Chapter 1

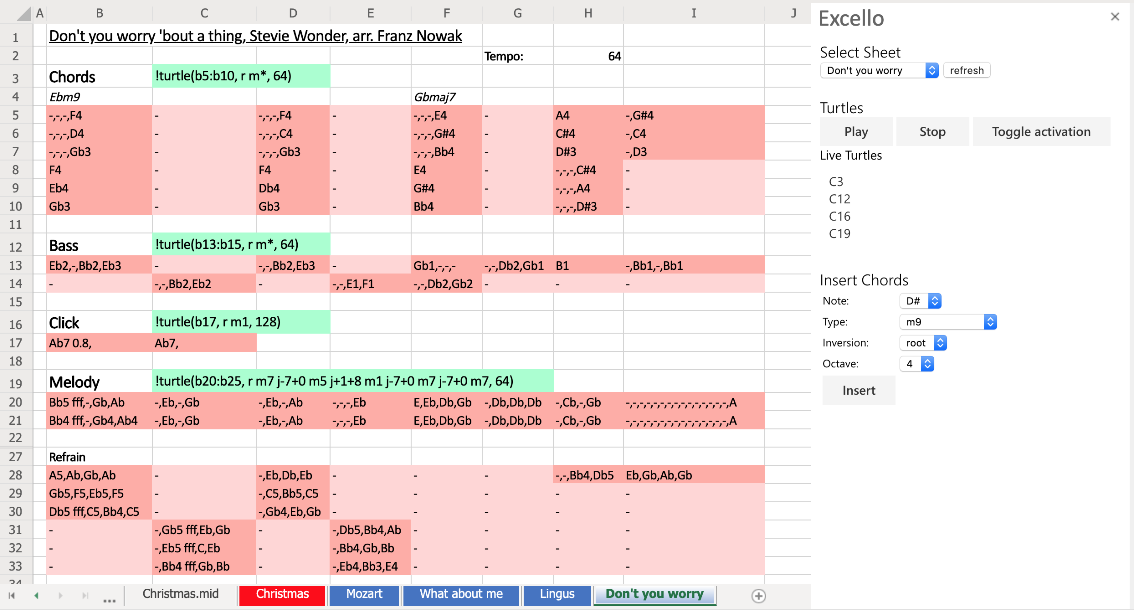
Evaluation

This chapter shall discuss the effectiveness of the Excello implementation with regards to the success criteria and by showcasing examples. The conversion of an existing MIDI corpus to the Excello notation using the converter from MIDI will demonstrate the expressiveness of the notation. Next I shall explain the summative evaluation and use the data from this to assess the features implemented in the participatory design process and to reason about Excello using the Cognitive Dimensions of Notation framework [4]. Finally, the ethics and data handling procedures shall be covered.

1.1 Excello Success

Both a notation for music expression and a program integrated into Excel for the playback of this expression has been implemented. As required by the success criteria, users can play multiple notes and chords of different durations. These can be combined into looped sequences with a defined tempo. In the participatory design process additional features were added as extensions. Defining multiple successive notes in a cell, turtles calculating how far they should move and nested instructions with repeats are additional features facilitating more efficient notation. Custom Excel functions, a chord adding tool and faster turtle toggling allows users to work more efficiently. Figure 1.1 shows Excello in use with participant's arrangement.

The first section of Piano Phase by Steve Reich consists of two equal piano melodies, one slightly faster than the other. The two parts move out of phase before aligning at different offsets. This is included as an example for many end-user programming tools. This is implemented in Manhattan using three rows of 24 columns [6]. Sonic Pi defined the notes in one line and eight additional lines are required for playback. Piano Phase can not be concisely notated by western staff notation. Excello only requires two cells to define two turtles of different speeds in addition to the notes. All three implementations are shown in figure 1.2.



	A	B	C	D	E	F	G	H	I	J
1		Don't you worry 'bout a thing, Stevie Wonder, arr. Franz Nowak								
2							Tempo:	64		
3		Chords	!turtle(b5:b10, r m*, 64)							
4		*Ebm9*				*Gbmaj7*				
5		-,-,-,F4	-	-,-,-,F4	-	-,-,-,E4	-	A4	-,G#4	
6		-,-,-,D4	-	-,-,-,C4	-	-,-,-,G#4	-	C#4	-,C4	
7		-,-,-,Gb3	-	-,-,-,Gb3	-	-,-,-,Bb4	-	D#3	-,D3	
8		F4	-	F4	-	E4	-	-,-,-,C#4	-	
9		Eb4	-	Db4	-	G#4	-	-,-,-,A4	-	
10		Gb3	-	Gb3	-	Bb4	-	-,-,-,D#3	-	
11										
12		Bass	!turtle(b13:b15, r m*, 64)							
13		Eb2,-,Bb2,Eb3	-	-,-,Bb2,Eb3	-	Gb1,-,-,-	-,-,Db2,Gb1	B1	-,Bb1,-,Bb1	
14		-	-,-,Bb2,Eb2	-	-,-,E1,F1	-,-,Db2,Gb2	-	-	-	
15										
16		Click	!turtle(b17, r m1, 128)							
17		Ab7 0.8,	Ab7,							
18										
19		Melody	!turtle(b20:b25, r m7 j-7+0 m5 j+1+8 m1 j-7+0 m7 j-7+0 m7, 64)							
20		Bb5 fff,-,Gb,Ab	-,Eb,-,Gb	-,Eb,-,Ab	-,-,-,Eb	E,Eb,Db,Gb	-,Db,Db,Db	-,Cb,-,Gb	-,-,-,-,-,-,-,-,-,-,-,-,-,-,-,A	
21		Bb4 fff,-,Gb4,Ab4	-,Eb,-,Gb	-,Eb,-,Ab	-,-,-,Eb	E,Eb,Db,Gb	-,Db,Db,Db	-,Cb,-,Gb	-,-,-,-,-,-,-,-,-,-,-,-,-,-,-,A	
22										
27		Refrain								
28		A5,Ab,Gb,Ab	-	-,Eb,Db,Eb	-	-	-	-,-,Bb4,Db5	Eb,Gb,Ab,Gb	
29		Gb5,F5,Eb5,F5	-	-,C5,Bb5,C5	-	-	-	-	-	
30		Db5 fff,C5,Bb4,C5	-	-,Gb4,Eb,Gb	-	-	-	-	-	
31		-	-,Gb5 fff,Eb,Gb	-	-,Db5,Bb4,Ab	-	-	-	-	
32		-	-,Eb5 fff,C,Eb	-	-,Bb4,Gb,Bb	-	-	-	-	
33		-	-,Bb4 fff,Gb,Bb	-	-,Eb4,Bb3,E4	-	-	-	-	

Christmas.mid | Christmas | Mozart | What about me | Lingus | Don't you worry

Excello ×

Select Sheet

Don't you worry refresh

Turtles

Play Stop Toggle activation

Live Turtles

C3
C12
C16
C19

Insert Chords

Note: D#

Type: m9

Inversion: root

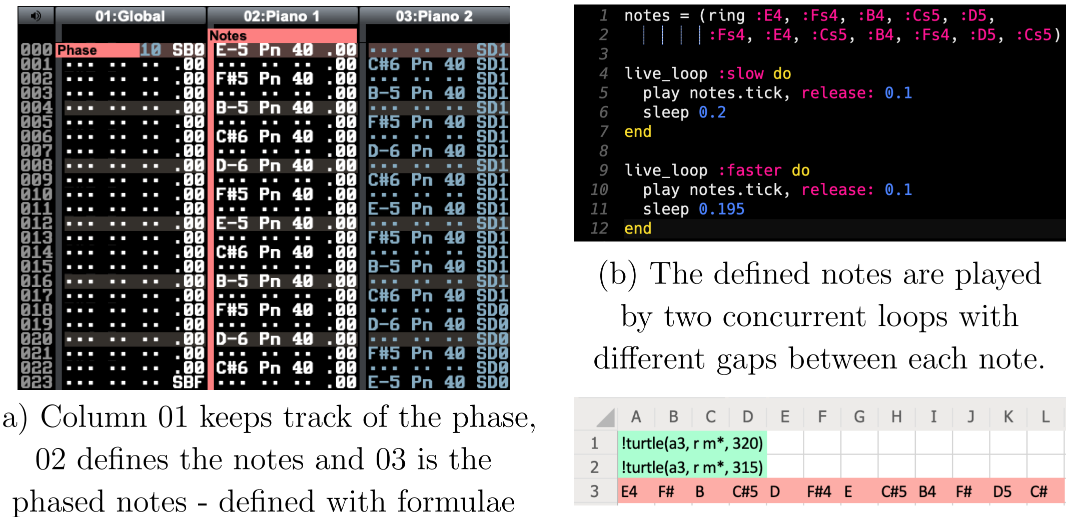
Octave: 4

Insert

Figure 1.1: An arrangement with separated and labelled parts per instrument. Turtles refer to a global tempo at the top of the spreadsheet.

1.2 MIDI Corpus Conversion

Whilst being able concisely notate music western notation and other end-user programming systems do not facilitate, Excello can exactly express any combination of concurrent note onset and offsets. Therefore any piece defined in MIDI can be expressed in Excello. In MIDI allows tempo can be redefined within a track, this is not be accounted for. If instead the time between note onsets and offsets is adjusted, the uncompressed file will account for this but compressing algorithm will fail as the difference between notes may become a non-integer multiple of the minimum. Instrument specific effects such as piano pedals are not supported. This naive conversion can result in unwieldy spreadsheet sizes. One conversion method compresses the representation by dividing the difference between notes by the minimum difference in not onset between any two notes. Provided the difference between any two notes is a multiple of the minimum difference, this compression method is lossless, whilst resulting in spreadsheets using orders of magnitude fewer cells. Therefore this method would not accurately convert quavers against triplets (three notes played in the same time as two) provided these notes were not multiples of a smaller note. Given the lengths of MIDI notes can be different to the time the note occupies in notation, to automate the compression, an assumption on the ratio of note lengths was required. The modal compression algorithm is lossy if the minimum note distance is not the modal distance. This is useful if there are ornaments or note within a piece that dramatically decrease the minimum distance but occur infrequently. Therefore their loss may be tolerable for a more efficient representation.



```
     01:Global          02:Piano 1      03:Piano 2
                        Notes
000  Phase      10 SB0  E-5 Pn 40 .00   ... .. .. SD1
001  ... .. ..     .00  ... .. .. .00   C#6 Pn 40 SD1
002  ... .. ..     .00  F#5 Pn 40 .00   ... .. .. SD1
003  ... .. ..     .00  ... .. .. .00   B-5 Pn 40 SD1
004  ... .. ..     .00  B-5 Pn 40 .00   ... .. .. SD1
005  ... .. ..     .00  ... .. .. .00   F#5 Pn 40 SD1
006  ... .. ..     .00  C#6 Pn 40 .00   ... .. .. SD1
007  ... .. ..     .00  ... .. .. .00   D-6 Pn 40 SD1
008  ... .. ..     .00  D-6 Pn 40 .00   ... .. .. SD1
009  ... .. ..     .00  ... .. .. .00   C#6 Pn 40 SD1
010  ... .. ..     .00  F#5 Pn 40 .00   ... .. .. SD1
011  ... .. ..     .00  ... .. .. .00   E-5 Pn 40 SD1
012  ... .. ..     .00  E-5 Pn 40 .00   ... .. .. SD1
013  ... .. ..     .00  ... .. .. .00   F#5 Pn 40 SD1
014  ... .. ..     .00  C#6 Pn 40 .00   ... .. .. SD1
015  ... .. ..     .00  ... .. .. .00   B-5 Pn 40 SD1
016  ... .. ..     .00  B-5 Pn 40 .00   ... .. .. SD1
017  ... .. ..     .00  ... .. .. .00   C#6 Pn 40 SD1
018  ... .. ..     .00  F#5 Pn 40 .00   ... .. .. SD0
019  ... .. ..     .00  ... .. .. .00   D-6 Pn 40 SD0
020  ... .. ..     .00  D-6 Pn 40 .00   ... .. .. SD0
021  ... .. ..     .00  ... .. .. .00   F#5 Pn 40 SD0
022  ... .. ..     .00  C#6 Pn 40 .00   ... .. .. SD0
023  ... .. ..     SBF  ... .. .. .00   E-5 Pn 40 SD0
```

(a) Column 01 keeps track of the phase, 02 defines the notes and 03 is the phased notes - defined with formulae that update depending on the phase and defined notes.

```
notes = (ring :E4, :Fs4, :B4, :Cs5, :D5,
  | | | | :Fs4, :E4, :Cs5, :B4, :Fs4, :D5, :Cs5)

live_loop :slow do
  play notes.tick, release: 0.1
  sleep 0.2
end

live_loop :faster do
  play notes.tick, release: 0.1
  sleep 0.195
end
```

(b) The defined notes are played by two concurrent loops with different gaps between each note.

	A	B	C	D	E	F	G	H	I	J	K	L
1	!turtle(a3, r m*, 320)											
2	!turtle(a3, r m*, 315)											
3	E4	F#	B	C#5	D	F#4	E	C#5	B4	F#	D5	C#

(c) Two turtles play the same notes at different speeds.

Figure 1.2: Implementations of Steve Reich's Piano Phase in a) Manhattan, b) Sonic Pi, c) Excello

I have converted three MIDI corpora. The first is a collection of 497 Bach chorales[1] made by Margaret Greentree. The second is 277 piano pieces[2] help by Bernd Krueger under a creative commons license. Finally 194 Bach pieces made available from "A Johann Sebastian Bach Midi Page"[3]. This is not all the files available from this site as some were not readable by the python MIDI reader. All 968 MIDI files were converted using all three methods.

The language of Excello is expressive enough to represent MIDI files and can do so concisely provided the condition of minimum note onset differences is maintained.

1.3 Summative Evaluation Sessions

Of the 21 users who participated in formative evaluation, 19 had continued using Excello. These users all filled out a summative evaluation questionnaire. First a review of the features that had been added since the initial sessions was given. To ensure users had a sufficient understanding of the interface before giving feedback, a short transcription task involving some original composition by the users was given.

The questionnaire first assessed the success of the features added during the participatory design process by comparing the interface before and after a feature had been added. Seven-point Likert scale questions were given to test if the issues had been solved and if overall the change rendered the system more preferable. The remaining questions were based on Blackwell and Green's cognitive dimensions questionnaire [2]. Cognitive

[1] Accessed from https://github.com/jamesrobertlloyd/infinite-bach/tree/master/data/chorales/midi
[2] http://piano-midi.de/midis.htm
[3] http://www.bachcentral.com/midiindex.html

dimensions of notations can be used to analyse musical notation [3] in addition to software systems [5], therefore it is a suitable tool for the discussion of the Excello notation and interface. Because dimensions have different significances for the different activities [4], users identified what percentage of their time they spent carrying out the five different user activities (searching for information, translating, incrementation, modification and exploratory design). Questions focusing on closeness of mapping, consistency, secondary notation, viscosity and visibility were used as planned in the project proposal. Usage and cognitive dimensions questions were also answered with respect to the user's preferred interface for music manipulation, composition or transcription. 12 users chose Sibelius, which shall be used for comparison.

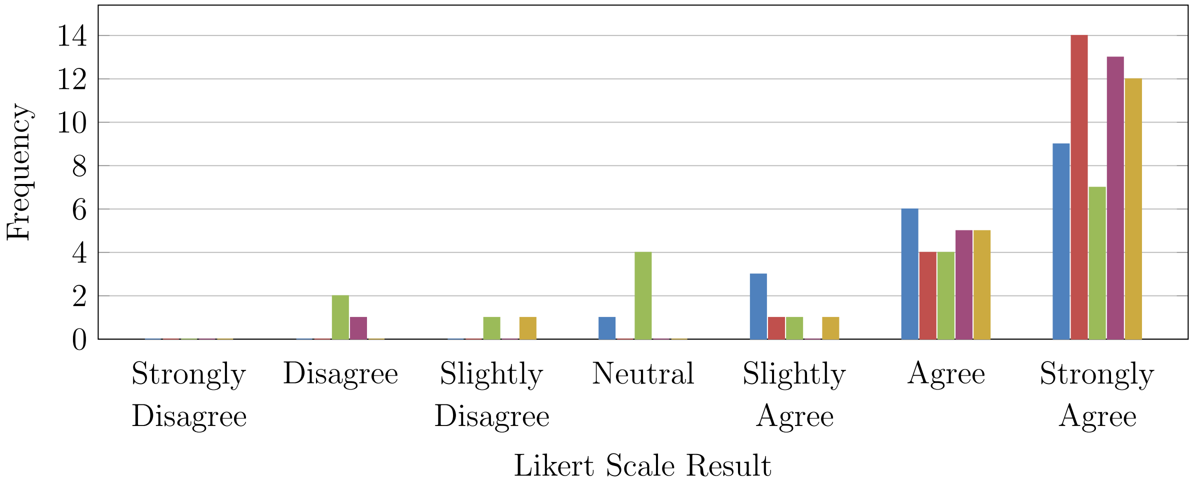
1.4 Success of Participatory Design

For each feature added, Excello with this feature (system 2) was compared to first prototype implementation. The following charts show the frequency of Likart scale responses for each question. Mean's of Likert scale data provide are not useful [1] so the mode is considered. A Chi-Squared goodness-of-fit test tests if the distribution is significantly different to uniform. As all expected values must be greater than 1 and 80% greater than or equal to five [7] and the expected frequency for one result is $19/7 \approx 2.7$, I combine Strongly Disagree with Disagree, Strong Agree with Agree and the remaining three options into a third group. The p-value given is from the Chi-Squared test using these three categories.

1.4.1 Dynamics in the Cell

Statement	**Mode**	**p-value**
🟦 It is easier to figure out the turtles path	Strongly Agree	0.0000
🟥 It is easier to figure out what dynamics different notes are played at	Strongly Agree	0.0146
🟩 It is easier to tell the order in which dynamics are applied	Strongly Agree	0.0000
🟪 It is easier to write dynamics in the correct place	Strongly Agree	0.0000
🟨 Overall system 2 is preferable	Strongly Agree	0.0000

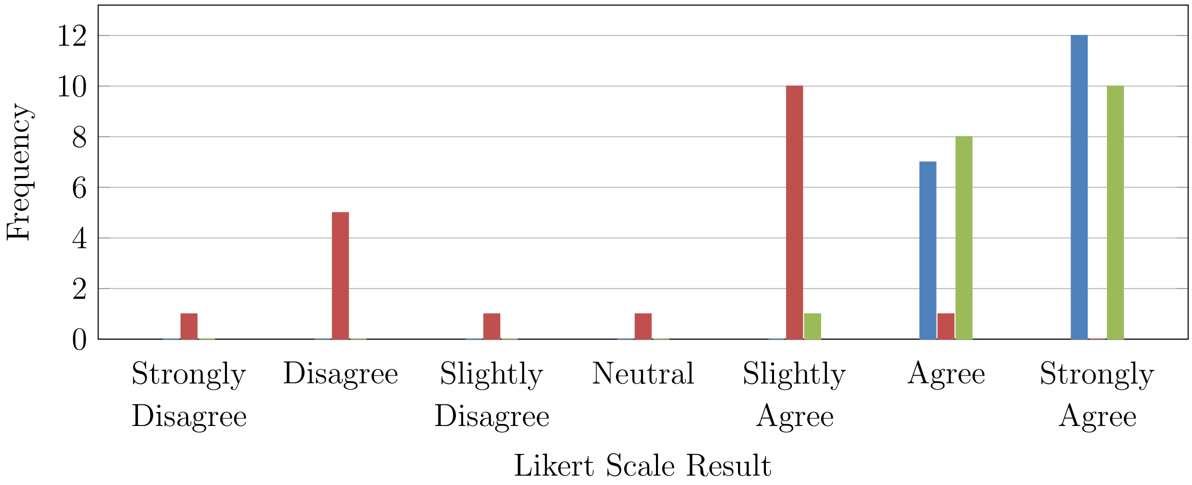
All issues that were aimed to be solved, users strongly agreed were improved resulting in a significantly improved system.



1.4.2 Inferred Octave

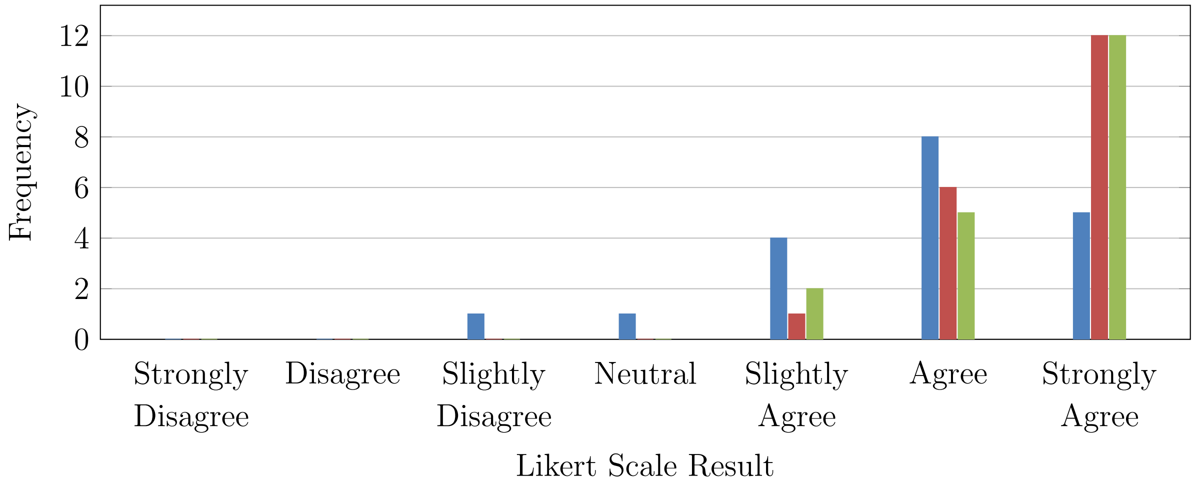
Statement	Mode	p-value
🟦 less effort is required to write a part	Strongly Agree	0.0000
🟥 it is harder to figure out what octave a note will be played in	Slightly Agree	0.0639
🟩 Overall, system 2 is preferable	Strongly Agree	0.0000

Depending on its use, the inferred octave notation has the tradeoff that the octave is less easier inferred. There were no strong agreements and only one agreement that this was the case but the modal response was slight agreement. However many users didn't find this an issue and the distribution of responses is not significantly different to uniform. Overall this addition offered a significant improvement.



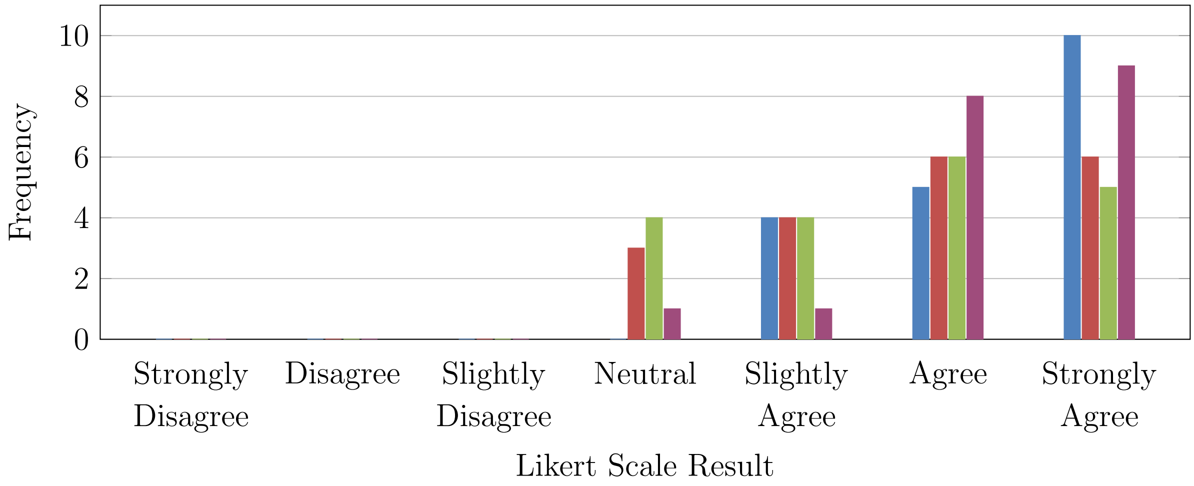
Statement	**Mode**	**p-value**
🟦 it is easier to parse the turtle instruction and tell what it will do.	Agree	0.0003
🟥 it is easier to repeat sections of notes.	Strongly Agree	0.0000
🟩 Overall, system 2 is preferable.	Strongly Agree	0.0000

1.4.3 Nested Instructions



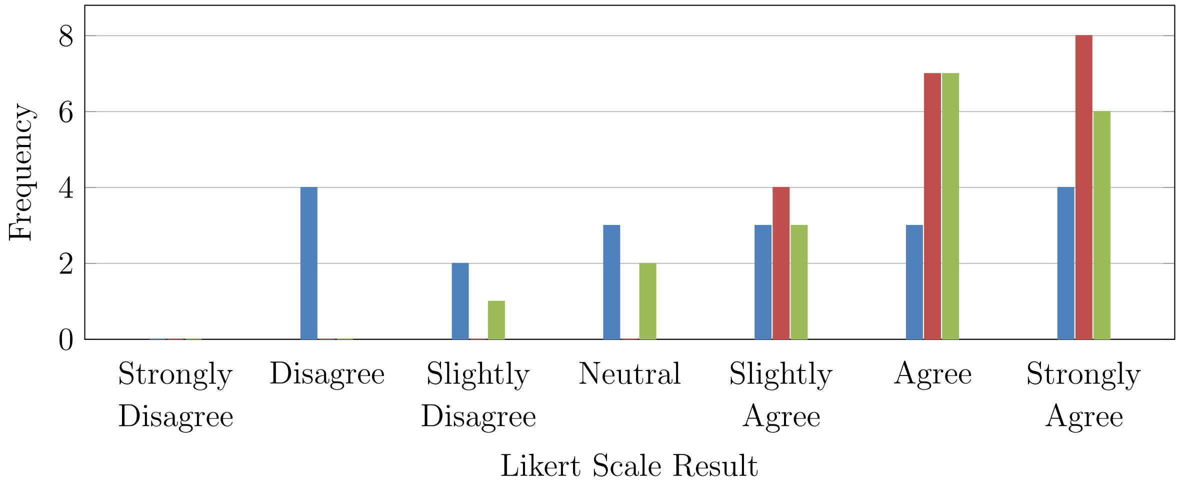
1.4.4 Active Turtles

Statement	**Mode**	**p-value**
🟦 it is easier to tell if a certain turtle has been registered.	Strongly Agree	0.0000
🟥 it is easier to see where the active turtles are.	(Strongly) Agree	0.0011
🟩 it is easier to toggle the activation of turtles.in the correct place	Agree	0.0038
🟪 Overall, system 2 is preferable.	Strongly Agree	0.0000



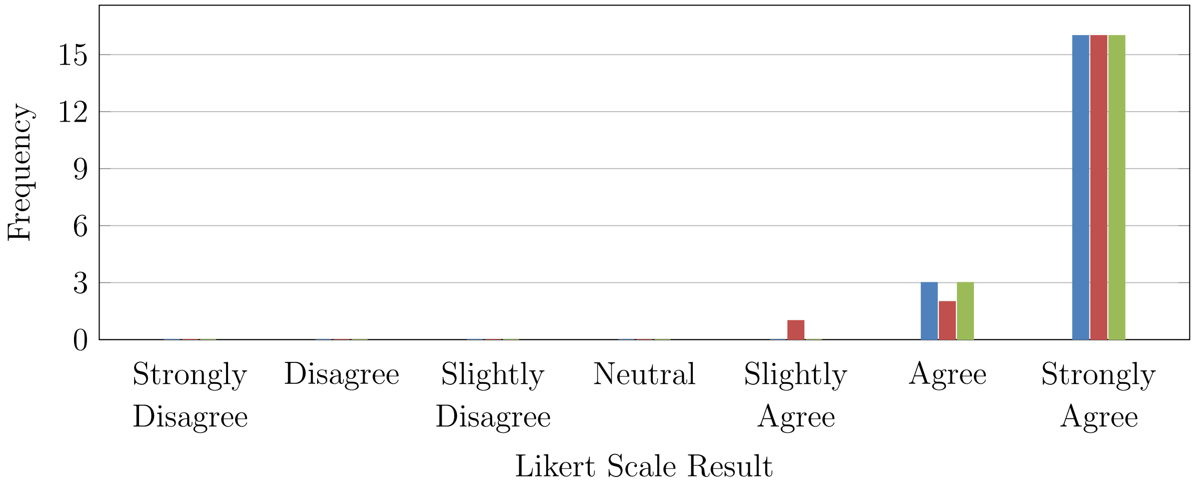
Statement	Mode	p-value
🟦 it is more intuitive how loud a note will be played.	Disagree Strongly Agree	0.6592
🟥 the volumes available are less limited.	Strongly Agree	0.0000
🟩 Overall, system 2 is preferable..	Agree	0.0003

1.4.5 Continuous Volume



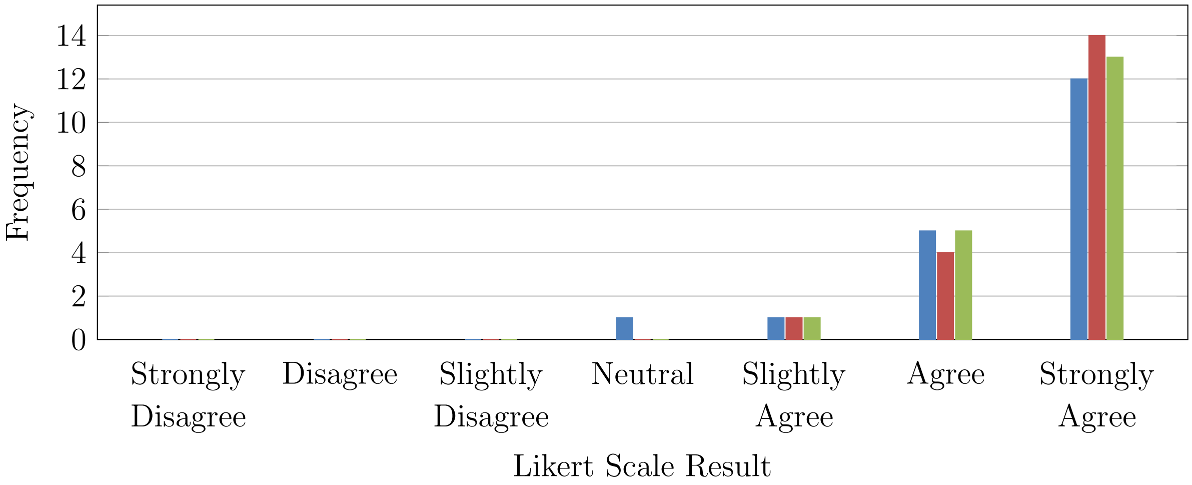
1.4.6 Automatic Stepping

Statement	Mode	p-value
🟦 less mental work is required to write the turtle instructions.	Strongly Agree	0.0000
🟥 less work is required when more notes wish to be added.	Strongly Agree	0.0000
🟩 Overall, system 2 is preferable.	Strongly Agree	0.0000



Statement	**Mode**	**p-value**
🟦 it is easier to tell what the speed instruction corresponds to	Strongly Agree	0.0000
🟥 giving an exact tempo (e.g. when transcribing sheet music) is easier	Strongly Agree	0.0000
🟩 Overall, system 2 is preferable.	Strongly Agree	0.0000

1.4.7 Absolute Tempo



Bibliography

[1] Dwight Barry. Do not use averages with likert scale data, 01 2017.

[2] Alan F. Blackwell and Thomas R. G. Green. A cognitive dimensions questionnaire optimised for users. In *PPIG*, 2000.

[3] Alan F. Blackwell, Thomas Green, and Dje Nunn. Cognitive dimensions and musical notation systems. *Workshop on Notation and Music Information Retrieval*, 11 2000.

[4] Thomas Green and Alan Blackwell. Cognitive dimensions of information artefacts: a tutorial. Technical Report Version 1.2, BCS HCI Conference, 1998.

[5] Thomas Green and Marian Petre. Usability analysis of visual programming environments: A 'cognitive dimensions' framework. *Journal of Visual Languages*, 7:131–, 06 1996.

[6] Chris Nash. Manhattan: End-user programming for music. In *NIME*, 2014.

[7] S.M. Ross. *Introductory Statistics*. Elsevier Science, 2010.