A Specialized Inter-Curve Similarity Measure for Agglomerative Diffusion MRI Streamline Clustering

Jadrian Miles (jadrian@cs.brown.edu)
Visualization Research Lab
Computer Science Department
Brown University, Providence, RI
Talk at NIH, 2009-05-19

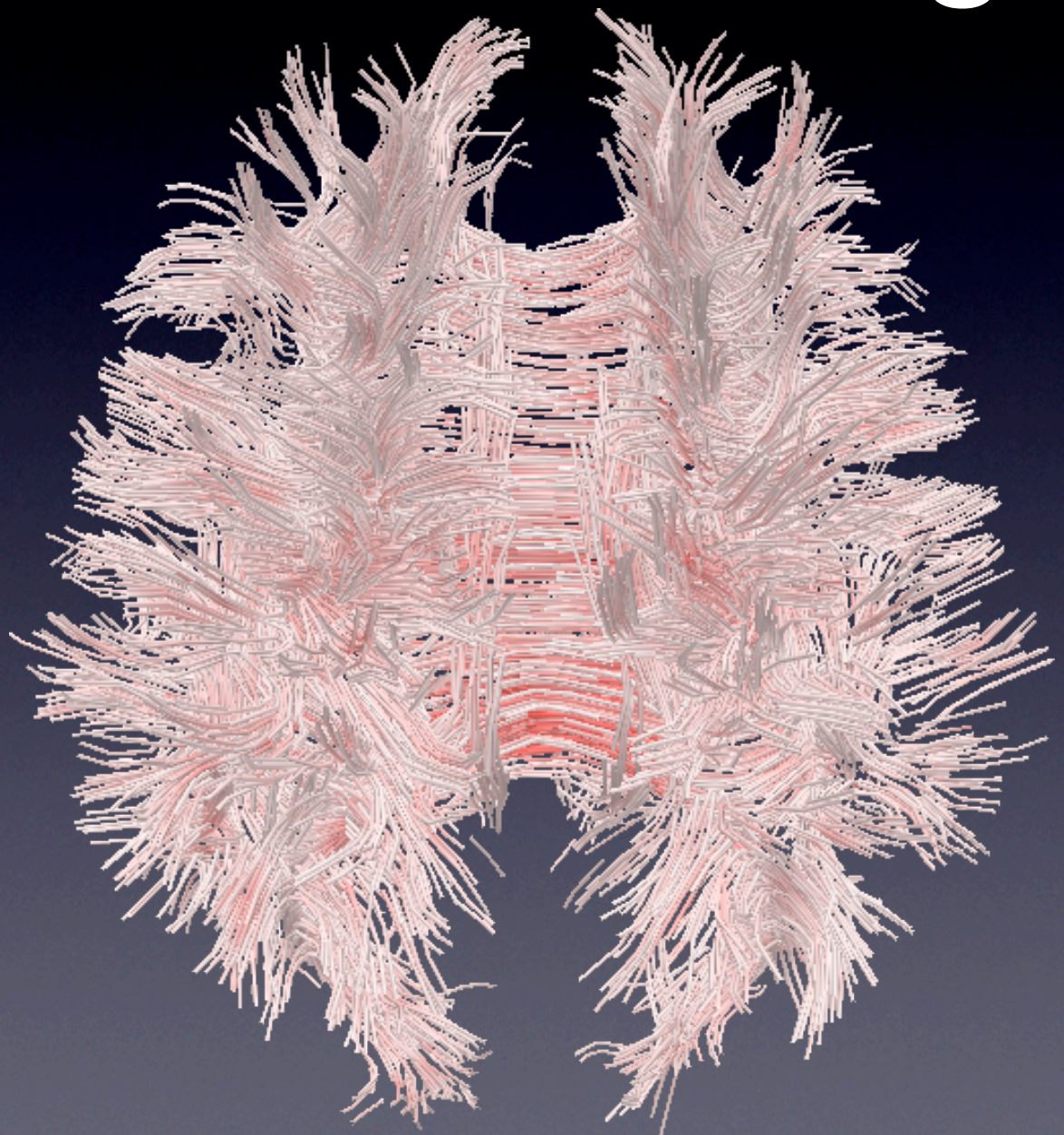
Overview

- Motivation
- Agglomerative clustering
- Inter-curve similarity measures
- Efficient clustering implementation

Overview

- **Motivation**
- Agglomerative clustering
- Inter-curve similarity measures
- Efficient clustering implementation

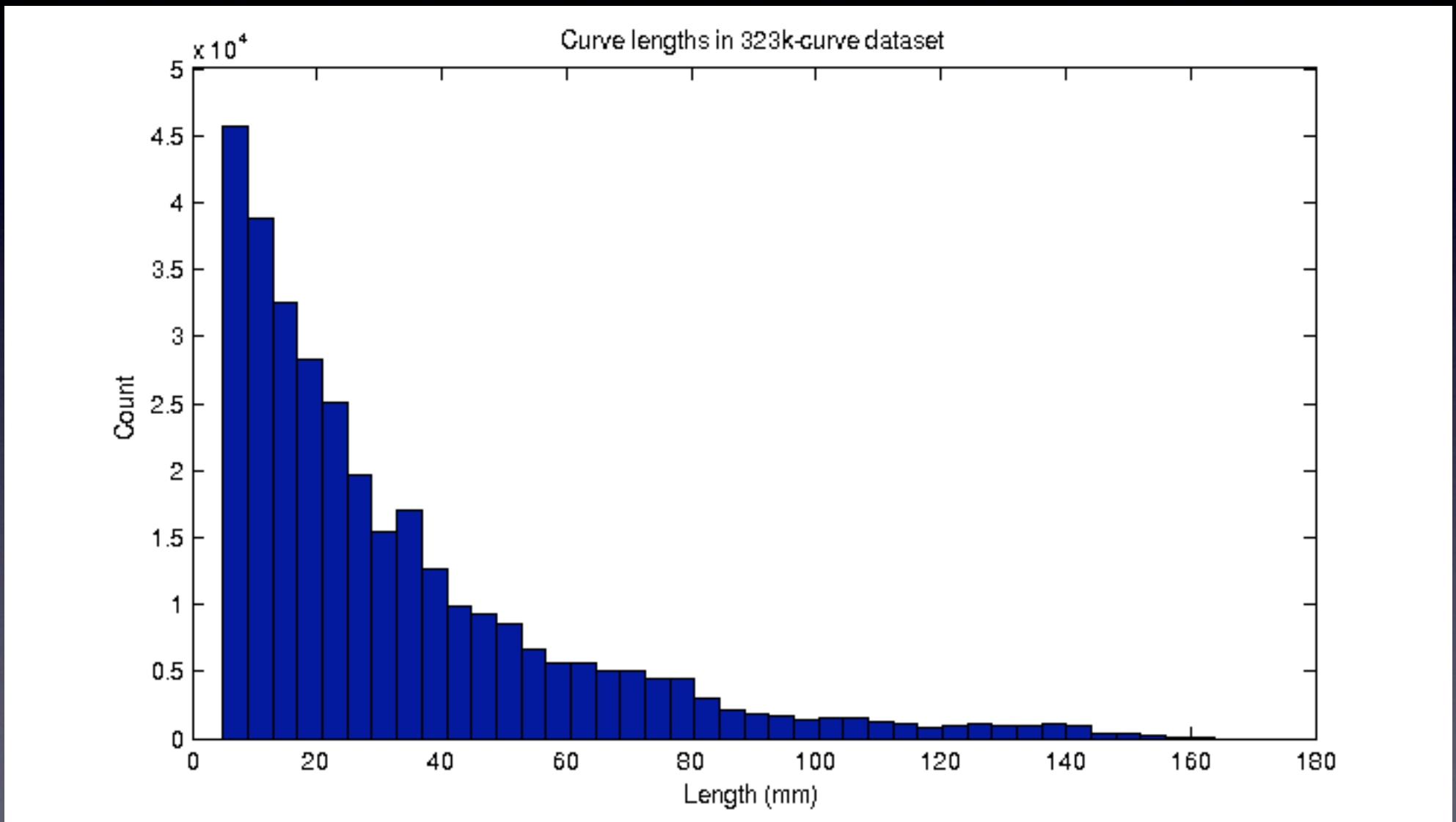
What are we dealing with?



What are we dealing with?

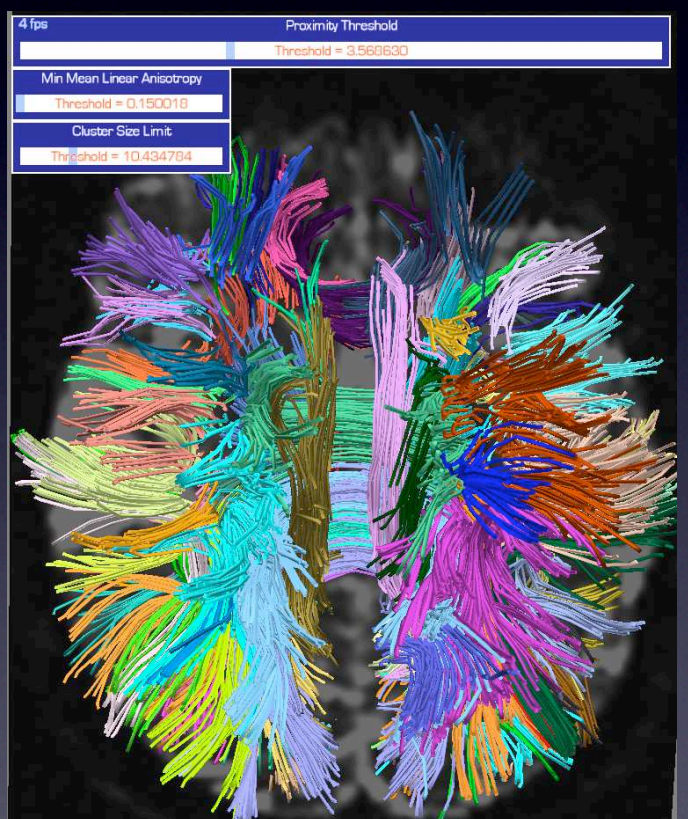
- DTI dataset (1mm isotropic voxels)
- Principle eigenvector streamlines
- 323k space curves
- Lots of short / broken segments

What are we dealing with?



Why clustering?

- Approximates tracts for:
  - Visualization and selection
  - Statistics
  - Non-local modeling



The Goal

- Automatic clustering
- Dense whole-brain tractogram
- Approximate anatomical tracts

The Hypothesis

- Short segments can aid clustering

Overview

- **Motivation**
- Agglomerative clustering
- Inter-curve similarity measures
- Efficient clustering implementation

Overview

- Motivation
- **Agglomerative clustering**
- Inter-curve similarity measures
- Efficient clustering implementation

How does it work?

1. Every curve begins as its own cluster
2. Join clusters of two closest curves
3. Repeat step 2 until termination

How does it work?

- Single-link distance
- $A \sim_\epsilon B \Leftrightarrow \exists \{X_1, X_2, ..., X_k\}$ s.t. $d(A, X_1) < \epsilon$, $d(X_k, B) < \epsilon$, $d(X_i, X_{i+1}) < \epsilon \; \forall \; 1 \leq i < k$

Why agglomerative?

- Provides a cluster hierarchy
- Easy to understand
- Parallelizable
- Flexible termination conditions

Overview

- Motivation
- **Agglomerative clustering**
- Inter-curve similarity measures
- Efficient clustering implementation

Overview

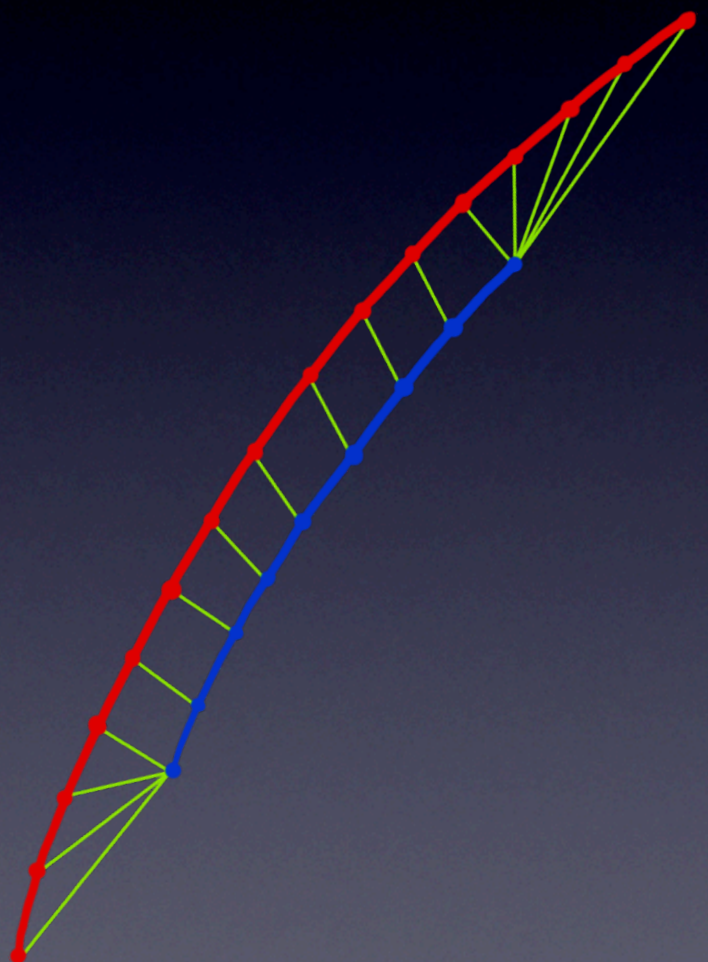
- Motivation
- Agglomerative clustering
- **Inter-curve similarity measures**
- Efficient clustering implementation

Why a new measure?

- Previous work:
  - Corouge
  - Brun
  - Zhang
  - Ding

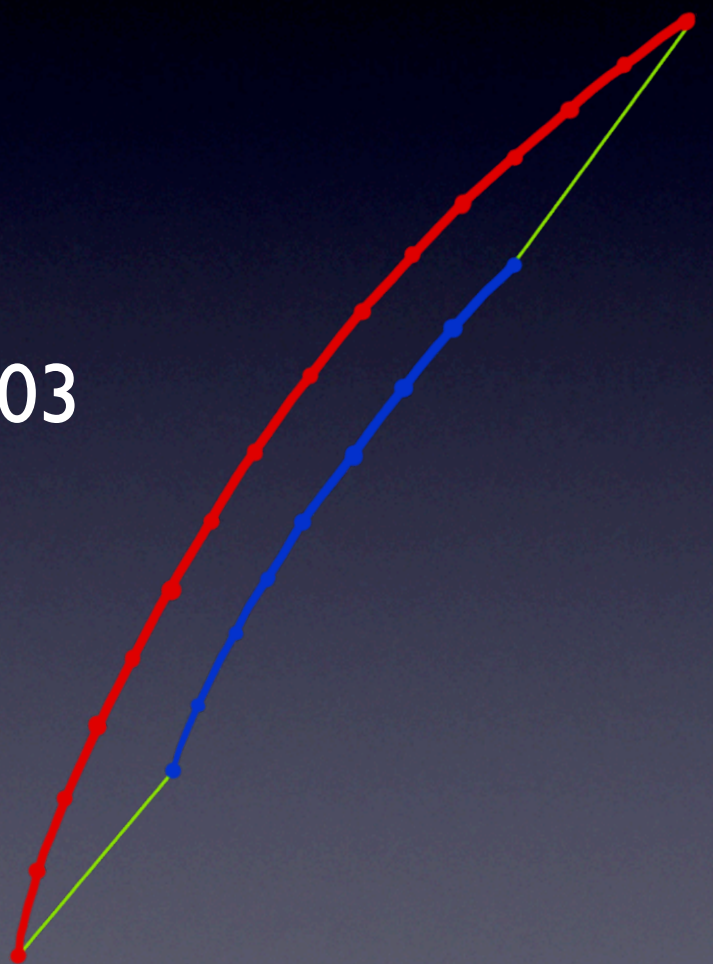
Previous Work

- Corouge, et al., ISBI 2004
  - Minimum-distance vertex pairs
  - Minimum, mean, max of pair distances



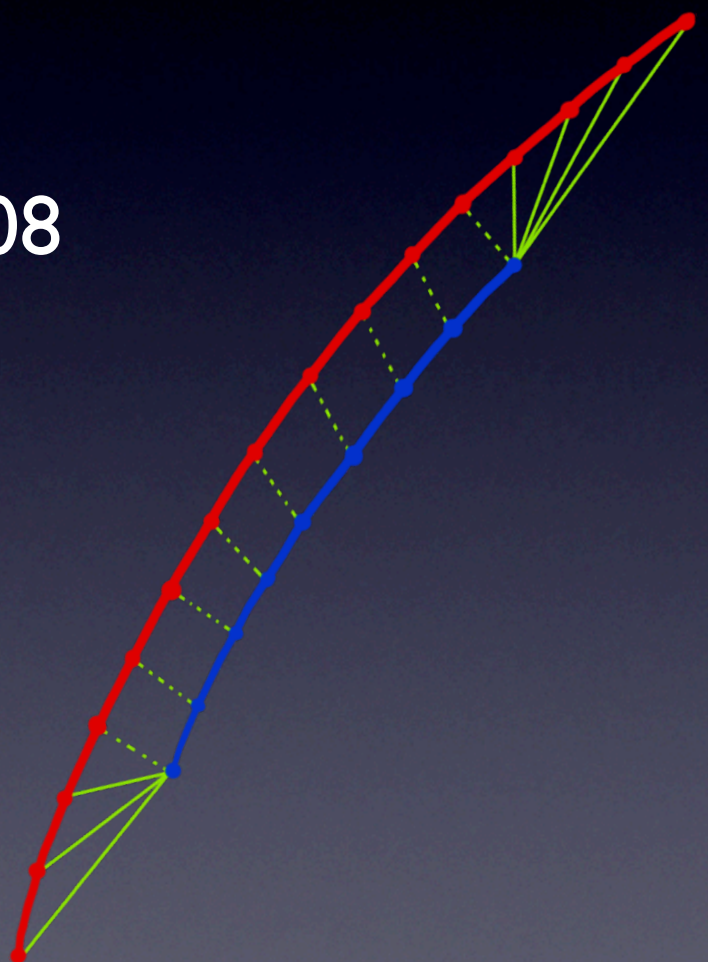
Previous Work

- Brun, et al., EUROCAST 2003
  - Endpoint distance



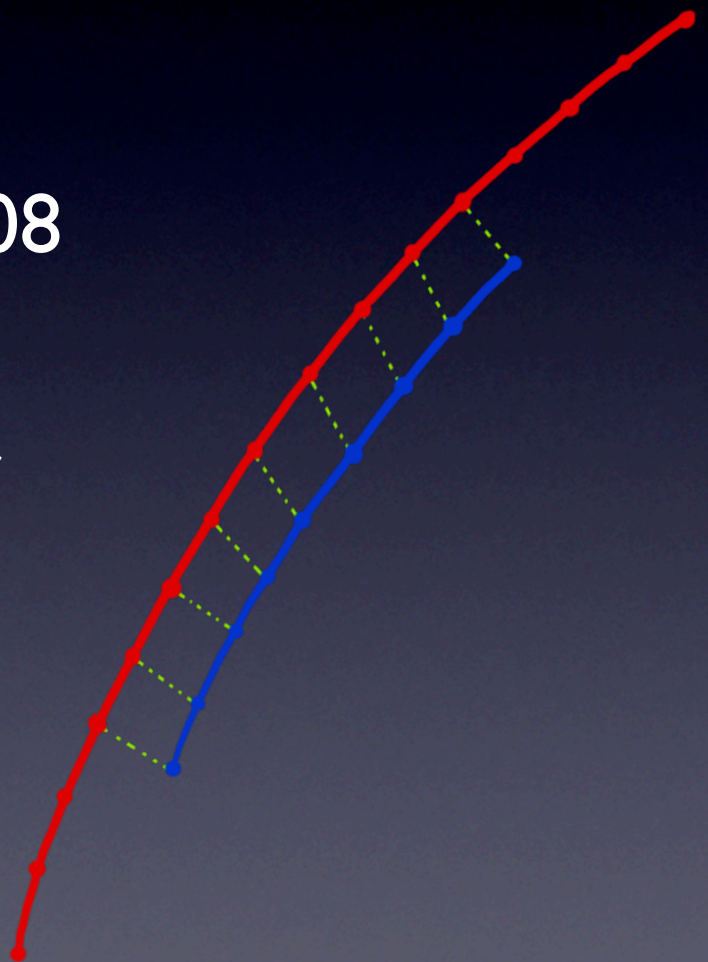
Previous Work

- Zhang & Laidlaw, TVCG 2008
  - Mean distance above threshold



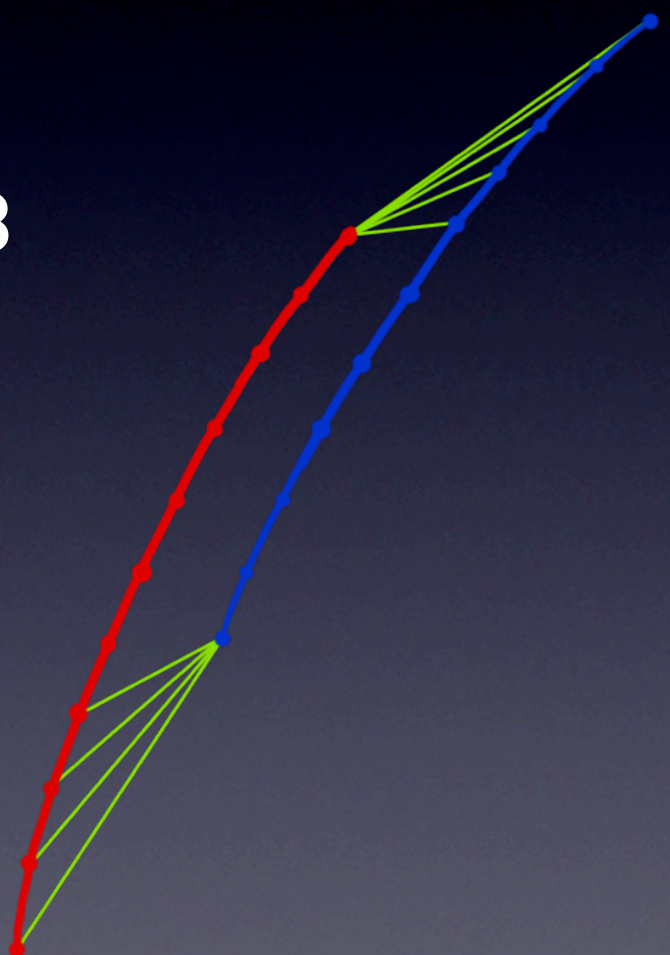
Previous Work

- Zhang & Laidlaw, TVCG 2008
  - Mean distance above threshold ... from shorter curve



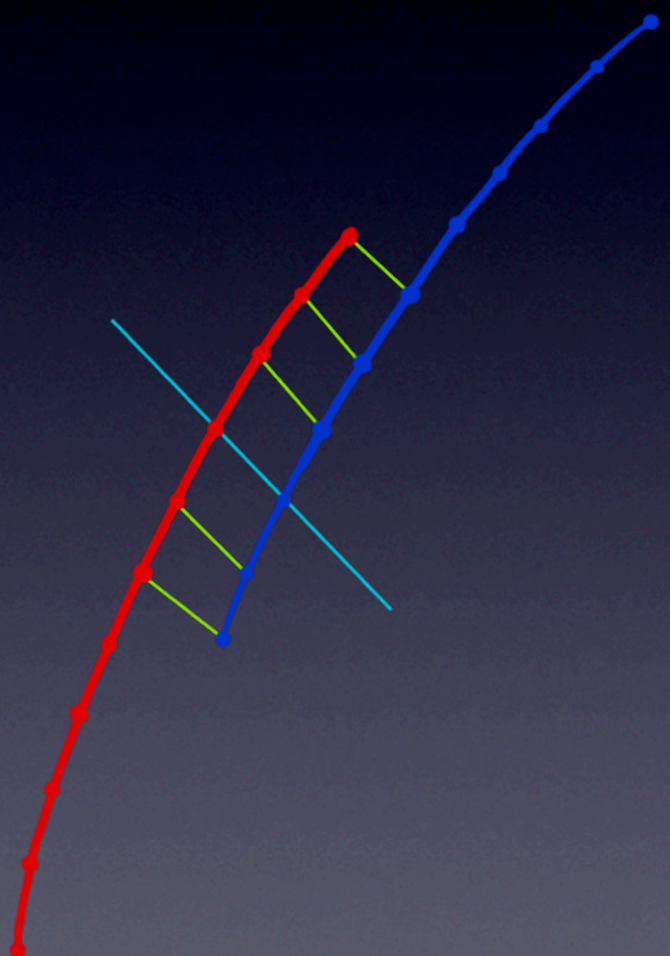
Previous Work

- Zhang & Laidlaw, TVCG 2008
  - Mean distance above threshold ... from shorter curve
  - But!



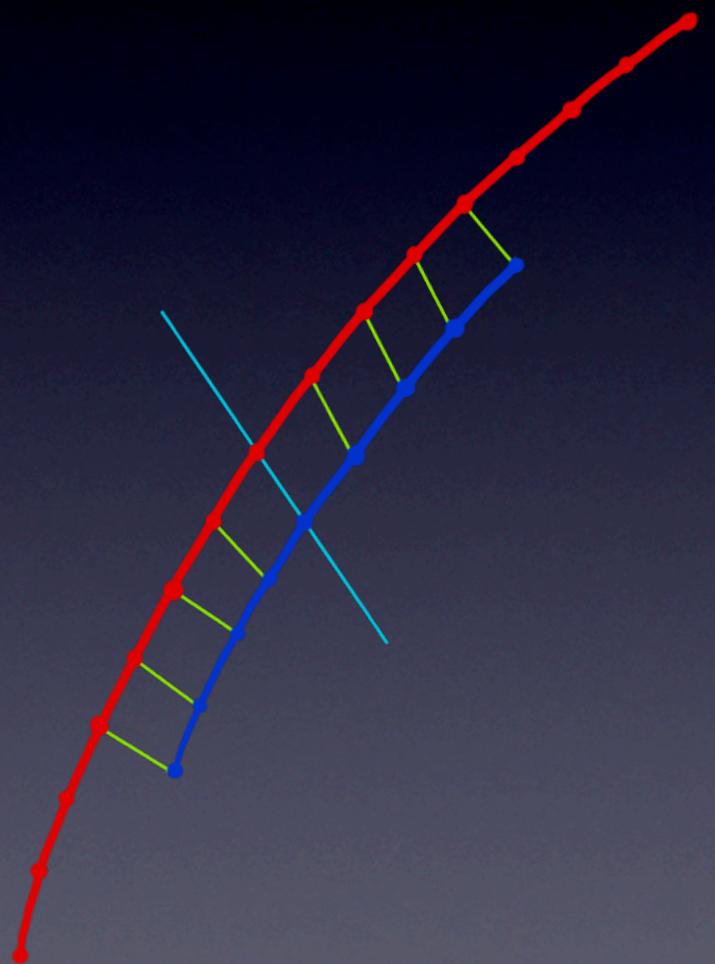
Previous Work

- Ding, et al., Vis 2001
  - Corresponding segment
  - Mean vertex distance in segment



Previous Work

- Ding, et al., Vis 2001
  - Corresponding segment
  - Mean vertex distance in segment



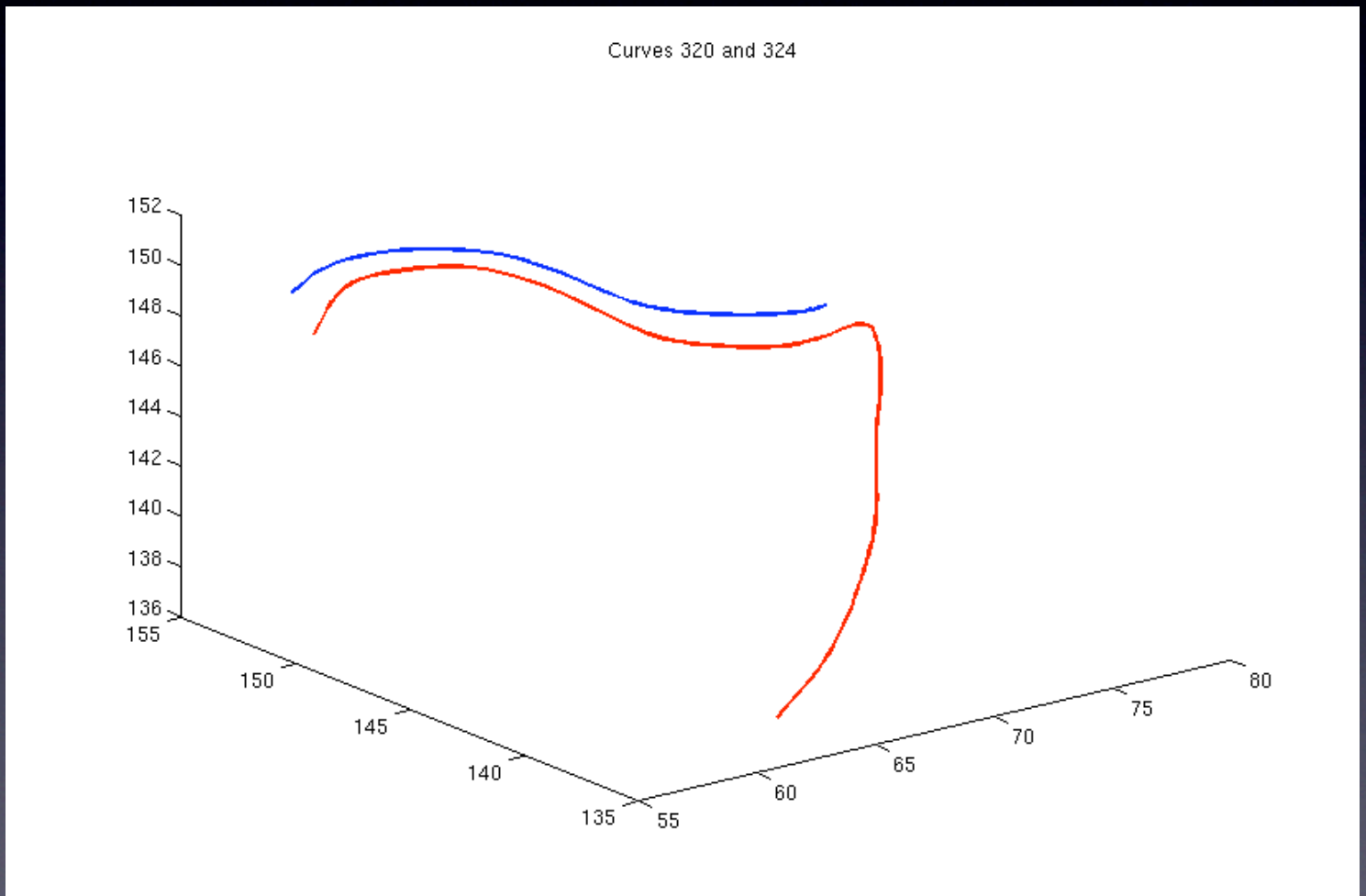
The New Measure

- Detects overlapping segment
- Penalizes skewed curves

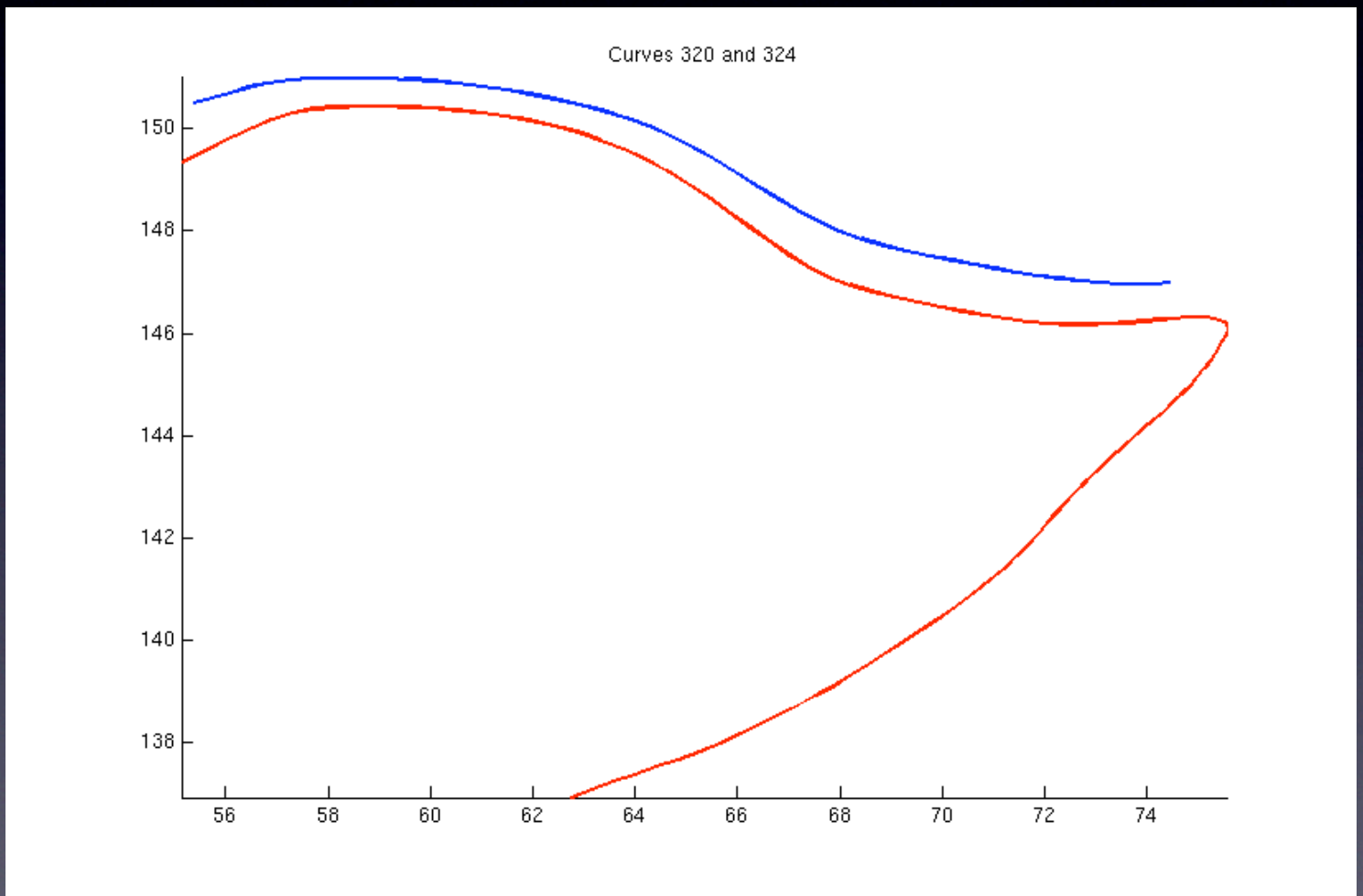
The New Measure

- Trimming
- Induced relative orientation
- Matching order
- Match length ratio

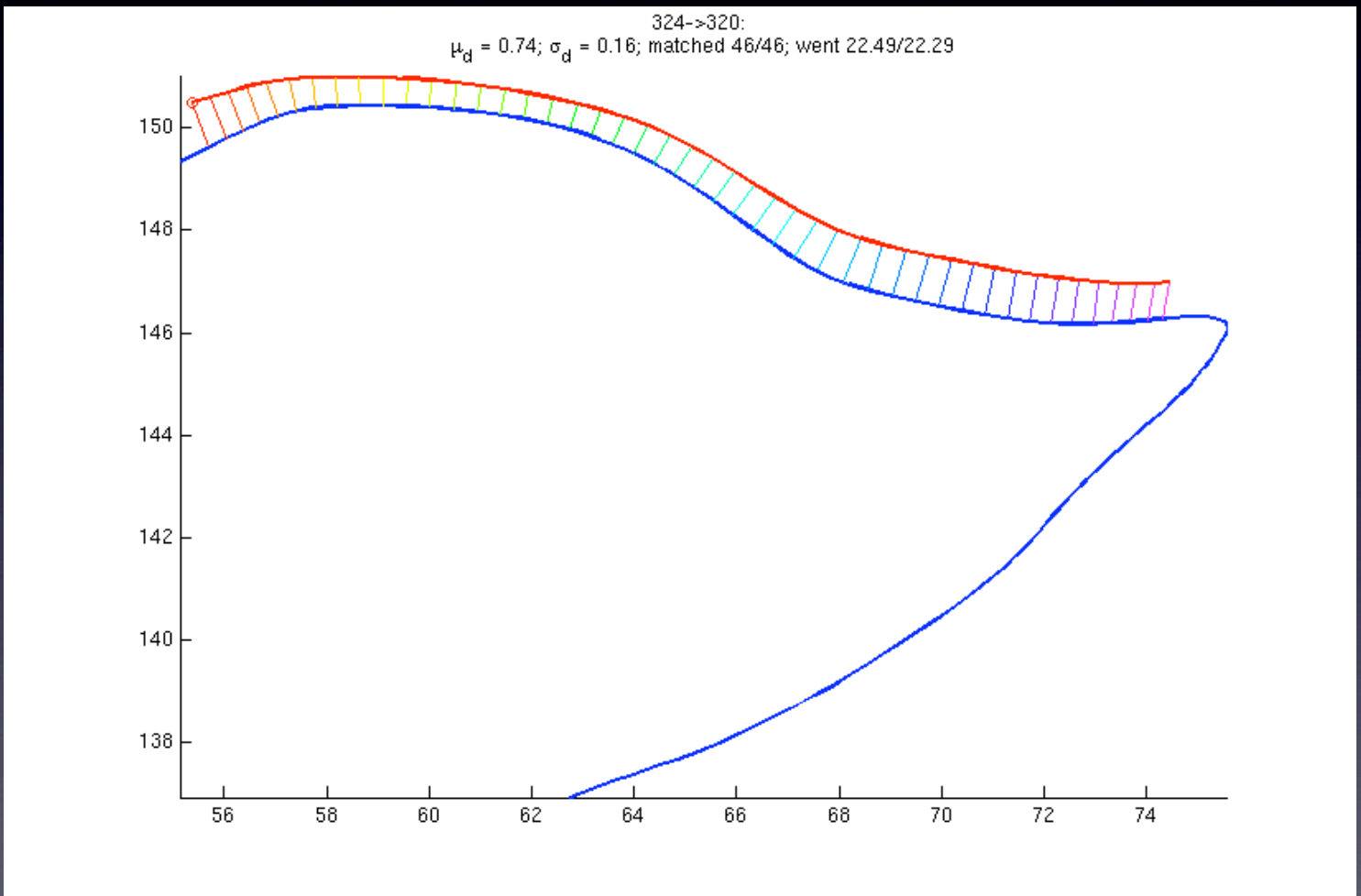
Trimming


Curves 320 and 324
152
150
148
146
144
142
140
138
136
155
150
145
140
135
55
60
65
70
75
80

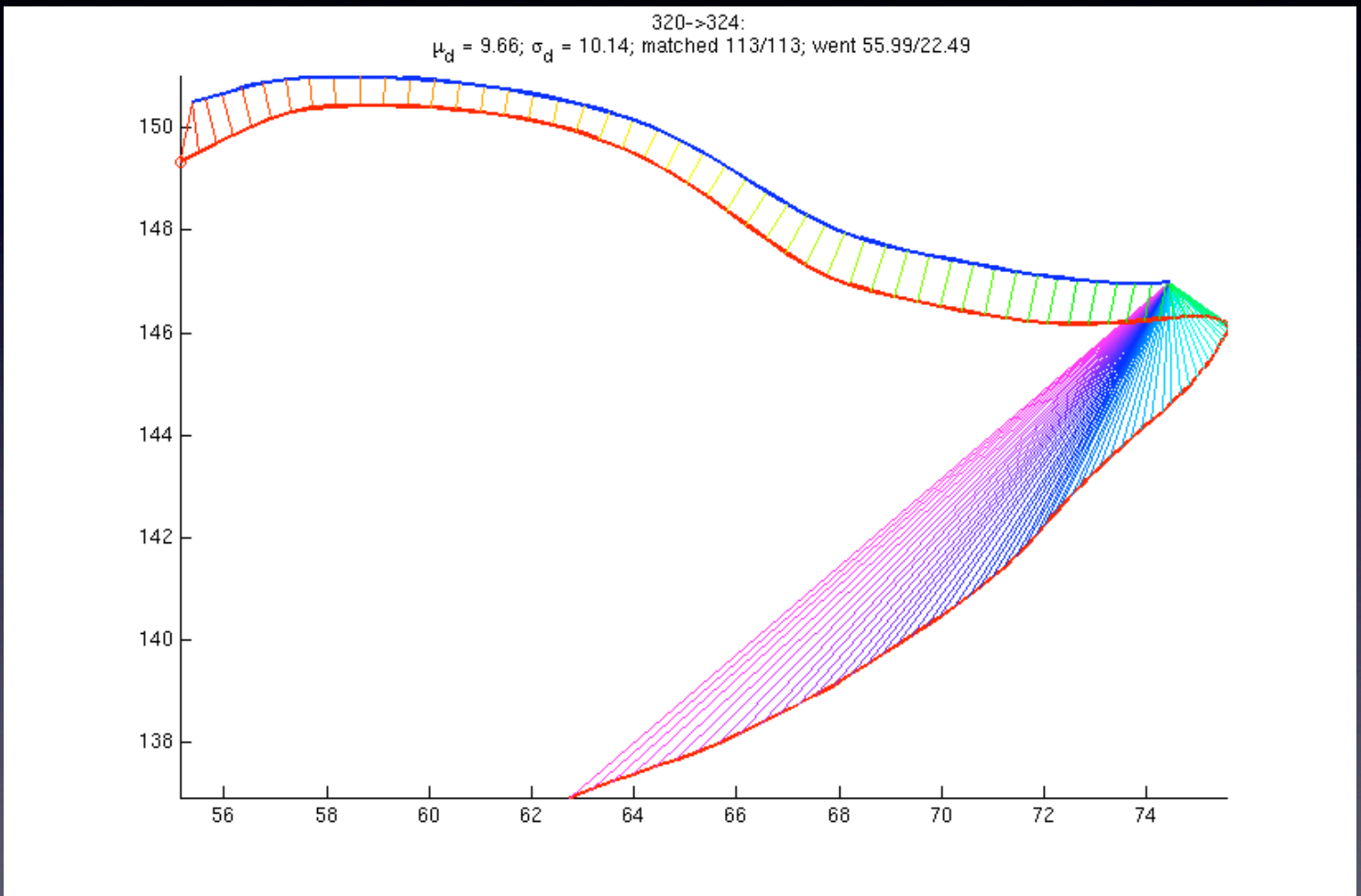
Trimming


Curves 320 and 324
150
148
146
144
142
140
138
56
58
60
62
64
66
68
70
72
74

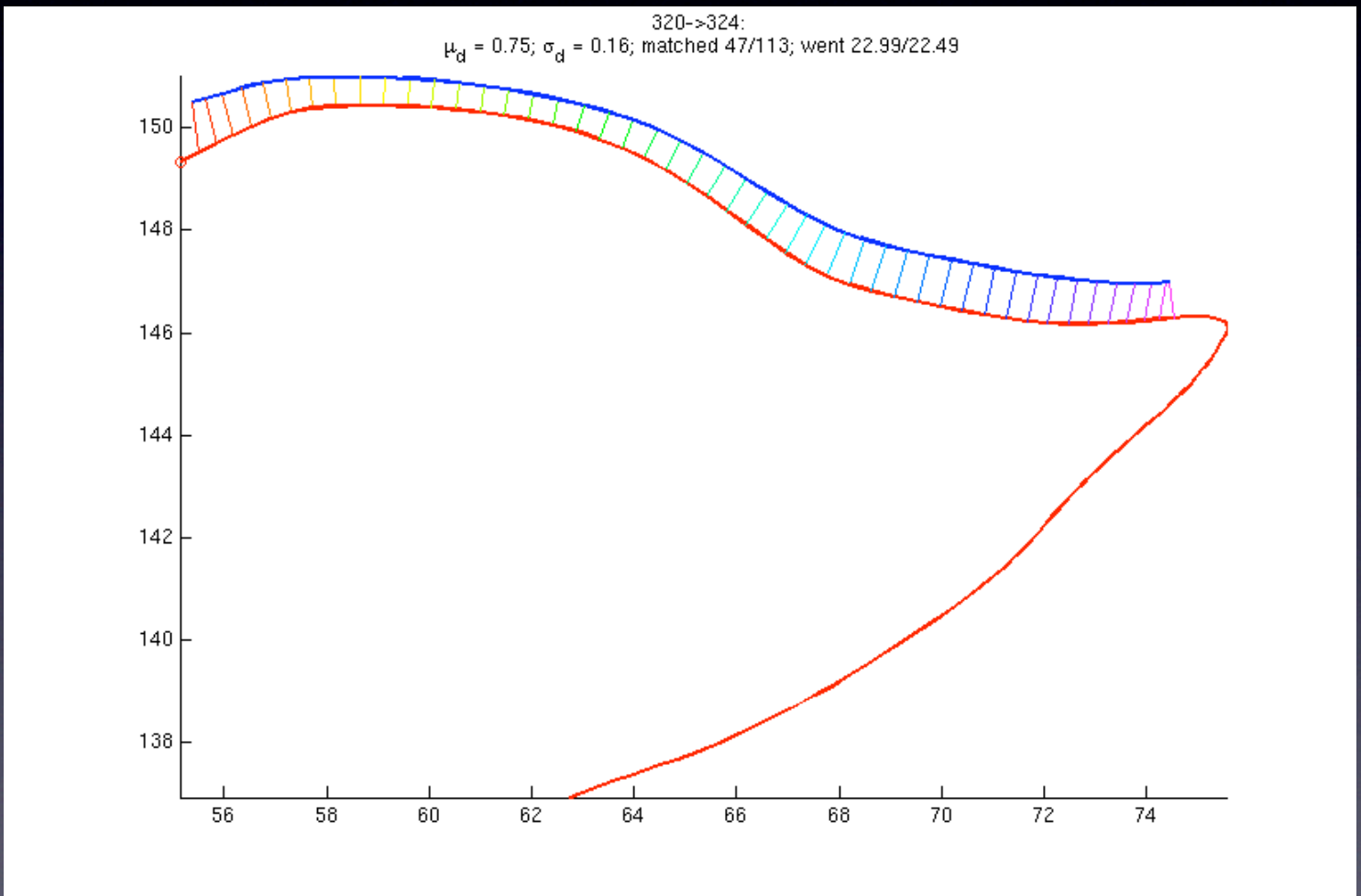
What we want


324->320:
μ_d = 0.74; σ_d = 0.16; matched 46/46; went 22.49/22.29
150
148
146
144
142
140
138
56
58
60
62
64
66
68
70
72
74

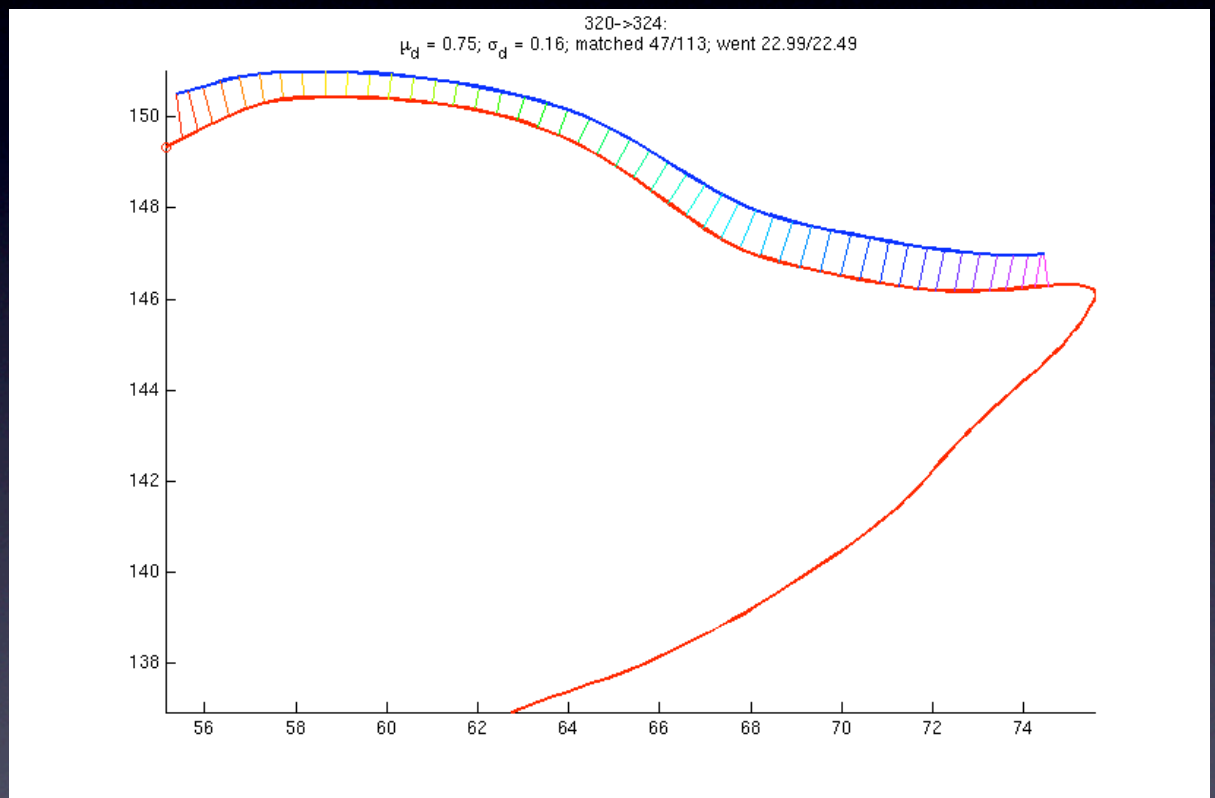
What we (can) get



Trimming Excess Matches



Trimming Excess Matches

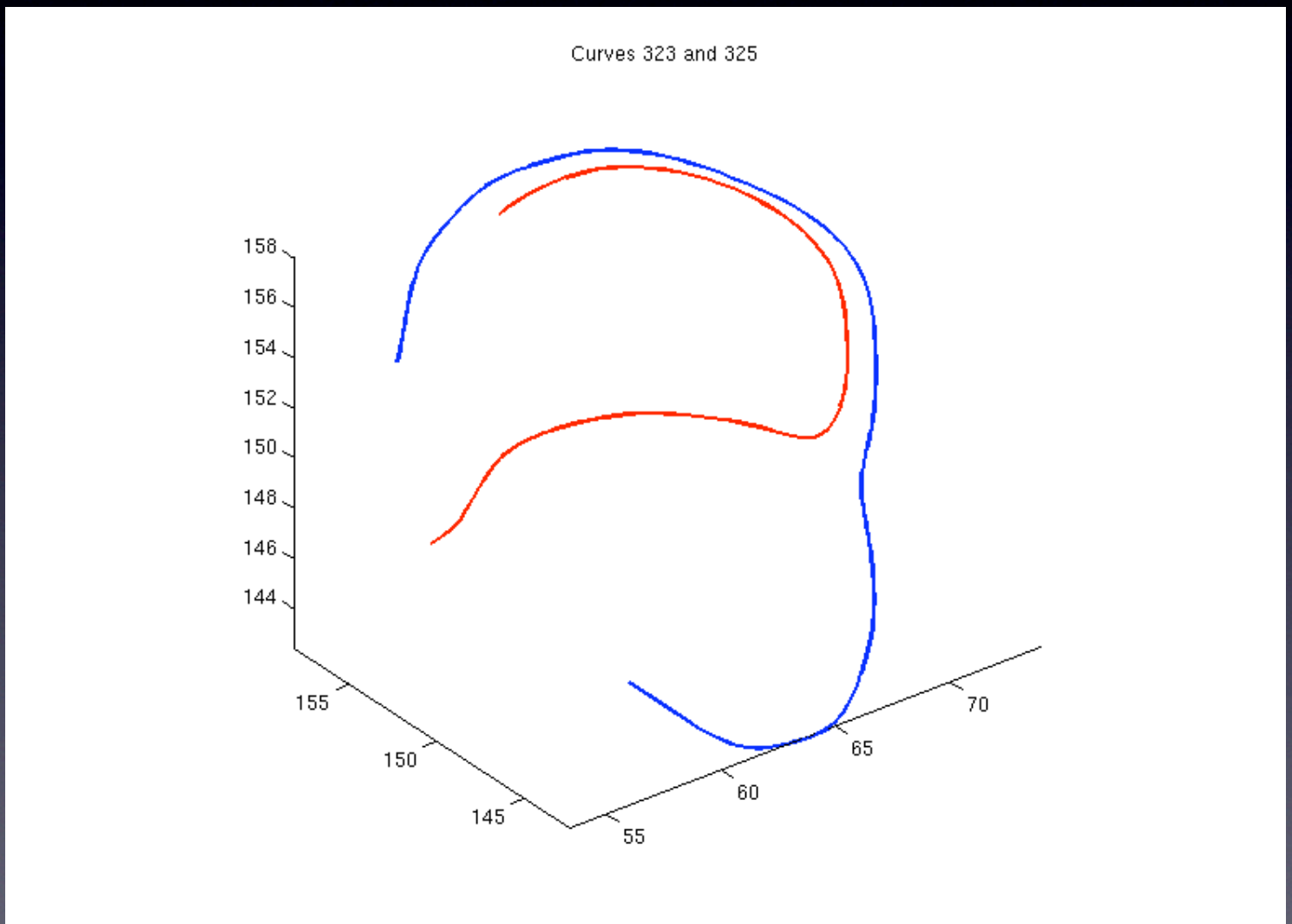


d(A,B) = mean distance in trimmed segment

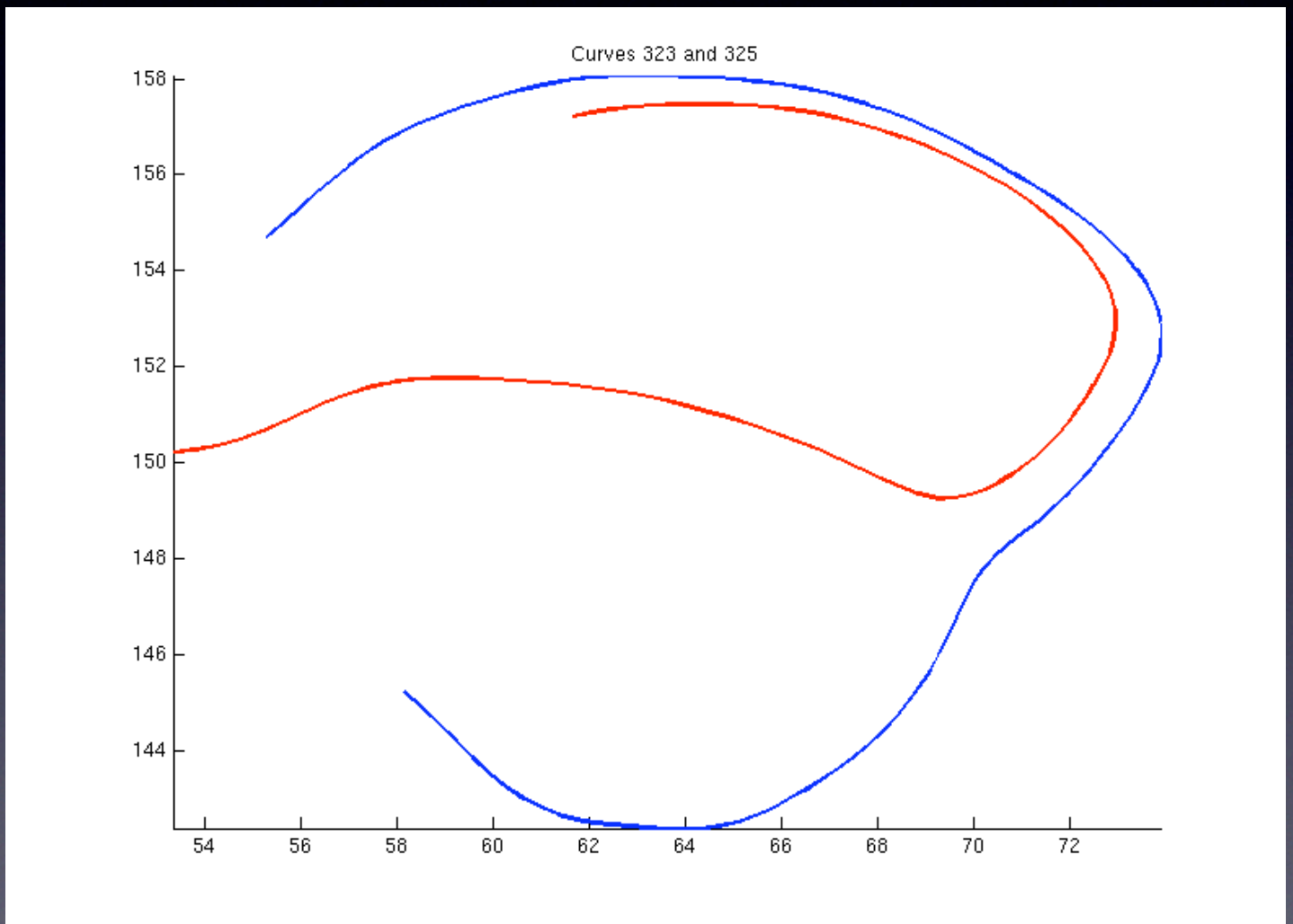
The New Measure

- Trimming
- Induced relative orientation
- Matching order
- Match length ratio

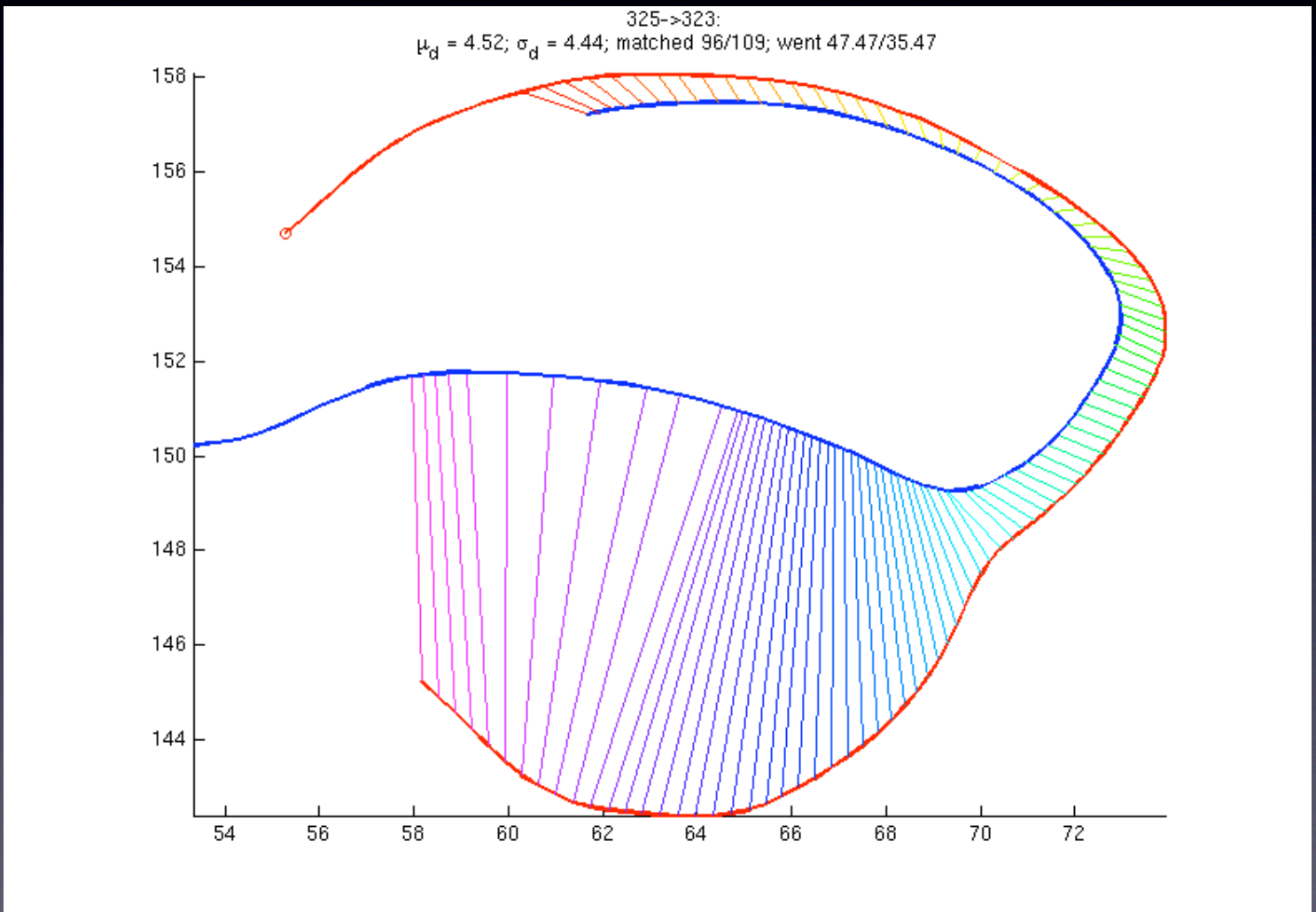
Relative Orientation



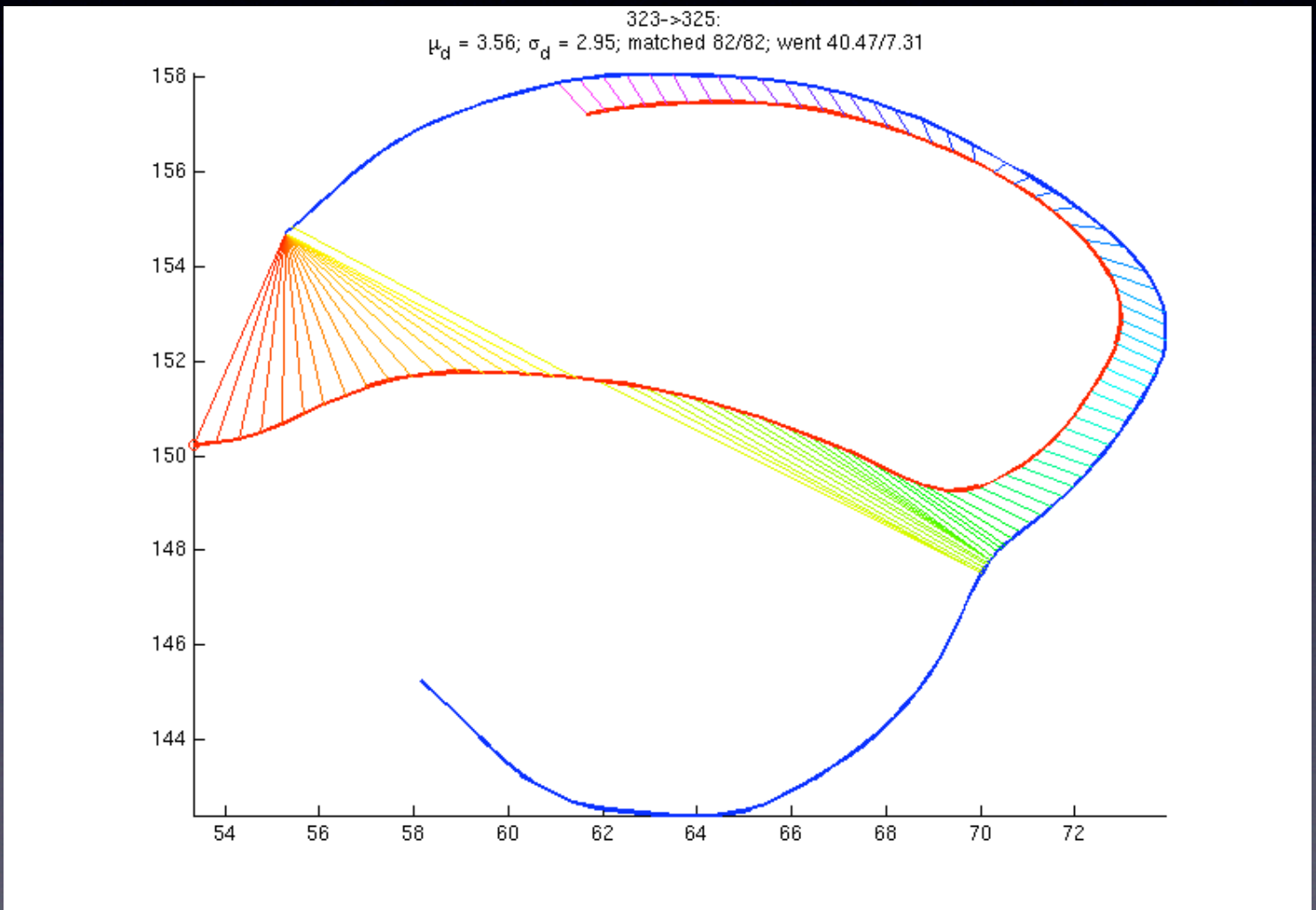
Relative Orientation



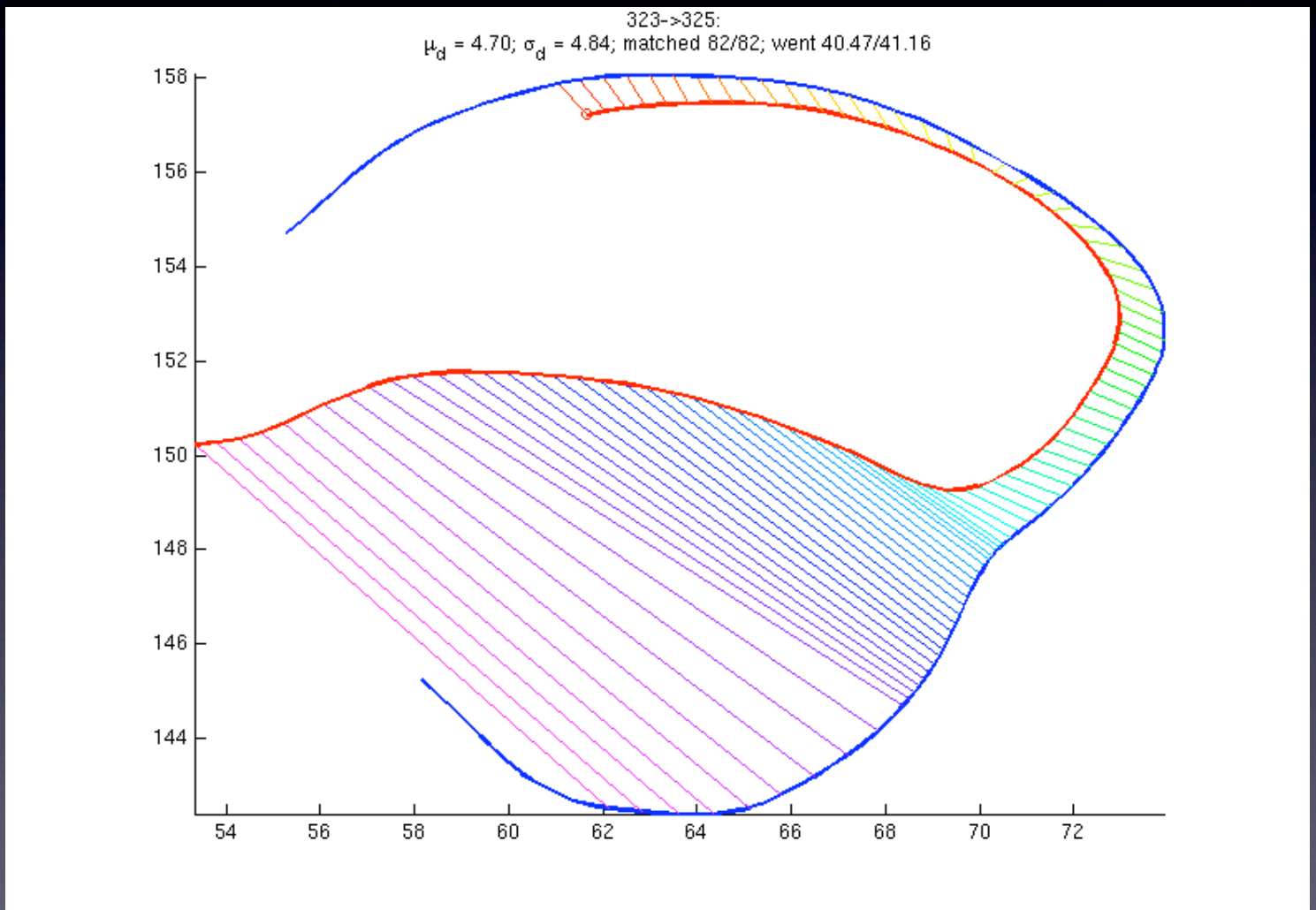
A Good Match


325->323:
μ_d = 4.52; σ_d = 4.44; matched 96/109; went 47.47/35.47
158
156
154
152
150
148
146
144
54
56
58
60
62
64
66
68
70
72

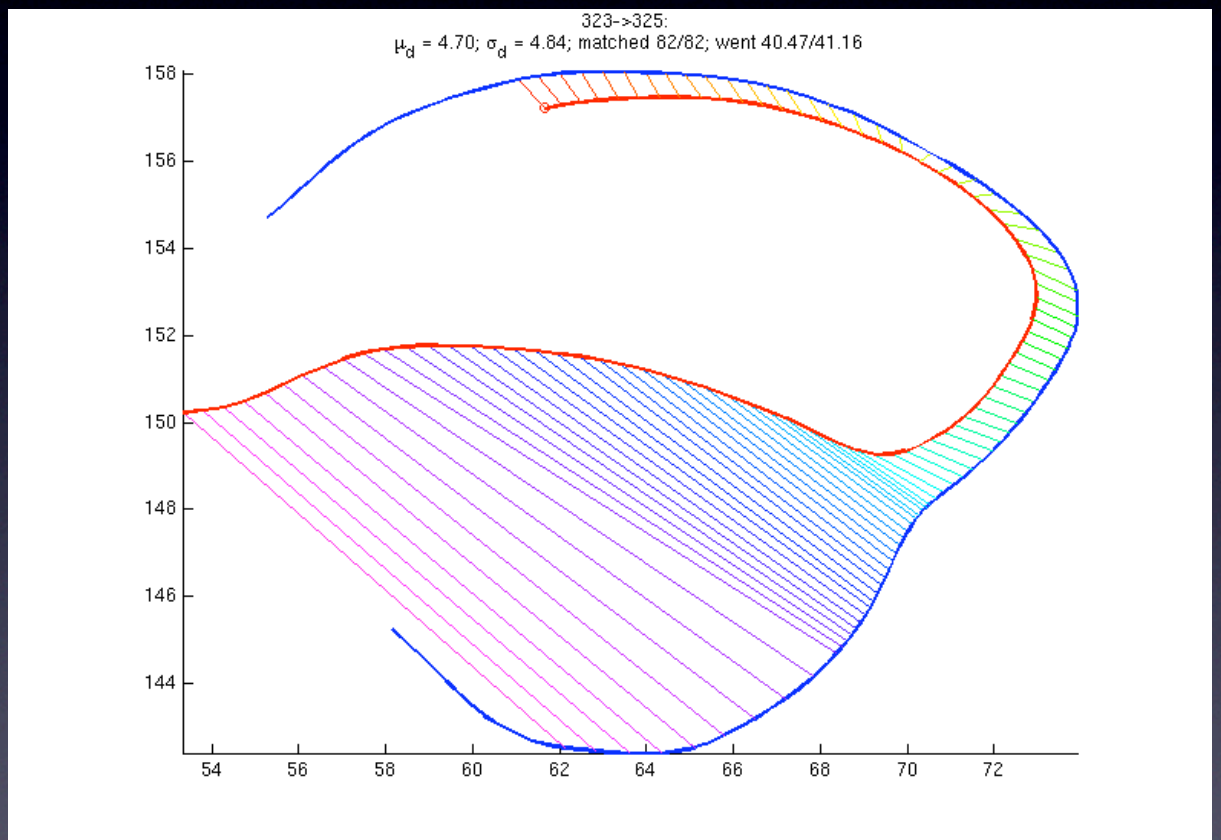
Not So Good


323->325:
μ_d = 3.56; σ_d = 2.95; matched 82/82; went 40.47/7.31
158
156
154
152
150
148
146
144
54
56
58
60
62
64
66
68
70
72

Forcing Relative Orientation



Forcing Relative Orientation

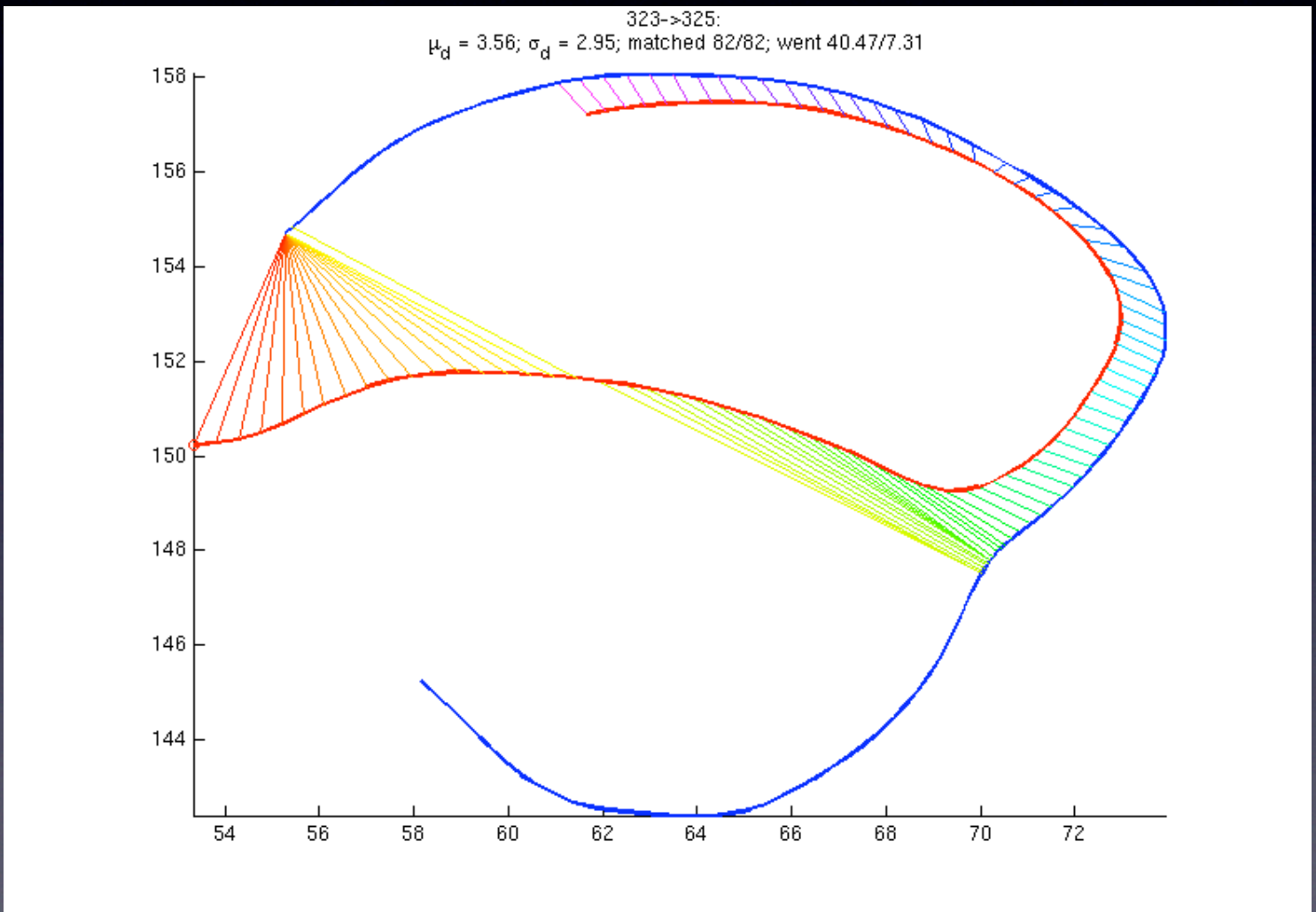


d(A,B) = mean distance in trimmed segment, with forced relative orientation

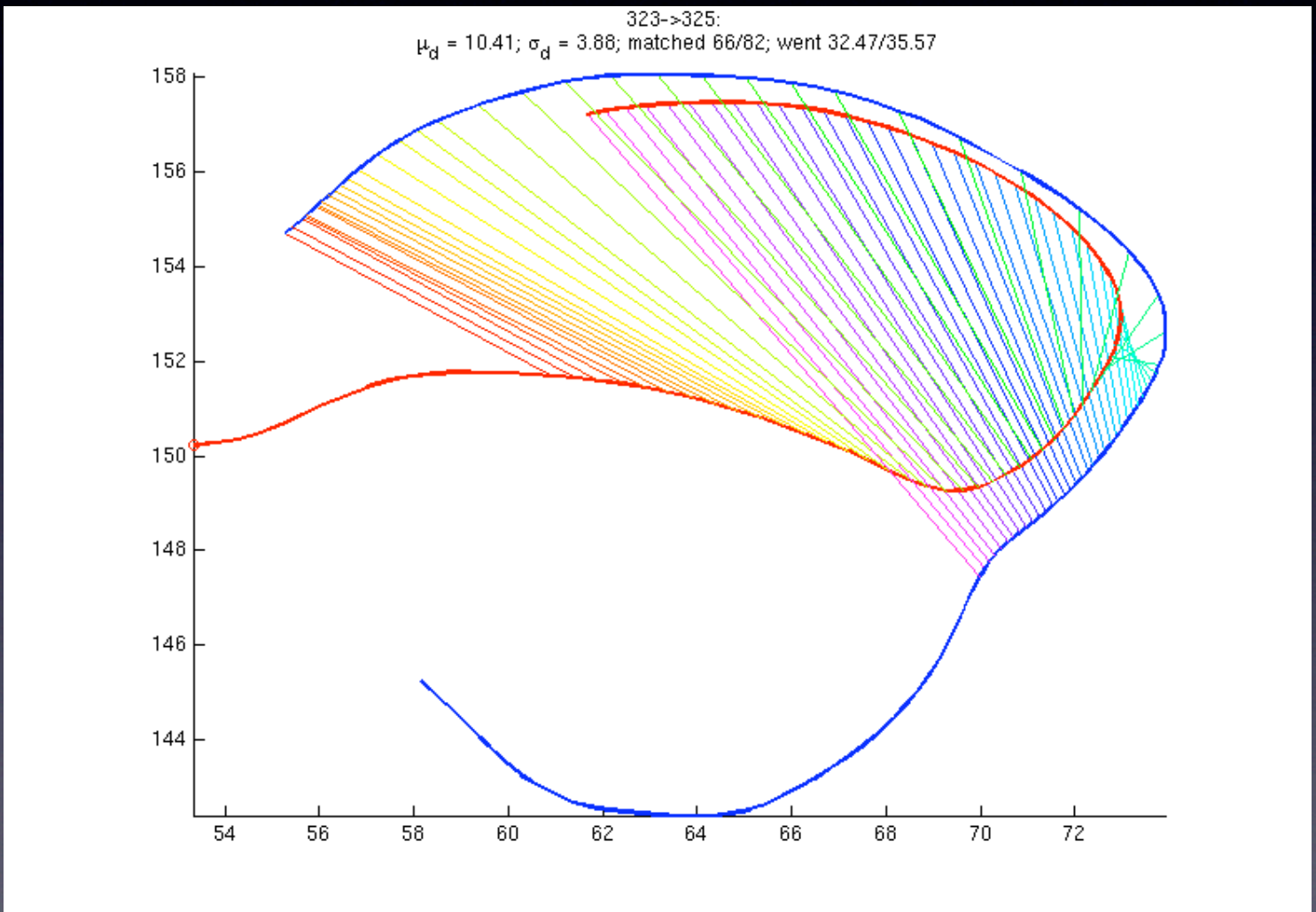
The New Measure

- Trimming
- Induced relative orientation
- Matching order
- Match length ratio

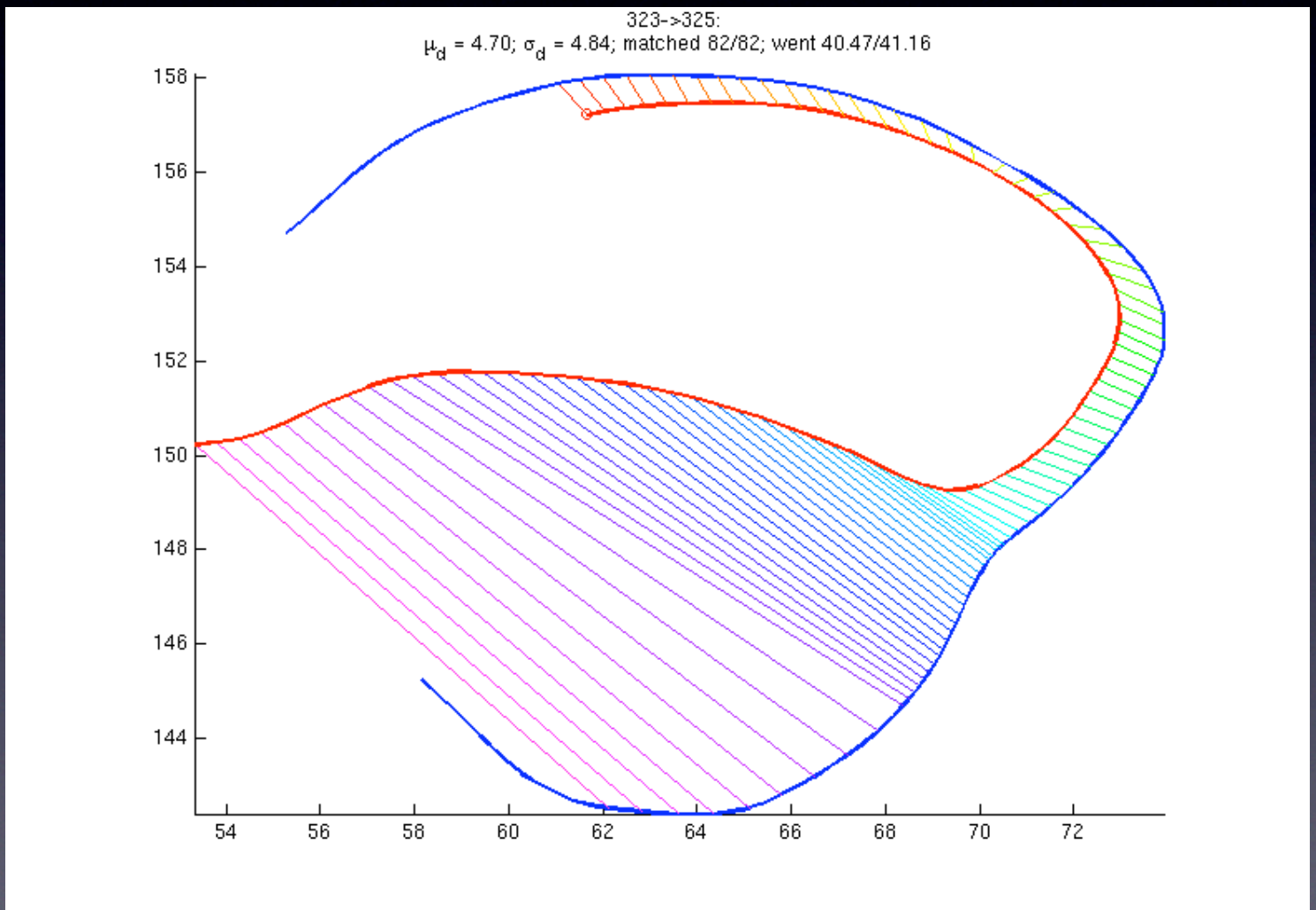
Without Orientation Forcing or Trimming



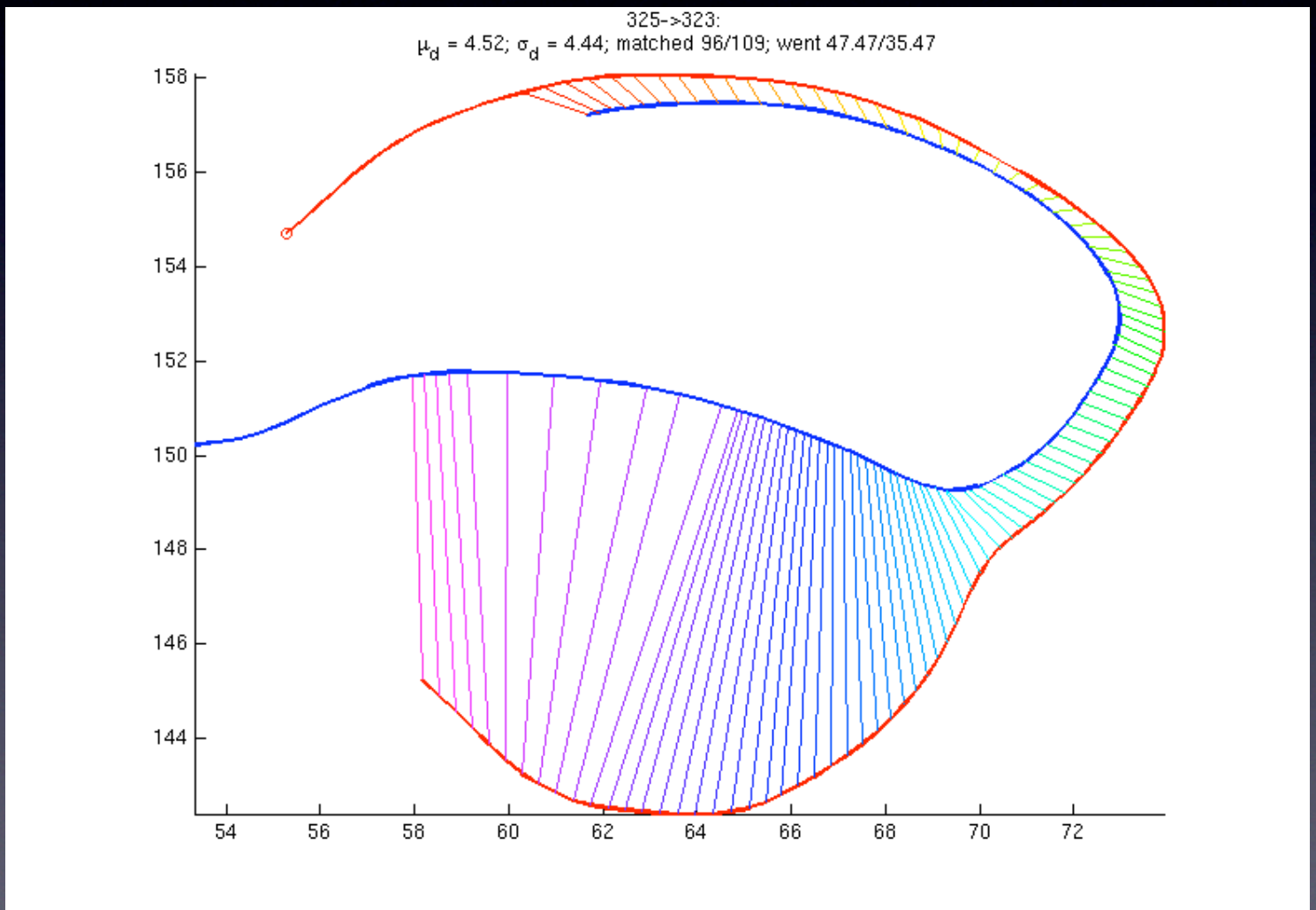
With Forced Orientation: Whoa!



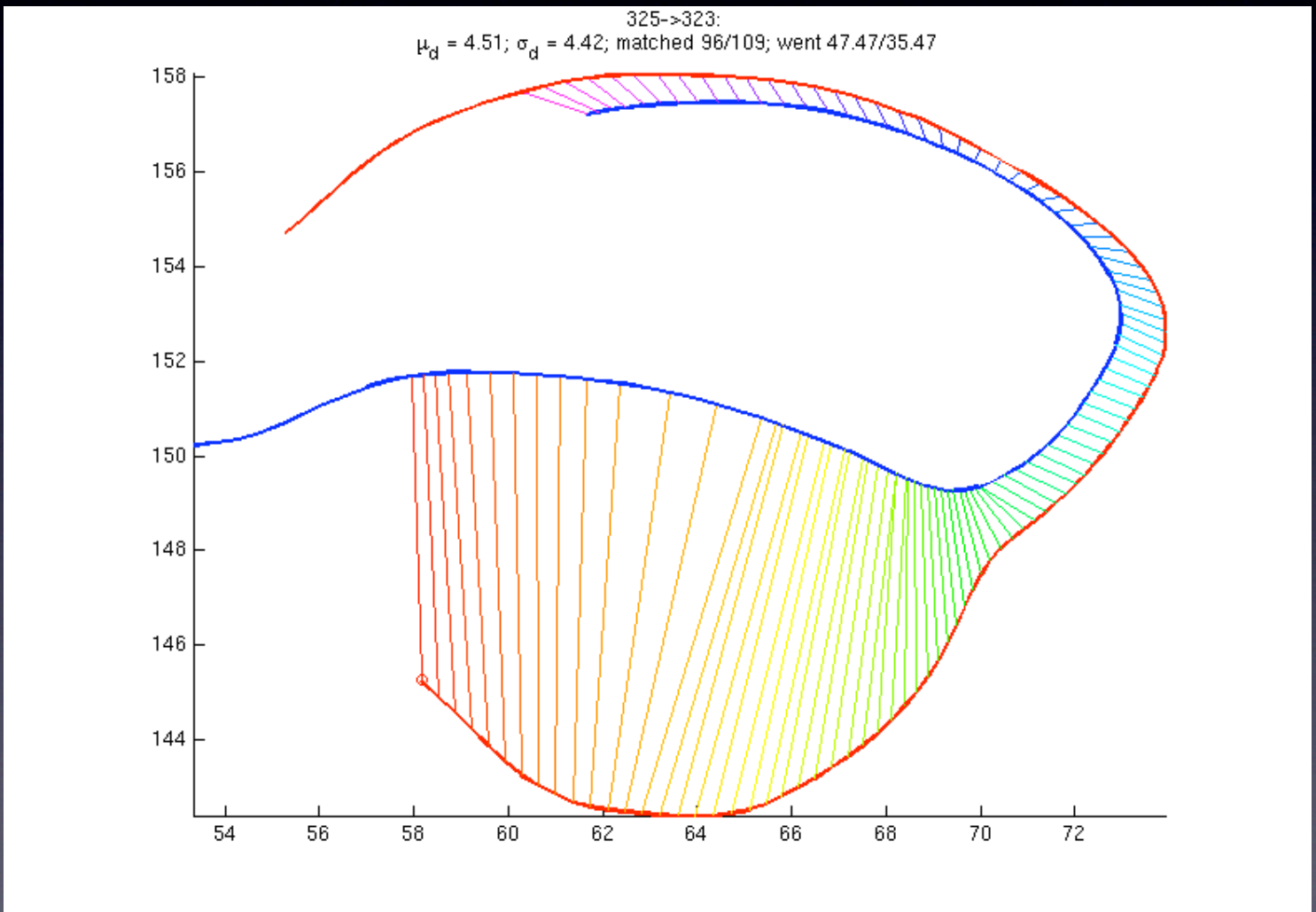
Matching Order



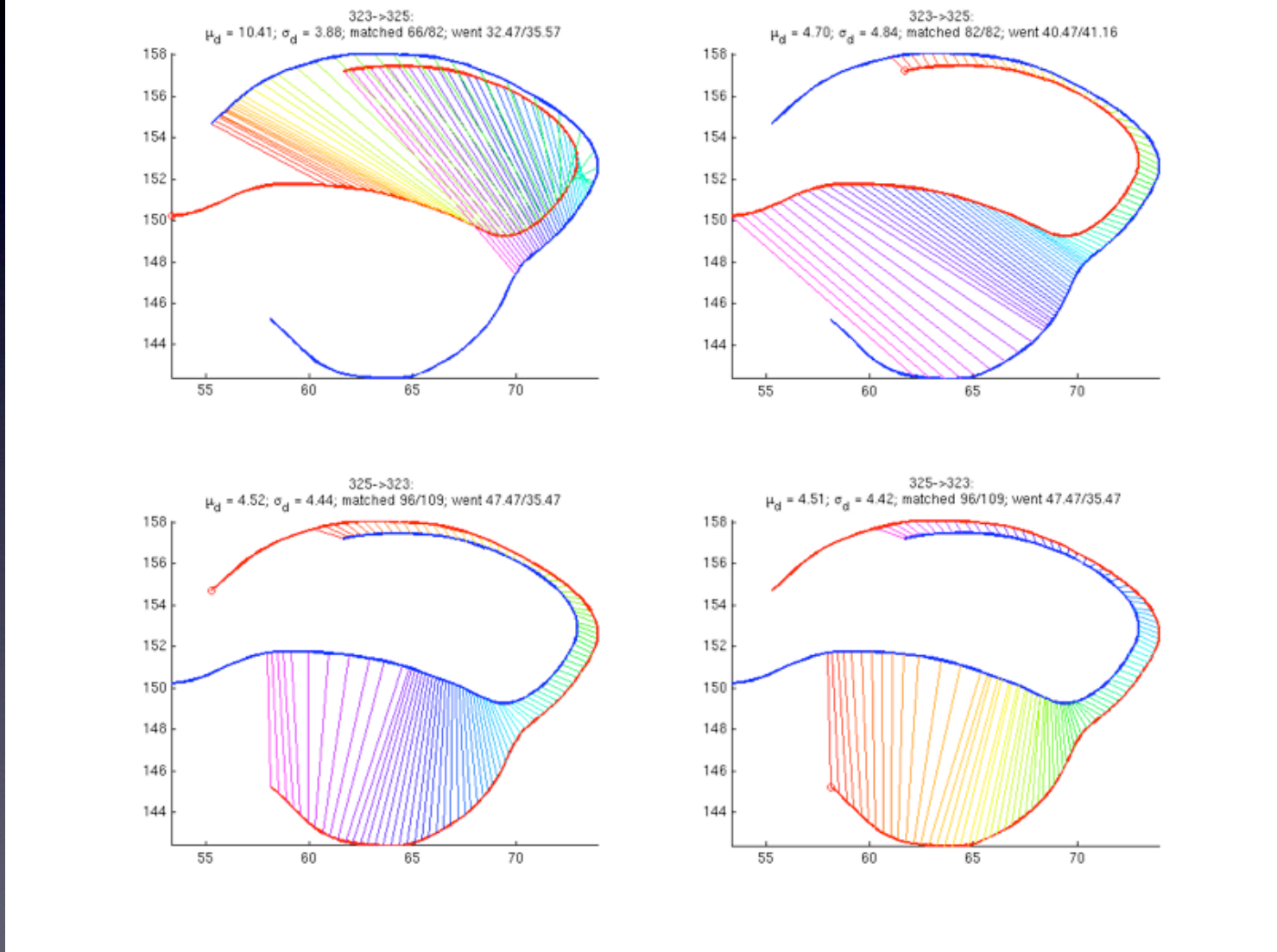
Matching Order



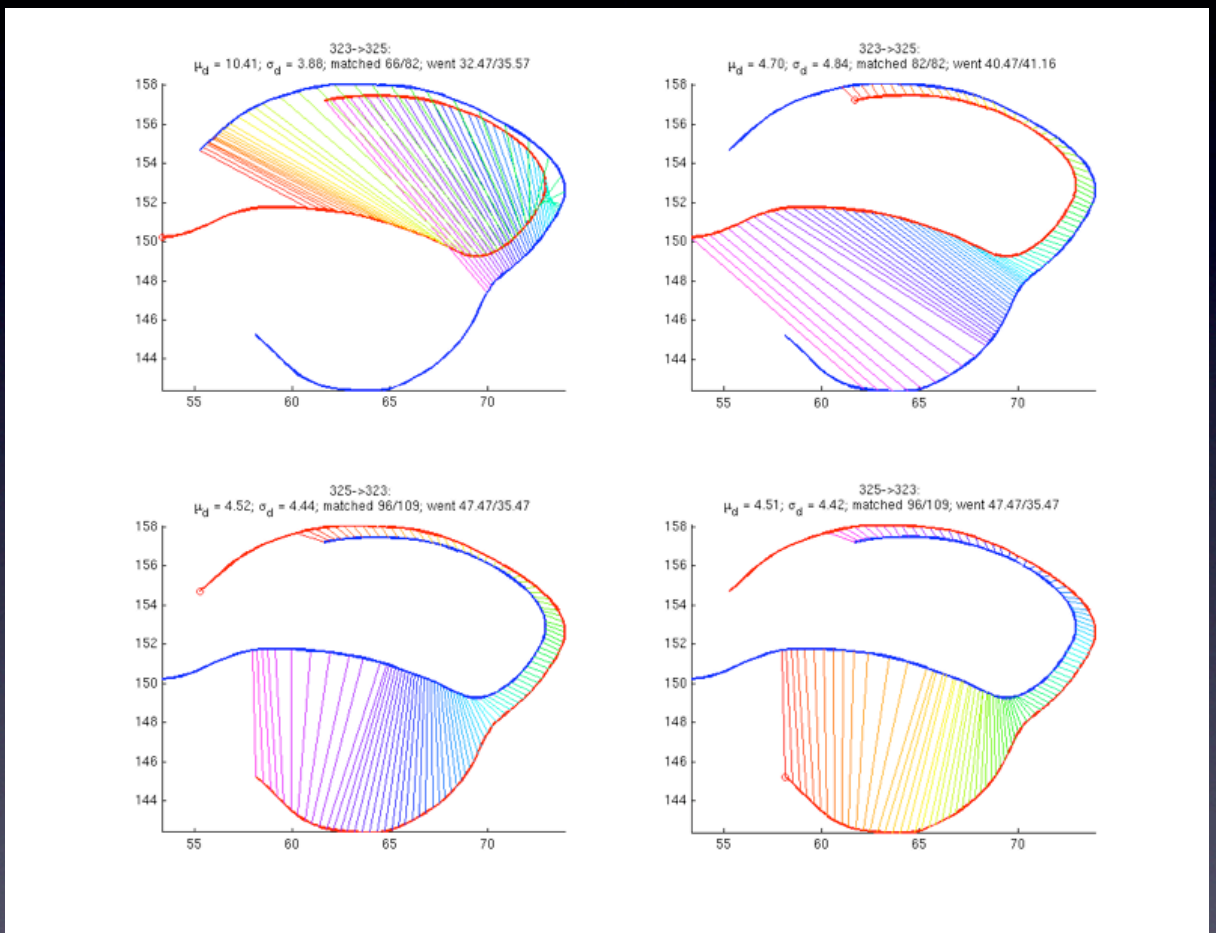
Matching Order



All Matching Orders



All Matching Orders

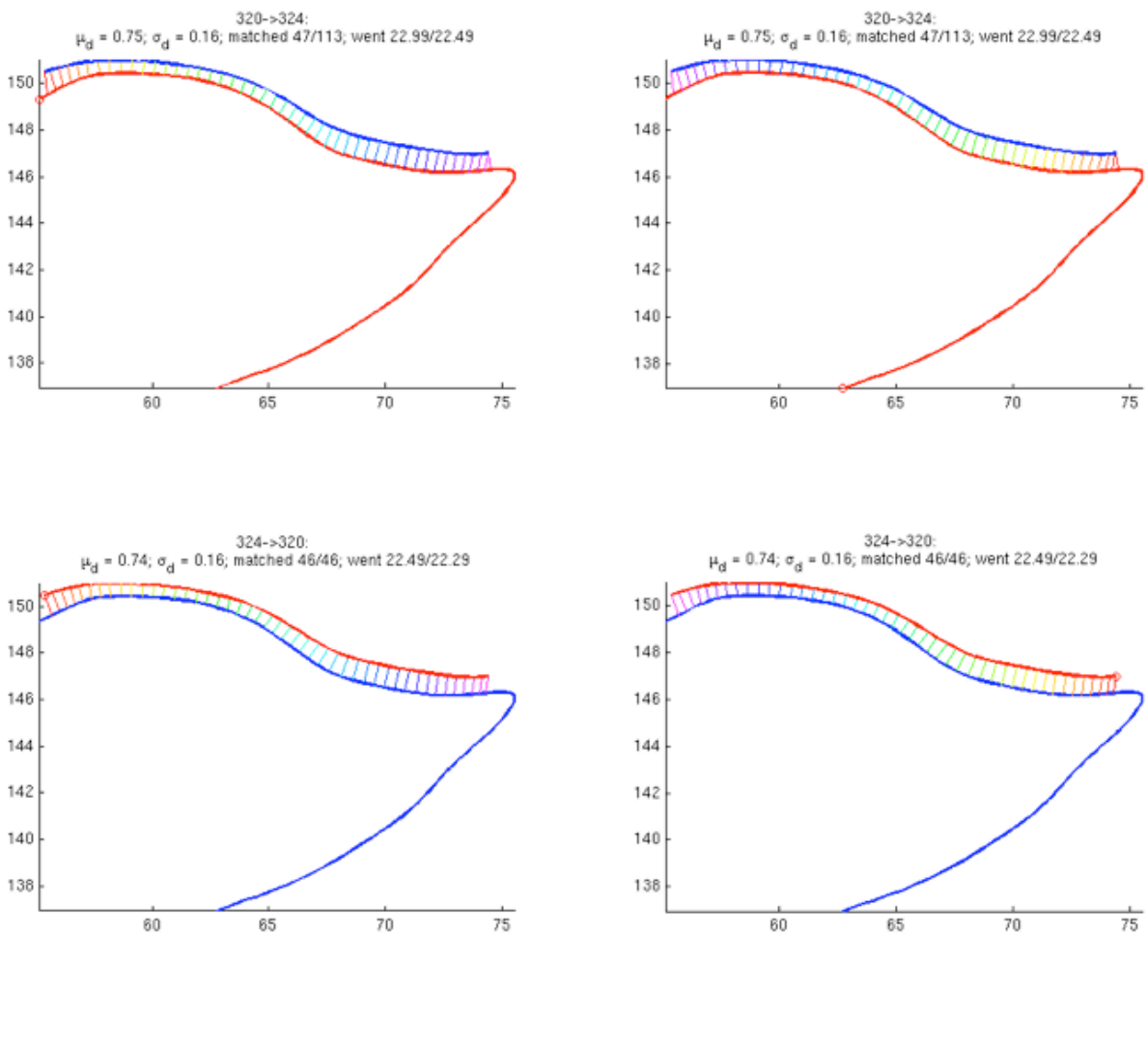


d(A,B) = min mean distance in trimmed segment with forced orientation, over all four matching orders

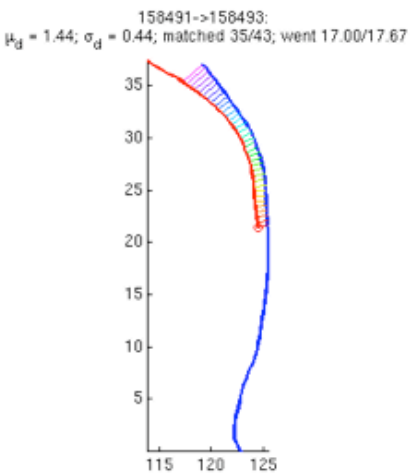
The New Measure

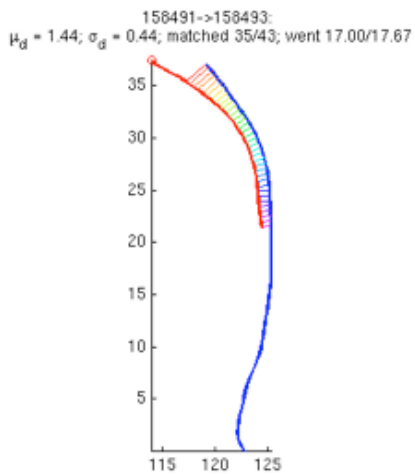
- Trimming
- Induced relative orientation
- Matching order
- Match length ratio

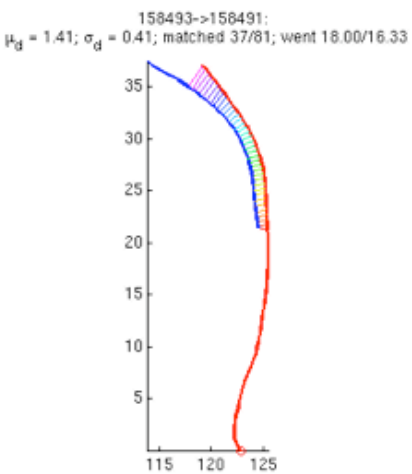
Success!

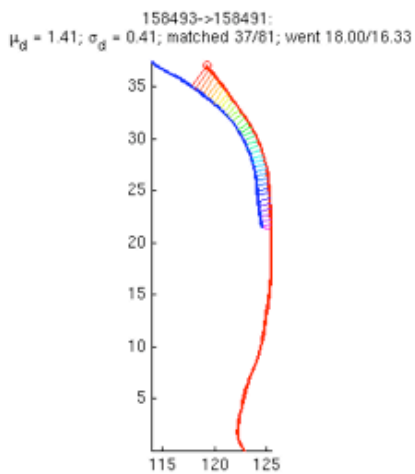

320->324:
μ_d = 0.75; σ_d = 0.16; matched 47/113; went 22.99/22.49
320->324:
μ_d = 0.75; σ_d = 0.16; matched 47/113; went 22.99/22.49
324->320:
μ_d = 0.74; σ_d = 0.16; matched 46/46; went 22.49/22.29
324->320:
μ_d = 0.74; σ_d = 0.16; matched 46/46; went 22.49/22.29
150
148
146
144
142
140
138
60
65
70
75

Success!

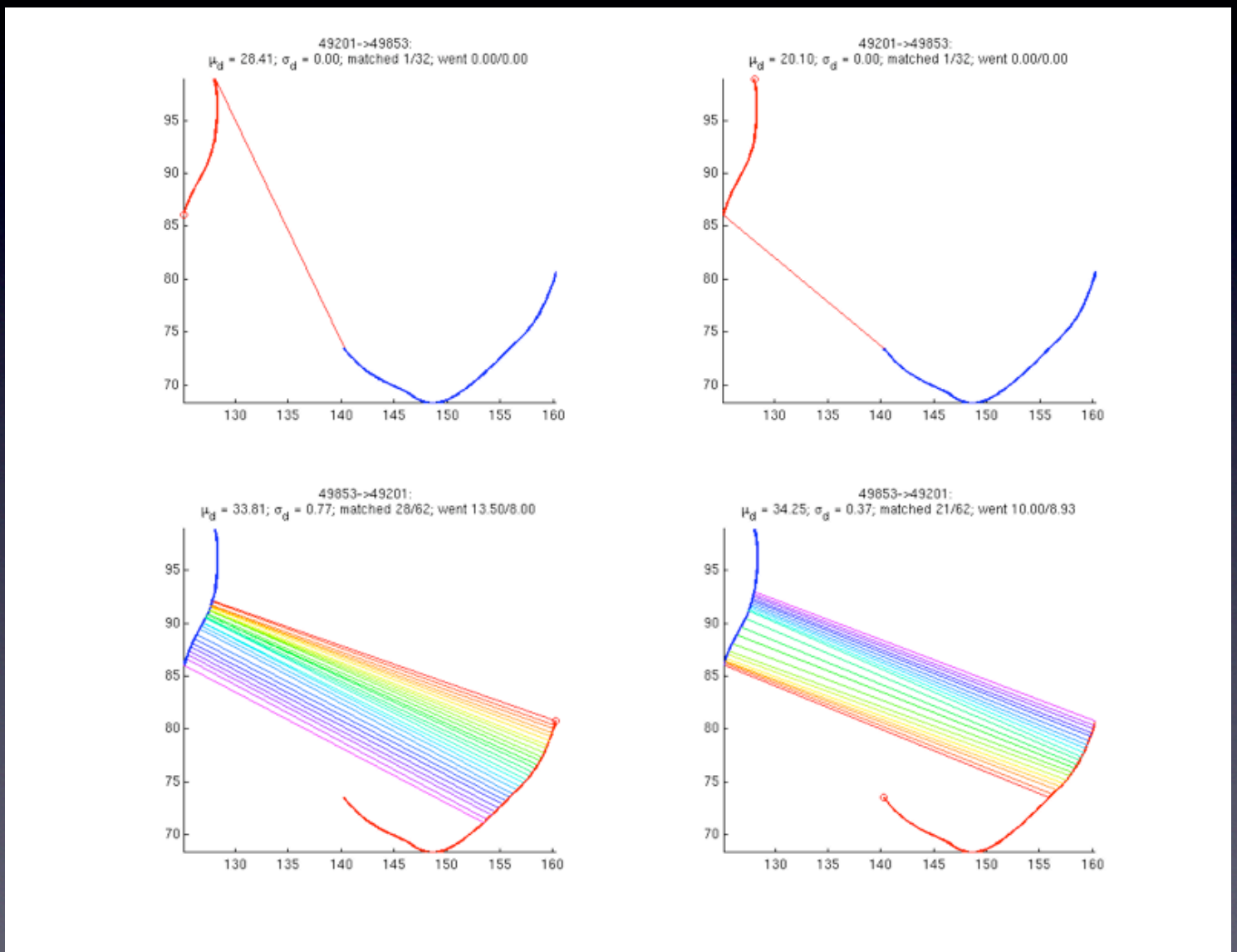

158491->158493:
μ_d = 1.44; σ_d = 0.44; matched 35/43; went 17.00/17.67
35
30
25
20
15
10
5
115
120
125


158491->158493:
μ_d = 1.44; σ_d = 0.44; matched 35/43; went 17.00/17.67
35
30
25
20
15
10
5
115
120
125

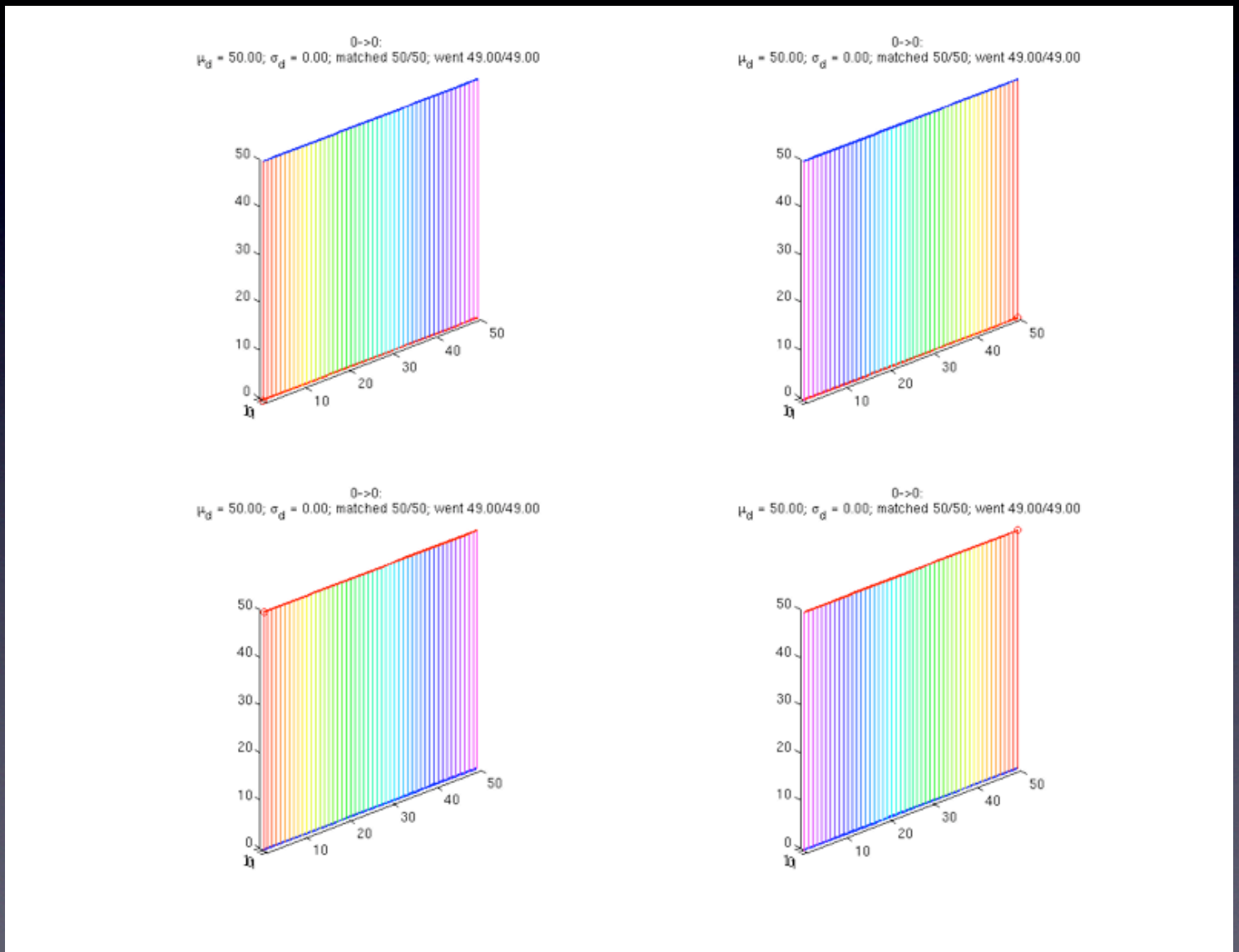

158493->158491:
μ_d = 1.41; σ_d = 0.41; matched 37/81; went 18.00/16.33
35
30
25
20
15
10
5
115
120
125


158493->158491:
μ_d = 1.41; σ_d = 0.41; matched 37/81; went 18.00/16.33
35
30
25
20
15
10
5
115
120
125

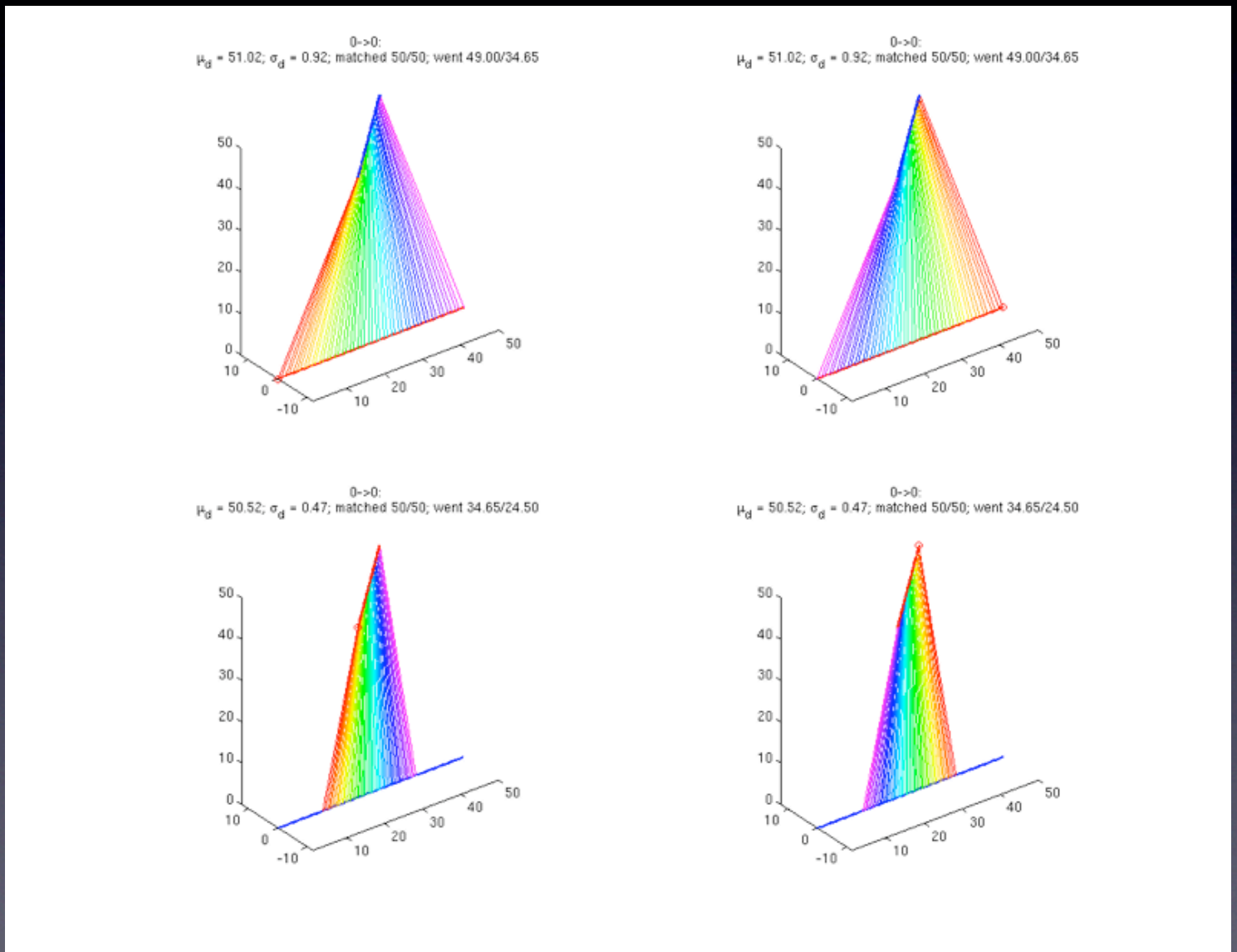
Still Not Quite Right



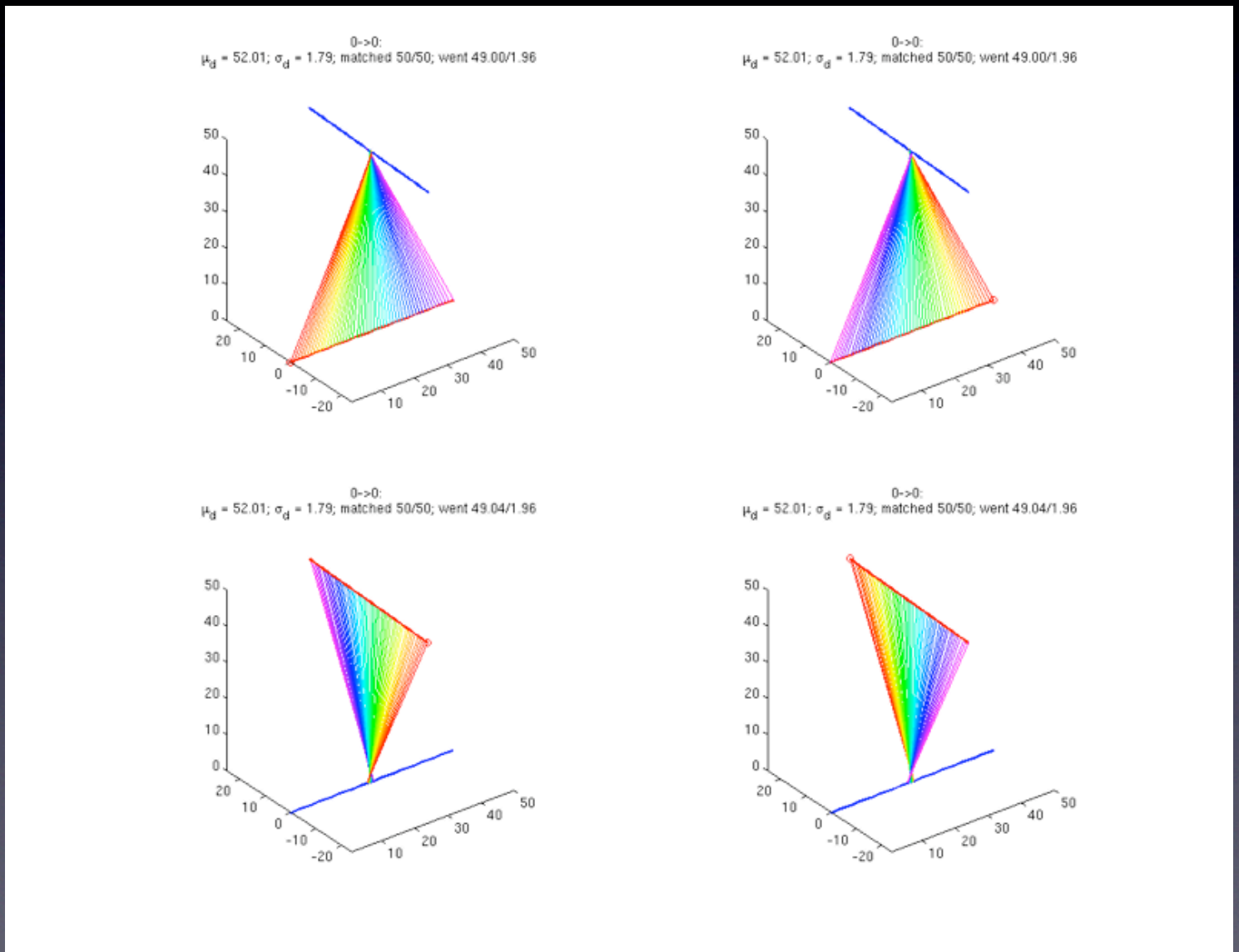
An Extreme Case



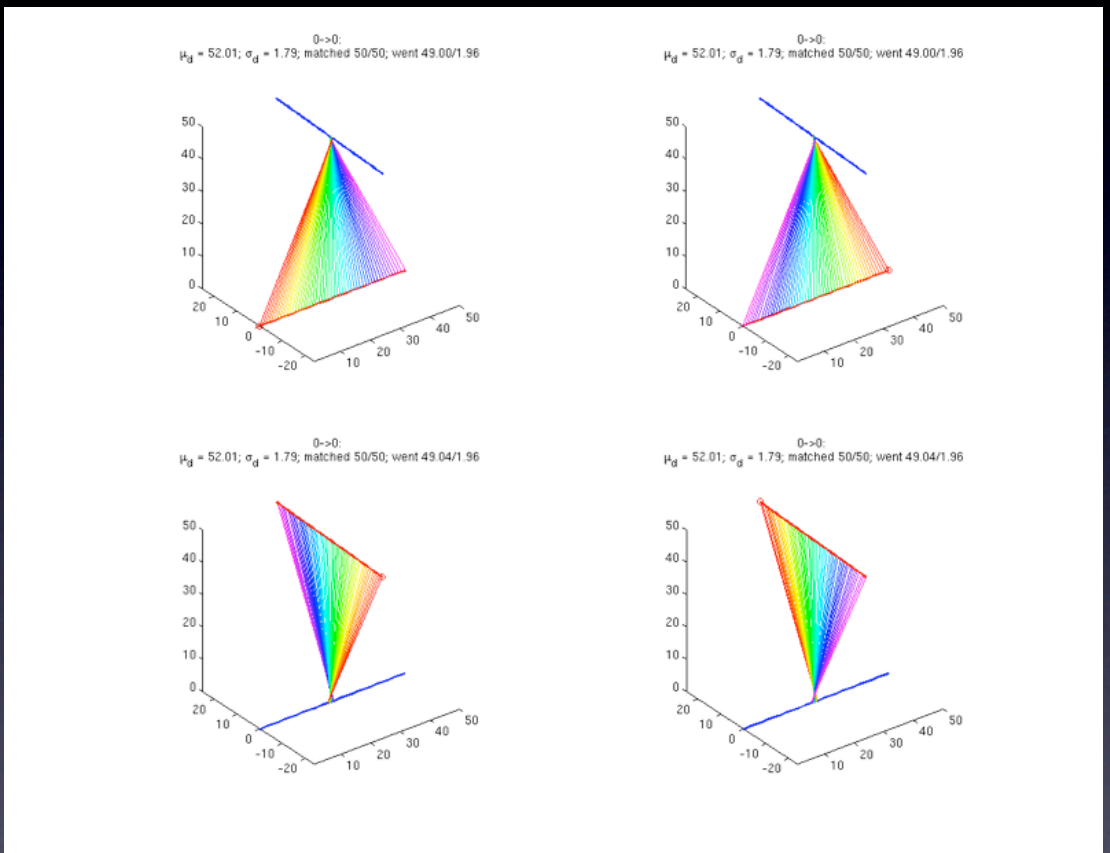
An Extreme Case



An Extreme Case



Match Length Ratio



- Infinite for perpendicular segments
- Infinite for end-to-end segments

The New Measure

- d(A,B) = minimum over all four matching orders of:
  - Mean distance between points
  - In trimmed segment
  - With forced relative orientation
  - Multiplied by the match length ratio

The New Measure

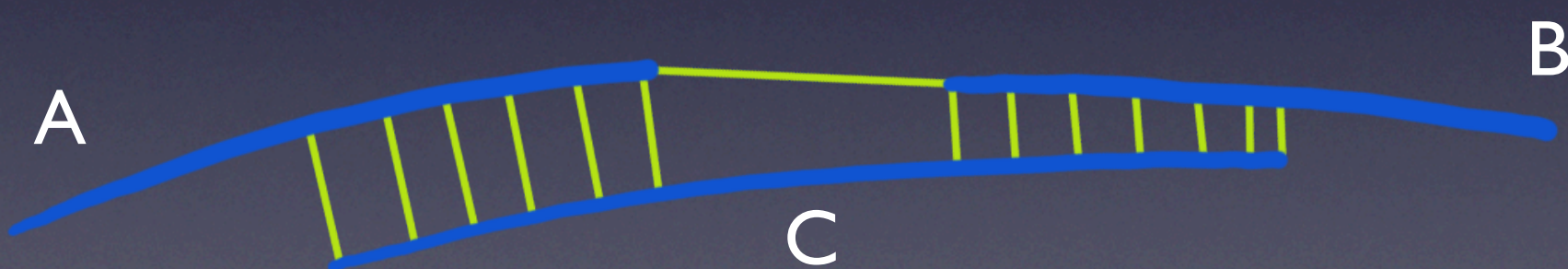
- Detects overlapping segment
- Penalizes skewed curves

The New Measure

- Exploits agglomerative clustering with single-link cluster distance

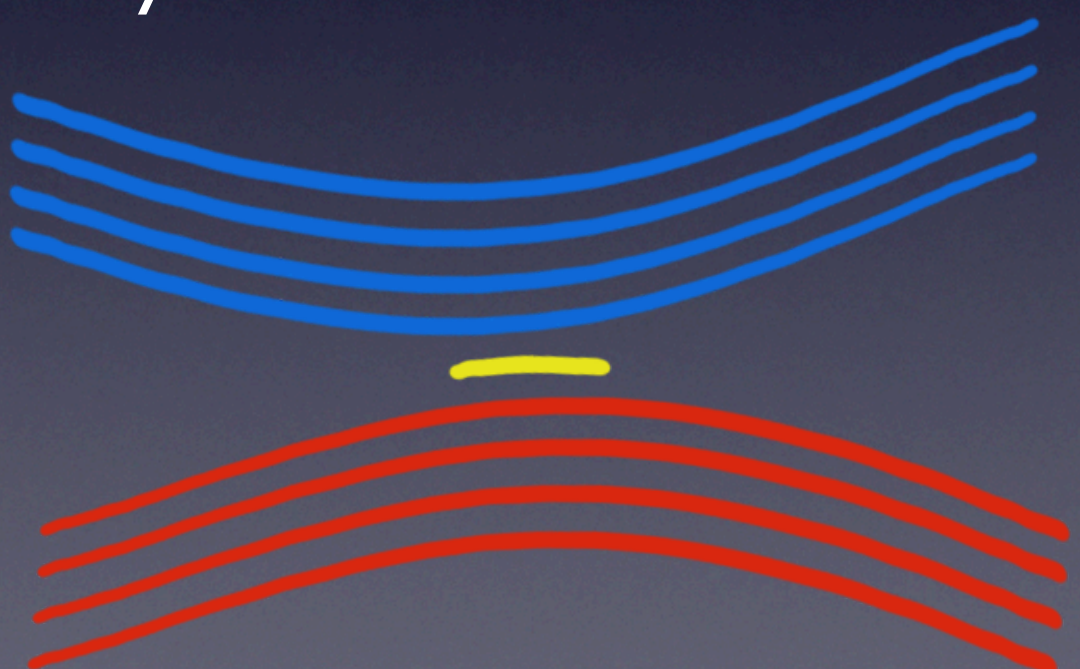
The New Measure

- Exploits agglomerative clustering with single-link cluster distance
- $d(A,C) < \epsilon$, $d(B,C) < \epsilon$
- $d(A,B) = \infty$ and yet $A \sim_\epsilon B$



A Failure Case

- A single short segment can join two otherwise strongly separated clusters
- Post-processing the clustering is likely necessary



Overview

- Motivation
- Agglomerative clustering
- **Inter-curve similarity measures**
- Efficient clustering implementation

Overview

- Motivation
- Agglomerative clustering
- Inter-curve similarity measures
- **Efficient clustering implementation**

What We'd Like

1. Compute all pairwise curve similarities
2. Find highest similarity
3. Merge those clusters
4. GOTO 2

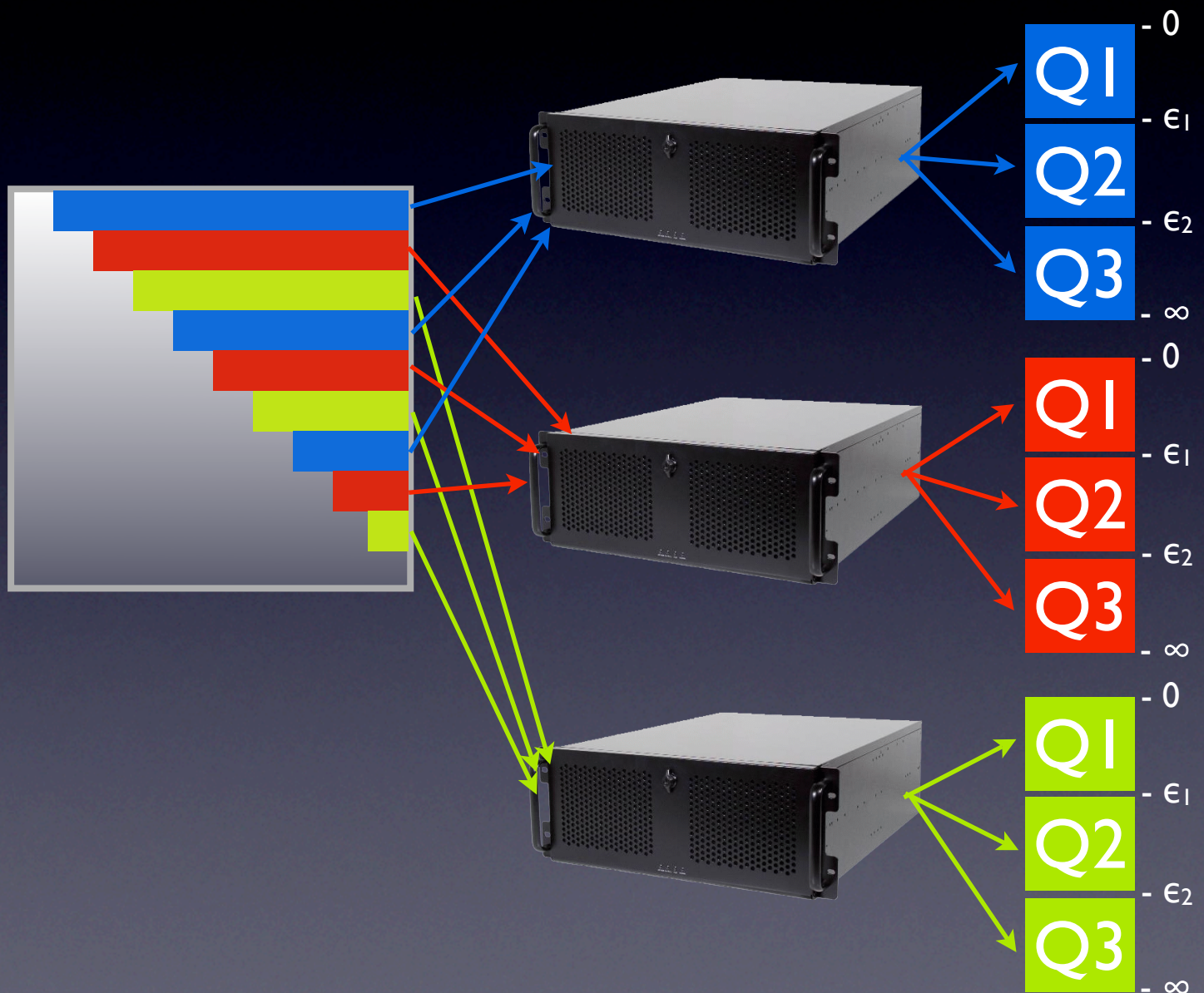
What We'd Like

- Can compute 1600 similarities / s
- But we have 300k curves!
  - 45 billion similarities, 400 GB, 11.5 months
- Even decimated, 30k curves
  - 450 million similarities, 5.1 GB, 83 hours

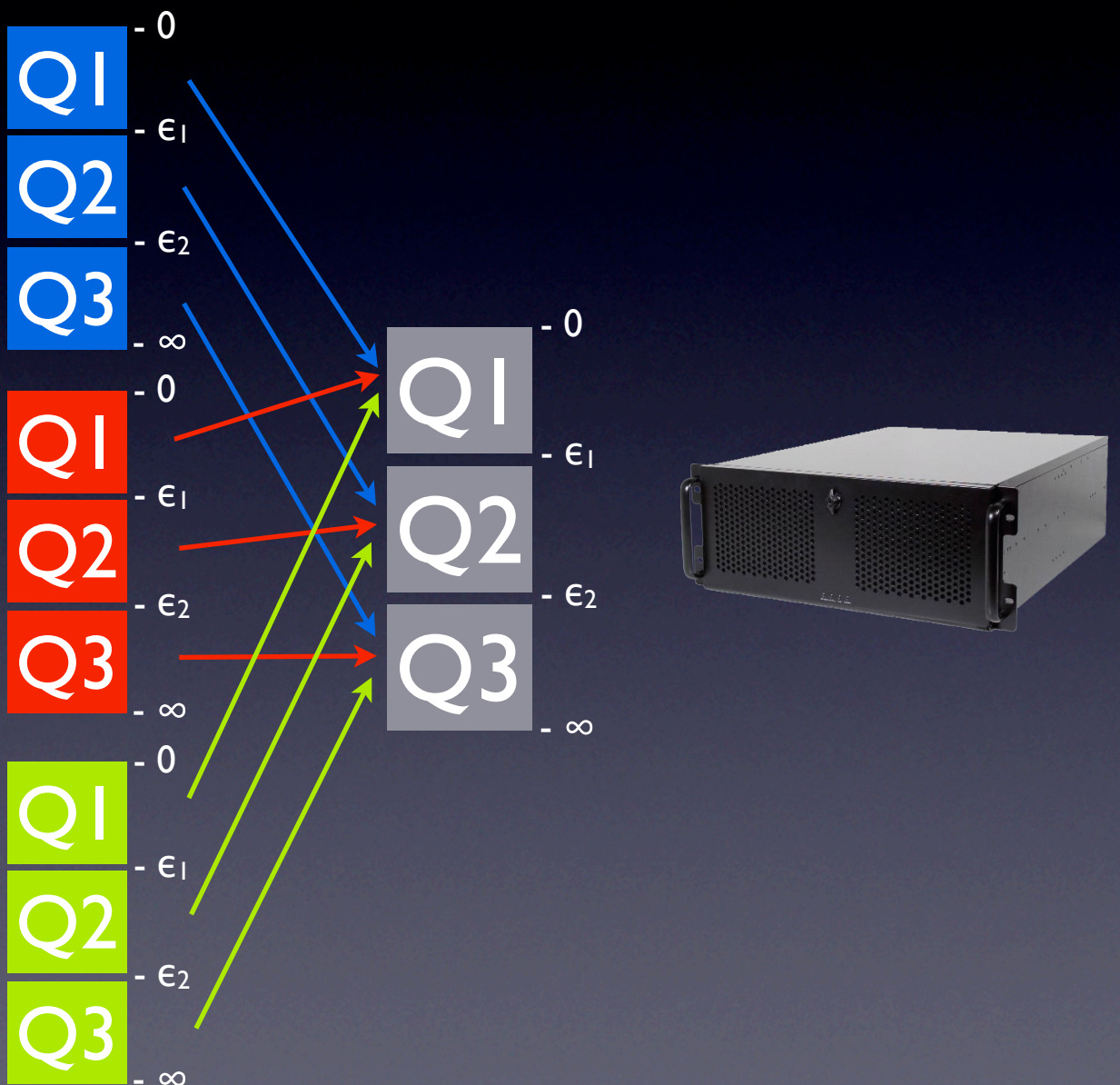
What we need

- Parallelism
- Only the highest elements of the similarity matrix in memory

Efficient Clustering



Efficient Clustering



Efficient Clustering

- 30k curves with 13 hosts
- 6.5 hours, 5.1 GB total
- 10 priority queues of increasing size: 8MB, 22MB, 56MB, 130MB... 1.1GB
- Only one priority queue must be in memory at a time to do clustering

Thanks!

- David Laidlaw, my advisor
- Evren Ozarslan for inviting me

Questions?

- Motivation
- Agglomerative clustering
- Inter-curve similarity measures
- Efficient clustering implementation