Fast Algorithms for Coevolving Time Series Mining

Lei Li

Computer Science Department

Carnegie Mellon University

Advisor:

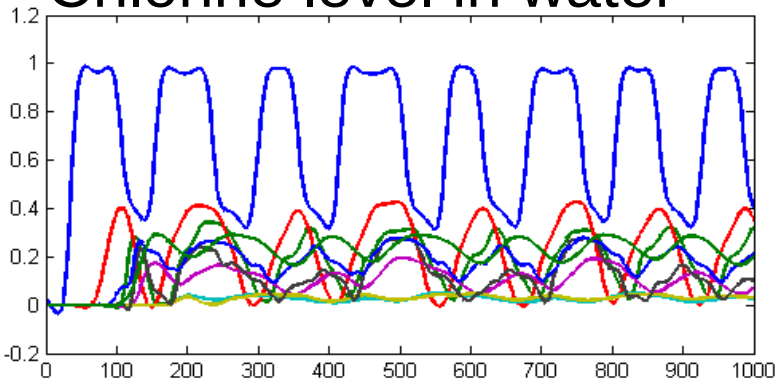
Christos Faloutsos

Thanks

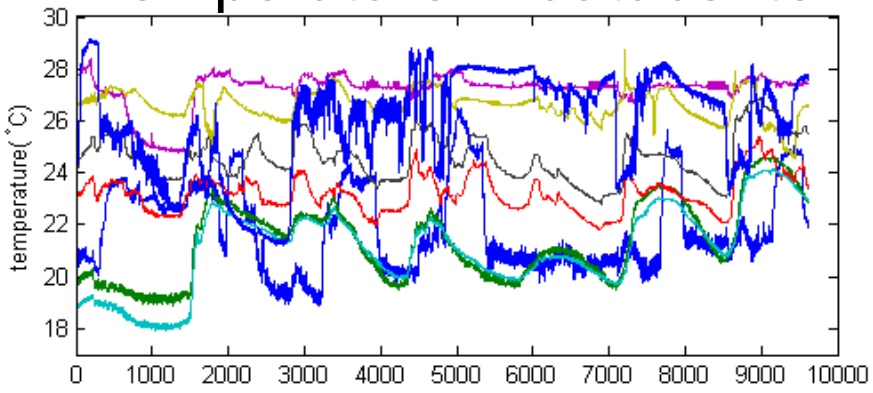
- Organizers:
  - Nikos Mamoulis
  - Yannis Papakonstantinou
  - Timos Sellis
- Travel fellowship from NSF
  - NSF grant IIS-0956600

Coevolving Time Series (TS)

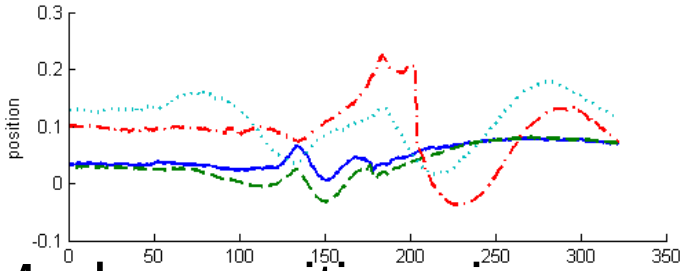
Chlorine level in water



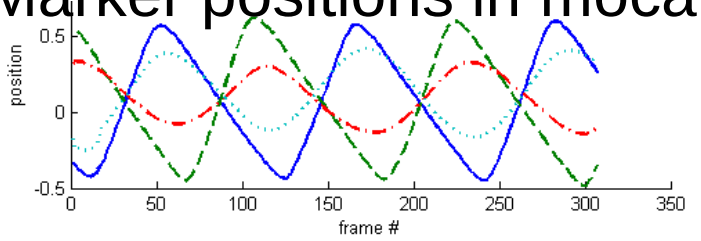
Temperature in datacenter



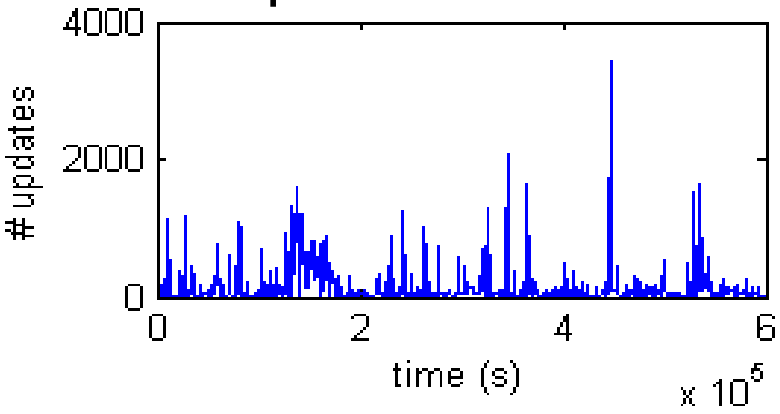
Need fast algorithms for time series mining



Marker positions in mocap



BGP updates in network



Outline

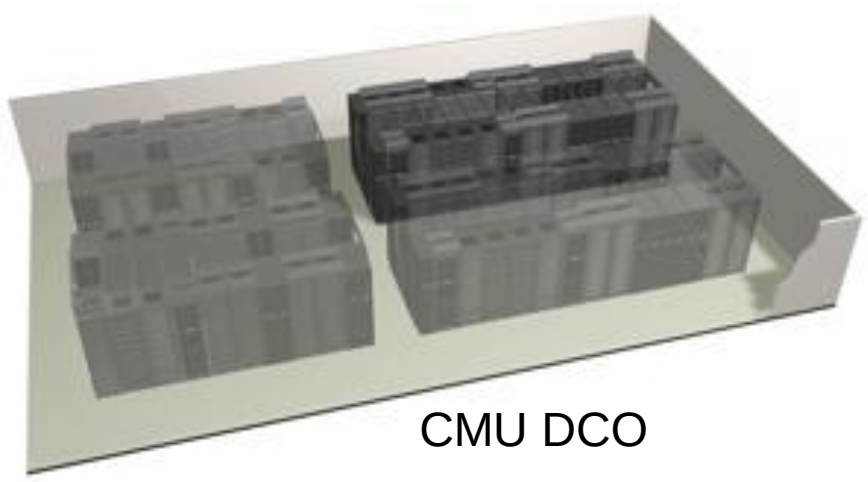
- Motivation
  - Mining tasks, goals, and problems
- Completed Work
  - P1:Mining w/ Missing Value [Li+ 2009]
  - P2:Parallel Learning [Li+ 2008b]
  - P3:Natural Motion Stitching [Li+ 2008a]
- Conclusion

M1: Natural Motion Generation

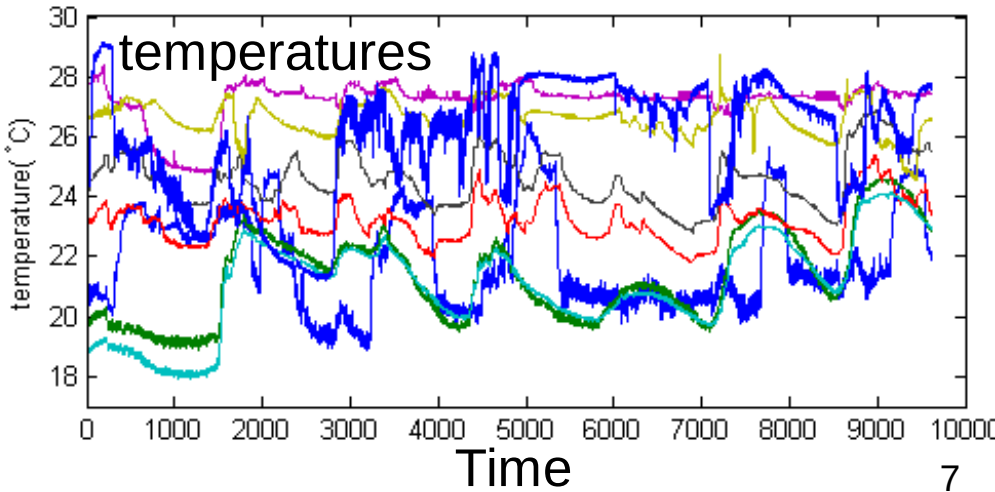
- How to generate new *realistic* motions from mocap database?
- e.g. “karate kick” → “boxing”
- Applications:
  - Game ($57billion 2009)
  - Movie animation
  - Quality of Life (assistive devices)

M2: Data Summarization

- How to compress & manage large time series?
  - A datacenter with 5000 servers: 1TB data per day, 55 million streams ([Reeves+ 2009])
- Goal: save energy in data center
  - $4.5billion power for US dc's 2006

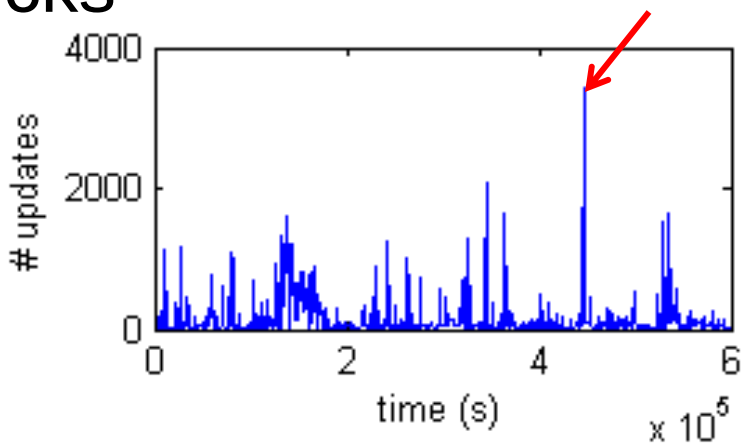


CMU DCO



M3: Anomaly Detection

- How to detect anomalies?
- Applications:
  - Intrusion computer network traffic (e.g. # of packets)
  - Detect leakage or attack in drinking water system by monitoring chlorine levels
  - Spam/robot in web clicks



Time Series Mining Tasks

- Pattern Discovery (e.g. cross-correlation, lag-correlation)
  - T1:Forecasting
  - T2:Summarization
  - T3:Segmentation (detecting change points)
  - T4:Anomaly detection
- Feature Extraction (e.g. wavelets coefficients)
  - T5:Clustering
  - T6:Indexing TS database
  - T7:Visualization

Goals for Mining Algorithms

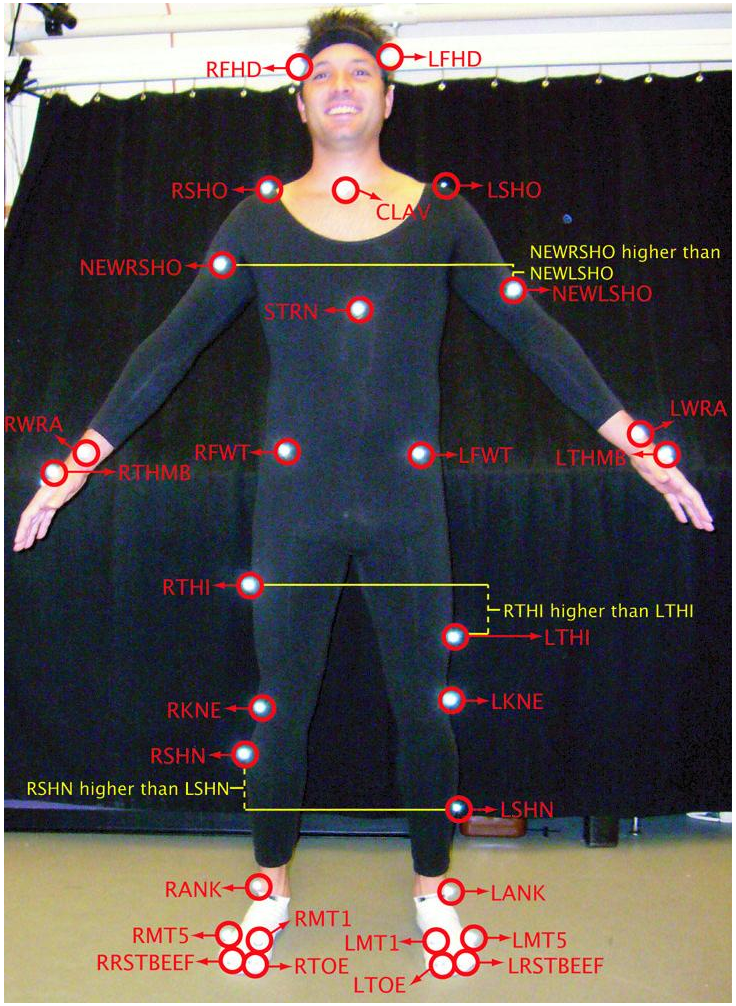
- G1:Effective:
  - achieve low reconstruction error (mean square error) (DynaMMo, [Li+2009])
  - high precision/recall, classification accuracy
- G2:Scalable:
  - to the size (e.g. length) of sequences
  - on modern hardware (Cut-And-Stitch [Li+2008b])

Outline

- Motivation
- Completed Work
  - P1: DynaMMo: Mining w/ Missing Value[Li+09]
    - Problem Definition
    - Intuition of Proposed Method { recovering, compression, segmentation }
    - Results
  - P2: Cut-And-Stitch: Parallel Learning [Li+08b]
  - P3: Natural Motion Stitching [Li+08a]
- Conclusion

Missing Values in Time Series

- Motion Capture:
  - Markers on human actors
  - Cameras used to track the 3D positions
  - Duration: 100-500
  - 93 dimensional body-local coordinates after preprocessing (31-bones)
- Sensor data missing due to:
  - Low battery
  - RF error



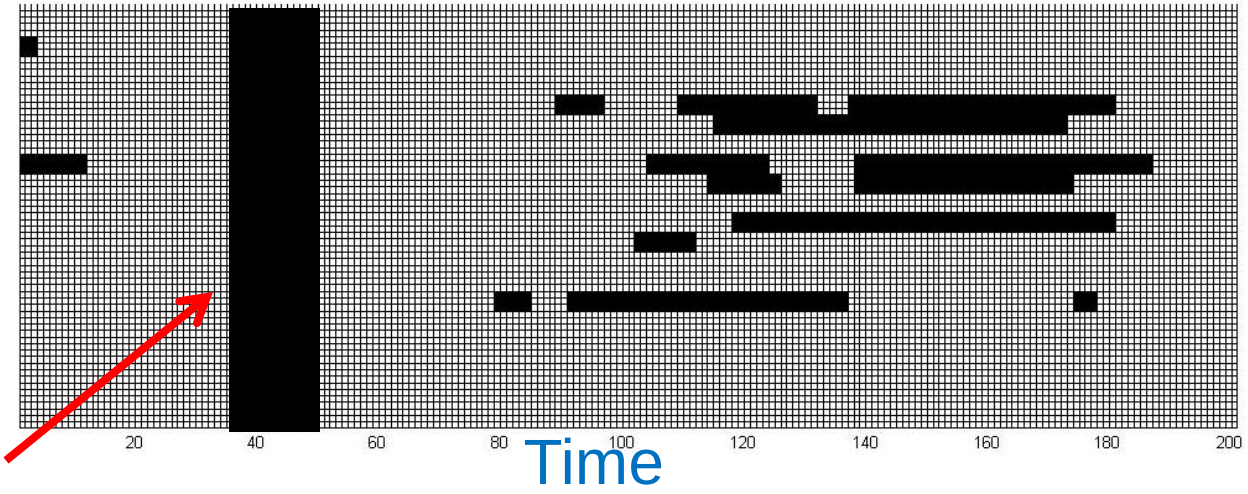
From mocap.cs.cmu.edu

joint work w/ C. Faloutsos, J. McCann, N. Pollard. [Li et al, KDD 2009]

Problem Definition [Li+2009]

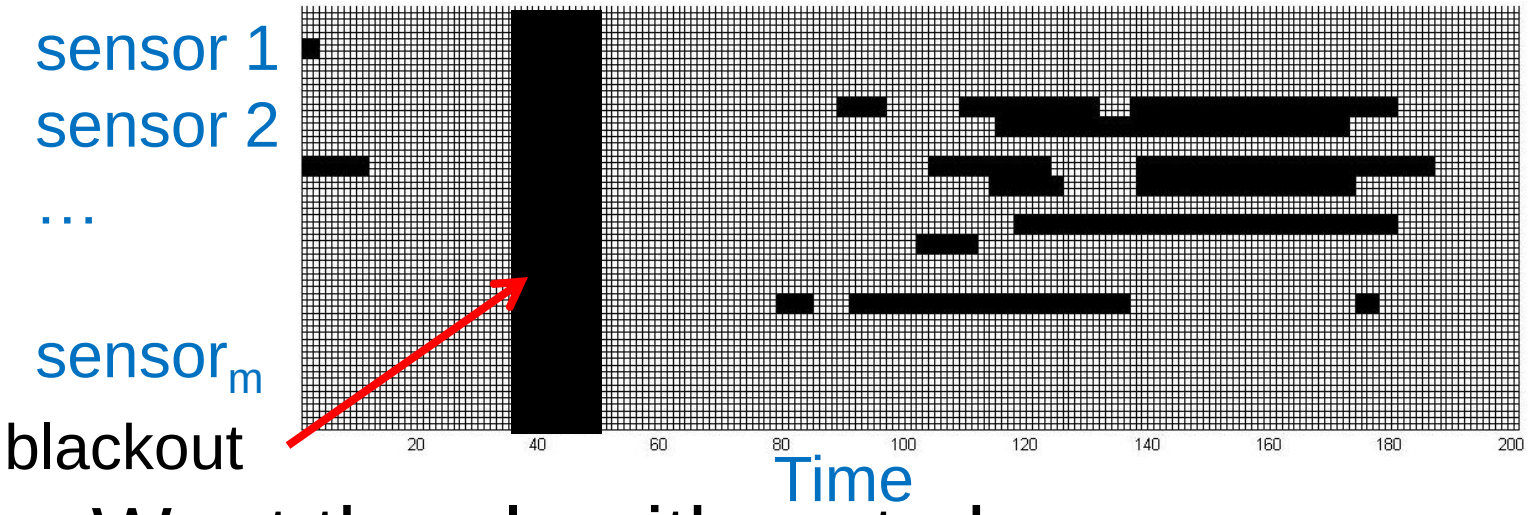
- Given

sensor 1
sensor 2
…
sensor$_m$
blackout



- Find algorithms for:
  - Recovering missing values
  - Compression/summarization (T2)
  - Segmentation (T3)

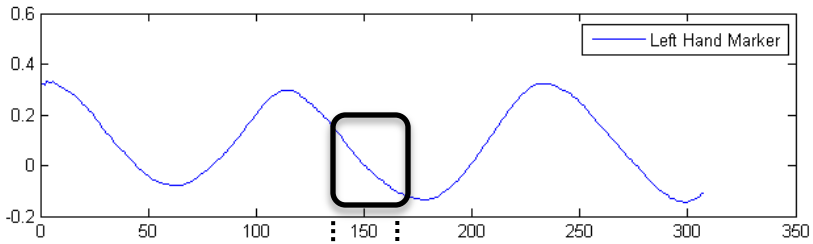
Problem Definition (cont')



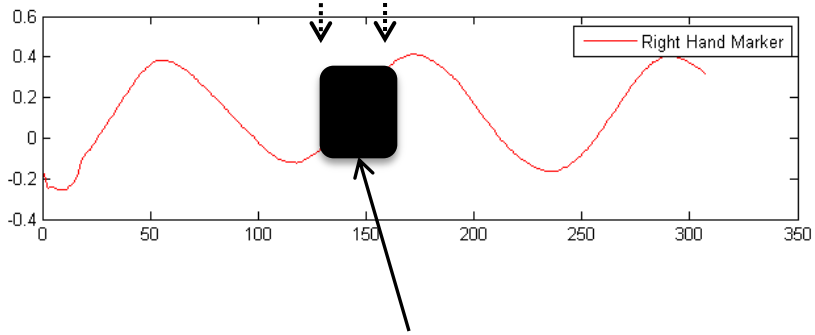
- Want the algorithms to be:
  - G1:*Effective*
  - G2:*Scalable:* to duration of sequences

Proposed Method: Intuition

Position of Left hand marker

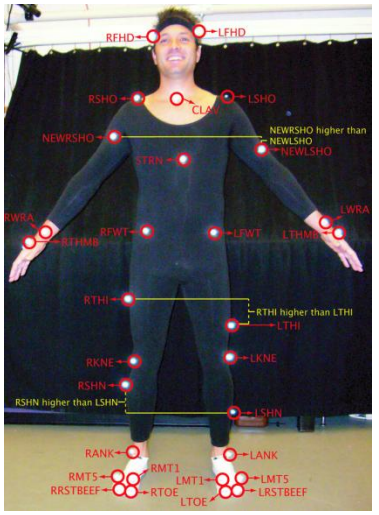


Position of right hand marker



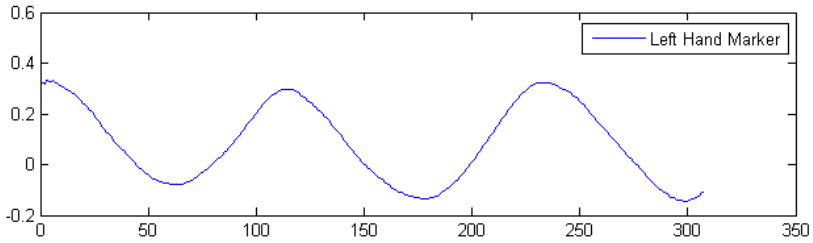
missing

Recover using Correlation among multiple sequences



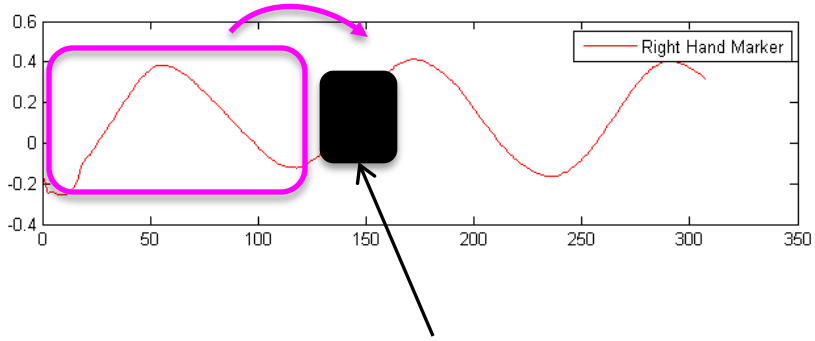
Proposed Method: DynaMMo Intuition

Position of Left hand marker



Recover using Dynamics temporal moving pattern

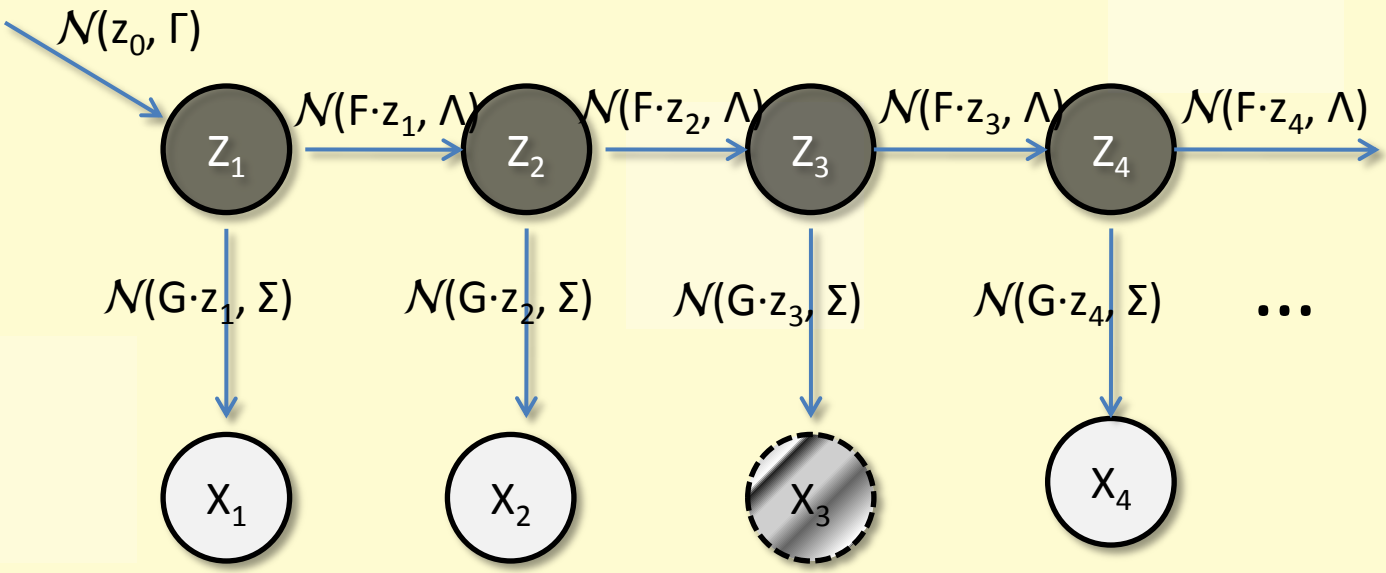
Position of right hand marker



more results in [Li et al 2009]

Underlying Model (details)

Use *Linear Dynamical Systems* to model whole sequence.



Model parameters:

$\theta = \{z_0, \Gamma, F, \Lambda, G, \Sigma\}$

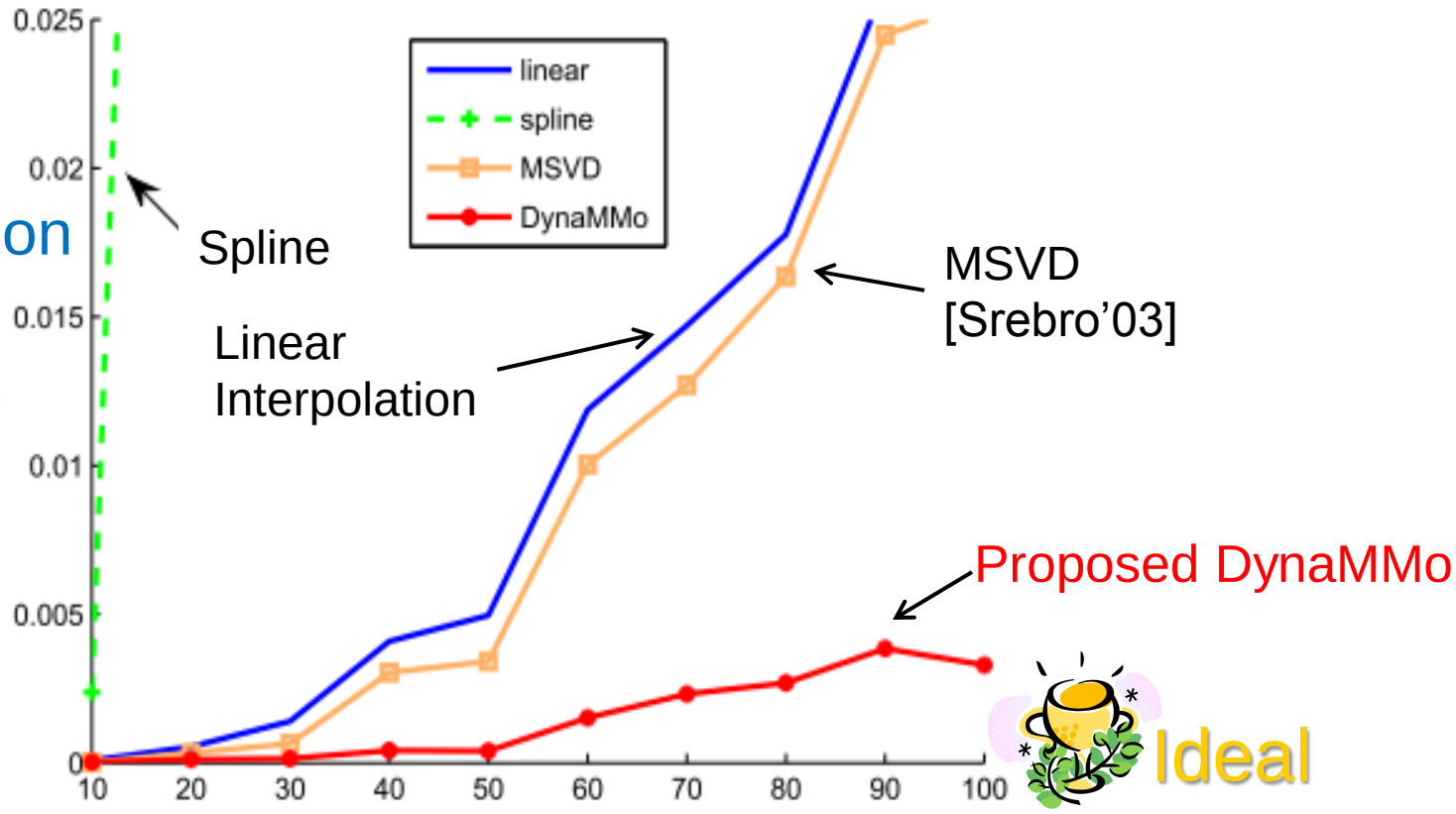
$$z_1 = z_0 + \omega_0$$

$$z_{n+1} = F \cdot z_n + \omega_n$$

$$x_n = G \cdot z_n + \varepsilon_n$$

Results – Better Missing Value Recovery

Reconstruction error

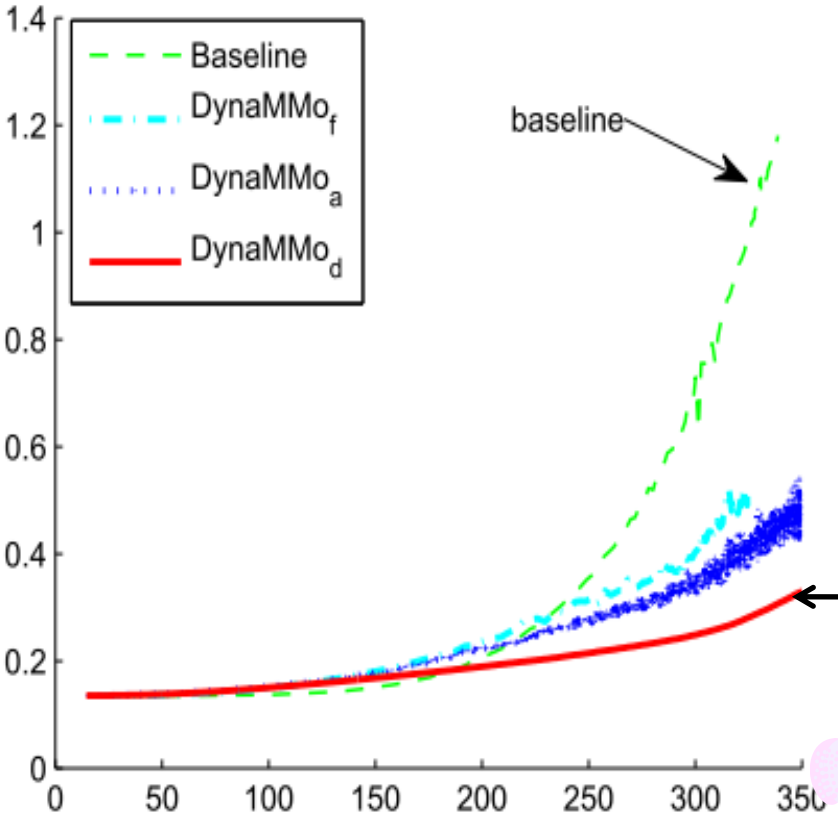


Dataset:
CMU Mocap #16
mocap.cs.cmu.edu

more results in [Li et al 2009]

Results – Better Compression

error



DynaMMo w/ optimal compression

Ideal

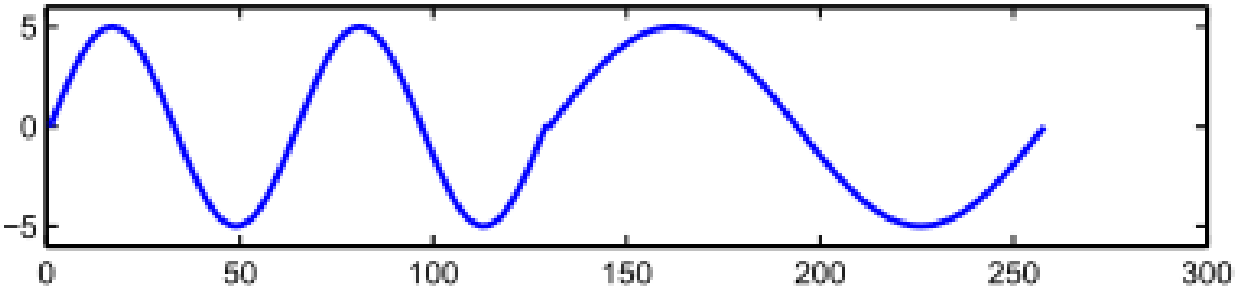
Dataset:
Chlorine levels

more results in [Li et al 2009]

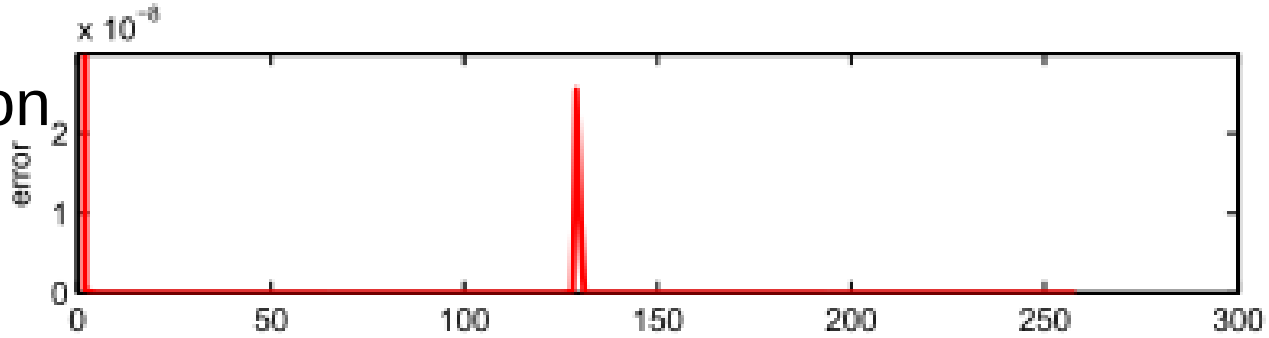
Results: segment synthetic data

- Segment by threshold on reconstruction error

original data



reconstruction error



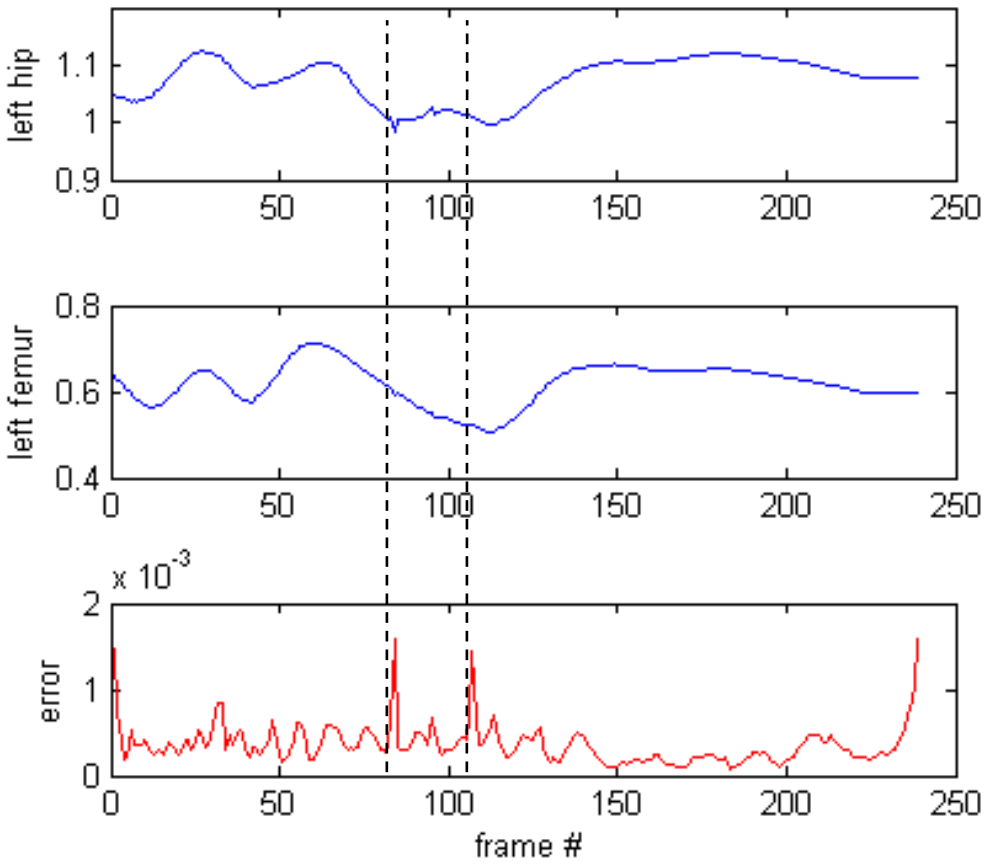
Results – Segmentation

- Find the *transition* during "running" to "stop".

left hip

left femur

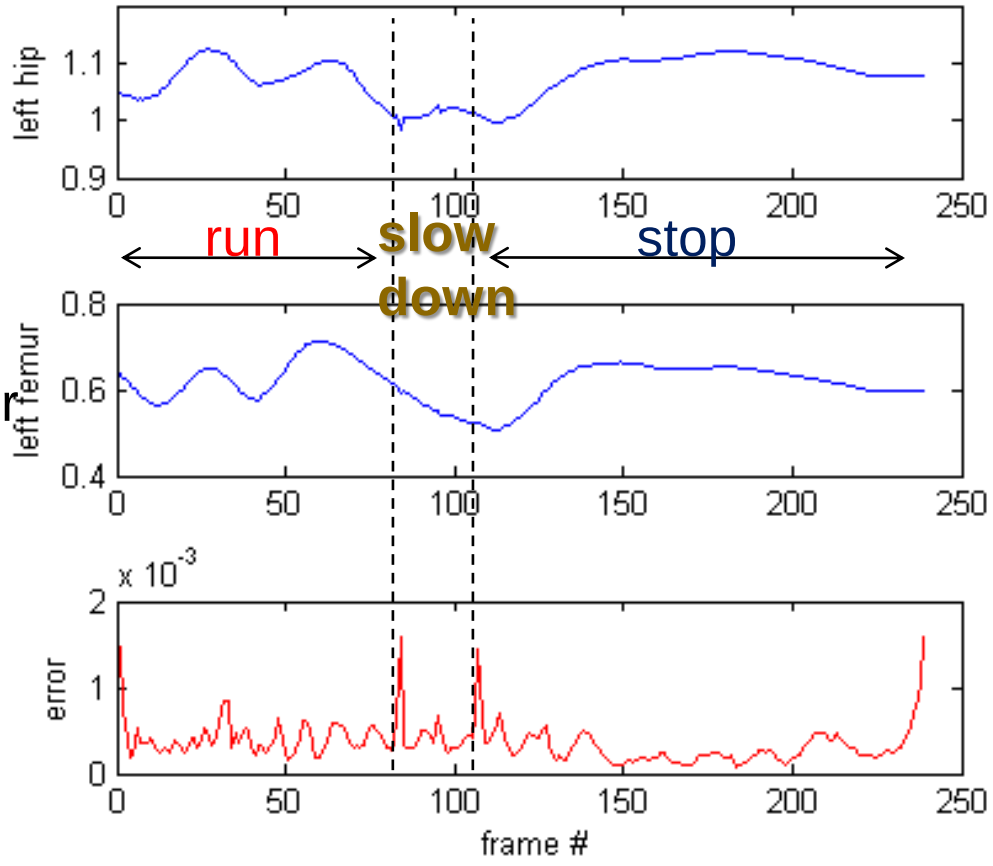
reconstruction error



Results – Segmentation

- Find the *transition* during "running" to "stop".

left hip



reconstruction error

Outline

- Motivation
- Completed Work
  - P1: DynaMMo: Mining w/ Missing Value [Li+09]
    - *Contribution*: the most accurate mining algorithms for TS with missing value so far.
  - P2: Cut-And-Stitch: Parallel Learning [Li+08b]
  - P3:Natural Motion Stitching [Li+08a]
- Conclusion

Outline

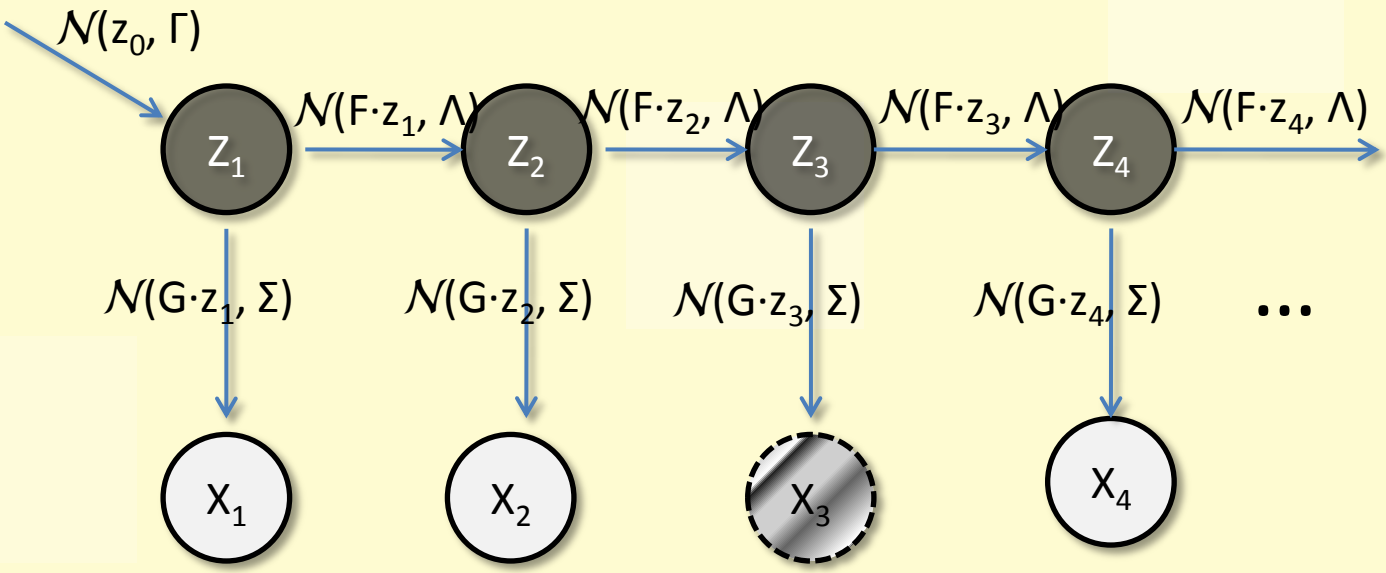
- Motivation
- Completed Work
  - P1: DynaMMo: Mining w/ Missing Value[Li 09]
  - P2: Cut-And-Stitch: Parallel Learning [Li 08b]
    - Problem Definition
    - Basic Intuition
    - Results

Goals for Mining Algorithms

- G1:Effective:
  - achieve low reconstruction error (mean square error) (DynaMMo, [Li 2009])
  - high precision/recall, classification accuracy
- G2:Scalable:
  - to the size (e.g. length) of sequences
  - on modern hardware (Cut-And-Stitch [Li 2008b])

Recap Model for DynaMMo (details)

Use *Linear Dynamical Systems* to model whole sequence.



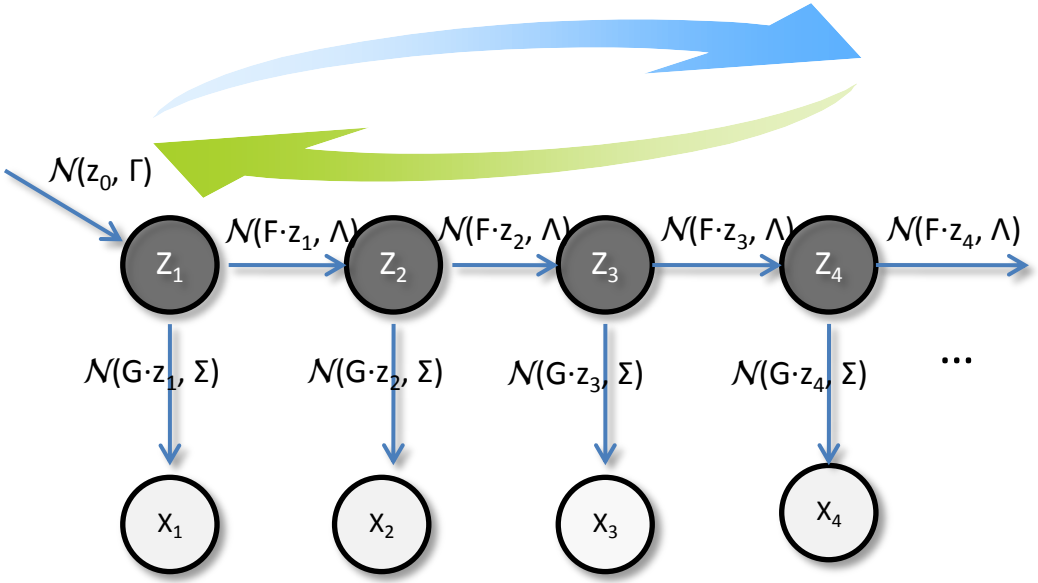
Model parameters:

$\theta = \{z_0, \Gamma, F, \Lambda, G, \Sigma\}$

$$z_1 = z_0 + \omega_0$$
$$z_{n+1} = F \cdot z_n + \omega_n$$
$$x_n = G \cdot z_n + \varepsilon_n$$

Challenge of Learning LDS: Expectation-Maximization Alg.

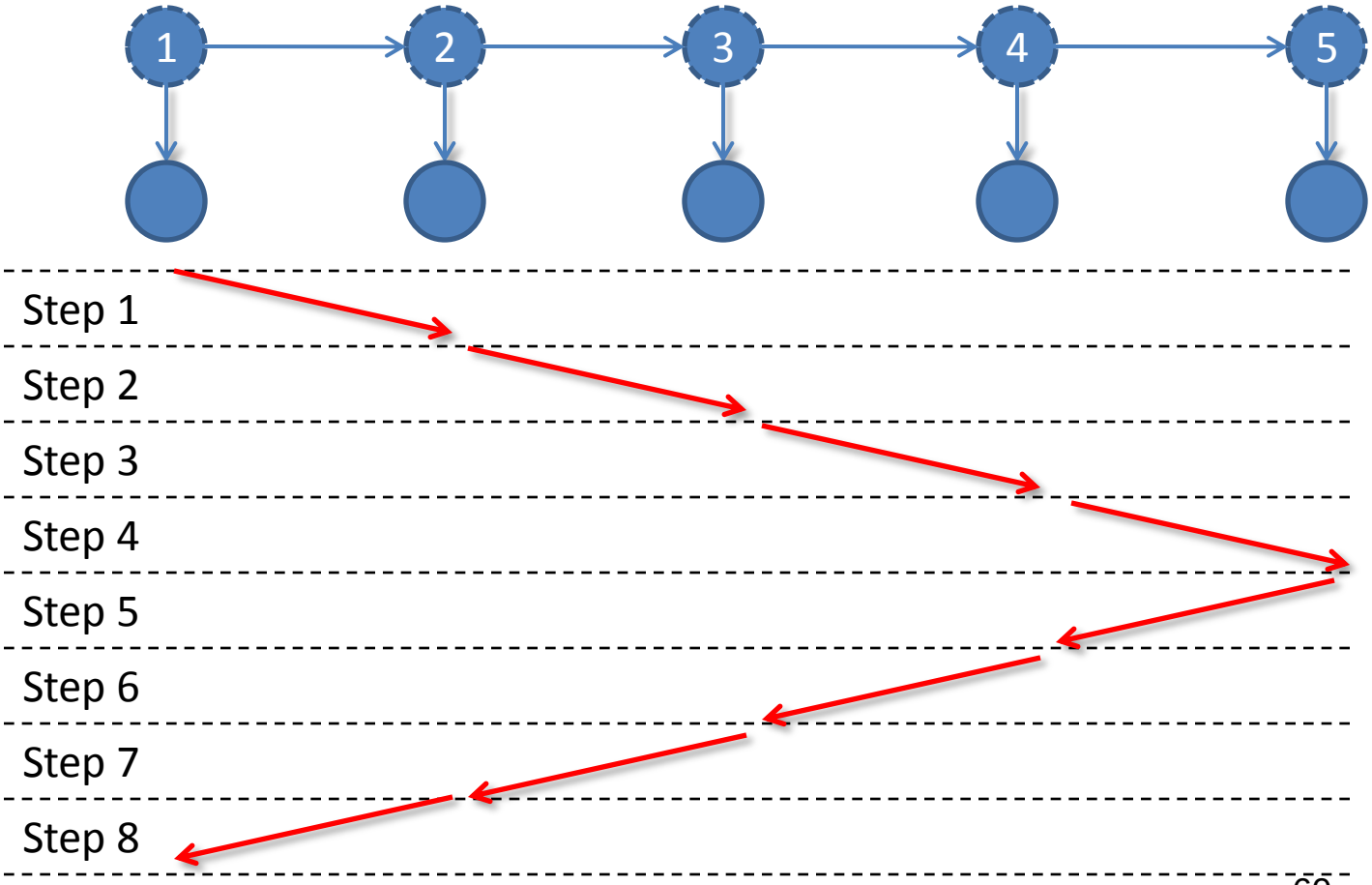
- Not easy to parallelize on multi-processors due to non-trivial data dependency (details in writeup)
- Q: How to parallelize the learning to achieve scalability?



Challenge illustration Expectation-Maximization Alg.

Timeline for E-step (forward-backward) in learning LDS

EM can only uses Single CPU Due to data dependency



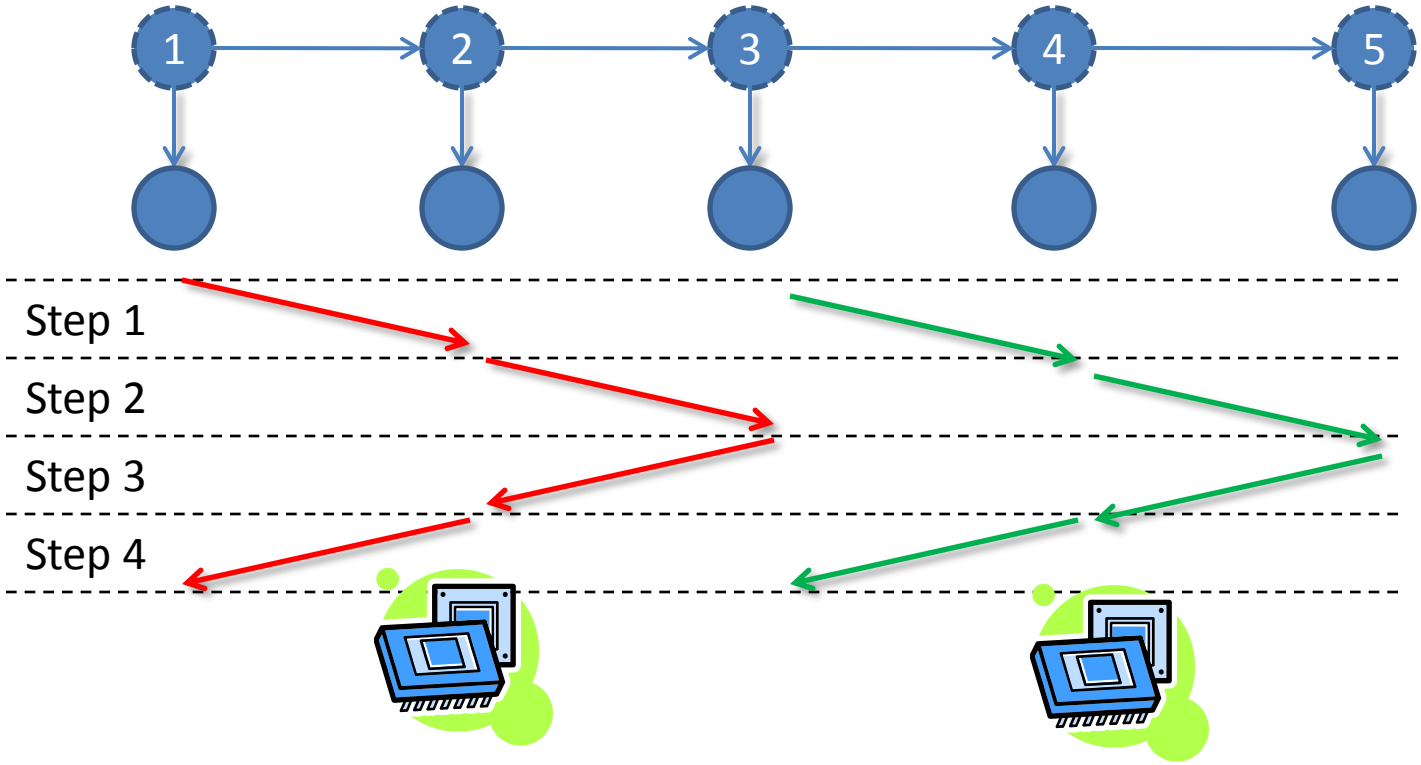
Problem Definition

- Problem:
  - **Given** a sequence of numbers, **design** a parallel learning algorithm to find the best model parameters for Linear Dynamical Systems
- Goal:
  - Achieve ~ linear speed up on multi-core
- Assumption:
  - *Shared memory architecture* (e.g. multi-core)

Proposed Method: Cut-And-Stitch

Intuition:

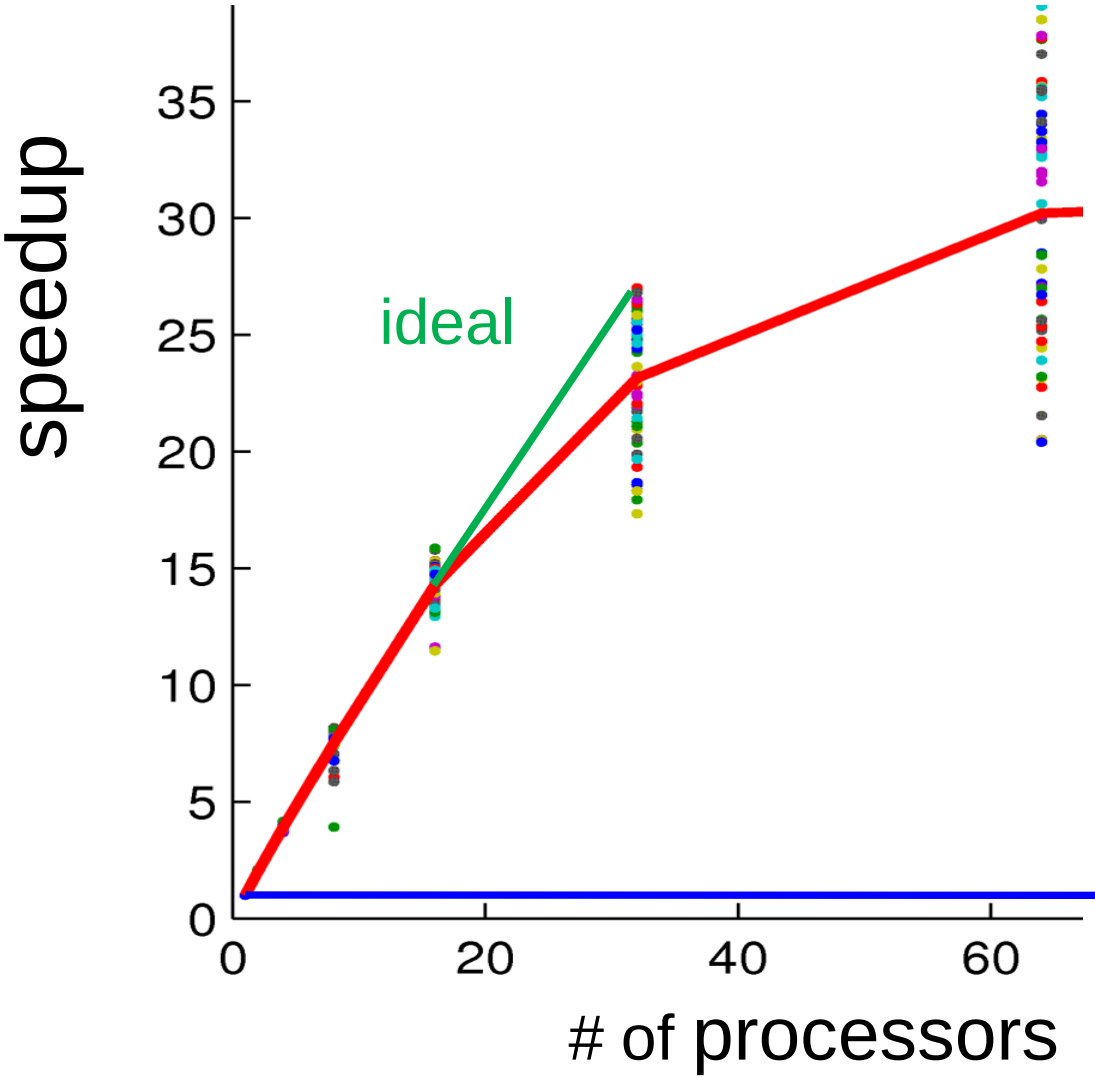
Goal:
with 2 CPUs



Details in [Li et al 2008b]:

Joint work w/ Wenjie Fu, Fan Guo, Todd C. Mowry, Christos Faloutsos.

Near Linear Speedup

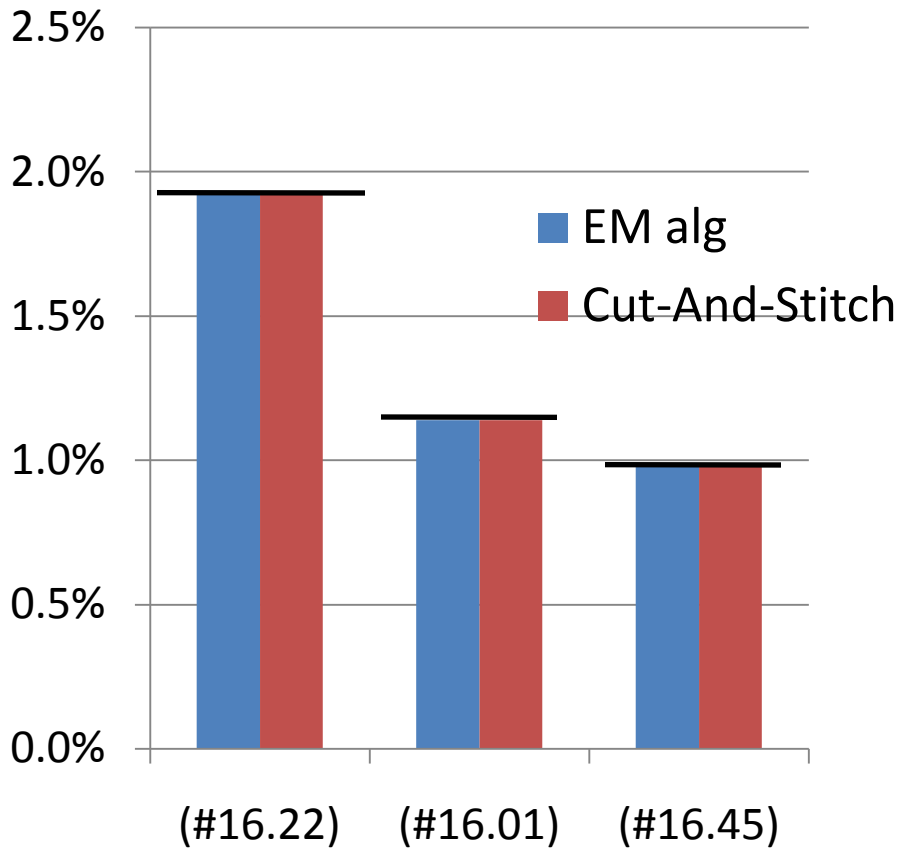


Dataset:
58 motion sequences
CMU Mocap #16
mocap.cs.cmu.edu,
tested on NCSA super computer,

more results in [Li et al 2008b]

No loss of accuracy

Normalized Reconstruction Error



~ IDENTICAL

more results in [Li et al 2008b]

Outline

- Motivation
- Completed Work
  - P1:DynaMMo: Mining w/ Missing Value [Li+09]
  - P2:Cut-And-Stitch:Parallel Learning [Li+08b]
    - *Contribution*: the 1st parallel algorithm for learning LDS

Goals for Mining Algorithms

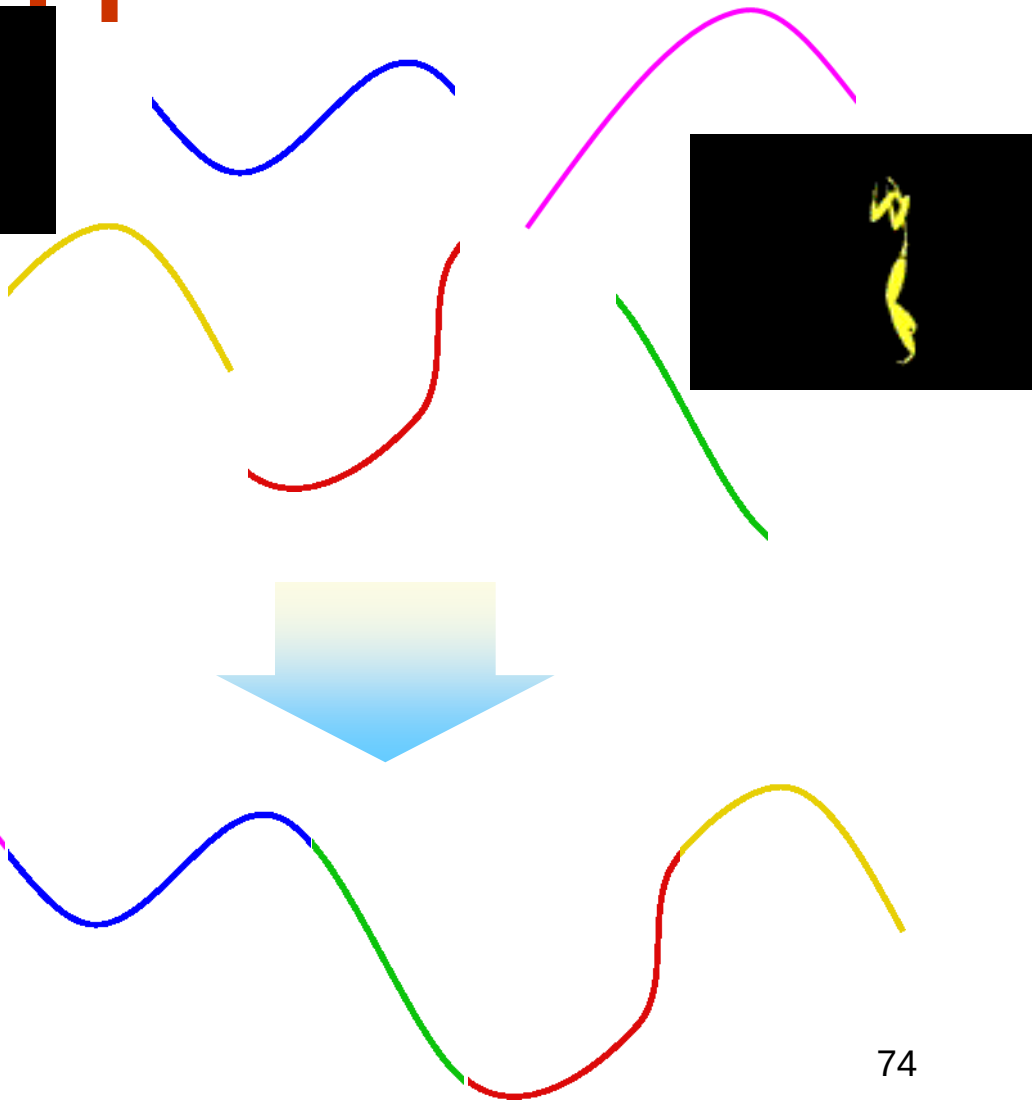
- G1:Effective:
  - achieve low reconstruction error (mean square error) (DynaMMo, [Li 2009])
  - high precision/recall, classification accuracy
- G2:Scalable:
  - to the size (e.g. length) of sequences
  - on modern hardware (Cut-And-Stitch [Li 2008b])

Outline

- Motivation
- Completed Work
  - P1:DynaMMo: Mining w/ Missing Value [Li+09]
  - P2:Cut-And-Stitch:Parallel Learning [Li+08b]
  - P3:Natural Motion Stitching [Li+08a]
    - Problem Definition
    - Proposed Method
    - Results
- Conclusion

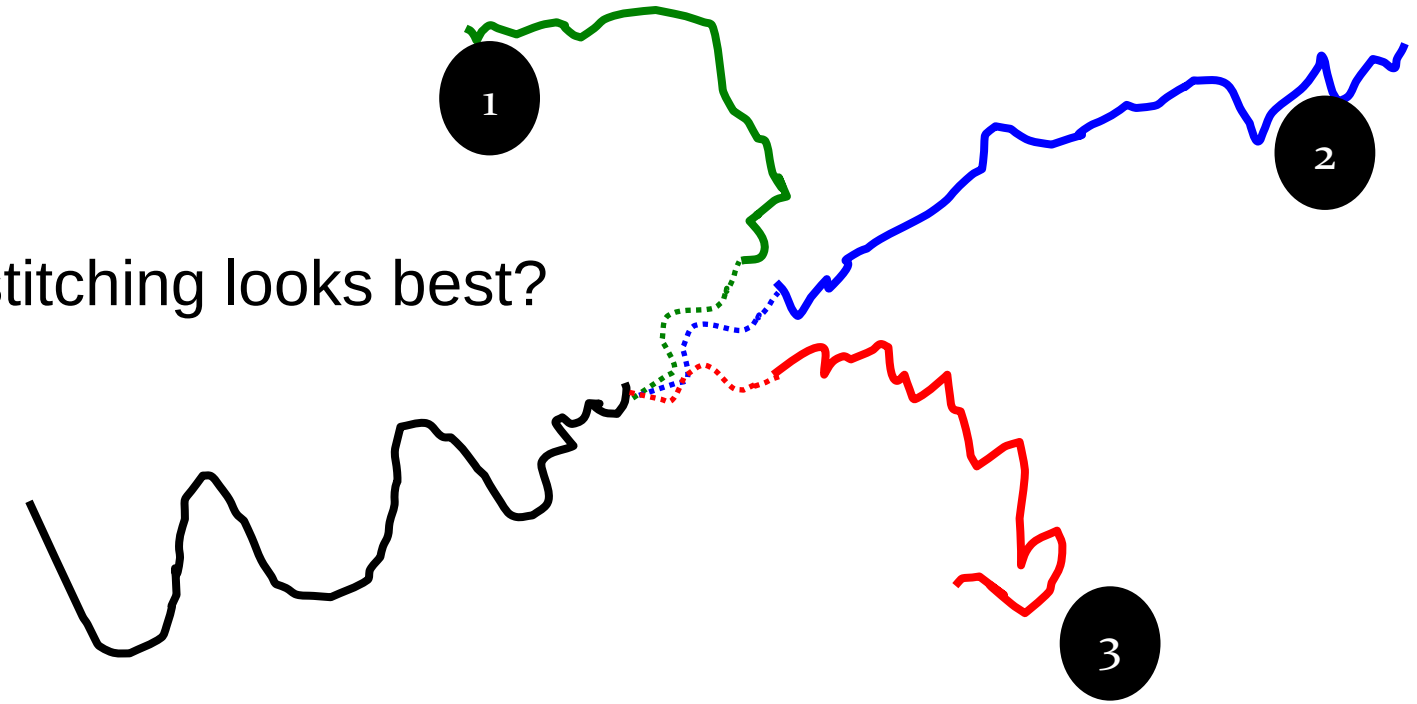
Motion Stitching
A Database Approach

- Select *best stitchable* segments from a set of basic motion pieces and generate new natural motions



Problem Definition

- Given two motion-capture sequences that are to be stitched together, how can we assess the goodness of the stitching?



Which stitching looks best?

Joint work w/ Jim McCann, Nancy Pollard, Christos Faloutsos [Li et al, Eurographics2008]

Competitor: Euclidean distance fail

straight moving

U-Turn

Equally “good” under Euclidean distance

Result – Synthetic Transition

straight moving

U-Turn

Laziness-score prefer straightforward moving

more results in [Li 2008a]

Conclusion

- Pattern discovery w/ missing values (DynaMMo)
  - Recovering missing values
  - Compression
  - Segmentation
- Scale up learning on multicore
  - Parallel learning algorithm for LDS (Cut-And-Stitch)
- Natural human motion stitching
  - An intuitive distance function(Laziness score)

References

- Lei Li, Jim McCann, Nancy Pollard, Christos Faloutsos. DynaMMo: Mining and Summarization of Coevolving Sequences with Missing Values. KDD '09.
- Lei Li, Wenjie Fu, Fan Guo, Todd C. Mowry, Christos Faloutsos. Cut-and-stitch: efficient parallel learning of linear dynamical systems on SMPs. KDD '08.
- Lei Li, Jim McCann, Christos Faloutsos, Nancy Pollard. Laziness is a virtue: Motion stitching using effort minimization. Eurographics 2008.

Question

- Thanks!

- contact: Lei Li (leili@cs.cmu.edu)
- paper, software, dataset on http://www.cs.cmu.edu/~leili