

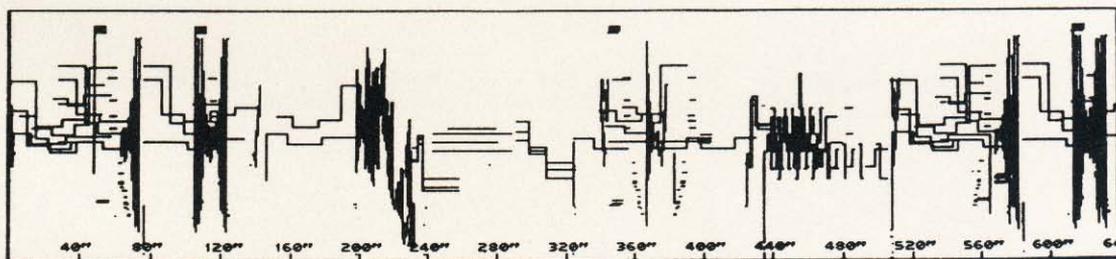
INSTITUT FÜR ELEKTRONISCHE MUSIK

AN DER HOCHSCHULE FÜR MUSIK UND DARSTELLEND KUNST IN GRAZ

KARLHEINZ ESSL

STRUKTURGENERATOREN

ALGORITHMISCHE KOMPOSITION IN ECHTZEIT



SONDERBAND ZUR RINGVORLESUNG
" DIE KLANGWELT AM RAND DER DATENAUTOBAHN "

BEITRÄGE ZUR ELEKTRONISCHEN MUSIK **5**

IMPRESSUM

Herausgeber: Robert Höldrich
Institut für Elektronische Musik (IEM) an
der Hochschule für Musik und
darstellende Kunst in Graz
© 1996

Redaktion: Robert Höldrich, Andreas Weixler

Druck: Druckwerk Graz, Druckerei Khil

Erscheinungsort: Graz, Österreich

Kontaktadresse: Institut für Elektronische Musik (IEM) an
der Hochschule für Musik und
darstellende Kunst in Graz

Jakoministr. 3-5
A - 8010 Graz, Österreich

Tel.: ++43/ +316/ 389 - 7010 (Sekretariat)
Fax: ++43/ +316/ 389 - 7008

Titelblatt: Graphik aus der Komposition "Natté" von
Helmut Dencker mit freundlicher
Genehmigung des Komponisten.

Bisher erschienen folgende "Beiträge zur elektronischen Musik":

BEM 1	HARALD FRIPERTINGER	ENUMERATION IN MUSICAL THEORY	1992
BEM 2	GREGOR WIDHOLM	HOLOGRAPHIE, CAD UND MODALANALYSE IM DIENSTE DER MUSIK	1993
BEM 3	HELWIG BRUNNER	DER NACHTIGALLENGESANG IN DER EUROPÄISCHEN KUNSTMUSIK	1994
BEM 4 1995	NORBERT SCHNELL	GRAINY - GRANULARSYNTHESE IN ECHTZEIT	

Die Reihe "Beiträge zur Elektronischen Musik" stellt Arbeiten des Instituts für Elektronische Musik Graz zu den Themenbereichen Akustik, Computermusik, Musikelektronik und Medienphilosophie vor. Dabei handelt es sich meist um Ergebnisse von Forschungsarbeiten am Institut oder um überarbeitete Vorträge von Institutsmitarbeitern. Darüber hinaus soll hier eine Diskussionsplattform zu den genannten Themen entstehen. Beiträge können auch eine Beschreibung von Projekten und Ideen sein, die sich noch in Entwicklung befinden und noch nicht fertiggestellt sind.

Beiträge bzw. Anfragen über bisher erschienene Artikel richten Sie bitte an umseitige Kontaktadresse.

Wir hoffen, daß die Schriftreihe "Beiträge zur Elektronischen Musik" eine Anregung für Ihre wissenschaftliche und künstlerische Arbeit bietet.

Robert Höldrich & Andreas Weixler
(Redaktion)

The series "Beiträge zur Elektronischen Musik" (contributions to electronic music) presents papers by the Institute of Electronic Music Graz on various topics such as acoustics, computer music, music electronics and media philosophy. The contributions present either results of research work performed at the institute or edited lectures held by members of the institute.

Furthermore, the series shall establish a discussion forum for the above mentioned fields. Articles will be written either in English or German. The contributions can also deal with the description of projects and ideas that are still in preparation and not yet completed.

Please send submissions and inquiries concerning already published articles to the address mentioned on the previous page.

We hope that the series "Beiträge zur Elektronischen Musik" will provide thought provoking ideas for your scientific and artistic work.

Robert Höldrich & Andreas Weixler
(the editors)

Vorwort des Herausgebers

Karlheinz Essl's Arbeit über musikalische Strukturgeneratoren ist die erste einer Serie von Sonderausgaben der "Beiträge zur Elektronischen Musik". Diese wird die Ergebnisse einer Veranstaltungsreihe zusammenfassen, die im Sommersemester 1995 unter dem Titel "Die Klangwelt am Rand der Datenautobahn - Ringvorlesung zu Gegenwart und Zukunft der Computermusik" am Institut für Elektronische Musik Graz stattgefunden hat.

Die Vortragenden dieser Ringvorlesung waren Miller Puckette, Mesias Maiguashca, Robin Minard, Karlheinz Essl, Klaus Hollinetz, Norbert Schnell, Günther Rabl, Winfried Ritsch und Robert Höldrich. Im Rahmen von mehrtägigen Seminaren wurden verschiedene künstlerische und wissenschaftliche Aspekte der Computermusik behandelt. Die Vorträge wurden von mehreren Konzerten begleitet, in denen Werke von Maiguashca, Minard, Essl und Hollinetz präsentiert wurden.

Die Herstellung dieser Publikation wurde von den Musikkuratoren des Bundesministers für Wissenschaft und Kunst unterstützt.

Robert Höldrich

KARLHEINZ ESSL

STRUKTURGENERATOREN

Algorithmische Komposition in Echtzeit

Abstract

Mit der Entwicklung von interaktiven Programmierumgebungen wie MAX ist es möglich geworden, Kompositionsalgorithmen als Generatoren zu implementieren, die musikalische Strukturen in Echtzeit erzeugen können.

Grundlegend dafür ist das Konzept des Strukturgenerators, daß sich theoretisch sowohl auf serielle Denkweisen als auch auf die bahnbrechenden Pionierarbeiten von Gottfried Michael Koenig bezieht: eine musikalische Struktur wird demnach als mögliche Variante einer allgemeineren Meta-Struktur ("Modell") verstanden, deren Eigenschaften durch Veränderung der Modellparameter gesteuert werden können.

1992 hat der Autor begonnen, für MAX eine umfassende Bibliothek von Softwaremodulen zu erstellen, mit deren Hilfe solche Strukturgeneratoren programmiert werden können. Diese „Real Time Composition Library“ umfaßt verschiedene Kategorien von Objekten: Zufallsgeneratoren, Selektionsmechanismen, Harmonie- und Rhythmusgeneratoren sowie umfangreiche Listen- und Toolbox-Funktionen.

Mit Verwendung dieser Library (die mittlerweile in der Version 2.2 vorliegt) wurde die interaktive Echtzeitkomposition "Lexikon-Sonate" (1992 ff.) für computergesteuertes Klavier erstellt, die in dem vorliegenden Text analysiert wird.

Thanks to the development of interactive programming environments such as MAX, compositional algorithms can now be implemented as generators which are capable of creating musical structures in real time.

This approach is based on the concept of structure generators which are drawn from serial philosophy of the 1950's as well as Gottfried Michael Koenig's primary research on composition theory. Here, musical structure is viewed as one possible variant drawn from a more general meta-model, whose properties can be controlled by changing the system's parameters.

In 1992, the author started developing a library of software modules written in MAX for the easy implementation of structure generators. This "Real Time Composition Library" is composed of different classes of objects:

random generators, selection mechanisms, harmony and rhythm generators, list operations and toolbox functions. "Lexikon-Sonate" (1992 ff.), an interactive real time composition for computer-controlled piano, was created using this library. This work will be analysed in the following article.

Inhaltsverzeichnis

Vorwort 7

1. Serielle Theorie	9
1.1. Serialismus als Methode kompositorischen Denkens	9
1.2. Tabula rasa & Musica pura	10
1.3. Bezugspunkte	11
1.4. Integrale Reihentechnik	11
1.5. Parameter	13
1.6. Reihenmanipulationen	14
1.7. Zufall	17
1.8. ...wie die Zeit vergeht...	19
2. Zur Theorie der Strukturgeneratoren	23
2.1. Struktur	23
2.2. Gestalt	24
2.3. Modellbildung – <i>Computer Aided Composition</i>	25
2.4. "Instrument"	25
2.5. Strukturgeneratoren	26
3. "Real Time Composition Library"	29
3.1. Komponieren mit Computerunterstützung	29
3.2. MAX	29
3.3. RTC-lib	30
3.3.1. Zufallsoperationen	35
3.3.2. Rhythmus	39
3.3.3. Harmonik	43
3.4. Konstruktion von Strukturgeneratoren	47
4. Lexikon-Sonate	49
4.1. Ausgangspunkte	49
4.2. Module	51
4.3. Benutzeroberfläche	53
4.4. Aktions- und Interaktionsmöglichkeiten	55
4.5. Rezeption	56

Literatur 58

Anhang 60

Vorwort

Dieser Aufsatz widmet sich der Frage, wie mit Hilfe des Computers komponiert werden kann, was die theoretischen Voraussetzungen dafür sind und wie sich dies in der Praxis gestaltet. Zentraler Punkt meiner Überlegungen ist das Konzept des Strukturgenerators, ein als Computerprogramm implementiertes kompositorisches Modell. Als wichtiger Anreger fungiert für mich Gottfried Michael Koenigs *Projekt 1*¹ (1964–66), das er an der Universität Bonn entwickelt hatte und die Idee des Strukturgenerators in bündiger und gut dokumentierter Form an die Öffentlichkeit brachte. Koenigs Computerprogramm war ursprünglich dazu konzipiert, eine Reihe von Strukturvarianten aus einem einzigen kompositorischen Modell abzuleiten, um damit eine ganze Werkreihe zu komponieren². Das ursprünglich in FORTRAN II geschriebene Programm selbst ist fixiert und geschlossen; der Benutzer kann zwar über Menüs die Eingabedaten manipulieren, den Verlauf der Berechnung aber darüber hinaus nicht beeinflussen.

Der Nachfolger – *Projekt 2*³ – erlaubt es dem Benutzer nun, weitgehend Einfluß auf den Berechnungsvorgang selbst zu nehmen. Durch Ausfüllen eines 60teiligen Fragebogens kann er nicht nur die Parameterdaten eingeben, sondern auch bestimmten Verarbeitungsalgorithmen auswählen. Trotzdem erweist sich auch dieses Programm als hermetisch geschlossen: die Interaktion mit dem Computer bleibt auf die Beantwortung der 60 Fragen beschränkt.

Koenigs grundlegende und unendlich wertvolle Ansätze wurden von mir nun erweitert: statt eines geschlossenen Programmes habe ich nun ein offenes und erweiterbares, modular aufgebautes "Environment"⁴ geschaffen, mit dessen Hilfe verschiedenste Strukturgeneratoren konstruiert werden können. Diese generieren nun ihre Resultate in Echtzeit ("Realtime Composition"), was einen interaktiven Live-Einsatz ermöglicht. Anders als bei Koenig handelt es sich dabei nicht um ein geschlossenes Programm, sondern um eine Bibliothek von Softwaremodulen, mit denen Strukturgeneratoren programmiert werden können.

¹Gottfried Michael Koenig, *Projekt 1 " – Modell und Wirklichkeit* (1979); in: ders., *Ästhetische Praxis. Texte zur Musik*, hrsg. von Stefan Fricke und Wolf Frobenius, Bd. 3 (Saarbrücken 1993), S. 223 – 230.

²Aus dieser Werkreihe sind zwei Ensemblestück – *Projekt 1 - Version 1* (1965/66) und *Projekt 1 - Version 3* (1967) bei Edition Peters (London) erschienen. Letzteres wurde 1969 vom Ensemble "die reihe" unter der Leitung von Kurt Schwertsik in Wien uraufgeführt.

³Gottfried Michael Koenig, *Project 2 / 82 - a program for musical composition. Manual* (Instituut voor Sonologie, Utrecht 1983) - Siehe auch: ders., *Über meine Projekte 1 und 2*; in: *Texte*, a.a.O., Bd. 3, S. 350–357.

⁴*Real Time Composition Library "for MAX* (1992 ff.): s. Kapitel 3.3 der vorliegenden Arbeit.

Bevor ich diese Software-Bibliothek vorstelle, möchte ich zunächst die theoretischen Grundlagen diskutieren, die sich – historisch – auf serielle Denkweisen beziehen, die in der von mir formulierten Theorie der “Realtime Composition” weitergedacht wurden.

Graz, im Herbst 1995 Karlheinz Essl

1. Serielle Theorie

Mein Konzept der Strukturgeneratoren basiert wesentlich auf Erkenntnissen der seriellen Kompositionstheorie, die in Bezug auf das Medium Computer weiterentwickelt wurden. Ich möchte in diesem Kapitel die verschiedenen Aspekte des seriellen Denkens darstellen, die als Ausgangspunkt für die Idee der "Realtime Composition" dienen.

1.1. Serialismus als Methode kompositorischen Denkens

Die grundlegende Absicht der seriellen Musik liegt in der Vermittlung zwischen dem extrem Kleinen und dem extrem Großen – den akustischen Eigenschaften des Klanges ("Parameter" genannt) und der Gesamtform der Komposition. Dabei stellt sich das Serielle weniger als Technik denn als Methode musikalischen Denkens dar, die in der uralten Idee von der Versöhnung der Gegensätze wurzelt. Diese geht auf Pythagoras zurück und fand über Giordano Bruno und Spinoza Eingang in die Monadenlehre Wilhelm Leibniz'. Die Welt wird nicht mehr dualistisch vorgestellt und die Form nicht mehr aus apriorischen Gegensätzen entwickelt, sondern als Einheit aufgefaßt, die sowohl rational als auch mystisch erfahrbar ist. Gegensätze werden nicht mehr als widerstreitende Prinzipien, sondern verschiedenartige Ausprägungen desselben übergeordneten Aspekts begriffen. Über Zwischenstufen vermittelt bilden sie ein **Kontinuum**, das sich als "digitale" Skala von Werten darstellt. Dadurch lassen sich Aspekte des Tonsatzes, die traditionell unmittelbar nichts miteinander zu tun haben – etwa Rhythmus und Klangfarbe – als Ausprägung ein und derselben Sache begreifen. Herkömmlicherweise wurden diese beiden "Parameter" auf verschiedenen Ebenen des Kompositionsprozesses organisiert, wobei das Rhythmische eindeutig über dem Klanglichen dominierte. Motivik als eine Funktion des Rhythmus trieb das musikalische Geschehen primär fort, während die Klangfarbe eingesetzt wurde, diese Vorgänge plastisch zu verdeutlichen. Im seriellen Denken lassen sich diese beiden Parameter als verschiedenartige Ausprägung zeitlicher Prozesse auffassen. Experimente im elektronischen Studio hatten gezeigt, daß rhythmische Pulsationen in Klang umschlagen können, wenn man deren Abfolge beschleunigt (was sich mit einem

Impulsgenerator leicht bewerkstelligen läßt)⁵.

Ein einfaches Experiment möge dies verdeutlichen: Man lasse eine Ping-Pong-Ball auf eine harte Oberfläche fallen. Der Rhythmus des Aufschlagens wird kontinuierlich schneller, bis er zuletzt in einen singenden Klang umschlägt. Das Umkippen des Rhythmischen in eine neue Zustandsqualität (die des Klanges) passiert in dem Moment, wo die Geschwindigkeit der Prallimpulse eine Frequenz von ca. 20 Hz (d.h. die untere Hörschwelle) übersteigen.

Diese im elektronischen Studio gewonnene Erkenntnis zeitigte wiederum Auswirkungen auf die Instrumentalmusik: die *Klangflächenkompositionen* György Ligetis und Friedrich Cerhas sind direkte Umsetzungen hiervon.

1.2. Tabula rasa & Musica pura

⁵Gottfried Michael Koenig, *Ligeti und die elektronische Musik*; in: Ligeti. Personalstil - Avantgardismus - Popularität (= S. Wertungsforschung, Bd. 19, hrsg. von Otto Kolleritsch, Wien 1987), S. 13.

Die Entwicklung des Serialismus ist eng mit der Nachkriegsgeschichte verknüpft. Sein Hauptbestreben lag in der Schaffung einer neuen musikalischen Syntax, die sich nicht auf die – verdächtig gewordenen – überlieferten Formen, Sujets und Techniken stützen konnte. Eine Weiterführung der Tendenzen der Romantik – die schließlich zum Faschismus geführt hatten – schien unmöglich, ebensowenig eine Wiederaufnahme des Hindemith'schen Hantierens im Werkstoff der Töne. Der Zustand der "tabula rasa", wie er sich den Komponisten um 1950 darstellte, bot die Chance für einen Neuanfang: *"Die Städte sind radiert",* frohlockte Stockhausen, *"man kann von Grund auf neu anfangen ohne Rücksicht auf Ruinen und »geschmacklose« Überreste!"*⁶

Zunächst wurde alles traditionell Verbürgte über Bord geworfen. In der geschichtlichen Situation des Wiederaufbaus begann man, das zertrümmerte musikalische Material vom Nullpunkt aus neu zu organisieren. Dabei wurde grundlegende Pionierarbeit geleistet, von der wir heute noch zehren. Durch methodische Strenge und eiserne Selbstkontrolle versuchte man sich gegen den Rückfall in die obsolet gewordenen traditionellen Sprachmuster abzusichern. Galt es doch, Grundsätzliches innerhalb des Tonmaterials und in bezug auf die Wahrnehmung zu erforschen, *"um in absehbarer Zeit das Handwerk zu beherrschen und die neue musikalische Sprache so zu sprechen, wie einem der Schnabel gewachsen ist."*⁷

Jeder Aspekt der Komposition sollte der kompositorischen Verfügungsgewalt unterstellt werden. Diese durchdringt die einst von einem "common sense" getragenen traditionellen Sprachmittel und gipfelt zuletzt in der Komposition des Klanges selbst. Doch bis dahin war es noch ein weiter Weg. In ihrer Bemühung nach Reinheit entstanden Werke wie Boulez' Extremstück *Structure 1a* (1952) für 2 Klaviere (das sich programmatisch auf das Gemälde "Am Rande des Fruchtlandes" von Paul Klee beruft) und Karel Goeyvaerts elektronisches Frühwerk *Nummer 5 met zuivere tonen* [mit reinen Tönen] (1953).

1.3. Bezugspunkte

⁶Karlheinz Stockhausen, *Zur Situation des Metiers (Klangkomposition)*, 1953; in: ders., *Texte zur Musik*, hrsg. von Dieter Schnebel (Köln 1963 ff.), Bd. I, S.48.

⁷ebda., S. 48.

Als Vaterfigur für die junge Komponistengeneration diente **Anton Webern**, in dessen Werk man bereits Ansätze des neuen musikalischen Denkens zu erkennen glaubte: Weberns (mißverständene) Sensibilität für den Einzelton wurden als Vorbote der "Punktuellen Musik" gesehen, seine Strukturproportionen wiederum als Vorstufe der elektronischen Musik⁸.

Daneben bezog man sich auch auf **Claude Debussy**, der in seinem Orchesterwerk *Jeux* (1911) von einem neuen Formkonzept ausgegangen war: dem irrationalen organisch-pflanzlichen Wuchern im Gegensatz zur funktionalen Formvorstellung der Wiener Klassik. Webern und Debussy stellen gleichsam die Extrempole einer Skala zwischen PUNKT (= punktuelle Musik) und FLÄCHE (= Feldkomposition) dar⁹. Im Kontinuum ihres Überganges entfaltet sich nun das ganze Formspektrum der seriellen Musik: vom Pointillismus der *Kontrapunkte* (1952/53) Stockhausens bis hin zu Cerhas Klangflächenkomposition *Fasce* (1959 ff.). Auch wenn es den Eindruck erwecken möchte, als wäre der Serialismus eine unmittelbare Weiterentwicklungen der Ansätze von Webern und Debussy, so zeigte es sich, daß diese Legitimation erst im Nachhinein geschah, als sich die neuen Methoden schon längst etabliert hatten. Webern und Debussy dienten in erster Linie dazu, das Neue zu verdeutlichen, worauf Stockhausen ausdrücklich hingewiesen hatte: *"In Weberns Werk entdeckte ich dann später – findend, was ich suchte –, daß vieles bei ihm schon vorbereitet war"*¹⁰.

1.4. Integrale Reihentechnik

⁸ders., *Weberns Konzert für 9 Instrumente op. 24. Analyse des ersten Satzes* (1953); in: Texte I, S. 24 f.

⁹ders., *Von Webern zu Debussy. Bemerkungen zur statistischen Form* (1954); in: Texte I, S. 75 f.

¹⁰ders., *Gruppenkomposition: Klavierstück I (Anleitung zum Hören)*, 1955; in: Texte I, S. 63 f.

Der Serialismus ist eine Methode, unterschiedliche Kräfte auszubalancieren, wobei Extreme durch Zwischenschritte vermittelt werden. Um die Einheit der verschiedenen musikalischen Bestimmungsgrößen ohne Zerstörung ihrer individuellen Elemente zu bewahren, werden diese demselben Organisationsprinzip unterworfen. Dabei handelt es sich um eine verbindliche Folge von Zahlenproportionen. Eine solche Abstraktion erwies sich als unumgänglich, zugleich aber als eminent fruchtbar: dadurch wurde es erst möglich, ursprünglich voneinander getrennte musikalische Aspekte miteinander in Beziehung zu setzen. Dies führte schließlich zum seriellen Formkonzept, in dem *„Großform und alle Detailformen eines Werkes aus einer einzigen Proportionsreihe abgeleitet werden.“*¹¹

Die in Weberns Spätwerk beobachtete Tendenz, die Reihentechnik auch auf andere Parameter des Tonsatzes auszudehnen, führte in Konsequenz schließlich zur sog. „Integralen Reihentechnik“.

Damit ändert sich auch die Funktion der Reihe. Bei Schönberg verdankte sie sich noch ganz dem Einfall und diente als thematische Keimzelle, aus der sich alle Gestalten der Komposition in ihrer horizontalen und vertikalen Dimension ableiten lassen.

Anders aber in der Seriellen Musik:

- Eine Reihe ist eine **spezifischen Anordnung** von gleichwertigen Elementen, die jeweils nur einmal vorkommen dürfen (somit eine Menge darstellen) und Proportionen zueinander bilden. Da aufgrund des seriellen Wiederholungsverbotens keines der Elemente bevorzugt auftritt, stellt sie eine nichthierarchische Organisationsform dar, deren dezentrale Struktur innere Mobilität garantiert.
- Die Reihe birgt eine potentielle **Möglichkeitsstruktur**, die im Verlaufe einer Komposition entfaltet wird. In diesem Sinne ist die Reihe mit einem Samenkorn vergleichbar, in dem bereits das ganze Potential einer Pflanze „in nuce“ enthalten ist.
- Damit wird die Reihe zu einem **Steuermechanismus** – zu einem Programm: in einem streng determinierten hierarchischen Prozeß steuern übergeordnete Reihen untergeordnete Funktionsabläufe (*top-down-approach*).
- Da die Reihe nun auf verschiedenen hierarchischen Ebenen wirkt, ist sie zugleich **Material und Gesetz**. Damit ist die Konsistenz der Methode garantiert: das Serielle zeigt sich in

¹¹ders., *Erfindungen und Entdeckungen. Ein Beitrag zur Form-Genese* (1961); in: Texte I, S. 228.

seinem Unifikationsbestreben ganz als geistiges Kind der Moderne.

1.5. Parameter

Die Reihe als Steuermechanismus beeinflusst nun das Verhalten der sog. "Parameter". Dieser Terminus bezeichnet im seriellen Sprachgebrauch zunächst die akustischen Toneigenschaften Höhe, Dauer, Lautstärke und Klangfarbe. Mit der zunehmend rationalen Durchdringung aller kompositorischen Teilaspekte wurde dieser Begriff auch auf übergeordnete Struktureigenschaften wie Dichtegrad, Gruppencharakteristik, Harmonik, Tonhöhenambitus etc. erweitert. Was als Parameter zu gelten hat, muß nun von Komposition zu Komposition jeweils aufs Neue definiert werden.

In der im Anschluß an die serielle Poetik entstandene Computermusik bedeutet der Ausdruck "Parameter" nunmehr eine **Variable** im formalisierbaren Kompositionsprozeß, die die Veränderungen der musikalischen Struktureigenschaften kontrolliert. In diesem Sinne verwenden wir ihn auch bei der Diskussion der Strukturgeneratoren.

Das Denken in Parametern läßt sich als Weiterführung polyphoner Techniken – des Kanons zumal – deuten. Erinnert sei auch an die unabhängige Organisation von Tonhöhen und Dauern – als "color" bzw. "talea" – in der isorhythmischen Motette des 14. und 15. Jahrhunderts. Faßt man Polyphonie als das geordnete gleichzeitige Ablaufen selbständiger Stimmen auf, so treten im Seriellen die ineinandergreifenden Parameterschichten an Stelle der kontrapunktischen Stimmzüge. Diese Parameterschichten, aufeinander projiziert, schießen zur Struktur zusammen.

Hatte man in den Anfängen der seriellen Musik noch geglaubt, die einzelnen Parameter nach dem gleichen Reihenschema organisieren zu müssen, so erkannten die Komponisten sehr bald, daß diese entsprechend ihrer eigenen Voraussetzungen zu differenzieren seien: als Ausdruck des angestrebten materialgerechten Denkens. Experimente aus der Frühzeit des Serialismus – wie Pierre Boulez' schon erwähnte *Structure 1a* (1951)¹² – führten zu einer kritischen Durchleuchtung der kompositorischen Theorie: Das ursprünglich als Unifikationstheorie konzipierte Serielle verwandelte sich schließlich durch die Tendenzen des Materials.

1.6. Reihenmanipulationen

Wir wollen uns im Folgenden einigen klassischen Verfahren der Reihenmanipulation widmen.

- Das serielle Wiederholungsverbot führt nun dazu, daß die Ursprungsreihe nicht unverändert beibehalten wird, sondern ständig variiert ("permutiert") wird. Die **Permutations**regel selbst kann wiederum aus der Grundreihe abgeleitet werden. Auf diese Weise wird ein komplexes hierarchisches – und selbstreferentielles – System errichtet, dessen funktionelle Aspekte jeweils von Varianten der Ausgangsreihe gesteuert werden; als Resultat erscheint ein Strom parallel ablaufender Parameter-Reihen, deren Bestimmungsgrößen – synchron gelesen – schließlich zur Struktur "zusammenschießen".
- Um aus einer gegebenen Reihe neue Varianten abzuleiten, muß sie zunächst durch eine **Zahlenfolge** substituiert werden. Nun läßt sie sich beliebig manipulieren, weil die daraus resultierenden numerischen Konstellationen jeder Interpretation offenstehen. Kompositionstechnisch wird es dadurch überhaupt erst möglich, Tonhöhen- und Dauernreihen miteinander in Beziehung zu setzen bzw. aus einer Grundreihe weitere Parameterserien abzuleiten.

¹²In der Absicht, an den "Rand des Fruchtlandes" (so der geplante Untertitel der Komposition in Bezug auf das gleichnamige Bild von Paul Klee) zu gelangen, treibt Boulez den Reihenschematismus bewußt auf die Spitze und leitet alle Reihen für Dauern, Lautstärken und Anschlagsarten aus der Struktur einer Zwölftonreihe ab. Dabei hat eine zwölfgliedrige Tonreihe ihre Berechtigung allein schon dadurch, da sie alle chromatischen Töne beinhaltet. Eine in Analogie dazu gebaute zwölfteilige Dauernreihe erscheint jedoch als willkürliche Konstruktion, die aus dem Material der Zeitdauern nicht zu rechtfertigen ist. Vollends absurd wird dies bei einer Reihe von zwölf Anschlagsarten oder zwölf Dynamikstufen, die kaum mehr differenzierbar sind.

- Die Reihe wird nicht allein im mathematischen Sinne (d.h. als Bezeichnung von Verhältnisgrößen) verwendet, sondern auch zur Bezeichnung der **Reihenfolge**. Damit wird der serielle Reihenbegriff erweitert und zugleich gesprengt: anstelle der Reihe tritt nun die Reihenfolge, deren einzelne Glieder Indizes auf eine Materialliste ("supply") darstellen. Diese beinhaltet alle möglichen Werte, die ein Parameter annehmen kann. Da die Werte dort in Form einer Skala – also nach Größe geordnet – auftreten, behält die Materialliste noch Relikte aus ihrer ursprünglichen Funktion, Verhältnisgrößen auszudrücken: die Numerierung ihrer Teilwerte entspricht gleichzeitig ihrem Größenverhältnis.

- Die Substituierung der Reihe durch eine Folge von Zahlenwerten erweitert diese um einen weiteren Aspekt: die Zahlenfolge ist nun imstande, **Mengen** zu definieren, wobei jeder Zahlenwert die Anzahl der Wiederholungen angibt, die ein Element einer Parameter-Liste erfährt. Damit wird eine Grundregel der seriellen Musik – das *Wiederholungsverbot* – angetastet und durch ein *Wiederholungsgebot* ersetzt, ohne damit aber serielles Denken preiszugeben: die Elementwiederholungen werden durch eine Multiplikationsreihe wiederum seriell gesteuert¹³.

¹³Gottfried Michael Koenig, *Aleatorische und serielle Verfahrensweisen in der Elektronischen Musik* (1965); in: Gottfried Michael Koenig, *Ästhetische Praxis*, hrsg. von Stefan Fricke und Wolf Frobenius, Bd. 2 (Saarbrücken 1992), S. 308.

Das Prinzip der Elementwiederholung beruht auf Erfahrungen bei der Realisierung elektronischer Musik. Beim Abspielen einer einzigen Grundphase eines Klanges wird dieser nur als Knacks vernommen. Erst durch mehrfache Wiederholung der Phase wird der Klang als solcher hörbar: er tritt als periodischer Vorgang in Erscheinung. Die Anzahl der Wiederholungen läßt sich durch eine Multiplikationsreihe seriell bestimmen. Je weniger periodische Vorgänge innerhalb eines Klang stattfinden, desto mehr tendiert er zum Geräusch, das sich bekanntlich durch Aperiodizität auszeichnet. Durch Vermittlung zwischen periodischen und aperiodischen Vorgängen lassen sich im Studio alle Zwischenbereiche von Klang zum Geräusch auskomponieren. Auf ähnlichen Überlegungen basiert Gottfried Michael Koenigs Kompositionsprogramm **Projekt 1** (1964 ff.)¹⁴, das als Stammvater von algorithmischen Strukturgeneratoren von besonderem Interesse ist.

Durch Einführung des Wiederholungsgebotes verliert die serielle Musik ihre "punktuellen" Charakter, der *"zugunsten übergeordneter Erlebnisqualitäten"*¹⁵ aufgehoben wird. Die von Stockhausen in seinem bahnbrechenden Aufsatz *"...wie die Zeit vergeht..."*¹⁶ formulierte Theorie der **Gruppenkomposition** stellt die Korrektur des ursprünglichen seriellen Konzepts dar, ohne aber ins vor-serielle Fahrwasser abzugleiten. Bestimmte musikalische Eigenschaften gelten nicht mehr für Einzeltöne, sondern für übergeordnete musikalische Einheiten, "Gruppen" genannt. Dadurch wird die Artikulation komplexer musikalischer Formen wieder möglich, wie es Stockhausen in seiner

¹⁴Die zuvor im Bereich des Klanges (also der "Mikro-Zeit") betrachteten Vorgänge erscheinen hier in die "Makro-Zeit" transferiert. Es werden periodische und aperiodische Texturen unterschieden. Im ersten Fall wird ein gewählter Parameterwert (z.B. eine Dynamikangabe) mehrmals wiederholt, ehe der nächste folgt. Eine rein aperiodische Textur hingegen reiht die Parameterdaten ohne Wiederholung aneinander und bildet so den klassischen Fall einer "Reihe". Zwischen diesen beiden Extreme wird in einem 7stufigen Prozeß interpoliert, der gleichsam "digitale" Übergänge zwischen periodischen und aperiodischen Vorgängen schafft. Jeder der fünf Parameter (Instrument, Rhythmus, Harmonie, Oktavlage und Dynamik) wird für einen Formteil hinsichtlich seiner Periodizität bestimmt; die Struktur und der spezifische Charakter eines Abschnittes entstehen durch Überlagerung aperiodischer bzw. periodischer Parameter-Ströme. – cf. Gottfried Michael Koenig, *Projekt Eins - Modell und Wirklichkeit* (1979); in: *Ästhetische Praxis*, a.a.O., Bd. 3, S. 223 ff.

¹⁵Karlheinz Stockhausen, *Gruppenkomposition*, a.a.O., S. 63.

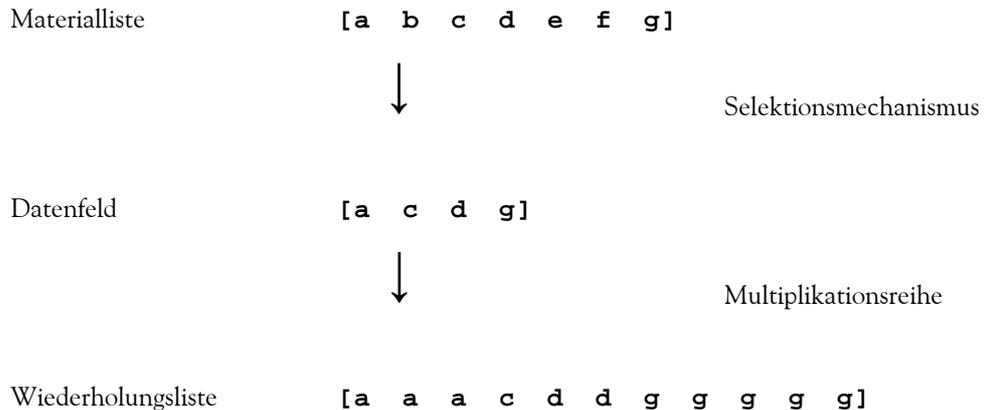
¹⁶ders., *"...wie die Zeit vergeht..."* (1956); in: *die Reihe*, Bd. III, hrsg. von Herbert Eimert unter der Mitarbeit von Karlheinz Stockhausen (Wien 1957), S. 13 ff. - Neudruck: *Texte I*, S. 99 ff.

programmatischen Komposition *Gruppen* für drei Orchester (1955-57) beispielhaft vorgeführt hat.

- In letzter Konsequenz führen diese Reihenmanipulation schließlich zur **Auflösung der Reihe** selbst. An ihre Stelle *tritt also ein Feld mit den Zuständen, die der jeweilige Parameter im jeweiligen Formteil annehmen kann. Das Feld ist Bestandteil einer Materialliste, welche sämtliche Werte beinhaltet, die der Komposition zur Verfügung stehen.*¹⁷

Für jeden Parameter (harmonische Konstellationen, Tempi, Einsatzabstände, Register, Dynamik etc.) wird nun eine Materialliste konstruiert, die alle in der Komposition vorkommenden Parameterdaten enthält. Deren Reihenfolge ist gleichgültig: sie werden ihrer Größe nach zur Skala sortiert. Nur im Grenzfall erscheint die Materialliste zur Gänze, meist aber in Auszügen, den Datenfeldern. Diese werden für jeden Abschnitt der Komposition gebildet und enthalten eine Auswahl an Elementen der Materialliste. Hinsichtlich ihrer Größe und ihres Inhaltes sind die Datenfelder variabel; sie stellen eine Selektion und damit auch eine Reduktion des zur Verfügung stehenden Ausgangsmaterials dar. Erst für die einzelnen Unterabschnitte eines Formteils wird die Häufigkeit und Anordnung der Elemente bestimmt. Aus einer amorphen Ansammlung von Daten wird so Schritt für Schritt strukturiertes Material "zubereitet".

Schematisch ließe sich dieser Vorgang so darstellen:



¹⁷Gottfried Michael Koenig, *Serielle und aleatorische Verfahrensweisen in der Elektronischen Musik* (1965); in: Gottfried Michael Koenig, *Ästhetische Praxis*, hrsg. von Stefan Fricke und Wolf Frobenius, Bd. 2 (Saarbrücken 1992), S. 308.



Permutationsregel

Ergebnis

[c d g g a g g d a g a]

Auf der linken Seite dieser Darstellung steht die Bezeichnung der Parameterlisten, in der Mitte in eckigen Klammern *eine* mögliche Erscheinungsform der Parameter, und rechts die Angabe der jeweils wirkenden Manipulationsregel.

Wir sehen hier, daß die Reihe durch eine Parameterliste ersetzt wurde. Anstatt diese Liste nun explizit auszufüllen, kann sie nun durch eine Prozedur (eine "Funktion" im Sinne der Informatik) ersetzt werden, die den Listeninhalt generiert. Dieses Verfahren hat den Vorteil, daß es allgemeiner und offener ist. Die Funktion ließe sich wiederum durch übergeordnete Parameter steuern.

1.7. Zufall

Die Auflösung der seriellen Reihenmechanik verdankt sich zudem der Erkenntnis, daß Determiniertes ins Unbestimmbare umzukippen imstande ist, wenn die Bedingungen, unter denen die Parameterwerte zusammentreffen, so kompliziert sind, daß der Komponist das Resultat im Einzelnen nicht vorausszusehen vermag. Die Ergebnisse geraten so komplex, daß die dahinter stehende Ordnung nicht mehr rezipiert werden konnte. Was zunächst als Widerspruch anmutet, ist eine Konsequenz des Seriellen: Gerade das hochgradig Organisierte verliert seine Bestimmtheit und schlägt um ins ununterscheidbar Graue, wenn die Determination zu weit getrieben wird. Den *"Umschlag allzu fortgeschrittener Differenzierung ins Indifferente"*¹⁸ bewirkt ein lebensnotwendiges Regulativ unserer Wahrnehmung: bei struktureller Überorganisation kann die Fülle von Information nicht mehr kognitiv verarbeitet werden. Dies führt zu einem Umschlagen von Ordnung in Beliebigkeit und somit zu einem Zustand, den wir als Unordnung empfinden.

Dieser Umstand zeitigte radikale Konsequenzen und führte bei Gottfried Michael Koenig zur Substituierung von seriellen Ordnungsmechanismen durch aleatorische Auswahlprinzipien –

¹⁸György Ligeti, *Form* (1966); in: Darmstädter Beiträge zur Neuen Musik, hrsg. von Ernst Thomas, Bd. X: "Form in der Neuen Musik", (Mainz1966), S. 28

und damit zur Auflösung des Reihendenkens¹⁹. Durch die bewußte Anwendung von Zufallsoperationen konnte paradoxerweise gerade das gerettet werden, was die "integrale Reihentechnik" preisgegeben hatte: nämlich die Wiedergewinnung von intendiertem Ausdruck und die tatsächliche Verfügbarkeit über die kompositorischen Mittel²⁰.

Dies führte zur Entwicklung der aleatorischen Musik. Unter Aleatorik versteht man die Anwendung von Zufallsoperationen auf Kompositionsprozesse, die ursprünglich dem Prinzip der vollständigen Determination unterworfen waren. Das Problem des Zufalls in der Musik steht dabei in unmittelbarem Zusammenhang mit der Entwicklung der abendländischen Musikgeschichte. Die ihr innewohnende Tendenz zu immer stärkerer Determination und Ausklammerung alles Zufälligen führte in der seriellen Musik zu einer Grenze, an der das Reglement so entropisch wird wie der Zufall selbst.

¹⁹ cf. Gottfried Michael Koenig, *Serielle und aleatorische Verfahrensweisen*, a.a.O., S. 300 ff.

²⁰cf. Karlheinz Essl, *Zufall und Notwendigkeit. Anmerkungen zu Gottfried Michael Koenigs "Streichquartett 1959" vor dem Hintergrund kompositionstheoretischen Überlegungen*; in: *Musik-Konzepte*, Bd. 66 "Gottfried Michael Koenig", hrsg. von Heinz-Klaus Metzger und Rainer Riehn (1989)

Die Verwendung des Zufalls in Kompositionsprozessen ist Ausdruck einer veränderten Weltansicht, die sich seit Beginn des Jahrhunderts vor allem in den Naturwissenschaften abzeichnete, wo der Newton'sche Determinismus zunehmend durch statistische Zustandsbeschreibungen ersetzt wurde. Im atomaren Bereich konnte Heisenberg nachweisen, daß durch eine "Unschärferelation" die gleichzeitige Messung verschiedener Eigenschaften eines Teilchens nie exakt sein kann, wodurch eine genaue Beschreibung unmöglich ist. Damit wurde auch mit dem Köhlerglauben aufgeräumt, die Welt sei nichts als eine komplizierte Maschine, die völlig in den Griff zu bekommen sei, fände man nur ihre exakte Formel. Sie offenbart sich vielmehr als komplexes System zwischen den Bereichen des Determinierten und Chaotischen. Jedoch sollte dies weder zur ideologischen Verherrlichung des Zufalls führen, noch zum alleinigen Vertrauen auf das rational Bestimmbare. Keinesfalls darf übersehen werden, wozu der Zufall eigentlich taugt: *"als ein Mittel zur Erweiterung des empirischen Horizontes, als ein Weg zu einer weiteren Kenntnis"*²¹ – niemals aber als Erkenntnis selbst.

Zufall und Ordnung werden im seriellen Denken nicht als Widersprüche, sondern als Extrempole aufgefaßt, zwischen denen vielfältige Übergänge möglich sind. In dieser Dialektik liegt etwas vom Geheimnis lebendiger Prozesse beschlossen: *Ordnung* als Stabilisierungsfaktor und Mittel zur Vertiefung; *Zufall* hingegen als Varianzfaktor, als Mittel zum Erkenntnisgewinn. Bildlich gesprochen: Wäre die Ameise nicht zufällig von der Ameisenstraße abgerrt, hätte sie nicht die neue Futterquelle entdecken können.

Dieses Denken zeitigt natürlich auch kompositionstechnische Korrelate. Der Vermittlung zwischen Ordnung und Zufall hat sich Koenig verschrieben: in seinem Kompositionsprogramm Projekt 1 (1964 ff.) implementierte er drei verschiedene sog. Selektionsmechanismen, die gleichsam digitale Übergänge zwischen bestimmt und unbestimmt erzeugen: SEQUENCE, SERIES und ALEA²².

1.8. ...wie die Zeit vergeht...

²¹Luigi Nono / Helmut Lachenmann, *Geschichte und Gegenwart in der Musik von heute* (1960); in: Darmstädter Beiträge zur Neuen Musik, Bd. 3, hrsg. von Wolfgang Steinecke (Mainz 1960), S. 47

²²Diese werden im Kapitel 3.3.2 ausführlich diskutiert.

Das serielle Denken nimmt seinen Ausgangspunkt in der Vorstellung von der *„Einheit der musikalischen Zeit“*, die Stockhausen in seiner Abhandlung *...wie die Zeit vergeht...*²³ proklamiert hatte. Besagte Theorie möchte aufzeigen, daß man es in der Musik primär mit zeitlichen Prozessen zu tun hat. Die Bestimmungsgrößen eines Tones – seine Höhe, Dauer und Klangfarbe – lassen sich als Zeitfunktionen beschreiben: die Frequenz als Anzahl der Grundtonschwingungen pro Sekunde, die Dauer als zeitliche Ausdehnung, zuletzt die Farbe als Überlagerung von Obertönen und Spektren in der Zeit.

Dabei handelt es sich um unterschiedliche Wahrnehmungsbereiche, in denen sich zeitliche Vorgänge in den verschiedenen Wirkungsebenen der Zeit ausdrücken: so sind Farbe und Tonhöhe im Bereich der *„Mikro-Zeit“* angesiedelt, während sich Rhythmus und Form in der *„Makro-Zeit“* abspielen. Koenig zufolge wären zum Beispiel *„fünf Minuten - eine Formeinheit, fünf Sekunden - eine Tondauer, eine fünftel Sekunde - eine rhythmische Größe, eine fünfhundertstel Sekunde - das Maß für eine Tonhöhe“*²⁴.

Die Grenzen dieser Zeitbereiche sind jedoch nicht hermetisch gegeneinander abgeriegelt, sondern permeabel. So können qualitativ unterschiedliche Parameter – etwa Rhythmus und Tonhöhe – unmerklich ineinander übergehen, wie dies bereits am Beispiel des Pingpong-Ball-Experiments demonstriert werden konnte²⁵.

Wenn aber nun Rhythmus, Tonhöhe und Klangfarbe Zeitfunktionen darstellen, die in verschiedenen Wahrnehmungsbereichen angesiedelt sind und ineinander übergehen können, so verlangt dies auch die kompositionstechnische Vereinheitlichung der Parameterbehandlung. Die gleichen Prinzipien, die im Bereich der Tonhöhen (*„Mikrozeit“*) gelten, haben auch auf dem Gebiet der Dauern (*„Makro-Zeit“*) ihre Berechtigung. So lassen sich – ebenso wie Tonhöhen – auch Dauernwerte in *„Zeitoktaven“* zusammenfassen. Die Oktave, als Verhältnis von 2 : 1 definiert, wird im temperierten chromatischen Tonsystem in 12 gleich groß empfundene Glieder unterteilt. Das Frequenzverhältnis zwischen zwei Halbtönen ist immer konstant und wird durch den *„Proportionalitätsfaktor“* ausgedrückt; die

²³Stockhausen, *...wie die Zeit vergeht...*, a.a.O.

²⁴in: Ursula Stürzbecher, *Werkstattgespräche mit Komponisten* (Köln 1971), S. 19 f.

²⁵Vgl. Kapitel 1.1 der vorliegenden Arbeit.

12. Wurzel aus 2.

Auf gleiche Weise läßt sich eine *“chromatisch temperierte Skala der Dauern”*²⁶ konstruieren. Die Dauer kann im herkömmlichen Notationssystem indes nur metronomisch bezeichnet werden, wobei die metrische Bezugseinheit (der sog. “Grundzeitwert”: zum Beispiel die ganze Note) durch Tempoangaben differenziert wird. In Stockhausens Beispiel wird eine Dauernskala im Bereich zwischen einer und einer halbe Sekunde (eine “Dauernoktav” also) konstruiert, wobei folgende Metronomzahlen, bezogen auf die ganze Note, auftreten²⁷:

MM	=	60.0	MM	=	84.9
	=	63.6		=	89.9
	=	67.3		=	95.2
				=	71.4
				=	100.9
	=	75.6		=	106.9
	=	80.1		=	113.3

Die nächste Dauernoktave liegt im Bereich einer halben und einer viertel Sekunde, also zwischen MM = 120 und MM = 226. Diese läßt sich aber ebenso durch den Wechsel der metrischen Bezugseinheit, des Grundzeitwertes, erreichen. Statt einer ganzen Note steht nun die Halbe, in der folgenden Zeitoktave die Viertel etc.

Die Verwendung von logarithmischen Zeitskalen (die also durch chromatisch-temperierte Teilung entstanden sind) garantiert ein wohlproportioniertes Ausgangsmaterial, das sich an unserer sinnlichen Wahrnehmung orientiert: wir empfinden eine solche Skala als ausgewogen und stimmig. Ein gleichmäßiges Ritardando oder Accelerando entsteht nun ebenfalls durch “temperierte” Teilung, ohne daß man sich dessen bewußt sein muß: die Verlangsamung bzw. Beschleunigung wird dann als regelmäßig erlebt, wenn das Verhältnis zwischen zwei benachbarten Zeitwerten konstant ist.

Würde man jedoch eine Zeitskala durch lineare Teilung erzeugen (bei der jeder folgende Wert

²⁶Stockhausen, *...wie die Zeit vergeht...*, a.a.O., S. 24.

²⁷Für eine zwölfwache Unterteilung von “Dauernoktaven” und Grundzeitwerten besteht freilich keine zwingende Notwendigkeit. Die Zwölfteiligkeit, als unserem westlichen Tonsystem historisch vermittelt, hat allein dort ihre Berechtigung. Diese Verhältnisse jedoch auf die übrigen Parameter zu übertragen, ist daraus nicht zu rechtfertigen, selbst wenn erkannt wurde, daß die Bildungsprinzipien der Tonhöhenreihe auch für andere Zeitfunktionen Geltung besitzen.

um den gleichen Betrag größer wird), ist das Verhältnis zwischen den Glieder kein proportionales mehr: die langen Zeitwerten werden die kurzen dominieren, was in einer langsamen Durchschnittsgeschwindigkeit resultiert.

Eine Gegenüberstellung dieser beiden Unterteilungsmöglichkeiten möge den Unterschied demonstrieren:

log. Zeitskala	lin. Zeitskala	Index
100	100	0
147	250	1
215	400	2
316	550	3
464	700	4
681	850	5
1000	1000	6

In beiden Fällen handelt es sich um einen siebenstufigen Übergang von Zeitwerten zwischen 100 und 1000 Millisekunden (ms). Während die lineare Skala konstant ansteigt, wächst die logarithmische um den gleichen Faktor. Dies führt dazu, daß die kleineren Werte dort stärker vertreten sind und die Durchschnittsgeschwindigkeit dort schneller ist, wohingegen in der linearen Zeitreihe die kleineren Werte von den größeren "aufgefressen" werden²⁸.

Diese beiden Zeitunterteilungsprinzipien lassen sich als Computerprogramm allgemein formulieren:

```
trans (start, end, steps, mode)
```

Die Funktion trans führt einen Übergang zwischen einem Startwert (start) und einem Endwert (end) in einer bestimmten Anzahl von Schritten (steps) durch, wobei zwischen mehreren Übergangsfunktionen (mode) gewählt werden kann: logarithmisch, linear oder exponentiell.

²⁸Boulez hat in seinem Klavierstück *Structure 1a* (1952) eine lineare (arithmetische) Zeitreihe verwendet (siehe György Ligeti, *Die Entscheidung und Automatik in der Structure 1a*; in: die reihe, Bd. 4, hrsg. von Herbert Eimert, Wien 1958, S. 39. - Siehe auch das Note Stockhausen, *„wie die Zeit vergeht...“*, a.a.O., S. 17). Das Problem der rhythmischen Ausdünnung löst Boulez dadurch, daß er immer mehrere Rhythmusschichten synchron ablaufen läßt, wodurch sich eine Verdichtung ergibt, die im Einzelnen aber dem Zufall der Überlagerung unterliegt.

In einem von mir seit 1989 entwickelten, auf xLOGO basierenden COMPOSE-Environment würde die Funktion

```
trans 100 1000 7 log  
als Ergebnis eine Liste
```

```
> [100 147 215 316 464 681 1000]
```

liefern, die beliebig weiterverarbeitet werden kann.

Die gleiche Funktion ist aber auch in MAX implementiert (siehe Abb. 1) und ist als solche Teil der "Real Time Composition Library for MAX". Im Unterschied zu seinem xLOGO-Pendant liefert sie aber als Ergebnis nicht die vollständige Skala, sondern nur das jeweilige durch den Index spezifizierte Listenelement. Allerdings erfolgt dort die Berechnung in "Realtime": zu dem Zeitpunkt, wo ein Listenindex in den linken Eingang der Funktion trans geschickt wird, wird der dazugehörige Listenwert als Resultat zurückgegeben.

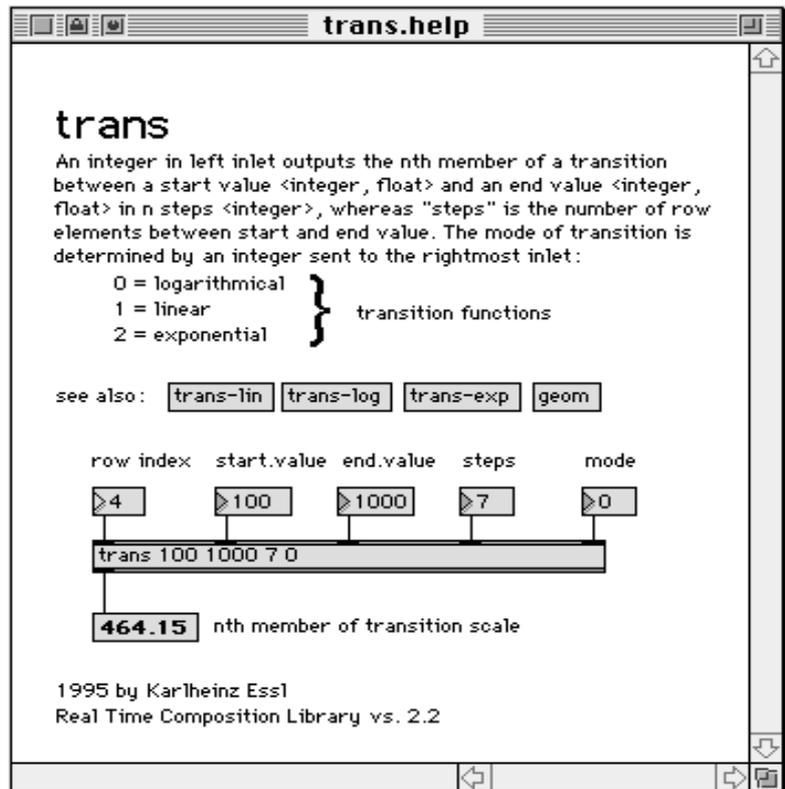


Abb. 1: Help-Window der Funktion trans
 (aus: Karlheinz Essl's "Real Time Composition Library for MAX 2.2")

2. Zur Theorie der Strukturgeneratoren

Die in der seriellen Theorie aufgeworfenen Fragen ziehen eine neue Sichtweise des musikalischen Kunstwerks nach sich. Dieses wird nun nicht mehr als einmaliger "Spezialfall" gesehen, vielmehr aber als eine von mehreren Realisierungsmöglichkeit innerhalb vorgegebener Grenzen ("constraints"). Wenn Musik nicht länger als Ikone, sondern als Modell gedacht wird, wendet sich der Blick zunächst von der sinnlichen Oberfläche (dem "Besonderen") ab und dringt ein ins Verborgene, "Allgemeine", Abstrakte: in die Zusammenhänge im Inneren eines Kunstwerkes. Diese können nun als abstraktes Modell beschrieben werden, als Gefüge von Kräften und Steuerungsmechanismen. Dieses Denken zielt darauf ab, die die Musik hervorbringende "Tiefenstruktur" (im Sinne der generativen Grammatik von Noam Chomsky) als Algorithmus zu beschreiben.

2.1. Struktur

Das Wort "Struktur" bedeutete ursprünglich "Gefüge" und bezeichnete im römischen Bauwesen den Modus, nach welchem Steine aufgeschichtet wurden. Hier geht es also um einen Beziehungszusammenhang – die Art und Weise, wie Materialien miteinander in Relation gebracht werden.

Eine philosophisch-semantische Interpretation findet sich bei Merleau-Ponty: *"Struktur ist die unlösliche Verbindung zwischen einer Idee und einer Existenz, das kontingente Arrangement, durch das Materialien vor unseren Augen einen Sinn annehmen."*²⁹ Sein Strukturbegriff zielt also nicht auf etwas Abstraktes, sondern auf eine Aussage, auf einen Sinn.

²⁹Maurice Merleau-Ponty, *Die Struktur des Verhaltens* (Berlin 1976).

Helmut Lachenmann wiederum betont den Netzcharakter des Strukturellen, der sich einer "Polyphonie von Anordnungen" verdankt: *Dem Strukturbegriff liegt eine schematische Vorstellung eines charakteristischen Gefüges, eine Art Polyphonie von Anordnungen, einer Zuordnung von wie auch immer gearteten »Familien« zugrunde, deren einzelne Familien-Glieder bei verschiedengradiger Individualität im Hinblick auf den ihnen zugeordneten Charakter als dessen Komponenten oder Varianten zusammenwirken.*³⁰ In dieser Definition klingt ein Aspekt an, der für unser Konzept der Strukturgeneratoren von Bedeutung ist: Struktur entsteht aus dem Ineinandewirken von hierarchisch abgestuften, miteinander verknüpften Teilaspekten, die – Lachenmann nennt sie "Familien" – in einem Klassenzusammenhang stehen.

Lassen Sie mich nun einen weiteren Aspekt des Strukturellen herausgreifen – den der Gestalt.

2.2. Gestalt

Nehmen wir als Beispiel das Wort »Baum«. Darunter lassen sich verschiedene Gewächse subsummieren, wie Tanne, Buche oder Weide. Der Begriff »Baum« ist zunächst ein Abstraktum: er meint keine bestimmte Pflanze, sondern eine ganze Familie von Pflanzen, die – trotz aller Unterschiede – gemeinsame Eigenschaften – eben eine charakteristische *Gestalt* – aufweisen. Das Interessante dabei ist, daß wir aufgrund des in uns implementierten Modells »Baum« auch Baumformen denken können, die in der Realität gar nicht existieren. Dies wurde vor 200 Jahre bereits von Goethe beobachtet, der seinem Freund Herder über die Idee der *Urpflanze* schreibt:

*Die Urpflanze wird das wunderlichste Geschöpf von der Welt, um welches mich die Natur selbst beneiden soll. Mit diesem Modell und dem Schlüssel dazu kann man alsdann noch Pflanzen bis ins Unendliche erfinden, die konsequent sein müssen, das heißt, die, wenn sie auch nicht existieren, doch existieren könnten und nicht etwa malerische und dichterische Schatten und Scheine sind, sondern eine innerliche Wahrheit und Notwendigkeit haben. Dasselbe Gesetz wird sich auf alles übrige Lebendige anwenden lassen.*³¹

Die Eigenschaften eines Baumes werden durch ein *Modell* – ein Inbegriff von Strukturmerkmalen – bestimmt. Eine konkrete Form (z.B. Fichte) kann auch als Strukturvariante dieses Strukturmodells »Baum« beschrieben werden.

³⁰Helmut Lachenmann, *Bedingungen des Materials. Stichworte zur Praxis der Theoriebildung*; in: Darmstädter Beiträge XVII (Mainz 1977).

³¹Johann Wolfgang von Goethe, *Italienische Reise: Brief an Herder vom 17. Mai 1787*; in: Goethes Werke (Hamburger Ausgabe), hrsg. v. Erich Trutz (München 1982/11), S. 323/324.

ließe sich gleichermaßen auch auf Musik übertragen: ein bestimmtes Modell (als Beschreibung einer kompositorischen *Gestalt*) würde die Generierung tausender verschiedener Varianten, ganz im Sinne der *Urpflanze*, erlauben.

2.3. Modellbildung – *Computer Aided Composition*

Wenn wir nun in der Lage sind, einen Baum im Sinne eines Modells zu beschreiben, dann könnte dieses Modell auch als Generator implementiert werden, der zwar keine echten Bäume, aber die verschiedensten Bilder von Bäumen erzeugen kann. Die geeignete Umsetzung wäre ein Computerprogramm, das aufgrund des implementierten Modells durch Veränderung der Parameter neue Baumformen generiert. Solche Software existiert bereits und erlaubt nicht nur die Herstellung pflanzlicher Artefakte³², sondern auch von Wolkenformationen, Landschaften und Oberflächentexturen.³³

Das gleiche Prinzip – nunmehr auf musikalische Komposition angewandt – stellt eine meiner zentralsten Arbeitsmethoden dar. Das Modell beschreibt strukturelle Verknüpfungen und die Steuerparameter, die zur Erzielung einer bestimmten musikalischen Gestalt notwendig sind.

2.4. "Instrument"

Ein solches Modell kann auch als »Instrument« aufgefaßt werden, auf dem der Komponist, der es zuvor selbst gebaut hat, spielt. So ist dieser nicht nur Tonesetzer, sondern zugleich Instrumentenbauer, Instrumentalist und Spracherfinder. Helmut Lachenmann hat in diesem Zusammenhang von der Analogie zum Orgelwerk gesprochen: "Bildlich gesprochen heißt Komponieren dann nicht nur: auf einem persönlich geprägten Instrumentarium spielen, sondern daraus ein unverwechselbares Instrument, ein »Werk« bilden, dessen Traktur durch die spezifische Anordnung und Gestaltung nicht nur der »Manuale« (der »Familien«), sondern auch der »Tasten« (der »Familienmitglieder«) so eindeutig präzisiert ist, daß der Vorgang, in dem sich das »Werk« offenbart, als eine Art »Abtastprozeß« im Grunde mit einem »Arpeggio«, wenn auch in

³²Anmerkung des Herausgebers: Die mathematische Beschreibung von Pflanzenstrukturen ist unter anderem in der Theorie der Lindenmayer-Systeme formalisiert. Siehe: Prusinkiewicz, Przemyslaw, Hanan, James: *Lindenmayer Systems, Fractals, and Plants*. - New York: Springer-Verlag, 1989. 2. Auflage 1992.

³³Bryce – Kay's *Power Tools for Photoshop*. Zu erinnern sei auch an die Lucas Film Ltd., die sich auf computergenerierte utopische Szenarien für Science Fiction Filme spezialisiert hat.

vielfach gefächerter Variante, vergleichbar ist. »Instrument«, »Werk« und »Spiel« fallen dann in eins zusammen.³⁴

³⁴nach Helmut Lachenmann, *Bedingungen des Materials*, a.a.O., S. 95.

Letztlich wird Abschied genommen von einer hermetischen Strukturvorstellung: *‘Musik hat Sinn doch nur, weil sie über ihre eigene Struktur hinausweist auf Strukturen – das heißt: Wirklichkeiten und Möglichkeiten – um uns und in uns selbst.’*³⁵

2.5. Strukturgeneratoren

Wie sich dies in der Praxis gestalten kann, möchte ich nun am Beispiel eines Strukturgenerators erläutern, der Triller generiert.

Ein Triller im traditionellen Sinn besteht aus dem schnellen Alternieren zweier benachbarter Skalentöne. Nun läßt sich das Prinzip des Trillers aber erweitern: es bleibt bei der raschen Bewegung, nur können jetzt aber mehr als 2 Töne auftreten, die zur Vermeidung von repetitiven Mustern jedoch nicht zyklisch durchlaufen, sondern unregelmäßig permutiert werden. Neben der Dauer eines Trillers ist auch seine Geschwindigkeit als variabel zu denken, und zudem können auch Geschwindigkeitsveränderungen (ritardandi, accelerandi) auftreten. Als letzter Parameter kommt noch der Dynamikverlauf dazu: an- und/oder abschwellend oder eine komplexere Hüllkurve.

Das Modell dieses Triller-Generators basiert auf einem Satz von Parameterlisten, aus denen ausgewählt werden kann.

Dauer	<min> bis <max> in <Anzahl> Schritten
Anfangs-Tempo	<max> bis <min> in <Anzahl> Schritten
End-Tempo	<max> bis <min> in <Anzahl> Schritten
Anzahl Töne	<2> bis <12>
Töne	Auswahl von <Anzahl der Töne> innerhalb einer Oktave
max. Dynamik	<min> bis <max>

Das Trillertempo wird sich – im Falle von Instrumentalmusik – am menschlichen Maß orientieren. Als Untergrenze käme ein entsprechend langsamer Wert in Frage, der eben noch gerade das Gefühl von Triller vermittelt. Hier aber wird das Modell durchlässig: verlangsamt

³⁵ ebd., S. 99.

man das Trillertempo immer weiter, verschwindet mit einem Mal der Trillercharakter und ein neuer qualitativer Zustand entsteht, der – abhängig von der Anzahl der Töne – als Akkordzerlegung oder Melodiefloskel erlebt werden kann.

Die Parameter eines Strukturgenerators können nun von einer äußeren Instanz gesteuert werden. „Steuermann“ kann entweder ein Mensch sein, der seine Vorstellungen ausdrücken will, oder aber eine „M“ (Computerprogramm, Algorithmus). Die Parameter können

- starr fixiert sein („Preset“),
- seriell permutiert (Permutationsalgorithmus) werden,
- aleatorisch bzw. stochastisch gelenkt werden (Zufallsoperationen innerhalb definierter Grenzen),
- gerichtet oder ungerichtet verändert werden,
- an kausale Mechanismen gebunden sein oder auch nicht,
- chaotischen Mechanismen (Autopoïese) unterworfen sein.³⁶

Auf diese Weise läßt sich ein weites Feld von Strukturvarianten erzeugen, dessen Variabilität durch verschiedensten Methoden gesteuert werden kann.

Ein Strukturgenerator (wie der eben beschriebene Triller-Generator) existiert zunächst als reines Gedankenkonzept. In einem weiteren Arbeitsschritt seine Umwandlung in ein Computerprogramm erfährt. Die Wahl der Programmiersprache spielt dabei eine nicht unwesentliche Rolle, wenngleich sich die meisten Algorithmen in (fast) jeder formalen Sprache beschreiben lassen. Soll jedoch der Strukturgenerator Resultate in Echtzeit liefern, ist man in der Wahl der Mittel sehr eingeschränkt. Aus diesem Grund habe ich mich hier auf die Programmiersprache MAX³⁷ konzentriert: eine in der 1980er Jahre entwickelte graphische Entwicklungsumgebung für Echtzeit-Applikationen. Im Unterschied zu Fertiglösungen wie kommerzielle Software-Sequenzern oder Notationsprogrammen handelt es sich hier um eine Programmiersprache, in der musikalische Applikationen geschrieben werden können.

In MAX habe ich nun die sog. „Real Time Composition Library“ entwickelt, die die Programmierung von Strukturgeneratoren auf höchster Ebene gestattet. Die

³⁶ nach: Helmut Lachenmann, *Bedingungen des Materials*, a.a.O., S. 95.

³⁷ MAX – An Interactive Graphic Programming Environment by Miller Puckette and David Zicarelli (IRCAM/Opcode 1988)

Benutzeroberfläche des eben beschriebenen Triller-Generators würde darin folgendermaßen aussehen:

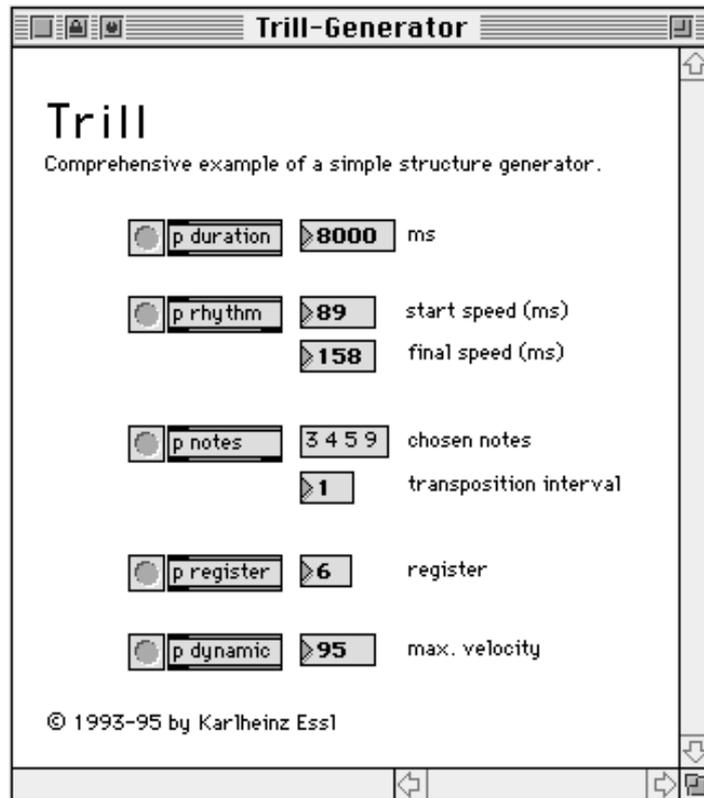


Abb. 2: Benutzeroberfläche eines Triller-Generators

3. "Real Time Composition Library"

Eine Softwarebibliothek zum Design von Strukturgeneratoren

3.1. Komponieren mit Computerunterstützung

Es existieren im Wesentlichen zwei grundsätzliche Methoden, den Computer als Werkzeug im Komposition einzusetzen. Die am häufigsten praktizierte verwendet den Rechner als komfortable Schreibmaschine, musikalische Einfälle aufgezeichnet und manipuliert werden können. Als Ergebnis liefern sie entweder elektronisch erstellte Partitur, die dann von Musikern gespielt wird (Notationsprogramm), oder eine F Steueranweisungen, mit der elektronische Musikinstrumente angesteuert werden (Sequenzler).

Die andere Methode, der ich mich ausschließlich widme, sieht den Computer als experimentelle Werkb kompositorische Ideen zu entwickeln und erproben. Mit Hilfe einer geeigneten Programmiersprache la kompositorische Algorithmen formulieren und umsetzen. In der Interaktion mit dem Computer kann der K die Tragweite seiner kompositorischen Absichten erkennen, ohne jedoch das Zepter aus der Hand zu geben. den Resultaten nicht einverstanden, kann er das Programm solange modifizieren, bis Intention und Ergo Deckung gebracht sind. Dabei umschließt das künstlerische Subjekt als erste und letzte Instanz wie eine K den gesamten Kompositionsprozeß: es entwirft ein Modell, analysiert dessen Output und gewinnt dadu Erkenntnisse, die dazu führen mögen, die Ansätze neu zu formulieren oder das Modell zu modifizieren. So Computer zu einer — selbstgeschaffenen — Kontrollinstanz werden, zum unbestechlichen Spiegel der Vorstellung.

3.2. MAX

Als ich 1992 ans IRCAM kam, um dort an einem Kompositionsauftrag³⁸ für Ensemble und die "IRCAM Workstation" (ISPW)³⁹ zu arbeiten, lernte ich auch die Programmiersprache MAX in kennen, welche die gr Benutzeroberfläche der ISPW darstellt.

³⁸*Entsagung* (1991-93) für Flöte, Baßklarinette, präpariertes Klavier, Schlagzeug und 4-kanaliges interaktives Klang-Environment (Musikverlags GmbH, Darmstadt).

³⁹Die ISPW basiert auf einem NeXT-Computer mit spezieller DSP-Hardware und erlaubt die Synthese und Manipulation von Echtzeit.

Die Entwicklung von MAX⁴⁰ wurde 1986 am IRCAM begonnen, als man das Paradigma eines Analogstudios⁴¹, dessen einzelnen Komponenten durch Kabelverbindungen zu komplexeren "Patches" verschaltet werden können, auf eine graphische Programmierumgebung übertrug. Die grundlegende Absicht war es, ein Echtzeitsystem zu entwickeln, das vor allem im Live-Einsatz Verwendung finden sollte. Aus einem kleinen Vorrat von System-"Primitives" (graphischen Objekten mit Ein- und Ausgängen) lassen sich durch Verbindungen mit sog. "patch chords" höherstrukturierte Objekte erzeugen. Ein solcher "Patch", dessen interne Komplexität hinter einer "object box" verborgen werden kann, kann nun als eigenes Objekt angesprochen werden. Dadurch lassen sich komplizierte Strukturen aus einigen hochspezialisierten Objekten (Unterprogrammen vergleichbar) erzeugen, die wiederum aus untergeordneten Objekten bestehen usw.

Solche Objekte, die Erweiterungen des Sprachschatzes darstellen, können nun in verschiedenen Kontexten verwendet werden, ohne daß man sich jeweils die genaue Kenntnis ihrer internen Organisation vor Augen führen muß. Sie fungieren als "black boxes", deren Verhalten man zwar genau kennen soll, ohne aber über die programmtechnischen Details Bescheid wissen zu müssen.

3.3. RTC-lib

⁴⁰Bei MAX (ursprünglich "Patcher" genannt) handelt es sich um eine objekt-orientierte Entwicklungsumgebung für Echtzeitapplikationen. Sie wurde von 1987 von Miller Puckette entworfen und später von David Zicarelli weiterentwickelt. Die Software existiert in zwei unterschiedlichen Varianten: in einer mit Klangverarbeitungs-Funktionen angereicherten Fassung dient sie als Benutzeroberfläche der ISPW, während die von OPCODE Ltd. vertriebene stark erweiterte Macintosh-Version in erster Linie für die Generierung und Manipulation von MIDI-Daten Verwendung findet.

⁴¹Im Analogstudio ließ sich die Idee der Programmierbarkeit von Musik, wie sie die serielle Kompositionstheorie als Möglichkeit anvisiert hatte, erstmals realisieren. Die einzelnen Komponenten (Oszillatoren, Filter, Funktions- und Hüllkurvengeneratoren etc.) werden mittels Steuerspannungen kontrolliert ("voltage control"), können aber wiederum auch selbst Steuerspannungen erzeugen. Diese Ambivalenz ermöglicht die Konstruktion kybernetischer Maschinen, die sich selbst kontrollieren und dabei Klangstrukturen erzeugen (wie etwa in Gottfried Michael Koenigs zwischen 1967 und 1969 entstandene Werkreihe der *Funktionen*). Eines der wichtigsten Studios dieser Art wurde unter der Leitung von Koenig am "Instituut voor Sonologie" in Utrecht aufgebaut. Der allseits bekannte Moog-Synthesizer stellt die kompakte und bühnentaugliche Version eines Analogstudios dar, wenngleich er mehr zur Erzeugung von "sounds" verwendet wurde als zur Generierung musikalischer Strukturen.

MAX dient in erster Linie zur Manipulation von MIDI⁴²-Daten, also Steuerinformationen für elektronische Musikinstrumente. Was bislang aber fehlte, war eine Bibliothek mit speziellen Funktionen für "Computer Aided Composition". Diese Lücke füllt nun die seit 1992 entwickelte "Real Time Composition Library for MAX". Mit dieser Bibliothek von Softwaremodulen lassen sich Kompositionsalgorithmen auf einer hohen Abstraktionsebene formulieren, ohne daß der Benutzer sich mit untergeordneter Systemprogrammierung ablagen muß.

Als Beispiel sei etwa das Objekt `scale` angeführt, das Werte aus einem bestimmten Eingangsintervall in ein bestimmtes Ausgangsintervall umrechnet ("skaliert"):

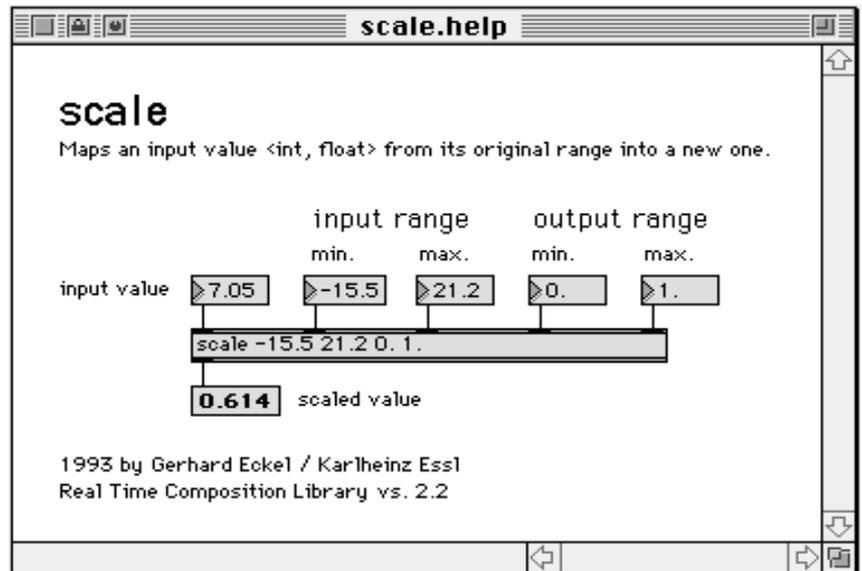


Abb. 3: Help-File der Funktion `scale`

Im Inneren dieser Funktion verbirgt sich eine aus MAX-Primitives konstruierte Programmstruktur, die implementiert – in den verschiedensten MAX-Patches verwendet werden kann ("re-usability"), ohne daß jedesmal den Berechnungsalgorithmus vergegenwärtigen muß. Ist die Funktion einmal ausgetestet und ihr Verhalten bekannt, kann sie wie eine Systemfunktion benutzt werden.

⁴²MIDI (Musical Instruments Digital Interface) – ein 1983 von verschiedenen Synthesizerherstellern entwickeltes standardisiertes Protokoll zur Steuerung elektronischer Musikinstrumente.

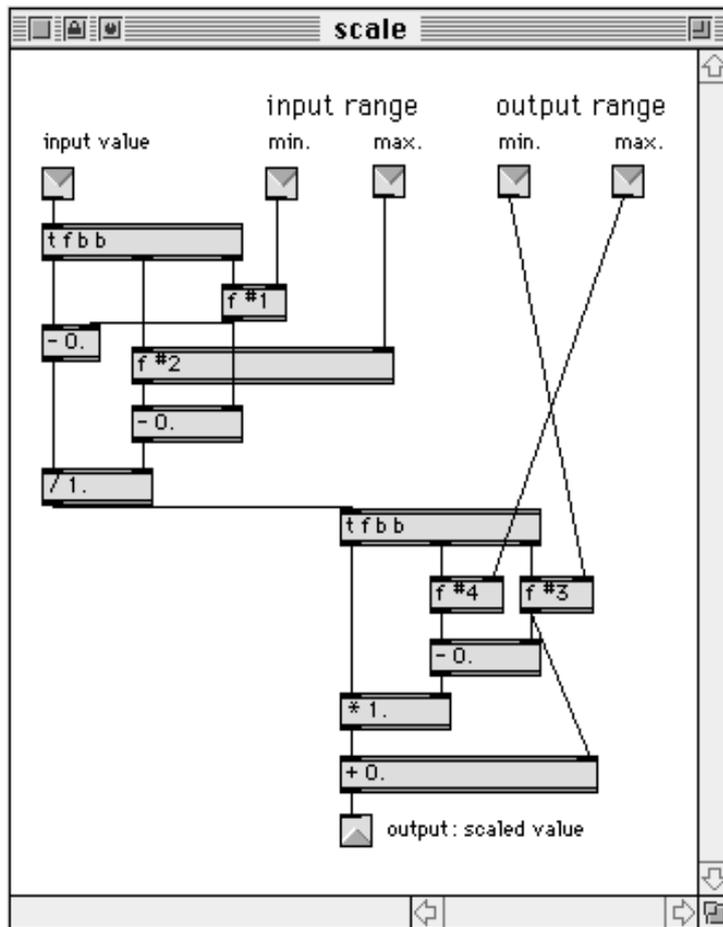


Abb. 4: Das verborgene Innenleben von *scale*

Das Prinzip, durch Programmierung von Softwaremodulen den Grundwortschatz einer Computersprache zu erweitern, habe ich bereits in meinem auf xLOGO basierendem COMPOSE-Environment (1988 ff.) angewandt. Dieses System verwende ich in erster Linie zur Generierung von "score listings", deren symbolisch-numerische Erscheinung zunächst in musikalische Notation übersetzt werden müssen, um zu klanglichen Ergebnissen zu gelangen⁴³.

In MAX hingegen bietet sich die Möglichkeit, alle Operationen in Echtzeit durchzuführen. Damit läßt sich nunmehr ein lang gehegter Traum verwirklichen: die Schaffung einer experimentellen Werkbank zur Entwicklung und Erprobung kompositorischer Strategien, deren Ergebnisse sofort hörbar gemacht werden können. Diese Unmittelbarkeit bringt naturgemäß enorme Vorteile für den Entwicklungsprozeß einer Komposition, da augenblicklich sinnliche Erfahrungen gesammelt werden können. Dadurch läßt sich eine zunächst nur grob skizzierte kompositorische Idee schrittweise verfeinern, bis Intention und Resultat – in einem Rückkopplungsprozeß zwischen Komponist und Computer – zur Deckung gebracht werden ("rapid prototyping").

⁴³cf. Karlheinz Essl, *Computer Aided Composition*; in: Distel, Nr. 46/47 "Mensch Maschine" (Bozen 1991).

Der unmittelbare Respons eröffnet aber auch die Möglichkeit, Musikautomaten zu erfinden, die Musik in einem immerwährender Prozeß in Echtzeit generieren⁴⁴. Diesen Ansatz nannte ich einmal scherzhaft "Realtime Composition", wohl um die Paradoxie dieses Wortpaares wissend. Mittlerweile hat sich dieser Begriff aber eingebürgert, und eine ganze Bibliothek von Softwaremodulen – eben die "Real Time Composition Library" (RTC-lib) trägt diesen Namen.

Die "Real Time Composition Library" stellt ein work-in-progress dar und erscheint mittlerweile in der Version 2.2. Die Arbeit daran wurde 1992 begonnen, als ich gemeinsam mit Gerhard Eckel (IRCAM) Überlegungen anstellte, welche grundlegenden MAX-Funktionen für musikalische Komposition notwendig wären. Ihm verdanke ich auch die Entwicklung einiger toolbox-Objekte und die Einführung in einen strukturierten und objekt-orientierten Programmierstil.

Die Version 1.0 wurde ursprünglich vom IRCAM im Rahmen der "IRCAM Usergroup" vertrieben. Die mittlerweile als Version 2.2. erscheinende RTC-lib (verbessert und stark erweitert, mit Hypertext-artiger Online-Hilfe, Tutorial und Anwendungsbeispielen) ist nunmehr Public Domain-Software und kann im Internet von verschiedenen ftp-servern bezogen werden⁴⁵. Neben Gerhard Eckel haben auch andere Personen Beiträge zur RTC-lib geleistet, denen an dieser Stelle gedankt sein soll: allen voran Peter Elsea (University of California, Santa Barbara), der manche bereits bestehende RTC-Objekte als Externals in C implementierte und die Erlaubnis gab, einige seiner Listenobjekte in die RTC-lib zu integrieren; weiters Orm Finnendahl (Berlin), James McCartney (University of Texas), David Zicarelli (IRCAM / Paris), Gary Lee Nelson (Oberlin College) und Charles Baker.

Die RTC-lib ist in ständiger Veränderung begriffen und reflektiert den momentanen Stand meiner kompositionstheoretischen Auseinandersetzung. Ihre Charakteristika seien hier in Kürze zusammengefaßt:

- ein offenes und erweiterbares, objekt-orientiertes "Realtime Programming Environment" anstelle eines starren, geschlossenen Computerprogrammes,
- modular aufgebaut,

⁴⁴Als Beispiel dafür werde ich im Kapitel 4 die *Lexikon-Sonate* (1992 ff.) – eine unendliche und interaktive Realtime-Komposition für computergesteuertes Klavier – diskutieren.

⁴⁵Nähere Informationen finden sich im Anhang des vorliegenden Textes.

- die RTC-Objekte können in verschiedensten Kontexten verwendet werden (“re-usability of code”),
- Komplexität kann durch Kapselung (encapsulation) verborgen werden.

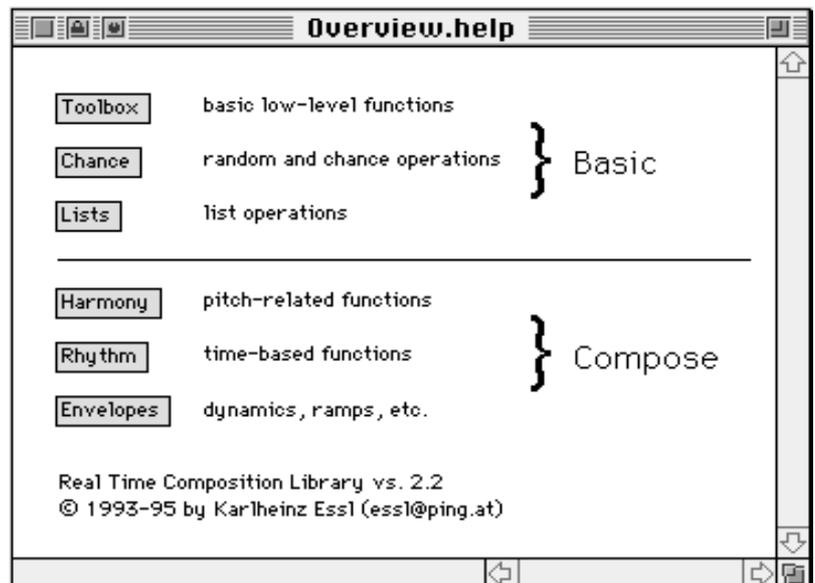


Abb. 5: Aufbau der RTC-lib

Die “Real Time Composition Library” besteht aus 145 Objekten, die sich in zwei grundsätzliche Kategorien unterlassen: **Basic**-Objekte für die Bewältigung programmtechnischer Probleme und **Compose**-Objekte für kompositorische Fragestellungen. Diese Grundkategorien lassen sich nun weiter klassifizieren.

BASIC

Toolbox:	grundlegende low-level Funktionen
Chance:	Zufallsoperationen
Lists:	Listenoperationen ⁴⁶

⁴⁶Um bestimmte Funktionen – zum Beispiel zufallsbasierte Selektionsmechanismen – zu implementieren, war es notwendig, Listenoperationen zu erweitern. Diese waren ursprünglich nur in rudimentärster Ausprägung im Sprachkonzept von MAX enthalten. Index-Operationen (wie wir sie im Kapitel über serielle Theorie diskutiert hatten) lassen sich aber ohne das Konzept der Liste (oder des “arrays”, wie es in Programmiersprachen heißt) nicht realisieren. Deshalb konnte die Entwicklung der RTC-lib erst in großem Umfang gestartet werden, nachdem MAX um bestimmte Listenoperationen erweitert wurde. Hier bin ich vor allem Serge Lemouton (IRCAM) zu größtem Dank verpflichtet.

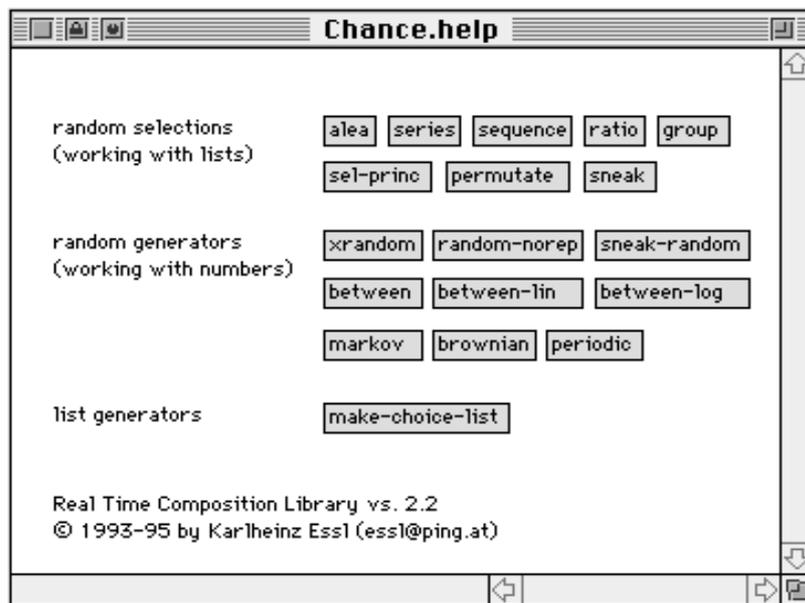
COMPOSE

Harmony:	tonhöhenbezogene Funktionen
Rhythm:	rhythmus- und zeitbezogene Funktionen
Envelopes:	Hüllkurven und zeitvariante Funktionen ("ramps")

Die Library erlaubt es dem Anwender, sich mehr auf die kompositorischen Fragestellungen zu konzentrieren, da die Entwicklung von Programmstrukturen auf einem sehr hohen Level erfolgt. Dafür werden dem Benutzer eine Vielzahl von Werkzeugen zur Verfügung gestellt. Davon möchte ich im Folgenden nun einige Kategorien exemplarisch herausgreifen – und zwar Zufallsoperationen, Rhythmus- und Harmoniegeneratoren.

3.3.1. Zufallsoperationen

Die nachfolgende Übersicht zeigt alle Zufallsoperationen, die bislang in der RTC-lib verfügbar sind:



mit der Implementierung eines in C geschriebenen "externals" namens nth (das das n-te Element aus einer Liste ausliest) den Grundstein für viele von mir entwickelten Listenoperationen gelegt hat.

Abb. 6: Zufallsoperationen in der RTC-lib

Es lassen sich zwei grundsätzliche Kategorien unterscheiden: a) Zufallsgeneratoren und b) Selektionsmechanismen.

ad a) Zufallsgeneratoren generieren Zufallszahlen innerhalb eines gegebenen Wertebereichs. Das Verhalten wird gelegentlich durch Zusatzparameter beeinflusst werden. Um die verschiedenen Generatoren miteinander zu vergleichen, wurden von jedem Algorithmus 40 Zufallszahlen zwischen 0 und 5 berechnet, und die Ergebnisse in einem Raster dargestellt.

Der einfachste Zufallsgenerator – `between` – liefert jede beliebige Zahl aus einem angegebenen Intervall. Er verkörpert so den "blinden Zufall". Betrachten wir die Verteilung der Zahlen über die Zeit, so erkennen wir keine

Regelmäßigkeiten

oder Strukturen.

Sicher ist nur, daß

im statistischen

Mittel alle Werte

gleich häufig

auftreten.

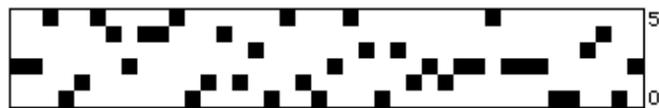


Abb. 7a: `between`

Eine etwas "intelligere" Variante stellt `xrandom` dar. Aufgrund eines implementierten *Wiederholungsverbot*s darf eine Zahl erst dann wieder ausgegeben werden, wenn alle anderen bereits vorgekommen sind (worin sich noch altes serielles Denken widerspiegelt). Die Wiederholung gleicher Werte kann nur an den "Nahtstellen" erfolgen (in untenstehender Graphik eingekreist) – wenn also die letzte Zahl eines Durchlaufs und die erste des nächsten zufälligerweise gleich sind.

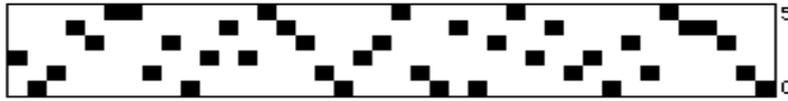


Abb. 7b: xrandom

Der in der serielle Gruppentheorie durch Koenig eingeführte Begriff der "Periodizität"⁴⁷ hat im Zufallsgenerator `periodic` seinen Niederschlag gefunden. Hier wird das in `xrandom` geltende Wiederholungsverbot durch ein *Wiederholungsgebot* ersetzt. Ein Periodizitätsfaktor (zwischen 0 und 1) bestimmt, wie oft ein zufällig gewählter Wert wiederholt werden kann, ehe ein neuer berechnet wird. Die durchschnittliche Gruppengröße steigt also mit dem Periodizitätsfaktor; ist dieser 1, wird immerfort der gleiche Wert repetiert, ist er hingegen 0, herrscht wiederum das Wiederholungsverbot.

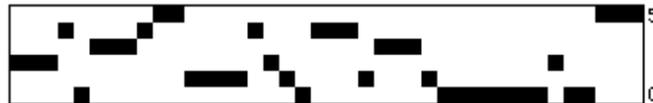


Abb. 7c: periodic

In den drei zuvor beobachteten Zufallsgeneratoren spielte für die Generierung der nächsten Zahl die vorhergehende keine Rolle. Anders in `brownian`, wo die Idee der Brown'schen Molekularbewegung ("random walk") umgesetzt wurde. Ein zusätzlicher Parameter (der sog. "brown factor", auch als Freiheitsgrad bezeichnet) bestimmt hier, wie groß das Intervall zwischen zwei benachbarten Zufallszahlen sein darf. Ein Faktor von 1 bedeutet, das Intervall zweier benachbarter Zahlen beliebig sein kann, was exakt dem Verhalten des "blinden" Zufallsgenerators `between` entspricht. Ein Faktor von 0 wiederum würde dazu führen, daß immer nur eine einzige Zahl ausgegeben werden kann. Bei einem entsprechend gewählten "brown factor" (0.27 in untenstehender Graphik) kann das Verhalten so eingestellt werden, daß immer nur eng benachbarte Zahlenwerte generiert werden können.

⁴⁷Siehe Kapitel 1.6 ("Reihenmanipulationen").

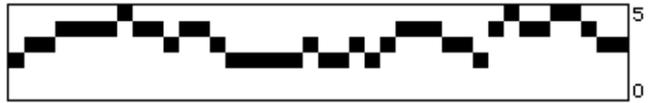


Abb. 7d: brownian

ad b) Selektionsmechanismen hingegen wählen Elemente aus einem gegebenem Repertoire ("supply") aus, deshalb auf einer höheren Abstraktionsebene: Das Augenmerk richtet sich hier auf eine Materialkonstellation (eine Liste), die entweder frei definiert werden kann, oder (funktionell gedacht) wieder Hilfe einer listengenerierenden Funktion erzeugt werden kann.

Die von Koenig in *Projekt 1* und *Projekt 2* definierten Selektionsmechanismen *alea*, *series*, *sequence*, *ratio* und *group* wurden für die RTC-lib neu in MAX implementiert. So liefert beispielsweise *sequence* jeweils nächste Element der Repertoireliste,

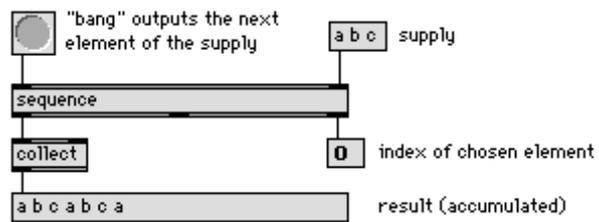


Abb. 8: sequence

während *alea* jedes beliebige Element ausgeben kann.

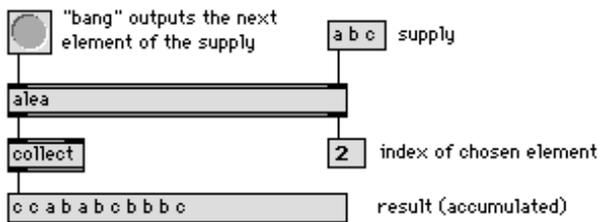


Abb. 9: alea

3.3.2. Rhythmus

Bevor ich mich nun den Rhythmusgeneratoren widme, möchte ich zunächst einige Anmerkungen zur Nomenklatur machen:

Der Abstand zwischen zwei rhythmischen Impulsen wird Einsatzabstand ("entry delay") genannt und ist eine relative Zeitangabe. Die absolute Zeitbestimmung wäre nun der Einsatzpunkt ("entry point"), den man durch sukzessives Aufaddieren der Einsatzabstände erhält. Die Dauer wiederum bezeichnet, wie lange ein Ereignis zu einem gegebenen Einsatzpunkt dauert. Als zusätzliche Bestimmungsgröße des Rhythmus ließe sich auch die Akkordgröße (vertikale Dichte) bestimmen, die angibt, wie viele Ereignisse ("events") an einem Einsatzpunkt zugleich stattfinden.

Folgende "score list" eines kurzen rhythmischen Ablaufs:

Einsatzabstand	Akkordgröße	Dauer
0.0	3	0.8
1.2	1	3.0
0.8	2	0.1
1.5	1	1.7
0.2	4	0.7

würde folgendes – graphisch interpretiertes – Ergebnis liefern:

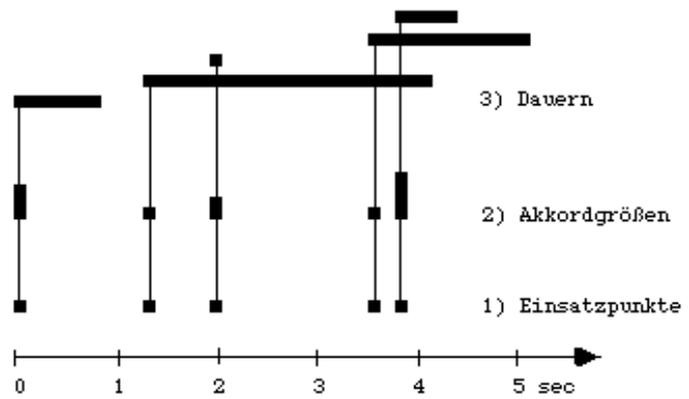


Abb. 10:

Rhythmus-Parameter

Die drei den Rhythmus beschreibenden Parameter (Einsatzabstand, Akkordgröße und Dauer) wurden graphischen Repräsentation voneinander isoliert aufgezeichnet, fallen aber in der musikalischen Realität zusammen und bilden ein Ganzes.

Rhythmusgeneratoren

In der RTC-lib gibt es eine Reihe von Rhythmusgeneratoren mit unterschiedlichen Eigenschaften. Hier sind verschiedene kompositionstheoretische Ansätze als Modelle implementiert, deren Parameter innerhalb festgelegter Grenzen verändert werden können.

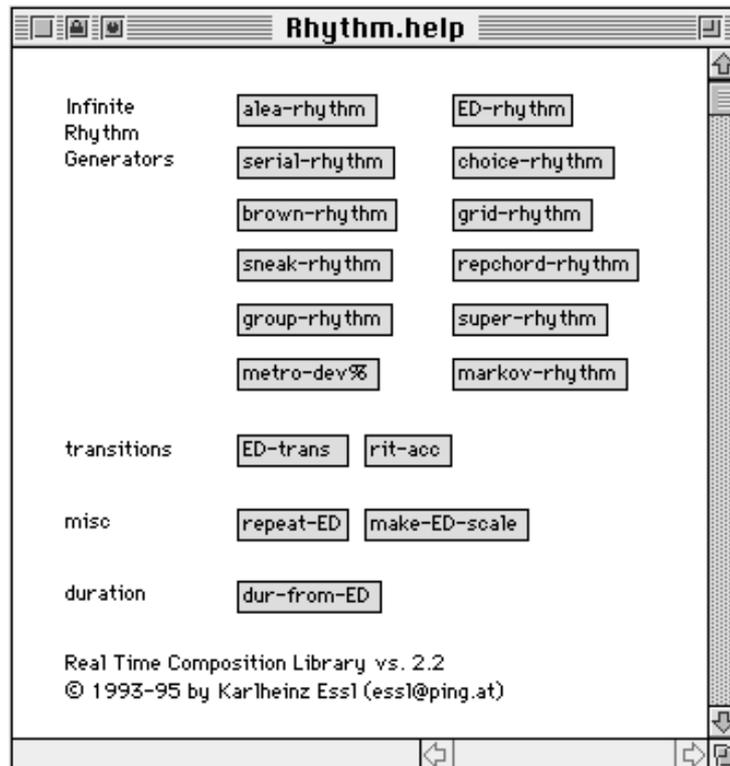
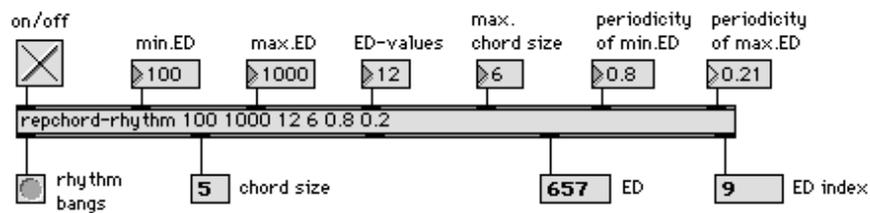


Abb. 11: Rhythmusobjekte der RTC-lib

Aus den oben angeführten Rhythmusgeneratoren möchte ich nun einen einzelnen herausgreifen: repchord-



rhythm.

Abb. 12: repchord-rhythm

Es handelt sich dabei um einen “unendlichen” Rhythmusgenerator, der – einmal eingeschaltet – fortwährend eine charakteristische rhythmische Faktur erzeugt. Diese ist abhängig von den 6 Parametern:

- **min.ED** (“minimum entry delay”) : kleinster Einsatzabstand (ms)⁴⁸
- **max.ED** (“maximum entry delay”): größter Einsatzabstand (ms)
- **nr.of.ED** (“number of entry delays”): Anzahl der Einsatzabstände
- **max.chordsize**: maximale Akkordgröße
- **periodicity of min.ED**: Periodizität des kleinsten Einsatzabstandes
- **periodicity of max.ED**: Periodizität des größten Einsatzabstandes

Zunächst wird als rhythmisches Ausgangsmaterial eine geometrische Reihe aus **min.ED** und **max.nr.of.ED** Gliedern gebildet⁴⁹. Aus diesem Vorrat werden mit Hilfe des Selektionsmechanismus **alea** Einsatzabstände ausgewählt.

Die Entscheidung, wie oft nun der gewählte rhythmische Wert wiederholt wird, ist von den Periodizitätsfaktoren abhängig. In der Regel wird man den Periodizitätsgrad der **min.ED** relativ hoch ansetzen, während der von **max.ED** geringer zu veranschlagen ist. Das würde bedeuten, daß schnellere rhythmische Ketten bilden (also ein periodischeres Verhalten an den Tag legen), während lange Notenwerte wiederholt werden.

Zuletzt wird in Abhängigkeit zum gewählten Einsatzabstand noch die Akkordgröße des rhythmischen Wertes bestimmt. Dabei herrscht folgende Relation: große Einsatzabstände können Akkordgrößen bis zur **max.chordsize** aufweisen, während der kleinste rhythmische Wert nur Akkordgrößen von 1 (= Einstimmigkeit) zugewiesen bekommt.

Aus diesem Modell lassen sich durch Veränderung der Parameter unterschiedlichste Varianten bilden, die wiederum unendliche viele rhythmische Gestalten erzeugen können.

⁴⁸Rhythmische Werte werden in der RTC-lib nie in metrischen Einheiten (z.B. Viertel, Achtel etc.) angegeben, sondern immer Zeitangaben in Millisekunden (ms). Da wir uns hier in einem Bereich außerhalb der traditionellen Notenschrift befinden, können wir uns frei Zahlen bewegen, ohne auf metrische Einschränkungen Rücksicht nehmen zu müssen.

⁴⁹Siehe den Abschnitt über Zeitreihen im Kapitel 1.8.

3.3.3. Harmonik

Ein rhythmischer Ablauf, wie ihn ein Rhythmusgenerator der RTC-lib generieren kann, bedarf zu seiner Konkretisierung – Versinnlichung! – eines weiteren Parameters: den des Klanges. Erst im Klang selbst manifestiert sich das Rhythmische als sinnlich wahrnehmbare Gestalt.

Klang läßt sich nun ebenfalls parametrisch beschreiben. Ein wichtiger Aspekt ist seine Frequenz, spezifischer: seine Tonhöhe. Tonhöhe tritt nur in den seltensten Fällen als isolierte Größe auf – wir hören sie immer in Relation zueinander, und konstruieren uns bei Hören harmonische Bezüge. »Harmonik« bezieht sich nun eben nicht auf das punktuelle Ereignis, sondern immer auf den Kontext – also die Beziehungen zwischen den Tönen bzw. Klängen.

Die RTC-lib beinhaltet nun eine ganze Reihe von Funktionen, um harmonische Abläufe zu gestalten.

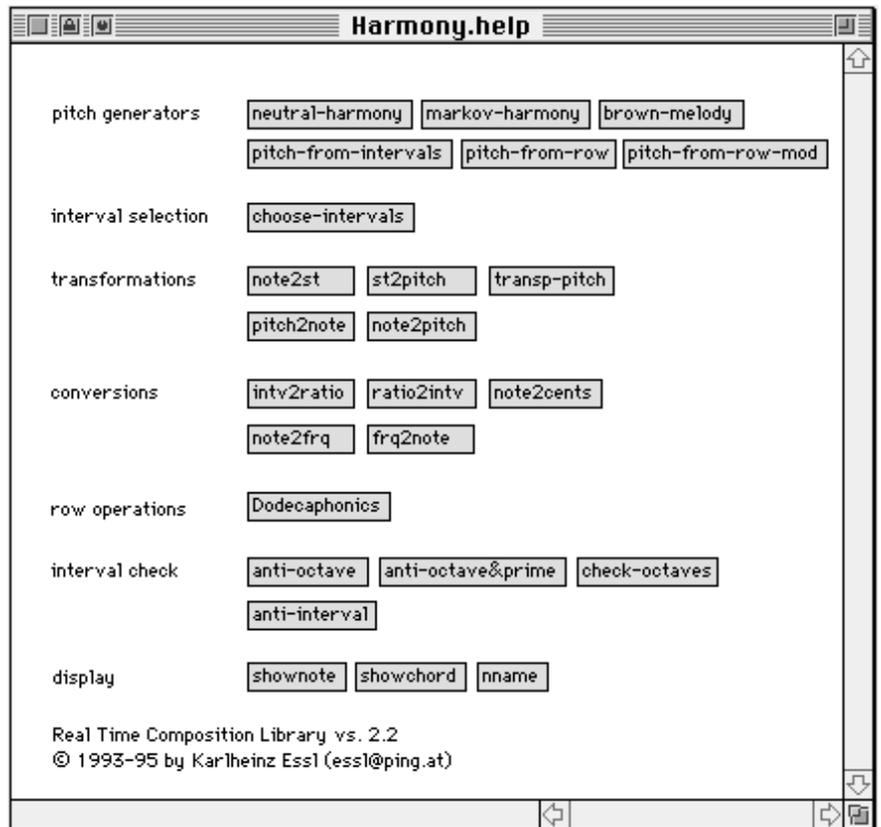


Abb. 13: Harmonie-Objekte der RTC-lib

An dieser Stelle noch ein Exkurs zur Nomenklatur:

- **pitch (class):** "abstrakte" Tonqualität ohne Registerangabe
z.B. c, f#
- **note:** "konkrete" Tonhöhe mit Registerangabe
z.B. c1, F#
- **register:** Oktavlage
z.B. 5 = eingestrichene Oktave
- **interval:** Abstand zwischen zwei "notes", gemessen in Halbtonschritten
z.B. -1 = absteigende kleine Sekunde
19 = aufsteigende Duodezime
0.5 = ein Viertelton aufwärts

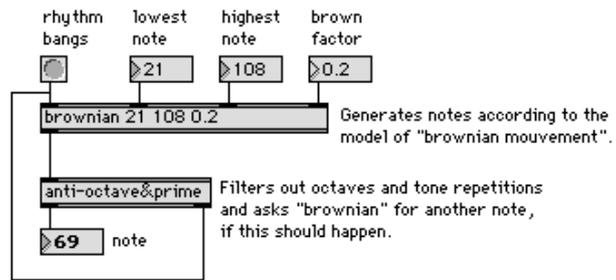
Zur Veranschaulichung möchte ich nun drei verschiedene Prämissen zur Generierung harmonischer Strukturen vorstellen: ausgehend a) von Tonhöhen ("notes"), b) von Tonqualitäten ("pitch classes"), oder c) von Intervallen. Dies werde ich im Folgenden an Hand von drei verschiedenen Harmoniegeneratoren erläutern, von denen allerdings nur der erste in der RTC-lib aufscheint. Die anderen wurden ausschließlich für diese Publikation und vor allem in Hinblick auf Anschaulichkeit konstruiert.

ad a) Tonhöhenbasierte Harmoniegeneratoren

Nehmen wir nochmals das Modell der "Brown'schen Molekularbewegung", das wir zuvor im Zusammenhang mit den Zufallsgeneratoren diskutiert haben. Dieses Prinzip läßt sich nun auf die Generierung von Tonhöhen übertragen. Als Wertebereich nehmen wir z.B. den Ambitus eines Klaviers, dessen 88 Tasten den MIDI-Notennummern 21 bis 108 entsprechen. Mit Hilfe des "brown factors" läßt sich nun regulieren, wie groß das Intervall zwischen zwei generierten Tönen maximal sein darf.

Da dieser Ansatz nicht harmonisch (also kontextbezogen) konzipiert ist, treten eine Reihe von möglicherweise

unerwünschten Nebeneffekten auf wie etwa “unmotivierte” Tonwiederholungen oder Oktaven. Diese lassen



herausfiltern, wie es im RTC-Objekt `brown-melody` der Fall ist:

Abb. 14: brown-melody

Dieses einfache Mittel der Filterung verleiht dem, auf einem “dummen” Algorithmus basierten Tonhöhengenerator, mit einem Male Charakter und Ausgewogenheit.

ad b) Tonqualitätenbasierte Harmoniegeneratoren

Gehen wir nun von einer vorgegebenen Skala von Tonqualitäten aus, die als Material für eine harmonische Skala dient. Diese Skala wird in einer Liste zusammengefaßt. Aus diesem “supply” werden nun mit Hilfe eines Selektionsmechanismus’ die “pitch class” ausgewählt.

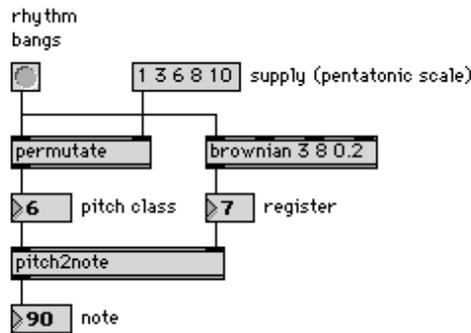


Abb. 15: scale-harmony

Der hier verwendete Selektionsmechanismus heißt `permutate`: die im “supply” enthaltenen Werte werden zufällig ausgelesen, wobei jedoch direkte Wiederholungen des selben Elements ausgeschlossen sind. Das Resultat ist zunächst aber noch eine nackte Tonqualität (“pitch class”) ohne konkrete Positionierung im Tonraum. Dies besorgt nun die Registerangabe, die wieder mittels `brownian` ermittelt wird. Das Objekt `pitch2note` faßt pitch classes und “register” zusammen und berechnet daraus die Tonhöhe (“note”).

Den “supply” haben wir in diesem Fall explizit als pentatonische Skala übergeben. Stattdessen ließe sich aber auch eine Funktion erfinden, die verschiedenste Skalen algorithmisch konstruiert und zur Verfügung stellt. Man könnte noch einen Schritt weitergehen und ein weiteres Objekt gestalten, das eine gegebene Tonskala (also den “supply”) sukzessive verändert, indem es etwa immer ein Element der Liste entweder zufügt oder löscht⁵⁰.

ad c) Intervallbasierte Harmoniegeneratoren

Schreiten wir zuletzt zu dem abstraktesten, gleichwohl aber reizvollsten Möglichkeit, der Harmonieerzeugung mit Hilfe von Intervallen.

Wir beginnen wieder mit einem gegebenen Material, einer Liste von Intervallen. Diese wird nun aber in diesem Beispiel nicht mehr händisch eingegeben, sondern von dem RTC-Objekt `choose-intervals` bestimmt⁵¹. Dieses

⁵⁰Ein solches Objekt namens `scale-changer` existiert bereits in der Version 2.3 der RTC-lib, die sich momentan (Januar 1996) noch im Entwicklungsstadium befindet.

⁵¹Es würde den Rahmen dieser Arbeit übersteigen, wollte ich hier die genaue Funktionsweise von `choose-intervals` erläutern. Ich beschränke mich, die Eigenschaft dieses Objekts zu beschreiben (und zitiere dabei aus der Online-Dokumentation der RTC-lib): *“Chooses a supply*

“supply” wird nun mittels `sequence` der Reihe nach ausgelesen. Ist man am Ende der Liste angelangt, wird von vorne begonnen (“looping”).

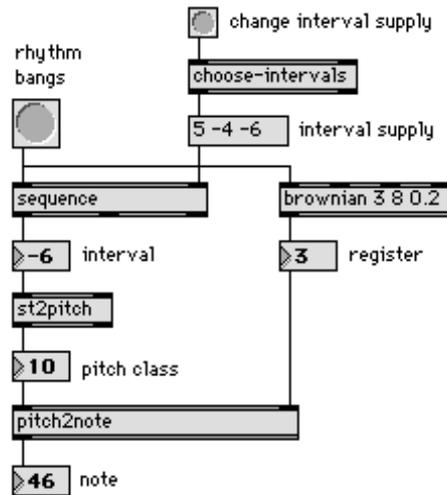


Abb. 16: intervalic-harmony

Auf diese Weise erhält man zunächst Intervalle: hier eine sich zyklisch wiederholende Folge von aufsteigend (= 5), absteigender großer Terz (= -4) und absteigendem Tritonus (= -6). Diese Intervalle werden nun in `st2pitch` sukzessive aufaddiert und mittels der Operation “modulo 12” die “pitch class” bestimmt. Zur Tonhöhe gelangt schließlich – wie in obigem Beispiel – wieder durch `pitch2note`, wo “pitch class” und “register” zusammengeführt werden.

Wir sehen in diesem Beispiel deutlich die schrittweise “Zubereitung” der harmonischen Gestalt: vom Intervall zum “pitch class” zur Tonhöhe.

3.4. Konstruktion von Strukturgeneratoren

of up to five different intervals. This set fullfills certain requirements: the sum of the chosen intervals will never be an octave, any pairs of that supply will never be an octave and there will never occur diminished chords from these intervals.”

Nachdem wir nun verschiedenen Teilbereiche der RTC-lib kennengelernt haben, möchte ich nun zeigen, wie damit Strukturgeneratoren konstruiert werden können.

Die im Folgenden wiedergegebene Graphik zeigt das Meta-Modell eines Strukturgenerators zur Generierung von Klaviermusik. Dieses (Meta-Modell) besteht aus vier Patches (mit p gekennzeichnete Objektkästchen, die die konkreten Algorithmen einer musikalischen Struktur beinhalten und nicht einer Softwarebibliothek entnommen sind).

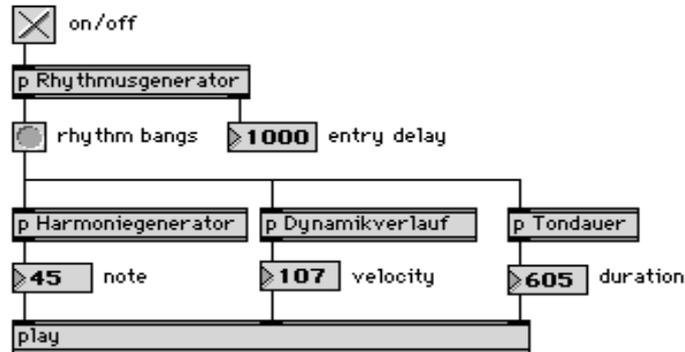


Abb. 17: Flußdiagramm eines Meta-Modells für Strukturgeneratoren zur Generierung von Klaviermusik

Ein Rhythmusgenerator erzeugt einen bestimmten Rhythmus, indem er rhythmische Impulse (“rhythm bangs”) aussendet. Diese werden an drei weitere Generatoren für Tonhöhe, Tonstärke und Tondauer geschickt und veranlassen diese Objekte, aufgrund ihres implementierten Algorithmus bei jedem Impuls einen entsprechenden Wert zu berechnen. Tonhöhe (“note”), Tonstärke (“velocity”) und Tondauer (“duration”) werden vom Objekt `play` zusammengefaßt und als MIDI-Informationen an ein angeschlossenes MIDI-Instrument gesendet und dort gespielt.

Wichtig erscheint mir die Feststellung, daß es sich bei dem obigen Beispiel um ein Meta-Modell handelt, das – bei Verwendung verschiedener Rhythmus-, Harmonie-, Hüllkurven- und Dauerngeneratoren – die Konstruktion völlig unterschiedlicher Strukturgeneratoren erlaubt. Die prinzipielle Organisation des Meta-Modells bleibt davon unangetastet: Der primäre Impulsgeber ist immer der Rhythmus, der, mit Tonhöhe, Dynamik und Dauer versehen wird und dadurch erst als musikalische Größe (“Ton”) in Erscheinung tritt. Dies sind auch genau die Parameter, die das Klavierspiel beschreiben: nämlich welche Taste (“note”) zu welchem Zeitpunkt (“entry point”) wie stark angeschlagen (“velocity”) und wie lang gehalten wird (“Dauer”).

Das klingt für's erste recht nüchtern und gibt Anlaß zur Befürchtung, daß damit nur sehr hölzerne und mechanisch klingende musikalische Abläufe zu gestalten sind. Daß dem nicht so sein muß, soll im letzten Kapitel an Hand der

Lexikon-Sonate demonstriert werden.

4. Lexikon-Sonate

Eine unendliche und interaktive Realtime-Komposition für computergesteuertes Klavier (1992 ff.)

In einem mit Hilfe der RTC-lib komponierten Werk – der *Lexikon-Sonate* (1992 ff.) – habe ich versucht, Klaviermusik seit Johann Sebastian Bach (über Beethoven, Liszt, Brahms, Schönberg, Webern, Stockhausen bis Boulez) mit ihren Topoi, Sujets und Klischees geistig zu erfassen und dieses Wissen als Strukturgeneratoren zu implementieren. Da das Werk nicht als Notentext existiert, sondern einzig und allein von einem Computerprogramm in Echtzeit generiert wird, habe ich auch Interpretationsparameter wie Rubato, Espressivo, Phrasieren etc. berücksichtigt.

4. 1. Ausgangspunkte

Die Entstehung des Werkes verdankt sich einem Roman, genauer: dem *Lexikon-Roman* (1970)⁵² von Andreas Okopenko. Dieses Buch, das die *„sentimentale Reise zum Exporteurtreffen in Druden“* (so der Untertitel) zu Lande sein kann nicht wie ein herkömmliches Druckwerk von vorne bis hinten gelesen werden. Seine viele hundert Kapitel sind wie in einem Lexikon alphabetisch sortiert. Mittels Verweis Pfeilen (‘) kann der p.t. Leser nun wieder eine Reise durch das Buch antreten, und sich nach eigenem Gutdünken durch den Text bewegen. Was heute durch Internet und Web-Browser schon Allgemeingut geworden ist, war damals noch völlig unbekannt. Ja, nicht einmal der Terminus dafür – HyperText – existierte, als Okopenko sein visionäres Werk verfaßte (er wurde einige Jahre später von Ted Nelson „erfunden“).

Die Buchform ist für einen Hypertext freilich das denkbar ungeeignetste Medium. Nur gab es 1970, als das Werk erschien, kaum Computer, geschweige denn World-Wide Web oder Hypertext-Autorensysteme. Deshalb wurde unter der Federführung von Franz Nahrada ein Gruppe von Computerfachleuten und Medienkennern zusammengelassen, die unter dem Namen *Libraries of the Mind* die elektronische Umsetzung des *Lexikon-Roman* (als CD-ROM) in Angriff nahm. Okopenko, der selbst dieser Gruppe angehört, hat nun angeregt, sein Werk multimedial zu erweitern, also Bilder, Photos, Klänge und Musik darin zu integrieren.

⁵²Andreas Okopenko, *Lexikon-Roman einer sentimental Reise zum Exporteurtreffen in Druden*, (Frankfurt/Main, Berlin, Wien 1980)

Ich gelangte 1992 zu den *Libraries* und wurde mit der Aufgabe betraut, den Musikpart zu gestalten. Nach der ersten Lektüre des Buches wurde mir allerdings rasch klar, daß die ursprünglich vorgesehene Aufgabe – kurze Musikclips für die einzelnen Kapitelchen zu komponieren – nicht zielführend ist. Die Struktur des Buches selbst forderte, so schien es mir, einen gänzlich anderen Weg. Seine potentielle Unendlichkeit und Freiheit der “Wegfindung” – seine explizite Prozessualität – stehen im krassen Widerspruch zur Komposition kleiner, abgeschlossener Einheiten. Außerdem wurde mir bald klar, daß man ein Kapitel auf die verschiedensten Arten lesen kann (andächtig meditierend, flüchtig überschlagend, vor- und zurückblättern etc.), daß – mit einem Wort – die Verweildauer des Lesers ein unkalkulierbarer Parameter blieb.

Ich wollte nun eine Musik schaffen (nicht “schreiben”), die sozusagen das Leserverhalten des “Benutzers” reflektiert. Wie lange auch immer die Verweildauer in einem Kapitel sein mag: während dieser Zeit ertönt eine charakteristische Musik, die sich jedoch verändert, wenn man zu einem anderen Abschnitt wechselt. Dieser Wechsel sollte aber nicht abrupt erfolgen, sondern Teilaspekte des Vorgegangenen weiterführen. Wie in einem Lexikon, wo der Verweis auf ein’ Stichwort immer noch einen semantischen Rückbezug bedeutet. Okopenko hat dieses “lexikalische Prinzip” im Vorwort des *Lexikon-Romans* wunderbar illustriert:

“Wer hat nicht schon im Lexikon, GOLDSCHMINKE nachschlagen wollend, erst einmal den Artikel über GOLDONI, dann den über GOLDREGEN gelesen, dort auf LABURNUM verwiesen, die Einrichtung von LABORATORIEN gestreift, Interesse an der Herstellung eines Chlorkalziumröhrchens gefaßt, die Glasblasen erlernt, dabei einen Wangenriß erlitten, pflasterbeklebt einem Clown geöhelt, nachgedacht was zum Clown noch fehlte, dabei Blanc und Rouge aufgefunden und so den Gedanken zurückgewonnen daß er ja GOLDSCHMINKE nachschlagen wollte – was er nun endgültig tat.”⁵³

So begann ich nun peu-a-peu, Topoi aus der Klavierliteratur zu analysieren und sie als Strukturgeneratoren (genannt) zu implementieren: Espressivo-Melodien, Akkordstrukturen, Triller⁵⁴, aber auch idiomatische Triolen, Arpeggi und Glissandi. Jedes dieser Module legt ein charakteristisches musikalisches Verhalten an den Tag und generiert unendlich viele Varianten seines implementierten Strukturmodells.

4.2. Module

An Hand von ESPRIT (einem Generator für Espressivo-Melodien) möchte ich den prinzipiellen Aufbau erläutern.

⁵³ebda, S. 6.

⁵⁴Das Strukturmodell eines Trillergenerators wurde im Kapitel 2.4. erläutert.

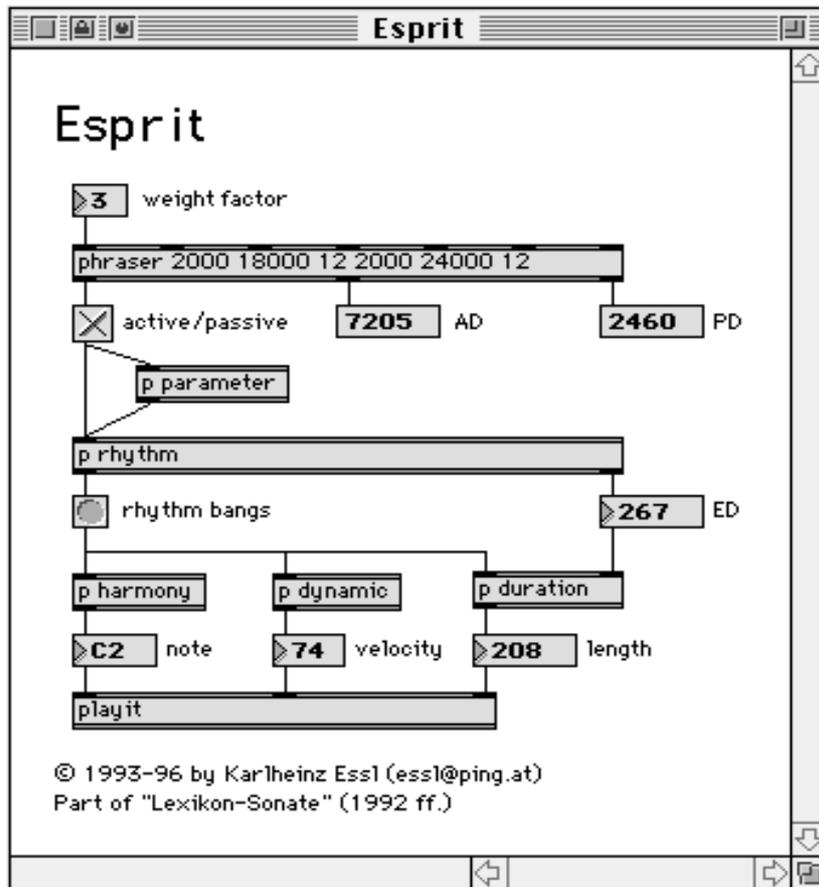


Abb. 18: Strukturgenerator ESPRIT

- **phraser:** bestimmt die Dauer einer musikalischen Phrase (aktive Phase) und der daran anschließenden Pause (passive Phase). Der "weight factor" bestimmt die statistische Länge der Aktions (AD)- und Pausendauern (PD)⁵⁵, wobei folgende Relation

⁵⁵Dieses Konzept geht auf Karlheinz Stockhausen zurück und wurde von Herbert Henck beschrieben; siehe: ders, *Karlheinz Stockhausens Klavierstück X. Ein Beitrag zum Verständnis der seriellen Kompositionstechnik* (Köln 1980), S. 19-23.

gilt:

“weight”	Aktionsdauer	Pausendauer	Ergebnis
1	kurz	lang	Hintergrund
2	mittel	mittel	Mittelgrund
3	lang	kurz	Vordergrund

Damit wird der phraser zu einem Regulator der musikalischen Dichte und – wie wir später sehen – zu einem wichtigen formalen Gestaltungsmittel. Mit anderen Worten: der “weight factor” bestimmt den Wahrnehmungsgrad eines Moduls – ob es nun deutlich im Vordergrund zu hören ist, sich bescheiden in der Mitte hält in oder gar im Hintergrund versteckt.

- **parameter:** nach jeder Pause (also vor Berechnung einer neuen musikalischen Phrase) werden die Parameter des Strukturgenerators innerhalb der vorgegebenen Grenzen zufällig geändert. Je nach Parameter unterscheiden sich die Phrasen mehr oder weniger stark voneinander. Sie erscheinen als Variationen des Strukturmodells, sind aber immer als Derivate davon erkennbar.

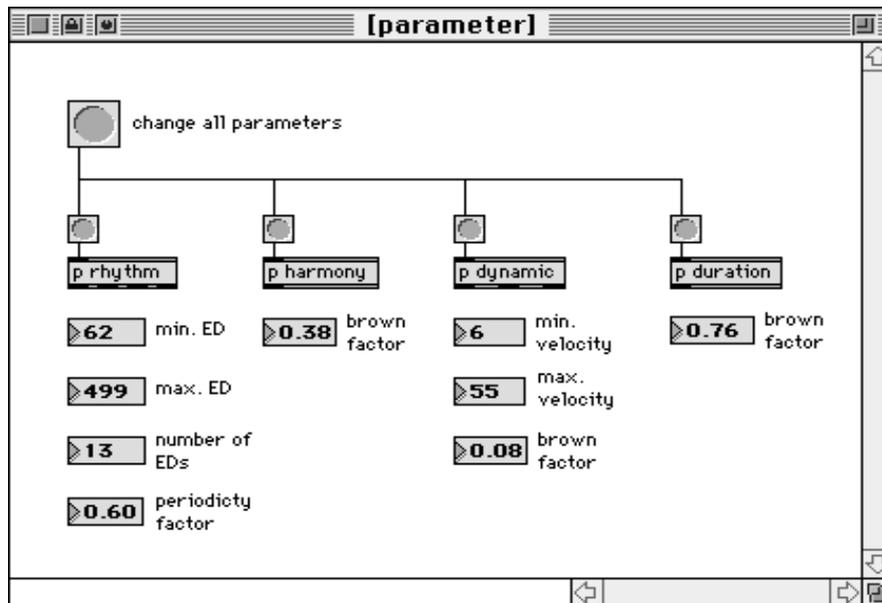


Abb. 19: Parameter-Patch von ESPRIT

- **rhythm**: erzeugt während der Dauer einer Phrase eine Folge von rhythmischen Impulsen ("rhythm bangs").

Jeder dieser Impulse markiert den Einsatzpunkt eines Tones, dessen Parameter Tonhöhe, Tonstärke und Tondauer von den folgenden Objekten berechnet werden.

- **harmony, dynamic, duration:** erzeugen aufgrund eines bestimmten Algorithmus für jeden Einsatzpunkt die entsprechenden Parameterwerte.
- **playit:** faßt diese Ton-Parameter zusammen und wandelt sie in MIDI-Code um, der vom angeschlossenen MIDI-Instrument (Player Piano, Sampler, Synthesizer) interpretiert wird.

4.3. Benutzeroberfläche

Zur Zeit besteht die *Lexikon-Sonate* aus 24 verschiedenen Modulen. Da sie jedoch als "work-in-progress" konzipiert ist, kommen laufend neue dazu.

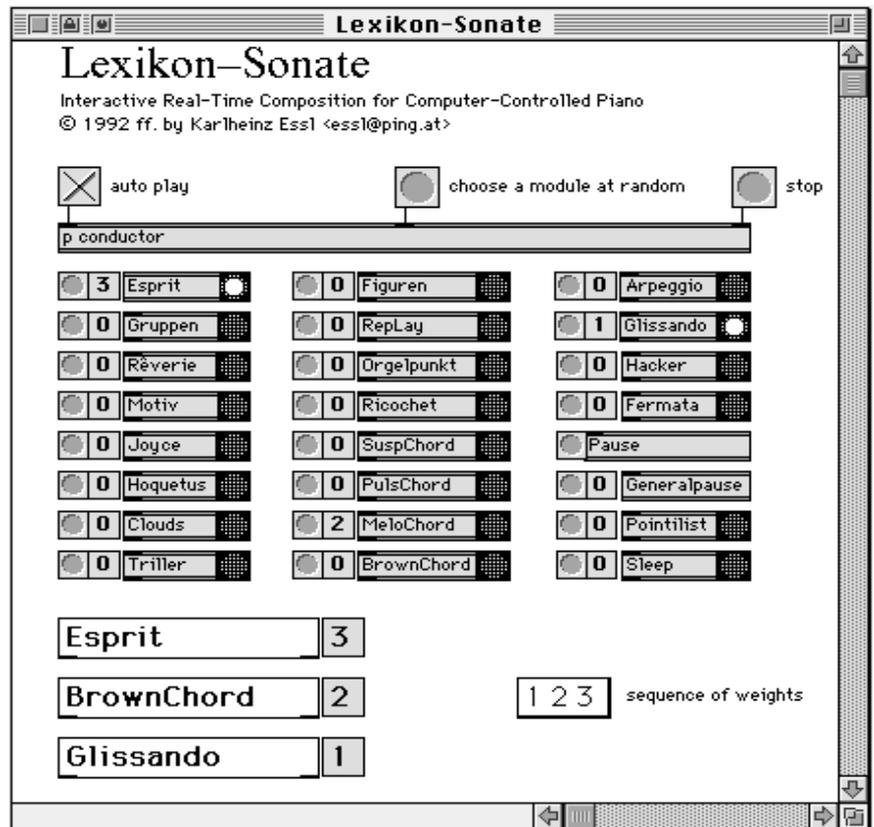


Abb. 20: Benutzeroberfläche der Lexikon-Sonate

Wir sehen in der Mitte die 24 verschiedenen Strukturgeneratoren, darunter einen sog. *conductor*, der die Auswahl der Module besorgt, und darunter eine dreistufige "Registerbank", in der die selektierten Module von oben nach unten eingetragen werden.

Jedem Kästchen der "Registerbank" ist eine Zahl zwischen 1 und 3 zugeordnet – der "weight factor". In dem obigen Beispiel werden gerade die Module **ESPRIT**, **BROWNCORD** und **GLISSANDO** miteinander kombiniert, wobei **ESPRIT** im Vordergrund (3) spielt, **BROWNCORD** im Mittelgrund (2) und **GLISSANDO** im Hintergrund (1). Das unterschiedliche Verhältnis von Aktions- und Pausendauern in den drei Modulen könnte etwa folgender Graphik entsprechen, wobei die schwarzen Kästchen die aktiven Phrasen versinnbildlichen:



Abb. 21: Aktions- und Pausendauern dreier Module

Deutlich hier die lange Phrasen von **ESPRIT**, garniert von gelegentlichen kurzen **GLISSANDI**, während sich **BROWNCORD** ausgewogen in der Mitte hält.

4.4. Aktions- und Interaktionsmöglichkeiten

Wenn nun ein neues Modul (z.B. **TRILLER**) vom *conductor* in die "Registerbank" geschickt wird, rutscht **ESPRIT** eine Stufe nach unten und erhält den "weight factor" 2. **BROWNCORD** wiederum nimmt im untersten Kästchen Platz und bekommt dort den "weight factor" 1 zugewiesen, währenddem **GLISSANDO** verschwindet.

Wir sehen also, daß bei Aufruf eines neuen Moduls das "älteste" verschwindet, zwei aber erhalten bleiben und mit dem neu dazugekommenen kombiniert werden. Dadurch wird ein gleitender formaler Übergang erzielt.

Der **conductor** selbst kann auf zwei Arten gesteuert werden: entweder durch einen **auto-piloten**, der ihn in unregelmäßigen Abständen anweist, eine Aktion zu setzen⁵⁶. Oder aber durch den Benutzer, der willentlich entscheiden kann, wann der **conductor** ein neues Modul auswählen soll⁵⁷. Im ersten Fall ist der sog. "toggle" mit der Aufschrift "auto play" einzuschalten, im letzteren wird immer dann, wann eine Änderung gewünscht wird, der Button "choose a module at random" angeklickt.

Anstelle des **conductors** kann aber auch der Benutzer die Entscheidung treffen, welche Modul "Registerbank" geschickt werden soll. Er muß dazu nur auf den Button neben dem gewünschten Strukturgen klicken.

Darüber hinaus besteht auch noch die Möglichkeit, unter Umgehung der "Registerbank" direkt bestimmte auszuwählen und ihnen beliebige "weight factors" zuzuweisen.

4.5. Rezeption

Der Mechanismus einer "Registerbank" erlaubt die Kombination von drei verschiedenen Modulen, wobei jedes bestimmten Grad von Präsenz (ausgedrückt durch den "weight factor") aufweist. Die selektierten Strukturgener laufen synchron als eigenständige Schichten ab, die jedoch nicht miteinander koordiniert sind. Sie durchdringen verschmelzen miteinander oder stoßen sich ab. Es ist schließlich der Hörer, der im Hören eine Synthese im Sinn "Radikalen Konstruktivismus"⁵⁸ leistet und aufgrund seiner individuelle Voraussetzungen den "ästhetischen Gegenstand"⁵⁹ in sich Realität werden läßt.

⁵⁶Diese Methode wurde von Gerhard Eckel gewählt, als er im November 1995 die *Lexikon-Sonate* als Klanginstallation im "Banff Centre for the Arts" (Kanada) präsentierte. - Näheres dazu findet sich im World-Wide Web unter: <http://www.ping.at/users/essl/bibliogr/eckel.html>

⁵⁷So geschehen bei der öffentlichen "Uraufführung" am 10. Februar 1994 im Großen Sendesaal des Österreichischen Rundfunks handelte sich dabei um ein Konzert mit dem "Bösendorfer SE Grand Piano", das im Rahmen der Sendung KUNSTRADIO - RADIOK ausstrahlt wurde. Das Publikum vor seinen Radioapparaten hatte die Möglichkeit, durch Wählen einer bestimmten Telefonnummer ein Computerprogramm anzuweisen, ein neues Modul in die "Registerkette" einzuspeisen. Damit konnte der Hörer unmittelbar den musikalischen Verlauf beeinflussen, und das Ergebnis seiner Intervention sofort im Radio hören. - Näheres dazu findet sich im World-Wide Web unter: <http://www.ping.at/users/essl/bibliogr/lexson-kunstradio.html>

⁵⁸Karlheinz Essl, *Kompositorische Konsequenzen des Radikalen Konstruktivismus*; in: Positionen. Beiträge zur Neuen Musik, hrsg. von Gerhard Eckel, Nr. 11 "Mind Behind" (Berlin 1992).

⁵⁹cf. Clemens Hausmann, *Kunstrezeption und ästhetischer Gegenstand*. - Vierter Teil: Anwendung und Diskussion des Modells von Roman Okopenko, *Lexikon-Roman / Karlheinz Essl, Lexikon-Sonate* (phil. Diss., Salzburg 1995), S. 113 - 120.

Die Zusammenhänge sind in der Regel komplex und vieldeutig, da sich die einzelnen Strukturgeneratoren nicht eindeutig voneinander abgrenzen. An den Rändern erscheinen sie "offen", wodurch sie mit anderen Modulen konvergieren können: Wenn zum Beispiel eine mehrtönige Trillerfiguration durch Verlangsamung als Akkordbrechung (' ARPEGGIO) oder Melodie in Erscheinung treten kann⁶⁰, oder ein Akkordgenerator (' BROWNCHORD) einstimmige Akkorde (welch' herrliche Paradoxie!) erzeugt und der Akkord so zur Melodie wird.

Ein wichtiger Aspekt der Lexikon-Sonate besteht in ihrem Vermögen, Allusionen an bereits bestehende Musik hervorzurufen. Obwohl hier keinerlei Zitate verwendet werden (sondern nur algorithmische Beschreibungen von Klaviertopoi), kommt es immer wieder zu Déjà-vu-Erlebnissen. Diese tragen zu einer Semantisierung des Gehörten bei und fordern den Hörer zu einer persönlichen Deutung und Sinngebung heraus. In dieser Weise kann das Hören zu einem aktiven Gestaltungsvorgang werden, indem der Hörer das Gehörte in sich zu Musik werden läßt – und damit zum Mitschöpfer wird.

⁶⁰Dargestellt an Hand des "Trillergenerators", s. Abschnitt 2.4.

Literatur

Eckel, Gerhard: *About the Installation of Karlheinz Essl's "Lexikon-Sonate" at the Banff Centre for the Arts, Canada 1995*; in:

<http://www.ping.at/users/essl/bibliogr/lexson-eckel.html>

Essl, Karlheinz: *Aspekte des Seriellen bei Stockhausen. Relevante Ordnungsstrukturen gegen Vergleichsgültigkeit*; Almanach "Wien modern '89", hrsg. von Lothar Knessl (Wien 1989).

- *Zufall und Notwendigkeit. Gottfried Michael Koenigs "Streichquartett 1959" vor dem Hintergrund kompositionstheoretischen Überlegungen*; in: Musik-Konzepte, Bd. 66, "Gottfried Michael Koenig", hrsg. von Klaus Metzger und Rainer Riehn (München 1989).

- *Computer Aided Composition*; in: Distel, Nr. 46/47 "Mensch Maschine" (Bozen 1991).

- *Kompositorische Konsequenzen des Radikalen Konstruktivismus*; in: Positionen. Beiträge zur neuen Musik, Hrsg. von Gisela Nauck, Heft 11 "Mind Behind" (Berlin 1992).

- *Lexikon-Sonate. Darmstadt-Lecture 1994*; in: ton-gemisch. darmstadt-lectures, hrsg. von Lothar Knessl (Darmstadt 1994/95).

- *Lexikon-Sonate. An Interactive Realtime Composition for Computer-Controlled Piano*; in: Proceedings of the Brazilian Symposium on Computer Music" (Canela 1995).

- *Wie entsteht ein Komposition? Grenzüberschreitung als kompositorisches Prinzip*; in: Bericht über das B. Symposium 1995 "Zum Schaffensprozeß in den Künsten" (Linz 1997).

- *Plädoyer für "Das Offene Kunstwerk". Zur Frage: "Für und wider das Kunstwerk"*; in: Positionen. Beiträge zur neuen Musik, hrsg. von Gisela Nauck, Heft 26 "Interpretation" (Berlin 1996).

Günther, Bernhard: *Irreal-Enzyklopädie – einer metaphorischen Reise – zur Lexikon-Sonate von Karlheinz Essl* (msch. 1995).

Hausmann, Clemens: *Kunstrezeption und ästhetischer Gegenstand. - Vierter Teil: Anwendung und Diskussion des Modells: Andreas Okopenko, Lexikon-Roman / Karlheinz Essl, Lexikon-Sonate* (phil. Diss., Salzburg 1995), S. 113 - 120.

Koenig, Gottfried Michael: *Ästhetische Praxis. Texte zur Musik*, hrsg. von Stefan Fricke und Wolf Frobenius (= Quellentexte zur Musik des 20. Jahrhunderts, 1), 3 Bd. (Saarbrücken 1991–93).

Lachenmann, Helmut: *Bedingungen des Materials. Stichworte zur Theoriebildung*; in: *Darmstädter Beiträge zur Neuen Musik*, Bd. XVII (Mainz 1978).

Andreas Okopenko, *Lexikon – einer sentimental Reise zum Exporteurtreffen nach Druden – Roman* (Salzburg 19701, Frankfurt-Berlin-Wien 19832).

Stockhausen, Karlheinz: *Texte zur elektronischen und instrumentalen Musik*, Bd. 1, hrsg. von Dieter Schnebel (Köln 1963).

Anhang

Die *Lexikon-Sonate* und die *Real Time Compositions Library* (momentan in der Version 2.2 verfügbar) sind Public Domain Software und werden mit MAX 3.0 (IRCAM / OP CODE) vertrieben. Darüber hinaus können sie im Internet von folgenden ftp-servern bezogen werden:

1. Software

- `ftp://ftp.ircam.fr/pub/IRCAM/programs/max-patches/compositions/RTC-lib_2.2.sea.bin`
- `ftp://kahless.isca.uiowa.edu/pub/max/RTC-lib_2.2.sea.hqx`

Eine Runtime-Version der *Lexikon-Sonate* (die ohne MAX läuft) ist auf der Neoism-Homepage unter:

- `http://www.neoism.org/neoism/squares/lexicon.html`

erhältlich.

2. Disklavier-Disk

In Robert Willey's Disklavier-Archive befindet sich eine Realisation der *Lexikon-Sonate* als sog. Disklavier-Disk, die auf einem "Yamaha Disklavier" abgespielt werden kann:

- `http://crca-www.ucsd.edu/bobw/disk3.html`

3. MIDI-Files

Verschiedene, mit Hilfe der *Lexikon-Sonate* generierte MIDI-Files können von:

◦ <http://www.ping.at/users/essl/div/LexSon-MIDI.sit.hqx>

heruntergeladen werden.

4. Audio

Ein Ausschnitt von der Uraufführung der *Lexikon-Sonate* am 10.2.1994 (mit dem Bösendorfer SE Grand Piano) findet sich auf der CD

Karlheinz Essl: Rudiments (1995)

die von der TONOS Musikverlags GmbH (Ahastr. 9, D-64285 Darmstadt) bezogen werden kann. Eine weitere Live-Aufnahme (diesmal mit einem Yamaha Disklavier) wird in Kürze auf der von Bruno Degazio herausgegebenen SoundPrint-CD "The Devil's Staircase II: Random Sense" (Toronto 1996) erhältlich sein.

* * *

NB: Die hier zitierten Texte von Karlheinz Essl sowie weitere Informationen zur *Lexikon-Sonate* und zur "Real Time Composition Library" können im World-Wide Web unter der URL:

<http://www.ping.at/users/essl/index.html>

abgerufen werden.