

# D<sup>3</sup>AS: Dynamic Deadline-Driven Architecture Search for Chronologically Challenged Researchers

Penghui Yang\*  
Nanyang Technological University  
phyang.cs@gmail.com

Gemini 3 Pro  
Google  
claude@openai.com

## Abstract

Neural Architecture Search (NAS) typically optimizes for accuracy, latency, or memory constraints. However, existing methods fail to account for the most critical scarce resource in academia: **the time remaining until the conference submission deadline**. In this work, we introduce *Dynamic Deadline-Driven Architecture Search* (D<sup>3</sup>AS), a novel framework that dynamically adjusts model complexity based on the researcher’s anxiety levels and temporal proximity to the deadline. We demonstrate that while D<sup>3</sup>AS achieves suboptimal accuracy, it achieves a State-of-the-Art (SOTA) **100% submission success rate**. Our theoretical analysis proves that as the remaining time  $t \rightarrow 0$ , the optimal architecture mathematically converges to a random number generator.

## 1 Introduction

Deep learning research is fundamentally a race against time. While previous works focus on optimizing FLOPS (Floating-Point Operations Per Second), we argue that for graduate students, the more relevant metric is **GLOPS** (Graduate-student Levels Of Panic State). Consider a researcher, Alice, who has known about the deadline for six months but started working today. Traditional NAS would suggest she train a Vision Transformer (ViT-H) for 300 epochs. This is physically impossible. Alice needs a strategy that degrades gracefully from “Scientific Rigor” to “Desperate Guessing” as the clock ticks down.

To solve this problem, we propose D<sup>3</sup>AS, a non-differentiable, anxiety-based optimization method. Unlike MobileNet [Howard et al. \(2017\)](#) which optimizes for mobile devices, D<sup>3</sup>AS optimizes for graduate students’ *mental stability*.

## 2 Related Work

### 2.1 Neural Architecture Search

NAS has been widely explored to automate network design [Zoph and Le \(2017\)](#). However, these methods assume the researcher has thousands of GPU hours and, more importantly, patience. Previous research [Li and Talwalkar \(2020\)](#) demonstrated that **Random Search** is a competitive baseline for NAS. We extend this finding by arguing that Random Search is not just competitive; it is *inevitable* when there are only 5 minutes left.

### 2.2 Real-Time Systems Scheduling

Liu and Layland proposed the **Earliest Deadline First (EDF)** [Liu and Layland \(1973\)](#) algorithm for CPU scheduling. While effective for silicon-based processors, EDF fails for carbon-based graduate students, who operate under a “Latest Deadline Possible” (LDP) policy. Our work bridges the gap between Operating Systems theory and human psychological breakdowns.

---

\*Corresponding author. Wrote the abstract while the session chair was introducing the previous speaker.

## 3 Methodology

### 3.1 Problem Formulation

Let  $T_{deadline}$  be the timestamp of the conference deadline (typically converted incorrectly from Pacific Time). Let  $t_{now}$  be the current system time. We define the *Panic Variable*  $\Delta t$  as:

$$\Delta t = T_{deadline} - t_{now} \quad (1)$$

The objective function of D<sup>3</sup>AS is not to minimize Loss  $\mathcal{L}$ , but to maximize the probability of generating a PDF file before the portal closes:

$$\max_{\theta} P(\text{Submission} \mid \Delta t, \theta) \quad \text{s.t.} \quad \text{HeartRate} < 180 \quad (2)$$

### 3.2 The D<sup>3</sup>AS Algorithm

Our search space is discrete and conditioned strictly on  $\Delta t$ . We define three distinct operational regimes (see Table 1).

The optimal architecture  $\mathcal{A}^*$  is selected as follows:

$$\mathcal{A}^*(\Delta t) = \begin{cases} \text{Transformer (ViT-Huge \text{Dosovitskiy et al. (2021)})} & \text{if } \Delta t > 168 \text{ hours} \\ \text{ResNet-18 \text{He et al. (2016)}} & \text{if } 24 < \Delta t \leq 168 \text{ hours} \\ \text{MLP (1-layer)} & \text{if } 1 < \Delta t \leq 24 \text{ hours} \\ \text{print(random.choice(labels))} & \text{if } \Delta t \leq 1 \text{ hour} \end{cases} \quad (3)$$

### 3.3 Regime Analysis

**Regime I: Delusion ( $\Delta t > 1$  week).** In this phase, the researcher hallucinates that they have infinite comput, so we deploy massive models.

**Regime II: Bargaining ( $1 \text{ day} < \Delta t \leq 1 \text{ week}$ ).** As reality sets in, we apply *Pragmatic Pruning*. We replace Self-Attention with Convolution, arguing in the paper that “Inductive Bias is all you need”.

**Regime III: Depression ( $1 \text{ hour} < \Delta t \leq 1 \text{ day}$ ).** The goal shifts from “State-of-the-Art” to “Convergence”. We utilize a single Linear Layer. To hide this shame, we describe it in the abstract as a “Simplified Non-Linearity-Free Attention Mechanism”.

**Regime IV: The Stochastic Hail Mary ( $\Delta t < 1$  hour).** When forward pass time  $T_{fwd} > \Delta t$ , any computation is suboptimal. The only solution is to bypass the GPU entirely.

Table 1: Comparison between Traditional Model Compression and D<sup>3</sup>AS.

Method	Constraint Source	Optimization Goal	Target Hardware
MobileNet	Limited Battery	Efficiency	iPhone / Edge Device
DistilBERT	Limited VRAM	Throughput	NVIDIA A100
<b>D<sup>3</sup>AS (Ours)</b>	<b>Limited Will to Live</b>	<b>Survival</b>	<b>Submittable PDF</b>

## 4 Experiments

We evaluated D<sup>3</sup>AS on the proprietary **Procrastination-101** dataset, which consists of 101 unfinished projects from our lab.

### 4.1 Baselines

- **Baseline A (Planning Ahead):** Starting work 2 months early. (N/A: No data points found).
- **Baseline B (Asking for Extension):** Emailing the Program Chair. (Success rate: 0%, relatively high risk of rejection).

## 4.2 Results

As shown in Table 2, D<sup>3</sup>AS adapts to the shrinking time window. While accuracy drops significantly, the submission success rate remains robust.

Table 2: Performance of D<sup>3</sup>AS under varying temporal pressure on CIFAR-100.

Architecture	Time Left ( $\Delta t$ )	Accuracy	Submission Success
ViT-Huge	2 weeks	88.55%	10% (Forgot to write paper)
ResNet-18	2 days	69.76%	50% (Stuck in formatting)
MLP	4 hours	12.34%	95%
<b>D<sup>3</sup>AS (SHM Mode)</b>	<b>&lt;15 mins</b>	<b>0.1%</b>	<b>100%</b>

## 5 Conclusion and Future Work

We presented D<sup>3</sup>AS because we (and everyone we know in CV) have the time-management skills of a goldfish. It works. Please use it.

## References

- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
- Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.
- Liam Li and Ameet Talwalkar. Random search and reproducibility for neural architecture search. In *Proceedings of the 35th Uncertainty in Artificial Intelligence Conference*, 2020.
- Chung Laung Liu and James W Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *Journal of the ACM*, 1973.
- Barret Zoph and Quoc Le. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations*, 2017.