A guide to understand Stepmania's gimmicks

Pedro G. Bascoy
pepo_gonba@hotmail.com

March 2, 2023

1 From song time to sequencer time

1.1 Stepmania definition

The SSC file provides two methods of artificially pausing the note scrolling when the song is playing. Those gimmicks are referred as #STOPS and #DELAYS. Although there is a subtle difference between them, they are essentially identical w.r.t. the effect that produces.

One STOP definition might look as follows:

```
#STOPS: 4=5,6=2;
```

Let us convert the definition into a JSON-like structure:

```
{
  [
    beat: 4,
    stop: 5
  ],
  [
    beat: 6,
    stop: 2
  ]
}
```

This #STOPS definition is telling us a couple of things.

1. At beat 4, stop the scrolling for 5 seconds. Then resume.
2. At beat 6, stop the scrolling for 2 seconds. Then resume.

Similarly, one DELAY defition could look like this:

```
#DELAYS: 4=5,6=2;
```

And after converting it into the friendly structure:

```
{
  [
    beat: 4,
    delay: 5
  ],
  [
    beat: 6,
    delay: 2
  ]
}
```

The `#DELAYS` definition above is telling us essentially the same story as the STOPS definition shown before, i.e.:

1. At beat 4, stop the scrolling for 5 seconds. Then resume.
2. At beat 6, stop the scrolling for 2 seconds. Then resume.

The difference between these to is that notes that lie exactly at the stop/delay beat, will need to be tapped before and after the waiting time for the stops and delays, respectively.

1.2 Challenge

We would like to have a pair of functions $t_{(s)}, t_{(d)} : \mathbb{R} \to \mathbb{R}$ that would retrieve the song time after stops and delays (sequencer time) from the song time. Additionally, we would like to have two inverse functions of $t_{(s)}, t_{(d)}$, $t_{(s)}^{-1}$ for STOPS and $t_{(d)}^{-1}$ for DELAYS so we are able to map from the sequencer time into the song time.

1.3 Solution

To do so, let us imagine that we have a function $f : \mathbb{R} \to \mathbb{R}$ that given a beat, it calculates the song time. To simplyfy things further, imagine that this song is has constant BPM of 60, so each second is worth 1 beat. Then $f(b) = b$, being b a beat.

Having this in mind, then we can write the piecewise functions

$$t_{(s)}(x) = t_{(d)}(x) = \begin{cases} x\,, & \text{if } x \leq f(4)\,; \\ 4\,, & \text{if } 4 < x \leq 4+5\,; \\ x-5\,, & \text{if } 4+5 < x \leq 6+5\,; \\ 6+5\,, & \text{if } 6+5 < x \leq 6+5+2\,; \\ x-2-5\,, & \text{if } x > 6+5+2\,; \end{cases} \tag{1}$$

that will map from song time to sequencer time for the STOPS and DELAYS, respectively. A plot of this function can be see in Figure 1.

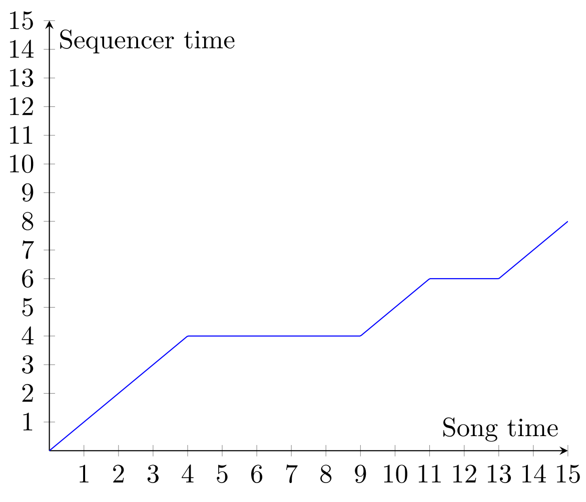

Figure 1: Plot of $t_{(d)}$

We can also easily calculate the two different inverse functions to model when notes at stops or delays should be tapped w.r.t. the song time. On the one hand side, we write the function

$$t_{(s)}^{-1}(x) = \begin{cases} x\,, & \text{if } x \leq 4\,; \\ x+5\,, & \text{if } 4 < x \leq 6\,; \\ x+5+2\,, & \text{if } x > 6\,; \end{cases} \tag{2}$$

to map from the sequencer time into song time for STOPS. Similarly, the function

$$t_{(d)}^{-1}(x) = \begin{cases} x\,, & \text{if } x < 4\,; \\ x + 5\,, & \text{if } 4 \le x < 6\,; \\ x + 5 + 2\,, & \text{if } x \ge 6\,; \end{cases} \tag{3}$$

will map from the sequencer time into the song time for DELAYS. Note that the signs at the conditions are slightly different.

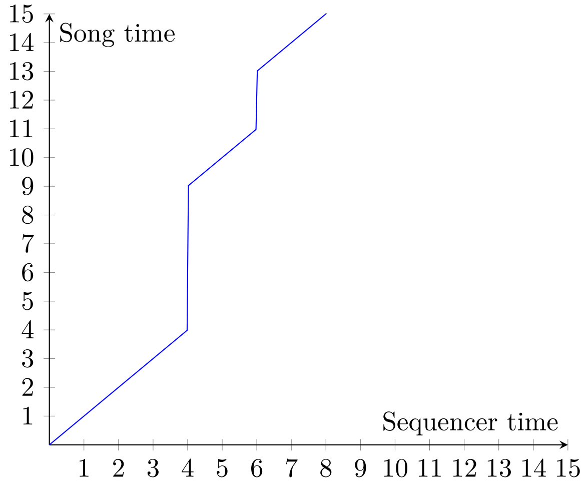

Figure 2: Plot of t^{-1}

1.4 Formalization

Let $\mathcal{T} = \left\{\left(b_i^{(t)}, r_i\right)\right\}_{i=1}^{n}$ be a sequence of STOPS or DELAYS, where r_i is the stop or delay(measured in seconds) at beat $b_i^{(t)}$. Let $f : \mathbb{R} \to \mathbb{R}$ be a function that retrieves the second from the start of the song given a current beat.

We define a new set

$$\mathcal{T}' = \{(c_i, r_i)\}_{i=1}^{n} = \left\{\left(f\left(b_i^{(t)}\right), r_i\right)\right\}_{i=1}^{n} \tag{4}$$

where c_i is the second from the start of the song of beat $b_i^{(t)}$. We define the functions $t_{(s)}, t_{(d)} : \mathbb{R} \to \mathbb{R}$

$$t_{(s)}(x) = t_{(d)}(x) = \begin{cases} x - \displaystyle\sum_{j=1}^{i-1} r_j\,, & \text{if } c_{i-1} + \displaystyle\sum_{j=1}^{i-1} r_j < x \le c_i + \displaystyle\sum_{j=1}^{i-1} r_j\,, \quad \forall i = 1, \ldots, n\,; \\ c_i\,, & \text{if } c_i + \displaystyle\sum_{j=1}^{i-1} r_j < x \le c_i + \displaystyle\sum_{j=1}^{i} r_j\,, \quad \forall i = 1, \ldots, n\,; \\ x - \displaystyle\sum_{j=i}^{n} r_j & \text{if } x > c_n + \displaystyle\sum_{j=1}^{n} r_j\,; \end{cases} \tag{5}$$

when $\mathcal{T}' \neq \emptyset$, where $c_0 := -\infty$, that maps from the song time into the sequencer time.

We define the function $t_{(s)}^{-1} : \mathbb{R} \to \mathbb{R}$

$$t_{(s)}^{-1}(x) = \begin{cases} x\,, & \text{if } x \le c_1\,; \\ x + \displaystyle\sum_{j=1}^{i} r_i\,, & \text{if } c_i < x \le c_{i+1}\,, \quad \forall i = 1, \ldots, n\,; \end{cases} \tag{6}$$

with $c_{i+1} := \infty$ which maps from the sequencer time into the song time for the STOPS, if $\mathcal{T}' \neq \emptyset$, and the function $t_{(d)}^{-1} : \mathbb{R} \to \mathbb{R}$

$$t_{(d)}^{-1}(x) = \begin{cases} x\,, & \text{if } x < c_1\,; \\ x + \displaystyle\sum_{j=1}^{i} r_i\,, & \text{if } c_i \le x < c_{i+1}\,, \quad \forall i = 1, \ldots, n\,; \end{cases} \tag{7}$$

that maps from the sequencer time into the song time for the DELAYS, if $\mathcal{T}' \neq \emptyset$. If $\mathcal{T}' = \emptyset$, then $t_{(s)}(x) = t_{(s)}^{-1}(x) = x$ and $t_{(d)}(x) = t_{(d)}^{-1}(x) = x$, for STOPS and DELAYS respectively. The final mapping from song time into sequencer time is the composition of $t_{(d)}$ and $t_{(s)}$, $t_{(d)} \circ t_{(s)}$. Similarly, the mapping from sequencer time into the song time is $t_{(s)}^{-1} \circ 1_{(d)}^{-1}$.

2 From sequencer time to beat

2.1 Stemania definition

A `SSC` file gives a list of pairs which defines the bpms. The first item in the pair is the target beat, and the second item is the desired BPM from that beat on. Let us imagine we have a `SSC` file with the following definition:

```
#BPMS:0,120:8,70:13,200;
```

Let us convert this cumbersome definition into a friendly structure:

```
{
  [
    beat: 0,
    bpm: 120
  ],
  [
    beat: 8,
    bpm: 180
  ],
  [
    beat: 13,
    bpm: 60
  ]
}
```

This `#BPMS` definition is telling us three things:

1. From beat $-\infty$ to beat 8, the BPM is 120.
2. From beat 8 to beat 13, the BPM is 180.
3. From beat 13 to beat $+\infty$, the BPM is 60.

2.2 Challenge

We want to find a function $f : \mathbb{R} \to \mathbb{R}$ that retrieves the current beat given the second. This function is useful when a song is playing and we want to know at what beat we are at if we know how much time has passed since the start of the song. Notes move at the speed of the BPM, so if we can have a function f, we can sort of know where the steps should be drawn.

2.3 Solution

First, let us convert BPMS to BPSS (Beats Per Second), since we are going to provide the input in seconds instead of minutes. We can do so by dividing the BPMS by 60, i.e.

$$\text{BPS}(x) = x \times \frac{\text{Beats}}{\text{Minute}} = x \times \frac{1 \times \text{Minute}}{60 \times \text{Seconds}} \frac{\text{Beats}}{\text{Minute}} = \frac{x}{60} \times \frac{\text{Beats}}{\text{Second}}\,. \tag{8}$$

Next, let us define a piecewise function $f' : \mathbb{R} \to \mathbb{R}$ that gives the current BPS given the current Beat. Taking the `#BPMS` toy example from the previous section, we get that

$$f'(x) = \begin{cases} 2\,, & \text{if } x \leq 8\,; \\ 3\,, & \text{if } 8 < x \leq 13\,; \\ 1\,, & \text{if } x > 13\,. \end{cases} \tag{9}$$

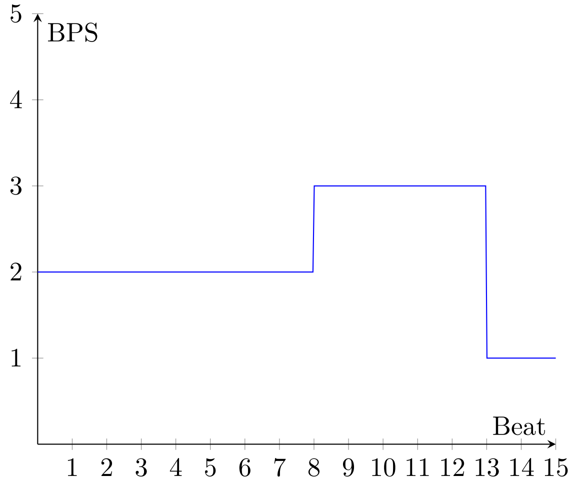

Figure 3: Plot of f'

In Figure 3 you can see the plot of f' we just defined in (9).

Note that by using f', we can get the BPS at any beat of the song. This is great, but it does not quite solve our problem.

Next, we can calculate the SPB (Seconds Per Beat) by just inversing the BPS, i.e.

$$\mathrm{SPB} = \frac{1}{\mathrm{BPS}}\,, \tag{10}$$

and therefore we can define a function $t : \mathbb{R} \to \mathbb{R}$

$$t(x) = x \times \mathrm{SPB} \tag{11}$$

that given a beat x retrieves the current second.

Let

$$f''(x) = \begin{cases} \dfrac{x}{2}\,, & \text{if } x \leq 8\,; \\[2ex] \dfrac{8}{2} + \dfrac{x-8}{3}\,, & \text{if } 8 < x \leq 13\,; \\[2ex] \dfrac{8}{2} + \dfrac{5}{3} + x - 13\,, & \text{if } x > 13\,; \end{cases} \tag{12}$$

be the function that given a beat x retrieves the current second. This function is the result of plugging (10) and (11) into (9), and can be rewritten recursively as

$$f''(x) = \begin{cases} \dfrac{x}{2}\,, & \text{if } x \leq 8\,; \\[2ex] f''(8) + \dfrac{x-8}{3}\,, & \text{if } 8 < x \leq 13\,; \\[2ex] f''(13) + x - 13\,, & \text{if } x > 13\,. \end{cases} \tag{13}$$

Figure 4 depicts the function f''. Note that this is just the opposite of the desired solution!

As it turns out, the function f that we are looking for is just the inverse function of f'', in short

$$f = f''^{-1}\,, \tag{14}$$

thus

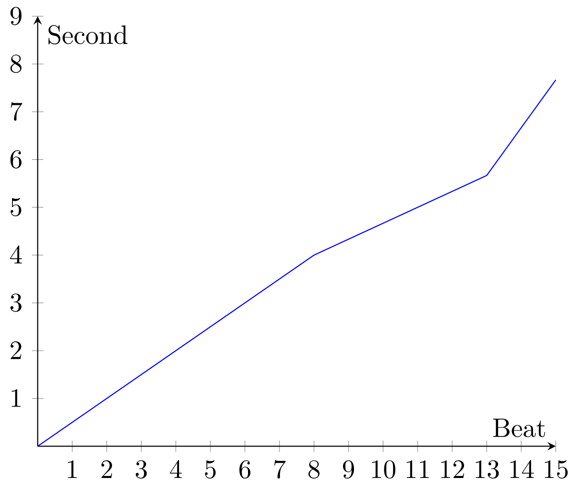

Figure 4: Plot of f''

$$f(x) = \begin{cases} 2x\,, & \text{if } x \leq 4\,; \\ (x-4) \times 3 + 8\,, & \text{if } 4 < x \leq 5.6\,; \\ x - 5.6 + 13\,, & \text{if } x > 5.6\,. \end{cases} \tag{15}$$

A plot of f can be seen in Figure 5.

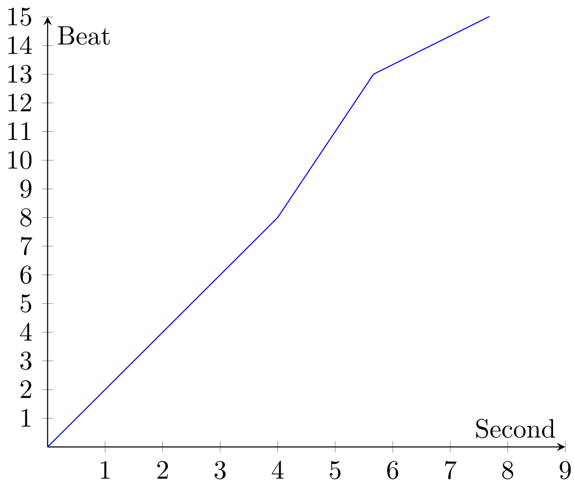

Figure 5: Plot of f

2.4 Formalization

Let $d = \{(b_i, v_i)\}_{i=1}^{n}$ be a sequence of n beat signatures, where v_i is the BPS value at beat b_i. Let $f : \mathbb{R} \to \mathbb{R}$ be a function that provided a beat, returns the seconds from the first beat. We define this function as a n-step piecewise function

$$f(x) = \begin{cases} \dfrac{x}{v_1}\,, & \text{if } x \leq b_2\ ; \\ f(b_i) + \dfrac{x - b_i}{v_i}\,, & \text{if } b_i < x \leq b_{i+1}\,; \quad \forall i = 2, \ldots, n\,; \end{cases} \tag{16}$$

where $b_1 := 0$, and $b_{n+1} := \infty$.

Analogously, let $g : \mathbb{R} \to \mathbb{R}$ be a function that provided a second, returns the beat from the zero second. We define this function as a n-step piecewise function

$$g(x) = \begin{cases} v_1 x\,, & \text{if } x \leq f(b_2)\ \ ; \\ [x - f(b_i)] \times v_i + b_i\,, & \text{if } f(b_i) < x \leq f(b_{i+1})\,; \quad \forall i = 2, \dots, n\,, \end{cases} \tag{17}$$

where $g = f^{-1}$.

3 From beat to note position

3.1 Stepmania definition

There are a pair of stepmania definitions that influence the position where notes should be placed on the screen. One of them is `#BPMS`, which arguably is the rate at what notes travel upwards towards the receptor w.r.t. to the music rhythm. As well see later on, we will consider the BPMS as a speed function from which we can discover the position of a note given the beat it should be tapped. As a toy example, we will use the same definition as provided in Section 2.1.

However, there is another gimmick that plays a role in the note positioning: `#SCROLLS`. Let's have a look at an example:

```
#SCROLLS:0=1,4=0,10=2;
```

Again, let us convert this cumbersome definition into a friendly structure:

```
{
  [
    beat: 0,
    scroll: 1
  ],
  [
    beat: 4,
    scroll: 0
  ],
  [
    beat: 10,
    scroll: 2
  ]
}
```

The scroll gimmick changes the effective BPM at a given beat by a rate defined by the scroll value. Thus, in this example, the SCROLLS gimmick is changing the BPMs as follows:

1. From beat 0 to beat 4, the BPM is its value times 1.
2. From beat 4 to beat 10, the BPM is its value times 0. (all the steps in between this beats, will have the same position)
3. From beat 10 on, the BPM is its value times 2.

3.2 Challenge

We would like to have a function $p : \mathbb{R} \to \mathbb{R}$ that given a beat, it retrieves the position w.r.t. the origin (or where the receptor is) where a note at that beat should be drawn.

3.3 Solution

Let us start by defining a function $h : \mathbb{R} \to \mathbb{R}$ that retrieves the current BPS (Beats per Second) from the current beat. Given the example, this function is identical to that of (9),

$$h(x) = \begin{cases} 2\,, & \text{if } x \leq 8\,; \\ 3\,, & \text{if } 8 < x \leq 13\,; \\ 1\,, & \text{if } x > 13\,. \end{cases} \tag{18}$$

You can see a plot of this function in Figure 3. Now, let us define a new function $g : \mathbb{R} \to \mathbb{R}$ that given a beat, retrieves the effective BPS (i.e., BPS with applied scrolls). For that matter, we just need to check out in what beats the scroll is taking place, and change the BPS accordingly. For our toy example the resultant g function looks like this

$$g(x) = \begin{cases} 1 \times 2\,, & \text{if } x \leq 4\,; \\ 0 \times 2\,, & \text{if } 4 < x \leq 8\,; \\ 0 \times 3\,, & \text{if } 8 < x \leq 10\,; \\ 2 \times 3\,, & \text{if } 10 < x \leq 13\,; \\ 2 \times 1\,, & \text{if } x > 13\,. \end{cases} \tag{19}$$

The function g is depicted in Figure 6. Now, this is great! By asking g, now we have the effective BPS. Note that g is a function that models speed, i.e. the velocity that the notes should move towards the receptor. To retrieve the position at each beat, we can just consider that $\frac{dp}{dx} = g(x)$. Therefore, to get p, we just have to take in integral of g w.r.t. x,

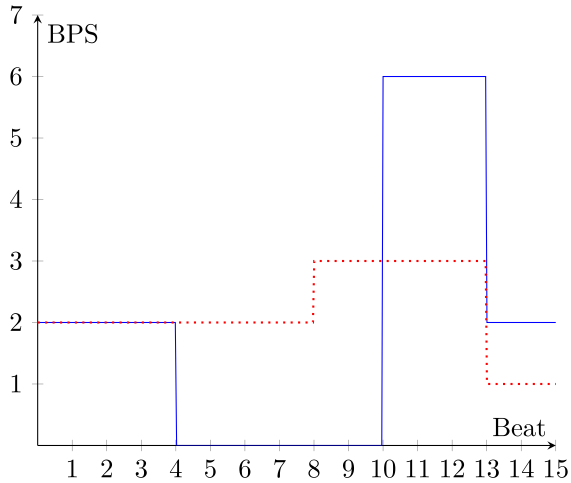

Figure 6: Plot of g (blue line), Plot of h (dotted, red line)

$$p(x) = \int g(x)dx\,. \tag{20}$$

For instance, if we want to know the position of a note at the beat 11, it would only take to calculate

$$\text{position} = \int_0^{11} g(x)dx\,. \tag{21}$$

You can see an example in Figure 7. As shown, the position at beat 11 is the area under the curve (in light blue) from 0 to 11.

The resulting integral of $g(x)$ in our toy example is

$$p(x) = \begin{cases} 2x\,, & \text{if } x \leq 4\,; \\ p(4) + 0\,, & \text{if } 4 < x \leq 8\,; \\ p(8) + 0\,, & \text{if } 8 < x \leq 10\,; \\ p(10) + (x - 10) \times 6\,, & \text{if } 10 < x \leq 13\,; \\ p(13) + (x - 13) \times 1\,, & \text{if } x > 13\,, \end{cases} \tag{22}$$

and its plot can be seen in Figure 8.

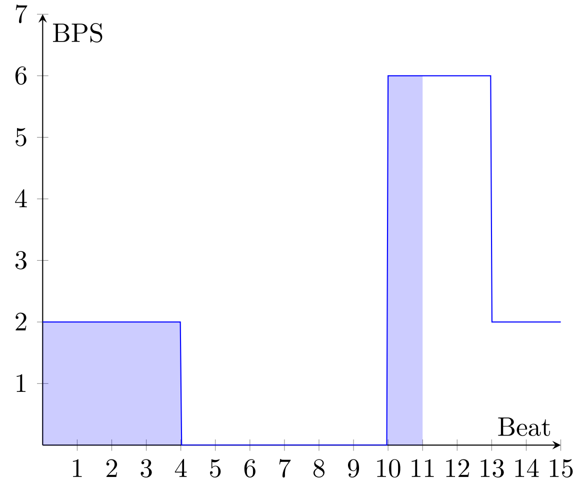

Figure 7: The position at beat 11 will be the area underneeth the function g.

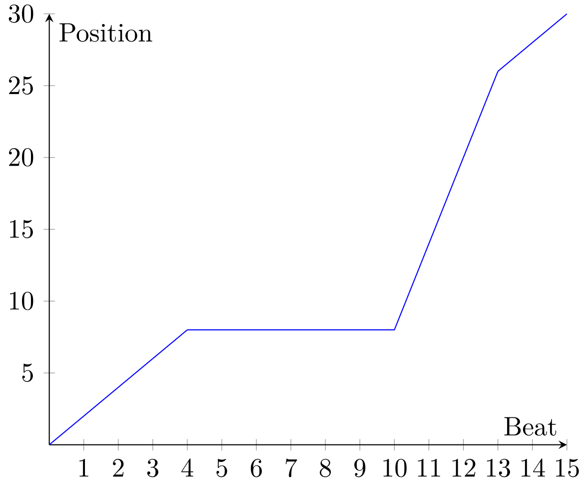

Figure 8: Plot of p

3.4 Formalization

Let

$$\left\{\left(b_i^{(b)}, v_i\right)\right\}_{i=1}^{n} = \mathcal{B} = B^{(b)} \times V \tag{23}$$

be a sequence of n beat signatures, where $v_i \in V = \{v_j\}_{j=1}^{n}$ is the BPS value at beat $b_i^{(b)} \in B^{(b)} = \{b_i^{(b)}\}_{j=1}^{n}$.

Let

$$\left\{\left(b_i^{(s)}, s_i\right)\right\}_{i=1}^{m} = \mathcal{S} = B^{(s)} \times S \tag{24}$$

be a sequence of m scroll signatures, where $s_i \in S = \{s_j\}_{j=1}^{m}$ is the scroll value at beat $b_i^{(s)} \in B^{(s)} = \{b_j^{(s)}\}_{j=1}^{m}$.

Let

$$\{(b_i, z_i)\}_{i=1}^{n'} = \mathcal{Z} = B^{(b)} \cup B^{(s)} \times V \cup S \tag{25}$$

be a sequence of n' BPSs with applied scrolls, where z_i is the effective BPS at beat b_i constructed from $\mathcal{S}$ and $\mathcal{B}$ as

$$\mathcal{Z} = \bigcup_{i=1}^{m} \left\{\left(b_i^{(s)}, h(b_i^{(s)}) \times s_i\right)\right\} \bigcup_{j=1}^{n} \begin{cases} \left\{\left(b_j^{(b)}, v_j \times s_i\right)\right\}, & \text{if } b_i^{(s)} < b_j^{(b)} < b_{i+1}^{(s)}\,; \\ \emptyset\,, & \text{otherwise}\,; \end{cases} \tag{26}$$

where $h : \mathbb{R} \to \mathbb{R}$ is a function that given a beat, returns the BPS for that beat, and $b_{m+1}^{(s)} := \infty$. We define

$$g(x) = \begin{cases} z_1\,, & \text{if } x \le b_2\,; \\ z_i\,, & \text{if } b_i < x \le b_{i+1}\,; \quad \forall i = 2, \ldots, n'\,, \end{cases} \tag{27}$$

as a function that retrieves the effective BPS given a beat, and $b_{n'+1} := \infty$. Finally, we define

$$p(x) = \int g(x)dx = \begin{cases} xz_1\,, & \text{if } x \le b_2\,; \\ p(b_i) + (x - b_i) \times z_i\,, & \text{if } b_i < x \le b_{i+1}\,; \quad \forall i = 2, \ldots, n'\,, \end{cases} \tag{28}$$

as the function that retrieves the position given a beat.