# GENERATING EXPRESSIVE PERFORMANCE

**Romane Clerc**[1] **Octavio Profeta**[2] **Cindy Tang**[1] **Shu Yang**[1]
[1] Digital Humanities, EPFL
[2] Robotics, EPFL

## ABSTRACT

Expression lies at the heart of music performance, infusing the listening experience with depth, vitality, and emotions. In this study, we undertake the creation of a model designed to translate a raw MIDI file into an artificial performance that reflects the nuanced expressiveness of a live rendition of Mozart's *Piano Sonata n°8 I*. Our model encompasses various elements, including rhythm, musical phrasing, dynamics, and articulation. Finally, our generated piece exhibits less robotic sounds, with a notable enhancement in the clarity of dynamic variations between phrases.

## 1. INTRODUCTION

Expression lies at the heart of music performance, infusing the listening experience with depth, vitality, and emotions. It can include subtle deviations in timing, dynamics, pitches or timbres, that distinguish one rendition of a composition from another. Building upon insights gleaned from previous studies analysing [1–4] and modelling [5] the components contributing to expressive rendition, we embark on a parallel journey. Our study aims to firstly understand the elements that makes a performance expressive, and secondly, to computationally model these aspects in order to generate an artificial expressive performance from a raw MIDI file. For this purpose, we choose to focus on Mozart's *Piano Sonata n°8 I*, from the Aligned Scores and Performances (ASAP) dataset [1], and to use the Jia's performed version as a reference to evaluate our model.

## 2. LISTENING AND ANALYZING

After comparing performed and unperformed versions, we have identified the following key differences, and categorized them for clarity:

### 2.1 Rhythm

Our findings primarily concern the aspect of rhythm. We are pleased to discover that the musical phrase is a vital concept where musical pieces are structured in relation to velocity. We noticed that the velocity in phrases follows a mountain-like shape: starting slowly, speeding up in the middle, and then slowing down (ritardando) towards the end. The Inter-Onset Interval (IOI) contrasts are reduced, especially where two adjacent notes originally have a 2:1 nominal IOI ratio, and the IOI is shortened in sections where the melody ascends. Additionally, longer notes tend to be extended while shorter notes are shortened in the performed version. Interestingly, in ascending leaps, the first note is shortened while in descending leaps, it's lengthened. Also, final chords in the measures are played more slowly.

### 2.2 Musical Phrases

In particular, listening to the performed and unperformed piece one after the other, it appears very clearly that the phrases are not emphasized in the same way. The sound level is constant within each phrase in the unperformed version of the piece, which accentuates the changes of volume between phrases, producing a inhuman sound. The phrases are actually beginning and ending quieter than in the middle, creating a mountain-like shape.

### 2.3 Dynamics

Dynamics is another key factor that turns musical pieces into something more vibrant. We found that the volume of each phrase matches its tempo, with slower tempos paired with softer volumes. Also, in the left hand, particularly during repeated chords, the playing is staccato and softer, especially noticeable at the beginning of the piece. Moreover, when the right hand plays a sequence like BbGED, the Bb is emphasized more than the others, which sometimes also applies to the left hand.

### 2.4 Articulation

Articulation is essential in expressing performer's intentions when playing. We found that when a measure contains only chords, they are typically arpeggiated to enhance their texture. Sometimes, the trills are not played as expected, deviating from the traditional execution. Additionally, there are added and elongated rests that introduce pauses, maintaining suspense within the piece. At the beginning, when the right hand plays arpeggios, the notes on the half beats are articulated staccato; this articulation eventually extends to the following notes.

[1] https://github.com/fosfrancesco/asap-dataset

## 3. GENERATING EXPRESSIVE MIDI

In order to generate our music with unique expressiveness, we select insights that are general and have the potential for application across various pieces. Therefore, not every discovery will be mathematically incorporated as a feature in our models. For specifics on which differences will be retained, please refer to the following subsections.

The main toolkit employed is music21, a Python-based toolkit designed for computer-aided musicology, to extract essential features from the dataset. These features include chords, pitches, tempo, velocity, offsets, duration and others. All the code can be accessed on GitHub [2] .

### 3.1 Rhythm

To generate an expressive performance, we have to get rid of all the robotic feelings we get while listening to a unperformed MIDI file. As mentioned before in the analysis, for two successive notes with a nominal IOI ratio of 2:1, for example a quarter note followed by a eighth note, the IOI contrast will be slightly reduced. This is achieved by slightly shifting the onset second note, allowing it to happen a tad bit sooner than programmed, giving an impression of "rush and wait". This is done in the `decrease_ioi_contrast` function, which simply iterates through all events in a part; for two consecutive sounded events, we simply check the ratio of their duration length, and in the case of a 2:1 ratio, we shift the onset of the second note by a value $x$, which in our case is equal to one forth of a beat, i.e. $x = 0.25$.

As also mentioned in the analysis, longer notes are lengthened and shorter notes are shortened in a performance. However, this is just so subtle when done by an actual human, and we did not manage to recreate it with our toolkit. We decided in the end not to implement this generation into our performance.

### 3.2 Musical Phrases

If we draw our attention to the volumes of the notes, we find out that they change quite brutally. Reading literature on the subject, we learned that within a phrase of a performance, the loudness tend to follow a mountain-like shape, being louder in the middle of the phrase [6]. The performer is doing a *crescendo* until the middle of the phrase where *decrescendo* begins. These fluctuations of loudness are continuous, in a sort of wave pattern, which creates a natural feeling, like long breaths throughout the piece. Listening to the unperformed midi file, we noticed that the different musical phrases were already separated by velocity. In fact, the whole piece is alternating between *forte* and *piano* phrases.

To create a wave effect in loudness within the phrases of the piece, we created the functions `adjust_velocity` and `adjust_phrase_velocities`. As the phrases are distinguishable by changes in velocity, separating them in the function `adjust_velocity` is pretty easy: we check the velocity for each note and if it is different than the last one's, we consider this note and the followings as a new phrase. Phrase's velocities are then modified in `adjust_phrase_velocities`. The function increases the velocity of a note according to its position in the phrase. In this way, the velocity is following a way more natural curve, which participate greatly to the living impression of the performance.

### 3.3 Dynamics

Drawing from our study results and piano expertise, we emphasizes two main reasons for the right hand's (RH) heightened focus: its role in carrying the melody, which is typically more pronounced in the upper register of multi-voice compositions, and the greater perceptual impact of RH errors on listeners [7]. So based on the research and what we heard from the performed version, in the function `decrease_left_hand_dynamics`, we aimed to selectively reduce the dynamics of notes and chords in the left-hand part. It applies a designated decrease to the velocity of single notes and a more pronounced reduction to the velocity of notes within chords. By applying this, the left hand plays more softly, enhancing musical expression and contrast.

Also, it is evident that there are several instances where the left hand consistently plays the same chord within one measure, particularly notable at the beginning of the piece. Through literature review, we identified that such a playing pattern could be categorized as an *ostinato* or chord progression. The term *ostinato* is used in this discussion refers to a short melody or a motive which is normally in the lowest voice and is repeated several times in succession, one characteristics of this type of *ostinato* is the melody is sometimes ornamented during the course of its repetition [8]. Another effect we assume this pattern would serve is as a percussive function, enhancing the rhythmic quality of the music. Drew inspiration from our performed piece, we find the player tends to render these sequences with a detached articulation, adding breaks between chords to create a *staccato*-like effect. Moreover, we think a variance in dynamics between the same chords is natural because it is quite challenging and almost impossible to strike all chords with the exact same volume, and this slight unpredictability adds a layer authenticity to the performance. And such difference also makes the chords, which might originally like a purely metronomic function, less monotonous and more dynamic. Consequently, in our function `modulate_left_hand_sequence_chords`, we added a *staccato* effect to mimic this detached playing style. Additionally, we randomize the dynamics to each chord within a defined range, to create a more percussive and dynamic feel.

[2] `https://github.com/cindytangch/generating-expressive-performance`

### 3.4 Articulation

In our pursuit to enhance the expressiveness of MIDI performances, we have developed two functions about articulation: `add_arpeggios` and `custom_trills`.

The `add_arpeggios` function is designed to infuse *arpeggio* into measures exclusively composed of chords, mirroring the cascading effect often encountered in live musical renditions. This function iterates through each measure within the piece, identifying those comprised solely of chords, and calculates the offset for each note within these chords to introduce the desired *arpeggio* effect. The spacing between arpeggiated notes is determined by dividing the duration of the chord by the number of notes it contains raised to a power parameter that we can adjust. Manipulating this power parameter enables us to adjust the intensity of the *arpeggio* effect: increasing the power decreases the offsets between notes, resulting in a reduction of the *arpeggio* effect.

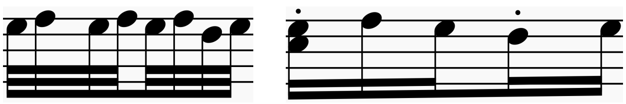

**Figure 1**. Comparison of Trills.
On the left, trills as depicted in the musical partition.
On the right, trills as performed in Jia's performance.

Furthermore, our observations have revealed variations in the performer's rendition of the "trills" structure, as illustrated in Figure 1. In response, the `custom_trills` function endeavors to emulate these nuanced trills. Upon studying the MIDI file encoding of this structure, we noticed the following representations for these trills:

```
{3.0} <music21.chord.Chord E5 F5 E5>
{3.3333} <music21.chord.Chord F5 E5 F5>
{3.6667} <music21.note.Rest 1/12ql>
{3.75} <music21.chord.Chord D5 E5>
```

Therefore, as an attempt to mirror the performer's approach, we model a function to identify chords containing precisely three notes, examine the pitch relationships among these notes, and remove the last note of the chord if it is redundant.

## 4. REFLECTING ON THE GENERATED MIDI

In the analysis of the differences between our exported MIDI file and the original, we can clearly feel that our music is less robotic. A major change is that we can more clearly hear the differences in dynamics between phrases, making it easier to distinguish when the music moves to the next phrase. Also, within the phrases, the gradual increase and decrease in volume helps to emotionally engage us, making the listening experience better connecting music with listeners. Additionally, other adjustments we made, such as *arpeggios*, trills, and some *staccato* settings for the left hand, can also be perceived when we listen carefully. What's more, the main melody in the right hand also feels distinct compared to the original one. Overall, we are satisfied with our work.

We believe that the most vital aspect of musical expressiveness is its ability to vividly evoke human emotions. Moreover, an important fundamental point is that computational changes do not necessarily linearly affect human cognitive changes. As the Weber-Fechner Law indicates, the perceived intensity of stimuli by humans is logarithmic. This means that the changes in perception are not directly proportional to the physical changes in stimuli, but rather related to the proportional changes in stimuli [9]. We believe this law is also applicable to auditory perception. Therefore, in order to make our music more human-like in the future, that is, more capable of moving emotions, we need to make more pronounced changes in computation. When we are considering future plans, one possibility that emerged is to computationally represent the small imperfections and irregularities that occur in human playing, which might be key to making performances more human-like. It may be the truth that imperfection is human's innate nature.

## 5. DISCUSSION

After getting familiar with the tools for computationally analyzing music in the previous study, the present investigation enables us to expand our exploration into the hierarchical structure of musical compositions. This encompasses elements such as phrases, measures, chords, and notes. Furthermore, we have the opportunity to delve into a broader range of musical characteristics beyond expressive timing, including pitch and dynamics.

However, our exploration has not been devoid of challenges. We mainly encountered difficulties when switching between MuseScore and music21. Discrepancies occasionally surfaced between MIDI files loaded in MuseScore and those processed with music21, causing confusion regarding the correspondence of parts and impeding comparisons. For instance, in the trills depicted in Figure 1, we struggled to comprehend why they were encoded as chords while they are not played simultaneously. Additionally, some discrepancies in clef types can be displayed in MuseScore, but not in music21. Moreover, grasping the method by which music21 extracts left-hand and right-hand performances has also presented a hurdle.

Additionally, one issue we encountered while organizing our respective functions was that the structure of the exported music files was not consistent. We spent some time harmonizing our models, and this issue will be addressed in our future group projects.

## 6. CONCLUSION

The combination of all of our generative functions gives us a generated expressive performance; compared to an unperformed piece, our result feels more "human", which was the objective of this research.

This second project was a deeper dive into the crafts of generative performance. From rhythm modification, which we firstly explored in the first project, to physical dynamics of the hands, we saw that generating an expressive performance is not easy task. It implies listening, reflection and attention to the most minute details of a performance. But the resulting performance rewards us with a feeling that, us too, are kind of musicians ourselves.

## 7. AUTHOR CONTRIBUTIONS

Initially, we collectively listened to, analyzed, and compared both the unperformed and performed versions of Mozart's *Piano Sonata n°8 I*. Then, the workload was distributed among the team members: the analysis and generation of rhythm was made by O. Profeta, musical phrases by R. Clerc, dynamics by S. Yang and the articulation by C. Tang. Each member wrote their respective sections of the report. Subsequently, all team members actively engaged in reviewing, providing feedback, and refining each section. This collaborative approach enabled each section to benefit from the diverse expertise of each team member.

## 8. REFERENCES


[1] A. Lerch, C. Arthur, A. Pati, and S. Gururani, "An interdisciplinary review of music performance analysis," *arXiv preprint arXiv:2104.09018*, 2021.

[2] R. Bresin and A. Friberg, "Emotional coloring of computer-controlled music performances," *Computer Music Journal*, vol. 24, no. 4, pp. 44–63, 2000.

[3] R. Bresin and G. Umberto Battel, "Articulation strategies in expressive piano performance analysis of legato, staccato, and repeated notes in performances of the andante movement of mozart's sonata in g major (k 545)," *Journal of New Music Research*, vol. 29, no. 3, pp. 211–224, 2000.

[4] J. Sundberg, A. Friberg, and R. Bresin, "Musician's and computer's tone inter-onset-interval in mozart's piano sonata k 332, 2nd mvt, bar 1-20," *Quarterly progress and status report, TMH, Speech, Music and Hearing*, 2003.

[5] G. D. Poli, "Methodologies for expressiveness modelling of and for music performance," *Journal of New Music Research*, vol. 33, no. 3, pp. 189–202, 2004.

[6] J. Langner and W. Goebl, "Visualizing expressive performance in tempo—loudness space," *Computer Music Journal*, vol. 27, no. 4, p. 69–83, Dec 2003.

[7] R. Parncutt, J. A. Sloboda, and E. F. Clarke, "Interdependence of right and left hands in sight-read, written, and rehearsed fingerings of parallel melodic piano music," *Australian Journal of Psychology*, vol. 51, no. 3, pp. 204–210, 1999.

[8] W. Logan, "The ostinato idea in black improvised music: A preliminary investigation," *The Black Perspective in Music*, pp. 193–215, 1984.

[9] S. Dehaene, "The neural basis of the weber–fechner law: a logarithmic mental number line," *Trends in cognitive sciences*, vol. 7, no. 4, pp. 145–147, 2003.



The writing of this work was assisted by OpenAI's ChatGPT (version 3.5) for text reformulations and grammar checks. Retrieved from `https://www.openai.com/chatgpt`.