Spiking Graph Neural Network on Riemannian Manifolds

Li Sun, Zhenhao Huang, Qiqi Wan,

Hao Peng and Philip S. Yu

Contact: ccesunli@ncepu.edu.cn
North China Electric Power University, Beijing, China

Source code: *https://github.com/ZhenhHuang/MSG*

Motivations and Contributions

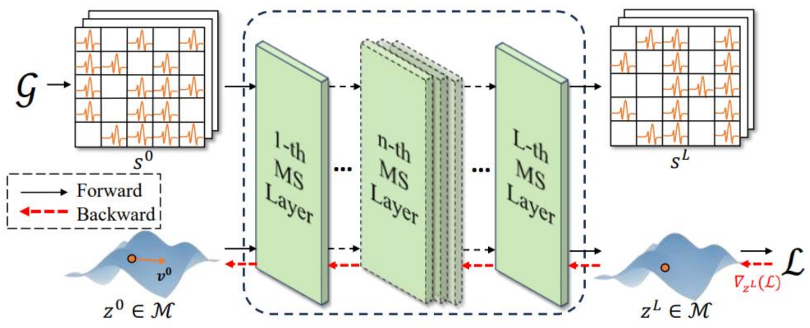
Motivations

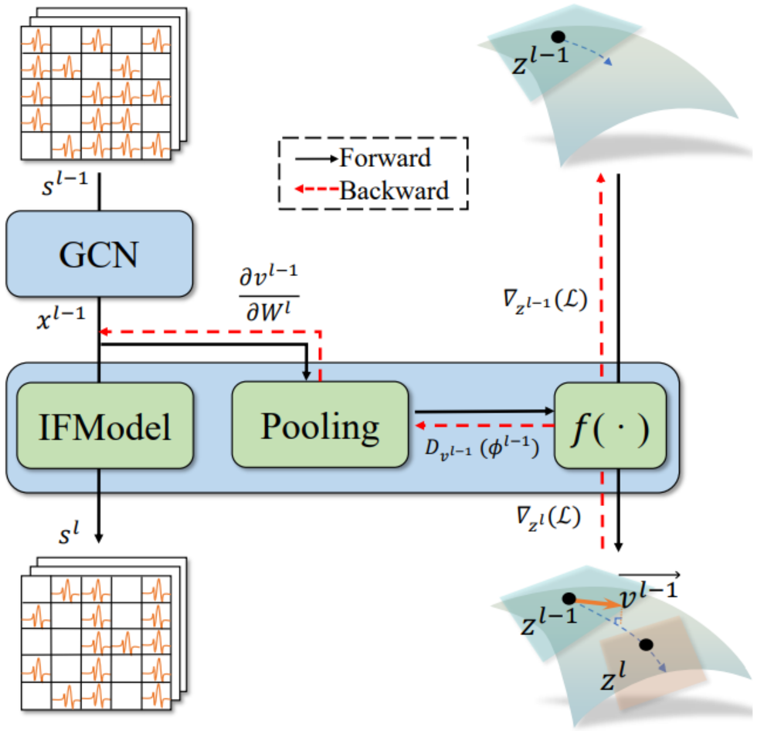
- Graph data is non-Euclidean, and it’s hard to represent its geometric properties by binary spike trains.
- BPTT training for SNN is non-differentiable and high time latency delay.
- Surrogate gradient with BPTT may cause gradient vanishing and explosion.
- Float operations on the Riemannian manifold contradict the binary property of SNN.

Contributions

- First Graph-SNN on the Riemannian manifolds (*MSG*) without manifold operations on forward phase.
- New training methods *DvM* builds relation to the manifold.
- Avoids high latency and gradient vanishing and explosion.
- Lower energy consumption and effective model performance.
- Theoretically show *MSG*’s connection to manifold ODEs.

Manifold-valued Spiking GNN





BPTT vs DvM

BPTT:
$$\nabla_{W^l}\mathcal{L} = \sum_t \left[\frac{\partial s^l[t]}{\partial W^l}\right]^* \nabla_{s^l[t]}\mathcal{L}$$

Differentiation via Manifold:

Theorem 4.1 (Backward Gradient). *Let $\mathcal{L}$ be the scalar-valued function, and $\mathbf{z}^l$ is the output of l-th layer with parameter $\mathbf{W}^l$, which is delivered by tangent vector $\mathbf{v}^l$. Then, the gradient of function $\mathcal{L}$ w.r.t $\mathbf{W}^l$ is given as follows:*

$$\nabla_{\mathbf{W}^l}\mathcal{L} = [\frac{\partial \mathbf{v}^{l-1}}{\partial \mathbf{W}^l}]^*[D_{\mathbf{v}^{l-1}}\phi^{l-1}]^*\nabla_{\mathbf{z}^l}\mathcal{L}, \quad \nabla_{\mathbf{z}^l}\mathcal{L} = [D_{\mathbf{z}^l}\psi^l]^*\nabla_{\mathbf{z}^{l+1}}\mathcal{L}, \qquad (12)$$

where $\phi^{l-1}(\cdot) = \mathrm{Exp}_{\mathbf{z}^{l-1}}(\cdot)$, $\psi^l(\cdot) = \mathrm{Exp}_{(\cdot)}(\mathbf{v}^l)$, and $[\cdot]^*$ means the matrix form of pullback.

BPTT via spikes：

1. High time steps latency delay
2. Non-differentiable
3. Gradient vanishing/explosion

Differentiation via Manifold:

1. Recurrence-free
2. Differentiable by smooth function
3. Avoid gradient vanishing/explosion

Algorithm 1 Training MSG by the proposed Differentiation via Manifold

Input: Graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{F}, \mathbf{A})$, Manifold $\mathcal{M}$, Loss function over the manifold $\mathcal{L}(\cdot)$, Number of spiking layers L, Original points $\mathbf{O}$.
Output: Parameters $\{\mathbf{W}^l\}_{l=0,\cdots,L}$

```
while not converging do
    ▷ forward pass
    Input current X^0 = GCN(A, F; W^0);
    Initialize [S^0, Z^0] = MSNeuron(X^0, O);
    for each spiking layer l = 1 to L do
        X^(l-1) = GCN(A, S^(l-1); W^l);
        [S^l, Z^l] = MSNeuron(X^(l-1), Z^(l-1));
    end for
    ▷ backward pass
    Compute ∇_{z^L} L from L(Z^L).
    for layer l = L - 1 to 1 do
        Compute D_{z^l} ψ^l, D_{v^{l-1}} φ^{l-1}, ∂v^{l-1}/∂W^l.
        Compute ∇_{z^l} L, ∇_{W^l} L as Eq. 12.
        Update W^l.
    end for
end while
```

Theory: MSG as Neural ODE Solver

Definition 5.1 (Dynamic Chart Solver [17]). *The manifold ODE in Eq. (13) with initial condition* $\mathbf{z}(0) = \mathbf{z}$ *can be solved with a finite collection of successive charts* $\{(U_i, \phi_i)\}_{i=1,\dots,L}$*. If* ode_i *is the numerical solver to Euclidean ODE corresponding to the* i*-th chart,* $\mathbf{y}(t) = \mathrm{ode}_i(t)$ *on* $[\tau_i, \tau_i + \epsilon_i]$*, then* $\mathbf{z}(t)$ *in Eq. (13) is given as*

$$(\phi_L^{-1} \circ \mathrm{ode}_L \circ (\phi_L \circ \phi_{L-1}^{-1}) \circ \dots \circ (\phi_2 \circ \phi_1^{-1}) \circ \mathrm{ode}_1 \circ \phi_1)(t). \tag{15}$$

That is, a manifold ODE can be solved in Euclidean subspaces given by a series of successive charts.

Theorem 5.2 (MSG as Dynamic Chart Solver). *If* $\mathbf{y}(t) : [\tau, \tau + \epsilon] \to \mathbb{R}^n$ *is the solution of*

$$\frac{d\mathbf{y}(t)}{dt} = (D_{\mathrm{Exp}_{\mathbf{z}}(\mathbf{y}(t))} \mathrm{Log}_{\mathbf{z}}) u(\mathrm{Exp}_{\mathbf{z}}(\mathbf{y}(t)), t), \tag{16}$$

then $\mathbf{z}(t) = \mathrm{Exp}_{\mathbf{z}}(\mathbf{y}(t))$ *is a valid solution to the manifold ODE of Eq. (13) on* $t \in [\tau, \tau + \epsilon]$*, where* $\mathbf{z} = \mathbf{z}(\tau)$*. If* $\mathbf{y}(t)$ *is given by the first-order approximation with the* ϵ *small enough,*

$$\mathbf{y}(\tau + \epsilon) = \epsilon \cdot (D_{\mathbf{z}} \mathrm{Log}_{\mathbf{z}}) u(\mathbf{z}(\tau), \tau), \tag{17}$$

then the update process of Eqs. (4) and (8) in MSG is equivalent to Dynamic Chart Solver in Eq. (15).

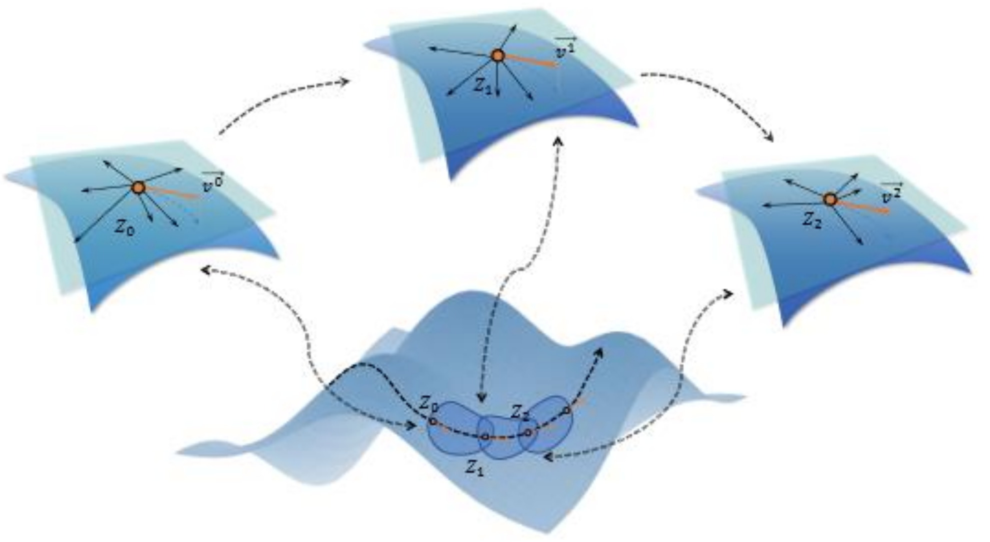


Figure 3: Charts given by the logarithmic map.

1. MSG approximates a solver of manifold Ordinary Differential Equations (ODEs).
2. Each layer solves the ODE of a smooth path, the endpoint is parameterized by a GNN related to the spikes.
3. Layer-by-layer forwarding solves the manifold ODE from the current chart to the successive chart.

Experiment Results

华北电力大学
NORTH CHINA ELECTRIC POWER UNIVERSITY

Table 1: Node Classification (NC) in terms of classification accuracy (%) and Link Prediction in terms of AUC (%) on Computers, Photo, CS and Physics datasets. The best results are **boldfaced**, and the runner-ups are underlined. The standard derivations are given in the subscripts.

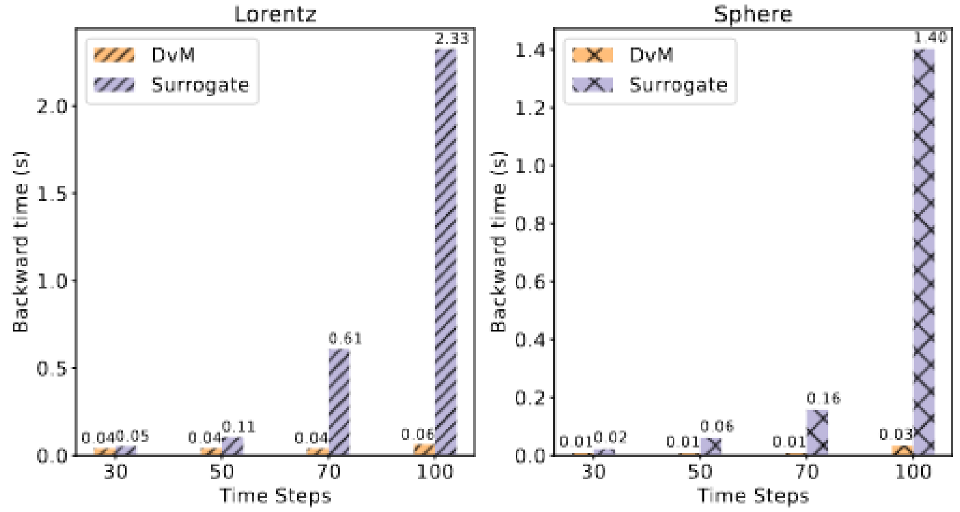
		Computers		Photo		CS		Physics	
		NC	LP	NC	LP	NC	LP	NC	LP
ANN-E	GCN [18]	83.55$_{\pm 0.71}$	92.07$_{\pm 0.40}$	86.01$_{\pm 0.20}$	88.84$_{\pm 0.39}$	91.68$_{\pm 0.84}$	93.68$_{\pm 0.84}$	95.03$_{\pm 0.19}$	93.46$_{\pm 0.39}$
	GAT [2]	86.82$_{\pm 0.04}$	91.91$_{\pm 1.08}$	86.68$_{\pm 1.32}$	88.45$_{\pm 0.07}$	91.74$_{\pm 0.22}$	94.06$_{\pm 0.70}$	95.11$_{\pm 0.29}$	93.44$_{\pm 0.70}$
	SGC [19]	82.17$_{\pm 1.25}$	90.46$_{\pm 0.80}$	87.91$_{\pm 0.65}$	89.84$_{\pm 0.40}$	92.09$_{\pm 0.05}$	**95.94**$_{\pm 0.43}$	94.77$_{\pm 0.32}$	95.93$_{\pm 0.70}$
	SAGE [1]	81.69$_{\pm 0.86}$	90.56$_{\pm 0.48}$	89.41$_{\pm 1.28}$	89.86$_{\pm 0.90}$	**92.71**$_{\pm 0.73}$	95.22$_{\pm 0.14}$	95.62$_{\pm 0.26}$	95.75$_{\pm 0.37}$
ANN-R	HGCN [3]	88.71$_{\pm 0.24}$	96.88$_{\pm 0.53}$	89.18$_{\pm 0.50}$	94.54$_{\pm 0.20}$	90.72$_{\pm 0.16}$	93.02$_{\pm 0.26}$	94.46$_{\pm 0.20}$	94.10$_{\pm 0.64}$
	κ-GCN [13]	89.20$_{\pm 0.50}$	95.30$_{\pm 0.20}$	92.22$_{\pm 0.62}$	94.89$_{\pm 0.15}$	91.98$_{\pm 0.15}$	94.86$_{\pm 0.18}$	95.85$_{\pm 0.20}$	94.58$_{\pm 0.22}$
	$\mathcal{Q}$-GCN [4]	85.94$_{\pm 0.93}$	**96.98**$_{\pm 0.05}$	92.50$_{\pm 0.95}$	97.47$_{\pm 0.03}$	91.18$_{\pm 0.28}$	93.39$_{\pm 0.20}$	94.84$_{\pm 0.25}$	OOM
	HyboNet [54]	86.29$_{\pm 2.30}$	96.80$_{\pm 0.05}$	92.67$_{\pm 0.09}$	**97.70**$_{\pm 0.07}$	92.34$_{\pm 0.03}$	95.65$_{\pm 0.26}$	95.56$_{\pm 0.18}$	**98.46**$_{\pm 0.49}$
SNN-E	SpikeNet [45]	88.00$_{\pm 0.70}$	-	92.90$_{\pm 0.10}$	-	92.15$_{\pm 0.18}$	-	92.66$_{\pm 0.30}$	-
	SpikeGCN [5]	86.90$_{\pm 0.30}$	91.12$_{\pm 1.79}$	92.60$_{\pm 0.70}$	93.84$_{\pm 0.03}$	90.86$_{\pm 0.11}$	95.07$_{\pm 1.22}$	94.53$_{\pm 0.18}$	92.88$_{\pm 0.80}$
	SpikeGCL [6]	89.04$_{\pm 0.08}$	92.72$_{\pm 0.03}$	92.50$_{\pm 0.17}$	95.58$_{\pm 0.11}$	91.77$_{\pm 0.11}$	95.13$_{\pm 0.24}$	95.21$_{\pm 0.10}$	94.15$_{\pm 0.29}$
	SpikeGT [55]	81.00$_{\pm 1.06}$	-	90.66$_{\pm 0.38}$	-	91.86$_{\pm 0.61}$	-	94.38$_{\pm 1.57}$	-
	MSG (Ours)	**89.27**$_{\pm 0.19}$	94.65$_{\pm 0.73}$	**93.11**$_{\pm 0.11}$	96.75$_{\pm 0.18}$	92.65$_{\pm 0.04}$	95.19$_{\pm 0.15}$	**95.93**$_{\pm 0.07}$	93.43$_{\pm 0.16}$

Table 2: Ablation study of geometric variants. Results of node classification in terms of ACC (%).

	Computers	Photo	CS	Physics
$\mathbb{H}^{32}$	**89.27**$_{\pm 0.19}$	**93.11**$_{\pm 0.11}$	92.65$_{\pm 0.04}$	**95.93**$_{\pm 0.07}$
$\mathbb{S}^{32}$	87.84$_{\pm 0.77}$	92.03$_{\pm 0.79}$	92.72$_{\pm 0.06}$	95.85$_{\pm 0.02}$
$\mathbb{E}^{32}$	88.94$_{\pm 0.24}$	92.93$_{\pm 0.21}$	**92.82**$_{\pm 0.04}$	95.81$_{\pm 0.04}$
$\mathbb{H}^{16} \times \mathbb{H}^{16}$	89.18$_{\pm 0.25}$	92.06$_{\pm 0.14}$	92.67$_{\pm 0.10}$	95.90$_{\pm 0.04}$
$\mathbb{H}^{16} \times \mathbb{S}^{16}$	88.00$_{\pm 1.05}$	91.97$_{\pm 0.08}$	92.33$_{\pm 0.21}$	95.73$_{\pm 0.11}$
$\mathbb{S}^{16} \times \mathbb{S}^{16}$	82.49$_{\pm 1.18}$	92.31$_{\pm 0.45}$	92.18$_{\pm 0.21}$	95.81$_{\pm 0.10}$

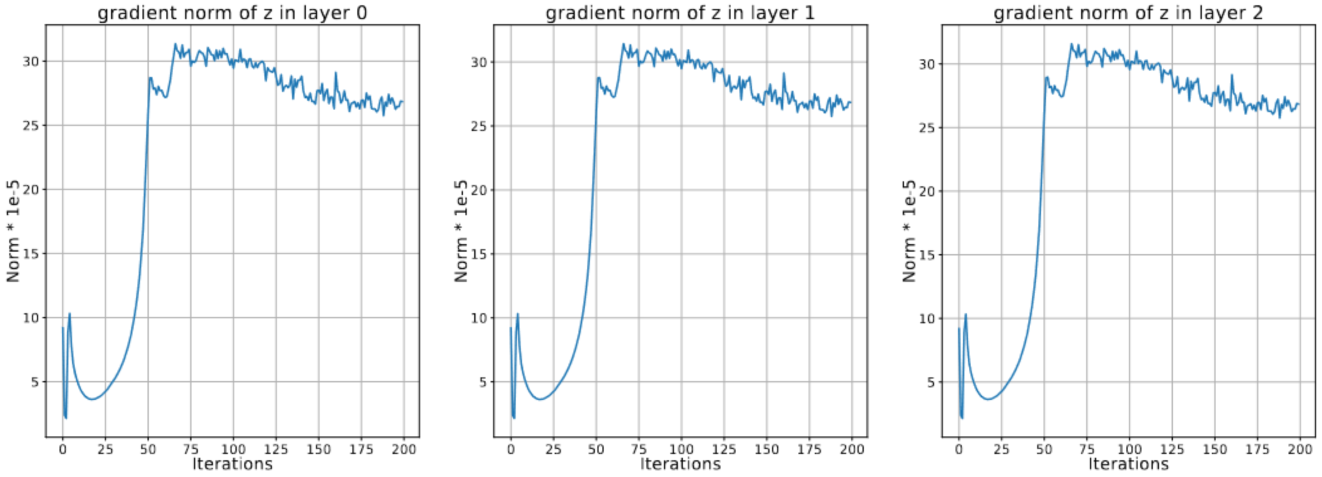
Table 3: Energy cost. The number of parameters at the running time (KB) and theoretical energy consumption (mJ) on Computers, Photo, CS and Physics datasets. The best results are **boldfaced**, and the runner ups are underlined.

		Computers		Photo		CS		Physics	
		#(para.)	energy	#(para.)	energy	#(para.)	energy	#(para.)	energy
ANN-E	GCN [18]	24.91	1.671	24.14	0.893	218.29	18.444	269.48	42.842
	GAT [2]	24.99	2.477	24.22	1.273	218.38	28.782	269.55	81.466
	SGC [19]	**7.68**	0.508	**5.97**	0.219	**102.09**	8.621	**42.08**	6.688
	SAGE [1]	49.77	1.671	48.23	0.893	436.53	18.444	538.92	42.842
ANN-R	HGCN [3]	24.94	1.614	24.96	0.869	217.79	18.390	269.31	42.800
	κ-GCN [13]	25.89	1.647	25.12	0.889	218.24	18.440	269.44	42.836
	$\mathcal{Q}$–GCN [4]	24.93	1.629	24.96	0.876	217.83	18.393	269.34	42.809
	HyboNet [54]	27.06	1.625	26.29	0.875	219.94	18.399	271.47	42.825
SNN-E	SpikeNet [43]	101.22	0.070	98.07	**0.040**	438.51	0.218	540.04	0.334
	SpikingGCN [5]	38.40	0.105	29.84	0.046	510.45	1.871	210.40	1.451
	SpikeGCL [6]	59.26	0.121	57.85	0.067	445.69	0.128	548.74	0.214
	SpikeGT [55]	77.07	1.090	74.46	0.584	365.28	6.985	355.77	12.524
	MSG(Ours)	26.95	**0.047**	25.68	0.043	226.15	**0.026**	143.72	**0.029**

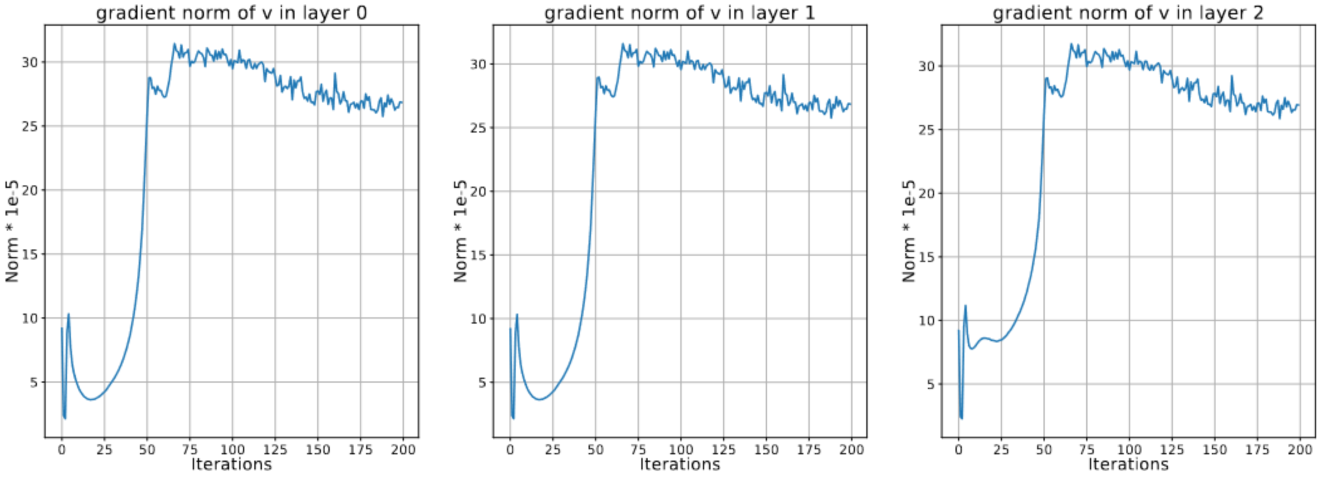


(a) Backward times in model training.

Visualizations



(a) The norm of backward gradient of L with respect to $\mathbf{z}$ in each spiking layers.



(b) The norm of backward gradient of L with respect to $\mathbf{v}$ in each spiking layers.