Proceedings of CONF-APMM 2025 Symposium: Simulation and Theory of Differential-Integral Equation in Applied Physics
DOI: 10.54254/2753-8818/2025.DL27324

# *Multi-Particle Dynamical Systems Modeling Transformers*

**Yuxuan Zhang**

*Beanstalk International Bilingual School, Haidian Academy, Beijing, China*
*yuxuanZhang25hda@bibs.com.cn*

***Abstract.*** Deep neural networks can be understood as discretizing a continuous dynamical system. This literature review analyzes how the multi-particle dynamical system formulation models the self-attention mechanism in transformers. We will discover how this formulation enables the systematic study of the system's convergence towards clusters and its relation with the Kuramoto oscillator.

***Keywords:*** Transformer, Dynamical System, Kuramoto Model, Sumformer

## 1. Introduction

Transformers have gained immense popularity in deep learning in the past few years. They have achieved many advanced results and practical applications in deep learning tasks, such as machine translation, file creation, and image processing. The key to the success of this algorithm relies on the self-attention mechanism, which can encode input data in parallel, improving efficiency and capturing complex correlations across different types of large datasets [1-4].

Building on our understanding on the transformer mechanism and the reasons for its success in partical applications, we are going to review [5], a paper that delves into the mathematical intricacies of the attention mechanism of the transformer from a multi-particle dynamical system perspective. This will enable a systematic study of the attention's convergence towards clusters and its relation with the Kuramoto oscillator in the simplified case of having only 2 particles. We end the paper by introducing a practical application of Transformers to approximate Sumformers.

## 2. Background

### 2.1. Multi-Particle Dynamical System

The subject of multi-particle dynamical systems concerns the evolution in time of systems of $n$ particles. Where a particle is an element from the set $X$, e.g. $X = \mathbb{R}^d$. More precisely, an homogeneous, continuous-time dynamical system $\phi$, can be defined as the continuous map $\phi : \mathbb{R} \times X^n \to X^n$ that satisfies the following two relations for any $x = (x_1, ..., x_n) \in X^n$ and $s, t \in \mathbb{R}$

1. $\phi(s + t, x) = \phi(t, \phi(s, x))$
2. $\phi(0, x) = x$

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2753-8818/2025.DL27324

The partial map fixing the time variable, $\Phi_t : \phi(t, \cdot)$ , is called the system's flow, and the partial map fixing the particles $\xi_x(t) = \phi(\cdot, x)$ is called the system's trajectory.

Smooth dynamical systems can be modeled using ODEs. If we have a system $\phi$ with flows $\{\Phi_t\}$ , then its trajectories $x(t) := \xi_{x_0}(t)$ satisfy the initial value problem, $\dot{x}(t) = f(x(t))$ with $x(0) = x_0$ in which the vector field $f(x) = \frac{d}{dt}|_{t=0} = \Phi_t(x)$ .

In the case of multi-particle dynamical systems, the vector field $f_i$ corresponding to each particle $x_i$ , $i = 1, ..., n$ is the sum of two factors: convection and diffusion. The convection factor concerns the particle movement regardless of other particles, e.g., caused by an external force like gravity. The diffusion factor concerns the particle movement that results from interacting with other particles.

### 2.2. Neural Networks as dynamical systems

Deep neural networks can be thought of as discretizations of continuous dynamical systems [6]. This interpretation has been widely used in the literature in recent years since it allows numerical analysis tools to understand and better design neural networks.

**Resnet, a simple example** The ResNet architecture is a simple example that illustrates well the suitability of the dynamical system interpretation. We start from a simple initial value problem for a first-order ODE,

$$\begin{cases} \frac{dx}{dt} = f(t, x), & t > t_0 \\ x(t_0) = w \end{cases} \tag{1}$$

where $x : [t_0, \infty) \to X^n$ and $w \in X^n$ is the value of the system at $t_0$. As simple as it is, it is not always possible to solve (1) analytically. Nevertheless, numerical methods can find an approximate solution at a given time $T$ . For instance, the Euler method can find an approximate solution to this problem in $L$ steps by discretizing the time variable with step size $\gamma = (T - t_0)/L$ and using the first order approximation of the derivative $\frac{x(t_{l+1}) - x(l_i)}{t_{l+1} - t_l} \approx f(t_l, x(t_l))$ . By doing so, we can estimate $x(T)$ from $x_0 = x(t_0)$ by sequentially estimating $x_{l+1} = x(t_{l+1})$ with the iterative rule

$$x_{l+1} = x_l + \gamma f(t_l, x_l) \tag{2}$$

where $l = \{0, ..., L - 1\}$ , $t_l = t_0 + \gamma l$ . This mathematical formulation of the update rule is equivalent to the formulation of a ResNet layer. Therefore, the function $\gamma f(t_l, x_l)$ can be considered a neural network block, where the $t_l$ time variable indicates the $l$ -th layer and $x_l$ corresponds to the skip connection present in this architecture.

## 3. Transformers as Muli-Particle Dynamic Systems

### 3.1. The transformer architecture

In the same way, we can identify a transformer network with some initial value problem and find the corresponding dynamical system representing its layers. Recall the transformer consists of an attention layer 3 and a feed-forward layer 4:

DOI: 10.54254/2753-8818/2025.DL27324

$$Att_l(x_l, i) = \sum_{j=1}^{N} Softmax(\beta \langle Q_l x_{l,i}, K_l x_{l,j} \rangle) V_l x_{l,j} \tag{3}$$

$$FFN_l(x_{l,i}) = W_l^2 \sigma(W_l^1 x_{l,i} + b_l^1) + b_l^2 \tag{4}$$

The attention layer outputs a linear combination of the system particles, depending on the query $Q$, key $K$, value $V$ matrices, their scalar product, and the temperature value $\beta$. The feed-forward layer outputs the non-linear transformation of the particle $x_i$ according to the matrices $W^2, W^1$, the vectors $b^2, b^1$, and the non-linear function $\sigma$.

### 3.2. Dynamical system formulation

The introduction of dynamical system notation into the transformer problem was first done in [7], modeling the multi-headed self-attention layer as the diffusion term and the feed-forward network as the convection term. This MPDS can be approximated using the Lie-Trotter splitting scheme by the iterative solving of the diffusion and the convection ODEs. Nevertheless, this formulation is still very complex to be analyzed analytically.

### 3.3. Simplification of the problem

To perform a deeper mathematical analysis, [5] relaxes the typical experimental formulation of the transformer and focuses solely on a simplified version of the attention mechanism (3). The simplified problem is the following:

- Each particle lies in the unit sphere $X = \mathbb{S}^{d-1}$. Therefore, after the attention mechanism, the particle is normalized again into the sphere. To study the evolution of a particle position over non-linear manifolds such as the sphere, where we do not have a notion of "sum" or "difference," we rely on the concept of the tangent bundle. The change of position will then be measured using infinitesimal displacements on the point's tangent hyperplane. This is empirically attained by projecting the attention output into the tangent hyperplane at the particle's position, using $P_{x_i}^{\perp} : \mathbb{R}^d \to \mathbb{S}^{d-1}$, $y \mapsto y - \langle y, x_i \rangle x_i$.
- The attention parameters: query $Q$, key $K$, and value $V$ are considered constant across time (i.e., layers) and equal to the identity unless stated otherwise. Therefore, the problem dynamics follow:

$$\dot{x}_i(t) = P_{x_i(t)}^{\perp}\left(\frac{\sum_{j=1}^{n} e^{\beta \langle x_j, x_i \rangle} x_j}{\sum_{j=1}^{n} e^{\beta \langle x_j, x_i \rangle}}\right) \tag{5}$$

- To avoid the asymmetry introduced by the denominator in (5) they also study a variant of the attention mechanism normalized by a factor of $n$, which is also equivalent to studying the case $\beta \ll 1$. This other formulation is:

$$\dot{x}_i(t) = P_{x_i(t)}^{\perp}\left(\frac{\sum_{j=1}^{n} e^{\beta \langle x_j, x_i \rangle} x_j}{n}\right) \tag{6}$$

DOI: 10.54254/2753-8818/2025.DL27324

With these relaxations, the main focus of the paper is to study the evolution of the systems (5) and (6). This study gives us insights into the following questions: what is, mathematically, the attention mechanism? Is attention guaranteed to converge? If so, is this method deterministic?

### 3.4. Formal result

Formally, the main result of the section holds for $d \geq n$ , when the initial configuration is uniformly sampled over the $(\mathbb{S}^{d-1})^n$. The paper states that under these conditions, the unique solution to the Cauchy problem for (5) and (6) converges almost surely and at an exponential rate towards a single particle $x^* \in \mathbb{S}^{d-1}$ . This is, for any particles $i \in \{1, ..., n\}$

$$\|x_i(t) - x^*\| \leq Ce^{-\lambda t} \tag{7}$$

for some $C, \lambda > 0$ . Not only this, but the same results hold for more general formulations of the problem, where the key $K$ and query $Q$ matrices are arbitrary $d \times d$ matrices.

This theorem follows as a direct corollary of a result they call the *cone collapse*. In this previous lemma, they show that any solution to the Cauchy problems (5) and (6) converges and at an exponential rate towards a single particle $x^* \in \mathbb{S}^{d-1}$ if the initial configuration lies in an open hemisphere. This indeed happens with probability one when $d \geq n$ .

When $n$ is fixed and $d \to \infty$ in high dimensional spaces, we can better model the entire dynamics evolution with high probability. This has an intuitive explanation since, when $d \gg n$, any two particles will likely be almost orthogonal. By concentration of measure, the evolution of this system is comparable to the evolution of an orthonormal system, in which a single parameter describes the dynamics. In this simplified model, where all different initial particles are orthogonal, the unique solution to (5) and (6) preserves an equal angle between all different particles whose value depends only on time $t$ and the temperature of the attention mechanism $\beta$ . Equivalently:

$$\angle(x_i(t), x_j(t)) = \theta_\beta(t) \quad i \neq j \tag{8}$$

The most surprising result in this section is that the metastability and phase transition between clustering and non-clustering regimes can also be modeled by this parameter of the system dynamics $\gamma_\beta(t) = cos(\theta_\beta(t))$ when $d \gg n$. Further work in this topic is done in [8].

## 4. Angular Dynamics Equation and the Kuramoto Model

We review some further equations and models mentioned in [5] in section 7.1 and section 7.2 on Angular Dynamics Equation and Kuramoto Model .

### 4.1. Derivation of the Angular Dynamics Equation

In this section, based on [5] section 7.1, we focus on reviewing the dynamics of particles constrained to the unit circle $S^1 \subset \mathbb{R}^2$ ,i.e., the case when $d = 2$ specifically under the dynamics equation 6 (USA).This model, parametrized by angles and related to the celebrated Kuramoto model. Each particle $x_i(t) \in S^1$ can be represented by an angle $\theta_i(t) \in T = [0, 2\pi)$ as follows [5]:

DOI: 10.54254/2753-8818/2025.DL27324

$$x_i(t) = \cos(\theta_i(t))e_1 + \sin(\theta_i(t))e_2,$$

where $e_1 = (1, 0)$ and $e_2 = (0, 1)$ are the standard basis vectors in $\mathbb{R}^2$.

To derive the dynamics of $\theta_i(t)$ under equation 6 (USA), we proceed with the following steps as we follow the section 7.1 in [5]:

Firstly, we assume we only discuss the dynamic equation 6 (USA) . To express this equation in terms of $\theta_i(t)$ , we start by using the relation that $\cos(\theta_i(t)) = \langle x_i(t), e_1 \rangle$ . we then take the time derivative of both sides of the equation which yields the following equation:

$$\dot{\theta}_i(t) = -\frac{1}{n \sin(\theta_i(t))} \sum_{j=1}^{n} e^{\beta \langle x_i(t), x_j(t) \rangle} (\langle x_j(t), e_1 \rangle - \langle x_i(t), x_j(t) \rangle \langle x_i(t), e_1 \rangle).$$

Secondly, noting that $\langle x_i(t), x_j(t) \rangle = \cos(\theta_i(t) - \theta_j(t))$ , we could substitute this relation and rewrite it into the following equation:

$$\dot{\theta}_i(t) = -\frac{1}{n \sin(\theta_i(t))} \sum_{j=1}^{n} e^{\beta cos(\theta_i(t) - \theta_j(t))} [\cos(\theta_j(t)) - \cos(\theta_i(t) - \theta_j(t)) \cos(\theta_i(t))].$$

Thirdly, by using some elementary trigonometric relations, this finally leads us to define the following expression of the equation:

**Definition 1.** The expression of Angular Dynamics under equation 6 is:

$$\dot{\theta}_i(t) = -\frac{1}{n} \sum_{j=1}^{n} e^{\beta \cos(\theta_i(t) - \theta_j(t))} \sin(\theta_i(t) - \theta_j(t)),$$

which governs the evolution of $\theta_i(t)$ in the presence of interactions weighted by $\beta$ and the relative angle between particles.

**Remark 1.** when $\beta = 0$ , the dynamics reduce to the well-known Kuramoto model which describes the synchronization phenomenon in coupled oscillators.

$$\dot{\theta}_i(t) = -\frac{1}{n} \sum_{j=1}^{n} \sin(\theta_i(t) - \theta_j(t)).$$ .

**Remark 2.** Considering the dynamics in definition 1 for $\beta > 0$ , we observe that it can also be written as a gradient flow with interaction energy $E_\beta : T^n \to \mathbb{R}_{\geq 0}$ :

$$E_\beta(\theta) = \frac{1}{2\beta n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} e^{\beta \cos(\theta_i - \theta_j)},$$

which reaches its maximum when all $\theta_i$ align at a single fixed real value in $T = [0, 2\pi)$ .

### 4.2. The Kuramoto Model and Its Genalizations

In this section, we reviewed some Kuramoto models and its genalizations in this section by following section 7.2 in [8] As mentioned in the last section, when $\beta = 0$ , the dynamics in the

DOI: 10.54254/2753-8818/2025.DL27324

previous section simplify to a particular case of the Kuramoto model. In fact, the Kuramoto model could be described in the following definition:

**Definition 2.**The Kuramoto model for oscillator $i$ is given by:

$$\dot{\theta}_i(t) = \omega_i + \frac{K}{n} \sum_{j=1}^{n} \sin(\theta_j(t) - \theta_i(t)),$$

where $K > 0$ is a coupling constant and $\omega_i \in T$ is the intrinsic frequency of oscillator $i$ .

**Remark 3.** In this model in the previous definition, for small $K$ , oscillators do not synchronize over long time. As $K$ exceeds a critical threshold, some oscillators begin to synchronize. For very large $K$ , all oscillators eventually synchronize in the long term.

**Observation 1.**

If all intrinsic frequencies $\omega_i$ are equal to a real number like $\omega$ in the previous definition, we can actually shift and rewrite variables by setting $\theta_i(t) \to \theta_i(t) - \omega t$. This transforms the dynamics in definition 2 into the following gradient flow form:

$$\dot{\theta}(t) = -n \nabla F(\theta),$$

where the energy $F : T^n \to \mathbb{R}_{\geq 0}$ is defined by:

$$F(\theta) = -\frac{K}{2n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \cos(\theta_i - \theta_j).$$

**Remark 4.** This energy $F$ is exactly maximized when all oscillators synchronize (i.e., $\theta_i = \theta^*$ for some fixed $\theta^* \in T$ and for all $i$ in $1, 2..., n$ ), with equilibrium states occurring at the critical points of $F$ .

**Definition 3.**

The Kuramoto model can also be generalized to include more general non-linear interaction functions. In particular, an extension of the form can be written as following:

$$\dot{\theta}_i(t) = \omega_i + \frac{K}{n} \sum_{j=1}^{n} h(\theta_j(t) - \theta_i(t)),$$

where $h : T \to \mathbb{R}$ is a general non-linear function, which captures both the classic Kuramoto model (when $h(\theta) = \sin(\theta)$) in definition 2 and the model in definition 1 as specific cases.

**Example 1.** One example of such a generalization is when $h(\theta) = e^{\beta \cos(\theta)}$ , leading to the interaction function:

$$h_\beta(\theta) = e^{\beta \cos(\theta)} = \sum_{k \in \mathbb{Z}} I_k(\beta) e^{ik\theta},$$

where $I_k(\beta)$ denotes the modified Bessel function of the first kind. \end{example}

DOI: 10.54254/2753-8818/2025.DL27324

## 5. A practical transformer - Sumformer

### 5.1. Definition of Sumformer

A Sumformer is a sequence-to-sequence function $S:\mathbb{R}^{n\times d}\to\mathbb{R}^{n\times d}$, defined for input sequence $X=[x_1,\ldots,x_n]$ as:

$$\Sigma=\sum_{i=1}^{n}\phi(x_i),$$

$$S(X)=[\psi(x_1,\Sigma),\ldots,\psi(x_n,\Sigma)],$$

where $\phi:\mathbb{R}^d\to\mathbb{R}^{d_1}$ and $\psi:\mathbb{R}^d\times\mathbb{R}^{d_1}\to\mathbb{R}^d$ are learnable functions.

### 5.2. Approximation theorem of Sumformer

Let $f$ be a continuous permutation-equivariant sequence-to-sequence function on compact sets, defined as $f:\mathbb{R}^{n\times d}\to\mathbb{R}^{n\times d}$. Then, for any $\epsilon>0$, there exists a Transformer $T$ such that:

$$\sup_{X\in\mathbb{R}^{n\times d}}\|f(X)-T(X)\|_\infty<\epsilon.$$

### 5.3. Proof

We aim to prove that a Transformer can approximate the Sumformer $S$ and hence approximate any equivariant function $f$. The proof consists of two main steps:

Step 1: Sumformer Approximation of $f$

1. Goal: Construct a Sumformer $S$ that approximates $f$.

2. Constructing $\Sigma$ using $\phi$: For each input sequence $X=[x_1,\ldots,x_n]$, define

$$\Sigma=\sum_{i=1}^{n}\phi(x_i),$$

where $\phi(x_i)$ encodes information from each input $x_i$ in a way that captures permutation-equivariant properties. For example, using multisymmetric polynomials to approximate $f$'s behavior.

3. Defining $S(X)$ using $\psi$: Using $\Sigma$, define

$$S(X)=[\psi(x_1,\Sigma),\ldots,\psi(x_n,\Sigma)],$$

where $\psi(x_i,\Sigma)$ combines each $x_i$ with the aggregate information $\Sigma$ to approximate $f(x_i)$.

4. Approximation Error of $S$ for $f$: Given a continuous $f$, we can choose $\phi$ and $\psi$ such that:

$$\sup_{X\in\mathbb{R}^{n\times d}}\|f(X)-S(X)\|_\infty<\frac{\epsilon}{2}.$$

Step 2: Transformer Approximation of Sumformer

DOI: 10.54254/2753-8818/2025.DL27324

1. Input Encoding: For each input $x_i$, construct the sequence

$$X' = [[x_1, \phi(x_1)], \ldots, [x_n, \phi(x_n)]] \in \mathbb{R}^{n\times(d+d_1)}.$$

2. Using Attention to Approximate $\Sigma$: Set the Transformer attention query and key matrices as:

$$W_Q = W_K = [1\&0\& \ldots \&0]^\top \in \mathbb{R}^{(d+d_1)\times 1},$$

so that each attention score is constant, allowing us to approximate the sum $\Sigma$ by aggregating $\phi(x_i)$ terms across the sequence.

3. Output Generation Using Feed-Forward Layers: Use two feed-forward layers to approximate $\psi(x_i, \Sigma)$ for each $x_i$, ensuring that:

$$\sup_{X\in\mathbb{R}^{n\times d}} \|S(X) - T(X)\|_\infty < \frac{\epsilon}{2}.$$

Step 3: Combined Approximation By combining these two steps, we have:

$$\sup_{X\in\mathbb{R}^{n\times d}} \|f(X) - T(X)\|_\infty \le \sup_{X\in\mathbb{R}^{n\times d}} \|f(X) - S(X)\|_\infty + \sup_{X\in\mathbb{R}^{n\times d}} \|S(X) - T(X)\|_\infty < \epsilon.$$

This completes the proof.

### 5.4. Conclusion

In summary, the multi-particle dynamical systems perspective provides a powerful lens to understand transformers. By modeling self-attention as interacting particles, one can rigorously study convergence, clustering, and stability, while also revealing parallels with classical systems such as Kuramoto oscillators. This framework not only deepens theoretical insight but also opens pathways for principled analysis and potential improvements in transformer architectures.

## References

[1] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, June). BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers) (pp. 4171–4186). Minneapolis, Minnesota: Association for Computational Linguistics. Retrieved from https: //aclanthology.org/N19-1423

[2] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., . . . Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. In (Vol. abs/2010.11929). Retrieved from https: //api.semanticscholar.org/CorpusID: 225039882

[3] E, W. (2017). A proposal on machine learning via dynamical systems. Communications in Mathe- matics and Statistics, 5 , 1 - 11. Retrieved from https: //api.semanticscholar.org/CorpusID: 64849498

[4] Geshkovski, B., Koubbi, H., Polyanskiy, Y., & Rigollet, P. (2024). Dynamic metastability in the self- attention model. Retrieved from https: //arxiv.org/abs/2410.06833

[5] Geshkovski, B., Letrouit, C., Polyanskiy, Y., & Rigollet, P. (2024). A mathematical perspective on transformers. Retrieved from https: //arxiv.org/abs/2312.10794 6

[6] Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., & Shah, M. (2022, September). Transformers in vision: A survey. ACM Comput. Surv., 54 (10s). Retrieved from https: //doi.org/10.1145/ 3505244 DOI: 10.1145/3505244

DOI: 10.54254/2753-8818/2025.DL27324

[7] Lu, Y., Li, Z., He, D., Sun, Z., Dong, B., Qin, T., . . . Liu, T.-Y. (2019). Understanding and improving transformer from a multi-particle dynamic system point of view. ArXiv, abs/1906.02762 . Retrieved from https: //api.semanticscholar.org/CorpusID: 174801126

[8] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., . . . Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21 (140), 1–67. Retrieved from http: //jmlr.org/papers/v21/20-074.html