BEAM IT UP! — A CLASSIFICATION GRID FOR HISTORIC AND CONTEMPORARY PRACTICES OF BEAMING BY MATHEMATICAL RE-MODELLING

Markus Lepper
semantics gGmbH
Berlin, DE
post@markuslepper.eu

Baltasar Trancón y Widemann
Nordakademie
Elmshorn, DE
baltasar@trancon.de

ABSTRACT

In conventional Common Western Notation (CWN) there are different notations styles for flags and beams, evolved historically. We present a classification based on an algorithm which sets beams according to positions in a musical metric space. This algorithm contributes to more clarity also for the human discussion of the historic phenomena thanks to its stratified architecture: (a) assignment of canonical beaming to nodes of the metric tree, then (b) data transformations coming from pauses, dotted notations, etc., (c) breaking of beams according to further parameters like motifs, playing techniques, etc., and finally (d) transformations according to the needs of graphic appearance. For phases (a) and (b) an exact algorithm is presented; for (c) and (d) a semi-formal classification grid.

1. INTRODUCTION

In Common Western Notation (CWN, starting in the mid-dle of the seventeenth century) a beam is a means to no-tate the duration of a note event. It has basically the same meaning as the (much older) flag: The maximum number of beams from each side of a stem is taken for its flag count.

Early examples of beams in printing can be found in *Hans Neusidler: Ein Newgeordent Künstlich Lautenbuch* (Nuremberg, 1536), urn:nbn:de:bvb:12-bsb00041542-7. The duration symbols were borrowed from the white men-sural notation. Here prevails the “fusa”, which looks like the modern eighth. And instead of drawing three flags each for four neighboring “thirty-second notes”, it suggests it-self to simply join them into three continuous lines that link all four stems. See Figure 1, from page 99 of the cited work. In its foreword this is called “Leiterlein” = “little ladder”. [1]

The idea of *mathematical re-modelling* is to mimic the syntactic operations and semantic outcomes of symbol sys-tems from historically evolved cultural techniques by a col-lection of mathematical definitions and algorithms. Its main

[1] According to [1, p. 88] beams appear not before 1690 in printed *staff* notation.

Figure 1. Early Example of Beams in Lute Tabulature

aims are (a) to define more precisely the terminology necessary for human discourse, (b) to lay the foundations for a transparent documentation of automated processing, and (c) to provide a collection of parameters which modify the behavior of the model and can thus be used as a grid for more precise classification and comparison of artefacts, methods and automated processing tools. [2] [3] [4]

Mathematical re-modelling is applied in the following to the problem of finding the adequate beaming configuration for given meter and rhythm, after the sequence of basic symbols (note heads, stems, and flag counts) has been found. The described algorithm has recently been added to our metricSplit Java implementation, found at `http://bandm.eu/downloads/DemoMetric.jar`. This also can be used as a library for own programming, see `http://bandm.eu/sig/doc/api`.

2. BEAMING RULES AS A TRANSFORMATION PIPELINE

Complex transformations which have evolved during centuries of historic practice, can best be analyzed and documented by modelling them as a transformation pipeline, a sequence of distinct transformation phases, each with well defined input and output interfaces and well defined inner behavior. In this paper we present a proposal as a basis for discussion, research and implementation. The first three phases have been implemented and thoroughly tested.

Figure 2 shows the chosen architecture: The first phase operates on the metric tree as such, the second gets the rhythmic information, and the third phase incorporates additional data like sung text, hand distribution, tempo, etc. The very last phase deals with the concrete rendering of the graphic appearance according to pitch heights. Every phase gets as input its specific main input data (= left column in the Figure), the output of the preceding phase (center column), and a collection of further parameter settings, to modify its way of operation (right column).

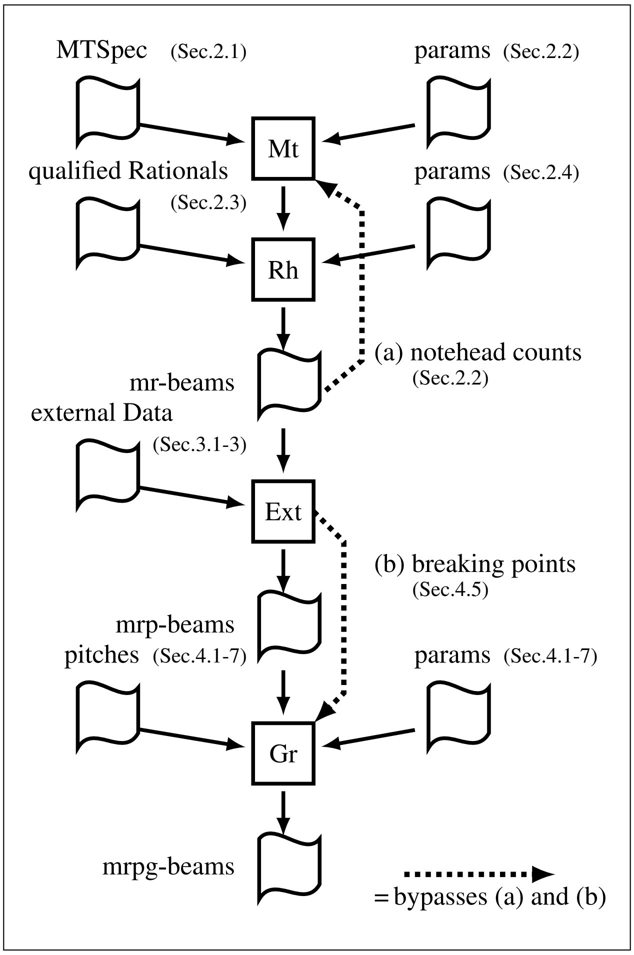


Figure 2. Beam Layout Processing Pipeline: Inputs, Result, Data Flow, and Bypasses

A natural classification grid for different notation styles, epochs, software systems, etc., is given by the selection of the applied transformations, together with their parameter values.

Dividing a complex transformation into distinct phases accomplishes a clean separation of concepts, data, and information flow. But it also helps to identify those critical aspects which can *not* be restricted to only one phase. We have found two such bypasses, shown in Figure 2 by the dotted lines and discussed in sections 2.2 and 4.5.

2.1 Foundation: Genuine Beams

The first phase deals only with the structure of the meter, not with a particular rhythm. A good starting point for any rendering (including beaming) in a metric context is a *metric tree*. Widely varying concepts of metric trees have been defined, esp. for the purpose of music recognition, automated transcription and automated interpretation ([5] [6] [7] [8] [9] [10] [11] [12] [13] [14] [15] [16]—for a survey see [17].) They all have in common:

- A musical meter is represented by a tree of nodes.
- The root of the tree represents the flow of time during a complete measure.
- Each node has an ordered sequence of child nodes; they represent consecutive and adjacent sub-intervals of the time interval represented by their parent node.

Many further attributes, like "metric weight", agogics, harmonic roles and rules, etc., can be attached to each node of such a graph. The following is based on the metric trees from metricSplit [16]. There the only additional requirement is that durations and start and end points are given as rational numbers and the measure starts at timepoint 0. [2] MetricSplit supports arbitrary complex metric tree specifications like $Mts = \mathtt{7/8+8/7}$.

For notation of rhythms in general and for beaming in particular, it is ergonomically crucial that the notation expresses the relation of each single notation event (as a foreground structure) to a particular node of such a metric graph (as its middleground structure) in a direct and easily readable way. As a consequence, not every sequence of neighboring notes can be joined by a beam, but only those which represent tree nodes under the same parent. [18, p. 27 pp.][1, p. 91 pp.] [19, p. 43 pp.,47] [20, p. 80 pp.][21, p. 153 pp.]

This was already understood by Neusidler: the "sixteenths" in measure three in Figure 1 are only partially joined.

So the first and fundamental rules [3] are:

PROP. nota.trabes.trabesUtVexilia*: Add the numbers of beams and beamlets from both sides of the stem separately. The higher of these sums indicate the duration of the note in the same way as the same number of flags would do.*

PROP. nota.trabes.notaeUtNota*: Let there be a single note A, which directly corresponds to a node of a metric tree, and a group of notes B by which this note is replaced. Then the left(/right) side of the leftmost (/rightmost) note in B has the same beam appearance as the left(/right) side of A, resp.*

Both rules seem easy and trivial, and indeed they are widely applied in conventional engraving. Together they determine completely a beaming for each sequence of notes which corresponds to the complete list of child nodes of a particular metric tree node. These are called *genuine beams*, as shown in the top part of Figure 5 for a very regular "$\mathtt{2/4}$" meter. For instance, the third and fourth thirty-second notes have the same beaming structure at their left and right end as the sixteenth note one line above, which they can replace. [4]

But two severe caveats arise: (1) A node in the metric tree may not correspond to one single note symbol, for instance a node with the duration $\frac{5}{16}$. (2) Historical practice often violates this principle, see the next section.

[2] In particular, the mathematically intricate problem whether the nodes represent open or half-open intervals does not require consideration.

[3] The LMN project [4] has collected about eight hundred properties to classify conventional usage of CWN, all identified by a hierarchical nomenclature of Latin terms. The property names in this article are intended to fit in neatly.

[4] All "one-and-a-half-dimensional" renderings in this article have been produced automatically by our implementation. Their graphics is a mere control instrument, with no claim for beauty. Esp. it employs only linear proportional space allocation, without any "psychological adjustment", which is for instance in musixTex executed by a dedicated external program run. [22]

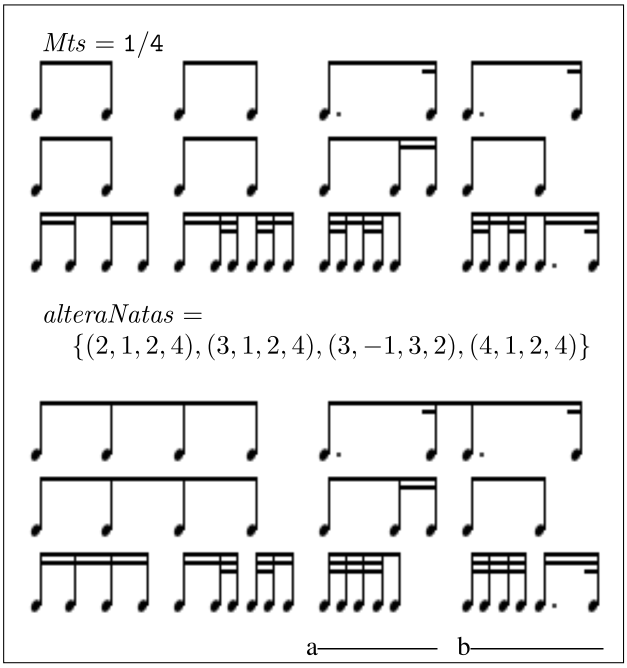


Figure 3. Modified Genuine Beams I

2.2 Modification of Genuine Beams

As mentioned, the historic practice has deviated from these simple rules above:

(A-a) Neusidler did not follow them: In the second measure of Figure 1, the first two "sixteenth" are *not* connected to the "thirty-seconds" by a single "eighth" beam.

(A-b) Nowadays, esp. in the context of traditional 4/4 meters, *all* four sixteenth notes of one quarter are joined by two beams, including the middle ones.

(A-c) Contrarily, when groups of four thirty-second notes follow, these are often also completely joined by three beams, but the single beam connecting the eighth nodes is dropped.

Modifications of these kinds can be modelled by the data type

PROP. nota.trabes.alteraNatas*:* $\mathbb{P}(\mathbb{N} \times \mathbb{Z} \times \mathbb{N} \times \mathbb{N}) \rightarrow$
Each contained tuple (a,b,c,d) *says that on the level* a *of the metric tree, (counted with positive numbers descending from the top node at level* 0*) the value* b *is added to the number of beams if and only if the conditions given by* c *and* d *are met.*

If b *is positive, then* c *gives a maximum number of note symbols which must not be exceeded by both halfs to be connected, and* d *is a maximum for the sum of these numbers. If* b *is negative, than* c *is a minimum which must be reached on either side, and* d *a minimum for both sides.*

These rules are a violation of the pipeline architecture, see the bypass line (a) in Figure 2: Conceptually the genuine beams are determined by the meter only, but the number of actual note heads on both sides of these formulas is not fixed until merging transformations (the "MX" in the next phase, see below) have been applied. When programming a concrete implementation, this twist causes real problems for documentation, testing, and maintenance.

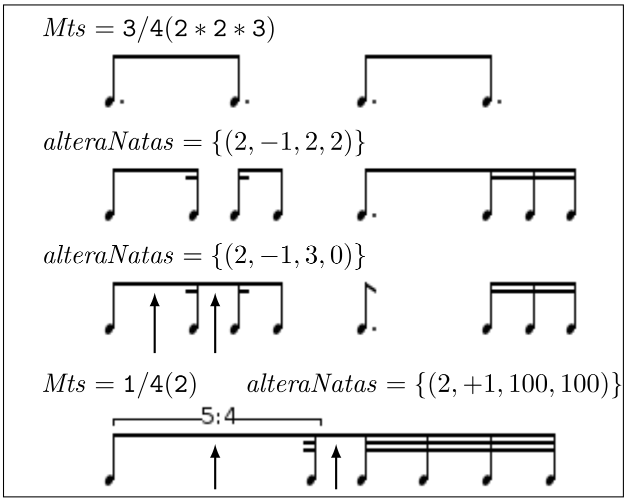


Figure 4. Modified Genuine Beams II

We found that these rather coarse and somehow arbitrary rules are sufficient to model most of the modified genuine beamings found in historic and contemporary engraving practice:

$\{(2,-1,0,0)(3,1,100,100)\}$ produces the notation chosen by Neusidler in Figure 1, by removing the connecting beam between the eighths, when reading it (a-historically!) as modern notation.

The top of Figure 5 shows the genuine beams, and Figure 3 shows modern standard engraving of some rhythms in a $\frac{4}{4}$ meter: Code value $(2,1,2,4)$ adds an additional beam between sibling quarters (= at level 2 of the metric tree), but only in case there are at most two notes on each side, see the second example line.

Similarly, $(3,1,2,4)$ joins sibling eighths by two beams instead of one, of course only if possible w. r. t. the rhythmic values, compare lines three and six. [5]

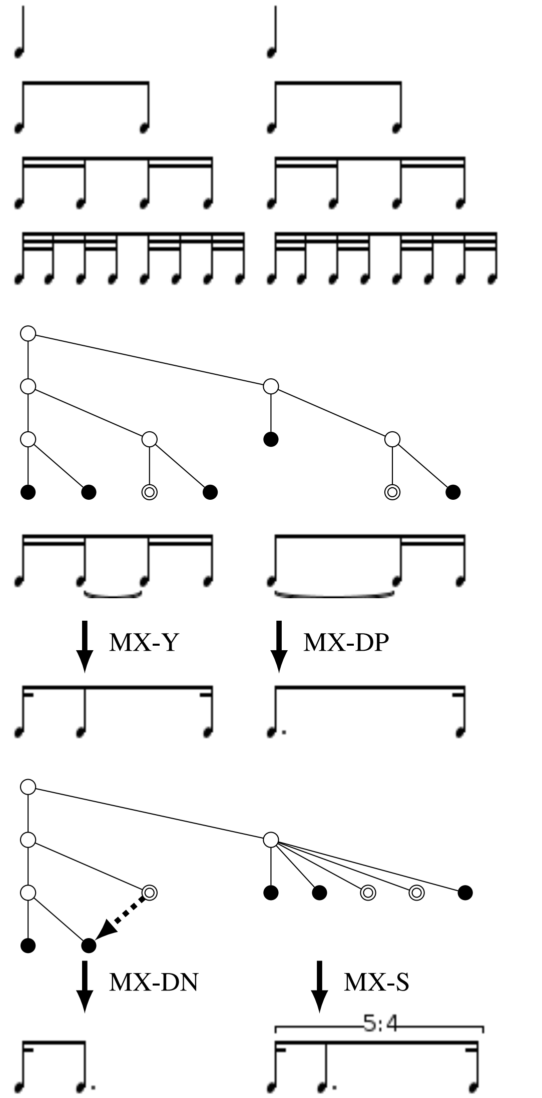
Contrarily, $(3,-1,3,2)$ suppresses the beam between two groups of notes which fill sibling eighths, if one of them is too crowded, for a better separation of these groups for ergonomic reasons. Comparing the groups (a) and (b) at the end of Figure 3 shows that $c \geq 3$ must be fulfilled by only one of their constituting eighths, but $d \geq 2$ by both.

Finally, $(4,1,2,4)$ joins four thirty-second notes by an additional beam connecting the sibling sixteenths, but only if not divided finer.

Also counter-intuitive and confusing modifications can be applied. These are not prevented automatically by the current implementation. For instance, the widely spread textbook [18] shows on page 27 a table of examples for rhythmic notations, which notoriously contains patterns like

This is not wrong but somehow paradox: When it shall be stresses that not "two times two" thirty-seconds belong

[5] In the context of metricSplit [23], [16]. the same printing effect could also be realized by changing the definition to `4*1/16`, which means constructing four sibling nodes on the same level, under one common parent node. But then also the metric structure with all other roles and functions is a different one, not only the graphical rendering.



$\langle(1/16, \mathsf{true}), (3/16, \mathsf{true}), (1/20, \mathsf{true})$
$(3/20, \mathsf{true}), (1/20, \mathsf{true})\rangle$

Figure 5. Genuine Beams and Merging Transformations

together, but all four of these on equal terms, which thus cover the complete duration of an eighth, it is confusing to obfuscate these eighths by sixteenths' beams.

(B)

It is a common phenomenon in the history of notation, that a well-proven device is ab-used or re-used in a new context. So the prolongation dot has been advanced from part of the fixed topos "3:1" to a general sign for Mersenne numbers $2^n - 1$ as duration factors.

As a consequence, one and the same graphical beam level can become necessary on two different but neighboring levels of the metric tree, see the arrows in Figure 4. The first line shows an ergonomically sensible layout, but with finer divisions, cancellations of the higher beam become sensible, as also shown.

The same can happen with local divisions by non-Mersenne numbers, see the last line in the figure.

2.3 Beams for Rhythms

A rhythm to be rendered is given as a sequence of pairs of rational numbers (representing time points relative to the measure start, or durations to be added up) and one Boolean value each, qualifying the event as sound or pause, see the last lines in Figure 5. An *initial coverage* is the minimal front in the metric tree which has a node at every start point occurring in the rhythm, see again Figure 5. The initial coverage can always be rendered immediately, using the genuine beaming from the top of the Figure: When a sounding event is represented by more than one note symbol, these must be connected by a *tie symbol*.

Only the first nodes assigned to any event show up in the rendering necessarily. All others can possibly be *merged* with their predecessors. The rules when to apply a particular merging transformation (MX), and the different style parameters to re-model the different historical practices can become very complicated, especially when a *total* rendering function is to be described, which supports arbitrary meters and rhythms—for details see [16].

By each MX, a contiguous sequence of note symbols is cancelled from the notation and the immediate predecessor is prolongated. This may affect beaming. On the one hand, positive dottings (MX-DP) increase the duration, but they never reach the factor 2; therefore they never affect beaming. On the other hand, with negative dottings (MX-DN) the last of the merged note symbols appears at the rhythmic position of the first one and the beam structure must be transferred, see the dotted arrow in the figure.

Syncopes (MX-Y) and hemiolas (MX-H) print the parent node's note symbol at a child node's position. Thus the left(/right) side of this note must copy the left(/right) side of the left(/right) parent involved, resp.

The merging of equidistant siblings (MX-S) enlarges the duration of the first (=the only printed) note and thus reduces the beaming according to the resulting multiplication factor f. (More precisely: by the highest n such that $2^n \leq f$.)

Thus after all these transformations a subsequent *cutting down* step is needed: All beams which are foreseen as genuine but may only touch one of their two stems, because the other's note value has been prolongated by a merging transformation, are cut down to *beamlets*. [6]

2.4 Local Transformations of Beam Patterns

Concerning pauses, there are three alternative ways how to apply the resulting beams and beamlets to the note symbols [18, p. 49][19, p. 46][24, p. 15 p.][20, p. 88, 213]

PROP. nota.trabes.sopraPausam.perCaudulam: *Each pause is treated like a sounding note and gets a* stemlet.
. . . transiens: *Pauses are not connected to beams, but a beam may span over a pause.*
. . . separans: *A pause is never spanned by a beam, but cuts all beams pointing to it down to beamlets.*

The format *perCaudulam* is the most recent developed format, and also the the most canonical: Each pause gets a

[6] Beamlets can also occur as genuine beams, but only in advanced use cases, see the last example in section 2.2.

"stemlet" and is treated like a sounding event, see line ① in Figure 6. The representation is somehow redundant because the duration of the pause is encoded twice: in the beam (but without prolongation dots) and in the pause symbol. Nevertheless, from the systematic perspective this is the canonical form.

For the variant *transiens* these stemlets are deleted, all beams between a stemlet and a stem are cut down to beamlets, and all beams between two stemlets disappear completely, see line ②.

Line ③ shows the form *separans*: All beams which touch pauses are cut down to beamlets. This variant is dominant in conventional sheet music. But it gives up one particular role of beaming which goes beyond the mere replacement of flags:

PROP. nota.trabes.significantVocem: *Connected beams implicitly have the role of voice-leading indicators (German "Stimmweiser").*

There are rare but relevant critical cases:

Here (BWV 543, ms. 16) the first version (Neue Bach Ausgabe = NBA) of engraving forbids to read a crossing of the voices allowed by the second (Breitkopf Sämtliche Orgelwerke Urtext), which would but be the correct resolution of the d", the seventh of the dominant. [7] The collection of all beams and beamlets which are connected by at least one contiguous top-level beam are called a *beam aggregate* in the following.

All forms ① to ③ are a basis to which further *local transformations* are applied. These aim at eliminating the following properties:

PROP. nota.trabes.trabulaeContraIdem: *All beamlets appear on the same height on both sides of a particular stem.*

This pattern is caused by an isolated inner node from a group of more than two equidistant siblings, see the triplets in Figure 6 line ③. The beamlets remaining from the above-mentioned down-cuts appear on both sides of the stem with equally good reasons. It may seem desirable for an engraver (for whatever reason) to eliminate this property. This can be done by replacing them by beamlets only to one side. This transformation reduces the "sensible ergonomic information", because the fact that the note is connected *in the same way* to its left and to its right neighbor is no longer expressed.

Line ⑦ in Figure 6 shows an example and Table 1 shows a mathematical model. Each stem end is represented by five natural numbers: number of flags, number of left beams and beamlets, number of right beams and beamlets. The properties and the eliminating transformations are specified on these 5-tuples, the former as pre-conditions for the latter. [8]

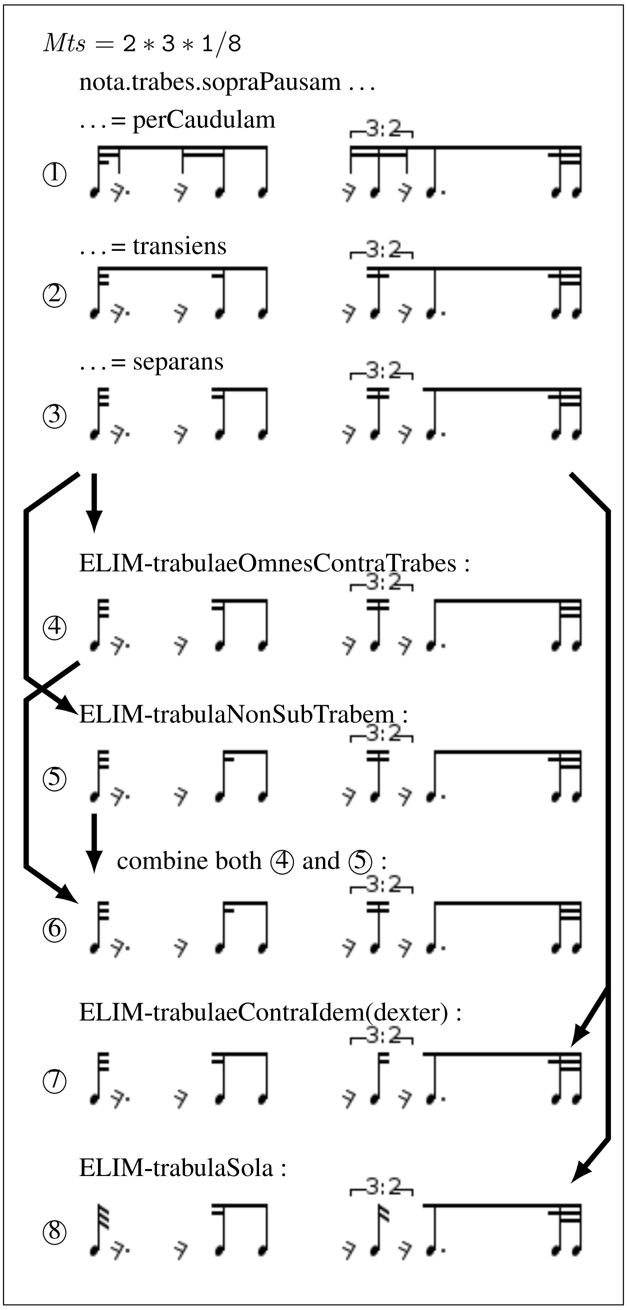
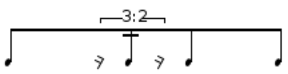
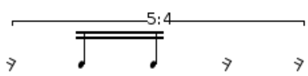


Figure 6. Local Transformations

The formula for *nota.trabes.trabulaeContraIdem* only requires the number of left and right long beams to be equal ($= a$). So it also holds for beamlets on both sides under (the same number of) beams on both sides, a case not depicted in Figure 6: 

A similar property can be defined to match cases like 

where a very similar distribution of beamlets is found at both sides not of a single

[7] Thus the voice identification algorithm in [25] in a preparatory step (p. 307) replaces beams (and slurs) by explicit voice-leading signs

[8] The prefix "ELIM-" works as on mere technical meta-level, not as part of the nomenclature. Then of course "elim-trabul*as*-contra-idem would be the correct case.

note but of a group of siblings. This case is *not supported* by the current model. It is left to future extensions of our work, because its eliminating transformation would not be a mere *local* one.

PROP. nota.trabes.trabulaeOmnesContraTrabes*: All beamlets one one side of the stem appear on the same height as a (long) beam on the other side.*

Due to *trabesUtVexilia*, each such beamlet is redundant for calculating the duration. It nevertheless indicates the relation of this stem's note and node to its neighbor in the indicated direction. In conventional engraving, this beamlet is removed—compare the last two beamlets in line 3 of Figure 6 to line 4.

While the eliminating transformations above reduce the information content, they do not introduce confusing contradictions to the genuine beams. This changes when trying to eliminate

PROP. nota.trabes.trabulaNonSubTrabem*: On one side of the stem one or more beamlets appear, but no single beam. On the other side there is at least one beam.*

Then the beamlets traditionally simply switch the side, see the second sounding note in Figure 6, the transformation's definition in Table 1 and the result in Line 5. Here the direction into which the sixteenth-beamlets point is *positively wrong* and *contradicts* the original genuine beams. Nevertheless this is a standard transformation in traditional engraving. This also eliminates

PROP. nota.trabes.trabulaSola*: There is a stem which carries beamlets but no beams.*

This is eliminated by the final transformation in Table 1 and Figure 6, namely by replacing all beamlets by the corresponding number of flags.

Conventional CWN engraving uses all these four transformations, but ELIM-trabulaeContraIdem is often superseded by ELIM-trabulaSola, compare lines 7 and 8 of Figure 6.

Not all possible cases are covered by these transformations, for instance $(0 \mid 0, a \mid 0, b) \wedge a \neq b \wedge a \neq 0 \wedge b \neq 0$. But in conventional usage of CWN it is transformed by ELIM-trabulaSola to $(max(a, b) \mid 0, 0 \mid 0, 0)$ anyhow.

3. ADDITIONAL EXTERNAL DATA

The beam aggregates constructed so far are determined solely by meter and rhythm. They are called *mr-beams* in the following. In each particular notation, they will be employed in contexts where *further parameters* influence their concrete appearance.

3.1 Indirect Influence by Stem Direction

First of all, the *stem direction* is of course relevant. Assume that every sequence of notes is part of a particular *notational voice*. Then there are basically three variants for determining the stem direction:

PROP. nota.cauda.significat. . .
. . . vocem*: the stem direction is needed to identify the voice among other voices in the same staff.*
. . . instrumentum/accentum/manum/modumAgitur/. . .*: the stem direction is employed to represent the named (binary valued) parameter.*
. . . nihil*: the stem direction carries no meaning at all, but is free to change according to graphic/ergonomic requirements.*

In the first two cases the stem direction is fixed and must be respected by all further transformations; it is a further input parameter to any beam layout algorithm. Normally it will change in the first case less frequently than in the second. Only in the last case it is free and thus is an output of the subsequent transformation processes.

For beam aggregates there are further sub-cases:

PROP. nota.cauda.significat.vocem. . .
. . . trabsSeparataCaudaMutata*: beams are broken when the stem direction changes.*
. . . trabsSeparataLineaMutata*: beams are broken when the voice moves into another (mostly: a neighboring) staff.*

In most conventional engravings of classical piano music the first property holds (to thin out the optical appearance for readability), the second does not (to make the transition of the voice even clearer, according to *trabes.significantVocem*).

Whenever such a breaking takes place, the situation is as if a pause had occurred in the mode *nota.trabes.sopraPausam.separans*: Subsequent application of *ELIM-trabulaNonSubTrabem* and *ELIM-trabulaSola* may or may not be applied.

A minimal case is a one-line percussion staff, with one notated pitch only and two stem directions for a parameter. Then a frequently found transformation is

PROP. nota.voces.unaUtDue_pausasPerdatas*: The voice is split into two voices, separating up and down stems, and both are beamed independently, except that the events in one voice stand in for the pause symbols in the other.*

This can also be applied in cases which are only slightly more complicated. Gould [21, p. 312] gives an example from piano notation, where the parameter encoded by stem direction is the hand selection = *cauda.significat.manum*:

With respect to the (invisible) pauses either *nota.trabes.-sopraPausam.transiens* or *. . . separans* can be applied to both resulting voices. The example shows the former, followed by *ELIM-trabulaeOmnesContraTrabes*.

$$(\mathit{flags} \mid \mathit{leftLong}, \mathit{leftShort} \mid \mathit{rightLong}, \mathit{rightShort}) : \mathbb{N} \times \mathbb{N} \times \mathbb{N} \times \mathbb{N} \times \mathbb{N}$$

$$\frac{b > 0}{\begin{array}{c}\mathsf{ELIM}\text{-}\mathit{trabulaeContraIdem}((0 \mid a, b \mid a, b), \mathsf{dexter}) = (0 \mid a, 0 \mid a, b) \\ \mathsf{ELIM}\text{-}\mathit{trabulaeContraIdem}((0 \mid a, b \mid a, b), \mathsf{sinister}) = (0 \mid a, b \mid a, 0)\end{array}}$$

$$\frac{b > 0 \qquad a + b \leq c}{\begin{array}{c}\mathsf{ELIM}\text{-}\mathit{trabulaeOmnesContraTrabes}(0 \mid a, b \mid c, d) = (0 \mid a, 0 \mid c, d) \\ \mathsf{ELIM}\text{-}\mathit{trabulaeOmnesContraTrabes}(0 \mid c, d \mid a, b) = (0 \mid c, d \mid a, 0)\end{array}}$$

$$\frac{b > 0 \qquad c > 0 \qquad x = \mathsf{MAX}(d, b - c)}{\begin{array}{c}\mathsf{ELIM}\text{-}\mathit{trabulaNonSubTrabem}(0 \mid 0, b \mid c, d) = (0 \mid 0, 0 \mid c, x) \\ \mathsf{ELIM}\text{-}\mathit{trabulaNonSubTrabem}(0 \mid c, d \mid 0, b) = (0 \mid c, x \mid 0, 0)\end{array}}$$

$$\frac{b + d > 0}{\mathsf{ELIM}\text{-}\mathit{trabulaSola}(0 \mid 0, b \mid 0, d) = (\mathsf{MAX}(b, d) \mid 0, 0 \mid 0, 0)}$$

Table 1. Historically Defined Local Transformations

3.2 Direct Influence

The preceding section covered the case that an additional parameter breaks an mr-beam indirectly by determining the stem direction. But such a parameter can also affect the beamings directly. The result is called an *mrp-beaming*. Basically there are two cases:

PROP. nota.trabes.extera…
…separans: *a parameter value causes the breaking of a beam from the mr-beaming.*
…ligans: *a parameter value joins two groups of beams which are foreseen as separate by mr-beaming.*

The first case is much more frequent.

With *sung lyrics* one of the following properties may apply, which all are of type *extera.separans*:

PROP. nota.trabes.cumVerborum…
…syllabis: *beams may not extend further than the sung syllable.*
…nominibus: *beams may not extend further than the sung word.*
…lineis: *beams may not extend further than the sung text line.*

The first property had been standard in all engravings of classical and romantic music. The last property can sometimes be found in contemporary popular sheet music. For the second we have not found any evidence, but it logically closes a gap.

In contemporary advanced music neither is applied, but vocals are beamed as “any other” instrument. [1, p. 8] This can be called *NON-trabes.cumVerbis*.

PROP. nota.trabes.separatae.cumMelo/cumLigato: *Especially in piano music, but also in orchestral monodies, a sequence of adjacent motifs with the same rhythmic structure is clarified by breaking the beams between every on-beat and the following up-beat part.*

See chorale prelude op. 122, Nr. 3, ms. 7 f. by Johannes Brahms, where this technique is applied in the first measure (redundantly doubling the slurs), but not in the second:

Please note that this operation gives up *trabes.significantVocem*: The voice leading may become less clear.

Confusingly, two modifications with *contrary* meaning have the same optical result:

PROP. nota.trabes.ligataeContraNates: *Notes are connected contrarily to the grouping by the genuine beams.*

PROP. nota.metraMulta.perTrabem: *A polymetric situation is clarified by beam patterns in a particular subset of voices which are shifted against the beams in the other voices and/or the sequence of measure bars.*

These two are totally different techniques: The former only affects the way of writing (= the sphere of syntax) but still means the original pattern of stress, unchanged relations in agogics and motif, etc. (= the same semantics). But the second means that the meter shall indeed be shifted with all consequences in interpretation (= in the semantic sphere).

Often *ligataeContraNates* is applied to up-beats and their targets. For example, LilyPond [26], which claims to implement standard engraving conventions, renders the source text “`r8. d16 e r8.  r2`” as[9]

The appearance of this pattern in Chopin’s sonata in b minor,[10] Largo, ms. 5–19, left hand, is a typical example for its combination with

PROP. nota.trabes.cumPositioneManus: *Change and identity of the hand position are expressed by breaks and continuity of the beams.*

[9] So done by LilyPond version 2.20.0.

[10] Engraving “Collection Litolff No 1087”, `IMSLP638961--PMLP2364-ChopinSonataBMinor-KohlerEdition.pdf`.

This is one of the rare cases of the above-mentioned *nota.-trabes.extera.ligans*, where *additional* beams are caused by external data.

Both techniques can and often do lead to beams crossing bar lines and even line breaks, called *nota.metrum.secat.trabem* and *nota.linea.fracta.secat.trabem* by [4, p. 223, 229].

Examples of polymetrics in Romantic piano literature are often of a most simple type, namely a shift by a certain distance d with otherwise unchanged structure (= same duration, metric tree and tempo, labelled "$(d, =, =, =)$" by [4, p. 243 pp.]). But also constellation $(0, \neq, \neq, =)$ is possible and can easily expressed by beaming only, see again Brahms, Capriccio op. 67, Nr. 5 , m. 111 ff:

For more recent examples see [24, p. 116 pp.][21, p. 171, 175 pp.][20, p. 170 pp.]

3.3 Beams expressing Tempo – "Feathered" Beams

A more advanced concepts are "feather" or "fan" beams (German "Fächerbalken"). [19, p. 47] [20, p. 94] [24, p. 124, 141] [21, p. 158]

Primary, secondary, ternary beams etc. do not run in parallel but converge like a fan to a common point. The intuition is a ritardando when at the left end there are three beams (which would mean a thirty-second note) which collapse into only one beam at the right end (which stands for the much slower eighth note). Vice versa, starting at one point and running apart means an accelerando.

While in literature this principle is only defined for a "free" interpretation and nearly always written with three beams, indeed it is totally independent from the notated and played pattern and can (in sensible limits, for readability) be combined with any rhythm:

PROP. nota.trabes.accelerans*:* $\mathsf{Seq}(\mathbb{Q} \times \mathbb{N} \times \mathbb{Q})$.
= list of tempo changes which shall be expressed graphically by vertical beam dimensions.

Let this sequence be sorted by the first components ascending. Then each contained triple (a, b, c) says that at time point a the tempo b BPM shall rule, that this is expressed by the factor c applied to the widths and distances of all beams at this point, and that between these triples linear interpolation shall be applied.

So with *trabes.accelerans* $= \langle(0, 30, 1/1)(1/4, 90, 3/2), (1/4, 60, 1/1), (1/2, 60, 1/1), (1/2, 30, 1/2), (1/1, 120, 3/2)\rangle$ we get the rendering

which possibly is not too intuitive, but the canonical continuation of the principle of feathered beams.

4. TWO-DIMENSIONAL LAYOUT: VERTICAL POSITION AND PITCH HEIGHT

Up to here, all considerations have been related to a "one-and-a-half-dimensional" space: The x-axis is mapped to the flow of time, but the y-axis is made up by only the beam selections and the stem direction.

The most frequent context for beams is to be attached to noteheads which represent *pitches* by their vertical position relative to the staff lines. So now graphical criteria come into play, to produce a true two-dimensional arrangement of the beams. Here a beam aggregate from the mrp-beaming can even be broken again.

Any *layout algorithm*, whether applied manually or automatically, must always answer *four distinct questions*. These are very different in nature, but their solutions are tightly mutually dependent. [27, p. 153] How they are priorized or whether they are solved separately or in an intermangled way may differ.

The questions are:

PROP. nota.trabes.inclinatioSignificans*: How does the steepness of the top-most beam symbol indicate a tendency in the distribution of the pitches, or even a musical gesture?*
. . . ponuntCaudas*: Only in case that the stem directions are still free at this point of the processing pipeline: Does the fact that all notes shall be beamed together determine a preferred stem direction?*
. . . visio*: What are the graphical coordinates of the whole beam aggregate? Or those of its fragments, if a printable solution can only be found after breaking it?*
. . . inLineolas*: How does the graphic appearance of the beam symbols interact with the individual lines of the staff?*

Each of the relevant publications on historic engraving treats all or most of these questions, but only separately, see Table 3. There are *no complete and explicit* algorithms outside the heads of the engravers, who have done this job over centuries. On the other hand, nowadays digital note setting programs necessarily contain such an algorithm, but those are not published. Further research will thus include reverse engineering.

Table 2 shows a possible modelling of the algorithm's input data, and possible properties of its result. Again, restrictions on these can serve as further input parameters.

4.1 Ergonomic Significance of Beam Inclination

The first property, *inclinatioSignificans*, is discussed with very different results. [19, p. 42] [18, p. 46 p.] [27, p. 155, 168 pp.] [21, p. 22 pp., 169 pp.] A general consensus is that an overall tendency of the pitches shall be expressed by the inclination of the beam, which historically had been decided by the taste of the engraver (unless restricted by collisions, see below.) But the opinions in details, given only by examples, do differ widely. [19, p. 42] demands a maximum steepness of 30 degrees. Many authors on nineteenth to twentieth century engraving practice impose a *maximal lift*, not a steepness—required by the physics of copper plate engraving, see section 4.4 below. [11]

In general, this area appears to require *non-local* considerations, corresponding to **nota.trabes.priorInfluit/successorInfluit** from Table 2. For instance, to use horizontal beams for Alberti bass figures is motivated by the fact that these are immediately repeated, so that the distribution of the note head heights is stationary—not over a single beam aggregate, but over their sequence. [27, p. 154 p., 159 p.][19, p. 40 p.][21, p. 25] Here future work is required—our current formalization is restricted to *local* phenomena.

4.2 Stem Direction of Beam Aggregates

The question how the decision that a group of notes shall be beamed together influences the stem directions (of course only in cases when these are still free), is discussed by [1, p. 94 pp.] [20, p. 88] [21, p. 24 p.][27, p. 154] [19, p. 40 p.]

4.3 Graphical Placement of Beam Aggregates

The raison d'être of any layout algorithm is to find a final graphical position of the beam aggregate. Its input are the mrp-beaming computed so far, pitches of the note heads, and additional parameters. The properties listed in Table 2 can all be made input parameters by defining restrictions. Its output are the coordinates of one or more beam fragments, esp. their inclinations. [18, p. 45 pp.] [1, p. 97 pp.] [1, p. 115 pp.] [19, p. 42][21, p. 17 pp.][27, p. 155 pp.]

In most cases such an algorithm will be a *partial* function:

PROP. nota.trabes.conditionesConfligentes: *no (simple) solution can be found which fulfills all requirements stated by the input data (mrp-beaming, pitches, and parameters).*
. . . vocesConfligentes: *a conflict with the positions of the graphical representations of notes from* another voice *obstructs finding a solution.*

The second case is not formalized in our approach so far. A possible idea is to give a list of "blocked rectangles" as additional input parameters. Whenever no single beam aggregate can be found, the remedy of further splits can perhaps be applied, as described in sections 4.5 pp.

4.4 Fine Tuning against the Staff Lines

Historically, much attention had been payed to *nota.trabes.inLineolas*, the relation of the beam to the staff line, mainly to avoid too small areas of white paper, problematic with traditional mechanical engraving technologies. [18, p. 43 pp.] [1, p. 98 pp.] [19, p. 41 p.] [24, p. 9 pp.][21, p. 17 pp.][27, p. 25 p.,42, 161 pp.] But also nowadays, "wedges" and gaps should still be avoided for their bad psychological effects on legibility.

[11] See the sections "3.5.2.1 traditional steepness" v. "3.5.2.2 contemporary" in [27].

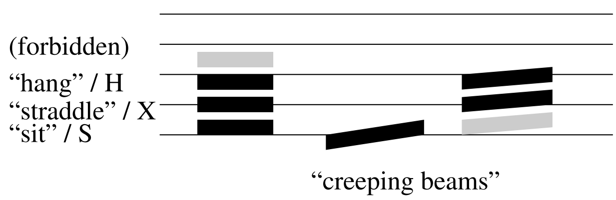


Figure 7. Beams versus Staff Lines

Most authors agree on

PROP. nota.trabes.tresInTresLineolis: *Let h be the height of a beam, d the distance between two beams, and s between two staff lines. Then it holds that $3*h+2*d = 2*s$.*

(Remarkably, all authors treat the width of the staff line as zero.) Most of the authors agree on the solution $4*d = 2*h = s$, but e.g. [24, p. 9] allows reducing h while sticking to the formula. Only [27, p. 42 p.] proposes the formula $4*h+3*d = 3*s$ (**...quatuorInQuatuorLineolis**, which is indeed much more flexible) and $h = 1.52*d$.

Under *trabes.tresInTresLineolis*, there are four positions of a horizontal beam relative to staff lines, one of which is forbidden. The others are called "sit" (S), "straddle" (X), and "hang" (H), and must follow in this order bottom-up for a horizontal 1/32 beam aggregate, see Figure 7. Continuing this rule, an aggregate of four or more parallel horizontal beams necessarily includes the forbidden position between the lines. [1, p. 125 p.] and [24, p. 11 p.] propose for this case to enlarge d to $s/2$, but only when the beams indeed fall into the staff. This implies

PROP. nota.trabes.sineLineolis: *Beams drawn in a staff and beams outside are treated differently.*

For *slanted* beams, the same relative positions S, X, and H apply for their starts and endings.

Most authors prefer

PROP. nota.trabes.subLineola: *beams which hold contact to one and the same staff line throughout.*

These are called "creeping beams" ("schleichende Balken") by Chlapik [19, p. 42], but he explicitly restricts their applicability, due to *inclinatioSignificans*.

A single creeping beam thus can have a maximum lifting of $s/2$, a double beam of $s/4$, a triple beam is not possible, see again Figure 7. (Creeping beams are the typical case where not the maximum steepness but the maximum lifting is fixed as an input parameter. Some authors allow the three slanted beams from the Figure with stems going up [="downstemmed" noteheads], because then no "wedge" appears. Else the beams must be shifted down by $s/4$.)

4.5 Resolving Conflicts by Breaking Beams

Giving the beam aggregates, the note head positions, and the additional parameters, an algorithm may fail to find a solution. The most simple and most frequently used

$$\mathsf{Heads} = \mathsf{seq}(\mathbb{Q} \times \mathbb{P}\mathbb{Q})$$

$$H \in \mathsf{Heads} \wedge 1 \leq k_1 < k_2 \leq \#H \Longrightarrow \pi_1(H(k_1)) < \pi_1(H(k_2))$$

// = input data: sorted list of x-coordinates of stems and y-cooordinates of the attached note heads (possibly chords!)

// Sequence indexing starts with 1

$$\mathsf{Beams} = \mathsf{seq}(\mathbb{Q} \times \mathbb{Q} \times \mathbb{Q} \times \mathbb{Q})$$

$$B \in \mathsf{Beams} \wedge 1 \leq m \leq \#B \wedge 1 \leq n \leq \#B - 1 \Rightarrow \pi_1(B(m)) < \pi_3(B(m)) \wedge \pi_3(B(n)) \leq \pi_1(B(n+1))$$

// = output data: list of start and end point coordinates of the top-level beam or of its fragments

$$\mathsf{Layout} : \mathsf{Heads} \times \mathsf{Params} \nrightarrow \mathsf{Beams}$$

$$\mathsf{Layout}(H, P) = B$$

$$\mathsf{beamedBy} : \mathbb{N} \leftrightarrow \mathbb{N} \qquad \mathsf{beamHeight} : (\mathbb{N} \times \mathbb{N}) \nrightarrow \mathbb{Q} \qquad \mathsf{sides} : \mathbb{N} \to \mathbb{P}\{-1, +1\}$$

$$\mathsf{beamedBy}(m, n) \Longleftrightarrow H(m) = (x_m, \_) \wedge B(n) = (x_1, \_, x_2, \_) \wedge x_1 \leq x_n \leq x_2$$

$$\mathsf{beamHeight}(m, n) = y \Longleftrightarrow H(m) = (x_m, \_) \wedge B(n) = (x_1, y_1, x_2, y_2) \wedge y = (y_2 - y_1)/(x_2 - x_1) * (x_n - x_1) + y_1$$

$$x \in \mathsf{sides}(m) \Longleftrightarrow \exists n : \mathbb{N}, y : \mathbb{Q} \bullet y \in \pi_2(H(m)) \wedge x = \mathsf{sgn}(y - \mathsf{beamHeight}(m, n))$$

$$\mathsf{knees} = \{m : 2..\#H \mid \mathsf{sides}(m) \neq \mathsf{sides}(m-1)\}$$

Further sensible input parameters to a Layout Algorithm:

$nota.trabes.cauda.max = c_{\mathrm{A}} : \mathbb{Q}_{>0}$ the maximal length of the very first and last stems.
$nota.trabes.cauda.maxInterior = c_{\mathrm{B}} : \mathbb{Q}_{>0}$ the maximal length of all other stems in a beam aggregate.
$nota.trabes.cauda.min = c_{\mathrm{I}} : \mathbb{Q}_{>0}$ the minimal length of the very first and last stems.
$nota.trabes.cauda.minInterior = c_{\mathrm{J}} : \mathbb{Q}_{>0}$ the minimal length of all other stems in a beam aggregate.
($nota.trabes.cauda.solaMin/solaMax : \mathbb{Q}_{>0}$ the minimal/maximal length of a stem with no beams.)
$nota.trabes.maxInclinatio = d_{\mathrm{A}} : \mathbb{Q}_{\geq 0}$ the maximal steepness of a beam.
$nota.trabes.maxAltitudo = h_{\mathrm{A}} : \mathbb{Q}_{\geq 0}$ the maximal lifting of a beam (= absolute difference of first and last y pos).
nota.trabes.priorInfluit = the appearance of a beam aggregate changes with the contents of a preceding aggregate.
nota.trabes.successorInfluit = the appearance of a beam aggregate changes with the contents of a following aggregate.

Table 2. Graphical Properties of Beam Layout

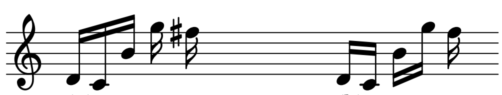
method for solving such a conflict is to break the beam into two parts, which are then dealt with separately. [21, p. 25] Since music is read from left to right, and the reader should be surprised as little as possible, this is naturally done at the rightmost position, i.e. as "late" as possible.

This break leads to a gap in the beaming; the resulting parts are then processed with the same layout algorithm recursively. The resulting aggregates are called *mrpg-beamings*. In Table 2 this is modelled by the result type Beams being a sequence of coordinate pairs. A gap after index position m of the input note sequence is characterized by $\mathsf{beamedBy}(m) \cap \mathsf{beamedBy}(m+1) = \varnothing$.

Again it is worth mentioning that possibly *trabes.significantVocem* is given up and hence in multi-voice writing the voice leading may become less clear.

Applying just the break (in the narrow sense of this word) necessarily produces stemlets. These can subsequently be processed/removed in the same way as known from the pauses in the mode *nota.trabes.sopraPausam.separans*: by applying *ELIM-trabulaNonSubTrabem*, *ELIM-trabulaOmnesContraTrabes*, *ELIM-trabulaSola*, and (in rare cases) *ELIM-trabulaeContraIdem*, as already specified above in section 2.4 and in Figure 6.

The breaks inserted by this algorithm are in the syntactic sphere alone, they are introduced for mere graphical reasons. A straightforward solution is (a):



(a) (b)

Whenever the semantics shall influence these breaks, they must be applied in the preceding phase explicitly. Only (b) expresses correctly the jump of the hand position on a keyboard, according to *nota.trabes.cumPositioneManu*. Thus, for a flute (a) is correct, for a piano (b). Our strict pipeline architecture does not support these considerations, but could be enhanced by "predetermined breaking points" as known from the TEX typesetting system: the preceding semantic processing could assign "preferences and penalties" to breaking points which are used by the subsequent layout procedure only in case they are required. This is one of the bypasses we identified in our transformation pipeline, see the dotted line (b) in Figure 2 on page 2.

4.6 Resolving Conflicts by Knees

Another remedy with less loss of substance but more graphical overhead is the "extraordinary beam", called "knee" (from the German "Knie") in the following. It means a beam from which stems go *both up and down*. [1, p. 118 p.] [19, p. 43] [24, p. 12 pp.] [20, p. 86, 88 pp.] [21, p. 26] [27, p. 93, 135, 153] More formally the indexes after a knee are given by knees from Table 2.

PROP. nota.trabes.genus.inLineae: *a knee is contained totally in one staff.*
. . . interLineas: *a knee is contained in or crosses the space between two staves.*

The latter is a frequent case in two-staff writing. [12]

[12] [19, p. 43] says about this case even "[Die] Balkenverbindung von einer Zeile in die andere ist nicht als ≫Knie≪ anzusprechen." ("Beams from one staff into another shall not be called ≫knee≪.")

Significant properties are:

PROP. nota.trabes.numerusGenuum: *number of "knees" = points where the stems change the side = $\#\mathsf{knees}$ in Table 2.*
…numerusInGenui: *the number of beams which cross the knee.*
…numerusInterGenibus: *the number of consecutive note heads between two knees.*
…caudaVersusCaudam: *whether two stems to both sides can appear at the same x-coordinate. This corresponds to $\{-1,1\} \in \mathsf{ran(sides)}$ in Table 2.*

Restrictions to these properties again can be used as further input parameters to an algorithm.

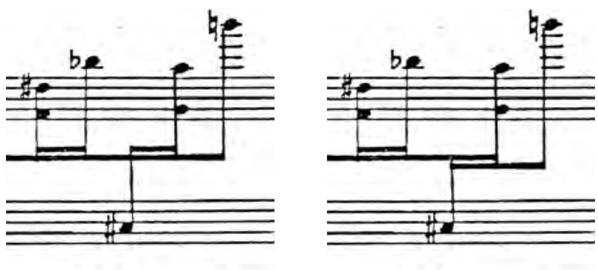
When multiple beams go "through" the knee (i.e. appear with a stem to one side immediately followed by a stem to the other side, or more complicated cases of change), they completely exchange their meanings. For instance in a group of three, the graphical symbol which is the "highest" beam on the one side, representing thus the "eighth" flag, becomes the "lowest" on the other, representing the thirty-second flag, et vice versa. [13] Anyway it should hold

PROP. nota.trabes.summaeInGenu: *The bundle of beams which cross the "point of change" at a knee are on both sides the top-most beams in the aggregate; all further beams are added to them "below" = towards the note heads.*

[24, p. 13 p.] exhaustively discussed this problem of beam addition, without finding a simple rule as our *summaeInGenu*. Herma by Xenakis violates *summaeInGenu*, but instead follows

PROP. nota.trabes.inGenuCumPluribus: *If there is a vast majority of one stem direction over the other, this defines the meaning of the beams.*

This alternative leads (in m. 87, as executed) to the left version, while our rule would produce the right one, clearly demonstrating the change of meaning:



The insertion of a knee may conflict with *nota.trabes.inclinatioSignificans*. In Figure 8 the cases for the sequence "up-stem followed by down-stem" are labelled by the comparison results of the note heads and the stem ends. (For the opposite sequence the comparison operators signs must be inverted.)

PROP. nota.trabes.genuParadoxum: *in the case "up-stem followed by down-stem", the knee cases $(<,>)$ and $(<,>>)$ are called* paradox *because an inclining sequence of note heads / pitches is graphically realized by a declining slope of the beam. (Vice versa in the case "down-stem to up-stem").*

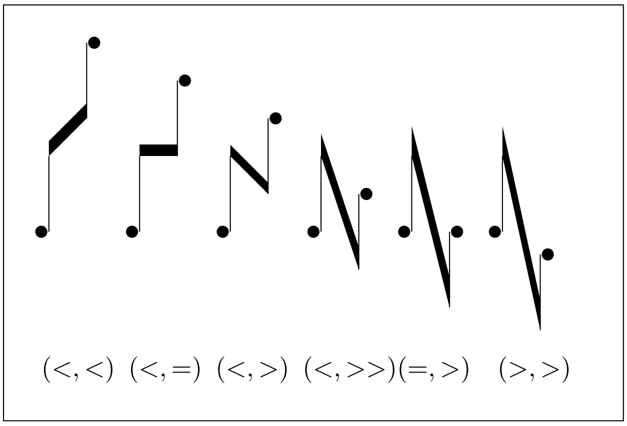


Figure 8. Categories of Knees

We leave it open if also the cases $(<,=)$ and $(=,>)$ shall be called "paradox". It holds:
a) If the lower note is stemmed down and the higher note is stemmed up, then no paradox shows up. (The tendency of the note heads is even *amplified* by the beam.)
b) Otherwise, if the absolute distance of the note heads is smaller than double the minimal stem length, then a paradox is unavoidable.

Paradox beams are explicitly forbidden by [19, p. 43].

While beams of the last category $(>,>)$ seem unpractical, they are indeed frequently required when more than two voices share a (piano) staff. See Brahms, Intermezzo op. 119 Nr. 1 m. 61: (Simrock, 1893, plate 10055):

4.7 Resolving Conflicts by Changing Height and/or Inclination

A rather modern means to resolve graphical beaming conflicts is to change the inclination or the height of the beam "underway", when crossing a particular stem S. Systematically this can be treated as breaking the beam into two fragments which overlap just at S. This is not one single break process, as in section 4.5, but two, namely immediately left and right of S. Then all beamlets resulting from these breaks are removed and both fragments are laid out independently. The last stem of the first and the first stem of the second resulting beaming aggregate must point into the same direction and will be unified in the final printed result. [14]

Such a result is represented in context of Table 2 by two elements of the sequence B according to the pattern

[13] While this indeed poses severe problems to formal definitions of syntax and semantics, it has no consequences in practice because only the *number* of beams is relevant, not their "undisturbed self-identity". This effect is called "non-deterministic determinedness" by [4, p. 298].

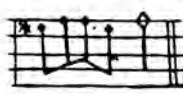
[14] These considerations do not cover to case of giving *two* stems (one in each direction) to the common notehead at S. This must be treated separately, because it possibly has stronger impact on the notated voice leading.

$\langle \ldots (x_0, y_0, X, Y_1), (X, Y_2, x_3, y_3), \ldots \rangle$. It can exhibit any combination of

PROP. nota.trabes.translataeInCaudis: *the distance of the top-level beam from the note head is different on both sides of the stem ($Y_1 \neq Y_2$).*
. . . angulusFractumInCaudae: *the top-level beams on both sides of the stem are not in 180° but in a different angle* $(Y_1 - y_0)/(X - x_0) \neq (y_3 - Y_2)/(x_3 - X)$.

See Figure 9 a) and b). The graphical syntax of these properties is modelled by *nota.graph.simplex.XIV* and *XV* in [4, p. 108]. The selection of allowed properties and combinations can be a further input parameter. The occurrence of neither would indicate inconsistent programming, because in this case the original rendering attempt (without any breaking) should have succeeded.

Both properties and their combination are rather frequent in classical avantgarde notation styles, see Stockhausen, Klavierstück X. But indeed *angulusFractumInCaudae* are found in much older prints and hand-writings. [1, p. 88 p.] gives two examples from 1690 (regrettably not specifying the source):



5. ASPECTS NOT COVERED

A mathematical re-modelling of inter-human symbol systems determined by history and culture is not and should not try to be exhaustive. But it should clearly circumscribe the areas of non-formalization. Whether extended formalization beyond these limits is sensible may be left to future work.

Some of the properties we have not (yet?) formalized:

PROP. nota.trabes.angulusFractumInterCaudis: *the top-level beam is bent at a point* between two *stems.*

This variant, see Figure 9 c), can be integrated into our model by breaking the aggregate between the stems, laying out the fragments independently and printing the resulting beams in a prolongated way, extending to their meeting point between the aggregates. Of course such a meeting point must exist, and research on its preconditions is necessary. This appears to be related to the problems of inter-aggregate dependencies and graphical harmonizations, see *nota.trabes.priorInfluit* etc. above.

PROP. nota.trabes.multaAdCaudam: *Multiple groups of beams are connected to the same side of the same stem.*

Different patterns of this kind can be found in advanced notation. A simple sub-kind is

PROP. nota.trabes.multaAdCaudam.perOrnamentum: *Additional beam groups extend only locally and indicate an ornament, i.e. a figure played in a sub-ordinated local organization of time.*

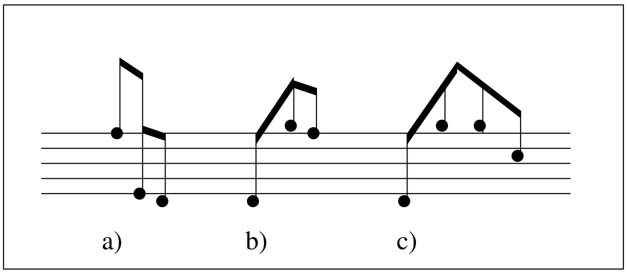
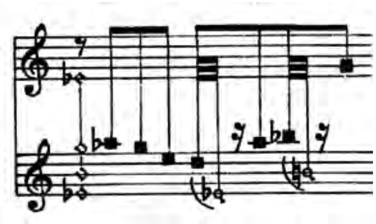


Figure 9. Displaced and Bent Beams

The following example (Lachenmann, Toccatina, p. 3, cited by [28] p. 24) is a such a simple case, where local "Nachschläge" are notated by a subordinate group of beams: [15]



PROP. nota.trabes.trabulaUtVexilium : *A beamlet is represented in the form of a flag, even when the stem has (long) beams.*

This is a mere graphical transformation. [20, p. 94] mentions Boulez, Le marteau sans maître, and rejects this kind of writing sharply.

6. CONCLUSION

We have presented a four-staged pipeline architecture for calculation and layout of a beam aggregate for a given meter, rhythm, and pitches. We have identified about sixty properties which influence the result. We have assigned statements from literature (which nearly exclusively is concerned with conventional CWN) to the different stages of processing.

The pipeline architecture helps to clarify input conditions, output specifications, tests of both, documentation, etc. considerably. Nevertheless we found two bypasses which do not fit cleanly into the sequential order, see Figure 2: (a) the number of note heads modifying the genuine beaming (an issue not treated in the literature so far), and (b) predetermined breaking points, prepared by semantic parameters to be used by the graphic layout process if required.

Table 3 in the Appendix shows where the cited standard works treat the different topics. Tables 4 and 5 list all properties found so far, referring to the corresponding section of this article.

Acknowledgments

Many thanks to Hartmuth Kinzler, Osnabrück, for a rich collection of beaming examples.

[15] Another important example is Stockhausen, Klavierstück X. But in its foreword special semantics of beams are explicitly defined anyhow.

7. REFERENCES

[1] T. Ross, *The Art of Music Engraving and Processing*. New York: Hansen Books, 1987.

[2] M. Lepper, M. Oehler, H. Kinzler, and B. Trancón y Widemann, "Diminuendo al bottom —- clarifying the semantics of music notation by re-modeling," *plosOne*, 2019. [Online]. Available: https://doi.org/10.1371/journal.pone.0224688

[3] M. Malt and B. Sluchin, "Geometric notation for time-bracket works, application and performance. the case of John Cage's Music for ___," in *Proceedings of the International Conference on Technologies for Music Notation and Representation – TENOR'20*, R. Gottfried, G. Hajdu, J. Sello, A. Anatrini, and J. MacCallum, Eds. Hamburg, Germany: Hamburg University for Music and Theater, 2020, pp. 114–119, http://tenor-conference.org/proceedings.html[20210102].

[4] M. Lepper, "De linguis musicam notare," Ph.D. dissertation, Universität Osnabrück, Osnabrück, 2021.

[5] D. A. Byrd, "Music Notation By Computer," Ph.D. dissertation, Indiana University, Bloomington, 1984.

[6] M. Gieseking, "Code-basierte Generierung interaktiver Notengraphik," Ph.D. dissertation, Universität Osnabrück, 2000.

[7] A. Fleischer, G. Mazzola, and T. Noll, "Computergestützte Musiktheorie: zur Konzeption der Software RUBATO für musikalische Analyse und Performance," *Musiktheorie*, vol. 15, pp. 314–325, 2000.

[8] G. W. Cooper and L. B. Meyer, *The Rhythmic Structure of Music*. Chicago: The University of Chicago, 01 1960.

[9] A. T. Cemgil, P. Desain, and B. Kappen, "Rhythm quantization for transcription," *Computer Music Journal*, vol. 24, no. 2, pp. 60–76, 2000.

[10] A. Cogliati, Z. Duan, and D. Temperley, "Transcribing human piano performances into music notation," in *Proc. of International Society for Music Information Retrieval Conference (ISMIR)*, J. Devaney, M. I. Mandel, D. Turnbull, and G. Tzanetakis, Eds., 2016, https://wp.nyu.edu/ismir2016/event/proceedings/.

[11] A. Ycart, F. Jacquemard, J. Bresson, and S. Staworko, "A supervised approach for rhythm transcription based on tree series enumeration," in *Proceedings of the International Computer Music Conference (ICMC 2016)*, H. Timmermans, Ed. Utrecht: Michigan Publishing, 2016, pp. 369–376.

[12] C. Agon, K. Haddad, and G. Assayag, "Representation and rendering of rhythmic structures," in *Second International Conference on WEB Delivering of Music*, C. Busch, M. Arnold, P. Nesi, and A. Schmuckler, Eds. Los Alamitos, CA, USA: IEEE Computer Society, 2002, pp. 109 – 113.

[13] M. Laurson and M. Kuuskankare, "From rtm-notation to enp-score-notation," in *Proceedings of Journées d'Informatique Musicale*, J. Clos, Ed. Montbeliard: L'ecole nationale de musique du pays te Montbeliard, 2003.

[14] A. Agostini and D. Ghisi, "A max library for musical notation and computer-aided composition," *Comput. Music J.*, vol. 39, no. 2, pp. 11–27, Jun. 2015. [Online]. Available: http://dx.doi.org/10.1162/COMJ_a_00296

[15] N. Didkovsky, "Java music specification language, v103 update," in *Proceedings of the International Computer Music Conference (ICMC 2004)*. Miami, Florida, USA: Michigan Publishing, 2004, pp. 742–745.

[16] M. Lepper and B. Trancón y Widemann, "metricsplit: An automated notation of rhythm aligned with metric structure," *Journal of Mathematics and Music*, vol. 13, no. 1, pp. 60–84, 2019.

[17] F. Gouyon and S. Dixon, "A review of automatic rhythm description systems," *Computer Music Journal*, vol. 29(1), pp. 34–54, 2005.

[18] A. C. Vinci, *Fundamentals of Traditional Musical Notation*. Kent, OH: Kent State University Press, 1985.

[19] H. Chlapik, *Die Praxis des Notengraphikers*. Wien: Doblinger, 1987.

[20] G. Read, *Music Notation: a Manual of Modern Practice*. New York: Taplinger, 1979.

[21] E. Gould, *Behind Bars: the Definitive Guide to Music Notation*. London: Faber, 2011.

[22] D. Taupin, R. Mitchell, and A. Egler, "MusixTEX – Using TEX to write polyphonic or instrumental music," 2002, www.texdoc.net/texmf-dist/doc/generic/musixtex/musixdoc.pdf[20200922].

[23] M. Lepper and B. Trancón y Widemann, "Metricsplit – Automated Notation of Rhythms, Adequate to a Metric Structure," Technische Universität Ilmenau, Ilmenau, Tech. Rep., Mar 2017. [Online]. Available: http://nbn-resolving.de/urn:nbn:de:gbv:ilm1-2017200259

[24] K. Stone, *Music Notation in the Twentieth Century: a Practical Guidebook*. New York: Norton, 1980.

[25] M. Lepper, M. Oehler, H. Kinzler, and B. Trancón y Widemann, "Crowded staves – rules for semantics and style of conventional multiple-voices-per-staff musical notation," in *Proceedings of the TENOR 2020 conference*, Hamburg, 2020, pp. 1–12, https://www.tenor-conference.org/proceedings.html.

[26] H. W. Nienhuys and J. Nieuwenhuizen, "Lilypond, a system for automated music engraving," in *Proceedings of the XIV Colloquium on Musical Informatics*, Firenze, 2003, pp. 167–172.

• Genuine beams:
[18, p. 27 pp.] [1, p. 91 pp.] [19, p. 43 p.] [19, p. 46 pp.] [20, p. 80 pp.] [21, p. 153 pp.]
• Modified genuine beams:
(no predecessors)
• Stem direction of beam aggregate:
[1, p. 94 pp.] [20, p. 88] [21, p. 24 p.] [27, p. 154]
• Beams crossing pauses:
[18, p. 49] [19, p. 46] [24, p. 15 p.] [20, p. 88, 213]
• Height of beam symbols:
[1, p. 94 pp.][1, p. 119 pp.][19, p. 41][24, p. 9] [20, p. 80][21, p. 17][27, p. 42 p.]
• Significance of the beam's steepness:
[19, p. 42] [18, p. 45 p.] [21, p. 22 pp., 169 pp.] [27, p. 155, 168 pp.]
• Value of the beam's steepness:
[18, p. 45 pp.] [1, p. 97 pp.] [1, p. 115 pp.] [19, p. 42][21, p. 17 pp.][27, p. 155 pp.]
• Knees:
[1, p. 126 p.] [19, p. 43 p.] [24, p. 12 pp.] [29, p. 56] [20, p. 86, 88 pp.] [21, p. 26] [27, p. 93, 135, 153]
• Relation to stafflines / micro positioning:
[18, p. 43 p., 47] [1, p. 98 pp.] [1, p. 119 pp.] [19, p. 41 p.] [24, p. 9 pp.][21, p. 17 pp.][27, p. 25 p.,42, 161 pp.]
• Beams for polymetrics:
[24, p. 116 pp.][21, p. 171, 175 pp.] [20, p. 170 pp.]
• Feathered Beams:
[19, p. 47] [20, p. 94] [24, p. 124, 141] [21, p. 158]

Table 3. Synopsis of the Literature w. r. t. the Discussed Issues

[27] H. Wanske, *Musiknotation – Von der Syntax der Notenstichs zum EDV-gesteuerten Notensatz.* Mainz: Schott, 1988.

[28] C. Dimpker, *Extended Notation – The depiction of the unconventional.* Berlin: Lit Verlag, 2013.

[29] K. Hader, *Aus der Werkstatt eines Notenstechers.* Wien: Waldheim-Eberle, 1948.

A. APPENDICES

Table 3 gives a synopsis of the discussion in standard literature. Tables 4 and 5 list all properties appearing in this article.

A.1 Polymetric Constellations Expressible by Beams

Following the classification grid in [4], a simple polymetric situation can be characterized by a quadruple indicating the relations between (1) the start points of the metric pattern, (2) the physical lengths, (3) the inner metric structure, and (4) the notated tempo, i.e. the relation from notated values to physical time.

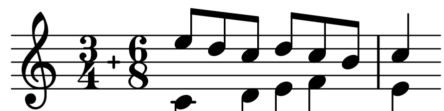
Many simple constellations can be expressed by beams alone. Examples without further comments:

$(\neq, =, =, =)$ (Petterson, Sinf 11)

$(=, \neq, \neq, =)$ (Petterson, Sinf 10)

$(=, =, \neq, \neq)$ (Beethoven op. 111)

$(=, =, \neq, =)$ (constructed example)



$(=, =, =, \neq)$ (Mozart Klaviersonate K 457)
nota.trabes.unificataIndicantRubatum[Kinzler]

• *nota.trabes.utVexilia*
= basic principle how beams express duration, see section 2.1.
• *nota.trabes.notaeUtNota*
= basic principle that a single note and a replacing group behave identically towards their neighbors, see section 2.1.
• *nota.trabes.alteraNatas* : $\mathbb{P}(\mathbb{N} \times \mathbb{Z} \times \mathbb{N} \times \mathbb{N})$
= collection of rules to modify the genuine beaming, see section 2.2
• *nota.trabes.sopraPausam* : $\{\mathsf{perCaudulam}, \mathsf{transiens}, \mathsf{separans}\}$
= way how beams cross pauses, see Section 2.4.
• *nota.trabes.significantVocem*
= a beam fulfills also the role to indicate voice identity, thus voice leading.
• *nota.trabes.trabulaeContraIdem*, *...trabulaeOmnesContraTrabes*, *...trabulaNonSubTrabem*, *...trabulaSola*
= local transformations of beamlets, see Section 2.4.
• *nota.cauda.significat.vocem*
= the stem direction indicates the notational voice.
• *...instrumentum/accentum/manum/modumAgitur*
= the stem direction indicates the employed instrument (with percussion) / the active hand / a certain way of sound production.
• *nota.cauda.significat.nihil*
= the stem direction is free and can be determined according to graphical needs.
• *nota.cauda.significat.vocem.trabsSeparataCaudaMutata*
= when the stem direction changes midways, the beam is broken.
• *nota.cauda.significat.vocem.trabsSeparataLineaMutata*
= when the voice changes the staff midways, the beam is broken.
• *nota.voces.unaUtDue_pausasPerdatas*
= one single voice with two kinds of events (e.g. by two hands) is notated as *two* voices, where the events in the one voice are read as pause symbols for the other.
• *nota.trabes.extera.separans*
= external parameters make a beam break.
• *nota.trabes.extera.ligans*
= external parameters make two separated beams join.
• *nota.trabes.cumVerborum.syllabis/nominibus/lineis*
= the beams are broken according to the syllables/words/lines of the sung text.
• *nota.trabes.separatae.cumMelo/cumLigato*
= the beams are broken according to motif structure/legato execution.
• *nota.trabes.ligataeContraNates*
= beams are connected against the genuine beams prescribed by the meter.
• *nota.metraMulta.perTrabem*
= the beams are used to clarify a multi meter situation (shifted or shifted meter in different voices=.
• *nota.trabes.cumPositioneManu*
= the beams follow the positional changes ("jumps") of a hand on a keyboard.
• *nota.trabes.accelerans* : $\mathsf{Seq}(\mathbb{Q} \times \mathbb{N} \times \mathbb{Q})$
= tempo curve for compressed and expanded beam heights and distances.
• *nota.trabes.inclinatioSignificans*
= the steepness of the beam is related to the distribution of the pitches, or even to the gesture of the motif.
• *nota.trabes.ponuntCaudas*
= the fact that the notes are beamed together influences the stem direction.
• *nota.trabes.visio*
= the final visual layout of the beams in its graphical context.
• *nota.trabes.inLineolas*
= the relation of the particular beams to the lines of the staff.
• *nota.trabes.cauda.max/maxInterior/min/minInterior* : $\mathbb{Q}$,
nota.trabes.maxInclinatio/maxAltitudo : $\mathbb{Q}$
= input parameters for a layout algorithm.
• *nota.trabes.conditionesConfligentes*
= the case that the input parameters conflict and prevent a (simple) solution.
• *nota.trabes.vocesConfligentes*
= the case that the spatial requirements of another voice (in the same staff) prevents a (simple) solution.

Table 4. All Properties Found for Beams and Their Transformations—Part I

• *nota.trabes.tresInTresLineolis*
= formula from copper engraving which allows to put three beams into two spaces of a staff.
• *nota.trabes.sineLineolis*
= the case that beams outside a staff are treated differently than inside.
• *nota.trabes.subLineola*
= beams which hold contact to one and the same staff line throughout. = "creeping beams".= "schleichende Balken".
• *nota.trabes.priorInfluit/successorInfluit*
= context dependency: a change in the input data for a preceding/following beam aggregate influences the layout.
• *nota.trabes.genus.inLineae*
A "knee" ("exceptional beams", German "Knie") is contained in the graphic area of a staff's lines.
• *nota.trabes.genus.interLineas*
A knee is contained in graphic space between two staves.
• *nota.trabes.numerusGenuum*
= the number of knees in one particular rendering.
• *nota.trabes.numerusInGenui*
= the number of beams which "go through" the knee.
• *nota.trabes.numerusInterGenibus*
= the number of stems between two knees/a knee and the end of the aggregate.
• *nota.trabes.caudaVersusCaudam*
= whether a stem in both directions appears at the same point of a beam.
• *nota.trabes.summaeInGenu*
= the rule that the top-most beams only cross a knee.
• *nota.trabes.inGenuCumPluribus*
= the rule that the majority side of stems decides the meaning of the beams across a knee.
• *nota.trabes.genuParadoxum*
= that the beam in a knee aggregate goes contrarily to the pitches.
• *nota.trabes.translataeInCaudis*
= that the beams on two sides of a stem start at different distances.
• *nota.trabes.angulusFractumInCaudae*
= that the beams on two sides of a stem start in different angles.
• *nota.trabes.angulusFractumInterCaudis*
= that a beam makes a "turn" between two stems.
• *nota.trabes.multaAdCaudam*
= multiple groups of beams at the same side of a stem.
• *nota.trabes.multaAdCaudam.perOrnamentum*
= multiple groups of beams at the same side of a stem, but only for sub-ordinated, local time with fast events.
• *nota.trabes.trabulaUtVexilium*
= beamlets are printed as flags, even under beams.

Table 5. All Properties Found for Beams and Their Transformations—Part II