Professorial Inaugural Lecture, 26 April 2001

DIGITAL SYNTHESIS OF MUSICAL SOUNDS

B.T.G. Tan
Department of Physics
National University of Singapore

MUSICAL SYNTHESIS

The many music synthesizers and keyboards available today from manufacturers such as Yamaha can generate musical sounds which are reasonably close to that of musical instruments such as the clarinet and trumpet.

The generation or synthesis of musical sounds aims to reproduce as closely as possible the harmonic structure of the instrument being imitated.

MUSIC AND VIBRATIONS

All musical notes are the result of **vibrations** i.e. an object must vibrate to make the air vibrate, giving rise to a sound wave which reaches our ears. We are able to hear sound vibrations from about **20Hz to 20,000Hz**.

Musical instruments can give rise to a sound wave by:

- **Scraping** a stretched string and causing it to vibrate i.e. **string** instruments.
- **Blowing** into a tube and causing the air column to vibrate - **wind** instruments.
- **Hitting** a solid object and causing it to vibrate - **percussion** instruments.

The nature of the vibration gives each instrument its particular colour or **timbre**.

TIMBRE AND HARMONICS

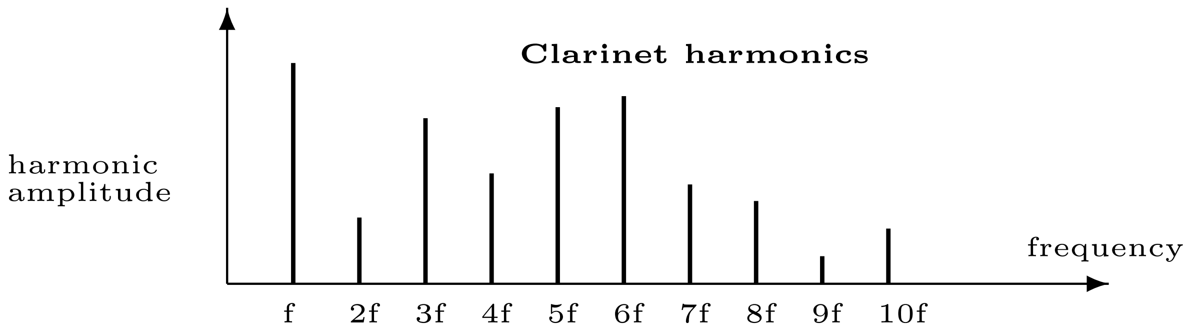
The timbre can be shown to be due to the **harmonics** of the musical sound produced.

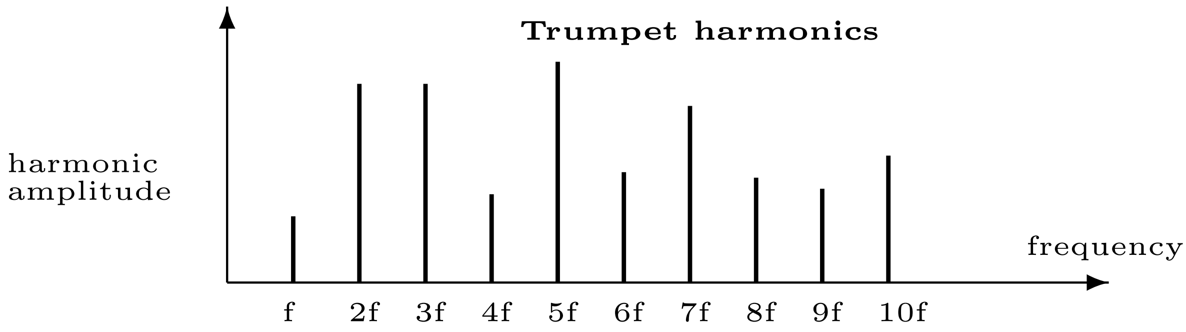
For example, a clarinet playing the note A4 will have a series of harmonics with

1. The first harmonic or **fundamental** at 440 Hz.
2. The second harmonic at double this frequency i.e. 880 Hz.
3. The third harmonic at three times i.e. 1320 nHz, and so on.

It is the relative strength of the harmonics which gives the clarinet its characteristic sound.

A trumpet which is playing the same note will also have its fundamental at 440 Hz, but the relative strength of its harmonics will be different from the clarinet. This gives the trumpet sound a different colour or timbre from the clarinet.





f = fundamental, 2f = 1st harmonic, 3f = 2nd harmonic etc

TYPES OF MUSICAL SYNTHESIS

- **Additive synthesis**: The required harmonics are generated separately and added together.
- **Subtractive synthesis**: A sound rich in harmonics (e.g. white noise) is the starting point and filters are used to subtract the undesired harmonics.
- **FM synthesis**: (invented by John Chowning of Stanford) Two or more waveforms are used, one modulating the frequency of the other, to generate a rich harmonic structure. To obtain the desired harmonics, the frequencies and amplitudes of the two waveforms must be optimized.
- **Sampling/wavetable synthesis**: The waveform of an actual musical instrument is sampled and its shape stored in memory as a set of wavetables.

FM AND RELATED SYNTHESIS TECHNIQUES

In (Frequency Modulation) **FM synthesis**, the frequency of a sinusoidal carrier wave is modulated by another sinusoidal waveform to give a complex waveform which is rich in harmonics.

By suitable choice of the frequencies and degree of modulation, the harmonic structure of the resultant FM waveform can be made to approximate to that of a desired waveform, such as that of a musical instrument.

We consider a carrier waveform of circular frequency ω_c which is frequency modulated by a sinusoidal waveform of $x(t)$ of circular frequency ω_m and amplitude A_m:

$$x(t) = A_m cos(\omega_m t)$$

SYNTHESIS EQUATION FOR FM

The instantaneous frequency f of the carrier wave thus becomes:

$$f = f_c + A_m cos(\omega_m t)$$

The resultant carrier wave is therefore

$$x(t) = A_c sin(\omega_c + 2\pi K_f \int_0^t A_m cos(\omega_m t')dt')$$

i.e.

$$x(t) = A_c sin(\omega_c t + \frac{2\pi K_f A_m}{\omega_m} sin(\omega_m t))$$

If we define the degree of modulation by the modulation index, I as

$$I = \frac{2\pi K_f A_m}{\omega_m} = \frac{A_m K_f}{f_m}$$

we have the basic synthesis equation for FM :

$$x(t) = A \sin[\omega_c + I \sin(\omega_m t)]$$

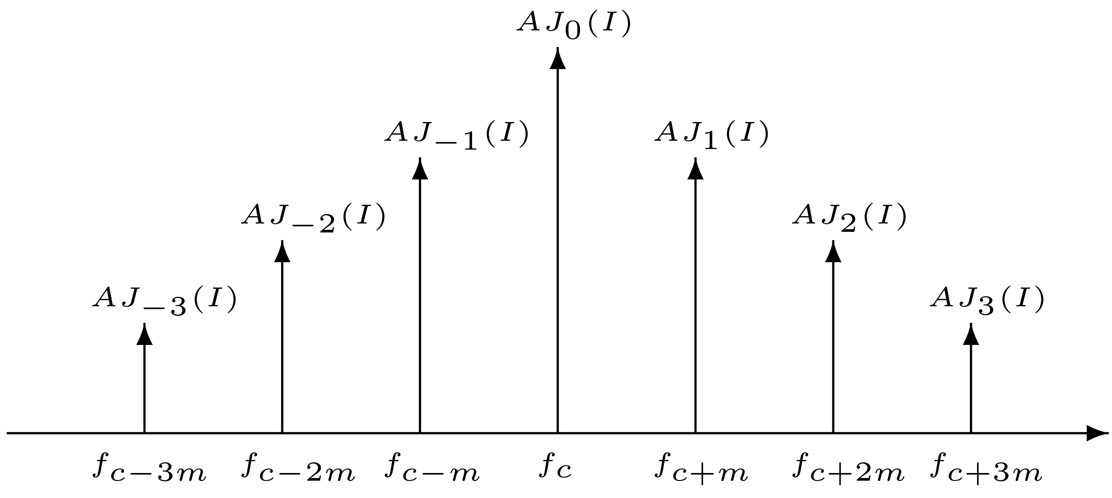
BESSEL FUNCTIONS

$x(t)$ can be expressed as a sum of sinusoids of frequency and amplitude given by

$$x(t) = A \sum_{n=-\infty}^{\infty} J_n(I) \sin(\omega_c + n\omega_m)t$$

where $J_n(I)$ are Bessel Functions.

The FM spectrum thus consists of a central carrier frequency with symmetrical harmonics around it.



REFLECTED HARMONICS

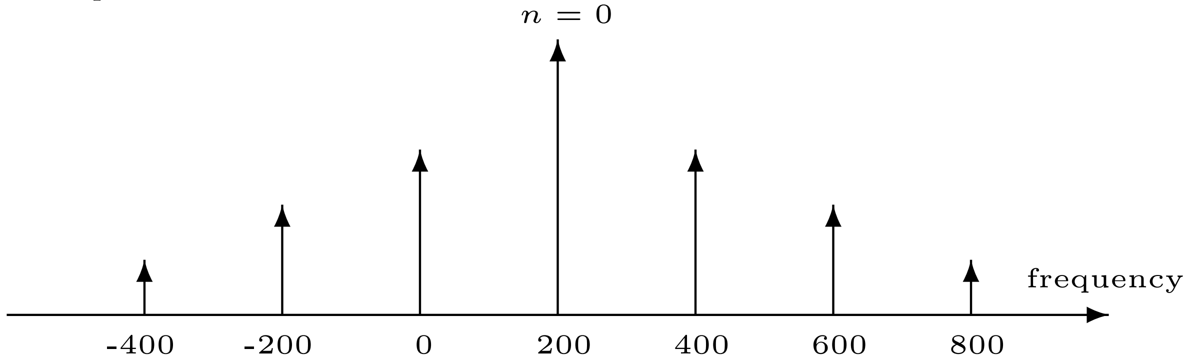
Looking at a typical FM spectrum, as I increases, the number of non-zero harmonics increases. The spectral envelope is mainly dependent on I, but the frequencies of the harmonics and the intervals between them are determined by the ω_c and ω_m.

For example, with $I = 1.0$, $\omega_c = 200$ and $\omega_m = 200$, we obtain a resulting FM spectrum with seven harmonics. The harmonics of negative frequency are reflected from the zero to give harmonics with opposite phase.

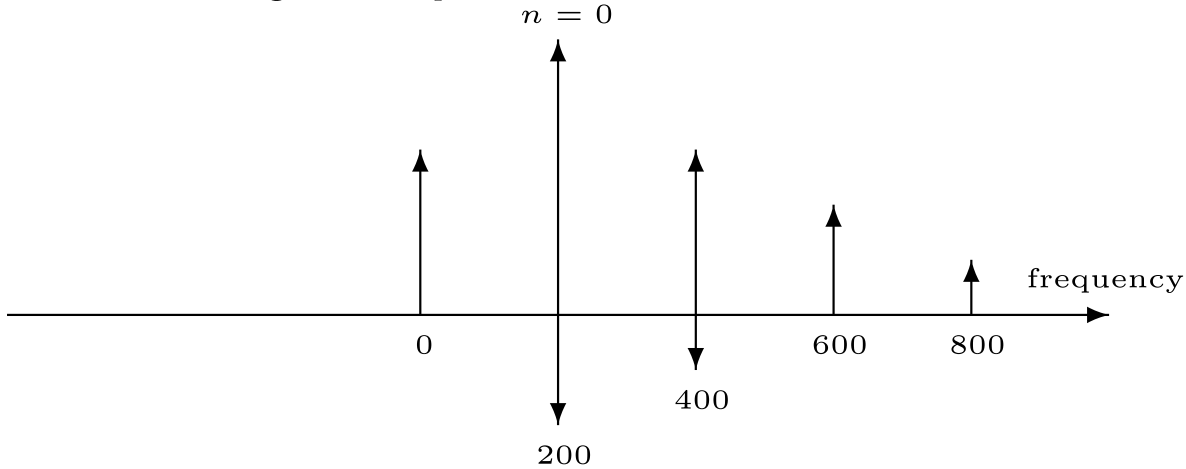
It may happen that some of the reflected harmonics coincide with the unreflected ones. In this example, the -200 Hz harmonic coincides with the +200 Hz harmonic, and the -400 Hz harmonic conicides with the +400 Hz harmonic.

The resultant FM spectrum has five harmonics with the +200 Hz and +400 Hz harmonics reduced in amplitude after reflection.

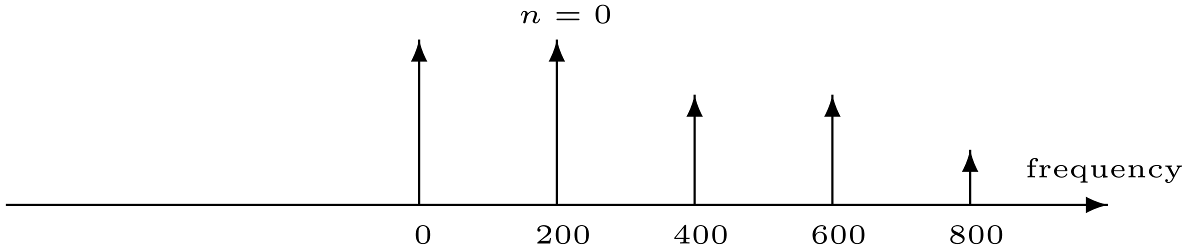
FM spectrum



Reflection of negative frequencies



Resulting FM spectrum



HARMONICITY

If the ratio of the carrier frequency and modulating frequency:

$$\frac{\omega_c}{\omega_m} = \frac{R_c}{R_m}$$

and there is a minimum non-zero $|R_c - nR_m| = 1$ where $n = 1, 2, 3, 4...$

then a harmonic spectrum will be obtained where the harmonics are multiples a fundamental frequency.

Typical examples of such $\frac{R_c}{R_m}$ are:

$\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5} \cdots$

$\frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{6}{7}, \frac{3}{2}, \frac{4}{3}, \frac{5}{2}$ etc

On the other hand if the minimum non-zero $|R_c - nR_m| \neq 1$ where $n = 1, 2, 3, 4...$, then the harmonics will not be multiples of a fundamental frequency.

VARIANTS OF FM SYNTHESIS

Asymmetrical FM or AFM synthesis

A parameter r^n is inserted thus:

$$x(t) = A \sum_{n=-\infty}^{\infty} r^n J_n(I) \sin(\omega_c + n\omega_m)t$$

which gives harmonics which are NOT symmetrical about the carrier frequency.

Hence the harmonics can be more complex and more like actual harmonics of real instruments.

However the synthesis equation is more complex:

$$\begin{aligned} x_c(t) \quad &= \quad A \exp[\frac{I}{2}(r - \frac{1}{r}) \cos(\omega_m t)] \\ & \quad\quad \sin[\omega_c t + \frac{I}{2}(r + \frac{1}{r}) \sin(\omega_m t)] \end{aligned}$$

and hence requires more computation. If $r = 1.0$, AFM is equivalent to FM.

DOUBLE FREQUENCY MODULATION

Instead of one frequency being the carrier and the other the modulator, in **double frequency modulation (DFM)** they have equal status thus:

$$x(t) = A \sin[I_1 \sin(\omega_1 t) + I_2 \sin(\omega_2 t)]$$

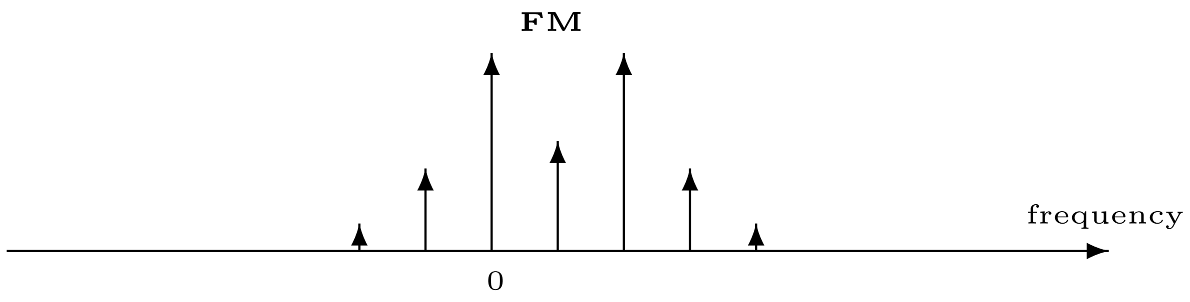
where I_1 and I_2 are the modulation indices of the two frequencies and ω_1 and ω_2 are their respective frequencies.

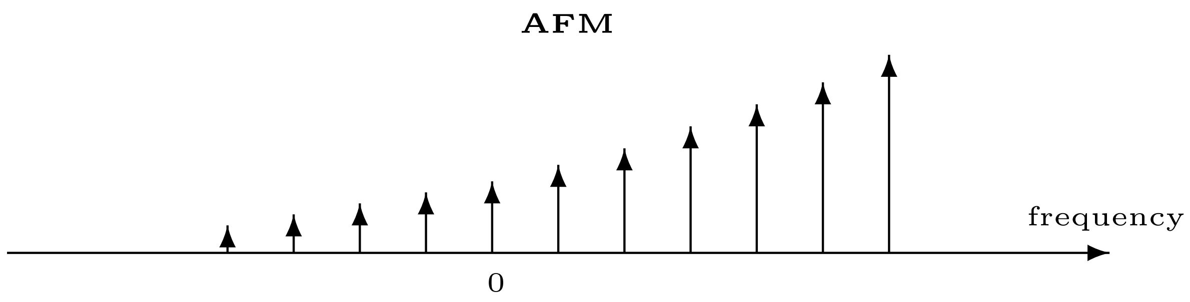
We can show that

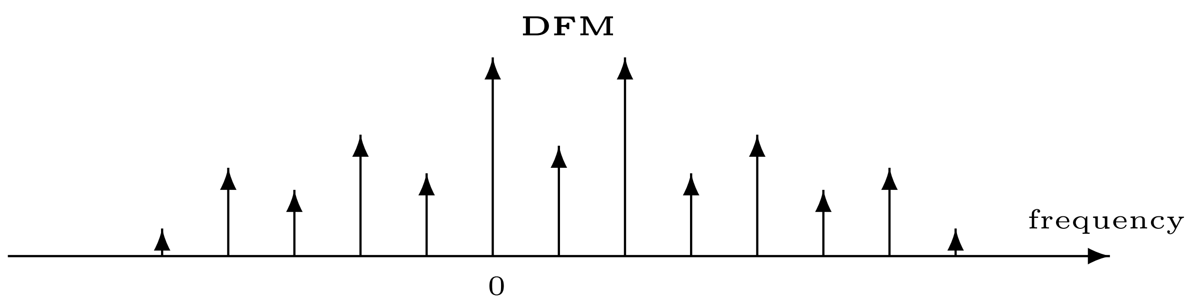
$$x(t) = \sum_p \sum_q J_p(I_1) J_q(I_2) \sin(p\omega_1 t + q\omega_2 t)$$

where p and q are integers of different parity.

DFM thus generates harmonics of angular frequency $p\omega_1 + q\omega_2$. It also generates many more significant harmonics than FM, due to the harmonics depending on product of the *two* Bessel functions instead of the single Bessel function of FM.


FM
frequency
0


AFM
frequency
0


DFM
frequency
0

COMPARISON OF AFM AND DFM

DFM is able to generate harmonics of greater complexity than FM, but with much less computational load than AFM.

In AFM there are

- two sine functions
- one cosine function
- one exponential function
- five multiplication
- two divisions
- two additions
- one subtraction.

In DFM we have only

- three sine functions
- two multiplications
- one addition

OPTIMISATION OF FM PARAMETERS

For FM synthesis, the parameters for the carrier and modulator frequencies i.e.

ω_c, ω_m and I

have to be optimized for the closest fit to the harmonics of the instrument to be imitated.

For AFM synthesis, the parameters

ω_c, ω_m, I and r

have to be optimized.

For DFM synthesis, the parameters for the two frequencies,

I_1, I_2, ω_1 and ω_2

have to be optimized.

This has normally been done by trial and error, which is a tedious and lengthy process.

FM, AFM, DFM SOLUTION SPACES

For FM, AFM, and DFM synthesis, the relevant parameters which determine the spectrum of the synthesized waveform can be evaluated by a **fitness** parameter.

The smaller the fitness for a given set of parameters, the closer is the resultant spectrum to the desired one.

The fitness can be plotted as a function of the relevant parameters in a **solution space**. The lowest valley of this space determines the set of parameters which generate the waveform having a spectrum most closely resembling that of the musical instrument to be synthesized.

The objective of optimization is thus to find the set of parameters which gives the best fitness value.

REAL MUSICAL INSTRUMENT SAMPLES

The real musical instrument samples were obtained from the standard **McGill University Master Samples** (MUMS) CD. This CD contains recordings of musical instruments playing notes at standard frequencies such as A=440 Hz.

Selected recordings were analyzed by dividing each note into time frames of 31.25 milliseconds. This time frame was then sampled and fed into a 1024 point FFT program.

For a single time frame, the plot of the amplitudes of the harmonics versus their frequency shows the envelope or shape of the harmonic spectrum.

By plotting the harmonic amplitudes for all the time frames successively, it is possible to obtain a plot of the spectrum as it changes through the duration of the musical note.

FM SOLUTION SPACE

Having obtained a plot of the harmonics for a particular time frame, we can then attempt to synthesize the musical instrument tone using FM, AFM or DFM synthesis to replicate the harmonic spectrum as closely as possible.

For example, for FM synthesis, we can vary three parameters: ω_c, ω_m and I. For convenience, we define $f_c = \frac{\omega_c}{2\pi}$ and $f_m = \frac{\omega_m}{2\pi}$.

In practice, for a tone at A=440 Hz, we fix $f_c = 440$ and $f_m = 440$. We then need to find the value of I for which the fitness of the synthesized note is at an optimum value.

The nature of the optimization problem can be more clearly shown by plotting the value of the fitness for a range of values of I. For example, for the trumpet tone, it can be seen from such a plot that the minimum fitness occurs at $I = 4.65$.

AFM AND DFM SOLUTION SPACES

Similar solution spaces can be plotted for AFM and DFM synthesis.

We consider the same trumpet tone at A = 440 Hz.

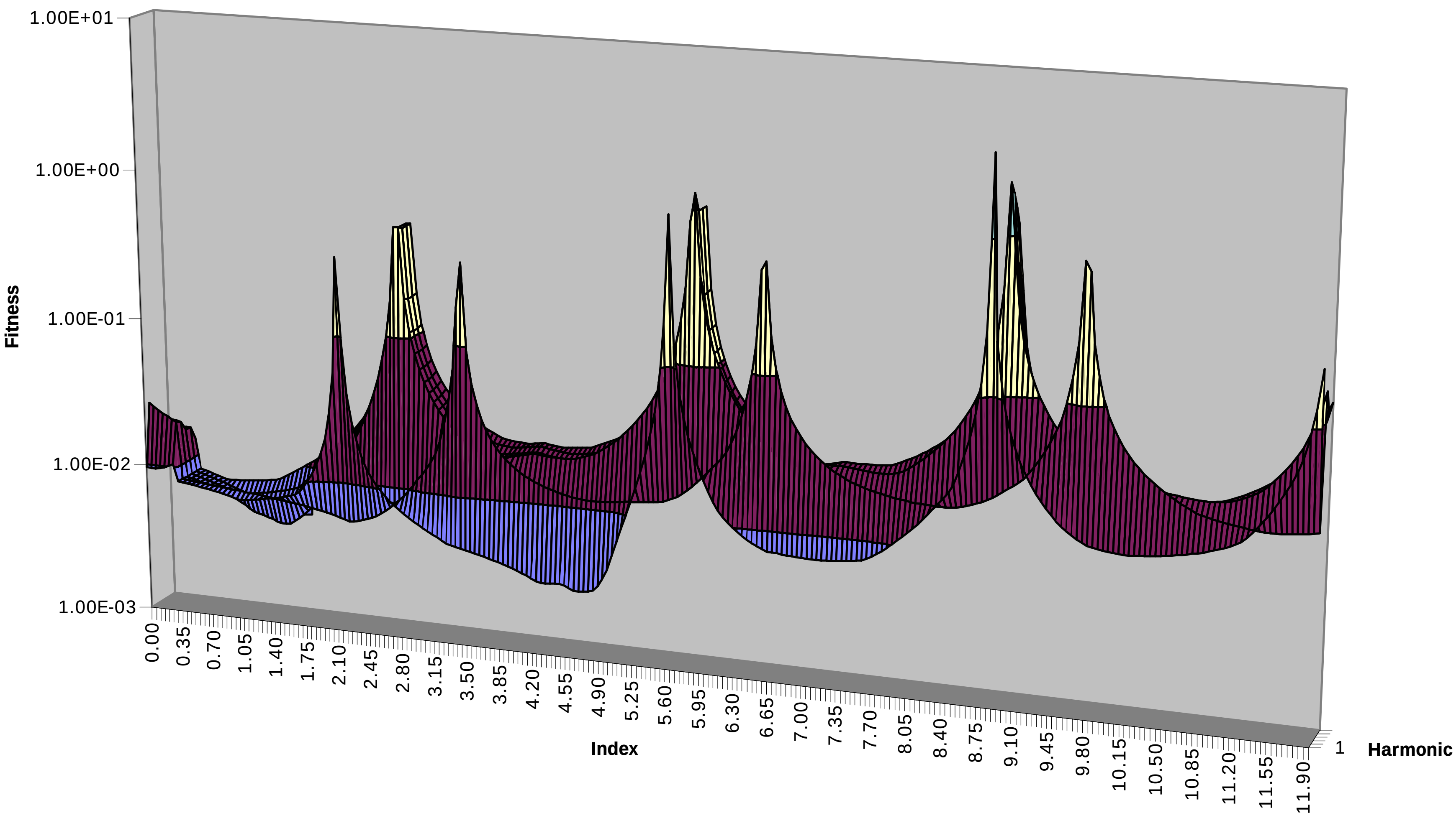
For AFM synthesis, we can fix $f_1 = 440$ and $f_2 = 880$. The solution space is then a plot of the fitness value for varying I and r. The minimum fitness occurs at $I = 0.55$ and $r = 2.10$.

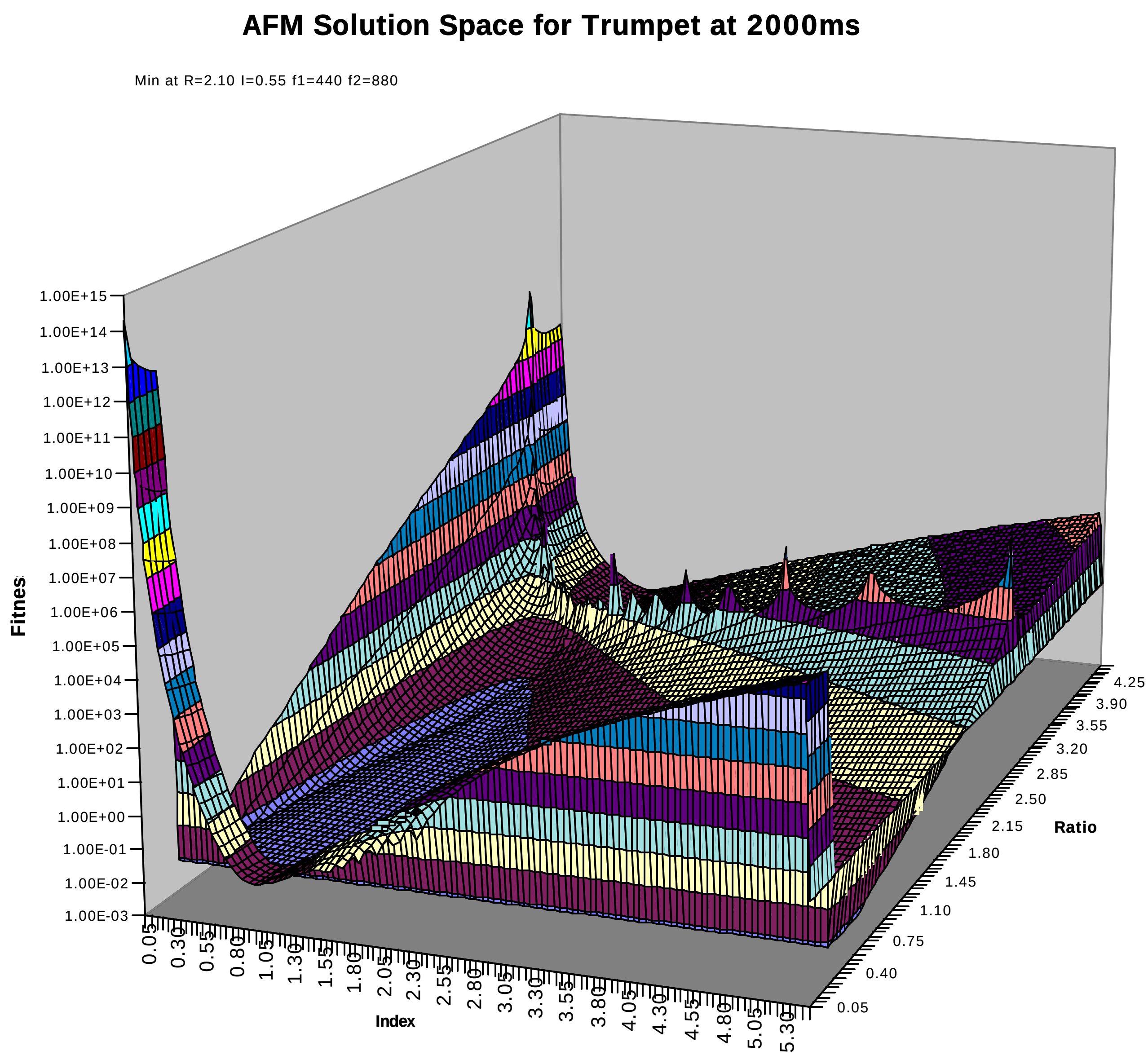
Likewise, for DFM synthesis, we fix $f_1 = 440$ and $f_2 = 880$ to plot the fitness against I_1 and I_2. From the solution space, the minimum fitness occurs at $I_1 = 3.05$ and $I_2 = 1.40$

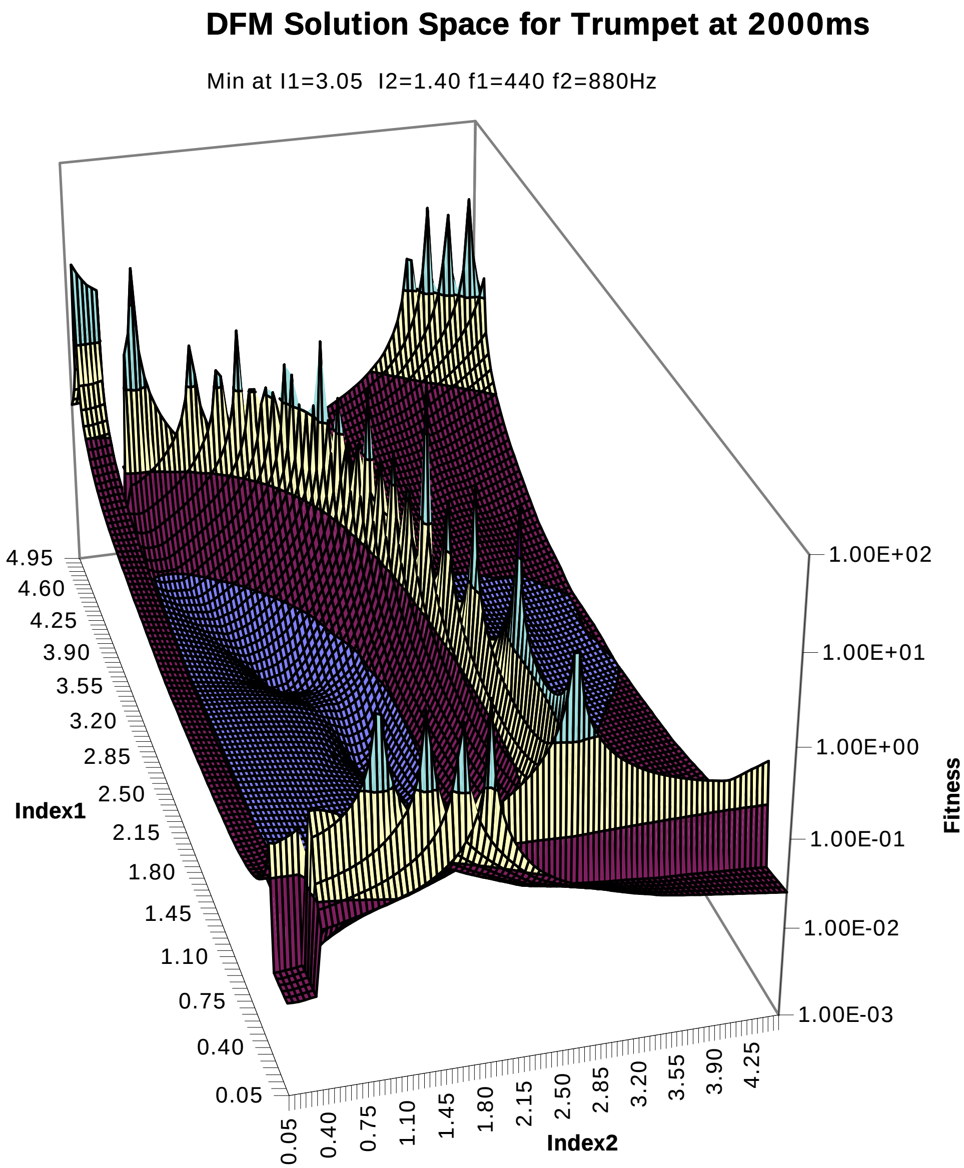
The FM, AFM and DFM all employ a single operator. By employing two or more operators, it is possible to obtain solution spaces with minimum fitness of even lower values.

FM Solution Space for Trumpet at 2000ms

Min at Index = 4.65 & freq2 = 440Hz when Freq1 = 440Hz




AFM Solution Space for Trumpet at 2000ms
Min at R=2.10 I=0.55 f1=440 f2=880
Fitnes
1.00E+15
1.00E+14
1.00E+13
1.00E+12
1.00E+11
1.00E+10
1.00E+09
1.00E+08
1.00E+07
1.00E+06
1.00E+05
1.00E+04
1.00E+03
1.00E+02
1.00E+01
1.00E+00
1.00E-01
1.00E-02
1.00E-03
Index
0.05 0.30 0.55 0.80 1.05 1.30 1.55 1.80 2.05 2.30 2.55 2.80 3.05 3.30 3.55 3.80 4.05 4.30 4.55 4.80 5.05 5.30
Ratio
0.05 0.40 0.75 1.10 1.45 1.80 2.15 2.50 2.85 3.20 3.55 3.90 4.25



DFM Solution Space for Trumpet at 2000ms
Min at I1=3.05 I2=1.40 f1=440 f2=880Hz
Index1
4.95
4.60
4.25
3.90
3.55
3.20
2.85
2.50
2.15
1.80
1.45
1.10
0.75
0.40
0.05
Index2
0.05
0.40
0.75
1.10
1.45
1.80
2.15
2.50
2.85
3.20
3.55
3.90
4.25
Fitness
1.00E+02
1.00E+01
1.00E+00
1.00E-01
1.00E-02
1.00E-03

SOLUTION SPACES FOR THE VIOLIN

For comparison we have also plotted solution spaces for the violin for FM, AFM and DFM (for single operators).

FM synthesis:

$f_c = f_m = 440$

Minimum fitness occurs at $I = 0.8$

AFM synthesis:

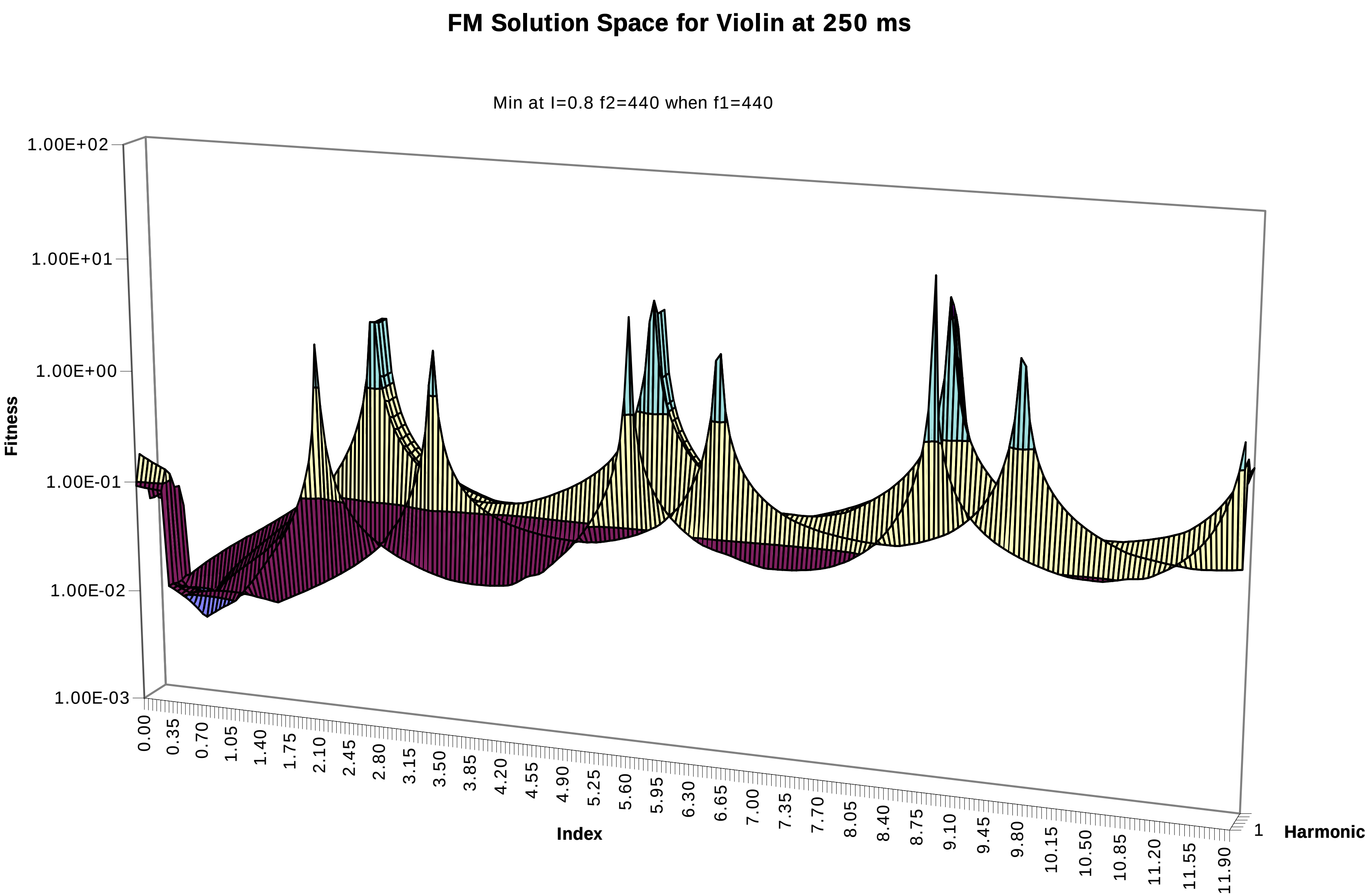
$f_c = f_m = 440$

Minimum fitness occurs at $I = 0.1$ and $r = 0.51$

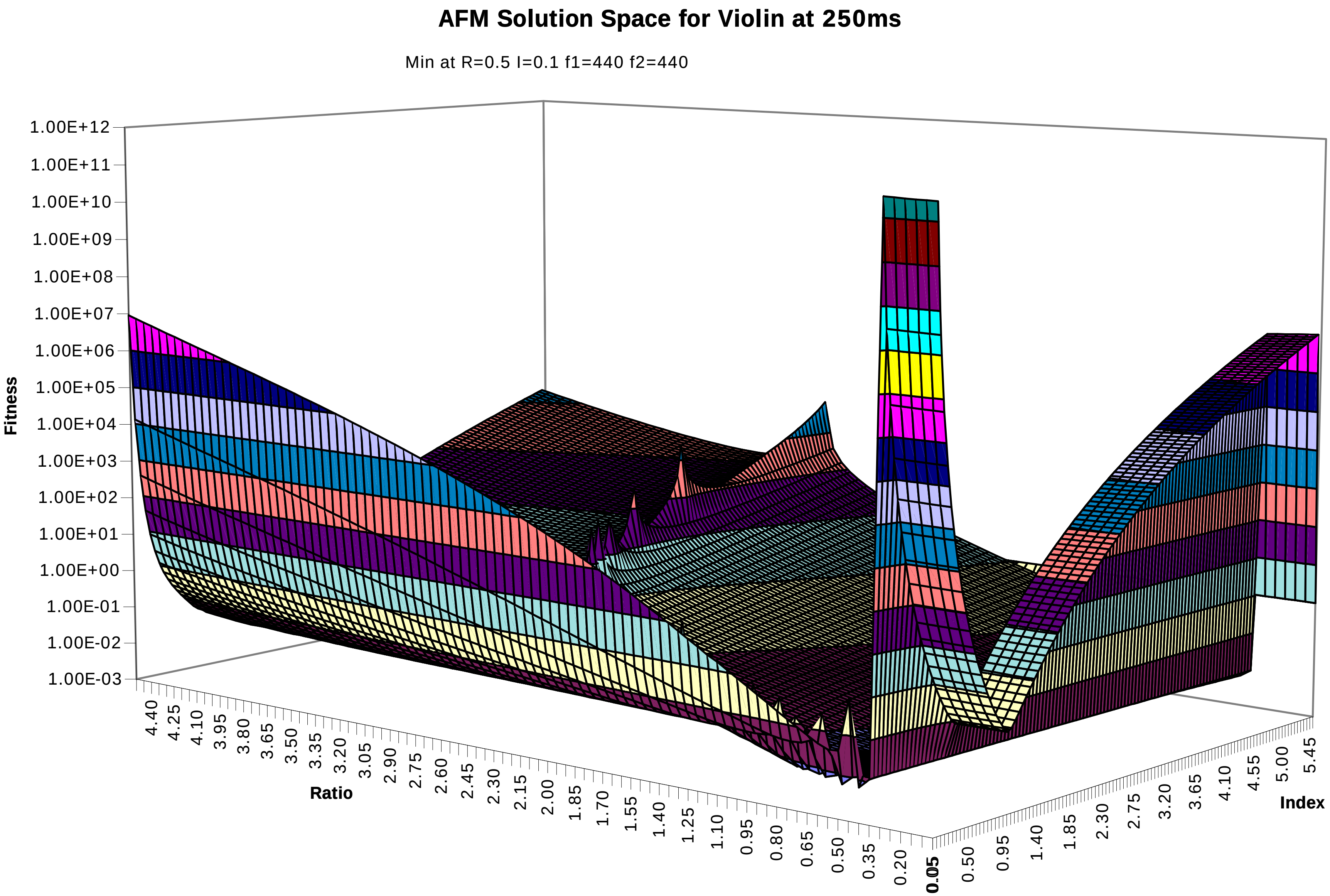
DFM synthesis:

$f_1 = 440$ and $f_2 = 880$

Minimum fitness occurs at $I_1 = 1.45$ and $I_2 = 0.65$

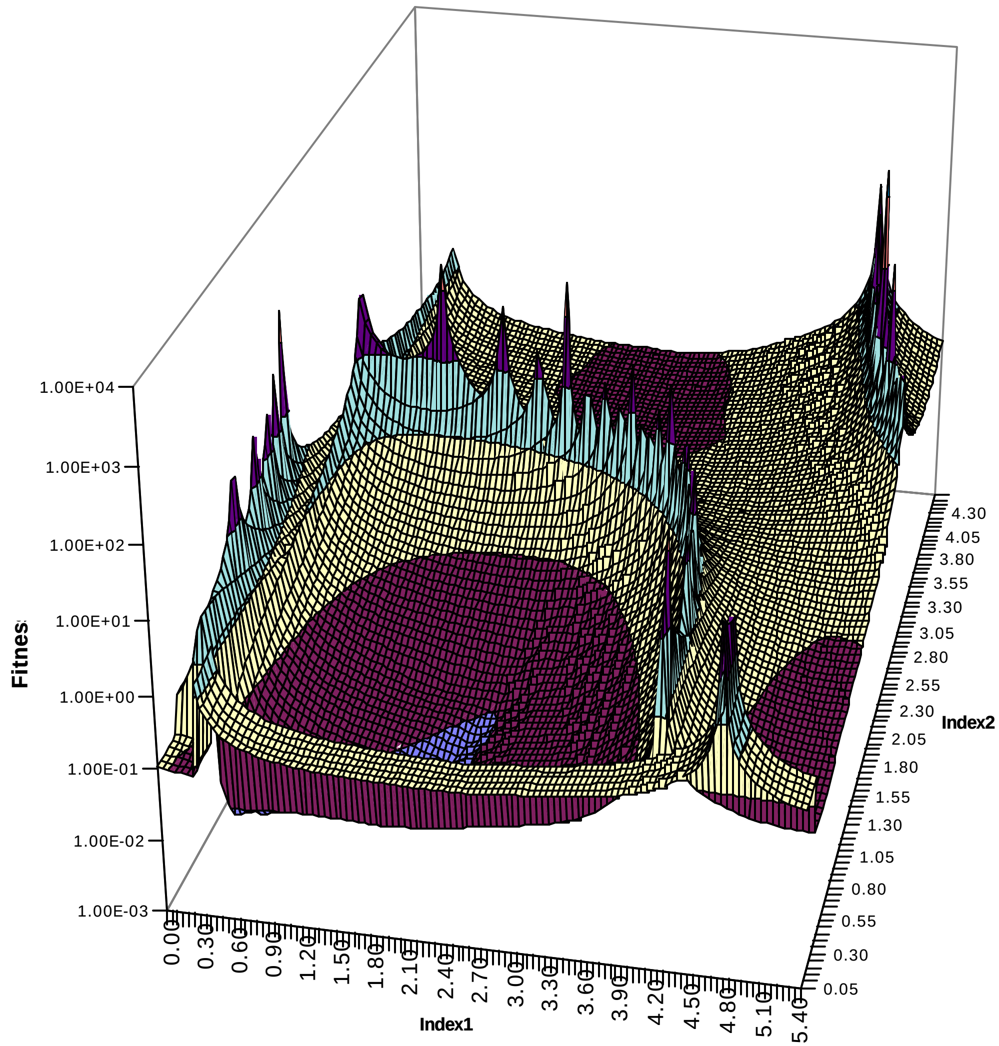


FM Solution Space for Violin at 250 ms
Min at I=0.8 f2=440 when f1=440
Fitness
1.00E+02
1.00E+01
1.00E+00
1.00E-01
1.00E-02
1.00E-03
Index
0.00
0.35
0.70
1.05
1.40
1.75
2.10
2.45
2.80
3.15
3.50
3.85
4.20
4.55
4.90
5.25
5.60
5.95
6.30
6.65
7.00
7.35
7.70
8.05
8.40
8.75
9.10
9.45
9.80
10.15
10.50
10.85
11.20
11.55
11.90
1
Harmonic


AFM Solution Space for Violin at 250ms
Min at R=0.5 I=0.1 f1=440 f2=440
Fitness
1.00E+12
1.00E+11
1.00E+10
1.00E+09
1.00E+08
1.00E+07
1.00E+06
1.00E+05
1.00E+04
1.00E+03
1.00E+02
1.00E+01
1.00E+00
1.00E-01
1.00E-02
1.00E-03
Ratio
4.40
4.25
4.10
3.95
3.80
3.65
3.50
3.35
3.20
3.05
2.90
2.75
2.60
2.45
2.30
2.15
2.00
1.85
1.70
1.55
1.40
1.25
1.10
0.95
0.80
0.65
0.50
0.35
0.20
0.05
Index
0.05
0.50
0.95
1.40
1.85
2.30
2.75
3.20
3.65
4.10
4.55
5.00
5.45

DFM Solution Space for violin at 250 ms

Min at I1=1.45, I2=0.65, f1=440, f2=880



OPTIMIZATION TECHNIQUES

It is difficult to do obtain the best set of parameters analytically, and traditionally this has been done manually, which is a tedious and lengthy process.

We have speeded up the optimization process considerably by searching the solution space for the optimum set of parameters using various optimization techniques to search the solution space:

- **Genetic Algorithm**
- **Simulated Annealing**
- Combination of genetic algorithm and simulated annealing: **Genetic Annealing**
- **Tree Evolution Algorithm**

This has resulted not only in considerable speeding up, but in obtaining more accurate parameters i.e. better optimized harmonics and waveforms synthesized.

GENETIC ALGORITHM (GA)

By starting with an initial set of points randomly spread in solution space, the genetic algorithm processes each ”generation” of points using a technique similar to that in natural selection to arrive at solutions which are better than the previous generation.

The genetic algorithm consists of 4 major processes:

1. Recruitment process to form initial population.
2. Selection processes to select fittest members of population.
3. Crossover process on selected population; pairs of selected individuals to produce offspring for the next generation.
4. Mutation process to produce occasional changes in population.

SIMULATED ANNEALING

This is a probabilistic optimization algorithm based on annealing process in solids.

In annealing, a solid in a disordered state at a high temperature is allowed to cool down to a highly ordered state in stages, each time lowering its energy state.

For a given state c with an energy $E(c)$, the probability of its being in that state is given by the Boltzmann distribution:

$$B(c) = \frac{1}{Z(T)} \exp(\frac{E(c)}{k_B T})$$

where T is the temperature, k_B is the Boltzmann constant and $Z(T)$ is the partition function defined by

$$Z(T) = \sum_{allc} \exp(\frac{E(c)}{k_B T})$$

STATE TRANSITION

When we have a transition on cooling from a state c_{t-1} to the next state c_t, whether the new state is accepted to replace the previous state depends on the ratio P:

$$P = \frac{B(c_t)}{B(c_{t-1})}$$

$$P = \exp(\frac{-\partial E}{k_B T})$$

where ∂E is the difference between the two energy states.

Hence if the new state has a lower energy than the previous state, it will be accepted with a probability of one, while if it is higher, the probability is determined by the difference in energies.

SIMULATED ANNEALING PROCESS

The simulated annealing process starts with an initial state or position in the solution space:

1. Choose a sequence (T_k, t_k) starting with $k = 0$ and an initial state.
2. Perturb this state to a neighbouring state.
3. Compare the energy of the new state with the old state. If the new state Has lower energy, keep it. Otherwise it is kept only in accordance with Probability defined above using the Boltzmann distribution.
4. Repeat 2 and 3 t_k times.
5. Increase k by one and repeat steps 2,3 and 4 K times

At the end of the process, we arrive at a final cooled state at temperature T_k for which the energy is in the lowest possible state.

GA AND SIMULATED ANNEALING COMPARED

The genetic algorithm is able to search the whole of the solution space quite effectively, but is less good at converging to an optimum solution.

The simulated annealing algorithm is good at converging to a minimum once it finds one, but is less good at searching the entire solution space.

The dependence of the probability function on the temperature makes it less likely that the search can move from a less optimum minimum over a large barrier to a better minimum as the temperature is lowered.

GENETIC ANNEALING ALGORITHM (GAA)

The Genetic Annealing algorithm combines the best features of the genetic algorithm with simulated annealing, using the Genetic Algorithm as a basis.

Its main feature is the Anneal_Cross crossover process in which one parent is the fittest individual in the population and is mated with another parent selected randomly. The offspring will replace the parents with a simulated annealing-like algorithm: they replace the parents if they are fitter, and if not, only according to a probability determined by the Boltzmann function.

This ensures that the solutions are still able, like the Genetic Algorithm, to explore the solution space, but that they converge rapidly once a minimum is found.

GAA PROCESS

1. Recruitment process to set up a set of initial states.
2. Choose a sequence (T_k, t_k) starting with $k = 0$ and select the best state.
3. Do crossover of this state with a random partner.
4. Compare the energy of the offspring with the fittest parent and keep it in accordance with the simulated annealing. algorithm.
5. Repeat 2 and 3 t_k times.
6. Increase k by one, select new random partner and repeat steps 2,3 and 4 K times.

In effect, a complete simulated annealing process is incorporated into the crossover process of the genetic algorithm. At the end of the process, we arrive at a final cooled state at temperature T_k for which the energy is in the lowest possible state.

COMPARISON OF GAA AND GA FOR DFM

GAA and GA were used to optimize the DFM parameters of a number of synthesized tones of real instruments. Using the real instrument recordings on the standard McGill University Master Samples (MUMS) CD, the harmonics of the sampled waveforms were analyzed using FFT.

Using a one-operator DFM algorithm, the ω_c and ω_m were kept fixed while the I_c and I_m were varied.

For each instrument, the solution space was plotted so that the actual optimum minumum could be found.

GAA and GA were then applied to the DFM parameters to obtain the closest set of parameters to the optimum values.

GAA AND GA COMPARISON RESULTS

	Minimum		GAA		
Instrument	I_1	I_2	Fitness	I_1	I_2
Violin	1.45	0.65	0.260	1.46	0.66
Saxophone	4.95	1.35	0.533	4.99	1.37
Piano	0.65	0.40	0.011	0.60	0.39
Oboe	3.50	1.50	0.406	3.40	1.54
Trumpet	3.05	1.40	0.312	3.07	1.38

	Minimum		GA		
Instrument	I_1	I_2	Fitness	I_1	I_2
Violin	1.45	0.65	0.306	0.88	0.46
Saxophone	4.95	1.35	0.557	5.01	1.45
Piano	0.65	0.40	0.050	1.03	0.53
Oboe	3.50	1.50	0.457	3.57	1.34
Trumpet	3.05	1.40	0.331	2.22	1.82

TWO OPERATOR DFM

We define one DFM operator as:

$$x(t) = A\sin[I_1 \sin(\omega_1 t) + I_2 \sin(\omega_2 t)]$$

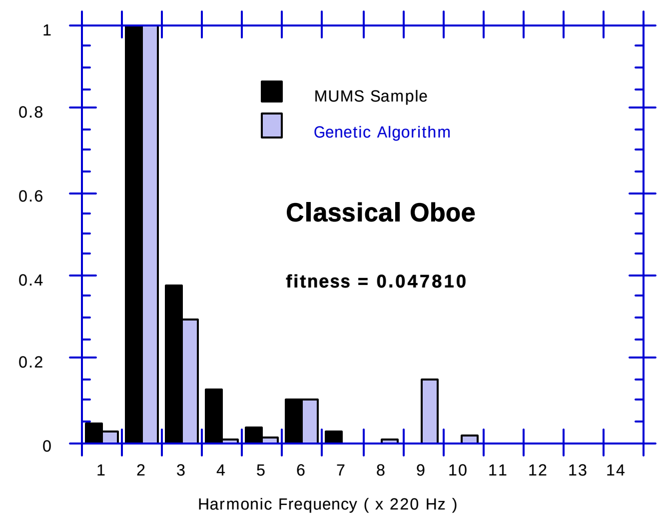
One or more DFM operators may be used to synthesize an instrument.

For two DFM operators, A and B, we may weight the operators accordingly with the weights W_A and W_B before adding the two operators together to obtain the resultant waveform $X(t)$:

$$\begin{aligned} X(t) &= W_A \sin[I_{1A} \sin(\omega_{1A}) + I_{2A} \sin(\omega_{2A})] \\ &+ W_B \sin[I_{1B} \sin(\omega_{1B}) + I_{2B} \sin(\omega_{2B})] \end{aligned}$$

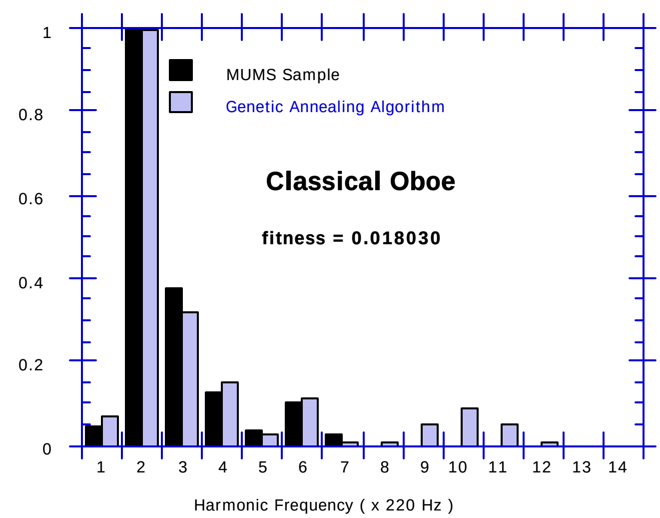
This can be extended to three or more DFM operators.

DFM OPERATOR A			DFM OPERATOR B		
AmpA	=	0.648809	AmpB	=	1.849750
IndexA1	=	2.605	IndexB1	=	1.437
FreqA1	=	660 Hz	FreqB1	=	440 Hz
IndexA2	=	0.097	IndexB2	=	0.037
FreqA2	=	880 Hz	FreqB2	=	220 Hz



Classical Oboe. DFM parameters estimated by the genetic algorithm.

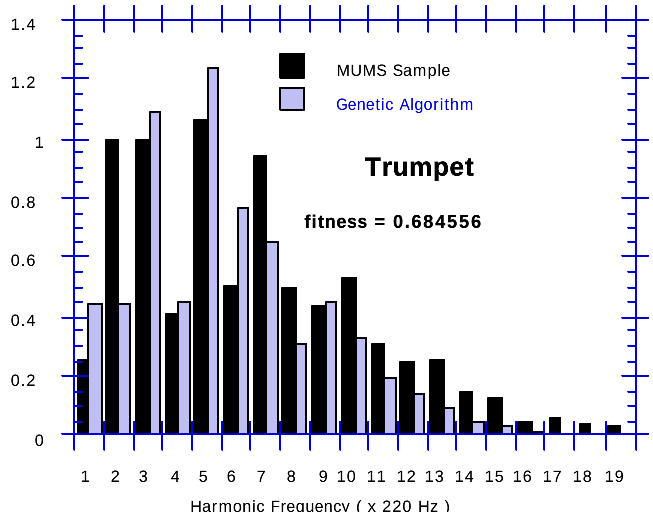
DFM OPERATOR A			DFM OPERATOR B		
AmpA	=	0.776809	AmpB	=	1.977013
IndexA1	=	1.618000	IndexB1	=	1.440000
FreqA1	=	660 Hz	FreqB1	=	440 Hz
IndexA2	=	0.961000	IndexB2	=	0.118000
FreqA2	=	880 Hz	FreqB2	=	220 Hz



Classical Oboe. DFM parameters estimated by the genetic annealing algorithm.

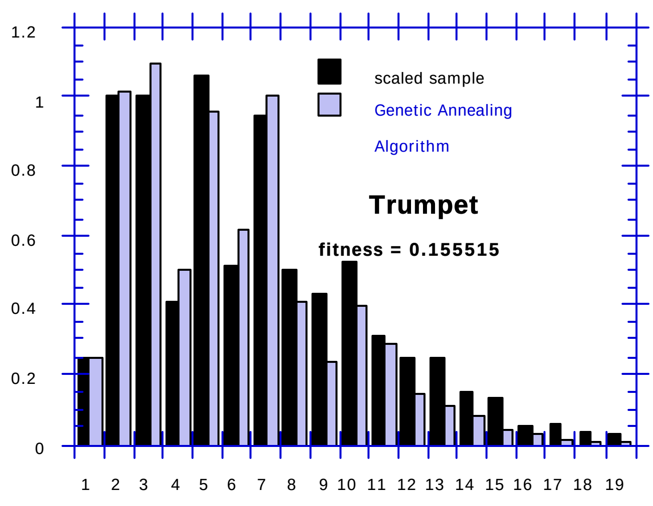
.

DFM OPERATOR A	DFM OPERATOR B	DFM OPERATOR C
AmpA = 2.948724	AmpB = 3.374904	AmpC = 5.366608
IndexA1 = 2.503000	IndexB1 = 0.019000	IndexC1 = 1.535000
FreqA1 = 440 Hz	FreqB1 = 440 Hz	FreqC1 = 440 Hz
IndexA2 = 5.722000	IndexB2 = 5.238000	IndexC2 = 1.939695
FreqA2 = 220 Hz	FreqB2 = 220 Hz	FreqC2 = 220 Hz



Trumpet. DFM parameters estimated by the genetic algorithm.

DFM OPERATOR A	DFM OPERATOR B	DFM OPERATOR C
AmpA = 5.262182	AmpB =2.815760	AmpC = 4.787540
IndexA1 = 1.390000	IndexB1 = 4.835000	IndexC1 = 5.246000
FreqA1 = 220 Hz	FreqB1 = 220 Hz	FreqC1 = 220 Hz
IndexA2 = 1.288000	IndexB2 = 3.326000	IndexC2 = 0.479000
FreqA2 = 440 Hz	FreqB2 = 440 Hz	FreqC2 = 440 Hz



Trumpet. DFM parameters estimated by the genetic annealing algorithm.

COMPARISON OF TWO OPERATOR FM, AFM AND DFM

Though Yamaha's famous DX7 synthesizer is described as using 6 operator FM synthesis, one carrier and one modulating waveform are counted by Yamaha as 2 operators, hence it is 3 operator synthesis in our terminology.

Using GAA to optimize 2 operator FM, AFM and DFM synthesis for a number of instruments gave the following results:

Instrument	FM	AFM	DFM
Trumpet	0.001426	0.000935	0.000658
Saxophone	0.001061	0.000831	0.000280
Oboe	0.001894	0.000887	0.001904
Piano	0.000440	0.000112	0.000088
Violin	0.003970	0.005517	0.002895
Cornet	0.003980	0.002886	0.004440

DYNAMIC SYNTHESIS AND RECYCLING

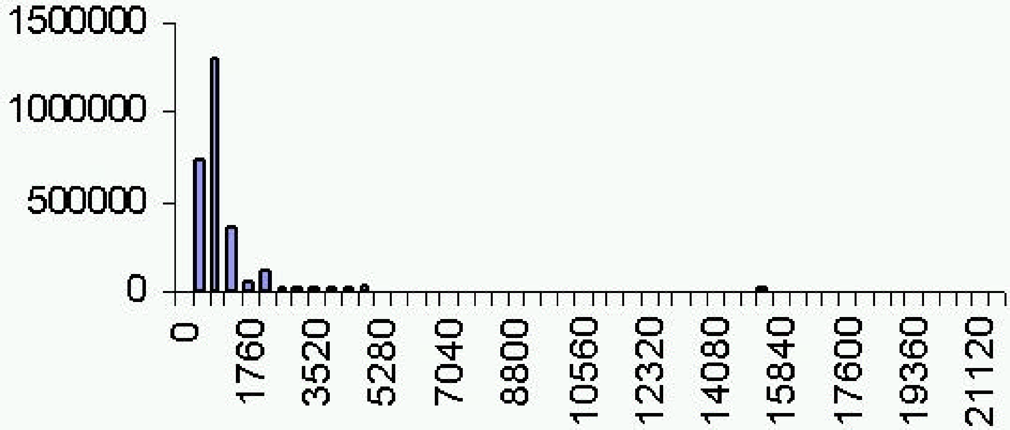
In order to synthesize dynamic sounds, Each set of optimum parameters for a time frame is **recycled** as the initial set of parameters for the recruitment phase for the next time frame. As the spectrum does not vary much from the last time frame, this enables the next time frame to start with a set of near-optimized parameters.

This enables the dynamic sound spectrum to be synthesized very efficiently.

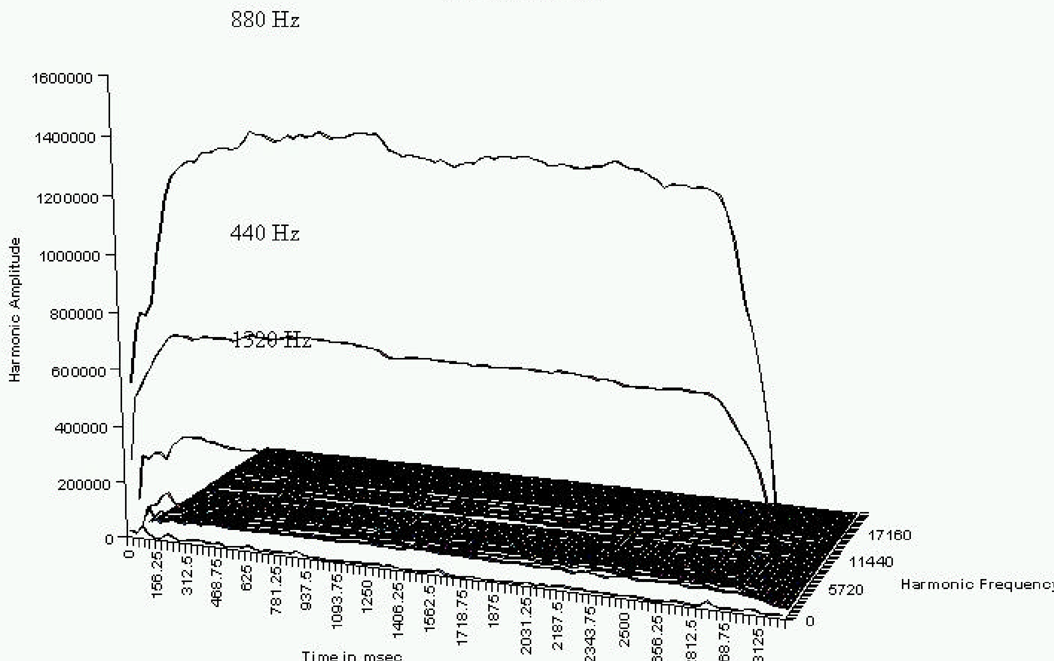
As the GAA process is very fast, typically about 7 seconds for the initial time frame and about 2 seconds for each subsequent time frame, it is conceivable that with faster computers the synthesis process could proceed in real time.

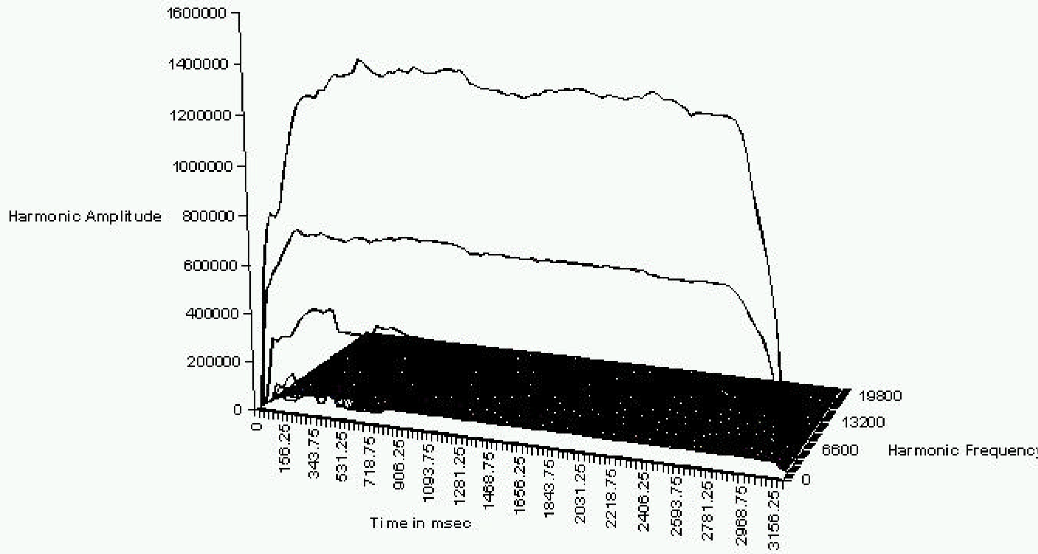
DFM could then be used as a highly efficient method of data compression for the storage and transmission of the synthesized waveform, as only the DFM parameters need be stored or transmitted.

Typical Trumpet Spectrum

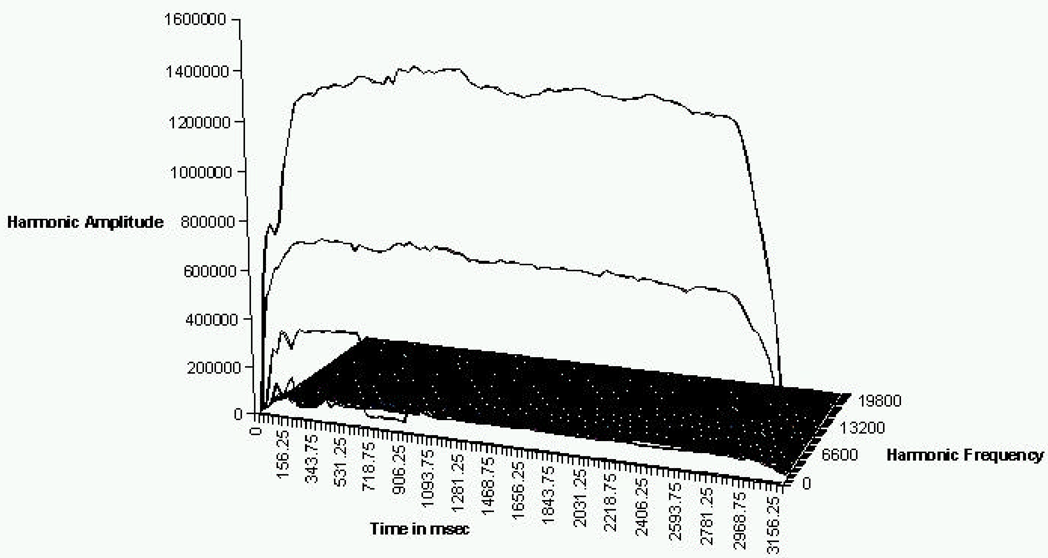


Sample Trumpet

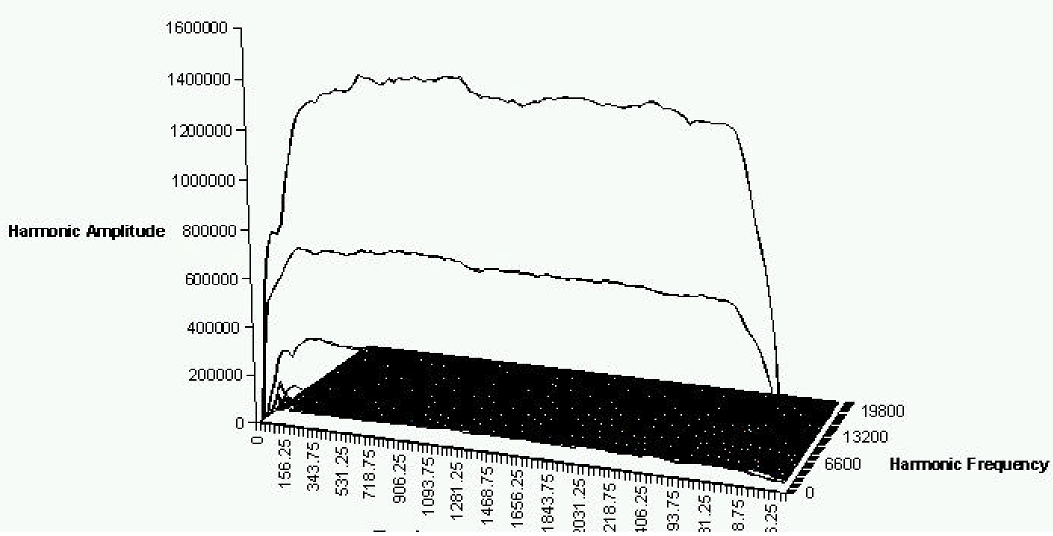


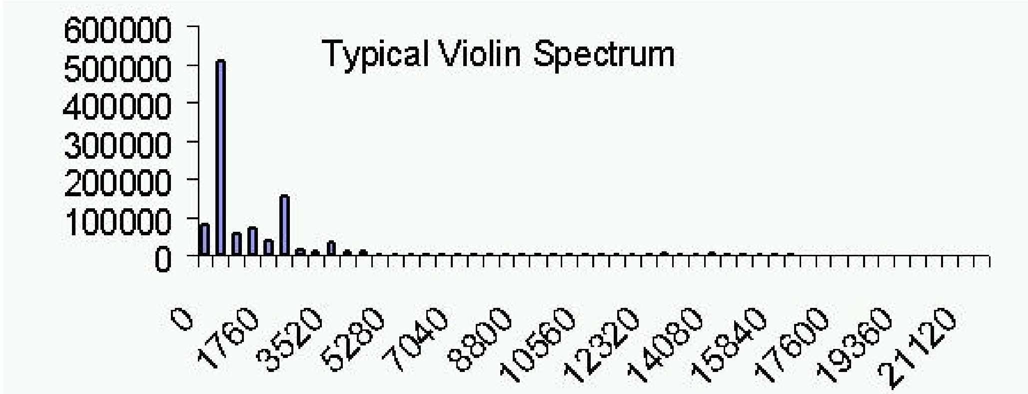


AFM Synthesized Trumpet by GAA

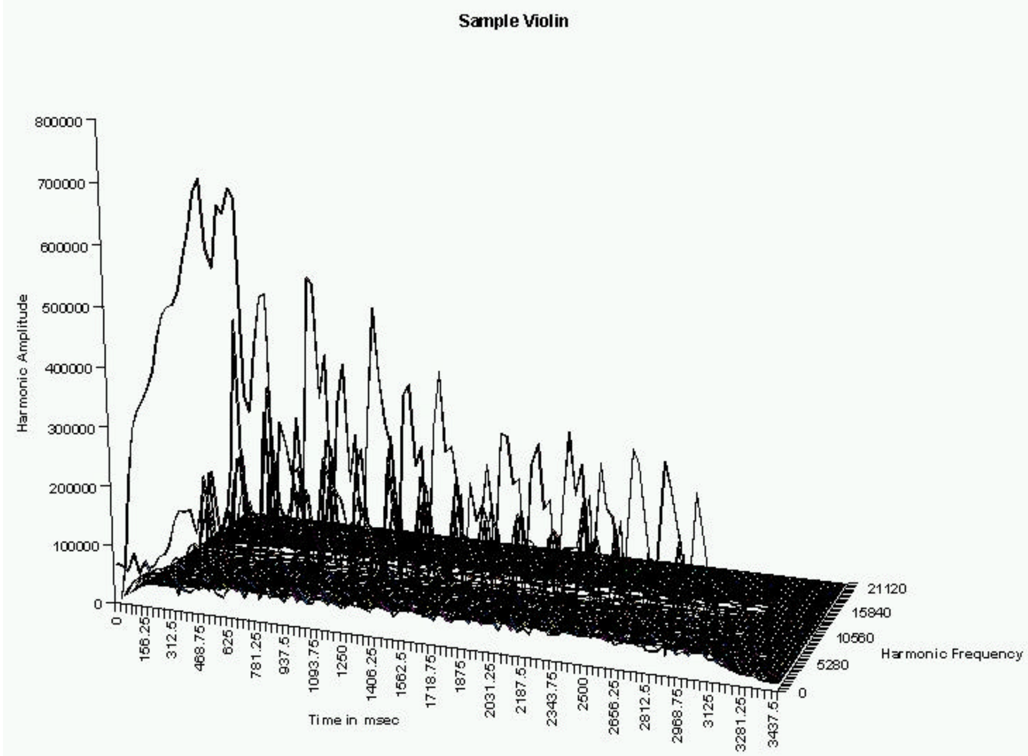


DFM Synthesized Trumpet by GAA

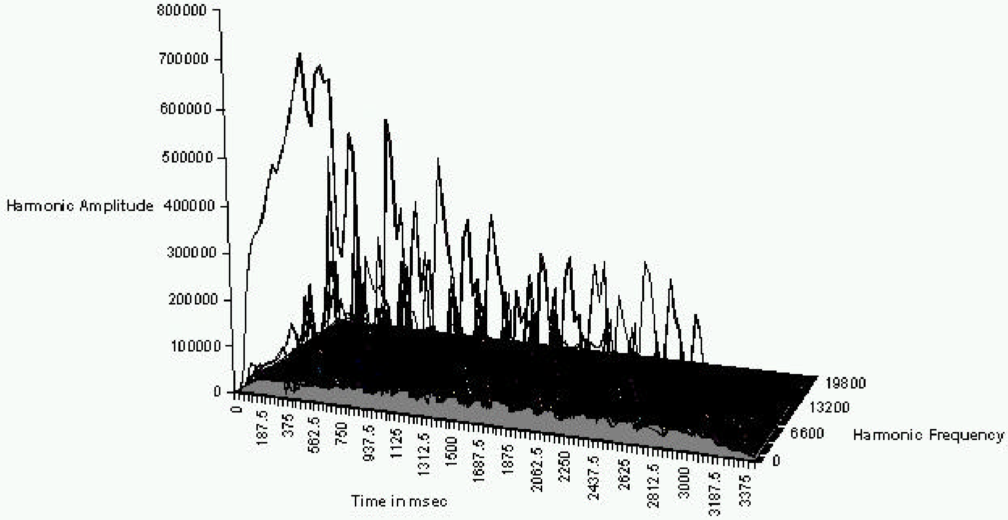




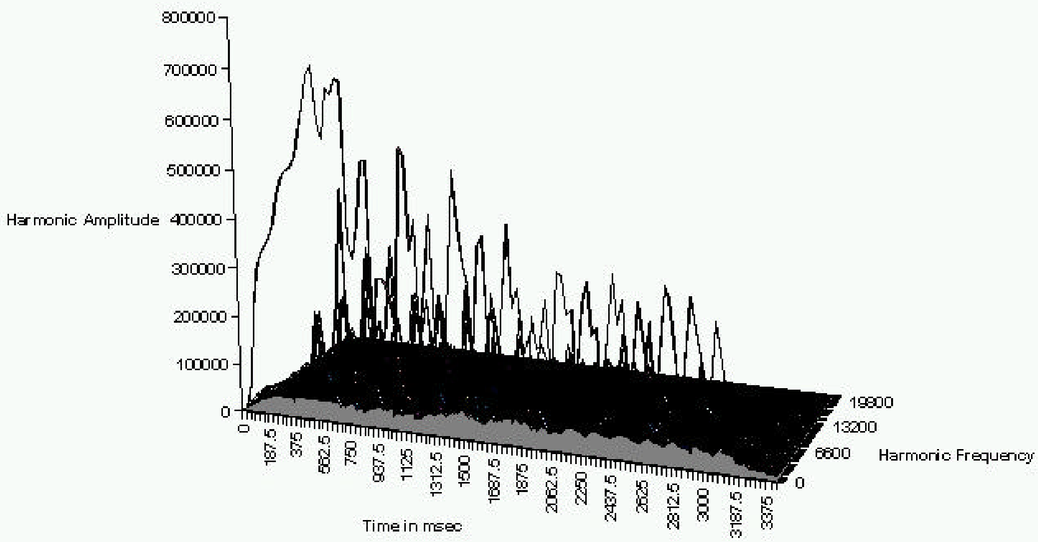
Typical Violin Spectrum
600000
500000
400000
300000
200000
100000
0
0
1760
3520
5280
7040
8800
10560
12320
14080
15840
17600
19360
21120



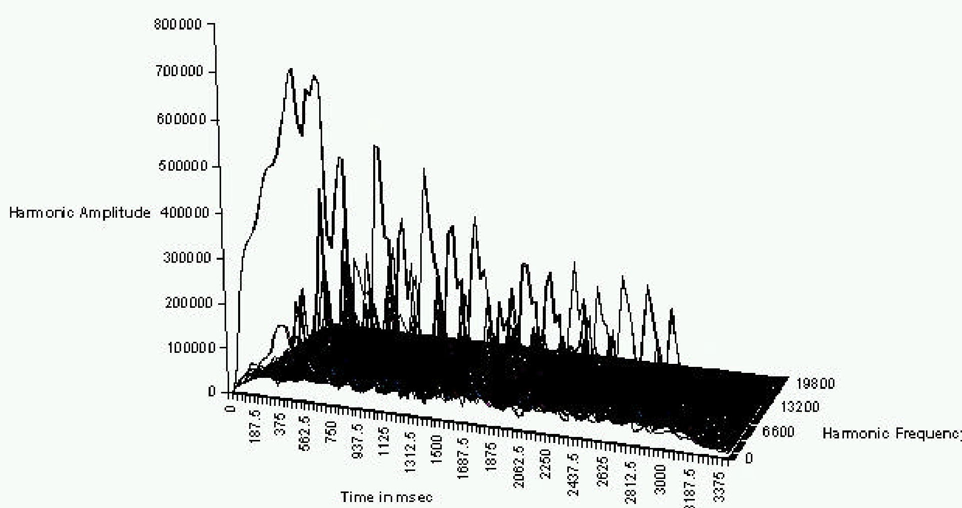
Sample Violin
Harmonic Amplitude
800000
700000
600000
500000
400000
300000
200000
100000
0
Time in msec
0
156.25
312.5
468.75
625
781.25
937.5
1093.75
1250
1406.25
1562.5
1718.75
1875
2031.25
2187.5
2343.75
2500
2656.25
2812.5
2968.75
3125
3281.25
3437.5
Harmonic Frequency
21120
15840
10560
5280
0



AFM Synthesized Violin by GAA



DFM Synthesized Violin by GAA



TREE EVOLUTION ALGORITHM

We have proposed a new algorithm named **Tree Evolution Algorithm (TEA)** based on GAA, which searches the solution space more thoroughly than GAA.

- TEA searches each local minimum separately by splitting the population into several parts.
- Each part forms a new population called a species and evolves in isolation by focusing on the closest local minimum through a GAA-like process independently.
- The parents for each GAA crossover are chosen only from the same group of individual, analogous to nature in that only organisms of the same species can mate.
- The crossover process of GAA is modified in order to restrict the offspring to the same species as their parents.
- The overall best local solution is selected.

VIBRATO IN STRING INSTRUMENTS

Musical instruments may have an oscillation of their amplitude and frequency known as **vibrato** which is part of the performer's technique imposed on the basic tone of the instrument.

We have initiated a study of the phenomenon of vibrato in the violin and other string instruments.

We have obtained the Time-Varying Spectrum (TVS) of the dynamic violin tone by employing an 2048 point FFT to extract the peaks of the harmonics of the dynamic tone.

Because of the vibrato, each peak was broader than for a tone without vibrato. By assuming that the energy under each peak is approximately constant, and that the actual spectrum consists of time-varying delta peaks, we can obtain the equivalent amplitude of each delta peak.

TIME VARYING SPECTRUM (TVS)

The **TVS** of a tone can be split into 2 parts:

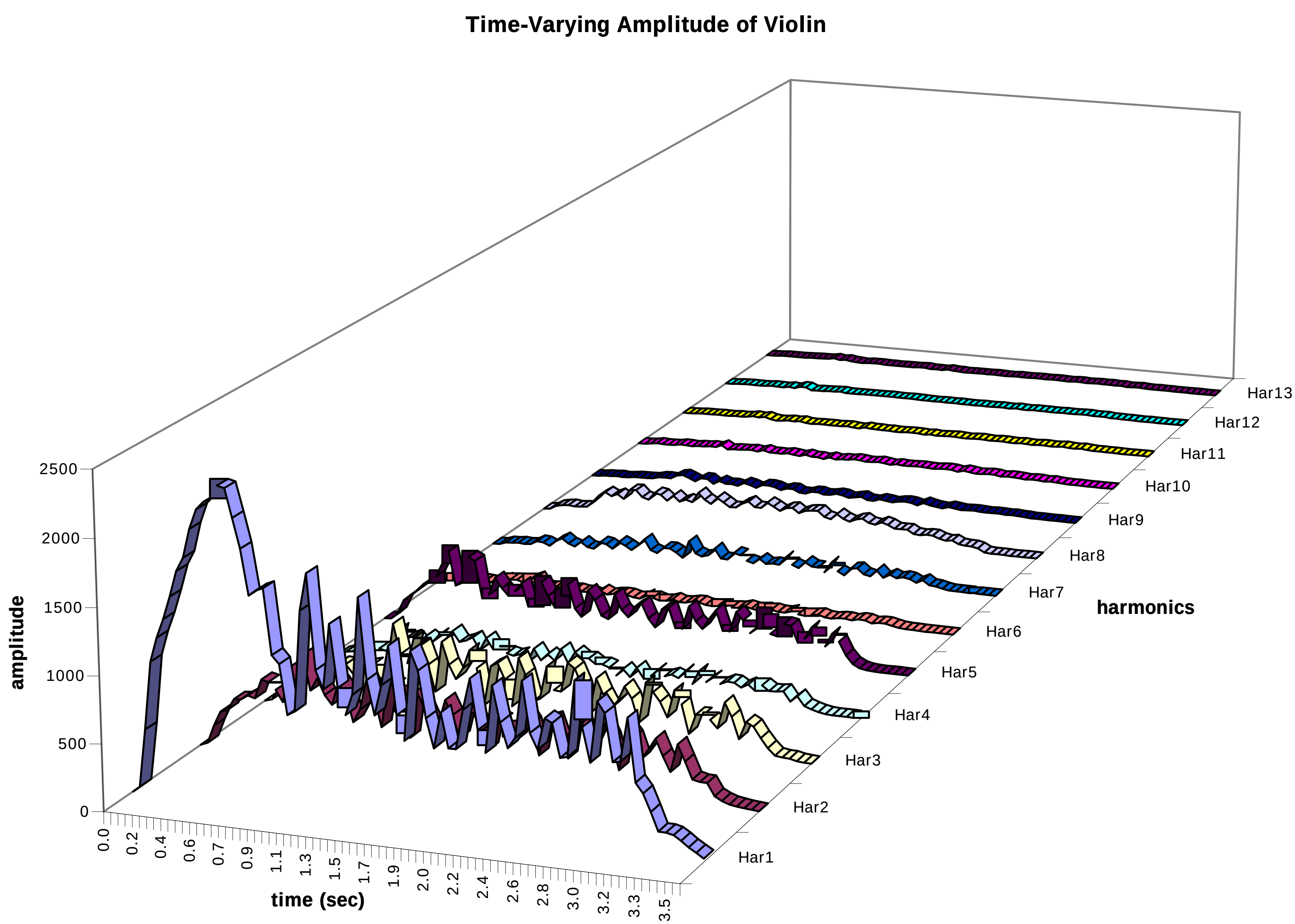
- The **Time Varying Amplitude (TVA)**, the variation of the amplitude.
- The **Time Varying Frequency (TVF)**, the variation of the frequency.

The TVA in turn can be split with a low-pass filter into:

- The **Time Varying Principal Amplitude (TVPA)**, which is the non-vibrato part of the amplitude variation.
- The **Time Varying Amplitude Modulation (TVAM)**, which is due to the vibrato.

Likewise the TVF can be split into:

- The **Time Varying Principal Frequency (TVPF)**.
- The **Time Varying Frequency Modulation (TVFM)**.



Time-Varying Amplitude of Violin
amplitude
2500
2000
1500
1000
500
0
0.0
0.2
0.4
0.6
0.7
0.9
1.1
1.3
1.5
1.7
1.9
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.3
3.5
time (sec)
Har1
Har2
Har3
Har4
Har5
Har6
Har7
Har8
Har9
Har10
Har11
Har12
Har13
harmonics

Fig. 1.1

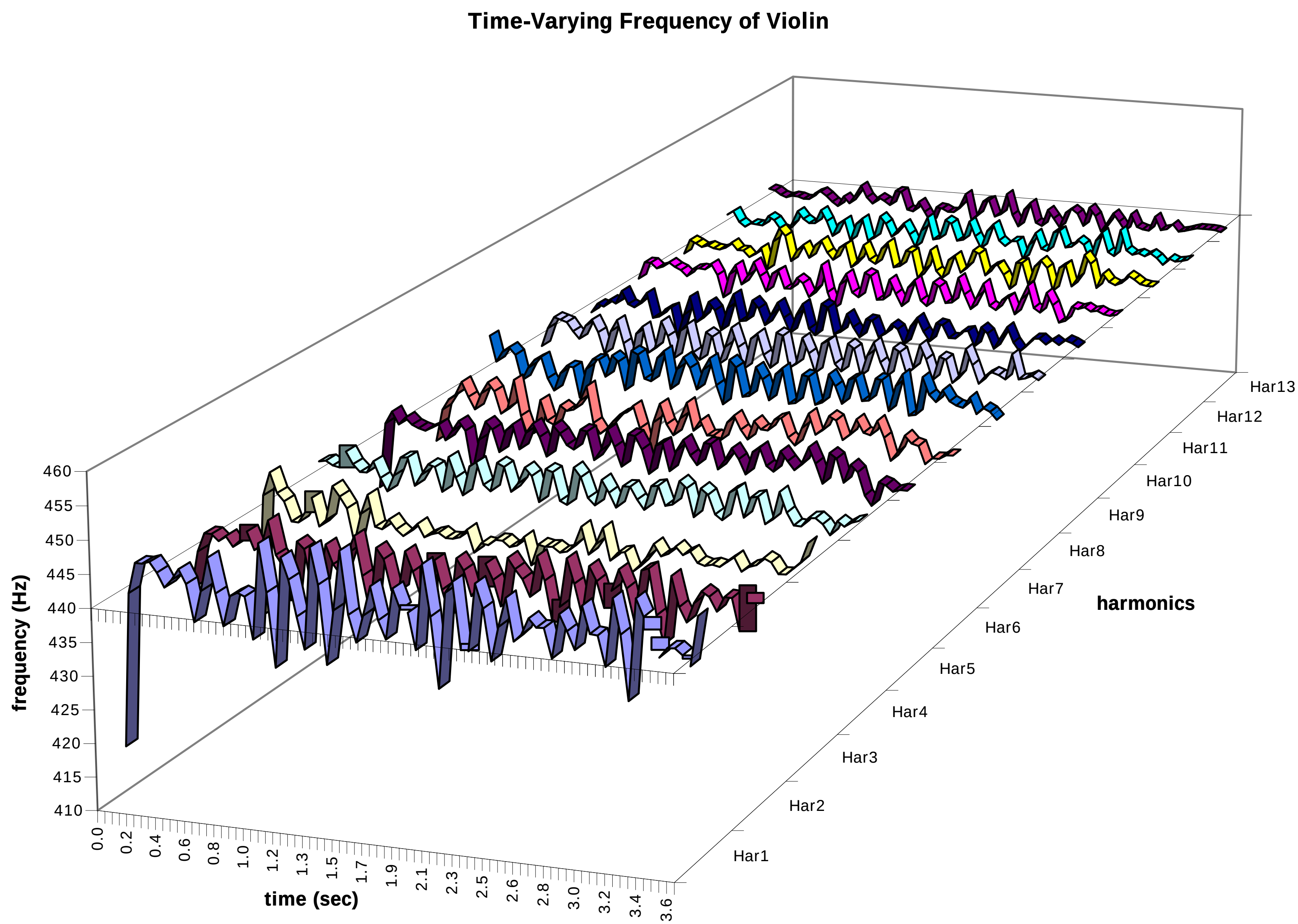
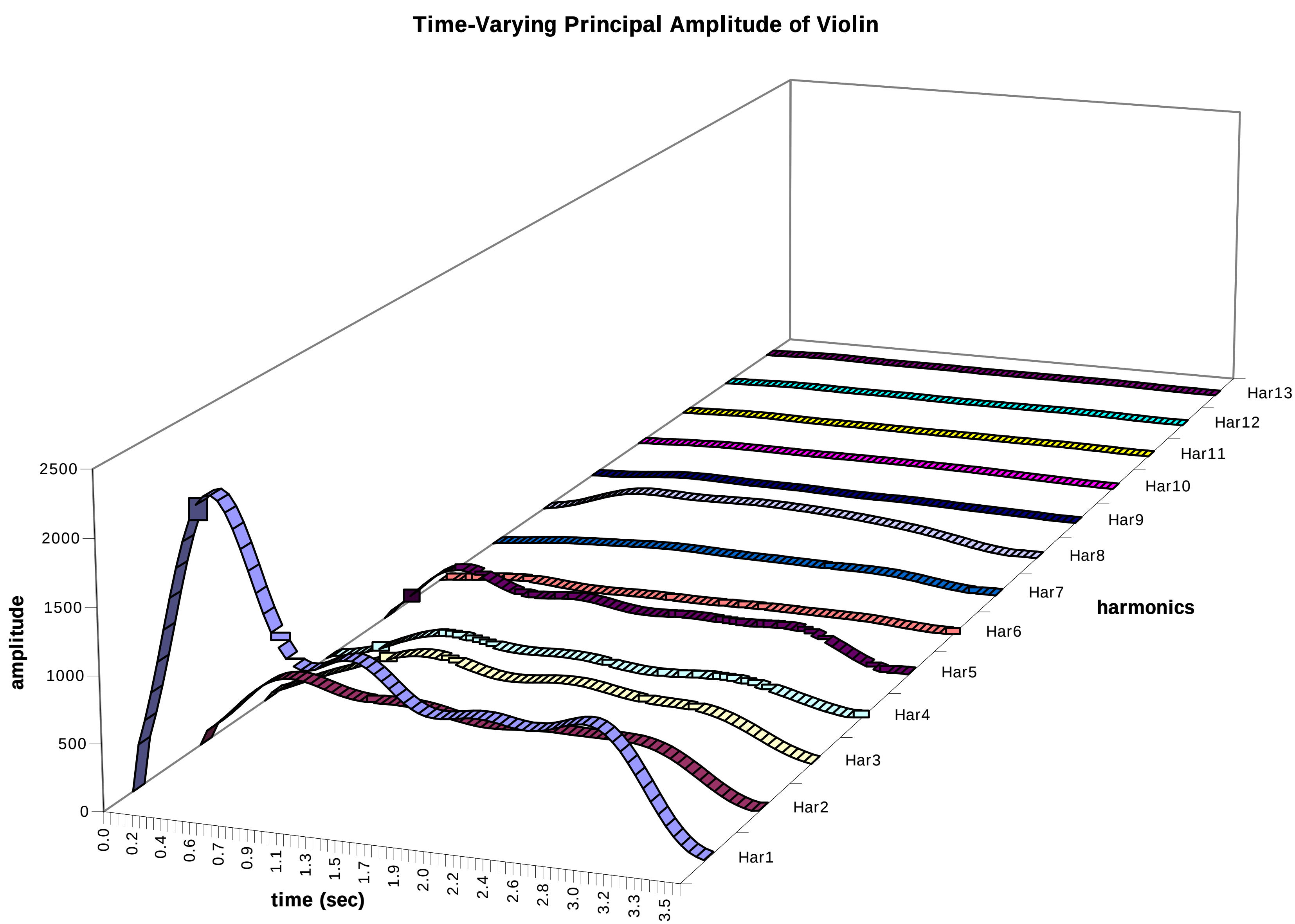

Time-Varying Frequency of Violin
frequency (Hz)
460
455
450
445
440
435
430
425
420
415
410
time (sec)
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.3
1.5
1.7
1.9
2.1
2.3
2.5
2.6
2.8
3.0
3.2
3.4
3.6
harmonics
Har1
Har2
Har3
Har4
Har5
Har6
Har7
Har8
Har9
Har10
Har11
Har12
Har13

Fig. 1.2



Time-Varying Principal Amplitude of Violin
amplitude
2500
2000
1500
1000
500
0
time (sec)
0.0
0.2
0.4
0.6
0.7
0.9
1.1
1.3
1.5
1.7
1.9
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.3
3.5
harmonics
Har1
Har2
Har3
Har4
Har5
Har6
Har7
Har8
Har9
Har10
Har11
Har12
Har13

Fig. 7.1

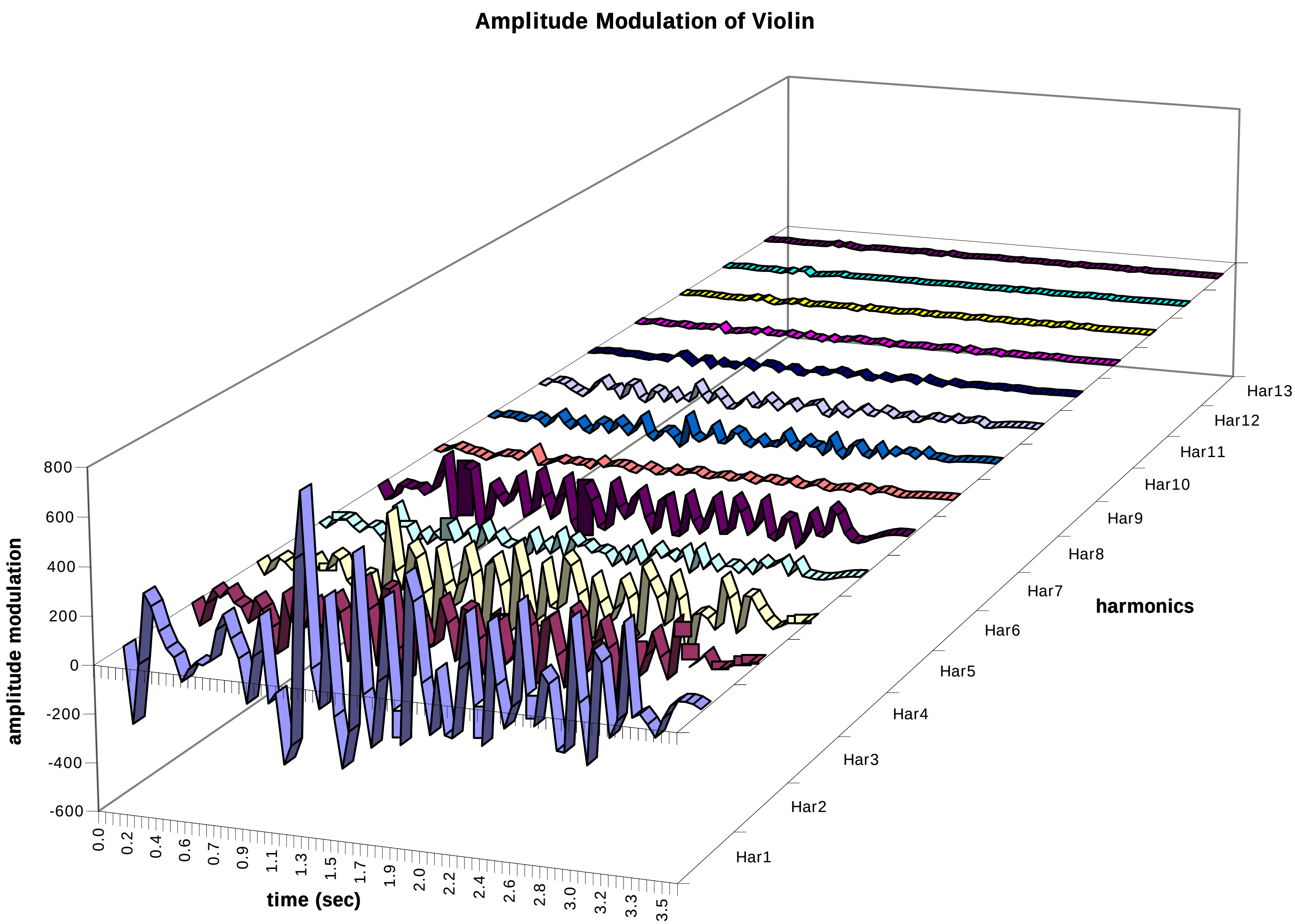
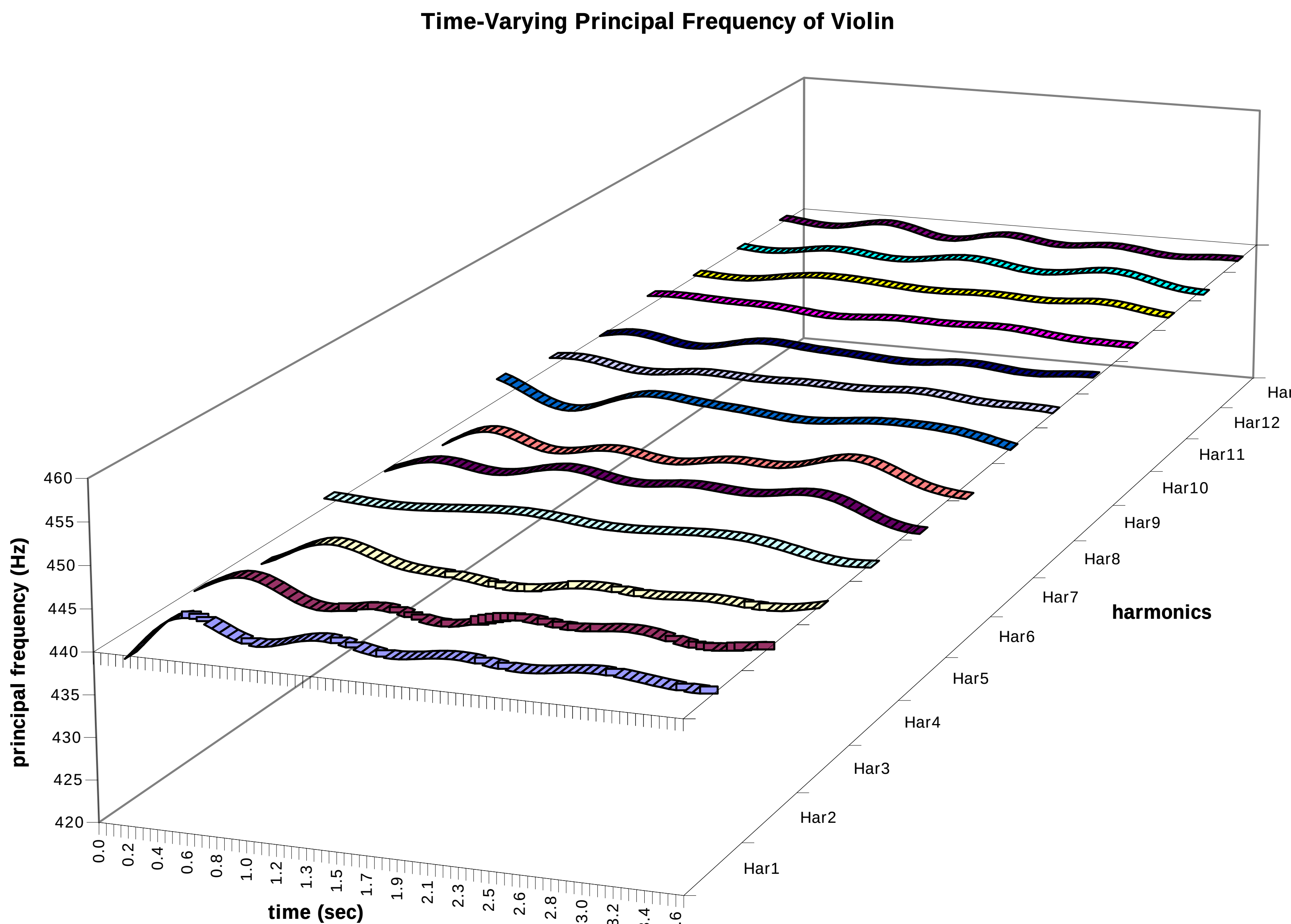
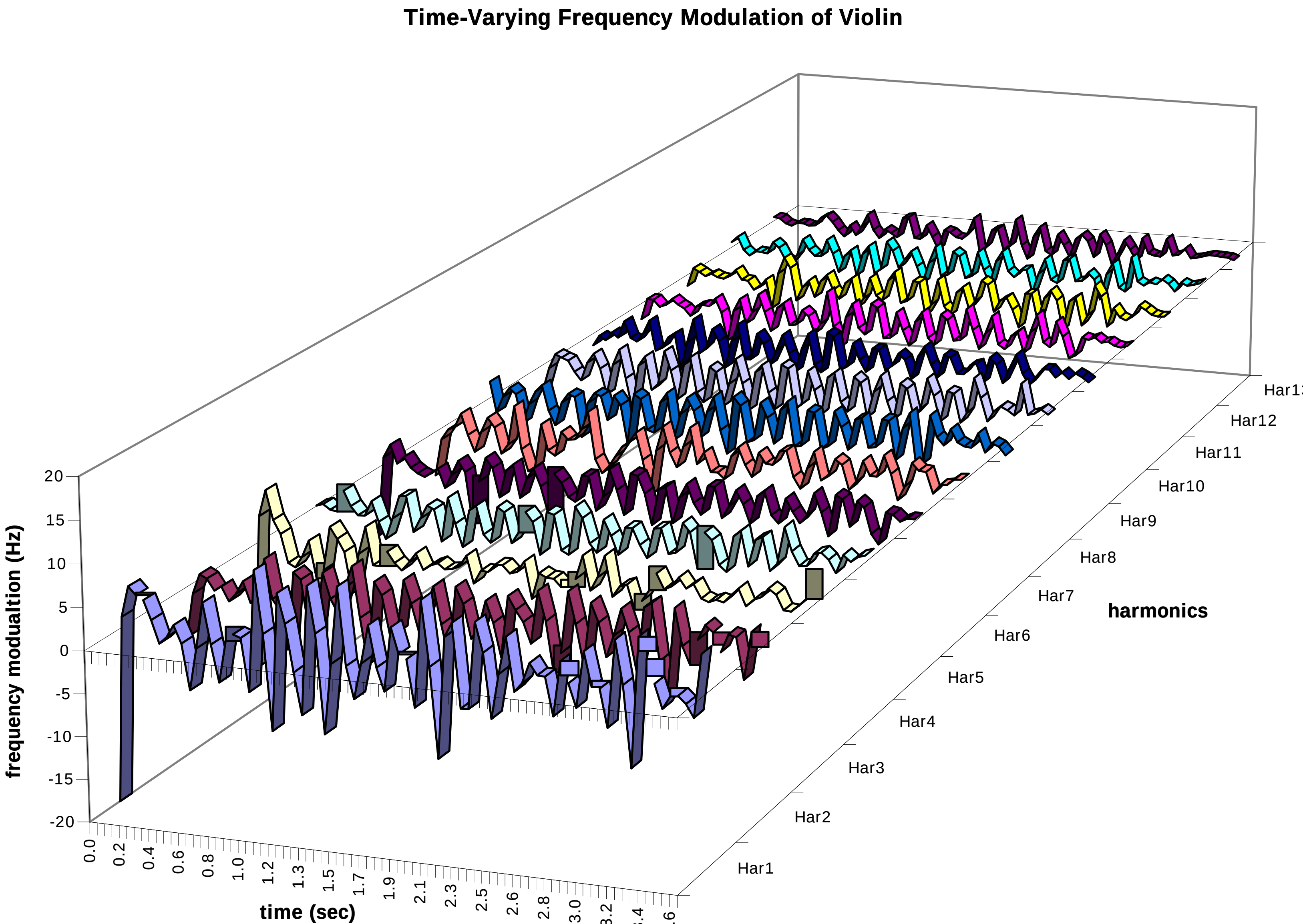

Amplitude Modulation of Violin
amplitude modulation
800
600
400
200
0
-200
-400
-600
0.0
0.2
0.4
0.6
0.7
0.9
1.1
1.3
1.5
1.7
1.9
2.0
2.2
2.4
2.6
2.8
3.0
3.2
3.3
3.5
time (sec)
Har1
Har2
Har3
Har4
Har5
Har6
Har7
Har8
Har9
Har10
Har11
Har12
Har13
harmonics

Fig. 7.2



Time-Varying Principal Frequency of Violin
principal frequency (Hz)
460
455
450
445
440
435
430
425
420
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.3
1.5
1.7
1.9
2.1
2.3
2.5
2.6
2.8
3.0
3.2
3.4
3.6
time (sec)
Har1
Har2
Har3
Har4
Har5
Har6
Har7
Har8
Har9
Har10
Har11
Har12
Har13
harmonics

Fig. 8.1



Time-Varying Frequency Modulation of Violin
frequency modualtion (Hz)
20
15
10
5
0
-5
-10
-15
-20
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.3
1.5
1.7
1.9
2.1
2.3
2.5
2.6
2.8
3.0
3.2
3.4
3.6
time (sec)
Har1
Har2
Har3
Har4
Har5
Har6
Har7
Har8
Har9
Har10
Har11
Har12
Har13
harmonics

Fig. 8.2

ADDITIVE SYNTHESIS OF VIBRATO

We have

TVS = TVA + TVF where
TVA = TVPA + TVAM and
TVF = TVPF + TVFM.

We have chosen to use such an additive model for the vibrato because the information extracted through the filter is additive and such a model provides great flexibility in analysis and simplicity in the real time synthesis of the vibrato.

The maximum average excursion of the for the violin and the other string instruments analyzed is about 3.4Hz to 4.0Hz for A=440 HGz, and the major vibrato rate is about 5.5Hz to 5.9Hz.

We synthesized a vibrato tone by adding the synthesized principal spectrum and the synthesized modulation spectrum. The synthesized vibrato tone was very close to the original real violin tone with vibrato.

BT/26.04.2001