Empirical Mode Decomposition for Trend Extraction. Application to Electrical Data

Jean-Michel Poggi
Laboratoire de Mathématique, Université d’Orsay, France

Joint work with Farouk Mhamdi, Mériem Jaïdane-Saïdane
U2S, ENIT, Tunis, Tunisia
Outline

• Motivation: Blind Trend Extraction
• Empirical Mode Decomposition (EMD)
• EMD trend vs. Hodrick Prescott (HP) trend
• Simulated seasonal series
• Tunisian daily peak load 2000-2006
• EMD vs. HP, wavelet trends

This work was supported by 2005/2009 VRR research project between ENIT and Tunisian Society of Electricity and Gas (STEG)
**Trend?**

- **Trend** = some “smooth additive component that contain information about global change” *Alexandrov et al. (2009)*
- **The problem:** to *extract trends from seasonal time series* without strong modeling and global estimation

- **Ex:** long term electricity load or airline traffic forecasting

---

Seasonal time series intrinsic trend

- **Additive time series model:**
  \[ Y(t) = T(t) + SC(t) + e(t) \]
  i.e. trend + seasonal-cycles + error

- Several methods are used for time series components extraction including
  - local or global regressions
  - MA filtering, X11, X12
  - Hodrick Prescott filter
  see *Alexandrov et al. (2009)* for a recent review

- The idea: to *directly extract trend without identifying the other components* of the observed signal
EMD? EMD and trend?

- EMD transformation is **nonlinear** and suitable for **non-stationary signals** *Huang et al. (2004)*
- Signal is decomposed as a superposition of local sums of oscillatory components called **Intrinsic Mode Functions (IMF)** of different time scales intrinsic to the signal

\[
Y(t) = \sum IMF_k(t) + r(t)
\]

with **IMF** = function with zero mean and having as many zero crossings as maxima or minima

and **r** = a monotone function

- IMFs are **fully data-driven** and local in time
- After the IMF extraction process (sifting), it remains **r(t) a monotone function candidate to be an estimate of T(t) since it is a trend** free of oscillatory components
EMD extraction algorithm

Residue = s(t), I_1(t) = Residue

i = 1, k = 1

while Residue not equal zero or not monotone

while I_i has non-negligible local mean

U(t) = spline through local maxima of I_i
L(t) = spline through local minima of I_i
Av(t) = 1/2 (U(t) + L(t))

I_i(t) = I_i(t) - Av(t), i = i + 1
end

IMF_k(t) = I_i(t)

Residue = Residue - IMF_k
k = k + 1
end

Credit:
Suz Tolwinski
University of Arizona
Program in Applied Mathematics
Spring 2007 RTG
EMD acting on an example

Analyzed signal = \textit{tone} + \textit{chirp}

Credit: Rilling and Flandrin
from http://perso.ens-lyon.fr/patrick.flandrin/emd.html
Residue = s(t)
I_1(t) = Residue
i = 1
k = 1

while Residue not equal zero or not monotone
    while I_i has non-negligible local mean
        U(t) = spline through local maxima of I_i
        L(t) = spline through local minima of I_i
        Av(t) = 1/2 (U(t) + L(t))
        I_i(t) = I_i(t) - Av(t)
        i = i + 1
    end
IMF_k(t) = I_i(t)
Residue = Residue - IMF_k
k = k+1
end
EMD acting on an example

*Extract local maxima*

Residue = s(t)

\[ I_1(t) = \text{Residue} \]

i = 1

k = 1

while Residue not equal zero or not monotone

\[ I_i(t) = \text{Residue} \]

while \( I_i \) has non-negligible local mean

\[ U(t) = \text{spline through local maxima of } I_i \]

\[ L(t) = \text{spline through local minima of } I_i \]

\[ Av(t) = \frac{1}{2} (U(t) + L(t)) \]

\[ I_i(t) = I_i(t) - Av(t) \]

i = i + 1

end

\[ \text{IMF}_k(t) = I_i(t) \]

Residue = Residue - \( \text{IMF}_k \)

k = k + 1

end
EMD acting on an example

Maxima envelope by interpolation

Residue = s(t)
I_1(t) = Residue
i = 1
k = 1

while Residue not equal zero or not monotone
  while I_i has non-negligible local mean
    U(t) = spline through local maxima of I_i
    L(t) = spline through local minima of I_i
    Av(t) = 1/2 (U(t) + L(t))
    I_i(t) = I_i(t) - Av(t)
    i = i + 1
  end
IMF_k(t) = I_i(t)
Residue = Residue - IMF_k
k = k+1
end
EMD acting on an example

**Extract local minima**

Residue = s(t)

\( I_1(t) = \text{Residue} \)

i = 1

k = 1

while Residue not equal zero or not monotone

while \( I_i \) has non-negligible local mean

U(t) = spline through local maxima of \( I_i \)

L(t) = spline through local minima of \( I_i \)

Av(t) = 1/2 \((U(t) + L(t))\)

\( I_i(t) = I_i(t) - Av(t) \)

i = i + 1

end

IMF_k(t) = \( I_i(t) \)

Residue = Residue - IMF_k

k = k+1

end
EMD acting on an example

Minima envelope by interpolation

Residue = s(t)
I_1(t) = Residue
i = 1
k = 1

while Residue not equal zero or not monotone
  while I_i has non-negligible local mean
    U(t) = spline through local maxima of I_i
    L(t) = spline through local minima of I_i
    Av(t) = 1/2 (U(t) + L(t))
    I_i(t) = I_i(t) - Av(t)
    i = i + 1
  end
IMF_k(t) = I_i(t)
Residue = Residue - IMF_k
k = k+1
end
EMD acting on an example

**Mean of maxima and minima envelopes**

Residue = s(t)
I_1(t) = Residue
i = 1
k = 1

**while** Residue not equal zero or not monotone
**while** I_i has non-negligible local mean
U(t) = spline through local maxima of I_i
L(t) = spline through local minima of I_i
Av(t) = 1/2 \((U(t) + L(t))\)
I_i(t) = I_i(t) - Av(t)
i = i + 1

**end**
IMF_k(t) = I_i(t)
Residue = Residue - IMF_k
k = k+1

**end**

Local low frequency component
EMD acting on an example

**Subtract mean** envelope from signal

Residue = \( s(t) \)
\[ I_1(t) = \text{Residue} \]
\( i = 1 \)
\( k = 1 \)

**while Residue not equal zero or not monotone**

**while** \( I_i \) has non-negligible local mean

\[ U(t) = \text{spline through local maxima of } I_i \]
\[ L(t) = \text{spline through local minima of } I_i \]
\[ \text{Av}(t) = \frac{1}{2}(U(t) + L(t)) \]
\[ I_i(t) = I_i(t) - \text{Av}(t) \ \text{("residue"-->)} \]
\( i = i + 1 \)

**end**

\[ \text{IMF}_k(t) = I_i(t) \]

Residue = Residue - \( \text{IMF}_k \)
\( k = k+1 \)

**end**
EMD acting on an example

Is the residue a IMF? No

Residue = s(t)
l_1(t) = Residue
i = 1
k = 1

while Residue not equal zero or not monotone
  while l_i has non-negligible local mean
    U(t) = spline through local maxima of l_i
    L(t) = spline through local minima of l_i
    Av(t) = 1/2 (U(t) + L(t))
l_i(t) = l_i(t) - Av(t) ("residue" -->
    i = i + 1
  end
IMF_k(t) = l_i(t)
Residue = Residue - IMF_k
k = k + 1
end
EMD acting on an example

No, so iterate the loop (sifting)

Residue = s(t)
I_1(t) = Residue
i = 1
k = 1

while Residue not equal zero or not monotone
  while I_i has non-negligible local mean
    U(t) = spline through local maxima of I_i
    L(t) = spline through local minima of I_i
    Av(t) = 1/2 (U(t) + L(t))
    I_i(t) = I_i(t) - Av(t)
    i = i + 1
  end
IMF_k(t) = I_i(t)
Residue = Residue - IMF_k
k = k+1
end
EMD acting on an example
IMF1 - iteration 1 - maxima

EMD for trend extraction
EMD acting on an example
IMF1 - iteration 1 - minima

EMD for trend extraction
EMD acting on an example
IMF1 - iteration 2 – mean

EMD for trend extraction
EMD acting on an example
IMF1 - iteration 3

EMD for trend extraction
EMD acting on an example

IMF1 - iteration 4

EMD for trend extraction
EMD acting on an example

IMF1 - iteration 5

EMD for trend extraction
EMD acting on an example
IMF1 - iteration 6

EMD for trend extraction
EMD acting on an example

IMF1 - iteration 7

EMD for trend extraction
EMD acting on an example

IMF1 - iteration 8

Residue = s(t)
\( I_1(t) = \text{Residue} \)
i = 1
k = 1

while Residue not equal zero or not monotone
  
  while \( I_i \) has non-negligible local mean
    
    \( U(t) = \text{spline through local maxima of } I_i \)
    
    \( L(t) = \text{spline through local minima of } I_i \)
    
    \( \text{Av}(t) = 1/2 \ (U(t) + L(t)) \)
    
    \( I_i(t) = I_i(t) - \text{Av}(t) \)
    
    i = i + 1
  end

  \( \text{IMF}_k(t) = I_i(t) \)
  
  Residue = Residue - \( \text{IMF}_k \)
  
  k = k+1
end

EMD for trend extraction
EMD acting on an example

**IMF2 - iteration 0**

Residue = s(t)
I_1(t) = Residue
i = 1
k = 1

while Residue not equal zero or not monotone
    while I_i has non-negligible local mean
        U(t) = spline through local maxima of I_i
        L(t) = spline through local minima of I_i
        Av(t) = 1/2 (U(t) + L(t))
        I_i(t) = I_i(t) - Av(t)
        i = i + 1
    end

IMF_k(t) = I_i(t)
Residue = Residue - IMF_k
k = k+1
end
EMD acting on an example

IMF2 – iteration 1

EMD for trend extraction
EMD acting on an example

IMF2 – iteration 2

EMD for trend extraction
EMD acting on an example

IMF2 – iteration 3

EMD for trend extraction
EMD acting on an example

IMF2 – iteration 4

EMD for trend extraction
EMD acting on an example
IMF2 – iteration 5

EMD for trend extraction
Trend definitions and EMD

• No consensus about what is a trend, various definitions related to data peculiarities and fields of application.

• EMD already used to extract trends:
  – *Flandrin et al. 2004*, sum of nonzero mean IMFs.
  – *Zhou et al. 2008*, power-system oscillation data.

• *This paper*: long-term trend for seasonal time series and comparison with *Hodrick Prescott (HP)* and a remark about *wavelets*.

• Three weeks ago:

  *The idea*: select low frequency IMFs from coarse to fine and when it does not differ from some noise reference.

EMD for trend extraction
Hodrick-Prescott filtering

- Comparison with the nonparametric trend extraction method based on HP filtering
- Widely used by economists, *Pollock (2003)*
  \[ \text{Trend} = \arg\min \{ (y - T)^2 + \lambda \nabla^2 (T) \} \]
- Penalized least squares estimation
- Usual values for $\lambda$ in economic time series:
  - Quarterly UK Gas data $\lambda = 1600$
  - Monthly Airline data $\lambda = 14400$

HP filter / simulated seasonal time series

- To select $\lambda$ automatically, see *Schlicht (2005)*
- Here, we prefer to use Empirical tuning based on *simulated load curve* for $\lambda \in [10^2, 10^{15}]$, a bootstrap-like scheme

- one week simulated daily load

- one year simulated daily peak load
- and additive trend (linear or exp.)

EMD for trend extraction
HP vs EMD and suitable $\lambda$

• *Calibration of HP parameter* $\lambda$ for different kinds of artificial trends for daily data

• *Comparison* of HP and EMD trends for different *linear*, *quasi linear* and *exponential* trends

• EMD-trend very close to the optimal HP one
  $\Rightarrow$ EMD is an effective alternative for trend extraction

• Then we can use these values to compare *EMD-trend* and “optimal” *HP one* on real electrical daily data
Tunisian daily peak load 2000-2006

Daily peak load

IMFs + final trend

EMD for trend extraction
IMFs / seasonal load components

**Sum of IMFs 1-2**
Short term week component

**Sum of IMFs 6-7-8**
Long term annual component

EMD for trend extraction
Tunisian daily peak load
HP and EMD long-term trends

EMD for trend extraction
Tunisian daily peak load
**HP and EMD trends: end effects**

Dealing with EMD end effects:
- **Windowing the signal**, *Ren et al. (2006)*
- **Extrapolate** end maxima and minima, *Zhaohua et al. (2009)*

End effects
Tunisian monthly electrical data
HP and EMD trends (1980-2006)
Wavelet trends: A7, A8, A9?

Approximations (/trends?)

Details (/IMFs?)

Choosing Good Level?
Wavelet trends: $daub5, \ sym8$?

Choosing Good Wavelet?

EMD for trend extraction
Conclusion

• EMD eligible method for *trend extraction from seasonal time series*

• Comparison with
  – HP Filter: widely used method in economics (λ?)
  – Wavelets: another time-scale/frequency method (wavelet?)

• EMD trend extraction does not require any tuning parameter thanks to its *adaptive nature*

• *Perspective: extensions to other trends and comparisons with proposals from Moghtaderi et al. (2010)*
References

• Alexandrov, Bianconcini, Bee Dagum, Maass, Mcelroy (2009) U.S. Census Bureau
• Flandrin, Goncalves, Rilling (2004) EUSIPCO
• Ould Mohamed Mahmoud, Mhamdi, Jaidane-Saidane (2009) IEEE PowerTech
• Moghtaderi, Borgnat, Flandrin (april 2010) Submitted CSDA and AADA
• Pollock (2003) Journal of Statistical Planning and Inference
• Schlicht (2005) Journal of Japan Statistic Society
• Suling, Yanqin, Qiang, Jian (2009) World Congress on Engineering and Computer Science
• Wu, Huang, Long, Peng (2007) PNAS