, 4): Complex shapes composed of multiple polygons.

Relationship Definition

The relationship component is pivotal in defining the goal condition of each puzzle within the PhysIQ benchmark. Specifically, it identifies the indices of the two bodies that must make contact for the puzzle to be considered solved. In our benchmark, the relationship type is consistently ”touching,” meaning that the two target bodies, green and blue/purple, must remain in contact for three seconds during the simulation. Although the PHYRE framework supports other types of relationships, such as distance-based and orientation-based, these were not utilized in our study.

Extraction Process

The extraction process from PHYRE to PhysIQ involves several steps, starting with loading the original PHYRE puzzles using the `phyre.loader` module, converting each puzzle's scene, bodies, and relationships to a standardized JSON format, and processing shape-specific data such as extracting vertices for polygons and radius for circles. This sequence of actions ensures that the base geometrical and relational data is preserved accurately while also being reformatted for consistency.

Following this, the process continues by mapping PHYRE-specific indices and values to PhysIQ's standardized format and storing the processed puzzles in a MongoDB database for efficient retrieval during experiments. Throughout the extraction, special attention is paid to preserving the physical properties of each body; while the geometric and relational data is explicitly extracted from PHYRE, the physical simulation parameters (such as friction, elasticity, and density) are standardized across all puzzles as detailed in subsection 4.3.2.

JSON Structure Example

A simplified example of the JSON structure for a puzzle might look like:

```
{
  "scene_dimensions": [256, 256],
  "bodies": [
    {
      "position": [128, 200],
      "body_type": 0,  // static
      "angle": 0,
      "color": 2,  // green
      "shape_type": 1,  // circle
      "radius": 10
    },
    {
      "position": [200, 50],
      "body_type": 1,  // dynamic
      "angle": 0.5,
      "color": 3,  // blue
      "shape_type": 0,  // polygon
      "vertices": [[0,0], [20,0], [20,20], [0,20]]
    }
  ],
  "relationship": {
    "bodyId1": 0,
    "bodyId2": 1,
    "relationships": [0]  // contact relationship
  },
  "metadata": {
    "description": "Make the green circle touch the blue square",
    "tier": 0  // single ball puzzle
  }
}
```

This structured representation enables consistent simulation across different puzzles while capturing the unique geometric and physical characteristics of each scenario. The standardized format also facilitates the generation of visual representations for the VLM evaluation tasks and ensures reproducibility of results across experiments.

4.3 Physics Simulation Setup

4.3.1 Simulation Engine Selection and Implementation

To ensure a high degree of fidelity in our simulations, closely mirroring the dynamics of the PHYRE benchmark, we meticulously adopted the identical physical and simulation parameters. This was achieved through a detailed examination of the source code within the original PHYRE repository, allowing for a direct replication of their setup.

To ensure scalability, facilitate the incorporation of future puzzle templates, allow the creation of other types of proposals and obtain the possibility to add features at will, we completely re-engineered the simulation process from the ground up. This reimplementation allowed us to eliminate redundant and outdated code, which was essential given that the original PHYRE repository is archived and no longer maintained. Moreover, this redesigned architecture provides a robust foundation for integrating advanced features, as elaborated in the following sections.

Initially, we experimented with the `pymunk` 2D physics engine[3]. However, these preliminary experiments uncovered notable discrepancies in the simulation behavior compared to PHYRE. Specifically, we observed instances where objects intended to be stable would unexpectedly collapse, and, in some cases, puzzles were inadvertently solved without any intervention. These inconsistencies suggested that `pymunk` was not suitable for achieving the desired level of simulation accuracy required for a faithful replication of PHYRE.

As a result, we investigated alternative physics engines and found that PHYRE's native engine, `Box2D`[4], implemented in C, was accessible via its Python API, `pybox2d`[5]. The adoption of `pybox2d` led to a significant improvement, enabling a near 1:1 replication of the PHYRE physics dynamics. While minor discrepancies persisted in a few specific puzzles, as further detailed in subsection 4.3.3, the overall simulation fidelity was substantially enhanced.

4.3.2 Simulation Constants and Parameters

To ensure the accurate replication of the intended physical dynamics, we implemented a comprehensive set of constants. These parameters are categorized into **Physical Constants**, which govern the fundamental properties of the simulated environment; **Simulation Constants**, which manage the numerical aspects of the simulation process; and **Stopping Constants**, which define the criteria for terminating the simulation. Table 4.1 provides a detailed overview of these constants, accompanied by concise descriptions for each parameter.

[3] Available at https://www.pymunk.org/en/latest/index.html

[4] Available at https://box2d.org/

[5] Available at https://github.com/pybox2d/pybox2d

Category	Parameter	Description
Physical	Scene dimensions	Size of the simulation scene (256 × 256 pixels)
	Gravity	Acceleration due to gravity (9.81 m/s^2)
	Density	Mass per unit area (0.25 kg/m^2)
	Friction coefficient	Coefficient determining resistance to sliding (0.5)
	Elasticity (restitution)	Bounciness of collisions (0.20)
	Angular damping	Reduction factor for rotational motion (0.01)
	Linear damping	Reduction factor for linear motion (0.0)
	Min Proposal Radius	Minimal length of the proposals (2)
	Max Proposal Radius	Maximum length of the proposals (32)
Simulation	Frame rate	Number of frames per second (60 FPS)
	Time scale	Factor to adjust simulation speed (1.0)
	Scene dimensions	Rendered scene dimensions (256 × 256 pixels)
	Velocity iterations	Iterations for the velocity solver (10)
	Position iterations	Iterations for the position solver (10)
Stopping	Stop velocity threshold	Threshold below which an object is considered static (0.1)
	Required frames for early stop	Consecutive frames with static objects to trigger early stopping (400)
	Required frames for goal verification	Consecutive frames where the two target objects had to remain in contact (360)
	Max frames	Maximum frames allowed for simulation before forced termination (3000)

Table 4.1: Physical, Simulation, and Stopping Constants Used in the Experiments

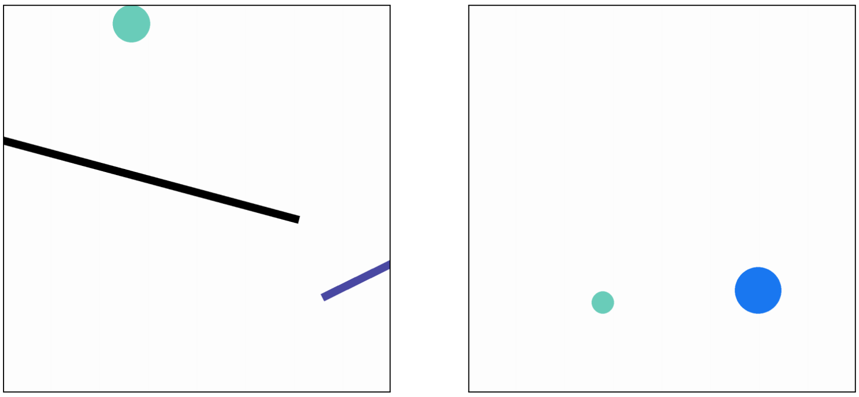
In our simulation framework, we made deliberate adjustments to the stopping parameters to ensure both practical evaluation and fidelity to the desired physical dynamics, while we left unchanged the physical and simulation ones.

The `stop velocity threshold` constant value of 0.1 was determined by analyzing the velocities of objects in templates where they merely fell under gravity. Although a completely static object would ideally have zero velocity, minor fluctuations or flickering caused the measured speeds to remain slightly above zero. Observations showed that these velocities were consistently under 0.1, making it an effective threshold.

For the `required frames for goal verification` constant (finally set at 3 seconds of execution), the chosen value was a compromise. It is sufficiently long to ensure that in templates like 00011 – where the target objects naturally touch for only a limited time – the goal is not automatically achieved, hence requiring the insertion of an additional proposal (see subsection 4.3.5). At the same time, it is not overly stringent for puzzles in which both target objects are dynamic (e.g. template 00000), where maintaining a lengthy collision would be impractical. Both templates can be seen in Figure 4.2

The `required frames for early stop` constant was deliberately set higher than the goal verification threshold. This ensures that the simulation does not terminate prematurely when the two target objects are touching but still moving slightly. At the same time, a ”not-so-long” interval guarantees that the series of screenshots capture key stages of the simulation instead of the same ones with the objects motionless in the same place, by providing more informative data for the interactive evaluation (see subsection 4.4.7).

Finally, the `max frames` constant was set to allow a simulation duration of up to 25 seconds. This value ensures that puzzles, even with an inserted proposal, remain solvable in a reasonable time frame. While setting this constant too low might lead to premature termination, a higher value primarily affects the proposal identification time, which is an acceptable trade-off.



(a) Example template 00011

(b) Example template 00000

Figure 4.2: Examples of puzzle templates used in the PhysIQ benchmark. (a) shows an example where the two target objects automatically touch even without a proposal, and (b) shows an example with both targets dynamic.

4.3.3 Verification of Puzzle Integrity

All puzzles from the initial datasets were simulated without the addition of any new proposals. In instances where the simulation reached the goal state naturally, such samples were deemed `untestable` and excluded from subsequent phases of dataset creation. Specifically, only 13 out of 5000 puzzles were removed from further processing because they solved themselves, indicating potential issues in this setting.

It is our conjecture that these discrepancies arise from subtle differences between the original PHYRE framework and our re-engineered simulation environment. While small variances in the version of the physical engine or in minor details of the simulation code could be responsible, the root cause remains somewhat ambiguous. The comprehensive adjustments to the stopping parameters, as detailed in subsection 4.3.2, were pivotal in enlarging the pool of valid samples while ensuring that the simulation remained feasible for identifying a correct proposal.

Through this rigorous verification process, we ensured that the majority of the puzzles were retained, thereby upholding the integrity and evaluative quality of the resulting dataset, while also keeping space for the correct proposal identification.

4.3.4 Simulation Process

The simulation engine in our framework serves as the cornerstone for both dataset creation and evaluation. Beyond the physics dynamics detailed in subsection 4.3.1, the simulation plays a critical role in several key aspects of the benchmark. It is responsible for generating visual representations (screenshots) of the puzzle states at different time points, which are particularly important in the Interactive Evaluation phase (see subsection 4.4.7), where they provide visual feedback to the model about the evolution of the simulation after a proposal has been inserted. Capturing the frames at regular intervals of 60 frames (half second), it allows to later take key screenshots based on regular intervals of multiples of half second, or evenly distributed screenshots (like it has been done in the Interactive Evaluation) to extract a small set of key frames. These screenshots allow models to observe the physical consequences of their proposed solutions, facilitating iterative refinement. The screenshots have been also leveraged for the other static evaluations, where either the first or last frames (or both) has been given to the models to enrich the prompt and extend it's context by attaching visual informations critical for this kind of scenario. Moreover, the simulation determines the outcome of each puzzle attempt through one of three possible termination conditions: **Goal Achievement:** the simulation detects, via its collision handler, that the target objects (green and blue/purple) have maintained contact for the required duration (3 seconds), representing a successful solution; **Static Termination:** the simulation stops because the world has remained static (object velocities below the threshold) for the duration specified by the `required frames for early stop` parameter, indicating that the objects have settled into a stable configuration without achieving the goal; or **Frame Limit**

Reached: the simulation terminates after exceeding the maximum number of frames allowed (`max frames`), indicating that the objects continue to move indefinitely without either achieving the goal or settling into a stable state. These termination conditions are essential not only for dataset creation, where they determine whether a generated proposal is classified as correct or incorrect, but also for the evaluation phase, where they provide the ground truth against which model predictions are assessed.

The simulation process maintains a consistent workflow across all phases of the benchmark by beginning with loading the puzzle configuration from its JSON representation and initializing the physics world with the specified physical parameters. It then creates and positions all bodies according to the puzzle specification, followed by the insertion of any proposed objects (for evaluation). The simulation proceeds by stepping through the simulation, capturing screenshots at regular intervals while monitoring for goal achievement or termination conditions, before finally returning the outcome (success/failure) and relevant metrics. This standardized simulation process ensures consistency across both dataset creation and model evaluation, allowing for reliable comparison of model performance on physical reasoning tasks.

4.3.5 Correct and Incorrect Proposal Generation

To systematically explore the proposal space and reliably identify correct proposals for each puzzle template, the pipeline was designed to execute a maximum of 10,000 proposal brute-force insertions per puzzle. This specific threshold of 10,000 attempts was adopted because it mirrors the approach used in the original PHYRE benchmark for identifying correct proposals, ensuring consistency with the established methodology. For puzzles derived from the one-ball templates, a single proposal was generated in each iteration, whereas for the two-ball templates, two simultaneous proposals were created—one for each ball. In each iteration, the proposals were generated by sampling a random position uniformly within the bounds of the fixed 256×256 pixel scene and by selecting a random radius from a continuous distribution with limits set between 2 and 32 pixels (as described in Table 4.1). After generating a proposal, the simulation was executed until one of two conditions occurred: either the simulation halted because it reached the maximum number of allowed steps (or the objects' velocities fell below the predefined stop velocity threshold), or the goal state was achieved.

The pipeline continues iterating until it records exactly 20 correct proposal iterations for each template. The rationale behind limiting to 20 samples per template stems from a balance between computational cost and diminishing returns; after a certain number of correct proposals, additional proposals tend to yield less new information about the puzzle's solution space. This limit is especially practical for more challenging templates (e.g., templates 122, 123 and 124) that inherently yield fewer correct solutions, as illustrated in Figure 4.3, which shows the mean attempts required to generate correct proposals across different templates. As the figure illustrates, there is a significant variation in difficulty across templates, categorized into easy, medium, and hard. Templates in the hard category (right section) required substantially more attempts—often in the thousands—to find correct solutions, while easy templates (left section) typically required fewer than 500 attempts. This quantitative measure of generation difficulty aligns with our qualitative assessment of template complexity and provides empirical justification for our difficulty categorization. Furthermore, the difference between 1_ball and 2_ball models is evident, with 2_ball templates generally requiring more attempts to find correct solutions.

Once a correct proposal is identified, the process of generating corresponding incorrect proposals is initiated. Similar to the correct proposal generation, a brute-force approach with a maximum of 10,000 attempts was also employed to find suitable incorrect proposals for each correct one. In most instances, incorrect proposals were identified within the first few attempts, though in some cases, more iterations were necessary. Notably, only one incorrect proposal, a 'hard' one for template 00010, could not be found within the attempt limit. For each correct proposal, three distinct incorrect proposals are constructed by modifying the parameters of the correct proposal. These modifications involve two key adjustments. First, the radius is altered by selecting a new value within a range defined between half of the correct proposal's radius and twice that value, while also ensuring that the new radius falls within the absolute bounds used for generating correct proposals (i.e., between 2 and 32 pixels). Second, a spatial displacement is introduced by moving the center of the proposal by a distance, to classify the incorrect proposals into three levels based on the degree of deviation:

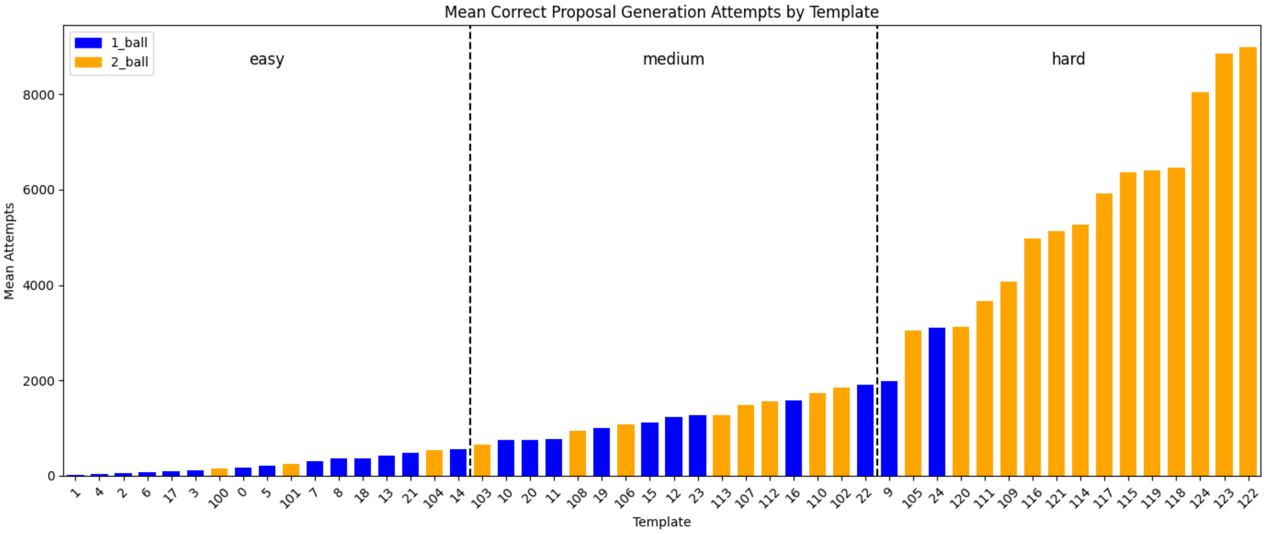


Figure 4.3: Mean Correct Proposal Generation Attempts by Template.

- **Hard** incorrect proposals involve minimal modification, with a displacement of 0 to 1× the original radius, maintaining proximity to the correct solution.

- **Medium** incorrect proposals are generated by adjusting the radius to lie between 1× and 2× the original radius difference, representing a moderate deviation.

- **Easy** incorrect proposals are produced by a more significant adjustment, where the radius is altered by a distance between 2× and 4× the original radius difference, making the proposal clearly distinct from the correct one.

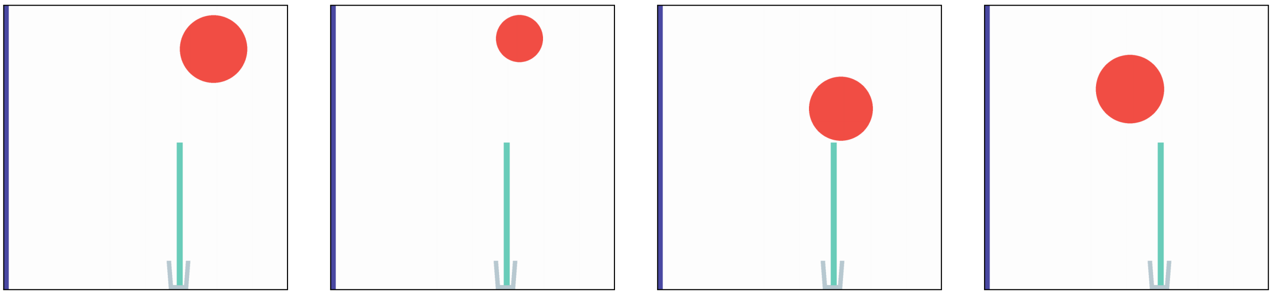
The motivation for incorporating this granularity is to understand whether easier alterations are more readily detectable as incorrect compared to subtler, harder modifications. By having three varying levels of difficulty for each correct proposal, the evaluation framework not only assesses the model's ability to recognize correct physical interactions but also its sensitivity to increasingly challenging deviations. Overall, for every correct proposal found, three incorrect proposals are generated, thereby enriching the evaluation metrics and providing a nuanced view of model performance in physical reasoning tasks. While 'easy' incorrect proposals are generally expected to be more obviously incorrect than 'hard' ones, the non-deterministic nature of physical puzzles means this is not strictly guaranteed. It is possible, though less probable, that an 'easy' incorrect proposal might, in some cases, inadvertently lead to a state closer to the goal than a 'hard' one. However, the design principle is that 'hard' incorrect proposals represent more subtle and thus more challenging deviations from the correct solution.

As shown in Figure 4.4, the four proposals discovered for template 00003 iteration 000 — one correct proposal and three incorrect proposals (hard, medium, and easy) — illustrate the systematic variation in the proposal perturbations.

4.3.6 Dataset Organization

The final PhysIQ dataset consists of a comprehensive collection of puzzles and proposals structured as shown in Table 4.2.

In our data generation process, for each of the 50 templates we attempted to brute-force the insertion of a correct proposal over 20 puzzle iterations. If a correct proposal was not found within 10,000 proposal attempts for a given puzzle, that puzzle was discarded. This disciplined curation procedure resulted in exactly 1000 puzzles with a correct proposal. Subsequently, for every correct proposal, three distinct incorrect proposals were generated, corresponding to easy, medium, and hard difficulty levels. Thus, the final dataset comprises 1000 correct proposals and 3000 incorrect proposals.



(a) Correct (b) Incorrect (Hard) (c) Incorrect (Medium) (d) Incorrect (Easy)

Figure 4.4: The 4 proposals found for template 00003 iteration 000.

Component	**Count**
Templates	50
Puzzle iterations per template	20
Total puzzles considered	1000
Correct proposals per puzzle	1
Total correct proposals	1000
Incorrect proposals per correct proposal	3
Total incorrect proposals	3000

Table 4.2: Summary of the PhysIQ dataset organization

4.4 Evaluation Tasks and Procedures

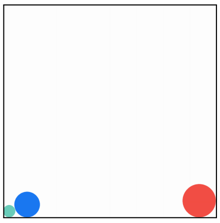
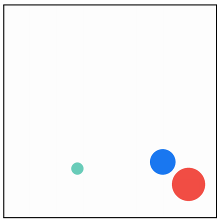
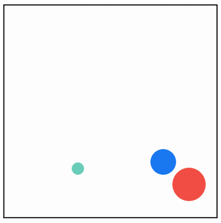
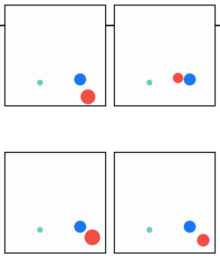
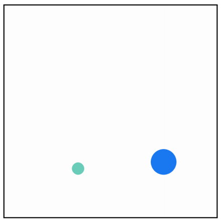
4.4.1 Overview of Evaluation Approach

To comprehensively assess the physical reasoning capabilities of Vision Language Models (VLMs), we designed a multi-faceted evaluation framework comprising five distinct tasks of increasing complexity. This approach allows us to systematically probe different aspects of physical understanding, from basic perception to advanced predictive reasoning and adaptive problem-solving. Table 4.3 provides a comprehensive overview of these tasks, including the presented frames, questions, simulation details, response types, and few-shot settings for each task.

Our evaluation strategy integrates both static and dynamic assessment methods. The static evaluations (sanity check, confidence estimation, binary classification and ranking) present the model with fixed scenarios and assess its ability to make predictions without iterative feedback. In contrast, the dynamic evaluation (interactive) enables the model to engage with the environment iteratively, learning from simulation feedback to refine its proposals.

The five evaluation tasks, in order of increasing complexity, are the following. First, we perform the **Sanity Check Evaluation**, the most fundamental evaluation that assesses whether models can correctly identify if two objects are touching in a final state image; this serves as a prerequisite filter—models that fail this basic perceptual task are not subjected to more complex evaluations. Next, the **Confidence Estimation** tests the model's calibration by requiring it to provide a probability estimate for a proposal's success, revealing how well the model can quantify uncertainty in physical predictions. Then, the main evaluation method, **Binary Classification** requires models to predict whether a given proposal will successfully solve a physics puzzle, directly measuring the model's ability to reason about physical outcomes from static images. Following this, the **Ranking** offers a more nuanced assessment that challenges models to order multiple proposals by their likelihood of success, evaluating the model's ability to discriminate between subtle differences in physical configurations. Finally, the **Interactive Insertion** is the most sophisticated assessment, allowing models to iteratively refine their proposals based on simulation feedback, and measuring not only physical reasoning but also the ability to learn from experience and adapt strategies.

This comprehensive evaluation framework enables us to not only assess the raw predictive capabilities of VLMs in non-interactive contexts but also evaluate their ability to adapt and improve through interactive feedback. Through this multi-layered approach, we aim to highlight both the strengths and limitations of current models in handling physical reasoning tasks within complex environments.

Task	**Sanity Check Evaluation**	**Confidence Estimation**	**Binary Classification**	**Ranking Task**	**Interactive Insertion**
Presented Frame(s)					
Frame(s) Info	Last frame with proposal	First frame with proposal	First frame with proposal	First frame of each proposal	First frame without proposal
Simulation Info	No	Yes	Yes	Yes	Yes
Few-Shots	0	0	0,2,4	0,1,2	0
Question	Are the target objects touching each other?	What's your confidence the goal is going to be reached?	Is the goal going to be reached?	What's the proposal's ranks from most to least likely to accomplish the goal?	Insert the proposal in order to reach the goal.

Response types	Yes/No	A percentage from 0% to 100%	Yes/No	A list like:
				$[1, 2, 3, 4]$
					A JSON like:
					`{`
					`"x": 20,`
					`"y": 45,`
					`"radius": 10`
					`}`

Table 4.3: Evaluation tasks table for the PhysIQ benchmark: This table provides a comprehensive comparison of the five evaluation tasks used to assess VLM physical reasoning capabilities. For each task, the table details: (1) the visual stimuli presented to the models (showing example frames), (2) specific frame information provided, (3) whether simulation context was included, (4) the number of few-shot examples used in different experimental conditions, (5) the revisited and shorter question formulation presented to the models, and (6) the expected response format. The progression from left to right represents increasing task complexity, from basic perceptual tasks to sophisticated interactive reasoning.

4.4.2 Models and Experimental Setup

In this study, we employed a selection of nine state-of-the-art vision-language models (VLMs) to evaluate their physical reasoning capabilities on the PhysIQ benchmark. All experiments were conducted using a fixed sampling temperature of 1.0. The models were chosen based on several criteria: the overall performance in non-reasoning tasks, model size, source availability (closed or open weights), and domain-specific attributes. In particular, GPT-4o [24] and Claude 3.5 Sonnet [1] represent some of the best non-reasoning models currently available, both being closed weights with undefined sizes. Gemini 2.0 Flash [2] is a small language model, also closed weights, selected for its efficiency. Grok-2-Vision-1212 [3] from xAI is another closed weights model included to represent emerging competitors in the VLM space.

For open weights models, we included Qwen [7], a state-of-the-art Chinese VLM with 72 billion parameters, which is medium in scale. Pixtral-Large-2411 [4] from MistralAI (a French company specializing in open weights models), was included due to its large scale and promising performance in complex tasks. Additionally, we evaluated two models from Meta's Llama family: Llama-3.2-90B-Vision-Instruct [18], a medium-sized model with 90 billion parameters, and Llama-3.2-11B-Vision-Instruct, a smaller model with 11 billion parameters, to assess how model size affects physical reasoning capabilities within the same model family.

Table 4.4 summarizes the key characteristics of the models used in this study.

Name	**Provider**	**Size**	**Weights**
Pixtral-Large-2411	MistralAI	123B	Open
Qwen2.5-vl-72b-instruct	Qwen	72B	Open
Llama-3.2-90B-Vision-Instruct	Meta	90B	Open
Llama-3.2-11B-Vision-Instruct	Meta	11B	Open
Grok-2-Vision-1212	xAI	Big	Closed
Gemini 2.0 Flash	Google	Small	Closed
Claude 3.5 Sonnet	Anthropic	Big	Closed
GPT-4o	OpenAI	Big	Closed

Table 4.4: Summary of the VLMs used in the experimental setup, including model name, provider, size, and weights availability.

The model selection was driven by a desire to cover a wide spectrum of architectural designs and performance characteristics, ranging from small to large open weights models to leading closed weights systems. This diverse selection provides insights into how different models perform under equivalent experimental conditions, particularly in the context of physical reasoning tasks as defined in the PhysIQ benchmark. The inclusion of models from the same provider but with different sizes (such as the Llama models) also allows for analysis of how model scale affects physical reasoning capabilities within a consistent architecture.

4.4.3 Sanity-Check Evaluation

In the context of our evaluation framework, the sanity check task serves as the most fundamental test of a model's capability to understand and interpret a fully resolved outcome of a physics simulation, as introduced in subsection 4.4.1. This task is expressly designed to determine whether the model can reliably ascertain if the target objects the green object and the blue (or purple) object—are in contact, based solely on the final state image produced by the simulation.

The task is intentionally straightforward. The accompanying stimulus is a final state image of the simulation, and the model is prompted with a clear, unambiguous question, such as: "Is the green object touching the blue object in this image?" This minimalistic prompt ensures that the model's response is evaluated in a zero-shot setting. Because the ability to correctly identify contact (or lack thereof) is a basic perceptual and interpretative skill, it should already be inherent in the pre-trained model without needing additional contextual examples.

In our setup, we include a prompt found in the attachments (see the Sanity-Check prompt in Appendix A) which illustrates the practical example provided to the model. This prompt reinforces the expected response format by instructing the model to answer exclusively with "Yes" or "No". By doing so, it eliminates any ambiguity and prevents the model from providing explanations or extraneous details, which might otherwise confound a binary evaluation framework.

Furthermore, for the sanity-check task, we have chosen to select five correct and five incorrect proposals (randomly sampled from the available easy, medium, and hard categories) for each template. This approach results in a total of 500 queries per model, ensuring a balanced and comprehensive evaluation of the model's perceptual and interpretative capabilities.

Models that fail to meet the sufficient performance criterion on this basic task are not evaluated further. The justification for this strict benchmark is that if a model cannot reliably confirm or deny the occurrence of the requisite contact in such an elementary setting, it is indicative of an absence of the core competencies required for the more advanced and nuanced evaluation stages. Essentially, failure in the sanity check task serves as an early elimination mechanism, ensuring that only models possessing the necessary baseline physical reasoning abilities are subjected to more challenging tasks such as binary classification, ranking, and iterative interactive evaluation.

To optimize computational resources and ensure the most meaningful results, we decided that only the four best-performing models from the sanity check would proceed to the subsequent, more complex evaluations. This approach not only reduces computational costs but also prevents the collection of potentially nonsensical results from models that lack fundamental physical perception capabilities. By focusing on the "smarter" models that demonstrate basic competence, we can obtain more insightful and reliable data on advanced physical reasoning abilities in the later evaluation stages.

4.4.4 Confidence Estimation

This evaluation, positioned as the second level in our assessment framework (see subsection 4.4.1), is designed to assess the model's ability to gauge, in a zero-shot manner, the likelihood that a given proposal will succeed in a physics simulation. In this task, the model is prompted—using a carefully constructed query (see the Confidence Prompt in Appendix B) to provide a single numerical probability estimate representing its confidence that the proposed solution will achieve the goal (i.e., maintain contact between the target objects for the required duration).

We deliberately chose not to use example prompts in this evaluation for three primary reasons. First, including examples could introduce bias into the model's responses; for instance, if the examples implicitly set expectations (such as 100% for correct proposals or 75% for a certain type of incorrect proposal), it might steer the model away from its unbiased assessment. Second, there is uncertainty as to whether the probability of success varies significantly among the three types of incorrect proposals, as discussed in subsection 4.3.5. Providing examples could inadvertently enforce a false equivalence or difference between them. Third, the probability estimate is not an absolute or strictly quantifiable value, but rather a subjective 'vibe' test that reflects the model's intuitive judgment about the dynamic simulation scenario.

For this confidence estimation evaluation, we obtained three unique samples for each proposal type for every template, leading to a total of 600 queries per model. This approach ensures that the evaluation is both robust and free from the bias that could be introduced by example-based prompting, thereby providing a more authentic measure of the model's calibration and its sensitivity to the nuances of physical reasoning tasks.

4.4.5 Binary Classification Task

The Binary Classification Task serves as the secondary evaluation method in our benchmark, as outlined in subsection 4.4.1, directly assessing a model's ability to predict whether a given proposal will successfully solve a physics puzzle. In this task, models are presented with an image of a puzzle that includes an inserted proposal (a red ball), and are asked to determine whether this proposal represents a correct or incorrect solution. This binary decision—correct or incorrect—provides a straightforward measure of the model's physical reasoning capabilities.

To comprehensively evaluate model performance, we conducted this task across multiple experimental conditions. First, we tested models on proposals of varying difficulty levels, as described in subsection 4.3.5, including correct proposals and incorrect proposals categorized as easy, medium, or hard. This stratification allows us to assess whether models exhibit different levels of discrimination ability depending on how subtly incorrect a proposal is.

Second, we systematically varied the prompting strategy to investigate how additional context affects model performance. Specifically, we implemented three distinct prompting conditions:

- **Zero-shot:** Models were provided with only the puzzle image containing the proposal and asked to classify it as correct or incorrect, without any examples.

- **Two-shot:** Models were given two examples (one correct and one incorrect) before being asked to classify the target puzzle. This balanced approach was chosen to avoid biasing the model toward either outcome.

- **Four-shot:** Models were provided with four examples (two correct and two incorrect) before the classification task, offering more extensive context while maintaining balance between outcome types.

Furthermore, for both the two-shot and four-shot conditions, we introduced an additional experimental variable: the number of frames shown in the examples. In the single-frame condition, examples included only the initial state of the puzzle with the proposal inserted. In the dual-frame condition, examples included both the initial state and the final state of the simulation. This design choice was motivated by the hypothesis that showing the final outcome might provide models with additional insights into the physics simulation, potentially strengthening their internal world model and improving prediction accuracy.

It is important to note that in the dual-frame scenario, the examples differed structurally from the main question, which contained only the initial frame. This methodological choice introduced an interesting tension: on one hand, the additional information about simulation progression could enhance the model's understanding; on the other hand, the structural difference between examples and the target question could potentially confuse the model, particularly if it had developed an association between the ground truth and the presence of a second frame.

For each model, we conducted a total of 3000 classification trials, distributed across the various prompting conditions and proposal types. This comprehensive approach allows us to not only assess overall physical reasoning capabilities but also to investigate how different prompting strategies and contextual information affect model performance on this fundamental task.

The prompt for this task was deliberately straightforward, asking: "Given this configuration, will the goal be satisfied?" This simplicity ensures that any performance differences can be attributed to the models' inherent reasoning capabilities rather than to variations in prompt complexity. The complete prompt template can be found in Appendix D.

4.4.6 Ranking Task

The Ranking Task, as described in subsection 4.4.1, is designed to assess the model's nuanced understanding of the physical dynamics and subtleties inherent in the proposals generated for each puzzle. Unlike the binary classification or confidence estimation tasks, which require a simple Yes/No or probability-based response, the ranking task challenges the model to order multiple proposals by their likelihood of successfully meeting the puzzle's goal.

Specifically, the task presents four distinct proposals for a given puzzle—comprising one correct proposal and three incorrect proposals that have been categorized according to varying levels of difficulty (incorrect_hard, incorrect_medium, and incorrect_easy). The explicit instruction is: "Given the 4 proposals for a puzzle, rank them from most likely to succeed to least likely." This approach offers a window into the model's ability to identify and discriminate subtle differences among the proposals, effectively gauging the extent to which the model can perceive partial correctness in a multi-faceted physical simulation. A key feature of the task is the use of explicit labeling within the prompt. As demonstrated in the example provided in the ranking prompt from the attachments (see the example in Appendix C), each proposal is clearly numbered and associated with its corresponding image. This careful labeling ensures that there is no ambiguity regarding which proposal index corresponds to which image, thereby allowing the model to have a reference framework.

Furthermore, the ranking prompt and its delivery have been tested under different experimental conditions—specifically in 0-shot, 1-shot, and 2-shot configurations. These tests were designed to determine whether including prompt examples aids the model in correctly ordering the proposals or, conversely, introduces confusion becuase the proposals are too similar between each other or because the numerous tokens used to represent the images increase the noise. The 0-shot setup provides a baseline where the model must rely solely on its pre-trained understanding; the 1-shot and 2-shot experiments, on the other hand, provide additional context through one or two examples. This systematic variation in prompting allows for a detailed analysis of the trade-offs between guiding the model and maintaining an unbiased evaluation of its inherent reasoning abilities.

For each model, we conducted a total of 450 ranking trials, distributed evenly across the three prompting conditions: 150 trials with 0-shot prompting, 150 trials with 1-shot prompting, and 150 trials with 2-shot prompting. Each template was evaluated three times under each prompting condition, resulting in a comprehensive assessment across all 50 templates.

To mitigate potential biases and ensure a fair evaluation, the four proposals (one correct and three incorrect of varying difficulty) were always randomly presented to the models. This randomization prevents the models from developing positional biases. For subsequent analysis and accurate performance measurement, the randomized indices of the proposals were recorded, allowing us to correctly map the model's rankings back to the actual proposal categories (correct, incorrect_hard, incorrect_medium, incorrect_easy).

Through this task, the evaluation framework directly examines the model's ability to synthesize visual cues and simulation dynamics, providing critical insights into its physical reasoning capabilities. Ultimately, the Ranking Task not only acts as a quick diagnostic tool for discriminating correct from incorrect proposals but also informs future prompt design strategies by revealing whether additional examples benefit or hinder the model's performance.

4.4.7 Interactive Evaluation

For the interactive evaluation, we deployed only the two best performing models from the Binary Evaluation (see subsection 4.4.5) to gauge their in-context learning ability under a dynamic and iterative setup. As described in subsection 4.4.1, unlike static assessments—such as binary classification, ranking, or confidence estimation—this evaluation is designed not only to test the model's inherent physical reasoning but also to analyze its capacity to learn from real-time feedback through iterative refinement. In our approach, a model is allowed up to five sequential attempts per puzzle instance to correct or refine its proposed solution. It is important to emphasize that this five-iteration limit was deliberately imposed not only as a reasonable threshold for assessing learning capabilities but also as a practical measure to control computational costs, as each iteration requires significant processing resources and API usage fees when working with commercial models.

This evaluation is markedly distinct from the others because the assessment mechanism is interactive and manually curated. However, the same logic could readily be integrated into the clembench framework [11] as an additional game. The evaluation protocol relies on a dedicated prompt (available in the Interactive Prompt section in Appendix E of the attachments) where the model is required to provide its proposal strictly in JSON format. Depending on the received proposal (or proposals, in the case of puzzles demanding two proposals), the evaluation framework considers five distinct outcomes:

1. If the model's response is poorly formatted and cannot be parsed as valid JSON, the framework immediately signals a formatting error and returns the required JSON template—forcing the model to retry.

2. If one or more proposals lie outside the simulation boundaries, the framework notifies the model that the proposals are "outside of the screen."

3. If one or more proposals overlap with an existing body in the scene, the framework reports that the proposal(s) are "overlapping with a body present in the image."

4. If the simulation is executed with the provided proposal and the goal state is not reached, the framework displays five screenshot frames, evenly distributed from the start to the end of the simulation. This outcome allows the model to observe the evolution of the scenario and refine its next proposal accordingly.

5. If the goal is reached during the simulation (i.e., the target objects make sustained contact for the required duration), the evaluation is immediately halted and the sample is marked as "solved."

It is important to note that for this evaluation, we focused exclusively on the 1-ball templates to optimize computational resources while still obtaining meaningful insights. Additionally, only the best-performing models from the previous evaluations were tested. This strategic selection allowed us to reduce the overall computational cost of the evaluation while focusing on the most sophisticated models that had already demonstrated strong physical reasoning capabilities in the non-interactive tasks.

Figure 4.5 shows an example of the five frames returned from a simulation for template 00000 iteration 000.

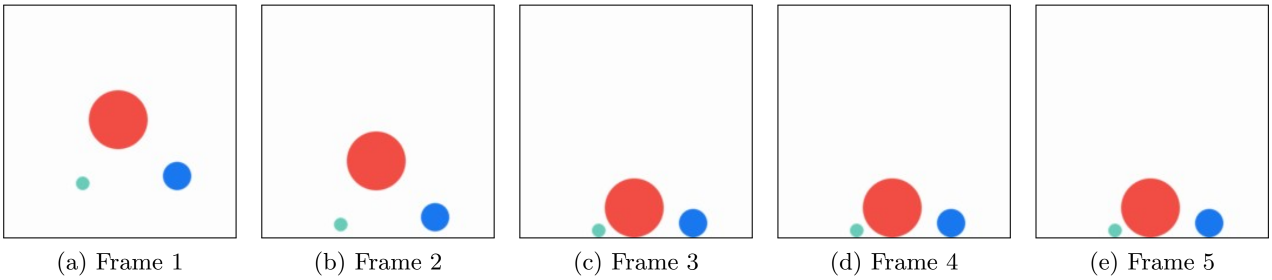

(a) Frame 1 (b) Frame 2 (c) Frame 3 (d) Frame 4 (e) Frame 5

Figure 4.5: Example of the five frames returned from a simulation for template 00000 iteration 000. These frames are evenly distributed from the start to the end of the simulation, providing a comprehensive view of the physical dynamics.

Overall, the interactive evaluation comprises the following key stages:

- Puzzle configuration is loaded, and the model is prompted (through the JSON-based interactive prompt as detailed in Appendix E) to insert a proposal (or two, depending on the puzzle type).
- The evaluation framework verifies the proposal format and spatial validity (ensuring that proposals remain within the visible simulation boundaries and do not overlap with predefined bodies).
- The simulation is then executed, and the framework determines the outcome, which can result in one of the four responses mentioned above.
- This iterative procedure is repeated—up to five attempts per puzzle—until either a valid sequence of proposals leads to the successful achievement of the puzzle goal or the maximum iteration count is reached.
- For further robustness, this evaluation was performed over three iterations for each template across all chosen models.

This extensive evaluation process highlights not just the models' capacity for physical reasoning, but also their ability to benefit from, and adapt to, sequential and contextual feedback. Such an approach provides an insightful measure into the model's in-context learning capabilities, setting it apart from more static evaluation methodologies.

Analytical Insights from Interactive Evaluation

The interactive evaluation methodology offers several analytical opportunities that extend beyond the capabilities of static evaluations, even with our limited dataset of 75 total executions per model (3 iterations per template). By analyzing the results of this evaluation, we can gain preliminary insights into key aspects of model performance:

1. **Basic Learning Capability:** By observing whether models improve their proposals after receiving feedback, we can assess if they demonstrate rudimentary in-context learning. While our limited sample size prevents detailed efficiency analysis, we can identify whether models show any improvement across attempts.

2. **Simple Adaptation Patterns:** The sequential nature of the evaluation allows us to observe basic adaptation patterns in response to feedback. This provides initial indications of how models adjust their proposals after unsuccessful attempts.

3. **Success Rate Analysis:** By tracking whether models eventually solve puzzles within the five-attempt limit, we can compare overall success rates between models, offering a straightforward metric of interactive performance.

4. **General Template Difficulty:** Even with limited data, comparing success rates across different templates can identify which physical scenarios appear more challenging for current VLMs, providing directional insights for future research.

The interactive evaluation also allows us to identify some basic failure patterns, though the limited sample size prevents comprehensive analysis:

- **Repeated Failures:** Cases where models fail to improve despite feedback, suggesting potential limitations in their ability to incorporate physical feedback.

- **Proposal Variability:** Observations on whether models make minor adjustments or dramatic changes between attempts, providing initial insights into their problem-solving approach.

By focusing on the best-performing models from previous evaluations and limiting the assessment to 1-ball templates, we ensure that even our modest dataset (75 executions per model across 3 iterations per template) can provide valuable directional insights. The strict five-iteration limit per puzzle was a critical constraint imposed to manage the substantial computational costs associated with running these evaluations, particularly when using commercial API-based models where each interaction incurs usage fees. Despite these practical limitations, this targeted approach allows us to gain preliminary understanding of how current VLMs perform in interactive physical reasoning tasks, while acknowledging that more extensive evaluation would be required for definitive conclusions.

4.5 Implementation Summary and Workflow

This section provides a high-level overview of how the entire system—from puzzle extraction and proposal generation to final evaluations—fits together into a coherent pipeline. Figure **??** conceptually illustrates the workflow.

1. **Puzzle Template Extraction.** We begin by selecting 50 distinct templates from the PHYRE benchmark, split evenly into single-ball and two-ball puzzle tiers. A custom script leverages the `phyre.loader` module to load each template's puzzle data (shapes, positions, relationships). The puzzle data is then converted into a standardized JSON format, capturing all geometry and metadata. During this phase all relevant puzzle information is stored in a MongoDB database for efficient retrieval and manipulation.

2. **Physics Simulation and Verification.** Each puzzle (now in JSON form) is simulated in our re-engineered `Box2D/pybox2d` environment, using precisely matched physical constants (e.g., gravity, friction). The simulation stops if: (i) the two target bodies have remained in contact for three seconds, (ii) objects become static, or (iii) the maximum frame limit is reached. Early on, we validated that the puzzle states (with no newly inserted proposals) are stable and do not inadvertently solve themselves; any that do are discarded.

3. **Proposal Generation and Labeling.** For each puzzle:

   - A brute force sampler attempts up to 10,000 insertions (one or two red balls, depending on the puzzle's template) until it finds a valid *correct proposal* that achieves the goal.
   - Once a correct proposal is found, three *incorrect proposals*—graded as *easy*, *medium*, or *hard*—are generated by perturbing the correct proposal's position or radius. Each incorrect proposal is tested in the same simulation engine to verify that it fails the goal. If all three fail in the intended ways, they are attached to that correct proposal.

- This process yields exactly 20 ”puzzle + correct proposal” pairs per template. Each pair includes one correct proposal and three systematically determined incorrect proposals

In this manner, the final dataset is curated to contain enough correct/incorrect pairs for subsequent evaluations.

4. **Non-Interactive Evaluation:** Four evaluation tasks—Sanity Check, Confidence Estimation, Binary Classification, and Ranking—are all conducted in a non-interactive, single-turn manner:

   - **Sanity Check:** Models receive the *final frame* (showing the puzzle with the red ball) and must answer whether the two target objects are in contact: Poor performers are filtered out here.
   - **Confidence Estimation:** Models see the *initial frame* (puzzle + red ball) plus limited textual context about simulation details. They return a single percentage indicating their confidence in goal success.
   - **Binary Classification:** Models are shown one puzzle + proposal in its initial frame, then predict whether the goal will be achieved (Yes/No). We systematically vary prompting (0-, 2-, or 4-shot) and test the addition of the final frames (other than the starting ones) in the prompt.
   - **Ranking:** Models view four proposals (one correct + three incorrect) side by side in their initial configuration and must order them from most to least likely to succeed. We experiment with 0-, 1-, and 2-shot prompting.

5. **Interactive Evaluation:** The highest-performing models from the non-interactive tasks proceed to a multi-step, interactive scenario:

   (a) The puzzle is presented *without* a red ball. The model must propose a solution in strict JSON format (x, y, radius).

   (b) The system validates if the ball is within boundaries and not overlapping existing objects. If invalid, the model is asked to correct it.

   (c) Once valid, the system simulates the puzzle. If the goal is *not* achieved, the model receives 5 ”key frames” capturing the puzzle's evolution over time and can propose another ball—up to 5 iterations total.

   (d) The evaluation ends when either the goal is met or the model exhausts its 5 attempts.

   This interactive procedure tests iterative refinement and in-context learning capabilities that are not captured by single-turn predictions.

This workflow forms the end-to-end pipeline: from an initial puzzle template and raw geometry, to labeled puzzle variants with correct/incorrect proposals, and finally to comprehensive multi-task VLM evaluations that assess everything from basic perceptual checks to advanced iterative problem-solving.

4.5.1 Web-based Prompt Visualization Tool

An additional component of our methodology involved creating a small web-based application to facilitate quick inspection and testing of prompts. This tool (built with Streamlit) operates locally and is not yet accessible online. Its primary purpose is to visualize how the final prompts are composed, changed, or expanded with few-shot examples. By using a simple interface, one can load a specific puzzle template, select the prompt type, adjust variables such as the number of frames to display, set the proposal tier (correct/incorrect), and choose from among the different VLM models evaluated in the experiment.

Appendix F presents a screenshot of this web interface. On the left-hand side, the user can dynamically modify the prompt's degrees of freedom—e.g., how many examples to show for few-shot settings, whether to attach the final simulation frame, etc. Once these inputs are set, the tool generates the exact textual prompt. This helps confirm that each model receives precisely the intended prompt structure and visual resources.

In addition, the tool also supports the iterative prompt design for the interactive evaluation. That is, as the model receives intermediate simulation screenshots or rejects an overlapping placement, the interface updates in real time so that testers can visualize how the conversation evolves. Although we employed it mostly for debugging and demonstration purposes, it can serve as a lightweight local environment for quick iteration on prompt formats.

Because this site is designed purely for local testing, we did not host it publicly. Nonetheless, the framework can be reused for internal evaluation or adapted in future research to broaden community access. The utility lies in having a rapid, visual overview of each puzzle's baseline conditions, the prompt expansions (few-shot examples, number of frames, final states), and the model's responses, all in one interactive environment.

5 Results

5.1 Quantitative Results

5.1.1 Sanity Check Evaluation Results

The sanity check evaluation served as our initial assessment of the models' fundamental physical reasoning capabilities. This evaluation focused on a straightforward task: determining whether target objects were in contact based on final state images. This basic assessment provides critical insights into each model's ability to interpret visual physical relationships before proceeding to more complex reasoning tasks.

Figure 5.1 presents the accuracy of each model in verifying whether the target objects are in contact based on the final state images. This visualization reveals substantial performance variations across the evaluated Vision Language Models (VLMs).

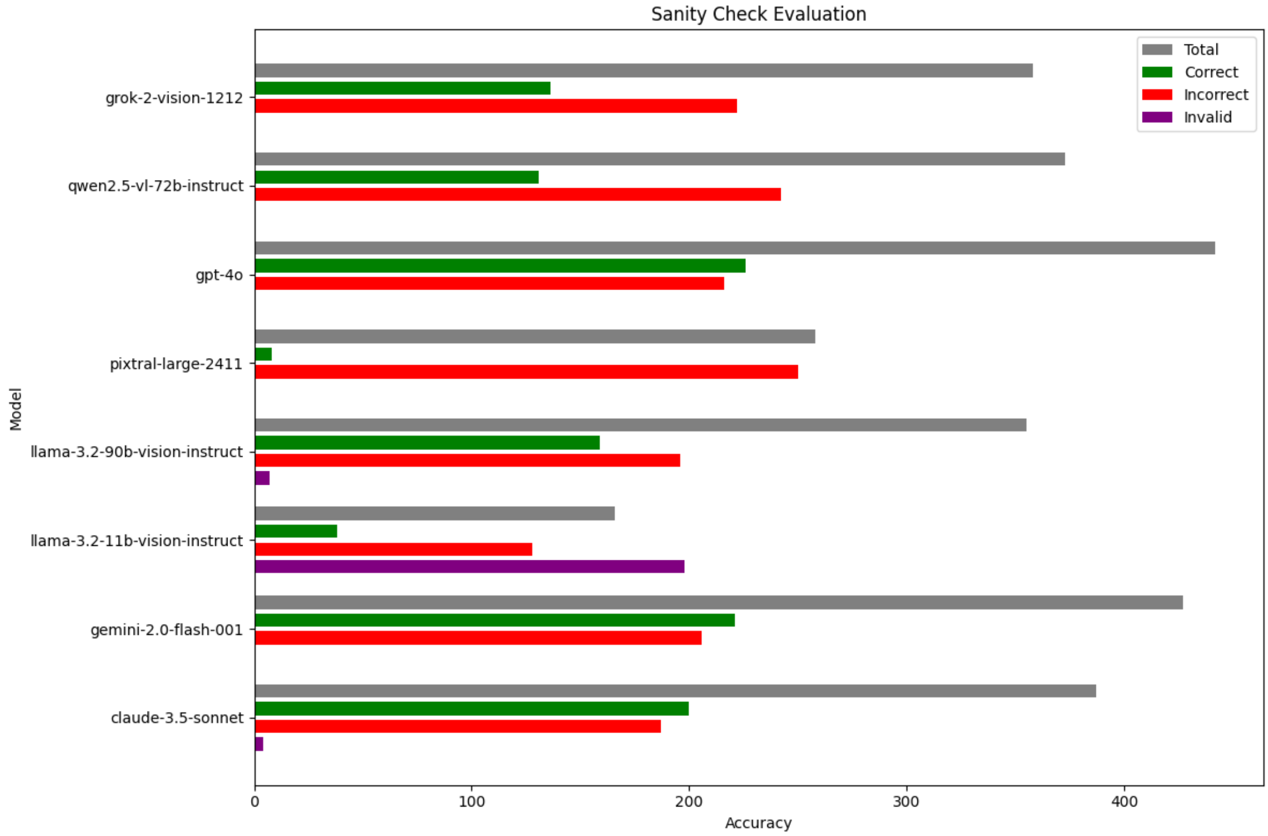


Figure 5.1: Accuracy per model in the Sanity Check Evaluation. The chart illustrates significant performance disparities across evaluated models, with GPT-4o, Claude-3.5-sonnet, Gemini-2.0-flash-001, and Qwen2.5-vl-72b-instruct demonstrating notably superior accuracy compared to other models.

Performance Distribution Patterns

Our analysis revealed a clear stratification in model performance across the evaluated VLMs. Four models emerged as definitive performance leaders: **GPT-4o**, **Claude-3.5-sonnet**, **Gemini-2.0-flash-001**, and **Qwen2.5-vl-72b-instruct**. These models demonstrated substantially higher accuracy compared to their counterparts, with consistently balanced performance across the evaluation tasks.

The superior results from these top-performing models indicate a robust capacity to interpret visual cues necessary for determining goal achievement in physical scenarios. This proficiency suggests these models possess more sophisticated visual reasoning capabilities, enabling them to accurately assess spatial relationships and object interactions within the provided images.

In stark contrast, several models exhibited notably poor performance. **Pixtral-large-2411** and **llama-3.2-11b-vision-instruct** struggled considerably with the evaluation tasks, recording significantly lower accuracy rates in identifying correct outcomes. **Pixtral-large-2411**, in particular, produced almost negligible correct responses, revealing substantial limitations in its ability to understand and interpret the target scenarios.

This performance gap highlights the considerable variation in visual reasoning capabilities among current VLMs, even for relatively straightforward physical relationship tasks. The stark performance disparity suggests fundamental differences in how these models process and interpret visual information related to physical interactions.

Response Validity Analysis

Beyond raw accuracy, response validity emerged as another critical differentiating factor in our evaluation. The ability to generate valid responses—those conforming to task instructions and providing meaningful assessments—varied significantly across models.

The **llama-3.2-11b-vision-instruct** model generated a disproportionately high number of invalid responses, indicating fundamental challenges in either processing visual data or conforming to task instructions. This pattern was particularly pronounced across the llama-3.2 model series, with the 11b version responsible for the majority of invalid outputs.

Such response invalidity suggests architectural or training limitations that impair these models' ability to engage meaningfully with visual physical reasoning tasks. The prevalence of invalid responses indicates that some models struggle with the basic task formulation, failing to properly interpret the instructions or generate responses in the expected format.

This finding has significant implications for real-world applications, as models producing invalid responses would require additional error handling and validation mechanisms, potentially limiting their utility in autonomous physical reasoning scenarios.

Model Selection Criteria

Based on the comprehensive evaluation of accuracy, invalid response rates, and economic considerations, we selected only four models for the more demanding subsequent evaluation tasks: **GPT-4o**, **Claude-3.5-sonnet**, **Gemini-2.0-flash-001**, and **Qwen2.5-vl-72b-instruct**.

While **Grok-2-Vision-1212** and **llama-3.2-90b-vision-instruct** demonstrated performance levels comparable to **Qwen2.5-vl-72b-instruct**, they were excluded from further evaluation due to cost efficiency considerations and the higher incidence of invalid responses observed in the llama-3.2 series.

This selection process ensured that subsequent, more complex evaluations would focus on models with demonstrated baseline competence in physical reasoning tasks. By concentrating on the most capable models, we could explore the upper bounds of current VLM capabilities in physical reasoning while avoiding the confounding effects of models struggling with basic task comprehension.

Implications for Advanced Reasoning Tasks

The performance patterns observed in this sanity check have significant implications for the models' expected capabilities in more complex tasks. The strong performance demonstrated by the selected models suggests they are likely to excel in more sophisticated evaluations, including confidence estimation, ranking, binary classification, and interactive scenarios.

Conversely, models that struggled with this fundamental assessment, particularly those producing numerous invalid responses, would likely encounter even greater difficulties with the increased complexity of advanced evaluation tasks. This observation supports our decision to focus subsequent evaluations on the top-performing models.

A particularly noteworthy observation was the substantial discrepancy between the total number of evaluations and the number of correct responses for most models. With the exception of **GPT-4o**, **Gemini-2.0-flash-001**, and **Claude-3.5-sonnet**, all evaluated models showed significant performance gaps, highlighting the varying capabilities among current VLMs in interpreting visual instructions related to physical relationships.

Summary of Sanity Check Findings

Our sanity check evaluation revealed several key insights into the current state of physical reasoning capabilities in Vision Language Models:

1. A clear performance hierarchy exists among current VLMs, with only a select few models demonstrating robust physical reasoning capabilities even in basic tasks.
2. Response validity varies significantly across models, with some models struggling to generate meaningful responses that conform to task instructions.
3. The top-performing models (**GPT-4o**, **Claude-3.5-sonnet**, **Gemini-2.0-flash-001**, and **Qwen2.5-vl-72b-instruct**) demonstrate sufficient capability to warrant further evaluation in more complex physical reasoning tasks.
4. Among the top performers, **GPT-4o** demonstrated the most balanced and reliable performance, suggesting it possesses the most robust visual physical reasoning capabilities among the evaluated models.

These findings establish a baseline understanding of current VLM capabilities in physical reasoning tasks and provide context for interpreting the results of subsequent, more complex evaluations. The clear stratification in model performance highlights the ongoing challenges in developing VLMs with robust physical reasoning capabilities, while also identifying the most promising models for further investigation.

5.1.2 Confidence Estimation Results

Following the sanity check evaluation, we conducted a comprehensive confidence estimation analysis to assess how well the selected models could gauge their own certainty when evaluating physical reasoning tasks. This evaluation provides critical insights into model calibration—the alignment between confidence scores and actual performance—as well as patterns of under- or overconfidence across varying difficulty levels. Notably, all models produced valid responses in this evaluation phase, with zero invalid samples detected.

Figure 5.2 presents violin plots illustrating confidence distributions across different template difficulties (easy, medium, hard) and proposal correctness (correct vs. incorrect). These distributions reveal significant patterns in how models assess their own certainty in physical reasoning tasks.

Confidence Distribution Patterns

Our analysis revealed distinct confidence distribution patterns across the evaluated models. **Claude-3.5-sonnet** and **GPT-4o** demonstrated wide-ranging confidence scores, producing classic violin-shaped distributions that span from low to high confidence values. This pattern indicates these models are willing to express varying degrees of certainty depending on the specific scenario presented.

In stark contrast, **Gemini-2.0-flash-001** and **Qwen2.5-vl-72b-instruct** exhibited markedly different behavior, with strongly skewed, water drop-shaped distributions heavily weighted toward lower confidence values. This conservative approach suggests these models are programmed to express hesitancy in their physical reasoning capabilities, rarely assigning high confidence even when their predictions are correct.

These qualitative differences in confidence distribution shapes reveal fundamentally different uncertainty estimation strategies implemented in these vision language models. The broader distributions of Claude and GPT-4o suggest these models may have been designed or trained to express more nuanced levels of certainty, while Gemini and Qwen appear to implement a more conservative confidence estimation approach.

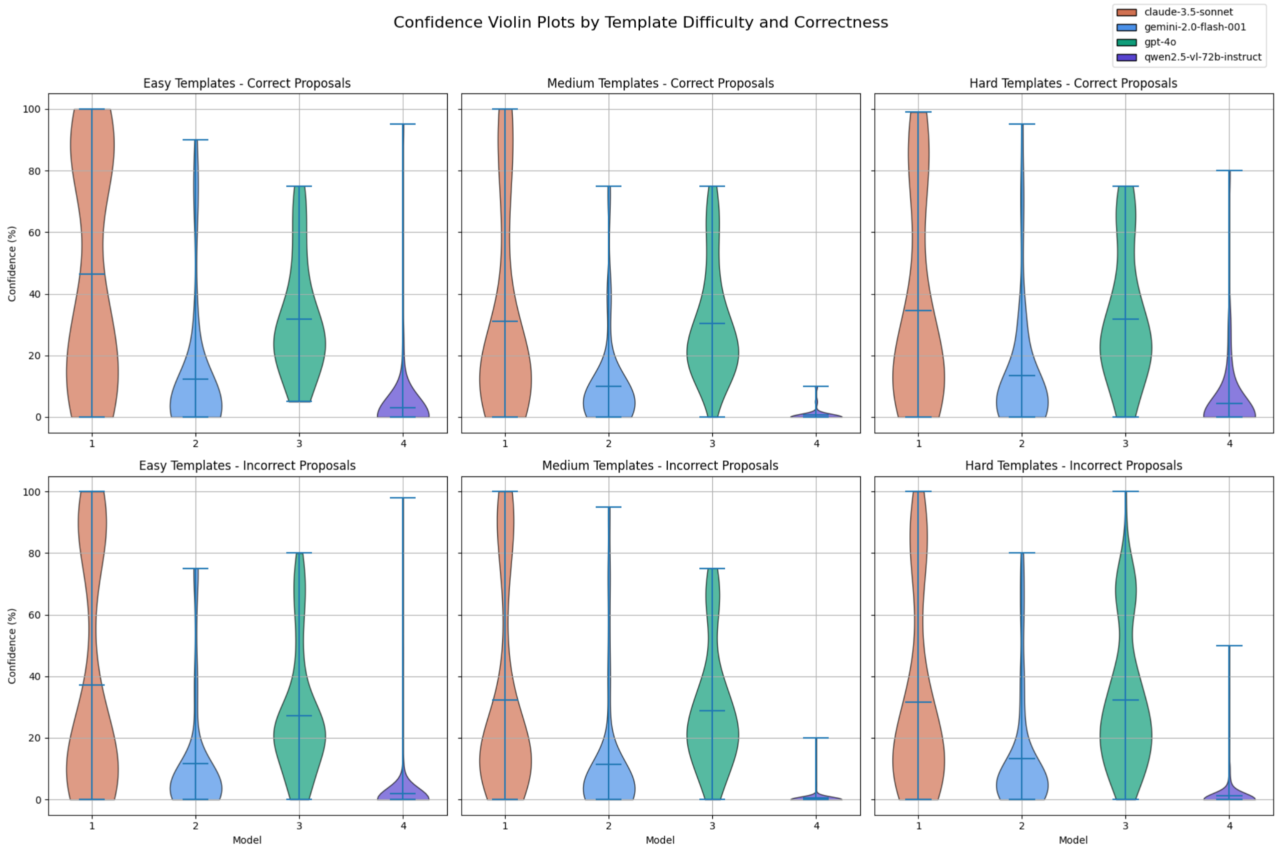


Figure 5.2: Confidence Violin Plots by Template Difficulty and Correctness. The plots illustrate confidence distributions across different template difficulties (easy, medium, hard) and proposal correctness (correct vs. incorrect) for the four selected models. The violin shapes reveal distinct confidence estimation strategies, with Claude-3.5-sonnet and GPT-4o showing wider distributions while Gemini-2.0-flash-001 and Qwen2.5-vl-72b-instruct exhibit more conservative, lower-value distributions.

Sensitivity to Proposal Correctness

A critical aspect of effective confidence estimation is the ability to assign higher confidence to correct predictions and lower confidence to incorrect ones. As illustrated in Figure 5.2, all models demonstrated at least partial sensitivity to proposal correctness, generally assigning higher confidence scores to correct proposals compared to incorrect ones. This pattern was particularly pronounced in **Claude-3.5-sonnet** and **GPT-4o**, especially evident in the correct-easy template category.

However, this sensitivity was inconsistent across difficulty levels and models. The distinction between confidence assigned to correct versus incorrect proposals diminished considerably for medium and hard templates, suggesting that as physical reasoning tasks become more complex, models struggle to accurately assess their own performance. This finding aligns with the intuitive expectation that more challenging physical scenarios would introduce greater uncertainty in model predictions.

Sensitivity to Template Difficulty

An intriguing finding emerged regarding model sensitivity to intrinsic puzzle complexity. The **Claude-3.5-sonnet** model uniquely demonstrated a clear pattern of higher confidence for easier puzzle templates compared to medium or hard ones. This suggests Claude possesses at least partial sensitivity to the inherent difficulty of physical reasoning tasks, a capability not clearly observed in the other evaluated models.

This difficulty-aware confidence calibration represents a sophisticated aspect of physical reasoning, as it indicates the model can recognize and assess the relative complexity of different scenarios. The absence of this pattern in other models suggests that while they may perform well on physical reasoning tasks, they lack the meta-cognitive capability to recognize when a task is inherently more challenging.

Implications for Model Calibration

The observed confidence patterns have significant implications for model calibration in physical reasoning tasks. The ideal calibrated model would demonstrate both high confidence for correct predictions and low confidence for incorrect ones, with confidence levels appropriately adjusted based on task difficulty.

Our findings suggest that current vision language models have achieved only partial calibration. While they show some ability to distinguish between correct and incorrect predictions through confidence scores, this ability deteriorates as task complexity increases. Furthermore, most models (with the exception of Claude-3.5-sonnet) do not appear to adjust their confidence based on the intrinsic difficulty of the physical reasoning task.

The conservative confidence approach adopted by **Gemini-2.0-flash-001** and **Qwen2.5-vl-72b-instruct** may represent a design choice to avoid overconfidence, which is often considered more problematic than underconfidence in AI systems. However, this conservative approach limits these models' ability to express high certainty even when their predictions are correct, potentially reducing their utility in applications where confidence-based decision thresholds are important.

Summary of Confidence Estimation Findings

Our confidence estimation analysis revealed several key insights into how current vision language models assess their own certainty in physical reasoning tasks:

1. Models employ distinctly different confidence distribution strategies, with Claude-3.5-sonnet and GPT-4o utilizing wide-ranging confidence values, while Gemini-2.0-flash-001 and Qwen2.5-vl-72b-instruct adopting a consistently conservative approach.

2. All models demonstrated at least partial sensitivity to proposal correctness, assigning higher confidence to correct solutions than incorrect ones, though this sensitivity diminished with increasing task complexity.

3. Only Claude-3.5-sonnet showed clear sensitivity to template difficulty, suggesting most current models lack sophisticated meta-cognitive awareness of task complexity in physical reasoning scenarios.

4. The absence of clear differentiation in confidence scores among incorrect proposal types indicates models struggled significantly with subtle physical distinctions, highlighting a limitation in their physical reasoning capabilities.

These findings highlight both the potential and current limitations of vision language models in physical reasoning tasks. While these models have made significant progress in basic physical reasoning capabilities, their confidence estimation abilities remain imperfectly calibrated, particularly for complex physical scenarios. This suggests an important direction for future research: developing models that can not only solve physical reasoning tasks but also accurately assess their own certainty across varying levels of task complexity.

5.1.3 Binary Classification Results

Following our confidence estimation analysis, we conducted a comprehensive evaluation of the models' ability to distinguish between correct and incorrect proposals in physical reasoning tasks. This binary classification task represents a fundamental challenge in physical reasoning, requiring models to predict whether a proposed solution would successfully achieve the target goal. Notably, invalid responses were negligible, with only 1 from **Claude-3.5-sonnet** and 5 from **Gemini-2.0-flash-001**.

Figure 5.3 presents the accuracy of each model across different few-shot configurations (0-shot, 2-shot, 4-shot) and frame conditions (1 frame vs. 2 frames), broken down by proposal type (correct vs. incorrect).

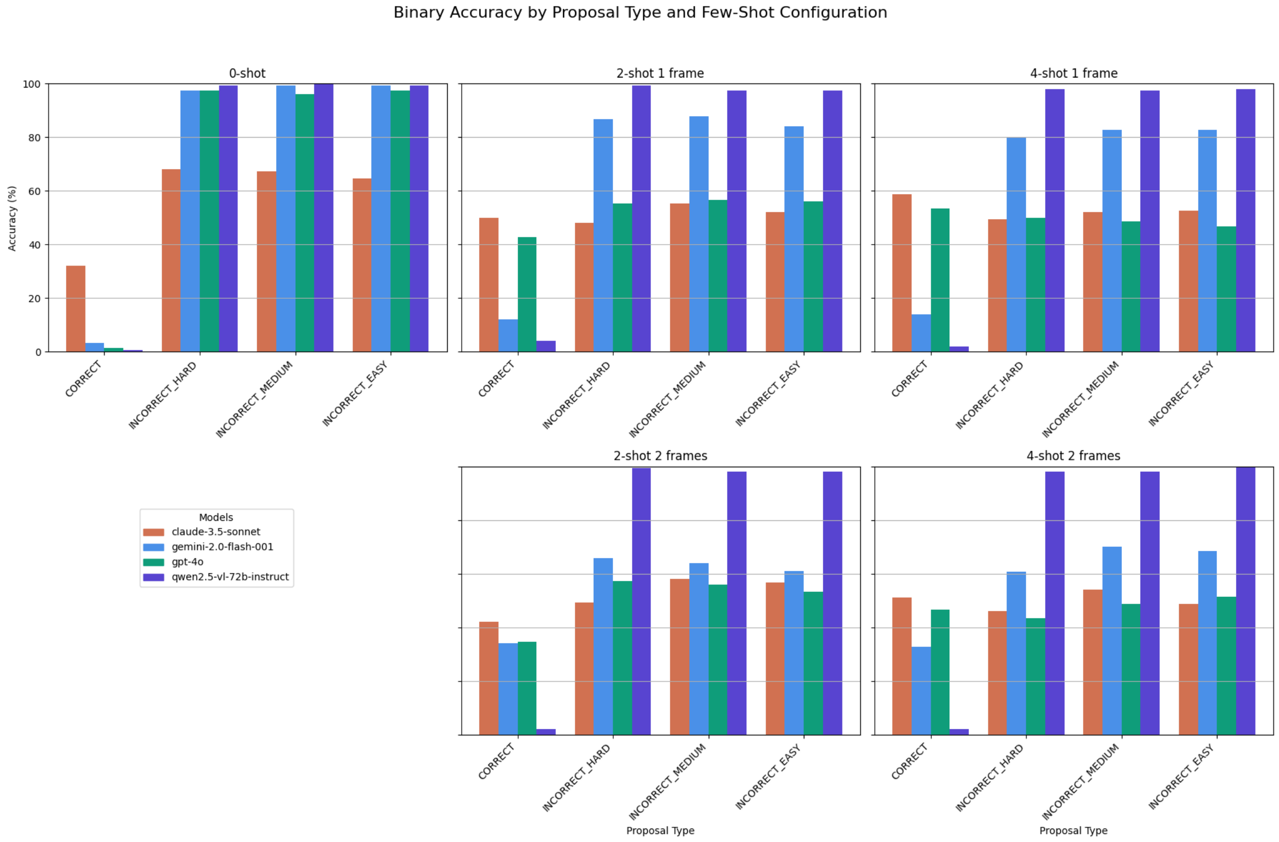


Figure 5.3: Binary Accuracy by Proposal Type and Few-Shot Configuration. The chart illustrates the performance of each model in distinguishing between correct and incorrect proposals across different few-shot counts (0-shot, 2-shot, 4-shot) and frame conditions (1 frame vs. 2 frames). The visualization reveals significant variations in model biases and the impact of additional examples on classification performance.

Model Bias Patterns

Our analysis revealed distinct bias patterns across the evaluated models. **Claude-3.5-sonnet** demonstrated a notable positive bias, performing relatively better than other models at identifying correct proposals in 0-shot conditions (approximately 32% accuracy), but significantly underperforming in identifying incorrect proposals across all conditions. This positive bias suggests Claude has a tendency to predict successful outcomes in physical reasoning tasks, even without supporting evidence.

In stark contrast, **Gemini-2.0-flash-001**, **GPT-4o**, and **Qwen2.5-vl-72b-instruct** exhibited strong negative biases in 0-shot scenarios. These models achieved remarkably low accuracy on correct proposals (below 5%) while maintaining high accuracy on incorrect proposals (approximately 95%). This pattern indicates a default assumption of failure in physical reasoning tasks when no examples are provided, suggesting these models may be calibrated toward conservative predictions in uncertain scenarios.

Particularly noteworthy was **Qwen2.5-vl-72b-instruct**, which maintained this strong negative bias even with additional examples. While this resulted in the highest accuracy for incorrect proposals, it came at the expense of extremely poor performance on correct proposals, making its overall performance questionable despite seemingly high accuracy on negative cases. This persistent negative bias suggests fundamental limitations in Qwen's ability to recognize successful physical interactions, even with contextual guidance.

Impact of Few-Shot Examples

The introduction of few-shot examples had a profound and varied impact across the evaluated models. In the 0-shot condition, most models defaulted to strong biases (either positive or negative), suggesting limited intrinsic physical reasoning capabilities without contextual guidance.

As we increased the number of examples to 2-shot and 4-shot configurations, **Claude-3.5-sonnet** and **GPT-4o** showed substantial improvements in accuracy for correct proposals. Claude's accuracy on correct proposals increased from approximately 32% in 0-shot to 58% in 4-shot conditions, while GPT-4o demonstrated an even more dramatic improvement from below 5% to approximately 55%. These improvements suggest these models can effectively leverage contextual examples to refine their physical reasoning capabilities.

A particularly intriguing pattern emerged in the 4-shot condition for Claude and GPT-4o, where the sum of accuracy on correct and incorrect proposals exceeded 100% (e.g., Claude: 58% correct + 50% incorrect; GPT-4o: 55% correct + 50% incorrect). This pattern suggests that with multiple examples, these models develop genuine discriminative capabilities rather than simply shifting their bias from negative to positive. The ability to simultaneously improve on both correct and incorrect proposal identification indicates a more sophisticated understanding of physical dynamics rather than a mere prediction inversion.

Gemini-2.0-flash-001 also showed improvement with additional examples, though less pronounced than Claude and GPT-4o. In contrast, **Qwen2.5-vl-72b-instruct** remained strongly negatively biased even with multiple examples, suggesting fundamental limitations in its ability to incorporate contextual information for physical reasoning tasks.

Effect of Frame Count on Performance

Our evaluation included variations in the number of frames provided to the models (1 frame showing only the initial state vs. 2 frames showing both initial and final states). Contrary to intuitive expectations, including the final frame of simulations generally did not improve overall accuracy for **Claude-3.5-sonnet**, **GPT-4o**, and **Qwen2.5-vl-72b-instruct**.

Gemini-2.0-flash-001 emerged as a notable exception to this pattern. It demonstrated a clear improvement when provided with the final frame, particularly in identifying correct proposals. This unique sensitivity to additional visual context suggests Gemini may employ different reasoning mechanisms that benefit from seeing the outcome state, while other models may rely more heavily on predictive reasoning from the initial state alone.

This finding has significant implications for model design and application. It suggests that simply providing more visual information does not necessarily enhance physical reasoning capabilities and may potentially introduce confounding factors that obscure intuitive reasoning. The exception of Gemini indicates that model architecture and training methodology may significantly influence how effectively additional visual context is utilized in physical reasoning tasks.

Complementarity Between Correct and Incorrect Proposals

A clear complementary pattern emerged between accuracy for correct and incorrect proposals in most conditions, particularly in the 0-shot scenario. For example, at 0-shot:

- Claude: approximately 32% correct vs. 68% incorrect
- Gemini: approximately 5% correct vs. 95% incorrect
- GPT-4o: approximately 2% correct vs. 95% incorrect

This complementarity suggests that in the absence of examples, models tend to adopt a consistent bias (either positive or negative) rather than making nuanced distinctions between scenarios. However, this complementary relationship weakened at higher few-shot counts, particularly at 4-shot for Claude and GPT-4o. This weakening complementarity, coupled with improved accuracy on both proposal types, indicates a transition from biased prediction to more sophisticated discriminative reasoning with additional contextual guidance.

Implications for Physical Reasoning Capabilities

The binary classification results have significant implications for our understanding of current VLM capabilities in physical reasoning. The strong default biases observed in the 0-shot condition suggest that without contextual guidance, current models rely heavily on prior assumptions rather than sophisticated physical reasoning. This finding highlights the importance of carefully designed prompts and examples in eliciting effective physical reasoning from these models.

The dramatic improvements observed with multiple examples, particularly for Claude and GPT-4o, suggest these models possess latent physical reasoning capabilities that can be effectively activated with appropriate contextual guidance. This pattern indicates that while current VLMs may not have robust intrinsic physical reasoning abilities, they can rapidly adapt their reasoning processes when provided with relevant examples.

The persistent negative bias of Qwen, even with multiple examples, highlights the significant variations in physical reasoning capabilities across different model architectures. This suggests that architectural design and training methodology may play crucial roles in determining a model's capacity for physical reasoning and its ability to incorporate contextual information effectively.

Summary of Binary Classification Findings

Our binary classification evaluation revealed several key insights into the current capabilities and limitations of Vision Language Models in physical reasoning tasks:

1. Models exhibit strong default biases in the absence of examples, with Claude showing a positive bias while Gemini, GPT-4o, and Qwen demonstrate strong negative biases.
2. Multiple few-shot examples significantly enhance discriminative capabilities, particularly for Claude and GPT-4o, enabling them to more accurately identify both correct and incorrect proposals rather than simply shifting their bias.
3. Additional visual context (final frames) generally does not improve performance, with Gemini being a notable exception that benefits from seeing outcome states.
4. Qwen's persistent negative bias, even with multiple examples, highlights significant variations in how effectively different models can incorporate contextual information for physical reasoning.

5. The transition from complementary accuracy patterns in 0-shot to more balanced performance in 4-shot conditions suggests a shift from biased prediction to genuine discriminative reasoning with appropriate contextual guidance.

These findings underscore both the potential and current limitations of vision language models in physical reasoning tasks. While these models demonstrate significant improvements with contextual guidance, their strong default biases and varying responses to additional information highlight the ongoing challenges in developing VLMs with robust, intrinsic physical reasoning capabilities. Future research should focus on reducing these default biases and enhancing models' ability to effectively utilize both visual context and examples for more sophisticated physical reasoning.

5.1.4 Ranking Task Results

Following our binary classification analysis, we conducted a comprehensive evaluation of the models' ability to rank multiple solution proposals in order of effectiveness. This ranking task represents a more complex challenge than binary classification, requiring models to make nuanced distinctions between multiple viable approaches to physical puzzles. Notably, only 2 samples from **Gemini-2.0-flash-001** were invalid (responding with chain-of-thought reasoning rather than the requested numerical ranking).

Figure 5.4 presents the accuracy of each model across different few-shot counts (0-shot, 1-shot, and 2-shot), while Figure 5.5 illustrates the distribution of correct proposal positions in the ground truth ranking.

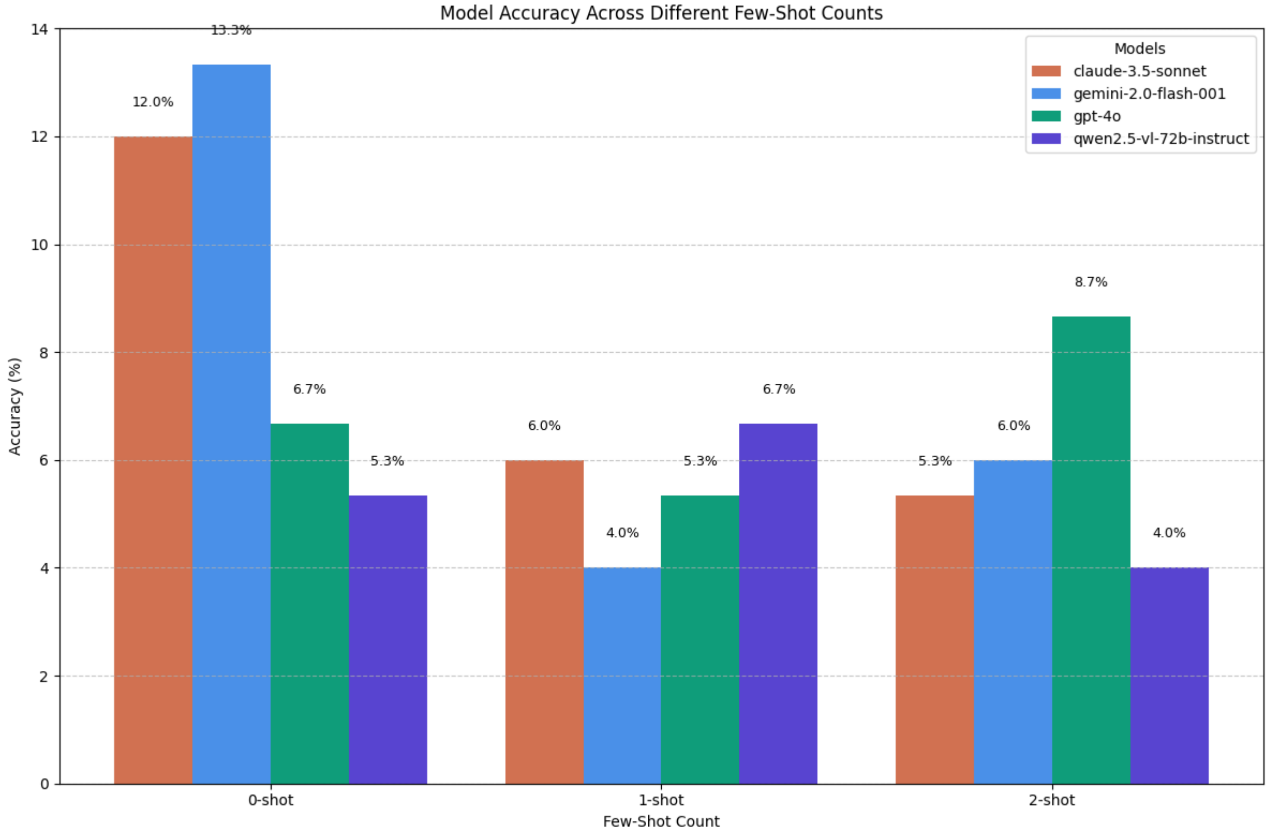


Figure 5.4: Model Accuracy Across Different Few-Shot Counts. The chart illustrates the performance of each model with varying numbers of example demonstrations (0-shot, 1-shot, 2-shot). Overall accuracy remains low (4% to 13%) across all models and shot counts, with Gemini-2.0-flash-001 and Claude-3.5-sonnet performing best in the 0-shot condition, while GPT-4o shows improvement from 1-shot to 2-shot conditions.

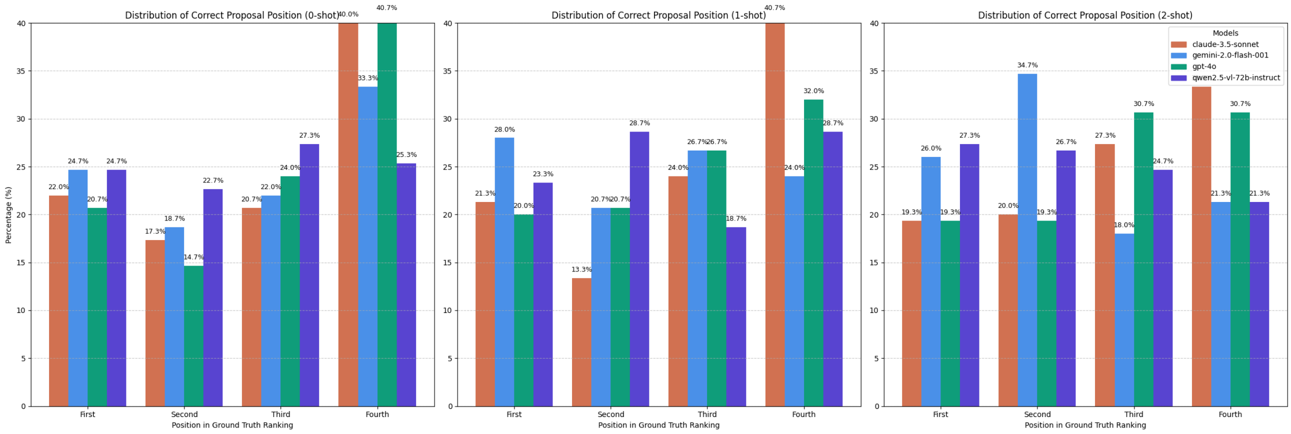


Figure 5.5: Distribution of Correct Proposal Position across different few-shot counts. The charts illustrate the percentage of times each model selects proposals corresponding to each of the four positions in the ground truth ranking. The relatively even distribution across positions suggests models struggle to consistently identify the optimal proposals, with a slight tendency to select the worst-ranked (fourth position) proposals more frequently as shot count increases.

Overall Accuracy Analysis

Our analysis revealed remarkably low overall accuracy across all models and few-shot conditions, ranging from approximately 4% to 13%. This indicates that the ranking task presents a substantial challenge for current Vision Language Models, regardless of the number of few-shot examples provided. The difficulty stems from the need to make fine-grained distinctions between multiple proposals, requiring sophisticated physical reasoning capabilities that exceed those needed for simpler binary classification tasks.

A particularly intriguing pattern emerged regarding the effect of few-shot examples on model performance. Rather than improving with additional examples, we observed a clear performance drop after introducing a single example (1-shot condition), suggesting that models may become confused or biased by individual examples rather than benefiting from them. While some minor recovery occurred with 2-shot prompting, accuracy remained consistently low, confirming the inherent difficulty of the ranking task even with additional contextual guidance.

Among the evaluated models, **Gemini-2.0-flash-001** and **Claude-3.5-sonnet** demonstrated marginally superior performance in the 0-shot condition (approximately 12-13%), suggesting stronger baseline intuition or zero-shot reasoning capabilities for physical ranking tasks. Conversely, **GPT-4o** showed modest improvement from the 1-shot to 2-shot condition (increasing from 5.3% to 8.7%), indicating it may begin to benefit from additional examples more effectively than other models when provided with sufficient context.

Proposal Position Distribution Analysis

Beyond overall accuracy, we analyzed the distribution of correct proposal positions within the ground truth ranking. As illustrated in Figure 5.5, no clear pattern emerged from increasing the number of few-shot examples (0 → 1 → 2 shots). Instead, models appeared to distribute their choices almost evenly among the four possible positions, suggesting near-random selection rather than informed physical reasoning.

A subtle but noteworthy trend was the slightly higher percentage of selections at the ”Fourth” position (the worst-ranked proposal in ground truth) as the number of shots increased. This counter-intuitive pattern suggests that additional examples may actually introduce confusion or misinterpretation rather than clarity, potentially causing models to misidentify the worst proposals as optimal solutions.

The near-uniform distribution across positions further reinforces the conclusion that current VLMs struggle significantly with the subtle physical distinctions required for effective proposal ranking.

This finding contrasts with the more pronounced performance patterns observed in our confidence estimation and sanity check evaluations, highlighting the particular challenge posed by tasks requiring comparative physical reasoning across multiple options.

Potential Explanations for Observed Results

Several factors may contribute to the observed performance patterns in the ranking task. The most evident explanation is the intrinsic difficulty of the task itself—ranking multiple proposals requires models to make fine-grained distinctions between physical scenarios that may appear visually similar but have significantly different physical outcomes. This level of nuanced physical reasoning appears to exceed the current capabilities of even the most advanced VLMs.

Additionally, the ineffectiveness of few-shot examples suggests potential issues with how these examples are presented or interpreted by the models. Rather than providing clarifying guidance, the examples may introduce ambiguity or conflicting patterns that confuse the models' reasoning processes. This highlights the challenge of designing effective few-shot prompts for complex physical reasoning tasks, where subtle differences between examples may lead to unintended interpretations.

The ambiguity of the ranking task formulation itself may also contribute to the observed results. If models interpret the ranking criteria inconsistently or struggle to apply consistent evaluation metrics across different proposals, this would naturally lead to the near-random distribution observed in our analysis.

Implications for Physical Reasoning Capabilities

The ranking task results have significant implications for our understanding of current VLM capabilities in physical reasoning. The consistently low performance across all models and conditions suggests a fundamental limitation in how these models process and compare multiple physical scenarios simultaneously. While they may demonstrate reasonable performance in binary classification or confidence estimation tasks, the more complex cognitive process of ranking multiple options appears to exceed their current capabilities.

This limitation is particularly relevant for real-world applications requiring nuanced physical reasoning, such as robotics planning or scene understanding, where systems must often evaluate and prioritize multiple potential actions. The observed performance suggests current VLMs would struggle significantly in such applications without substantial improvements in their comparative physical reasoning abilities.

Summary of Ranking Task Findings

Our ranking task evaluation revealed several key insights into the current limitations of Vision Language Models in complex physical reasoning tasks:

1. All evaluated models demonstrated consistently low accuracy (4-13%) across few-shot conditions, indicating fundamental difficulties with comparative physical reasoning tasks.

2. Increasing the number of few-shot examples did not improve performance and in some cases led to performance degradation, suggesting challenges in effectively utilizing contextual examples for physical reasoning.

3. The near-uniform distribution of selections across proposal positions indicates models struggle to make meaningful distinctions between physical scenarios of varying quality.

4. **Gemini-2.0-flash-001** and **Claude-3.5-sonnet** showed marginally better zero-shot capabilities, while **GPT-4o** demonstrated modest improvement with additional examples, highlighting different strengths across model architectures.

These findings underscore the significant gap between current VLM capabilities and the level of physical reasoning required for effective ranking of multiple proposals. Future research should focus on enhancing models' ability to make fine-grained comparative judgments across multiple physical scenarios, potentially through improved training methodologies or architectural innovations specifically targeting comparative reasoning tasks.

5.1.5 Interactive Insertion Results

Our interactive evaluation, though limited in scope with only 75 total executions per model (3 iterations per template), provides preliminary insights into how models perform when given multiple attempts with feedback. This section presents the key observations from this constrained dataset, acknowledging that more extensive evaluation would be required for definitive conclusions.

Overall Success Rates

Despite the limited sample size, we observed minimal differences in overall success rates between the two evaluated models. As illustrated in Figure 5.6, both models demonstrated remarkably low success rates: **GPT-4o** achieved a 6.7% success rate (5/75), while **Claude-3.5-sonnet** achieved 5.3% (4/75). This negligible difference suggests that both models face similar fundamental challenges in interactive physical reasoning tasks.

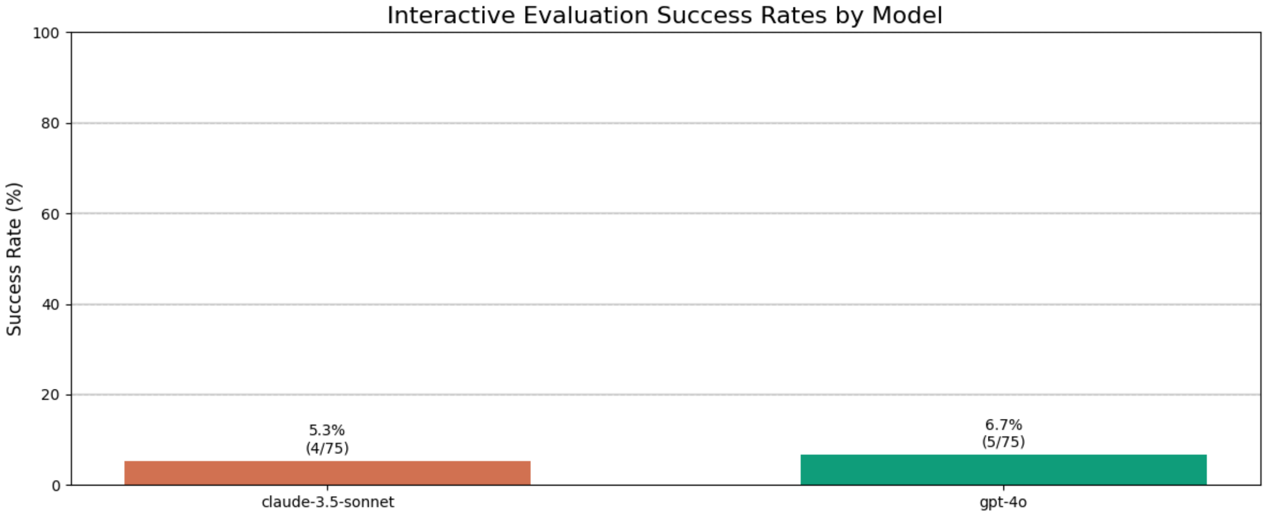


Figure 5.6: Interactive Evaluation Success Rates by Model. The chart illustrates the percentage of puzzles each model successfully solved within the five-attempt limit. Both models demonstrate similarly poor performance, with GPT-4o achieving 6.7% (5/75) and Claude-3.5-sonnet achieving 5.3% (4/75) success rates.

Status Distribution Analysis

Beyond raw success rates, we analyzed the distribution of different outcome statuses across all attempts. As shown in Figure 5.7, both models exhibited remarkably similar patterns in their failure modes.

The most prevalent outcome for both models was GOAL_NOT_REACHED (approximately 74% of attempts), indicating that while the proposed ball placements were valid (not overlapping with obstacles or outside boundaries), they failed to achieve the target physical goal. The second most common outcome was OVERLAPPING (approximately 24% of attempts), where the proposed ball placement intersected with existing obstacles, representing a fundamental spatial reasoning error. Notably, placements OUTSIDE_BOUNDARIES were exceptionally rare (less than 1%), suggesting both models generally understood the basic constraint of keeping the ball within the play area.

A particularly interesting observation is that both models consistently produced valid JSON outputs across all attempts, reflecting their strong capabilities in generating structured data—a strength likely attributable to their synthetic training backgrounds. This contrasts sharply with their difficulties in spatial and physical reasoning tasks.

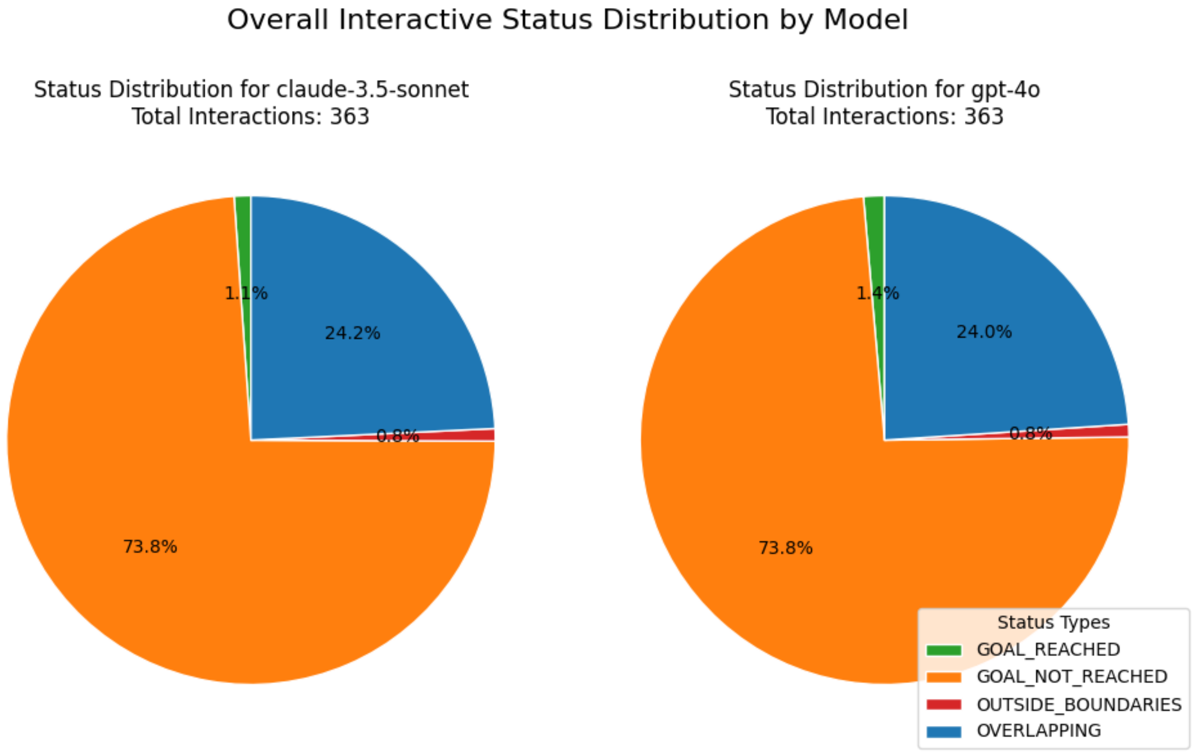


Figure 5.7: Overall Interactive Status Distribution by Model. The pie charts illustrate the distribution of different outcome statuses across all attempts for each model. Both models show nearly identical distributions, with approximately 74% of attempts resulting in GOAL_NOT_REACHED, 24% in OVERLAPPING placements, and only about 1% in placements OUTSIDE_BOUNDARIES. The GOAL_REACHED status represents successful attempts at approximately 1.1% for Claude-3.5-sonnet and 1.4% for GPT-4o.

Template-Specific Performance Patterns

Examining performance across different templates revealed distinct patterns in how models struggled with specific physical scenarios. Figure 5.8 presents a detailed breakdown of outcome statuses by template for each model.

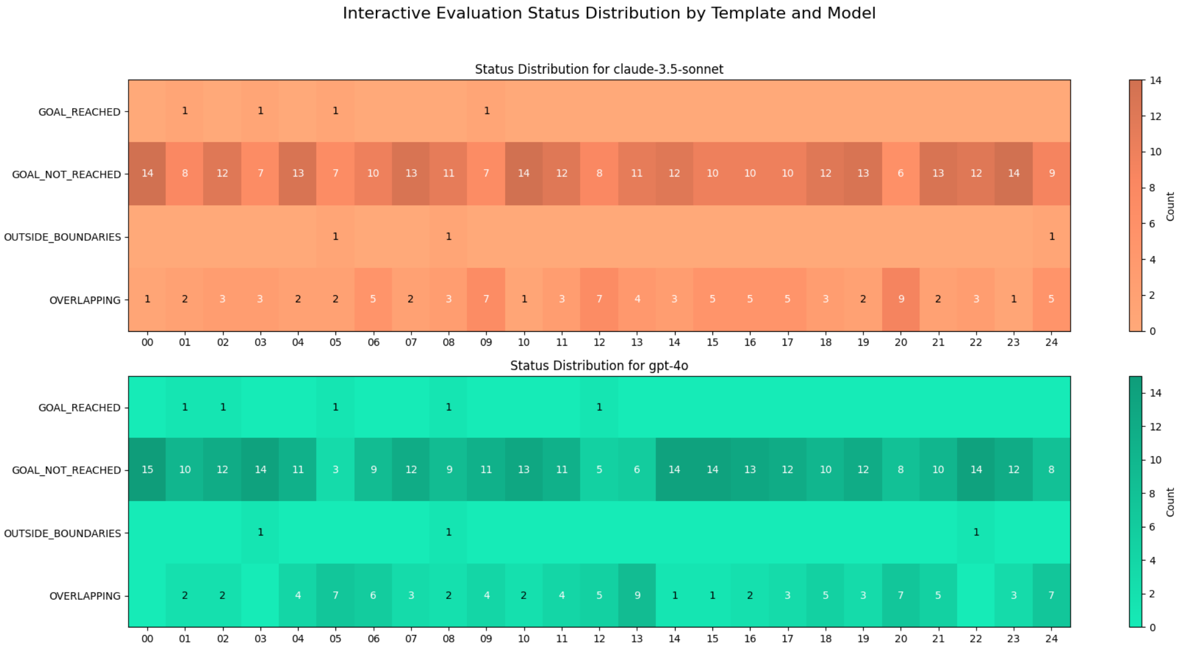


Figure 5.8: Interactive Evaluation Status Distribution by Template and Model. The heatmaps illustrate the distribution of different outcome statuses across all templates for each model. Claude-3.5-sonnet shows frequent overlapping errors in templates 9, 12, and 20, while GPT-4o exhibits similar issues particularly in templates 5, 13, 20, and 24. Successful attempts (GOAL_REACHED) are sparse and sporadic for both models.

Claude-3.5-sonnet exhibited frequent overlapping errors in templates 9, 12, and 20, suggesting particular difficulty with the spatial configurations presented in these scenarios. Successful attempts were sporadic and sparse, with no clear pattern of templates where the model consistently excelled.

Similarly, **GPT-4o** demonstrated concentrated overlapping errors in templates 5, 13, 20, and 24. Its success distribution was also minimal and showed no systematic pattern across templates. The similarity in template-specific challenges across both models suggests that certain physical configurations present inherent difficulties for current VLMs, regardless of their specific architecture or training methodology.

Learning Patterns Across Attempts

A critical aspect of interactive evaluation is understanding whether models improve their proposals after receiving feedback from previous attempts. Figure 5.9 illustrates the distribution of outcome statuses across successive attempts for each model.

Contrary to what might be expected in an iterative learning scenario, neither model demonstrated systematic improvement across successive attempts. Both models continued to produce overlapping errors at similar rates throughout all iterations, suggesting limited ability to learn from previous failures. Successful attempts occurred sporadically in early attempts (primarily attempts 1 or 3) rather than showing a pattern of incremental improvement, indicating that successes may have been more attributable to fortunate initial guesses rather than deliberate refinement based on feedback.

This lack of clear learning progression suggests that current VLMs struggle to effectively utilize feedback in physical reasoning tasks, potentially due to limitations in their ability to model physical dynamics or interpret the implications of previous failures.

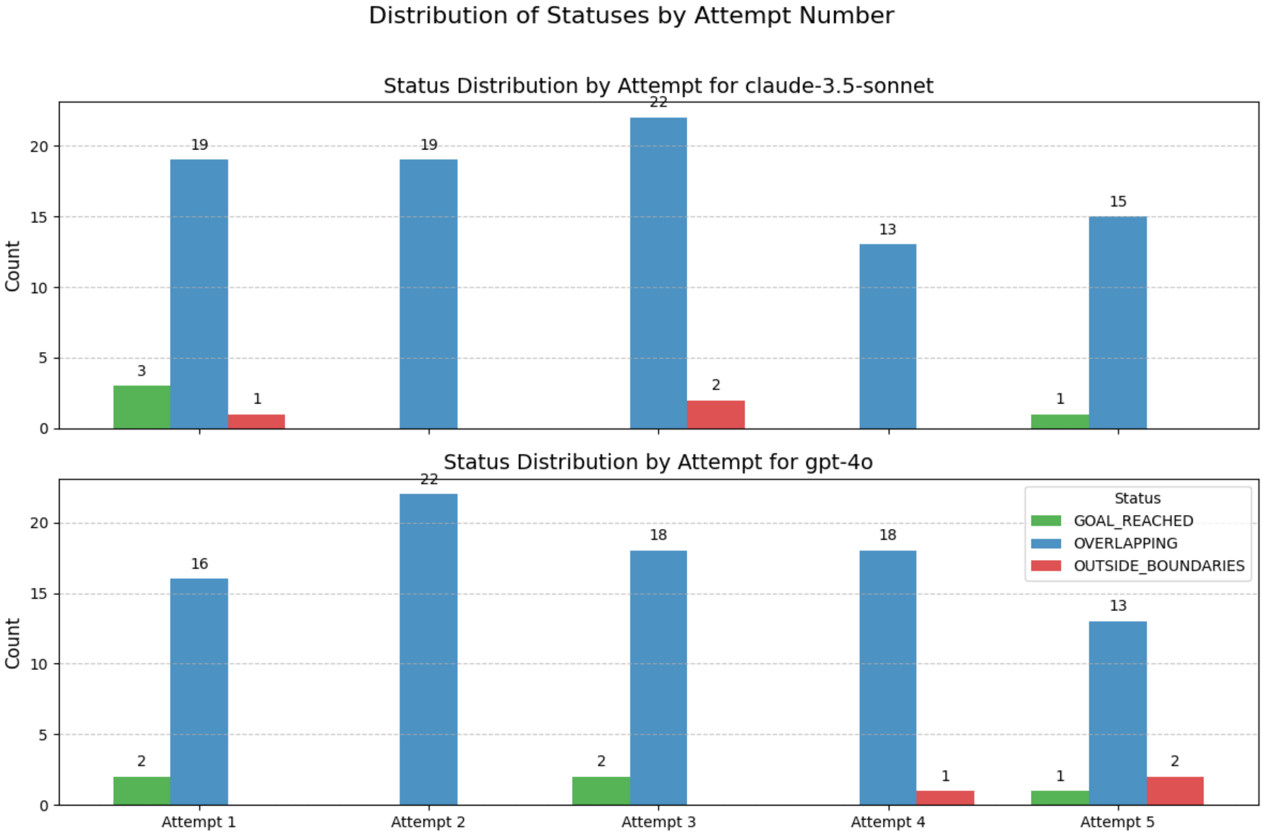


Figure 5.9: Distribution of Statuses by Attempt Number. The bar charts illustrate how outcome distributions evolve across successive attempts for each model. Neither model demonstrates systematic improvement across attempts, with overlapping errors persisting throughout all iterations. Successful attempts (GOAL_REACHED) occur sporadically in early attempts (1 or 3) without showing sustained incremental learning.

Proposal Adjustment Strategies

To gain deeper insights into how models attempted to refine their proposals across attempts, we analyzed the magnitude of changes between consecutive attempts in both position and radius parameters. Figure 5.10 presents violin plots illustrating the distribution of these adjustments across different attempts.

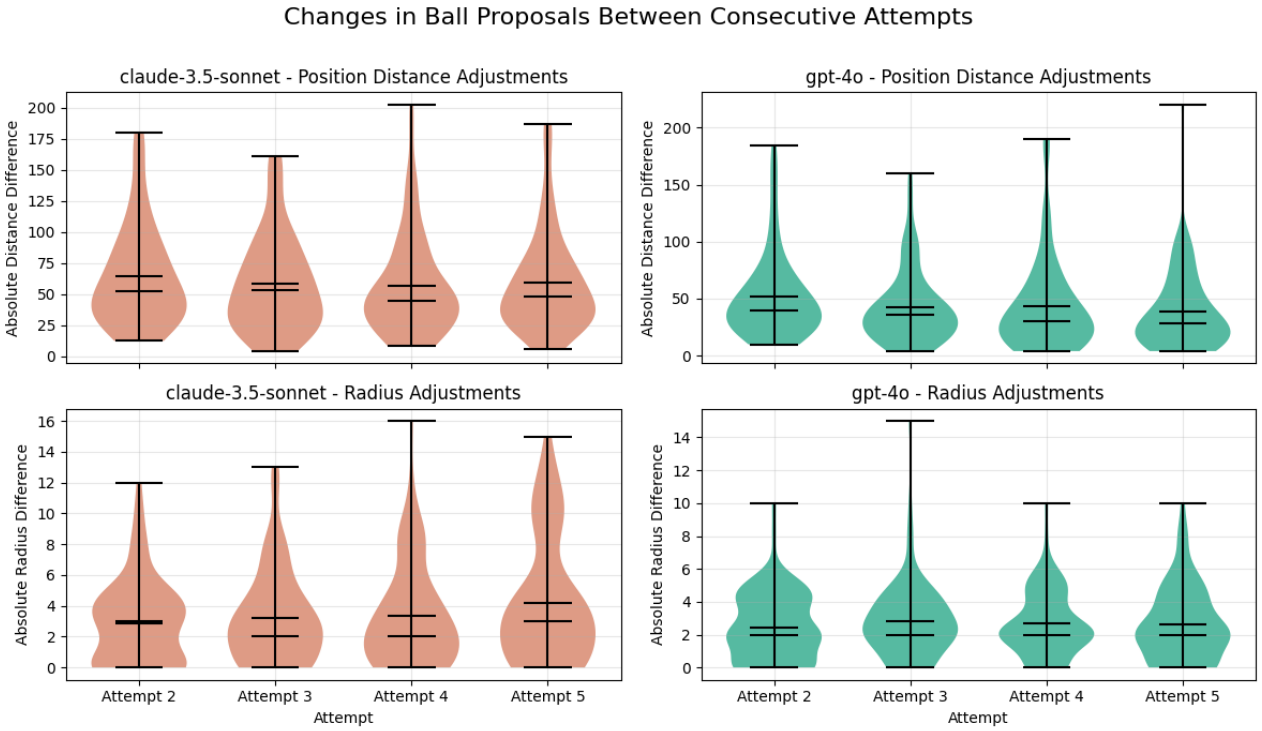


Figure 5.10: Changes in Ball Proposals Between Consecutive Attempts. The violin plots illustrate how models adjust their proposals between consecutive attempts in terms of position (Euclidean distance) and radius. Both models tend to make modest position adjustments (around 50 pixels) and minimal radius adjustments (2-3 pixels), with consistent patterns across attempts 2 through 5.

For position adjustments, **Claude-3.5-sonnet** consistently made modest changes with a median of approximately 50-60 pixels between consecutive attempts. The violin plots reveal a wide distribution, with the majority of adjustments concentrated in the 25-75 pixel range, though some outliers extend beyond 150 pixels. Similarly, **GPT-4o** exhibited comparable position adjustment patterns, with median values around 40-50 pixels and a similar distribution shape.

Regarding radius adjustments, both models demonstrated extremely conservative strategies. **Claude-3.5-sonnet** typically made minimal radius changes of approximately 2-4 pixels between attempts, with the distribution showing a concentration of adjustments in the lower range. **GPT-4o** showed similarly small radius adjustments, generally around 2-3 pixels, with a slightly narrower distribution compared to Claude-3.5-sonnet.

The violin plots reveal that the distribution patterns remain relatively consistent across attempts 2 through 5, suggesting that models maintain similar adjustment strategies throughout the interactive process rather than dramatically changing their approach in later attempts. The predominance of small, incremental adjustments suggests both models adopted conservative refinement strategies, making minimal changes between attempts rather than dramatic revisions. This approach indicates uncertainty in how to effectively respond to feedback or a lack of clear understanding regarding which aspects of their proposals needed significant modification.

Summary of Interactive Evaluation Findings

Our interactive evaluation revealed several key insights into the current limitations of Vision Language Models in iterative physical reasoning tasks:

1. Both evaluated models demonstrated remarkably poor overall performance, with success rates of only 5.3% (Claude-3.5-sonnet) and 6.7% (GPT-4o), indicating fundamental difficulties with interactive physical reasoning.

2. The distribution of failure modes was strikingly similar across models, with approximately 74% of attempts failing to reach the goal despite valid placements, and 24% producing overlapping placements that violated basic spatial constraints.

3. Neither model showed systematic improvement across successive attempts, suggesting limited ability to effectively learn from feedback or previous failures.

4. Both models adopted conservative adjustment strategies, making small, incremental changes to position (approximately 50 pixels) and radius (2-3 pixels) between attempts, without clear strategic refinements.

5. While both models excelled at generating valid structured outputs (100% valid JSON), this strength in formal representation contrasted sharply with their poor performance in spatial and physical reasoning tasks.

These findings highlight a significant gap in current VLM capabilities: while these models can process and generate structured data effectively, they struggle to leverage feedback for iterative refinement in physical reasoning tasks. This limitation suggests that future research should focus on enhancing models' ability to learn from physical feedback and develop more effective strategies for proposal refinement in interactive scenarios.

Limitations and Future Directions

It is important to acknowledge the significant limitations of our interactive evaluation. With only 75 executions per model and 3 iterations per template, the findings presented here should be considered preliminary and directional rather than definitive. Additionally, the restriction to only 5 attempts per puzzle significantly constrains the models' opportunity to learn and adapt their strategies, potentially underestimating their capacity for iterative improvement with extended interaction. A more comprehensive evaluation with larger sample sizes and more attempts per puzzle would be required to draw robust conclusions about model learning patterns and adaptation strategies.

Future work should expand both the breadth (more templates), depth (more iterations per template), and length (more attempts per puzzle) of interactive evaluation to provide more statistically significant insights into how models learn from physical feedback over time. Increasing the number of allowed attempts would be particularly valuable for understanding whether models can eventually converge on successful solutions given sufficient trial-and-error opportunities. Additionally, exploring different feedback mechanisms and prompt structures could potentially enhance models' ability to effectively utilize feedback for iterative refinement in physical reasoning tasks.

A key limitation in interpreting the interactive evaluation results is that each puzzle was attempted only a handful of times. Because the models often failed multiple attempts in a row, we could not definitively characterize if or when the model might eventually discover a correct placement. Knowing the *attempt index* at which each puzzle was first solved would have provided far greater resolution in analyzing how quickly the model grasped the physical constraints or adapted its strategy. Such extended trial-based data collection, however, would have required a substantially larger computational budget (and hence, higher costs) than allotted. This constraint should be kept in mind when generalizing from the current findings to a broader notion of model adaptability.

Another constraint in our study is that multiple puzzles exhibit more than one valid solution—any of several position-radius configurations can lead to a successful outcome. Consequently, we could not define a single "error distance" between the model's chosen proposal and *the* correct proposal. In principle, an analysis of all "correct solution regions" (i.e., continuous areas of valid positions and radii) could have allowed us to calculate how far each proposed placement was from these viable zones. That, however, would require extensive sampling of the proposal space and fine-grained simulation of each candidate, imposing a steep computational overhead. Future research might explore embedding-based or region-based metrics to estimate a more nuanced "error measure" for each puzzle proposal, thereby capturing partial correctness and clarifying how close a model's attempt is to a workable solution.

In addition, while the benchmark's final objective consistently involves making two specific objects touch for three seconds, the puzzles themselves are intricately designed with embedded complexities such as balancing, multi-object collisions, and nuanced dynamic interactions. These underlying physical phenomena require a comprehensive understanding of stability, collision mechanics, and spatial dynamics, meaning that success in these tasks depends on a broad spectrum of physical reasoning rather than mere contact detection.

Finally, the manner in which we prompt the models also shapes the limitations of these evaluations. Attempts to supply additional frames or to give more examples did not always translate to better performance, which suggests that the mere presence of extra data is no guarantee of more sophisticated reasoning. It is possible that different prompt designs—perhaps with structured descriptions of object states, or guided question sequences—might elicit better explanations or more strategic proposals. The unpredictable interplay between prompt design and model performance remains an open question. Some of the negative or positive biases we observed in different models appear to reflect how the examples are shown or the instructions are phrased, rather than purely the underlying physical reasoning ability.

Altogether, these limitations imply that while the PhysIQ evaluations demonstrate progress in capturing some facets of puzzle-based physical reasoning, one must not overstate the breadth or depth of what these tasks measure. They cannot, in their current form, disentangle genuine causal modeling from either memorized or superficial heuristics, nor do they exhaustively test the wide variety of phenomena that would constitute "human-level" intuitive physics. Addressing these concerns—in particular by expanding puzzle diversity, refining the interactive feedback loop, and systematically exploring prompt designs—would yield more conclusive insights into whether current Vision-Language Models can truly internalize and apply the laws of physical reality.

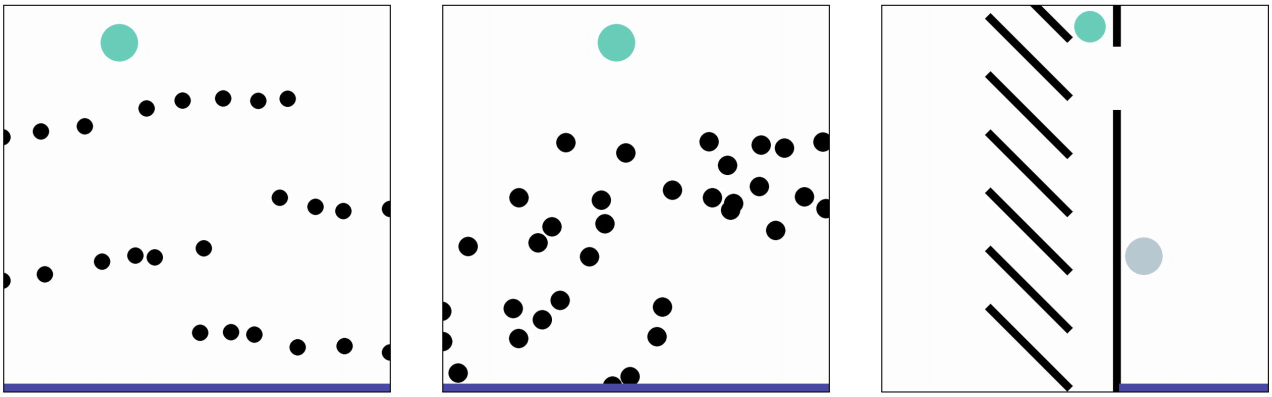
Overall, these insights underline an important conclusion: while sporadic successes in simpler puzzles are promising, they do not yet constitute evidence of a comprehensive understanding of physical reasoning within these models. The path forward involves not only optimizing architectures and training protocols to eliminate overlapping errors in densely populated scenes but also developing more nuanced evaluation strategies to discern between genuine reasoning ability and fortuitous successes in streamlined scenarios.

5.2 Qualitative Observations

In this section, we provide a detailed qualitative examination of the performance of the evaluated Vision Language Models (VLMs) on the physical reasoning puzzles. Our analysis is aimed at uncovering the underlying factors that differentiate failure cases from successes, especially in terms of spatial complexity and the inherent structure of the puzzles.

The experimental results reveal a marked dichotomy between certain templates. Specifically, puzzles corresponding to templates 12, 13, and 20 exhibit a high frequency of overlapping proposals (see Figure 5.11). In these challenging cases, the models are required to insert proposals with high precision. For template 20, the proposal must be placed in close proximity to other objects, necessitating an exact spatial alignment. Similarly, templates 12 and 13 are characterized by scenes where a multitude of objects are scattered throughout the frame, thereby raising the likelihood of accidental overlapping when proposals are generated. Such densely populated configurations demand fine-grained spatial reasoning, and the overlapping outcomes observed suggest that the current models struggle to meet these requirements.

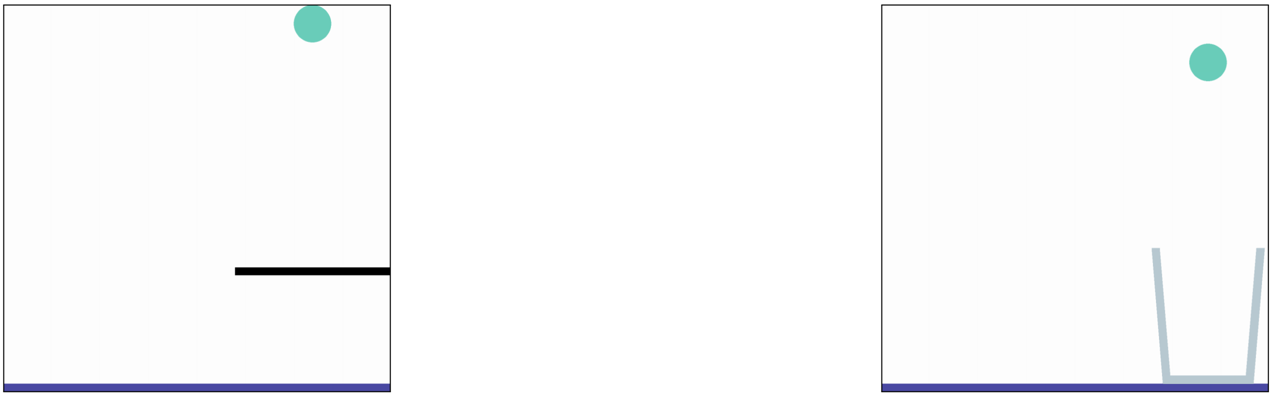
In stark contrast, templates 2 and 5 display a remarkably uniform structure that appears to



(a) Example template 00012 (b) Example template 00013 (c) Example template 00020

Figure 5.11: Examples of Selected Failures. In templates 12, 13, and 20, the generated proposals often exhibit substantial overlapping. This is likely due to the need for precise proposal placement near adjacent objects or within densely scattered scenes.

facilitate more successful outcomes, albeit sparingly (see Figure 5.12). These puzzles share a common layout: there are no angular configurations; the scene contains only a single additional object aside from the green ball; the green ball is positioned at a significant height; and the task requires the proposal to be placed beneath the green ball, with the vertical positioning being relatively flexible. The simplicity of these conditions effectively reduces the spatial constraints of the task, making the determination of a correct proposal more straightforward.



(a) Example template 00002 (b) Example template 00005

Figure 5.12: Structured Simplicity in Templates 2 and 5. The absence of angles and the presence of only one extra object, combined with a high starting position for the green ball, create an uncomplicated scenario where the correct placement is generally just below the green ball.

A closer inspection of the easier templates (2 and 5) reveals that their lower complexity might lend itself to occasional correct outputs by the VLMs. However, given the overall low success rate observed across the dataset, it remains unclear whether these successes are indicative of a genuine internalization of physical reasoning principles or simply fortunate instances due to the minimal spatial demands. In other words, while the structured simplicity of these puzzles could theoretically enable the models to accomplish them, the sparse number of correct responses makes it difficult to confidently attribute these successes to learned capability rather than random chance.

The stark differences between the failure-prone templates and the simpler ones underscore the challenges inherent in physical reasoning tasks. When confronted with scenarios requiring high precision—such as those with a myriad of objects or where the correct insertion point is in near contact with other elements (as seen in template 20)—the VLMs tend to default to generating overlapping proposals. This behavior suggests a reliance on heuristics that are insufficiently refined for handling complex spatial relationships.

Moreover, the occurrence of overlapping proposals in densely populated scenes (templates 12 and 13) further highlights the limitations of current training paradigms. The models appear to prioritize visual pattern recognition over precise spatial alignment, a trade-off that becomes particularly detrimental in complex scenes. This observation signals the need for future work to explore training methods that integrate explicit spatial or physical reasoning components, potentially by combining neural processing with rule-based spatial simulations.

On the other hand, the successes observed in templates 2 and 5 prompt an intriguing question: do these correct proposals reflect an embryonic capacity for genuine physical reasoning in simplified contexts, or are they merely the product of serendipitous alignment with an oversimplified task structure? Although the lack of complexity in these puzzles reduces ambiguity, the low frequency of success prevents a definitive conclusion. A more extensive set of experiments, especially focusing on puzzles with similar low complexity, would be required to determine whether these achievements represent an early form of learned spatial reasoning.

In summary, the qualitative observations presented here serve as a microcosm of the broader challenges faced by current Vision Language Models. The significant divergence in performance between complex and simple templates illustrates not only the limitations of existing VLM architectures in handling detailed spatial reasoning tasks but also the potential pathways for future improvements. Enhancing the models' abilities to accurately capture and process intricate spatial relationships is a critical step toward achieving more robust and reliable physical reasoning. Future research must address these limitations by exploring hybrid approaches that couple data-driven learning with explicit physical modeling, thereby paving the way for VLMs that can more effectively navigate both simple and complex spatial scenarios.

Overall, these insights underline an important conclusion: while sporadic successes in simpler puzzles are promising, they do not yet constitute evidence of a comprehensive understanding of physical reasoning within these models. The path forward involves not only optimizing architectures and training protocols to eliminate overlapping errors in densely populated scenes but also developing more nuanced evaluation strategies to discern between genuine reasoning ability and fortuitous successes in streamlined scenarios.

5.3 Error Analysis

In this section, we undertake a detailed examination of the factors contributing to the models' errors in physical reasoning. One prominent source of confusion arises from the extended collision dynamics required for goal attainment. Models often struggle to differentiate between scenarios that demand a prolonged sequence of collisions and those that involve a simple, direct collision. The simulation timeline, which projects approximately 25 seconds into the future, further exacerbates this challenge by introducing cumulative uncertainties that hinder precise prediction of physical interactions.

Another significant challenge originates from the complexity inherent in scenes populated with numerous objects. As the number of elements increases, so does the number of potential interactions, making it substantially more difficult for models to account for all physical relationships accurately. The high resolution of the puzzles adds another layer of complexity; the fine-grained spatial details necessary for accurate positioning can easily lead to misinterpretation. Compounding these issues, the descriptions of simulation parameters are often insufficient, leaving critical aspects of the physical environment under-specified. Although the prompt does not list every specific parameter (such as body sizes, positions, shapes, and types) explicitly, it nevertheless conveys all essential information required to understand the physical scene. The simulation description, in tandem with the scene image, unambiguously communicates the number, positions, shapes, dynamic properties, and target designations of all bodies. Therefore, the prompt is not under-specified; any challenges in interpreting the physical interactions should be attributed to the inherent complexity of the task and limitations of the models' reasoning capabilities, rather than a lack of provided data. It should be noted that while assessments of prompt completeness can vary and influence evaluation outcomes, all main parameters necessary for robust analysis are indeed present, either explicitly or implicitly.

An analysis based on distance-dependent difficulty categories reveals that models tend to exhibit higher confidence and accuracy when dealing with simpler scenarios, where the spatial relationships are less intricate. In contrast, as the complexity of the spatial arrangement increases—evidenced by longer interaction distances and more convoluted configurations—the reliability of the models' predictions deteriorates noticeably.

Furthermore, a recurring pattern in the models' responses indicates prevalent misconceptions in their approach to physical reasoning. Specifically, models such as GPT-4o, Qwen, and Gemini often operate under the assumption that if a puzzle appears overly challenging, the proposed solution is unlikely to achieve the intended goal. This predisposition leads to a narrow range of responses that may overlook feasible solutions. In contrast, models like Claude adopt a more exploratory approach, considering a wider array of potential outcomes, which may contribute to more diverse and sometimes more accurate predictions.

In summary, the errors observed can be attributed to several intertwined factors: (1) the difficulty of managing extended collision dynamics and long simulation timelines, (2) the increased complexity arising from high-resolution, multi-object scenes, (3) the inadequacy of simulation parameter descriptions leading to ambiguous problem settings, and (4) inherent reasoning biases that predispose models to discount the possibility of success in complex scenarios. These insights underscore the multifaceted challenges faced by current vision language models in achieving robust physical reasoning.

5.4 Limitations of the Current Evaluation

A recurring theme across all the experiments is that, despite the apparent diversity of tasks (from one-shot binary judgments to multi-step interactive insertions), the evaluations still capture only a narrow slice of what "robust physical reasoning" could entail. Although the PhysIQ framework draws from a broad range of puzzle templates inspired by PHYRE, it ultimately remains limited to relatively simple two-dimensional scenes with basic shapes and constrained physics. The strict focus on having two target objects make contact for three seconds—while useful for consistency—overlooks more complex interactions such as tool usage, multi-phase collisions, detailed occlusions, or fluid and deformable-body behaviors. That narrow scope means that success in these tasks should not be interpreted as conclusive evidence of a model's deeper or more generalizable understanding of physical laws.

Further issues arise from the strong reliance on visual cues and minimal textual context. Models receive an initial snapshot or a short set of frames, but they do not receive rich, continuous streams of data describing frictional forces, precise mass differentials, or even partial real-time updates about collisions. In principle, providing more explicit physics parameters or a more detailed simulation trace could have helped the models build sturdier internal representations, but it might also have overwhelmed them with extraneous details—especially given current limits on token context length and the generally fragile nature of large language model prompts. Because the puzzle tasks hinge on friction, elasticity, and density values that are only described textually, the models are forced to guess or approximate the relevant dynamics. Hence, although each puzzle is grounded in a well-defined simulation, the model's perspective on that simulation remains incomplete and can lead to errors that would not arise if the system had fuller information on the underlying physics parameters.

Another source of limitation stems from the size and structure of the dataset used in each evaluation tier. While the final compilation of correct and incorrect proposals offers a decent variety of success and failure cases, the absolute number of tasks still pales in comparison to what might be needed for robust generalization, especially in the hardest or "least typical" puzzle configurations. For instance, certain templates with heavily cluttered scenes (e.g., requiring extremely precise placements to avoid overlap) yielded disproportionate rates of failure. This indicates that we have not fully balanced the puzzle distribution to account for subtle object-placing challenges. When scenes become especially dense or angular, the margin for error shrinks substantially, and the random proposals or incremental adjustments (in the iterative tasks) fail at high rates. Conversely, some simpler puzzles (like those featuring only one static obstacle and a target object in a straightforward layout) may artificially inflate a model's sense of confidence, since even naive "close to the target" placements can sometimes succeed. The interplay between puzzle complexity and model accuracy remains underexplored in our methodology.

The interactive evaluation, meanwhile, suffers from the small scale of attempts (a maximum of five tries per puzzle) and the limited sample size of templates. Although multi-step refinements are precisely how one might expect an agent to learn from real-time feedback, we observed that very few proposals actually made significant progress from attempt to attempt. The fundamental reason is that adjusting a ball's center by a handful of pixels or changing its radius by a small margin can hardly compensate for a major conceptual misunderstanding. By capping the iteration count and not systematically modeling the step-by-step chain of thought, we have only scratched the surface of how a system might adapt across repeated trials. It is unclear whether allowing, say, ten or fifteen attempts would have allowed either model to converge on correct solutions in more templates. Furthermore, the puzzle success criterion (a single ”Yes/No” at the end of each simulation attempt) does not reveal whether the model learned intermediate physical insights; it only reveals a final pass/fail outcome. Consequently, the interactive setup does not disentangle partial improvements in strategy from pure guesswork.

A key limitation in interpreting the interactive evaluation results is that each puzzle was attempted only a handful of times. Because the models often failed multiple attempts in a row, we could not definitively characterize if or when the model might eventually discover a correct placement. Knowing the *attempt index* at which each puzzle was first solved would have provided far greater resolution in analyzing how quickly the model grasped the physical constraints or adapted its strategy. Such extended trial-based data collection, however, would have required a substantially larger computational budget (and hence, higher costs) than allotted. This constraint should be kept in mind when generalizing from the current findings to a broader notion of model adaptability.

Another constraint in our study is that multiple puzzles exhibit more than one valid solution—any of several position-radius configurations can lead to a successful outcome. Consequently, we could not define a single ”error distance” between the model's chosen proposal and *the* correct proposal. In principle, an analysis of all ”correct solution regions” (i.e., continuous areas of valid positions and radii) could have allowed us to calculate how far each proposed placement was from these viable zones. That, however, would require extensive sampling of the proposal space and fine-grained simulation of each candidate, imposing a steep computational overhead. Future research might explore embedding-based or region-based metrics to estimate a more nuanced ”error measure” for each puzzle proposal, thereby capturing partial correctness and clarifying how close a model's attempt is to a workable solution.

In addition, while the benchmark's final objective consistently involves making two specific objects touch for three seconds, the puzzles themselves are intricately designed with embedded complexities such as balancing, multi-object collisions, and nuanced dynamic interactions. These underlying physical phenomena require a comprehensive understanding of stability, collision mechanics, and spatial dynamics, meaning that success in these tasks depends on a broad spectrum of physical reasoning rather than mere contact detection.

Finally, the manner in which we prompt the models also shapes the limitations of these evaluations. Attempts to supply additional frames or to give more examples did not always translate to better performance, which suggests that the mere presence of extra data is no guarantee of more sophisticated reasoning. It is possible that different prompt designs—perhaps with structured descriptions of object states, or guided question sequences—might elicit better explanations or more strategic proposals. The unpredictable interplay between prompt design and model performance remains an open question. Some of the negative or positive biases we observed in different models appear to reflect how the examples are shown or the instructions are phrased, rather than purely the underlying physical reasoning ability.

Altogether, these limitations imply that while the PhysIQ evaluations demonstrate progress in capturing some facets of puzzle-based physical reasoning, one must not overstate the breadth or depth of what these tasks measure. They cannot, in their current form, disentangle genuine causal modeling from either memorized or superficial heuristics, nor do they exhaustively test the wide variety of phenomena that would constitute ”human-level” intuitive physics. Addressing these concerns—in particular by expanding puzzle diversity, refining the interactive feedback loop, and systematically exploring prompt designs—would yield more conclusive insights into whether current Vision-Language Models can truly internalize and apply the laws of physical reality.

6 Discussion

This chapter presents a broader reflection on the significance and implications of our findings within the context of physical reasoning and, more specifically, the emergent capabilities and limitations of contemporary Vision-Language Models (VLMs). We begin by interpreting the quantitative and qualitative results from the preceding chapters and exploring whether current models are genuinely grasping the underlying physics. We then compare non-interactive versus interactive evaluations, highlighting the strategic and practical insights gained from each approach. Subsequently, we address core limitations of present-day VLMs for physical reasoning, particularly in puzzle-based environments. We conclude with an assessment of the benchmark's effectiveness and propose several directions for future research and benchmarking efforts.

6.1 Interpretation of Key Findings

6.1.1 Revisiting the Role of Physical Reasoning in VLMs

The evaluations presented in Chapter 5 indicate that large-scale multimodal models indeed possess some capacity to parse visual scenes and discriminate outcomes that align with simplistic Newtonian mechanics. However, across nearly all tasks—from basic sanity checks to advanced puzzle-solving—a notable gap remains between human-level intuitive physics and VLM performance.

On the one hand, the top-tier models (GPT-4o, Claude-3.5-Sonnet, Gemini-2.0-Flash, and Qwen2.5) show promise in identifying clear-cut, ”easy” scenarios. Their performance in binary classification (Section 5.1.3) improves substantially with carefully designed few-shot prompts and minimal example-based guidance. This result underscores how large pretrained models can internalize enough statistical patterns from broad text and image data to handle moderately intricate questions about collisions and object contact.

On the other hand, the significant performance drops on tasks that demand nuanced, comparative judgments (Section 5.1.4) or iterative problem-solving (Section 5.1.5) confirm that existing models still rely heavily on memorized correlations and shallow heuristics rather than deeply grounded simulation. Their difficulty in ranking multiple proposals or adapting ball placements over repeated attempts suggests an inability to maintain structured, causal representations of dynamic events.

6.1.2 Pattern Recognition vs. Genuine Understanding

The pervasive question throughout this thesis has been whether current VLMs truly ”understand” physical dynamics or merely exploit patterns learned from massive datasets. Our results strongly suggest the latter explanation is more accurate. For instance, the stark default biases (negative or positive) observed in zero-shot binary classification (Figure 5.3) betray a lack of robust internal simulation. Without example prompts, models systematically guess that *every* proposal is doomed to fail or bound to succeed, revealing a reliance on prior dataset biases rather than any environment-specific logic.

When few-shot context or final-frame images are introduced, the models' performance improves. However, even these improvements often appear *localized and incremental.* They reflect a short-range adaptation to demonstration examples rather than a cohesive mental model that can be reliably extended to new puzzle configurations. Moreover, the minimal advantage gained by providing the final simulation frames (as opposed to only the initial state) for most models indicates that additional visual context does not necessarily translate into more thorough or stable predictive reasoning. Arguably, these patterns again point to robust correlation matching rather than a coherent theory of forces, collisions, and friction.

6.1.3 Temporal Complexity and Failure Modes

In tasks where the physical process unfolds over a longer time horizon—notably, contact-based puzzles requiring sustained collisions and extended movement—the models exhibit greater confusion and produce ambiguous predictions. Many proposals fail not because they are unsound in principle, but rather because the model never pinpoints a correct enough location or radius to maintain contact for three seconds.

From a temporal perspective, the compounding of small errors in forward predictions is a well-known problem in physics-based simulation [30]. Yet in the context of VLMs, these errors appear more closely tied to a shortfall in learned parametric knowledge and an inability to track partial states across multiple time steps. In essence, the "mental simulation" required to forecast an extended chain of collisions remains out of reach for these models, which typically process only static glimpses or short sequences.

6.1.4 Insights into Object-Centric Reasoning

Chapters 2 and 3 discussed how object-centric inductive biases can significantly improve a system's capacity for robust physical reasoning. Our puzzle-based results strengthen this perspective. Situations with *fewer* objects in simpler, well-spaced configurations (e.g., template 00002) were solved occasionally or at least approached more reasonably than scenes requiring high-precision insertion within cluttered environments. This pattern highlights the possibility that advanced object-centric frameworks—such as those explored in Interaction Network approaches [10]—could be integrated into VLM pipelines to achieve more reliable forward predictions.

Overall, while the results confirm that modern VLMs have some superficial competency, they stop well short of the "human-like" mental simulation described in theories of intuitive physics [9]. Instead, they show that bridging the gap from pattern-based correlation to genuine causal understanding remains a formidable challenge for AI systems today.

6.2 Interactive vs. Non-Interactive Evaluation

A key methodological contribution of this thesis is our use of both static and iterative tasks to dissect the models' problem-solving strategies.

6.2.1 Static Evaluations: Efficiency and Limitations

Static tasks—binary classification, ranking, or single-pass confidence estimation—offer a convenient measure of immediate physical reasoning from a given snapshot. These tasks are computationally lightweight, easily parallelized, and facilitate detailed ablation (e.g., zero-shot vs. few-shot). Indeed, we saw how few-shot prompts can radically shift a model's classification bias, revealing the malleability of correlation-driven systems.

Yet static tasks fundamentally limit the depth of insights into how an agent might *improve* its reasoning with experiential feedback. A single forward pass does not capture "trial-and-error" or dynamic world exploration, both of which characterize real-world physical problem-solving [5]. Consequently, strong static performance can mask the absence of deeper causal logic. The ranking results are emblematic here: models show near-random behavior when forced to discriminate subtle differences among four proposals (one correct, three incorrect). The partial success in binary classification thus does not translate to success in more nuanced comparative tasks.

6.2.2 Interactive Evaluations: Potential for Adaptation

By contrast, interactive tasks (Section 5.1.5) allow repeated attempts, immediate feedback on success/failure, and time-distributed frames capturing the simulation's progression. In principle, this setup should encourage "online" or in-context learning, letting the model refine its approach with each new piece of evidence—akin to how humans or advanced RL agents might systematically adjust strategy.

Our results, however, reveal that the two top-performing models (GPT-4o and Claude-3.5-Sonnet) show only minimal success (roughly 5–7% success rates) in discovering correct proposals within five attempts. Crucially, neither model exhibits consistent improvement across attempts; success, when it occurs, appears haphazard. The repeated tendency to propose overlapping or out-of-bounds placements suggests the models lack a stable representation of the puzzle constraints.

6.2.3 Implications for Real-World Usage

Despite these lackluster results, interactive evaluations remain essential for diagnosing how existing systems handle iterative feedback. In scenarios such as robotics, user-guided puzzle solving, or tool manipulation, real-time responses are standard. A single-shot classification approach cannot approximate the complexity of repeated attempts with partial successes or near-misses along the way.

One notable takeaway is that standard LLM architectures are not trivially ready for "closed-loop" physical tasks, even with sophisticated prompting. The combination of limited short-term memory, shallow chaining of attempts, and no stable internal simulator hamper success. Future work may require either extended context windows, a specialized memory of prior attempts, or explicit symbolic expansions (e.g., a vector-based or graph-based simulator module) that the model queries after each attempt.

6.3 Limitations of Current Models for Physics Reasoning

6.3.1 Sensitivity to Spurious Correlations

As highlighted by the binary classification biases, one fundamental pitfall of large-scale data-driven models is their reliance on spurious patterns. When the input distribution does not strongly match their training set (for instance, puzzle scenes in 2D with simplified geometry), the model reverts to naive guesses. These include attributing *extremely low* probabilities to physically feasible solutions (negative bias) or granting nearly *unconditional optimism* (positive bias).

6.3.2 Lack of Long-Horizon Simulation

Even when the model infers that an object might move in a certain direction, it rarely accounts for multi-step collisions, friction-driven slowdowns, or multi-object chain reactions. Errors compound over frames, especially once the puzzle timeline extends beyond a few seconds. The ranking task (Section 5.1.4) underscores how the ability to compare multiple hypothetical futures is beyond what the model can handle via static embeddings or large token contexts alone.

6.3.3 Format Adherence vs. Spatial Reasoning

Interestingly, the interactive attempts confirm that even the best VLMs rarely produce format errors when asked to supply JSON coordinates. They excel at structured text generation, presumably due to robust training on code and JSON data. Yet simultaneously, they falter at the essential geometry: they choose proposals that lie outside the puzzle boundary or overlap existing bodies. This disconnect between textual coherence and geometric viability exemplifies the gap between general language capabilities and grounded physical reasoning.

6.3.4 Underdefined Prompts and Physics Parameters

A partial cause of failure is that the textual instructions, though explicit in listing friction or gravity constants, may not be integrated effectively into the model's "reasoning." Some clues suggest that if the puzzle environment is not extremely straightforward (e.g., stable direct drop, minimal clutter), the model's attempts remain random or incompetent. This might be addressed by more sophisticated prompts that systematically recast the puzzle in object-centric form. However, absent a robust internal simulator, additional text alone may not suffice.

6.4 Benchmark Effectiveness

6.4.1 Strengths of the PhysIQ Approach

By leveraging the PHYRE environment [8], we introduce a puzzle-based test with a well-defined success criterion (sustained contact), a wide variety of object configurations, and the potential for out-of-distribution generalization checks. Unlike purely observational tasks (e.g., labeling collisions in pre-generated videos), our approach also accommodates iterative attempts. This synergy helps pinpoint distinctions among model performance levels.

Additionally, the suite of tasks—from quick binary classification to full interactive insertion—

provides incremental levels of difficulty that expose subtle model behaviors and biases. Observing how a single prompt can drastically alter performance clarifies the delicate interplay between prompting design and physical reasoning success.

6.4.2 Limitations and Coverage Gaps

While PhysIQ marks a step forward, several limitations constrain its broader applicability. One such limitation is the single physical goal: we focus on making two target objects collide, and more diverse tasks (e.g., stability, supporting, multi-step chain reactions) remain unexplored. Another limitation is the 2D environment, as real-world tasks often involve 3D geometry, friction complexities, or fluid and soft-body interactions that are absent here. Additionally, the approach is confined by the single puzzle format; the uniform "ball insertion" approach, though straightforward, might not fully capture advanced tool use or composite strategies (e.g., chaining multiple collisions across extended time). Finally, the design includes limited interactive attempts, since we cap iterations at five, which is enough to reveal some adaptation deficits but might understate a model's potential if given more tries and memory capacity.

Nevertheless, these controlled constraints also serve as an advantage for systematically dissecting model responses and avoiding confusion from unstructured tasks. In practice, adopting complementary benchmarks—such as CLEVRER [31] (causal queries in short videos) or IntPhys [30] (violation-of-expectation tasks)—can further enrich the overall evaluation of intuitive physics.

6.5 Future Directions

6.5.1 Expanding Puzzle Diversity and Goals

An immediate extension is to incorporate puzzle templates demanding other forms of object interaction, such as balancing or hooking an object on top of another. A multi-goal environment (akin to [17]) would also stress the generality of a model's physical reasoning.

6.5.2 Combining Symbolic Simulators with VLMs

Given the challenges of purely data-driven LLM approaches, hybrid paradigms may hold promise: a model could invoke an internal or external physics simulator for small rollouts before returning a final textual response. This approach aligns with prior frameworks that treat the LLM as an orchestrator of a symbolic or differentiable simulation engine [29, 13].

6.5.3 Interactive Learning with More Attempts and Memory

Although the iterative evaluation shows that the model does not forget previous attempts since they remain in the context, limiting the evaluation to only five attempts may restrict the potential for continuous improvement. Increasing the number of allowed attempts could better support iterative refinement and learning. However, directly appending all past attempts into the context risks exceeding the limited context window. To overcome this, alternative memory mechanisms—such as embedding-based summarization or dedicated memory modules—should be employed to compactly store and retrieve past outcomes. This approach retains crucial historical insights while efficiently managing context, enabling a more robust strategy refinement over extended interactions.

6.5.4 Refined Prompting and Multi-Modal Representations

Future work could explore finer-grained prompts that break down each puzzle step, asking the model to list possible collisions or rank partial states. Another approach is to integrate textual, geometric, and event-based representations into a single architecture. By systematically highlighting relevant objects (positions, shapes, friction) in a structured, multi-modal prompt, we may better harness the advanced text generation capabilities of large models while mitigating their guesswork in geometry.

6.5.5 Comparisons with Human Data

Finally, bridging the gap between AI and human intuitive physics would benefit from parallel user studies. Observing how humans solve these same puzzle templates—especially under time or attempt constraints—would reveal which errors are "human-like" and which are purely model artifacts [26, 23]. Crowd-sourced or lab-based experiments can provide robust baselines to gauge progress.

In summary, while the PhysIQ benchmark brings valuable insights into the puzzle-based reasoning capabilities of state-of-the-art VLMs, our findings make it clear that these models still struggle to form robust, causal, and temporally extended physical simulations. By expanding puzzle formats, integrating symbolic sub-modules, enhancing interactive learning paradigms, and comparing with human baselines, future endeavors can move us closer to the goal of achieving a truly grounded ”intuitive physics” in AI systems.

7 Conclusion

In this thesis, we presented a new benchmark—named PhysIQ—to assess how well Vision-Language Models (VLMs) handle puzzle-based physical reasoning. By adapting and expanding upon the original PHYRE puzzle environment, we established several key contributions. First, we present a curated set of puzzle templates that emphasize varied degrees of difficulty and object complexity, thereby offering a diverse range of physical reasoning challenges. Next, we developed a simulation pipeline that faithfully replicates PHYRE's physics while incorporating robust checks to ensure puzzle integrity. We further constructed a proposal-based dataset comprising both correct and incorrect solutions, which enables systematic analysis of each model's ability to classify or rank proposals. Additionally, we implemented multiple evaluation modes—from simple perceptual sanity checks and confidence estimation to static one-shot or few-shot classifications and multi-turn interactive insertion tasks—that collectively probe the strengths, biases, and limitations of state-of-the-art vision-language models. Finally, we introduce an interactive web-based prompt tester page that facilitates dynamic inspection and iterative refinement of prompt configurations.

In doing so, we not only preserved the classical contact-based objective (sustained collision between target objects) but also introduced iterative attempts, thus testing a model's capacity to integrate feedback and refine solutions dynamically. By examining different few-shot prompting conditions and interactive loops, we illustrated how the same puzzle can reveal different strengths, biases, and blind spots in large-scale vision-language models.

7.1 Key Takeaways

Our comprehensive evaluation revealed significant insights into the current state of physical reasoning capabilities in Vision Language Models. Even fundamental physical perception tasks proved challenging, with only a select few models——GPT-4o, Claude-3.5-sonnet, Gemini-2.0-flash-001, and Qwen2.5-vl-72b-instruct——demonstrating reliable performance in basic contact detection. This finding underscores the non-trivial nature of even elementary physical perception tasks for current VLMs. More concerning was the discovery of strong default biases in zero-shot binary judgments, where models systematically classified nearly all proposals as incorrect, revealing a fundamental lack of robust internal simulation capabilities rather than genuine physical reasoning.

The introduction of few-shot examples produced divergent outcomes across models. While Claude-3.5-sonnet and GPT-4o showed substantial improvements in discriminating between correct and incorrect proposals when provided with contextual examples, other models like Qwen2.5-vl-72b-instruct maintained persistent biases regardless of additional context. This pattern suggests that some models possess latent physical reasoning capabilities that can be activated through appropriate prompting, while others rely more heavily on fixed heuristics that resist contextual refinement. Particularly noteworthy was the finding that providing final-state frames generally did not improve performance for most models, with Gemini-2.0-flash-001 being the notable exception that benefited from seeing outcome states.

Comparative reasoning tasks revealed perhaps the most significant limitations in current VLMs. The ranking evaluation demonstrated consistently poor performance (4-13% accuracy) across all models and few-shot conditions, with models distributing their selections almost randomly among possible positions. This finding indicates that fine-grained discrimination between multiple physical scenarios remains beyond the capabilities of current systems, even when they perform adequately on simpler binary tasks. The interactive evaluation further highlighted these limitations, with both GPT-4o and Claude-3.5-sonnet achieving success rates of only 5-7% when given multiple attempts to solve physical puzzles. Critically, neither model showed systematic improvement across successive attempts, making only small, incremental adjustments between trials rather than demonstrating strategic refinement based on feedback.

The qualitative analysis revealed a stark contrast between performance on structurally simple puzzles versus those requiring precise spatial reasoning. Models occasionally succeeded in templates with minimal objects and straightforward configurations but consistently failed when confronted with densely populated scenes or scenarios requiring high-precision placements. This pattern suggests that current VLMs rely primarily on pattern recognition rather than causal physical understanding, succeeding only when the spatial constraints are sufficiently relaxed to accommodate imprecise reasoning. Collectively, these findings highlight the persistent gap between human-like physical intuition and the more correlation-driven approaches of current VLMs, underscoring the need for architectural innovations that can better integrate causal modeling with visual perception.

7.2 Closing Remarks

Overall, the PhysIQ benchmark captures diverse puzzle configurations that stress both *static* and *interactive* reasoning. Our results highlight the persistent gap between human-like, iterative physical intuition and the more correlation-driven approaches of leading VLMs. Although certain models succeed at simpler tasks or exhibit partial improvement with carefully designed prompts, the failure patterns in more complex puzzles underscore the need for deeper causal modeling. Future expansions could incorporate broader object shapes, three-dimensional scenes, or multi-step causal chains, while also offering extended interactive attempts and hybrid symbolic-data-driven architectures. We hope PhysIQ facilitates new directions in physically grounded AI research and encourages the development of models capable of genuine, adaptive reasoning in dynamic environments.

Bibliography

[1] Claude 3.5 sonnet. https://www.anthropic.com/news/claude-3-5-sonnet. Last update: Jun 21, 2024.

[2] Gemini 2.0 is now available to everyone. https://blog.google/technology/google-deepmind/gemini-model-updates-february-2025/. Last update: Feb 05, 2025.

[3] Grok-2 beta release. https://x.ai/blog/grok-2. Last update: Aug 13, 2024.

[4] Pixtral large. https://mistral.ai/news/pixtral-large. Last update: Nov 18, 2024.

[5] Kelsey R. Allen, Kevin A. Smith, and Joshua B. Tenenbaum. Rapid trial-and-error learning with simulation supports flexible tool use and physical reasoning. *Proceedings of the National Academy of Sciences*, 117(47):29302–29310, 2020.

[6] Tayfun Ates, M. Samil Atesoglu, Cagatay Yigit, Ilker Kesen, Mert Kobas, Erkut Erdem, Aykut Erdem, Tilbe Goksun, and Deniz Yuret. Craft: A benchmark for causal reasoning about forces and interactions, 2022.

[7] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025.

[8] Anton Bakhtin, Laurens van der Maaten, Justin Johnson, Laura Gustafson, and Ross Girshick. Phyre: A new benchmark for physical reasoning, 2019.

[9] Peter W. Battaglia, Jessica B. Hamrick, and Joshua B. Tenenbaum. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences*, 110(45):18327–18332, 2013.

[10] Peter W. Battaglia, Razvan Pascanu, Matthew Lai, Danilo Rezende, and Koray Kavukcuoglu. Interaction networks for learning about objects, relations and physics, 2016.

[11] Anne Beyer, Kranti Chalamalasetti, Sherzod Hakimov, Brielen Madureira, Philipp Sadler, and David Schlangen. clembench-2024: A challenging, dynamic, complementary, multilingual benchmark and underlying flexible framework for llms as multi-action agents, 2024.

[12] Meng Cao, Haoran Tang, Haoze Zhao, Hangyu Guo, Jiaheng Liu, Ge Zhang, Ruyang Liu, Qiang Sun, Ian Reid, and Xiaodan Liang. Physgame: Uncovering physical commonsense violations in gameplay videos, 2024.

[13] Thomas Carta, Clément Romac, Thomas Wolf, Sylvain Lamprier, Olivier Sigaud, and Pierre-Yves Oudeyer. Grounding large language models in interactive environments with online reinforcement learning, 2024.

[14] Zhenfang Chen, Kexin Yi, Yunzhu Li, Mingyu Ding, Antonio Torralba, Joshua B Tenenbaum, and Chuang Gan. Comphy: Compositional physical reasoning of objects and events from videos. In *International Conference on Learning Representations*, 2022.

[15] Ching-An Cheng, Andrey Kolobov, Dipendra Misra, Allen Nie, and Adith Swaminathan. Llf-bench: Benchmark for interactive learning from language feedback, 2023.

[16] Jiafei Duan, Samson Yu Bai Jian, and Cheston Tan. Space: A simulator for physical interactions and causal learning in 3d environments, 2021.

[17] Chathura Gamage, Vimukthini Pinto, Cheng Xue, Peng Zhang, Ekaterina Nikonova, Matthew Stephenson, and Jochen Renz. Novphy: A testbed for physical reasoning in open-world environments, 2023.

[18] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal,

Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024.

[19] Oliver Groth, Fabian B. Fuchs, Ingmar Posner, and Andrea Vedaldi. Shapestacks: Learning vision-based physical intuition for generalised object stacking, 2018.

[20] Aum Kendapadi, Kerem Zaman, Rakesh R. Menon, and Shashank Srivastava. Interact: Enabling

interactive, question-driven learning in large language models, 2024.

[21] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks, 2019.

[22] Ziqiao Ma, Zekun Wang, and Joyce Chai. Babysit a language model from scratch: Interactive language learning by trials and demonstrations, 2024.

[23] M Keith Moore and Andrew N Meltzoff. New findings on object permanence: A developmental difference between two types of occlusion. *British Journal of Developmental Psychology*, 17(4):623–644, nov 1999.

[24] OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Madry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao

Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. Gpt-4o system card, 2024.

[25] OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Quiñonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singhal, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov,

Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. Openai o1 system card, 2024.

[26] Jasmin Perez and Lisa Feigenson. Stable individual differences in infants' responses to violations of intuitive physics. *Proceedings of the National Academy of Sciences*, 118(27):e2103805118, 2021.

[27] Luis S. Piloto, Ari Weinstein, Peter Battaglia, and Matthew Botvinick. Intuitive physics learning in a deep-learning model inspired by developmental psychology. *Nature Human Behaviour*, 6(9):1257–1267, 2022. Published on 2022/09/01.

[28] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[29] Nazneen Fatema Rajani, Rui Zhang, Yi Chern Tan, Stephan Zheng, Jeremy Weiss, Aadit Vyas, Abhijit Gupta, Caiming XIong, Richard Socher, and Dragomir Radev. Esprit: Explaining solutions to physical reasoning tasks, 2020.

[30] Ronan Riochet, Mario Ynocente Castro, Mathieu Bernard, Adam Lerer, Rob Fergus, Véronique Izard, and Emmanuel Dupoux. Intphys: A framework and benchmark for visual intuitive physics reasoning, 2020.

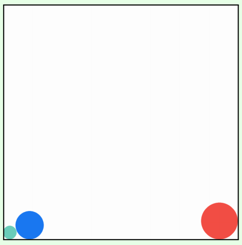
[31] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. Clevrer: Collision events for video representation and reasoning, 2020.

Appendix A Sanity Check Prompt

System message

You are a physics expert analyzing object interactions in physics simulations.
Task: You will receive an image depicting the current state of a simulation. Determine if the specified objects are currently in contact.
Response format: Answer clearly and exclusively with ”Yes” or ”No”.

User message



Based on the current state, are the green ball and azure ball in contact?

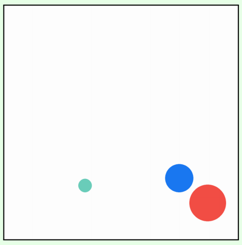
Appendix B Confidence Prompt

System message

You are an expert evaluator tasked exclusively with providing concise numerical probability estimates for outcomes of physics simulations.
Simulation conditions for all tasks:
- Gravity: 9.81 m/s^2 (downward)
- Objects density: 0.25
- Friction coefficient: 0.5
- Elasticity coefficient: 0.2
- Simulation duration: 25 seconds or until objects stop moving
- Black and purple objects: static (fixed)
- Green, red, grey objects: dynamic (can move)
- Goal criterion: Objects must remain in contact $\geq$ 3 seconds
- Objects cannot leave the visible simulation boundaries

Task: Given an initial image, estimate the probability that the specified objective is successfully achieved.
Response format: Provide ONLY a numerical percentage (e.g., "85%"). Do NOT include explanations, reasoning, or any other additional text.

User message



What is the probability that the green ball and azure ball will come into contact for 3 seconds?
Answer:

Appendix C Ranking Prompt

System message

You are a physics expert analyzing physics simulations.
Simulation conditions for all tasks:
- Gravity: 9.81 m/s² (downward)
- Objects density: 0.25
- Friction coefficient: 0.5
- Elasticity coefficient: 0.2
- Simulation duration: 25 seconds or until objects stop moving
- Black and purple objects: static (fixed)
- Green, red, grey objects: dynamic (can move)
- Goal criterion: Objects must remain in contact ≥ 3 seconds
- Objects cannot leave the visible simulation boundaries

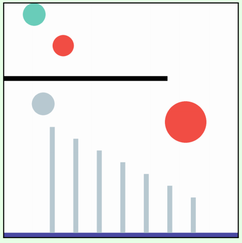
Task: You will receive 4 images, each depicting a different proposal for solving the puzzle. Rank the proposals clearly based on their likelihood of satisfying the goal, from highest likelihood (first) to lowest likelihood (last).

Response format: List ONLY the proposal indices in order from highest to lowest likelihood (e.g., `[3, 1, 4, 2]`). Do NOT include explanations, reasoning, or any other additional text.

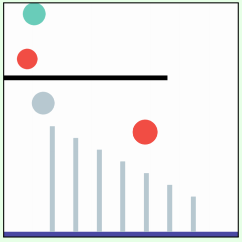
— Optional Few-shot Examples —

User message

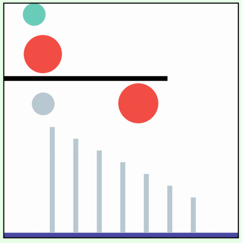
Example 1:
Proposal 1:



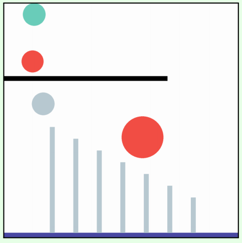
Proposal 2:



Proposal 3:



Proposal 4:



Will the green ball and purple bar come into contact for 3 seconds?
Rank the proposals by their likelihood of success:

Assistant message

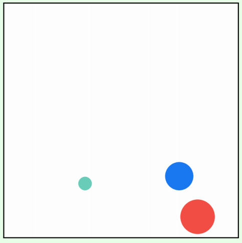
$[4, 2, 3, 1]$

...

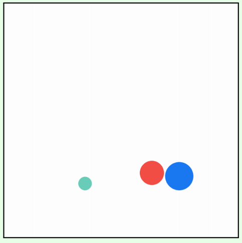
— Optional Few-shot Examples —

User message

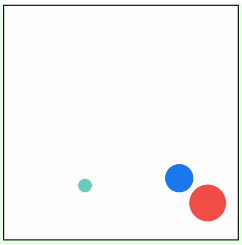
Proposal 1:



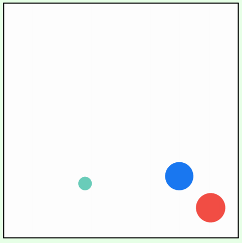
Proposal 2:



Proposal 3:



Proposal 4:



Will the green ball and azure ball come into contact for 3 seconds?
Rank the proposals by their likelihood of success:

Appendix D Binary Prompt

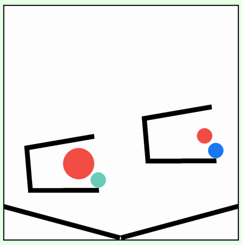
System message

You are a physics expert analyzing object interactions in physics simulations.
Simulation conditions for all tasks:
- Gravity: 9.81 m/s^2 (downward)
- Objects density: 0.25
- Friction coefficient: 0.5
- Elasticity coefficient: 0.2
- Simulation duration: 25 seconds or until objects stop moving
- Black and purple objects: static (fixed)
- Green, red, grey objects: dynamic (can move)
- Goal criterion: Objects must remain in contact $\geq$ 3 seconds
- Objects cannot leave the visible simulation boundaries

Task: Given an initial image of a physics simulation, predict whether the specified objects will come into contact for the required duration.
Response format: Answer clearly and exclusively with ”Yes” or ”No”.

— Optional Few-shot Examples - Only initial frame —

User message



Will the green ball and azure ball come into contact for 3 seconds?

Assistant message

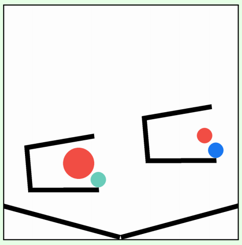
Yes

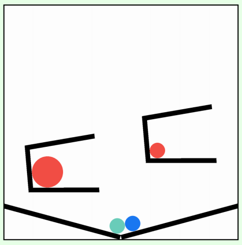
...

— Optional Few-shot Examples - Only initial frame —

— Optional Few-shot Examples - Initial and final frames —

User message





Will the green ball and azure ball come into contact for 3 seconds?

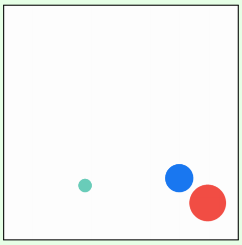
Assistant message

Yes

...

— Optional Few-shot Examples - Initial and final frames —

User message



Will the green ball and azure ball come into contact for 3 seconds?

Appendix E Interactive Prompt

System message

You are a physics expert creating solutions for physics simulations.
Simulation conditions for all tasks:
- Gravity: 9.81 m/s^2 (downward)
- Objects density: 0.25
- Friction coefficient: 0.5
- Elasticity coefficient: 0.2
- Simulation duration: 25 seconds or until objects stop moving
- Black and purple objects: static (fixed)
- Green, red, grey objects: dynamic (can move)
- Goal criterion: Objects must remain in contact $\geq$ 3 seconds
- Objects cannot leave the visible simulation boundaries

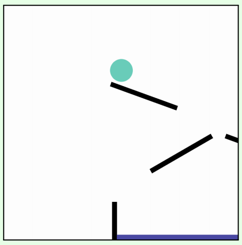
Task: Your objective is to place a new ball inside the simulation area to achieve the specified goal. Clearly define your solution by specifying:
- ”x”: horizontal position of the ball center (0 is left, maximum is 256 on the right)
- ”y”: vertical position of the ball center (0 is bottom, maximum is 256 at the top)
- ”radius”: size of the ball (minimum 2, maximum 32)

Important rules for placing the ball:
- The ball must remain fully within the visible simulation boundaries.
- The ball cannot overlap with existing objects.

Response format (always follow exactly): Provide your solution strictly in the following JSON format:

```
“‘json
{
    ”x”: ”<x-coordinate>”,
    ”y”: ”<y-coordinate>”,
    ”radius”: ”<ball_radius>”
}
“‘
```

User message



Try 1:

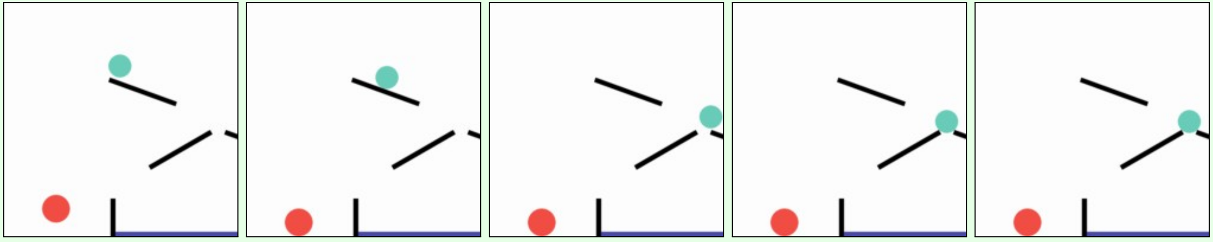
— Goal not reached —

Assistant message

```
“‘json
{
    ”x”: ”60”,
    ”y”: ”30”,
    ”radius”: ”15”
}
“‘
```

User message

The simulation ran successfully, but your proposal didn’t achieve the goal.
Review the following 5 simulation frames, each sampled at regular intervals, that illustrate key stages in the evolution of your previous proposal.



Carefully analyze the frames and try again. Provide a new proposal strictly in the same JSON format.
Try 2:

...

— Goal not reached —

— Outside boundaries —

Assistant message

```
“‘json
{
    ”x”: ”80”,
    ”y”: ”300”,
    ”radius”: ”10”
}
```

“‘

User message

Your previous proposal couldn’t be applied.
Reason: Your ball exceeds the simulation boundaries.
Try again, ensuring he ball fits entirely within the boundaries ($0 \leq x \leq 256$, $0 \leq y \leq 256$). Provide a new proposal strictly in the same JSON format.
Try 2:

...

— Outside boundaries —

— Overlapping —

Assistant message

```
“‘json
{
    ”x”: ”100”,
    ”y”: ”200”,
    ”radius”: ”32”
}
“‘
```

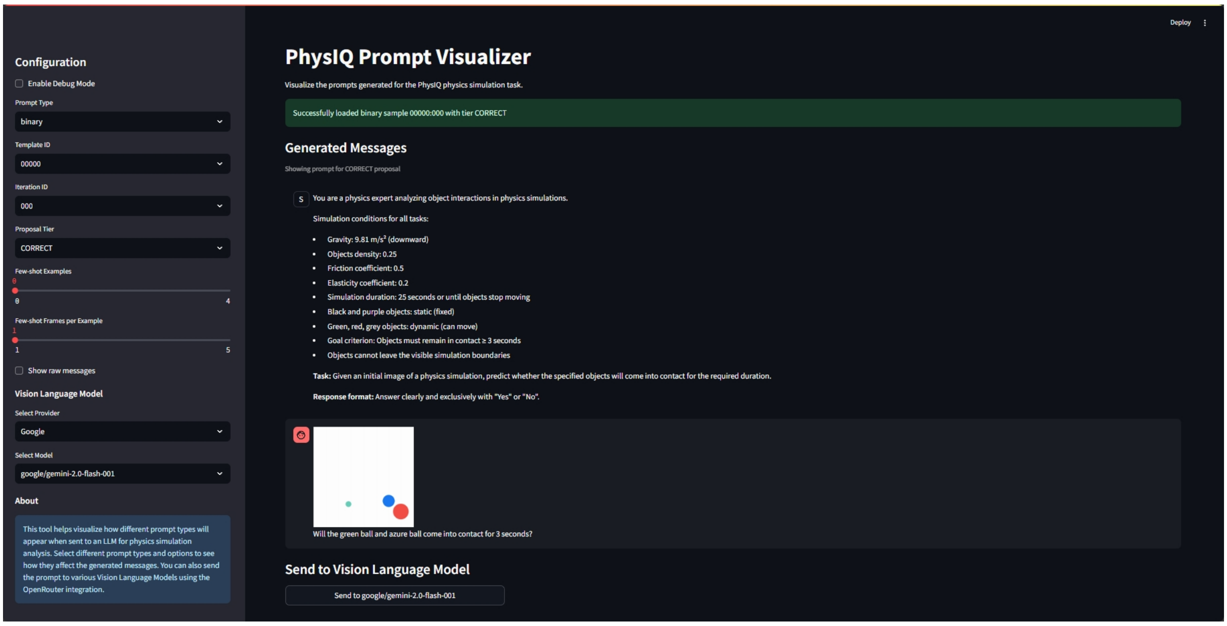
User message

Your previous proposal couldn’t be applied.
Reason: Your ball overlaps existing objects.
Try again, ensuring the ball does not overlap with any other object. Provide a new proposal strictly in the same JSON format.
Try 2:

...

— Overlapping —

Appendix F Prompt tester webpage



Configuration

□ Enable Debug Mode

Prompt Type: binary

Template ID: 00000

Iteration ID: 000

Proposal Tier: CORRECT

Few-shot Examples: 0 (0–4)

Few-shot Frames per Example: 1 (1–5)

□ Show raw messages

Vision Language Model

Select Provider: Google

Select Model: google/gemini-2.0-flash-001

About

This tool helps visualize how different prompt types will appear when sent to an LLM for physics simulation analysis. Select different prompt types and options to see how they affect the generated messages. You can also send the prompt to various Vision Language Models using the OpenRouter integration.

PhysIQ Prompt Visualizer

Visualize the prompts generated for the PhysIQ physics simulation task.

Successfully loaded binary sample 00000:000 with tier CORRECT

Generated Messages

Showing prompt for CORRECT proposal

S You are a physics expert analyzing object interactions in physics simulations.

Simulation conditions for all tasks:

- Gravity: 9.81 m/s² (downward)
- Objects density: 0.25
- Friction coefficient: 0.5
- Elasticity coefficient: 0.2
- Simulation duration: 25 seconds or until objects stop moving
- Black and purple objects: static (fixed)
- Green, red, grey objects: dynamic (can move)
- Goal criterion: Objects must remain in contact ≥ 3 seconds
- Objects cannot leave the visible simulation boundaries

Task: Given an initial image of a physics simulation, predict whether the specified objects will come into contact for the required duration.

Response format: Answer clearly and exclusively with "Yes" or "No".

Will the green ball and azure ball come into contact for 3 seconds?

Send to Vision Language Model

Figure F.1: Screenshot of the local web-based tool for prompt visualization and interactive testing.