Integrating Reasoning and Learning for Design Generation & Scientific Discovery
Application Track

Maxwell J. Jacobson, Nan Jiang, Md Nasim, Yexiang Xue
Purdue University, Department of Computer Science
{jacobs57, jiang631, mnasim, yexiang}@purdue.edu

Abstract

Machine learning and automatic reasoning stand as two fundamental pillars of artificial intelligence (AI). Machine learning, particularly through neural network techniques, has made significant strides in learning from complex data. Automatic reasoning, exemplified by constraint programming, brings interpretability and robustness to the table. However, machine learning faces significant challenges, primarily its lack of formal guarantees, leading to failures in situations divergent from its training data. Addressing some of the most pressing problems of our time necessitates integrating machine learning with automatic reasoning. We illustrate the power of this integration in two domains: automated design through our SPRING system and scientific discovery via our CVGP system. Both systems significantly outperform state-of-the-art machine learning-only baselines, underlining the critical importance of this integrative approach in advancing AI.

1 Introduction

Automatic reasoning and machine learning are two fundamental pillars of artificial intelligence (AI) Machine learning approaches – especially neural networks – have spearheaded major developments in learning from diverse and unstructured data, discovering hidden or fuzzy patterns, and producing effective predictive and generative models. Automatic reasoning approaches like constraint programs have produced efficient and reliable algorithms that can provide formal guarantees, interpretability, and robustness.

Despite its astounding success, pure neural-based learning is limited in several ways. Neural-based learning is perceptive – it creates knowledge from patterns in data. This means that it is adaptable but does not give formal guarantees. Therefore, it will often fail in contexts that are unlike its training data. This is exemplified by modern neural-based text-to-image generators, which often fail to uphold instructions regarding many objects or spatial relationships between objects.

Consequently, for many of the greatest problems of our time, *integrating machine learning and automated reasoning is the most effective approach*, as the strengths of each approach complement each other. We demonstrate this integration in two application domains. Firstly, we look at interior design generation. Good designs need to meet industry standards and user needs, while capturing subtle aspects such as aesthetics and convenience. Learning to generate objects with neural networks is a standard approach for this. Neural networks excel in learning to generate visually pleasing and functional designs. However, they often falter in adhering to precise user specifications, particularly when faced with constraint combinations not previously encountered in training datasets. For example, in Figure 1, a toaster must be added to the left of the oven and below the sink (already in the image). Additionally, a microwave needs to be added to the right of the oven. In the

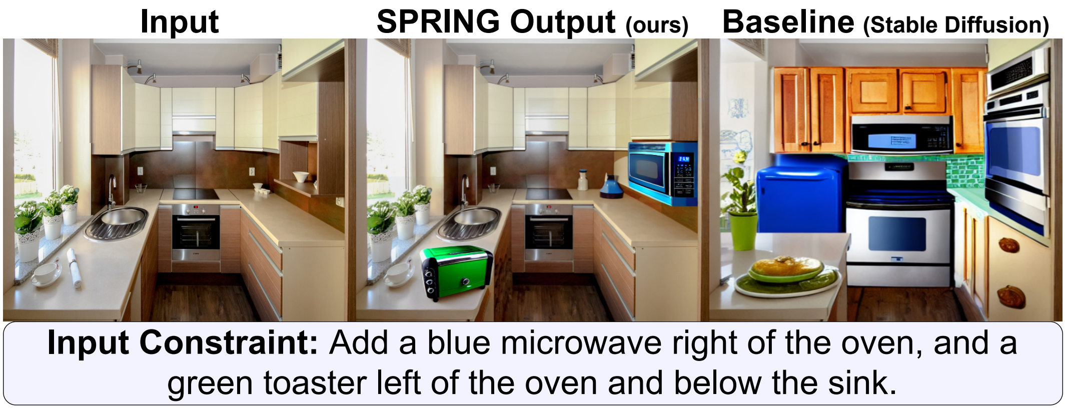

Figure 1: An interior design generated by our proposed SPRING model (middle) with a given background already containing an oven and a sink among other objects (left). The user specifications are at the bottom (provided to SPRING in the form of propositional logic; natural language text is used here to aid readability). SPRING creates a design satisfying the specifications. Text-to-image approaches like Stable Diffusion (right) often fail to meet these constraints.

right panel, the Stable Diffusion Rombach et al. [2022] model (one of the state-of-the-art neural generative models), taking the input of the initial kitchen configuration and the text specifications, simply alters the entire scene, producing results that look pleasing but do not fit the specification. This is emblematic of a deep issue with purely neural algorithms – thus far, they have failed to grasp the high-level symbolic understanding that automatic reasoning approaches can handle efficiently. We approach this problem by embedding a constraint reasoning solver within a neural architecture, allowing the system to iteratively reach a solution that is both explicitly constrained and implicitly natural.

In the domain of automatic scientific discovery, we also find that reasoning can correct for the innate limitations of learning. Specifically, symbolic regression, which involves learning the governing expression from experimental data, plays a crucial role in this automation. We integrate an automatic reasoning module into the discovery process to determine 1) the potential data to collect, and 2) the essential models to train, thereby enhancing the likelihood of identifying the correct expression. The reasoning algorithm initially fixes all independent variables except one, instructing the data generator to gather datasets for these controlled variables. Subsequently, it directs the symbolic regressor, a genetic programming algorithm, to uncover the simplified expression involving only the single variable left free. This approach of identifying an expression with fewer input variables, while keeping others controlled, simplifies the task compared to discovering an expression that includes all variables simultaneously. The reasoning module then sequentially releases one independent variable at a time. In each iteration, the GP-based regressor adapts the previously learned equations to include the new independent variable through processes such as mating, mutation, and selection. This reasoning process is repeated until all independent variables are incorporated into the symbolic expression. An illustrative example of this process is provided in Figure 2.

In both applications, our reasoning-embedded learners outperformed purely machine-learning approaches. In design generation, a human study was conducted to evaluate our approach in specification satisfaction, aesthetic appeal, background preservation, and spatial naturalness. Our approach demonstrated equal or superior performance in terms of aesthetic appeal and spatial naturalness when compared to the purely neural baseline. More importantly, it significantly outperformed the baseline in meeting specific design specifications and preserving the integrity of the background. In symbolic regression, we demonstrate that the reasoning-integrated method, i.e., control variable genetic programming (CVGP), finds the symbolic expressions with the smallest Normal-

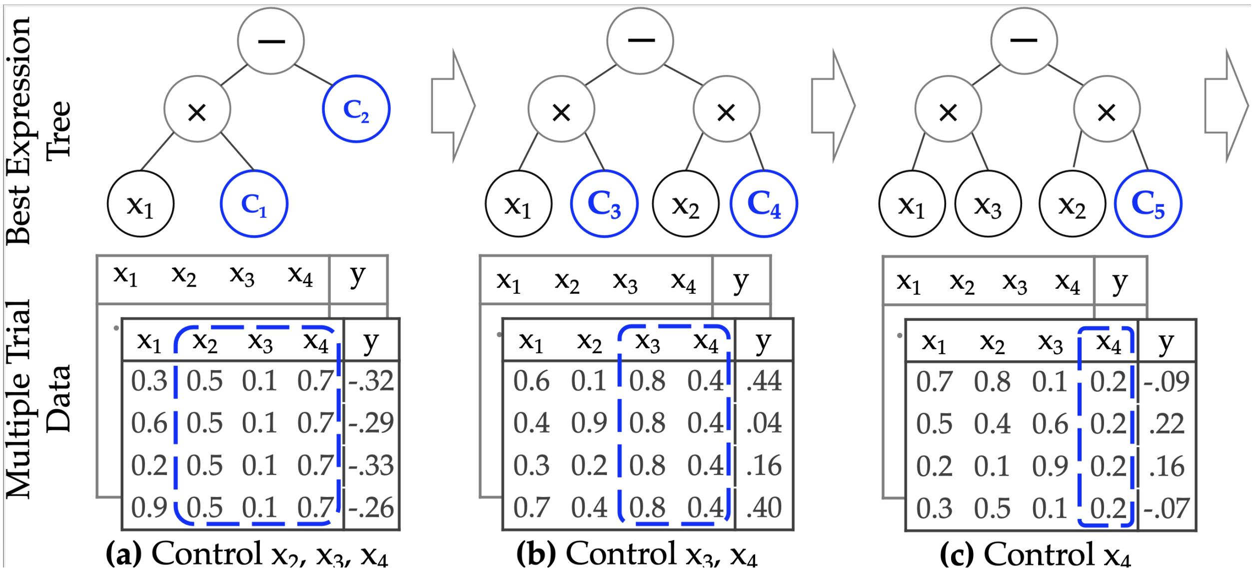

Figure 2: Running example of CVGP. The whole pipeline is determined by the reasoning module that query for the controlled variable dataset and asks the regressor to learn the reduced form expression. **(a)** Initially, a reduced-form equation $\phi' = C_1x_1 - C_2$ is found via fitting control variable data in which x_2, x_3, x_4 are held as constants and only x_1 is allowed to vary. **(b)** This equation is expanded to $C_3x_1 - C_4x_2$ in the second stage via fitting the data in which only x_3, x_4 are held as constants. **(c,d)** This process continues until the ground-truth equation $\phi = x_1x_3 - x_2x_4$ is found. The data generated for control variable experiment trials in each stage are shown at the bottom.

ized Mean-Square Errors (NMSE) among all 7 competing approaches on 21 noiseless benchmark datasets and 20 noisy benchmark datasets. In the ablation studies, we show our CVGP is consistently better than the baselines when evaluated in different evaluation metrics, evaluating different quantiles of the NMSE metric, with different amounts of Gaussian noise added to the data We also show our CVGP has a higher rate of recovering the ground-truth expressions than baselines.

2 Spatial Reasoning Integrated Generator for Interior Design

We introduce **Sp**atial **R**easoning **In**tegrated **G**enerator (SPRING) for design production. SPRING combines neural and constraint reasoning to analyze indoor scenes and generate object placements as bounding boxes. This process is refined by neural models and symbolic constraints, ensuring designs meet user specifications while maintaining aesthetic appeal. Given an initial indoor scene and user requirements described in propositional logic, the task is to generate a design that satisfies user specifications, looks pleasing, and follows common sense. The essence of SPRING is the ***embedding of a neural and symbolic integrated spatial reasoning module within the deep generative network***. The spatial reasoning module decides the locations of the objects to be generated in the form of bounding boxes, following an iterative refinement approach. The bounding boxes are predicted by a sequence-to-sequence neural model and are further filtered by symbolic constraint reasoning (forward checking). This integrated approach leverages the advantages of both neural and symbolic approaches: the constraint program deals with explicit specifications, such as user requirements, while neural networks handle aesthetics and common sense.

SPRING consists of three modules. The first perception module based on Detection Transformers (DETR) [Carion et al., 2020] extracts existing object positions from input images. It is followed by the spatial reasoning module, which uses neural and symbolic integrated approaches to generate the bounding boxes. When determining one coordinate of the bounding box (e.g., the x, y coordinates, width, or height), the recursive neural network in the spatial reasoning module iteratively halves the range of each coordinate until it is sufficiently small. During learning, the spatial reasoning module is trained to understand implicit spatial knowledge, such as potted plants usually being located on the floor, etc. Learned knowledge is reflected in the decisions it makes (that is, which half range of the coordinate falls into in every step). Training is completed by a teacher-forcing procedure that matches the bounding boxes predicted by the spatial reasoning module and those that contain the objects in the training images. During inference, explicit spatial constraints are enforced by a symbolic reasoning algorithm, which blocks decisions that necessarily lead to constraint violations. Finally, the bounding boxes are filled by a visual element generator, which is a diffusion model. The full version of this work is available at [Jacobson and Xue, 2023].

3 Reasoning-enhanced System for AI-driven Scientific Discovery

Our recently proposed Control Variable Genetic Programming (CVGP) implements the reasoning algorithm into the scientific discovery process based on Genetic Programming, for symbolic regression over many independent variables [Jiang and Xue, 2023]. The key insight of CVGP is to learn from *a customized set of control variable experiments*; in other words, the experiment data collection adapts to the learning process. This is in contrast to the current learning paradigm of most symbolic regression approaches, where they learn from a fixed dataset collected a priori.

In CVGP, first, we hold all independent variables except for one as constants and learn an expression that maps the single variable to the dependent variable using GP. GP maintains a pool of candidate expressions and improves the fitness of these equations via mating, mutating, and selection over several generations. Mapping the dependence of one independent variable is easy. Hence GP can usually recover the ground-truth reduced-form equation. Then, CVGP frees one independent variable at a time. In each iteration, GP is used to modify the equations learned in previous generations to incorporate the new independent variable, via mating, mutating, and selection. Such a procedure repeats until all the independent variables have been incorporated into the symbolic expression. See figure 2 for the high-level idea of algorithm execution. Theoretically, in the original paper, we show that CVGP as an incremental builder can reduce the exponential-sized search space for candidate expressions into a polynomial one when fitting a class of symbolic expressions. Experimentally, we show that CVGP outperforms a number of state-of-the-art approaches on symbolic regression over multiple independent variables.

References

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *CoRR*, abs/2005.12872, 2020. URL https://arxiv.org/abs/2005.12872.

Maxwell Joseph Jacobson and Yexiang Xue. Integrating symbolic reasoning into neural generative models for design generation, 2023.

Nan Jiang and Yexiang Xue. Symbolic regression via control variable genetic programming. In *Proceedings of the 2023 European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, Lecture Notes in Computer Science. Springer, 2023.