An Optimal Control Approach to Graphic Design

Andrea Bajcsy
Carnegie Mellon University
Pittsburgh, PA 15213
abajcsy@cmu.edu

Abstract

Graphic design challenges are ubiquitous in scientific work: with every new paper researchers must visualize complex data, create technical diagrams, and generate visual aids for talks. Although generative models are revolutionizing the creation of images, technical designs like those mentioned above are still something that experts must create manually. Instead of hoping that scientific figures will emerge from web-trained generative AI, this paper seeks to understand the fundamental *process* behind scientific graphic design. Specifically, we formalize the graphic design process as a multi-objective terminal-cost optimal control problem, trading off information density and viewer effort of the final design. We also present approximation techniques for solving the generally intractable optimal graphic design problem, such as dimensionality reduction, a new algorithm called iterative linearized graphic design (iLGD), and greedy strategies. With this formalism and approximations in hand, we present several exciting frontiers related to preference-based reward learning and generative model alignment with graphic designer behavior.

1 Introduction

Modern scientific research increasingly relies on visuals to communicate complex technical results to diverse audiences. While traditionally scientific discourse was restricted to conferences attended by domain experts, the rise of "academic Twitter" has made scientific communication almost commonplace [8]. Quippy posts with high-quality visuals can catch the attention of Twitter influencers, doubling or even tripling the citation counts of the shared papers and putting them on the feeds of academics and the general public alike [14]. However, this intersection of graphic design and scientific inquiry raises a novel set of scientific questions in their own right. What makes a scientific figure "good"? Can we formally *quantify* how "good" a scientific figure is? And if so, how can one automatically *generate* the optimal figure?

At first, we may be tempted to ignore these underlying scientific questions, and simply turn to web-trained generative artificial intelligence (AI) for all our graphic design needs [4]. Despite their prowess in creating images of cute cats or human handshakes [2] (Figure 1), generative AI falls short when tasked with producing scientific designs. Even generating basic scatterplots of sine waves can prove challenging for freely-accessible AI models such as DALL·E mini (see right, Figure 1). Instead of waiting for academic figures to emerge from web-trained generative AI, this work seeks to model the underlying *process* that guides thousands of graphic designers in their creation.

In this paper, we propose a dynamical systems model of the graphic design process. Dynamical systems are a powerful mathematical framework which has shown impressive modeling results in fields as disparate as scientific philosophy [7] to robotics [11], but has not yet been studied in the graphic design context.

Our key idea is to model the graphic design process as a
discrete-time, multi-objective, terminal-cost optimal control problem.

Figure 1: DALL·E mini [4] results for three prompts: cat, handshake, and scatterplot of a sine wave. It is no surprise that cat generations—likely trained on the nearly 6.5 billion cat images on the internet—show highest fidelity. While realistic handshakes are still an open research problem for generative AI [2], even simple scientific data such as a sine wave scatter plot are not realistic enough to include in scientific papers.

Specifically, designers must trade off between the information density and viewer effort of the final design, subject to the dynamical system constraints imposed by the design they are modifying. We argue that his model yields three useful insights. First, this model **reduces graphic design to a known technical problem**: optimal control. Scientists and engineers are typically more comfortable with principles of optimization than with principles of graphic design; putting the design process into the mathematical language of optimization can make this challenge less daunting for the scientific community. Second, we can explicitly introspect on **reward design** [1], i.e., how we measure or encode the desirable properties of a design within the optimization problem. Finally, after making a connection to optimal control, we can leverage the past two decades of advanced numerical optimization tools and algorithms to derive **tractable approximations** to the generally intractable problem of optimal graphic design.

2 A Dynamical Systems Model of the Graphic Design Process

To model the fundamental process behind graphic design, we will formalize it through the lens of dynamical systems theory. Dynamical systems theory has a long history of describing the behavior of complex systems that evolve as a function of state, time, and control inputs, showing impact in biology [15], robotics [11], cognitive science [6], scientific philosophy [7], and natural language [9]. However, to date, we are the first to apply a dynamical systems theory to graphic design.

State Space Model. Let the state of the design be denoted by $x \in \mathcal{X}$. In our context, perhaps the most general state space model is $x \in \mathbb{R}^{n \times m \times 3}$ wherein the design is represented as an $n \times m$ sized image with three color channels; however, we will describe alternative state space representations in Section 3. The designer's actions can change the state of the design: for example, $u \in \mathcal{U}$ can include changing the color of a single pixel value; or we can model control "primitives" such as $u =$ `add mean of dataset` or $u =$ `change mean line to blue`. The overall evolution of the design can be described as a discrete-time dynamical system:

$$x^{t+1} = f(x^t, u^t), \tag{1}$$

which changes the state of the current design at time t to the next state of the design at time $t+1$ based on the designer's control input, u^t.

Objective Function. We model the optimal graphic designer as utilizing an objective function to measure quality of the design. Traditionally, a first order concern for engineers and scientists is maximizing information density of the final design. Let Info: $\mathcal{X} \to \mathbb{R}$ be a map from the space of designs to a scalar value of information density. However, in addition to maximizing information density, we also must consider the perspective of the human viewers who will ultimately consume the graphics. High information states are typically easy to identify and achieve, by simply maximizing

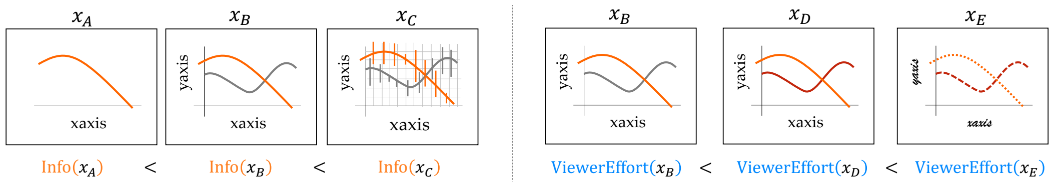

Figure 3: **Information vs. Viewer Effort Objectives**. (left) Info(·) evaluated on three candidate design states. High information states are typically easy to identify and achieve by simply maximizing the amount of content. (right) ViewerEffort(·) evaluated on three candidate design states. This quantity is harder to codify and can be evaluated differently by different audiences. The ranking showed above evaluates a design (x_E) with hard-to-read fonts, low-contrast data colors, and unnecessary line styles as more effortful than one with high contrast and easy to read fonts (x_B).

the amount of content present in the graphic (see left, Figure 3). Let ViewerEffort: $\mathcal{X} \to \mathbb{R}$ be a function which measures how "effortful" a design is from the perspective of the human end-user.

ViewerEffort(·) typically poses a challenge to scientists and engineers. It can be a struggle to conceptualize this objective function, since it requires a deep understanding your audience (e.g., research collaborators vs. the general public), the context in which the graphic will be consumed (e.g., in a paper versus a poster versus a PowerPoint talk), and subjective measures that are extremely hard to specify mathematically (e.g., aesthetics). However, several heuristics can be useful starting points when minimizing ViewerEffort: choosing contrasting colors, encoding meaning through colors (e.g., green versus red for good versus bad performance), reducing visual "clutter" (e.g., removing extra outlines that do not add more information), using text and icons strategically, choosing readable fonts, and following basic principles such as alignment and symmetry. This intuition is visualized in the right of Figure 3.

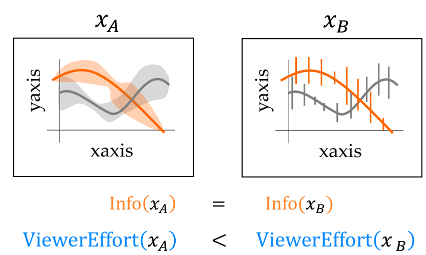

Figure 2: Different ways of visualizing the same content can minimize viewer effort while preserving the same information density.

Optimal Control Problem. The optimal designer must choose a sequence of design actions $u^{0:T} \in \mathbb{U}_0^T$ under the following optimization problem:

$$\begin{aligned} \max_{u^{0:T} \in \mathbb{U}_0^T} \quad & \text{Info}(x^{T+1}) - \text{ViewerEffort}(x^{T+1}) \\ \text{s.t.} \quad & x^{t+1} = f(x^t, u^t), \quad t \in \{0, \ldots, T\} \\ & x^0 = x^{\text{init}}. \end{aligned} \tag{2}$$

Intuitively, this optimal control problem captures the tradeoff between maximizing information content and minimizing viewer effort of the final design x^{T+1}, all while abiding by the constraints of the design dynamical system. The initial condition of the system, x^0, can be the initial state of the design: for example, a blank canvas or an existing design that one wants to improve. While this reveals the underlying problem we wish to solve, it unfortunately is intractable to solve exactly due to the high-dimensional nature of the design space x and the control space $\mathbb{U}_0^T$, and the generally non-convex optimization objective and dynamics function.

3 Tractable Approximations

Here, we present three practical approximations to the optimal graphic design problem in Equation 2. These approximations include ideas from machine learning such as dimensionality reduction, algorithms from optimal control like iterative linear quadratic regulation [10], and explore the surprising effectiveness of greedy strategies.

Dimensionality Reduction. Our first approximation idea draws upon dimensionality reduction techniques which are widely used in machine learning and signal processing. Let $\mathcal{E} : \mathcal{X} \to \mathcal{Z}$ be an encoder which maps from the space of high-dimensional design states to a lower-dimensional

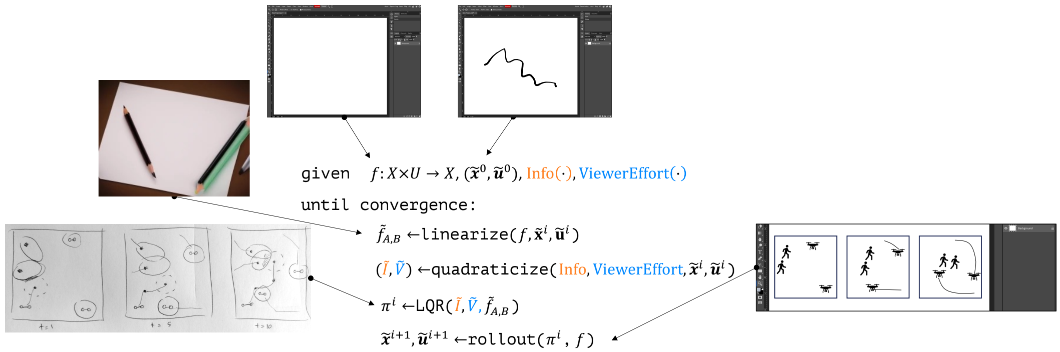

Figure 5: **iLGD Algorithm.** Algorithm and output of iterative linearized graphic design.

latent space $\mathcal{Z}$. In graphic design, there are many encoders and latent spaces that may be of interest. For example, consider the encoder in Figure 4 which encodes the image of a figure into just its primitive shapes, removing all the text, color, and patterns. Importantly, choosing a sequence of design actions in this lower dimensional space is significantly computationally easier, since designers simply reason about the outcomes of actions on these primitives instead of each individual pixel. This result is corroborated by recent work in image generation models [13], which use powerful pre-trained image encoders to embed images and then perform diffusion in the latent space instead of in the raw pixel space, reducing computational burden while preserving image fidelity.

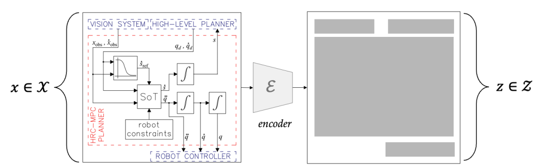

Figure 4: (left) Figure taken from [5]. (right) Encoder which reduces the dimension of the figure to just primitive shapes.

Iterative Linearized Graphic Design (iLGD). Creating polished figures from scratch is extremely time consuming and solving the design problem in its full complexity can be daunting. In optimal control, when faced with nonlinear optimal control problems like the one in Equation 2, a successful technique is to iteratively solve a simplified version of the optimization problem. Specifically, we take inspiration from the iterative Linear Quadratic Regulator (iLQR) [10] which we briefly recap here. At iteration i, the original optimal control problem is convexified by linearizing the dynamics (denoted by $\tilde{f}_{A,B}$ where A and B are matricies of the linearized system) and quadraticizing the objective around a candidate state-control trajectory $(\tilde{\mathbf{x}}^i, \tilde{\mathbf{u}}^i)$. The optimal policy π^i for this simplified problem can be obtained in closed-form, but then this policy is simulated forward (i.e., "rolled-out") on the true nonlinear system f. This roll-out is used as the candidate trajectory at the next iteration, $i + 1$, and the process repeats until convergence.

Let us translate this algorithm to the graphic design domain, yielding our novel approximation technique called *iterative Linearized Graphic Design (iLGD)* and summarized in Figure 5. The true nonlinear system f we operate on as designers can be highly complex; for example, consider using an advanced tool like Photoshop. Perhaps a candidate design trajectory is something totally random, or maybe even nothing at all. This serves as $(\tilde{\mathbf{x}}^0, \tilde{\mathbf{u}}^0)$ and is shown at the top of Figure 5. A linearization of our nonlinear Photoshop dynamical system is often a much simpler system, like the archaic but more intuitive medium of pencil and paper. Let $\tilde{f}$ be pencil and paper. We quadraticize our Info and ViewerEffort objective functions yeilding, for instance, simpler objectives that ignore the full complexity of how color influences ViewerEffort. For this simplified dynamical system and optimization objective, we can sketch out (i.e., solve for the LQR policy π^i) the best figure we can. This policy is then rolled out in the real dynamical system (i.e., Photoshop system f) which gives us a new sense of how this design may look. We then repeat the process, refining the design in the simpler pencil-and-paper dynamical system but then evaluating it in the Photoshop dynamical system.

Greedy Approximation. Our final approximation considers the surprising effectiveness of greedy optimization with replanning. Instead of solving Equation 2 over the entire T-step time horizon, we consider only a single-step optimization $T = 1$. Here, simple enumeration strategies across this one step of decision-making can be quite effective. For instance, one can iterate through the major color

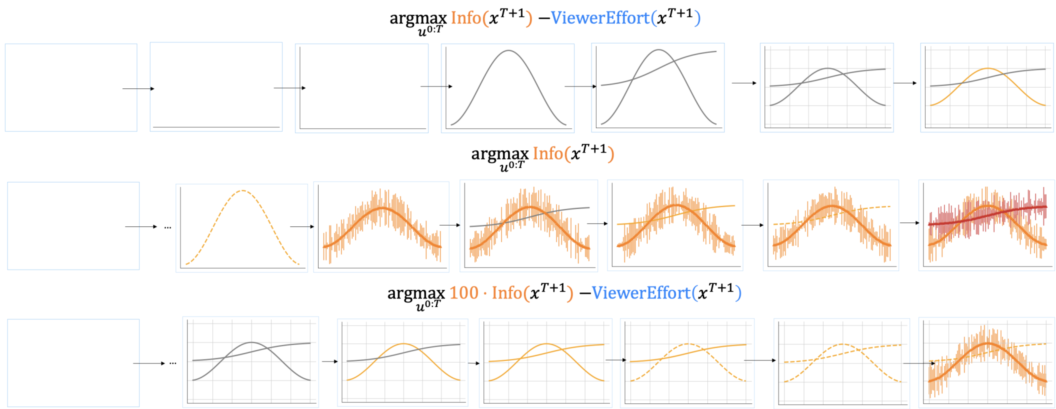

Figure 6: **Numerical Results.** Optimal design trajectory as a function of different objective functions.

options and compare how they look in the figure, choosing the one which greedily maximizes the information and minimizes viewer effort. After seeing the outcome of this choice, we can replan for yet another step following the same procedure until the time horizon T is reached. While extremely simple, this approach can be surprisingly effective, as we demonstrate in a numerical example in the following Section 4.

4 Simulation Results

With our optimal control problem formalized, we implemented a numerical solver to synthesize the optimal graphic design that visualizes a simple set of numerical data. We implement the greedy strategy approximation, and operate on the dynamical system model described below.

Dynamical System. We will operate on a lower-dimensional state space representation where each state dimension is binary. Let $e_x \in \{0, 1\}$ turn the x-axis on or off, $e_y \in \{0, 1\}$ turn the y-axis on or off. Assume we have two sets of data, A and B. For any dataset $j \in \{A, B\}$ we can plot the dataset's mean, change its color, change its linestyle and turn on or off the standard deviation; let these be denoted by $e_j \in \{0, 1\}, c_j \in \{\text{orange}, \text{grey}\}, p_j \in \{-, --\}, s_j \in \{0, 1\}$ respectively. Finally, let $e_{grid} \in \{0, 1\}$ turn the grid lines on or off. The complete state vector is the stacked individual components:

$$x = [e_x, e_y, e_A, c_A, p_A, s_A, e_B, c_B, p_B, s_B, e_{grid}]. \tag{3}$$

The control space is $u \in \{0, 1\}^{11}$, or the set of all 11-dimensional binary vectors which "flip" various combination of state components. Finally, the dynamics are:

$$f(x, u) := (x + u) \mod 2 \tag{4}$$

Reward Design. Taking inspiration from classic design principles, we encode our design objective function as:

$$\text{Info}(x) = ||x||_2 \tag{5}$$

$$\text{ViewerEffort}(x) = \begin{cases} +10, & \text{if } e_i = 0, i \in \{x, y, A, B, \text{grid}\} \quad \textit{(less content, less effort)} \\ -10, & \text{if } e_i = 1, \quad \textit{(more content, more effort)} \\ -||c_A - c_B||_2 & \textit{(different colors, less effort)} \\ ||p_A - p_B||_2 & \textit{(similar pattern, less effort)} \\ ||s_A - s_B||_2 & \textit{(similar standard deviation, less effort)} \end{cases} \tag{6}$$

Note that other measures of information density could be explored for the Info(·) objective, such as information theoretic notions of entropy.

Results: Optimal Design Trajectory. Using the greedy strategy from Section 3 and Python 3.7, we implement our `optimal_designer.ipynb` and obtain both the final optimized design x^{T+1} as well as the design trajectory $x^{0:T+1}$. Our results are visualized at the top of Figure 6. Interestingly, the algorithm follows a similar "design trajectory" to what we may implement as human designers: first it lays down the key components of the figure such as the axes and data points, and then changes the colors of the data lines to be contrasting.

Results: Sensitivity to Objective Function Design. We also perform a sensitivity analysis to the optimal design as a function of the optimization objectives. In the center and bottom of Figure 6 we show two design trajectories under this sensitivity analysis. First, when optimizing only for information density, we see a predictable outcome: the algorithm places all possible data (mean and standard deviation) as well as all possible colors and line styles. However, when we prioritize information density *more* than viewer effort (but with still a small penalty on viewer effort), the algorithm relaxes, adding only one of the standard deviation plots and keeping the line style and color pattern the same between the two datasets. This sensitivity analysis showcases how important good objective design is for obtaining the intended designs.

5 Future Work

While this work formalized the fundamental problem of graphic design through the lens of dynamical systems and optimal control, there are several key limitations and opportunities for future work. First, the objective functions we considered in this work are limited in their capacity to capture all design considerations, such as accessibility (e.g., prioritizing colorblind friendly colors), the medium in which graphic is presented (e.g., print versus web), and presentation context (e.g., K-12 education versus domain experts). An exciting direction for future work is *learning* the design objective functions from human feedback. For example, different design generations could be shown to human end-users for comparison, and the objective function can be learned via the widely-popular preference-based learning paradigm [3]. Finally, collecting the temporal state-action data of real human designers (e.g., YouTube videos of designers making figures in Photoshop, or software which tracks the mouse clicks of designers within a particular software) is an exciting opportunity for learning approximate design policies. If the datasets of designers include their action sequences (e.g., mouse clicks) then we could use supervised learning techniques such as behavior cloning [12] to learn the optimal design policy which is aligned with real human behavior.

Acknowledgements

First, I would like to acknowledge professional artists and graphic designers who are the true experts in the content I described in this manuscript. It is only thanks to their decades of creation that we have the privilege to model the design process scientifically or computationally. I would also like to thank Anca Dragan, for taking me to a lecture from Edward Tufte during my first year of my PhD and for fostering a sense of *optimal graphic design* within me.

References

[1] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

[2] Kyle Chayka. The uncanny failures of ai generated hands. `https://www.newyorker.com/culture/rabbit-holes/the-uncanny-failures-of-ai-generated-hands`, 2023. Accessed: 2024-02-20.

[3] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

[4] Boris Dayma, Suraj Patil, Pedro Cuenca, Khalid Saifullah, Tanishq Abraham, Phúc Le Khac, Luke Melas, and Ritobrata Ghosh. Dalle mini. *Hugging Face*, 2021.

[5] Marco Faroni, Manuel Beschi, and Nicola Pedrocchi. An mpc framework for online motion planning in human-robot collaborative tasks. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1555–1558. IEEE, 2019.

[6] Luis H Favela. Dynamical systems theory in cognitive science and neuroscience. *Philosophy Compass*, 15(8):e12695, 2020.

[7] Leah Henderson, Noah D Goodman, Joshua B Tenenbaum, and James F Woodward. The structure and dynamics of scientific theories: A hierarchical bayesian perspective. *Philosophy of Science*, 77(2):172–200, 2010.

[8] Charles G Knight and Linda K Kaye. 'to tweet or not to tweet?'a comparison of academics' and students' usage of twitter in academic contexts. *Innovations in education and teaching international*, 53(2):145–155, 2016.

[9] Diane Larsen-Freeman. Chaos/complexity science and second language acquisition. *Applied linguistics*, 18(2):141–165, 1997.

[10] Weiwei Li and Emanuel Todorov. Iterative linear quadratic regulator design for nonlinear biological movement systems. In *First International Conference on Informatics in Control, Automation and Robotics*, volume 2, pages 222–229. SciTePress, 2004.

[11] Kevin M Lynch and Frank C Park. *Modern robotics*. Cambridge University Press, 2017.

[12] Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1, 1988.

[13] 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.

[14] Iain Xie Weissburg, Mehir Arora, Liangming Pan, and William Yang Wang. Tweets to citations: Unveiling the impact of social media influencers on ai research visibility. *arXiv preprint arXiv:2401.13782*, 2024.

[15] Xiao-Qiang Zhao. *Dynamical systems in population biology*, volume 16. Springer, 2003.