Self-conditioned Flow Map Language Models via Fixed-point Flows

Compress prediction refinement and flow evolution

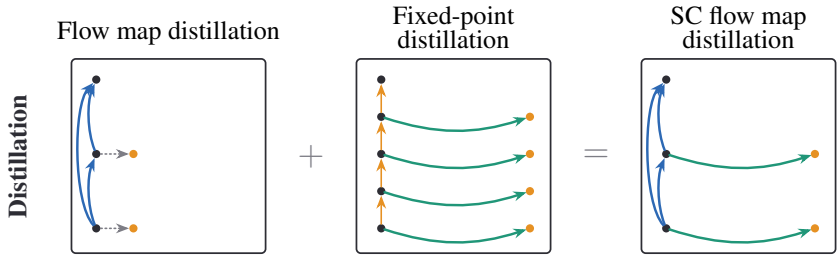
Paper by Jaehoon Yoo, Wonjung Kim, Floor Eijkelboom,
Chanhyuk Lee, Nicholas M. Boffi, Seunghoon Hong, and Jinwoo Kim

KAIST University of Amsterdam Carnegie Mellon University

Research discussion | 14 October 2026
arXiv:2607.00714v2, revised 1 October 2026

The method compresses two kinds of computation

- Self-conditioning reuses a denoising estimate across flow steps.
- Interpret this estimate as an iterate of a local fixed-point solver.
- Learn to jump both across refinement iterations and across flow time.



Blue: flow-map jump. Green: fixed-point jump.

Paper, Fig. 1 (bottom row), PDF p. 2.

A denoiser determines a continuous text flow

$$I_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1, \qquad \mathbf{x}_0, \mathbf{x}_1 \in \mathbb{R}^{L\times d}$$

$$D_t(\mathbf{x}) = \mathbb{E}[\mathbf{x}_1 \mid I_t = \mathbf{x}], \qquad b_t(\mathbf{x}) = \frac{D_t(\mathbf{x}) - \mathbf{x}}{1-t}$$

Generation by local updates

$$\mathbf{x}_{t_{i+1}} = \mathbf{x}_{t_i} + (t_{i+1} - t_i) b_{t_i}(\mathbf{x}_{t_i})$$

- L: sequence length; d: embedding dimension; t: flow time.
- Noise $\mathbf{x}_0$ evolves to text embeddings, then a decoder produces tokens.
- A flow map $X_{s,t}$ learns an entire interval in one evaluation.

Paper, Sec. 2, Eqs. (1)–(5), PDF pp. 2–3.

Self-conditioning trains an initial guess and a correction

$$\mathbf{z}^1 = \hat{D}_t(I_t, \mathbf{0}), \qquad \mathbf{z}^2 = \hat{D}_t(I_t, \mathrm{sg}(\mathbf{z}^1))$$
$$\mathcal{L}_\mu = \tfrac{1}{2}\mathbb{E}\|\mathbf{z}^1 - \mathbf{x}_1\|^2 + \tfrac{1}{2}\mathbb{E}\|\mathbf{z}^2 - \mathbf{x}_1\|^2$$

- sg: stop gradient. Expectations here also include flow time.
- Both predictions target the same clean embedding.
- If $\mathbf{z} \perp \mathbf{x}_1 \mid I_t$ and second moments are finite, the unrestricted population optimum is $D_t(I_t)$ on the training support.

Why it can help in practice

A finite learned network can use its first prediction to improve its next one.

Paper, Sec. 3.1, Eq. (7), Proposition 3.1; PDF p. 4 and Appendix A.1, p. 13.

At fixed time and state, prediction becomes an iteration

$$\boxed{\mathbf{z}^{j+1} = \hat{D}_t(\mathbf{x}, \mathbf{z}^j)} \qquad \mathbf{z}^\star = \hat{D}_t(\mathbf{x}, \mathbf{z}^\star)$$

Sufficient condition: contraction on a nonempty closed set O

$$f(O) \subseteq O, \quad \|f(\mathbf{z}) - f(\mathbf{z}')\| \le \eta \|\mathbf{z} - \mathbf{z}'\|, \quad 0 \le \eta < 1$$

$$\|\mathbf{z}^j - \mathbf{z}^\star\| \le \eta^j \|\mathbf{z}^0 - \mathbf{z}^\star\|$$

- t and $\mathbf{x}$ stay fixed; only the denoising estimate changes.
- Convergence is to the learned fixed point; better generation is tested empirically.

Paper, Eq. (10), Proposition 3.3, PDF pp. 4–5; Appendix A.3, pp. 15–17.

Warm starts reduce the work needed to find a fixed point

Sampling	Initial prediction at time t_i	Inner work
Cold start	$\mathbf{z}_{t_i}^0 = \mathbf{0}$	Iterate from scratch
Warm start	$\mathbf{z}_{t_i}^0 = \hat{\mathbf{z}}_{t_{i-1}}^\star$	Reuse previous estimate
Conventional SC	Warm start	One iteration

$$j \geq \left\lceil \frac{\log(\|\mathbf{z}^0 - \mathbf{z}^\star\|/\varepsilon)}{\log(1/\eta)} \right\rceil$$

- Bound applies when $0 < \eta < 1$ and initial error exceeds ε.
- If neighboring fixed points are close, the previous estimate is useful.
- With enough iterations in the contraction domain, both starts reach the same fixed point.

Paper, Sec. 3.3, PDF p. 6; Proposition A.5 and Eq. (37), PDF p. 18.

Solving the inner problem removes the history variable

$$D_t^\star(\mathbf{x}) := \mathbf{z}^\star, \qquad b_t^\star(\mathbf{x}) = \frac{D_t^\star(\mathbf{x}) - \mathbf{x}}{1-t}$$

$$\dot{\mathbf{x}}_t = b_t^\star(\mathbf{x}_t), \qquad t < 1$$

- A unique fixed point makes the velocity depend only on $(t, \mathbf{x})$.
- In the ideal case $D^\star = D$, this recovers the true data flow.
- Where this ODE has unique solutions, it defines a flow map:

$$X_{s,t}^\star = X_{u,t}^\star \circ X_{s,u}^\star$$

Paper, Propositions 3.4–3.5, Eqs. (13)–(20), PDF pp. 5–6. ODE uniqueness is a separate condition.

Fixed-point distillation replaces repeated refinement

$$\hat{D}_t(\mathbf{x}, \cdot) \xrightarrow{\text{iterate}} \mathbf{z}^\star \xrightarrow{\text{distill}} D_t^\star(\mathbf{x})$$

Basic regression target

$$\mathcal{L}(D^\star) = \int_0^1 \mathbb{E}\|D_t^\star(I_t) - \mathbf{z}^\star\|^2 \, dt$$

- Actual offline method: CDEQ consistency distillation, yielding ELF*.
- Teacher targets: 20 Anderson-accelerated iterations.
- Inference: one student evaluation per flow step, with self-conditioning input set to zero.

Paper, Eq. (18), PDF p. 6; Appendix B.2, PDF p. 21.

A two-time denoiser represents large flow-map jumps

$$v_{s,t}(\mathbf{x}) = \frac{X^\star_{s,t}(\mathbf{x}) - \mathbf{x}}{t - s}, \qquad \delta_{s,t}(\mathbf{x}) = \mathbf{x} + (1 - s)v_{s,t}(\mathbf{x})$$

Recover the next flow state

$$\boxed{X^\star_{s,t}(\mathbf{x}) = \frac{1 - t}{1 - s}\mathbf{x} + \frac{t - s}{1 - s}\delta_{s,t}(\mathbf{x})}$$

- On the diagonal: $\delta_{t,t}(\mathbf{x}) = D^\star_t(\mathbf{x})$.
- Off the diagonal: predict the effect of a finite time interval.
- At the endpoint extension: $X^\star_{0,1}(\mathbf{x}_0) = \delta_{0,1}(\mathbf{x}_0)$.

Paper, Eqs. (21)–(24), PDF pp. 6–7. Formal identities use $t < 1$; training also includes $(0, 1)$.

Consistency makes one jump agree with two smaller jumps

$$\mathcal{T}(\delta)_{s,u,t}(\mathbf{x}) = \gamma\delta_{s,u}(\mathbf{x}) + (1-\gamma)\delta_{u,t}(X^{\star}_{s,u}(\mathbf{x}))$$

$$\gamma = \frac{(1-t)(u-s)}{(1-u)(t-s)}$$

Two complementary losses

$$\mathcal{L}(\delta) = \mathbb{E}\|\delta_{s,t} - \mathrm{sg}(\mathcal{T}(\delta)_{s,u,t})\|^2 + \mathbb{E}\|\delta_{t,t} - D^{\star}_t\|^2$$

First term: compress flow evolution. Second: anchor local denoising.

Paper, Eqs. (25)–(27), PDF p. 7. In implementation, the intermediate state is computed from the current student under stop-gradient (Appendix B.3, p. 22).

Offline and online routes supply the same kind of target

Offline: two stages

$$\text{ELF} \longrightarrow \text{ELF}^\star \longrightarrow \text{FMLM}^\star$$

- First learn the fixed-point denoiser.
- Use one ELF* evaluation as the diagonal target.

Online: joint compression

$$\text{ELF} \xrightarrow{K \text{ FPIs}} \mathbf{z}^K$$

- Estimate the target during training.
- Train FMLM* without a separate ELF* model.

K counts teacher calls during training, including the initial zero-conditioned call.

Both final models generate with a two-time student.

Paper, Sec. 3.4, PDF p. 7; Appendix B.3, PDF p. 22.

Generation quality must be evaluated together with diversity

- OpenWebText; sequence length 1024.
- gPPL ↓: generated text scored by pretrained GPT-2 Large.
- Mean per-sample unigram entropy: target is near data entropy, 5.44 nats.
- Analysis: 256 samples. Distillation evaluation: 1024 samples.

Why the entropy column matters

Low gPPL can come from repetitive text. DFM + ESD at one step:

$$\text{gPPL} = 5.33, \qquad \text{entropy} = 0.26$$

Paper, experiment setup, PDF p. 7; Table 1, PDF p. 9. gPPL is an external evaluator score, not the generator's held-out likelihood.

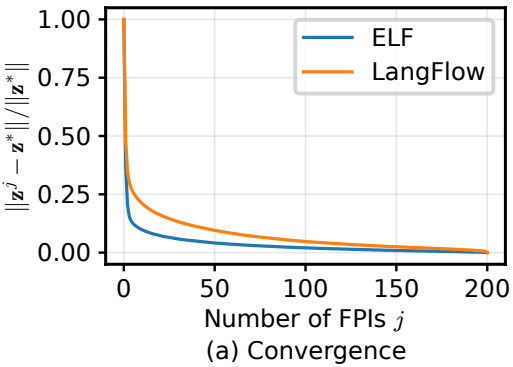
The experiments separate mechanism tests from distillation

Question	Experimental comparison
Q1: fixed-point behavior	ELF / LangFlow at fixed $(t, \mathbf{x})$; vary refinement count.
Q2: initialization	Warm versus cold starts at 1 and 100 iterations.
Q3: remove SC state	ELF* versus its ELF teacher, both with 32 flow steps.
Q4: few-step generation	FMLM* versus teacher and diffusion / flow-map baselines.

Distillation teacher: ELF-B variant with GPT-2 Large embeddings for tokenizer matching. Students: 5 epochs each, batch 512, 8 B200 GPUs.

Paper, Secs. 4.1–4.2, PDF pp. 8–9; Appendix B.2–B.3, pp. 21–22.

Q1: repeated predictions approach a stable reference



Paper, Fig. 2(a), PDF p. 8; Appendix B.1, PDF p. 21.

- Average over 256 examples and ten times $t = 0, 0.1, \ldots, 0.9$.
- Damped iteration:
$$\mathbf{z}^{j+1} = 0.3\mathbf{z}^j + 0.7\hat{D}_t(\mathbf{x}_t, \mathbf{z}^j)$$
- Reference: $\mathbf{z}^\star \approx \mathbf{z}^{200}$ from the same run.
- The trajectory supports stabilization; the final zero distance is by construction.

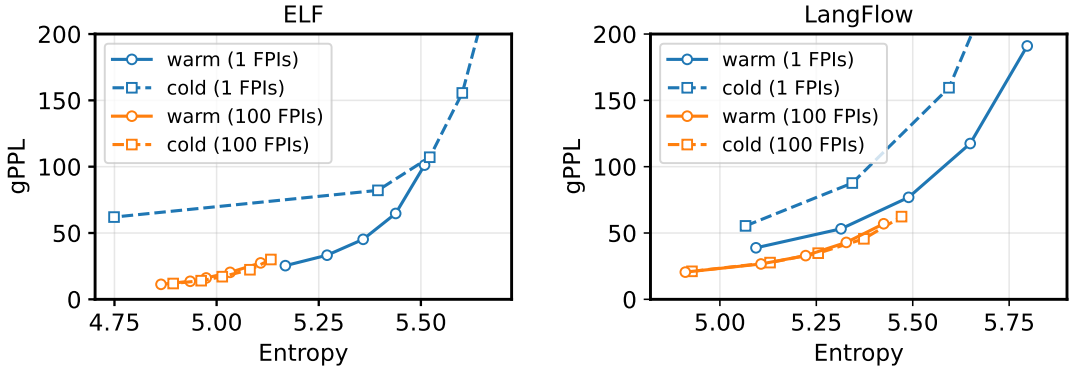
Q1: more refinement improves gPPL with extra computation

FPIs	ELF gPPL ↓	Entropy	LangFlow gPPL ↓	Entropy
1	101.19	5.51	77.26	5.49
2	66.50	5.42	57.99	5.42
3	56.67	5.37	52.63	5.40
4	50.25	5.33	49.37	5.38
5	46.31	5.31	47.27	5.36

- Warm starts, no damping; ELF: 32 flow steps; LangFlow: 64.
- Entropy declines modestly and remains near the data level of 5.44.
- Inner denoiser calls scale as $N \times K$: refinement is useful computation that distillation aims to compress.

Paper, Table 2, PDF p. 22; Appendix B.1, p. 21. ELF sampling $\gamma = 0$; LangFlow temperature 1.0.

Q2: enough refinement closes the warm–cold start gap



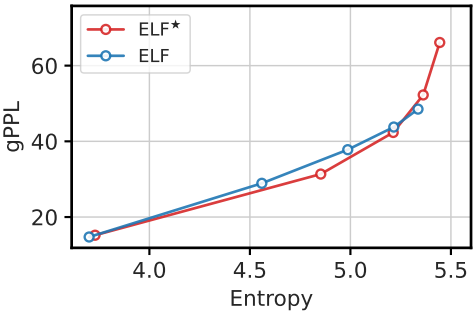
At 1 FPI, warm starts are better. At 100 FPIs, the frontiers overlap.

ELF: 32 flow steps; LangFlow: 64. Damping $\alpha = 0$ for 1 FPI and 0.3 for 100 FPIs. Compare initialization within each budget.

Paper, Fig. 3, PDF p. 8; Appendix B.1, PDF p. 21.

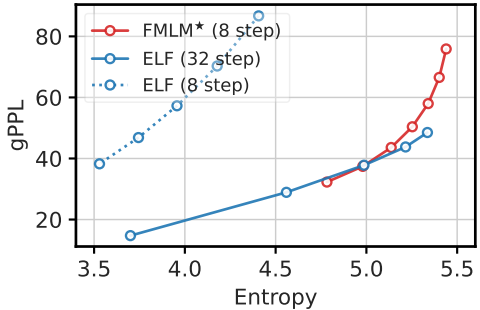
Q3–Q4: each compression retains much of the teacher's frontier

Remove the SC state



ELF* and ELF both use 32 steps; their frontiers are close.

Reduce flow steps



8-step FMLM* approaches 32-step ELF, with a gap at high entropy.

Paper, Figs. 4–5, Sec. 4.2, PDF p. 9. Step reduction is reported; a 4x wall-clock speedup is not measured here.

Q4: FMLM* improves the few-step quality–diversity tradeoff

Each cell: gPPL ↓ / entropy in nats. Data entropy: 5.44.

Model	1 step	2 steps	4 steps
Duo + DCD	5743.29 / 6.02	891.16 / 5.41	250.86 / 5.37
MDLM + SDTT	1260.86 / 5.26	877.22 / 5.34	339.73 / 5.38
FMLM	168.30 / 5.17	133.29 / 5.25	111.31 / 5.26
DFM + ESD	5.33 / 0.26	108.91 / 5.15	77.08 / 5.27
FMLM* offline	**113.91 / 5.37**	**93.66 / 5.46**	**73.86 / 5.35**
FMLM* online	117.89 / 5.39	98.96 / 5.47	80.26 / 5.44

Single-step gPPL drops 32.3% versus FMLM, with entropy closer to data.

Paper, Table 1 (selected rows), PDF p. 9. Online uses 8 training FPIs. Two-/four-step FMLM* use sampling $\gamma = 0.75/1.0$ (Appendix B.3, p. 22).

Eight online FPIs offer a useful training-cost compromise

Target route	1-step gPPL	2-step gPPL	4-step gPPL	Time
Online, 1 FPI	176.31	149.51	120.35	18 h
Online, 2 FPIs	141.97	115.25	89.32	18 h
Online, 4 FPIs	123.28	101.27	81.27	20 h
Online, 8 FPIs	117.89	98.96	80.26	23 h
Online, 16 FPIs	116.63	98.61	82.50	32 h
Offline	113.91	93.66	73.86	82 h

- 8 FPIs: about 72% less reported training time than offline.
- Single-step gPPL is 3.5% higher; entropy is 5.39 versus 5.37.
- 16 FPIs bring little improvement and worsen the 4-step result.

Paper, Table 3, PDF p. 23. Times compare distillation routes on the reported hardware; teacher pretraining is not separately itemized.

Local refinement helps; semantic quality remains a limitation

Evidence from fixed-state predictions

At $t = 0.25$, three ELF iterations improve grammatical agreement and make references to people more contextually consistent.

Evidence from complete generated samples

The 1-, 2-, and 4-step examples still contain malformed phrases, invented words, and topic drift, including in the 4-step output.

- gPPL and unigram entropy capture only part of text quality.
- Results concern OpenWebText generation at length 1024.
- The method still relies on a separately trained teacher.

Paper, Appendix D, Figs. 6–9, PDF pp. 23–27; Sec. 6, p. 10. Qualitative assessment of the shown samples is this reading's analysis.

The key contribution is a bridge from self-conditioning to flow maps

1. **Interpretation:** historical predictions serve as initial guesses for a local fixed-point iteration.
2. **Method:** distill both the local solver and the time evolution into a two-time denoiser.
3. **Tradeoff:** offline gives the best reported quality; online substantially reduces distillation time.

Keep distinct: convergence to a learned fixed point, agreement with the Bayes target, and empirical generation quality.

Yoo et al., Self-conditioned Flow Map Language Models via Fixed-point Flows.
arXiv:2607.00714v2, 1 October 2026. Main talk ends here; technical backups follow.

Backup: contraction does not imply improving Bayes error

Proposition 3.3 bounds the denoising error by

$$\|\mathbf{z}^j - D_t(\mathbf{x})\| \le \|\hat{D}_t(\mathbf{x}, \mathbf{z}^0) - D_t(\mathbf{x})\| + \frac{\eta - \eta^j}{1 - \eta}\|\mathbf{z}^1 - \mathbf{z}^0\|.$$

The extra term grows with j; the bound is not a monotonic-improvement result.

Teaching counterexample

Target $D = 0$, map $f(z) = z/2 + 1$, initialization $z^0 = 0$:

$$1,\ 1.5,\ 1.75, \ldots \longrightarrow 2.$$

The iteration contracts toward its fixed point while moving away from the target.

Paper, Eqs. (11)–(12), PDF p. 5; Appendix A.3, pp. 15–17. Counterexample is a teaching construction.

Backup: low loss gives a weaker, scale-dependent contraction

Let X be fixed, f continuous, and $Z \sim \mu$ supported on a closed O.

$$\mathcal{L}(f) = p\mathbb{E}\|f(Z) - X\|^2 + (1-p)\mathbb{E}\|f(f(Z)) - X\|^2, \quad 0 < p < 1$$

Assume $\overline{B}(X, r) \subseteq O$, $2r \leq \eta w$, and $\mathcal{L}(f) \leq \min(p, 1-p)\varepsilon r^2$. Define

$$A = \{z \in O : \|f(z) - X\| \leq r\}.$$

Then

$$\mu(A \cap f^{-1}(A)) \geq 1 - \varepsilon,$$

and for $z, z' \in A$ separated by at least w,

$$\|f(z) - f(z')\| \leq 2r \leq \eta\|z - z'\|.$$

This leaves small scales and deterministic invariance of all of A unresolved.

Paper, Appendix A.2, Lemmas A.2–A.3, PDF pp. 13–15. The main convergence analysis assumes strict contraction separately.

Backup: CDEQ trains agreement across inner iterates

$$\tau_j = \varepsilon + (1 - e^{-\rho j})(T - \varepsilon), \quad (\varepsilon, T, \rho) = (0.002, 5.0, 0.3)$$

$$g_j = c_{\text{skip}}(\tau_j)\mathbf{z}^j + c_{\text{out}}(\tau_j)P_j$$

$$\mathcal{L} = 0.8\|g_j - \mathbf{z}^\star\|^2 + 0.2\|g_j - \text{sg}(g_{j-1})\|^2$$

- $c_{\text{skip}} = (\tau - \varepsilon)/(T - \varepsilon)$; $c_{\text{out}} = 1 - c_{\text{skip}}$.
- P_j: two-point Anderson combination of two student evaluations.
- Teacher: 20 Anderson iterations; history 3, mixing 0.9, regularization 10^{-4}.
- Probability 0.25: directly regress g_0 to the terminal target.

Paper, Appendix B.2, PDF p. 21. Inference uses one evaluation at τ_0 with zero self-conditioning.

Backup: flow-map training uses a student-computed target

- Backbone: ELF; four time-prefix tokens split evenly between s, t.
- Diagonal fraction: 0.4 offline, 0.5 online. Of off-diagonal examples, $1/32$ use $(s, t) = (0, 1)$.
- Time marginal: logit-normal $(\mu, \nu) = (-1.5, 0.8)$; u is a midpoint in CDF coordinates.
- Evaluate $\delta_{s,u}$ and $\delta_{u,t}$ under stop-gradient; compute the intermediate state from the student's first jump.
- Compute the two mean-squared losses separately, then sum with equal weights.
- Sample guidance w log-uniformly in $[0.5, 5]$; hold it fixed across inference steps.

Paper, Appendix B.3, PDF p. 22. Semigroup coefficient γ, sampling parameter γ, and guidance weight w are different quantities.

Backup: the remaining baseline rows complete Table 1

Each cell: gPPL ↓ / entropy in nats; OpenWebText, 1024 tokens.

Model	1 step	2 steps	4 steps
Duo + Di4C	370.51 / 3.92	210.22 / 4.63	154.67 / 4.85
MDLM + Di4C	1298.80 / 5.29	758.23 / 5.35	239.27 / 5.40
DFM + PSD	180.29 / 4.91	152.83 / 5.03	122.32 / 5.10
FMLM* offline	113.91 / 5.37	93.66 / 5.46	73.86 / 5.35

- Duo and MDLM provide discrete diffusion comparisons.
- FMLM and DFM provide continuous flow-map comparisons.
- The headline advantage concerns low gPPL together with entropy near the data reference of 5.44.

Paper, Table 1, PDF p. 9. These rows complement the main results frame.

Backup: what the flow-map identities establish

- A unique ODE solution yields composition of flow maps on its interval of existence.
- The two-time formula transfers this composition into a weighted denoiser identity; the weights are fixed by the time endpoints.
- A true two-time denoiser has zero population loss. A zero-loss solution satisfies the diagonal and consistency identities almost surely on training distributions.
- Formal propositions use $t < 1$. Implemented training also includes the endpoint $(0, 1)$.

A useful coefficient check

For $s = 0$, $u = 1/2$, $t = 3/4$, the semigroup weights are $\gamma = 1/3$ and $1 - \gamma = 2/3$.

Paper, Propositions 3.5–3.7, PDF pp. 6–7; proofs in Appendix A.6–A.8, pp. 18–20. Coefficient example is pedagogical.