

prem pice

Intermezzo für Akkordeon und Live Electronik

Bruno Friedmann, 2009

Die Fibonacci Reihe entwickelt sich aus dem Vorangegangenen, beginnend bei 0 und 1 - eine andere Progression ist nicht denkbar - denn das Ende der Reihe, von dem man auch beginnen könnte ist nicht bekannt. Nichtsdestotrotz läuft sie immer steiler darauf zu; mit immer größeren Schritten, rasant beschleunigend, zielstrebig, direkt. Sie wird oft als beschreibend für natürliche Wachstumsprozesse, für Dynamik und Strukturen in der Natur gesehen, beschreibend auch für die gefühlte Beschleunigung des Lebens schlechthin.

Überraschenderweise kann man dieser monoton anwachsenden Reihe auch eine Periodizität entnehmen, etwas Wiederkehrendes, Pendelndes, gegensätzlich zur vorwärtstrebenden Dynamik der Fibonacci Reihe selbst. Der Algorithmus $f(n) \text{ modulo } 12$ schafft eine Folge, die sich nach $n = 24, 48, 72, ..$ Elementen der Fibonacci Reihe wiederholt:

0 1 1 2 3 5 8 1 9 10 7 5 0 5 5 10 3 1 4 5 9 2 11 1 (a)

Diese Reihe, als Tonskala - Halbtonabstände relativ zu einem Grundton - begleitet das zeitvariante Vorwärts von *prem pice* und stellt das Wiederkehrende darin dar, auch in sich mit Wiederholungen und Sprüngen versehen, ein Kontrapunkt zum zielstrebigem Vorwärts-aufwärts-immer-schneller. Diese limitierte Sequenz, am Anfang identisch mit der Fibonacci Reihe, laviert dann recht launisch hin und her und endet mit dem größten Rücksprung von 11 nach 1, um wieder von vorne zu beginnen.

Die Dynamik des Fortschritts, die Beschleunigung der Zeit, das unbeirrbar Vorwärtstreiben - es beinhaltet doch zyklische Elemente, Repetitionen, die identisch sind, die sich gleichen; ein Kontrapunkt zu der scheinbar unbeirrbar Beschleunigung ins Unendliche.

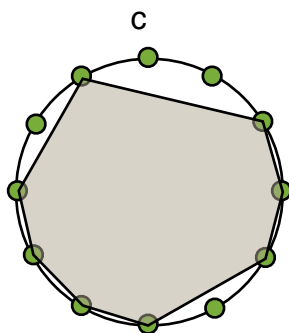


Abb. 1a: Musikal. Skala, auf Grundton f#, resultierend aus den ersten 10 Elementen der Fibonacci Reihe (b)

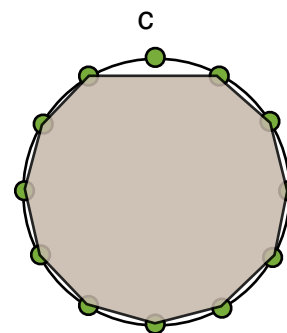


Abb. 1b: Musikal. Skala, auf Grundton f#, resultierend aus der Modulo-Reihe (a); lediglich der Tritonus fehlt

Komposition

prem pice ist ein Stück für Akkorden und Live Electronic. Es besteht aus zwei Stimmen, der Akkordestimme und einer elektronisch erzeugten. Letztere gibt die Töne der aufgenommenen Akkordestimme wieder, jedoch rasch und subtil verändert in der Tonhöhe, gesteuert durch die Lautstärke der ersten Stimme selbst. Die Tonhöhenänderung folgt, in Abhängigkeit von der Lautstärke der Akkordestimme, einer aus der Fibonacci Reihe entwickelten Skala.

Die zu Beginn der Komposition gewählte musikalische Skala resultiert aus der Fibonacci Reihe selbst, bis zum 10. Glied,

0 1 1 2 3 5 8 13 21 34 (b)

der Grundton ist f#, siehe Abb. 1. Das entspricht einer 12-Tonreihe mit Ausnahme von c, c# und f, siehe Abb. 1a.

Ab Takt 18, con brio, ist es die Modulo-Sequenz (a) selbst, mit f# als Basiston, die mit Inversion und Umkehrung als Tonmaterial dient. Das ist die 12-Tonskala mit Ausnahme des Tritonus', Abb. 1b. Die Tonhöhe ist nun nicht proportional zur detektierten Lautstärke, proportional ist der Index von (a).

Live Electronic

Das zu *prem pice* entwickelte Live Electronic Framework basiert auf Max/MSP und der Erweiterung FTM/Gabor (Norbert Schnell, Centre Pompidou, IRCAM). Die Lautstärke der Akkordestimme wird detektiert; damit werden proportional die Elemente der Fibonacci Sequenz (a) angesteuert. Diese geben die Transposition des gerade gespielten Tones an. Wie aus der Sequenz (a) zu sehen ist, ist damit die Tonhöhe keineswegs proportional zur detektierten Lautstärke.

Grundlegend für die Live Electronic ist die oben genannte Modulo-Sequenz (a) der Fibonacci Reihe. Das Element der Sequenz (a) wird durch die Lautheit eines Tones ausgewählt und dient der Transposition des momentan gespielten Tones.

Die verwendeten FTM/Gabor-Patches `gbr.slice~`, `gbr.fft`, `gbr.ifft`, u. a. (im Patcher `analysis_resynthesis_1`), sind eigentlich für eine monophone Anwendung gedacht. Deren polyphone Verwendung erzeugt jedoch interessante Zwischentöne und -klänge, die vom Musiker kontrolliert und gestaltet werden können, Takt 12, Takt 17 und ab Takt 25.

fibonacci 1 (*no of fibo scale = 1*), siehe Abb. 2:

Mit zunehmender Lautstärke wird die Modulo-Sequenz (a) vorwärts verwendet und liefert die Anzahl der Halbtöne, um den gerade gespielte Ton transponiert wird, relativ zu dessen eigener Tonhöhe. Diese Relativität

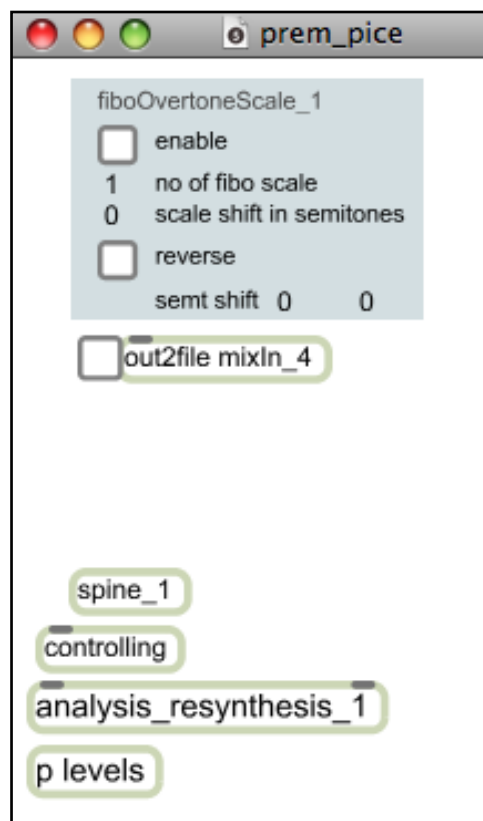


Abb. 2: Main patcher von *prem pice*

kann verändert werden, mit „scale shift in semitones“. Auch kann die Reihe in umgekehrter Folge verwendet werden, „reverse“.

fib0 2 (no of fibo scale = 2), siehe Abb. 2: Mit zunehmender Lautstärke wird eine modifizierte Moduloreihe vorwärts verwendet:

0 1 1 2 3 5 8 13 21 22 19 17 12 29 29 34 27 25 28 41 45 38 47 37 (c)

und liefert die Anzahl der Halbtöne, um den gerade gespielte Ton transponiert wird, relativ zu dessen momentaner Tonhöhe. Diese Relativität kann verändert werden, mit „scale shift in semitones“. Auch kann die Reihe in umgekehrter Folge verwendet werden, „reverse“.

fib0 3 (no of fibo scale = 3), siehe Abb. 2:

Durch Veränderung der Lautstärke wird aus der Moduloreihe zufällig ein Wert ausgewählt und liefert die Anzahl der Halbtöne, um den gerade gespielte Ton transponiert wird, relativ zu dessen Tonhöhe. Die Lage dieses Ton-Raumes kann mit „scale shift“ verändert werden.

Das Zusammenspiel der wichtigsten Max/MSP-Patches ist in Abb. 3 mit Tabelle 1 dargestellt.

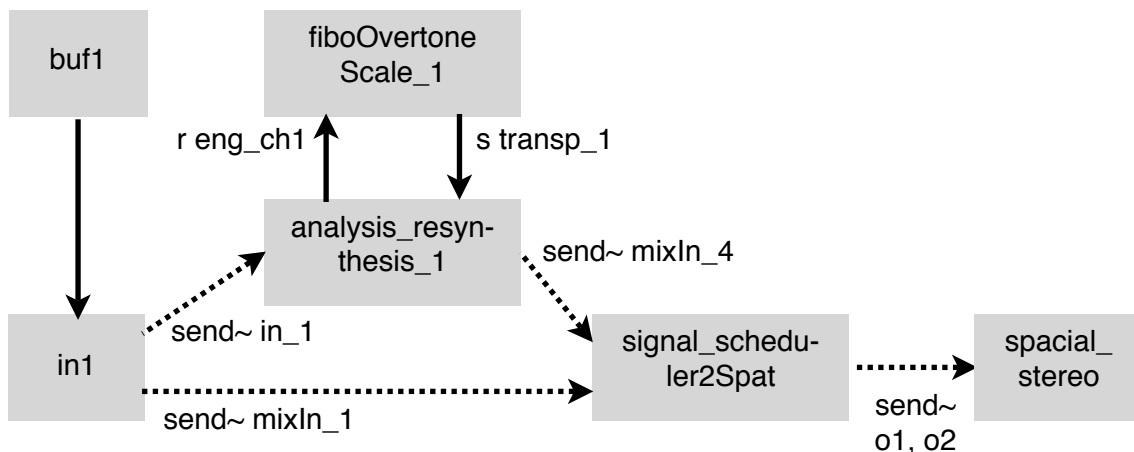


Abb. 3: Übersicht über den Datenfluss zwischen den Haupt-Patches (gestrichelte Linien stellen Audiosignale dar)

Die Notation

Die zweite Stimme, der Live Electronic Teil von *prem pice* wird über die Lautstärkemonulationen der Akkordeonstimme gesteuert. Ein zweites, größeres Notensystem soll diese Dynamik beschreibbar machen: die untere Notenlinie (e) stellt ein Pianissimo dar, die obere (f) ein Fortissimo. Bei *pp* sollte die Lautstärke gerade noch keinen Einfluss auf die Live Electronic haben, das andere Extrem, die maximale wirksame Lautstärke ist bei *ff* erreicht.

fib01 bezeichnet die direkte Anwendung der Sequenz (a). In fib02 wird diese Sequenz auf 4 Oktaven verteilt. fib03 wählt aus (a) zufällig eine Transpositionshöhe aus.

in1	<ul style="list-style-type: none"> • Nimmt Audio auf oder spielt Audio File ab
buf1	<ul style="list-style-type: none"> • definiert und initialisiert buffer Object
analysis_re-synthesis_1	<ul style="list-style-type: none"> • Analysiert das Audiosignal von in1 • gibt dessen Lautstärkewerte (eng_ch1) weiter an fiboOvertoneScale_1 • synthetisiert das Audiosignal und transponiert es um den Wert von transp_1 in cent
fiboOvertoneScale_1	<ul style="list-style-type: none"> • Empfängt von in1 Daten, die die Lautstärke wiedergeben (eng_ch1) • definiert daraus nach der gewählten Skala die Transposition in cent
signal_scheduler2Spat	<ul style="list-style-type: none"> • Audio-Mixer
spacial_stereo	Ermöglicht eine Verräumlichung

Tab. 1: Beschreibung der Max/MSP Haupt-Patches

Die Live Electronic ist wie üblich mit einem controlling versehen, Abb. 4, das durch einfaches Weiterschalten, z. B. mit der Leertaste, eine unkomplizierte Weiterschalten der Effekteinstellungen ermöglicht. Die verschiedenen Zustände sind nummeriert, das ist die erste Ziffer in der LE Notation, Abb. 5, es folgen als Hinweise die Parametereinstellungen der Max-Patches, siehe auch Abb. 2.

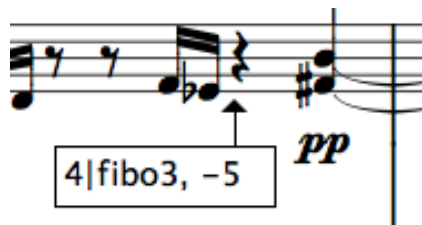


Abb. 5: Live Electronic Notation

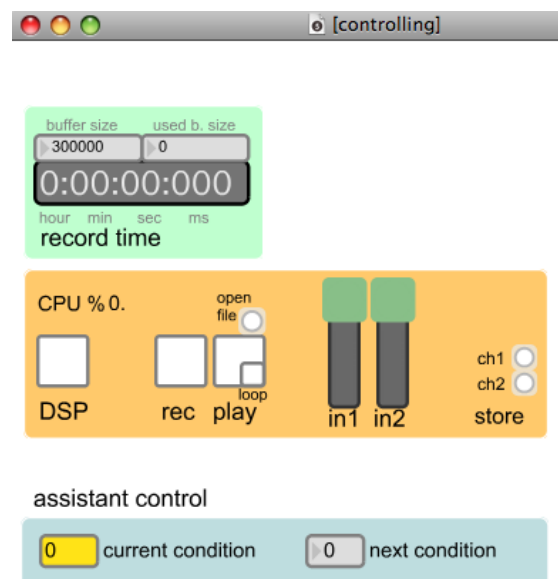


Abb. 4: Live electronic control patcher

prem pice ([mp3](#), [Noten](#))