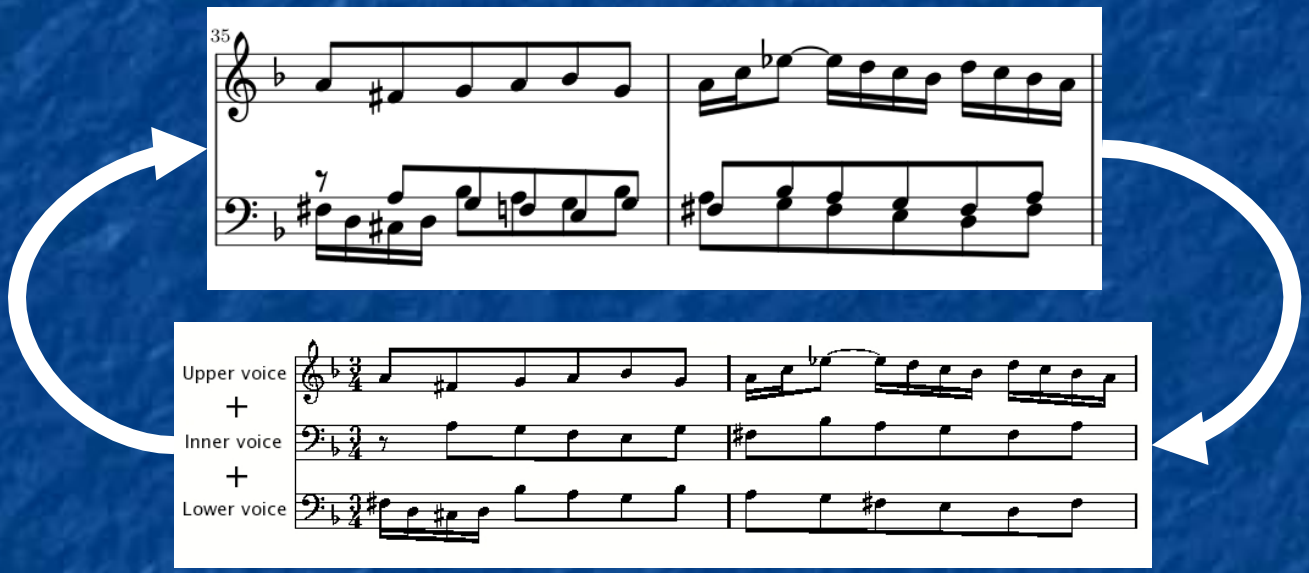
Voice separation in polyphonic music: A data-driven approach



Anna Jordanous

Music Informatics Research Centre
University of Sussex, Brighton, UK

- What is voice separation?
- Have there been attempts to automate voice separation?
- What solution do I propose?
- What results have I achieved with my approach?
- What conclusions can be drawn from this work?

Polyphony

Musical texture in two or more (though usually at least three) relatively independent parts.

The Oxford Companion to Music

A term used to designate various important categories in music: namely, music in more than one part, music in many parts, and the style in which all or several of the musical parts move to some extent independently.

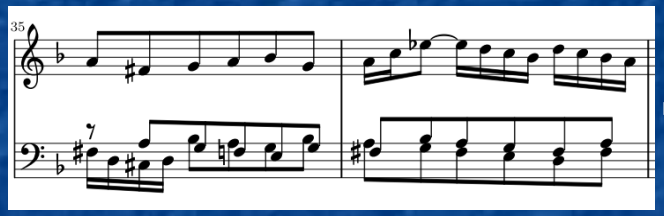
Grove Music Online

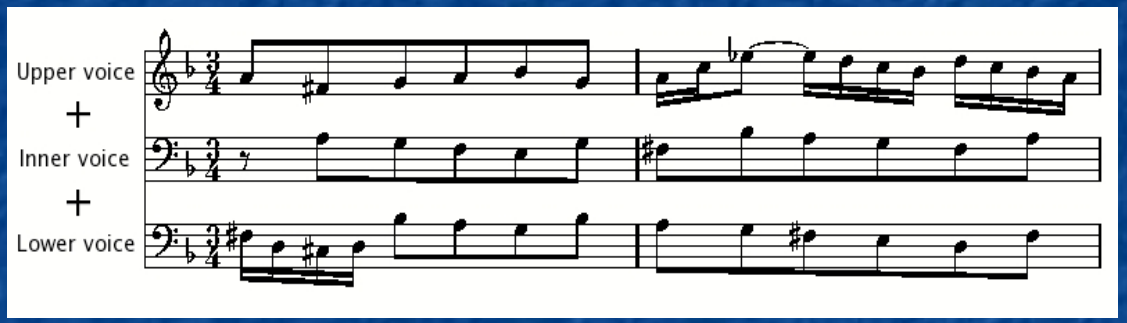
Many sounds. Mus. in which several simultaneous v. or instr. parts are combined contrapuntally, as opposed to monophonic mus. (single melody) or homophonic mus. (one melodic line, the other parts acting as acc.).

The Oxford Dictionary of Music

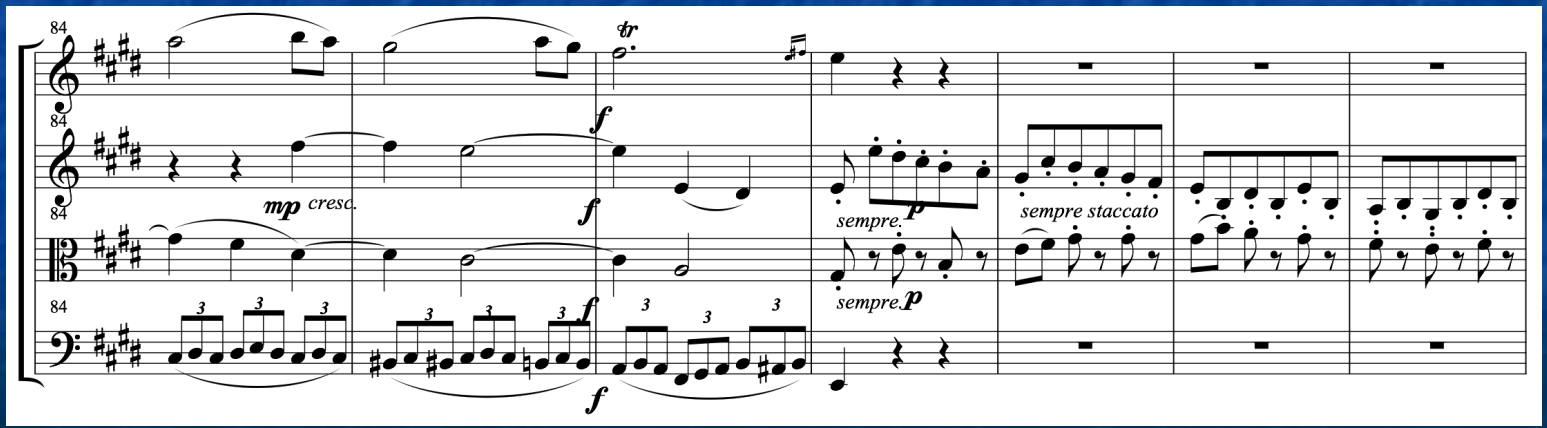
Polyphonic music can be *separated* into individual parts or *voices*

1) J.S. Bach: Fugue No 6 in D minor





2) Beethoven: String Quartet No 8 in E minor (3rd movement)



Motivations: Why do voice separation?

- Music analysis and understanding
- Music information retrieval
- Interactive music performance

- Can be time-consuming, tricky and monotonous to do by hand

Existing voice separation systems

Several systems exist

- *Kilian and Hoos (2002)*
- *Chew and Wu (2004)*
- *Madsen and Widmer (2006)*
- *Karydis, Nanopoulos, Papadopoulos and Cambouropoulos (2007)*

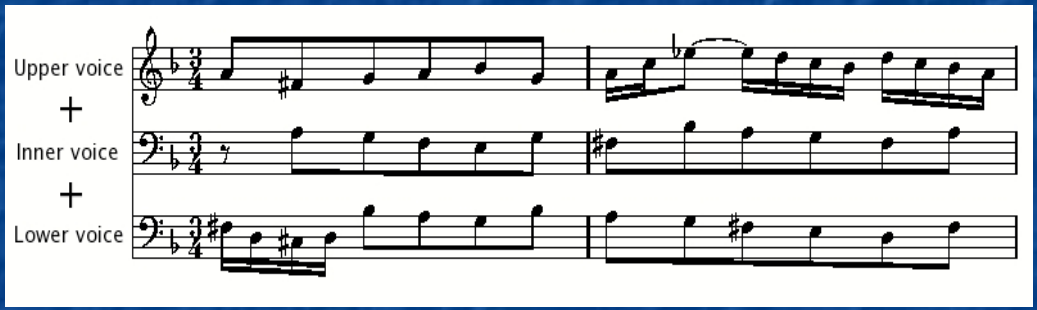
All these are rule-based, using specialist musical knowledge and heuristics

Q. To what extent are these rules and heuristics…

Comprehensive? Accurate? Encoded correctly?

This approach: Less reliance on encoding human knowledge

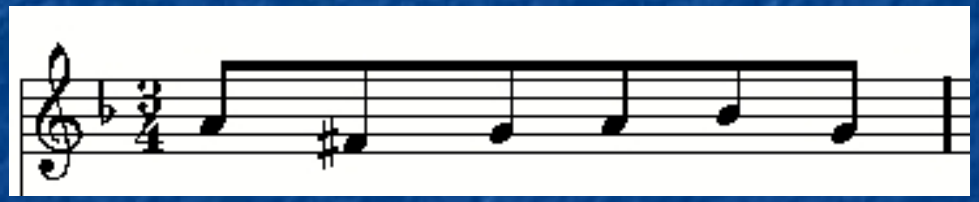
Examines several pieces of music:
making statistical observations about the pieces



- How often each note occurs in a voice
- Relationships between notes in each voice

Uses this information to judge how best to allocate voices in an unseen piece of music

Training the system: Example



(Using a training corpus of Bach 3-voice fugues, examining the upper voice)

Pitch occurrence

...	F#	G	G#	A	Bb	...
0	1	2	0	2	1	0

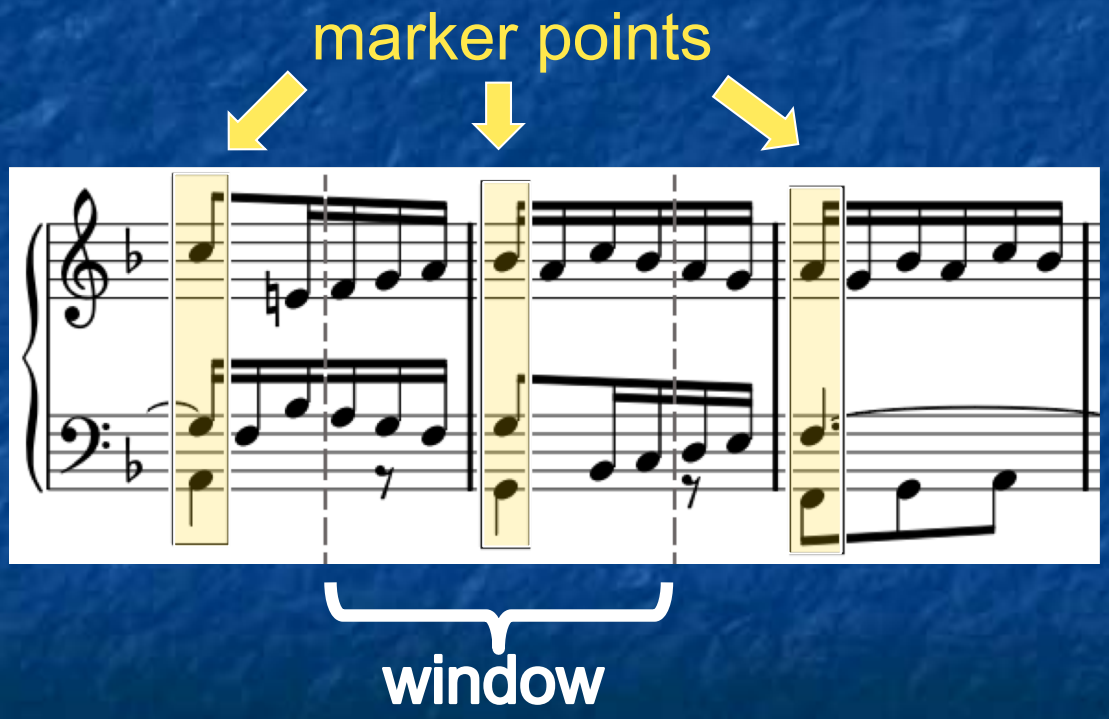
(NB In practice, all test pieces were transposed to the key of C before analysis. This was for normalisation purposes)

Interval occurrence

	...	F#	G	G#	A	Bb	B
...	0	0	0	0	0	0	0
F#	0	0	1	0	0	0	0
G	0	0	0	0	1	0	0
G#	0	0	0	0	0	0	0
A	0	1	0	0	0	1	0
Bb	0	0	1	0	0	0	0
...	0	0	0	0	0	0	0

The system in action

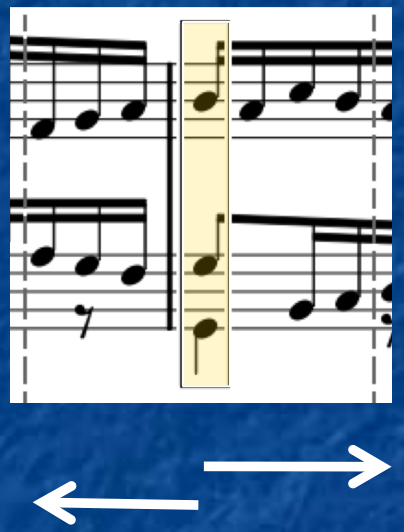
1. Find *marker points*: points where the voices are pitched very far apart. Work in *windows* around these marker points



The system in action

2. Working outwards from each marker point,

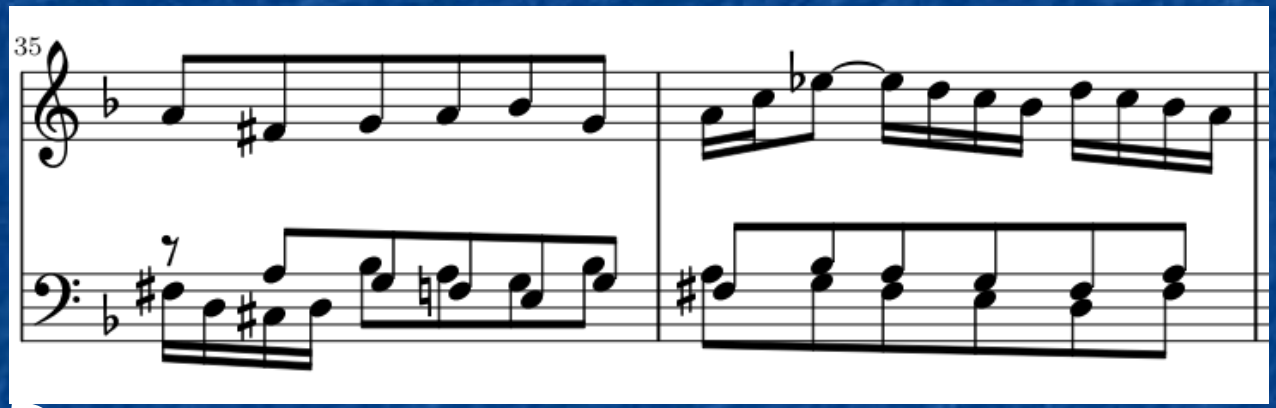
For each voice, **find the most likely next note in that voice** and allocate it to that voice

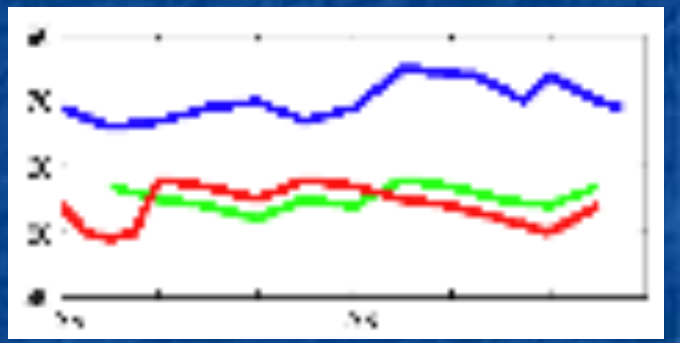


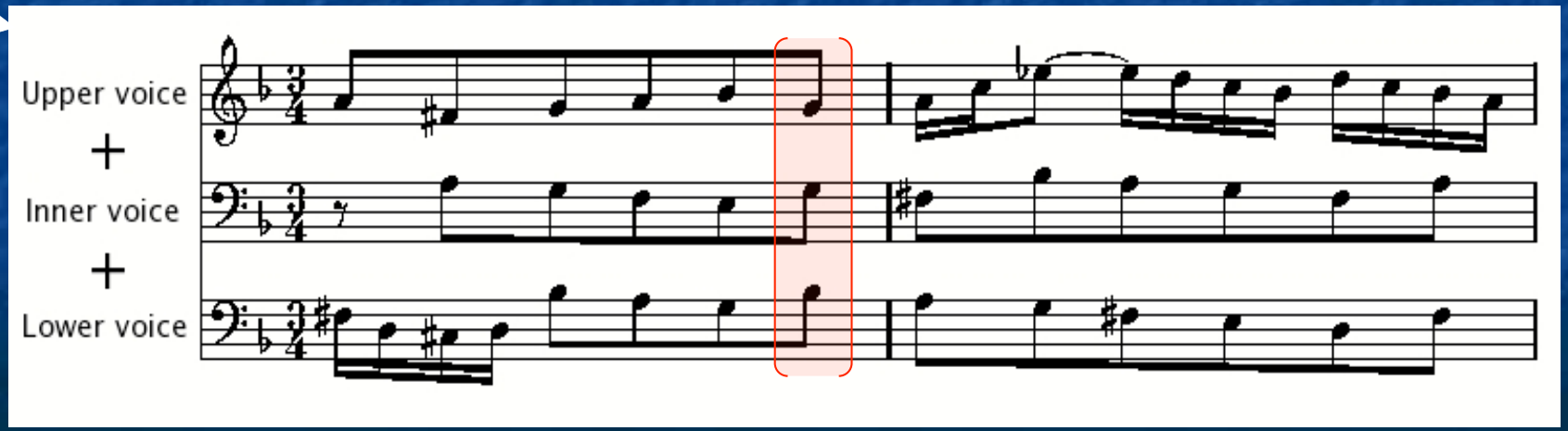
Determine the most likely next note using:

- Observations made during training
- Timing information
  - What notes are sounding when the current note for that voice has stopped sounding?
- Competition between voices for notes
  - When a note is allocated to more than one voice – which voice is it most likely to belong to?

Examples of the system in action





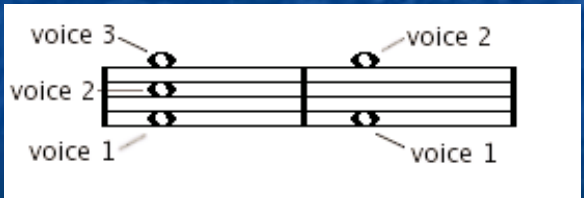


Comparative evaluation

Results compared against other systems' results:

- Chew and Wu
- Kirlin and Utgoff
- Madsen and Widmer
- Karydis et al

And with a baseline voice separation algorithm:




```
Allocate voices in ascending pitch order:
-    Lowest voice to lowest note
-    Next lowest voice to next lowest note
-    etc
Stop either when there are no more voices
     or when there are no more notes
```

Evaluation metrics

Taken from information retrieval/statistical classification.

2 aspects to successful allocation:
accuracy and completeness

System performance evaluated using

- precision
- recall
- F-measures (as an equal balance of precision and recall)

Results: Voice identification in Bach three-voice fugues

Method	Voice	Precision	Recall	F-measure
My system	v3	90.53%	88.84%	89.48%
My system	v2	82.92%	83.40%	83.01%
My system	v1	92.44%	92.80%	92.44%
Baseline	v3	97.27%	63.28%	76.01%
Baseline	v2	67.62%	74.89%	70.80%
Baseline	v1	80.09%	99.15%	88.37%

- Average of 10% improvement in F-measure scores compared to baseline algorithm
- Particular improvement in the two upper voices

Results: Voice identification in Bach four-voice fugues

Method	Voice	Precision	Recall	F-measure
My system	v4	79.85%	80.35%	79.73%
My system	v3	69.90%	67.45%	68.44%
My system	v2	69.31%	70.08%	69.35%
My system	v1	81.23%	83.04%	80.80%
Baseline	v4	94.97%	40.30%	54.92%
Baseline	v3	52.48%	49.66%	50.89%
Baseline	v2	52.99%	66.26%	58.42%
Baseline	v1	70.58%	99.43%	81.47%

- Average of 13% improvement in F-measure scores compared to baseline algorithm

Results: Voice identification in Beethoven string quartets

Method	Voice	Precision	Recall	F-measure
My system	violin1	79.84%	69.47 %	71.86%
My system	violin2	59.79%	57.91%	58.68%
My system	viola	60.86%	59.38%	60.00%
My system	cello	71.55%	72.02%	71.70%
Baseline	violin1	80.08%	51.41%	62.07%
Baseline	violin2	64.39%	57.03%	60.29%
Baseline	viola	63.52%	67.07%	65.06%
Baseline	cello	66.91%	90.68%	76.54%

- Nearly 10% improvement in F-measure scores for violin 1 but similar/poorer scores for other parts

Comparison of different systems

System	Precision	Recall	F-measure
This study	**80.88%**	**80.85%**	**80.86%**
Chew & Wu	n/a	88.98%	n/a
Kirlin & Utgoff *	88.65%	65.57%	75.38%
Madsen & Widmer	95.94%	70.11%	81.02%
Karydis et al	93.19%	n/a	n/a

- Results compared against results reported in the other authors' papers
- Compared over a corpus of Bach fugues (except * which was tested on sections of Bach's Ciaconna)

Further experimentation

Q. What would happen if the system was trained on one corpus and tested on a related but different piece of music?

Voice separation was carried out on **Mozart's *Fugue in C minor***, using a training corpus of **Bach fugues**.

Result: Mean F-measure of **75%**

Summary

Back to the questions I posed
at the start of this talk…

Summary

- **What is voice separation?**
  - Identifying the individual parts that make up a piece of polyphonic music
- **Have there been attempts to automate voice separation?**
  - Yes – but mostly they rely on encoding specialist knowledge correctly
- **What solution do I propose?**
  - Allocating voices in music using statistical observations
- **What results have I achieved with my approach?**
  - 12% overall improvement in performance over the baseline approach
  - 80.86% F-measure on Bach fugues: compares well with other methods
- **What conclusions can be drawn from this work?**
  - A complex problem can be solved using a set of simple observations
  - Rule-based systems can be (at least) matched in performance by a data-driven approach

References

E. Cambouropoulos. *'Voice' separation: theoretical, perceptual and computational perspectives.* In ICMPC, Italy, 2006.

E. Chew and X. Wu. *Separating Voices in Polyphonic Music: A Contig Mapping Approach.* In Computer Music Modeling and Retrieval: Revised Papers, Esbjerg, Denmark, 2005.

I. Karydis, A. Nanopoulos, A. Papadopoulos, and E. Cambouropoulos. *VISA: The Voice Integration/Segregation Algorithm.* In ISMIR, Austria, 2007.

J. Kilian and H. Hoos. *Voice separation - a local optimisation approach.* In ISMIR, France, 2002.

P. Kirlin and P. Utgoff. VoiSe*: Learning to Segregate Voices in Explicit and Implicit Polyphony.* In ISMIR, UK, 2005.

S. Madsen and G. Widmer. *Separating voices in MIDI.* In ISMIR, Canada, 2006.

Anna Jordanous

Music Informatics Research Centre
University of Sussex, UK

a.k.jordanous@sussex.ac.uk

Thanks to Nick Collins, Chris Thornton and Chris Darwin
for their helpful comments on this work