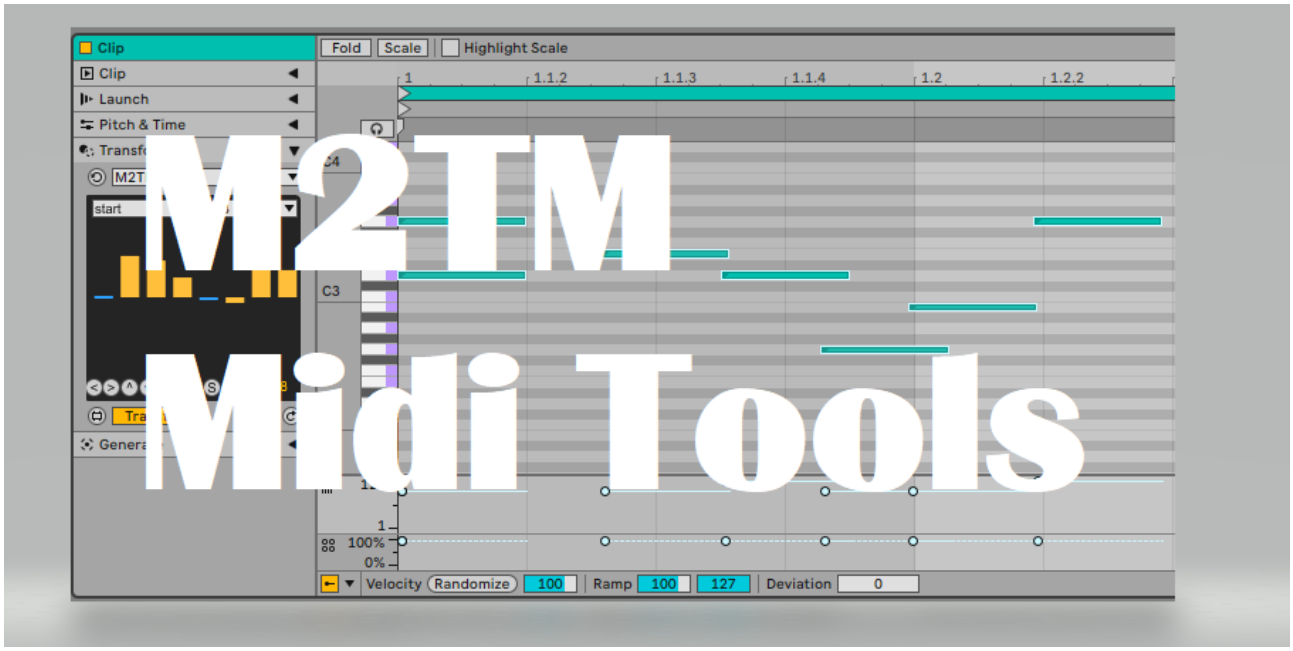


[M2TM Steps]
[M2TM Chords]

midi
2themax

The M2TM Midi Tools Collection

M2TM Midi Tools is a growing collection of MIDI Transformers and MIDI Generators for **Ableton Live Suite Edition version 12** or Ableton Live 12 Standard with support for Max-for-Live devices. These tools do NOT work in previous Live versions or in Live editions without Max-for-Live support.

> *As of this writing, the collection contains two components: the **M2TM Steps** transformer and the **M2TM Chords** generator. More tools are being developed and will be added in the future, either as part of the collection or as individual downloads*

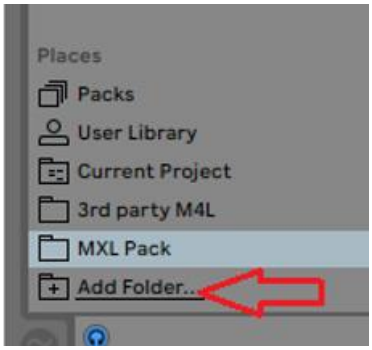
You can find the various **M2TM Midi Tools** here: http://www.gumroad.com/midi2themax

Setup and Documentation

Regardless of what individual **M2TM Midi Tools** components you downloaded or purchased, this manual encompasses all the tools in the collection.

Both the collection and individual components are distributed as a compressed ZIP file containing. We recommend that you create a folder on your hard disk named **M2TM Midi Tools,** and then uncompress the ZIP file in that folder.

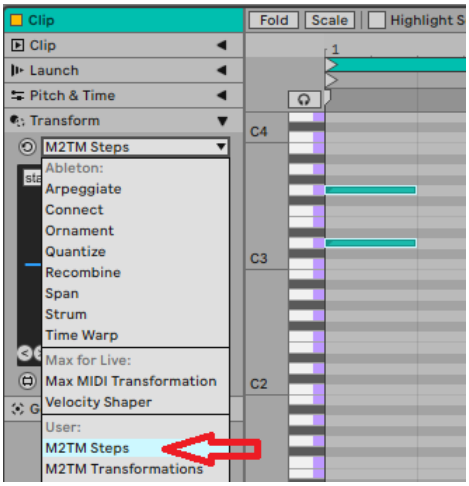
Inside Ableton Live, click in the **Add Folder** option in the left sidebar, and select the folder you just created. Alternatively, you can uncompress M2TM Midi Tools components in a subdirectory of a folder already listed in this section. For example, MXL Pack customers can create the M2TM Midi Tools folder as a subfolder of the MXL Pack folder.

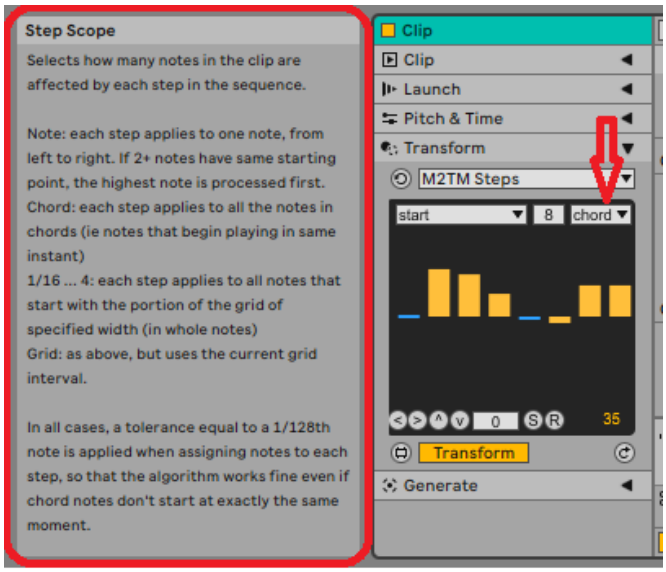


Next, create a MIDI clip, open the **Transform** tab of the clip properties, and click on the menu that lists all the installed MIDI Transformations, which should now contain the MIDI transformers in the M2TM Midi Tools collection.

***IMPORTANT:** if no M2TM component appears in the list, close and restart Ableton Live.*

Likewise, the menu in the **Generators** tab should include the MIDI generators in the M2TM Midi Tools collection, for example **M2TM Chords** (if you have purchased it)





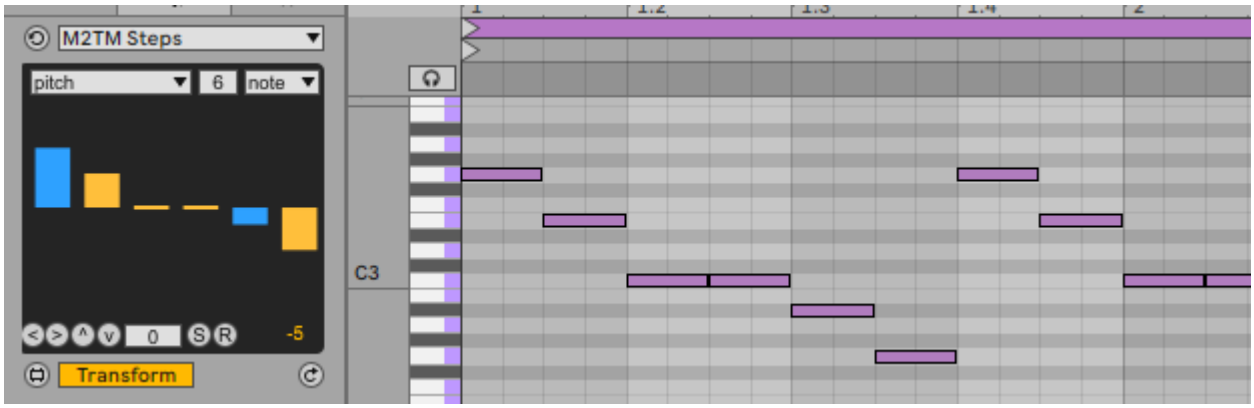
In addition to the PDF manual, you can learn how devices work by hovering the mouse on a field and look at Live's Help window (press the "**?**" key if the Help window is currently hidden).

Support

To report problems, propose ideas for new M2TM components, or suggest how to improve them, please contact us at midi2themax@gmail.com.

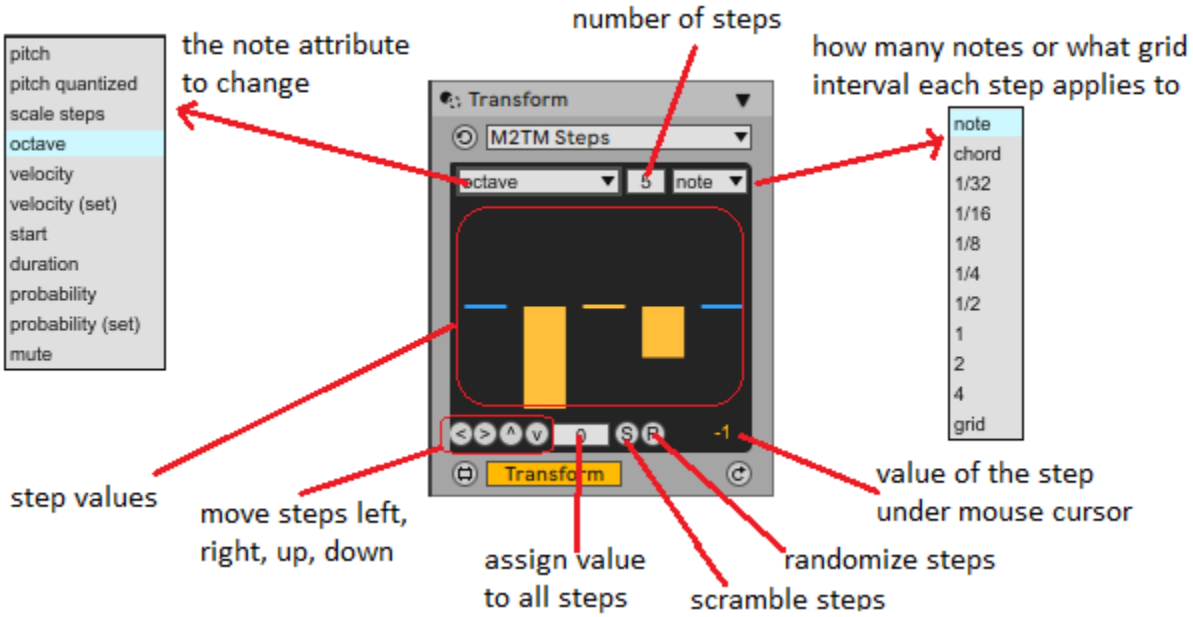
For more information about installing and using Ableton Live MIDI Tools, please see the documentation available on Ableton website: https://www.ableton.com/en/live-manual/12/midi-tools/

M2TM Steps



M2TM Steps is a MIDI Transformer that lets you change the main attributes of clip notes in a cyclic fashion. A "cycle" can contain from 2 to 64 steps and can be applied to pitch, velocity, start time, duration, probability and mute attribute of the clip notes that are selected (or all notes, if no note is currently selected).

For example, the figure above shows what happens if you apply a 6-step cycle to increment or decrement the **pitch** attribute of a series of C3 notes. In this specific case, the first and second notes in the clip are transposed up by 7 and 4 semitones, respectively; the third and fourth notes are not modified; the fifth and sixth notes are transposed down by 2 and 4 semitones, respectively. If the clip contains seven or more notes – as in the example above – the cycle is applied again, therefore the 7th and 8th notes are transposed up, the 9th and 10th notes are not modified, etc.



Note attribute and cycle length

The most important field is the **attribute** menu, which lets you decide which note attribute are modified and how steps are interpreted. There are the options available in the top-left menu:

pitch: transpose the note up or down by the specified number of semitones; steps values are in the range from -12 to +12.

pitch quantized: as above, but the resulting note is quantized to the current scale.

scale steps: transpose the note up or down by the specified number of scale steps; step values are in the range from -12 to + 12. For example, the value +2 transposes notes up by a third; the value -4 transposes the note down by a fourth.

Notice that the actual number of semitones depends on the current scale and the original note's pitch. For example, assuming that current scale is C Major and you are transposing one third up, C notes are transposed to E (four semitones up), whereas D notes are transposed to F (three semitones up).

octave: transpose the note up or down by the specified number of octaves; step values are in the range from -2 to 2. The resulting note is not quantized to the scale, therefore it is out-of-key if the original note was out-of-key.

velocity: increase or decrease the note's velocity by the specified step value, which can be in the range from -126 to +126.

velocity (set): the note's velocity is assigned the specified step value, which can be in the range from 1 to 127. The default value is 90.

start: anticipate or delay the note's starting time; step values are relative to the current grid size; the lowest value (-128) moves the note one grid position to the left, whereas the highest value (+128) moves the note one grid position to the right.

duration: change the note's duration; the lowest step value (-128) decreases the duration by the grid size, the highest step value (+128) increases the duration by the current grid interval.

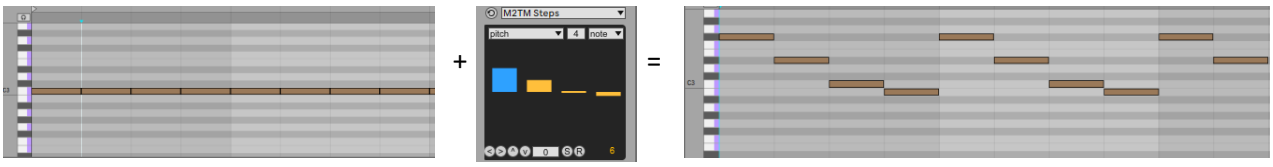
probability: increase or decrease the note's velocity probability by the specified step value, which can be in the range from -100 to +100.

probability (set): the note's probability is assigned the specified step value, which can be in the range from 0 to 100. The default value is 100.

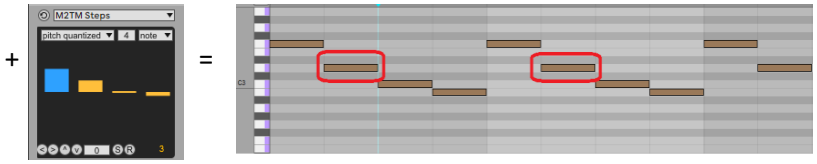
mute: step values can only be 0 (not muted) or 1 (muted).

The field in the top-center defines the **length of the cycle**, which can be a number between 2 and 64 (default is 8). Keep in mind that each note attribute uses a distinct cycle, with its own values and length.

Here are a few examples of how these options affect the notes in the clip. Let's start with a series of identical notes and apply a cycle of pitch transposition in semitones:

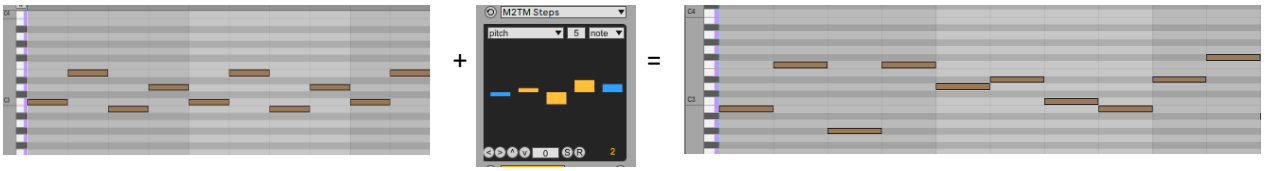


If the **pitch quantized** option is selected, the result is slightly different, because resulting notes automatically fit the current scale. Assuming that the C Major scale is selected, this is the result:

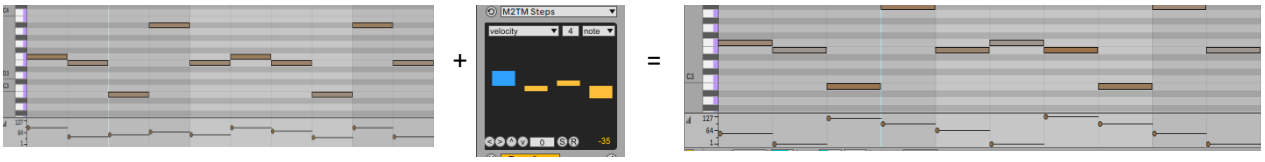


If the **scale steps** option is used, the resulting intervals are usually wider; in this case, notes automatically fit the current scale:

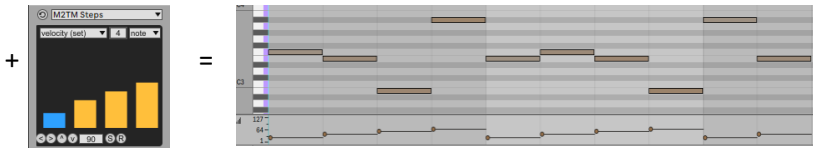
You get more interesting and less predictable results if you start with a melody rather than a group or repeated notes, or if you apply cycles of longer lengths. For example, you can start with a simple melody of four repeated notes, and apply a 5-steps cycle:



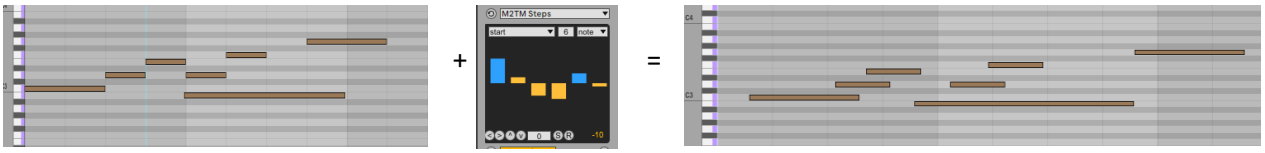
When working with velocities and probabilities, cycle steps can be applied as deltas of current values or as absolute values. To use deltas, you select the **velocity** or the **probability** option:



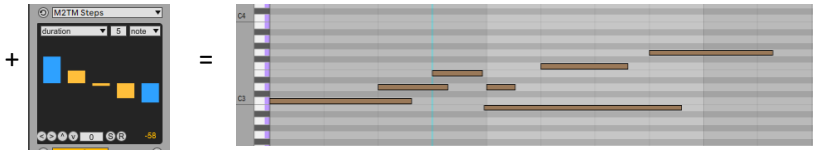
To use absolute values, you select the **velocity (set)** or the **probability (set)** option, and the current values of velocity or probability attributes are ignored:



The **start** option shifts notes horizontally on the grid. When the step value is at its minimum (-128) or its maximum (+128), the note is anticipated or delayed by the current grid interval. If grid interval is 1/16 or 1/32 you can use this option for minor adjustments of note attack, for example to “humanize” the resulting melody:



The **duration** option can decrease or increase the note duration; as for the **start** option, step values are relative to the current grid interval:



***NOTE**: When you use the top-left menu to switch a different attribute, all step values are reset to their default value, which is 90 for **velocity (set)**, 100 for **probability (set)**, and zero for all others. The only exception to this rule is when you switch from a pitch-related*

*attribute to another pitch-related attribute, e.g. from **pitch** to **pitch quantized** or **scale steps**: in these cases, step values and cycle length is preserved.*

Step scope

In general, steps are applied to notes in the order in which these notes appear in the note editor (from left to right); however, you can apply a given individual step to more than just one note. These options are available in the top-right menu:

note: steps are applied to individual notes; if two or more notes have the same start time (i.e. they are aligned vertically), notes are processed for the highest to the lowest note.

chord: if two or more notes play in the same moment, the are modified using the same step value.

1/32, 1/16, 1/4, 1/2: the clip is subdivided in intervals of this size (32nd-notes, 16th-notes, eight notes, quarter notes, and half notes) and notes in each interval are affected by the same step value; if a step corresponds to a grid portion that contains no notes, that step is ignored.

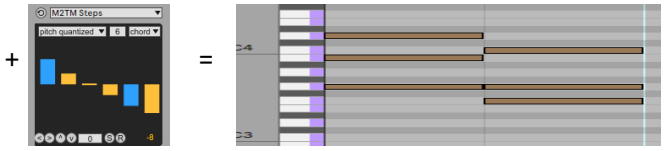
1, 2, 3, 4: like above, except the menu items correspond to number of whole notes (or measures, if current time is 4/4).

grid: like above, but the current grid interval is used.

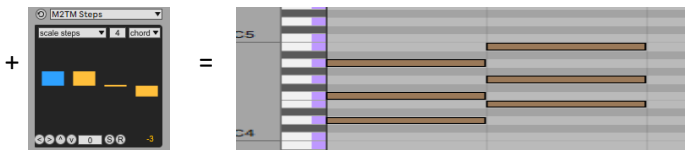
***NOTE**: a tolerance equal to 1/128th note is applied when assigning a note to a time interval. This approach is used so that notes that are slightly anticipated or delayed – for example, "strummed" chords or notes that have been "humanized" by means of the **start** option – are correctly grouped together.*

All the examples shown so far used the **note** option. However, when the clip contains chords, this option often delivers undesired results, because it destroys the relationship between chord notes. For example, see what happens if the clip contains chords built on minor or major thirds:

In such cases, the **chord** option is usually more appropriate:

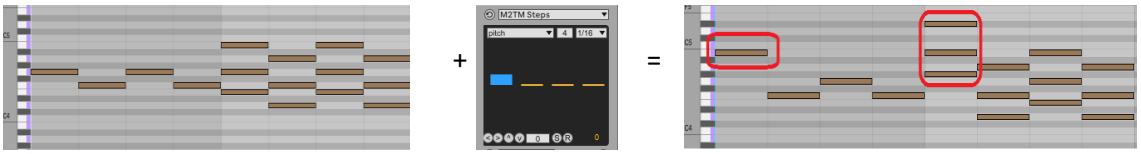


As you see, the first chord (C-E-G, or C major) has been transposed into G-B-D (G major); however, the second chord (D-F-A, or D minor) has been transposed into F-G-C, a chord that contains no thirds. This happened because of the "quantize to scale" adjustment after the transposition in semitones. To preserve the intervals in the original chord, the **scale step** option is often preferable:



At times, you might want to apply the same transformation to all the notes in a given time interval. For example, you might want to generate a variation of existing melody or harmony by transposing one

semitone up only the notes at the beginning of each quarternote interval. In this scenario, you might select the **1/16** option, set 4 as the cycle period, and select a non-zero value only for the first step in the cycle:



Step values

The buttons and fields near the bottom edge let you apply simple transformations to step values.

The first four buttons rotate, increase or decrease all values. The **set value** numeric field in the center allows you to assign the same value to all steps: you can either click and drag the mouse up or down, or click and then type a numeric value.

> ***TIP:*** *by double-clicking this field you reset all steps to their default value; it is an easy and quick way to undo all changes done so far to the currently selected note attribute.*

The **S** (for **scramble**) button randomly shuffles all current steps; it provides a quick way to try different combinations of the same step values.

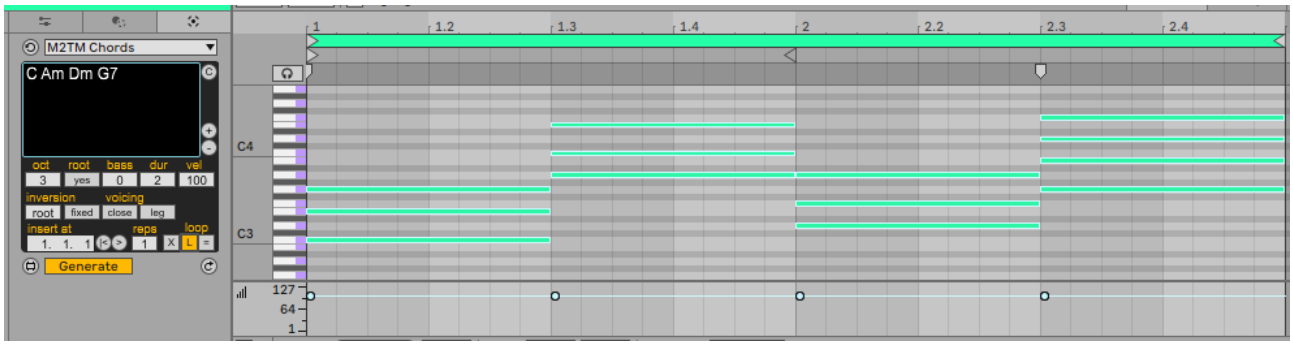
The **R** (for **randomize**) button randomizes step values. If all steps have the same value – for example, because you just selected the note attribute from the top-left menu or double-clicked the bottom-center field – the **randomize** command generates random values across the entire valid range, e.g. between 1 and 127 in the case of **velocity (set)**, or between -2 and +2 in the case of **octave**. However, if steps have different values, the **randomize** command generates values in the range comprised between the *current* minimum and maximum.

Here is a practical example. Say that you want to generate random velocities in the range between 30 and 110. Here is a quick way to do so: (1) use the bottom-center field to assign the 30 value to all steps, (2) use the mouse to increase the first step to 110, (3) click the **R** button.

> ***TIP:*** *to precisely assign a value to the step under you mouse, just check the number that appears in the bottom-right corner of the window.*

Notice that the **randomize** command preserves steps' minimum and maximum, i.e. it generates at least one step whose value is equal to the current minimum, and at least one step whose value is equal to the current maximum. This allows you to repeatedly click the **R** button to generate random values in the same initial range.

M2TM Chords



M2TM Chords is a MIDI tool generator that allows you to generate a complex chord progression by simply typing the names of its chords. It offers many advanced features that make it suitable for expert musicians, yet it can be used by those with little or no music theory knowledge, provided that they have a chord progression to try out.

Feature List

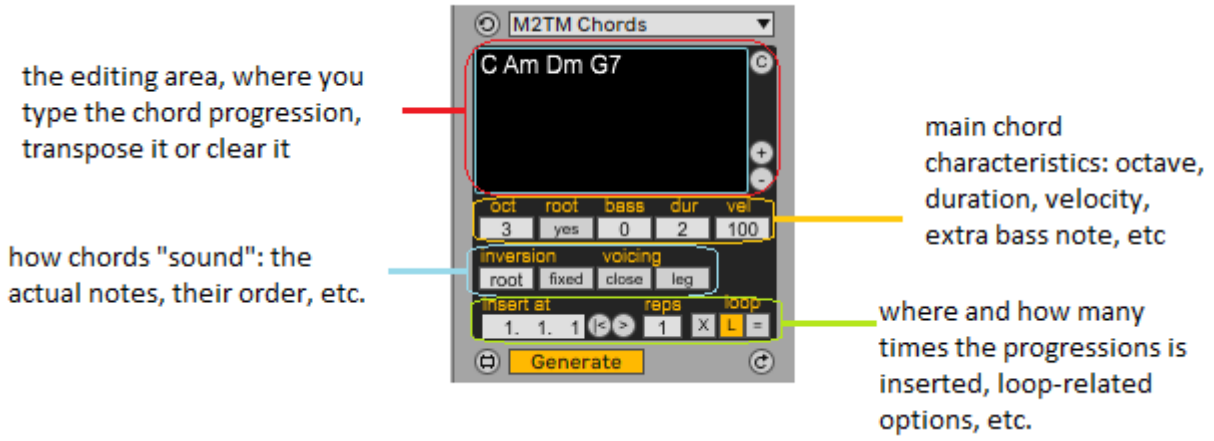
Here is the complete list of features and options:

- **support for virtually any chord type**, from simple triads to 6- and 7-note chords, including quartal, quintal and other non-standard chords;
- **chord synonyms:** recognizes countless name variations – for example, Dmin7, Dm7, Dmi7 and D-7 all refer to the same combination of notes (D F A C);
- **smart parsing of altered chords**: you can type alterations in any order, and separate them with or without the slash character; for example, Csus7b9, Csus7/b9, Csus4/7/b9, C7sus/b9, Csus4/7b9 all refer to the same chords (C F G Bb Db)
- **rootless chords:** optionally drop the root note from the chord; useful if you plan to assign the root note to another instrument
- **drop chords**: in addition to rootless chords, a special extension to the standard syntax allows you to "drop" the 3rd, 4th, 5th and 7th from any chord; for example, Cmaj7drop5 or Cmaj7^5 generate a C major 7th chord without the 5th (or C G B notes)
- **inversions:** you decide whether the lowest note is the root, the 3rd, the 5th or the 7th of the chord
- **auto-inversions:** M2TM Chords can automatically select the inversion that minimize the distance from notes in previous chord, just like skilled keyboard players do
- **slash chords:** enter chords such as Cmaj7/G to indicate the lowest note of the chord – it is another way to specify the inversion, on a chord-by-chord basis
- **transposition:** transpose the typed progression up or down to any of the 12 keys; chord names in the edit window are immediately updated to reflect the new key
- **octave:** select the octave for all chords in the progressions
- **closed/open voicings**: you decide whether chords are played in "closed" position (chord notes are as close as possible, as in the C-E-G triad) or in "open" position (as in the C-G-E triad)
- **automatic legato**: notes in common between consecutive chords can be played in *legato mode*, as a single, longer note – great for pads and string sections!
- **extra base note:** an additional root note can be added one or two octaves below (or above), for a thicker sound; this option is especially effective for simpler triads
- **default or custom chord duration(s):** decide the default duration of chords in the progression, while retaining the ability di specify a different duration for some of the chords

- **repetitions:** the progression can be inserted just one time (default behavior) or be repeated up to 64 times, a feature that is very handy when creating long clips in Arrangement view
- **custom insertion point:** unlike most MIDI Generators for Live, M2TM Chords lets you decide *where* in the clip the notes are added; this feature allows you to quickly create entire songs by appending individual progressions
    - you can specify the insertion point in terms of **bars.beats.16th** (as you do with Live's punch-in and punch-out positions) or – quite opportunely - you can just click a button to move the insertion point immediately after the last progression you inserted
- **add or replace clip notes:** most MIDI generators for Live *add* generated notes to those already in the clip, which is convenient – for example – when you are building a drum pattern one sound at a time; when working with chords, however, you typically want to *replace* existing notes and chords with a new progression, and this option is available in M2TM Chords
- **set or extend loop**: you can set the clip's loop position to match the progression just added, a feature that saves you time when trying out progressions of different length, or when you assign custom durations to each chord
    - you can also *extend* the loop so that it includes the progression just added; this feature is very handy when building a longer progression made of shorter, simpler ones.

The User Interface

M2TM Chords offers many features, thus its window is crowded with controls of all sorts. However, these controls are logically grouped in areas, each with a different purpose:



Let's examine each area in more details.

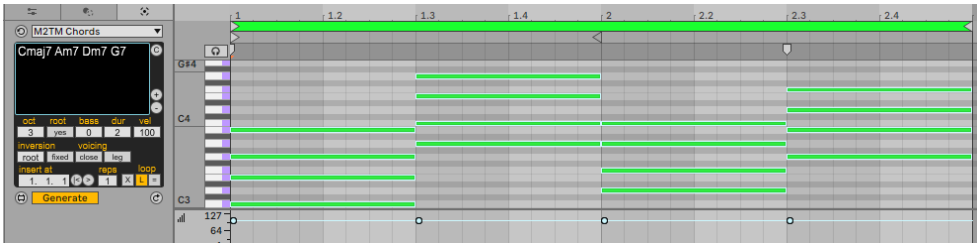
The Chord Progression

The upper half of the window is where you type the chords that make up your progression:

You can type chord names as you would write them down with pen and paper, you just need to leave one spaces (or more) *between* chords and ensure that no space is left *inside* the chord name. For example, here a 4-chord progression in C major:

Cmaj7 Dm7 Am7 G7

Just press the ENTER or TAB key to commit the chords to Live's note editor:



M2TM Chords recognizes virtually **all** chord types – from triads to chords with added 9th, 11th and 13th extensions, in natural or altered (lowered or raised) form, plus non-standard chords consisting of stacked fourths or fifths. Just as important, many syntax forms are accepted, up to ***several thousand syntax variations***. For example, the progression above could be rewritten as:

Cma7 Dmi7 Ami7 G7

or

CM7 D-7 A-7 Gdom7

If a chord name consists of a single note, it is assumed to be a major triad, so for example the C chord is resolved into the C E G notes. For more clarity, you can use the **M, maj** or **ma** suffixes (eg **C CM Cmaj Cma** all indicate the C major triad). You can use **min, mi, m** or – for minor chords (eg **Cmin Cmi Cm C-** are all resolved as the C minor triad). The **+** symbol can be used for augment triads (eg **C+** stands for C E G# notes), the **o** symbol can be used for diminished triads (eg **Co** stands for C E Gb notes); the **sus** suffix indicates a suspended fourth (eg **Csus** stands for C F G notes).

By appending the **6** or **7** digit you can create a 4-note chord. **C6 Cmaj6 CM6** are equivalent spellings for the major 6th chord (C E G A notes); **Cmaj7 CM7 Cma7** stand the major 7th chord (C E G B); **Cmin7 Cm7 Cmi7 C-7** indicate the minor 7th chord (C Eb G Bb); **Cmin6 Cm6 Cm6 C-6** all resolve to the minor 6th chord (C Eb G A). **C7** is the dominant chord (C E G Bb).

You can create many more chords by appending other suffixes. The following table shows ***a small subset*** of all supported chord symbols:

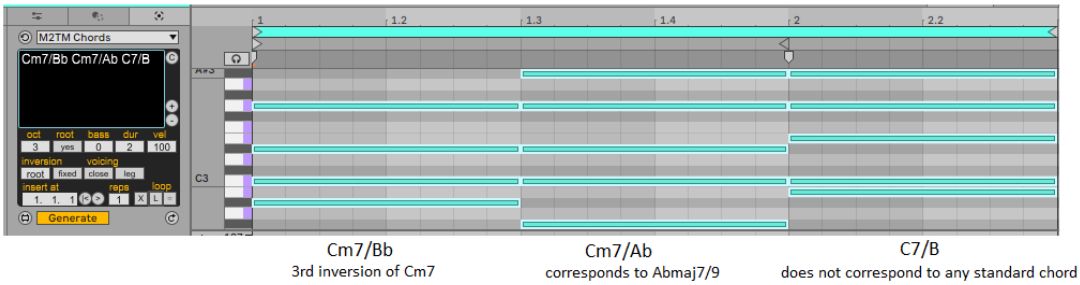
symbol	triad	notes
C Cmaj CM Cma	major triad	C E G
Cmin Cm Cmi C-	minor triad	C Eb G
C+ Caug	augmented triad	C E G#
Co Cdim	diminished triad	C Eb Gb
Csus Csus4 Cadd4	suspended triad	C F G
symbol	4-note chord	notes
Cmaj7 CM7 Cma7	major 7th	C E G B
Cmaj6 C6 CM6 Cma6	major 6th	C E G A
Cmin7 Cm7 C-7 Cmi7	minor 7th	C Eb G Bb
Cmin6 Cm6 C-6 Cmi6	minor 6th	C Eb G A
C7 Cdom7	dominant 7th	C E G Bb
Cm7b5 Cm7/b5 C-7b5	half-diminished	C Eb Gb Bb
Cdim7 Co7	diminished 7th	C Eb Gb A
Cmin/maj7 Cmi/ma7	minor triad + 7th	C Eb G B
C7/#5 C7#5	dominant aug 5th	C E G# Bb

C7/b5 C7b5	dominant dim 5th	C E Gb Ab
C7sus4 Csus7	suspended 7th	C F G Bb
Cadd9	major triad + 9th	C E G D
symbol	5-note chord	notes
Cmaj9 CM9	major 9th	C E G B D
Cmin9 Cm9	minor 9th	C Eb G Bb 9
C9	dominant 9th	C E G Bb D
Cmin/maj9	minor + 7th + 9th	C Eb G B D
Cmaj11 CM11	major + 9th + 11th	C E G B D
C7/#11 C7#11	dominant + #11th	C E G Bb F#
non-standard symbols	chord	notes
Cq Cq3	3 stacked 4ths	C F Bb
Cq4	4 stacked 4ths	C F Bb Eb
Cq5	5 stacked 4ths	C F Bb Eb Ab
CQ CQ3	3 stacked 5ths	C G D
CQ4 CQ4	4 stacked 5ths	C G D A

M2TM Chords also recognizes the so-called *slash chords*, i.e. chord names followed by a slash and the note that works as the bass note. For example, the standard **Cm7** is resolved to C Eb G Bb notes, whereas its slash variation **Cm7/Bb** generates the same notes but in a different sequence Bb C Eb G.

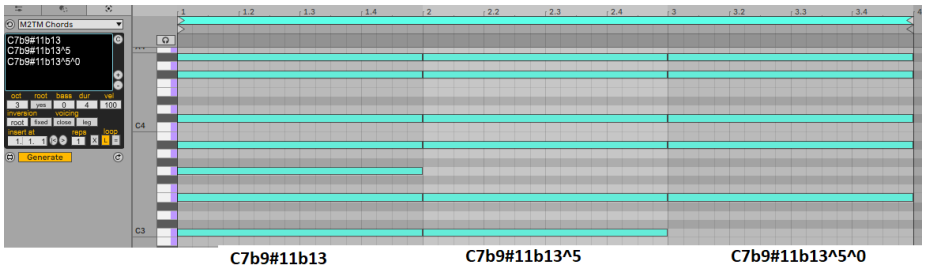
The slash note may or may not belong to the chord. In the former case, the slash notation specifies an inversion of the chord, so in previous example, **Cm7/Bb** is the 3rd inversion of the **Cm7** chord: in this case the Bb note is *moved* from the top to the bottom of the chord. However, if the slash note does ***not*** belong to the chord, it is *added* to regular chord notes, so for example **Cm7/Ab** symbol generates the Ab C Eb G Bb chord.

If the slash not does not belong to the chord, the resulting note sequence may or may not correspond to an existing chord: for example, **Cm7/Ab** is equivalent to **Ab7/9**, and **Dm/B** can be also written as **Bm7b5** (half-diminished chord). However, the **C7/B** chord generates the B C E G Bb notes, a rather dissonant sequence that does not correspond to any standard chord.

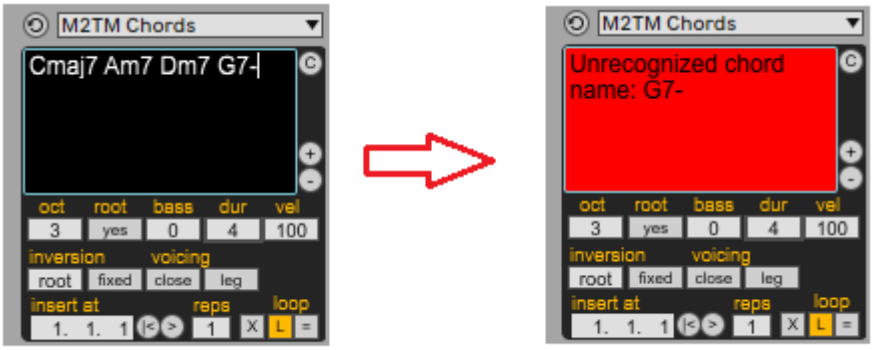


A special syntax – based on the **drop** suffix or the **^** symbol – lets you enter ***drop chords***, i.e. chords that do not contain a note that *should* be there. Drop chords are more useful with very complex chords. For example, the **C7b9#11b13** chord consists of 7 notes – namely C E G Bb Db F# Ab – thus you might want to simplify it while preserving notes that work as upper extensions.

The usual candidate for dropping is the 5th, thus for example **C7b9#11b13drop5** or **C7b9#11b13^5** generate the following 6-note chord: C E Bb Db F# Ab. You can apply the **drop** or **^** suffix multiple times and you can use the 0 or 1 digit to drop the root note, thus for example **C7b9#11b13^5^0** generates this 5-note chord: E Bb Db F# Ab.

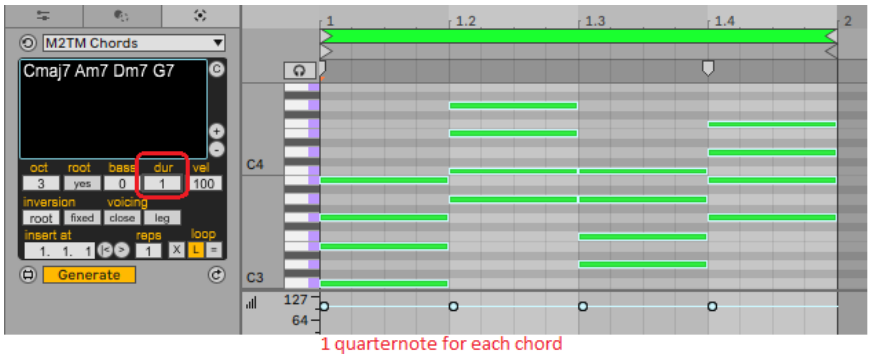


Don’t be afraid to experiment with chord symbols: if **M2TM Chords** fails to recognize one of the chords you type, it displays an error message that explains what symbol didn’t pass the parsing:



G7- isn't a valid chord symbol

By default, the duration of all chords is equal to one bar, or 4 quarternotes in the usual 4/4 time signature. You can set a different duration by changing the **dur** field immediately under the editing pane:

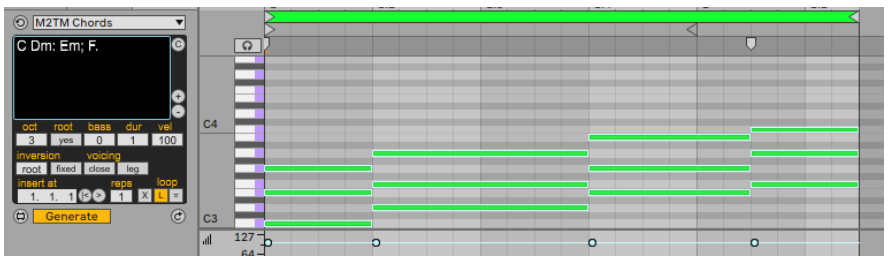


1 quarternote for each chord

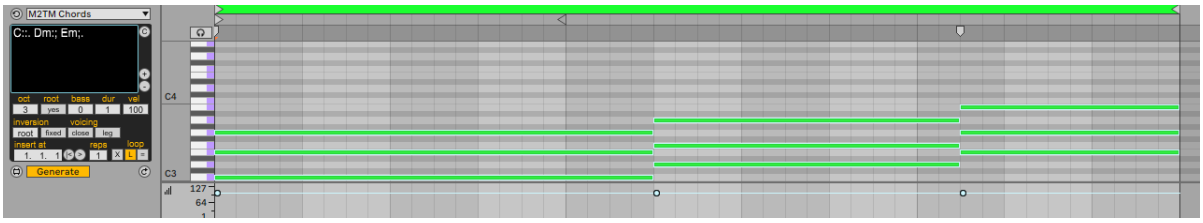
You can also specify a *different* duration for each chord, by appending the following symbols to chord names:

- append a colon symbol “:” for 2 quarternotes
- append a semicolon symbol “;” for 1.5 quarternotes
- append a dot “.” for 1 quarternote

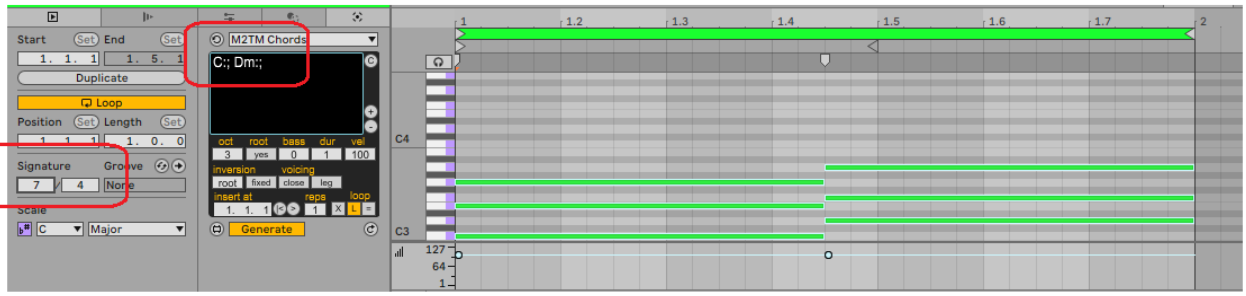
For example, the sequence **C Dm: Em; F.** generates a C major chord for 4 quarternotes (or whatever the default duration indicated by the **dur** field is), a D minor chord for 2 quarternotes, an E minor chord for 1.5 quarternotes, and an F major chord for 1 quarternote:



Duration suffixes can be combined in any order, to generate any duration equal to or greater than 1 quarternote. For example, the sequence **C::. Dm:; Em;.** generates a C major chord for 5 quarternotes, a D minor chord for 3.5 quarternotes, and an E minor chord for 2.5 quarternotes.

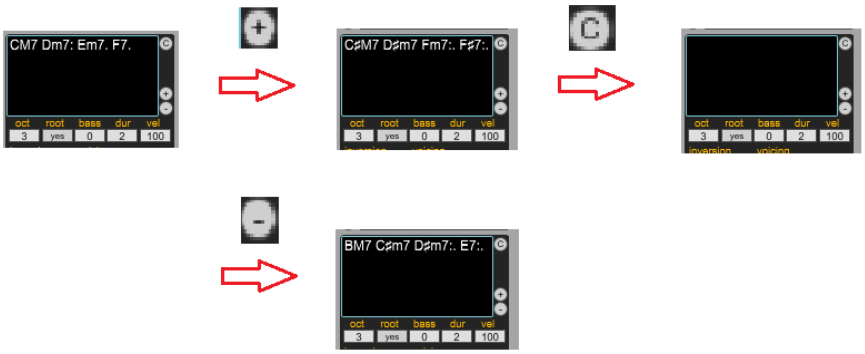


The semicolon ";" symbol is useful for durations that contain half quarternotes, for example to insert two chords of equal duration in a 3/4 or 5/4 bar. The figure below shows how to fit two chords in a 7/4 bar:



NOTE: you cannot specify durations shorter than 1 quarternote.

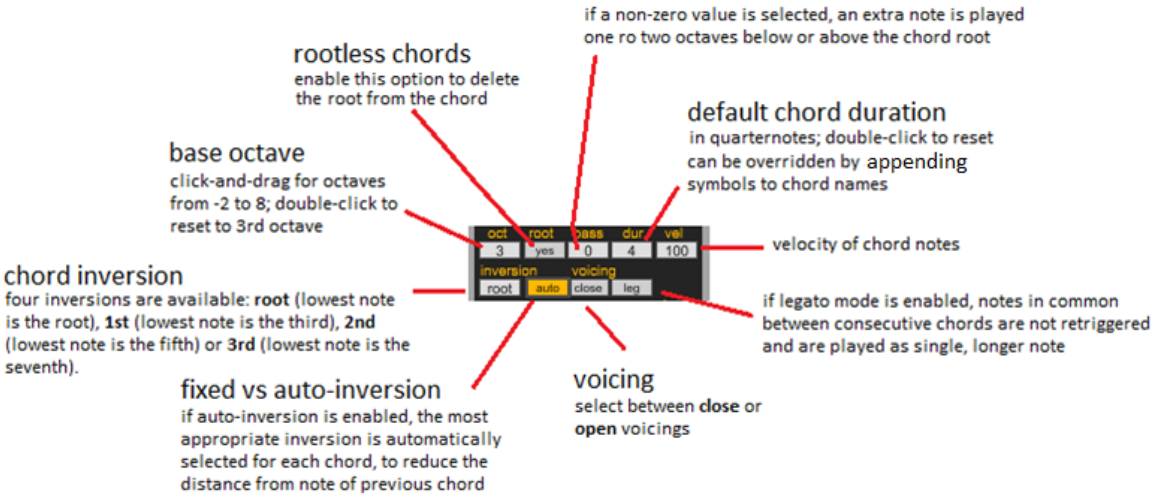
M2TM Chords lets you transpose your progressions up or down by semitones using the **+** and **–** buttons to the right of the editing pane, and you can quickly delete the current progression by clicking the **C** button. The text in the editing pane always reflect the updated progressions and chord durations are preserved:



Please be aware that *up/down buttons do not commit the current progression* to the note editor; instead, they translate the last progression that was processed (and found correct). This implies that you should always type the ENTER or TAB key before clicking the up/down buttons.

Chord Attributes

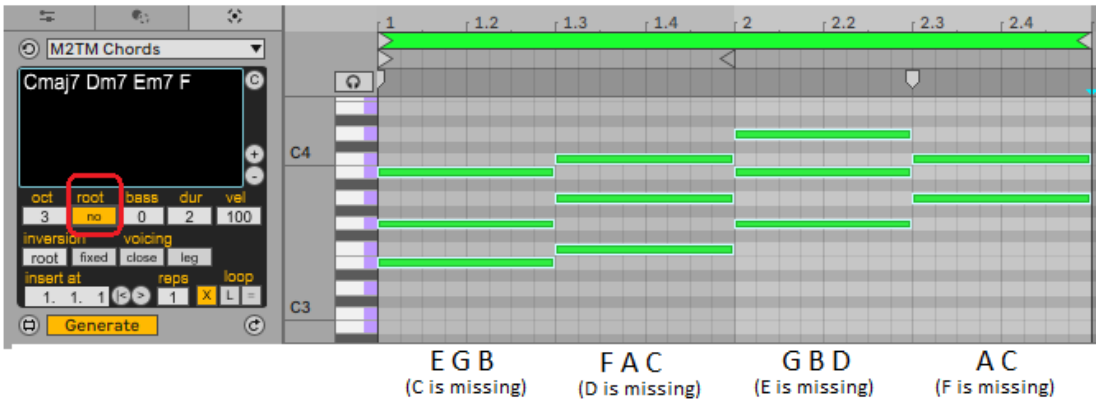
The next two horizontal sections let you define the octave, duration, velocity and other characteristics that affect how the chord "sounds":



The **oct** field is the base octave for the chord: the default value of 3 means that chords will extend from C3 (middle C) upward.

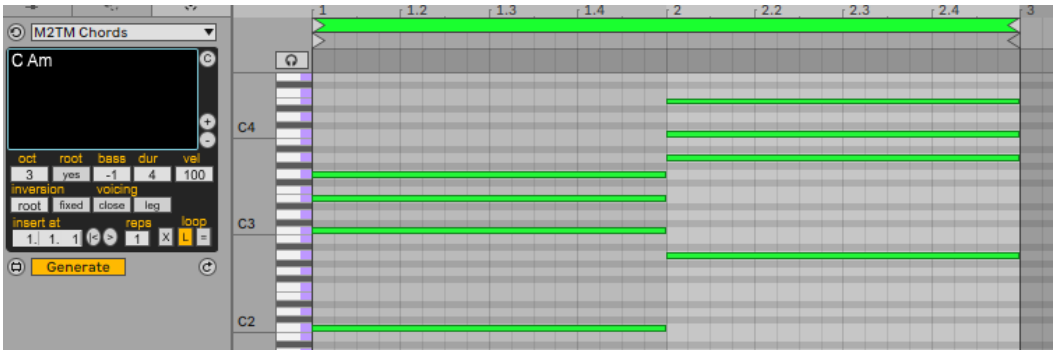
The **root** option lets you generate *rootless chords,* i.e. chords that lack their root note. For example, a rootless **Cmaj7** chord would contain the following notes: E G B. While omitting the root note might sound weird or useless, you might have good reasons to do so, for example when you want to assign the root to another instrument (in a different Live track).

> *Rootless chords are quite popular among jazz musicians, and it often happens that the piano player omits the root because the bass player will be playing it.*



The **root** option is useful when *all* the chords in the progression should be played as rootless chords; if only a few chords should be played without the root, you can use the **drop** or **^** suffixes in the chord symbol, as explained in previous section

You can create a “thicker” sound by doubling the lowest chord note one or two octaves below or above, using the **bass** field. (The lowest note is usually the chord root, but it can be a different note if you selected a different inversion.)



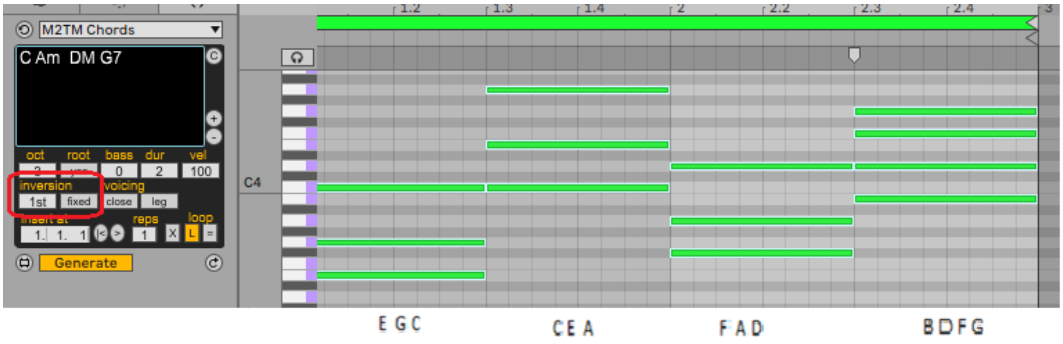
As explained in a previous section, the **dur** field affects the duration of all chords whose duration is not specified by means of dots, colons, and semicolon suffixes. You can double-click this field to reset it to the numerator digit in current time signature, for example 4 in the canonical 4/4 tempo, or 7 in the less common 7/8 time signature.

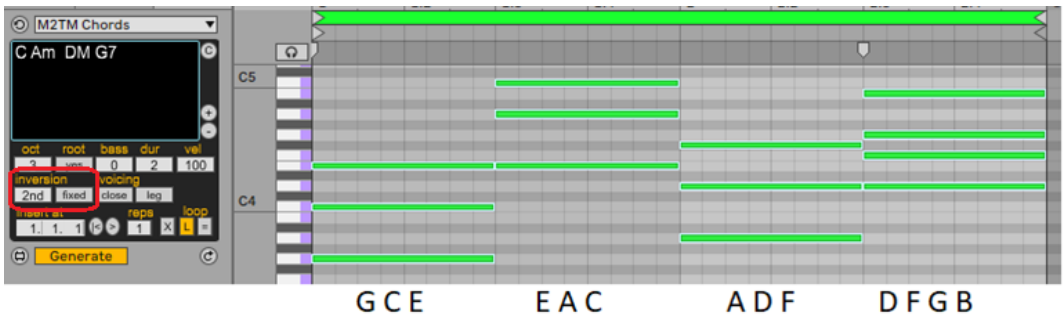
As its name suggests, the **vel** field is the MIDI velocity of all notes in the chord, including the bass note if there. (Its default value is 100.) In real-world scenarios, you might want to "humanize" the result by randomly changing this value using the Live's Velocity Shaper built-in MIDI tool or our **M2TM Steps**.

Inversions and Voicings

A single chord symbol actually corresponds to dozens chords that have the same "nature" and yet a different "sound". In fact, in addition to playing the chord in different octaves, you can also choose between different *inversions* and *voicings*.

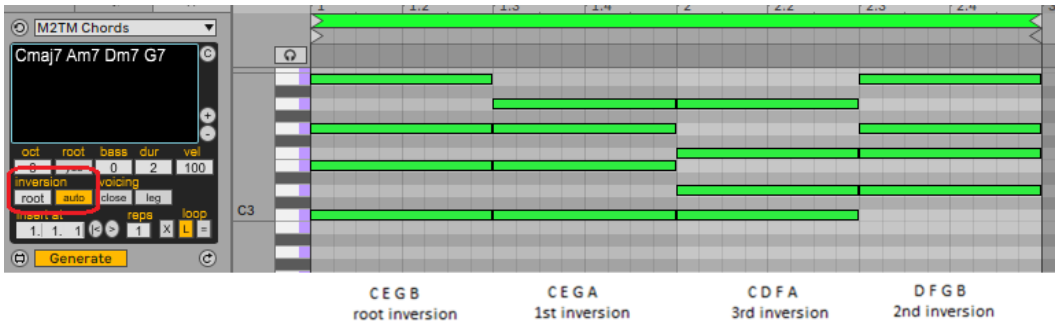
While beginners usually play chords in the so-called *root position* (i.e. C-E-G-Bb for C7), more expert musicians may opt for the *1st inversion* (lowest note is the third, or E-G-Bb-C in previous example), the *2nd inversion* (lowest note is the fifth, or G-Bb-C-E) or even the *3rd inversion* (lowest note is the seventh, or Bb-C-E-G). Here are a couple examples:



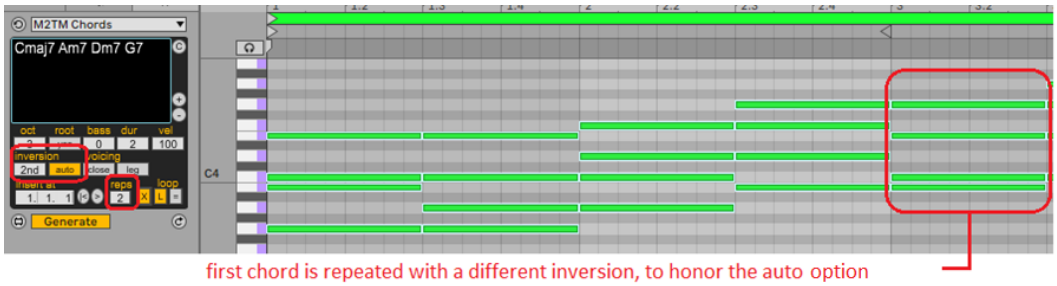


As you surely noticed, in all examples seen so far, chords move in parallel, and the same inversion is used for all chords in the progression. This is rarely what a seasoned keyboard player would do.

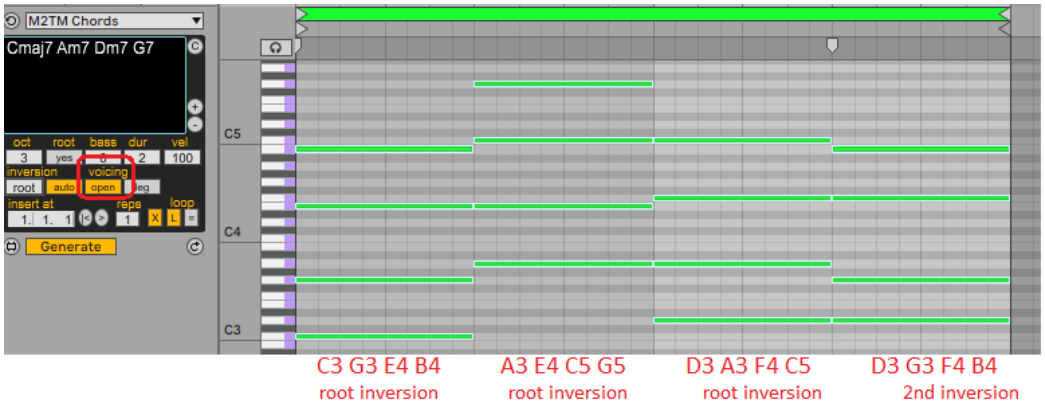
A unique – and very powerful – feature of **M2TM Chords** is the ability to automatically select the optimal inversion for each chord after the first one. In this context, the *optimal inversion* is the one that preserves the pitch of notes that are in common between consecutive chords or that, at least, minimize the distance between them. You enable this option by turning the **fixed** field into **auto**:



Interestingly, the **auto** feature works correctly even if the progression is repeated two or more times, using the **reps** field:

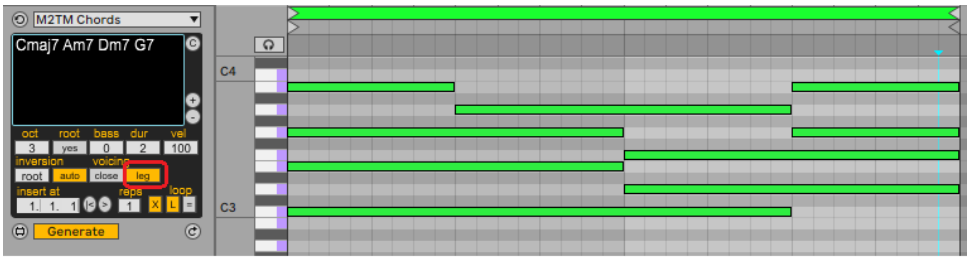


In addition to inversions, you can vary the sound of chord by opting for an *open voicing* that uses larger intervals between chord notes. You achieve this result by means of the **voicing** option, optionally combining it with the **inversion** field and the **auto** feature.



As you see in the figure above, when using the auto feature you may get different inversions for the same progression, depending on whether you select close or open voicing.

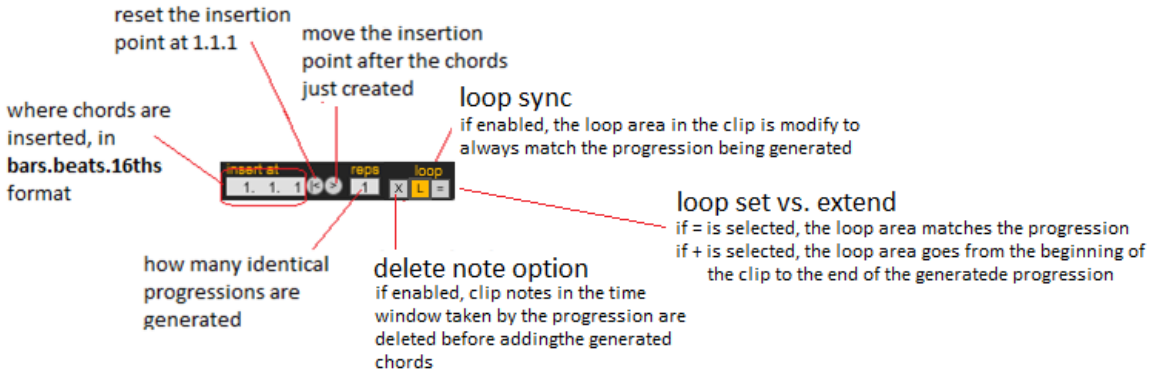
If two consecutive chords have one or more notes in common, a keyboard player might decide not to re-trigger the note on the latter chord. This is especially effective when creating background pads, or mimicking a string section. You enable this behavior using the **leg** (for **legato**) option:



The **legato** option works as expected even when the progression is repeated two or more times:

Insertion Point and Loop Handling

The fields near the bottom of the window allow you to define where chords are added and how many progressions are created, and to perform a few useful actions related to the loop area:

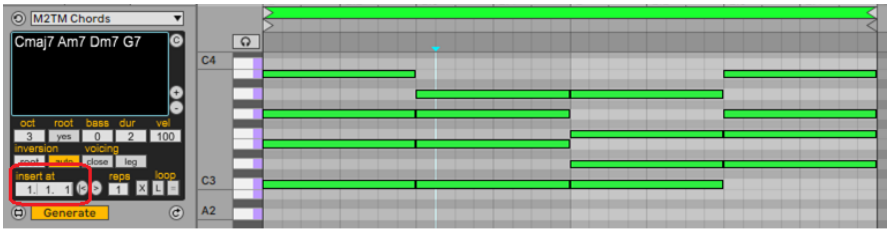


By default, **M2TM Chords** creates notes at the very beginning of the clip, corresponding to position **1.1.1** when expressed in the **bars.beats.16ths** format. Unlike most other MIDI Generators, however, you can change this insertion point by means of the first three fields in this area.

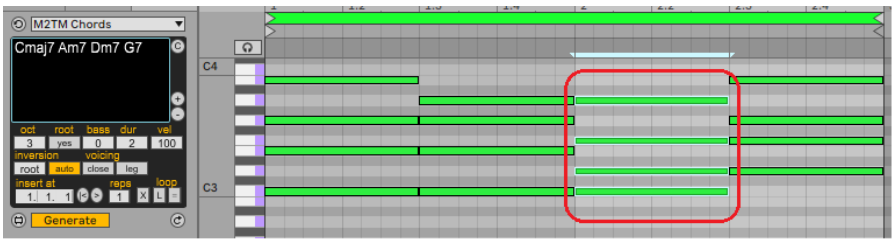
While it is possible to manually edit the numbers in these fields, you will typically change the insertion point by means of the two buttons to their rights: the **|<** button resets the insertion point to 1.1.1, whereas the **>** button moves the insertion point *immediately after* the chords you have just inserted. This feature allows you to create complex progressions that consists of two or more simpler progressions, or maybe the same progression with different root, voicings, etc.

Notice that for these buttons to work correctly you need to confirm the insertion of the current progression, according to the rules of all MIDI Generators for Ableton Live. In practice, this is the correct workflow:

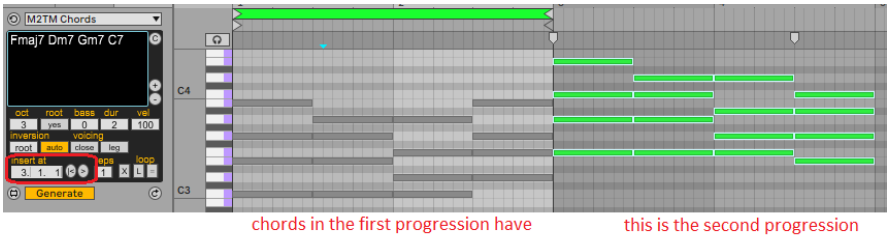
First, enter the first progression, as you would do normally. If the insertion point is other than 1.1.1, click on the **|<** button to reset to the beginning of the clip.



Next, confirm your decision to generate current chords, by using one of the methods Ableton Live offers; for example, use the mouse to select some notes in the MIDI editor.



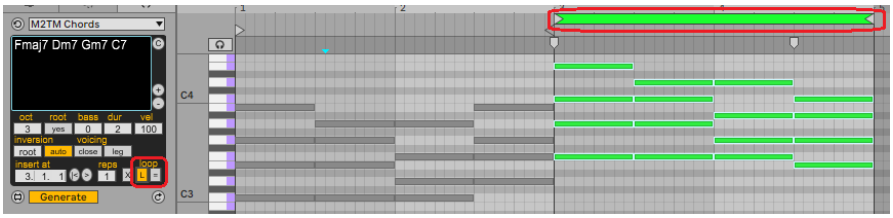
Click on the **>** button to move the insertion point after the chord you just inserted and enter a different progression (or just transpose the current one).



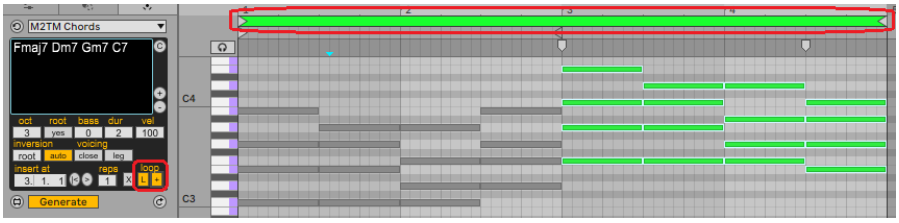
chords in the first progression have been committed to the clip

this is the second progression

Click the **L** button and ensure that the button to its right shows the **=** (equal) symbol, so that the loop area ***moves*** to match the progression you just added.



If you are satisfied with the new progression, you can click on the = button to turn it into the + (plus) symbol, which ***extends*** the loop area to include the new progression. You can now hear how the two progressions sound together.

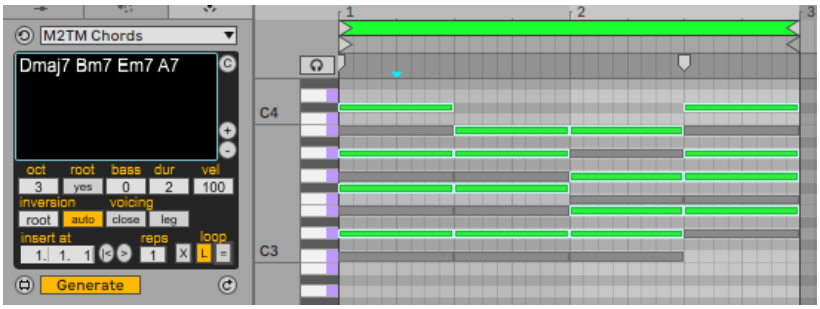


> ***NOTE**: changing the insertion point, either directly or by means of the **|<** and **>** buttons does not have immediate effect, and the new point is used for the **next** progression you define. If you want to apply them immediately, just force M2TM Chords to re-generate the current progression, for example by clicking twice the **auto** button.*

When experimenting with chord progressions you can often change your mind and need to restart the entire process, or just change a sub-progression somewhere in the clip. As you may already know, by default MIDI Generators for Ableton Live always ***add*** notes to those already in the clip, which makes sense in most scenarios, for example, when you create a drum loop and need to define the pattern for the hi-hat without deleting the patterns you already created for the kick and snare.

When working with chords, however, the chords you generate somewhere in the clip are typically meant to ***replace*** the notes in that area, and you would be forced to temporarily switch from **M2TM Chords** to the

clip editor. Unfortunately, this very operation has the effect to ***commit*** the chords you are currently working on, which is rarely what you would do.



In this scenario, the **X** option is very useful, because it automatically deletes all the notes that are currently in the area where the progression is about to be inserted.