# Confluence: decidability results & what to do without

Florent Jacquemard

5th International Workshop on Confluence - IWC'16

## 1.(un)decidability of confluence

- map of results on (un)decidability
  of confluence and Uniqueness of Normal Forms
- importance of linearity
- open problems
- links to regularity preservation

## 2.enumeration of equivalent terms

- case study: terms-based representation of durations
  in music western notation
- non-confluent TRS
- automata-based representation of equivalent terms sets
- lazy ordered enumeration of equivalence classes
- links to regularity preservation

decidability

undecidability

## confluence

ground TRS
[Oyamaguchi 87]
[Dauchet Tison 88]

left-linear
right-ground TRS
[Dauchet et al 90]

linear TRS (depth 2)
[Verma et al 01]

**EXPTIME**

**PTIME**

ground TRS
[Comon et al 01]
[Tiwari 02]

left-shallow-linear
right-ground TRS
[Tiwari 02]

length 2 SRS
[Sakai 07]

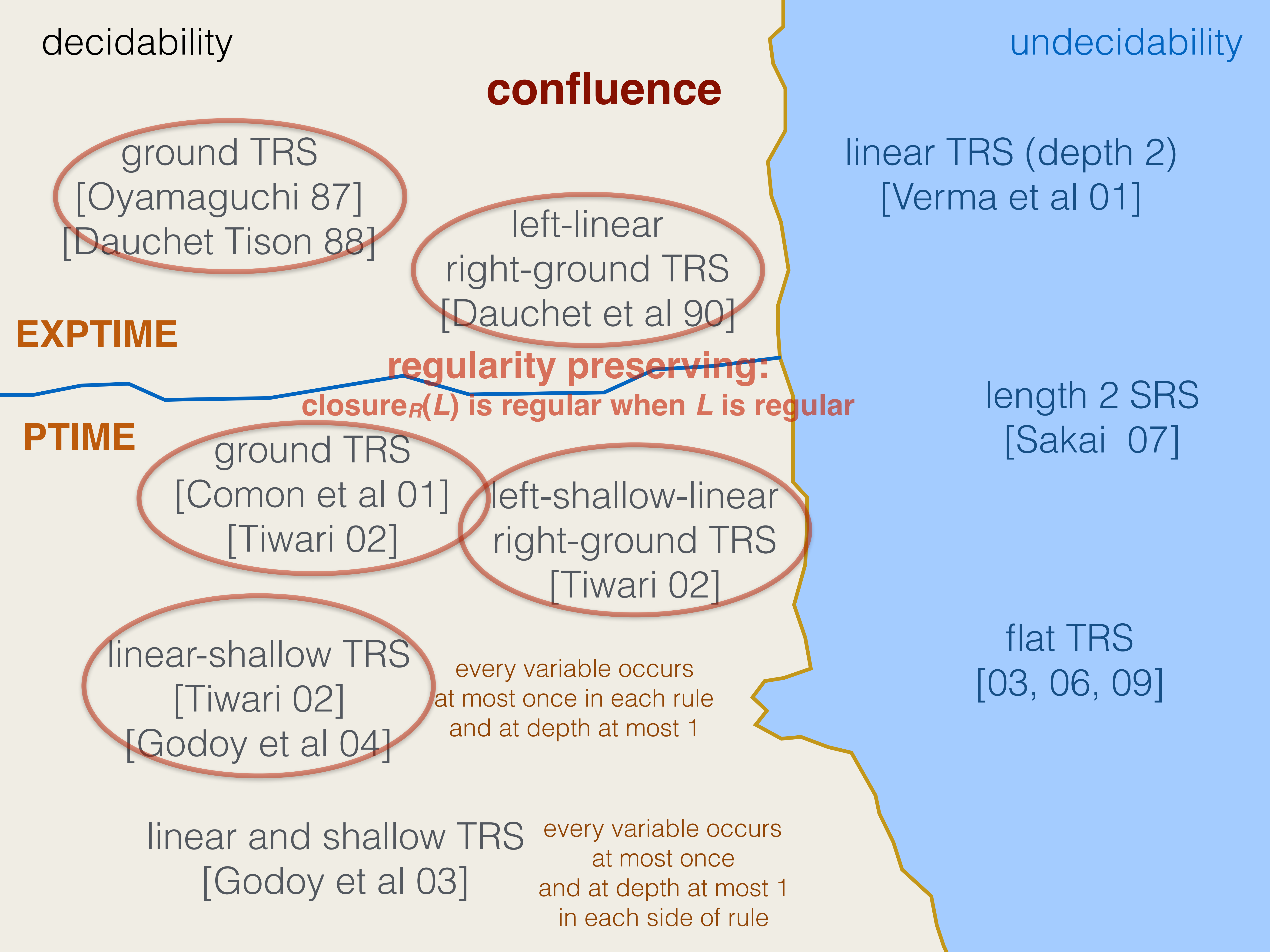
linear-shallow TRS
[Tiwari 02]
[Godoy et al 04]

every variable occurs
at most once in each rule
and at depth at most 1

flat TRS
[03, 06, 09]

linear and shallow TRS
[Godoy et al 03]

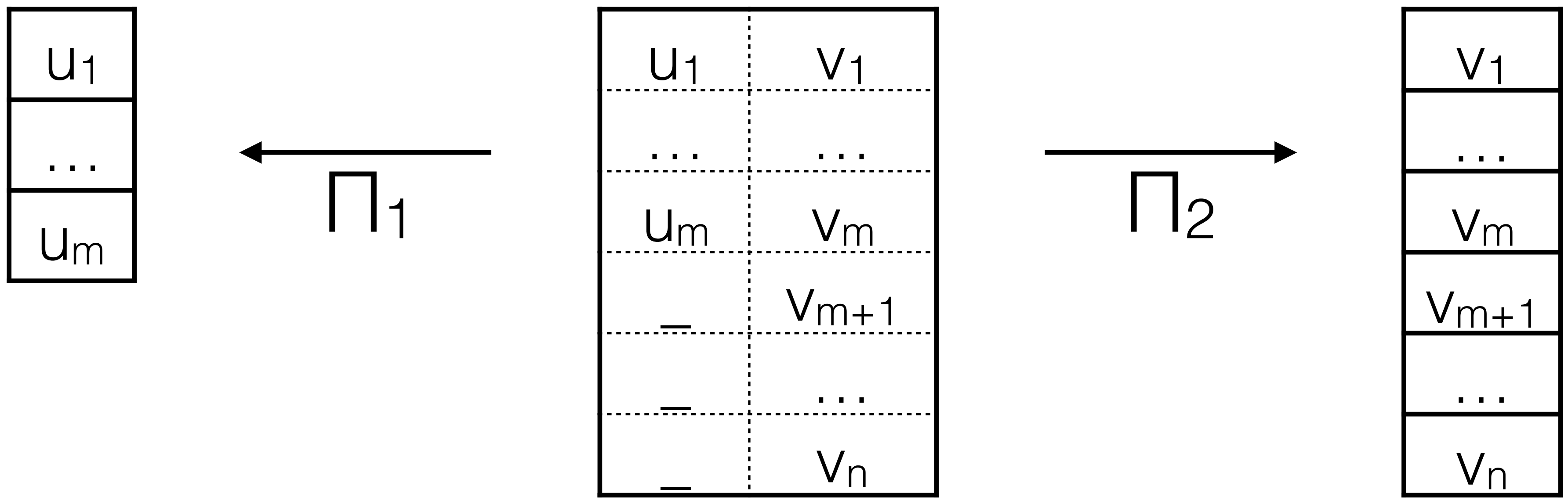
every variable occurs
at most once
and at depth at most 1
in each side of rule


decidability
undecidability
confluence
ground TRS
[Oyamaguchi 87]
[Dauchet Tison 88]
left-linear
right-ground TRS
[Dauchet et al 90]
linear TRS (depth 2)
[Verma et al 01]
EXPTIME
regularity preserving:
closure$_R$($L$) is regular when $L$ is regular
PTIME
length 2 SRS
[Sakai 07]
ground TRS
[Comon et al 01]
[Tiwari 02]
left-shallow-linear
right-ground TRS
[Tiwari 02]
flat TRS
[03, 06, 09]
linear-shallow TRS
[Tiwari 02]
[Godoy et al 04]
every variable occurs
at most once in each rule
and at depth at most 1
linear and shallow TRS
[Godoy et al 03]
every variable occurs
at most once
and at depth at most 1
in each side of rule

# undecidability of confluence for flat TRS

[Jacquemard 03, Mitsuhashi, Oyamaguchi, J. 06 ]

1. undecidability of reachability:
   PCP encoding (*shifted pairing* technique)
2. reduction of reachability to confluence




simpler proofs in [Godoy Hernandez 09]

# undecidability of confluence for flat TRS

[Jacquemard 03, Mitsuhashi, Oyamaguchi, J. 06 ]

1. undecidability of reachability:
   PCP encoding (*shifted pairing* technique)
2. reduction of reachability to confluence

$$
\begin{array}{llllllllllll}
0 & \to & f( & q_A^{(3)}, & q_A^{(4)}, & q_A^{(5)}, & q_B^{(13)}, & q_B^{(14)}, & q_A^{(6)}, & q_B^{(15)}, & q_B^{(16)}) & \\
 & & & T_A^{(3)} & T_A^{(4)} & T_A^{(5)} & T_B^{(13)} & T_B^{(14)} & T_A^{(6)} & T_B^{(15)} & T_B^{(16)} & \\
 & & & P^{(3,1)} & P^{(4,2)} & P^{(5,1)} & S^{(13,11)} & S^{(14,12)} & P^{(6,2)} & S^{(15,11)} & S^{(16,12)} & \\
 & & f( & x_1, & x_2, & x_1, & y_{11}, & y_{12}, & x_2, & y_{11}, & y_{12}) & \\
 & & \downarrow & & & & & & & & & \\
 & & g( & x_1, & x_2, & x_1, & y_{11}, & y_{12}, & x_2, & y_{11}, & y_{12}) & \\
 & & & P^{(1,0)} & P^{(2,0)} & \Pi_1^{(1,17)} & S^{(11,17)} & S^{(12,18)} & \Pi_2^{(2,18)} & S^{(11,10)} & S^{(12,10)} & \\
 & & g( & x_0, & x_0, & y_{17}, & y_{17}, & y_{18}, & y_{18}, & y_{10}, & y_{10}) & \to \; 1
\end{array}
$$

simpler proofs in [Godoy Hernandez 09]

decidability

undecidability

**confluence**
(non-linear TRSs)

right-ground TRS
[Kaiser 05]

shallow and right-linear TRS
[Godoy, Tiwari 05]

flat TRS
[03, 06, 09]

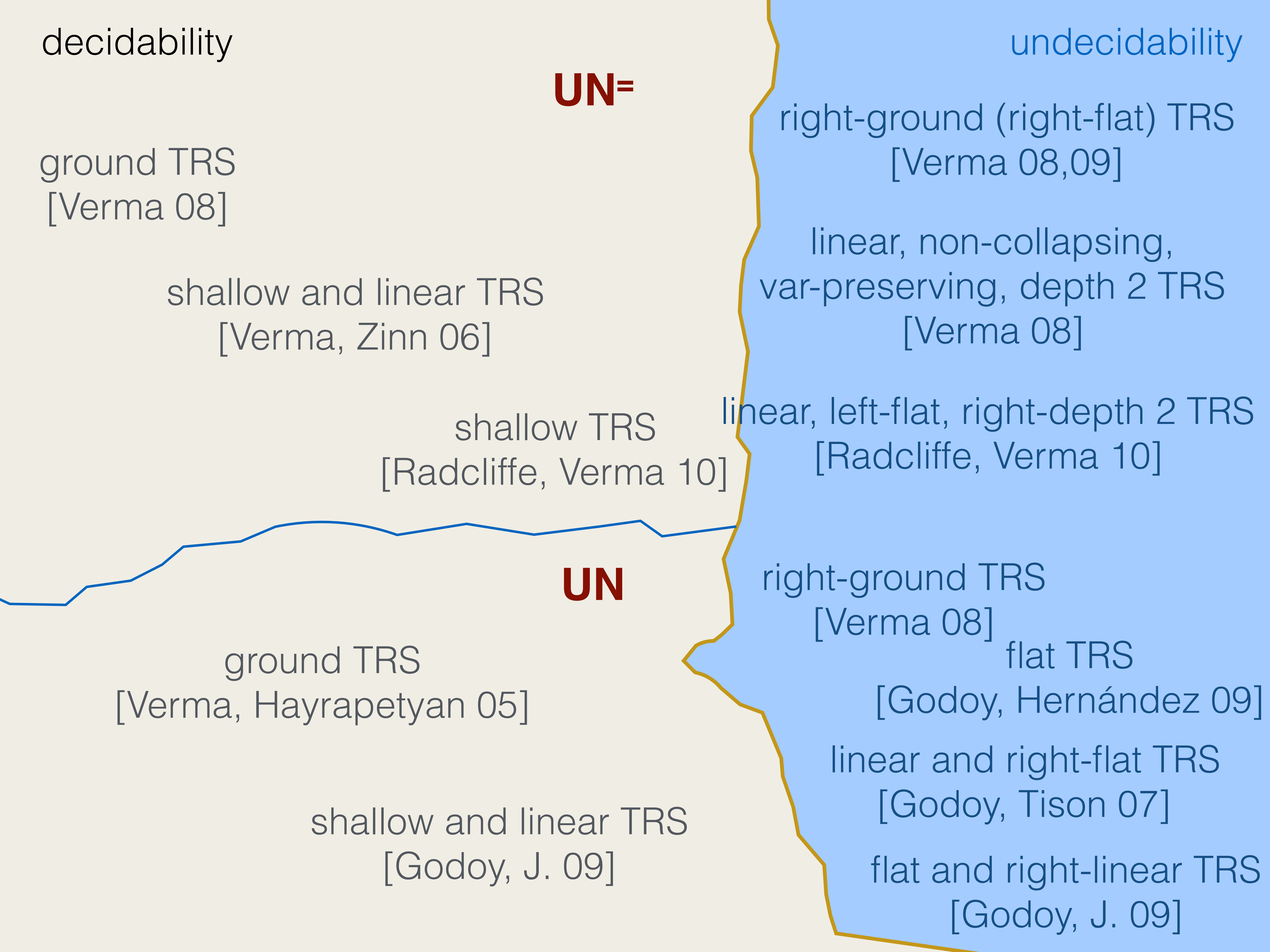
# uniqueness of NF

uniqueness of NF (**$UN^=$**):
no two distinct normal forms can be equivalent modulo the TRS.

confluence ⇒ $UN^=$

unique normalization (**UN**):
every term can reach at most one normal form using the TRS.

$UN^=$ ⇒ UN (the converse is not true)


decidability
undecidability
UN=
right-ground (right-flat) TRS
[Verma 08,09]
ground TRS
[Verma 08]
linear, non-collapsing,
var-preserving, depth 2 TRS
[Verma 08]
shallow and linear TRS
[Verma, Zinn 06]
linear, left-flat, right-depth 2 TRS
[Radcliffe, Verma 10]
shallow TRS
[Radcliffe, Verma 10]
UN
right-ground TRS
[Verma 08]
flat TRS
[Godoy, Hernández 09]
ground TRS
[Verma, Hayrapetyan 05]
linear and right-flat TRS
[Godoy, Tison 07]
shallow and linear TRS
[Godoy, J. 09]
flat and right-linear TRS
[Godoy, J. 09]

# confluence and UN under rewrite strategies

new [Ishizuki, Sakai, Oyamaguchi IWC 16]
conditions for confluence of innermost-terminating TRS

for bottom-up term rewriting?
[Durand, Sénizergues 07]

# open questions: decidability of confluence for

## flat and non-collapsing TRS?

$$\Pi_1 := \{\langle a, b\rangle(x) \to a(x) \mid a \in \Sigma, b \in \Sigma \cup \{\_\}\} \cup \{\langle \_, b\rangle(x) \to x \mid b \in \Sigma\}$$

$$\Pi_2 := \{\langle a, b\rangle(x) \to b(x) \mid a \in \Sigma \cup \{\_\} b \in \Sigma\} \cup \{\langle a, \_\rangle(x) \to x \mid a \in \Sigma\}$$

(collapsing rules in PCP reduction by shifted pairing)

## regularity preserving TRSs

- regularity preservation used in decision of confluence, e.g.
  - local confluence:
    $$s_1 \xleftarrow[\mathcal{R}]{} t \xrightarrow[\mathcal{R}]{} s_2 \Rightarrow \{s_1' \mid s_1 \xrightarrow[\mathcal{R}]{*} s_1'\} \cap \{s_2' \mid s_2 \xrightarrow[\mathcal{R}]{*} s_2'\} \neq \emptyset$$
  - original decidability proofs for ground TRS
- decidability for other regularity preserving TRSs?
  - right-linear and finite-path-overlapping TRS [Takai et al 00]
  - layered traducing TRS [Seki et al 02]

## 1.(un)decidability of confluence

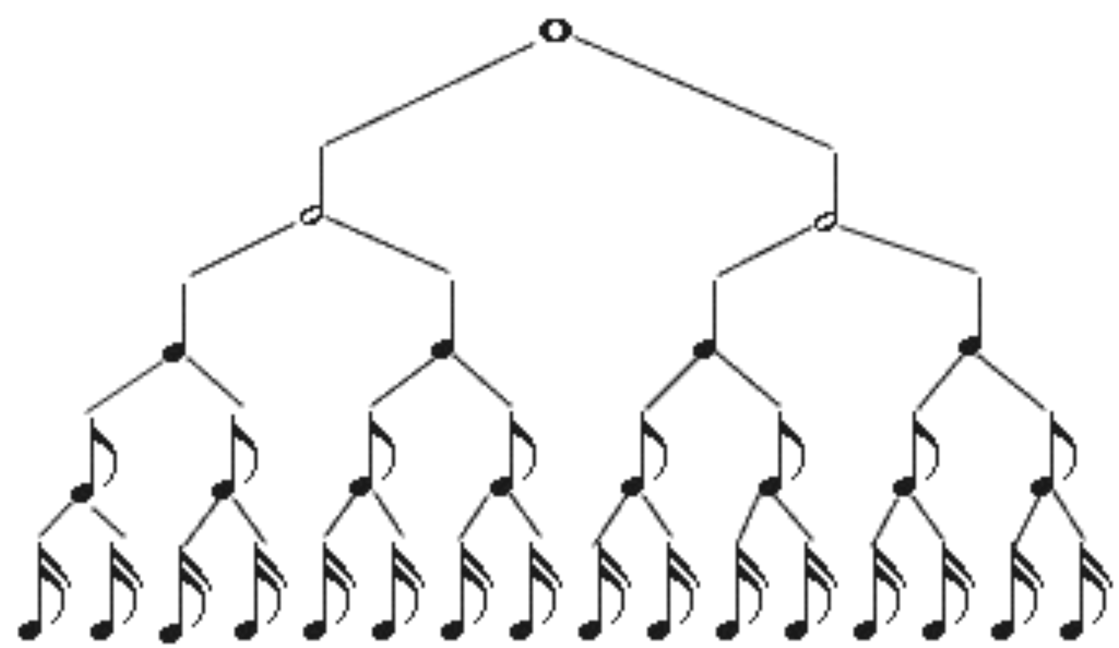
- map of results on (un)decidability
  of confluence and Uniqueness of Normal Forms
- importance of linearity and flatness
- open problems
- links to regularity preservation

## 2.enumeration of equivalent terms

- case study: terms-based representation of durations
  in music western notation
- non-confluent TRS
- automata-based representation of equivalent terms sets
- lazy ordered enumeration of equivalence classes
- links to regularity preservation

# notation of durations in music

in common western music notation,
durations are defined relative to a periodic pulse (beat)
and hierarchically, by recursive subdivisions (nested fractions)



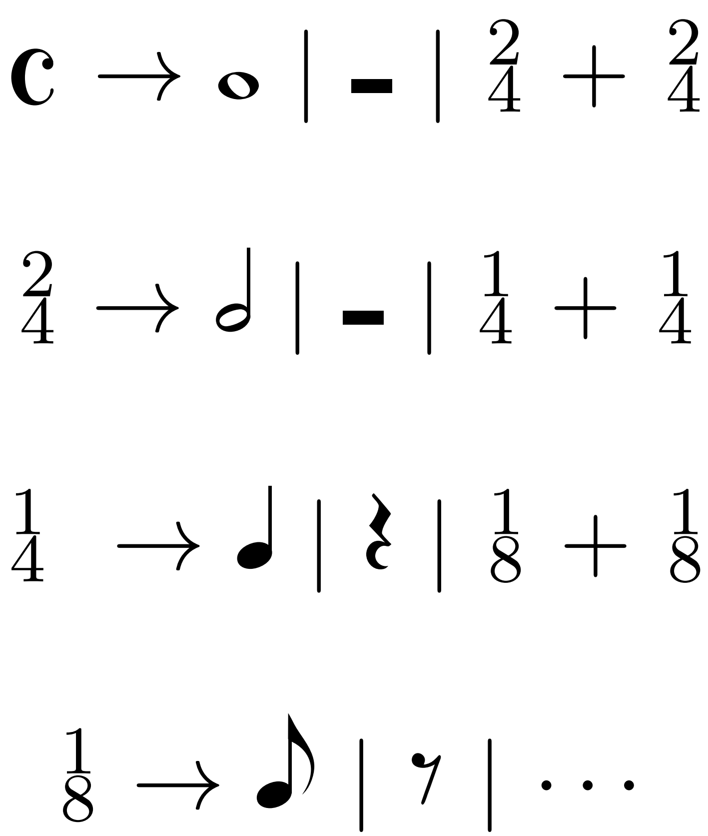
survey in

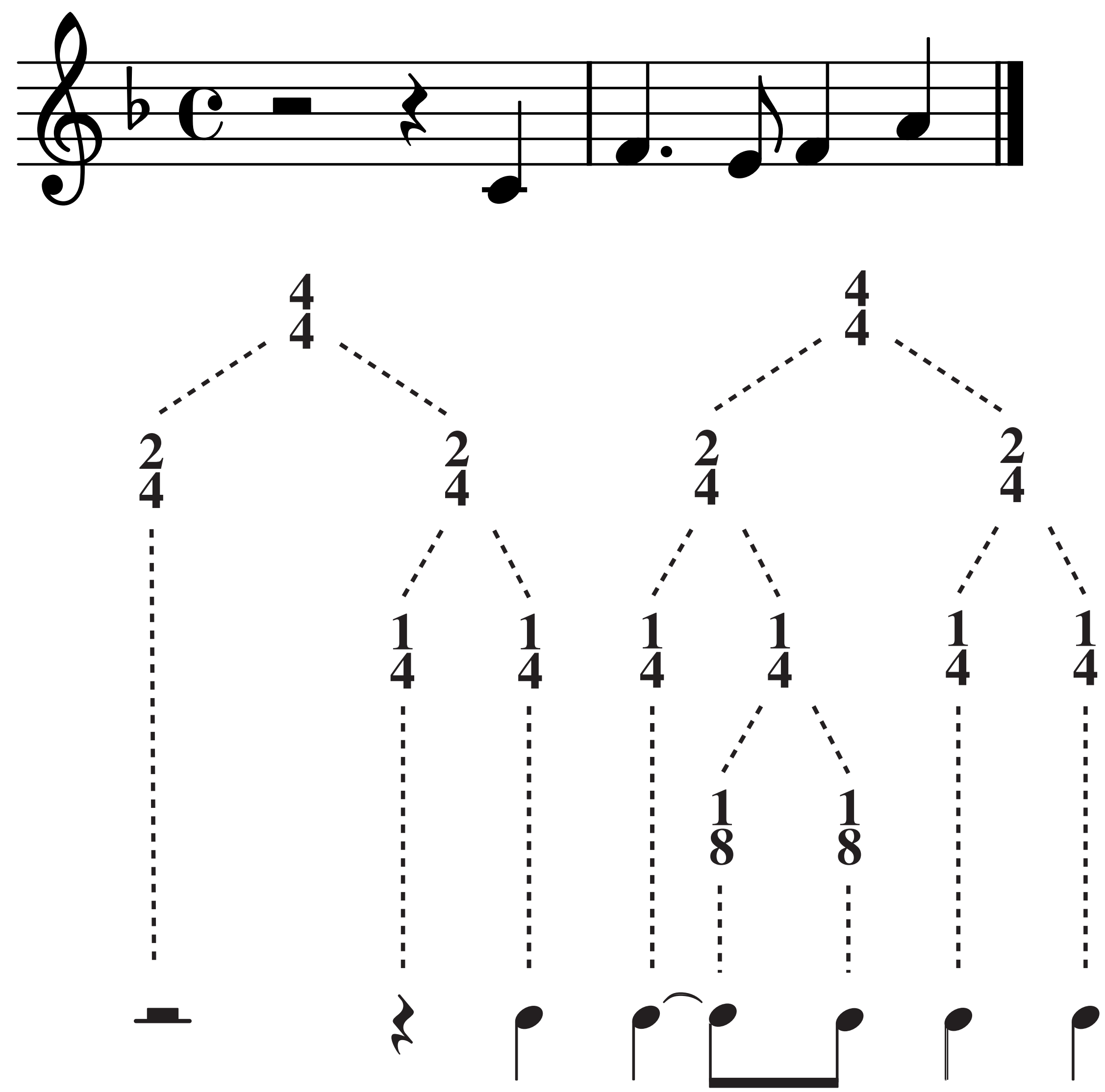
Rizo
Symbolic music comparison with tree data structures
PhD thesis U. Alicante, 2010

# rhythms as syntax trees

Longuet-Higgins
The perception of music
I.S.R., 1978

Lee
The rhythmic interpretation of simple musical sequences
Musical Structure and Cognition, 1985





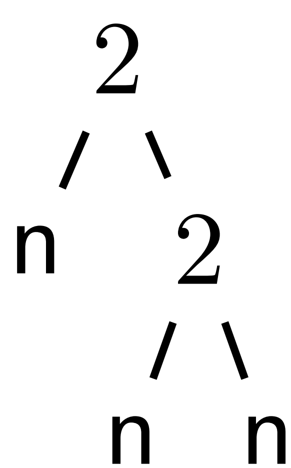
- ➡ symbolic constraints (*e.g.* sum = 1)
- ➡ definition of schemas for rhythm notations as regular tree languages
- ➡ definition of syntactic transformations or equations

# rhythm trees (RT): syntax
(simplified version)

Laurson
Patchwork: A Visual Programming Language
Helsinki: Sibelius Academy, 1996

hierarchical encoding of durations
as terms over a finite (and small) signature $\Sigma$

- one symbol *p* of arity *p* for each $1 < p \leq$ bound (typically 13)
- constant symbols: **n** (note), **r** (rest), **o** (*tie*=composition)




# rhythm trees (RT) : semantics

we associate a duration to every node:

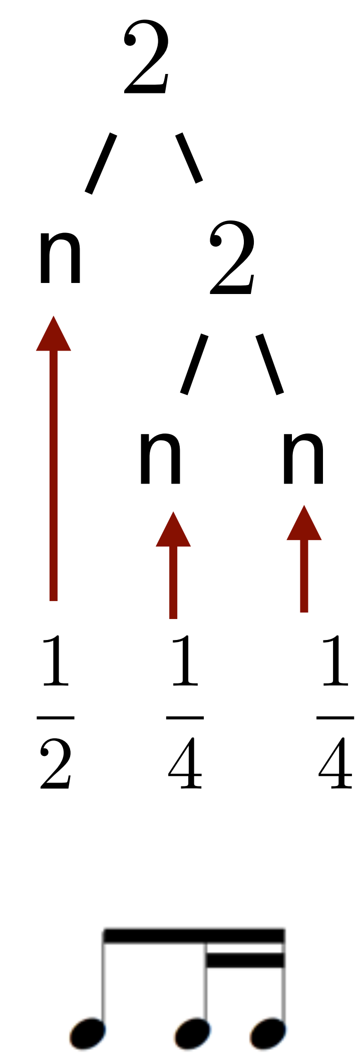
dur(*root*) = 1 beat or 1 measure

$$\text{dur}(node) = \frac{\text{dur}(parent)}{\text{arity}(parent)} + \text{pdur}(node)$$

pdur(*node*) = dur(*next-leaf*)
if *next-leaf* exists and is labeled with **o**

pdur(*node*) = 0 otherwise

rhythmic value = sequence of rational numbers
= duration of leaves (in dfs traversal) labelled **n** or **r**



# rhythm trees (RT) : semantics

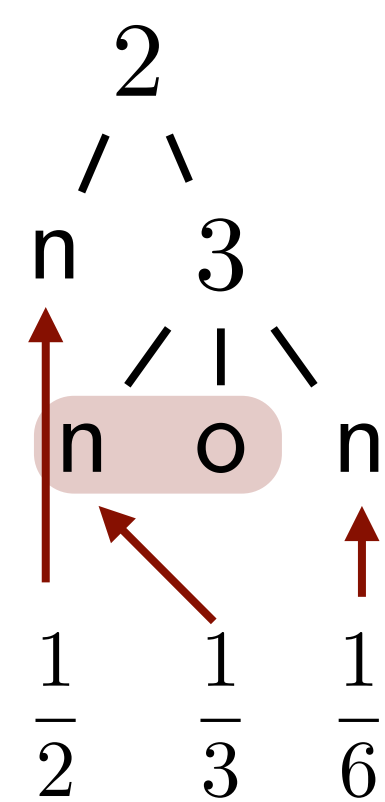
we associate a duration to every node:

dur(*root*) = 1 beat or 1 measure

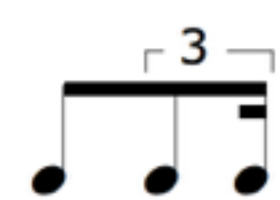
$$\mathrm{dur}(\mathit{node}) = \frac{\mathrm{dur}(\mathit{parent})}{\mathrm{arity}(\mathit{parent})} + \mathrm{pdur}(\mathit{node})$$

pdur(*node*) = dur(*next-leaf*)
if *next-leaf* exists and is labeled with **o**

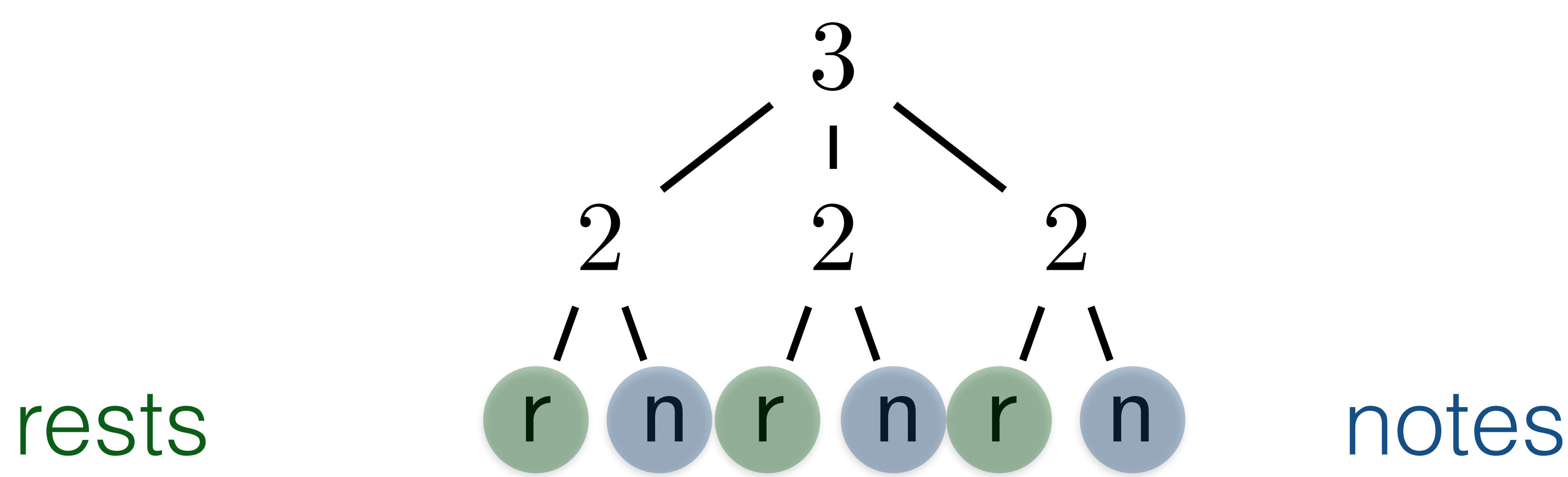
pdur(*node*) = 0 otherwise




rhythmic value = sequence of rational numbers
= duration of leaves (in dfs traversal) labelled **n** or **r**

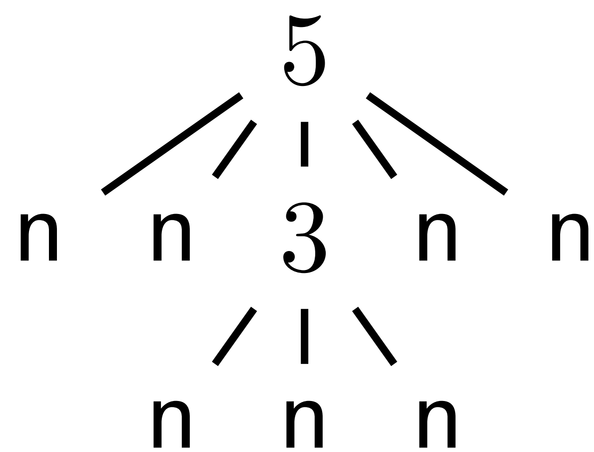


# rhythmic value

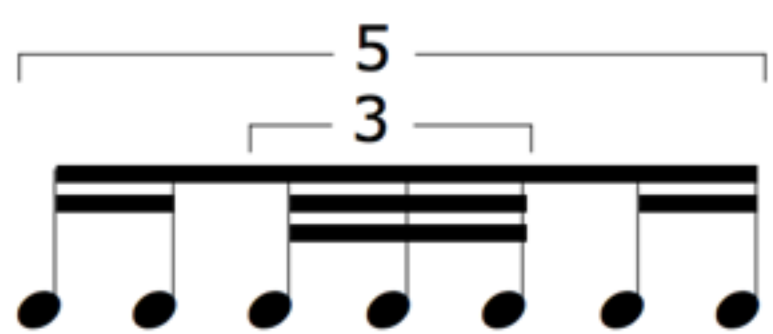



rhythmic value $\left[\frac{1}{6}\right]\ \frac{1}{6}\ \left[\frac{1}{6}\right]\ \frac{1}{6}\ \left[\frac{1}{6}\right]\ \frac{1}{6}$

# rhythmic value (nested tuplets)

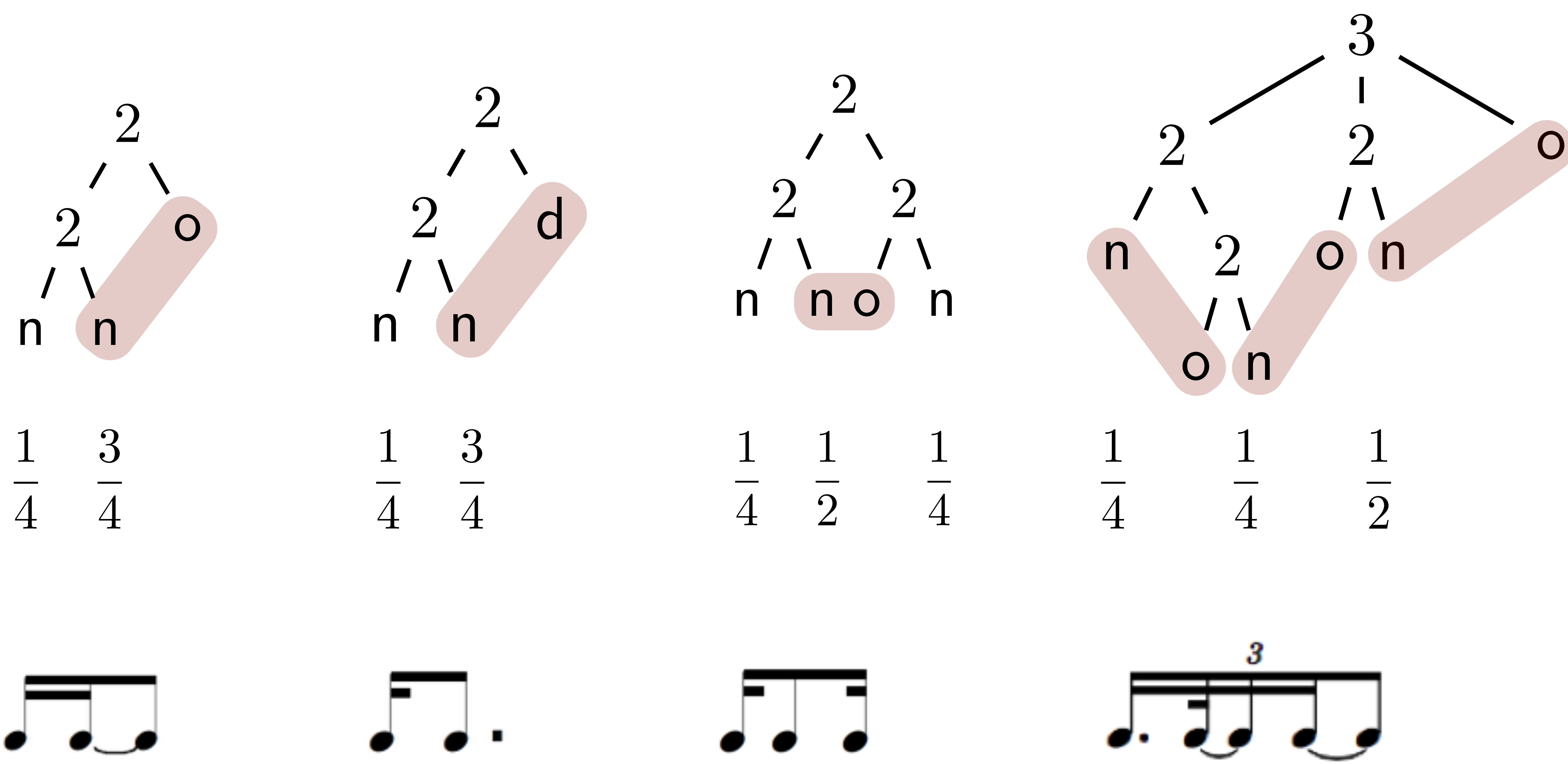



rhythmic value $\frac{1}{5}\ \frac{1}{5}\ \frac{1}{15}\ \frac{1}{15}\ \frac{1}{15}\ \frac{1}{5}\ \frac{1}{5}$



# ties and dots

we sum durations for subsequences
of leafs of the form **n o ... o**



# rewrite rules

Structural Theory of Rhythm Notation MEI'15, MCM'15

addition of rests

$$p(\mathsf{r}, \ldots, \mathsf{r}) \to \mathsf{r}$$

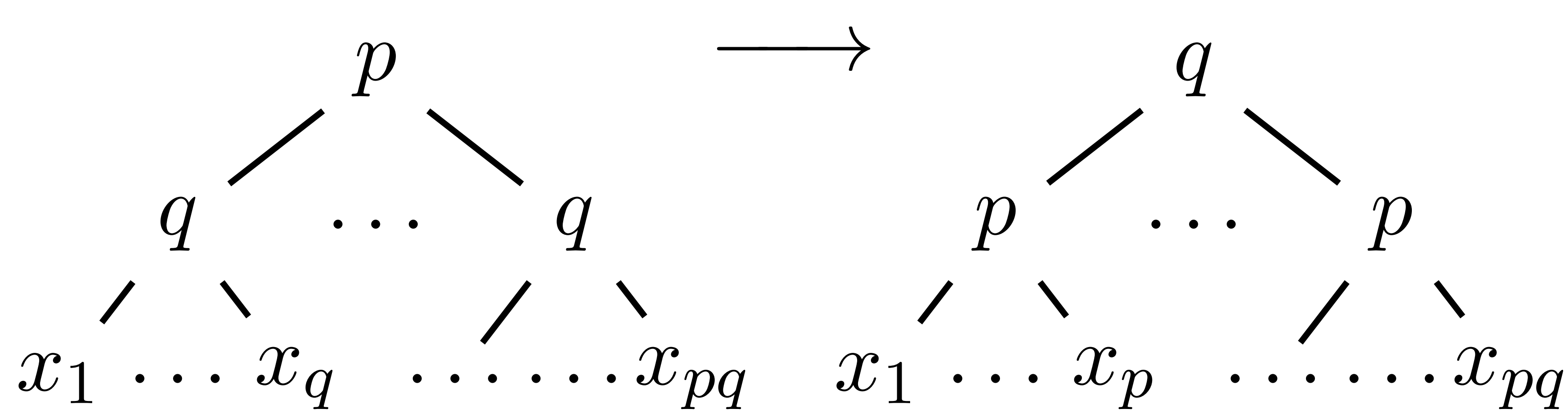
$$p(\mathsf{r}, \mathsf{o}, \ldots, \mathsf{o}) \to \mathsf{r}$$

normalization of ties

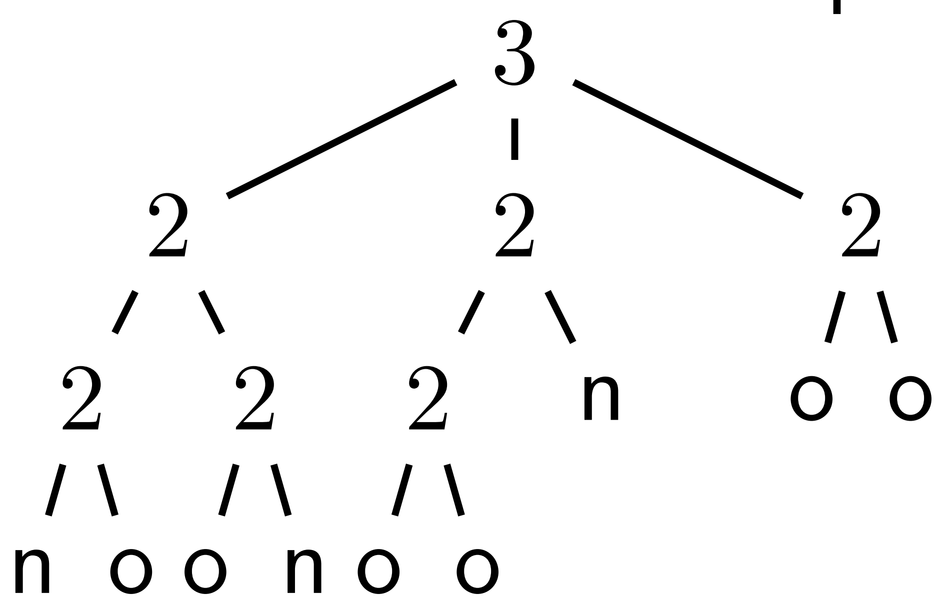
$$p(\mathsf{o}, \ldots, \mathsf{o}) \to \mathsf{o}$$

$$p(\mathsf{n}, \mathsf{o}, \ldots, \mathsf{o}) \to \mathsf{n}$$

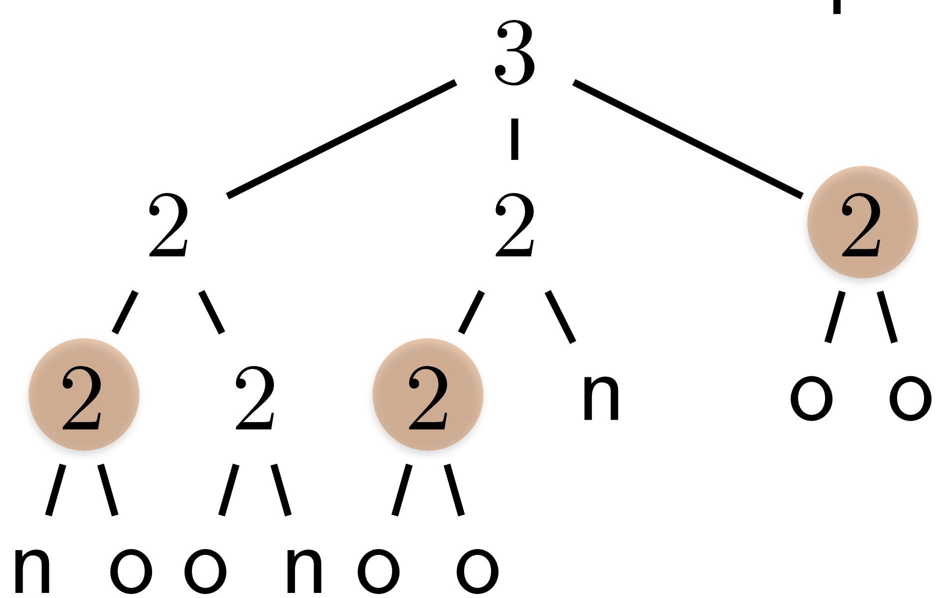
arity switch



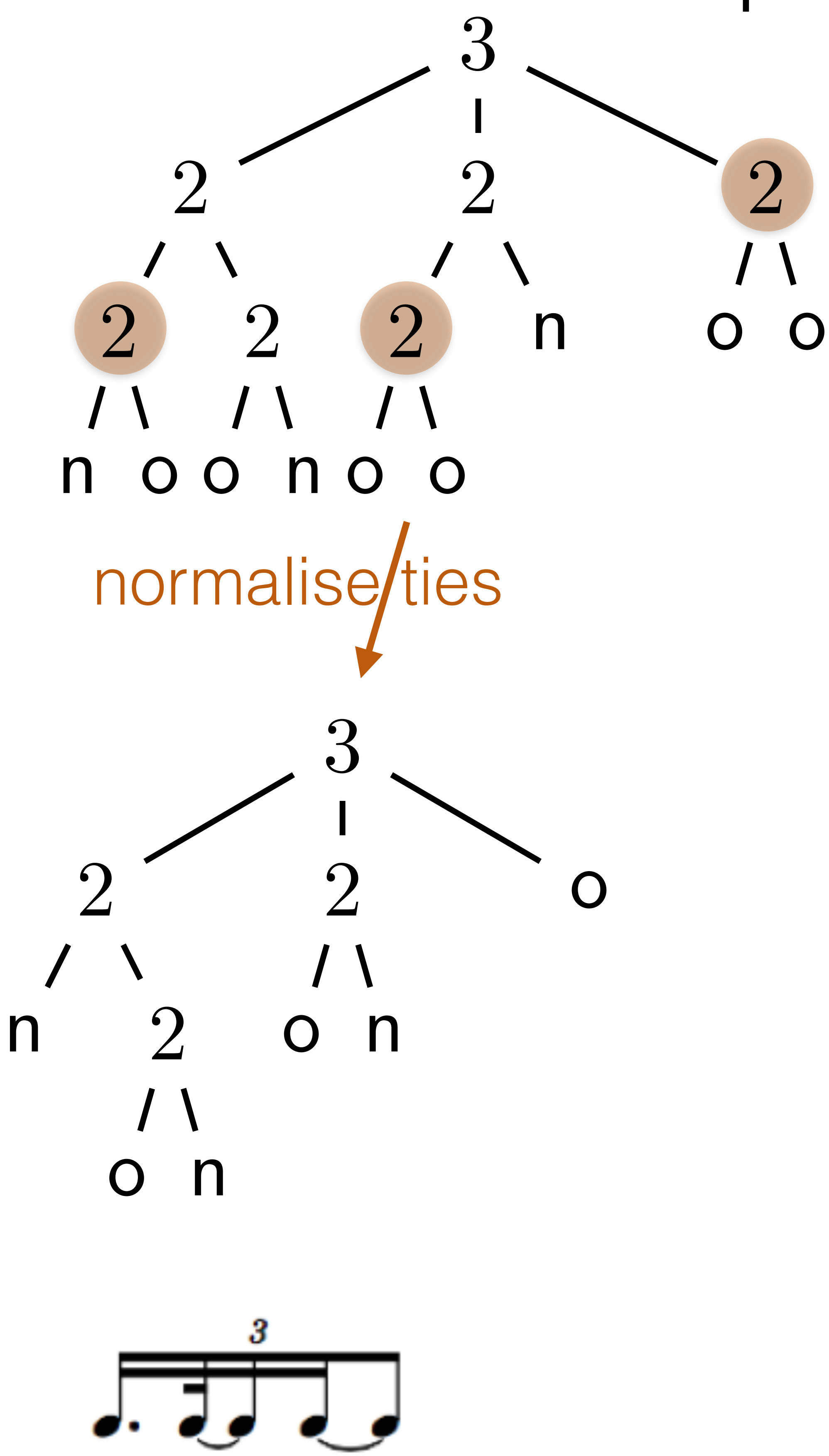
# peak example



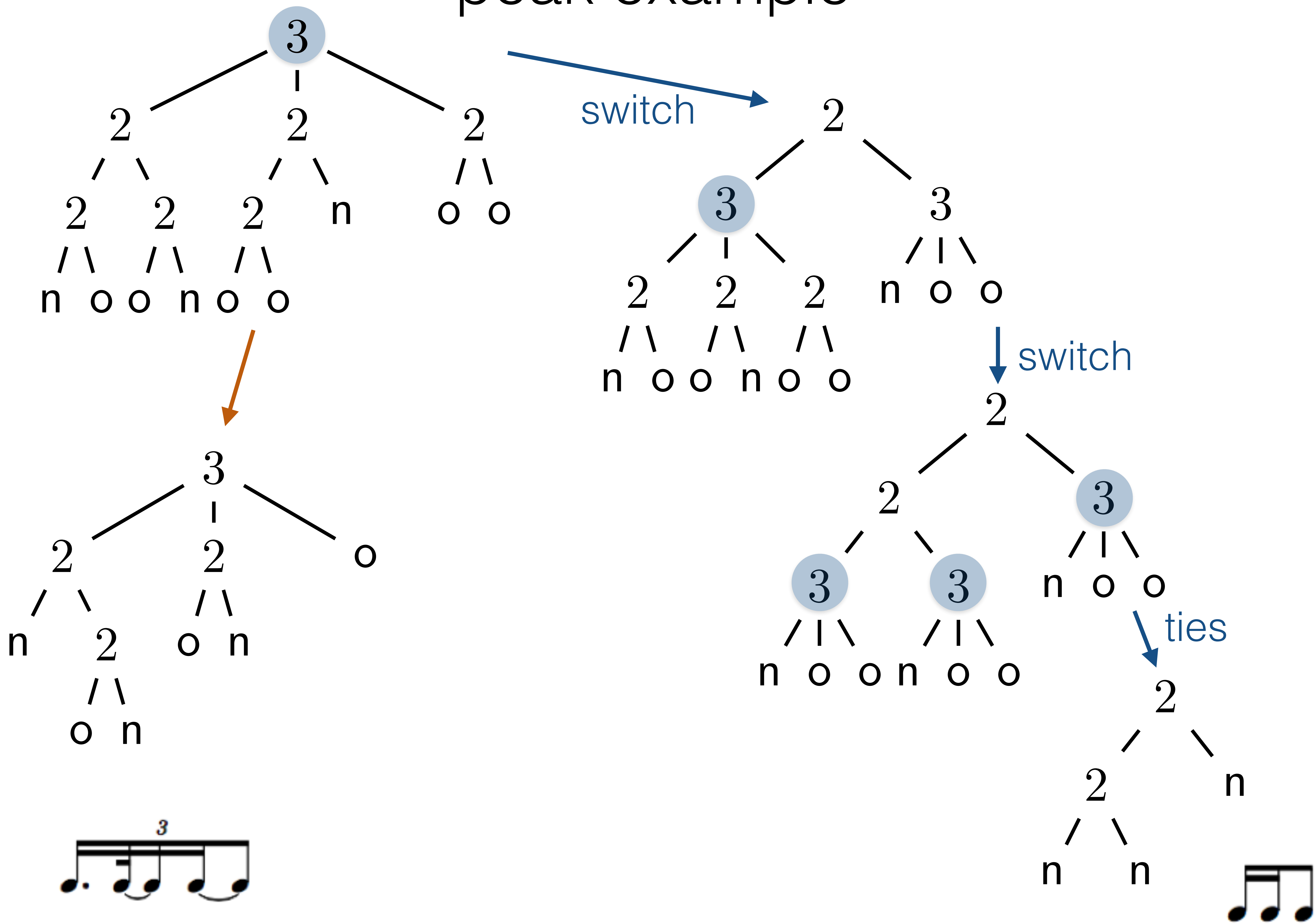
# peak example



# peak example



# peak example



# representation and enumeration of equivalence classes

given:

1. a finite description of a set *L* of allowed RT as CF grammar (RT *schema*)
2. a RT *t*

return:

- a finite description of the set *L'* of RT in *L* of same rhythmic value as *t*

variant:

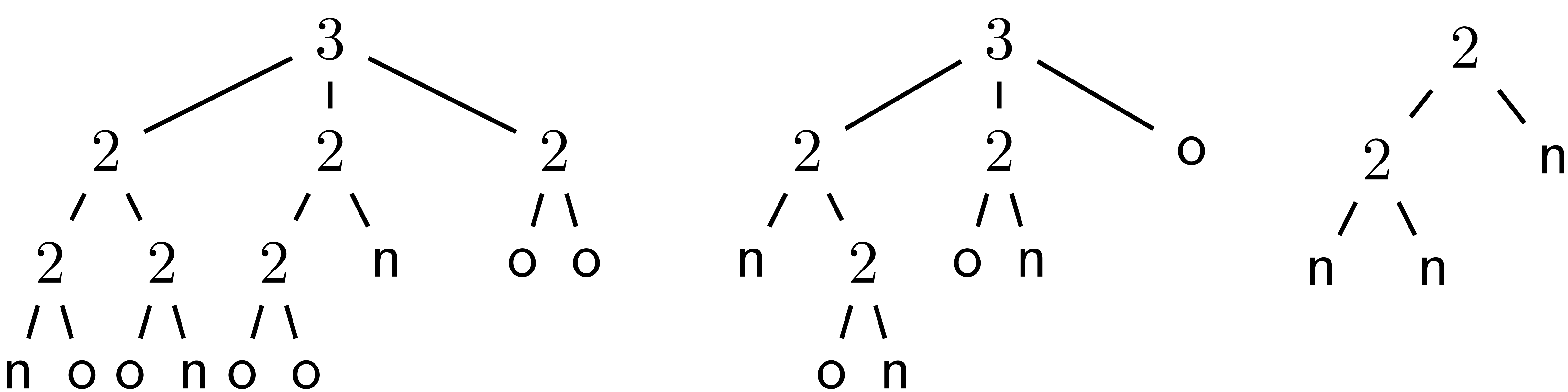
- *L* associates to every RT a weight in an ordered semiring
- lazy enumeration of *L'* according to weight

# RT schemas

acyclic CF grammars defining allowed divisions
defined RTs = derivation trees without n.t.
= tree automata language

<u>ex</u>: division by 2 or 3, then 2 then 2

$$
\begin{array}{llllllllllll}
q_0 & ::= & q_1 q_1 & & q_1 & ::= & q_2 q_2 & & q_2 & ::= & q_3 q_3 & \\
q_0 & ::= & q_1 q_1 q_1 & & q_1 & ::= & \mathsf{n}|\mathsf{r}|\mathsf{o} & & q_2 & ::= & \mathsf{n}|\mathsf{r}|\mathsf{o} & q_3 \ ::= \ \mathsf{n}|\mathsf{r}|\mathsf{o}
\end{array}
$$



# construction of schema for equivalence class

1. schema

$$q_0 ::= q_1 q_1 \qquad q_1 ::= q_2 q_2 \qquad q_2 ::= q_3 q_3$$
$$q_0 ::= q_1 q_1 q_1 \qquad q_1 ::= \mathsf{n}|\mathsf{r}|\mathsf{o} \qquad q_2 ::= \mathsf{n}|\mathsf{r}|\mathsf{o} \qquad q_3 ::= \mathsf{n}|\mathsf{r}|\mathsf{o}$$

2. initial RT of rhythm value $\frac{1}{4}\,\frac{1}{4}\,\frac{1}{2}$

- target schema for equivalence class

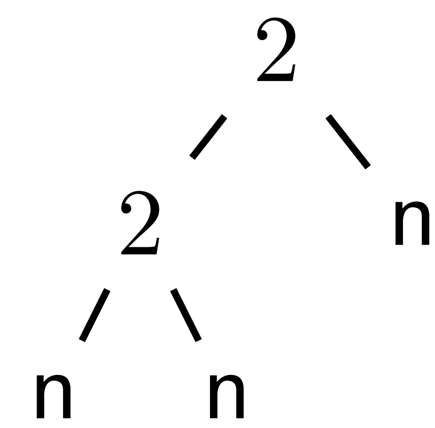
$$[\tfrac{1}{4}\,\tfrac{1}{4}\,\tfrac{1}{2}]^{q_0} := [\tfrac{1}{4}\,\tfrac{1}{4}]^{q_1}[\tfrac{1}{2}]^{q_1}$$
$$[\tfrac{1}{4}\,\tfrac{1}{4}]^{q_1} := [\tfrac{1}{4}]^{q_2}[\tfrac{1}{4}]^{q_2}$$
$$[\tfrac{1}{4}\,\tfrac{1}{4}\,\tfrac{1}{2}]^{q_0} := [\tfrac{1}{4}\,\tfrac{1}{12}]^{q_1}[\tfrac{\mathring{1}}{6},\tfrac{1}{6}]^{q_1}[\tfrac{\mathring{1}}{3}]^{q_1}$$
$$[\tfrac{1}{4}\,\tfrac{1}{12}]^{q_1} := [\tfrac{1}{6}]^{q_2}[\tfrac{\mathring{1}}{12}\,\tfrac{1}{12}]^{q_2}$$
$$[\tfrac{\mathring{1}}{12}\,\tfrac{1}{12}]^{q_2} ::= [\tfrac{\mathring{1}}{12}]^{q_3}[\tfrac{1}{12}]^{q_3}$$
$$[\tfrac{\mathring{1}}{6},\tfrac{1}{6}]^{q_1} ::= [\tfrac{\mathring{1}}{6}]^{q_2}[\tfrac{1}{6}]^{q_2}$$

$$[\tfrac{1}{2}]^{q_1} ::= \mathsf{n}|\mathsf{r}$$
$$[\tfrac{1}{4}]^{q_2} ::= \mathsf{n}|\mathsf{r}$$
$$[\tfrac{\mathring{1}}{3}]^{q_1} ::= \mathsf{o}$$
$$[\tfrac{1}{6}]^{q_2} ::= \mathsf{n}|\mathsf{r}$$
$$[\tfrac{\mathring{1}}{12}]^{q_3} ::= \mathsf{o}$$
$$[\tfrac{1}{12}]^{q_3} ::= \mathsf{n}|\mathsf{r}$$
$$[\tfrac{\mathring{1}}{6}]^{q_2} ::= \mathsf{o}$$
$$[\tfrac{1}{6}]^{q_2} ::= \mathsf{n}|\mathsf{r}$$

# schema for equivalence class and derivation trees

$$[\tfrac{1}{4}\,\tfrac{1}{4}\,\tfrac{1}{2}]^{q_0} \quad := \quad [\tfrac{1}{4}\,\tfrac{1}{4}]^{q_1}\,[\tfrac{1}{2}]^{q_1}$$

$$[\tfrac{1}{4}\,\tfrac{1}{4}]^{q_1} \quad := \quad [\tfrac{1}{4}]^{q_2}\,[\tfrac{1}{4}]^{q_2}$$

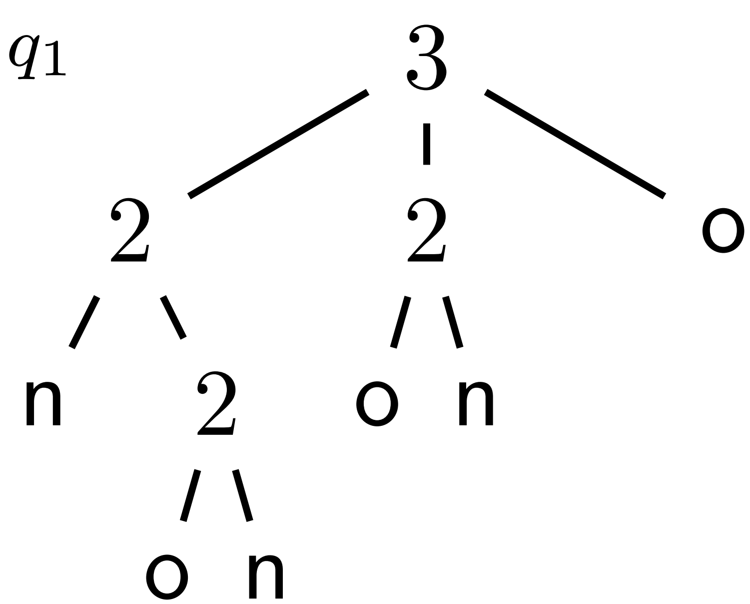


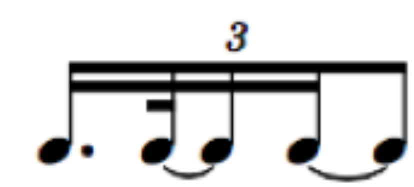

$$[\tfrac{1}{4}\,\tfrac{1}{4}\,\tfrac{1}{2}]^{q_0} \quad := \quad [\tfrac{1}{4}\,\tfrac{1}{12}]^{q_1}\,[\tfrac{\mathring{1}}{6},\tfrac{1}{6}]^{q_1}\,[\tfrac{\mathring{1}}{3}]^{q_1}$$

$$[\tfrac{1}{4}\,\tfrac{1}{12}]^{q_1} \quad := \quad [\tfrac{1}{6}]^{q_2}\,[\tfrac{\mathring{1}}{12}\,\tfrac{1}{12}]^{q_2}$$

$$[\tfrac{\mathring{1}}{12}\,\tfrac{1}{12}]^{q_2} \quad ::= \quad [\tfrac{\mathring{1}}{12}]^{q_3}\,[\tfrac{1}{12}]^{q_3}$$

$$[\tfrac{\mathring{1}}{6},\tfrac{1}{6}]^{q_1} \quad ::= \quad [\tfrac{\mathring{1}}{6}]^{q_2}\,[\tfrac{1}{6}]^{q_2}$$



# weights and lazy enumeration

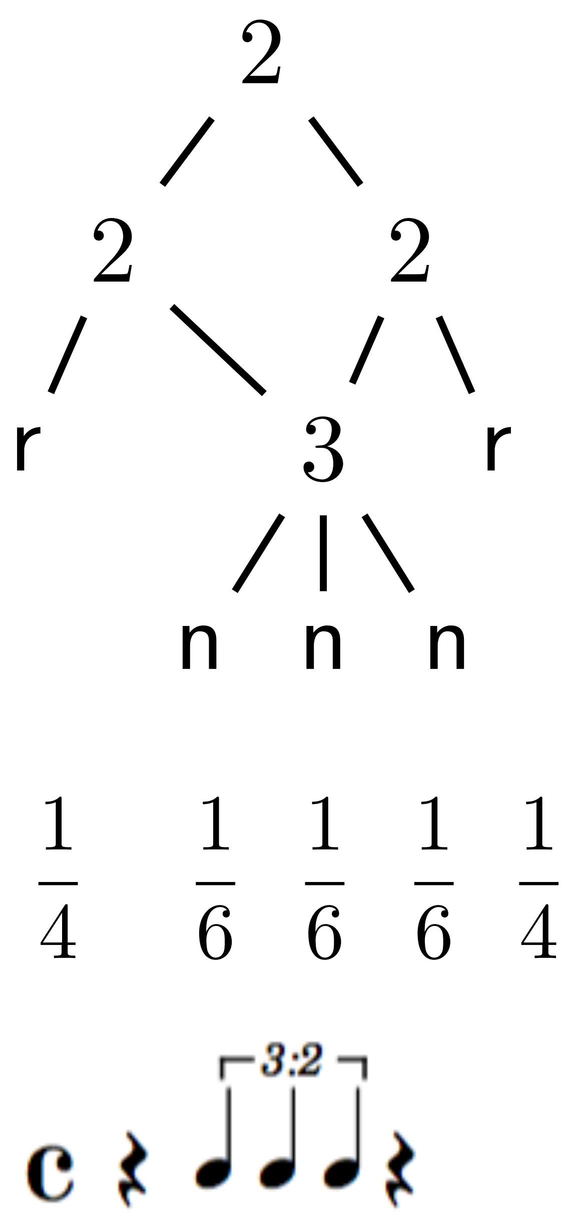
- add weights to CFG production rules
  → defines a notion of complexity of RT (size, penalty for tuples…)
- size of derivation tree = product of weights or rules
- size of RT = sum of sizes of weights of matching derivation trees

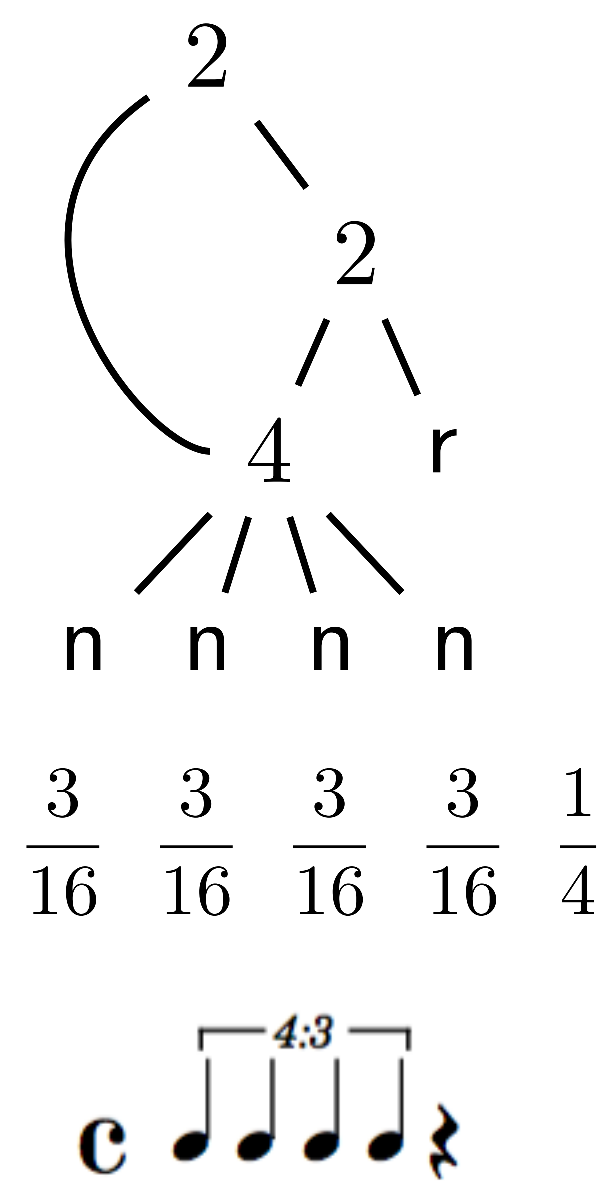
- lazy enumeration of *k* best derivation trees
  *k-best parsing* (dynamic programming) [Huang, Chiang 05]
  table based on the target schema (1 row for each NT).

  O(|target schema| + cmax . k . log(k))
  cmax: max number of production rules for one NT

# rhythm dags (RD)

- no symbol **o**
- sum of durations represented by node sharing (*the data is in the structure*)
- captures *ratio* notation (*p* in the time of *q*)



# RD schemas

acyclic CS grammars
define lang. of dag automata of [Kamimura, Slutzki 1981]

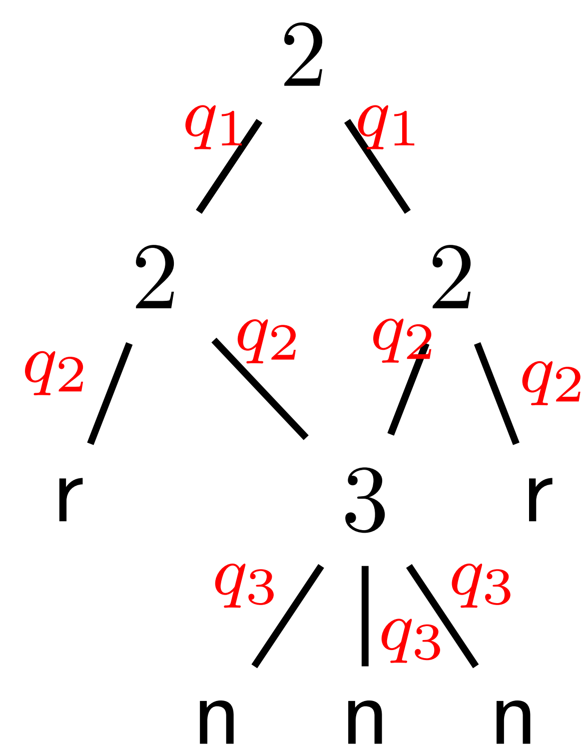
$$\varepsilon \quad ::= \quad q_1 q_1$$

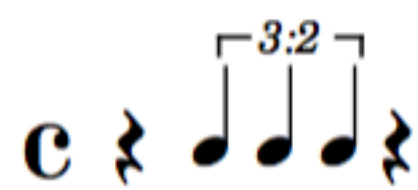
$$q_1 \quad ::= \quad q_2 q_2$$

$$q_2 q_2 \quad ::= \quad q_3 q_3 q_3 \quad \text{(3:2)}$$

incoming edges (ordered)

outgoing edges

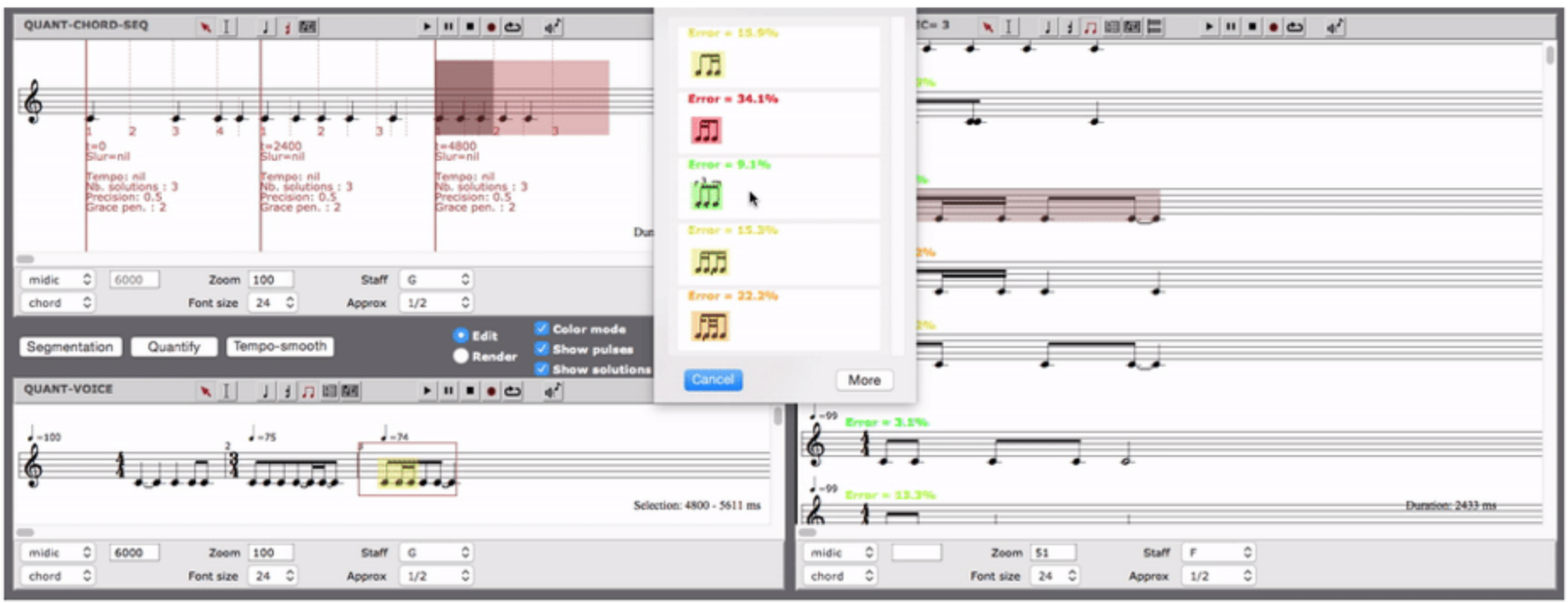


# OpenMusic Rhythm Trees

OpenMusic: graphical programming environment for algorithmic composition developed at Ircam

OM RT (nested lists) are a first class data structure for the representation of rhythms in OM




a library for rhythm transcription [Ycart et al ICMC'16]

# Rhythm Trees Applications

- algorithmic composition, music score editors
  transcription, assistance, transformations…
- computational musicology
  analysis of score corpora, data mining
- digital music score databases
  information retrieval, indexing, *query by tapping*
- metrical phonology in speech production

# automata-based representations of rewrite closure (*regularity preservation*)

given:

- a tree automaton (TA) *A* recognizing a set of terms *L*
- a TRS *R*

return:

- a tree automaton *A'* recognizing the forward closure of *L* by *R*

Used as theoretical tool in some proofs of decidability of confluence

Used as practical tool for enumeration of reachable terms
- e.g. counter examples, error configurations in verification…

Thank You