CAP representations
(The right(?) way for generic MR analysis of Internet data)

Amos Ron

University of Wisconsin* – Madison

WISP: UCSD, November 2004

*: Breaking news, 06/11/04: Wisconsin routed Minnesota 38:14, on its way to the national title.
Outline

• Possible goals behind “generic analysis on Internet signals”
• Why is that a non-trivial task?
• Predictability and pyramidal algorithms
• Performance of pyramidal representation
• CAMP and my favorite pyramidal representation
• What parameters to extract from the representation?
A mathematical view of Internet signals

• Main features in the signal:
  – burst types
  – rate of their appearances

• This is non-trivial
  (why? After all, nothing is easier than 1D signals...)
  – the amount of data may be overwhelming
  – there is no clear way to judge success

• It is also a cultural problem. really?
maybe it is time to show some images?
Pyramidal algorithms I: MR representation

$h : \mathbb{Z} \rightarrow \mathbb{R}$ is a symmetric, normalized, filter:
$h(k) = h(-k), \sum_{k \in \mathbb{Z}} h(k) = 1.$

\[\downarrow, \uparrow \text{ are downsampling & upsampling:}\]

\[y_\downarrow(k) = y(2k), \quad k \in \mathbb{Z}\]

\[y_\uparrow(k) = \begin{cases} 2y(k/2), & k \text{ even,} \\ 0, & \text{otherwise.} \end{cases}\]

\[(y_j)_{j=-\infty}^{\infty} \subset \mathbb{C}^{\mathbb{Z}} \text{ s.t:}\]

\[y_j = Cy_{j+1} := (h * y_j)_\downarrow, \quad \forall j.\]

*C is Compression or Coarsification*

\[y_{j+1}\] is then **predicted** from \(y_j\) by

\[y_{j+1} \approx Py_j := h * (y_j_\uparrow).\]

*P is Prediction or subdivision*
Pyramidal algorithms II: the detail coefficients

- Define the detail coefficients:

\[ d_j := (I - PC) y_j = y_j - P y_{j-1}. \]

- Replace \( y_j \) by the pair \( (y_{j-1}, d_j) \).

- Continue iteratively.

**Reconstruction.** Recovering \( y_m \) from \( y_0, d_1, d_2, \ldots, d_m \) is trivial:

\[ y_1 = d_1 + P y_0, \quad y_2 = d_2 + P y_1 \] and so on.
Wavelet pyramids, Mallat, 1987

Decompose the detail map $I - PC$:

$$I - PC = RD$$

$$D : y \mapsto (h_1 * y)_\downarrow =: w_{1,j-1}, \quad R : y \mapsto h_1 * y_\uparrow$$

with $h_1$ a real, symmetric, highpass: $\sum_{k \in \mathbb{Z}} h_1(k) = 0$.

Note that we can recover $y_m$ from $y_0, w_{1,0}, w_{1,1}, \ldots, w_{1,m-1}$ since

$$y_1 = Rw_{1,0} + Py_0, \quad y_2 = Rw_{1,1} + Py_1$$

and so on.
Performance

• Ability to predict. The best prediction are based on local averaging, and on nothing else = spline predictors

• Time blurring: good prediction requires long averaging filter. That blurs spontaneous events.

• Internet data exhibit different behaviour at “small” scales than other scales. Hence: non-stationary representation

• Standard wavelet systems are mediocre for Internet data: they blur time, and create artifacts, in order to gain unnecessary properties (orthonormality).
Poor prediction
My favorite representation

well, before we conducted any numerical tests

**Step I:** Build an MR representation based on

\[ h_1 = \frac{1}{4}(1 \ 2 \ 1) \]

**Step II:** Define the detail coefficients by:

\[
d_j(k) = \begin{cases} 
\frac{-y_j(k+1)+2y_j(k)-y_j(k-1)}{4}, & k \text{ even,} \\
\frac{y_j(k-3)-9y_j(k-1)+16y_j(k)-9y_j(k+1)-y_j(k+3)}{16}, & k \text{ odd.}
\end{cases}
\]

The “performance grade” here is 2 in the strict sense. (To compare, Haar’s grade is 1 in the non-strict sense, and 0 in the strict sense.)

This is an example of a new class of high-performance representations called **CAMP**
what to analyse? what to extract?

for \( p \geq 1 \), the \( p \)-norm is

\[
\|a\|_p = \left( \sum_k |a_k|^p \right)^{1/p}
\]

the best thing to analyse is the “compressibility” of the detail coefficients: choose a number \( N \), then

(1) replace the \( N \) “most important” detail coefficients by 0, to obtain a signal \( e_N \).

(2) reconstruct using \( e_N \) to obtain \( Y_N \).

(3) define \( e_p(N) := \|Y_N\|_p \).

(4) find the a parameter \( \alpha \) such that

\[
e_p(N) \approx CN^{-\alpha}.
\]
\[ \alpha(p) = \text{the predictability parameter in the } p\text{-norm} \]

“most important” = ?

(1) **Non-linear**: choose the largest ones

(2) **Linear**: go from coarse scale to fine scale.

Output: this way we have two functions \( p \mapsto \alpha(p) \).

**Goal**: learn how to judge properties of your signal based on these two functions

it might be that the detail coefficients behave rather differently at different scale (small scale vs. large scale).
CAP representations

Choose:

- two refinable functions $\phi_c, \phi_r$ with refinement filters $h_c, h_r$.
- A third (Auxiliary-Alignment) lowpass filter $h_a$.

Decompose: Fix $f : \mathbb{R} \rightarrow \mathbb{C}$.

For all $k, j \in \mathbb{Z}$, define $y_j(k) := 2^{j/2} \langle f, (\phi_c)_{j,k} \rangle$.

The CAP operators are:

- $C : y \mapsto (h_c \ast y)_{\downarrow}$, (Coarsification-Compression),
- $A : y \mapsto Ay := h_a \ast y$, (Alignment),
- $P : y \mapsto Py := h_r \ast (y_{\uparrow})$, (Prediction-subdivision).

Then $Cy_{j+1} = y_j, \forall j$. 
The detail coefficients are:

\[ d_j := (A - PAC)y_j = Ay_j - PACy_{j-1}. \]

This is the CAP representation with \((d_j)\) the CAP coefficients.

\[
\begin{array}{cccccc}
  y_m & \overset{C}{\rightarrow} & y_{m-1} & \overset{C}{\rightarrow} & y_{m-2} & \cdots \cdots & y_1 & \overset{C}{\rightarrow} & y_0 \\
  \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\
  d_m & & d_{m-1} & & d_{m-2} & & \cdots & & d_1 \\
\end{array}
\]

\(y_m\) is recovered from \(y_0, d_1, d_2, \ldots, d_m\) since

\[ Ay_1 = d_1 + PACy_0, \quad Ay_2 = d_2 + PACy_1, \quad \ldots, \quad Ay_m = d_m + PACy_{m-1} \]

and deconvolving \(A\) from \(Ay_m\).
Summary

Do they

<table>
<thead>
<tr>
<th></th>
<th>W</th>
<th>F</th>
<th>CAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>implemented by fast pyramid algorithms?</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>provides good function space characterizations?</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>avoid mother wavelets?</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>very short filters, with no artifacts?</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>have simple constructions?</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>avoid redundant representations?</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Wavelets are non-redundant. Caplets are only slightly redundant in high dimensions. Their redundancy is non-essential.
CAMP representations: Compression-Alignment-Modified Prediction

With CAP in hand, one can modify the process s.t.:

- The filters are shorter
- The performance (:= function space characterization) is the same

Example: Assume $h$ is interpolatory. Define the details as:

$$d_j := \begin{cases} y_j - h \ast y_j, & \text{on } 2\mathbb{Z}^d, \\ y_j - h \ast (y_j \downarrow \uparrow), & \text{otherwise}. \end{cases}$$

Let $\phi$ be the refinable function of $h$. If

$$\phi \in C_c^{s+\epsilon},$$

then the above detail characterize $L_p^s$. 

19
Example (2D): $h = \begin{bmatrix} 0 & 1/8 & 1/8 \\ 1/8 & 1/4 & 1/8 \\ 1/8 & 1/8 & 0 \end{bmatrix}$.

There are four (hidden) filters, for computing $d_j$:

$$\begin{bmatrix} 0 & -1/8 & -1/8 \\ -1/8 & +3/4 & -1/8 \\ -1/8 & -1/8 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & 0 & 0 \\ -1/2 & +1 & -1/2 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1/2 & 0 \\ 0 & +1 & 0 \\ 0 & -1/2 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & 0 & -1/2 \\ 0 & +1 & 0 \\ -1/2 & 0 & 0 \end{bmatrix}$$

Those are 7, 3, 3, 3-tap. There are four (hidden) “CAMPlets”, whose average area of support is about 2.

The performance is on par with tensor 3/5, whose filters are 25, 15, 15-tap. Each supported in $3 \times 3$ square.
Figure 1: First level $\tilde{d}$ CAMP coefficients, organized by cosets.
Wisconsin

From left, 1st row:
Julia Velikina, Youngmi Hur, Yeon Kim, Narfi Stefansson.

2nd row:
Thomas Hangelbroek, Sangnam Nam, Jeff Kline, Steven Parker.
Julia Velikina: undersampled MRI data

Schepp–Logan phantom

Conventional recon. from 90 projections, acceptable quality

Conventional recon. from 23 projections, unacceptable quality

TV–based recon. from 23 projections
Jeff Kline: new data representation in NMR

Approximate recovery of
and from

+ noise =
Adaptive framelet-based representation of a vibraphone recording

original signal (time)
Narfi Stefansson: sparse framelet representations
6/10  61440 coefficients

cubic spline  34608 coefficients

quartic spline  34452 coefficients

box15,box17,box18  35085 coefficients
Welcome to The IDR FrameNet Portal, a web-based, research and educational tool for time/frequency analysis of data. If you are new to this site, we encourage you to take the tour or visit the Site Help.

This tool provides facilities for uploading and management of scientific data, as well as dozens of available data sets from a variety of sources. Time/frequency analysis can be performed by classic wavelet systems, as well as by framelet systems (giving a redundant time/frequency description.) Furthermore, the FrameNet provides a collaboration mode, allowing researchers to work together on projects, and educators to demonstrate framelet analysis to their classes.

Group Leader: Amos Ron
Development Team Leader: Steven Parker
Development Team: Thomas Hangelbroek, Youngmi Hur, Jeff Kline, Narfi Stefansson, Bee-Chung Chen with contributions from Carl de Boor, Miron Livny, Kent Wenger and Remi Gribonval.

This site is a project of the Wavelet Center for Ideal Data Representation. It incorporates the Dervise data exploration system and the LastWave signal processing software. Web hosting is maintained by Computer Systems Lab of the Computer Sciences Department, University of Wisconsin - Madison.

To contact the FrameNet team, send mail to framenet@waveletidr.org.
The IDR FrameNet Portal

Manage Data  65 data are shown

Create an alias to each selected item (data or alias) in the table.
Copy the data source for each selected item (data or alias) in the table.
Edit the metadata for each selected item (data or alias) in the table.
Move each selected item (data or alias) to a different collection.
Delete each selected item (data or alias) in the table.

<table>
<thead>
<tr>
<th>select</th>
<th>collection</th>
<th>var 1</th>
<th>var 2</th>
<th>var 3</th>
<th>var 4</th>
<th>var 5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>09.20</td>
<td>out</td>
<td>bytes</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>09.20</td>
<td>out</td>
<td>flows</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>09.20</td>
<td>out</td>
<td>psts</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.07</td>
<td>out</td>
<td>bytes</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.07</td>
<td>out</td>
<td>flows</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.07</td>
<td>out</td>
<td>psts</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.14</td>
<td>out</td>
<td>bytes</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.14</td>
<td>out</td>
<td>flows</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.14</td>
<td>out</td>
<td>psts</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.21</td>
<td>out</td>
<td>bytes</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.21</td>
<td>out</td>
<td>flows</td>
<td>Info</td>
<td>View</td>
</tr>
<tr>
<td></td>
<td>framenet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Internet 1</td>
<td>10.21</td>
<td>out</td>
<td>psts</td>
<td>Info</td>
<td>View</td>
</tr>
</tbody>
</table>

---

32