

Neural Thermodynamic Laws for Large Language Model Training

Ziming Liu*, Yizhou Liu, Jeff Gore, Max Tegmark
Massachusetts Institute of Technology

Abstract

Beyond neural scaling laws, little is known about the laws underlying large language models (LLMs). We introduce *neural thermodynamic laws* (NTL) – a new framework that offers fresh insights into LLM training dynamics. On the theoretical side, we demonstrate that key thermodynamic quantities (e.g., temperature, entropy, heat capacity, thermal conduction) and classical thermodynamic principles (e.g., the three laws of thermodynamics and the equipartition theorem) naturally emerge under river-valley loss landscape assumptions. On the practical side, this scientific perspective yields intuitive guidelines for designing learning rate schedules.

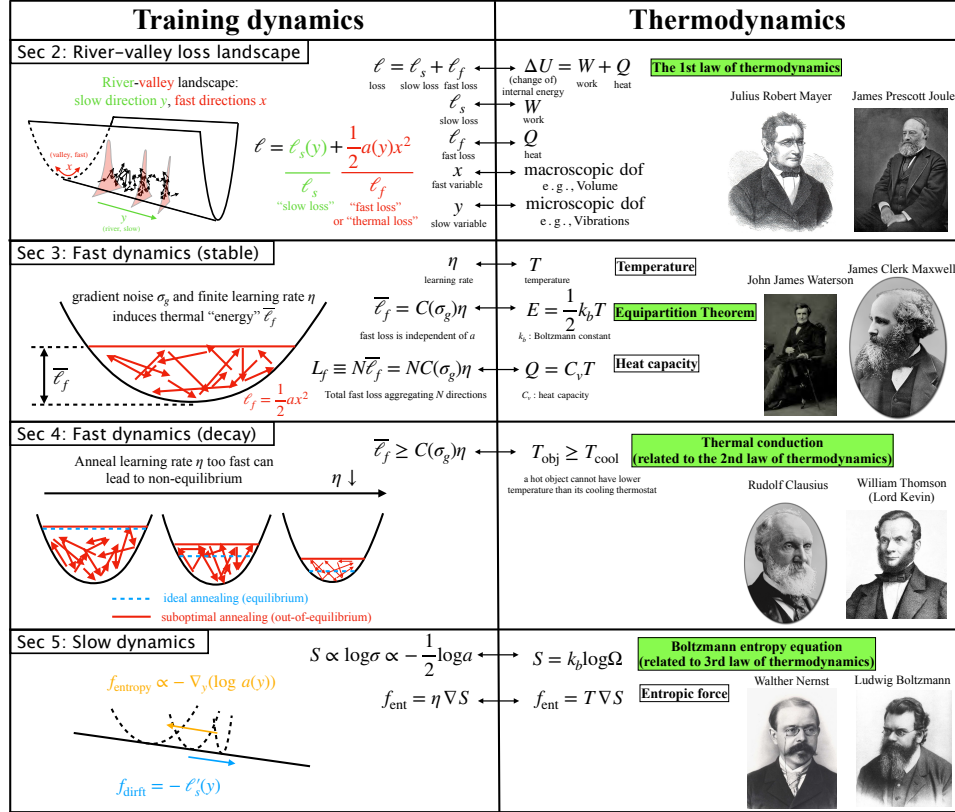


Figure 1: Connections between LLM training dynamics and thermodynamics.

*zmliu@mit.edu

1 Introduction

Large neural networks bear striking similarities to thermodynamic systems – both involve a vast number of degrees of freedom and exhibit stochastic dynamics. It is therefore not surprising that connections between neural networks and thermodynamics have been explored in prior work [1, 2, 3, 4]. However, these studies primarily focus on classical machine learning models with relatively well-understood loss landscapes. In contrast, recent research has only begun to shed light on the complex loss landscapes of large language models (LLMs), characterized by the so-called “river-valley” structure – comprising sharp, fast directions (valley) and flat, slow directions (rivers) [5, 6, 7]. Intuitively, the fast dynamics rapidly “equilibrate” within valleys, while the slow dynamics evolve gradually along rivers, subtly modulated by the fast components. The goal of this paper is to formalize this intuition through the lens of *Neural Thermodynamic Laws* (NTL). We show that key thermodynamic quantities and principles – including temperature, entropy, heat capacity, thermal conduction, the three laws of thermodynamics, and the equipartition theorem – emerge naturally from the training dynamics of LLM (see the connections between training dynamics and thermodynamics in Figure 1).

The duality between LLM training dynamics and thermodynamics is not only conceptually and theoretically compelling, but also provides practical insights – for example, into the design of learning rate schedules. A common learning rate schedule used in LLM pretraining is the warmup-stable-decay (WSD). According to [5, 8], the stable phase corresponds to motion along the river, with fluctuations in the valley directions, while the decay phase suppresses these valley variations. Motivated by this, we introduce a toy model of the river-valley landscape. This model is analytically solvable, admits a natural thermodynamic interpretation, and shows strong empirical agreement with actual LLM training dynamics.

The paper is organized as follows. The timescale separation between fast and slow dynamics allows us to decompose the total loss function ℓ into two components – the fast part ℓ_f and the slow part ℓ_s , which motivates our toy model of the river-valley landscape (Section 2). With a fixed learning rate, the fast dynamics converge to a steady-state distribution, analogous to thermal equilibrium (Section 3). When the learning rate decays, the distribution evolves accordingly – resembling annealing (Section 4). Moreover, the fast dynamics exert an effective entropic force on the slow dynamics, similar to entropic forces in physics (Section 5). Notably, the learning rate η plays a central role in all of these phenomena. By clarifying its complex and sometimes contradictory effects, we derive an intuitive guideline for designing efficient learning rate schedules (Section 6), followed by related works (Section 7) and conclusions (Section 8).

Unlike prior work that approaches LLM optimization, especially the design of learning rate schedules, from largely empirical or phenomenological perspectives, our characterizations are more mechanistic. Our technical contributions are as follows:

- **Formulation of Fast-slow decomposition.** In river-valley landscapes, we decompose training into two processes: (1) fast dynamics: either *equilibrium* (under fixed η) or *annealing* (under decaying η) along the *valley* and (2) slow dynamics: *drift* along the *river*.
- **An exactly solvable toy model.** We introduce a tractable toy model of the river-valley landscape that captures both fast and slow dynamics. This model admits analytical solutions for training behavior and optimal learning rate schedules.
- **Empirical relevance to LLMs.** We demonstrate that insights from the toy model generalize well to real LLM training, providing intuitive and effective heuristics for the learning rate schedule.
- **A bridge to physics** The duality between neural network training and thermodynamics provides a foundation for developing a deeper scientific understanding of deep learning.

2 River Valley Loss Landscape

Recent work [5] showed that LLM loss landscape resembles a river-valley landscape: a flat river lies at the bottom of sharp valleys. Training slowly progresses along the river while bouncing quickly between the sharp hillsides. Throughout the paper, we interchangeably use valley dynamics = fast dynamics, river dynamics = slow dynamics.

A dilemma for learning rate η A good learning rate should strike a good balance between the two objectives: (A) enabling progress along the river directions – where the loss typically decreases monotonically – which favors a large η ; and (B) minimizing variance along the valley directions – which favors small η . In WSD schedules, the stable phase takes care of (A), while the decay phase takes care of (B) [5]. To better understand this trade-off, we introduce a toy model that admits analytical characterization of the training dynamics along both river and valley directions.

2.1 Toy model

The toy model $\ell(x, y) = c(y) + \frac{1}{2}a(y)x^2$ is a loss function in 2D, resembling the river-valley landscape in the top left of Figure 1. It consists of a fast variable x and a slow variable y . For any fixed y , the loss is minimized at $x = 0$, which traces out the riverbed at the bottom of the valley, with corresponding loss $c(y)$. The loss function decomposes additively into two components: the valley component $\ell_f(x, y) \equiv \frac{1}{2}a(y)x^2$ (called *fast loss* or *thermal loss*) and the river component $\ell_s(x, y) \equiv c(y)$ (called *slow loss*). In the remainder of the paper, we analyze the training dynamics of SGD and SignGD on this landscape – under learning rate η and gradient noise σ_g – and demonstrate its relevance to the training behavior of large language models.

2.2 Thermodynamics: First law of thermodynamics

The decomposition of fast and slow dynamics is reminiscent of quasi-static equilibrium in thermodynamics. Consider a piston slowly changing the volume of a gas-filled chamber: while the piston (a slow variable) moves slowly, the gas molecules (fast variables) undergo rapid thermal motion and quickly reach a new thermal equilibrium. According to the first law of thermodynamics $\Delta U = W + Q$, the change in internal energy ΔU is composed of work W (associated with slow dynamics) and heat Q (associated with fast dynamics). This mirrors the decomposition of the loss function in the river-valley landscape $\ell = \ell_s + \ell_f$, where ℓ_s captures slow dynamics and ℓ_f captures fast dynamics. In this analogy, the slow variable y corresponds to macroscopic quantities such as volume, while the fast variable x corresponds to microscopic degrees of freedom, such as atomic vibrations. In the next section, we examine how the fast fluctuations in x contribute to the fast loss component ℓ_f . Although this decomposition is conceptually interesting, it does not yet provide a quantitative characterization of ℓ_f and ℓ_s , which we aim to address in the remainder of the paper.

3 Valley Dynamics in Equilibrium (Stable phase)

The fast-slow separation allows us to treat the slow variable y as fixed while analyzing the fast dynamics. Relevant to the fast dynamics is the fast loss $\ell_f(x, y) \equiv \frac{1}{2}a(y)x^2$. For simplicity, we drop the dependence on y and write $\ell_f(x) = \frac{1}{2}ax^2$. We consider stochastic gradient descent (SGD) or signed gradient descent (SignGD)² on this quadratic function, with learning rate η and gradient noise σ_g . Under this quadratic approximation, the training dynamics converge to a Gaussian steady-state distribution, $p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{x^2}{2\sigma^2})$, characterized by width σ . Our goal is to understand how σ depends on sharpness a , learning rate η , and gradient noise σ_g . In Section 3.1, we derive the functional form $\sigma = \sigma(\eta, a, \sigma_g)$ for SGD and SignGD. Section 3.2 provides a thermodynamic interpretation of the results, followed by empirical validation on LLM training in Section 3.3.

3.1 Toy model: steady distribution

SGD An SGD optimizer with learning rate η and gradient noise σ_g obeys the following dynamics:

$$x_{t+1} = x_t - \eta(ax_t + \sigma_g \dot{W}), \quad (1)$$

where $\dot{W} \sim \mathcal{N}(0, 1)$ is the standard Brownian motion. The equilibrium distribution is the Gaussian distribution $p_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$ (see derivtions in Appendix A). In equilibrium, the variance is preserved, i.e., $\text{Var}(x_{t+1}) = \text{Var}(x_t)$, giving rise to the equation $\sigma^2 = (1 - \eta a)^2 \sigma^2 + \sigma_g^2$, resulting

²For simplicity, we set momentums to zero. Note that SignGD is a special case of Adam when $(\beta_1, \beta_2) = (0, 0)$.

Table 1: Optimization on the 1D quadratic function $\ell(x) = \frac{1}{2}ax^2$

Optimizer	SGD	SignGD
Equation	$x_{t+1} = x_t - \eta(ax_t + \sigma_g \dot{W})$	$x_{t+1} = x_t - \eta \text{sign}(ax_t + \sigma_g \dot{W})$
Steady distribution width σ	$\frac{\sigma_g}{\sqrt{a(\frac{2}{\eta} - a)}}$	$\frac{\sqrt{\pi}}{4} \eta \sqrt{1 + \sqrt{1 + \frac{32}{\pi} (\frac{\sigma_g}{a\eta})^2}}$
σ (flat limit $a\eta \ll 1$)	$\sqrt{\frac{\eta}{2a}} \sigma_g \propto \eta^{1/2} a^{-1/2} \sigma_g$	$(\frac{\pi}{8})^{1/4} \sqrt{\frac{\sigma_g \eta}{a}} \propto \eta^{1/2} a^{-1/2} \sigma_g^{1/2}$
Thermal loss $\bar{\ell}_f$	$\frac{\sigma_g^2}{4} \eta$	$\sqrt{\frac{\pi}{32}} \sigma_g \eta$
Optimal decay schedule	$\eta_t = \frac{\frac{\eta}{2}}{1 + \frac{t}{t_h}} \quad (t_h = \frac{2}{a\eta})$	$\eta_t = \frac{\frac{\eta}{2}}{1 + \frac{t}{t_h}} \quad (t_h = \sqrt{2\pi} \frac{\sigma_g}{a\eta})$

in $\sigma = \sigma_g / \sqrt{a(2/\eta - a)}$. This formula is only well-defined when $0 < a < \frac{2}{\eta}$. When $a \rightarrow 2/\eta$, the learning rate reaches the so-called ‘‘edge of stability’’ [9]. Since most directions in an over-parametrized model are relatively flat, we are interested in the flat limit when $a \ll 2/\eta$, which simplifies σ to $\sigma \approx \sqrt{\eta/(2a)} \sigma_g \propto \eta^{1/2} a^{-1/2} \sigma_g$.

SignGD An SignGD optimizer with learning rate η and gradient noise σ_g obeys the dynamics:

$$x_{t+1} = x_t - \eta \text{sign}(ax_t + \sigma_g \dot{W}). \quad (2)$$

Using the variance preservation condition as in SGD, we derive the steady-state Gaussian width $\sigma = \frac{\sqrt{\pi}}{4} \eta \sqrt{1 + \sqrt{1 + \frac{32}{\pi} (\frac{\sigma_g}{a\eta})^2}}$. The flat limit $a \ll \sigma_g/\eta$ gives $\sigma \approx (\frac{\pi}{8})^{1/4} \sqrt{\frac{\sigma_g \eta}{a}} \propto \eta^{1/2} a^{-1/2} \sigma_g^{1/2}$. Detailed derivations are postponed to Appendix C. Results are summarized in Table 1 for reference.

Thermal loss The averaged thermal loss is $\bar{\ell}_f = \mathbb{E}_{x \sim p_\sigma(x)}(\frac{1}{2}ax^2) = \frac{1}{2}a\sigma^2$. Notice that for both SGD and SignGD, $\sigma \propto a^{-1/2}$, it follows that a is cancelled out in $\bar{\ell}_f$, i.e., $\bar{\ell}_f \propto \eta \sigma_g^2$ for SGD and $\bar{\ell}_f \propto \eta \sigma_g$ for SignGD. The independence of a leads to an interesting fact: given two directions with different sharpness $a_1 \neq a_2$, they induce the same $\bar{\ell}_f$ (as long as η and σ_g are the same). This also means: no matter how long the stable phase goes on (resulting in different points along the river with different sharpnesses), the reducible thermal loss is roughly the same in the decay phase, which explains the observations in [10]. The averaged thermal loss $\bar{\ell}_f$ can also be interpreted as being averaged over many valley directions.

3.2 Thermodynamics: Equi-partition theorem, Temperature, Heat capacity

The independence of sharpness corresponds to the *equipartition theorem* in thermodynamics, which states that: in a system in thermal equilibrium, energy is distributed equally among all degrees of freedom that appear quadratically in the system’s energy. Quantitatively, $E = \frac{1}{2}k_b T$, Where k_b is the Boltzmann constant and T is the temperature. In particular, the energy of a vibrational degree of freedom is independent of the spring constant, which corresponds to sharpness a in our case. Now we can make a mapping between optimization $\bar{\ell}_f \propto \sigma_g^n \eta$ ($n = 1$ for SignGD, $n = 2$ for SGD) and thermodynamics $E = \frac{1}{2}k_b T$. Ignoring constants, we have effective temperature $T \sim \eta$, i.e., the learning rate η can be interpreted as *temperature*. Given this, the slope $C \equiv \frac{\partial \bar{\ell}_f}{\partial \eta}$ can be interpreted as *heat capacity*. Now we can simply relate $\bar{\ell}_f$ and η by $\bar{\ell}_f = C\eta$. When there are N valley directions, the total thermal loss is summed over all valley directions $\bar{L}_f = N\bar{\ell}_f = NC\eta$.

3.3 Experiments

GPT-2 Experiment setup: We pre-train a GPT-2-small model (based on NanoGPT [11]) on OpenWebText. We use 8 V100 GPUs, choose block size 1024, batch size 480 blocks. We use the Adam Optimizer, with warmup-stable-decay learning rate schedules as shown in Figure 2 (a). We always use 2000-step linear warmup from 0 to 6×10^{-4} . The stable phase has a learning rate η . The decay phase starts from η and cosine decays to η_{\min} . The total number of training steps is 10k.

We have shown in the toy model that the averaged thermal loss $\bar{\ell}_f$ is linear to the learning rate η (assuming thermal equilibrium). We now show that this relation holds for large language models.

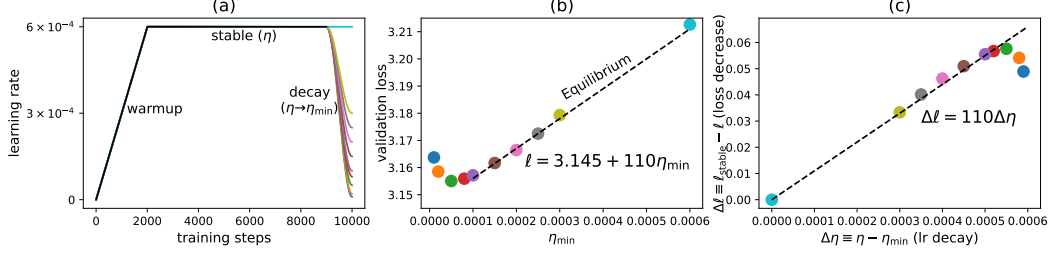


Figure 2: (a) LLM pretraining usually uses the WSD (warmup-stable-decay) learning rate schedule. η_{\min} is the final learning rate. (b) validation loss is a linear function of η_{\min} for large η_{\min} . (c) $\Delta \ell$ is a linear function of $\Delta \eta$ for small $\Delta \eta$.

We have 2k-step warmup, 7k-step stable ($\eta = 6 \times 10^{-4}$) and 1k-step cosine decay to η_{\min} , which is swept. In Figure 2 (b), we show that the final validation loss is linear to η_{\min} for large η_{\min} . Since the decay phase is short, we can assume that ℓ_s does not vary much across decay schedules. As a result, ℓ is representative of ℓ_f , and we measure that $\ell = 3.145 + 110\eta_{\min}$. Comparing to the theoretical thermal loss $\bar{L}_f = (\sqrt{\frac{\pi}{32}} N \sigma_g) \eta$ (for SignGD³), we have $N = \frac{110}{\sigma_g} \sqrt{\frac{32}{\pi}} \approx 5 \times 10^6 = 5\text{M}$ where $\sigma_g \approx 7 \times 10^{-5}$ is estimated using batches. Note that GPT-2 small has 124M parameters, and 5M valley directions are only 4% of its total parameters. This is expected: most directions of an over-parameterized model are flat (“river”), and only a small portion of directions are sharp (“valley”). However, too small η_{\min} leads to higher loss, and the lowest loss occurs around $\eta_{\min} \approx 5 \times 10^{-5} > 0$. The non-monotonic behavior at small η_{\min} is due to the breakdown of thermal equilibrium when η decays too fast, as we will elaborate on in the next section.

4 Valley Dynamics in Annealing (Decay phase)

In the last section, we have shown that the width of the steady distribution σ depends on the learning rate η as $\sigma \propto \sqrt{\eta}$. In the decay phase when η decays, σ will decay as a result. If η decays slowly enough, we could expect that $\sigma \propto \sqrt{\eta}$ always holds because of quasi-static thermal equilibrium. But is this optimal? Probably no. But the other extreme – too fast η decay – is also not optimal. This dilemma is because η plays two roles: (1) η is temperature that controls the Gaussian noise (we want η to be small); (2) η is step size that controls the time scale (we want η to be large). There exists an “optimal” η decay schedule in the sense that ℓ_f is reduced as quickly as possible.

4.1 Toy model: optimal learning rate decay

Now we consider a learning rate decay schedule, i.e., a sequence $\eta_0 \geq \eta_1 \geq \eta_2 \geq \dots \geq \eta_T$. the dynamics of SGD (Eq. (1)) now becomes $x_{t+1} = x_t - \eta_t(ax_t + \sigma_g \dot{W})$. Since the equation is linear, if $p(x_0)$ starts off as a Gaussian distribution, $p(x_t)$ remains a Gaussian distribution (with time-varying Gaussian width, denoted as σ_t) for all $t \geq 0$. Assuming that at $t = 0$, $p(x_0)$ is in thermal equilibrium with learning rate η whose Gaussian width is σ_0 . Gaussian widths σ_t obeys the following recursive relation $\sigma_{t+1}^2 = (1 - \eta_t a)^2 \sigma_t^2 + (\eta_t \sigma_g)^2 = (a^2 \sigma_t^2 + \sigma_g^2) \eta_t^2 - 2a \sigma_t^2 \eta_t + \sigma_t^2$, which is a quadratic function of η_t . We want to choose η_t such that σ_{t+1} is minimized $\eta_t = \frac{a \sigma_t^2}{a^2 \sigma_t^2 + \sigma_g^2}$, $\sigma_{t+1}^2 = \frac{\sigma_g^2 \sigma_t^2}{\sigma_t^2 + a^2 \sigma_g^2}$. By inverting the second equation, we get $\frac{1}{\sigma_{t+1}^2} = \frac{1}{\sigma_t^2} + \frac{a^2}{\sigma_g^2}$, which means that $\{\frac{1}{\sigma_t^2}\}$ forms an arithmetic sequence. It is clear that $\frac{1}{\sigma_t^2} = \frac{1}{\sigma_0^2} + \frac{a^2 t}{\sigma_g^2}$. Correspondingly,

$$\eta_t = \frac{\frac{1}{a}}{1 + \frac{\sigma_g^2}{a^2} (\frac{1}{\sigma_0^2} + \frac{a^2 t}{\sigma_g^2})} \approx \frac{\frac{\eta}{2}}{1 + \frac{t}{t_h}} \quad (t_h \equiv \frac{2}{a\eta}), \quad (3)$$

³SignGD is a special cases of Adam when $(\beta_1, \beta_2) = (0, 0)$.

where t_h is the time needed to decrease η_t by half, hence representing the characteristic time scale of learning rate decay. Recall that $\bar{\ell}_f = C\eta$, so $\bar{\ell}_{f,t} = C\eta_t$ has the same decay form as η_t . We make a few remarks about the optimal schedule.

Remark 1: Asymptotic behavior As $t \rightarrow \infty$, learning rate $\eta_t \propto t^{-1}$, standard deviation $\sigma_t \propto t^{-1/2}$, loss $\bar{\ell}_{f,t} \propto t^{-1}$, $I_t = 1/\sigma_t^2 \propto t$ (I_t is the fisher-information of the Gaussian mean).

Remark 2: non-continuity at $t = 0$. An interesting observation is that $\eta_0 \neq \eta$ but rather $\eta_0 \approx \frac{\eta}{2}$. This makes sense because neither $\eta_0 = \eta$ nor $\eta_0 = 0$ leads to a loss decrease. The non-continuity goes against the common wisdom of continuous learning rate schedules.

Remark 3: decay time is bounded when $\eta \rightarrow \infty$. Suppose we want to decrease the learning rate from the stable value η to the final value η_{\min} . The optimal decay takes time $T_d = \frac{2}{a\eta_{\min}}(1 - \frac{\eta_{\min}}{\eta}) < \frac{2}{a\eta_{\min}}$. Notice that the decay time T_d has an upper bound $T^* \equiv \frac{2}{a\eta_{\min}}$ independent of η , meaning that as long as one uses enough time (more than T^*) for decay, one can in principle arbitrarily increase the stable learning rate η without worrying about extra losses induced by insufficient decay.

Remark 4: The optimal schedule for SignGD is the same except for t_h . The optimal η decay schedule for SignGD has the same functional form $\eta_t = \frac{\eta}{1 + \frac{t}{t_h}}$ although with a slightly different $t_h \equiv \sqrt{2\pi} \frac{\sigma_g}{a\eta}$. Detailed derivations are deferred to Appendix C.

4.2 Thermodynamics: Fourier’s conduction law, Second law of thermodynamics

In the analysis above, the learning rate η has two roles: temperature and time scale, making it complicated to make a correspondence with thermodynamics. We now study a simplified two-temperature setting where the analogy becomes clearer. Suppose the fast parameter x reaches its thermal equilibrium with learning rate η_A . At $t = 0$, we suddenly switch the learning rate to $\eta_B < \eta_A$. How does thermal width σ_t and thermal loss $\bar{\ell}_{f,t} \equiv \frac{1}{2}a\sigma_t^2$ evolve in time?

Recalling basic facts for SGD at steady distribution: in the flat limit $a \ll \frac{2}{\eta}$, we have $\sigma \approx \sqrt{\frac{\eta}{2a}}\sigma_g$, and $\bar{\ell}_f = \frac{1}{2}a\sigma^2 = \frac{1}{4}\eta\sigma_g^2$. At $t = 0$, we have $\sigma_0 = \sqrt{\frac{\eta_A}{2a}}\sigma_g$. $\{\sigma_t\}$ evolves as follows $\sigma_{t+1}^2 = (a^2\sigma_t^2 + \sigma_g^2)\eta_B^2 - 2a\sigma_t^2\eta_B + \sigma_t^2$, where the crossed-out term can be ignored due to the flat limit. We are interested in how $\bar{\ell}_{f,t}$ evolves:

$$\bar{\ell}_{f,t+1} - \bar{\ell}_{f,t} = \frac{1}{2}a(\sigma_{t+1}^2 - \sigma_t^2) = \frac{1}{2}a\eta_B(\sigma_g^2\eta_B - 2a\sigma_t^2) = -2a\eta_B(\bar{\ell}_{f,t} - \bar{\ell}_{eq}(\eta_B)) \quad (4)$$

where $\bar{\ell}_{eq}(\eta_B) \equiv \frac{\eta_B\sigma_g^2}{4}$ is the averaged thermal loss given $\eta = \eta_B$ in equilibrium. This equation is similar to **Fourier’s law** in thermal conduction $Q = k(T_A - T_B)$: when a hot object (temperature T_A) touches a cooler surface (temperature T_B), the power of thermal conduction Q is proportional to their temperature difference, leading to an exponential convergence. Similarly, Eq. (4) bears an exponential decay solution $\bar{\ell}_{f,t} = \bar{\ell}_f(\eta_B) + (\bar{\ell}_f(\eta_A) - \bar{\ell}_f(\eta_B))\exp(-2a\eta_B t) \geq \bar{\ell}_f(\eta_B)$. The inequality $\bar{\ell}_{f,t} \geq \bar{\ell}_f(\eta_B)$ is related to the **second law of thermodynamics**. Simply put, when a hot object is in contact with a cool thermostat, the hot object cannot be cooled down to a temperature lower than the temperature of the cool thermostat (without extra work). As a sanity check, we show in Appendix D that: by making Eq. (4) continuous, we can also obtain the $1/t$ schedule as in Eq. (3).

4.3 Experiments

Toy experiments We test the optimality of the schedule we derived in Eq. (3). We choose the loss landscape $\ell = \sum_{i=1}^n \frac{1}{2}a\theta_i^2$ ($a = 2, n = 10000$) and initialize $\theta_i \sim \mathcal{N}(0, 1)$. We run SGD with $\eta = 0.1$ (manually injecting Gaussian gradient noise $\sigma_g = 0.1$) for 10000 steps to reach its steady distribution. We then perform a learning rate decay schedule $\eta_t = \frac{b\eta}{1 + \frac{t}{t_h}}$ with various combinations of (b, t_h) . We plot the final losses in Figure 3 (a). Our theoretical result implies that $b = 1/2$ and $t_h = \frac{2}{a\eta} = 10$ give the best result (the lowest loss), which is verified by the phase diagram. We also plot the two slices along t_h and b in (b) and (c), showing that the final loss is more sensitive to t_h than b . To simulate anisotropy, we also experiment with an anisotropic loss $\ell = \sum_{i=1}^n \frac{1}{2}a_i\theta_i^2$ ($a_i = 10^{-2+4i/n}, n = 10000$), $\theta_i \sim \mathcal{N}(0, 1)$. We apply the learning rate decay

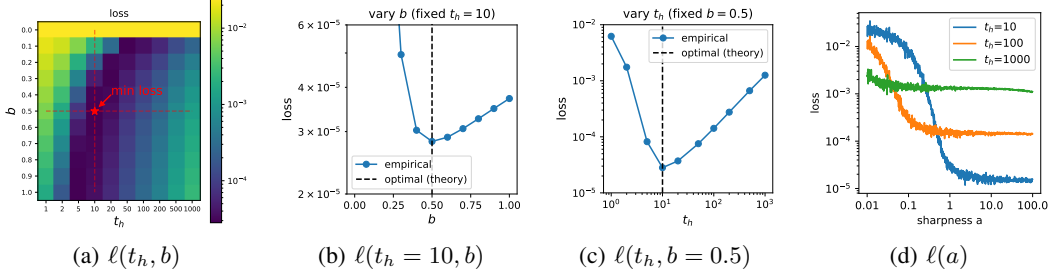


Figure 3: Annealing toy examples. (a)(b)(c) Isotropic loss $\ell = \sum_{i=1}^n \frac{1}{2} a \theta_i^2$ ($a = 2, n = 10000$). The final loss obtained by applying the decay schedule $\eta_t = b\eta_0/(1 + t/t_h)$. The theoretical minimum $(b, t_h) = (0.5, 10)$ (marked as a star) agrees with numerical results. (d) Anisotropic loss $\ell = \sum_{i=1}^n \frac{1}{2} a_i \theta_i^2$ ($a_i = 10^{-2+4i/n}, n = 10000$). We set $b = 0.5$ and try $t_h = 10, 100, 1000$. Small sharpness is slower to converge than large sharpness.

schedule $\eta_t = \frac{b\eta}{1 + \frac{t}{t_h}}$ with $b = 0.5$ and $t_h = 10, 100, 1000$, as shown in Figure 3 (d). The loss is roughly constant for large a , because of the equipartition theorem in Section 3. However, small sharpness directions have higher losses because their optimal decay time $t_h \propto 1/a$ is larger, hence requiring longer time to converge.

Implications for LLM The above results can provide insights on recent observations: (1) it is observed that the 1-sqrt decay [10] or an optimized decay [12] are better than the linear or cosine decay. These better decay schedules are aligned with our derived $1/t$ schedule. (2) Decaying to zero is sub-optimal, as observed in [13]. In fact, the optimal schedule implies that it should take infinite time to reach $\eta_{\min} = 0$.

5 River Dynamics

So far, we have been studying the fast dynamics of x , assuming the slow variable y is fixed. This section will study how the slow dynamics can be influenced by the fast dynamics via entropic forces.

5.1 Toy model: entropic forces

Recall that our 2D toy river-valley landscape is $l(x, y) = \frac{1}{2}a(y)x^2 + c(y) \equiv \ell_f + \ell_s$ where ℓ_f and ℓ_s are fast loss and slow loss, respectively. The sharpness of the valley is controlled by $a(y)$, while $c(y)$ controls the bottom of the valley. Optimizer dynamics can be viewed as fast dynamics along x and slow dynamics along y . Given a fixed y , the steady distribution for x is $p_y(x) = \frac{1}{\sqrt{2\pi\sigma(y)}} e^{-\frac{x^2}{2\sigma(y)^2}}$.

We have shown that $\sigma(y) = d(\eta, \sigma_g)/\sqrt{a(y)}$, where $d = \sqrt{\eta/2}\sigma_g$ for SGD and $c = (\pi/8)^{1/4}\sqrt{\sigma_g\eta}$ for SignGD. The entropic force is defined as the average gradient of ℓ_f along y :

$$F_{\text{ent}} = -\overline{g_y} = -\frac{1}{2}a'(y)\overline{x^2} = -\frac{1}{2}a'(y)\sigma(y)^2 = -\frac{d^2(\eta, \sigma_g)}{2} \frac{a'(y)}{a(y)} \quad (5)$$

The minus sign means that the entropic force points towards the direction of *decreasing* sharpness. The negative gradient of the valley bottom ℓ_s is $F_{\text{btm}} = -c'(y)$. The total “force” is $F = F_{\text{ent}} + F_{\text{btm}}$.

Defining entropy Notice that $a'(y)/a(y)$ in F_{ent} can be written into a more compact form $(\log a(y))'$. We can define $S \equiv -\frac{d^2(\eta, \sigma_g)}{2} \log a(x)$, and then $F_{\text{ent}} = \nabla S$.

Entropic trapping is defined as $F_{\text{btm}} \cdot F \leq 0$. Entropic trapping happens when the entropic forces prevent the optimizer from descending the river despite the fact that the optimizer is able to “see” the correct direction. Concrete examples are discussed in Appendix E.

5.2 Thermodynamics: entropy, the third law of thermodynamics

We want to justify a bit more from the thermodynamics perspective why $S(x) \propto -\frac{1}{2}\log a(x)$ can be interpreted as entropy. In physics, entropy is defined as $S_{\text{phy}} = -\sum_i p_i \log p_i$ for discrete systems or

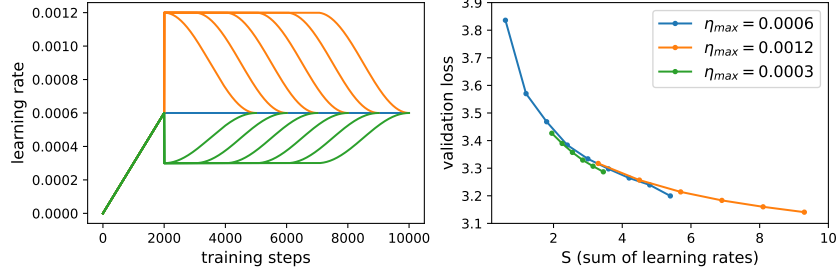


Figure 4: Test the existence of entropic forces in LLMs. Left: Various learning rate schedules with different stable $\eta_{\max} = 0.0003, 0.0006, 0.0012$ and the same $\eta_{\min} = 0.0006$. Right: Plot validation losses against learning rate sums. Curves for different η roughly align, suggesting slightly negative entropic forces, corresponding to a slightly narrowing valley along the river.

$S_{\text{phy}} = - \int dx p(x) \log p(x)$ for continuous systems. A Gaussian distribution $p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$ hence has entropy $S_{\text{phy}} = \frac{1}{2} \log(2\pi\sigma^2) + \frac{1}{2}$. Inserting $\sigma = d/\sqrt{a}$ gives $S_{\text{phy}} = -\frac{1}{2} \log a + \frac{1}{2} \log(2\pi d) + \frac{1}{2}$, which scales with a as $S \propto -\frac{1}{2} \log a$. This is also analogous to the **Boltzmann entropy equation** $S = k_b \log W$, where k_b is the Boltzmann constant, and W is the number of microstates (analogous to σ). The Boltzmann entropy equation is a foundation to **the third law of thermodynamics**.

5.3 Experiments

LLM experiments We want to determine to what extent the concerning entropic trapping phenomenon occurs in LLM training. Technically, the entropic force is hard to compute for large models: by definition, entropic forces are third-order derivatives, since they are gradients of sharpness (second-order derivatives). However, we may probe the existence of entropic forces via loss curves alignment against learning rate sums. In the gradient flow limit ($\eta \rightarrow 0$), the slow dynamics is only governed by the learning rate sum [12], i.e., we should expect the following two schedules to give the same final state and loss: (i) learning rate η with A steps and (ii) learning rate 2η with $A/2$ steps. However, for a finite η , the two schedules may not align due to the existence of entropic forces, so their misalignment is an indicator of the magnitude and direction of entropic forces. Since ℓ_s is not directly measurable, we can try to control ℓ_f the same (requiring η_{\min} to be the same, according to Section 4) and measure ℓ as a surrogate of ℓ_s .

We test a bunch of schedules with different stable learning rates $\eta = 0.0003, 0.0006, 0.0012$ (Figure 4 left). The stable phase may last for $1000i$ steps ($i = 0, 1, 2, 3, 4, 5$), and the decay phase lasts for 3k steps with cosine transition to the final learning rate $\eta_{\min} = 0.0006^4$. For each η , we plot its validation loss against the η sum, swept by varying the duration of the stable phase. Figure 4 (b) shows that curves of different η align reasonably well with each other, although smaller η seem to produce slightly lower losses given the same η sum, similar to the observations in [12]. This result implies that LLM has a slightly narrowing valley structure on average (correspondingly, the entropic force is slightly negative). However, our training is in the early stage due to computational constraints. It would be interesting to study whether entropic forces represent a more significant limit to the training of LLMs when more training steps are taken (where valleys may become sharper and entropic forces are larger).

6 Summary of Findings

The role of learning rates We have learned that the learning rate η has three roles in controlling training dynamics. (1) **Gaussian width**: η acts as temperature, controlling the Gaussian width along the fast direction. (2) **Entropic force magnitude**: the Gaussian width, combined with valley sharpness, jointly controls the entropic force. (3) **Time scale**: controls the step size.

⁴Although η_{\min} has the subscript “min”, it does not necessarily mean it is the minimum learning rate in the last phase. The last phase can either be decay ($\eta = 0.0012$), stable ($\eta = 0.0006$) or growth ($\eta = 0.0003$). The point is that all the schedules have the same final learning rate, denoted as η_{\min} .

What determines the final loss? In summary, our results show that the final loss largely depends on learning rate sum D (controlling ℓ_s , Section 5) and η_{\min} (controlling ℓ_f , Section 3), but there are also small correction terms due to the entropic force Δ_{entropic} (Section 5) and due to insufficient annealing Δ_{anneal} (Section 4).

$$\ell_{\text{final}} = \ell(D, \eta_{\min}) + \Delta_{\text{entropic}} + \Delta_{\text{anneal}}. \quad (6)$$

Empirically, for the early training stages of GPT2, we found $\Delta_{\text{entropic}} \sim 0$ holds roughly true, and $\Delta_{\text{anneal}} \sim 0$ when the decay phase is no smaller than 3k steps. If we assume Δ_{entropic} and Δ_{anneal} can be ignored, the only way to reduce loss is by reducing $\ell(D, \eta_{\min})$, which involves reducing η_{\min} and/or increasing D . Increasing D can be achieved by using a larger step size in the stable phase, verified by experiments in Appendix G.

7 Related Works

Physics of optimization Stochastic gradient descent with finite step sizes has different dynamics from gradient flow. Finite learning rate η can induce implicit regularization $\frac{\eta}{4} \|\nabla \ell\|^2$ [14], and stochastic gradients also have various implicit biases [15, 16, 17], guiding the optimization towards flatter minima [18, 9], large eigendirections [19], maximizes margin [20, 21], and redundant neurons/directions [16, 22]. Besides these quite general phenomena, research has also been carried out to understand the loss landscapes of neural networks, especially in the over-parametrized regime. Large models are shown to have mode connectivity [23, 24], which is also related to the recently discovered river-valley landscape of LLMs [5, 7].

Learning rate schedules for LLMs are diverse: cosine decay [25], cyclic [26], Noam [27] and weight-stable-decay (WSD) [8]. Recent research has started to show the advantages of the WSD schedule [5, 12, 10] and concerns about designing better decay schedules, e.g., the 1-sqrt schedule proposed in [10] and an optimized schedule in [12]. Our analysis provides yet another theoretical evidence for the use of the WSD schedule and analytically derives an optimal decay schedule which decays as $1/t$ (under the isotropic assumption).

Thermodynamics and learning Although we are the first to establish a mapping between thermodynamics and LLM training dynamics, thermodynamics has long inspired and has connections to machine learning: optimization as a thermodynamic process [1], statistical mechanics of learning [4], information bottleneck [28], entropy gradient descent [29], Boltzman machine [30], Hopfield networks [31], diffusion models [32, 33, 34], thermodynamic interpretations of networks [2].

8 Conclusions

We propose a toy model of a river-valley loss landscape and analyze the training dynamics under SGD and SignGD. The fast-slow separation enables us to treat valley and river directions independently, yielding analytically tractable results: thermal equilibrium and annealing for the fast dynamics, and drift for the slow dynamics. These analytical solutions bear qualitative—and in some cases quantitative—analogy to classical thermodynamic concepts and laws. Crucially, they are relevant to large language model (LLM) training, as recent work has shown that LLM loss landscapes exhibit river-valley structure. This duality between optimization and thermodynamics offers a novel perspective for understanding and evaluating modern optimizers. While we leave it for future work, we include a proof-of-concept analysis in Appendix F, where we analyze the recently proposed FOCUS optimizer—characterized by self-attracting forces [7]—through the lens of our theory.

Limitations. Many of the derivations in this paper adopt the physicist’s style of reasoning—emphasizing intuition, simplification, and tractable approximations—which may not satisfy the standards of mathematical rigor expected by theorists. For instance, the Gaussian approximation of steady states is not necessary for many results (only variances matter). In deriving the optimal learning rate decay schedule, we assume either uniform sharpness or a one-dimensional landscape; we treat the river as straight, though it is likely curved in practice; and we ignore momentum and weight decay for simplicity. Despite these simplifications, our analysis yields non-trivial, testable insights into LLM training dynamics. Natural extensions of this work include relaxing these assumptions, validating predictions at larger scales, and generalizing the framework. Although our focus is on transformer-based LLMs, the underlying physics-inspired principles may extend to other model architectures.

Acknowledgment Z.L. and M.T. are supported by IAIFI through NSF grant PHY-2019786. Z. L. is also supported by the Google PhD fellowship.

References

- [1] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688. Citeseer, 2011.
- [2] Shams Mehdi and Pratyush Tiwary. Thermodynamics-inspired explanations of artificial intelligence. *Nature Communications*, 15(1):7859, 2024.
- [3] Andreas Engel. *Statistical mechanics of learning*. Cambridge University Press, 2001.
- [4] Yasaman Bahri, Jonathan Kadmon, Jeffrey Pennington, Sam S Schoenholz, Jascha Sohl-Dickstein, and Surya Ganguli. Statistical mechanics of deep learning. *Annual review of condensed matter physics*, 11(1):501–528, 2020.
- [5] Kaiyue Wen, Zhiyuan Li, Jason Wang, David Hall, Percy Liang, and Tengyu Ma. Understanding warmup-stable-decay learning rates: A river valley loss landscape perspective. *arXiv preprint arXiv:2410.05192*, 2024.
- [6] Mingwei Wei and David J Schwab. How noise affects the hessian spectrum in overparameterized neural networks. *arXiv preprint arXiv:1910.00195*, 2019.
- [7] Yizhou Liu, Ziming Liu, and Jeff Gore. Focus: First order concentrated updating scheme. *arXiv preprint arXiv:2501.12243*, 2025.
- [8] Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, et al. Minicpm: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*, 2024.
- [9] Jeremy M Cohen, Simran Kaur, Yuanzhi Li, J Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. *arXiv preprint arXiv:2103.00065*, 2021.
- [10] Alex Hägele, Elie Bakouch, Atli Kosson, Leandro Von Werra, Martin Jaggi, et al. Scaling laws and compute-optimal training beyond fixed training durations. *Advances in Neural Information Processing Systems*, 37:76232–76264, 2024.
- [11] Andrej Karpathy. NanoGPT. <https://github.com/karpathy/nanoGPT>, 2022.
- [12] Kairong Luo, Haodong Wen, Shengding Hu, Zhenbo Sun, Maosong Sun, Zhiyuan Liu, Kaifeng Lyu, and Wenguang Chen. A multi-power law for loss curve prediction across learning rate schedules. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [13] Jordan Keller. modded-nanogpt. <https://github.com/KellerJordan/modded-nanogpt>, 2024. GitHub repository.
- [14] David GT Barrett and Benoit Dherin. Implicit gradient regularization. *arXiv preprint arXiv:2009.11162*, 2020.
- [15] Daniel Kunin, Javier Sagastuy-Brena, Lauren Gillespie, Eshed Margalit, Hidenori Tanaka, Surya Ganguli, and Daniel LK Yamins. The limiting dynamics of sgd: Modified loss, phase-space oscillations, and anomalous diffusion. *Neural Computation*, 36(1):151–174, 2023.
- [16] Feng Chen, Daniel Kunin, Atsushi Yamamura, and Surya Ganguli. Stochastic collapse: How gradient noise attracts sgd dynamics towards simpler subnetworks. *Advances in Neural Information Processing Systems*, 36:35027–35063, 2023.
- [17] Mandt Stephan, Matthew D Hoffman, David M Blei, et al. Stochastic gradient descent as approximate bayesian inference. *Journal of Machine Learning Research*, 18(134):1–35, 2017.

- [18] Zeke Xie, Issei Sato, and Masashi Sugiyama. A diffusion theory for deep learning dynamics: Stochastic gradient descent exponentially favors flat minima. *arXiv preprint arXiv:2002.03495*, 2020.
- [19] Jingfeng Wu, Difan Zou, Vladimir Braverman, and Quanquan Gu. Direction matters: On the implicit bias of stochastic gradient descent with moderate learning rate. *arXiv preprint arXiv:2011.02538*, 2020.
- [20] Kaifeng Lyu and Jian Li. Gradient descent maximizes the margin of homogeneous neural networks. *arXiv preprint arXiv:1906.05890*, 2019.
- [21] Bohan Wang, Qi Meng, Wei Chen, and Tie-Yan Liu. The implicit bias for adaptive optimization algorithms on homogeneous neural networks. In *International Conference on Machine Learning*, pages 10849–10858. PMLR, 2021.
- [22] Zhi-Qin John Xu, Yaoyu Zhang, and Zhangchen Zhou. An overview of condensation phenomenon in deep learning. *arXiv preprint arXiv:2504.09484*, 2025.
- [23] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. *Advances in neural information processing systems*, 31, 2018.
- [24] Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pages 3259–3269. PMLR, 2020.
- [25] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.
- [26] Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pages 464–472. IEEE, 2017.
- [27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [28] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.
- [29] Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-sgd: Biasing gradient descent into wide valleys. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124018, 2019.
- [30] David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowski. A learning algorithm for boltzmann machines. *Cognitive science*, 9(1):147–169, 1985.
- [31] Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Thomas Adler, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandve, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.
- [32] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.
- [33] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.
- [34] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Appendix

A SGD converges to Gaussian steady distribution

Suppose the initial point is x_0 at $t = 0$. The distribution is a delta function, effectively a Gaussian with mean $\mu_0 = x_0$ and $\sigma_0 = 0$. Due to the linearity of Eq. (1), a Gaussian distribution at step t evolved to become a Gaussian distribution at step $t + 1$, although with different means μ_t and standard deviations σ_t . Their recursive relations are ($t = 1, 2, 3, \dots$)

$$\begin{aligned}\mu_t &= (1 - \eta a)\mu_{t-1}, \\ \sigma_t^2 &= (1 - \eta a)^2 \sigma_{t-1}^2 + \eta^2 \sigma_g^2,\end{aligned}\tag{7}$$

which have solutions

$$\begin{aligned}\mu_t &= (1 - \eta a)^t x_0, \\ \sigma_t^2 &= (1 - (1 - \eta a)^{2t}) \sigma^2, \quad \sigma \equiv \frac{\sigma_g}{\sqrt{a(\frac{2}{\eta} - a)}},\end{aligned}\tag{8}$$

where σ is the steady-state standard deviation. Regardless of x_0 , $\mu_t \rightarrow 0$ and $\sigma_t \rightarrow \sigma$ as $t \rightarrow \infty$. The time scale of convergence is $t_c = -1/\log(1 - \eta a)$. In the flat limit $a\eta \ll 1$, $t_c \approx 1/(a\eta)$.

B Derivations for SGD

B.1 Fixed learning rate

Suppose a 1D loss function $l(x) = \frac{1}{2}ax^2$ where a is the second order derivative of the quadratic function. An SGD optimizer with learning rate η and gradient noise σ_g obeys the following dynamics:

$$x_{t+1} = x_t - \eta(ax_t + \sigma_g \dot{W}_t)\tag{9}$$

The equilibrium distribution is the Gaussian distribution $p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$. In equilibrium, we should have $\text{Var}(x_{t+1}) = \text{Var}(x_t)$, i.e.,

$$\sigma^2 = (1 - \eta a)^2 \sigma^2 + \sigma_g^2,\tag{10}$$

which gives

$$\sigma = \frac{\sigma_g}{\sqrt{a(\frac{2}{\eta} - a)}}.\tag{11}$$

This formula is only valid when $0 < a < \frac{2}{\eta}$. When $a \rightarrow 0$ (i.e., flat), finite σ_g can induce infinite σ (i.e., steady distribution does not exist). When $a \rightarrow \frac{2}{\eta}$, learning rate reaches the so-called “edge of stability”. In particular, when $\eta > \frac{2}{a}$, the $\{x_t\}$ sequence diverges when $\sigma_g \rightarrow 0$.

Eq. 11 is tested empirically in Figure 5. A somewhat unexpected feature of Eq. (11) is that $\sigma \rightarrow \infty$ when $a \rightarrow 0$ although we were previously viewing flat directions as good and viewing sharp directions as evil. Eq. (11) suggests that flat directions are as evil as sharp directions.

Equipartition theorem The original idea of equipartition (in classical statistical mechanics) is that, in thermal equilibrium, energy is shared equally among all of its degrees of freedom. Specifically, each degree of freedom would contribute to energy $\frac{1}{2}k_B T$ (k_B : Boltzmann constant, T : temperature) regardless of underlying details. We show that the loss incurred due to gradient noise is also (approximately) independent of sharpness a : $\langle l \rangle = \frac{1}{2}a\langle x^2 \rangle = \frac{1}{2}a\sigma^2$. When $a\eta \ll 2$, $\sigma \approx \frac{1}{\sqrt{2a}}\sigma_g\sqrt{\eta}$. So $\langle l \rangle = \frac{1}{2}a\sigma^2 \approx \frac{1}{2}a(\frac{1}{\sqrt{2a}}\sigma_g\sqrt{\eta})^2 = \frac{1}{4}\sigma_g^2\eta$. Ignoring constants, the effective temperature is $T_{\text{eff}} \propto \sigma_g^2\eta$. The gradient noise scales with batch size B as $\sigma_g \propto \frac{1}{\sqrt{B}}$. To reduce temperature, we can increase the batch size or decrease the learning rate. This has an interesting implication: during the training of a neural network, there might exist many such equilibrium directions. No matter how sharpness these directions are, they contribute equally to the total loss. Suppose there are N such directions, the total loss incurred by gradient noise is $l_g = N\langle l \rangle = \frac{1}{4}N\sigma_g^2\eta$.

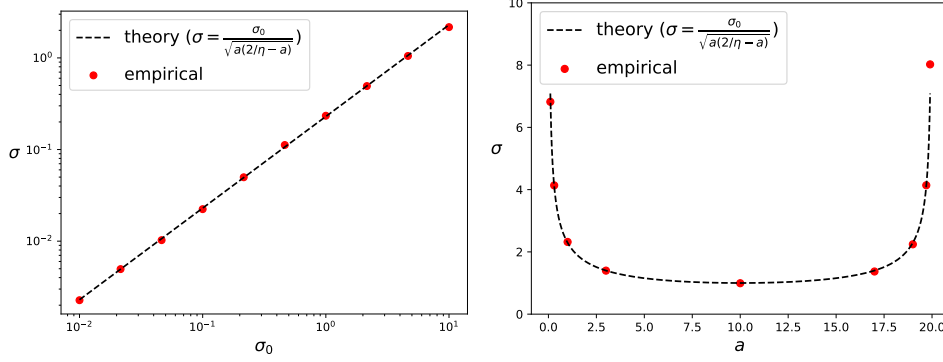


Figure 5: Dependence of σ on gradient noise σ_g and sharpness a .

B.2 Learning rate decay

Now we consider a learning rate schedule, i.e., a sequence of $\{\eta_t\}_{t=0}^T$. Eq. (1) now becomes:

$$x_{t+1} = x_t - \eta_t(ax_t + \sigma_g \dot{W}_t) \quad (12)$$

Since the equation is linear, if $p(x_0)$ starts off as a Gaussian distribution, $p(x_t)$ remain a Gaussian distribution (with time-varying Gaussian width, denoted as σ_t) forever. Assuming that at $t = 0$, $p(x_0)$ is already in thermal equilibrium whose Gaussian width is given by Eq. (11), i.e., the initial condition is $\sigma_0 = \sigma$. Gaussian widths σ_t obeys the following recursive relation:

$$\sigma_{t+1}^2 = (1 - \eta_t a)^2 \sigma_t^2 + (\eta_t \sigma_g)^2 = (a^2 \sigma_t^2 + \sigma_g^2) \eta_t^2 - 2a \sigma_t^2 \eta_t + \sigma_t^2, \quad (13)$$

which is a quadratic function of η_t . We want to choose η_t such that σ_{t+1} is minimized:

$$\eta_t = \frac{a \sigma_t^2}{a^2 \sigma_t^2 + \sigma_g^2}, \quad \sigma_{t+1}^2 = \frac{\sigma_g^2 \sigma_t^2}{\sigma_g^2 + a^2 \sigma_t^2}. \quad (14)$$

By inverting the second equation, we get

$$\frac{1}{\sigma_{t+1}^2} = \frac{1}{\sigma_t^2} + \frac{a^2}{\sigma_g^2}, \quad (15)$$

which means that $\{\frac{1}{\sigma_t^2}\}$ forms an arithmetic sequence. It is clear that $\frac{1}{\sigma_t^2} = \frac{1}{\sigma_0^2} + \frac{a^2 t}{\sigma_g^2}$. Correspondingly,

$$\eta_t = \frac{\frac{1}{a}}{1 + \frac{\sigma_g^2}{a^2 \sigma_t^2}} = \frac{\frac{1}{a}}{1 + \frac{\sigma_g^2}{a^2} \left(\frac{1}{\sigma_0^2} + \frac{a^2 t}{\sigma_g^2} \right)} \approx \frac{\frac{1}{a}}{\frac{2}{a\eta} + t} \propto \frac{1}{t + t_h}, \quad (t_h \equiv \frac{2}{a\eta}) \quad (16)$$

whose asymptotic behavior is $\eta_t \propto \frac{1}{t}$. t_h is the time needed to decrease η_t by half, hence representing the characteristic time scale of learning rate decay. Another interesting observation is that $\eta_0 \neq \eta$. In fact, when we assume $a\eta \ll 2$, $\eta_0 \approx \frac{\eta}{2}$. This makes sense: since σ_1 is an quadratic function of η_0 , and $\sigma_1 = \sigma_0$ for both $\eta_t = \eta$ (continue thermal equilibrium) and $\eta_0 = 0$ (freeze), the best η_1 must be at the middle point of η and 0, which is $\frac{\eta}{2}$. This suggests that the standard learning rate decay schedule (which is continuous at $t = 0$) may be suboptimal.

$\eta_t = \frac{\eta t_h}{t + t_h}$ ($t_h = \frac{2}{a\eta}$), to take it from η to η_m , it takes time $T_d = \frac{2}{a\eta_m} (1 - \frac{\eta_m}{\eta}) < \frac{2}{a\eta_m}$. So a larger η in the stable phase does not increase T_d significantly.

C Derivations for SignGD

C.1 Fixed learning rate

The optimization dynamics of SignGD is

$$x_{t+1} = x_t - \eta \text{sign}(ax_t + \sigma_g \dot{W}). \quad (17)$$

Define $\zeta(x) \equiv \int_{-ax/\sigma_g}^{\frac{1}{\sqrt{2\pi}}} \exp(-\frac{y^2}{2}) dy$. Then given a fixed x , $ax + \sigma_g \dot{W}$ is positive with probability $\zeta(x)$, or negative with probability $1 - \zeta(x)$, i.e.,

$$x_{t+1} = \begin{cases} x_t - \eta & \text{probability } \zeta(x) \\ x_t + \eta & \text{probability } 1 - \zeta(x) \end{cases} \quad (18)$$

We assume that x_t has a Gaussian steady distribution $x_t \sim \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{x^2}{2\sigma^2})$ ⁵. We can set up an equation by leveraging $\text{Var}(x_{t+1}) = \text{Var}(x_t)$, i.e.,

$$\int_{-\infty}^{\infty} dx p(x) x^2 = \int_{-\infty}^{\infty} dx p(x) (\zeta(x)(x - \eta)^2 + (1 - \zeta(x))(x + \eta)^2), \quad (19)$$

which can simplify to (with some derivations)

$$\int_{-\infty}^{\infty} dx p(x) \zeta(x) x - \frac{\eta}{4} = 0. \quad (20)$$

The first integral

$$\begin{aligned} \int_{-\infty}^{\infty} dx p(x) \zeta(x) x &= \int_{-\infty}^{\infty} dx \int_{-ax/\sigma_g}^{\infty} dy \frac{x}{\sqrt{2\pi}\sigma} \exp(-\frac{x^2}{2\sigma^2}) \frac{1}{\sqrt{2\pi}} \exp(-\frac{y^2}{2}) \\ &= \int_{-\infty}^{\infty} dy \int_{-\sigma_g y/a}^{\infty} dx \frac{x}{\sqrt{2\pi}\sigma} \exp(-\frac{x^2}{2\sigma^2}) \frac{1}{\sqrt{2\pi}} \exp(-\frac{y^2}{2}) \\ &= \int_{-\infty}^{\infty} dy \exp(-\frac{y^2}{2}) \int_{-\sigma_g y/a}^{\infty} dx \frac{x}{2\pi\sigma} \exp(-\frac{x^2}{2\sigma^2}) \\ &= \int_{-\infty}^{\infty} dy \exp(-\frac{y^2}{2}) \frac{\sigma}{2\pi} \exp(-\frac{1}{2}(\frac{\sigma_g y}{a\sigma})^2) \\ &= \frac{\sigma}{\sqrt{2\pi}} \frac{1}{\sqrt{1 + (\frac{\sigma_g}{a\sigma})^2}} \end{aligned} \quad (21)$$

Combining the previous two equations gives

$$\frac{\sigma}{\sqrt{1 + (\frac{\sigma_g}{a\sigma})^2}} = \sqrt{\frac{\pi}{8}} \eta, \quad (22)$$

or

$$\sigma = \frac{\sqrt{\pi}}{4} \eta \sqrt{1 + \sqrt{1 + \frac{32}{\pi} (\frac{\sigma_g}{a\eta})^2}}, \quad (23)$$

which is verified in Figure 6. The deviation at large a or small σ_g is due to discretization (x_t distribution cannot be viewed as a Gaussian). Comparing to SGD (Figure 5), the main difference is the $\sigma(a)$ is non-monotonic for SGD but is monotonic for SignGD. Another observation is that σ is lower bounded by $\frac{\sqrt{2\pi}}{4} \eta$ (when $\sigma_g \rightarrow 0$ or $a \rightarrow \infty$). In the limit of $\frac{\sigma_g}{a\eta} \gg 1$, we have $\sigma \propto \frac{\sqrt{\sigma_g \sqrt{\eta}}}{\sqrt{a}}$. Comparing this with SGD:

$$\sigma^{\text{SGD}} \propto \frac{\sigma_g \sqrt{\eta}}{\sqrt{a}}, \quad \sigma^{\text{SignGD}} \propto \frac{\sqrt{\sigma_g \sqrt{\eta}}}{\sqrt{a}}, \quad (24)$$

which suggests that SignGD is more robust to bigger noise since $\sigma^{\text{SGD}} \propto \sigma_g$ while $\sigma^{\text{SignGD}} \propto \sqrt{\sigma_g}$. Also since $\sigma^{\text{SignGD}} \propto \frac{1}{\sqrt{a}}$ just like $\sigma^{\text{SGD}} \propto \frac{1}{\sqrt{a}}$, the equipartition property applies to SignGD as well.

⁵The Gaussian approximation becomes increasingly more accurate as $\eta \rightarrow 0$.

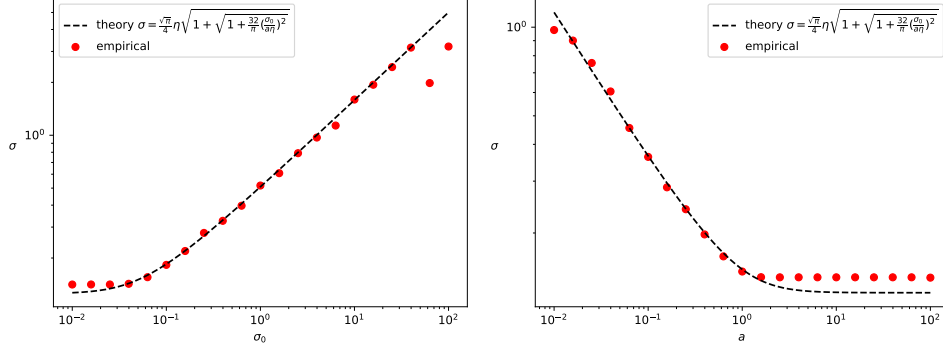


Figure 6: SignGD: Dependence of σ on gradient noise σ_g and sharpness a .

C.2 Learning rate decay

Considering time-varying learning rate $\{\eta_t\}_{t=0}^T$, Eq. (2) becomes:

$$x_{t+1} = x_t - \eta_t \text{sign}(ax_t + \sigma_g \dot{W}). \quad (25)$$

Similarly, we obtain the recursive relation for σ_t :

$$\sigma_{t+1}^2 = \eta_t^2 - 4\eta_t \int_{-\infty}^{\infty} dx p(x) \zeta(x) x + \sigma_t^2 = \eta_t^2 - \frac{4\sigma_t}{\sqrt{2\pi} \sqrt{1 + (\frac{\sigma_g}{a\sigma_t})^2}} \eta_t + \sigma_t^2, \quad (26)$$

which is a quadratic function against η_t . The optimal choice of η_t that minimizes σ_t is:

$$\eta_t = \frac{2\sigma_t}{\sqrt{2\pi} \sqrt{1 + (\frac{\sigma_g}{a\sigma_t})^2}}, \quad \sigma_{t+1}^2 = \sigma_t^2 - \frac{2}{\pi} \frac{\sigma_t^2}{1 + (\frac{\sigma_g}{a\sigma_t})^2} \approx \sigma_t^2 - \frac{2}{\pi} \left(\frac{a\sigma_t^2}{\sigma_g}\right)^2, \quad (27)$$

where the approximation holds when $\sigma_g \gg a\sigma_0 \geq a\sigma_t$ which is equivalent to $\sigma_g \gg a\eta$. Inverting both sides of the second equation gives

$$\frac{1}{\sigma_{t+1}^2} = \frac{1}{\sigma_t^2} \frac{1}{1 - \frac{2}{\pi} (\frac{a\sigma_t}{\sigma_g})^2} \approx \frac{1}{\sigma_t^2} \left(1 + \frac{2}{\pi} \left(\frac{a\sigma_t}{\sigma_g}\right)^2\right) = \frac{1}{\sigma_t^2} + \frac{2}{\pi} \left(\frac{a}{\sigma_g}\right)^2, \quad (28)$$

which is an arithmetic sequence. Hence $\frac{1}{\sigma_t^2} = \frac{1}{\sigma_0^2} + \frac{2}{\pi} \left(\frac{a}{\sigma_g}\right)^2 t$. Correspondingly

$$\eta_t \approx \sqrt{\frac{2}{\pi} \frac{a\sigma_t^2}{\sigma_g}} = \frac{\sqrt{\frac{\pi}{2} \frac{\sigma_g}{a}}}{t + \sqrt{2\pi} \frac{\sigma_g}{a\eta}} \propto \frac{1}{t + t_h}, \quad (t_h \equiv \sqrt{2\pi} \frac{\sigma_g}{a\eta}) \quad (29)$$

whose asymptotic behavior is $\eta_t \sim \frac{1}{t}$, just like SGD. We again observe that $\eta_0 \approx \eta/2$, which suggests that a continuous learning rate schedule might be suboptimal. Comparing the half time of SGD and SignGD:

$$t_h^{\text{SGD}} = \frac{2}{a\eta}, \quad t_h^{\text{SignGD}} = \sqrt{2\pi} \frac{\sigma_g}{a\eta}. \quad (30)$$

For a large model, the gradient (and gradient noise) of each individual parameter is usually small, i.e., $\sigma_g < 1$, suggesting the benefit of SignGD over SGD.

D Generalizing the two-temperature setup

In Section 4.2, we have shown that in the two-learning rate (temperature) setup $\eta_A > \eta_B$, the convergence from one equilibrium (η_A) to the other equilibrium (η_B) is exponential, similar to the situation when a hot object is in contact with a cooling thermostat, the temperature of the hot object converges exponentially to the temperature of the thermostat. However, this does not mean we can have $\bar{\ell}_f$ decay exponentially in time. What explains the difference? Note that the convergence rate $2a\eta_B$ is also proportional to η_B , meaning that η_B play both roles: temperature and time scale. This

is different from the situation in thermodynamics when temperature and time are independent (not controlled by a single parameter).

Making Eq. (4) continuous, we have

$$\frac{d\bar{\ell}}{dt} = -2a\eta(\bar{\ell} - \bar{\ell}_{eq}(\eta)), \quad \bar{\ell}_{eq}(\eta) \equiv C\eta. \quad (31)$$

Our goal is to design $\eta(t)$ such that $\bar{\ell}(t)$ is minimized. Given $\bar{\ell}$, the RHS is an inverted quadratic function of η , so the optimal η is $\eta^*(\bar{\ell}) \equiv \bar{\ell}/(2C)$. Inserting the relation to the RHS, we have

$$\frac{d\bar{\ell}}{dt} = -\frac{a}{2C}\bar{\ell}^2, \quad (32)$$

which solves to be $\bar{\ell}(t) = (\bar{\ell}(0)^{-1} + \frac{at}{2C})^{-1}$, and correspondingly $\eta(t) = (\frac{2C}{\bar{\ell}_0} + at)^{-1}$.

E Entropic Trapping

E.1 Example 1: Go down and stop

We set $c(x) = -cx$ ($c = 0.1$), $a(x) = a_0 + b|x|$ ($a_0 = b = 1$). Solving $F = 0$ gives

$$x_{-,+} = \frac{1}{\eta} \pm \sqrt{\left(\frac{1}{\eta}\right)^2 - \frac{b\sigma_g^2}{2c}}. \quad (33)$$

When $x > x_+$ or $x < x_-$, $F(x) < 0$, i.e., x moves to the left. When $x_- < x < x_+$, $F(x) > 0$, x moves to the right. As a result, when the initial point $x_0 < x_-$ or $(\frac{1}{\eta})^2 < \frac{b\sigma_g^2}{2c}$, x would decrease and end at 0. When $x_- < x < x_+$, x would decrease and end at x_+ . This is verified in Figure 7.

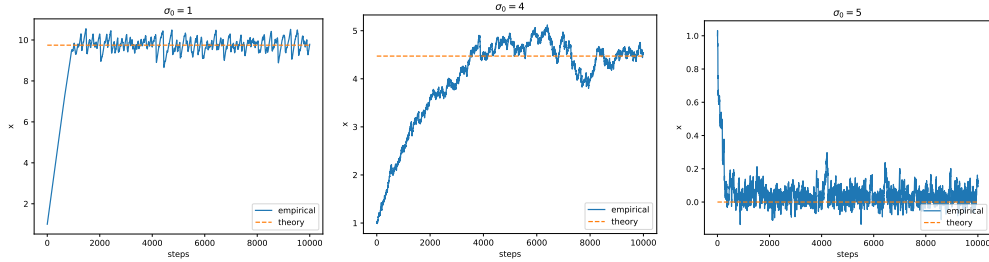


Figure 7: Increasing gradient noise σ_g makes the stop point shift to left (flatter region).

E.2 Example 2: Either left or right

Let us consider a specific case with $a(y) = \exp(ay)$ and $c(y) = -cy$. We have $F_{\text{btm}} = c$ and $F_{\text{ent}} = -\frac{d^2(\eta, \sigma_g)a}{2}$ which are both independent of y . When $d > \sqrt{\frac{2c}{a}}$, $F_{\text{ent}} + F_{\text{btm}} < 0$, meaning that y becomes smaller. $d > \sqrt{\frac{2c}{a}}$ translates to $\eta > \eta^* = \frac{4c}{a\sigma_g^2}$ (SGD) and $\eta > \eta^* \equiv \sqrt{\frac{32}{\pi}} \frac{c}{a\sigma_g}$ (SignGD). This means, to get deep down a narrowing valley, the learning rate should not be set too large, otherwise, the dynamic blocking could happen.

F Attraction forces

SGD with linear attraction force:

$$\bar{x}_t = \beta\bar{x}_{t-1} + (1-\beta)x_t, \quad x_t = x_{t-1} - \eta(ax_t + \gamma(x_{t-1} - \bar{x}_t) + \sigma_g\dot{W}_{t-1}), \quad (34)$$

with the standard deviation of x being

$$\sigma = \sqrt{\frac{\eta(1-\beta^2 + a\beta(1+\beta)\eta - 2\beta^3\gamma\eta)}{(a + \gamma(1-2\beta))(2(1+\beta) - \eta((a+\gamma)(1+\beta) - 2\beta^2\gamma))(1 + \beta(-1 + a\eta + 2(1-\beta)\gamma\eta))}} \sigma_g. \quad (35)$$

We verify that when $\gamma \rightarrow 0$, σ goes back to $\sigma = \frac{1}{\sqrt{a(\frac{2}{\eta} - a)}}\sigma_g$.

When $\beta \rightarrow 0$, $\sigma = \frac{1}{\sqrt{(a+\gamma)(\frac{2}{\eta} - (a+\gamma))}}\sigma_g$, which means that the role of γ is to shift sharpness from a to $a + \gamma$. So in the flat limit $a \ll 1/\eta$, a reasonable $\gamma < 1/\eta - a$ can make σ smaller, i.e., reducing valley variations.

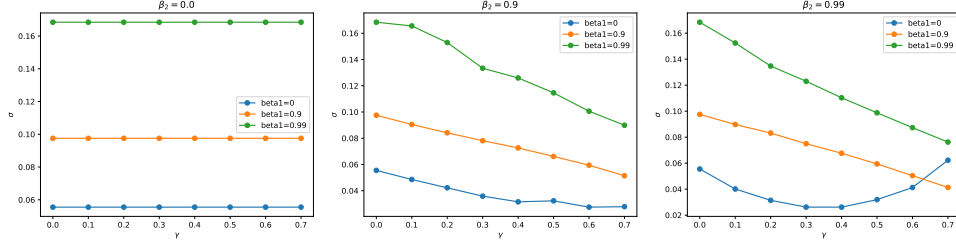


Figure 8: Gaussian width σ for different self-attracting force γ , $\eta = 0.1$, $a = 1$, $\sigma_g = 1$.

G Designing learning rate schedules

What insights can we gain to make training more effective? If we assume Δ_{entropic} and Δ_{anneal} can be ignored, the only way to reduce loss is by reducing $\ell(D, \eta_{\min})$, which involves reducing η_{\min} and/or increasing D .

However, reducing η_{\min} may have a non-trivial effect on Δ_{anneal} , because in Section 4 we have shown that the decay time $T_d \sim 1/\eta_{\min}$, meaning that if one wants to reduce η_{\min} by a factor of 2, the duration of the decay phase should be 2 times longer, which is not very efficient.

We now consider fixing η_{\min} but increasing D by choosing a larger η in the stable phase. There are two potential concerns with this strategy: (1) Perhaps a longer decay schedule is needed to decay a larger η to η_{\min} . Luckily, this is not a problem because Section 4 showed that T_d has an upper bound $O(1/\eta_{\min})$ which is independent of η . (2) Perhaps a larger η includes larger entropic forces. Our η sum alignment experiments in Section 5 show that the effect of entropic forces is negligible. With both concerns cleared, our experiments in Figure 9 indeed show the efficiency of choosing a larger stable learning rate. However, the learning rate cannot be too large to cause numerical problems. Note that our experiments are done on two V-100s with float16 precision. More advanced machines are supposed to allow even higher learning rates without a blowup. We also note that our strategy is not just running the stable phase longer, which inevitably faces a trade-off: longer stable phase reduces $\ell(D, \eta_{\min})$ by increasing D , but potentially increases Δ_{anneal} since the decay phase is eaten by the stable phase. The trade-off is shown in Figure 10.

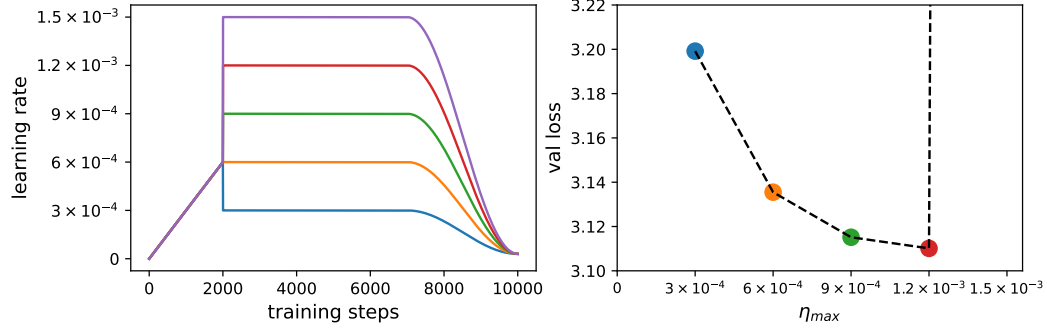


Figure 9: Learning rate schedules with different stable learning rate η . A larger η leads to a lower validation loss, unless NaN issues occur (at $\eta = 1.5 \times 10^{-3}$).

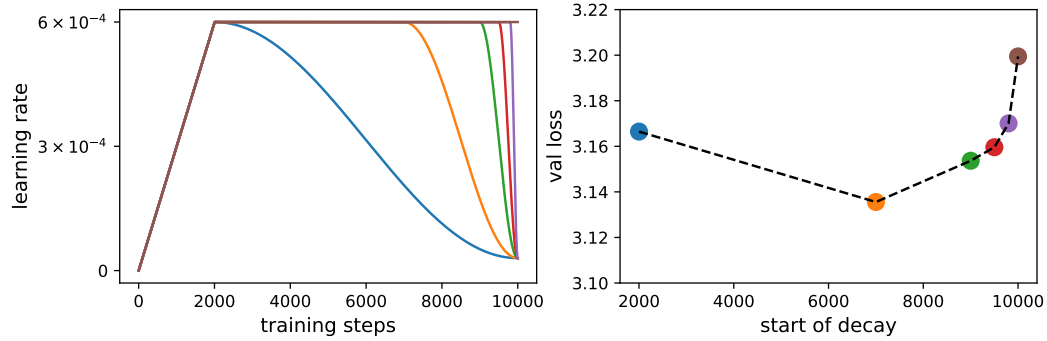


Figure 10: Learning rate schedules with different starting points of the decay phase. A longer stable phase does not necessarily lead to lower validation loss, since the decay phase cannot be too short due to annealing.