Geometric Deep Learning on Complexes

3rd Nepal Winter School in AI
Kathmandu, Nepal

Cristian Bodnar
Department of Computer Science
University of Cambridge

December 27, 2021

Overview

This presentation is largely based on [C. Bodnar, F. Frasca, et al., 2021a] and [C. Bodnar, F. Frasca, et al., 2021b].

1. Simplicial Complexes

1.1 Orientations

1.2 Chains

1.3 Hodge Theory

1.4 Convolutions and Symmetries

1.5 Application: Trajectory Classification

2. Cell Complexes

2.1 Definition and Construction

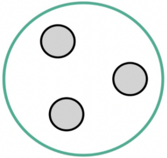
2.2 Cellular Weisfeiler Lehman

2.3 Message Passing on Cell Complexes

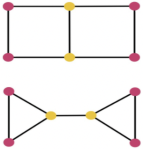
2.4 Application: Molecular Property Prediction

Beyond graphs

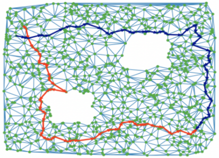
Groupwise interactions



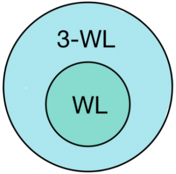
Higher-order structures



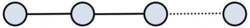
Higher-order signals



Expressive power



Long-range interactions

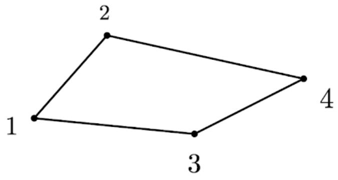


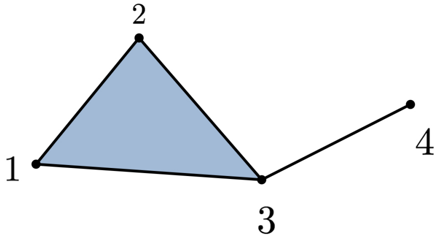
Simplicial Complexes

Definition (Simplicial Complex)

Let V be a non-empty vertex set. A *simplicial complex* K is a collection of nonempty subsets of V, called *simplices*, such that:

1. K contains all the singleton subsets of V.
2. K is closed under taking subsets.





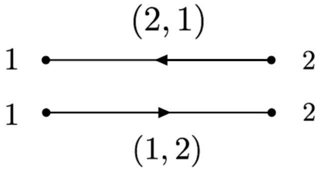
$\{\{1\},\{2\},\{3\},\{4\},\{1,2\},\{1,3\},\{2,4\},\{3,4\}\}$ $\{\{1\},\{2\},\{3\},\{4\},\{1,2\},\{1,3\},\{2,3\},\{3,4\},\{1,2,3\}\}$

Orientations

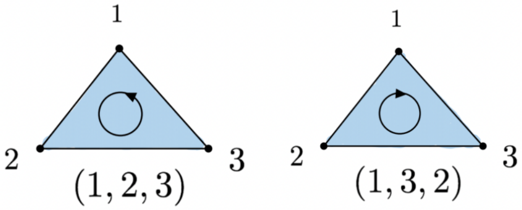
Definition

An *oriented* simplex is a simplex with a specified order of its vertices.

These can be visualised as a walk on the simplex in the order specified by the vertices.



The orientations of the 1-simplex $\{1, 2\}$

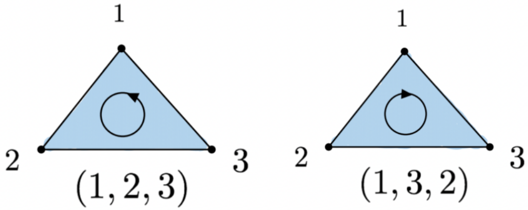


Two orientations of the 2-simplex $\{1, 2, 3\}$

We represent oriented simplices as tuples $(\cdot)$ and unoriented ones as sets $\{\cdot\}$.

Orientations

Ignoring the starting point of the walk, each k-simplex with $k > 0$ has two distinct orientations.



We can choose a representative for each of these two equivalence classes:

$$123 := \{(1,2,3),(2,3,1),(3,1,2)\} \leftarrow \text{Even permutations}$$
$$132 := \{(1,3,2),(2,1,3),(3,2,1)\} \leftarrow \text{Odd permutations}$$

Orientations

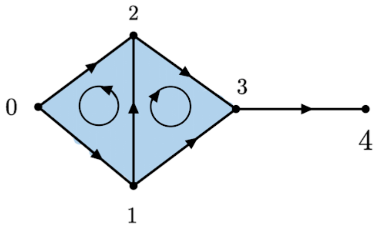
Definition (Oriented Simplicial Complex)

An oriented simplicial complex is a simplicial complex with a choice of orientation for each of its simplices.

Example

- $(1,2,3)$ and $(2,3)$ have the same orientation.
- $(1,2,3)$ and $(1,2,0)$ have the same orientation with respect to the lower-dim simplex $(1,2)$.
- $(0,1)$ and $(0,2)$ have different orientations with respect to the higher-dim simplex $(0,1,2)$.

When simplices share vertices, we can talk about their *relative orientation*.



Chains

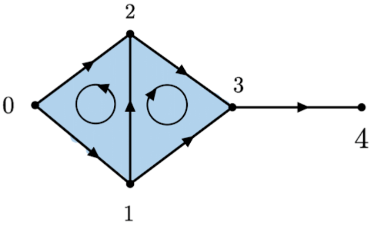
Definition (k-chains)

The vector space of *k-chains* $C_k(K, \mathbb{R})$ is the vector space with real coefficients with basis given by the oriented k-simplices of K.

Example

Chains $c_1 \in C_1(K, \mathbb{R})$ and $c_2 \in C_2(K, \mathbb{R})$

$$\begin{aligned} c_1 &= 2.5(0,1) - (3,4) + 3.14(1,3) \\ &= 2.5(0,1) + (4,3) + 3.14(1,3) \\ c_2 &= 5.0(0,1,2) + 1.3(1,2,3) \end{aligned}$$

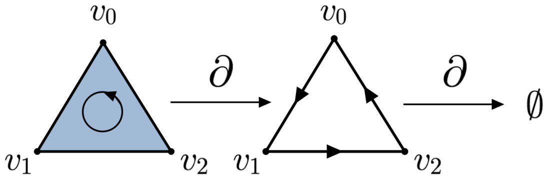


Boundary operator

Definition (Boundary operator)

The boundary operator $\partial_k : C_k(K, \mathbb{R}) \to C_{k-1}(K, \mathbb{R})$ is defined as

$$\partial_k(v_0, \ldots, v_k) = \sum_{i=0}^{k} (-1)^i (v_0, \ldots, \hat{v}_i, \ldots, v_k)$$



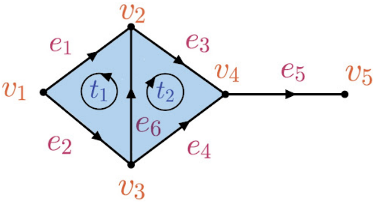
$$\partial(v_0, v_1, v_2) = (v_1, v_2) - (v_0, v_2) + (v_0, v_1) = (v_1, v_2) + (v_2, v_0) + (v_0, v_1) \tag{1}$$

$$\partial\partial(v_0, v_1, v_2) = \partial(v_1, v_2) + \partial(v_2, v_0) + \partial(v_0, v_1) \tag{2}$$

$$= (v_2 - v_1) + (v_0 - v_2) + (v_1 - v_0) = 0 \tag{3}$$

Boundary matrices

We can represent the boundary operator for each dimension using a matrix.



$$B_1 = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ +1 & 0 & -1 & 0 & 0 & +1 \\ 0 & +1 & 0 & -1 & 0 & -1 \\ 0 & 0 & +1 & +1 & -1 & 0 \\ 0 & 0 & 0 & 0 & +1 & 0 \end{bmatrix} \end{array}$$

$$B_2 = \begin{array}{c} \\ e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{array} \begin{array}{c} \begin{array}{cc} t_1 & t_2 \end{array} \\ \begin{bmatrix} -1 & 0 \\ +1 & 0 \\ 0 & +1 \\ 0 & -1 \\ 0 & 0 \\ +1 & +1 \end{bmatrix} \end{array}$$

Hodge Laplacian

Definition

The k-th Hodge Laplacian $\boldsymbol{L}_k : C_k(K,\mathbb{R}) \to C_k(K,\mathbb{R})$ is given by:

$$\boldsymbol{L}_k = \boldsymbol{L}_k^{\downarrow} + \boldsymbol{L}_k^{\uparrow} = \boldsymbol{B}_k^{\top}\boldsymbol{B}_k + \boldsymbol{B}_{k+1}\boldsymbol{B}_{k+1}^{\top}.$$

Importantly, $\boldsymbol{L}_0 = \boldsymbol{B}_1\boldsymbol{B}_1^{\top} = \boldsymbol{D} - \boldsymbol{A}$ is the usual graph Laplacian.

Denote by $\sigma_i \vee \sigma_j$ if σ_i, σ_j share a $(k-1)$-simplex and $\sigma_i \wedge \sigma_j$ if they are on the boundary of the same $(k+1)$-simplex.

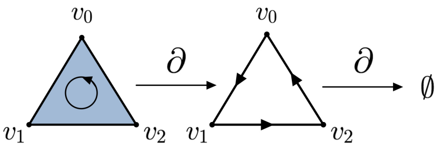
$$\boldsymbol{L}_k^{\downarrow}(i,j) = \begin{cases} k+1 & \text{if } i = j \\ \pm 1 & \text{if } i \neq j \text{ and } \sigma_i \vee \sigma_j \\ 0 & \text{otherwise} \end{cases} \qquad \boldsymbol{L}_k^{\uparrow}(i,j) = \begin{cases} \deg^{\uparrow}(\sigma_i) & \text{if } i = j \\ \pm 1 & \text{if } i \neq j \text{ and } \sigma_i \wedge \sigma_j \\ 0 & \text{otherwise} \end{cases}$$

For more details:
[A. Muhammad et al., 2006, L. Lim, 2019, S. Barbarossa, et al., 2020, M. Schaub et al., 2020]

Hodge Theory

Lemma

The boundary of a chain has no boundary: $\partial_k \circ \partial_{k+1} = 0$

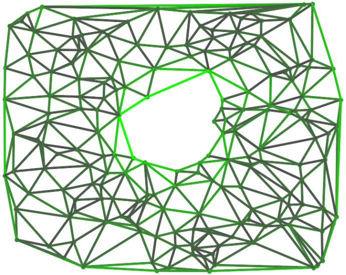


Theorem (Hodge Decomposition)

$$C_k(K, \mathbb{R}) = im(\partial_k^\top) \bigoplus ker(\boldsymbol{L}_k) \bigoplus im(\partial_{k+1})$$

Theorem

$$ker(\boldsymbol{L}_k) \cong H_k := ker(\partial_k)/im(\partial_{k+1})$$



Harmonic eigenvector of $\boldsymbol{L}_1$
Credits to Andrei C. Popescu

Convolutional operators

Equipped with a Laplacian operator, one can define a convolutional operator for 1-chains [S. Ebli et al., 2020, E. Bunch et al., 2020, T. Mitchell et al., 2021, C. Bodnar, F. Frasca, et al., 2021a].

Let $\boldsymbol{X}_0 \in \mathbb{R}^{V \times F}$, $\boldsymbol{X}_1 \in \mathbb{R}^{E \times F}$, $\boldsymbol{X}_2 \in \mathbb{R}^{T \times F}$ be matrices of 0-chains, 1-chains and 2-chains respectively, $\boldsymbol{W}_i$ a set of weights and ψ an activation function.

$$\boldsymbol{Y} = \psi\Big(\underbrace{\boldsymbol{L}_1^{\downarrow}\boldsymbol{X}_1\boldsymbol{W}_1}_{\text{Lower adjacencies}} + \underbrace{\boldsymbol{L}_1^{\uparrow}\boldsymbol{X}_1\boldsymbol{W}_2}_{\text{Upper adjacencies}} + \underbrace{\boldsymbol{B}_1^{\top}\boldsymbol{X}_0\boldsymbol{W}_3}_{\text{Boundary adjacencies}} + \underbrace{\boldsymbol{B}_2\boldsymbol{X}_2\boldsymbol{W}_4}_{\text{Coboundary adjacencies}} \Big) \tag{4}$$

This can be seen as a form of message passing with four types of adjacencies.

Symmetries: Permutation Equivariance

A simplicial complex of dimention d can be specified by all its boundary matrices $\mathcal{B} = (\boldsymbol{B}_1, \ldots, \boldsymbol{B}_d)$. Similarly define a tuple of permutation matrices $\mathcal{P} = (\boldsymbol{P}_0, \ldots, \boldsymbol{P}_d)$ and denote by $\mathcal{P}\mathcal{B} = (\boldsymbol{P}_0 \boldsymbol{B}_1 \boldsymbol{P}_1^\top, \ldots, \boldsymbol{P}_{d-1} \boldsymbol{B}_d \boldsymbol{P}_d^\top)$.

Definition

$f : C_k(K, \mathbb{R})^{F_1} \to C_k(K, \mathbb{R})^{F_2}$ is permutation equivariant if $f(\mathcal{P}\mathcal{B}, \boldsymbol{P}_k \boldsymbol{X}) = \boldsymbol{P}_k f(\mathcal{B}, \boldsymbol{X})$

Proposition

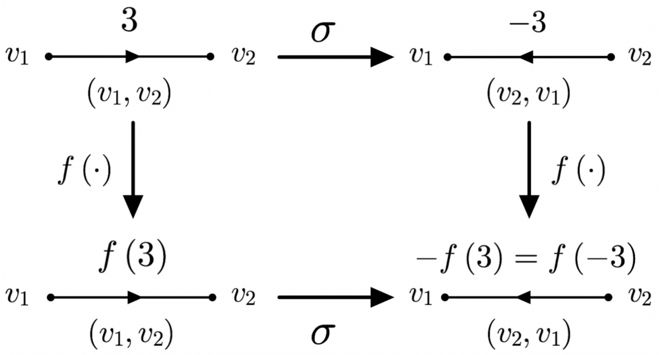
The function $f(\mathcal{B}, \mathcal{X}) := \psi\left(\boldsymbol{L}_1^{\downarrow} \boldsymbol{X}_1 \boldsymbol{W}_1 + \boldsymbol{L}_1^{\uparrow} \boldsymbol{X}_1 \boldsymbol{W}_2\right)$ is permutation equivariant.

Proof idea.

$$(\boldsymbol{P}_0 \boldsymbol{B}_1 \boldsymbol{P}_1^\top)^\top (\boldsymbol{P}_0 \boldsymbol{B}_1 \boldsymbol{P}_1^\top)(\boldsymbol{P}_1 \boldsymbol{X}_1) \boldsymbol{W}_1 = \boldsymbol{P}_1 \boldsymbol{B}_1^\top \boldsymbol{B}_1 \boldsymbol{X}_1 \boldsymbol{W} = \boldsymbol{P}_1 (\boldsymbol{L}_1^{\downarrow} \boldsymbol{X}_1 \boldsymbol{W}) \qquad \square$$

Symmetries: Orientation Equivariance

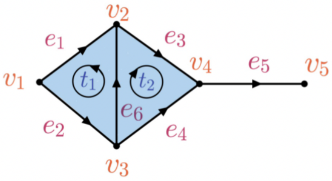
Mathematically, the choice of orientation is irrelevant. Therefore, we would like our model to produce the same outputs up to a change in orientation.



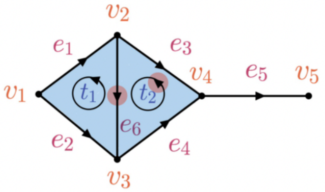
The function f must be odd.

Symmetries: Orientation Equivariance

If a simplex changes its orientation, then it flips its relative orientation with respect to its adjacent neighbours.



$$B_1 = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ +1 & 0 & -1 & 0 & 0 & +1 \\ 0 & +1 & 0 & -1 & 0 & -1 \\ 0 & 0 & +1 & +1 & -1 & 0 \\ 0 & 0 & 0 & 0 & +1 & 0 \end{bmatrix} \end{array} \qquad B_2 = \begin{array}{c} \\ e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{array} \begin{array}{c} \begin{array}{cc} t_1 & t_2 \end{array} \\ \begin{bmatrix} -1 & 0 \\ +1 & 0 \\ 0 & +1 \\ 0 & -1 \\ 0 & 0 \\ +1 & +1 \end{bmatrix} \end{array}$$



$$B_1 = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{array} \\ \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ +1 & 0 & -1 & 0 & 0 & -1 \\ 0 & +1 & 0 & -1 & 0 & +1 \\ 0 & 0 & +1 & +1 & -1 & 0 \\ 0 & 0 & 0 & 0 & +1 & 0 \end{bmatrix} \end{array} \qquad B_2 = \begin{array}{c} \\ e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{array} \begin{array}{c} \begin{array}{cc} t_1 & t_2 \end{array} \\ \begin{bmatrix} -1 & 0 \\ +1 & 0 \\ 0 & -1 \\ 0 & +1 \\ 0 & 0 \\ -1 & +1 \end{bmatrix} \end{array}$$

This amounts to flipping the sign in the corresponding rows and columns of the boundary matrices.

Symmetries: Orientation Equivariance

Consider a tuple of matrices $\mathcal{T} = (\boldsymbol{T}_0, \ldots, \boldsymbol{T}_d)$, where each $\boldsymbol{T}_i$ is a diagonal matrix with values in $\{\pm 1\}$. Additionally, because vertices always have a positive orientation, we restrict $\boldsymbol{T}_0 = I$. Then denote by $\mathcal{T}\mathcal{B} = (\boldsymbol{T}_0 \boldsymbol{B}_1 \boldsymbol{T}_1, \ldots, \boldsymbol{T}_{d-1} \boldsymbol{B}_d \boldsymbol{T}_d)$

Definition

$f : C_k(K, \mathbb{R})^{F_1} \rightarrow C_k(K, \mathbb{R})^{F_2}$ is orientation equivariant if $f(\mathcal{T}\mathcal{B}, \boldsymbol{T}_k \boldsymbol{X}) = \boldsymbol{T}_k f(\mathcal{B}, \boldsymbol{X})$

Proposition

$f(\mathcal{B}, \mathcal{X}) := \psi\left(\boldsymbol{L}_1^{\downarrow} \boldsymbol{X}_1 \boldsymbol{W}_1 + \boldsymbol{L}_1^{\uparrow} \boldsymbol{X}_1 \boldsymbol{W}_2\right)$ *is orientation equivariant when* ψ *is odd.*

Proof idea.

$\psi\left((\boldsymbol{T}_0 \boldsymbol{B}_1 \boldsymbol{T}_1)^{\top} (\boldsymbol{T}_0 \boldsymbol{B}_1 \boldsymbol{T}_1) \boldsymbol{T}_1 \boldsymbol{X}_1 \boldsymbol{W}\right) = \psi\left(\boldsymbol{T}_1 \boldsymbol{L}_1^{\downarrow} \boldsymbol{X}_1 \boldsymbol{W}\right)$

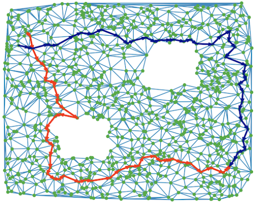
Note that ψ and $\boldsymbol{T}_1$ commute when ψ is odd. $\square$

Application: Trajectory Classification

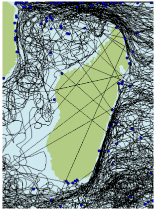
We are interested in classifying trajectories represent as 1-chains. The tasks are inspired from [M. Schaub et al., 2020].

Method	Synthetic Flow		Ocean Drifters	
	Train	Test	Train	Test
GNN L_0-inv	63.9±2.4	61.0±4.2	70.1±2.3	63.5±6.0
MPSN L_0-inv	88.2±5.1	85.3±5.8	84.6±4.0	**71.5±4.1**
MPSN - ReLU	**100.0±0.0**	50.0±0.0	**100.0±0.0**	46.5±5.7
MPSN - Id	88.0±3.1	82.6±3.0	94.6±0.9	**73.0±2.7**
MPSN - Tanh	97.9±0.7	**95.2±1.8**	**99.7±0.5**	**72.5±0.0**

Trajectory classification accuracy.



The task is to classify random walks.



The task is to classify ocean drifter trajectories around Madagascar.

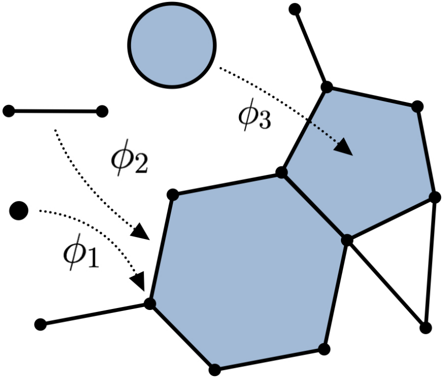
Cell complexes

The subset inclusion property of simplicial complexes can be restrictive, but we still want to exploit the topological tools they come with.

Definition (Informal)

A (regular) cell complex is a topological space X formed of a disjoint union of subspaces called *cells* such that:

1. Each cell is homeomorphic to $\mathbb{R}^n$, for some n.
2. The closure of each cell is homeomorphic to a closed ball in $\mathbb{R}^n$.

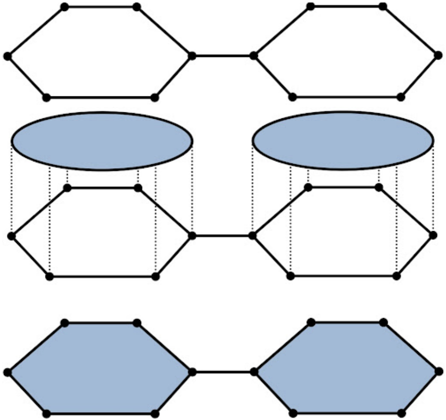


A cell complex X and the corresponding homeomorphisms to the closed balls for three cells of different dimensions in the complex.

Constructing cell complexes

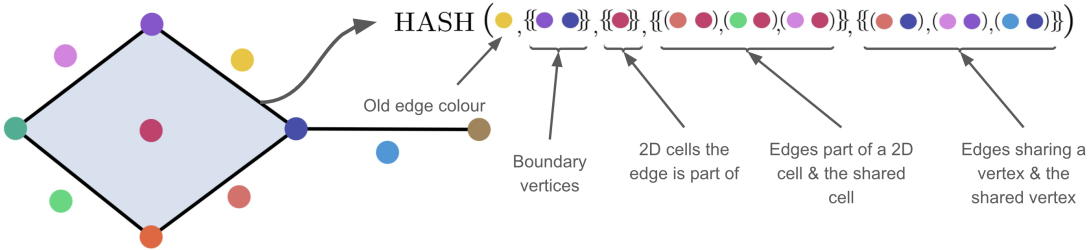
Cell complexes can be constructed hierarhically:

1. Start with a set of vertices.
2. Glue the boundary of a set of line segments to these vertices.
3. Glue the boundary of two-dimensional disks to cycles present in the graph previously obtained.



Cellular Weisfeiler Lehman

Generalising the Weisfeiler-Lehman algorithm for graphs, we can define a cellular version of the WL test. We call this *cellular WL*.



An example of a colour refinement step of CWL for an edge of the cell complex.

Theorem

CWL without coboundary and lower-adjacencies has the same expressive power as CWL with the complete set of adjacencies.

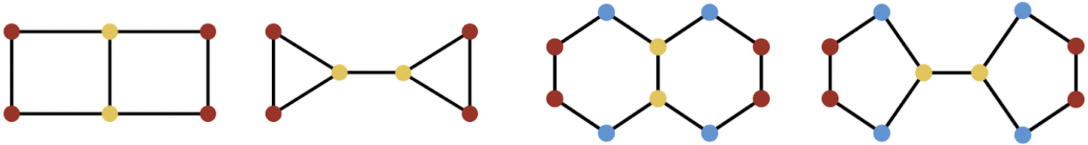
Expressive power of CWL

Definition

Let k-CL, k-IC, k-C be the "lifting" maps attaching cells to all the cliques, induced cycles and simple cycles, respectively, of size at most k.

Theorem

For $k \geq 3$, CWL(k-CL), CWL(k-IC) and CWL(k-C) are strictly more powerful than WL.



Pairs of graphs WL cannot distinguish but CWL can.

Message Passing with CW Networks

The cells receive two types of messages:

$$m_{\mathcal{B}}^{t+1}(\sigma) = \text{AGG}_{\tau \in \mathcal{B}(\sigma)}\Big(M_{\mathcal{B}}\big(h_\sigma^t, h_\tau^t\big)\Big)$$

$$m_{\uparrow}^{t+1}(\sigma) = \text{AGG}_{\tau \in \mathcal{N}_\uparrow(\sigma), \delta \in \mathcal{C}(\sigma,\tau)}\Big(M_{\uparrow}\big(h_\sigma^t, h_\tau^t, h_\delta^t\big)\Big)$$

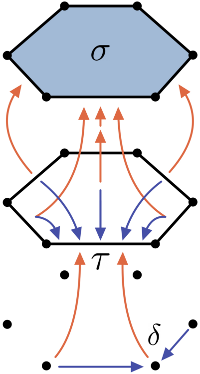
The update function takes as input these messages:

$$h_\sigma^{t+1} = U\Big(h_\sigma^t, m_{\mathcal{B}}^t(\sigma), m_{\uparrow}^{t+1}(\sigma)\Big)$$

A readout function computes a final representation:

$$\text{READOUT}(\{\{h_\sigma^L\}\}_{dim(\sigma)=0}, \{\{h_\sigma^L\}\}_{dim(\sigma)=1}, \{\{h_\sigma^L\}\}_{dim(\sigma)=2})$$

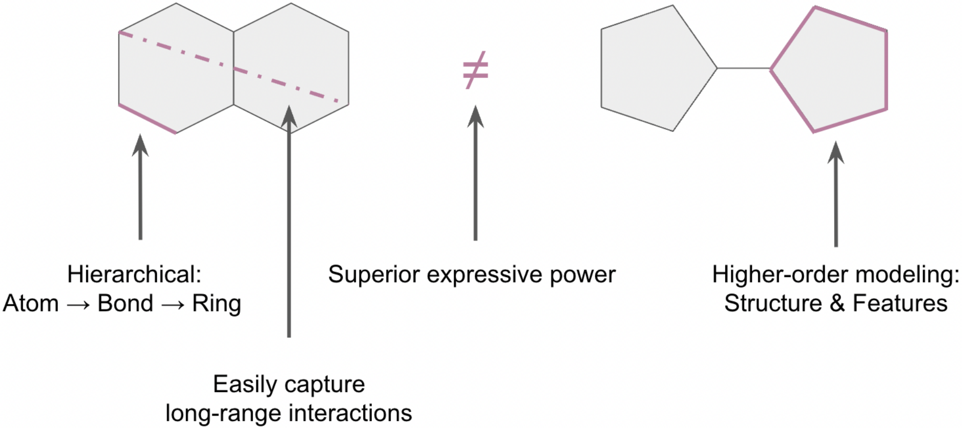
Related works: [M. Hajij, et al., 2020, M. Hajij, et al., 2021]



Orange arrows indicate boundary messages received by cells σ and τ, while blue ones show upper messages received by cells τ and δ

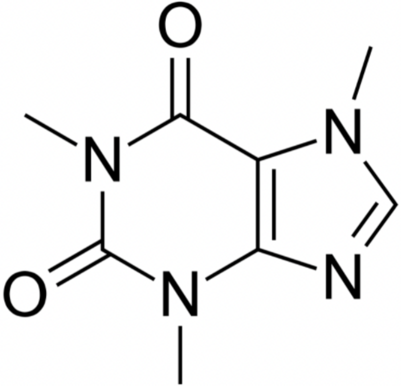
Advantages of CW Networks

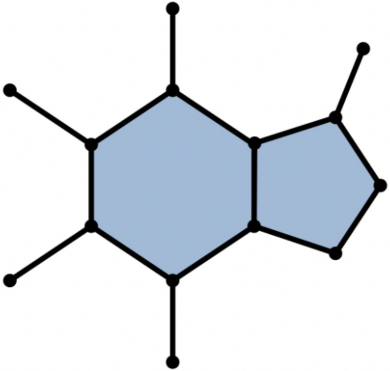
CW Networks are *hierarhical*, exploit *higher-order structures*, capture *long-range interactions* and have *superior expressive power*.



Molecules and Cell Complexes

Graph representations of molecules date back to the nineteenth century. However, it is not necessarily the best representation. We propose modelling molecules as cell complexes.





Molecular Property Prediction

Table 3: ZINC (MAE), ZINC-FULL (MAE) and Mol-HIV (ROC-AUC).

Method	ZINC $\downarrow$ No Edge Feat.	ZINC $\downarrow$ With Edge Feat.	ZINC-FULL $\downarrow$ All methods	MOLHIV $\uparrow$ All methods
GCN [45]	0.469±0.002	N/A	N/A	76.06±0.97
GAT [67]	0.463±0.002	N/A	N/A	N/A
GatedGCN [10]	0.422±0.006	0.363±0.009	N/A	N/A
GIN [72]	0.408±0.008	0.252±0.014	0.088±0.002	77.07±1.49
PNA [19]	0.320±0.032	0.188±0.004	N/A	79.05±1.32
DGN [5]	0.219±0.010	0.168±0.003	N/A	79.70±0.97
HIMP [26]	N/A	0.151±0.006	0.036±0.002	78.80±0.82
GSN [9]	0.139±0.007	0.108±0.018	N/A	77.99±1.00
CIN-small (Ours)	0.139±0.008	0.094±0.004	0.044±0.003	80.55±1.04
CIN (Ours)	**0.115±0.003**	**0.079±0.006**	**0.022±0.002**	**80.94±0.57**

Review

For more learning resources: [V. Nanda, 2020, Crane, 2021].

1. Simplicial Complexes

1.1 Orientations
1.2 Chains
1.3 Hodge Theory
1.4 Convolutions and Symmetries
1.5 Application: Trajectory Classification

2. Cell Complexes

2.1 Definition and Construction
2.2 Cellular Weisfeiler Lehman
2.3 Message Passing on Cell Complexes
2.4 Application: Molecular Property Prediction

References

Cristian Bodnar*, Fabrizio Frasca*, Yu Guang Wang*, Nina Otter, Guido Montúfar*, Pietro Liò and Michael Bronstein (2021)
Weisfeiler and Lehman Go Topological: Message Passing Simplicial Networks
ICML 2021

Cristian Bodnar*, Fabrizio Frasca*, Nina Otter, Yu Guang Wang, Pietro Liò, Guido Montúfar and Michael Bronstein (2021)
Weisfeiler and Lehman Go Cellular: CW Networks
NeurIPS 2021

Stefania Ebli, Michaël Defferrard, Gard Spreemann
Simplicial Neural Networks
Topological Data Analysis and Beyond Workshop at NeurIPS 2020

Eric Bunch, Qian You, Glenn Fung, Vikas Singh
Simplicial 2-Complex Convolutional Neural Nets
Topological Data Analysis and Beyond Workshop at NeurIPS 2020

References

T. Mitchell Roddenberry, Nicholas Glaze, Santiago Segarra
Principled Simplicial Neural Networks for Trajectory Prediction
ICML 2021

Michael T. Schaub, Austin R. Benson, Paul Horn, Gabor Lippner, Ali Jadbabaie
Random Walks on Simplicial Complexes and the normalized Hodge 1-Laplacian
SIAM Review, June 2020

Abubakr Muhammad, Magnus Egerstedt
Control Using Higher Order Laplacians in Network Topologies
Proc. of 17th International Symposium on Mathematical Theory of Networks and Systems, Kyoto, 2006

Lek-Heng Lim
Hodge Laplacians on Graphs
SIAM Review, 2019

References

Sergio Barbarossa, Stefania Sardellitti
Topological Signal Processing over Simplicial Complexes
IEEE Transactions on Signal Processing, March 2020

Mustafa Hajij, Kyle Istvan, Ghada Zamzmi
Cell Complex Neural Networks
Topological Data Analysis and Beyond Workshop at NeurIPS 2020

Mustafa Hajij, Ghada Zamzmi, Xuanting Cai
Simplicial Complex Representation Learning
Preprint, 2021

Vidit Nanda
Computational Algebraic Topology
University of Oxford Course, 2020-2021 - https://people.maths.ox.ac.uk/nanda/cat/

Keenan Crane
Discrete Differential Geometry
CMU Course, Spring 2021 - https://brickisland.net/DDGSpring2021/

The End