Subject: Multimedia signal synthesis
Student: Rubén Diab Martínez

SYNTH CHALLENGE 2017

Task 1 - Synthesis of the arbitrary instrument for the composition "Flight of the Bumblebee" by Nikolaj Rimskij Korsakov

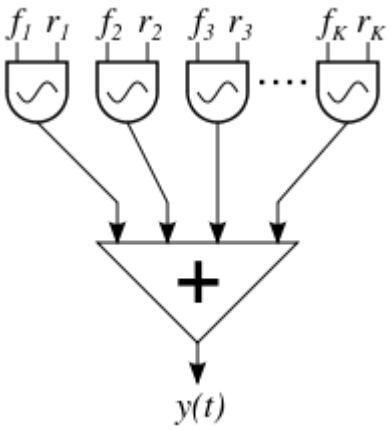
For this task, I used an additive synthesis of the piano for the composition "Flight of the Bumblebee" by Nikolaj Rimskij Korsakov.

I used "Bumble01.mid" file because, personally, is the best for making the synthesis.

Additive synthesis is a sound synthesis technique that creates timbre by adding sine waves together.

The timbre of musical instruments can be considered in the light of Fourier theory to consist of multiple harmonic or not harmonic partials or overtones. Each partial is a sine wave of different frequency and amplitude that swells and decays over time due to modulation from an ADSR envelope or low frequency oscillator.

Additive synthesis most directly generates sound by adding the output of multiple sine wave generators.



Schematic diagram of additive synthesis. The inputs to the oscillators are frequencies f_k and r_k →

Is needed to say that I applied envelopes for the additive synthesis which helped to obtain a better quality of piano sound.

I also applied first a Chebyshev type II low filter and after an Elliptic high filter.

I choosed these previous filters trying different kind of filters and combining them, until best synthesis was done from my point of view. These filters are nothing we haven't seen in class, but I have studied them in Digital Signal Processing subject.

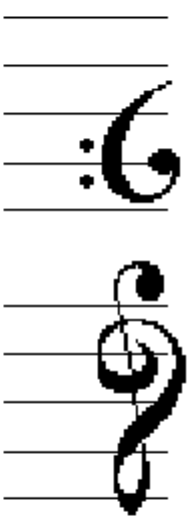
In my code, for this task I use the synthchallenge.m which makes the normalization of the signal, and I also changed the last part of the code to be able to change the name of the synthesized files that need to be saved in the folder results.

Task 2 - Two octaves of the major scale for one chosen instrument from the previous synthesis

For this task I used the part of the code corresponding to midi2freq.m which converts MIDI note number (m=0-127) to frequency f in (Hz).

I also used the next table to get the correct MIDI numbers corresponding to two octaves of the major scale:

MIDI number		Note name	Keyboard	Frequency Hz		Period ms	
21		A0		27.500		36.36	
23	22	B0		30.868	29.135	32.40	34.32
24		C1		32.703		30.58	
26	25	D1		36.708	34.648	27.24	28.86
28	27	E1		41.203	38.891	24.27	25.71
29		F1		43.654		22.91	
31	30	G1		48.999	46.249	20.41	21.62
33	32	A1		55.000	51.913	18.18	19.26
35	34	B1		61.735	58.270	16.20	17.16
36		C2		65.406		15.29	
38	37	D2		73.416	69.296	13.62	14.29
40	39	E2		82.407	77.782	12.13	12.86
41		F2		87.307		11.45	
43	42	G2		97.999	92.499	10.20	10.81
45	44	A2		110.00	103.83	9.091	9.631
47	46	B2		123.47	116.54	8.099	8.581
48		C3		130.81		7.645	
50	49	D3		146.83	138.59	6.811	7.216
52	51	E3		164.81	155.56	6.068	6.428
53		F3		174.61		5.727	
55	54	G3		196.00	185.00	5.102	5.405
57	56	A3		220.00	207.65	4.545	4.816
59	58	B3		246.94	233.08	4.050	4.290
60		**C4**		**261.63**		**3.822**	
62	61	D4		293.67	277.18	3.405	3.608
64	63	E4		329.63	311.13	3.034	3.214
65		F4		349.23		2.863	
67	66	G4		392.00	369.99	2.551	2.703
69	68	**A4**		**440.00**	415.30	**2.273**	2.408
71	70	B4		493.88	466.16	2.025	2.145
72		C5		523.25		1.910	
74	73	D5		587.33	554.37	1.703	1.804
76	75	E5		659.26	622.25	1.517	1.607
77		F5		698.46		1.432	
79	78	G5		783.99	739.99	1.276	1.351
81	80	A5		880.00	830.61	1.136	1.204
83	82	B5		987.77	932.33	1.012	1.073
84		C6		1046.5		0.9556	
86	85	D6		1174.7	1108.7	0.8513	0.9020
88	87	E6		1318.5	1244.5	0.7584	0.8034
89		F6		1396.9		0.7159	
91	90	G6		1568.0	1480.0	0.6378	0.6757
93	92	A6		1760.0	1661.2	0.5682	0.6020
95	94	B6		1975.5	1864.7	0.5062	0.5363
96		C7		2093.0		0.4778	
98	97	D7		2349.3	2217.5	0.4257	0.4510
100	99	E7		2637.0	2489.0	0.3792	0.4018
101		F7		2793.0		0.3580	
103	102	G7		3136.0	2960.0	0.3189	0.3378
105	104	A7		3520.0	3322.4	0.2841	0.3010
107	106	B7		3951.1	3729.3	0.2531	0.2681
108		C8	J. Wolfe, UNSW	4186.0		0.2389	



In my case, I decided to choose frequencies between 196 Hz and 783.99 Hz, which corresponds to notes G3 and G5, and also to MIDI numbers 55 and 79, respectively.

I choosed this range because it was, from my point of view, the one that sounds better.

The instrument used for this task is the piano, the same from the previous synthesis.

To end I normalized the signal and I saved the synthesized file in the folder results.

Task 3 - Arbitrary realization of audio signals in MATLAB

For this task I did an UFO sound. To achieve making this sound I first made a pure tone of 50 Hz that has 35 harmonics combined with 2 sounds with a sampling frequency of 48 kHz that has a duration of 10 seconds.

Then I created wind sound with a sampling frequency of 20 kHz.

I used “koupelna.wav” file and I used “konv_reverb” for mixing in one hand, “koupelna.wav” file with the pure tone created at the beginning. On other hand, for mixing “koupelna.wav” too with the wind sound created.

With the second mixing I just took a length of the signal equal to the first signal and I made the sum of both.

At the end I normalized the signal and I saved the synthesized file in the folder results.

References

- Lectures from Multimedia signal synthesis subject
- Additive synthesis: https://en.wikipedia.org/wiki/Additive_synthesis
- Two octaves of the major scale: https://newt.phys.unsw.edu.au/jw/notes.html
- Lectures from Digital Signal Processing