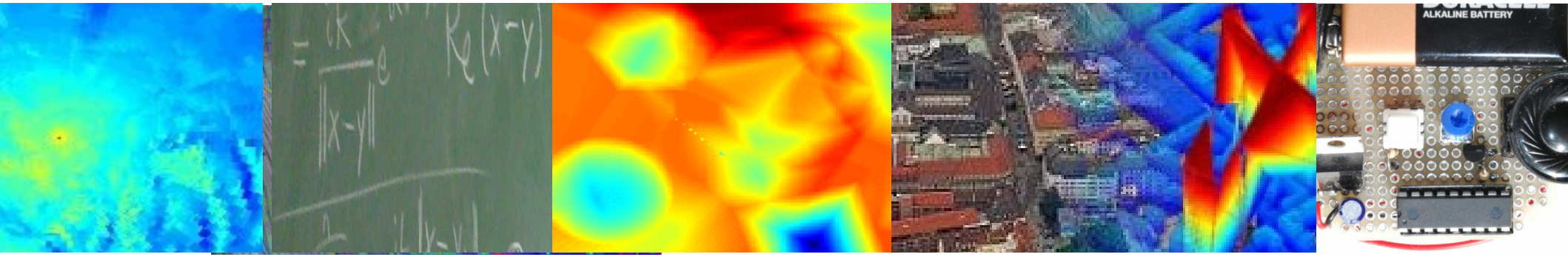
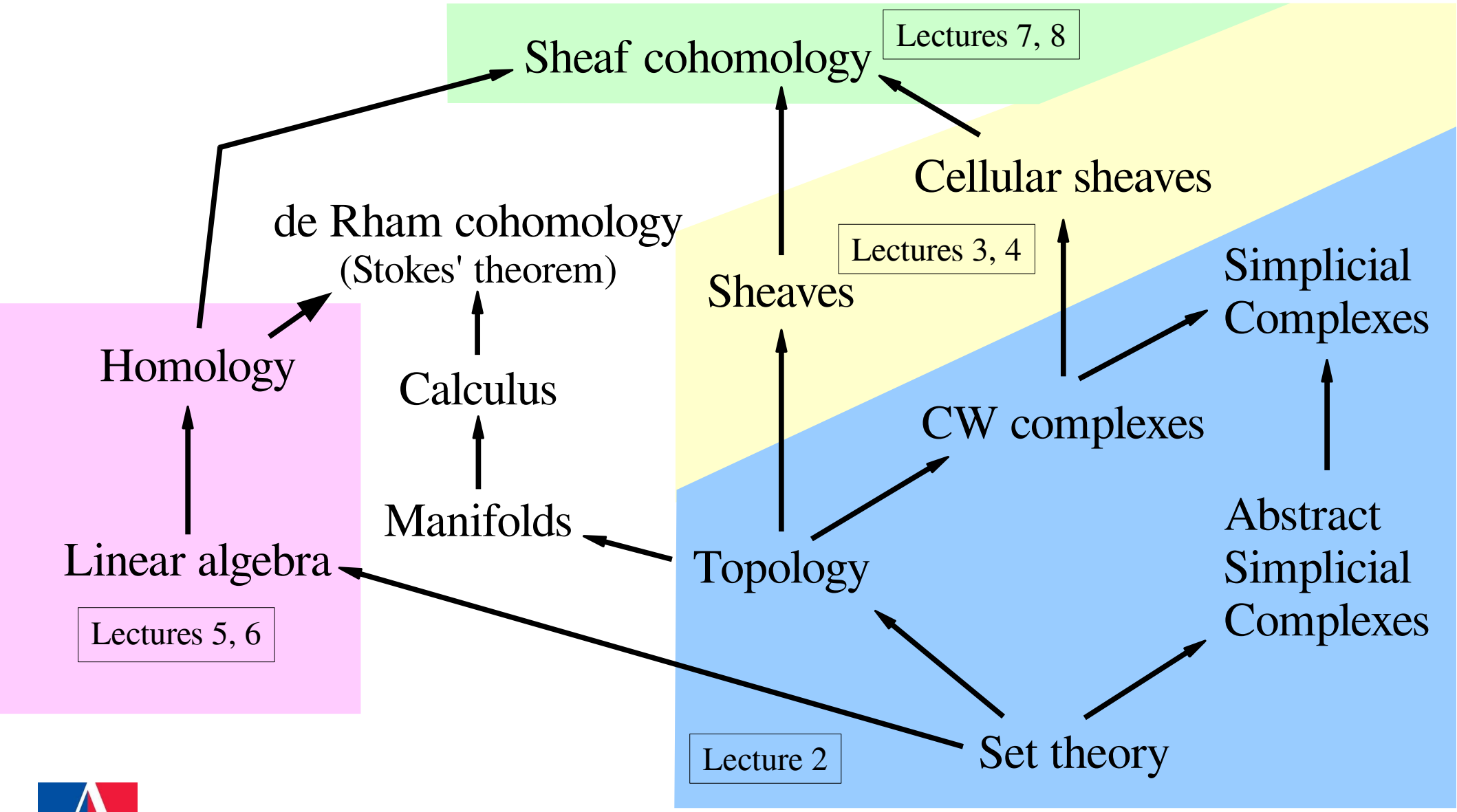
Data Structures as Sheaves



Michael Robinson

SIMPLEX Program

Mathematical dependency tree

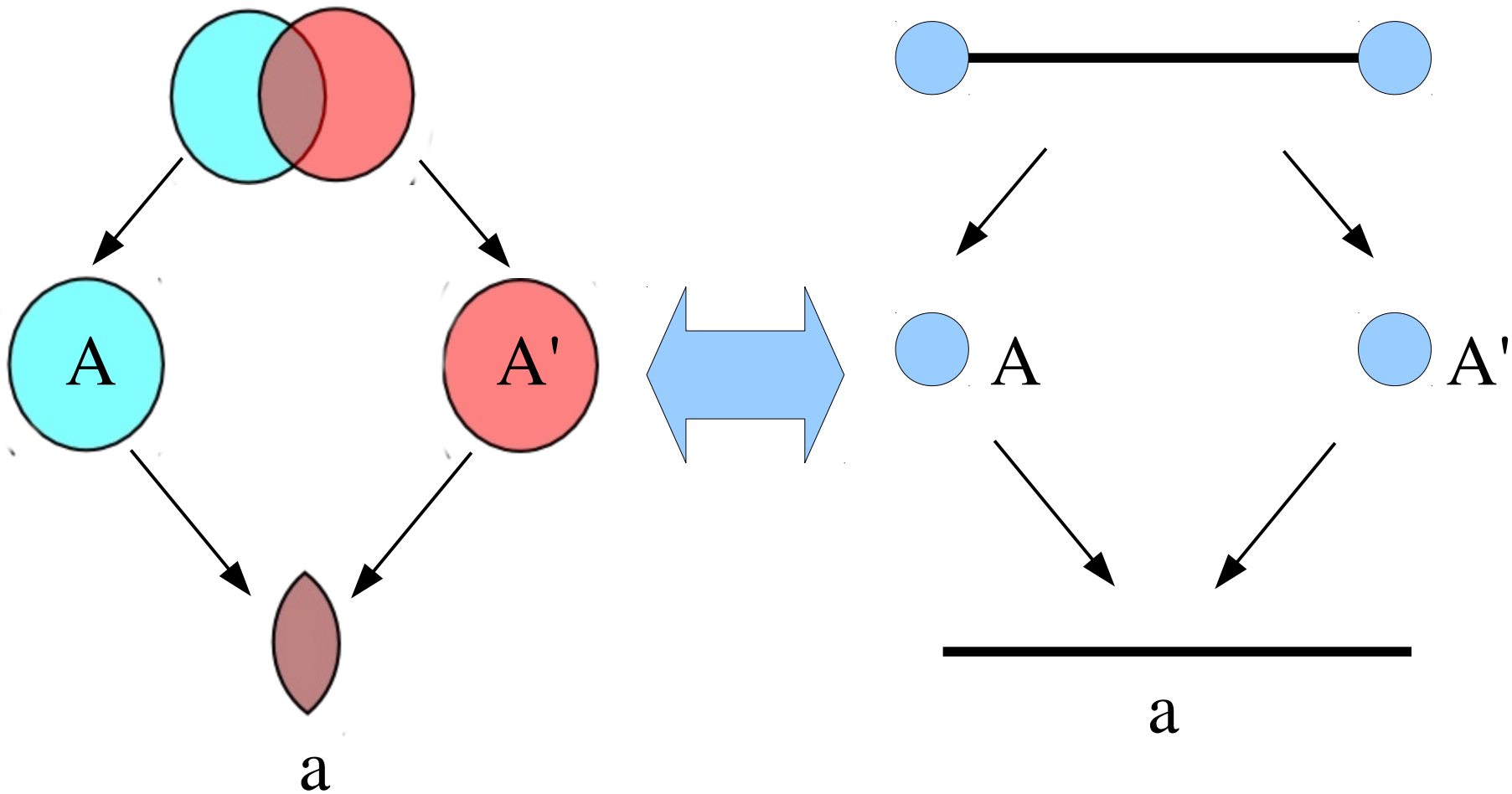


Michael Robinson

Session objectives

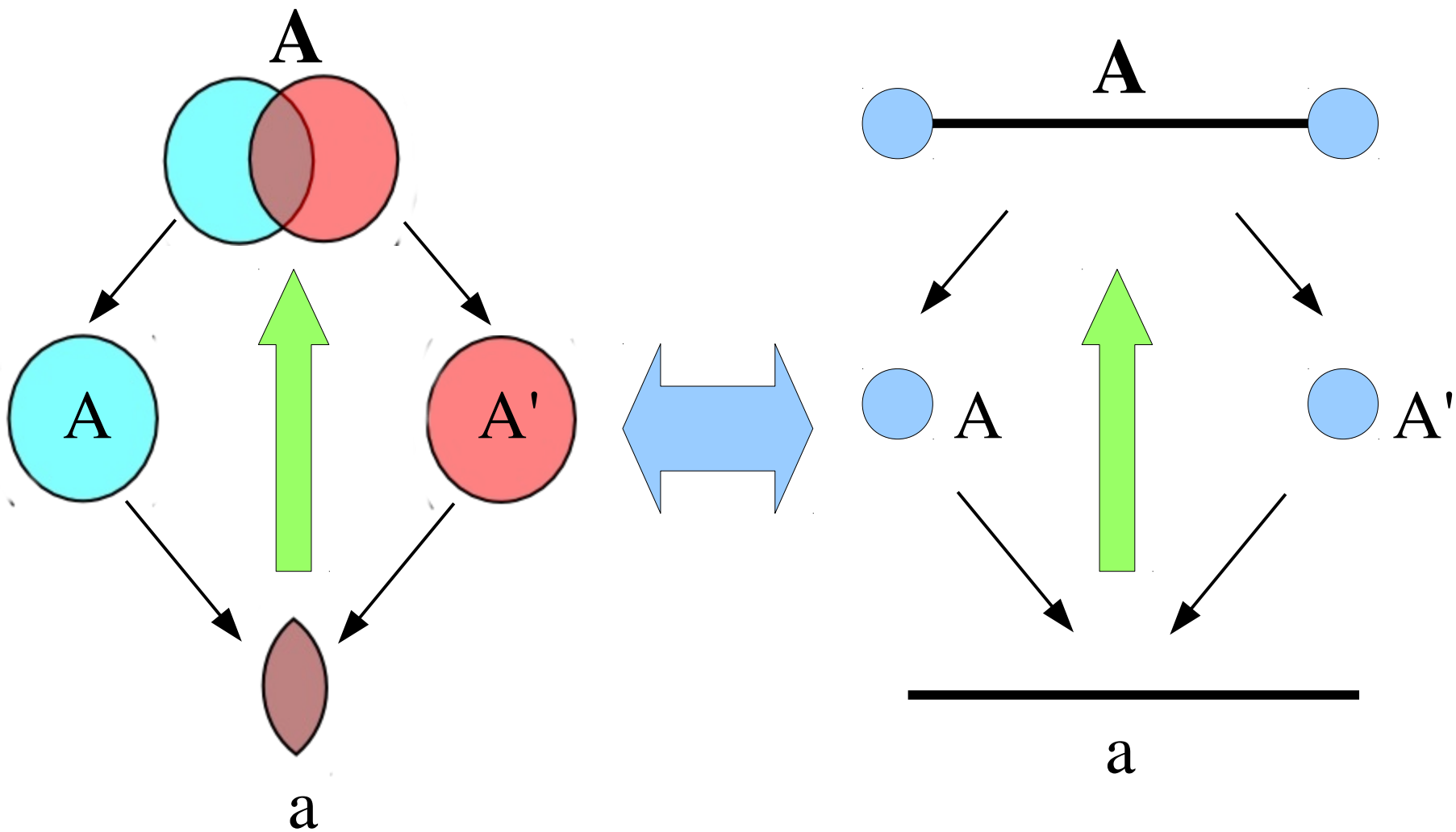
- How do sheaves extend well-known data structures?
- How do I translate between sheaf-based data structures?
- What can I do once I have a sheaf?

What is a sheaf?



Michael Robinson

What is a sheaf?



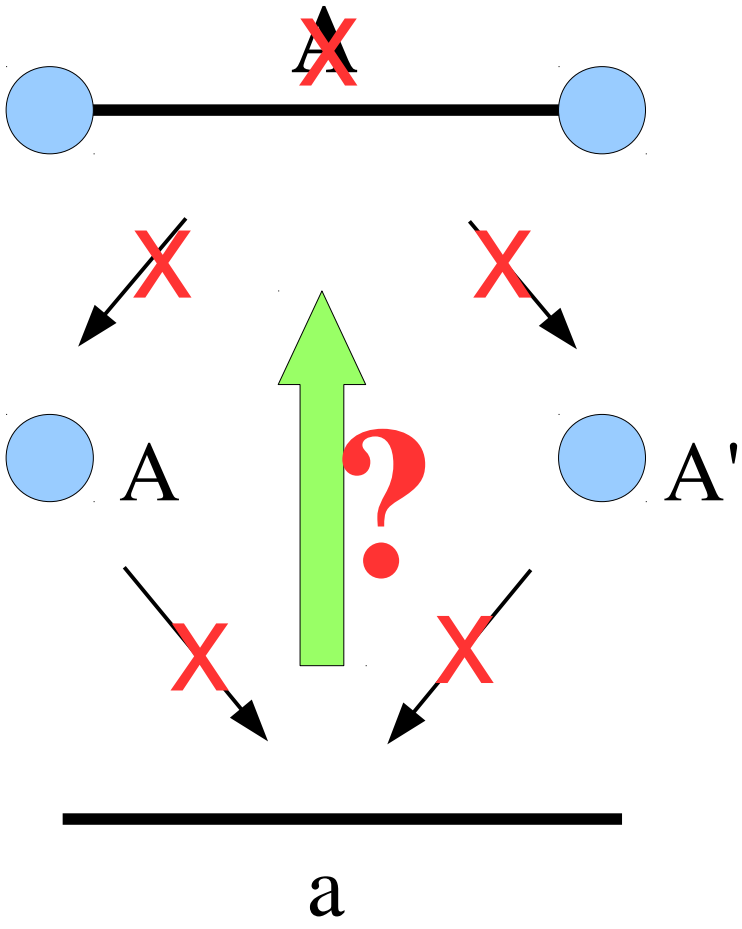
Michael Robinson

What isn't a sheaf?

If labels on the graph are not systematically related to one another...

Consistency checks become *ad hoc*

Cross-modality inference is no longer possible



Vertex- or (hyper)edge-weighted (hyper)graphs

Vertex weighted → sheaves

- Vertex has nontrivial stalk
- All restrictions are zero maps
- The resulting sheaf is flabby

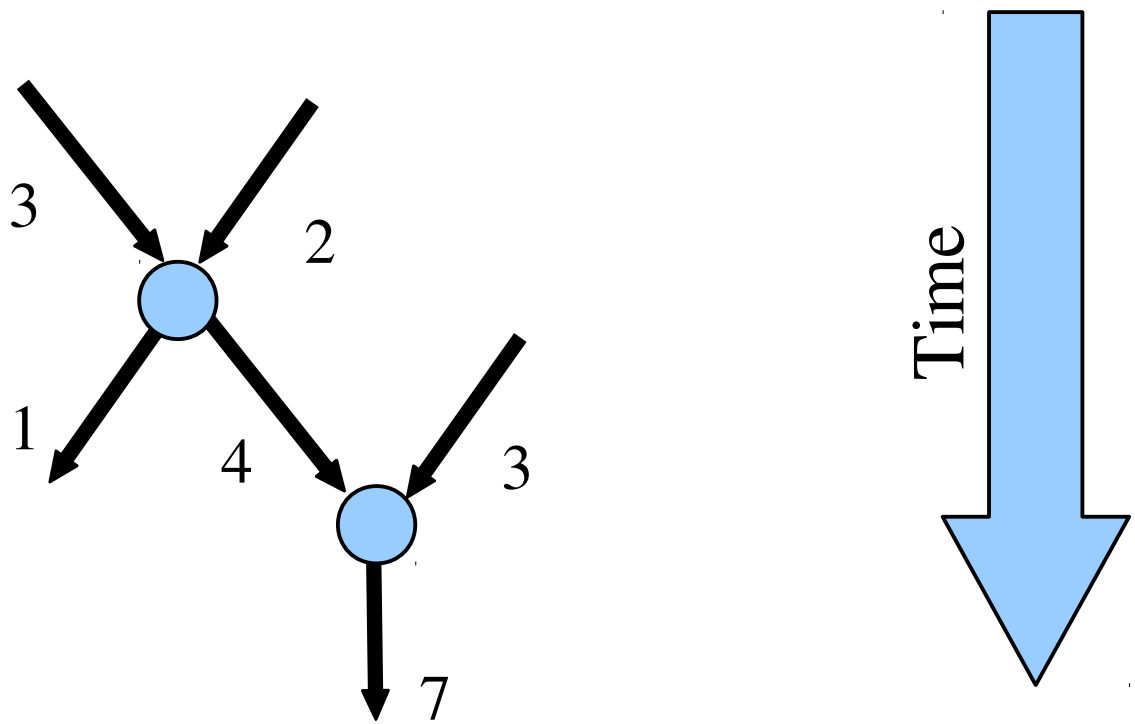
Hyperedge-weighted → cosheaves

- Toplex has nontrivial stalk
- All extensions are zero maps
- The resulting cosheaf is coflabby*

* This is such a fun term – but I don't see it much in use

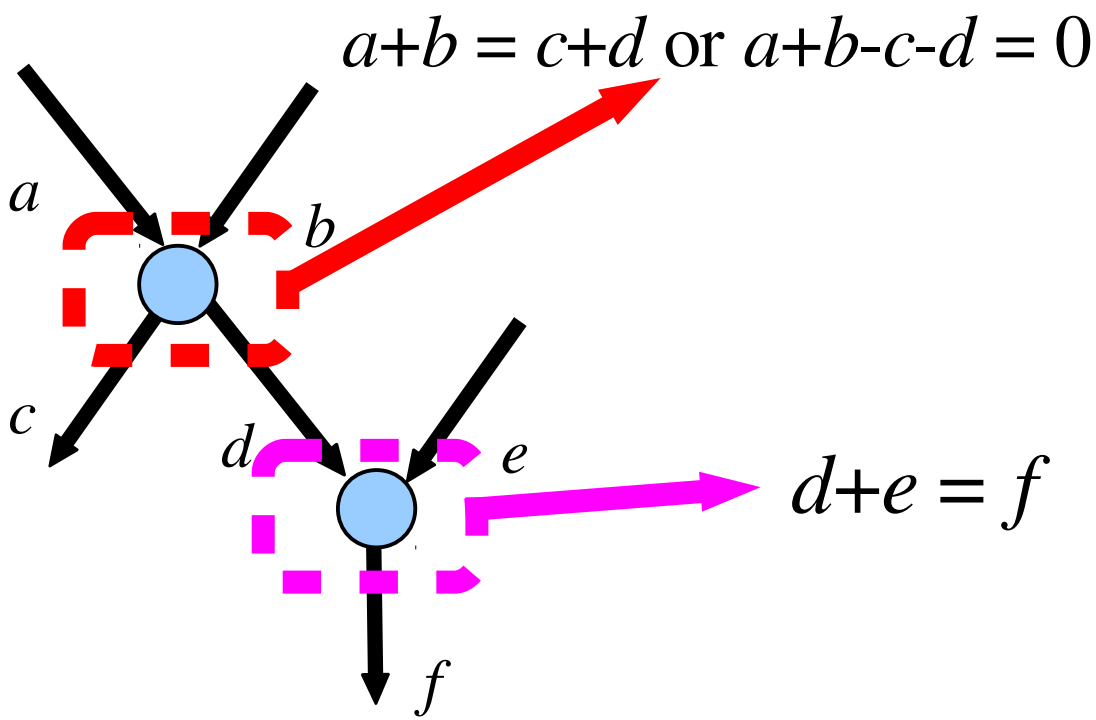
Flow sheaves

- Start with a collection of paths along which material flows
- Label each track segment with amount of material on that segment



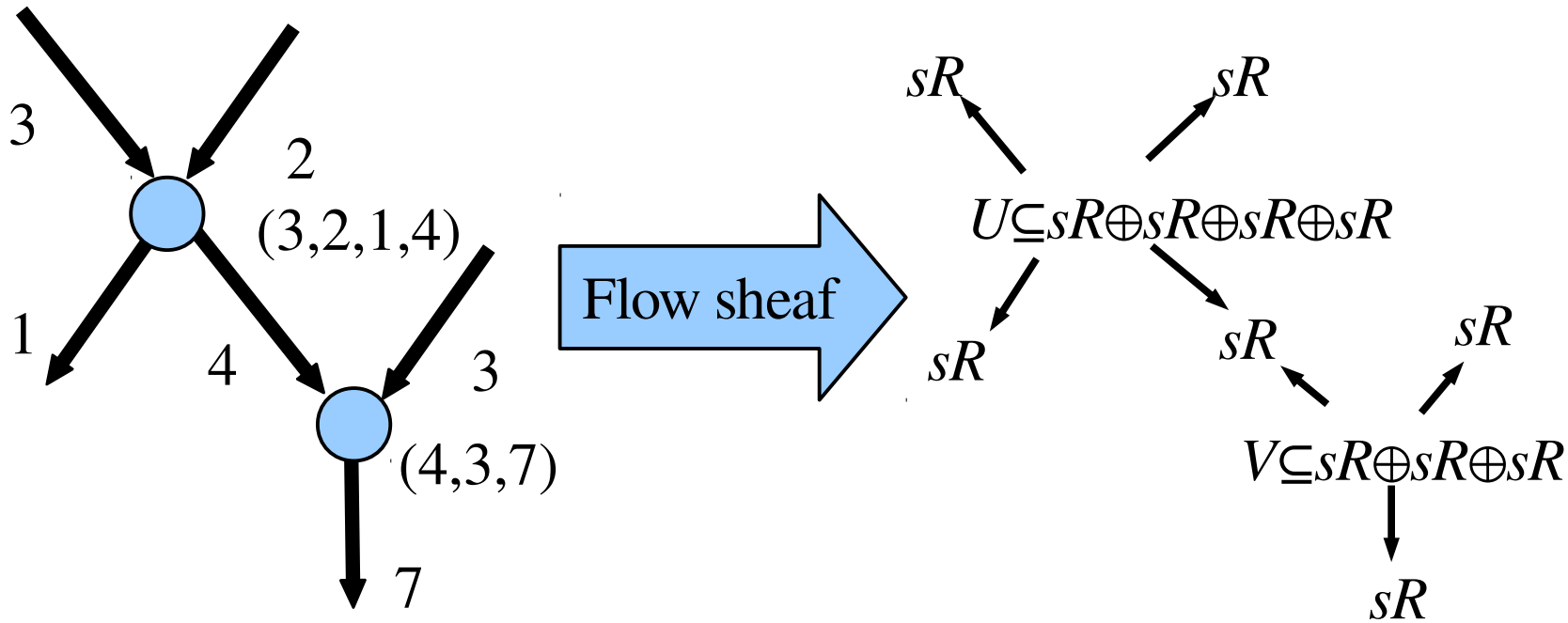
Flow sheaves

- Conservation law enforced at each vertex
- Depending on precisely how we count material (in $\mathbb{N}$ or $\mathbb{R}$, for instance), we might write the conservation law as



Flow sheaf

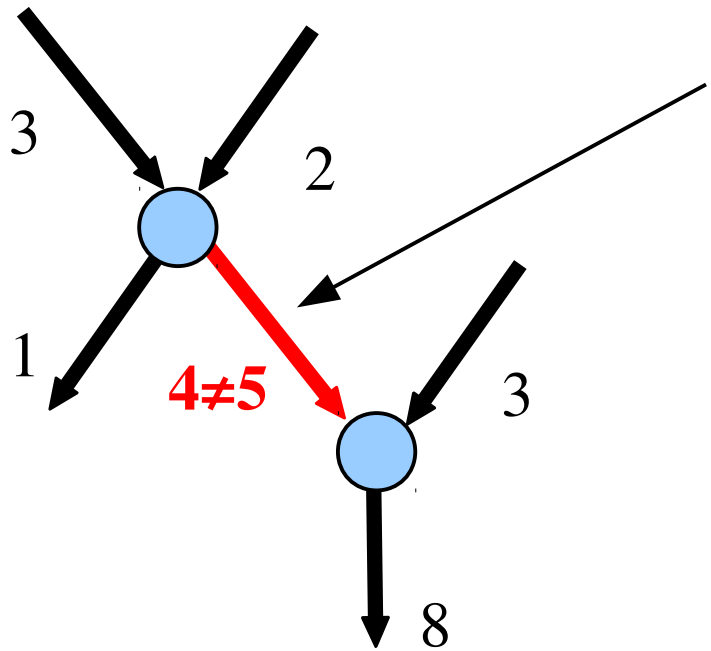
- Each degree n vertex is assigned a free sR-module* of rank $(n - 1)$ for material measured in a semiring sR
- Restriction maps are projections



* or just a vector space if it seems easier

Local consistency

A flow sheaves encodes a notion of consistency between adjacent faces



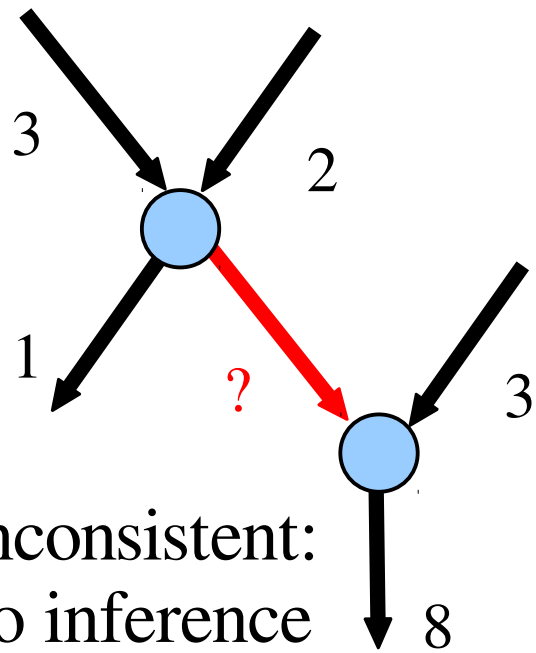
Cannot label this edge in a way consistent with the data

It's an indication that a flow was incorrectly measured somewhere in this vicinity

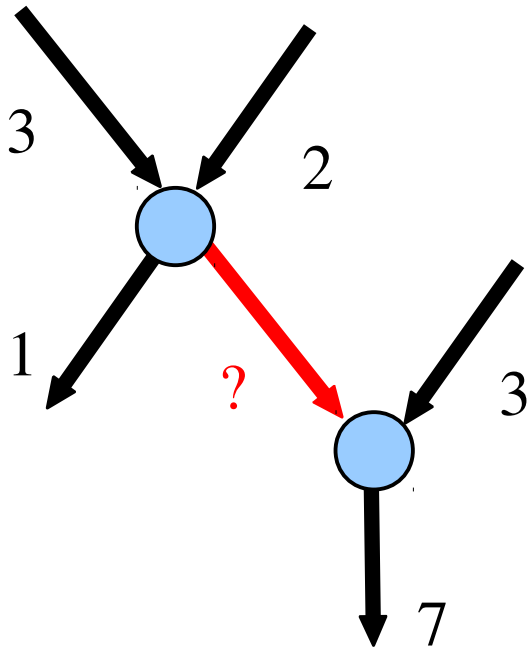
Michael Robinson

Inferential ambiguity

Depending on how we make measurements, we might not get "the full story" of the flow

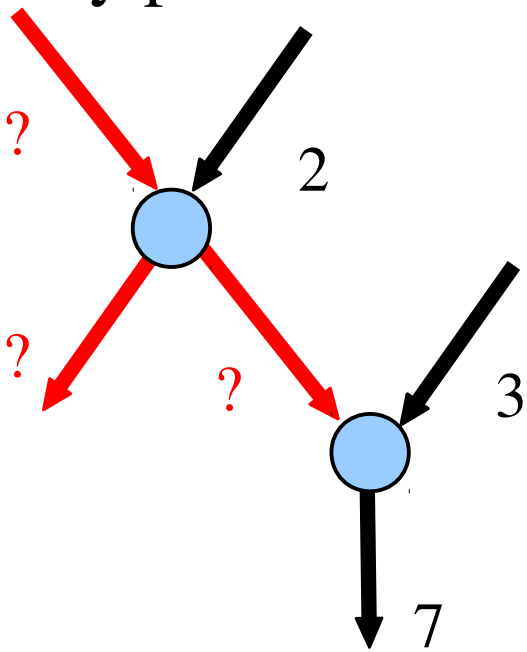


Inconsistent: no inference possible



Exactly one inference

Many possible inferences



Michael Robinson

Bayesian networks

Probability spaces

Start with a set of random variables $X_0, X_1, \ldots X_n$

Consider the set $P(X_0, X_1, \ldots, X_n)$ of all joint probability distributions over these random variables

- These are the nonnegative measures (generalized functions)

$$f = f(X_0, X_1, \ldots, X_n)$$

  with unit integral

- This is not a vector space – adding probability distributions doesn't yield another distribution

Marginalization cosheaf

There is a natural map

$$P(X_0, X_1, \ldots, X_n) \to P(X_0, X_1, \ldots, X_{n-1})$$

via *marginalization*, namely

$$f(X_0, X_1, \ldots, X_{n-1}) = \int f(X_0, X_1, \ldots, X_n)\, dX_n$$

- There similar maps for marginalizing out the other random variables, too
- This yields a cosheaf on the complete n-simplex!

Michael Robinson

Bayes' rule

Conditional probabilities produce maps going the other way… For instance,

$$P(X_0, X_1, \ldots, X_{n-1}) \to P(X_0, X_1, \ldots, X_n)$$

is parameterized by functions C

$$F(X_0, X_1, \ldots, X_n) = C(X_0, X_1, \ldots, X_n)\, f(X_0, X_1, \ldots, X_{n-1})$$

usually, one writes the arguments to C like

$$C = C(X_n \mid X_0, X_1, \ldots, X_{n-1})$$

So… conditional probabilities yield a sheaf on part of the n-simplex

Michael Robinson

Small Bayes net example

- Consider two binary random variables X and Y with a given conditional $C(Y \mid X)$

$$P(X) \xleftarrow{\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}} P(X,Y) \xrightarrow{\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}} P(Y)$$

Marginalization cosheaf

$$P(X) \xrightarrow{\begin{pmatrix} p(0|0) & 0 \\ p(1|0) & 0 \\ 0 & p(0|1) \\ 0 & p(1|1) \end{pmatrix} = C(Y \mid X)} P(X,Y) \qquad P(Y)$$

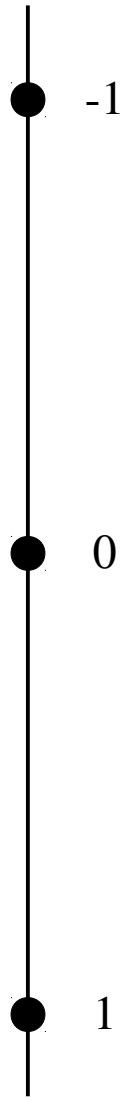
Conditional sheaf

Michael Robinson

Linear translation-invariant filters

How does a sheaf model a signal?

Simplicial complex for $\mathbb{R}$

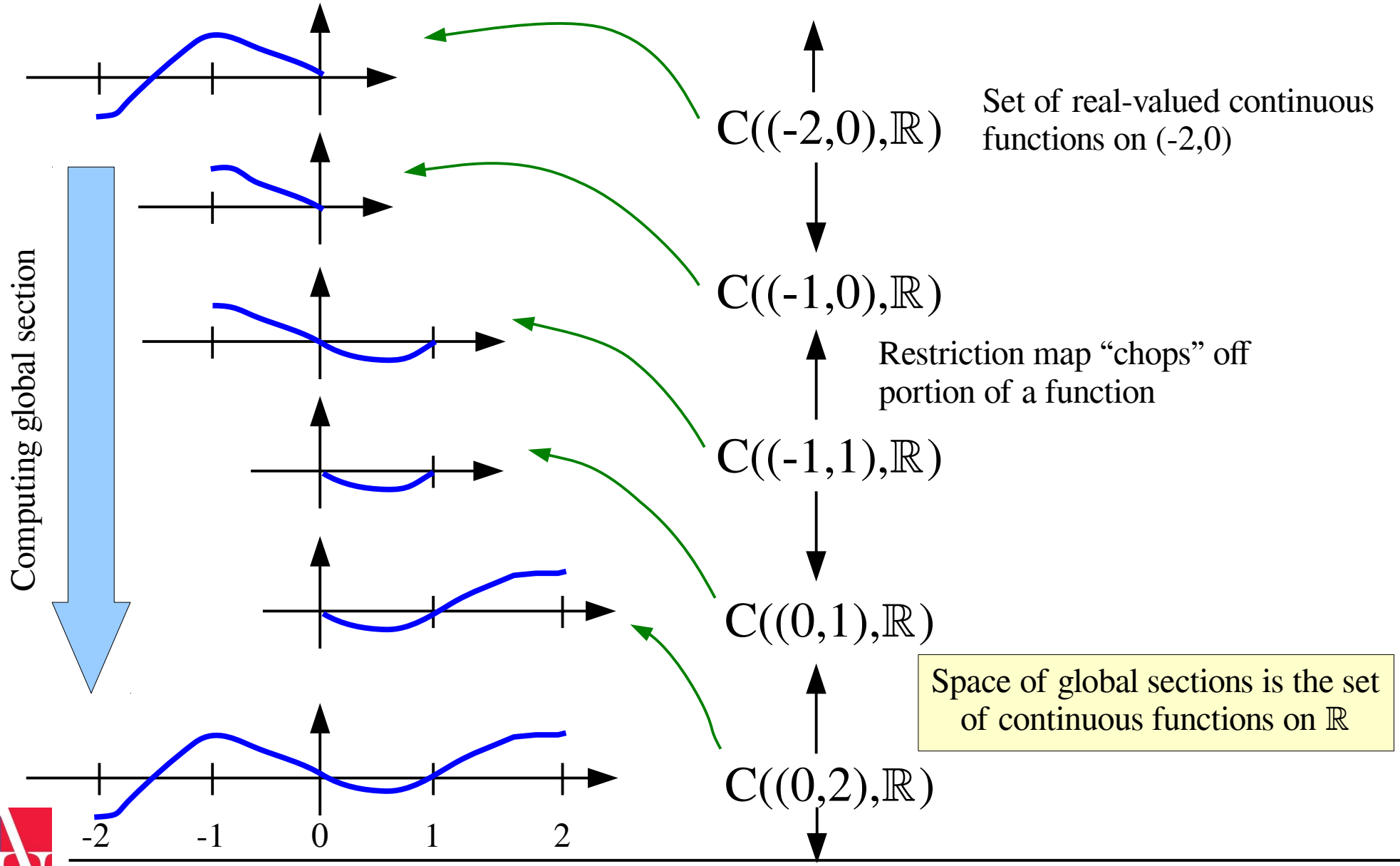


$$\begin{tikzcd}
C((-2,0),\mathbb{R}) \arrow[d] \\
C((-1,0),\mathbb{R}) \\
C((-1,1),\mathbb{R}) \arrow[u] \arrow[d] \\
C((0,1),\mathbb{R}) \\
C((0,2),\mathbb{R}) \arrow[u] \arrow[d]
\end{tikzcd}$$

The sheaf of continuous functions

Michael Robinson

How does a sheaf model a signal?



Michael Robinson

A *sheaf morphism* ...

- … takes data in the stalks of two sheaves …

$$\begin{tikzcd}
 & \mathbb{R}^2 \arrow[dl, "\begin{pmatrix}1&1\\2&0\end{pmatrix}"'] \arrow[dr, "\begin{pmatrix}2&0\\1&1\end{pmatrix}"] & \\
\mathbb{R}^2 & & \mathbb{R}^2 \\
 & \mathbb{R}^3 \arrow[ul, "\begin{pmatrix}0&1&0\\1&0&0\end{pmatrix}"] \arrow[ur, "\begin{pmatrix}1&0&0\\0&1&0\end{pmatrix}"'] &
\end{tikzcd}$$

$$\begin{tikzcd}
\mathbb{R}^2 \arrow[d, "\begin{pmatrix}1&0\\0&1\end{pmatrix}"] \\
\mathbb{R}^2 \\
\mathbb{R}^2 \arrow[u, "\begin{pmatrix}0&1\\1&0\end{pmatrix}"']
\end{tikzcd}$$

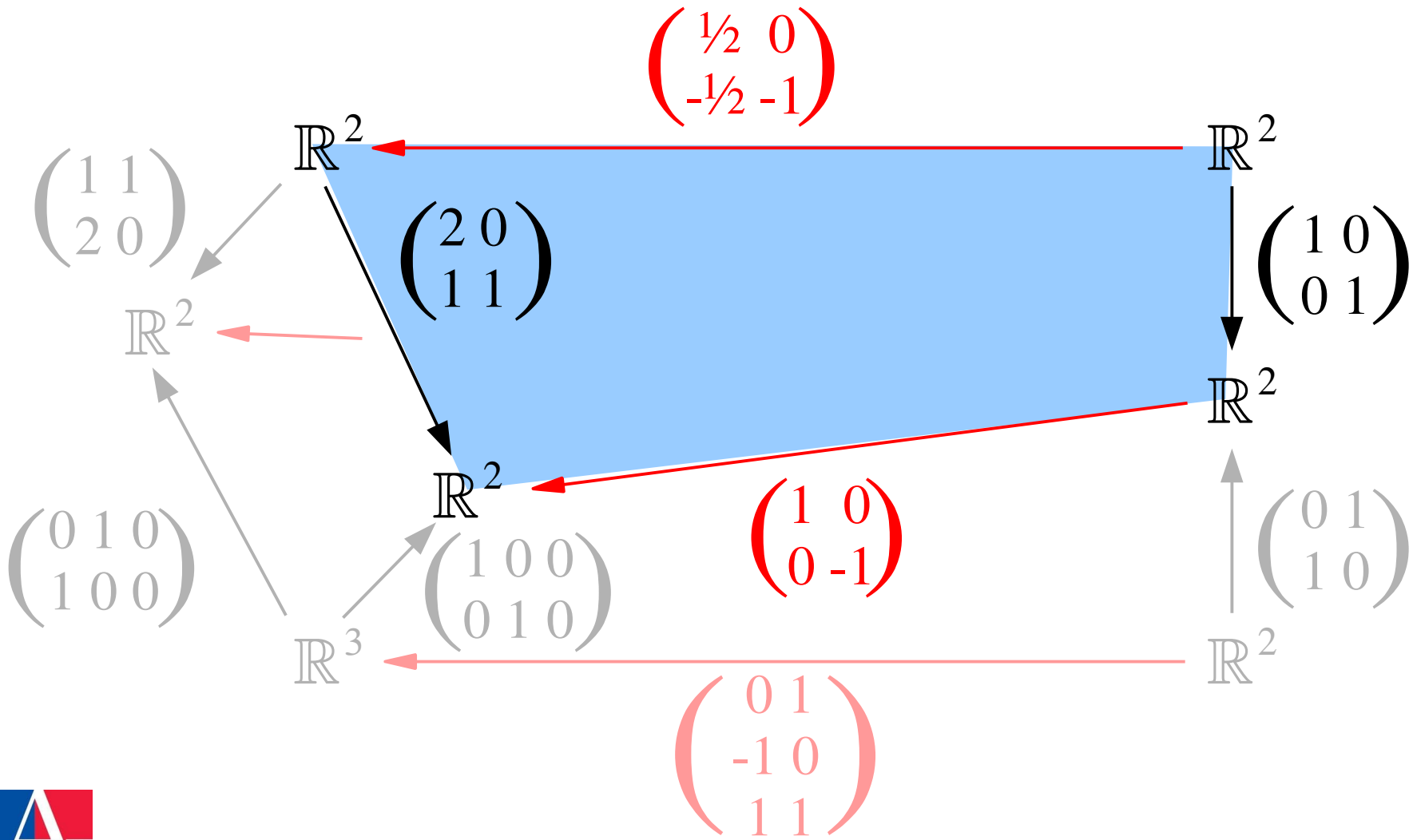
A *sheaf morphism* …

- … and relates them through linear maps …

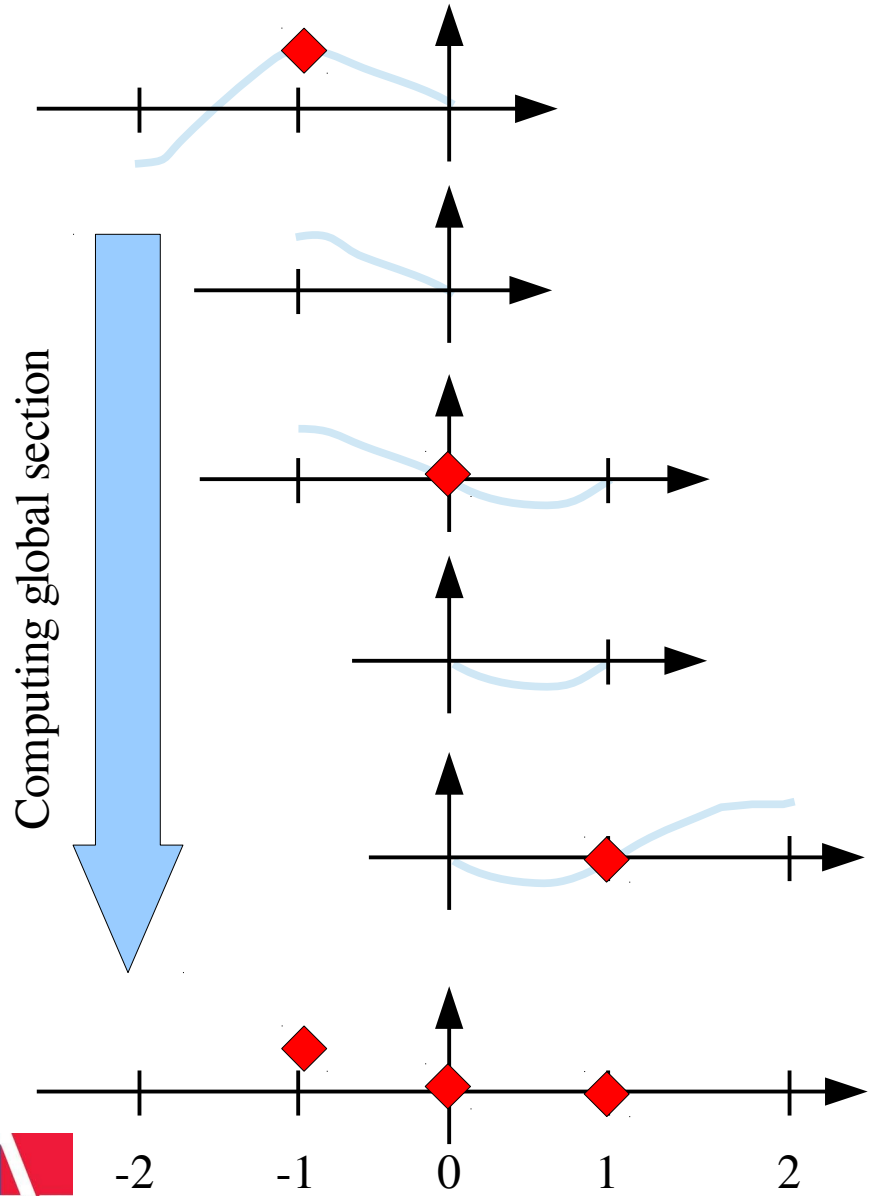
$$
\begin{tikzcd}
 & \mathbb{R}^2 \arrow[dl, "\begin{pmatrix}1&1\\2&0\end{pmatrix}"'] \arrow[dr, "\begin{pmatrix}2&0\\1&1\end{pmatrix}"] & & \mathbb{R}^2 \arrow[ll, "\begin{pmatrix}\frac{1}{2}&0\\-\frac{1}{2}&-1\end{pmatrix}"'] \arrow[d, "\begin{pmatrix}1&0\\0&1\end{pmatrix}"] \\
\mathbb{R}^2 & & \mathbb{R}^2 \arrow[ll, "\begin{pmatrix}0&-1\\1&0\end{pmatrix}"'] \arrow[dl, "\begin{pmatrix}1&0\\0&-1\end{pmatrix}"] \\
 & \mathbb{R}^3 \arrow[ul, "\begin{pmatrix}0&1&0\\1&0&0\end{pmatrix}"] \arrow[ur, "\begin{pmatrix}1&0&0\\0&1&0\end{pmatrix}"'] & & \mathbb{R}^2 \arrow[ll, "\begin{pmatrix}0&1\\-1&0\\1&1\end{pmatrix}"] \arrow[u, "\begin{pmatrix}0&1\\1&0\end{pmatrix}"']
\end{tikzcd}
$$

A *sheaf morphism* ...

- … so the diagram commutes!



A *sampling morphism* is ...



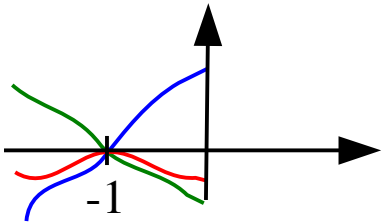
$$
\begin{tikzcd}
C((-2,0),\mathbb{R}) \arrow[r, "\text{Evaluate at } -1"'] & \mathbb{R} \\
C((-1,0),\mathbb{R}) \arrow[u] \arrow[d] \arrow[r] & 0 \arrow[u] \arrow[d] \\
C((-1,1),\mathbb{R}) \arrow[r, "\text{Evaluate at } 0"'] & \mathbb{R} \\
C((0,1),\mathbb{R}) \arrow[u] \arrow[d] \arrow[r] & 0 \arrow[u] \arrow[d] \\
C((0,2),\mathbb{R}) \arrow[r, "\text{Evaluate at } 1"'] & \mathbb{R}
\end{tikzcd}
$$

Michael Robinson

An *ambiguity sheaf* is ...

- The collection of *kernels* of these sampling maps

The set of continuous functions on (-2,0) that vanish at -1

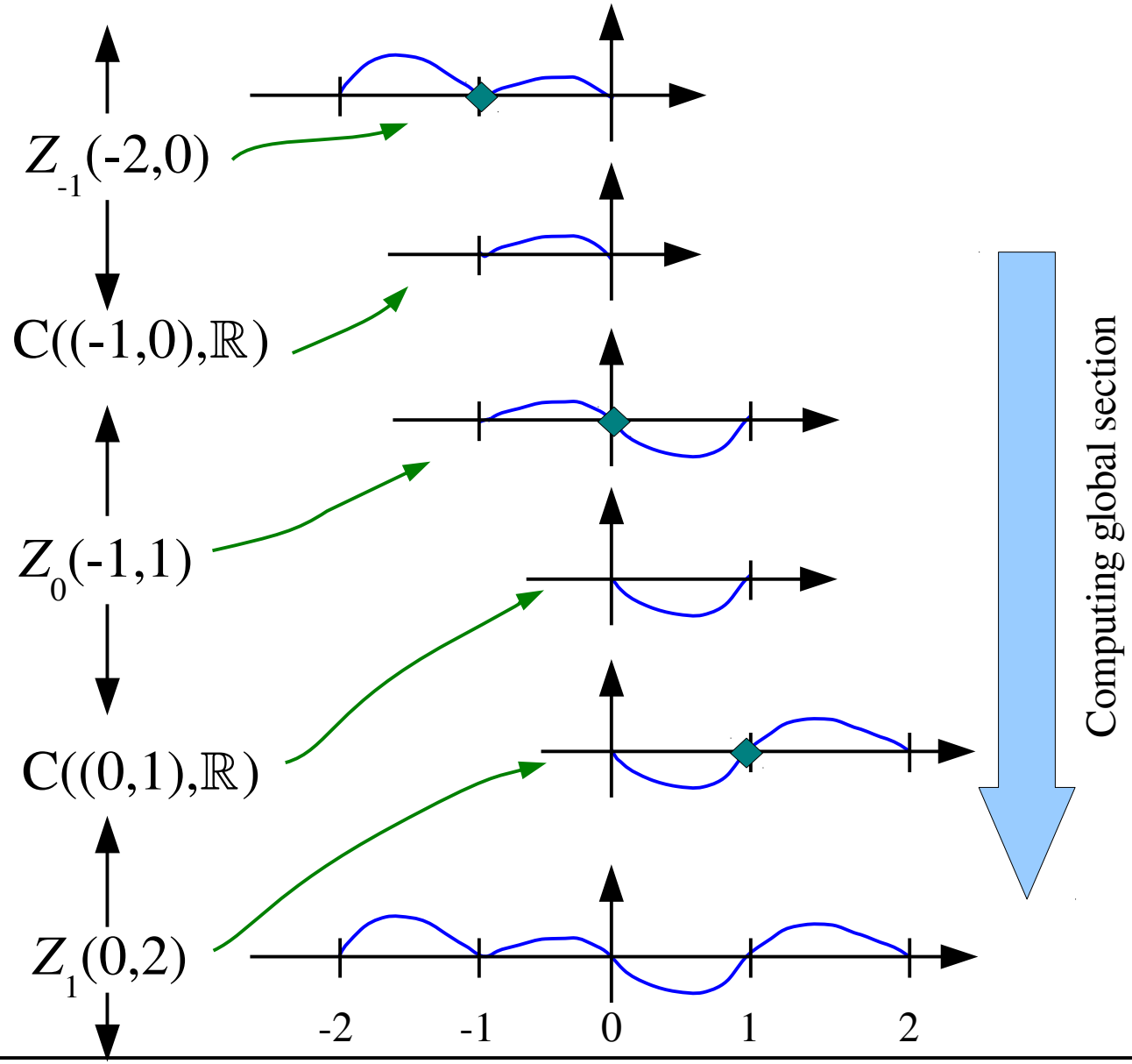


$$
\begin{tikzcd}
Z_{-1}(-2,0) \arrow[r] \arrow[d] & C((-2,0),\mathbb{R}) \arrow[r, "\text{Evaluate at } -1"'] \arrow[d] & \mathbb{R} \arrow[d] \\
C((-1,0),\mathbb{R}) \arrow[r] & C((-1,0),\mathbb{R}) \arrow[r] & 0 \\
Z_0(-1,1) \arrow[r] \arrow[u] \arrow[d] & C((-1,1),\mathbb{R}) \arrow[r, "\text{Evaluate at } 0"'] \arrow[u] \arrow[d] & \mathbb{R} \arrow[u] \arrow[d] \\
C((0,1),\mathbb{R}) \arrow[r] & C((0,1),\mathbb{R}) \arrow[r] & 0 \\
Z_1(0,2) \arrow[r] \arrow[u] & C((0,2),\mathbb{R}) \arrow[r, "\text{Evaluate at } 1"'] \arrow[u] & \mathbb{R} \arrow[u]
\end{tikzcd}
$$

Michael Robinson

An *ambiguity sheaf* is ...

Sections of the ambiguity sheaf are functions that appear to be all zero under the sampling



Michael Robinson

The general sampling theorem

- Given a sheaf S and a sampling of it, construct the ambiguity sheaf A

<u>Theorem</u>:

- Perfect reconstruction of global sections of S is possible if and only if the only global section of A is the zero function

- "No ambiguities means it's possible to reconstruct"

Michael Robinson

Nyquist-Shannon sampling

- Encode signals as a sheaf of bandlimited functions *BF* over $\mathbb{R}$, with bandwidth *B*
- It's easier to work in the frequency domain:

$$BF = \{ f \in C(\mathbb{R},\mathbb{C}) \mid \operatorname{supp} f \subseteq [-B, B] \}$$

- Samples are taken at integers, obtained by inverse Fourier transform
- For instance at *n*, we sample using the function M_n

$$M_n(f) = \int_{-B}^{B} f\, e^{-2n\pi i x}\, dx$$

Nyquist-Shannon sampling

- The ambiguity sheaf identifies bandlimited functions with zeros at specific locations

$$A_n = \{ f \in BF \mid M_n(f) = 0 \}$$

$$= \{\text{Bandlimited functions that are zero at } n\}$$

$$\begin{tikzcd}
 & A_{-1} \arrow[r] \arrow[d] & BF \arrow[d, "\text{identity}"] & A_0 \arrow[l] \arrow[r] \arrow[d] & BF \arrow[d, "\text{identity}"] & A_1 \arrow[l] \arrow[r] \arrow[d] & \\
\leftarrow \arrow[r, phantom] & BF \arrow[l, "\text{identity}"'] \arrow[r, "\text{identity}"] \arrow[d, "M_{-1}"] & BF \arrow[l, "\text{identity}"'] \arrow[d] & BF \arrow[l, "\text{identity}"'] \arrow[r, "\text{identity}"] \arrow[d, "M_0"] & BF \arrow[l, "\text{identity}"'] \arrow[d] & BF \arrow[l, "\text{identity}"'] \arrow[r, "\text{identity}"] \arrow[d, "M_1"] & \rightarrow \\
 & \mathbb{C} \arrow[r] & 0 & \mathbb{C} \arrow[l] \arrow[r] & 0 & \mathbb{C} \arrow[l] \arrow[r] &
\end{tikzcd}$$

Michael Robinson

Nyquist-Shannon sampling

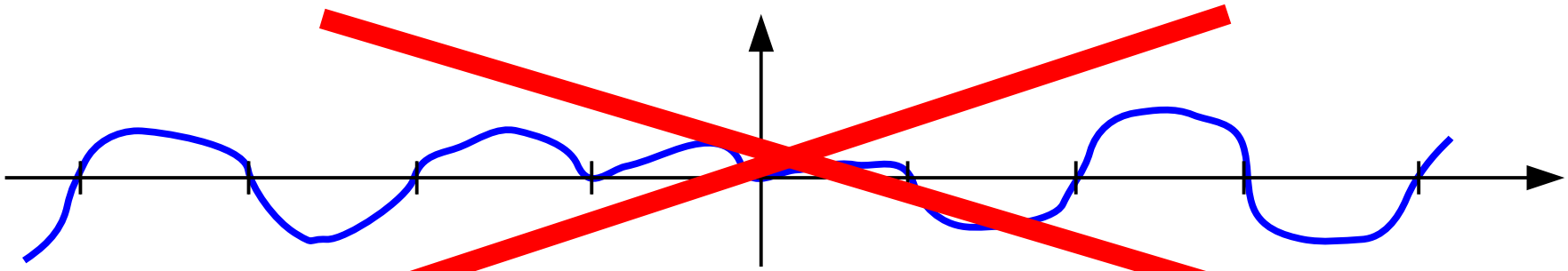
- The ambiguity sheaf identifies bandlimited functions with zeros at specific locations

$$A_n = \{ f \in BF \mid M_n(f) = 0 \}$$

$$= \{\text{Bandlimited functions that are zero at } n\}$$

$$\longleftarrow A_{-1} \longrightarrow BF \longleftarrow A_0 \longrightarrow BF \longleftarrow A_1 \longrightarrow$$

Global sections of the ambiguity sheaf are bandlimited functions that vanish at every integer



There aren't any if $B < 1/2$

Michael Robinson

Filters as sheaf morphisms

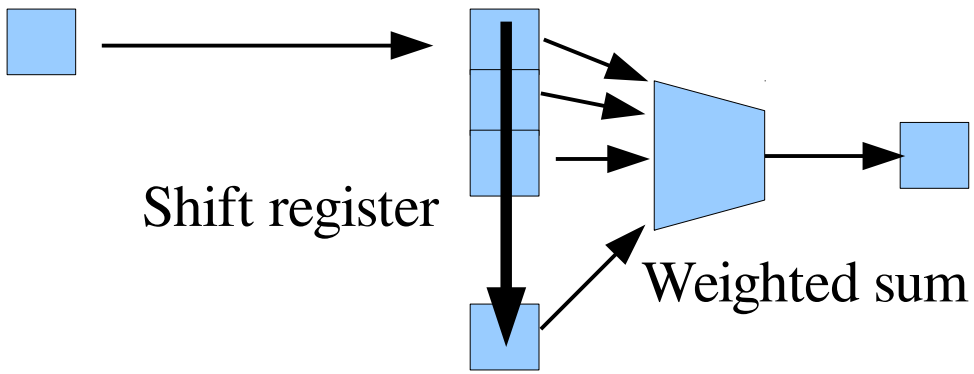
- <u>Theorem</u>: Every discrete-time LTI filter can be encoded as a sequence of two sheaf morphisms

$$S_1 \longleftarrow S_2 \longrightarrow S_3$$

Sheaf formalism

Input — Internal state — Output

Hardware



Michael Robinson

Input sheaf

- Sections of this sheaf are timeseries, instead of continuous functions

$$\longrightarrow 0 \longleftarrow \mathbb{R} \longrightarrow 0 \longleftarrow \mathbb{R} \longrightarrow 0 \longleftarrow \mathbb{R} \longrightarrow 0 \longleftarrow \mathbb{R}$$

Output sheaf

- The output sheaf is the same

$$\rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R}$$

$$\rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R}$$

The internal state

- Contents of the shift register at each timestep
- $N = 3$ shown

$$\rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R}$$

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

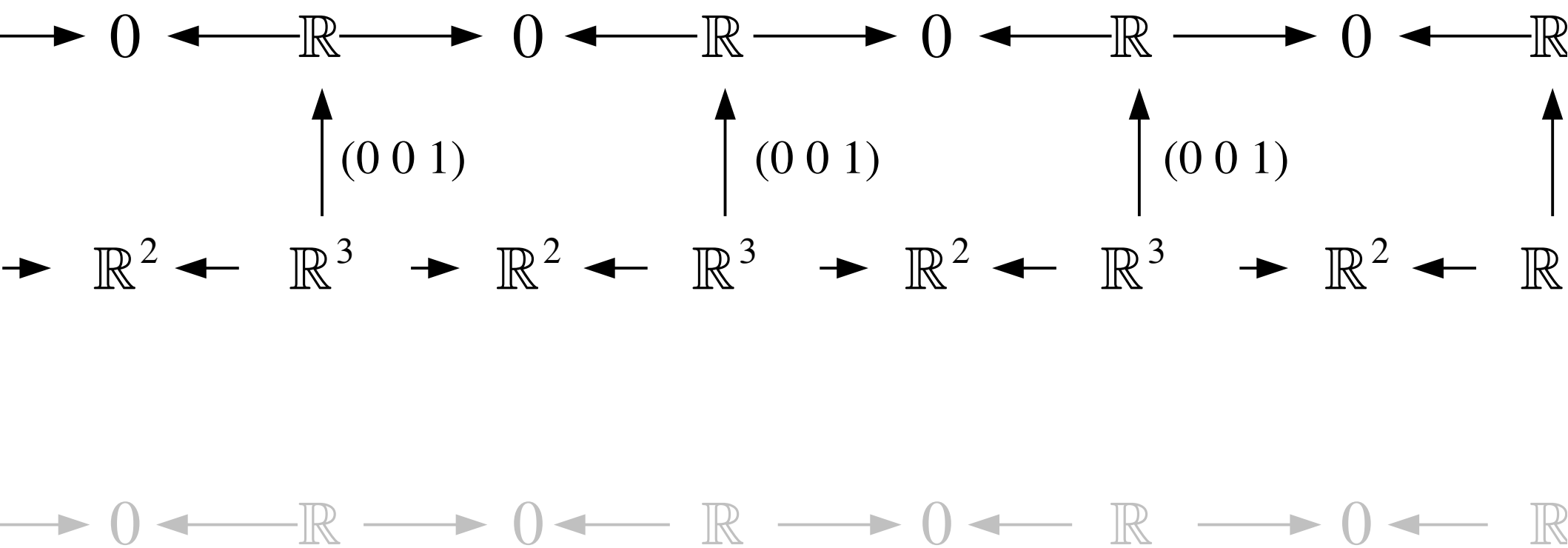
$$\rightarrow \mathbb{R}^2 \leftarrow \mathbb{R}^3 \rightarrow \mathbb{R}^2 \leftarrow \mathbb{R}^3 \rightarrow \mathbb{R}^2 \leftarrow \mathbb{R}^3 \rightarrow \mathbb{R}^2 \leftarrow \mathbb{R}$$

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R} \rightarrow 0 \leftarrow \mathbb{R}$$

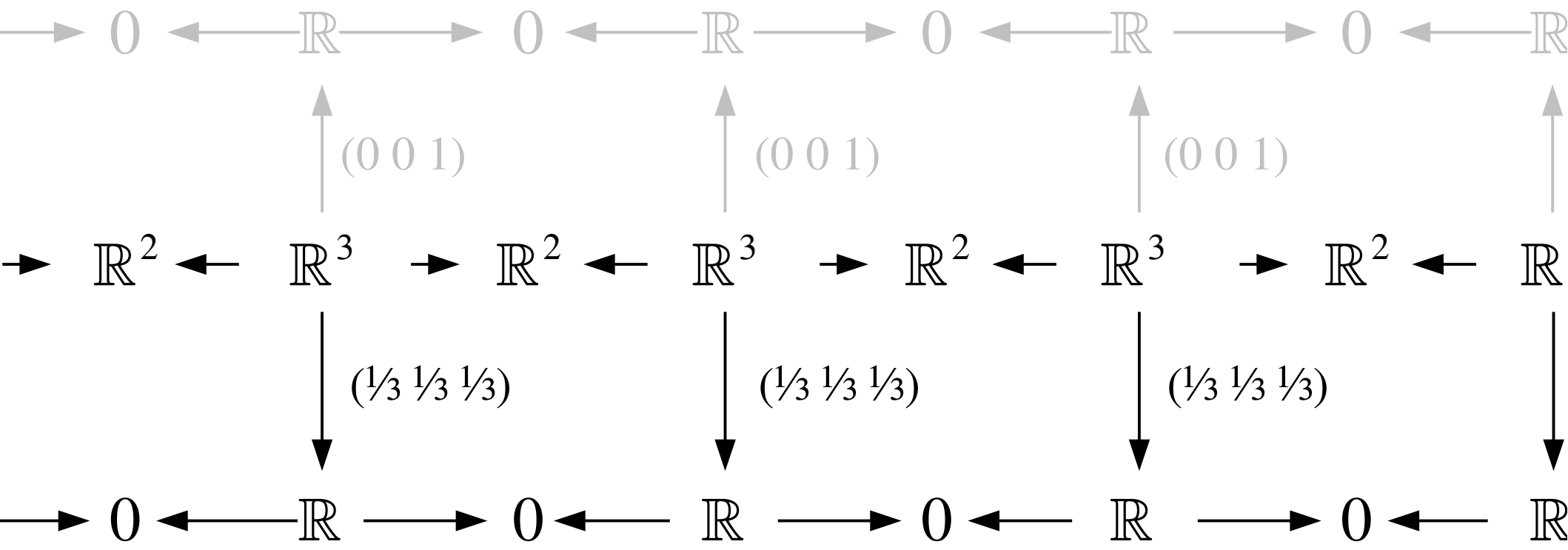
The internal state

- Loads a new value with each timestep



The internal state

- Produces average of the shift register at each timestep



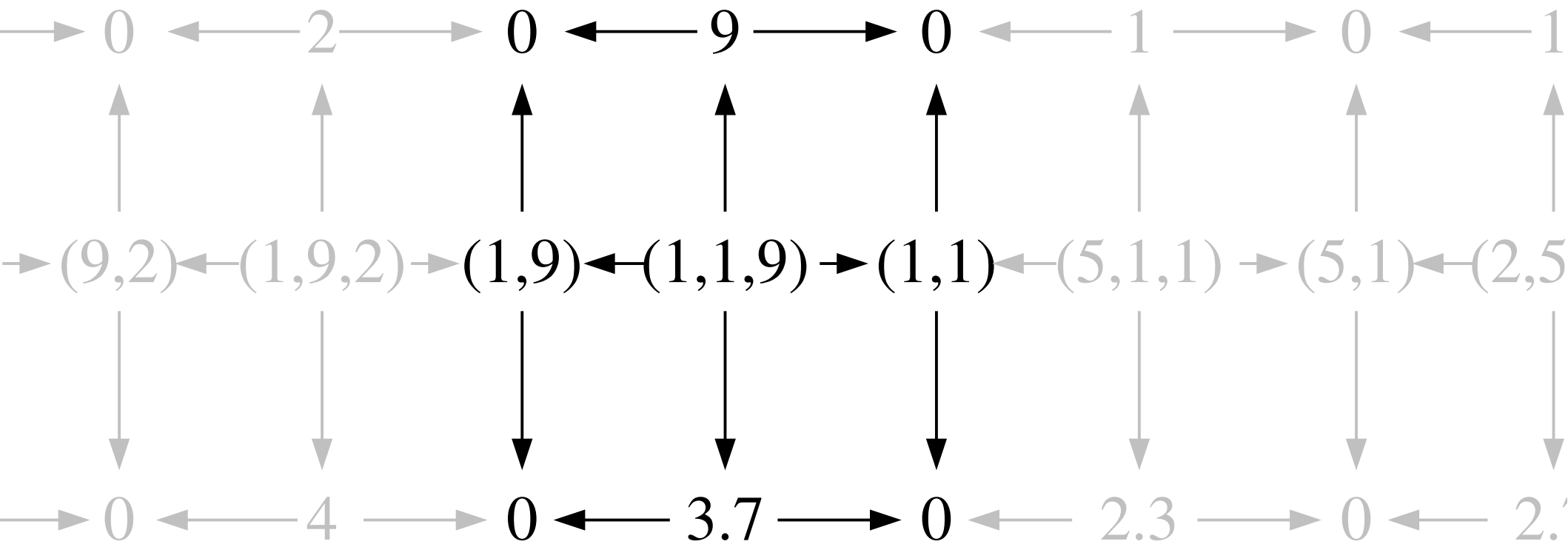
Finishing both morphisms

- Put in a few zero maps!

$$
\begin{tikzcd}
\arrow[r] & 0 & \mathbb{R} \arrow[l] \arrow[r] & 0 & \mathbb{R} \arrow[l] \arrow[r] & 0 & \mathbb{R} \arrow[l] \arrow[r] & 0 & \mathbb{R} \arrow[l] \\
\arrow[r] & \mathbb{R}^2 \arrow[u] \arrow[d] & \mathbb{R}^3 \arrow[l] \arrow[u] \arrow[d] \arrow[r] & \mathbb{R}^2 \arrow[u] \arrow[d] & \mathbb{R}^3 \arrow[l] \arrow[u] \arrow[d] \arrow[r] & \mathbb{R}^2 \arrow[u] \arrow[d] & \mathbb{R}^3 \arrow[l] \arrow[u] \arrow[d] \arrow[r] & \mathbb{R}^2 \arrow[u] \arrow[d] & \mathbb{R} \arrow[l] \arrow[u] \arrow[d] \\
\arrow[r] & 0 & \mathbb{R} \arrow[l] \arrow[r] & 0 & \mathbb{R} \arrow[l] \arrow[r] & 0 & \mathbb{R} \arrow[l] \arrow[r] & 0 & \mathbb{R} \arrow[l]
\end{tikzcd}
$$

A single timestep

- Sections of the sheaves linked together give possible input/output combinations of the filter



Of course, this extends...

- Sections of the sheaves linked together give possible input/output combinations of the filter

$$
\begin{tikzcd}
\rightarrow 0 & 2 \arrow[l] \arrow[r] & 0 & 9 \arrow[l] \arrow[r] & 0 & 1 \arrow[l] \arrow[r] & 0 & 1 \arrow[l] \\
\rightarrow (9,2) \arrow[u] \arrow[d] & (1,9,2) \arrow[l] \arrow[r] \arrow[u] \arrow[d] & (1,9) \arrow[l] \arrow[u] \arrow[d] & (1,1,9) \arrow[l] \arrow[r] \arrow[u] \arrow[d] & (1,1) \arrow[l] \arrow[u] \arrow[d] & (5,1,1) \arrow[l] \arrow[r] \arrow[u] \arrow[d] & (5,1) \arrow[l] \arrow[u] \arrow[d] & (2,5 \arrow[l] \arrow[u] \arrow[d] \\
\rightarrow 0 & 4 \arrow[l] \arrow[r] & 0 & 3.7 \arrow[l] \arrow[r] & 0 & 2.3 \arrow[l] \arrow[r] & 0 & 2. \arrow[l]
\end{tikzcd}
$$

How this formalism helps...

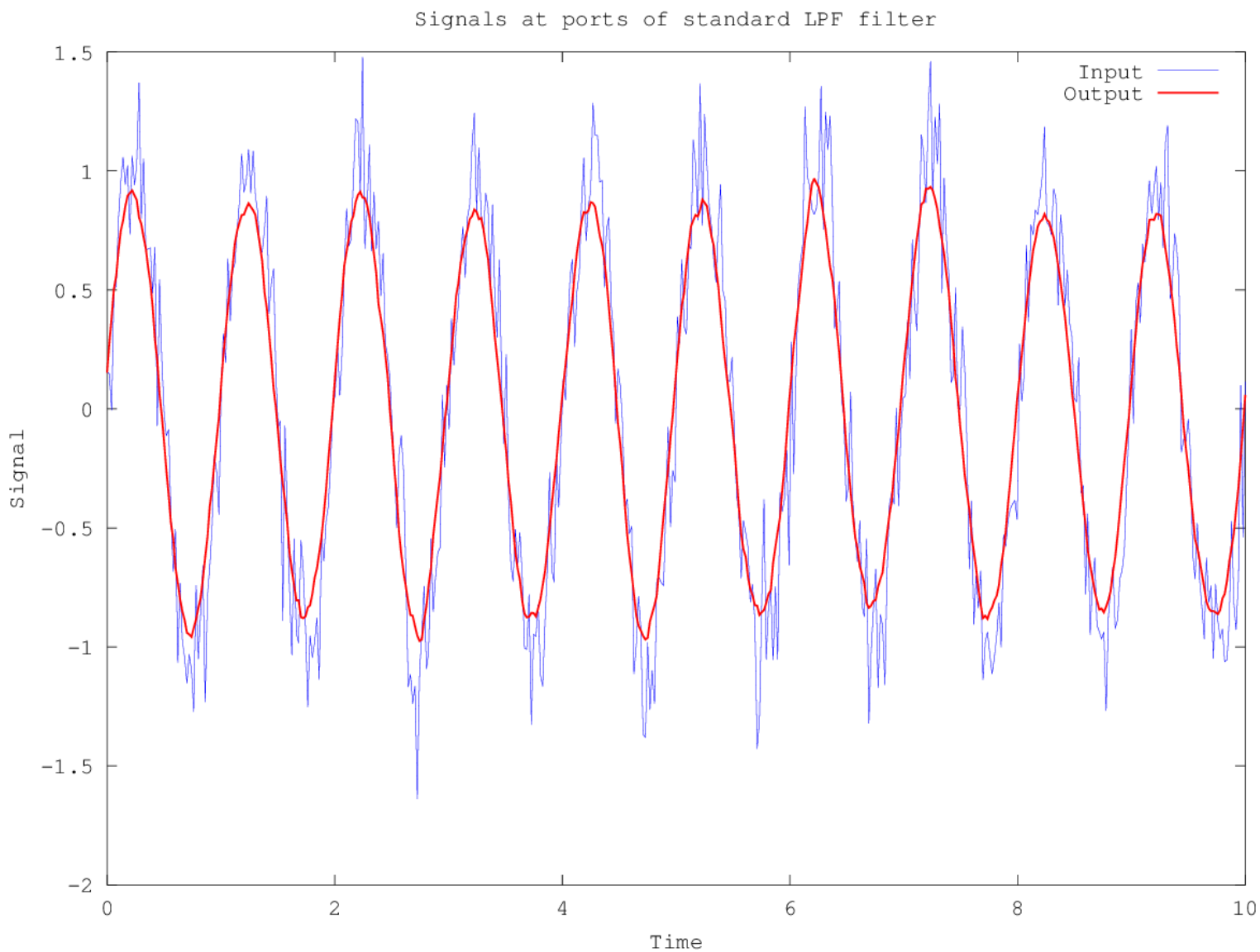
- Of course, it corresponds nicely to the hardware

… BUT...

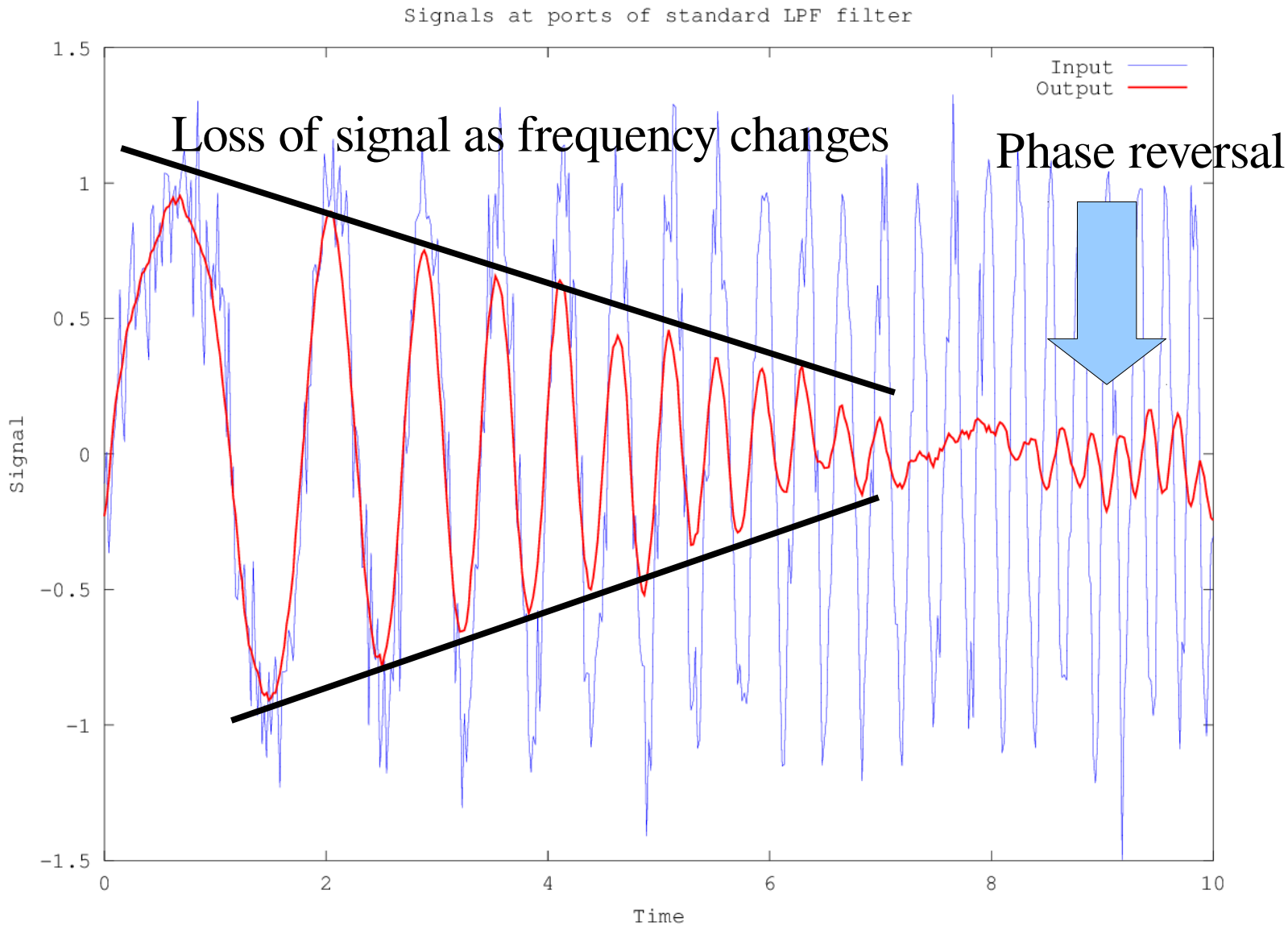
- It's easy to splice in **nonlinear** operations
- It works on nontrivial base spaces: A **systematic study** of filtering is **now possible** on cell complexes

Sheaves make it easy to invent *topological filters* that have controlled performance characteristics

Filtering sinosoids from noise

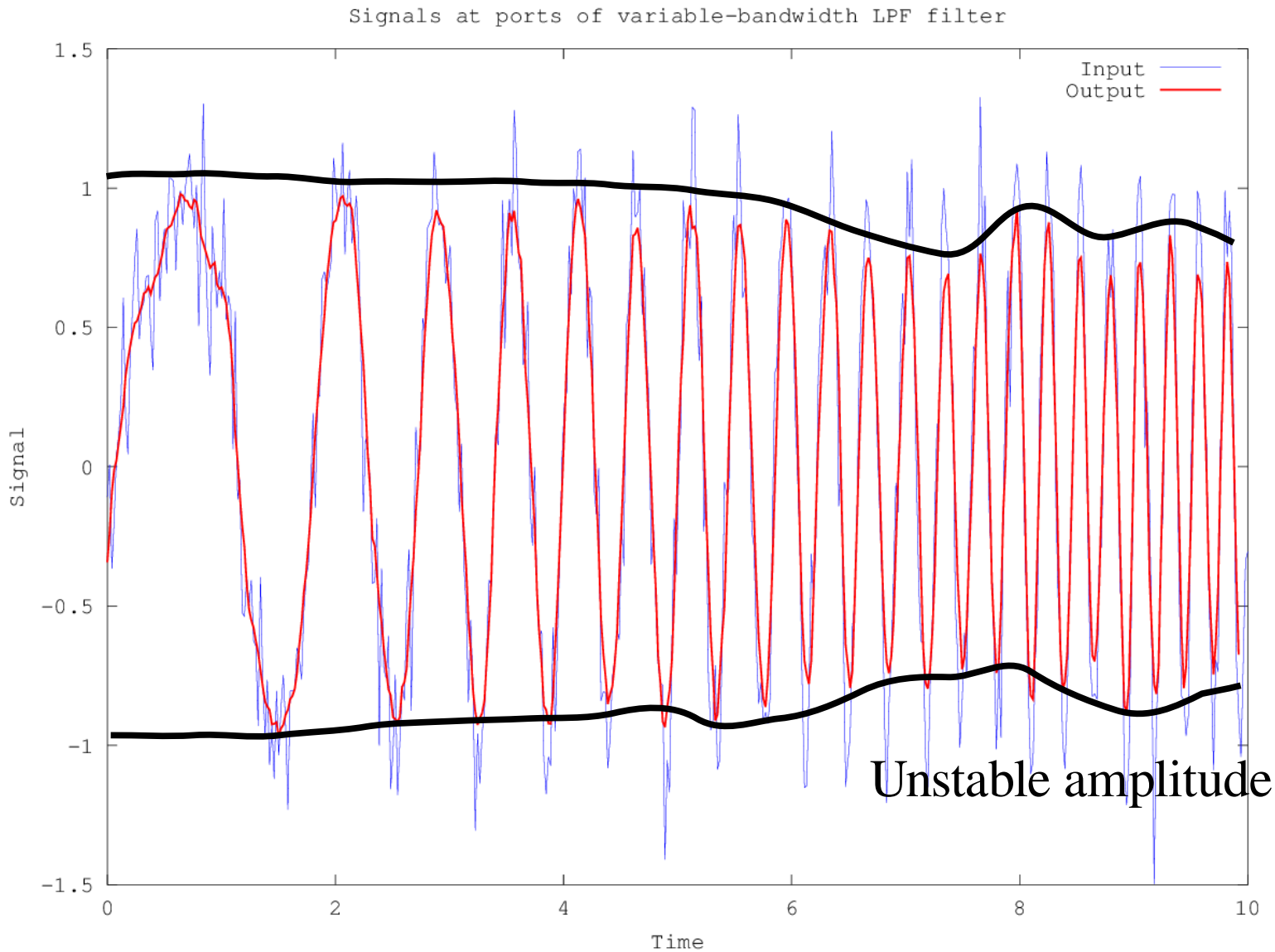


Filtering out chirpy signals



Michael Robinson

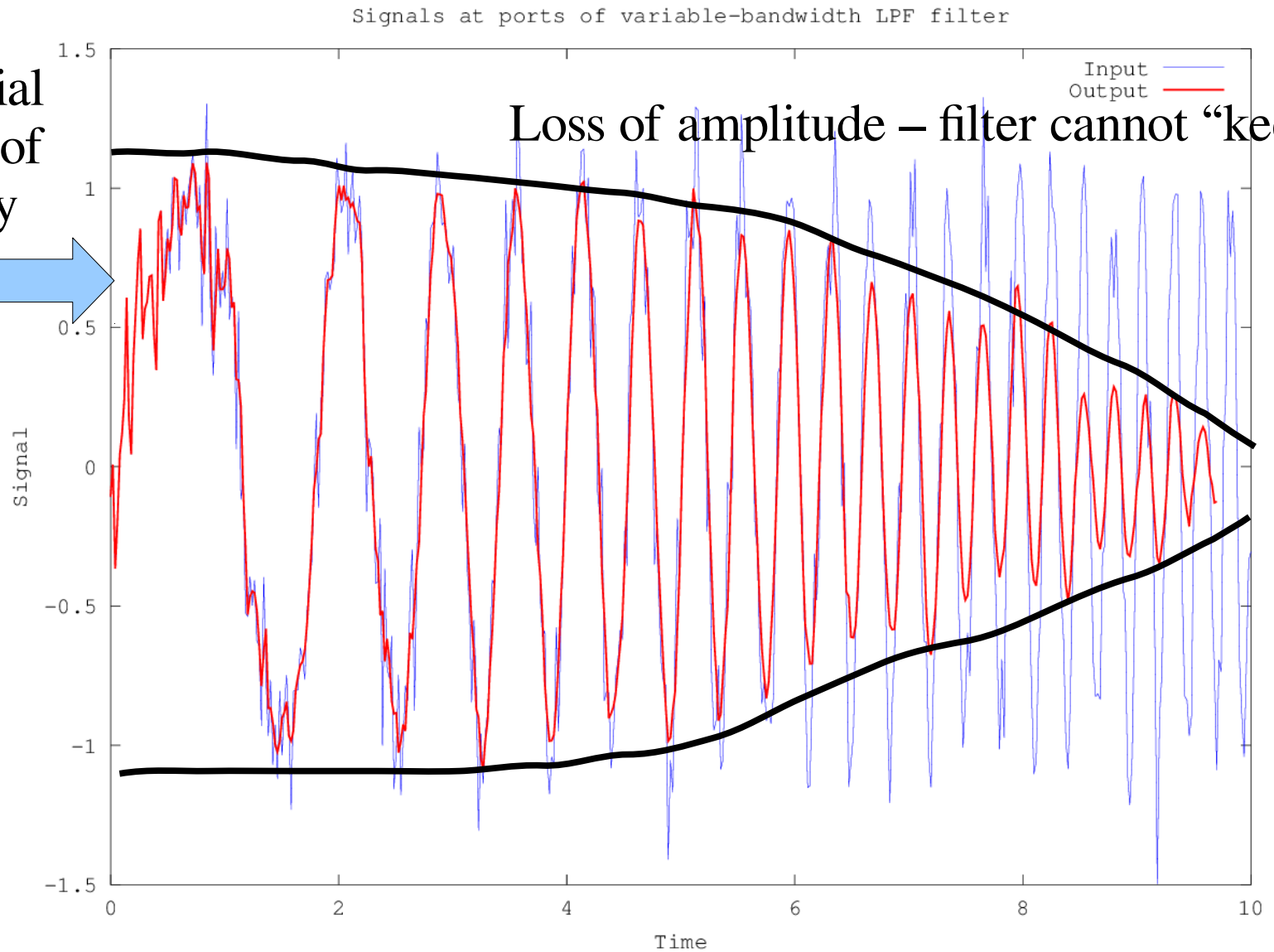
Filtering out chirpy signals



Michael Robinson

Filtering out chirpy signals

Poor initial estimate of frequency

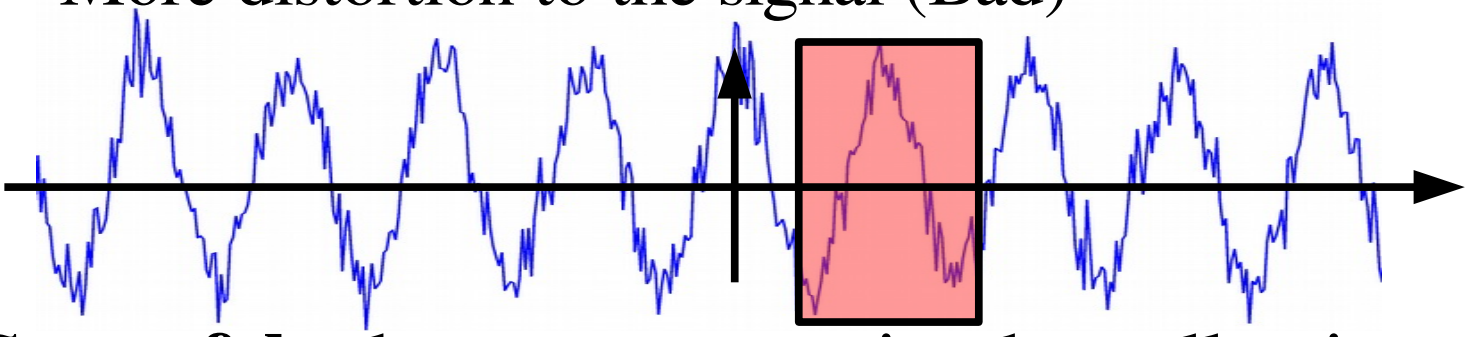


Loss of amplitude – filter cannot "keep up"!

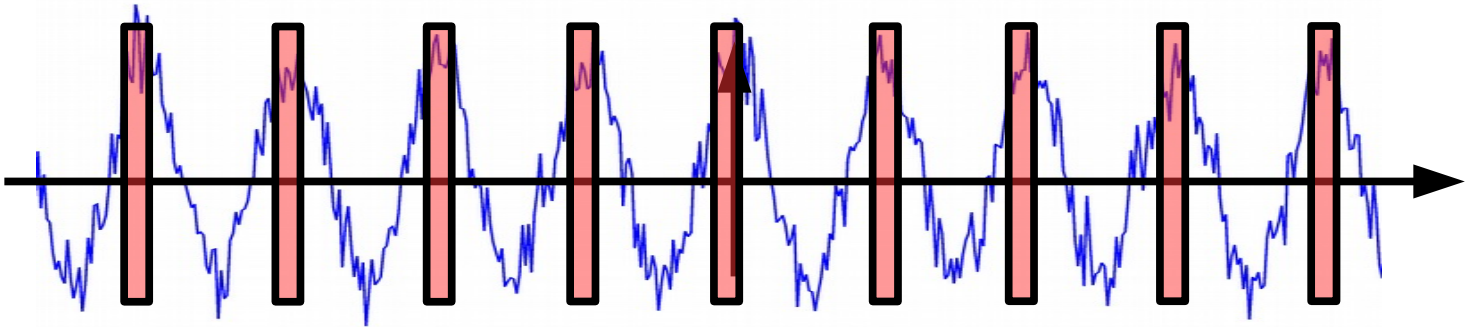
Michael Robinson

Circumventing bandwidth limits

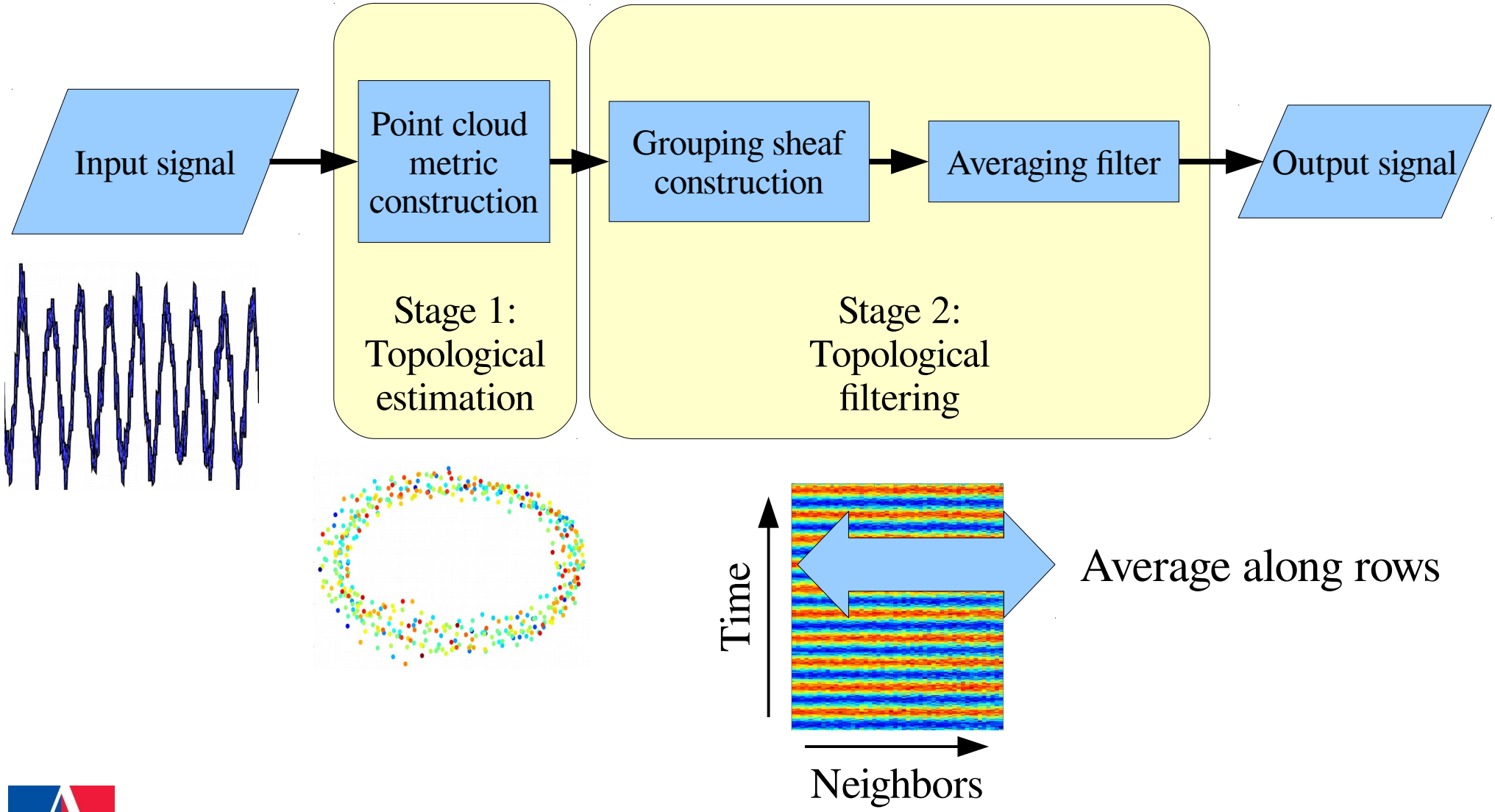
- More averaging in a connected window leads to:
  - More noise cancellation (Good)
  - More distortion to the signal (Bad)



- Can **safely** do **more** averaging by collecting samples at "similar places" across the **entire** signal

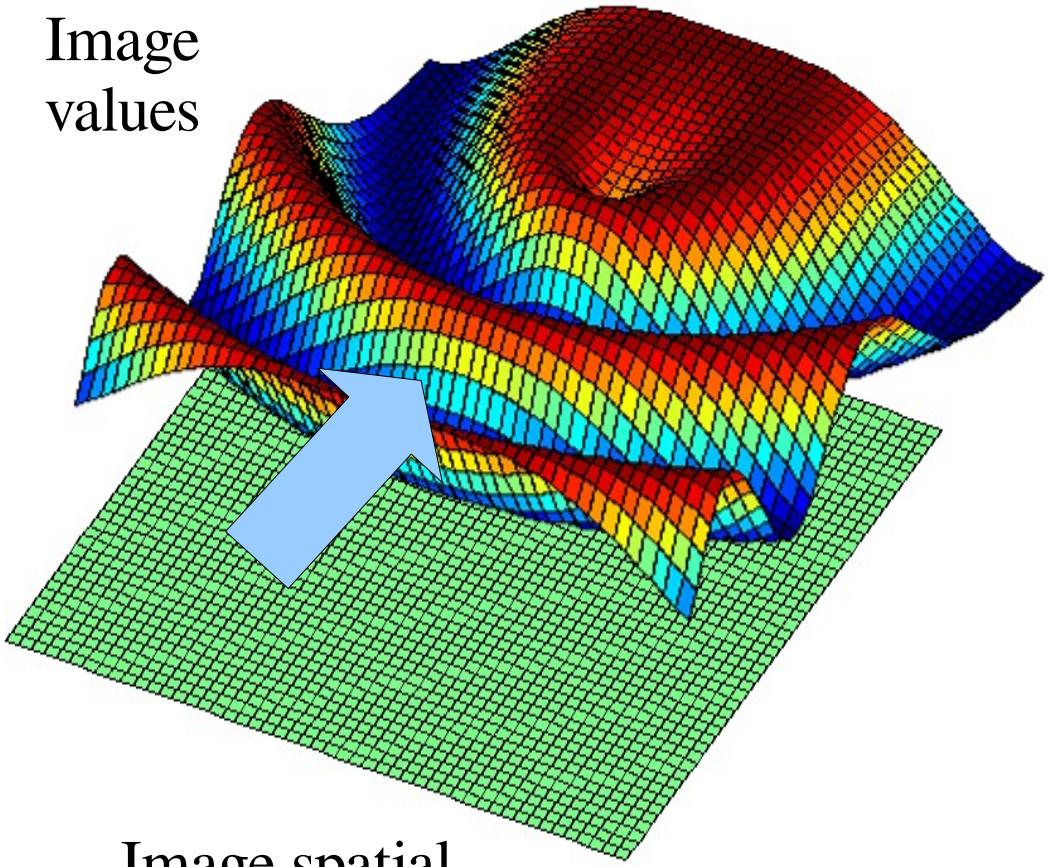


Filter block diagram

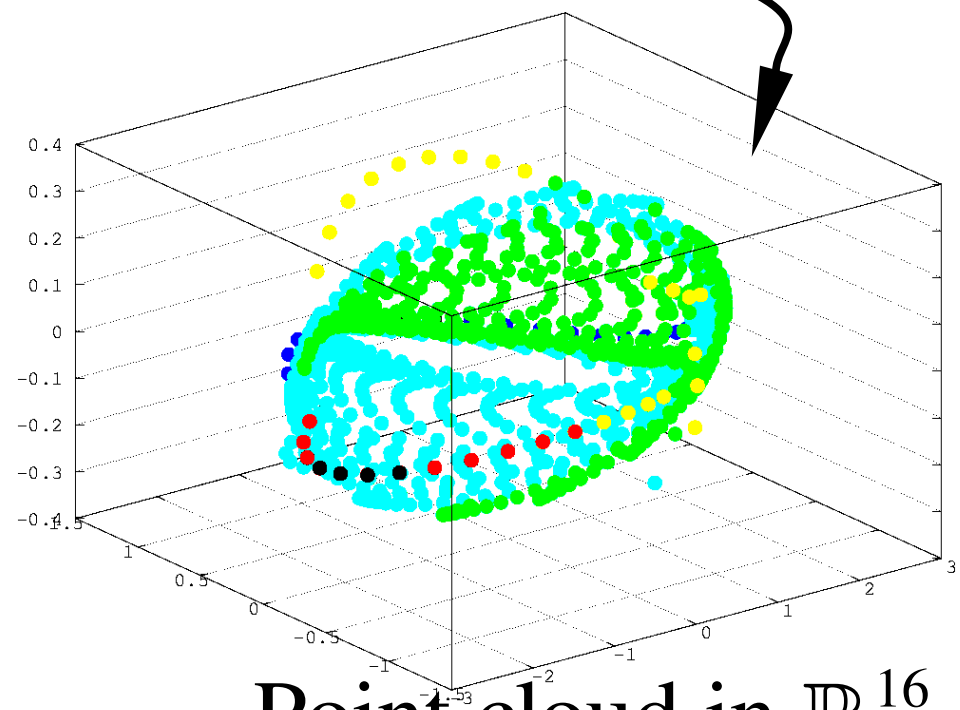


Michael Robinson

Stage 1: Topological estimation



Pixel neighborhood

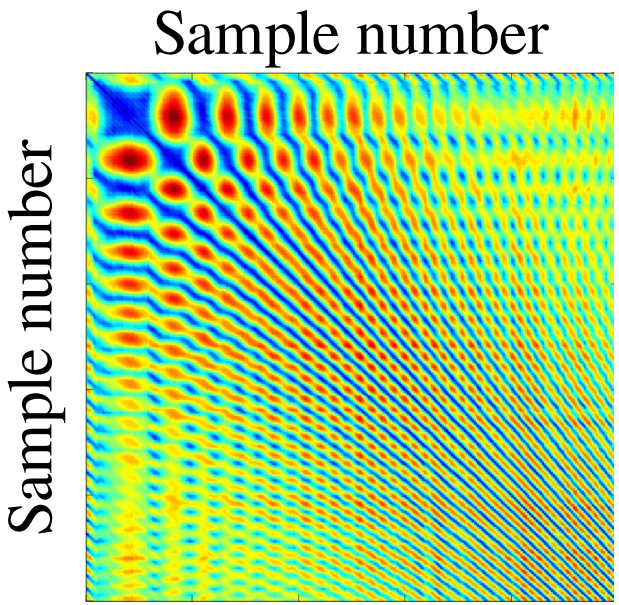


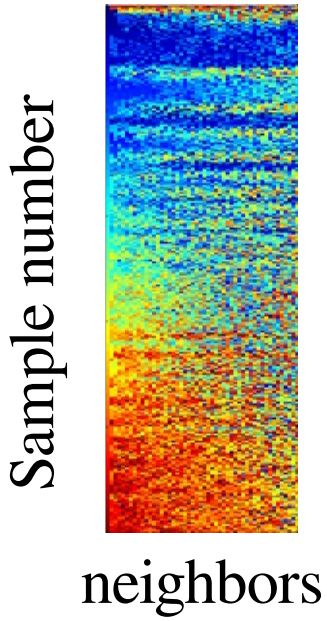
Point cloud in $\mathbb{R}^{16}$

Michael Robinson

Stage 2: Grouping sheaf

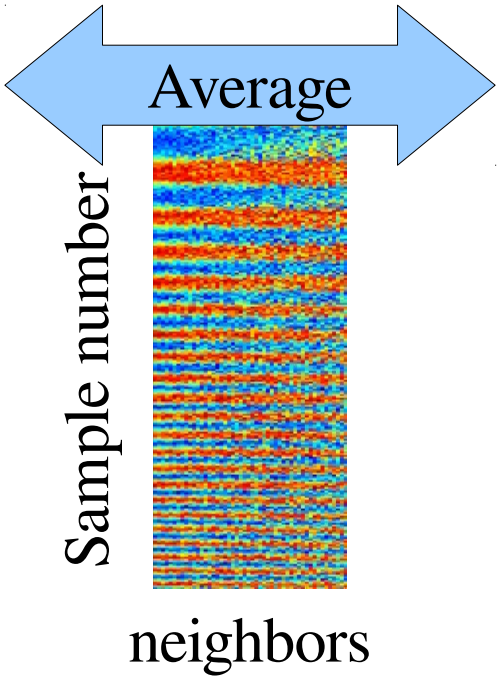
Distance matrix of point cloud





Locations of nearest neighbors

Values of the signal at the neighbors

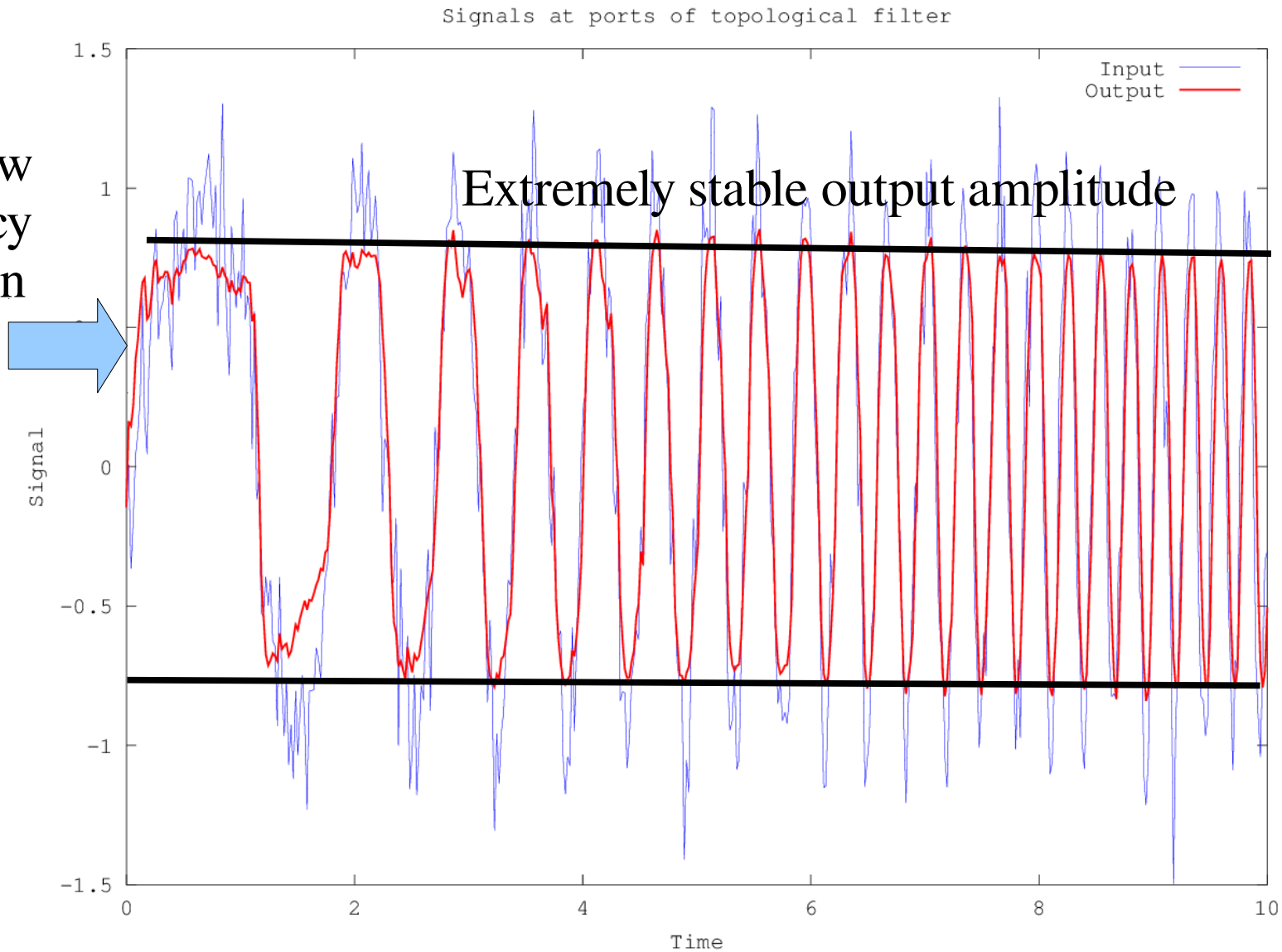


Process execution

Michael Robinson

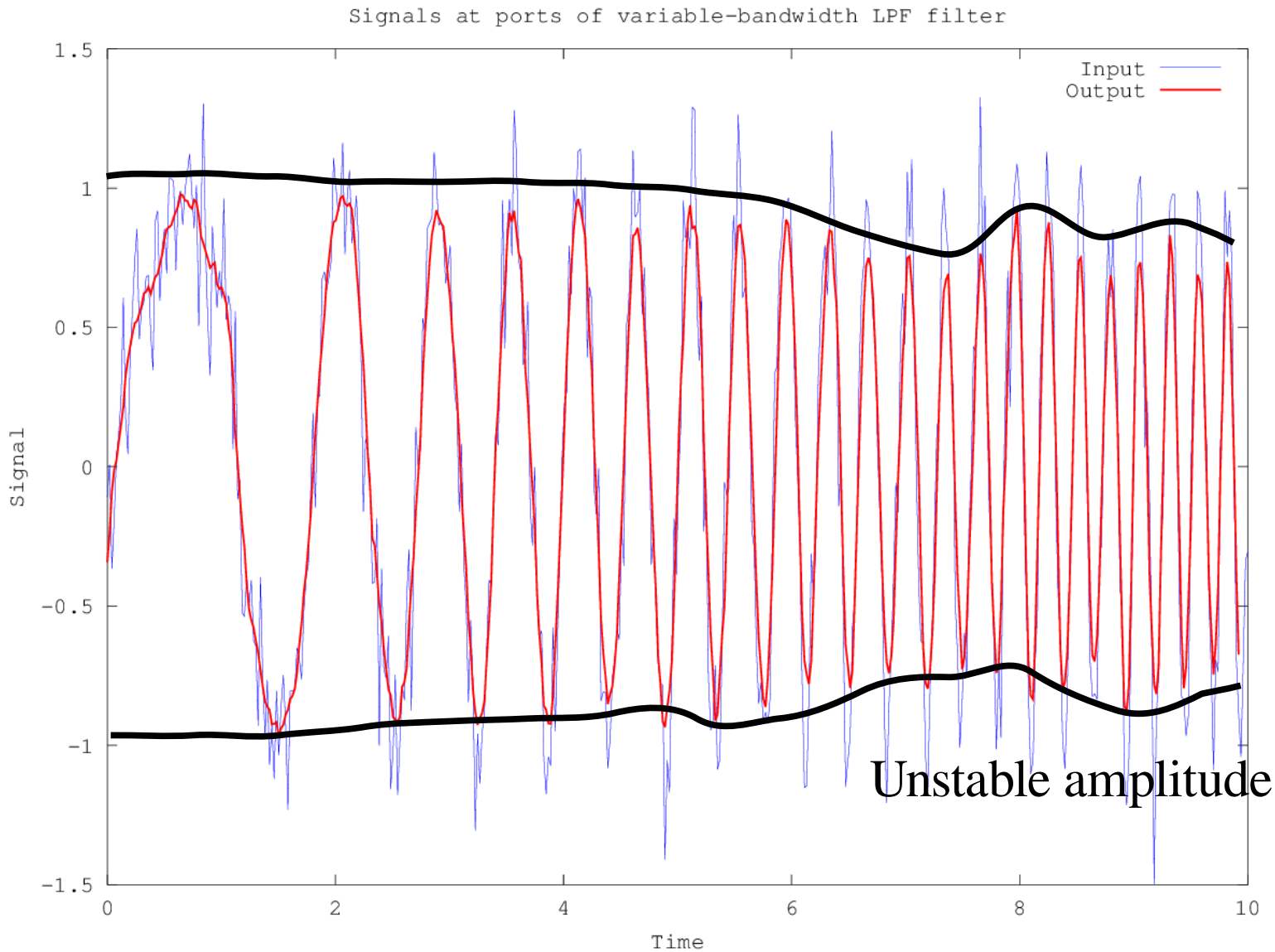
Topological filter results

Some low frequency distortion

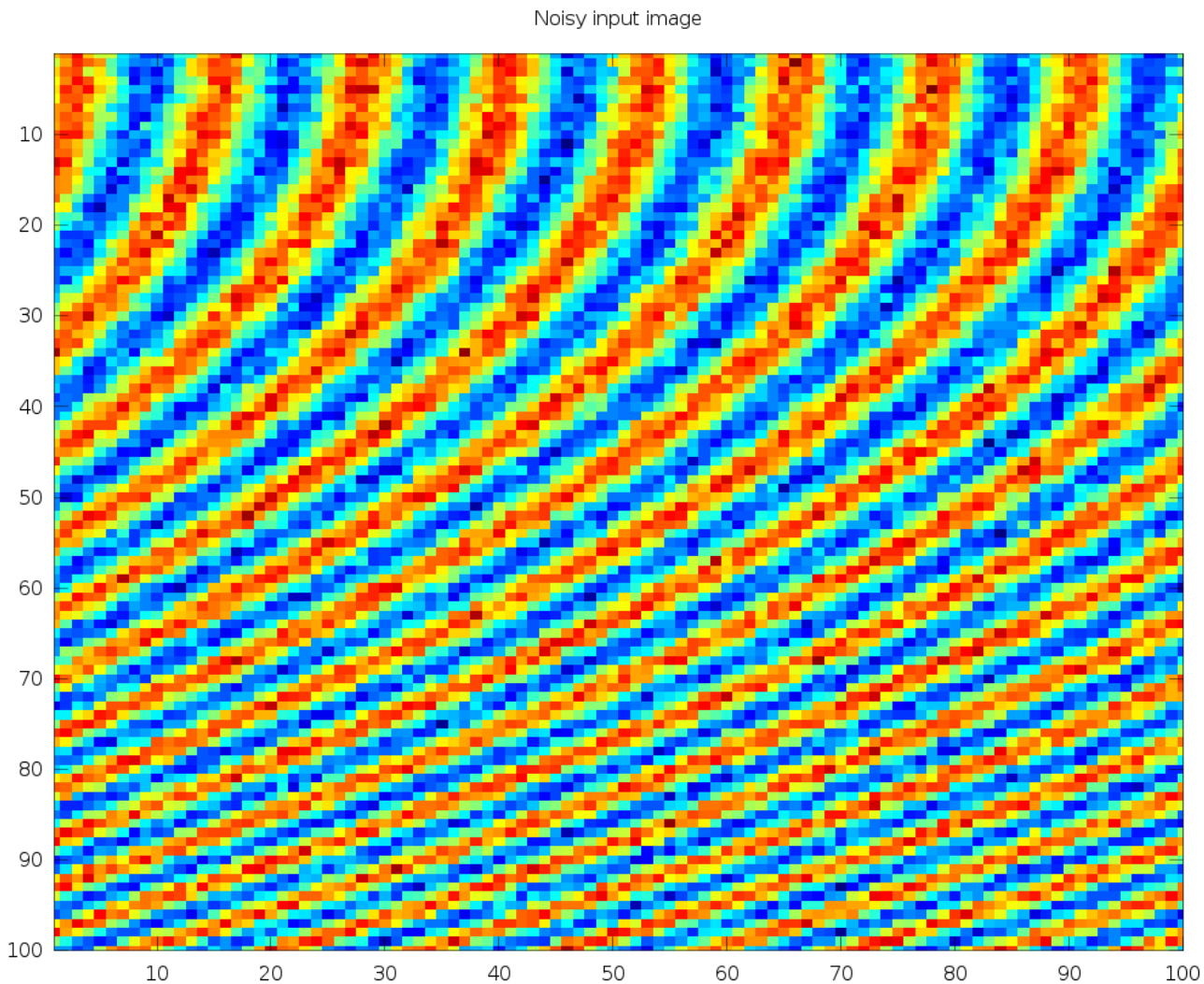


Michael Robinson

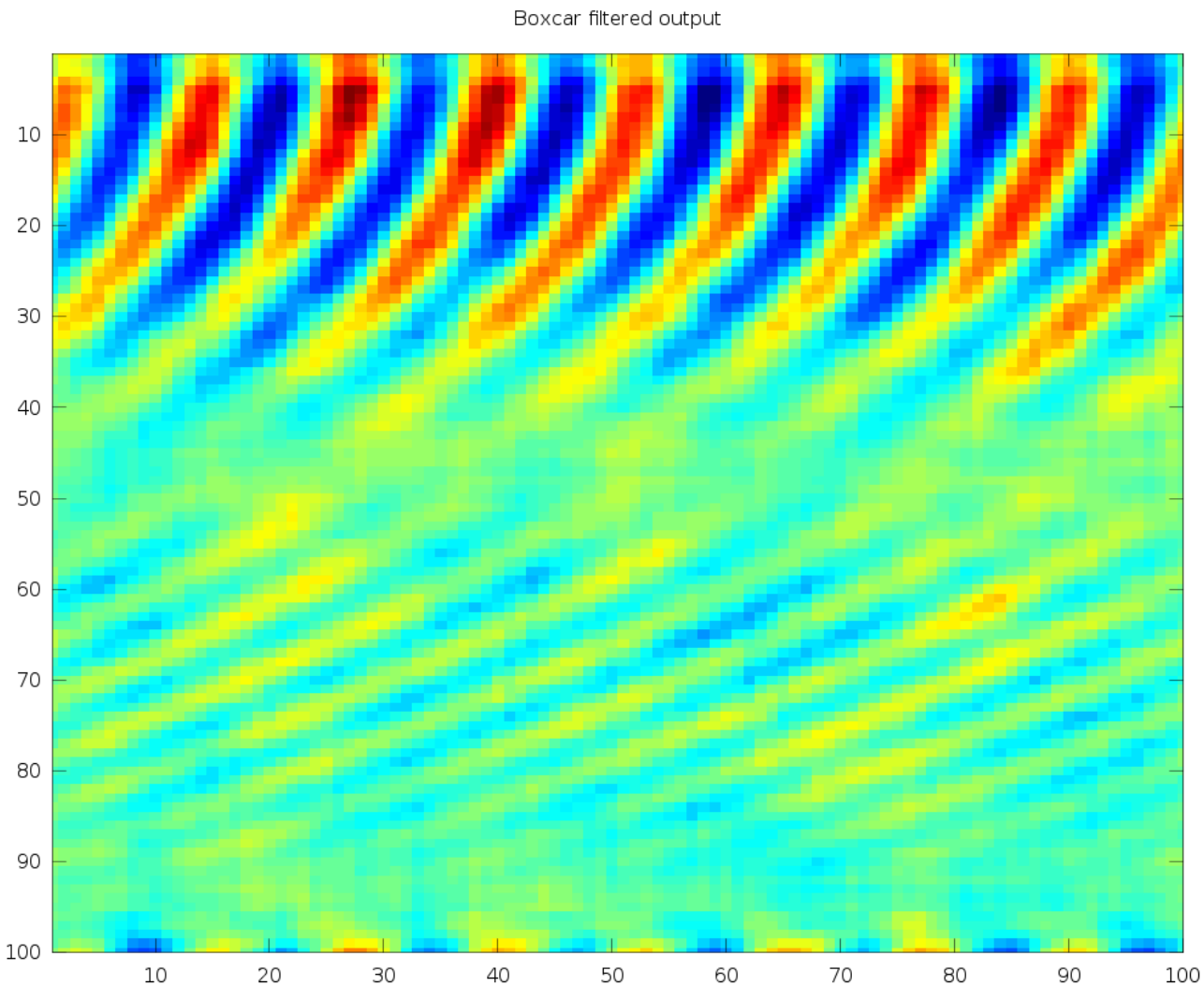
Compare: standard adaptive filter



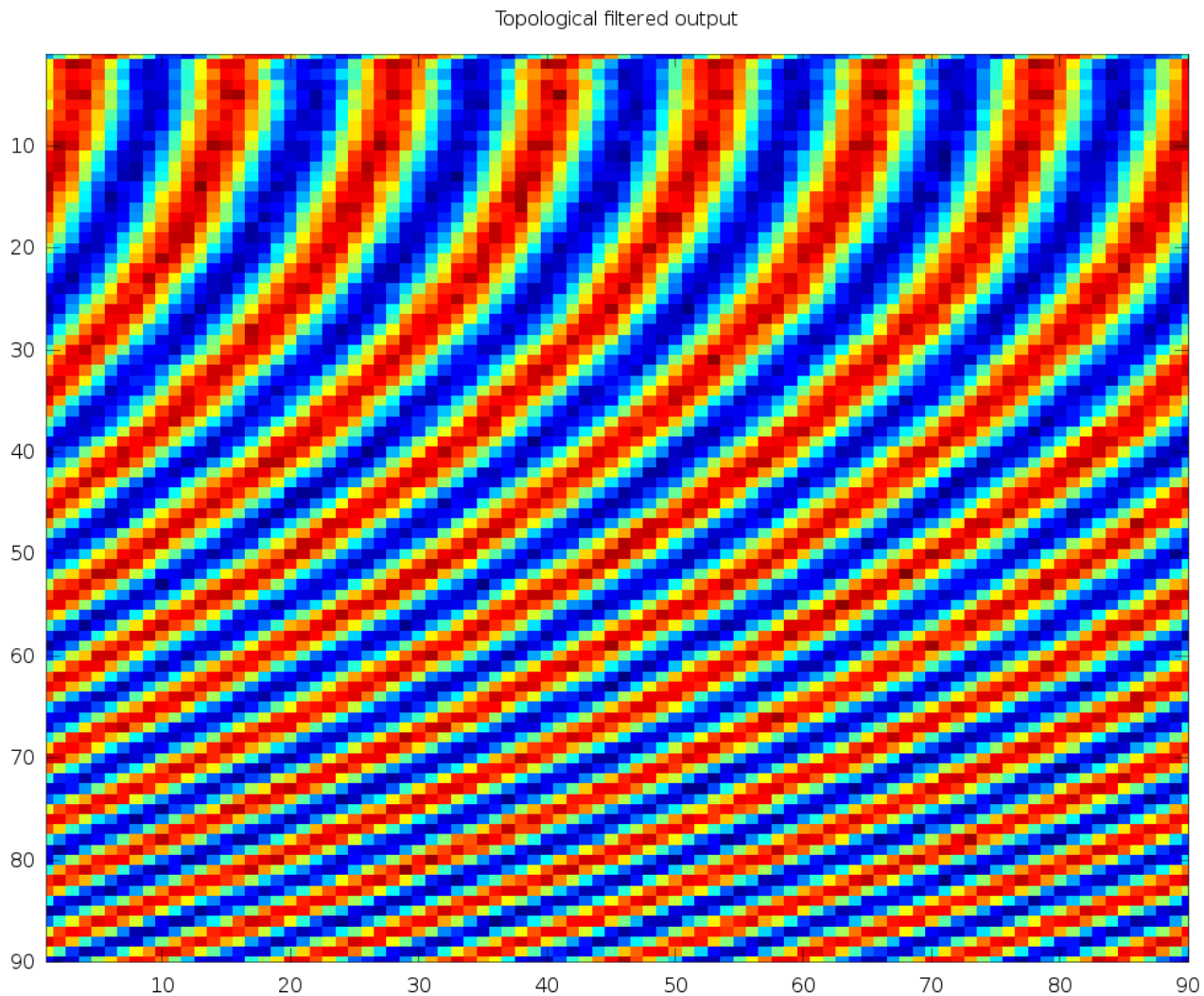
Input image


Noisy input image
10
20
30
40
50
60
70
80
90
100
10
20
30
40
50
60
70
80
90
100

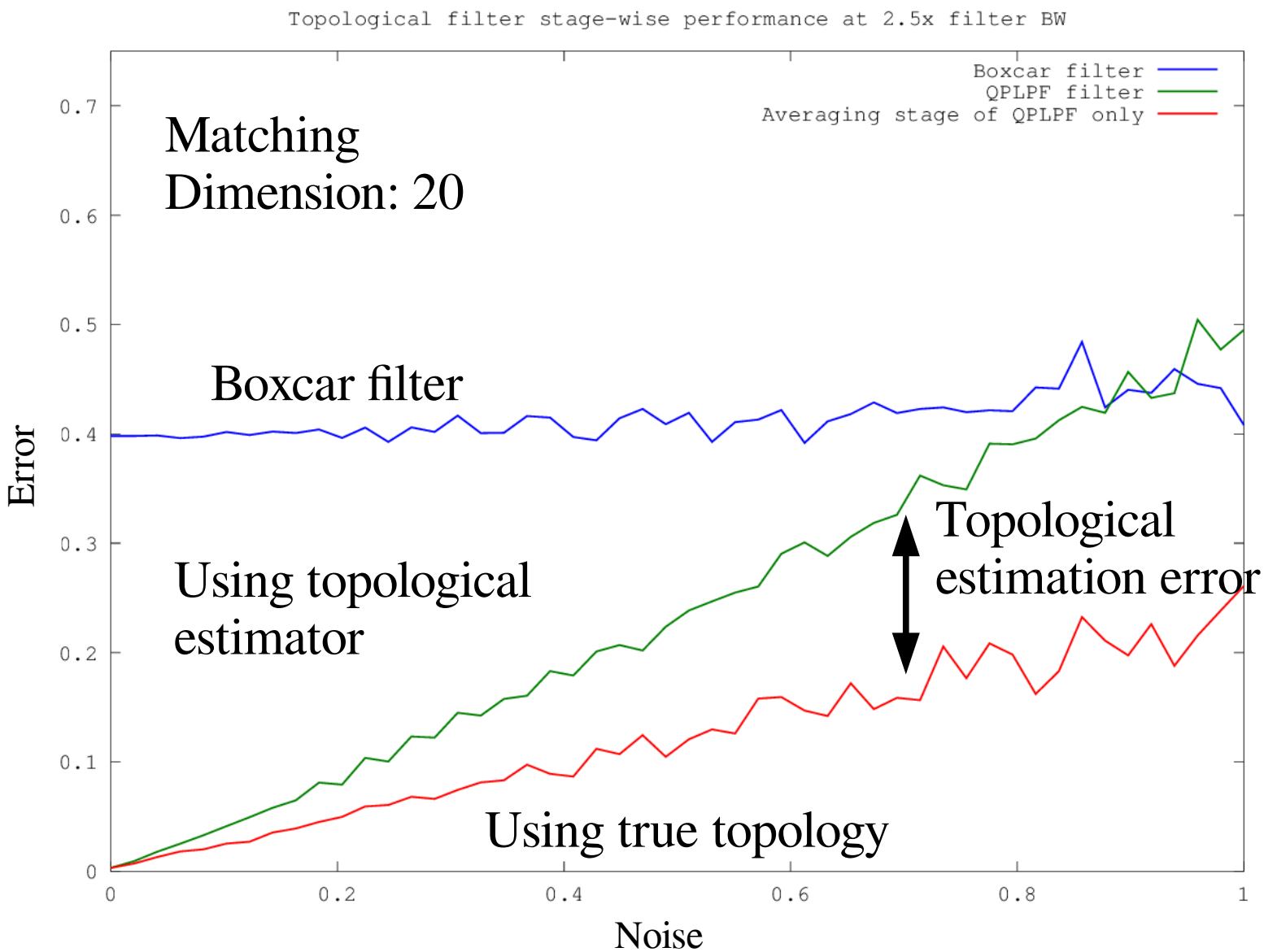
Fixed frequency image filter



Topological filter output

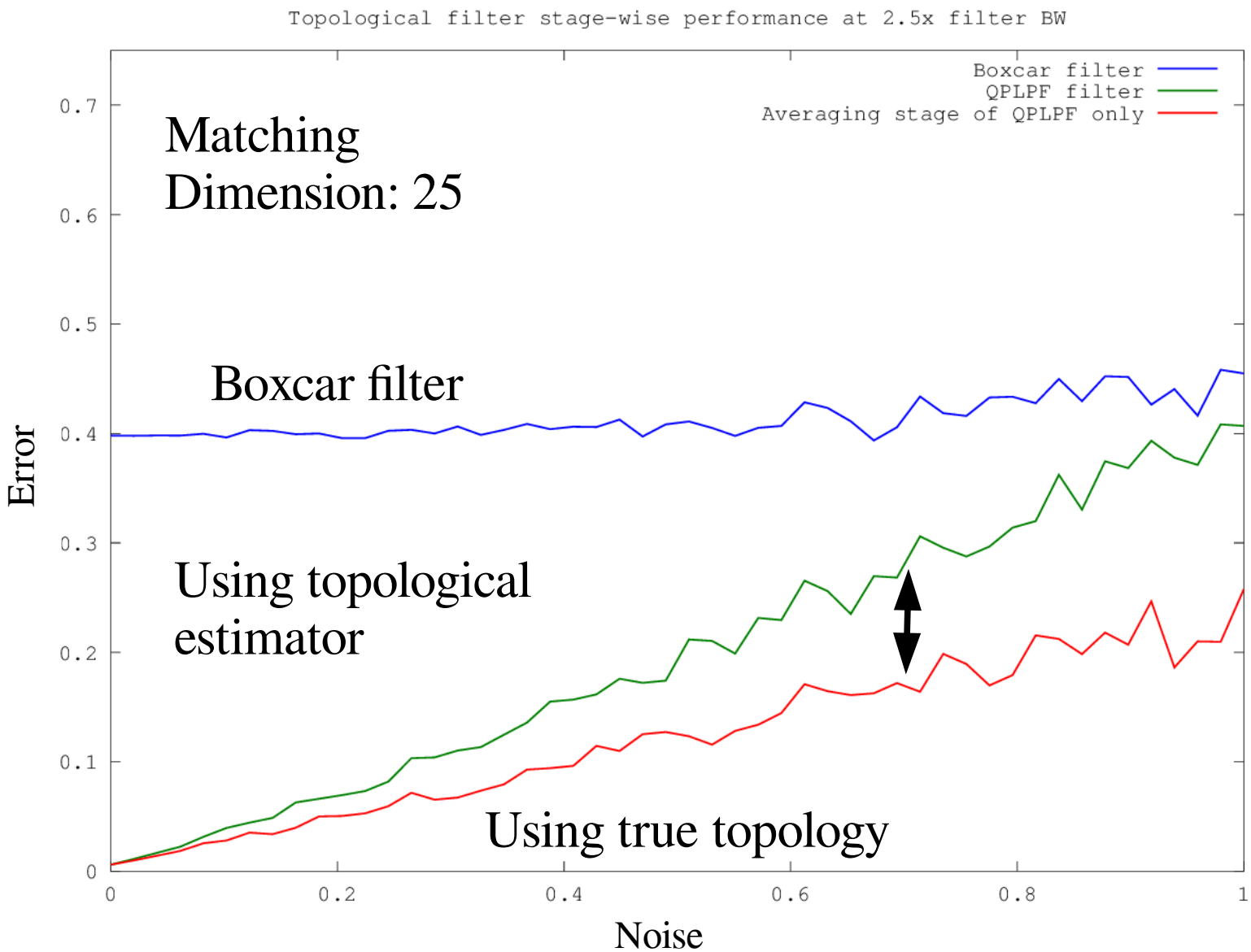


Error contributions



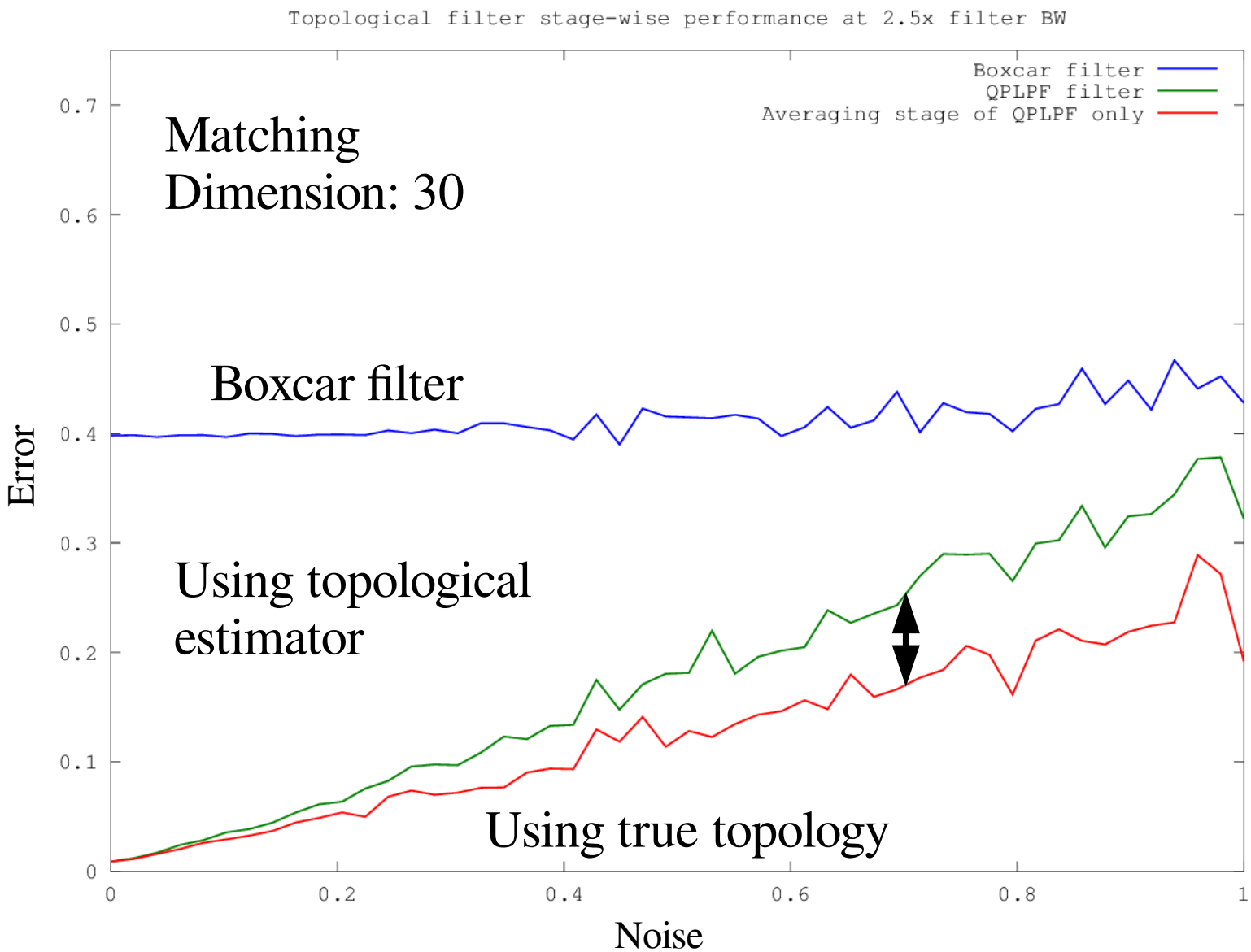
Michael Robinson

Error contributions



Michael Robinson

Error contributions



Michael Robinson

Next up...

- Informal social
- Next lecture (tomorrow morning): Categorification and Chain Complexes

Further reading...

- Sanjeevi Krishnan, "Flow-cut dualities for sheaves on graphs," http://arxiv.org/abs/1409.6712
- Robert Ghrist and Sanjeevi Krishnan, "A Topological Max-Flow Min-Cut Theorem," *Proceedings of Global Signals. Inf.*, (2013).
- Michael Robinson, "Understanding networks and their behaviors using sheaf theory," *IEEE Global Conference on Signal and Information Processing (GlobalSIP)* 2013, Austin, Texas.
- Michael Robinson, "The Nyquist theorem for cellular sheaves," *Sampling Theory and Applications 2013*, Bremen, Germany.