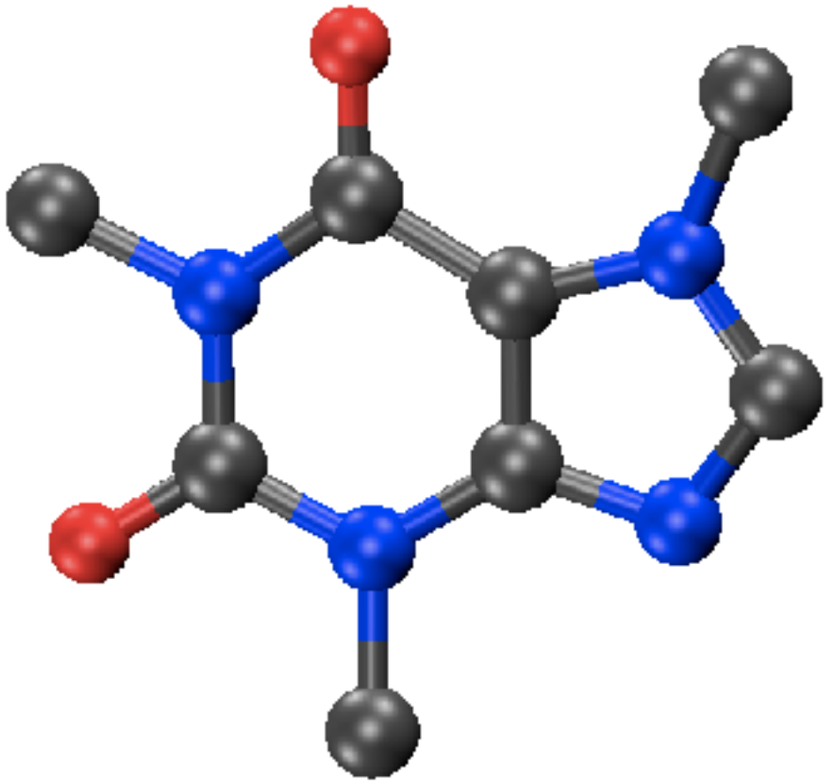
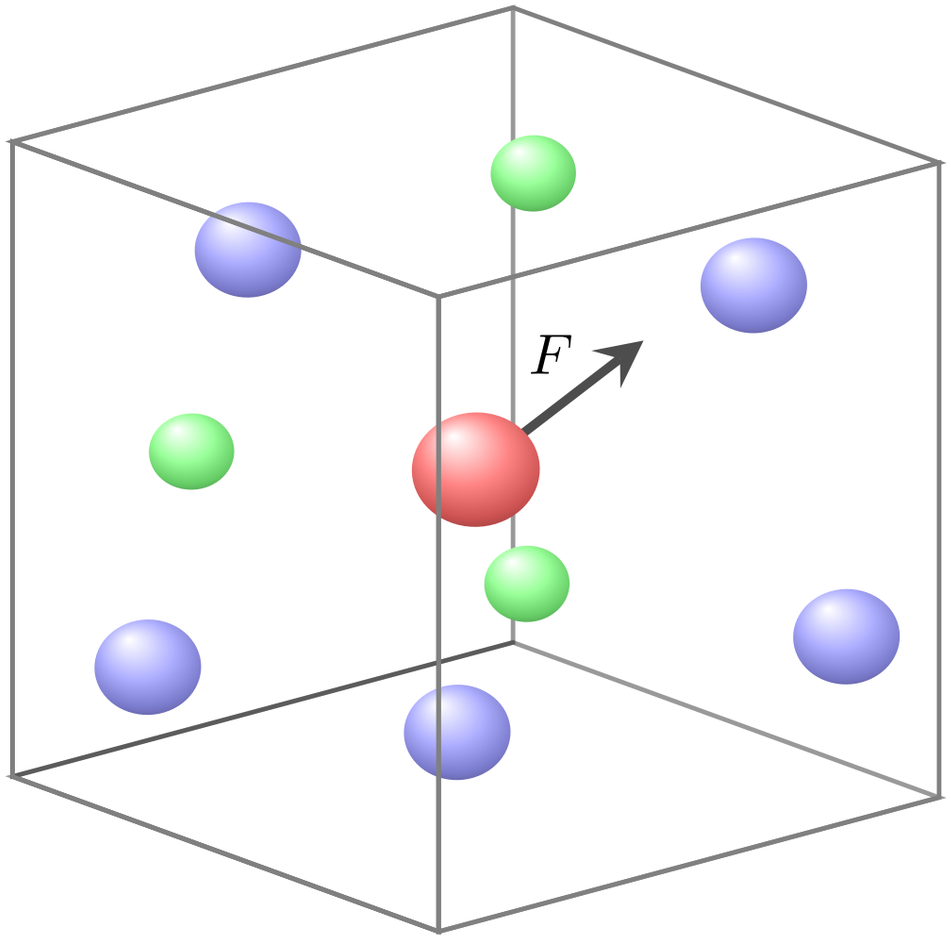
Covariant neural networks for learning physical systems

Risi Kondor
The University of Chicago



$\phi(G)$

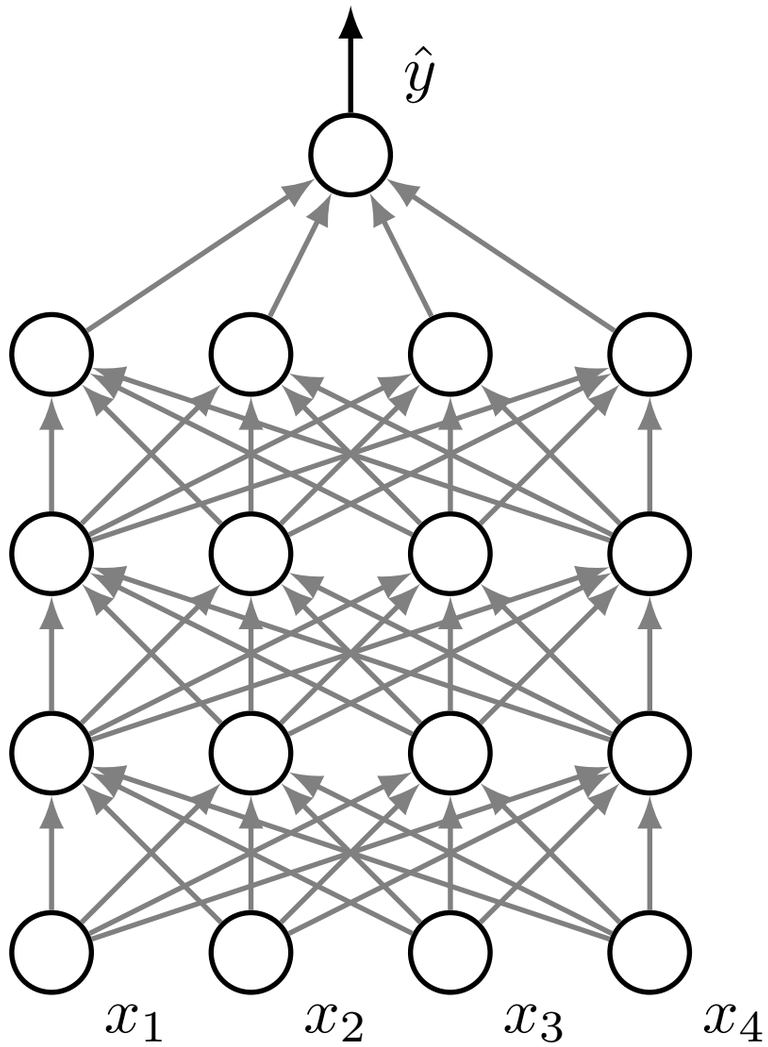


$F(\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_m)$

1. Multiscale structure and equivariance

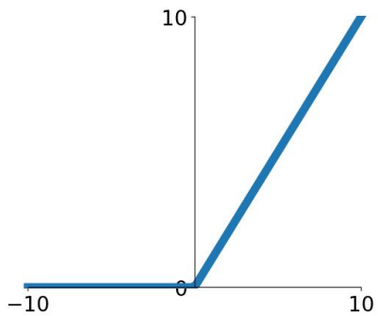
Feed-forward Neural Networks

$$f_{\text{out}} = \sigma\left(\sum_i w_i f_{\text{in}}^{(i)} + b\right)$$

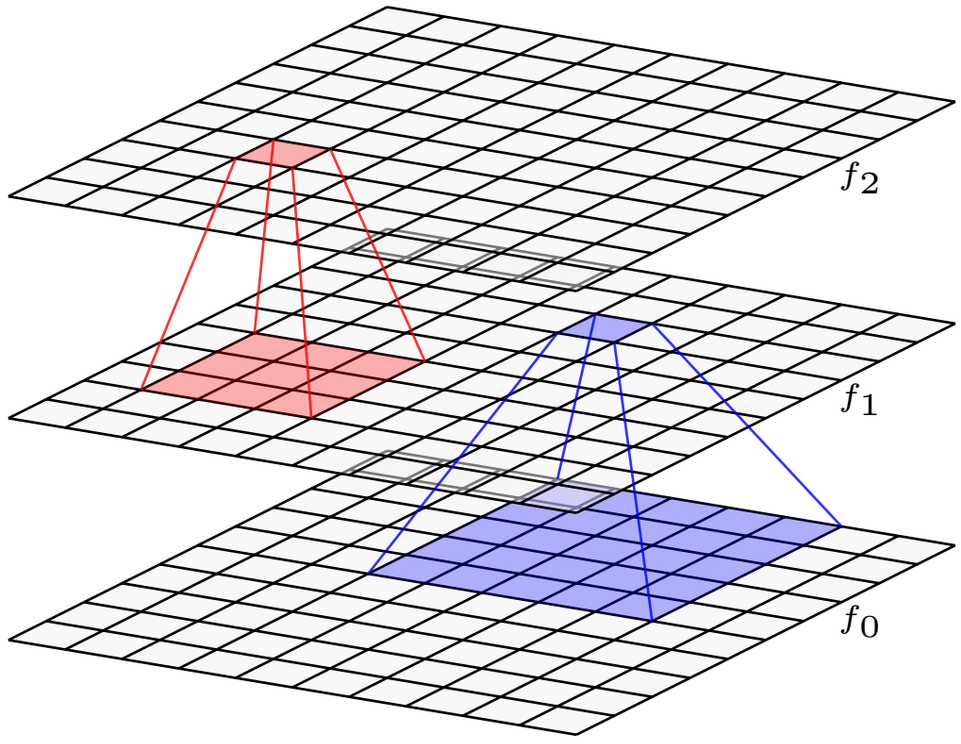


Common choice of nonlinearity:

$$\sigma_{\text{ReLU}}(z) = \max(0, x)$$



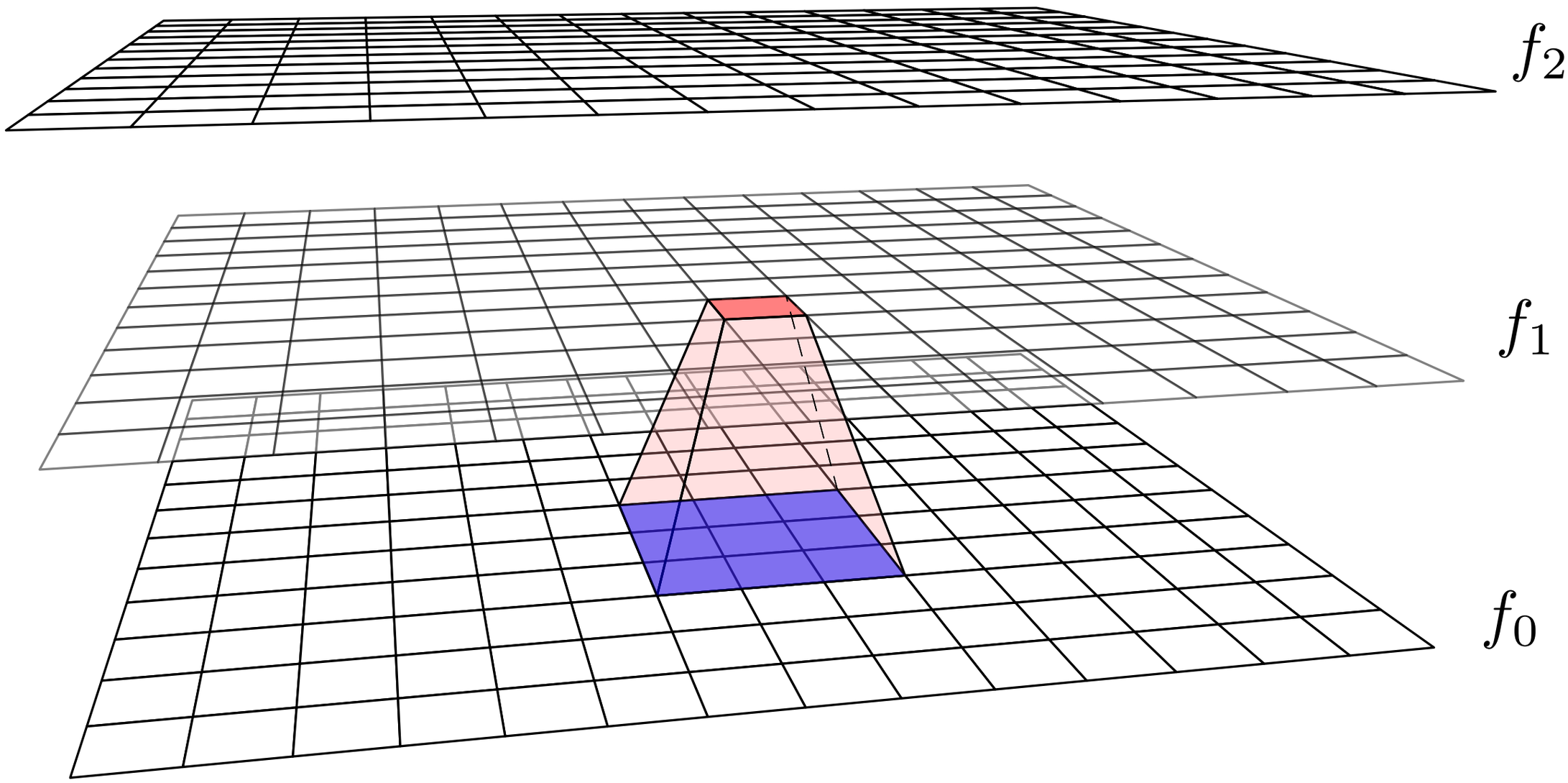
Convolutional Neural Networks



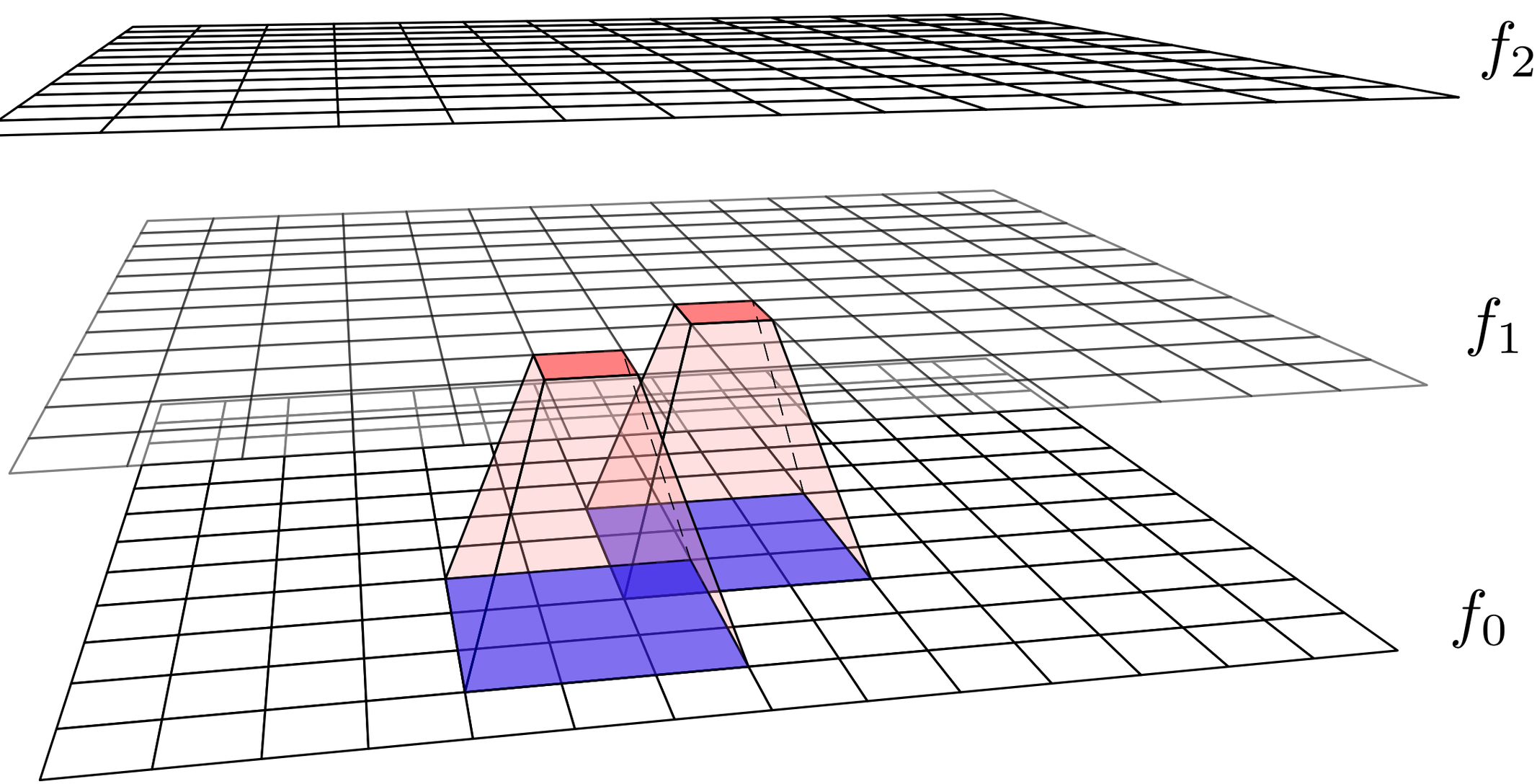
$$\phi_\ell(f_{\ell-1}) = (f_{\ell-1} * g_\ell)(x) = \sum_{y \in \mathbb{Z}^2} f_{\ell-1}(x-y)\, g_\ell(y)$$

Filter at layer ℓ

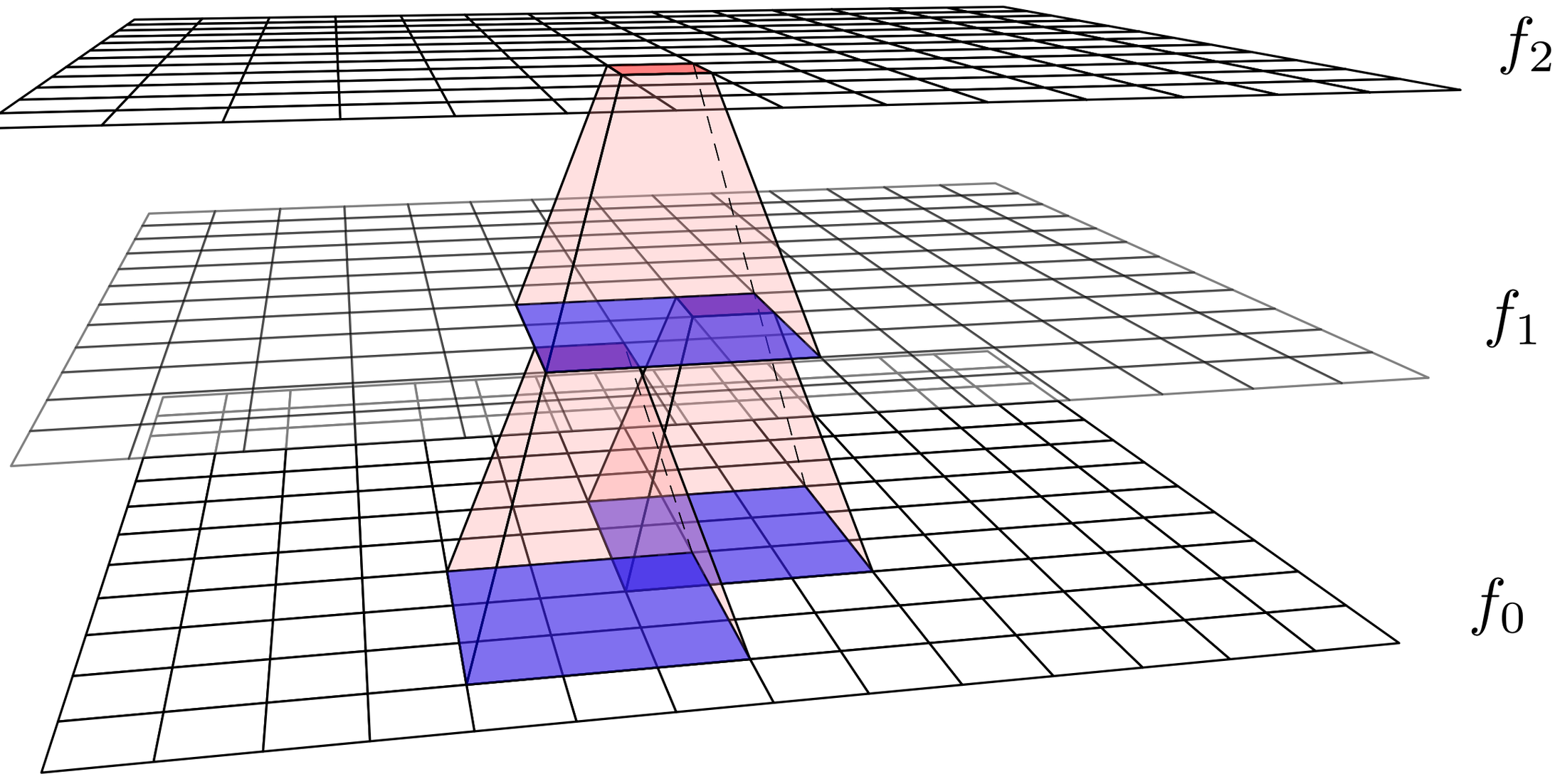
Multiscale structure



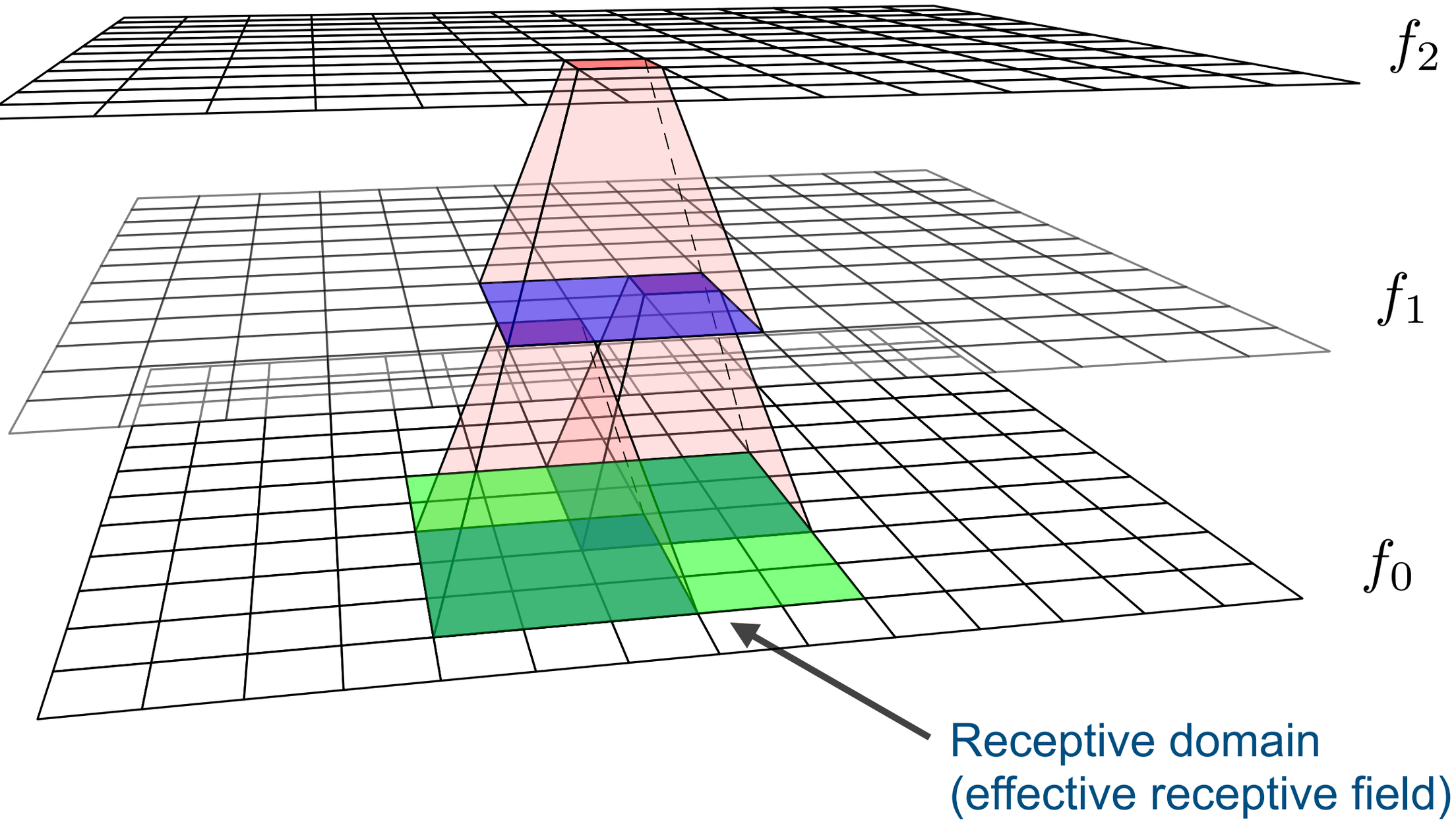
Multiscale structure



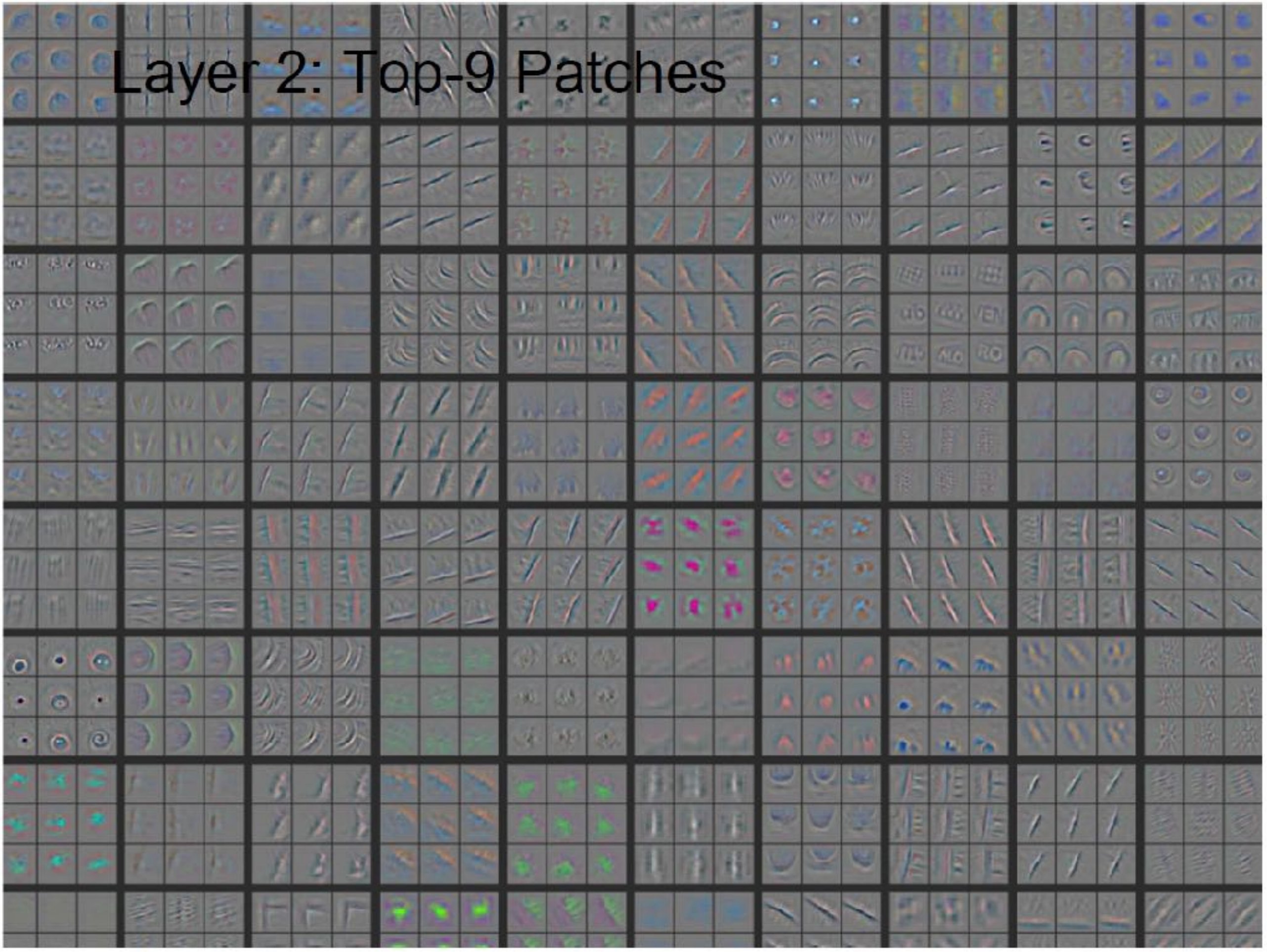
Multiscale structure



Multiscale structure



Layer 2: Top-9 Patches

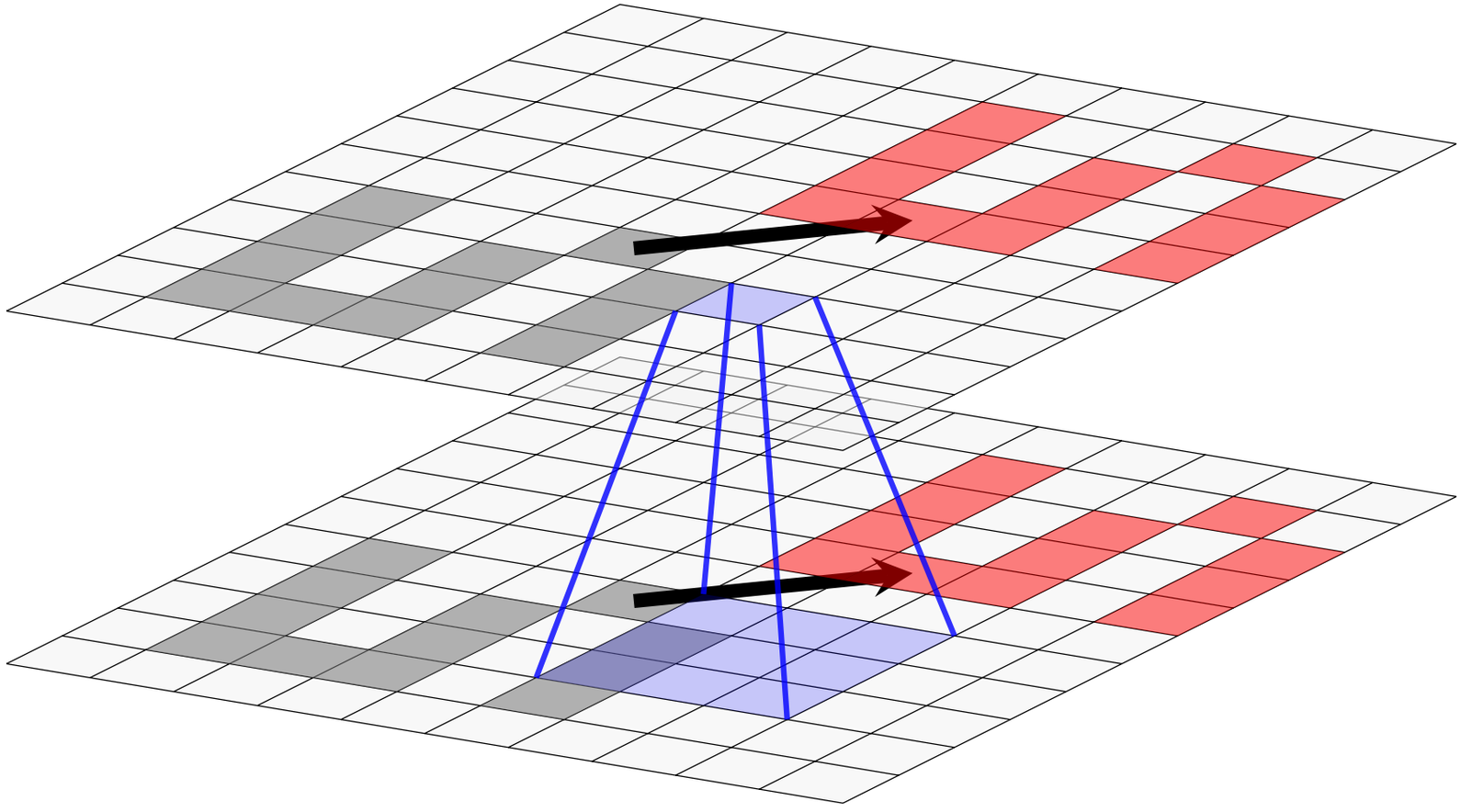


Layer 3: Top-9 Patches

Layer 4: Top-9 Patches

Equivariance to translations

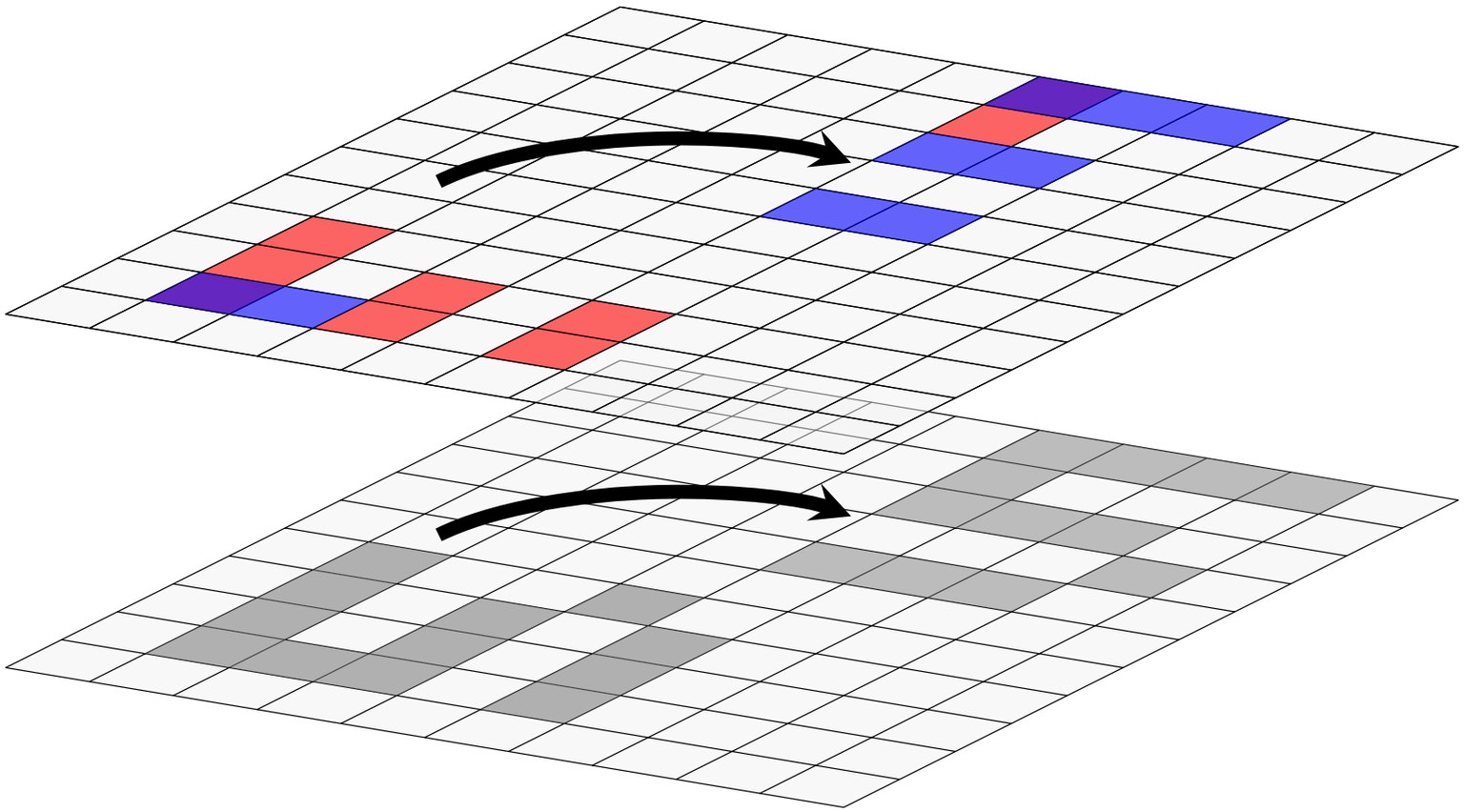
f_ℓ



$f_{\ell-1}$

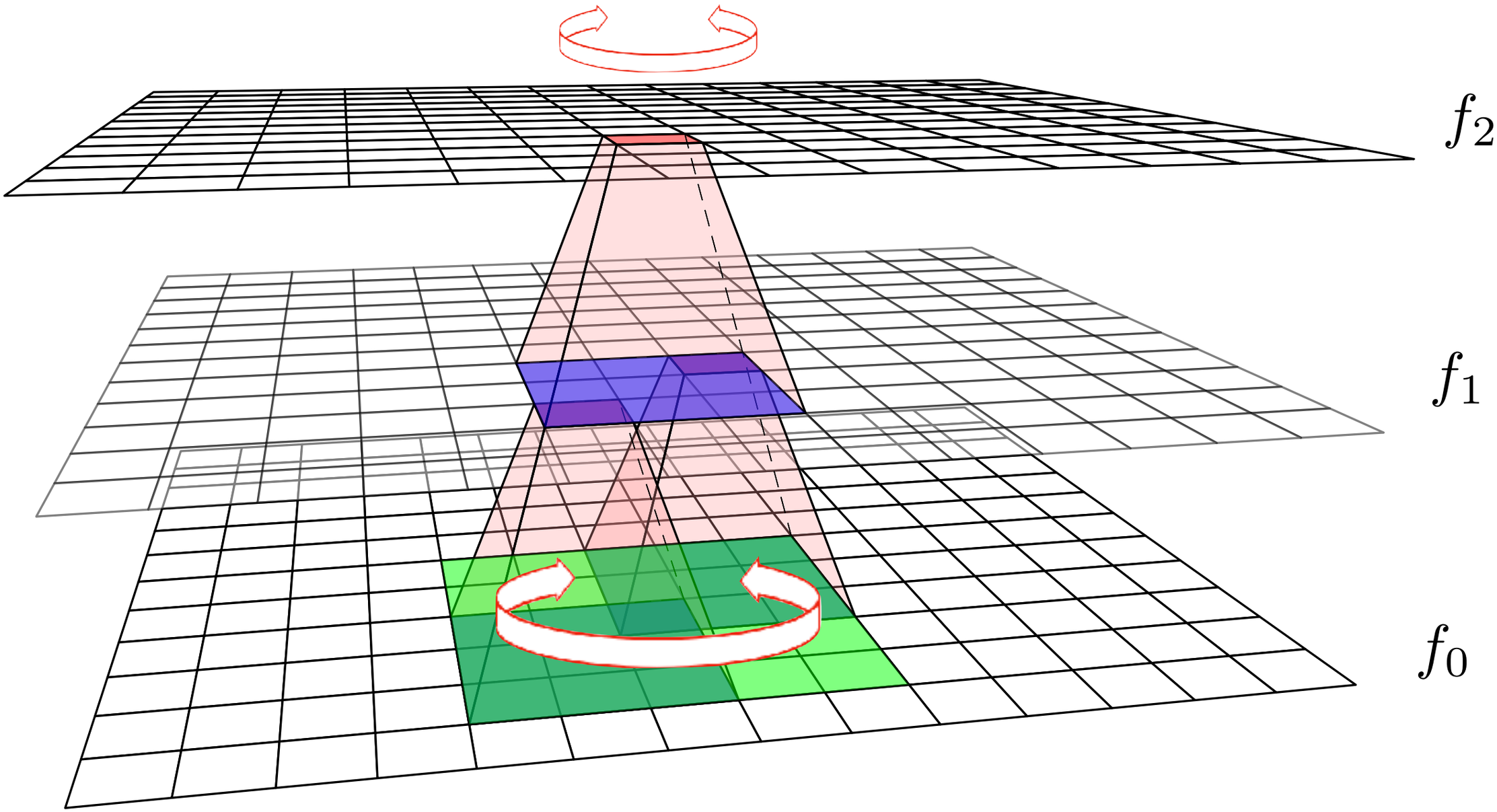
Steerability

f_ℓ



$f_{\ell-1}$

Equivariance (covariance)



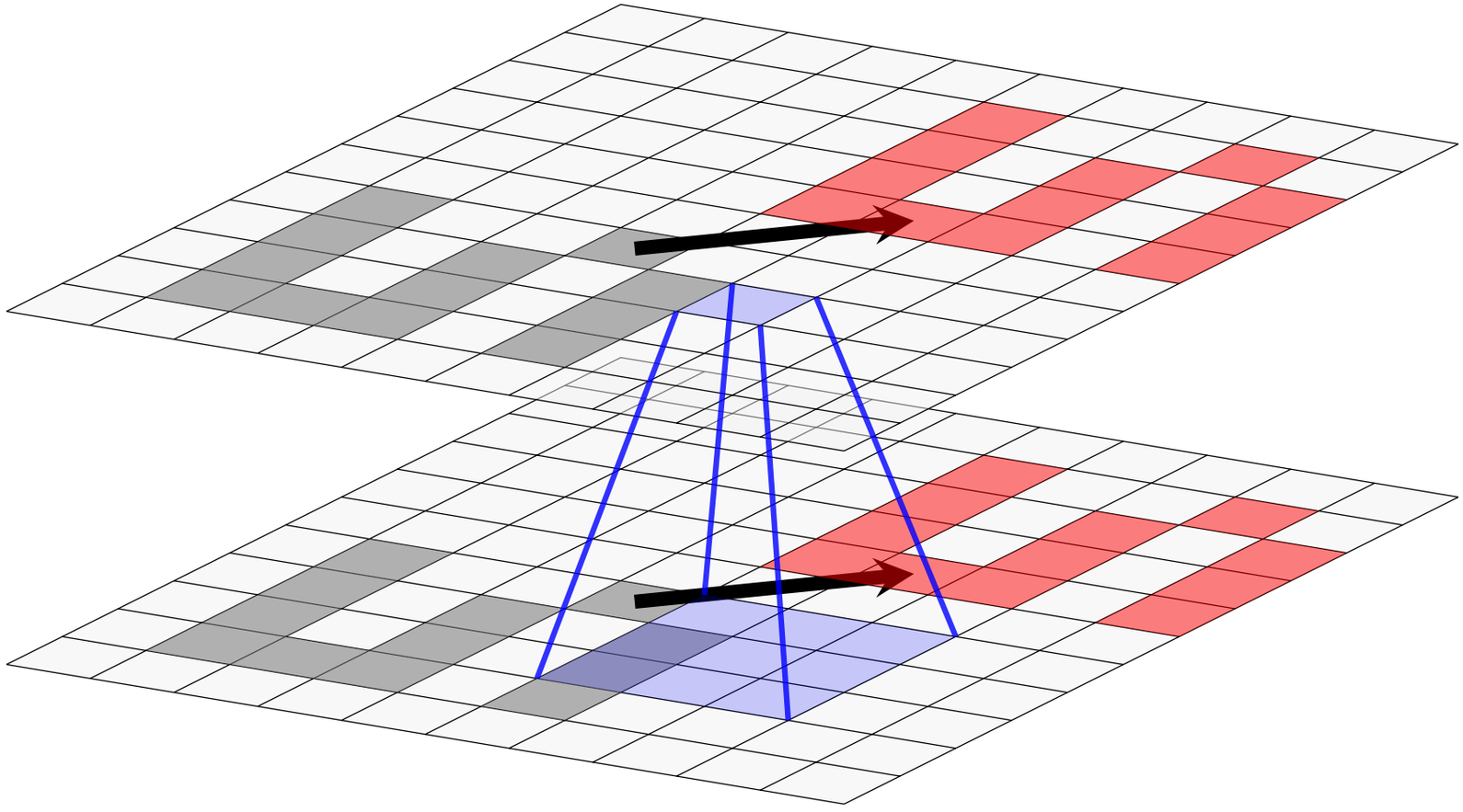
[Cohen & Welling, 2016]

What is the analog of convolutions on graphs?

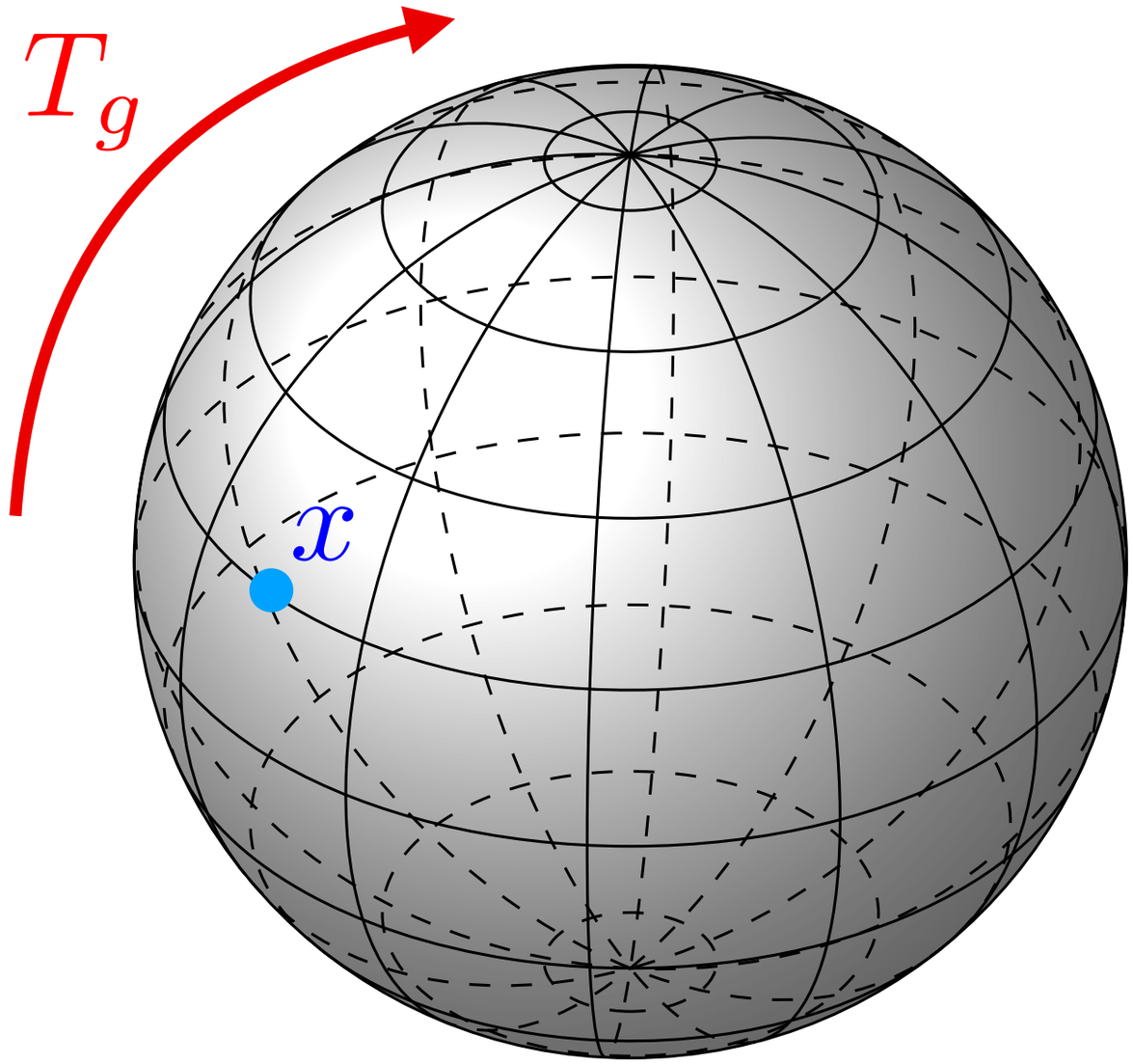
2. Theory

How do we generalize convolution to the action of general (compact) groups?

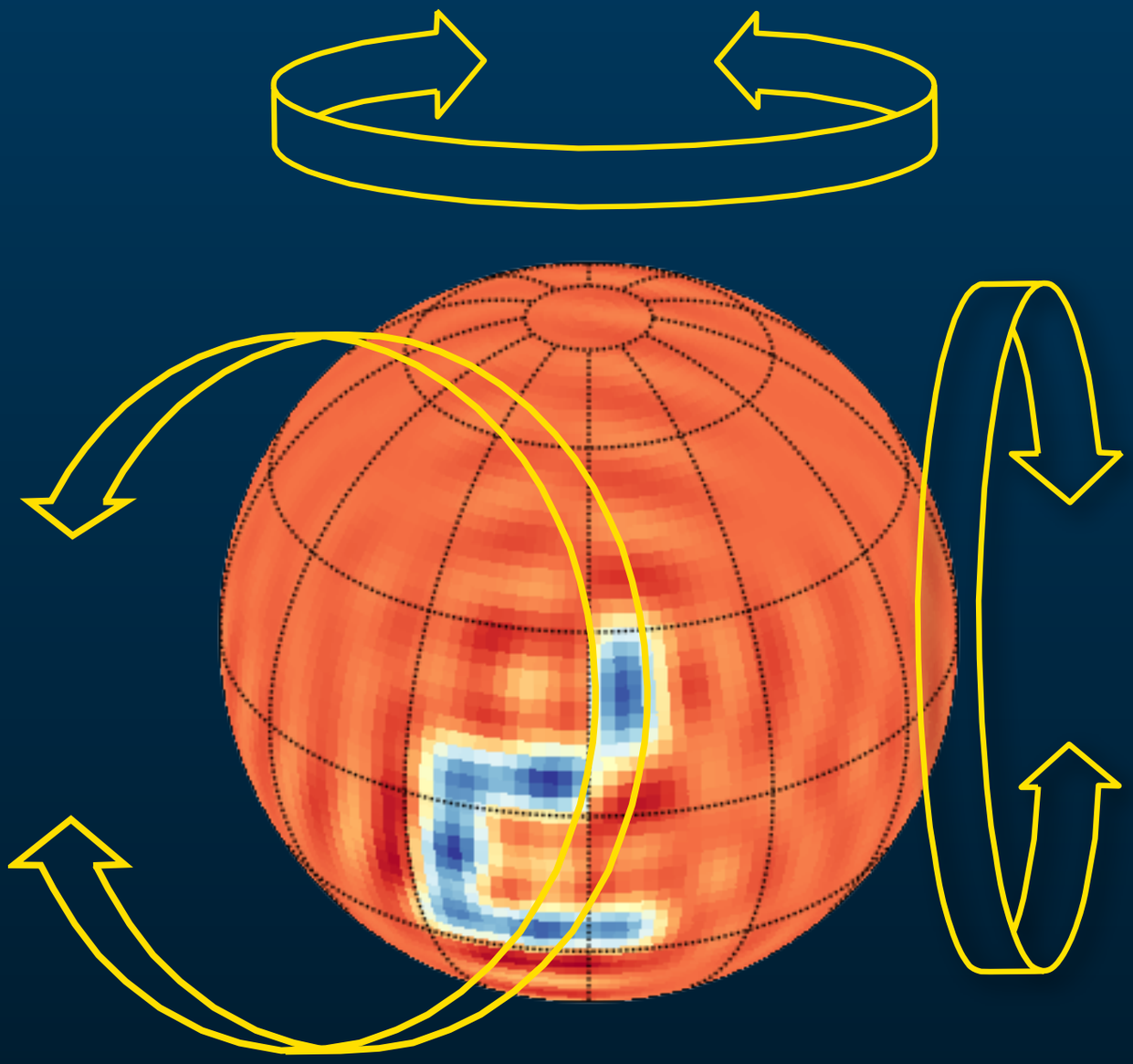
f_ℓ



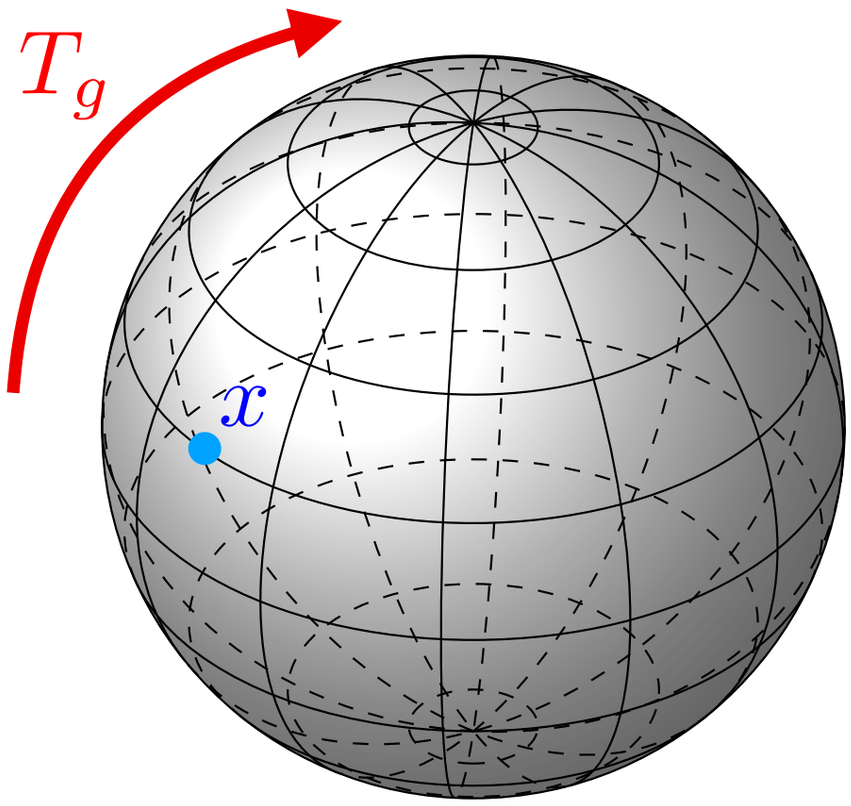
$f_{\ell-1}$



[Cohen, Geiger, Köhler & Welling, 2018]



Group actions



1. Our function lives on a space $\mathcal{X}$

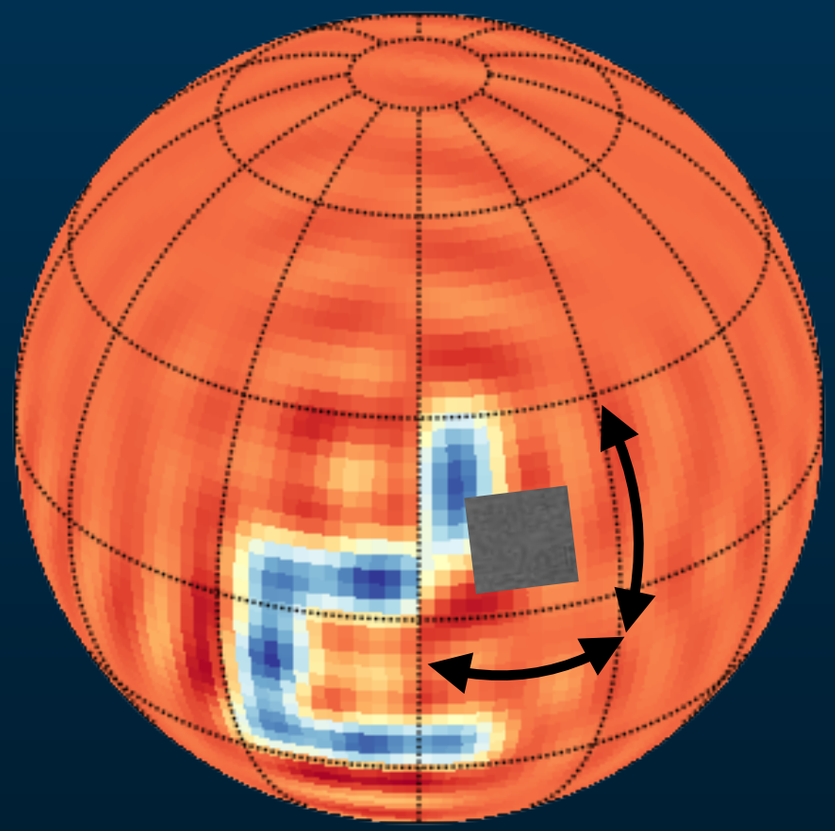
$$f : \mathcal{X} \to \mathbb{C}$$

2. We have a group G acting on $\mathcal{X}$

$$x \mapsto T_g(x)$$

3. This induces an action on functions

$$f \overset{\mathbb{T}_g}{\longmapsto} f' \qquad f'(x) = f(T_g^{-1}(x))$$



$$(h \star f)(R) = \frac{1}{4\pi} \int_0^{2\pi} \int_{-\pi}^{\pi} \left[h_R(\theta, \phi)\right]^* f(\theta, \phi)\ \cos\theta \, d\theta \, d\phi$$

Equivariance

$$
\begin{array}{ccc}
L(\mathcal{X}_1) & \xrightarrow{\mathbb{T}_g^{(1)}} & L(\mathcal{X}_1) \\
\downarrow{\scriptstyle \phi} & & \downarrow{\scriptstyle \phi} \\
L(\mathcal{X}_2) & \xrightarrow{\mathbb{T}_g^{(2)}} & L(\mathcal{X}_2)
\end{array}
$$

$$(f * g)(u) = \int_G f(uv^{-1})\, g(v)\, d\mu(v)$$

$$(f * g)(u) = \int_G f{\uparrow^G}(uv^{-1})\, g{\uparrow^G}(v)\, d\mu(v)$$

Theorem

A feed-forward neural network is equivariant to the action of a compact group G if and only if the linear operation in each layer is of the form

$$\phi_\ell(f_{\ell-1}) = f_{\ell-1} * g_\ell.$$

where $*$ denotes the generalization of convolution to homogeneous space of compact groups, defined

$$(f * g)(u) = \int_G f{\uparrow}^G(uv^{-1})\, g{\uparrow}^G(v)\, d\mu(v)$$

[K and Trivedi, 2018]

Convolution on groups

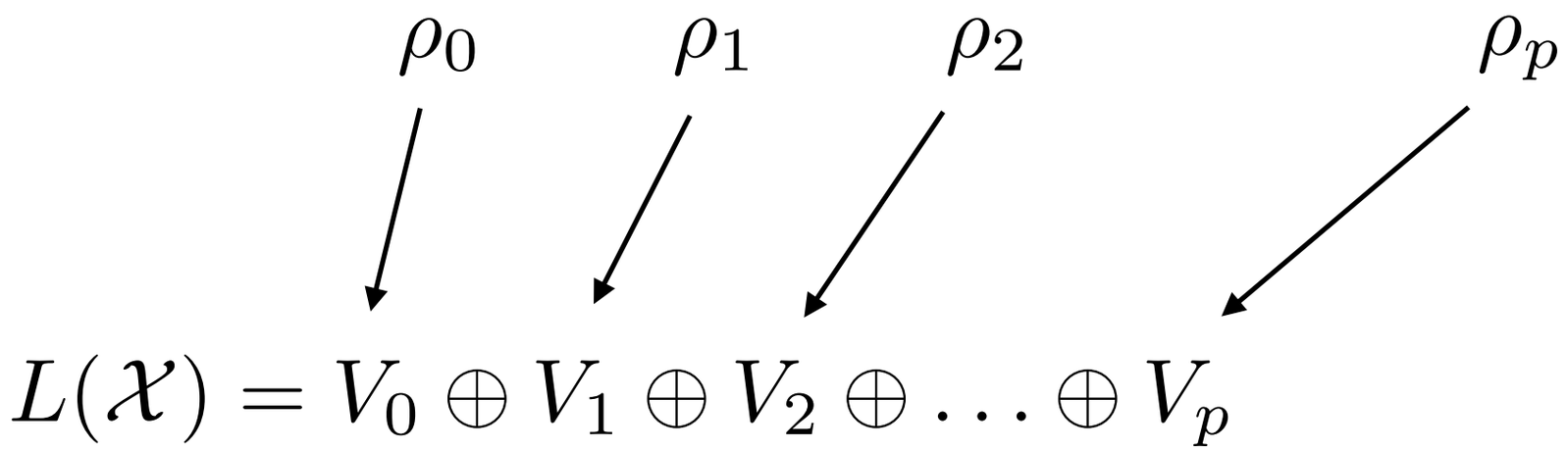
$$\widehat{f}(k) = \int f(x)\, e^{-ikx}\, dx$$

$$\widehat{(f * \chi)}(k) = \widehat{f}(k) \cdot \widehat{\chi}(k)$$

$$\widehat{f}(\rho) = \int f(x)\, \rho(x)\, d\mu(x)$$

$$(f * g)(u) = \int_G f(uv^{-1})\, g(v)\, d\mu(v)$$

$$\widehat{(f * \chi)}(\rho) = \widehat{f}(\rho) \cdot \widehat{\chi}(\rho)$$



$$V_i = W_i^1 \oplus W_i^2 \oplus \ldots \oplus W_i^{m_i}$$

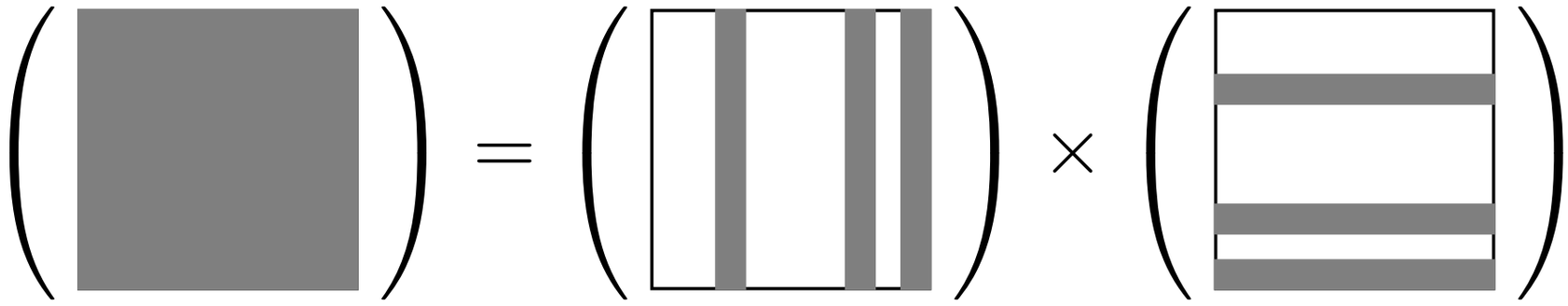
$$\widehat{f}(\rho_i) = \int_G f(u)\ \rho_i(u)\ d\mu(u) \qquad\qquad i = 0, 1, 2, \ldots$$

$$\widehat{f * g}(\rho_i) = \widehat{f}(\rho_i) \cdot \widehat{g}(\rho_i)$$

matrix multiplication

Case 1:

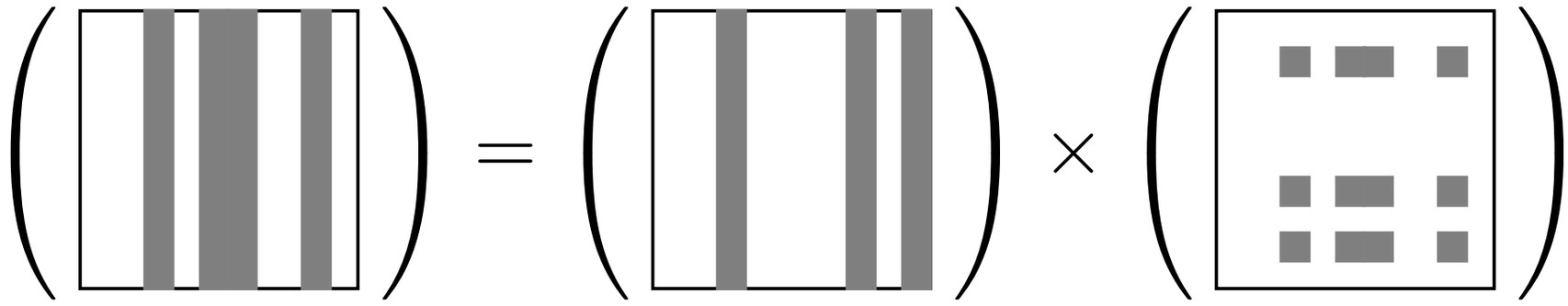
$$f_{\ell-1} \colon G/H \to \mathbb{C} \qquad f_\ell \colon G \to \mathbb{C}$$



$\widehat{f * g}(\rho)$ $\widehat{f\uparrow^G}(\rho)$ $\widehat{g\uparrow^G}(\rho)$

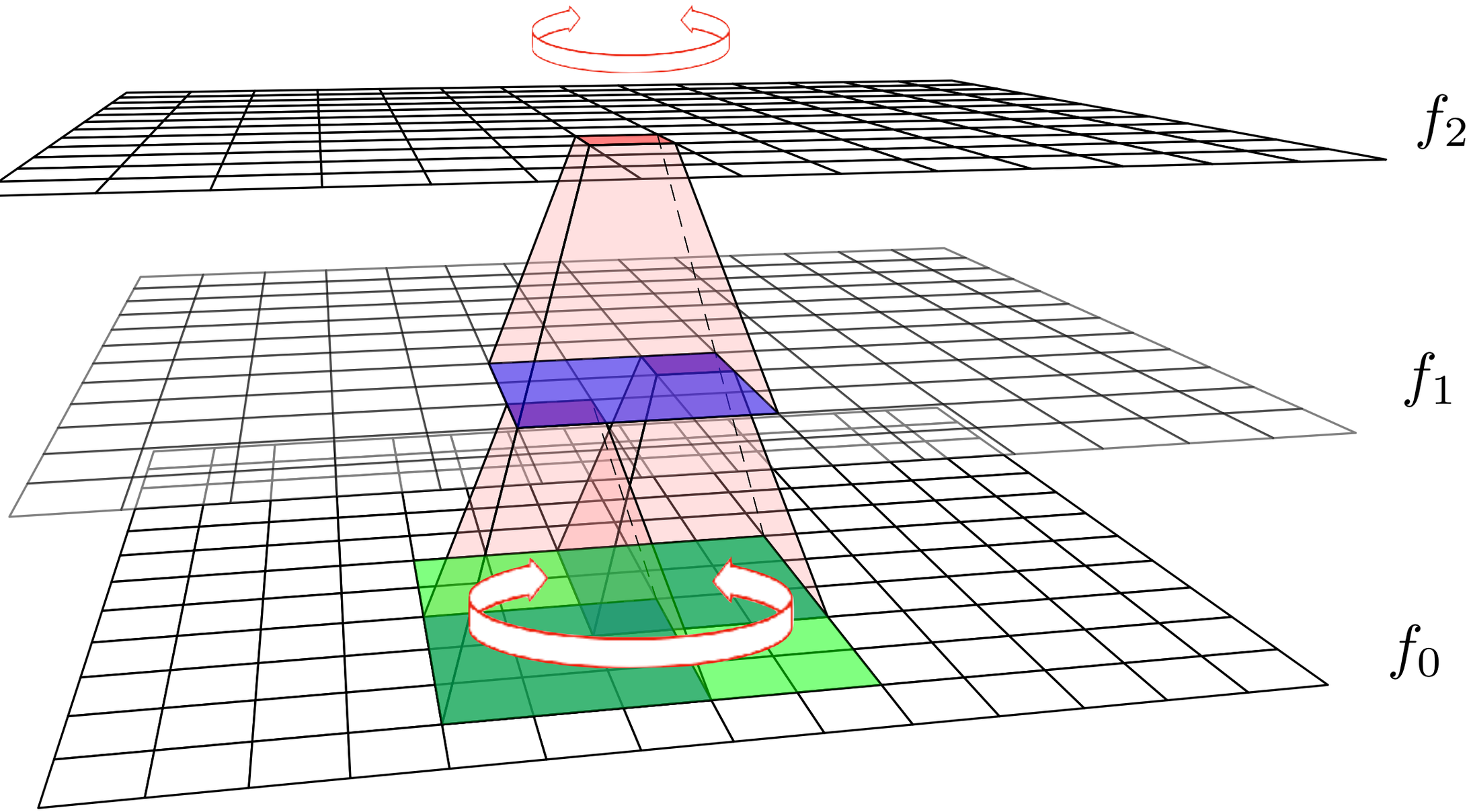
Case 2:

$$f_{\ell-1}\colon G/H \to \mathbb{C} \qquad f_{\ell}\colon G/K \to \mathbb{C}$$



$$\widehat{f * g}(\rho) \qquad \widehat{f\uparrow^G}(\rho) \qquad \widehat{g\uparrow^G}(\rho)$$

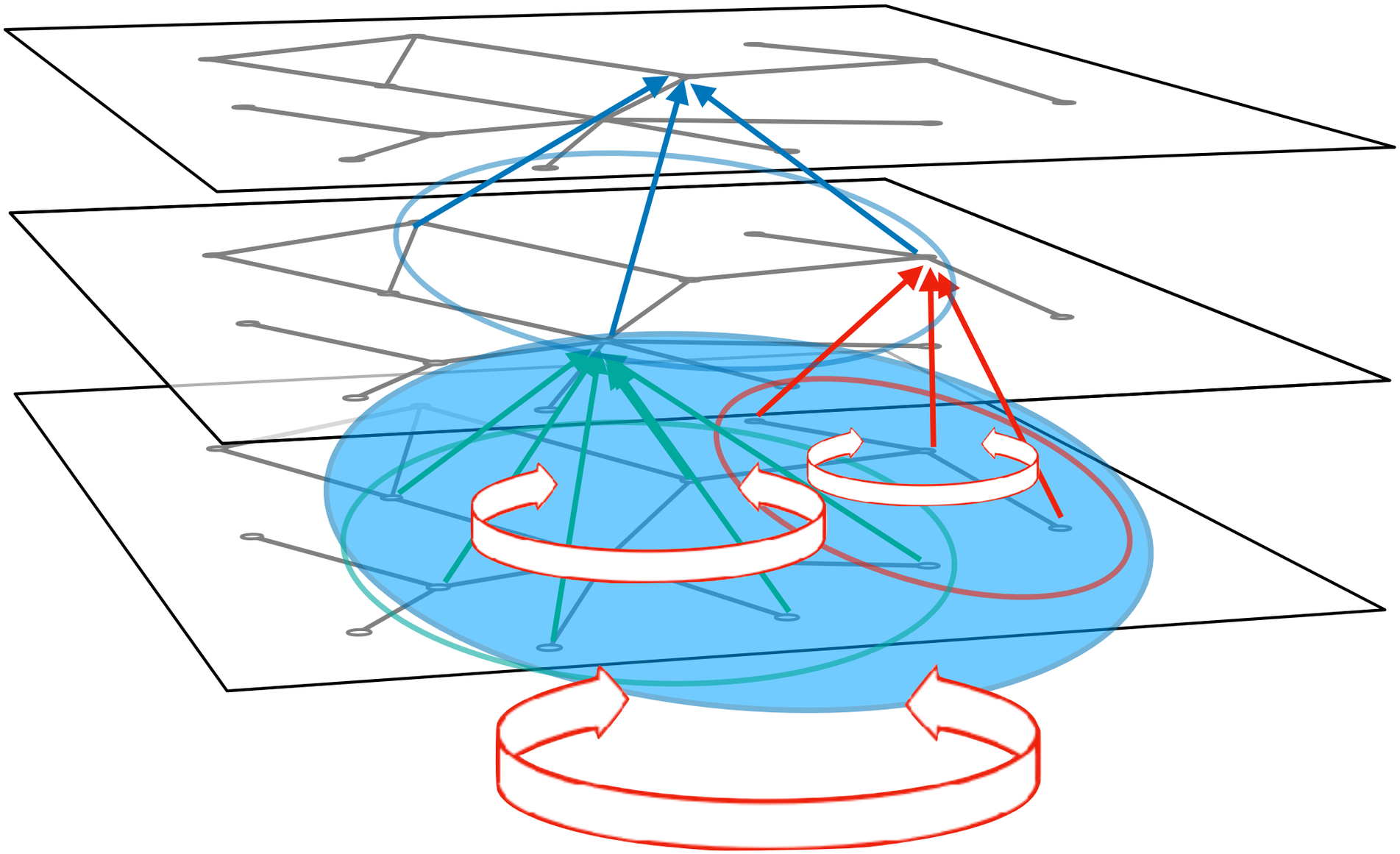
Fourier Space Neural Networks



3. Covariant graph neural networks

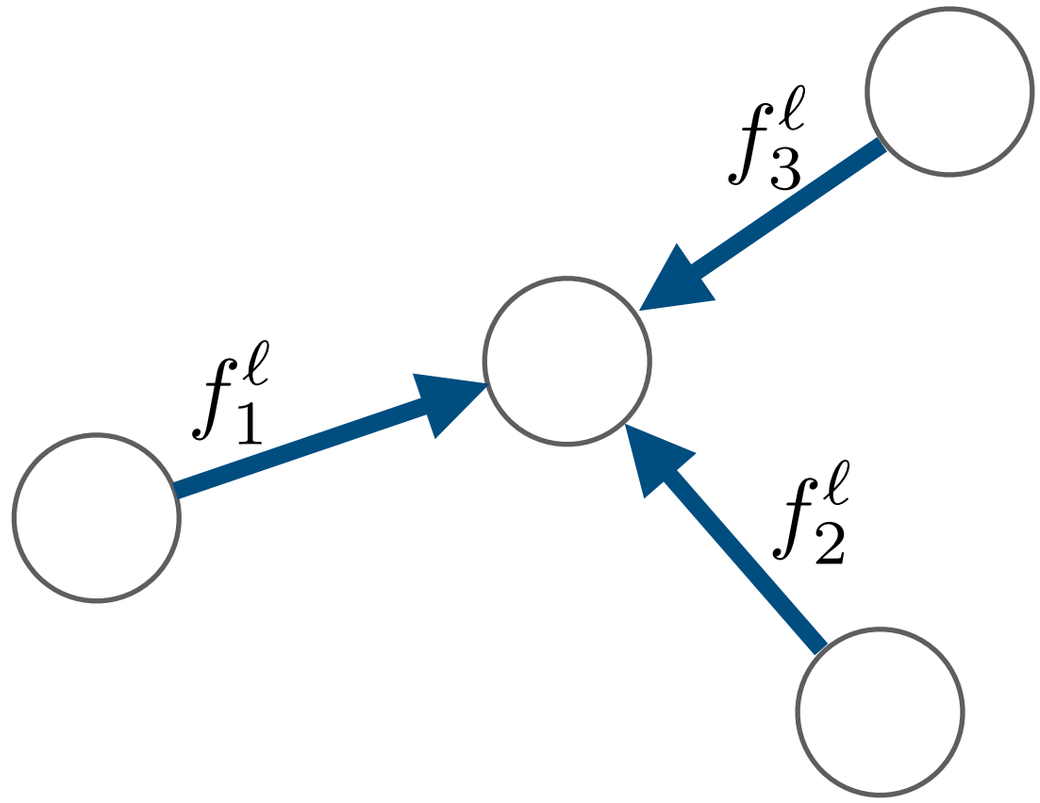
[K., Pan, Hy-Truong, Trivedi & Anderson]

Multiscale structure in graphs



Message passing neural networks (MPNNs)

$$f_i^{\ell+1} = \xi\left(W \sum_{j \in \mathcal{N}(i)} f_j^{\ell} + b\right)$$



[Gilmer et al, ’17] [Kriege, ’16] [Niepert, ’16] [Duvenaud et al., ’15] [Dai, Dai & Song, ’16]

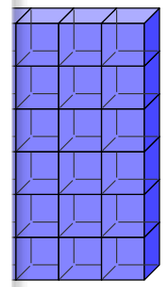
Permutation covariant activations

How does a given activation transform when its receptive field is subjected to a permutation σ?

Scalar activation
(0th

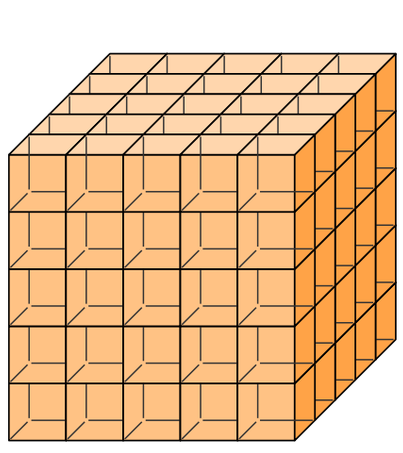
Vector activation
(1st

Matrix activation
(2nd r covariant)



$P_\sigma F_i P_\sigma^\top$

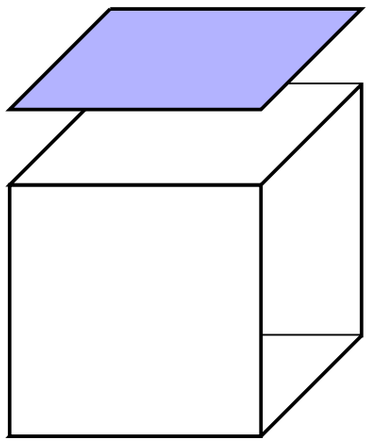
General tensor activation
(kth order covariant)



$$F_{i_1,i_2,\ldots,i_k} \overset{\sigma}{\longmapsto} [P_\sigma]_{i_1}^{j_1} [P_\sigma]_{i_2}^{j_2} \cdots [P_\sigma]_{i_k}^{j_k} F_{j_1,j_2,\ldots,j_k}$$

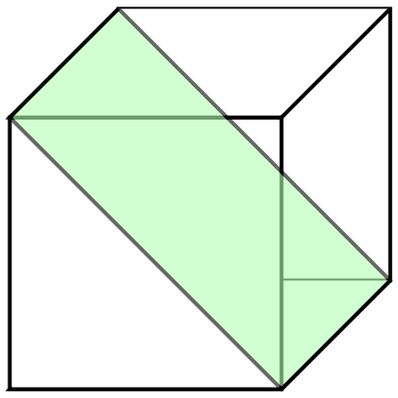
Permutation covariant operations:

1. Projections



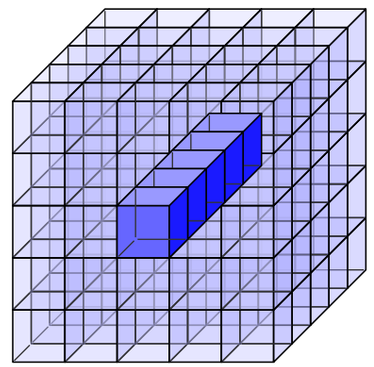
$$C_{i,j} = \sum_a A_{a,i,j}$$

2. Diagonals



$$C_{i,j} = \sum_i A_{i,i,j}$$

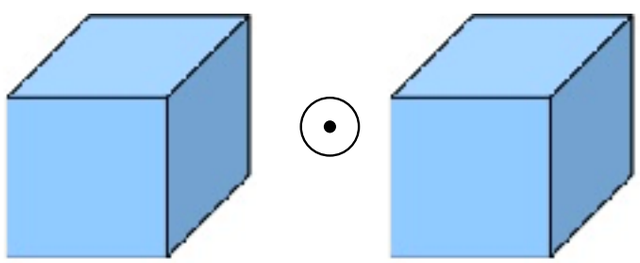
3. Contractions



$$C_k = \sum_{i,j} A_{i,j,k}$$

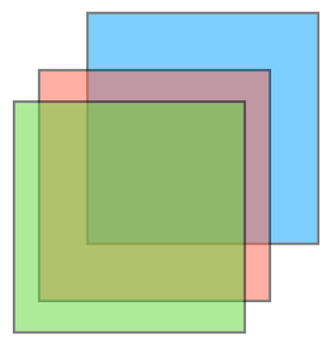
Permutation covariant operations:

4. Hadamard products



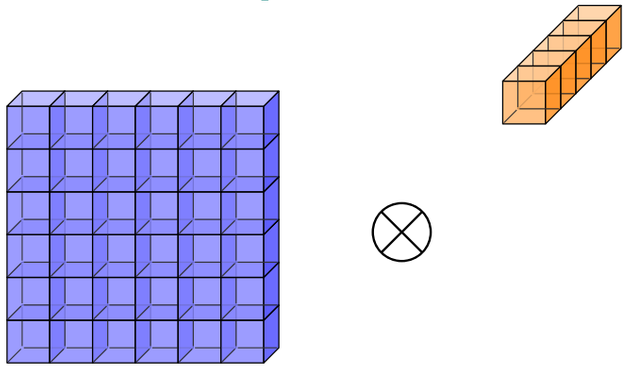
$$C_{i,j,k} = A_{i,j,k} B_{i,j,k}$$

5. Stacking



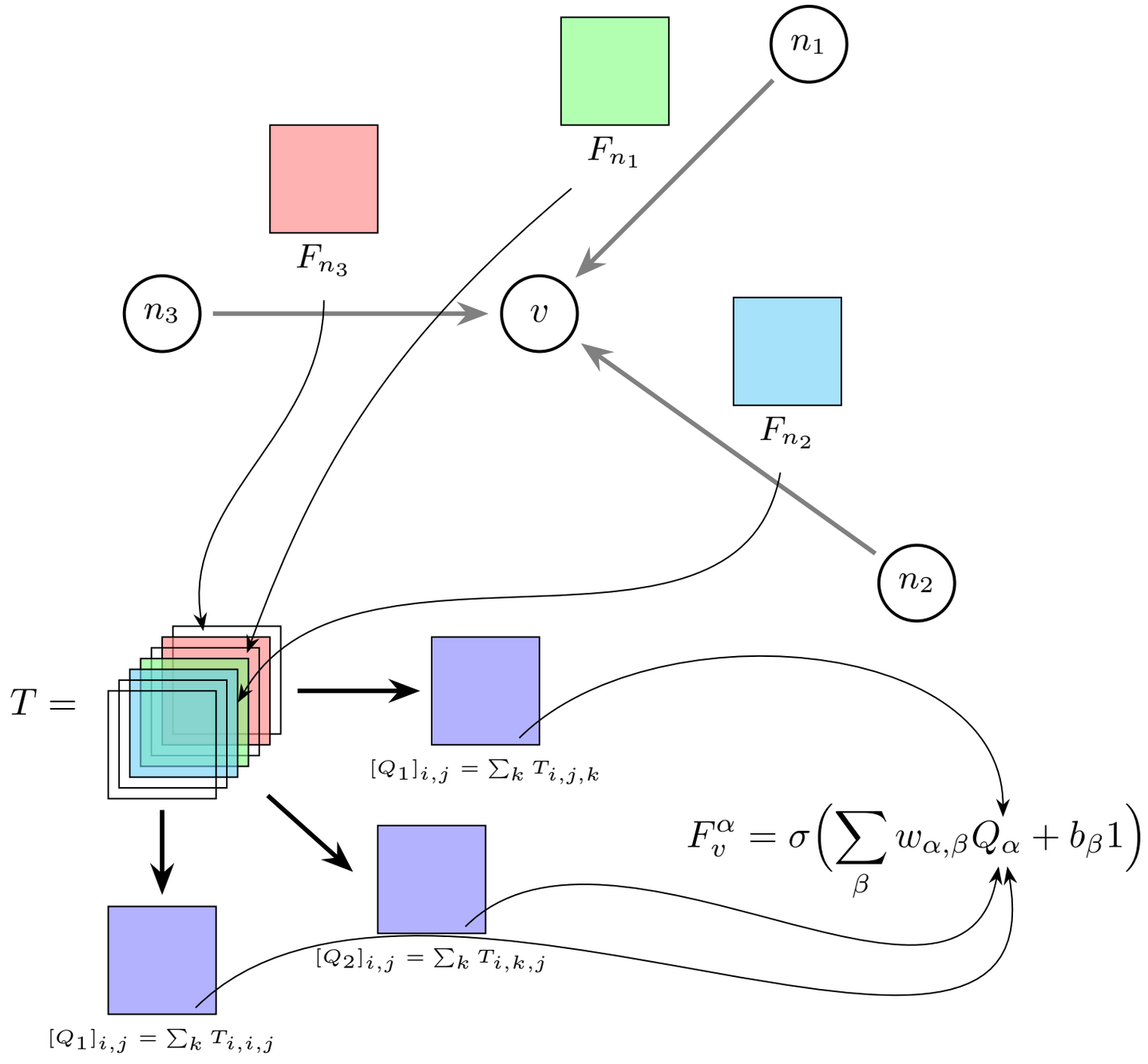
$$C_{i,j,k} = A^{(k)}_{i,j}$$

6. Tensor products



$$C_{i,j,k} = A_{i,j} B_k$$

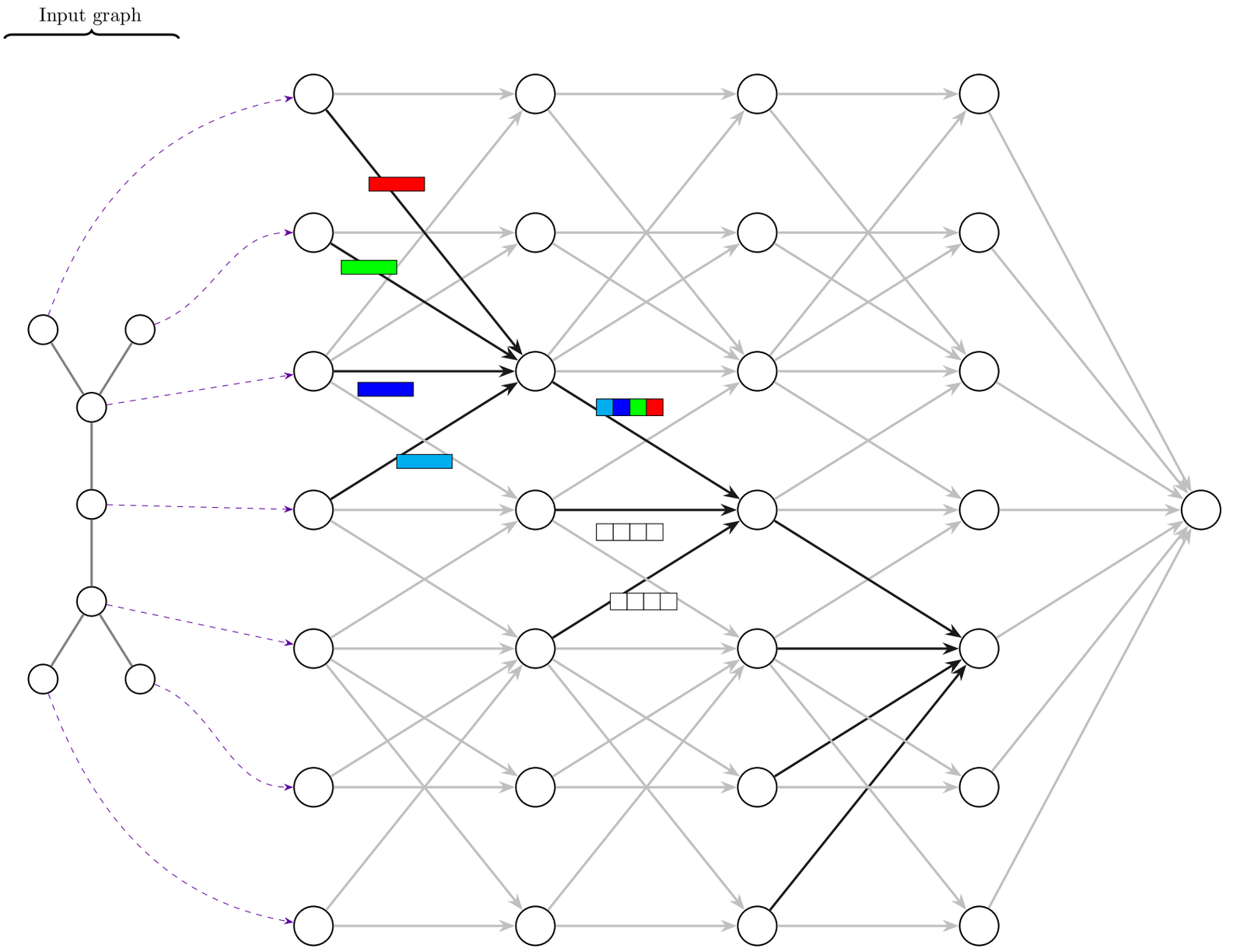
Second order vertex aggregation



Harvard Clean Energy Project results

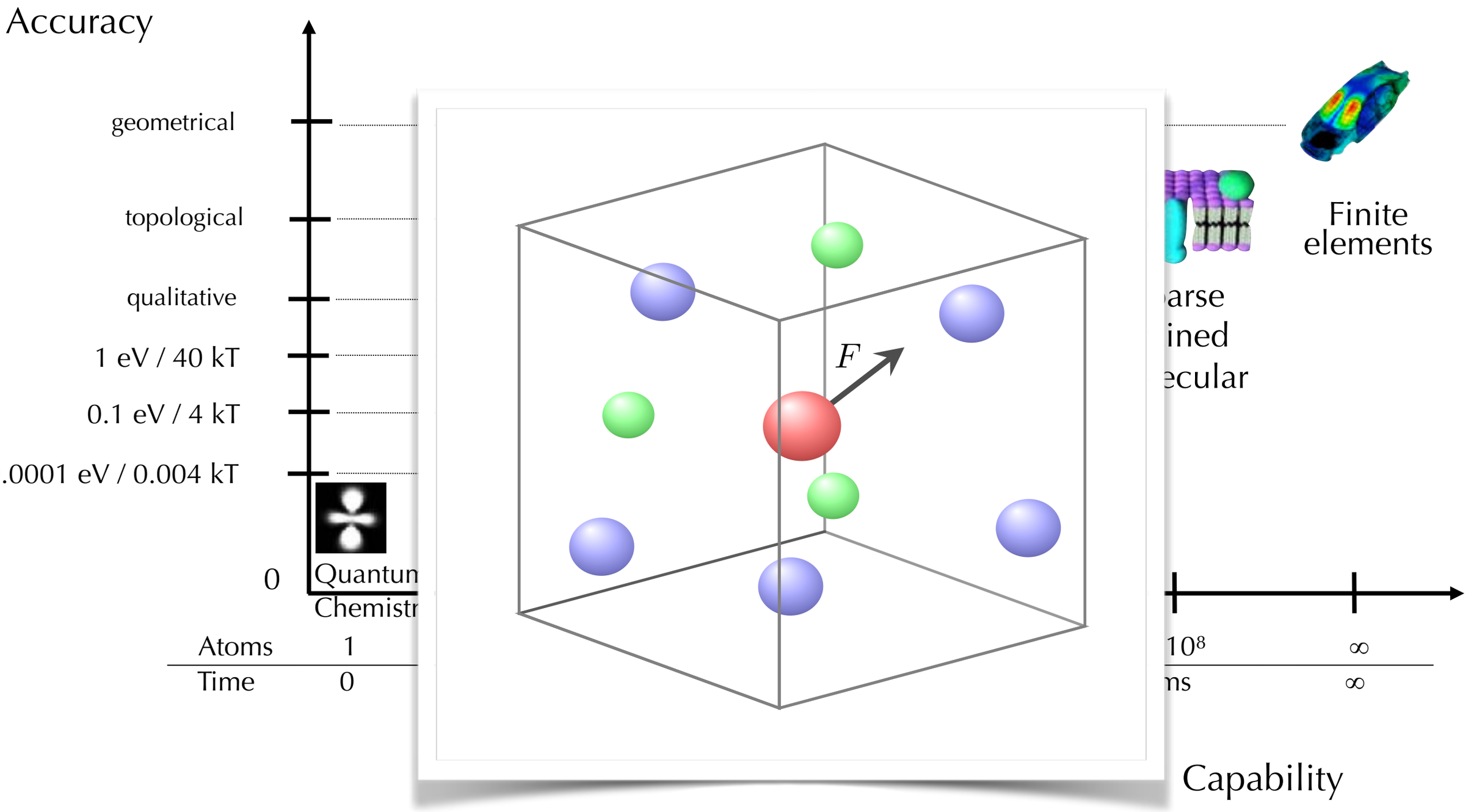
Method	Train MAE	Train RMSE	Test MAE	Test RMSE
Lasso	0.863	1.190	0.867	1.437
Ridge Regression	0.849	1.164	0.854	1.376
Random Forest	0.999	1.331	1.004	1.799
Gradient Boosted Tree	0.676	0.939	0.704	1.005
Weisfeiler-Lehman Graph Kernel	0.805	1.111	0.805	1.096
Neural Graph Fingerprint	0.848	1.187	0.851	1.177
Learning Convolution Neural Network	0.704	0.972	0.718	0.973
CCN 2D	0.562	0.773	0.570	0.773

[Duvenaud et al., 2015] [Kriege, 2016] [Niepert, 2016]
[Hachmann et al., 2011]


Input graph

4. SO(3) covariant neural networks

Multiple scales of materials modelling



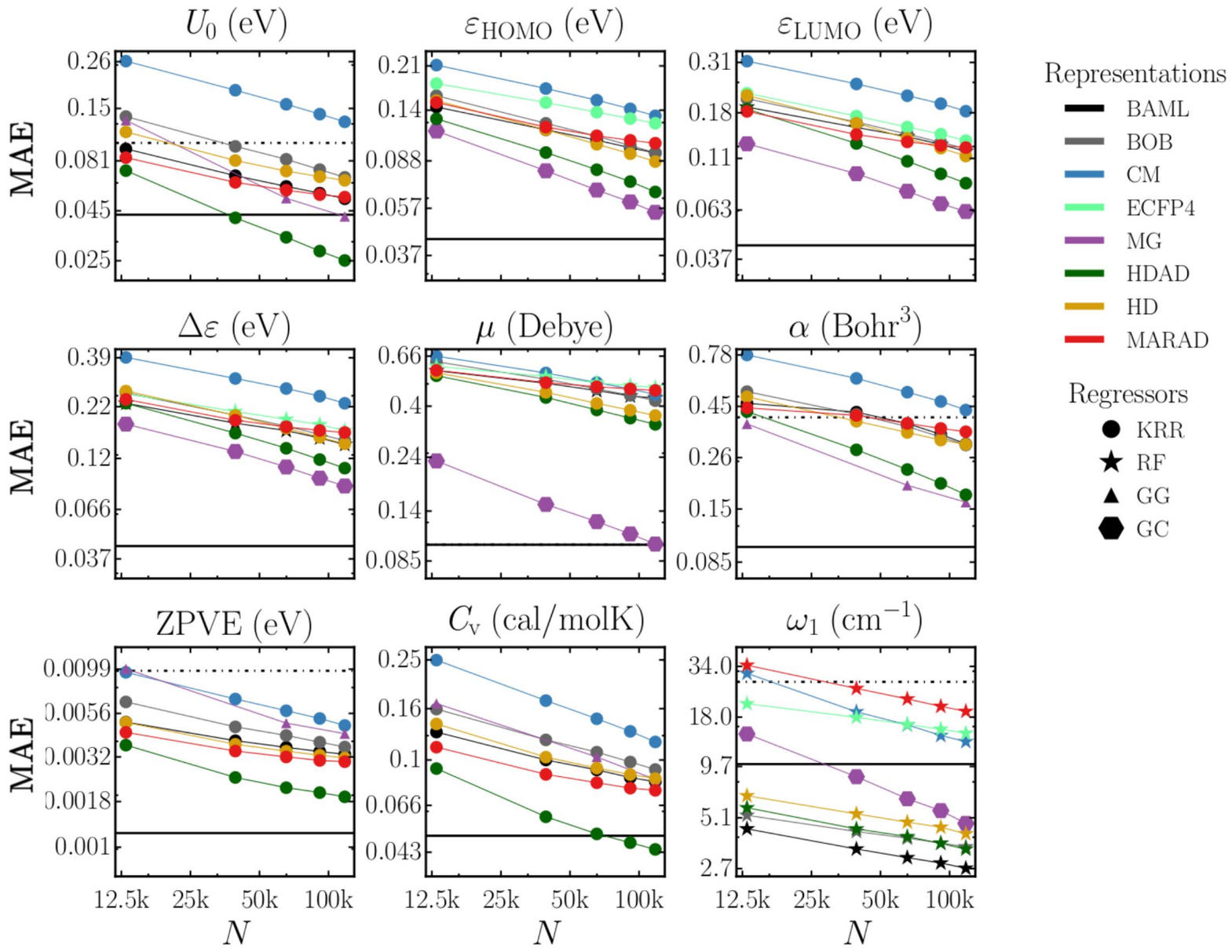
[Csányi]

The GDB-9 dataset

134k small organic molecules modeled with DFT

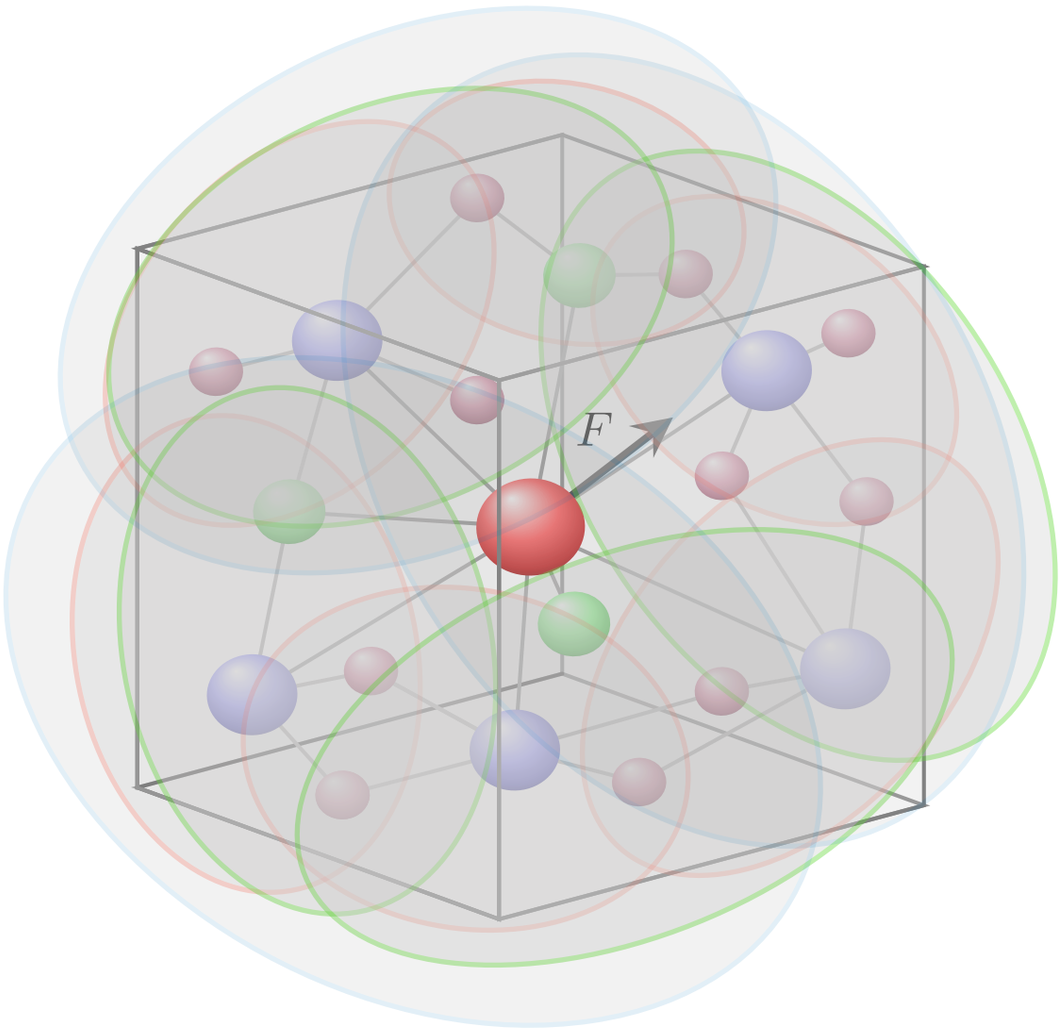
[B3LYP/6-31G(2df,p)]

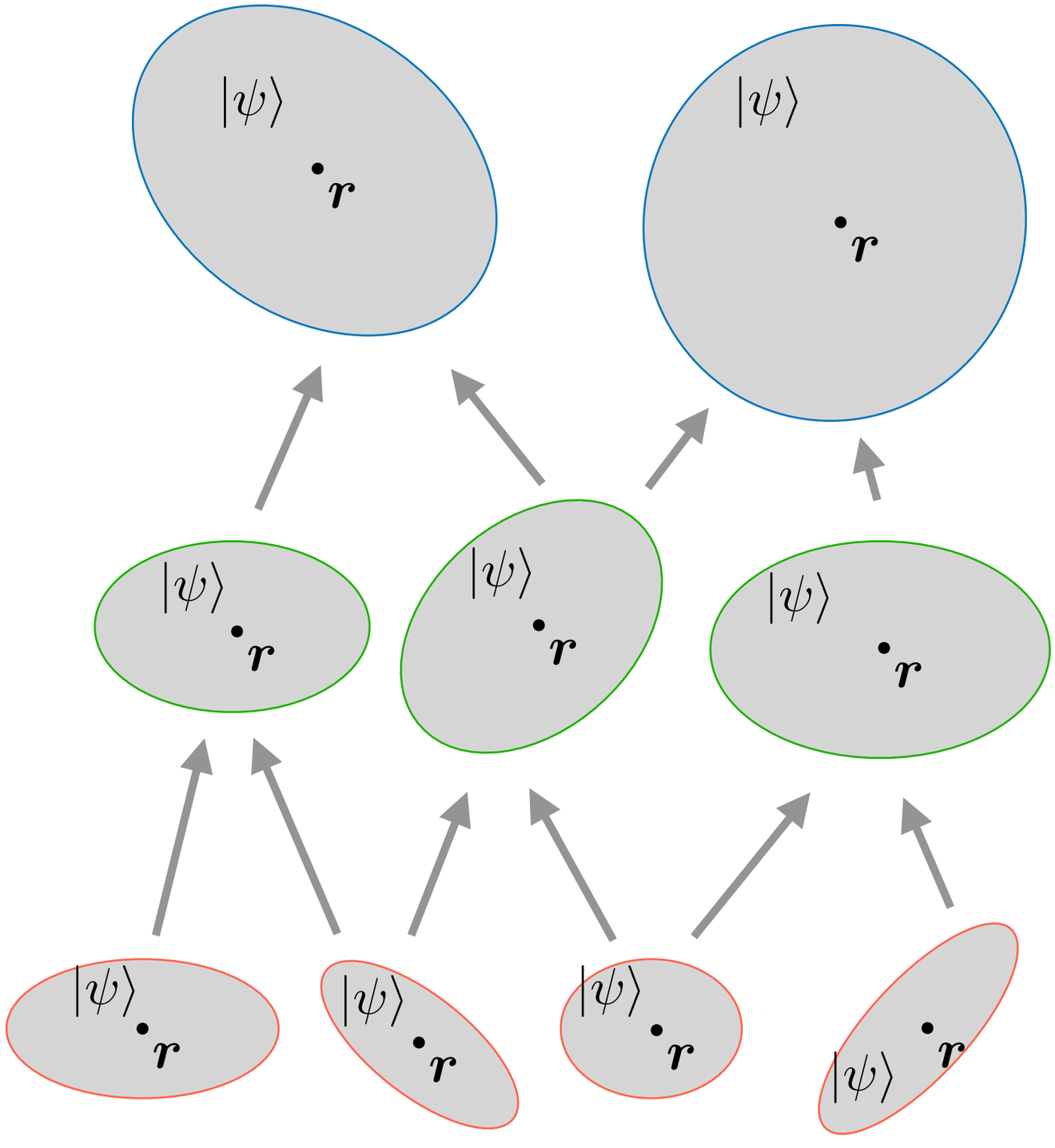
Targets: 15 physical/ chemical properties

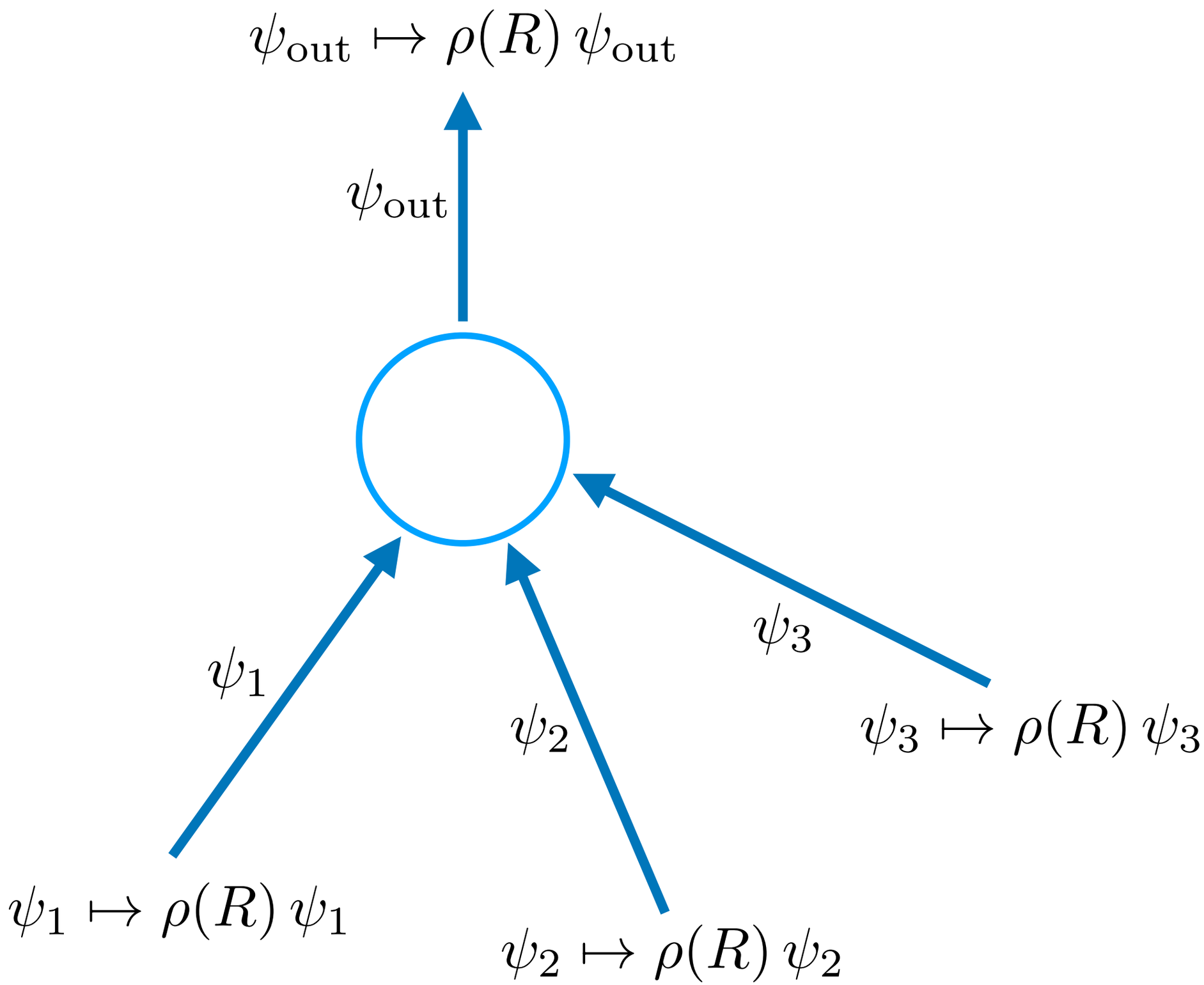


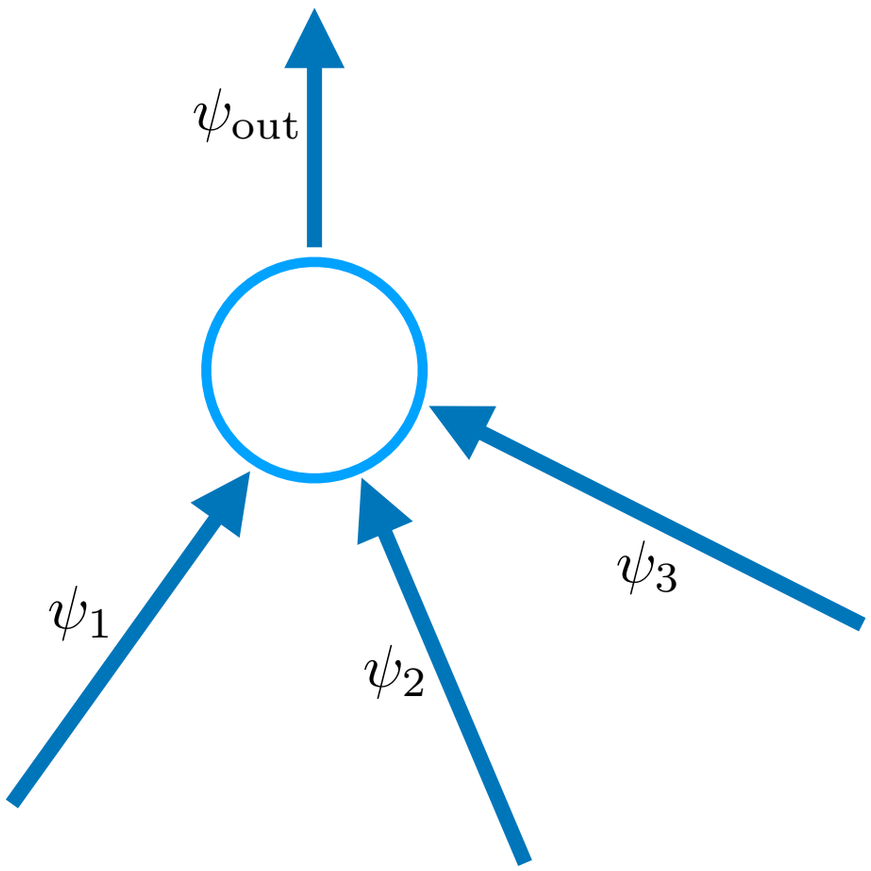
[Ramakrishnan et al., 2014] [von Lilienfeild et al.]

Compositional structure






$\psi_{\text{out}} \mapsto \rho(R)\,\psi_{\text{out}}$
ψ_{out}
ψ_1
$\psi_1 \mapsto \rho(R)\,\psi_1$
ψ_2
$\psi_2 \mapsto \rho(R)\,\psi_2$
ψ_3
$\psi_3 \mapsto \rho(R)\,\psi_3$



$$\psi_{\text{out}} = \sigma\left(\sum_i \psi_i * \chi\right)$$

$$\widehat{\psi}^{\ell'}_{\text{out}} = \sigma\left(\sum_i \widehat{\psi}^{\ell}_i \cdot \widehat{\chi}^{\ell}\right)$$

But what form should the nonlinearity take?

The Clebsch-Gordan product

The tensor product of two irreducible representations of a compact group G decomposes into irreducibles in the form

$$\rho_1(g) \otimes \rho_2(g) = \bigoplus_{\ell} \bigoplus_{m=1}^{\kappa(\ell)} C_m^\ell \cdot \rho_\ell(g) \cdot (C_m^\ell)^\dagger$$

Clebsch-Gordan gates

General form of aggregation:

$$|\psi\rangle = \sigma\Big(W \sum_i (\boldsymbol{r}_i - \boldsymbol{r})^{\otimes k} \otimes |\psi_i\rangle^{\otimes p} \Big)$$

The nonlinearity is a low order Clebsch-Gordan product

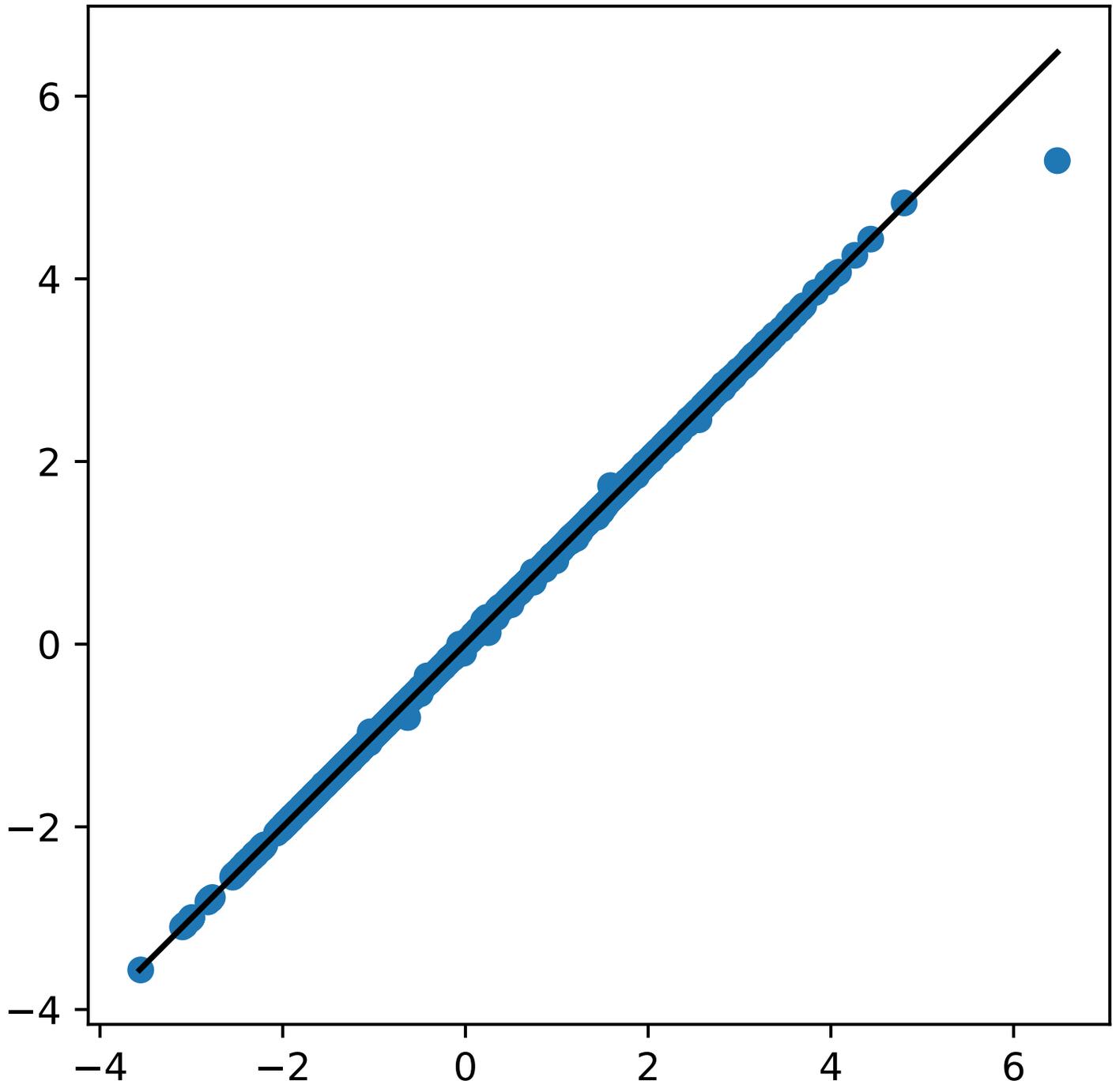
$$|\psi\rangle \mapsto \rho(g)\,|\psi\rangle$$

["Tensor Field Networks" by Thomas et al., 2018] ["Clebsch-Gordan networks", K., Trivedi & Zhen , 2018] ["3D steerable CNNs", Weiler et al., 2018]

QM9 results

	us	SchNett	google1	google2	googleML	L-Scat	T-Scat	NMP	CM
alpha (bohr3)	0.096	0.235	0.161	0.232	0.175	0.520	0.160	* 0.092	0.430
Cv (cal/molK)	* 0.031	0.033	0.084	0.097	0.044	0.100	0.049	0.040	0.120
gap (eV)	* 0.061	0.063	0.088	0.087	0.107	0.304	0.118	0.069	0.229
homo (eV)	* 0.036	0.041	0.057	0.055	0.066	0.177	0.085	0.043	0.133
lumo (eV)	* 0.032	0.034	0.063	0.062	0.084	0.234	0.076	0.038	0.183
mu (D)	0.046	0.033	0.247	0.101	0.334	0.630	0.340	* 0.030	0.450
omega1 (cm-1)	* 2.002	--	6.220	4.760	2.710	--	--	--	--
r2 (bohr2)	0.841	* 0.073	--	--	--	6.670	0.410	0.180	3.390
U0 (eV)	0.028	* 0.014	0.042	0.150	0.025	0.082	0.022	0.020	0.128
zpve (meV)	2.162	1.700	4.310	9.660	1.910	4.000	2.000	* 1.500	4.800

[Schütt et al., 2018] [Faber et al., 2017] [Eickenberg et al., 2018] [Zhang et al, 2018]

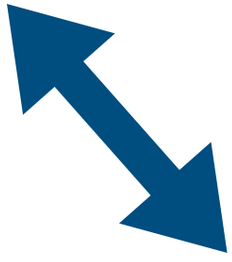

6
4
2
0
−2
−4
−4
−2
0
2
4
6

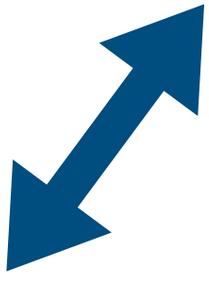
MD-17 results (kcal/mol)

	NBody50k	Deep95k	DTNN50k	SchE50k	SchEF50k	GDML1k	SchE1k	SchEF1k
Aspirin	* 0.103	0.201	--	0.250	0.120	0.270	4.200	0.370
Benzene	* 0.035	0.065	0.040	0.080	0.070	0.070	1.190	0.080
Ethanol	* 0.029	0.055	--	0.070	0.050	0.150	0.930	0.080
Malonaldehyde	* 0.056	0.092	0.190	0.130	0.080	0.160	2.030	0.130
Naphthalene	* 0.043	0.095	--	0.200	0.110	0.120	3.580	0.160
Salicylic_acid	* 0.072	0.106	0.410	0.250	0.100	0.120	3.270	0.200
Toluene	* 0.042	0.085	0.180	0.160	0.090	0.120	2.950	0.120
Uracil	* 0.045	0.085	--	0.130	0.100	0.110	2.260	0.140

Theory

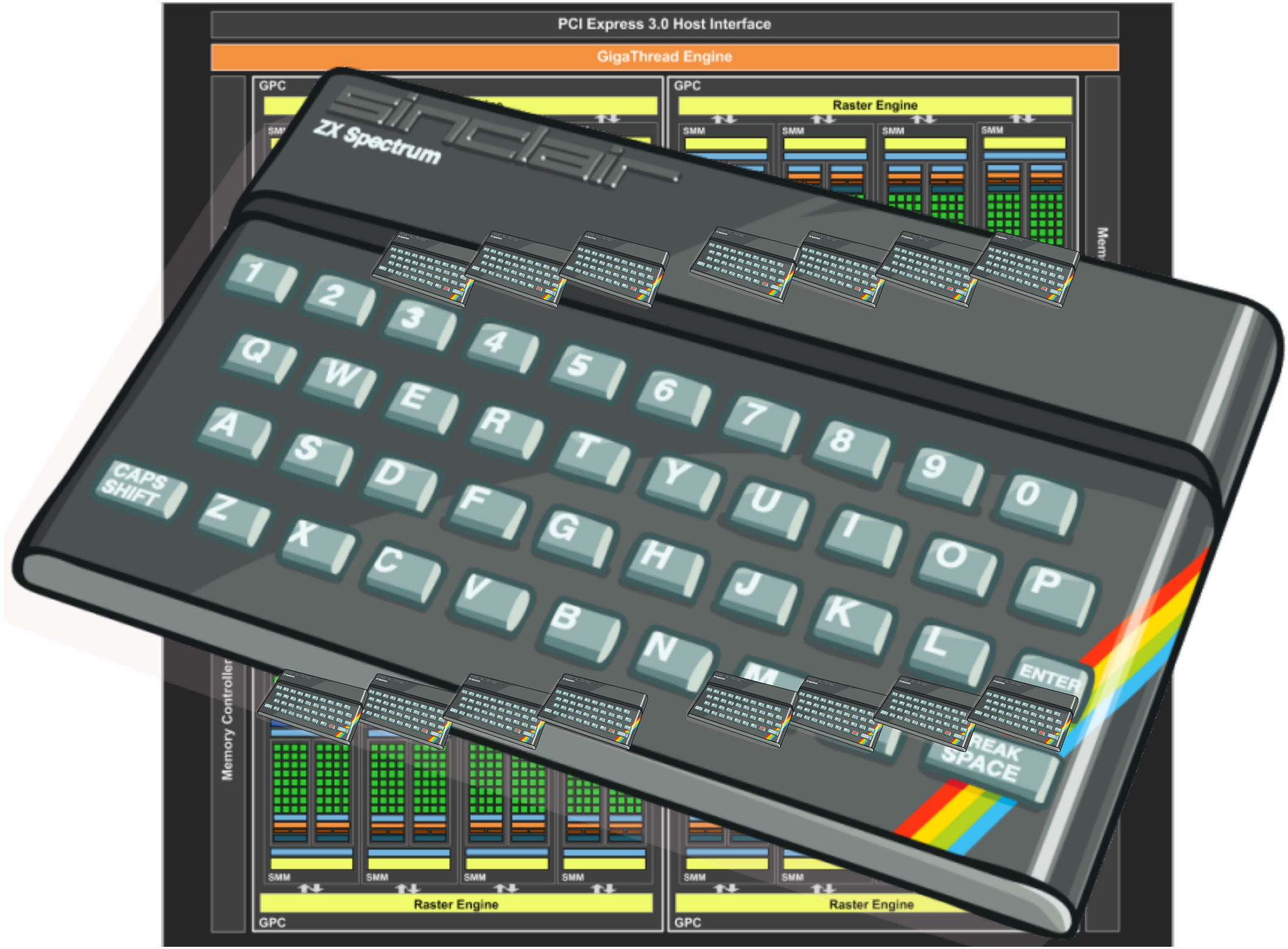
Experiment



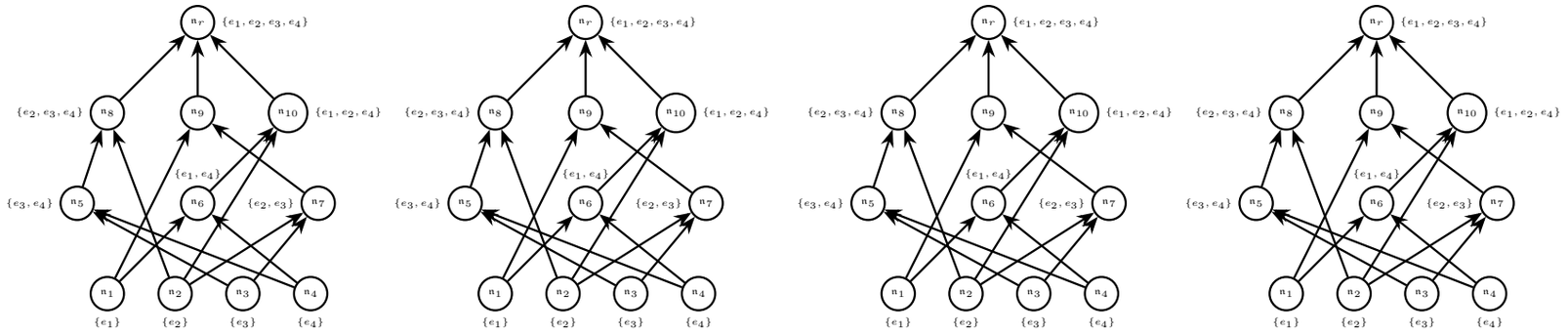


Implementation


PCI Express 3.0 Host Interface
GigaThread Engine
Memory Controller
GPC
Raster Engine
SMM
L2 Cache
Memory Controller
GPC
Raster Engine
SMM
Memory Controller
Memory Controller
GPC
Raster Engine
SMM
GPC
Raster Engine
SMM


PCI Express 3.0 Host Interface
GigaThread Engine
GPC
Raster Engine
SMM
Memory Controller
ZX Spectrum
CAPS SHIFT
ENTER
SPACE

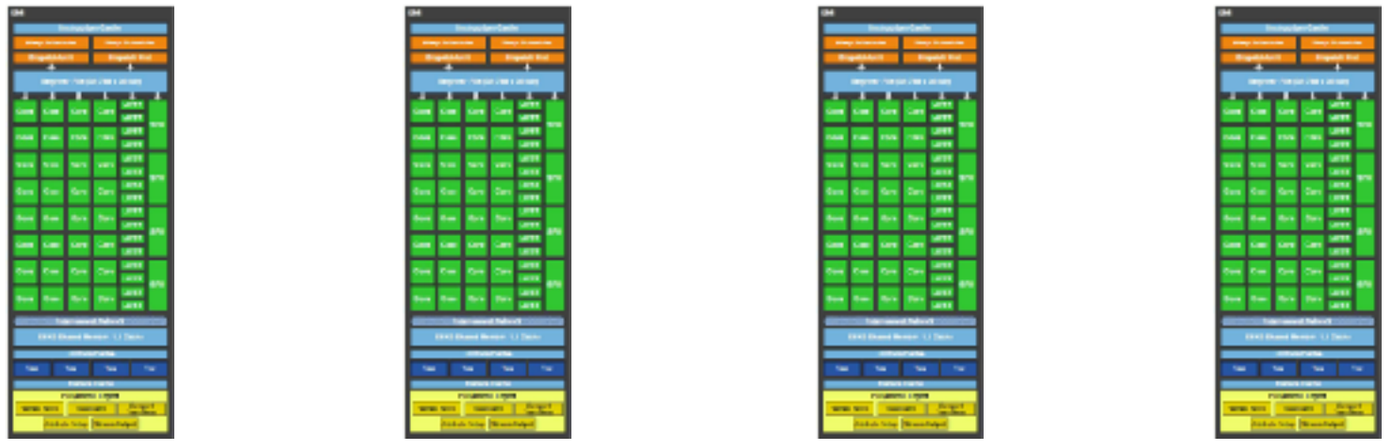
Computation graph



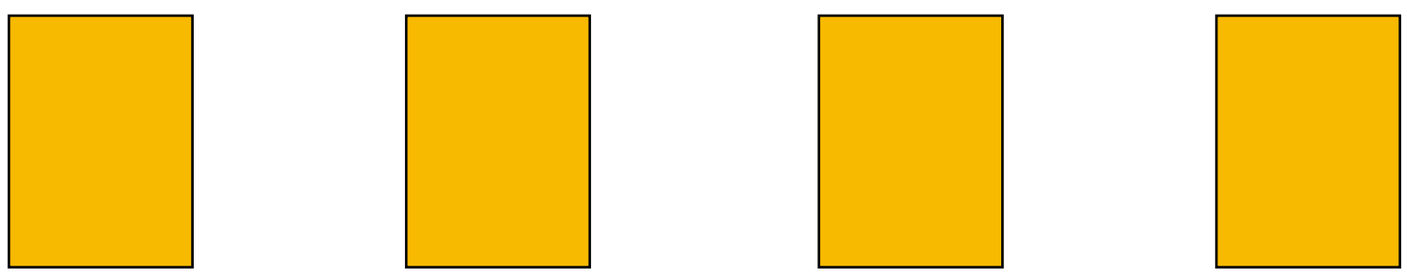
Bytecode

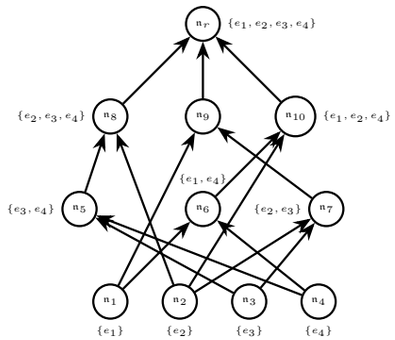
```
TPprogram  CGproduct(){
  TPpart0 (l=0)[0m (n=1){
    input(0,0);
  }
  TPpart1 (l=1) (n=1){
    input(0,1);
  }
  TPpart2 (l=0) (n=1){
    input(1,0);
  }
  TPpart3 (l=1) (n=1){
    input(1,1);
  }
  TPpart4 (l=0) (n=1){
    input(2,0);
  }
  TPpart5 (l=1) (n=1){
    input(2,1);
  }
  TPpart6 (l=0) (n=2){
    CG(2,4)[0];
    CG(3,5)[1];
  }
  TPpart7 (l=1) (n=3){
    CG(2,5)[0];
    CG(3,4)[1];
    CG(3,5)[2];
```

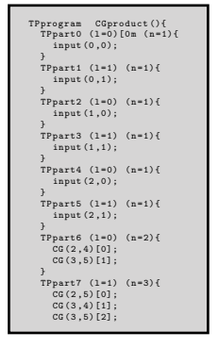
Same bytecode interpreter on each streaming mulitprocessor

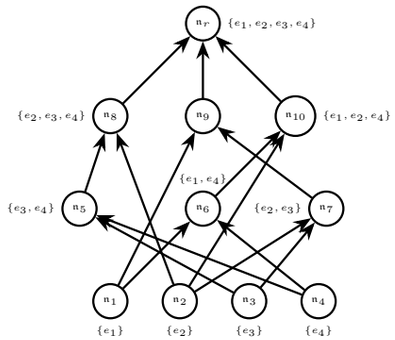


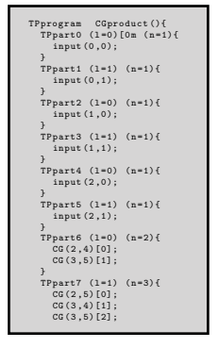
Result

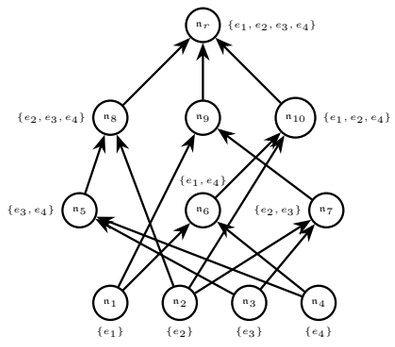


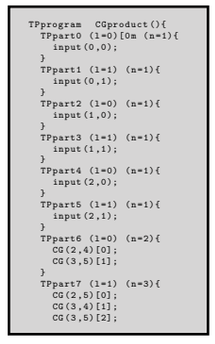


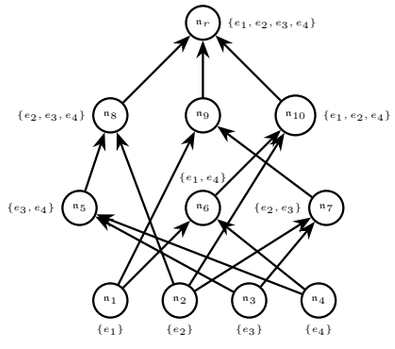


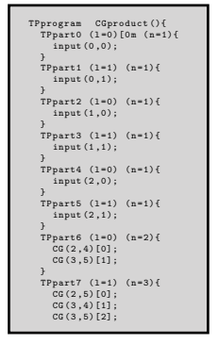












```
VertexFn inputs;

VertexFn L1 =
aggregate1D(perc(inputs,W1));
EdgeFn   L2 =
scatter(L1)*scatter(L1);
EdgeFn   L3 =
perc<ReLU>(scatter(L1),W2);
EdgeFn   L4 =
perc<ReLU>(concat(L2,L3));
VertexFn L5 = gather(L4);
VertexFn L6 =
perc<sigmoid>(L5,W4);

GlobalFn L8 =
average(aggregate0D(L6));
GlobalFn L9 =
perc<sigmoid>(L8,W5);
GlobalFn
output=squared_error_loss(L9,
target);
```

GNN grammar

```
CCNlayer0D 3
CCNlayer1D 3
ReLUlayer 2
CCNlayer0D 4
ReLUlayer 3
AveragingLayer 3
LinearFClayer 1
SquaredLossLayer
1
```

GNNlayerDescr

```
TPprogram  CGproduct(){
  TPpart0 (l=0)[0m (n=1){
    input(0,0);
  }
  TPpart1 (l=1) (n=1){
    input(0,1);
  }
  TPpart2 (l=0) (n=1){
    input(1,0);
  }
  TPpart3 (l=1) (n=1){
    input(1,1);
  }
  TPpart4 (l=0) (n=1){
    input(2,0);
  }
  TPpart5 (l=1) (n=1){
    input(2,1);
  }
  TPpart6 (l=0) (n=2){
    CG(2,4)[0];
    CG(3,5)[1];
  }
  TPpart7 (l=1) (n=3){
    CG(2,5)[0];
    CG(3,4)[1];
    CG(3,5)[2];
```

```
TPprogram  CGproduct(){
  TPpart0 (l=0)[0m (n=1){
    input(0,0);
  }
  TPpart1 (l=1) (n=1){
    input(0,1);
  }
  TPpart2 (l=0) (n=1){
    input(1,0);
  }
  TPpart3 (l=1) (n=1){
    input(1,1);
  }
  TPpart4 (l=0) (n=1){
    input(2,0);
  }
  TPpart5 (l=1) (n=1){
    input(2,1);
  }
  TPpart6 (l=0) (n=2){
    CG(2,4)[0];
    CG(3,5)[1];
  }
  TPpart7 (l=1) (n=3){
    CG(2,5)[0];
    CG(3,4)[1];
    CG(3,5)[2];
```

```
TPprogram  CGproduct(){
  TPpart0 (l=0)[0m (n=1){
    input(0,0);
  }
  TPpart1 (l=1) (n=1){
    input(0,1);
  }
  TPpart2 (l=0) (n=1){
    input(1,0);
  }
  TPpart3 (l=1) (n=1){
    input(1,1);
  }
  TPpart4 (l=0) (n=1){
    input(2,0);
  }
  TPpart5 (l=1) (n=1){
    input(2,1);
  }
  TPpart6 (l=0) (n=2){
    CG(2,4)[0];
    CG(3,5)[1];
  }
  TPpart7 (l=1) (n=3){
    CG(2,5)[0];
    CG(3,4)[1];
    CG(3,5)[2];
```

```
TPprogram  CGproduct(){
  TPpart0 (l=0)[0m (n=1){
    input(0,0);
  }
  TPpart1 (l=1) (n=1){
    input(0,1);
  }
  TPpart2 (l=0) (n=1){
    input(1,0);
  }
  TPpart3 (l=1) (n=1){
    input(1,1);
  }
  TPpart4 (l=0) (n=1){
    input(2,0);
  }
  TPpart5 (l=1) (n=1){
    input(2,1);
  }
  TPpart6 (l=0) (n=2){
    CG(2,4)[0];
    CG(3,5)[1];
  }
  TPpart7 (l=1) (n=3){
    CG(2,5)[0];
    CG(3,4)[1];
    CG(3,5)[2];
```

The CGNN GUI

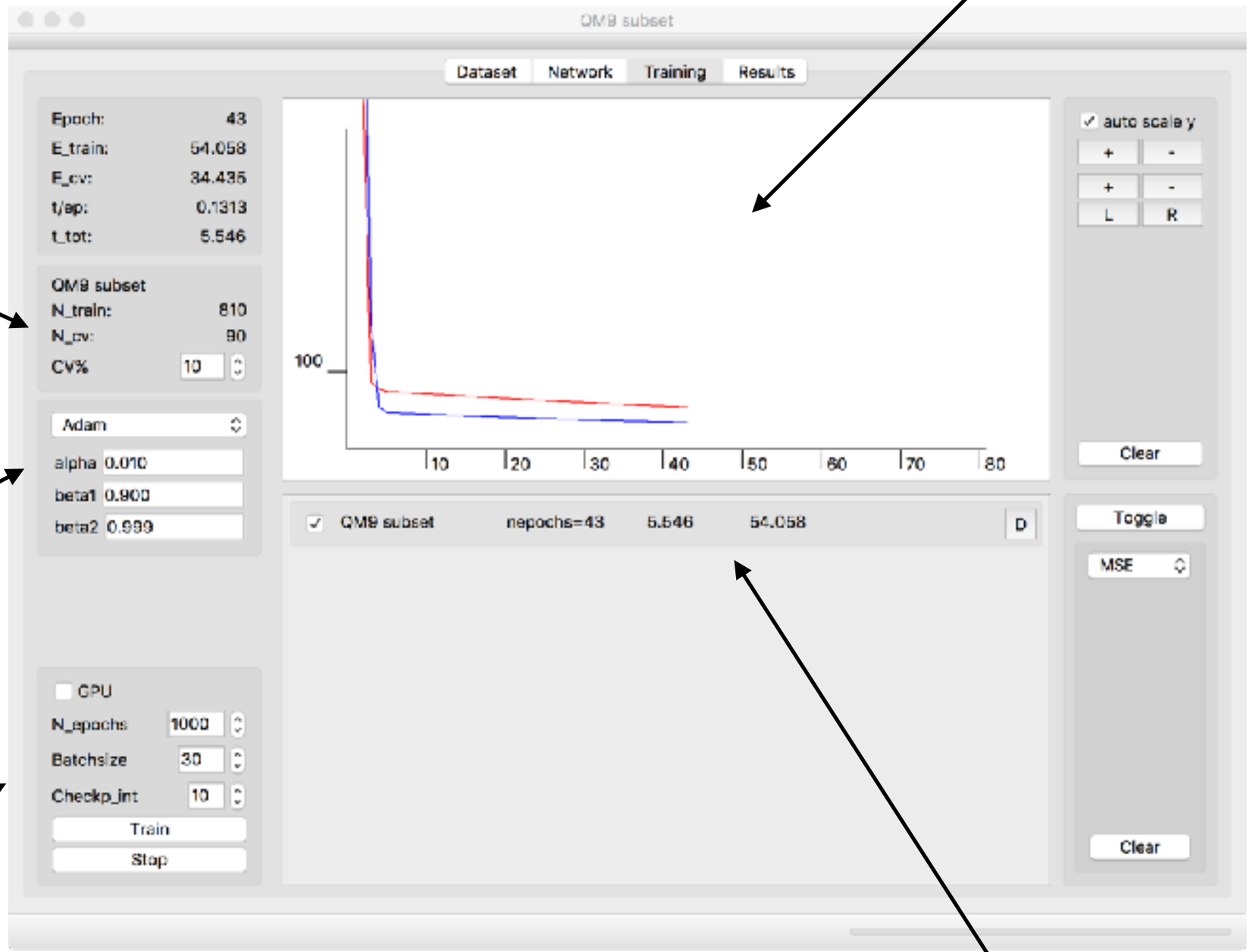
Training and CV error

Dataset

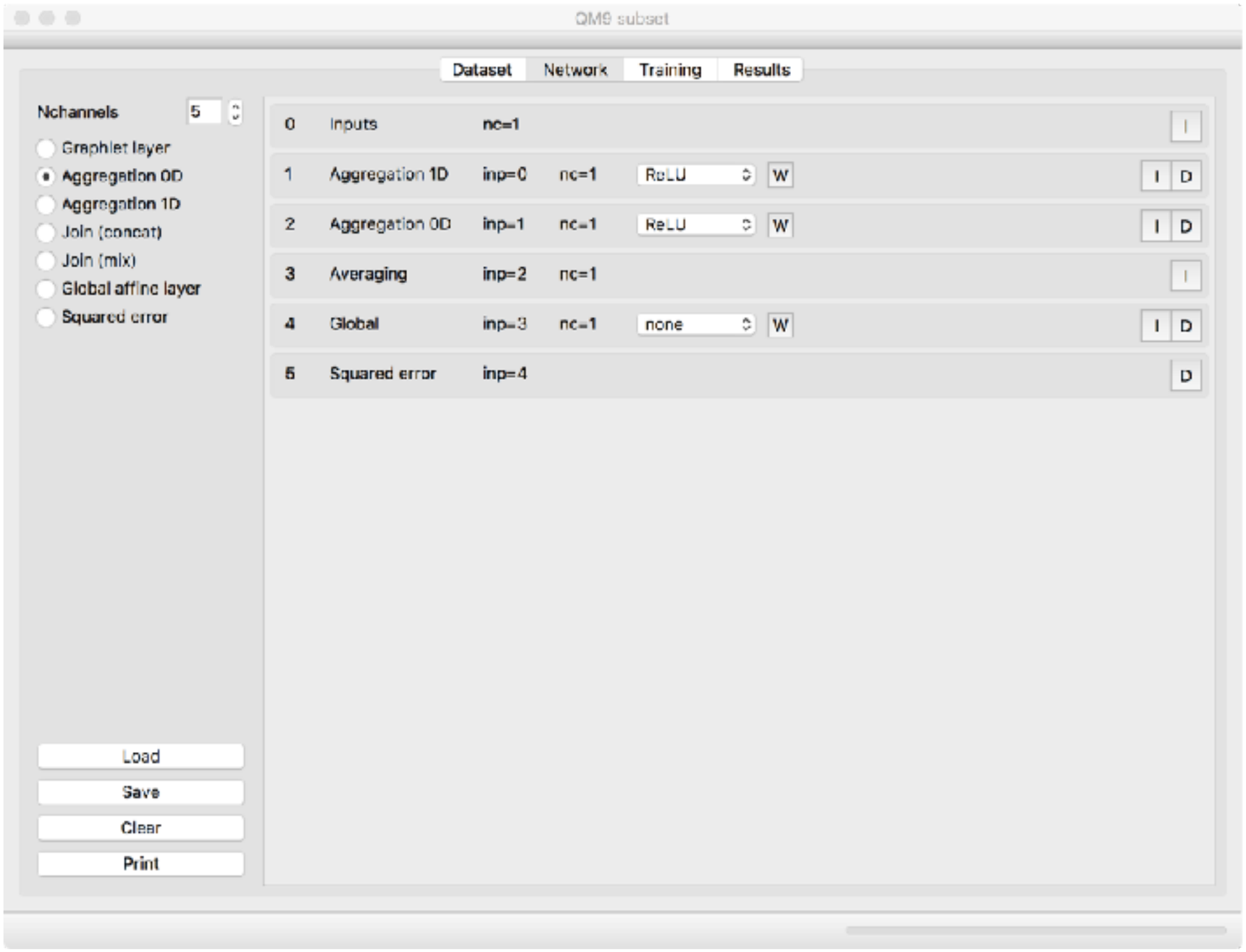
Optimizer

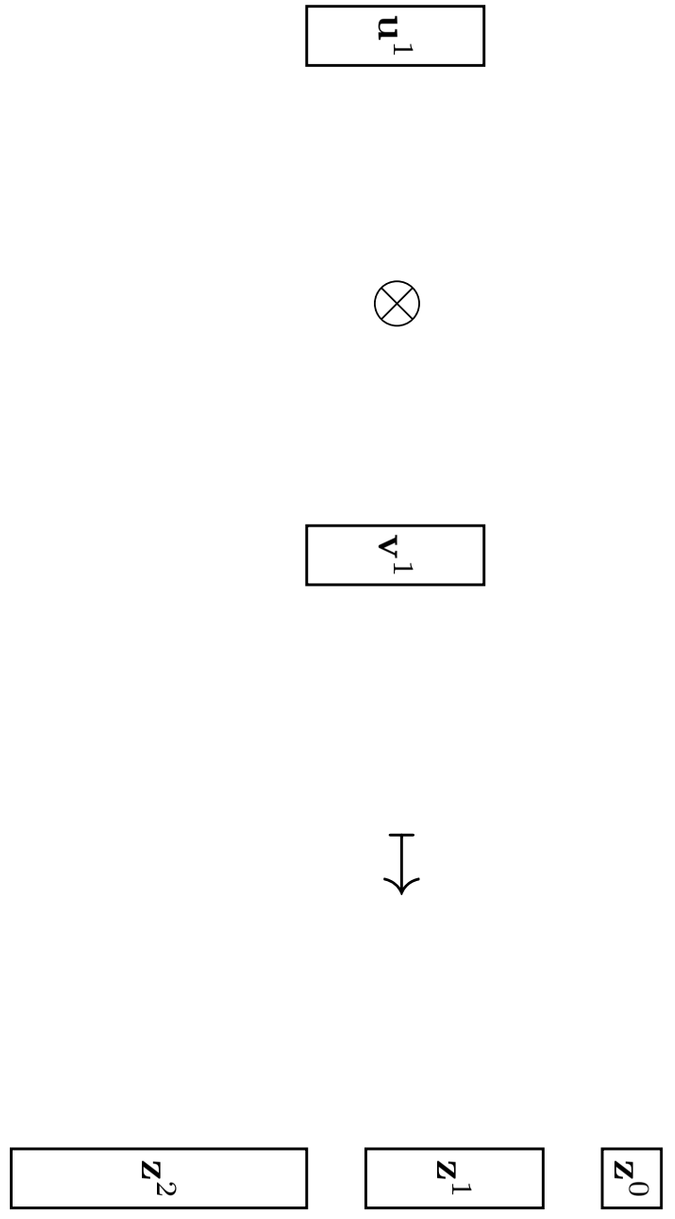
Training parameters

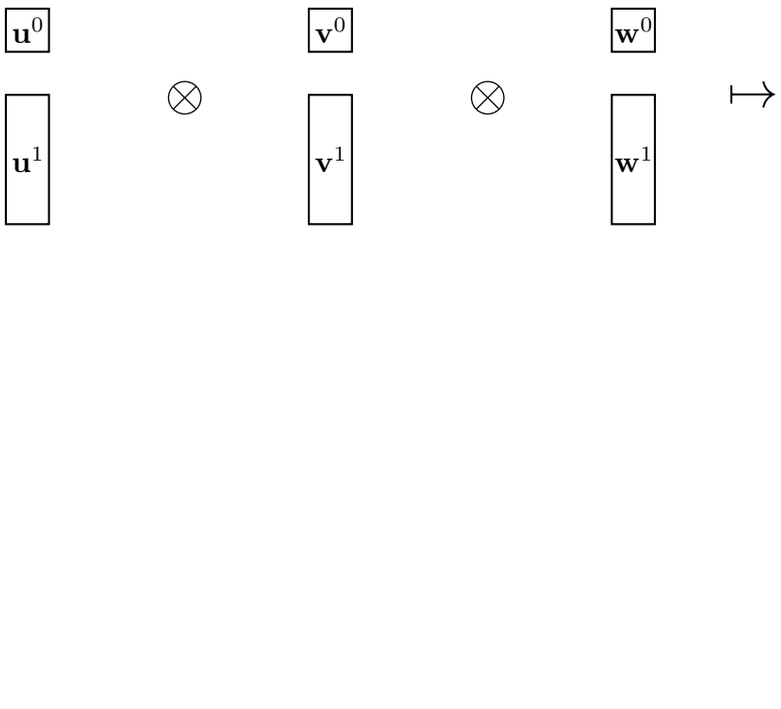
Run info strips

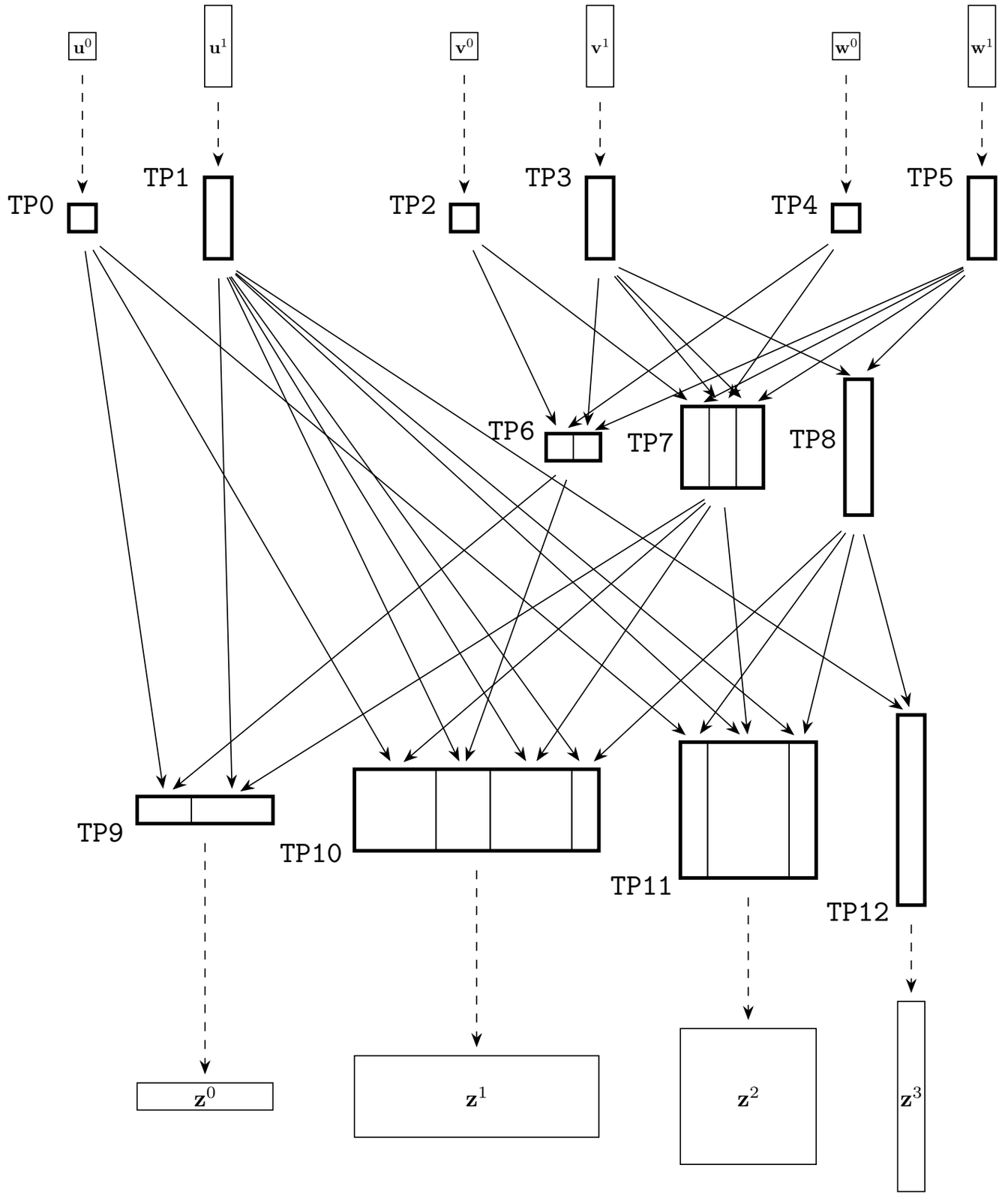


The CGNN GUI

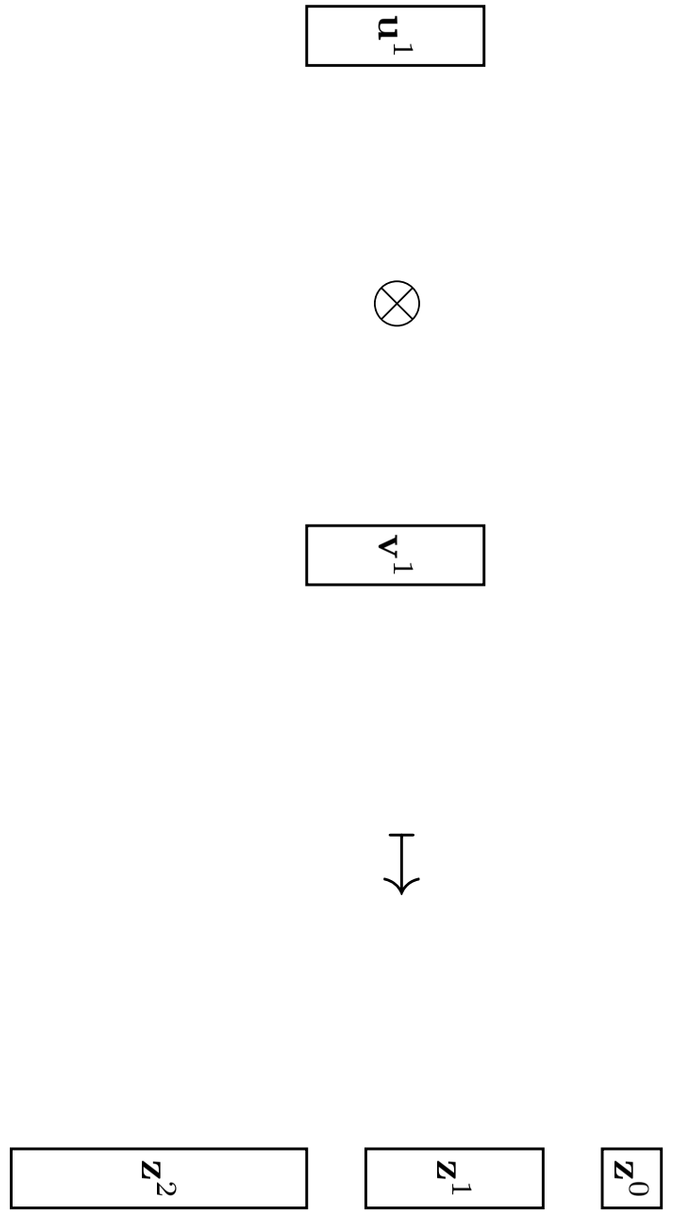

QM9 subset
Dataset
Network
Training
Results
Nchannels
5
Graphlet layer
Aggregation 0D
Aggregation 1D
Join (concat)
Join (mix)
Global affine layer
Squared error
0 Inputs nc=1 I
1 Aggregation 1D inp=0 nc=1 ReLU W I D
2 Aggregation 0D inp=1 nc=1 ReLU W I D
3 Averaging inp=2 nc=1 I
4 Global inp=3 nc=1 none W I D
5 Squared error inp=4 D
Load
Save
Clear
Print

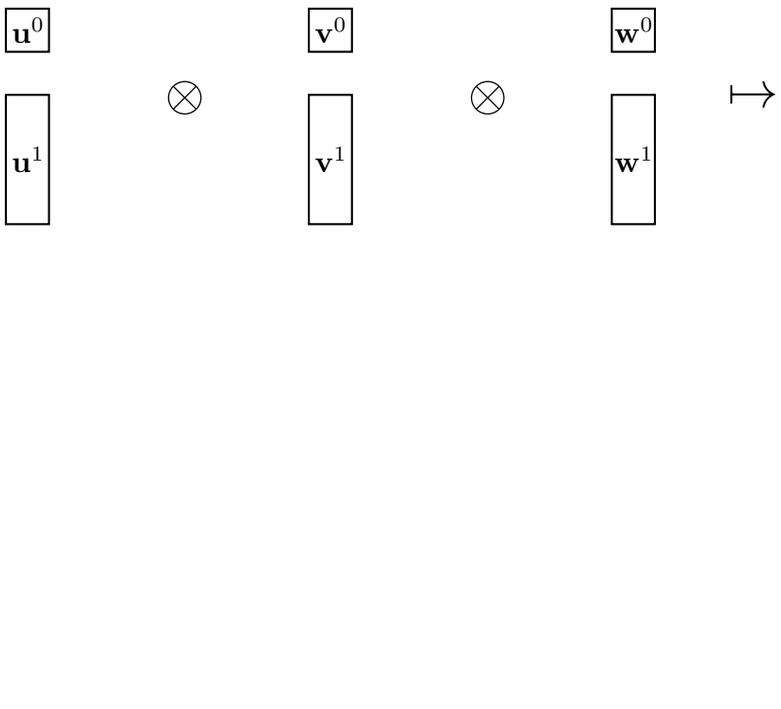

$\mathbf{u}^1$
$\otimes$
$\mathbf{v}^1$
$\mathbf{z}_2$
$\mathbf{z}_1$
$\mathbf{z}_0$

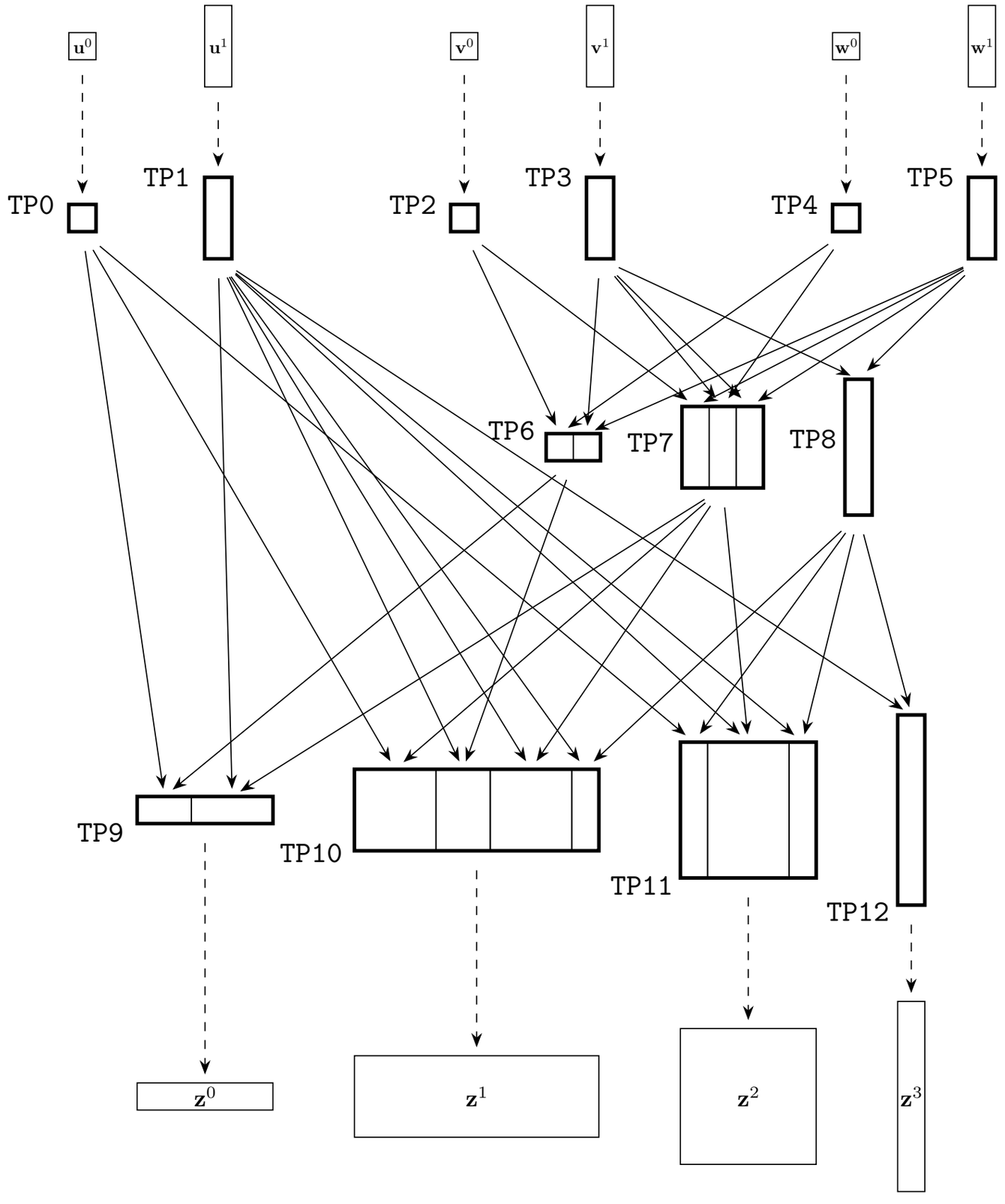

$\mathbf{u}^0$
$\mathbf{u}^1$
$\otimes$
$\mathbf{v}^0$
$\mathbf{v}^1$
$\otimes$
$\mathbf{w}^0$
$\mathbf{w}^1$
$\mapsto$
$\mathbf{u}^0 \otimes \mathbf{v}^0 \otimes \mathbf{w}^0$
$\mathbf{u}^0 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^0 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^0$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$
$\mathbf{u}^0 \otimes \mathbf{v}^0 \otimes \mathbf{w}^1$
$\mathbf{u}^0 \otimes \mathbf{v}^1 \otimes \mathbf{w}^0$
$\mathbf{u}^1 \otimes \mathbf{v}^0 \otimes \mathbf{w}^0$
$\mathbf{u}^0 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^0 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^0$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$

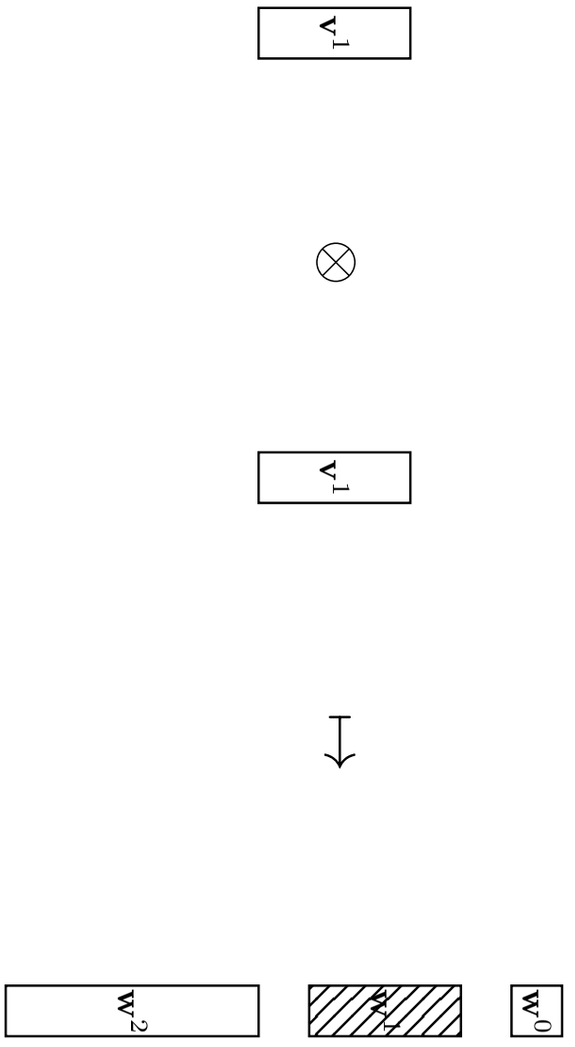

$\mathbf{u}^0$
$\mathbf{u}^1$
$\mathbf{v}^0$
$\mathbf{v}^1$
$\mathbf{w}^0$
$\mathbf{w}^1$
TP0
TP1
TP2
TP3
TP4
TP5
TP6
TP7
TP8
TP9
TP10
TP11
TP12
$\mathbf{z}^0$
$\mathbf{z}^1$
$\mathbf{z}^2$
$\mathbf{z}^3$

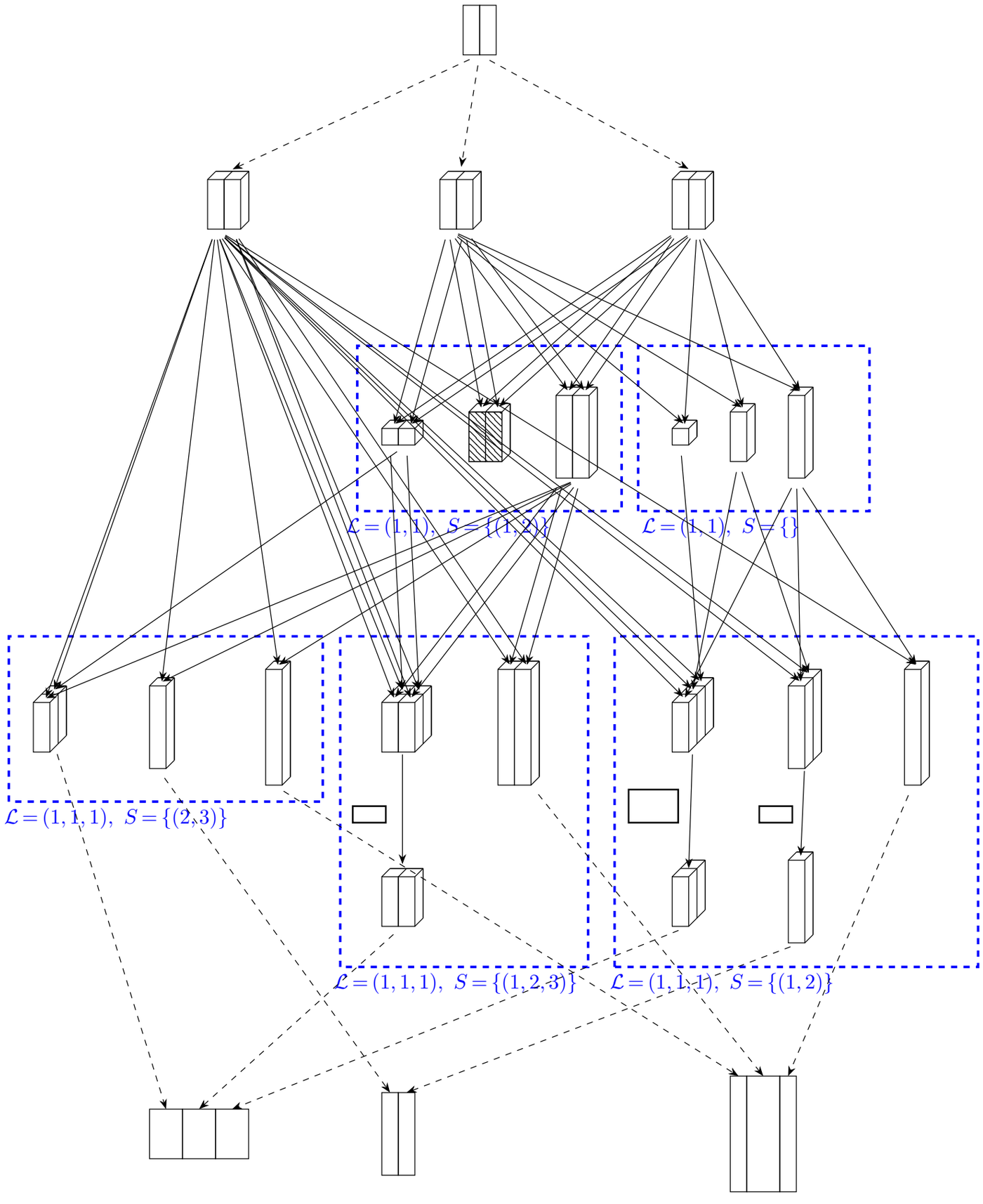
```
1   TPprogram  CGproduct(){
2     TPpart0 (l=0)[0m (n=1){
3       input(0,0);
4     }
5     TPpart1 (l=1) (n=1){
6       input(0,1);
7     }
8     TPpart2 (l=0) (n=1){
9       input(1,0);
10    }
11    TPpart3 (l=1) (n=1){
12      input(1,1);
13    }
14    TPpart4 (l=0) (n=1){
15      input(2,0);
16    }
17    TPpart5 (l=1) (n=1){
18      input(2,1);
19    }
20    TPpart6 (l=0) (n=2){
21      CG(2,4)[0];
22      CG(3,5)[1];
23    }
24    TPpart7 (l=1) (n=3){
25      CG(2,5)[0];
26      CG(3,4)[1];
27      CG(3,5)[2];
28    }
29    TPpart8 (l=2) (n=1){
30      CG(3,5)[0];
31    }
32    TPpart9 (l=0) (n=5){
33      output(0);
34      CG(0,6)[0];
35      CG(1,7)[2];
36    }
37    TPpart10 (l=1) (n=9){
38      output(1);
39      CG(0,7)[0];
40      CG(1,6)[3];
41      CG(1,7)[5];
```

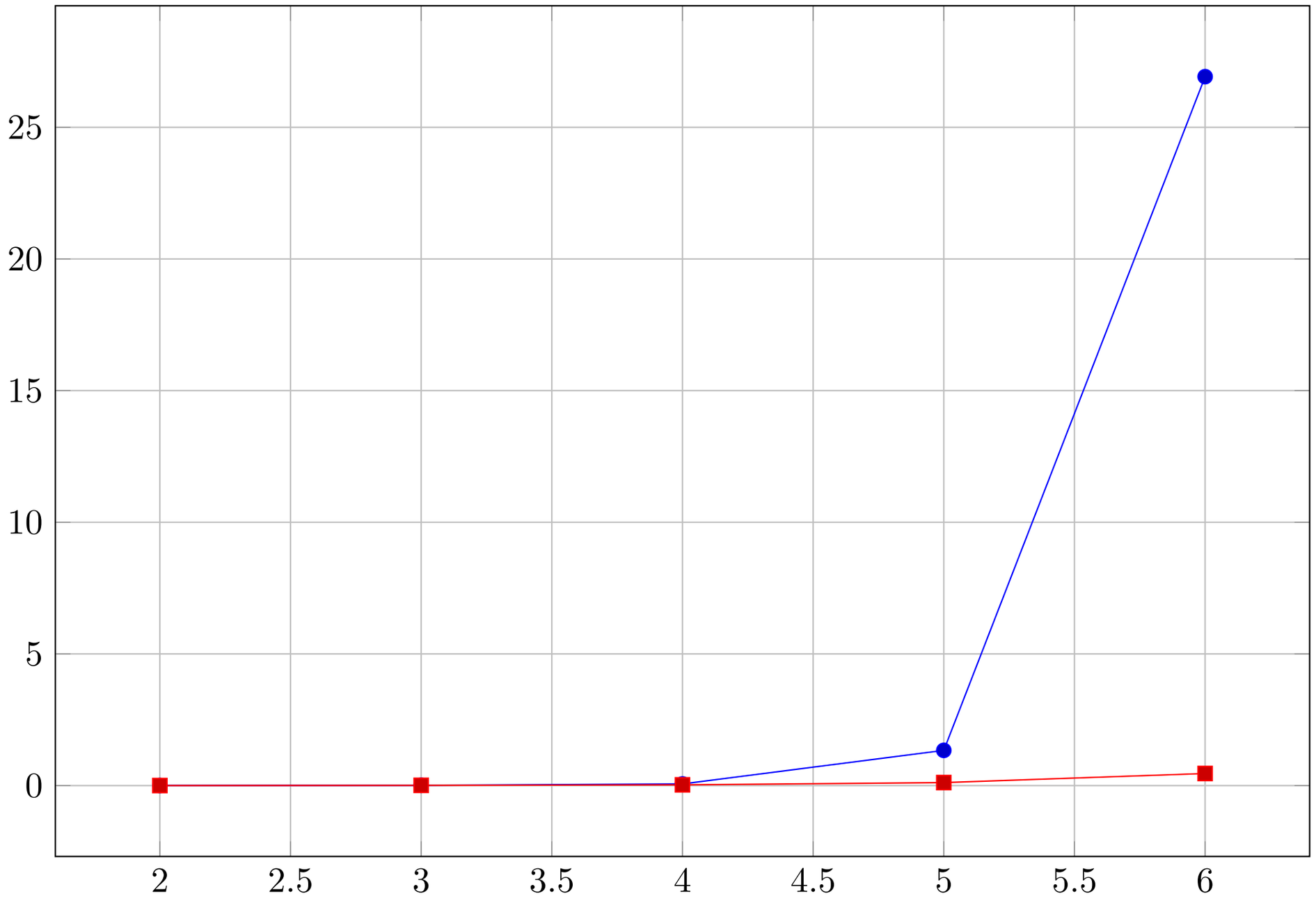

$\mathbf{u}^1$
$\otimes$
$\mathbf{v}^1$
$\mathbf{z}_2$
$\mathbf{z}_1$
$\mathbf{z}_0$


$\mathbf{u}^0$
$\mathbf{u}^1$
$\otimes$
$\mathbf{v}^0$
$\mathbf{v}^1$
$\otimes$
$\mathbf{w}^0$
$\mathbf{w}^1$
$\mapsto$
$\mathbf{u}^0 \otimes \mathbf{v}^0 \otimes \mathbf{w}^0$
$\mathbf{u}^0 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^0 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^0$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$
$\mathbf{u}^0 \otimes \mathbf{v}^0 \otimes \mathbf{w}^1$
$\mathbf{u}^0 \otimes \mathbf{v}^1 \otimes \mathbf{w}^0$
$\mathbf{u}^1 \otimes \mathbf{v}^0 \otimes \mathbf{w}^0$
$\mathbf{u}^0 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^0 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^0$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$
$\mathbf{u}^1 \otimes \mathbf{v}^1 \otimes \mathbf{w}^1$


$\mathbf{u}^0$
$\mathbf{u}^1$
$\mathbf{v}^0$
$\mathbf{v}^1$
$\mathbf{w}^0$
$\mathbf{w}^1$
TP0
TP1
TP2
TP3
TP4
TP5
TP6
TP7
TP8
TP9
TP10
TP11
TP12
$\mathbf{z}^0$
$\mathbf{z}^1$
$\mathbf{z}^2$
$\mathbf{z}^3$


v_1
v_1
w_2
w_1
w_0


$\mathcal{L} = (1,1),\ S = \{(1,2)\}$
$\mathcal{L} = (1,1),\ S = \{\}$
$\mathcal{L} = (1,1,1),\ S = \{(2,3)\}$
$\mathcal{L} = (1,1,1),\ S = \{(1,2,3)\}$
$\mathcal{L} = (1,1,1),\ S = \{(1,2)\}$


25
20
15
10
5
0
2
2.5
3
3.5
4
4.5
5
5.5
6

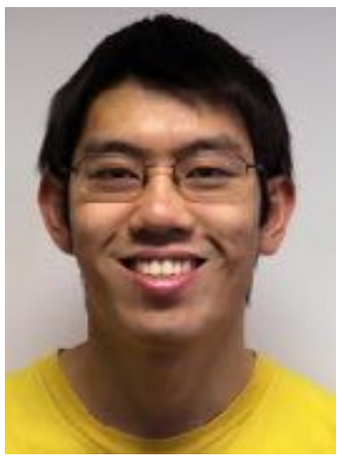
1. Compositional structure

2. Covariance

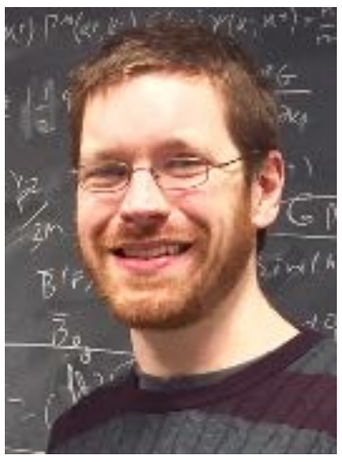
   → Fourier space activations

Shubhendu Trivedi

Hy Truong Son



Horace Pan



Brandon Anderson