Computational Model for Automatic Chord Voicing based on Bayesian Network

Tetsuro Kitahara, Makiko Katsura, Haruhiro Katayose, and Noriko Nagata

CrestMuse Project, CREST, JST, Japan/
Kwansei Gakuin University, Japan

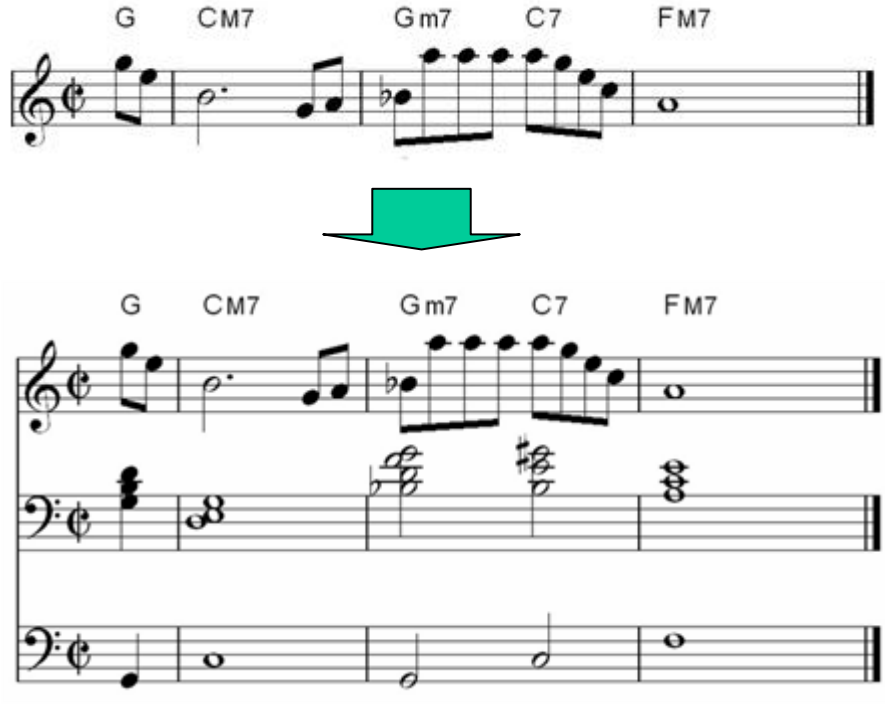
ICMPC 10 (August 2008 in Sapporo)

Our Goal

To develop a computational model for automatic *chord voicing*

Chord voicing:

Simultaneous vertical placement of notes in order to obtain the effective sound of a chord



What's the difficulty in chord voicing

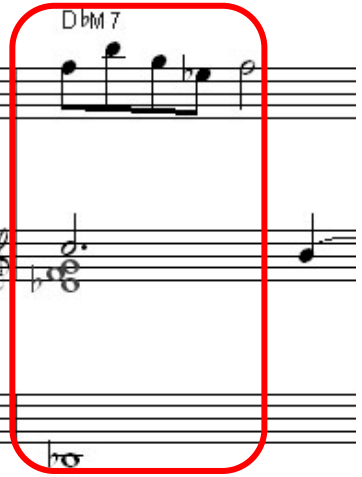
3 issues:

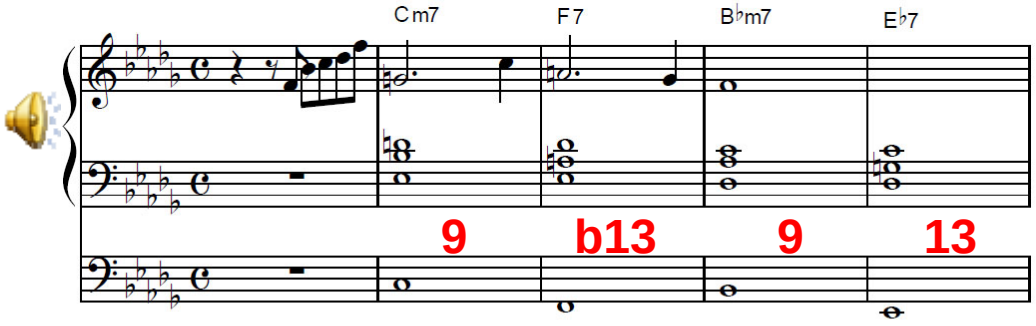
Extended notes	Omitted notes	Inversions
e.g. 9ths, 11ths, … Essential especially for jazz music	Only adding extended notes may cause dissonant sound. Physical limitation	Important to make temporally smooth voicings

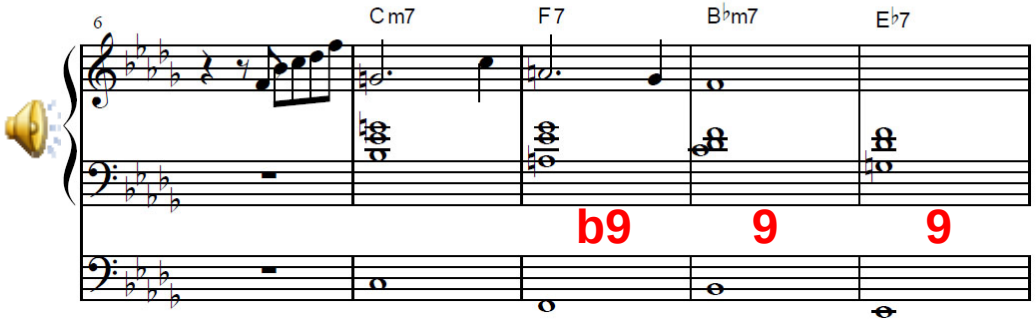
DbM7

DbM7(#11)

DbM7(9,13)



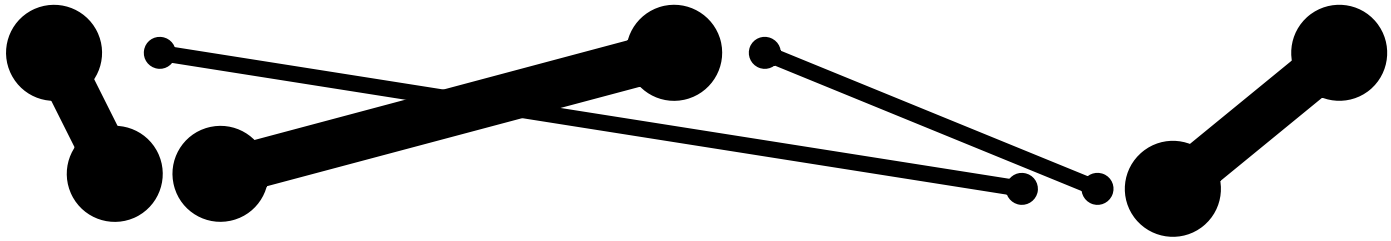




What's the difficulty in chord voicing

3 issues:

Extended notes	Omitted notes	Inversions
e.g. 9ths, 11ths, … Essential especially for jazz music	Only adding extended notes may cause dissonant sound. Physical limitation	Important to make temporally smooth voicings



Many-to-many relationship

2 aspects:

Musical simultaneity	Musical sequentiality
Matching with the simultaneous notes of other parts (e.g. melody)	Smooth connection of previous and following chords

3 possible solutions

- ***Rule-based***
  Design a set of rules manually
  --- [Emura, 2008]
- ***Case-based***
  Use existing examples (corpus) of chord voicings
  - ***Instance-based***
    e.g. Find part of the target chord progression in a corpus and copy-and-paste their voicings
    --- [Hirata, 2001] etc.
  - ***Probability-based***
    e.g. Train a probabilistic model with a corpus
    --- Few trials in previous studies

Merits (M) and Demerits (D)

- ***Rule-based***

  [M] If an expert carefully design rules,
  high quality can be expected.

  [D] Difficult to design a good rule set.

  ⇨Rules satisfying both simultaneity and sequentiality

  [D] Difficult for users to control the behavior

- ***Case-based***

  [M] No need to design a thorough rule set.

  [M] Users can control the behavior by switching the corpus.

  [D] If a probabilistic model, a large curpus is needed.

In this study…

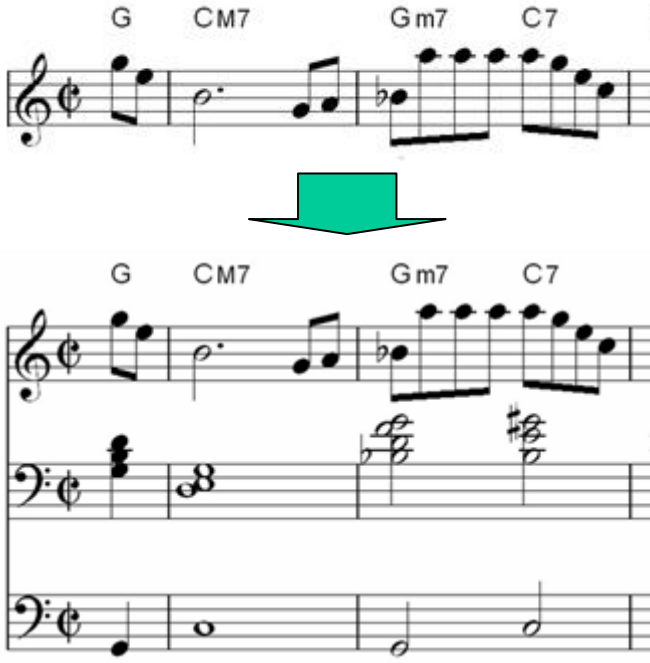
To develop a *probabilistic* model for automatic chord voicing

Aim: To show that…

- We can generate voicings satisfying simultaneity and sequentiality with a model where they are described as probabilistic dependencies.
- If a particular genre corpus is used, the generated voicing also has the characteristics of that genre.

Problem Statement

- **Target instrument:** Electronic organ (Electone)
  - Upper (right-hand) keyboard: Melody line
  - Lower (left-hand) keyboard: Chord
  - Pedal keyboard: Bass line
- **Input:** Data written on a lead sheet
  - A melody line
  - a sequence of chord names
- **Output:** A left-hand voicing and a bass note for each chord
  - No passing chords or passing bass notes are added.
- **Target genre:** Jazz

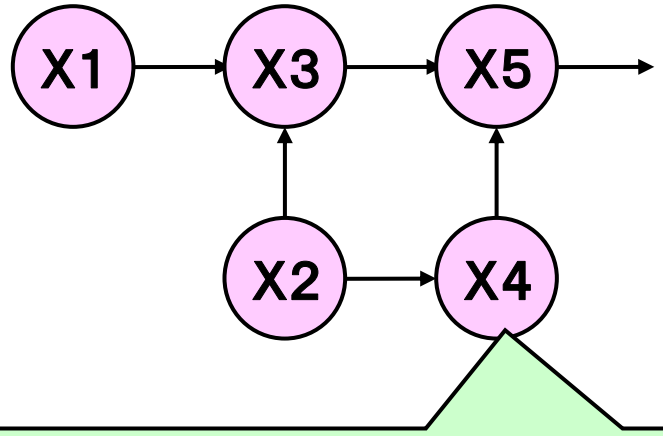


Probabilistic model we use

We use a *Bayesian network*

Bayesian network:
a probabilistic graphical model that represents variables and their probabilistic independencies

- Nodes: Random variables (typically discrete)
- Links: Conditional depend-encies between variables
- Each node has a conditional probability table (CPT).
- After observable variables are set, the probabilities of the others are estimated.



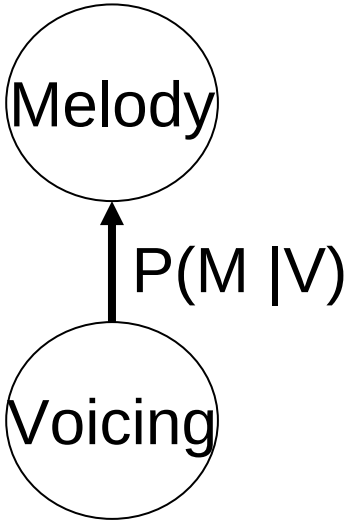
CPT (Conditional Probability Table)

P(X4|X2)

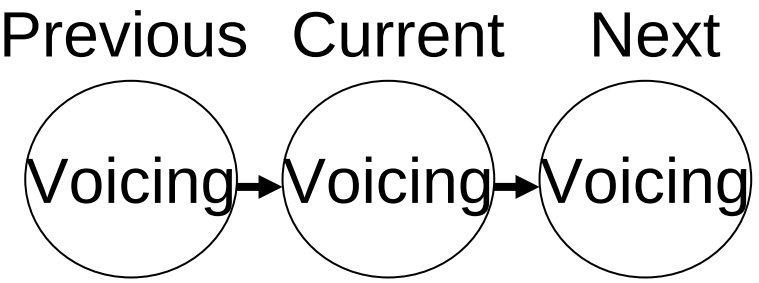
X4 \ X2	0	1
0	0.8	0.4
1	0.2	0.6

Basic policies

- ***Simultaneity***
  - Melody and voicing nodes are linked.
  - We consider an observed melody to be generated from the harmony including the voicings.



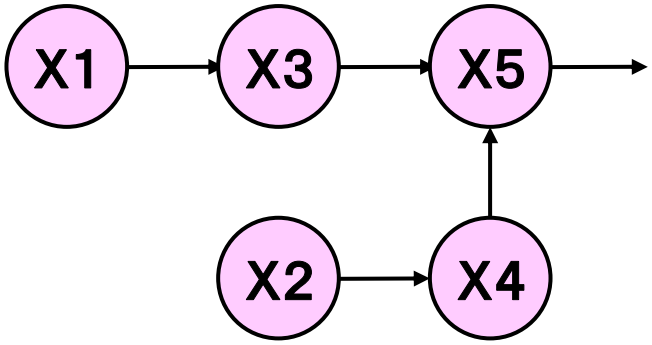
- ***Sequentiality***
  - The voicing nodes for the previous, current, and next chords are linked.
  - The nodes for top, middle, and bottom notes are separated.



Basic policies

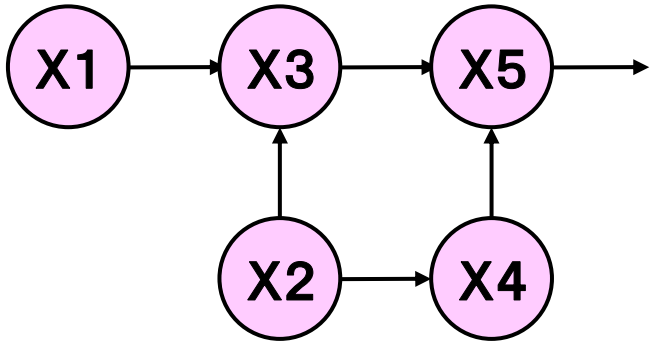
- Design a singly connected network

Singly connected network



A simple O(n) algorithm for probability calculation exists.

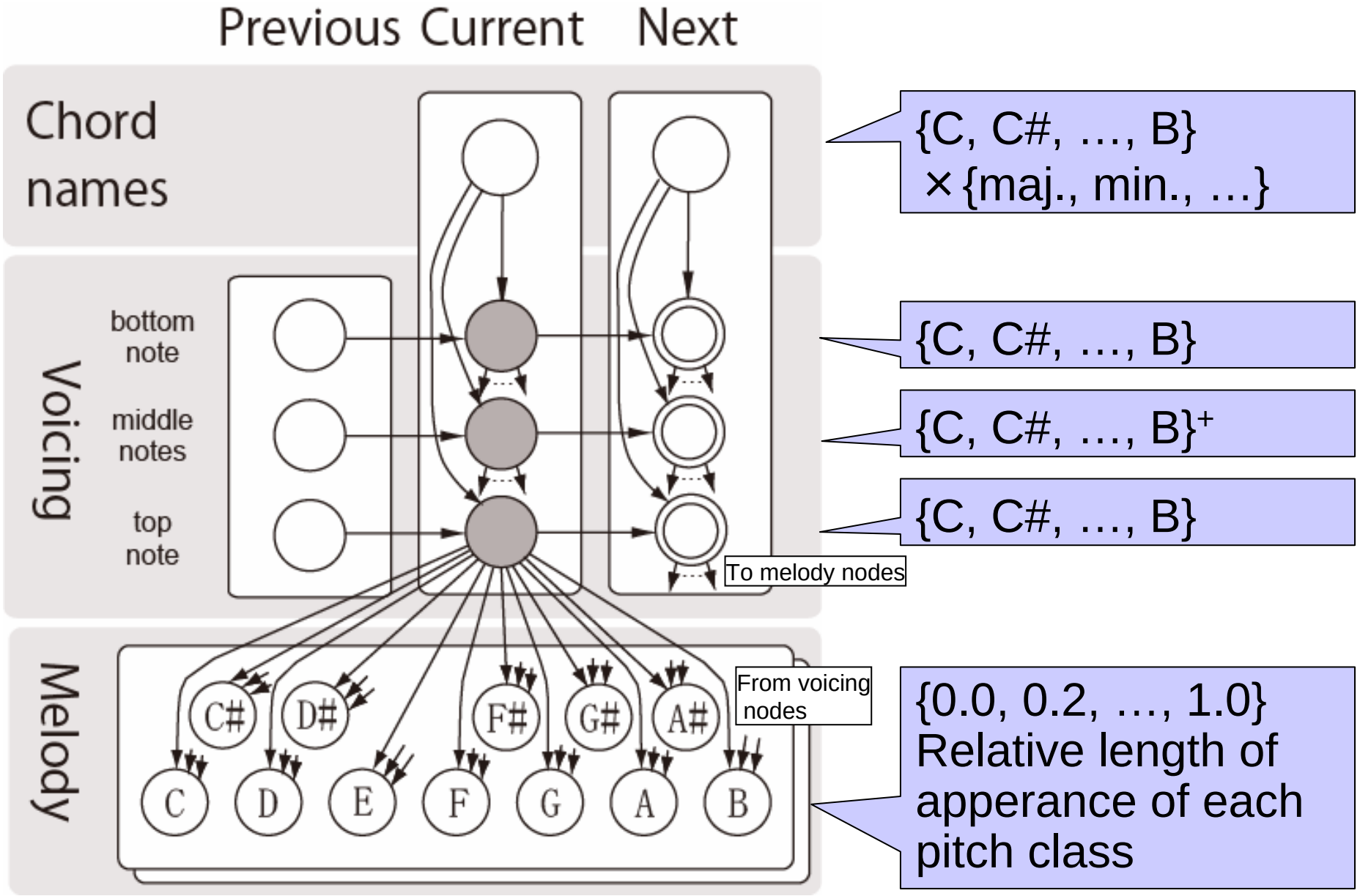
Multiply connected network



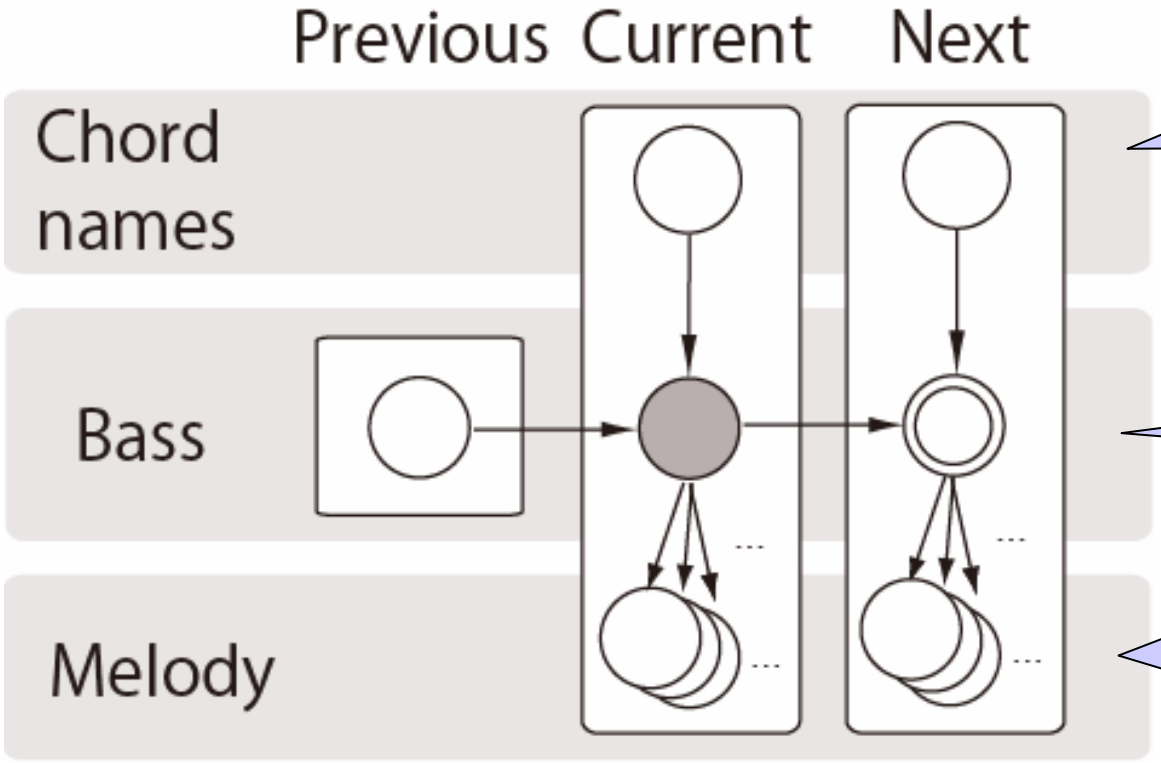
A complicated algorithm is needed.

- Design separate models for left-hand voicing and bass note determination
  - Models should be simple for training with limited data

Left-hand voicing model



Bass model



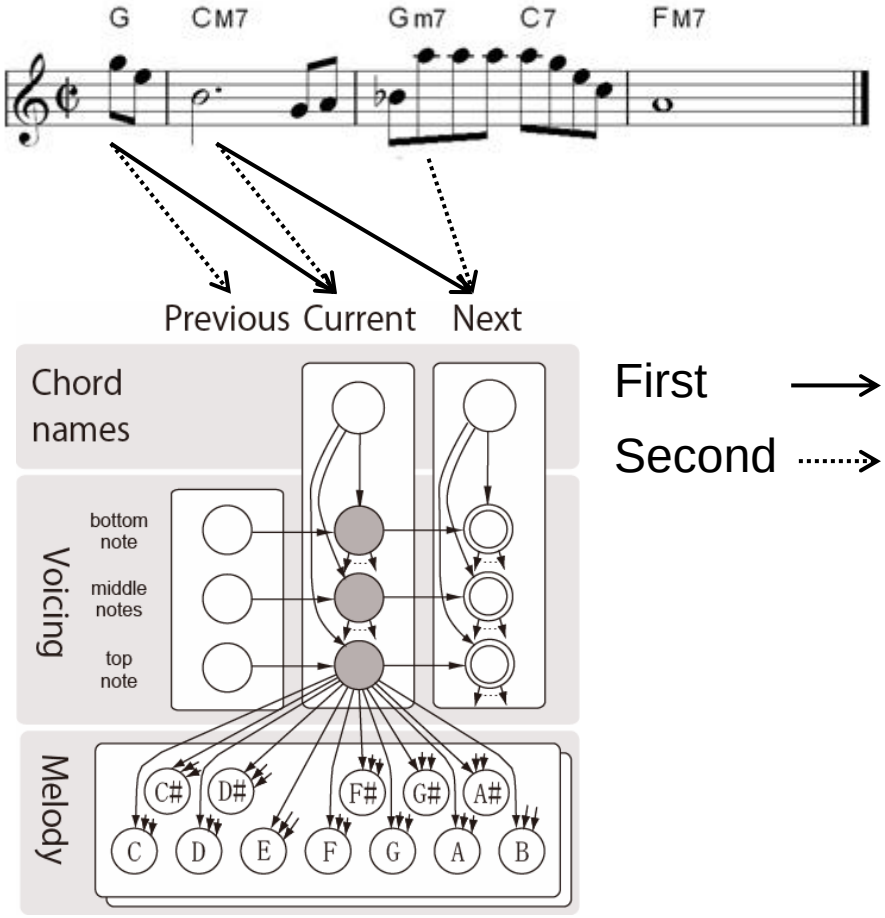
{C, C#, …, B}
×{maj., min., …}

{C, C#, …, B}

{0.0, 0.2, …, 1.0}
Relative length of apperance of each pitch class

System implementation

- Our models are applied to each chord from the beginning to the end of a given chord progression by shifting the current chord.
- After the probabilities of the nodes are calculated, the note with the highest probability is chosen from each current voicing node.
- Pitch range
  Left-hand: C3-A#4
  Bass: C2-G3
- Training: 30 jazz pieces



Example of Outputs

"Misty"

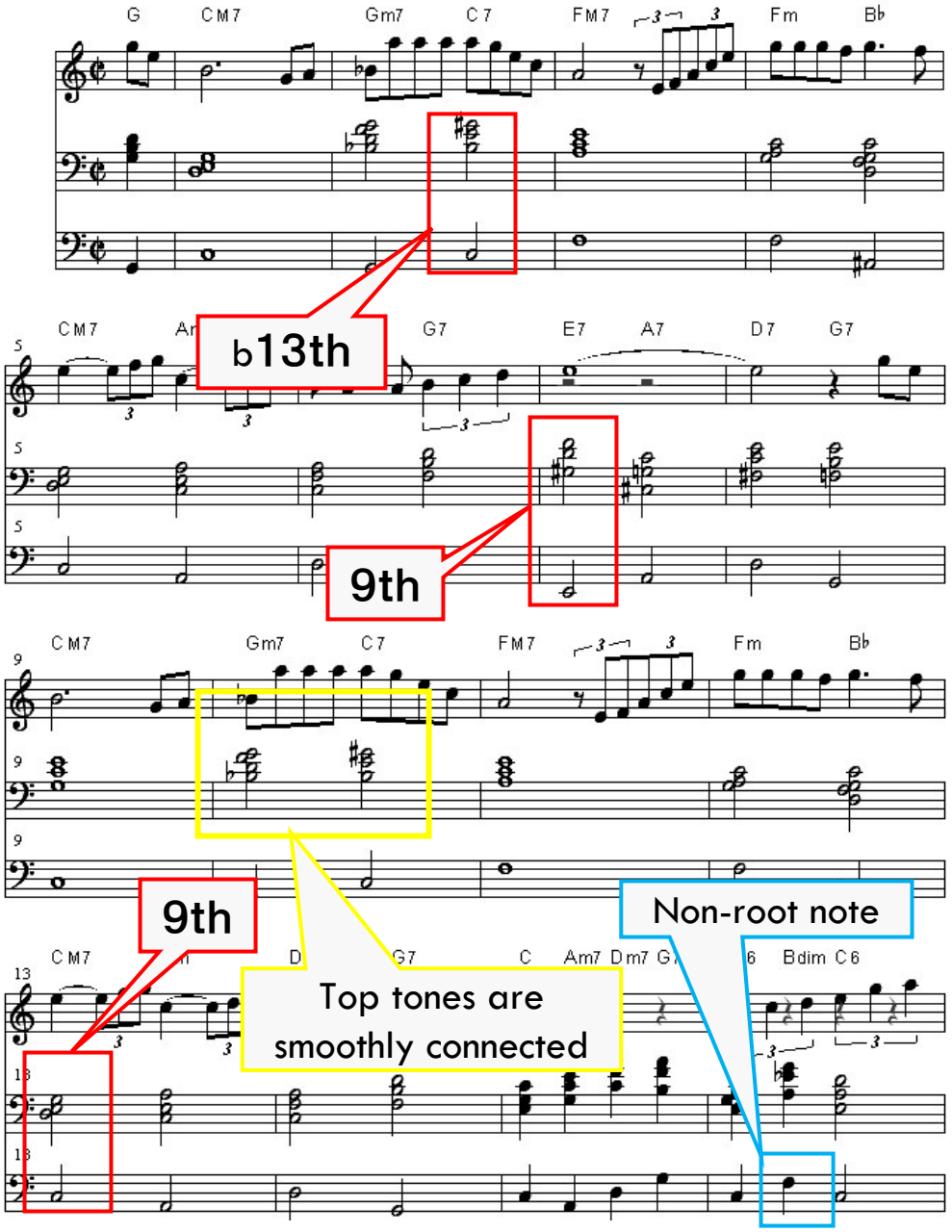
(by Erroll Garner)

Left-hand voicing

- Extended notes were added.
- Natural inversions were chosen.

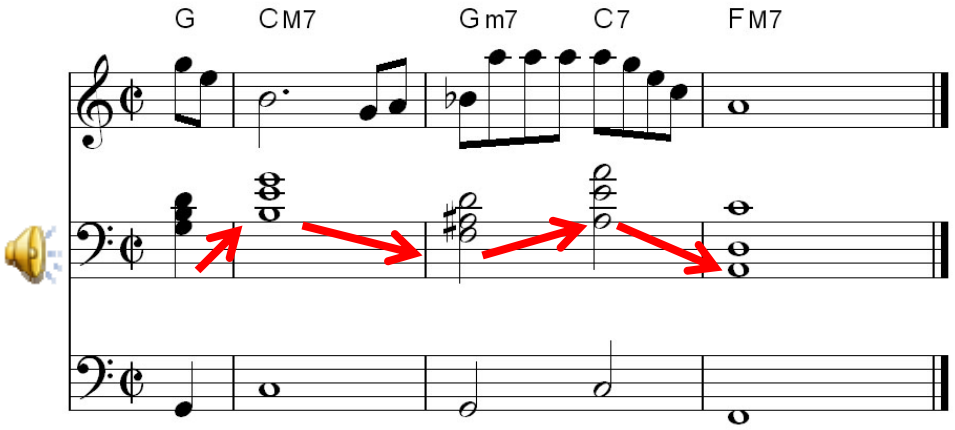
Bass

- Mostly root notes were chosen.

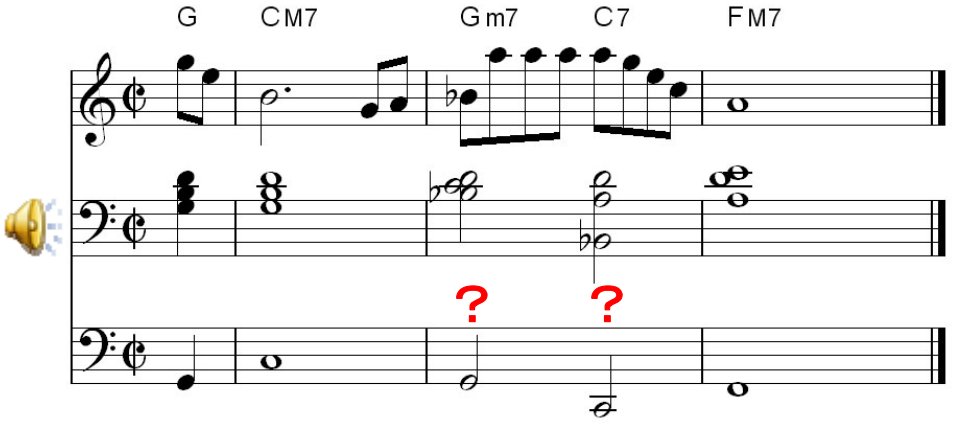


Qualitative evaluation of simultaneity and sequentiality

1. No previous- and next-chord nodes
   - ○ Extended notes match the melody.
   - × Temporal connection is not smooth.



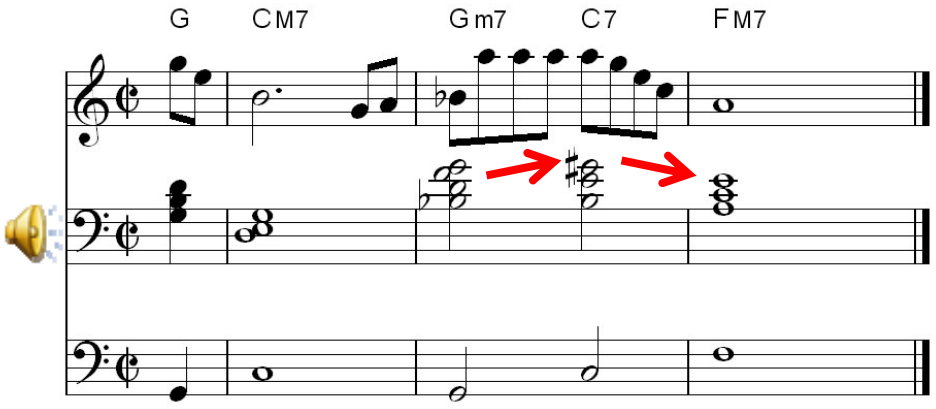
2. No melody nodes
   - ○ Temporal connection is smooth.
   - × Some notes do not match the melody



Qualitative evaluation of simultaneity and sequentiality

3. Our model

- The matching with the melody
- Smooth temporal connection



Subjective evaluation of simultaneity by music experts

Subjects rate the voicing for each chord

- ◎ : Beautiful
- ○ : Good
- △ : Acceptable
- × : Unacceptable

3 subjects

e.g. Electone instructor
Demonstrator

5 pieces (608 chords)

Results

	# of chords	Percent-age
Beautiful	82	9.0%
Good	719	78.8%
Accepable	63	6.9%
Unacceptable	48	5.3%

94.7%

Subjective evaluation of sequentiality by music experts

Subjects rate continuity of each chord change

- ○ : Good
- △ : Acceptable
- × : Unacceptable

2 subjects

5 pieces
(299 chord changes)

Results

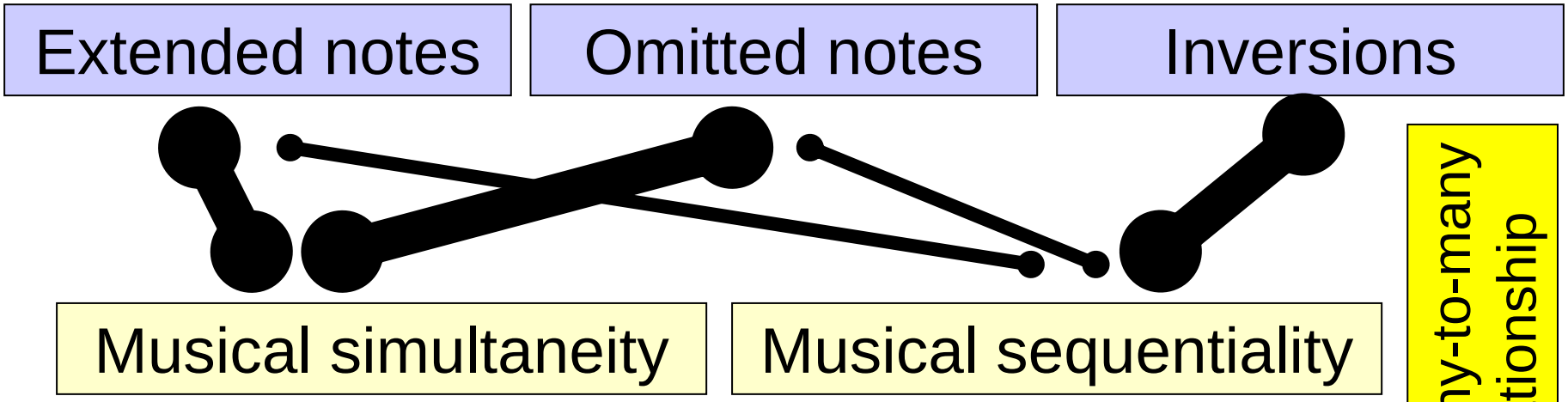
	# of chord changes	Percent-age
Good	180	25.8%
Accepable	407	58.3%
Unacceptable	111	15.9%

84.1%

Conclusions

Our goal: To develop a computational model for automatic chord voicing

Issue:



Our solution: Use a Bayesian network

Results: Generated voicings satisfy both simultaneity and sequentiality