Matematiksel Kanıt Yapan Yapay Zeka

Dil modelleri (LLM'ler) matematiksel ispatta nasıl bu kadar iyileşti? Ölçütlerin (benchmark) gizlediği şey ne?

Üç yıl önce ilkokul aritmetiğinde takılıyorlardı. Bugün IMO altın madalya seviyesinde sonuç bildiriliyor. Bu slaytlarda hem *neyin işe yaradığını* hem de *neyin abartıldığını* ayırmaya çalışacağız.

Artificial intelligence: an empirical science

Herbert A. Simon*

Department of Psychology, Carnegie Mellon University, Pittsburgh, PA 15213-3890, USA

Received August 1993; revised May 1995

Abstract

My initial tasks in this paper are, first, to delimit the boundaries of artificial intelligence, then, to justify calling it a science: is AI science, or is it engineering, or some combination of these? After arguing that it is (at least) a science, I will consider how it is best pursued: in particular, the respective roles for experiment and theory in developing AI.

I will rely more on history than on speculation, for our actual experience in advancing the field has much to tell us about how we can continue and accelerate that advance. Many of my examples will be drawn from work with which I have been associated, for I can speak with greater confidence about what motivated that work and its methods (and about its defects) than I can about the work of others. My goal, however, is not to give you a trip through history, but to make definite proposals for our future priorities, using history, where relevant, as evidence for my views.

Human mathematicians are being outcounterexampled

Posted on July 20, 2026 by xenaproject

AI makes a major breakthrough in a math problem that had stumped experts for decades

by Paul Arnold, Phys.org

edited by Gaby Clark, reviewed by Robert Egan

Editors' notes

AI Solved A Mathematical Problem That Had Stumped The World's Best Minds For Decades

By Anisha Sircar, Contributor. ⓘ Anisha Sircar is a journalist covering tech, fin...

Published Apr 17, 2026, 06:15am EDT

Follow Author

August 1, 2026 Publication

Ten advances in thematics and theoretical mputer science

You reposted

levent

@__alpoge__

hello there the jacobian conjecture is false thanx to my close friend akhil for asking about it and my other close friend fable for working during the world cup final

((1+xy)^3 z + y^2 (1+xy) (4+3xy), y + 3 x (1+xy)^2 z + 3 x y^2 (4+3xy), 2 x - 3 x^2 y - x^3 z): \C^3\to \C^3, has jacobian determinant -2, and sends (0, 0, -1/4), (1, -3/2, 13/2), and (-1, 3/2, 13/2) to (-1/4, 0, 0)

12:19 PM · Jul 20, 2026 · **20.7M** Views

Sunumun yol haritası

Giriş: LLM'ler ve training pipeline'ı
Model ne yapıyor, pretraining ve post-training nasıl işliyor

Modellerin neler başardıklarına dikkatli bir bakış
Sonuçlar gerçekten ne gösteriyor, ne göstermiyor

Matematik neden özel
Doğrulama asimetrisi

Pretraining tarafındaki gelişmeler
Veri, tokenization, uzun context (bağlam)

Post-training tarafındaki gelişmeler
SFT, RLVR, GRPO, tool use (araç kullanımı) + işe yaramayan denemeler

Ek bölüm: Reinforcement learning derinlemesine
PPO, GRPO ve process reward model'ler (PRM)

Tipik LLM training (eğitim) aşamaları

Çoğu üretici model bugün aşağı yukarı bu sırayı izliyor — laboratuvara göre adım sayısı ve isimler değişir.



NEDEN BU SIRA

Pretraining modele dil ve dünya bilgisini verir; SFT "biçimi" öğretir, RLVR doğru cevaba giden akıl yürütmeyi güçlendirir, RLHF genel kullanılabilirlik ve güvenliği hizalar.

Pretraining'de (ön-eğitim) LLM'ler ne yapıyor?

LLM (büyük dil modeli) pretraining sırasında tek bir şey öğrenir: verilen metnin devamındaki **token**'ı (metnin bölündüğü en küçük parça) tahmin etmek.

$$\mathcal{L}(\theta) = -\sum_{t=1}^{T} \log p_\theta(x_t \mid x_{<t})$$

Pretraining (ön eğitim — internetten toplanmış devasa metin yığını üzerinde bu kaybı minimize etme aşaması) bundan ibaret. Başka hedef fonksiyonu yok.

KRITIK NOKTA

İspat yazma yeteneği bu hedefin doğrudan amacı değil, *yan ürünü*.

Post-training'de LLM'ler ne yapıyor?

Pretraining'in tek kaybının aksine post-training aşamaları farklı hedeflerle çalışır. İki temel örnek:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) \;=\; -\sum_{t=1}^{T} \log p_\theta(y_t \mid x, y_{<t}), \quad (x,y) \sim \mathcal{D}_{\mathrm{demo}}$$

SFT (supervised fine-tuning - denetimli ince ayar) insan yazımı örnek cevapları taklit etmeyi öğretir; pretraining ile aynı kayıp, farklı veri.

$$J(\theta) \;=\; \mathbb{E}_{q\sim\mathcal{D},\, o\sim\pi_\theta(\cdot|q)}\Big[\, r(q,o) \,\Big] \;-\; \beta\, \mathrm{KL}\big(\pi_\theta \,\|\, \pi_{\mathrm{ref}}\big)$$

RL (reinforcement learning - pekiştirmeli öğrenme) LLM'lerin bir ödül fonksiyonu altında yüksel ödül üreteceği sekilde eğitimi hedefler.

"Reasoning" modeli ne demek

Cevabı doğrudan üretmek yerine, model önce uzun bir **chain-of-thought** (akıl yürütme zinciri) yazıyor (ara adımlar, dallanmalar, kendini kontrol); sonra nihai cevaba varıyor.

KLASIK MODEL

Soru → doğrudan cevap. Tek geçişte, geri dönüşsüz.

REASONING MODELI

Soru → uzun ara akıl yürütme (binlerce token) → cevap. Dener, geri döner, kendini denetler.

Sunumun yol haritası

Giriş: LLM'ler ve training pipeline'ı
Model ne yapıyor, pretraining ve post-training nasıl işliyor

Modellerin neler başardıklarına dikkatli bir bakış
Sonuçlar gerçekten ne gösteriyor, ne göstermiyor

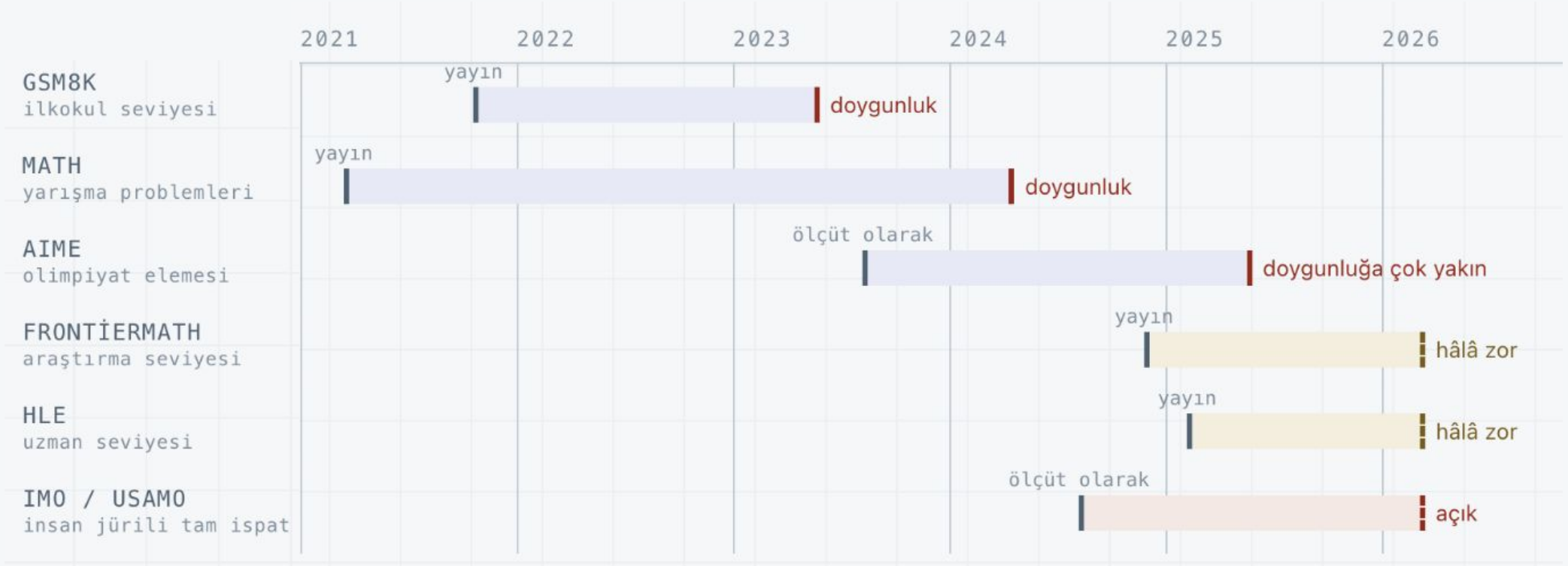
Matematik neden özel
Doğrulama asimetrisi

Pretraining tarafındaki gelişmeler
Veri, tokenization, uzun context (bağlam)

Post-training tarafındaki gelişmeler
SFT, RLVR, GRPO, tool use (araç kullanımı) + işe yaramayan denemeler

Ek bölüm: Reinforcement learning derinlemesine
PPO, GRPO ve process reward model'ler (PRM)

Yaygın matematik ölçütleri (benchmark)



Bu ölçütlerden birer örnek soru

ÖLÇÜT	ÖRNEK SORU
GSM8K ilkokul seviyesi	Ali'nin 5 kalemi var. Ayşe'nin kalem sayısı Ali'ninkinin iki katı. İkisinin toplam kaç kalemi var?
MATH yarışma problemleri	$x^2 - 5x + 6 = 0$ denkleminin köklerinin çarpımı kaçtır?
AIME	3^n sayısı 143 tabanında yazıldığında en sağdaki iki basamağı 01 olacak şekilde en küçük pozitif n tam sayısını bulun.
IMO 2025	Pozitif tam sayılar için $a^2 + b^2 = c^2$ olacak şekilde, $c \leq n$ koşulunu sağlayan üçlülerin sayısının belli bir formülle ifade edilebileceğini *ispatlayın*.
FrontierMath araştırma seviyesi	p, q farklı asal sayılar olsun. $f(n) = \sum_{k=1}^{n} \gcd(k, n)$ olmak üzere, $n = p^3 \cdot q^2$ formunda ve $f(n)/n > 6/5$ koşulunu sağlayan, 10^6'dan küçük en büyük n'i bulun.
HLE araştırma seviyesi	Mertebesi $2025 = 3^4 \cdot 5^2$ olan sonlu bir grup G'nin normal bir Sylow 3-altgrubu ve mertebesi 3'ü bölen 81 eleman bulunsun. G'nin, tamamı mertebesi 5 olan elemanlardan oluşan kaç farklı konjugasyon sınıfı vardır?

Sorular gösterim amaçlı yeniden yazılmıştır, gerçek ölçüt sorularının doğrudan alıntısı değildir.

Bu ölçütlerin çoğu ispatı değil, *nihai cevabı* ölçer

AIME SORUSU NASIL PUANLANIR

Cevap 0-999 arası bir tam sayıdır. Model bir sayı üretir, sistem onu doğru sayıyla karşılaştırır.

Yani bir `string` eşleşmesi.

BUNUN ÖLÇMEDİĞİ ŞEY

- ✕ Ara adımların geçerli olup olmadığı
- ✕ Kullanılan lemmanın gerekçelendirilip gerekçelendirilmediği
- ✕ Bütün durumların (case) ele alınıp alınmadığı
- ✕ Doğru cevaba *şansla* mı yoksa akıl yürüterek mi varıldığı

SONUÇ

Manşetlerdeki skorların büyük kısmı "ispat yazabiliyor" demiyor; "doğru sayıyı bulabiliyor" diyor. Bunlar aynı şey değil.

İspatlar insan jüriyle puanlandığında ne oldu?

MATHARENA · USAMO 2025

USAMO soruları yayımlandıktan hemen sonra, o dönemin en iyi modellerine verildi. Ürettikleri *tam ispatlar* matematikçiler tarafından olimpiyat rubriğiyle puanlandı.

Aynı modeller AIME'de çok yüksek skorlar alıyordu.

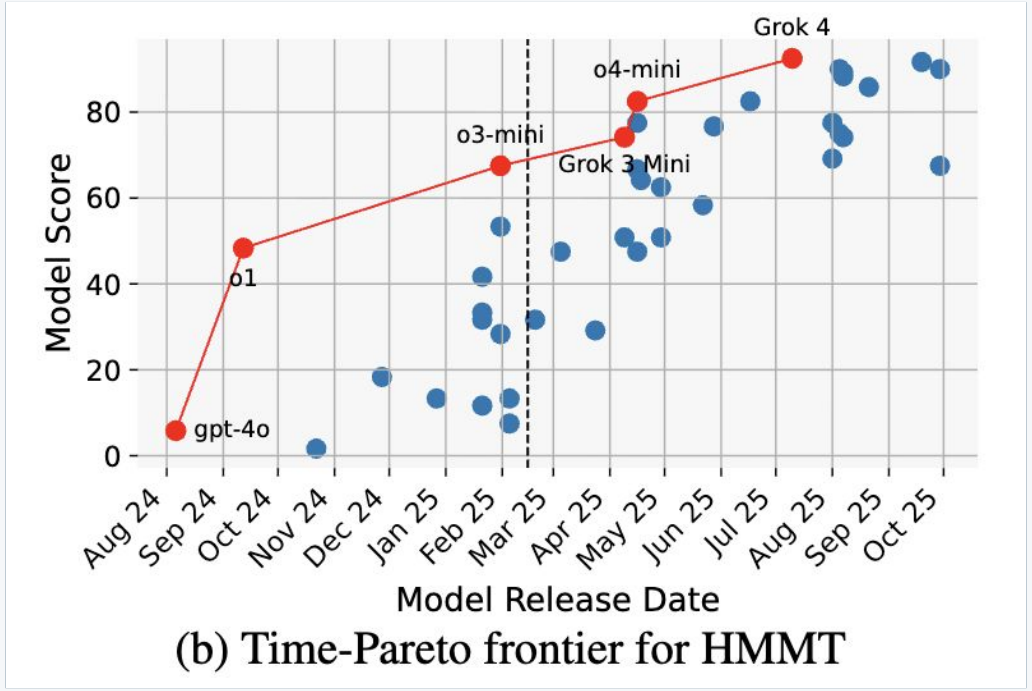
matharena.ai

SONUÇ

< %5

Ortalama ispat puanı.

MODELLER HIZLA IYILEŞIYOR



MathArena, Fig. 3b — HMMT için zaman-Pareto sınırı

LLM'lerin tipik hata biçimleri

1. GEREKÇESİZ LEMMA

"Açıkça görülüyor ki…", "kolayca gösterilebilir ki…" — ve ispatın tüm ağırlığı o cümlede.

2. ATLANAN DURUM

Genelliği bozmadan (WLOG) diyip simetrik olmayan bir ifadede sıralama varsayması. Vaka analizinin bir kolu sessizce kaybolur.

3. YANLIŞIN AKICI İSPATI

Doğru olmayan bir önerme verildiğinde modelin onu reddetmek yerine ikna edici bir "ispat" üretmesi.

Örnek: $a^3 + b^3 + c^3 - 3abc \geq 0$ için "genelliği bozmadan $a \geq b \geq c$ " — ifade simetrik olduğu için burada geçerli.

Aynı hamle $a^2b + b^2c + c^2a$ üzerinde **geçersiz** — bu ifade yalnızca *döngüsel*, simetrik değil.

> Bu tam olarak modelin yaptığı türden hata: doğru ortamda öğrenilmiş bir hamlenin yanlış ortama taşınması.

Matematik için kritik ayrım

KULAĞA ISPAT GIBI GELEN	İSPAT OLAN
Doğru sözdizimi, doğru sözcük dağarcığı	Her adım öncekilerden *zorunlu olarak* çıkar
Adımlar birbirini takip ediyormuş gibi duruyor	Hiçbir durum atlanmamış
Uzman olmayan bir okuru ikna eder	Bir makine tarafından kontrol edilebilir
Bir dil modelinin doğrudan optimize ettiği şey	**Ancak dolaylı yollarla optimize edilebilen şey**

TEZ

Son üç yılın bütün teknik gelişmeleri, bu iki sütunu birbirine yaklaştırma çabası olarak okunabilir.

BÖLÜM 1 · ÖZET

İlerleme gerçek. Ölçtüğümüz şey ispat değil, nihai cevap.

Aynı modeller AIME'de %90'ın üzerinde, insan jürili USAMO ispatlarında tek haneli yüzdelere düşüyorlardi. Aradaki fark, modelin zayıflığı kadar ölçütün ne ölçtüğüyle ilgili.

Bölüm 1 bitti

Sunumun yol haritası

Giriş: LLM'ler ve training pipeline'ı
Model ne yapıyor, pretraining ve post-training nasıl işliyor

Modellerin neler başardıklarına dikkatli bir bakış
Sonuçlar gerçekten ne gösteriyor, ne göstermiyor

Matematik neden özel
Doğrulama asimetrisi

Pretraining tarafındaki gelişmeler
Veri, tokenization, uzun context (bağlam)

Post-training tarafındaki gelişmeler
SFT, RLVR, GRPO, tool use (araç kullanımı) + işe yaramayan denemeler

Ek bölüm: Reinforcement learning derinlemesine
PPO, GRPO ve process reward model'ler (PRM)

Doğrulama asimetrisi

Matematikte bir çözümü *bulmak*, onu *kontrol etmekten* çok daha pahalıdır.

Nihai cevabı doğrulamak

AIME tipi kısa sayısal cevap, bilinen cevapla karşılaştırılır. Ucuz — ama ispatı değil, sonucu doğrular.

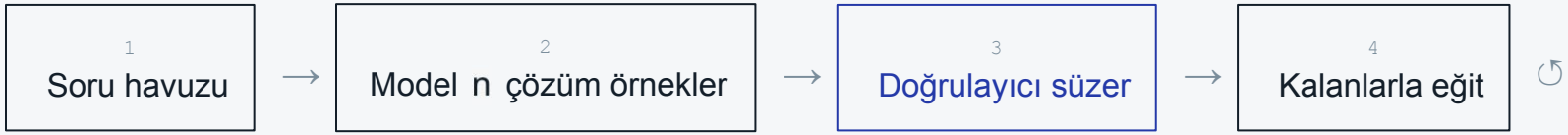
Formal ispatı doğrulamak

Lean gibi bir sistemde yazılmış ispatı kernel mekanik olarak kontrol eder. Ucuz *ve* gerçekten ispatı doğrular.

NEDEN ÖNEMLI

Ucuz bir doğrulayıcınız (verifier) varsa, iki şeyi bedavaya elde edersiniz: **sınırsız eğitim verisi** ve **insan etiketi gerektirmeyen bir ödül sinyali** . Sonraki iki slayt bunlar.

Kendi eğitim verinizi üretebilirsiniz



$$\mathcal{D}' = \left\{ (q, o) \;:\; o \sim \pi_\theta(\cdot \mid q),\;\; r(q, o) = 1 \right\}$$

Rejection sampling (reddetme örneklemesi — çok üret, doğru olanları tut)

Expert iteration (süzülmüş kendi çıktısıyla tekrar tekrar eğitme döngüsü)

Model n denemede bir kez doğru yapıyorsa, o bir doğru çözüm kalıcı eğitim verisine dönüşür. Beceri, seyrek başarıdan *güvenilir* başarıya doğru damıtılır.

Zelikman ve ark., 2022 — STaR: Bootstrapping Reasoning With Reasoning, arxiv.org/abs/2203.14465

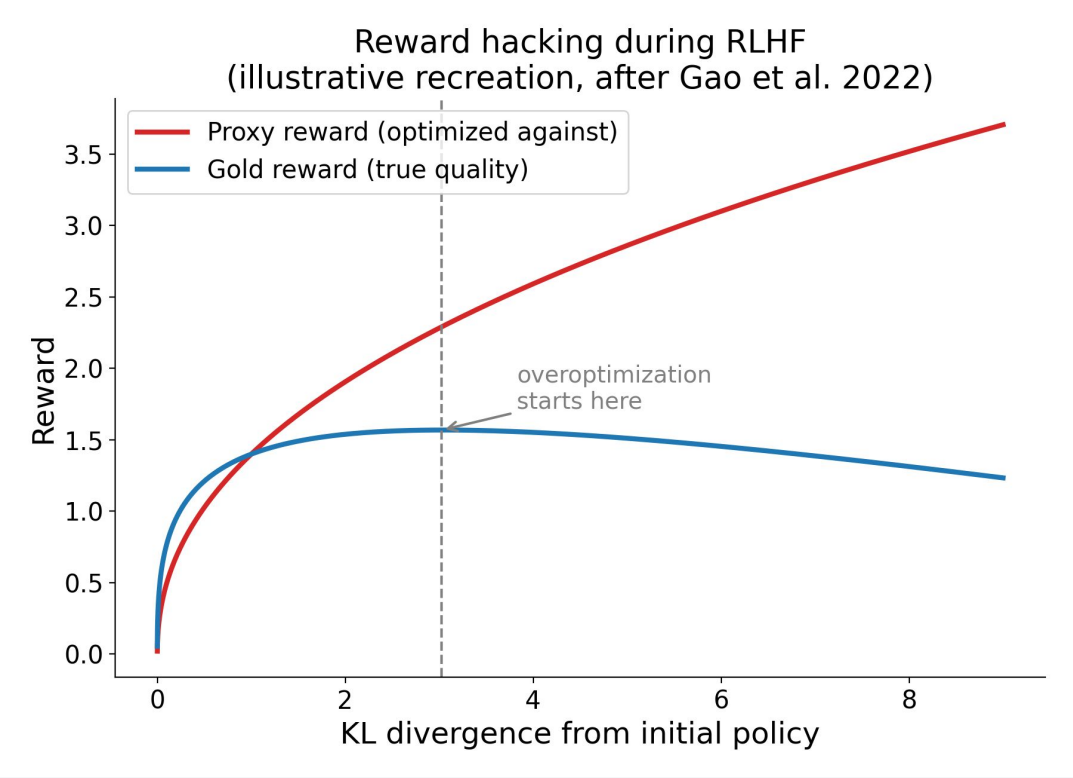
İnsan etiketi olmadan ödül sinyali

RLHF (PEKIŞTIRMELI ÖĞRENME, INSAN GERI BILDIRIMINDEN)

Standart **RLHF**'te (reinforcement learning from human feedback — insan geri bildiriminden pekiştirmeli öğrenme) ilk aşamada bir **reward (ödül) modeli** , insan tercihlerini tahmin etmeyi öğrenir. Ardından model, bu reward modeli altında yüksek ödül başaracak şekilde eğitilir. Öğrenilmiş her ödül modeli **reward hacking**'e (ödülü, asıl amacı gerçekleştirmeden yüksek puan alacak şekilde sömürme) açıktır.

$$r = R_\phi(q, o)$$

Öğrenilmiş, gürültülü, sömürülebilir.



İnsan etiketi olmadan ödül sinyali

RLHF (PEKIŞTIRMELI ÖĞRENME, INSAN GERI BILDIRIMINDEN)

Standart **RLHF**'te (reinforcement learning from human feedback — insan geri bildiriminden pekiştirmeli öğrenme) ilk aşamada bir **reward (ödül) modeli** , insan tercihlerini tahmin etmeyi öğrenir. Ardından model, bu reward modeli altında yüksek ödül başaracak şekilde eğitilir. Öğrenilmiş her ödül modeli **reward hacking**'e (ödülü, asıl amacı gerçekleştirmeden yüksek puan alacak şekilde sömürme) açıktır.

$$r = R_\phi(q, o)$$

Öğrenilmiş, gürültülü, sömürülebilir.

RLVR (DOĞRULANABILIR ÖDÜLLERLE PEKIŞTIRMELI ÖĞRENME)

RLVR (reinforcement learning with verifiable rewards — doğrulanabilir ödüllerle pekiştirmeli öğrenme) : ödül fonksiyonunun yerini bir `if` alır.

$$r = 1\left[\text{cevap}(o) = a^*\right]$$

Programatik, kesin, sömürülmesi zor.

Doğrulayıcının en temiz hâli: formal ispat

LEAN 4 · MATHLIB

```
-- Her n için n(n+1) çifttir
theorem two_dvd_mul_succ (n : ℕ) :
    2 ∣ n * (n + 1) :=
  (Nat.even_mul_succ_self n).two_dvd
```

✓ derlendi — kernel ispatı kabul etti

```
-- Bir adım eksik bırakılırsa
  ... := by sorry
error: declaration uses 'sorry'
```

İfade insan tarafından yazılır

Teoremi formal dile çevirmek hâlâ insan işi — ve asıl darboğaz burası.

İspatı makine kontrol eder

Küçük, güvenilir bir kernel her adımı mekanik olarak denetler. "Açıkça görülüyor ki" burada geçmez.

Sonuç ikili bir sinyaldir

Derlenir ya da derlenmez. Hakem yorgunluğu, ikna edici üslup, yarım bırakılan vaka yok.

Bu neden her alana genellenmiyor?

ALAN	UCUZ DOĞRULAYICI?	SONUÇ
Matematik (nihai cevap)	Var	RLVR çalışıyor, ilerleme hızlı
Formal ispat (Lean)	Var, üstelik ispatın kendisi için	En temiz kurulum
Kod	Kısmen (testler)	Testin kapsadığı kadar
Doğal dildeki ispat	Zayıf	İnsan jüri gerekiyor, ölçeklenmiyor
Deneme, analiz, tıbbi tavsiye	Yok	Karşılaştırılabilir bir sıçrama yok

BURADAN ÇIKARILMAMASI GEREKEN SONUÇ

"Matematikte bu kadar hızlı ilerlediyse her yerde ilerler." Matematikteki hız, matematiğin kendine özgü yapısının sonucu. Genel yetenek için *zayıf* bir öncü göstergedir.

BÖLÜM 2 · ÖZET

Ucuz doğrulayıcı = sınırsız eğitim verisi
ve
insan etiketi gerektirmeyen ödül.

Matematiğin ayrıcalığı bu. Doğrulayıcının zayıf olduğu yerlerde — doğal dildeki ispat dahil — aynı hızı beklemek için sebep yok.

Bölüm 2 bitti

Sunumun yol haritası

Giriş: LLM'ler ve training pipeline'ı
Model ne yapıyor, pretraining ve post-training nasıl işliyor

Modellerin neler başardıklarına dikkatli bir bakış
Sonuçlar gerçekten ne gösteriyor, ne göstermiyor

Matematik neden özel
Doğrulama asimetrisi

Pretraining tarafındaki gelişmeler
Veri, tokenization, uzun context (bağlam)

Post-training tarafındaki gelişmeler
SFT, RLVR, GRPO, tool use (araç kullanımı) + işe yaramayan denemeler

Ek bölüm: Reinforcement learning derinlemesine
PPO, GRPO ve process reward model'ler (PRM)

Başarı için en büyük faktör: veri seçimi

Matematiğe özel derlemeler

OpenWebMath ve benzerleri: web'den LaTeX ve matematiksel içeriğin ayıklanması. Genel web taramasının çok küçük bir kesiti.

Sınıflandırıcıyla veri toplanması

DeepSeekMath yaklaşımı: küçük bir çekirdek matematik kümesiyle bir sınıflandırıcı eğit, Common Crawl'ın tamamını onunla tara, bulduklarını çekirdeğe ekle, tekrarla. Yüz milyarlarca token'lık matematik derlemesi.

arXiv kaynak dosyaları

PDF değil, ham LaTeX. İspat yapısı işaretlenmiş hâlde geliyor: theorem, lemma, proof, □.

Formal derlemler

Lean/mathlib: makine tarafından doğrulanmış, kusursuz doğrulukta yüz binlerce teorem.

YAYIMLANAN ABLASYONLARIN ORTAK BULGUSU

Bu alanda **kalite, ölçekten daha çok kazandırdı.** Daha çok token değil, daha iyi seçilmiş token.

Az konuşulan, çok işe yarayan: mid-training

Mid-training / annealing (ön eğitimin son aşamasında veri karışımının değiştirilmesi ve öğrenme hızının düşürülmesi) : son yüz milyarlarca token'da matematik, kod ve akıl yürütme içeriğinin ağırlığı belirgin biçimde artırılır.

$$p_{\text{karışım}}(t) = (1-\lambda_t)\,p_{\text{genel}} + \lambda_t\,p_{\text{matematik}}, \qquad \lambda_t \nearrow \text{ eğitimin sonuna doğru}$$

Baştan sona yüksek matematik oranı kullanmaktan daha iyi sonuç veriyor
Toplam hesap bütçesinin küçük bir kısmına mal oluyor
Modelin "son gördüğü şey" orantısız etkili — pratikte herkesin bildiği, teorisi zayıf bir gözlem

Sayıların parçalanması konusunda düzeltme

Tokenization (metnin token'lara bölünmesi) sıklık temelli çalışır. Sayılar için bu felaket bir sonuç doğurur:

```
eski tokenizer'lar:  12345  →  123 45    veya 12346  →  1 2346
yeni tokenizer'lar:  12345  →  1 2 3 4 5
```

Eski şemada aynı basamak farklı sayılarda farklı yerlere düşerdi. Elde taşımalı toplama ise basamakların *hizalanmasını* gerektirir. Model, hiç olmaması gereken bir zorlukla boğuşuyordu.

Basamak başına ya da sabit üçlü gruplarla tokenization, aritmetik doğruluğunu ölçülebilir biçimde artırdı. Küçük, ucuz, gerçek bir kazanç.

Uzun düşünmeyi mümkün kılan altyapı

Modern bir model bir olimpiyat sorusunda on binlerce token'lık ara akıl yürütme üretir.

$$\underbrace{O(n^2 d)}_{\text{attention maliyeti}} \qquad \underbrace{2\,L\,H_{kv}\,d_h\,n}_{\text{KV cache boyutu (GPU içi transfer)}}$$

KV cache (üretim sırasında önceki token'ların ara hesaplarının bellekte tutulması) — uzun düşünce zincirinin asıl maliyeti burada

GQA / MLA

Sorgu başlıklarının anahtar-değer başlıklarını paylaşması ya da sıkıştırılması. H_{kv} küçülür, cache birkaç kat ucuzlar.

RoPE ölçekleme

Eğitimde görülenden çok daha uzun dizilere genelleme.

BÖLÜM 3 · ÖZET

Pretraining tavanı belirler. Post-training o tavana ulaşır.

Bu, alanın büyük ölçüde üzerinde anlaştığı formülasyon. Ama tamamen değil — "post-training tavanı yükseltebilir mi?" sorusu hâlâ açık.

Bölüm 3 bitti

Sunumun yol haritası

Giriş: LLM'ler ve training pipeline'ı
Model ne yapıyor, pretraining ve post-training nasıl işliyor

Modellerin neler başardıklarına dikkatli bir bakış
Sonuçlar gerçekten ne gösteriyor, ne göstermiyor

Matematik neden özel
Doğrulama asimetrisi

Pretraining tarafındaki gelişmeler
Veri, tokenization, uzun context (bağlam)

Post-training tarafındaki gelişmeler
SFT, RLVR, GRPO, tool use (araç kullanımı) + işe yaramayan denemeler

Ek bölüm: Reinforcement learning derinlemesine
PPO, GRPO ve process reward model'ler (PRM)

Tipik LLM training (eğitim) aşamaları

Çoğu üretici model bugün aşağı yukarı bu sırayı izliyor — laboratuvara göre adım sayısı ve isimler değişir.



NEDEN BU SIRA

Pretraining modele dil ve dünya bilgisini verir; SFT "biçimi" öğretir, RLVR doğru cevaba giden akıl yürütmeyi güçlendirir, RLHF genel kullanılabilirlik ve güvenliği hizalar.

Post-training adımlarından somut örnekler

Aynı soru üzerinden dört aşamanın ne yaptığını gösteren minyatür örnekler.

ADIM 1 · SFT

Bir soru ve insan yazımı örnek cevap çifti gösterilir: { **"instruction"**: "Kara deliğin ne olduğunu basitçe açıkla.", **"output"** : "Kara delik, uzayda kütle çekiminin o kadar güçlü olduğu bir bölgedir ki hiçbir şey (ışık bile) oradan kaçamaz" } Model bu **biçimi taklit etmeyi** öğrenir.

ADIM 2 · RLVR

Aynı soruya model çok sayıda çözüm dener. Nihai cevap otomatik kontrol edilir: doğruysa ödül = 1.
Sadece **doğru sonuca götüren** akıl yürütmeler pekiştirilir.

ADIM 3 · RLHF

Aynı soruya iki farklı cevap üretilir (örneğin biri net ve kısa, biri dağınık). İnsan hangisini tercih ettiğini işaretler. Ödül modeli bu tercihten **genel üslup ve kullanılabilirliği** öğrenir.

ADIM 4 · REJECTION SAMPLING

Model aynı soruya 100 çözüm üretir. Yalnızca doğru sonuca ulaşan ~30 tanesi süzülür. Model bu **süzülmüş kendi çıktısıyla** tekrar eğitilir.

Uzun düşünce zincirini öğretmenin en ucuz yolu

Post-training'in ilk adımı genellikle **SFT** (supervised fine-tuning — denetimli ince ayar: örnek girdi-çıktı çiftleri üzerinde normal eğitim) .

```
Girdi: "3x+5=20 ise x?" Çıktı (örnekten kopyalanan biçim): "3x=15 ⇒ x=5"
```

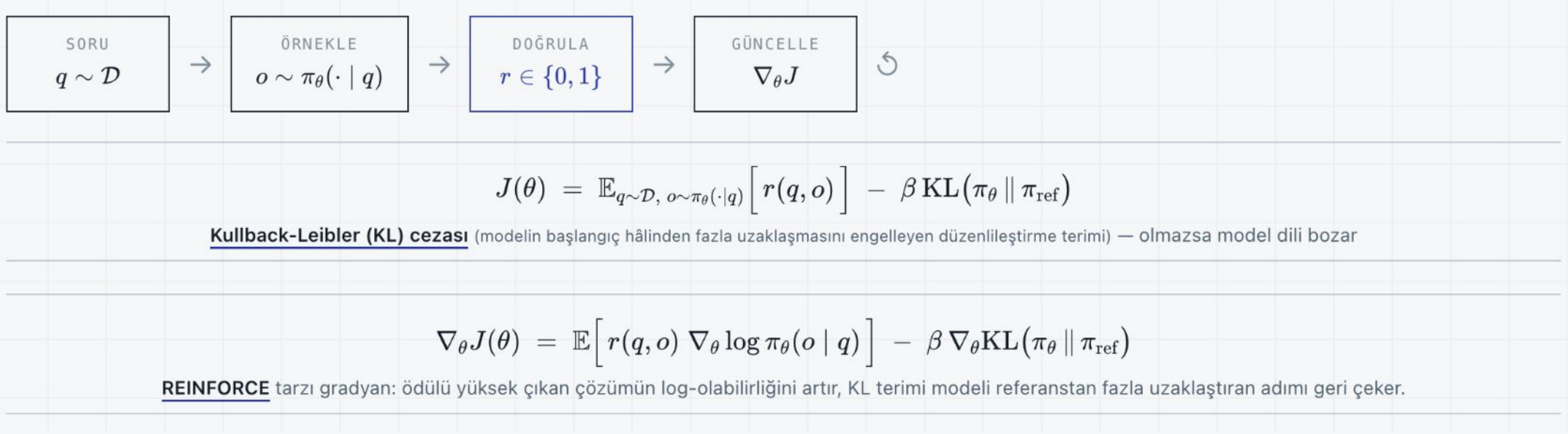
Chain of thought (düşünce zinciri — modelin cevabı vermeden önce ara adımları açıkça yazması) uzun ve dallanan hâliyle örneklerden öğretilebilir.

```
"Önce durumu inceleyelim… bu yaklaşım işe yaramazsa… o zaman şunu deneyelim… demek ki cevap 5."
```

ŞAŞIRTICI BULGU (R1-DİSTİLL)

Güçlü bir akıl yürütücünün ürettiği düşünce zincirleri toplanıp küçük modellere **düz SFT ile** verildiğinde, akıl yürütme davranışının büyük kısmı aktarılıyor. RL adımı olmadan.

RLVR döngüsü (reinforcement learning with verifiable rewards — doğrulanabilir ödüllerle pekiştirmeli öğrenme)



Gözlenen etki: model kendiliğinden daha uzun düşünmeye, geri dönüp denemeye ve cevabı kontrol etmeye başlar — **doğruluğu ödüllendirmenin yan ürünü**.

Aritmetiği parametrelerle yapmayı bırakmak

Tool use (modelin bir hesap makinesi, Python yorumlayıcısı ya da ispat kontrolcüsü çağırabilmesi) — ve bunun RL ile öğretilmesi: modelin ne zaman araç çağıracağı da ödülden öğrenilir.

ARAÇSIZ

$47^3 = ?$ modelin parametreleriyle, yaklaşık olarak, hata payıyla hesaplanır. Uzun bir zincirin ortasındaki tek bir aritmetik hatası her şeyi çöpe atar.

ARAÇLI

Model kodu yazar, yorumlayıcı çalıştırır, sonucu geri alır. Aritmetik hata sınıfı **tamamen** ortadan kalkar.

> **GENEL PRENSIP**
>
> Bir hata sınıfını modelde düzeltmeye çalışmak yerine, o işi hatasız yapan bir sisteme devretmek.

İşe yaramayanlar

Process reward model (PRM) (süreç ödül modeli — her ara adıma ayrı puan veren model)

Teoride tam ihtiyacımız olan şeydi: ispatın *adımlarını* ödüllendirmek. Pratikte adım etiketlemek pahalı, "doğru adım" tanımı belirsiz ve model ödül modelini sömürüyor. Birçok laboratuvar sonuç temelli ödüle döndü.

Aşırı uzun düşünce zinciri

Belli bir noktadan sonra daha fazla token daha fazla doğruluk getirmiyor; model kendi hatalarında dolanıyor.

Naif kendini düzeltme istemleri

"Cevabını gözden geçir" demek doğruluğu *düşürebiliyor*.

> **BUNUN ANLAMI**
>
> İspat kalitesini doğrudan hedefleyen en umut verici yöntem (PRM) şu ana kadar en çok hayal kırıklığı yaratan yöntem oldu. Ölçtüğümüz ile önemsediğimiz arasındaki boşluk bu yüzden kapanmıyor.

RL yeni akıl yürütme mi yaratıyor, yoksa var olana daha hızlı mı erişiyor?

Çoğu zorlu matematik sorusunun çözümü, ince ayar öncesinde bile modelin ağırlıklarında bir yerde zaten var. **CoT** ve **RL** , modelin bu cevaba daha erken varmasını sağlıyor — yeni bir çözüm yolu icat etmiyor.

$$\text{pass@64}_{\text{RL öncesi}} \approx \text{pass@2}_{\text{RL sonrası}}$$

RL öncesi 64 denemede bulunan doğru oran, RL sonrası 1–2 denemede yakalanan oranla örtüşüyor.

Yue ve ark., 2025: arXiv:2504.13837 — RLVR büyük `k`'de `pass@k` 'yı base model'e göre daraltıyor.

Shao ve ark., 2024: RLVR `pass@1`'i artırır ama `pass@k`'yı artırmaz.

QuestA, 2025: Zor problemlerde aynı `pass@k` bozulmasını doğruluyor.

BÖLÜM 4 · ÖZET

Post-training erişimi keskinleştiriyor. İspat kalitesini doğrudan hedefleyen yöntem yok.

SFT davranışı aktarıyor, RLVR doğru cevabı pekiştiriyor, araçlar bir hata sınıfını tamamen siliyor. Adımları puanlayan PRM ise hâlâ çalışmıyor — ölçtüğümüzle önemsediğimiz arasındaki boşluk bu yüzden duruyor.

Bölüm 4 bitti

EK BÖLÜM

Sunumun yol haritası

Giriş: LLM'ler ve training pipeline'ı
Model ne yapıyor, pretraining ve post-training nasıl işliyor

Modellerin neler başardıklarına dikkatli bir bakış
Sonuçlar gerçekten ne gösteriyor, ne göstermiyor

Matematik neden özel
Doğrulama asimetrisi

Pretraining tarafındaki gelişmeler
Veri, tokenization, uzun context (bağlam)

Post-training tarafındaki gelişmeler
SFT, RLVR, GRPO, tool use (araç kullanımı) + işe yaramayan denemeler

Ek bölüm: Reinforcement learning derinlemesine
PPO, GRPO ve process reward model'ler (PRM)

EK BÖLÜM

PPO (proximal policy optimization — yakınsal politika optimizasyonu)

PPO, standart RL algoritması. Adımları:

Model, soruya bir çözüm üretir; doğrulayıcı bir ödül `r` verir.

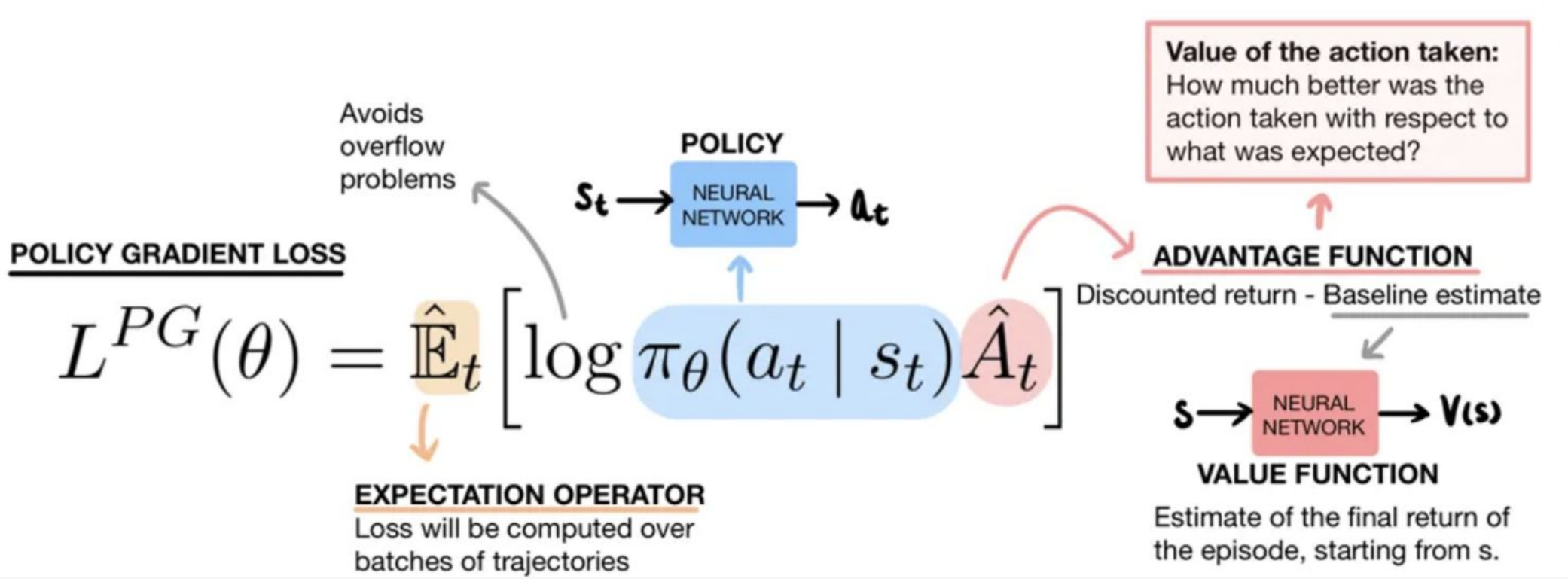
Value network (değer ağı — ayrı, eğitilen bir sinir ağı) her adımda "buradan itibaren beklenen ödül ne olur" diye tahmin eder.

Avantaj: `A = r − V(s)` — gerçekleşen ödülün, değer ağının tahmininden ne kadar iyi/kötü çıktığı.

Model, avantajı pozitif olan adımları güçlendirecek yönde güncellenir — ama eski hâlinden *çok uzaklaşmayacak* şekilde (bu yüzden "proximal").

> **MALIYET**
> Value network, model ağıyla kabaca aynı büyüklükte — bellek ve hesap ~2 katına çıkıyor. Ayrıca tahmini genelde gürültülü: yanlış tahmin edilen değer, yanlış avantaja, yanlış güncellemeye yol açar.

PPO özeti



Kaynak: miro.medium.com üzerinden barındırılan bir PPO kayıp fonksiyonu diyagramı.

EK BÖLÜM

Neden bir de PRM'e ihtiyaç duyuldu

RLVR ödülü *sonuç temelli* : yalnızca son cevap doğruysa 1, değilse 0. Onlarca adımlık bir ispatın hangi adımı hatalıydı bilgisi kayboluyor.

Seyrek sinyal: ödül sadece *son aşamada* bilinir; zincirdeki ara token'lar üretildikleri anda sonuçtan tamamen habersizdir — kredi atama zorlaşır.

Yanlış nedenle doğru cevap : model tesadüfen doğru sonuca ulaşabilir; sonuç ödülü bunu ayırt edemez.

Fikir: **process reward model (PRM)** (süreç ödül modeli) her adımı tek başına değerlendirsin; hatalı adım tam olarak işaretlensin.

> **VAAT**
> Daha yoğun bir eğitim sinyali, ve arama/en-iyi-n seçimi gibi çıkarım zamanı yöntemleri için adım bazında bir doğrulayıcı.

GRPO: neden bu kadar hızlı yayıldı

Önceki slayttaki maliyeti hatırla: **PPO**, her durumun beklenen değerini tahmin eden ayrı bir **value network** (değer ağı) gerektiriyordu — bellek ve hesabı yaklaşık ikiye katlayan bir sinir ağı.

GRPO'nun (group relative policy optimization — grup göreli politika optimizasyonu) fikri: aynı soru için G tane çözüm örnekle, değer tahmini yerine *grubun kendisini* referans al.

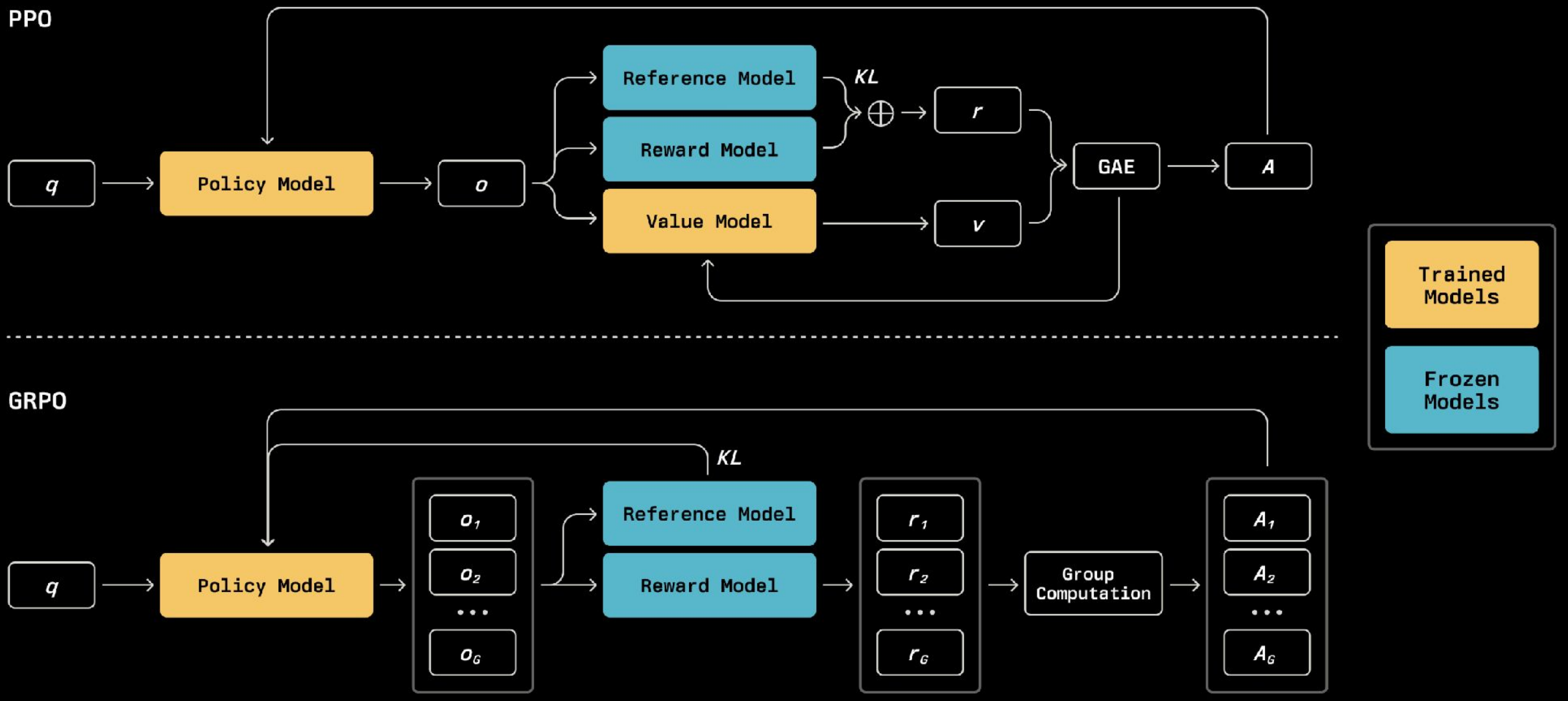
$$\hat{A}_i = \frac{r_i - \text{mean}\{r_1, \ldots, r_G\}}{\text{std}\{r_1, \ldots, r_G\}}$$

"Bu çözüm, aynı soruya verdiğim diğer çözümlerden daha mı iyiydi?"

DÜRÜST YORUM

Bu derin bir buluş değil — bir *maliyet* buluşu. Ama RL çalıştırmayı yarı fiyatına indirmesi, RLVR'ın sektöre bu hızda yayılmasının muhtemelen en büyük tek sebebi.

FROM PPO TO GRPO



Kaynak: research.perplexity.ai/articles/rl-training-for-math-reasoning · LLM'ler ve matematiksel ispat

Toparlarsak

İlerleme gerçek

Ve mekanizması gizemli değil: seçilmiş veri, doğrulanabilir ödül, uzun arama, dış araçlar.

Ama ölçtüğümüz şey çoğunlukla ispat değil

Nihai cevap. Bu boşluk kapanıyor, henüz kapanmadı.

Matematiğin hızı kendine özgü

Ucuz doğrulayıcı olmayan alanlara doğrudan genellemeyin.

Pretraining veriniz en kritik nokta

Post-training ne kadar iyi olursa olsun, pretraining verisinin kalitesi ve kapsamı tavanı belirlemeye devam ediyor.

SIZI NE IKNA EDERDI?

Bir modelin matematikte gerçekten "iyi" olduğuna sizi ne ikna ederdi? Cevabınız bir skor değilse — ki muhtemelen değil — o zaman bugün takip ettiğimiz sayılar yanlış sayılar.

EK · SUNUM SONRASI

İzini sürmek isteyenler için

VERI VE PRETRAINING

OpenWebMath — matematiksel web derlemi
DeepSeekMath — sınıflandırıcıyla veri madenciliği, GRPO
Lean mathlib — formal teorem kütüphanesi

POST-TRAINING

STaR — kendi çıktısıyla bootstrap
"Let's Verify Step by Step" — PRM'in umutlu dönemi
DeepSeek-R1 ve R1-Distill raporu

ÖLÇME VE ELEŞTIRI

MathArena — USAMO 2025 insan puanlaması
FrontierMath — araştırma seviyesi ölçüt

ŞÜPHECI TARAF

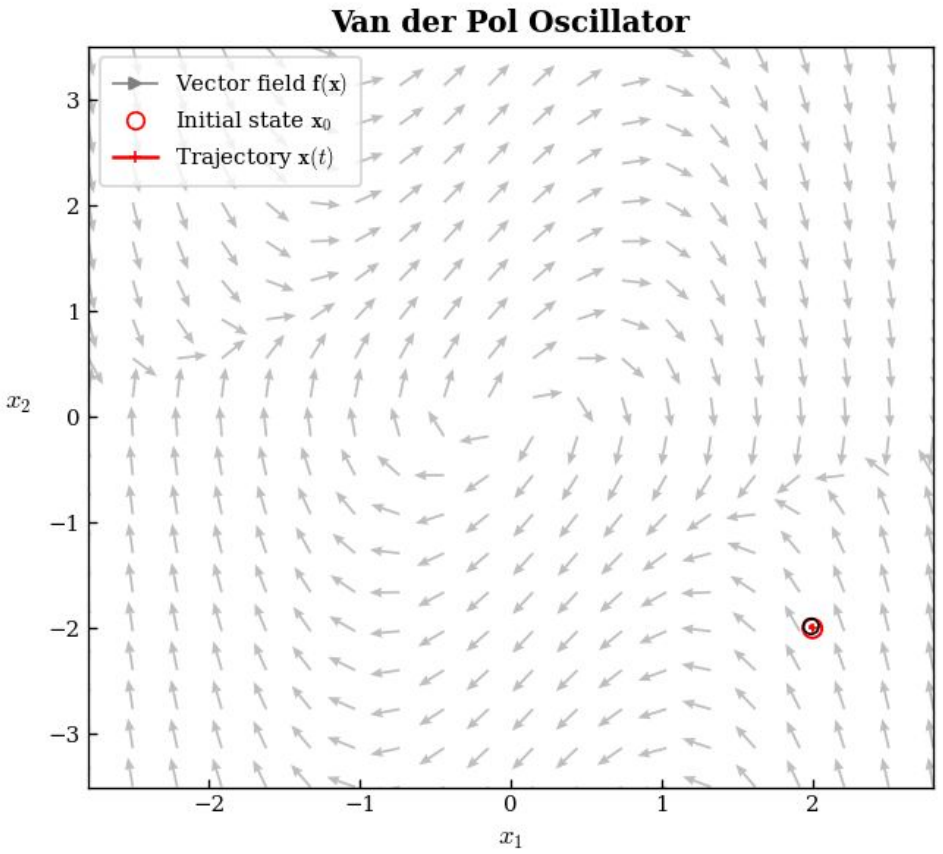
"LLMs Cannot Self-Correct Reasoning Yet" (Huang vd.)
RLVR ve pass@k daralması üzerine çalışmalar
Düşünce zinciri sadakati üzerine deneyler

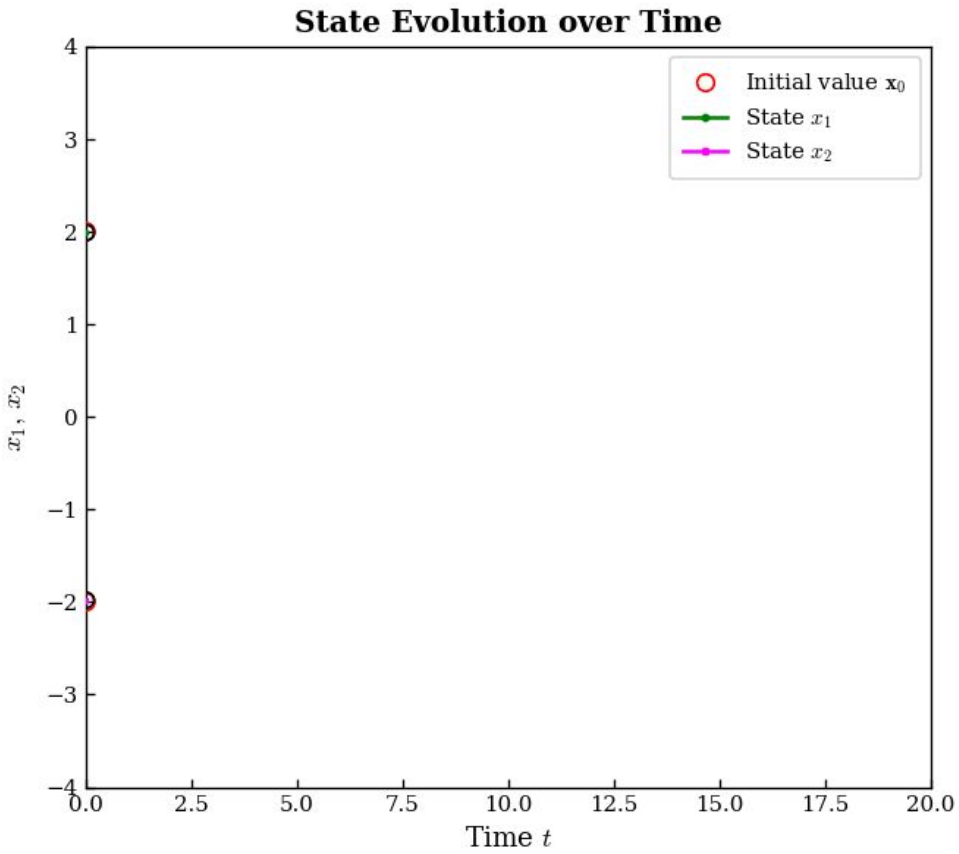
Bu liste tam değil ve alan hızlı ilerliyor; sunum tarihinden sonra çıkan işleri kontrol edin.

Doktora çalışmalarım

Continuous-time modelling in ML

Real world is continuous

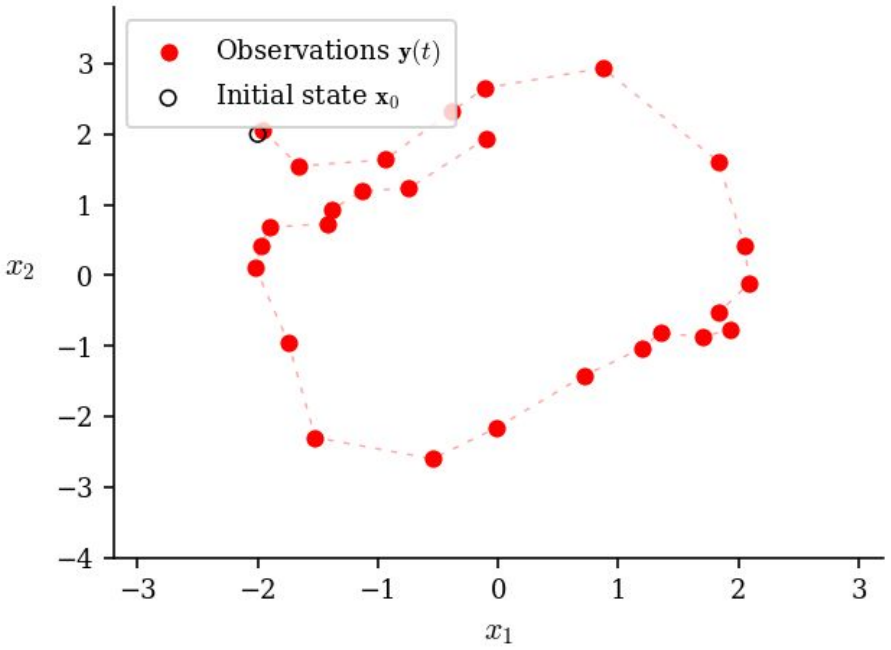




Continuous-time modelling in ML

Discrete observations

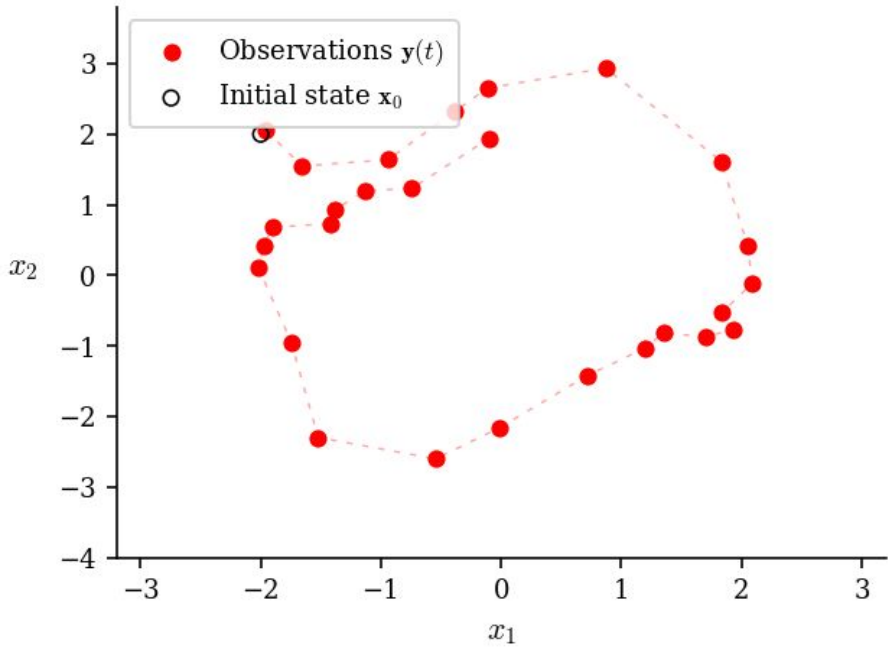
$$\mathbf{y}_i = \mathbf{x}(t_i) + \varepsilon$$



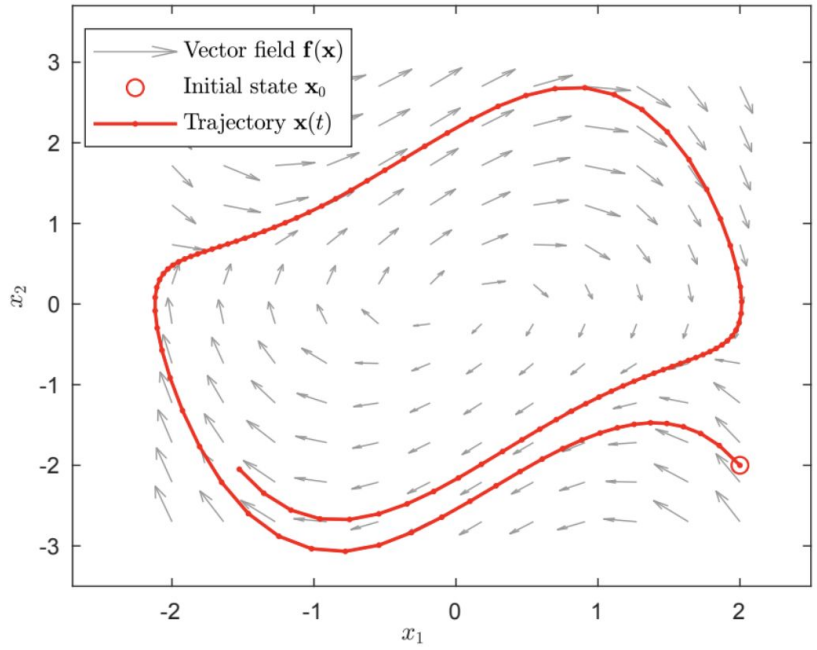
Continuous-time modelling in ML

Discrete observations → continuous-time models

$$\mathbf{y}_i = \mathbf{x}(t_i) + \varepsilon$$



$$\mathbf{x}(t_i) = \mathbf{x}_0 + \int_0^{t_i} \mathbf{f}(\mathbf{x}(\tau))d\tau$$



Continuous-time modelling in ML

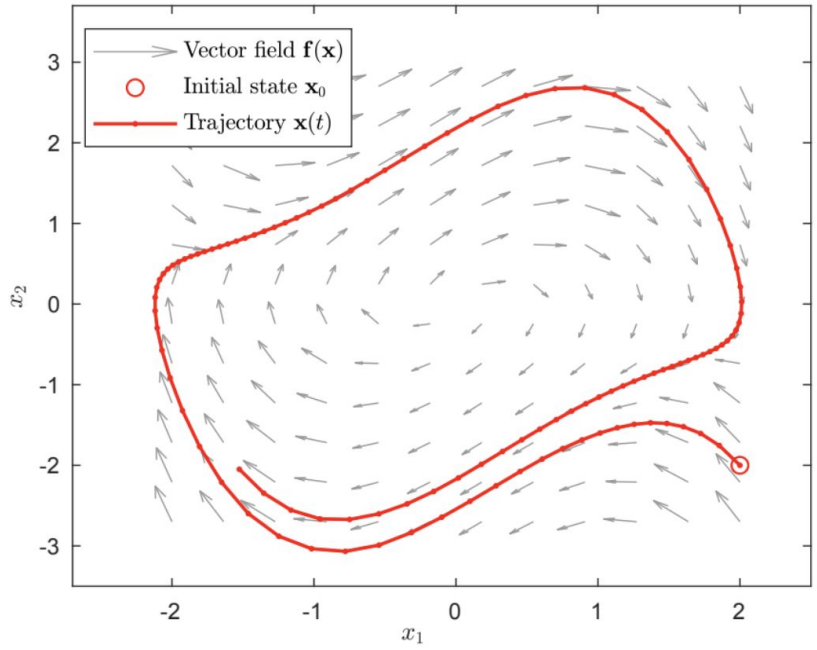
Gaussian process and neural network approximations

$$\mathbf{f}(\mathbf{x}, t) \approx \mathtt{GP}(\mathbf{x}, t) \text{ [1]}$$

$$\mathbf{f}(\mathbf{x}, t) \approx \mathtt{NN}(\mathbf{x}, t) \text{ [2]}$$

VAE stuff

$$\mathbf{x}(t_i) = \mathbf{x}_0 + \int_0^{t_i} \mathbf{f}(\mathbf{x}(\tau))d\tau$$

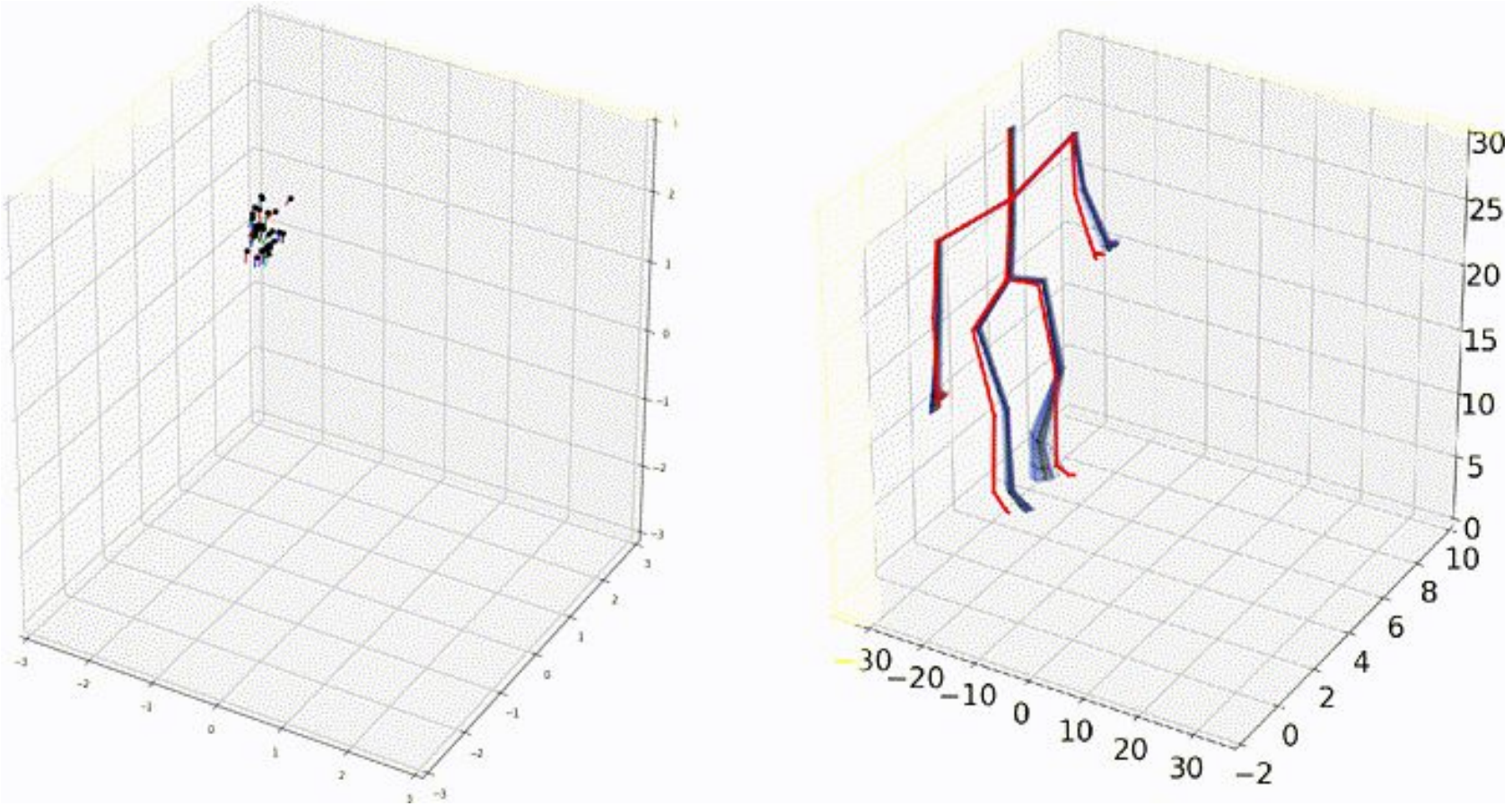


[1] Heinonen, M*., **Yildiz, C***., et al. Learning unknown ODE models with Gaussian processes. ICML 2018

[2] **Yildiz, C**., et al. Ode2vae: Deep generative second order odes with bayesian neural networks. NeurIPS 2019

Continuous-time modelling in ML

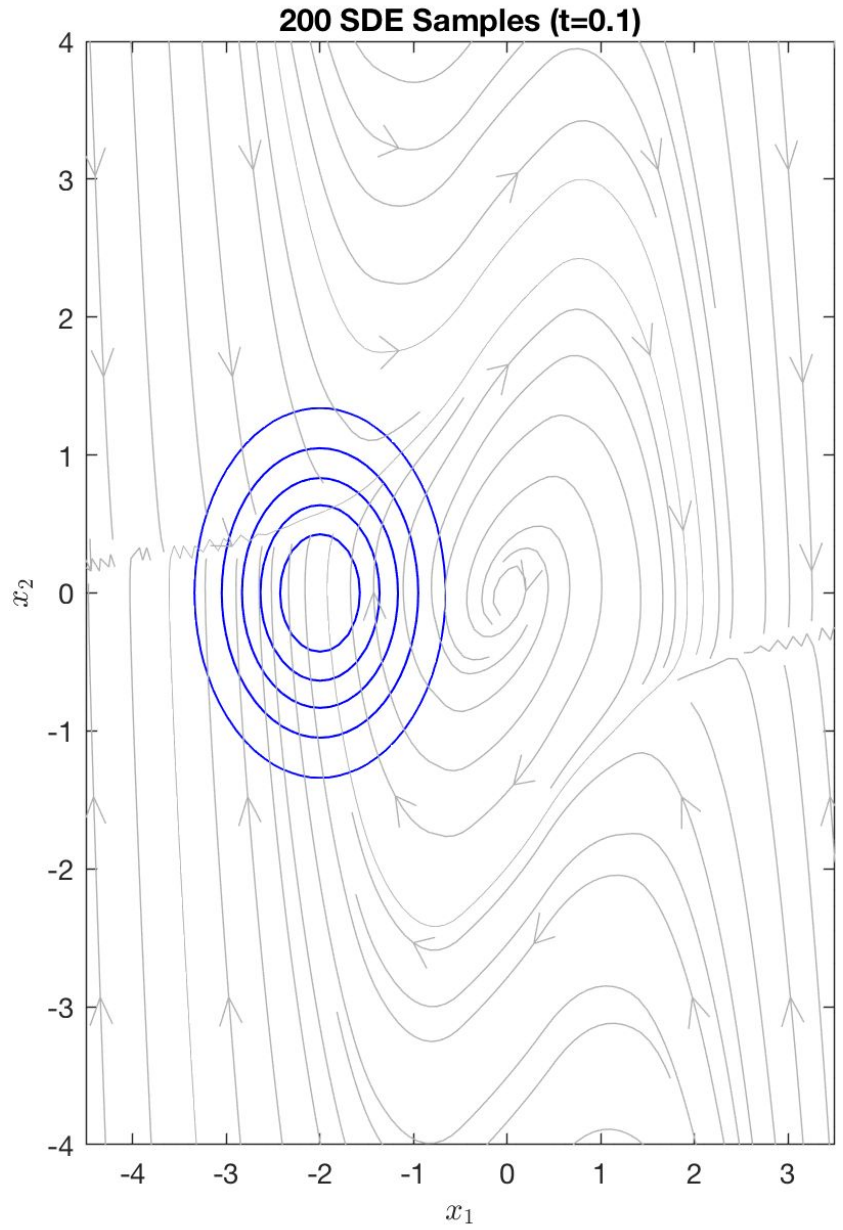
Generative modeling



[2] **Yildiz, C**., et al. Ode2vae: Deep generative second order odes with bayesian neural networks. NeurIPS 2019

Continuous-time modelling

Stochastic systems



[3] **Yildiz, C**., et al. Learning stochastic differential equations with gaussian processes without gradient matching. MLSP, 2018

Continuous-time modelling in ML

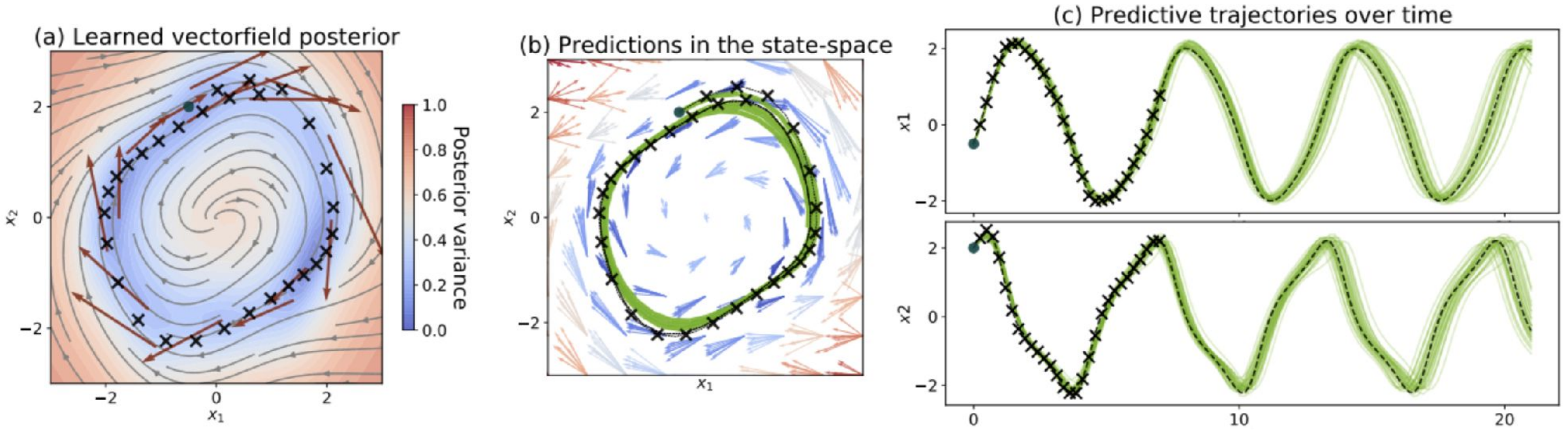
Reinforcement learning



[4] **Yildiz, C**., et al. Continuous-time model-based reinforcement learning. ICML 2021

Continuous-time modelling in ML

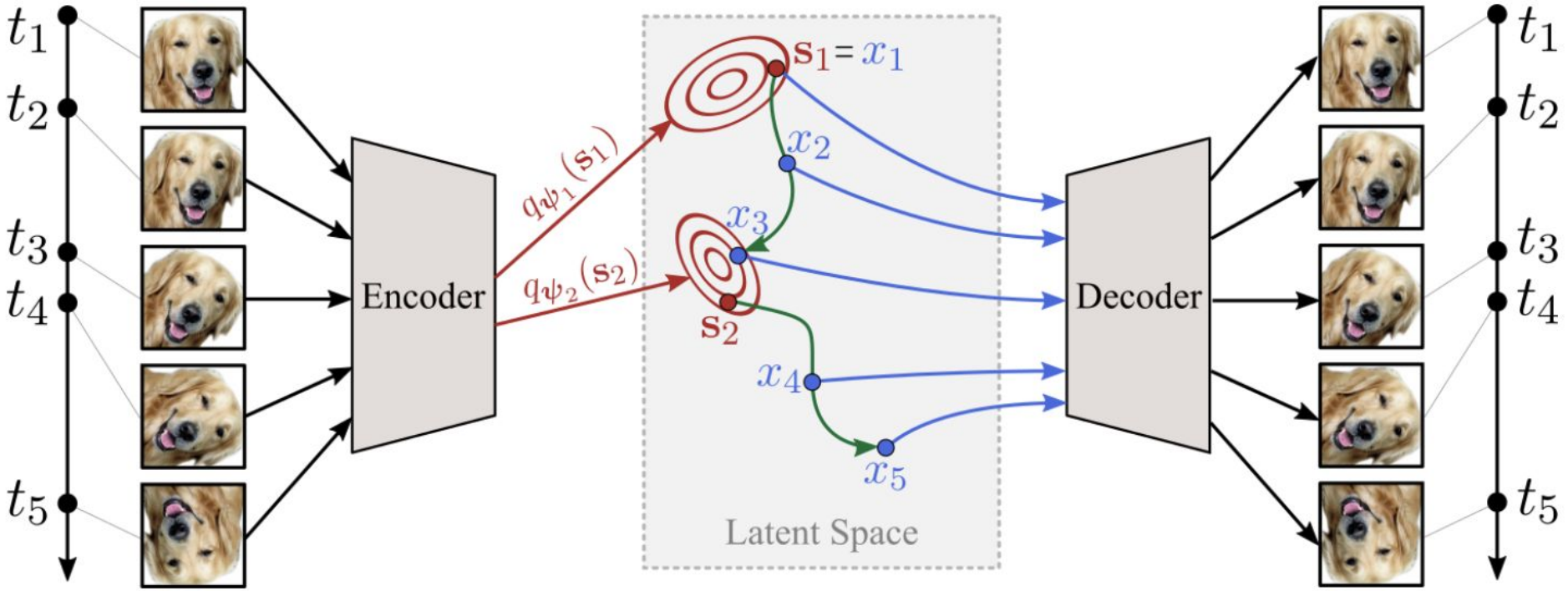
Vector field uncertainties



[5] Hegde, P., **Yildiz, C**., et al. "Variational multiple shooting for Bayesian ODEs with Gaussian processes." UAI, 2022.

Continuous-time modelling in ML

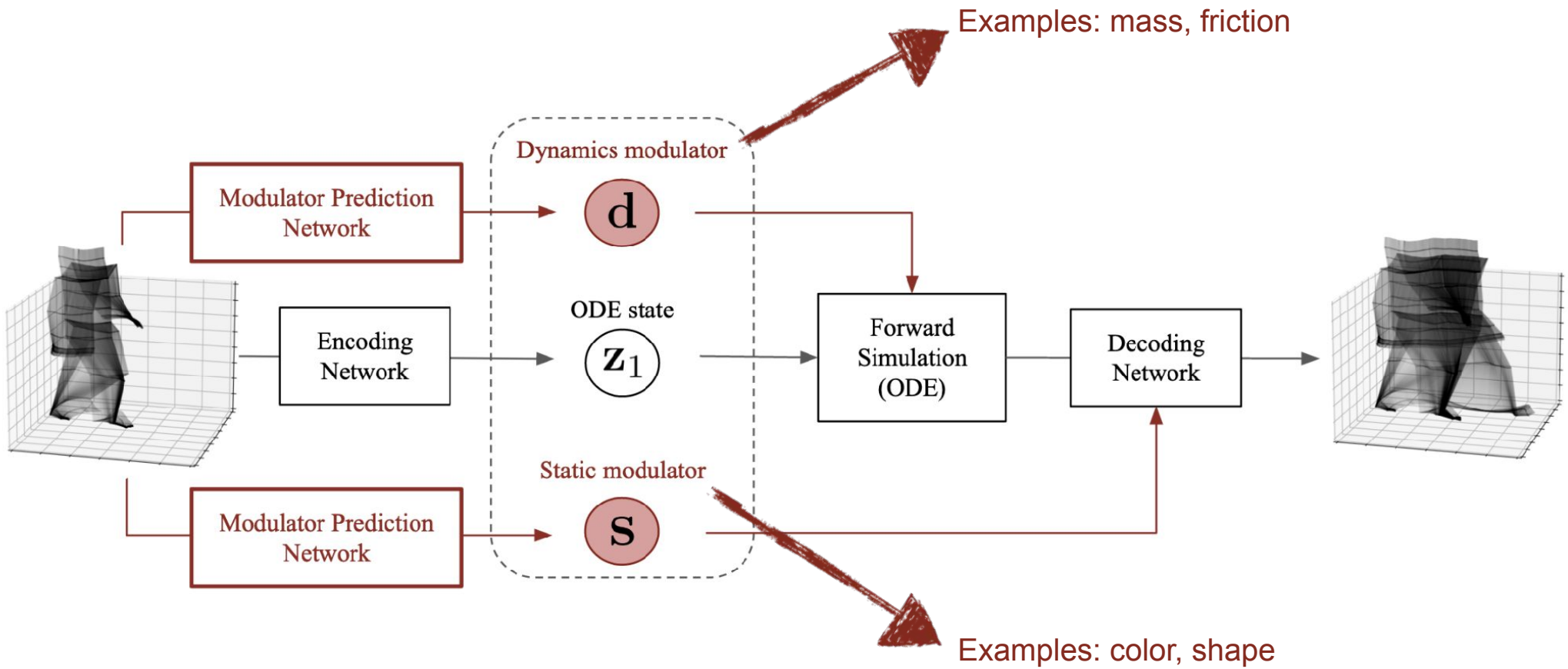
Improved optimisation + transformer encoder



[6] Iakovlev, V., **Yildiz, C**., et al. Latent neural ODEs with sparse bayesian multiple shooting. ICLR 2023

Continuous-time modelling in ML

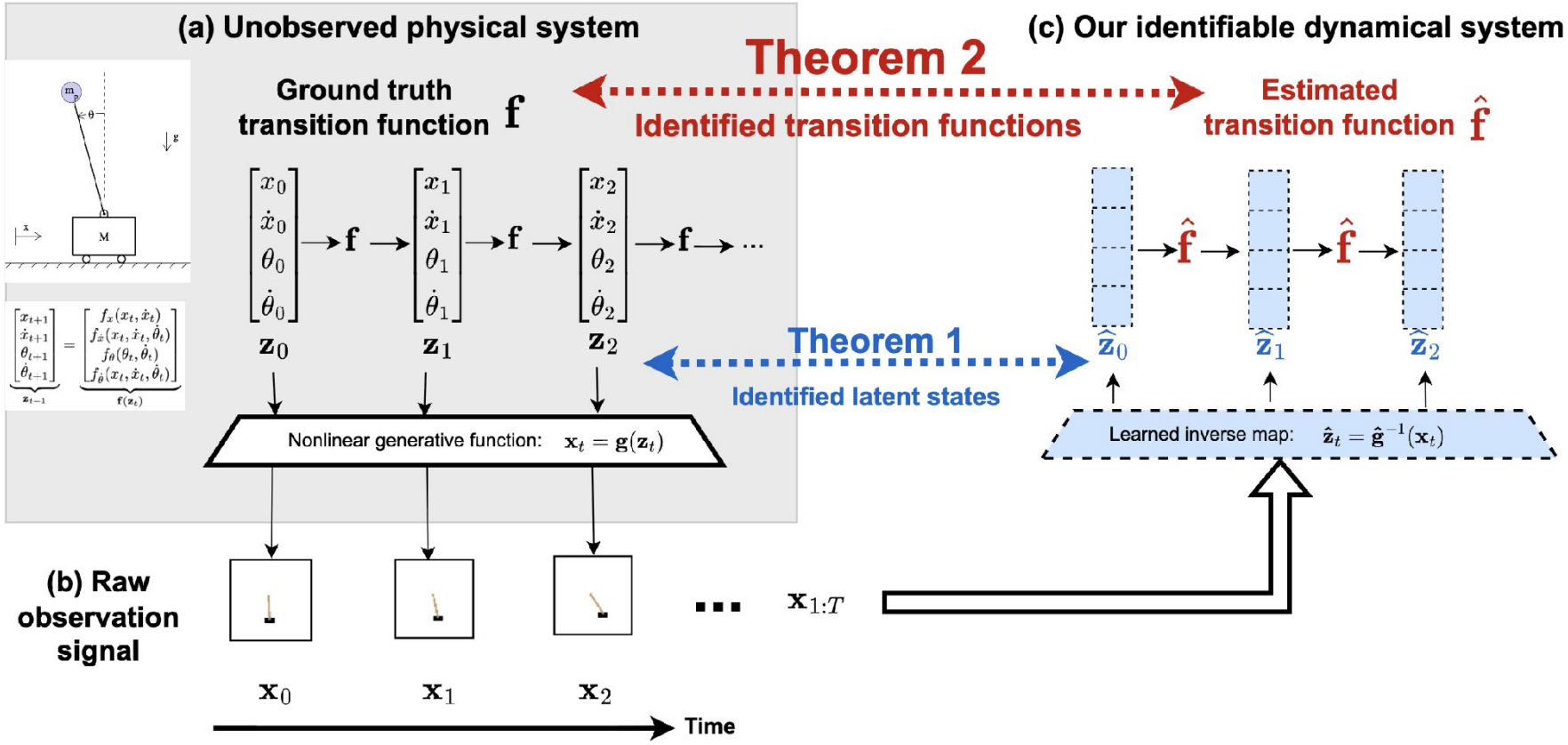
Modulated systems



[7] Auzina, I.A., **Yildiz, C.**, et al. Modulated Neural ODEs. NeurIPS 2023.

Continuous-time modelling in ML

Identifying latent dynamics



[8] Hızlı, Ç., **Yildiz, C**., et al. Identifying latent state transition in non-linear dynamical systems. ICLR 2025