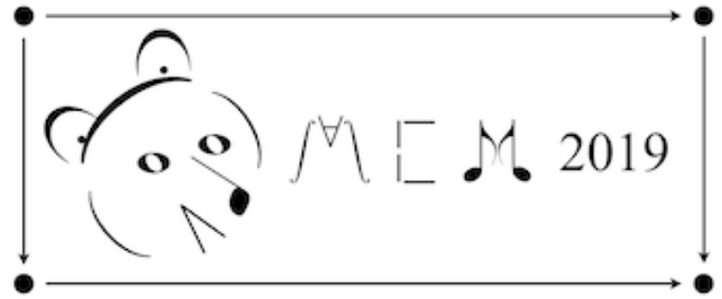


A Parse-based Framework for Coupled Rhythm Quantization and Score Structuring

Florent Jacquemard

Francesco Foscarin
Philippe Rigaux

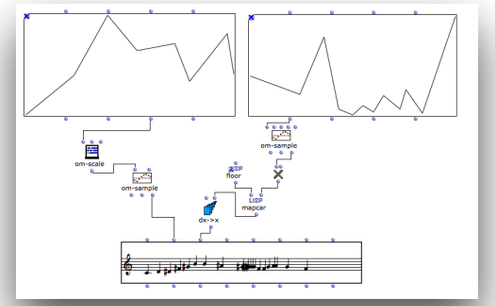
le cnam

Masahiko Sakai

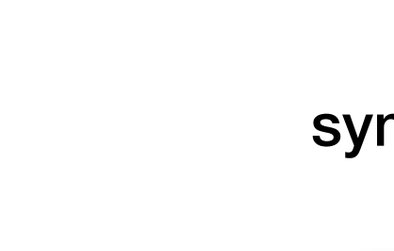
supported by:

Automated Music Transcription

symbolic generation (CAC)



MIDI device



audio recording

sound processing

piano roll
unquantized onsets
quantized pitches
(MIDI)

rhythm quantization

quantized piano roll

score engraving

score
(XML file)

audio-to-MIDI

MIDI-to-score

MIDI to score transcription with independent subtasks

1. Rhythm Quantization

unquantized MIDI →	quantized MIDI
sequential data →	sequential data

sequential model of durations (*HMM*)

Markov models of note values
[Raphael 2001], [Sagayama et al 2002], [Nakamura et al 2016]

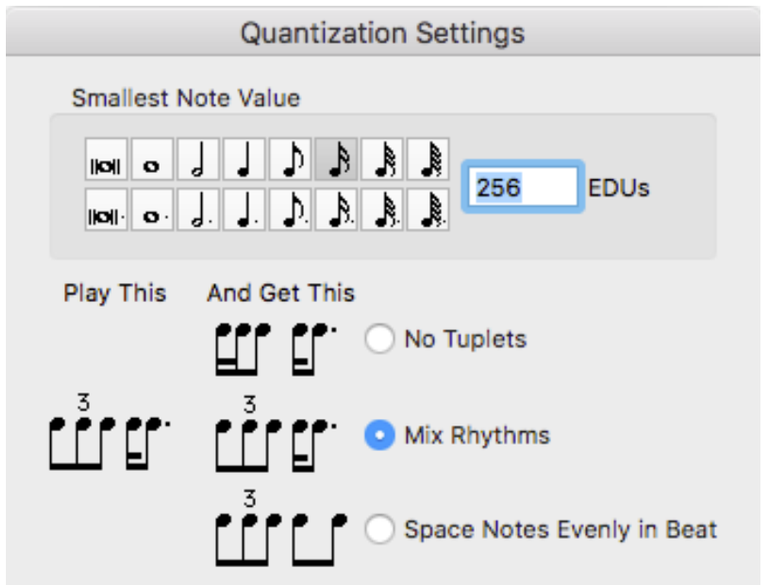
2. Score Engraving

quantized MIDI →	XML score file
sequential data →	1/2 structured data

delegated to functionality of a score editor (*MIDI import*)

3. Combination: Interface between the 2 subtasks?

- complex rhythm, deep nesting
- mixed tuplets
- rests, grace notes…



MIDI to score transcription with coupled subtasks

(our proposal)

unquantized MIDI	→	tree representations	~	XML score file
sequential data	→	1/2 structured data		1/2 structured data

integrated approach to MIDI-to-score transcription
coupling Rhythm Quantization & Score Production
based on:

- a hierarchical model of notation (tree-structured)
  similar to OpenMusic's Rhythm Trees [Agon et al 2002]
  expressive, accurate for complex rhythms

- quantitative parsing techniques
  for context-free grammars weighted in semirings
  efficient & modular

(focus on rhythm transcription)

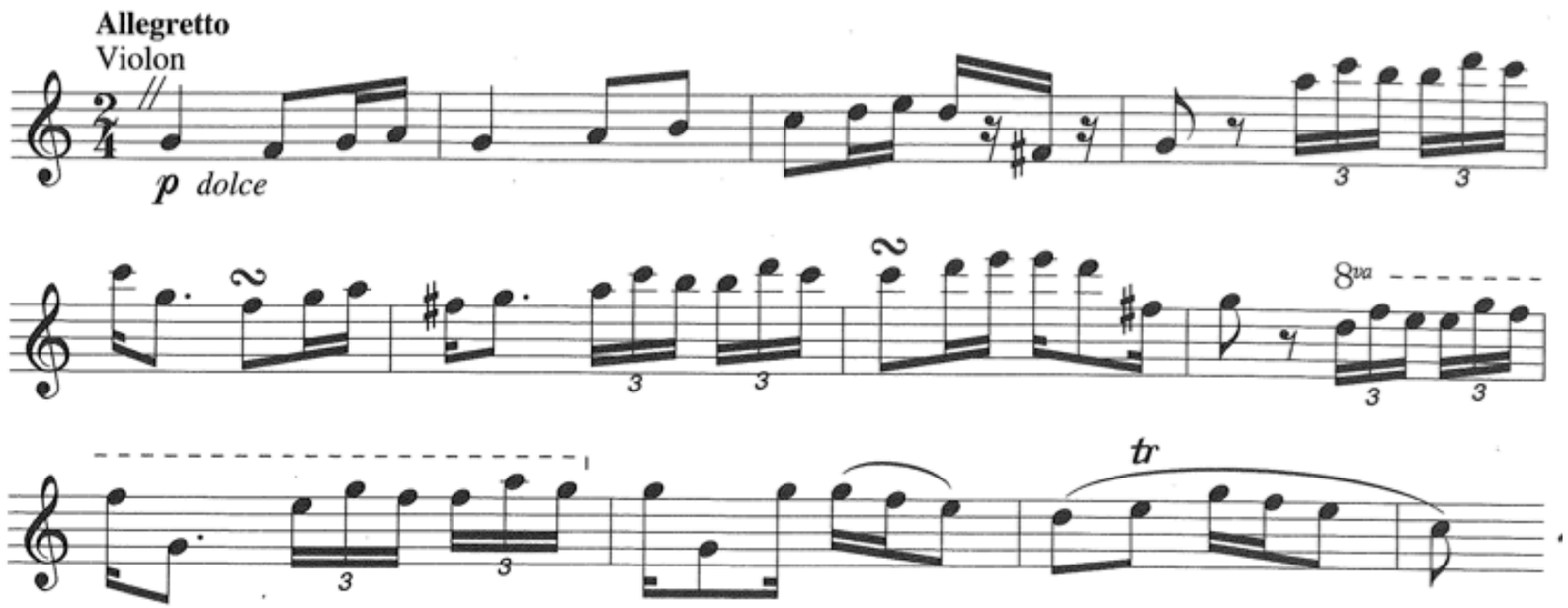
Implementation, Results

transcription: MIDI recording to XML/MEI

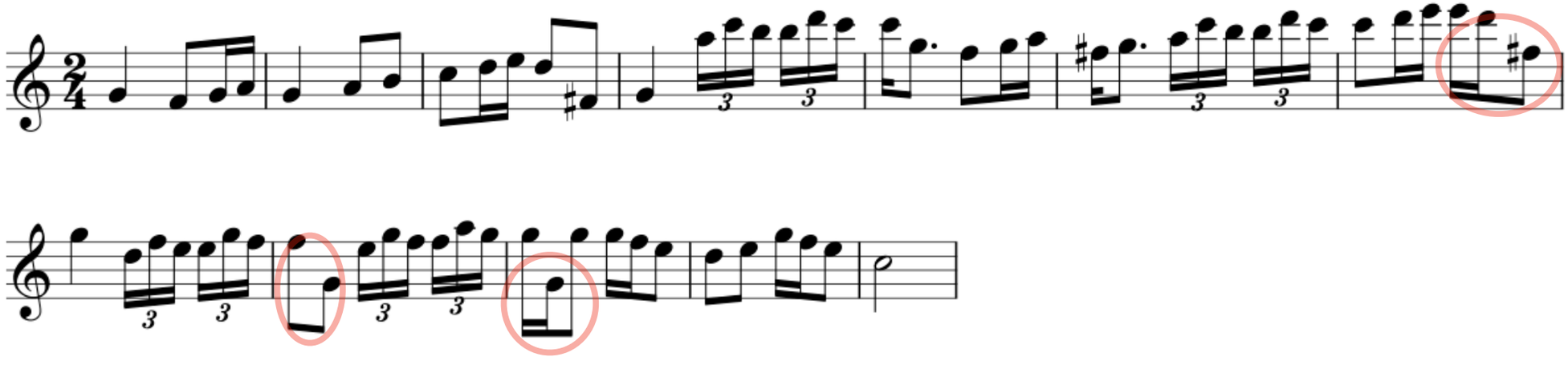
https://gitlab.inria.fr/qparse/qparselib
https://qparse.gitlabpages.inria.fr

original score

Beethoven, Trio for violin, cello and piano, op. 70 n.2 (2d mov)



transcription of MIDI recording with qparse



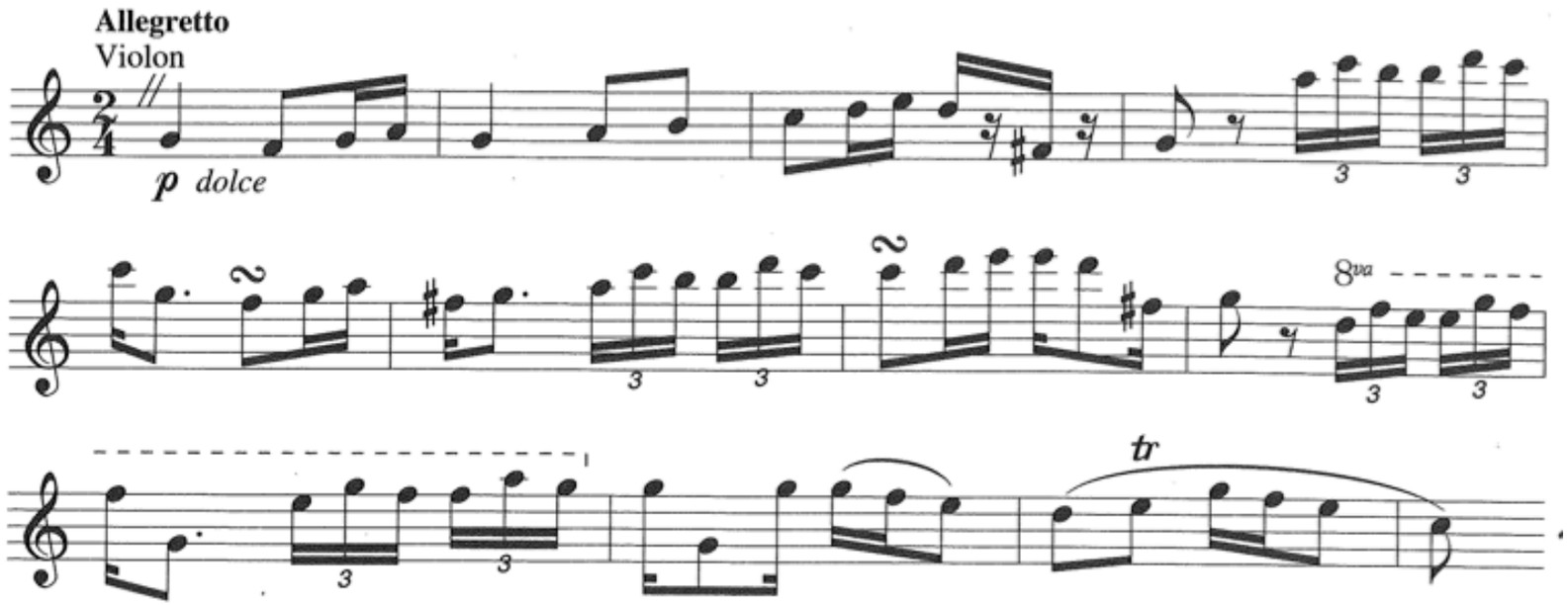
Implementation, Results

transcription: MIDI recording to MusicXML

finale
music notation software

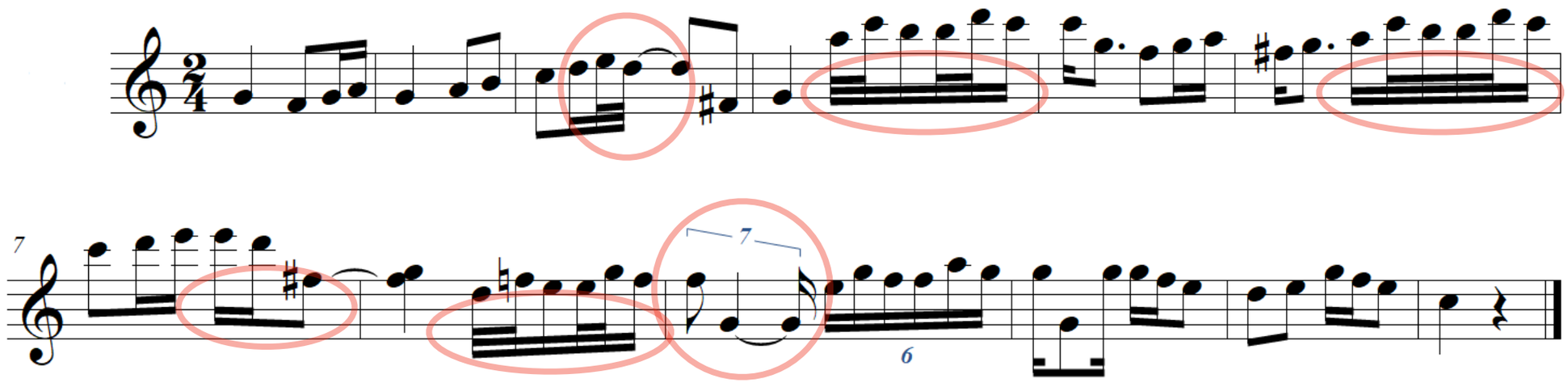
original score

Beethoven, Trio for violin, cello and piano, op. 70 n.2 (2d mov)



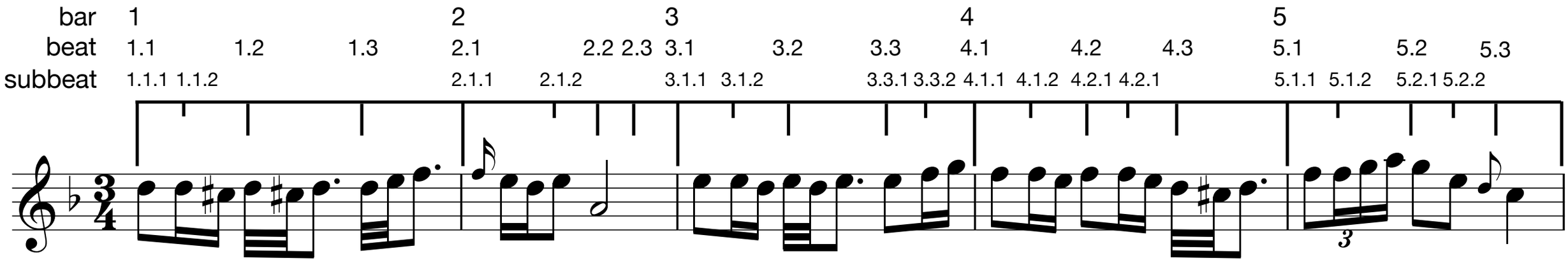
transcription of MIDI recording with Finale.

options:
mixed rhythms,
tuplets
smallest note = 32nd
The time signature and the tempo are given.

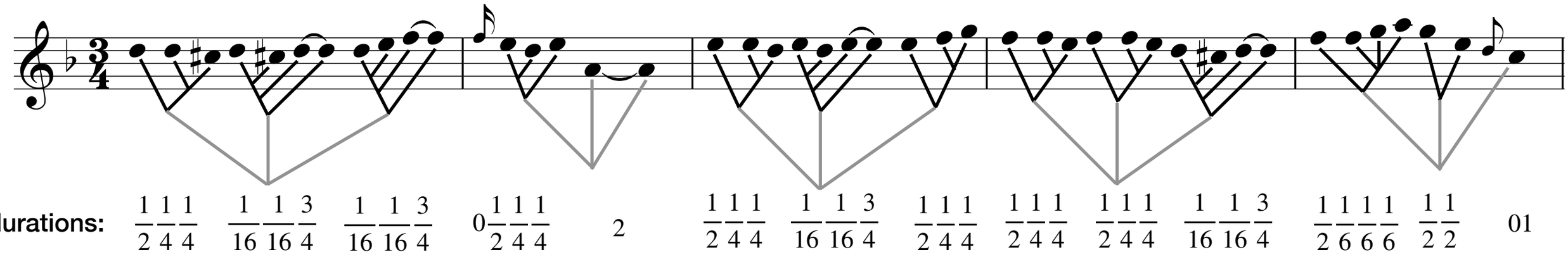


Hierarchical Structure of Music Notation

The notation gives clues (to player) of the metric structure



Polonaise in D minor from Notebook for Anna Magdalena Bach BWV Anh II 128



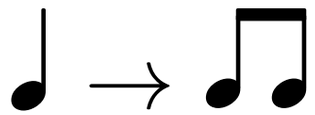
Tree representation of proportional rhythmic notation

Languages of rhythmic notation

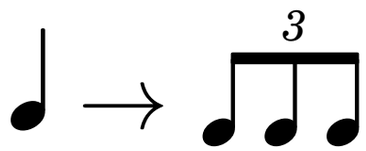
Context-Free Grammar (CFG): finite set of context-free production rules.

example: (very simple) CFG for $\frac{1}{4}$ bar sequences.

non terminal symbols: q (bar seq.), q_0 (1 bar = 1 beat), q_1, q_2,...
terminal symbols: $-$ (continuation),
• (1 note),
$\bullet_1$ (1 grace-note + 1 note),
$\bullet_2$ (2 grace-notes + 1 note),...



$q_0 \quad \rightarrow \quad q_1\ q_2$



$q_0 \quad \rightarrow \quad q_1\ q_2\ q_2$

Languages of rhythmic notation

Context-Free Grammar (CFG): finite set of context-free production rules.

example: (very simple) CFG for $\frac{1}{4}$ bar sequences.

non terminal symbols: q (bar seq.), q_0 (1 bar = 1 beat), q_1, q_2,...
terminal symbols: $-$ (continuation),
• (1 note),
$\bullet_1$ (1 grace-note + 1 note),
$\bullet_2$ (2 grace-notes + 1 note),...

$$q \rightarrow q_0 \qquad q \rightarrow q_0\ q$$

$$q_0 \rightarrow q_1\ q_2 \qquad q_0 \rightarrow q_1\ q_2\ q_2$$

Languages of rhythmic notation

Context-Free Grammar (CFG): finite set of context-free production rules.

example: (very simple) CFG for $\frac{1}{4}$ bar sequences.

non terminal symbols: q (bar seq.), q_0 (1 bar = 1 beat), q_1, q_2,...
terminal symbols: $-$ (continuation),
$\bullet$ (1 note),
$\bullet_1$ (1 grace-note + 1 note),
$\bullet_2$ (2 grace-notes + 1 note),...

$$q \to q_0 \qquad q \to q_0\ q$$

$$q_0 \to q_1\ q_2 \qquad q_0 \to q_1\ q_2\ q_2$$

$$q_0 \to - \qquad q_0 \to \bullet \qquad q_0 \to \bullet_1 \qquad q_0 \to \bullet_2$$

$$q_1 \to - \qquad q_1 \to \bullet \qquad q_1 \to \bullet_1 \qquad q_1 \to \bullet_2$$

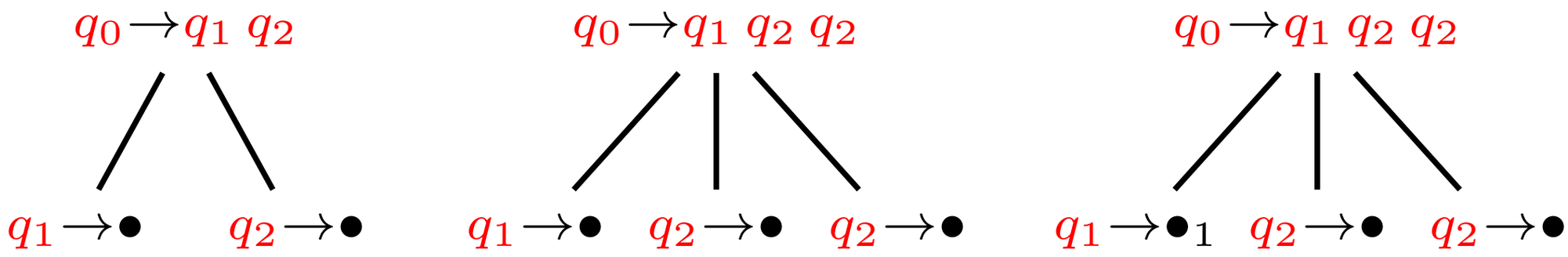
$$q_2 \to q_3\ q_3 \qquad q_2 \to \bullet$$

$$q_3 \to q_4\ q_4 \qquad q_3 \to - \qquad q_3 \to \bullet \qquad q_4 \to \bullet$$

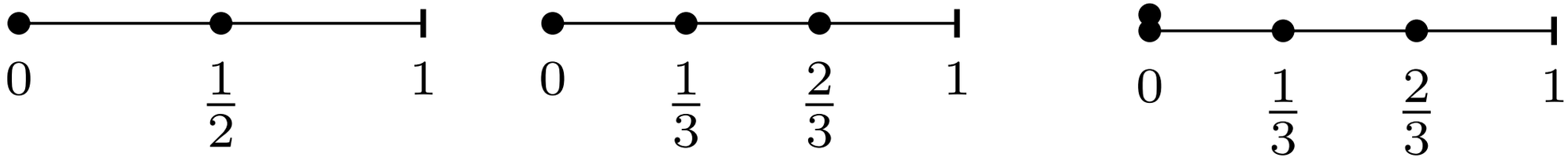
Parse trees

a parse tree is a tree labeled by grammar's production rules,
with tiling of non-terminals.

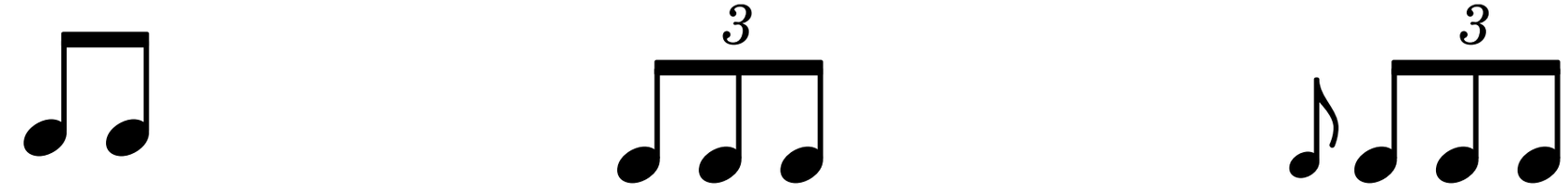
parse tree



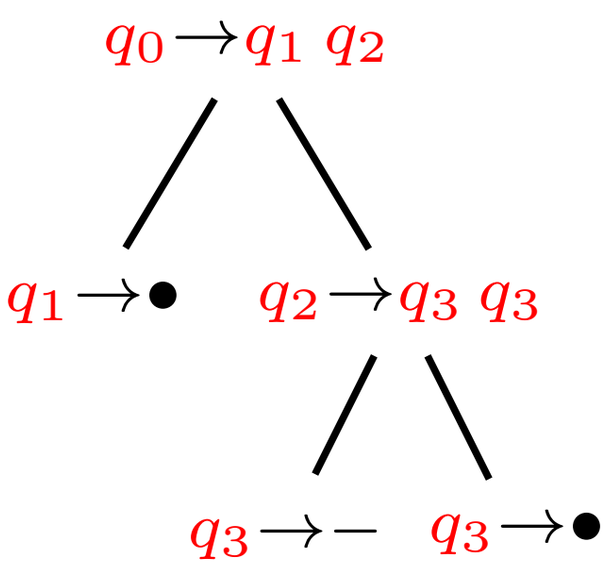
serialization

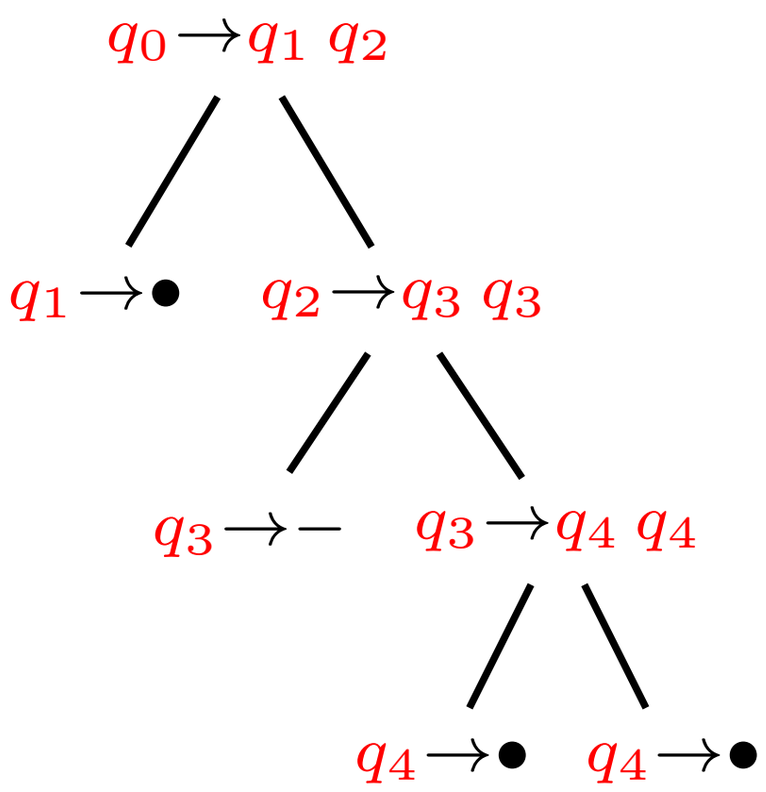


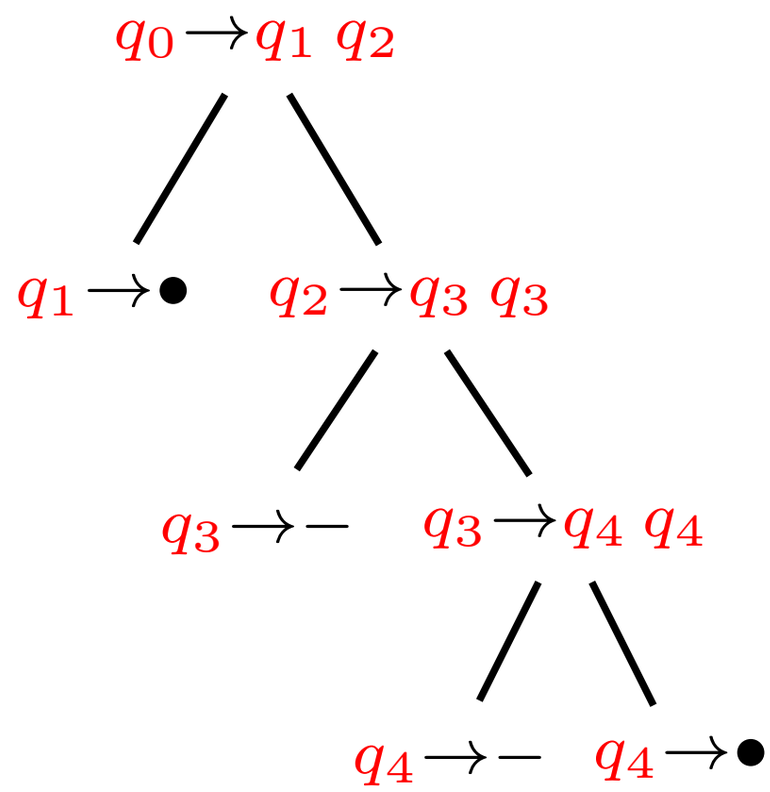
corresponding notation

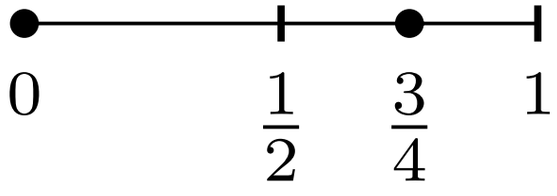


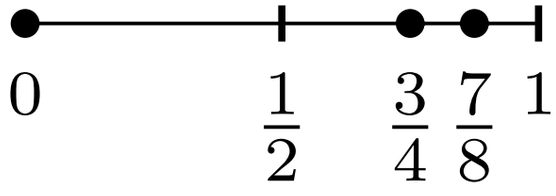
More parse trees

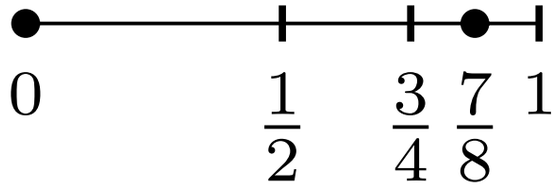


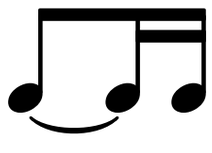












Closure of languages

- a word (sequence of terminal symbols) is parsed by a CFG
  if it is the yield of a parse tree

Properties:

- languages of (parsed) words are not closed under intersection.
- languages of parse trees = regular tree languages
  are closed under intersection.

tree languages:
→ capture regularity of rhythm
→ grammar composition (Cartesian product)
  combination of grammar representation of different aspects

Objectives (informal definition)

given:

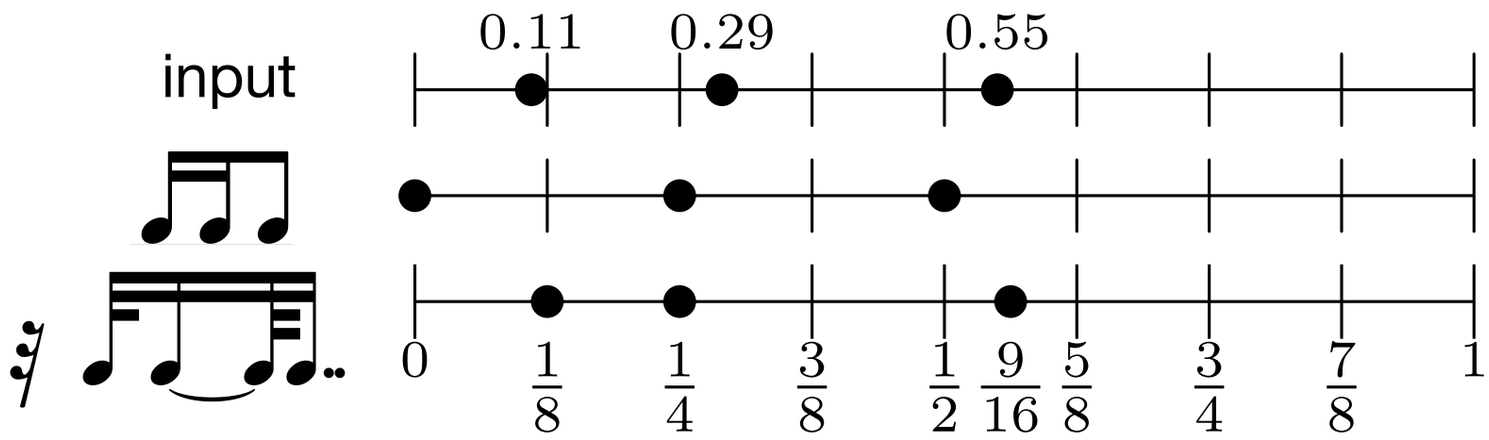
- a grammar $\mathcal{G}$ describing rhythmic notation preferences
- an input sequence of events σ

return:

- a parse tree t of $\mathcal{G}$ that *fits well* with σ

questions:

- How *well*? How to avoid complex parse trees (overfitting).



→ find good compromise between precision and complexity of notation (multicriteria optimisation)

→ quantitative parsing

Semirings

A *semiring* $\mathcal{S} = \langle \mathbb{S}, \oplus, \mathbb{0}, \otimes, \mathbb{1} \rangle$ is a structure with

- a domain $\mathbb{S} = dom(\mathcal{S})$
- two associative binary operators $\oplus$ and $\otimes$ with neutral elements $\mathbb{0}$ and $\mathbb{1}$; and such that
- $\oplus$ is commutative
- $\otimes$ distributes over $\oplus$: $\forall x, y, z \in \mathbb{S}$, $x \otimes (y \oplus z) = (x \otimes y) \oplus (x \otimes z)$,
- $\mathbb{0}$ is absorbing for $\otimes$: $\forall x \in \mathbb{S}$, $\mathbb{0} \otimes x = x \otimes \mathbb{0} = \mathbb{0}$

Intuitively,

$\oplus$ is for selection of a best value

$\otimes$ is for composition of values

Semirings

semiring	domain	$\oplus$	$\mathbb{0}$	$\otimes$	$\mathbb{1}$	natural ordering
Boolean	$\{0,1\}$	$\vee$	0	$\wedge$	1	$\mathbb{1} \leq_{\mathcal{S}} \mathbb{0}$
Viterbi	$[0,1] \subset \mathbb{R}_+$	max	0	.	1	$x \leq_{\mathcal{S}} y$ iff $x \geq y$
min-plus	$\mathbb{R}_+ \cup \{+\infty\}$	min	$+\infty$	$+$	0	$x \leq_{\mathcal{S}} y$ iff $x \leq y$
max-plus	$\mathbb{R} \cup \{-\infty\}$	max	$-\infty$	$+$	0	$x \leq_{\mathcal{S}} y$ iff $x \geq y$

These semirings are

commutative: $\otimes$ is commutative

idempotent: $\forall x,\ x \oplus x = x$

have an induced total natural ordering $\leq_{\mathcal{S}}$ defined by:

$$\forall x, y, x \leq_{\mathcal{S}} y \text{ iff } x \oplus y = x$$

monotonic *wrt* $\leq_{\mathcal{S}}$: $\forall x, y, z,\ x \leq_{\mathcal{S}} y$ implies

$$x \oplus z \leq_{\mathcal{S}} y \oplus z$$

$$x \otimes z \leq_{\mathcal{S}} y \otimes z$$

Weighted CF Grammars

every production rule is attached a weight value in a semiring

here, min-plus semiring:
weight values are penalties (costs)
for 1/4 bars:

$$
\begin{array}{lcl}
q & \xrightarrow{0} & q_0\ q \\
q_0 & \xrightarrow{6} & q_1\ q_2 \\
 & \xrightarrow{15} & - \\
q_1 & \xrightarrow{2} & - \\
q_2 & \xrightarrow{10} & q_3\ q_3 \\
q_3 & \xrightarrow{11} & q_4\ q_4
\end{array}
\qquad
\begin{array}{lcl}
q & \xrightarrow{0} & q_0\ q_0 \\
q_0 & \xrightarrow{12} & q_1\ q_2\ q_2 \\
 & \xrightarrow{1} & \bullet \\
q_1 & \xrightarrow{1} & \bullet \\
q_2 & \xrightarrow{2} & \bullet \\
q_3 & \xrightarrow{4} & -
\end{array}
\qquad
\begin{array}{lcl}
 & & 79 \\
q_0 & \longrightarrow & \bullet_1 \\
q_1 & \xrightarrow{25} & \bullet_1 \\
 & & \\
 & & \\
q_3 & \xrightarrow{1} & \bullet
\end{array}
\qquad
\begin{array}{lcl}
 & & 102 \\
q_0 & \longrightarrow & \bullet_2 \\
q_1 & \xrightarrow{64} & \bullet_2 \\
 & & \\
 & & \\
q_4 & \xrightarrow{1} & \bullet
\end{array}
$$

model with the Viterbi Semiring = PCFG (Probabilistic Context-Free Grammars)

s

weight = notational complexity

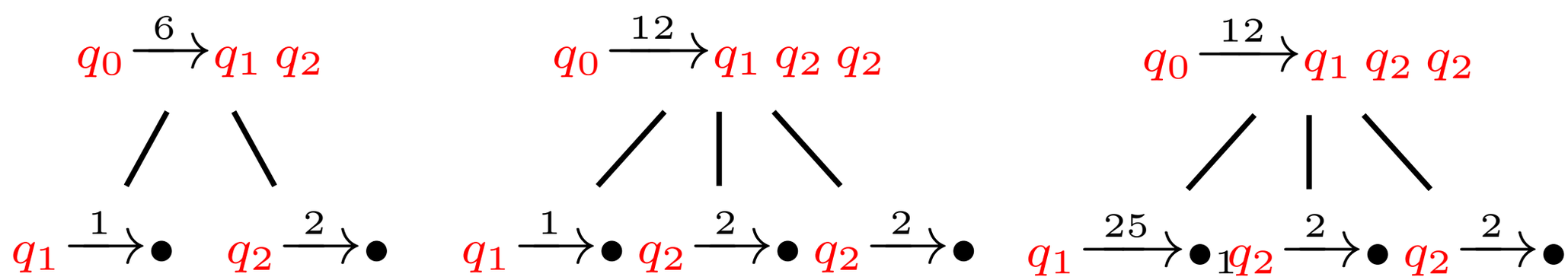
9

17

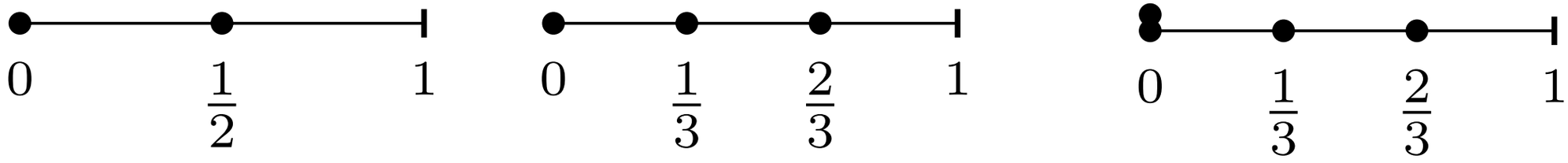
41

= product with $\otimes$ of weights of rules involved

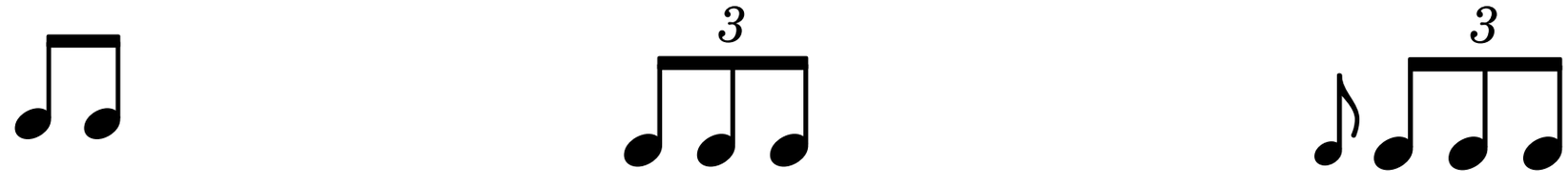
parse tree



serialization



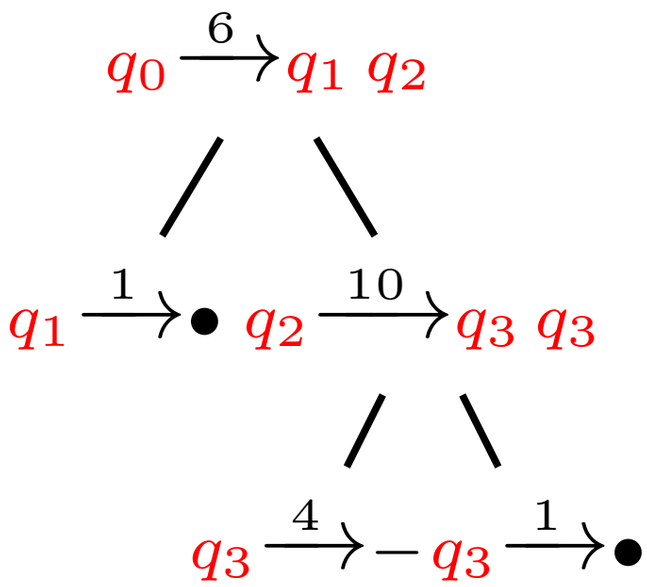
corresponding notation

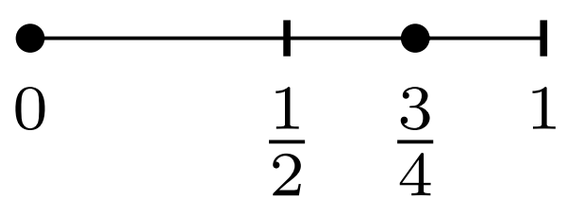


Weight of Parse Trees (2)

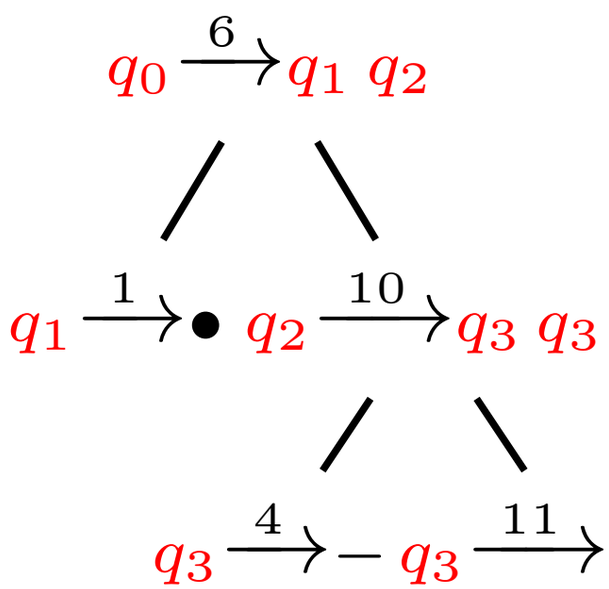
notational complexity

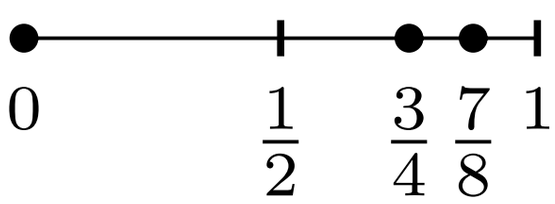
22



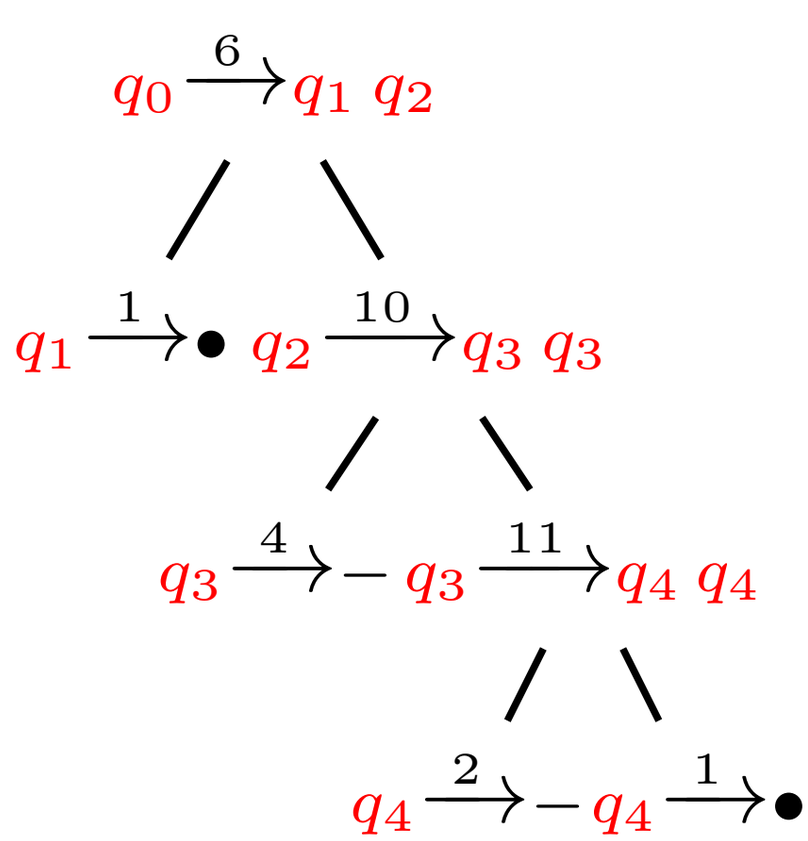


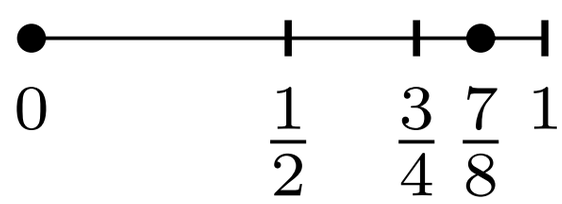
35





35





Time & Tempo

time units:

Real Time Unit (RTU), performance time, in seconds

Musical Time Unit (MTU), score time, in bars

A tempo curve is a monotonically increasing function

$\theta : \quad \mathbb{Q}_+ \to \mathbb{Q}_+$

$\theta : \quad \text{RTU value} \mapsto \text{MTU value}$

A tempo model $\mathcal{M}$ is a set of tempo curves, with restrictions (piecewise linear)

Objective

input: event sequence σ timestamped in RTU

function *fit:* pointwise comparison

tempo curve θ $\theta^{-1}(\|t\|)$ (RTU)

parse tree t of $\mathcal{G}$ $\xrightarrow{\text{serialization}}$ event sequence $\|t\|$ timestamped in MTU

---

given:

- a weighted CFG $\mathcal{G}$ over a semiring $\mathcal{S}$, describing rhythmic notation preferences
- an input sequence of events σ
- a tempo model $\mathcal{M}$

return:

- a tempo curve $\theta \in \mathcal{M}$
- a parse tree t of $\mathcal{G}$ minimizing $weight_{\mathcal{G}}(t) \otimes fit\big(\sigma, \theta^{-1}(\|t\|)\big)$ ***wrt*** $\leq_{\mathcal{S}}$

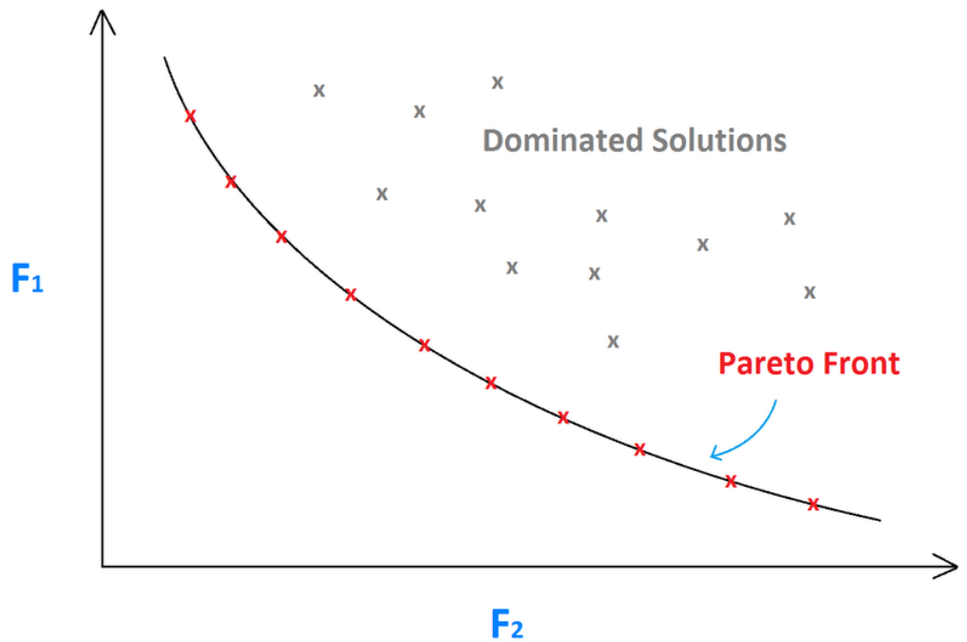
Optimization Problem

We want t that optimizes a combination (with $\otimes$) of

the weight of t *wrt* $\mathcal{G}$	$weight_{\mathcal{G}}(t)$
the fitness of t to the input σ	$fit\big(\sigma, \theta^{-1}(\|t\|)\big)$

- with a Viterbi semiring, $\otimes$ is a probability product to maximize.

- with a min-plus semiring $\otimes$ is a sum to minimize.
  This is similar to scalarization by weighted sum in multi-criteria optimization.



1-best parsing

goal: find a parse tree t of a weighted CFG over $\mathcal{S}$ with minimal weight *wrt* $\leq_{\mathcal{S}}$.

Hypotheses for the semiring $\mathcal{S}$

- $\otimes$ commutative
- $\oplus$ idempotent
- $\mathcal{S}$ monotonic *wrt* $\leq_{\mathcal{S}}$
- $\leq_{\mathcal{S}}$ is total: $\forall x, y,\ x \oplus y = x$ or $x \oplus y = y$

...

$q_0 \xrightarrow{6} q_1\ q_2$ $\quad$ $q_0 \xrightarrow{12} q_1\ q_2\ q_2$

$q_0 \xrightarrow{15} -$ $\quad$ $q_0 \xrightarrow{7} \bullet$ $\quad$ $q_0 \xrightarrow{79} \bullet_1$ $\quad$ $q_0 \xrightarrow{102} \bullet_2$

...

$$best(q_0) = min_{\leq_{\mathcal{S}}} \left\{ \begin{array}{l} 6 \otimes best(q_1) \otimes best(q_2), \\ 12 \otimes best(q_1) \otimes best_1(q_2) \otimes best_1(q_2), \\ 15, 7, 79, 102 \end{array} \right\} \quad \textit{etc}$$

for acyclic grammars. for cyclic ones: generalisation of Dijkstra algorithm to hypergraphs.

1-best parsing

goal: find a parse tree t of a weighted CFG over $\mathcal{S}$ with minimal weight *wrt* $\leq_{\mathcal{S}}$.

Hypotheses for the semiring $\mathcal{S}$

- $\otimes$ commutative
- $\oplus$ idempotent
- $\mathcal{S}$ monotonic *wrt* $\leq_{\mathcal{S}}$
- $\leq_{\mathcal{S}}$ is total: $\forall x, y,\ x \oplus y = x$ or $x \oplus y = y$

...

$q_0 \xrightarrow{6} q_1\ q_2$ $\quad$ $q_0 \xrightarrow{12} q_1\ q_2\ q_2$

$q_0 \xrightarrow{15} -$ $\quad$ $q_0 \xrightarrow{7} \bullet$ $\quad$ $q_0 \xrightarrow{79} \bullet_1$ $\quad$ $q_0 \xrightarrow{102} \bullet_2$

...

$$best(q) = \bigoplus_{\rho_0 = q \xrightarrow{w_0} a} \rho_0 \oplus \Big[\bigoplus_{\rho = q \xrightarrow{w} q_1 \dots q_n} \rho\big(best(q_1), \dots, best(q_n)\big) \Big]$$

for acyclic grammars. for cyclic ones: generalisation of Dijkstra algorithm to hypergraphs.

Transcription by 1-best parsing

principle: apply a 1-best algorithm to a weighted CFG $\mathcal{G} \times \mathcal{G}_\sigma$ combining $\mathcal{G}$ and a representation of the input σ

ex. steady tempo, all-left alignments.

▶ extend $q_0 \xrightarrow{12} q_1\ q_2\ q_2$ of $\mathcal{G}$ into

$$q_0.[0,1[\xrightarrow{12 \otimes \delta(\sigma,[0,1[)} q_1.[1,\frac{1}{3}[\ q_2.[\frac{1}{3},\frac{2}{3}[\ q_2.[\frac{2}{3},1[$$

where $\delta(\sigma,[0,1[) = \mathbb{1}$ if $\sigma|_{[0,1[} \neq \emptyset$ and $\delta(\sigma,[0,1[) = \mathbb{0}$ otherwise ($+\infty$ for min-plus)

▶ extend $q_1 \xrightarrow{1} \bullet$ of $\mathcal{G}$ into $q_1.[1,\frac{1}{3}[\xrightarrow{1 \otimes \varepsilon(\sigma,[1,\frac{1}{3}[)} \bullet$ where

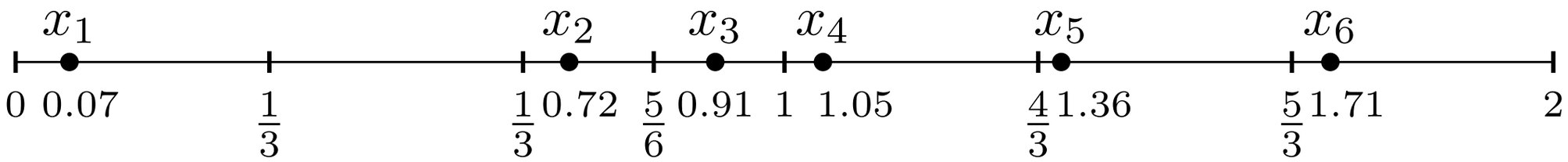
- if σ has 1 point x in the first half of $[1,\frac{1}{3}[$ and none in the second half, $\varepsilon(\sigma,[1,\frac{1}{3}[)$ is a normalized distance between x and 1,
- otherwise $\varepsilon(\sigma,[1,\frac{1}{3}[) = \mathbb{0}$

For efficiency, the combined grammar is computed on-the-fly

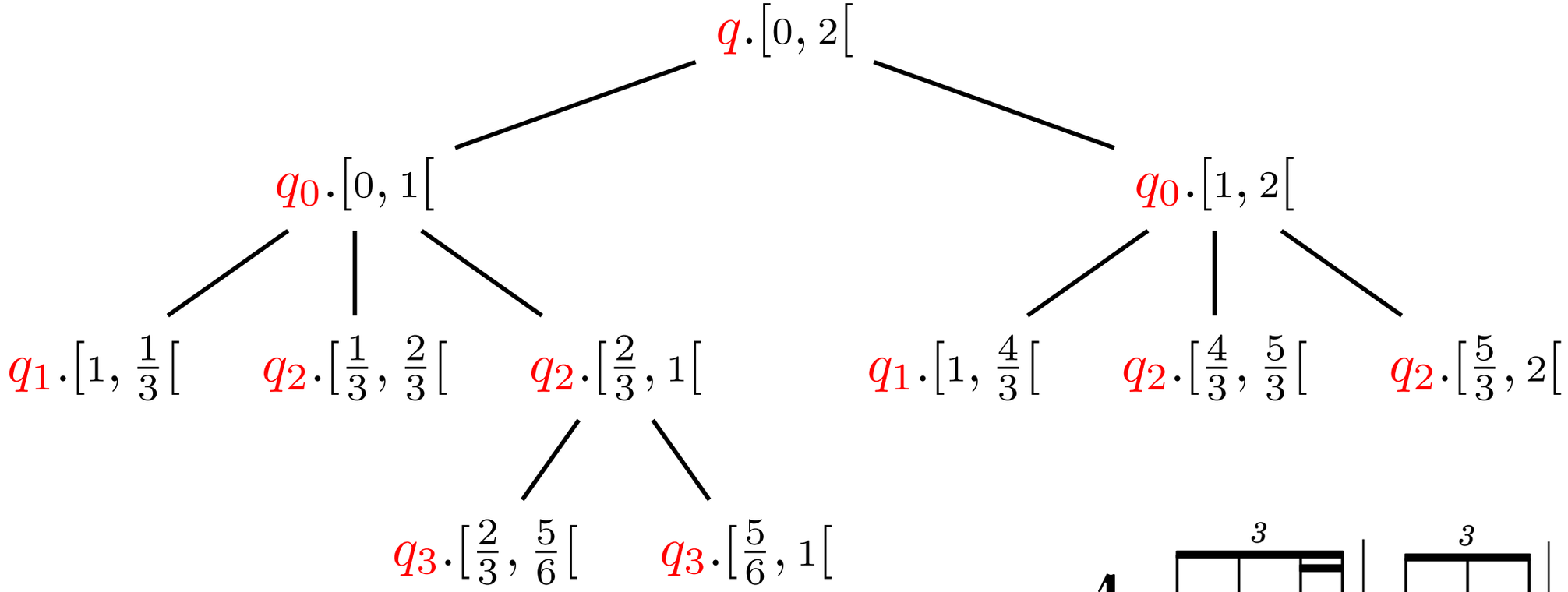
Transcription by 1-best parsing

ex.1: steady tempo, all-left alignments

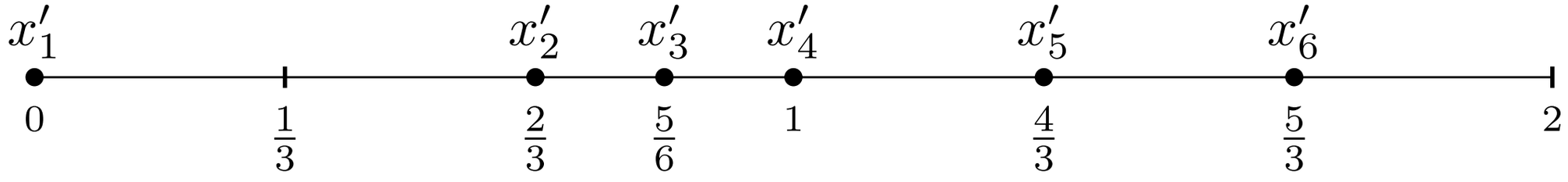
input



best parse tree



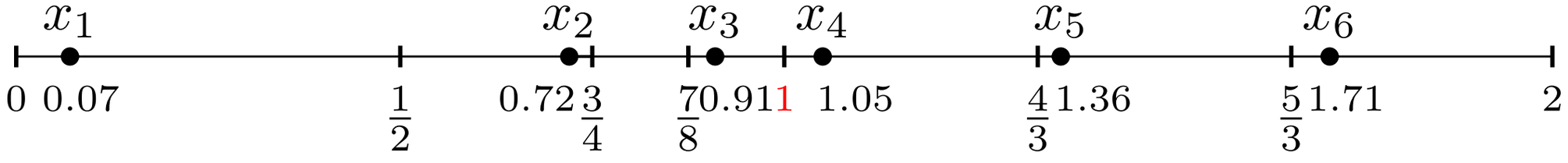
serialization



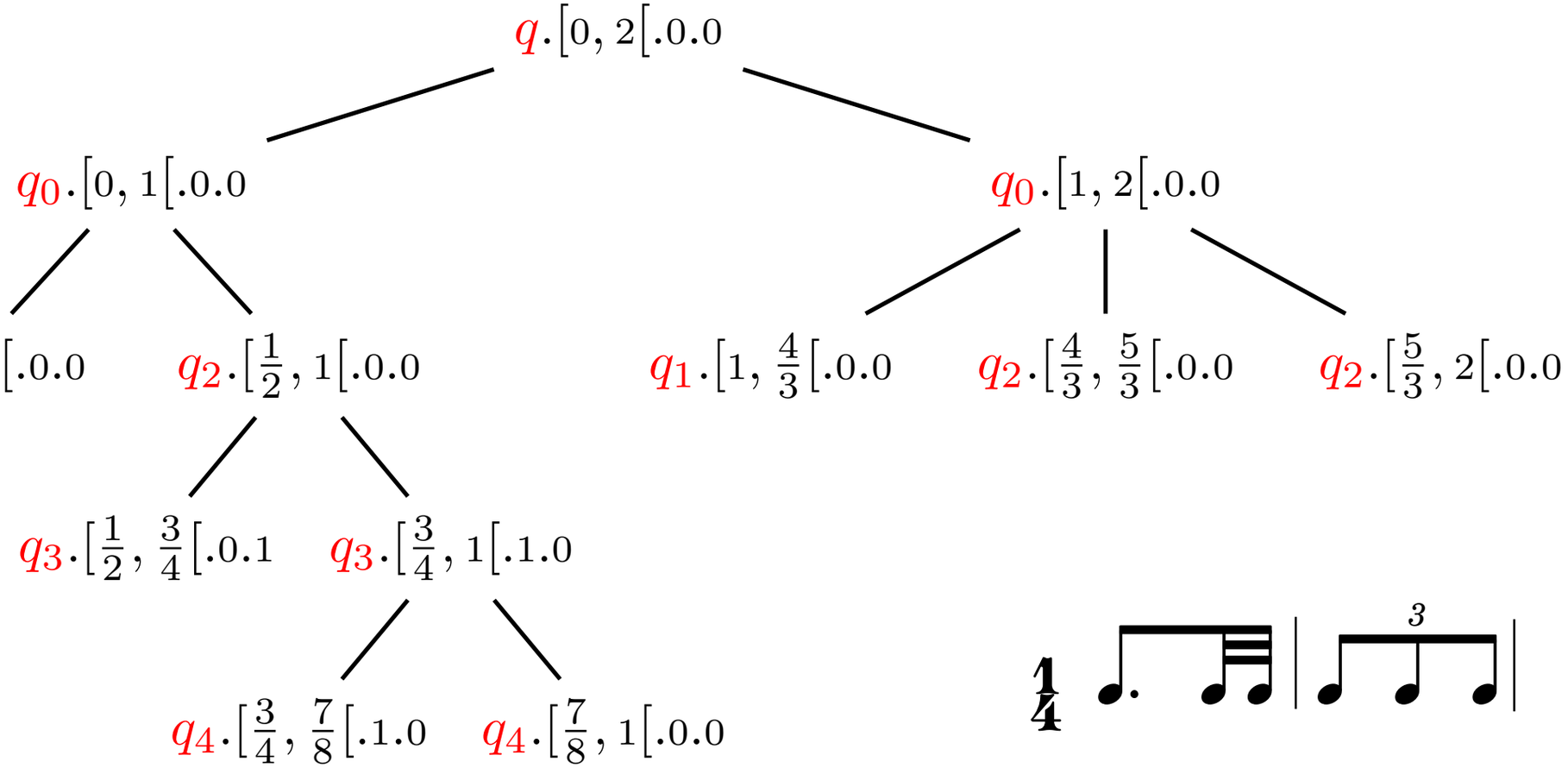
Transcription by 1-best parsing

ex.2: another extension for transcription with steady tempo, alignments to left or right

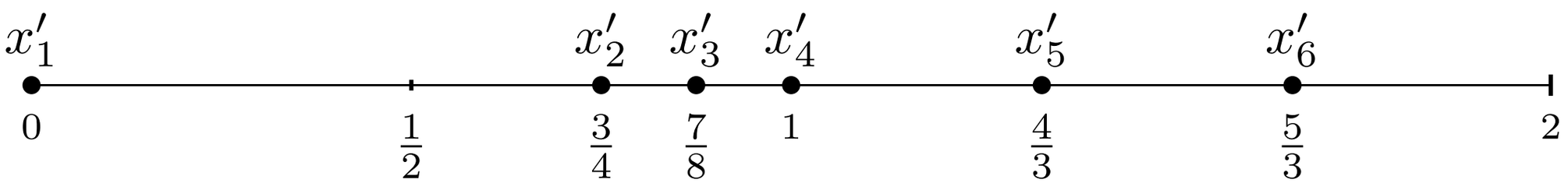
input



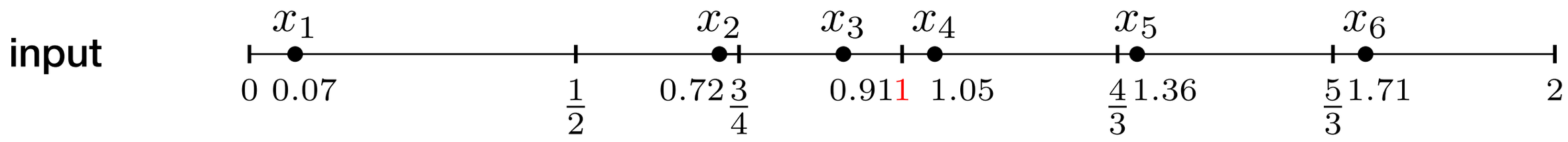
best parse tree



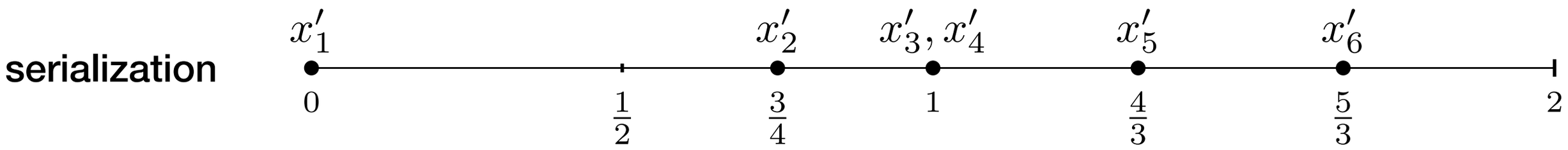
serialization



Transcription by 1-best parsing



if we reduce the penalty for grace-notes, $q_1 \xrightarrow{7} \bullet_1$
the best parse tree corresponds to:



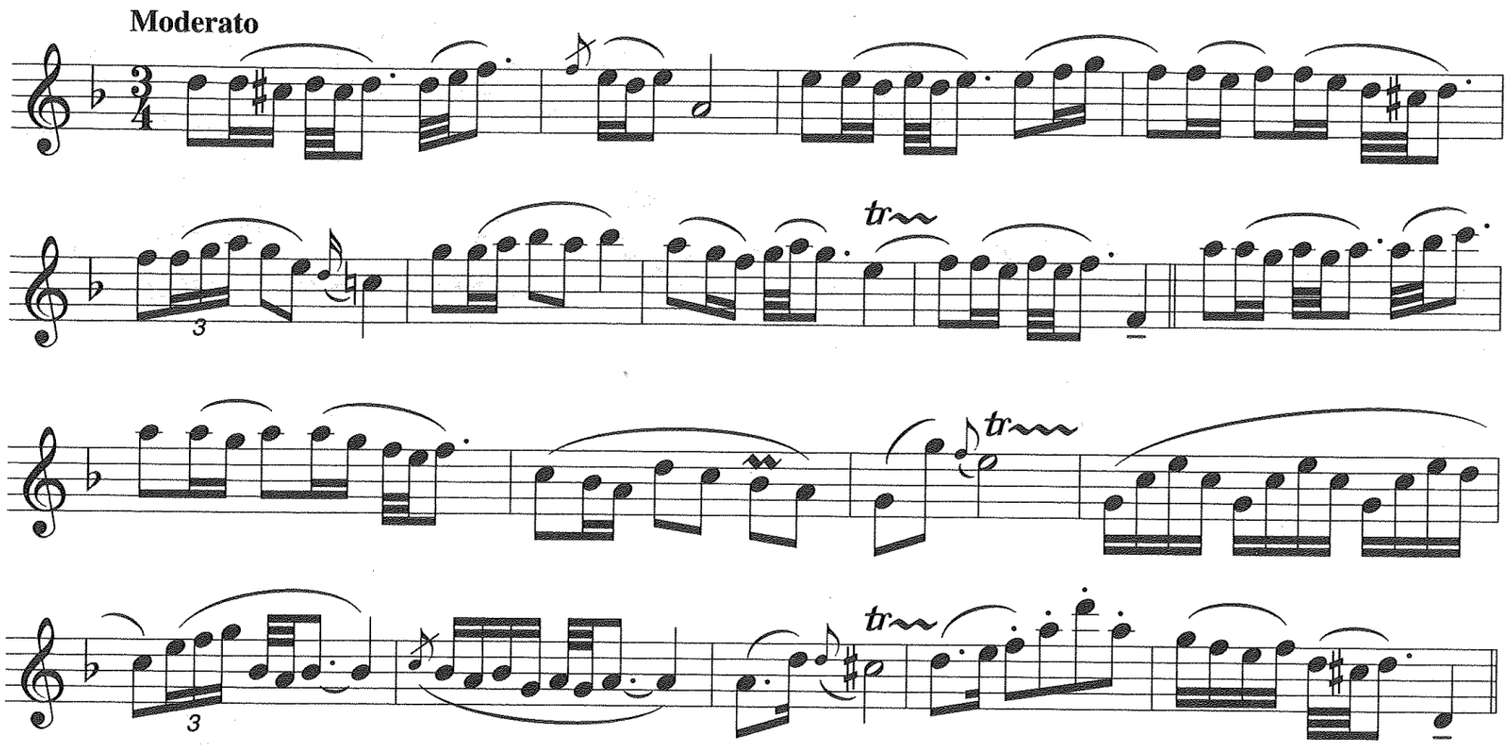
Implementation, Results

C++ library. MIDI input, XML/MEI or Lilypond output

https://gitlab.inria.fr/qparse/qparselib
https://qparse.gitlabpages.inria.fr

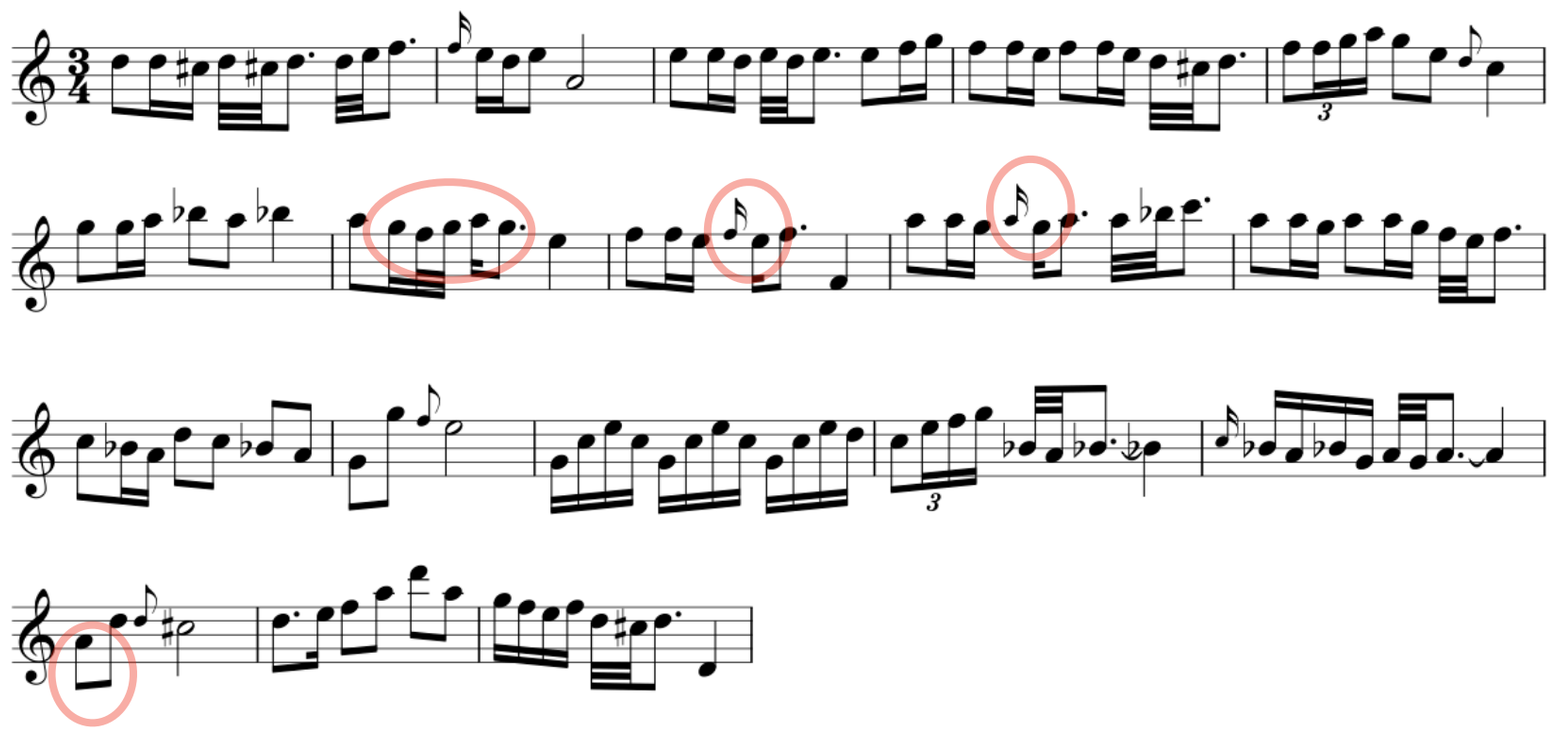
original score

Polonaise in D minor
from Notebook for
Anna Magdalena Bach
BWV Anh II 128



transcription of MIDI recording (100 ms) by qparse with generic grammar

MEI output, display with Verovio



Conclusion

https://gitlab.inria.fr/qparse/qparselib
https://qparse.gitlabpages.inria.fr

Rhythm transcription based on language theoretic models
- conversion of linear data (MIDI) into structured data (scores)
  = parsing
- quantitative (based on semiring)

- Symbolic Tree Automata (more general than CFG)
  - labels for inner nodes
  - infinite set of labels for leaves
  - guards in transitions

- Training rhythm grammars from corpus (Francesco, SMC 2019)

- Piano scores (process onsets & offsets)