

Agnes Seipelt (Detmold / Paderborn) / Paul Gulewycz (Wien) /
Robert Klugseder (Wien)

Digitale Musikanalyse mit den Techniken der Music Encoding Initiative (MEI) am Beispiel von Kompositionsstudien Anton Bruckners

Die Analyse musikalischer Werke als eines der Hauptgebiete der Musikwissenschaft trägt maßgeblich zum Verständnis dieser bei und hilft, „Sinnstiftendes bei unmittelbaren Klangfolgen oder unter größerem Blickwinkel zur Sprache zu bringen“.¹ Vor allem mit der harmonischen Analyse können (zumindest teilweise) musiktheoretische und harmonische „Regeln“ („Ein reiner Dur-Akkord in Grundstellung besteht aus mindestens drei verschiedenen Tonhöhen in Terzabständen. Die Töne dazu sind Grundton, große Terz über dem Grundton und reine Quinte über dem Grundton.“) nachvollzogen und mit verschiedenartigen Vokabeln und Zeichen um- und beschrieben werden. Allerdings, so Büsing, ist eine Analyse „keinesfalls eine Tatsachenfeststellung, sondern eine Interpretation, bei der Varianten und Alternativen oder auch Widersprüche einzukalkulieren sind“.²

Trotzdem gibt es schon seit den 1960er Jahren Ansätze, diese mehr oder weniger komplexen „Regeln“ (zumindest die, die sich festschreiben lassen) mehr oder weniger komplex formalisiert darzustellen und mit diesen extrem formalisierten Aussagen einen Computer die harmonischen „Regeln“ erkennen zu lassen.³ Harmonik kann bzw. wird mit diesen Systemen also in Algorithmen beschrieben.⁴

Als Basis für diese Algorithmen dienen Noten, die in codierter Form, d. h. maschinenles- und prozessierbar, vorliegen. Diese Formate können beispielsweise auf ASCII, Midi oder XML basieren. Zu letzterem gehört auch die Auszeichnungssprache MEI (Music Encoding Initiative)⁵. MEI hat sich inzwischen als Standard für Digitale Musikeditionen in der Musikwissenschaft durchgesetzt, womit ein wachsender Fundus an tief erschlossenen Daten in diesem Format zur Verfügung steht. Mit MEI können notentextliche Phänomene aufgenommen werden, die für die musikwissenschaftliche Forschung und Edition von großem Interesse sind, beispielsweise die Codierung mehrerer Quellen und ihrer Varianten oder skripturale Phänomene wie Streichungen und Ersetzungen von Notenmaterial etc. Bisher gibt es kein Programm oder System, das auf der Basis von MEI eine harmonische Analyse ausführt und die Ergebnisse in die MEI-Codierung an die jeweiligen Stellen zurückführt, obwohl in MEI sogar Elemente zur harmonischen Beschreibung vorhanden sind. Aus diesem Grund sollen im Folgenden Überlegungen für eine entsprechende Analyse angestellt

1 Otfried Büsing, *Harmonik als Netzwerk*, Hildesheim u. a. 2012, S.1.

2 Ebd.

3 Eine Übersicht dazu findet man bei Néstor Nápoles López, *Automatic harmonic analysis of classical string quartets from symbolic score* (Masters Thesis, Universitat Pompeu Fabra, Barcelona, 2017), online abrufbar unter: <<https://tinylink.net/2hTCC>>

4 Dass diese Auffassung viele Schwierigkeiten birgt und wo und wie die Grenzen der Formalisierbarkeit von harmonischen Gegebenheiten, die eben teilweise interpretiert sind, liegen, sind wichtige Fragen, die aber für diese Überlegungen zunächst ausgeklammert bleiben.

5 Homepage der Music Encoding Initiative: <www.music-encoding.org>, 27.6.2018.

werden. Dem voraus geht eine Vorstellung der für die Analyse herangezogenen Quelle, eines in seiner Lehrzeit angelegten Studienbuchs von Anton Bruckner, und ihrer historisch-musikwissenschaftlichen Einordnung sowie dem daraus entstandenen Codierungsmodell bzw. der Digitalen Edition.⁶

Bruckners Lehrzeit 1861–1863

Nach einem strengen Studium der Musiktheorie bei Simon Sechter nahm Anton Bruckner Unterricht beim Linzer Theaterkapellmeister Otto Kitzler. Kitzler unterrichtet ihn in der zeitgemäßen Art der Instrumentation, im Form- und Periodenbau und zu gattungsspezifischen Fragestellungen. Den Kompositionslehren des 19. Jahrhunderts folgend, werden zunächst einfache Kadenzübungen mit verschiedenen Schlüssen, Modulationsübungen, Übungen zum Periodenaufbau mit kontinuierlich steigendem Schwierigkeitsgrad und dreiteilige Liedkompositionen und daran anschließend Walzer, Polkas und Etüden komponiert. Erst im Anschluss widmet sich Bruckner komplexeren Gattungen wie der Klaviersonate, dem Streichquartett und der Symphonie. Besonders interessant sind in diesem Zusammenhang seine Instrumentationsübungen zu Beethovens *Pathétique*.

All diese Übungen sind im sogenannten Kitzler-Studienbuch erhalten, das 2013 von der Österreichischen Nationalbibliothek erworben wurde und nun der Öffentlichkeit zugänglich ist.⁷ Das Manuskript umfasst 326 Seiten und enthält zusätzlich zu den Übungen und deren Varianten und Korrekturen auch Kommentare, die den Austausch zwischen Bruckner und seinem Lehrer dokumentieren. Dadurch wird ein wertvoller Einblick in den Musikunterricht des 19. Jahrhunderts und die Arbeitsweise Bruckners ermöglicht. Ausführliche Untersuchungen zum Aufbau und Inhalt des Studienbuchs und der Arbeitsweise Bruckners finden sich in der Forschung von Paul Hawkshaw.⁸

6 Dieser Aufsatz basiert auf den Zwischenergebnissen des zweijährigen Projekts „Digitale Musikanalyse mit den Techniken der Music Encoding Initiative (MEI) am Beispiel von Kompositionsstudien Anton Bruckners“ der Österreichischen Akademie der Wissenschaften in Kooperation mit der Universität Paderborn und dem Zentrum Musik – Edition – Medien (ZenMEM) sowie dem ACDH (Austrian Center for Digital Humanities), MEI und Verovio unter der Leitung von Robert Klugseder. Mitarbeiter sind Paul Gulewycz (ÖAW) und Agnes Seipelt (Universität Paderborn) sowie Peter Provaznik vom ACDH, der für die Programmierungen der Digitalen Edition und des Analyse-Tools zuständig ist und Johannes Kepper (Projekt „Beethovens Werkstatt“), der bei der Konzeption und Entwicklung der harmonischen Analyse mit MEI berät. Gefördert wird es durch das Programm GO!DIGITAL 2.0 der ÖAW.

7 Paul Hawkshaw, *Das Kitzler-Studienbuch. Anton Bruckners Studien in Harmonie- und Instrumentationslehre bei Otto Kitzler (1861–63)*, Faksimile-Ausgabe, Wien 2014.

8 Paul Hawkshaw, *The manuscript sources for Anton Bruckner's Linz works: A study of his working methods from 1856 to 1868*, Columbia 1984; ders., „Das Kitzler-Studienbuch. Ein unschätzbare Dokument zu Bruckners Arbeitsweise“, in: *Bruckner Symposium. Zum Schaffensprozeß in den Künsten im Rahmen des internationalen Brucknerfestes Linz 1995*, Bericht, Linz 1997, S. 95–111.; ders., „Lied als Lehrmittel: Die Lieder in den Formenlehren Anton Bruckners während seiner Studienzeit bei Otto Kitzler 1861–1863“, in: *Bruckner-Tagung Bruckner – vokal, 2003, Sonderthema: Musikgeschichte Steyrs*, Wien 2009, S. 179–197.

Digitale Edition

Die Digitale Edition des Studienbuchs basiert auf der bereits genannten Auszeichnungssprache MEI. Mit Plugins und/oder XSLTs⁹ können die mit Notensatzprogrammen erstellten Notensätze mit teilweise erstaunlich guter Qualität nach MEI konvertiert werden. Die zahlreichen Eingriffe in den handschriftlichen Notentext wie beispielsweise Streichungen, Ausradierungen, Ersetzungen oder Varianten müssen dennoch von Hand nachträglich codiert werden. Dafür werden die jeweiligen Notentextstellen mit dafür vorgesehenen Elementen ausgezeichnet wie z.B. eine Substitution, die eine Hinzufügung (<add/>) und eine Löschung () enthält. Durch die geschachtelte Baumstruktur in MEI können solche Phänomene und ihre Abhängigkeiten voneinander beschrieben werden:

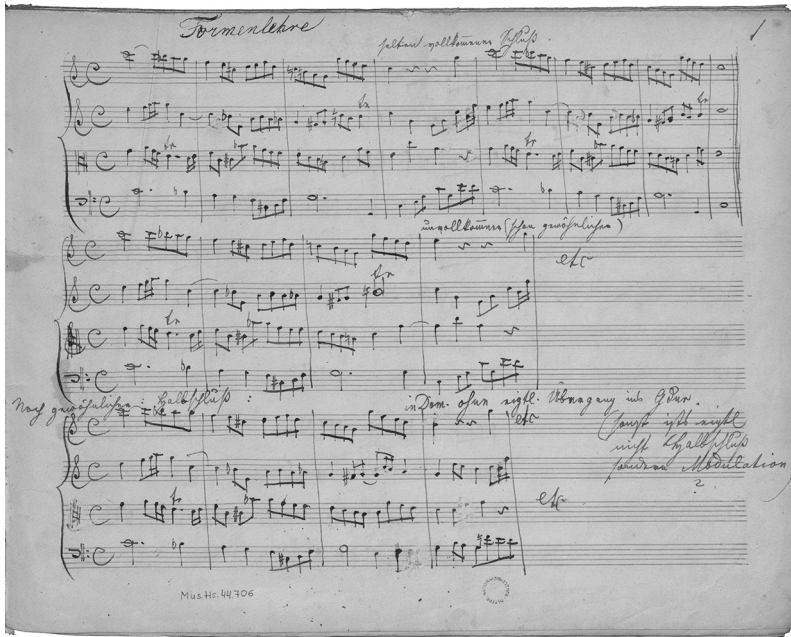


Abbildung 1: Kitzler-Studienbuch, S. 1

```
<subst>
  <del rend="erased">
    <note xml:id="kitzler-001_95" dur="8" oct="5" pname="c"/>
    <note xml:id="kitzler-001_96" dur="8" oct="4" pname="a"/>
  </del>
  <add>
    <note xml:id="kitzler-001_m-95" dur="4" oct="5" pname="c"/>
  </add>
</subst>
```

Abbildung 2: MEI-Code für die Substitution in Takt 2 in der zweiten Notenzeile

9 =Extensible Stylesheet Language Transformation, mit der man XML-Dateien transformieren, prozessieren oder Operationen damit durchführen kann.

Ebenfalls werden die zahlreichen Annotationen im Notentext händisch aufgenommen. Sie kommentieren musikalische Phänomene, stellen Fragen, kritisieren, oder waren einfach die Aufgabenstellung. Mit MEI kann außerdem zwischen verschiedenen Schreibern oder Schreibmitteln unterschieden werden. Für dieses Vorhaben wird zwar aller lesbarer Notentext aufgenommen, aber weitergehende Überlegungen zur Genese, also die Codierung einer eventuellen Reihenfolge der Veränderungen, bewusst ausgelassen. Auch genauere Beschreibungen bzw. Codierungen der Eingriffe können aus Zeitgründen nicht berücksichtigt werden. Die konkrete skripturale Ausführung der Tilgung, wie sie etwa im Projekt „Beethovens Werkstatt“¹⁰ erfasst wird, also ob z. B. eine Streichung horizontal oder vertikal, einfach oder mehrfach stattfand und wann sie getätigt wurde, wird hier vernachlässigt. Die Codierung soll daher in erster Linie den letztgültigen Text des Manuskripts wiedergeben, in dem alle erkennbaren vorhergehenden Eingriffe summarisch, d. h. ohne weitere zeitliche Differenzierung aufgenommen werden.¹¹

Für die Modellierung oder, im traditionellen Sinne, die Editionsrichtlinien, wird zunächst das Standard MEI-CMN-Schema¹² benutzt, das aber an die Bedürfnisse des Projekts angepasst wird. Das bedeutet, es wird entsprechend enger gestrikt, aber auch, falls nötig, mit projektspezifischen Elementen und Attributen angereichert oder verändert. Somit soll eine mit zeitgenössischen, digitalen Mitteln aufbereitete, codierte, durchsuchbare und vor allem maschinenlesbare und -prozessierbare Quellenedition erstellt werden.

Aus der Natur eines Studienbuchs heraus gibt es nicht immer eine strikt chronologische und logische Abfolge der Stücke. Vieles im Studienbuch ist fragmentarisch, unvollständig und skizzenhaft; die *Symphonie in f-Moll* zum Beispiel liegt im Großteil nur in einzelnen, voneinander unabhängigen Motiveinfällen vor, nur das Scherzo ist vollständig im Studienbuch vorhanden. Ein automatischer Durchlauf der Analyse würde in den meisten Fällen kein brauchbares Ergebnis liefern. Aus diesem Grund hat die Edition zwei Modi, die Werk- und die Quellenansicht, zwischen denen die Benutzerinnen und Benutzer wechseln können. Die Analyse-Anwendung (s. u.) wird in der ersten Ansicht implementiert.

Die Werkansicht bietet die Möglichkeit, die Werke nach Werkverzeichnis-Nummern geordnet und in der Edition¹³ aufrufen zu können. Einige Kommentare und Annotationen Bruckners werden ebenfalls in die Transkription aufgenommen. Auf Seite 4 (Abb. 3) kann man am Beginn der Übungsabschnitte „Plagalschluß“ oder „Trugschluß“ lesen. Vereinzelt sind zudem Verweise auf Modulation wie „nach Bdur“ und „ins Amoll“ eingetragen. All das soll zum besseren Verständnis der Übungen zusätzlich zur Musikedition dargestellt werden. Über einen Link kann direkt in die synoptische Quellenansicht gewechselt werden, in der die Seiten, unabhängig vom logischen Inhalt, in der Reihenfolge des Original-Manuskripts mit dem Faksimile angezeigt werden. Längere Kommentare sind oft zu umfangreich, um sie

10 Homepage des Projekts „Beethovens Werkstatt: Genetische Textkritik und Digitale Musikedition“: <<https://beethovens-werkstatt.de>>, 27.6.2018.

11 Damit wird eine spätere genetische Aufarbeitung möglich gemacht, andererseits können möglicherweise auch im Rahmen der Analyse eventuelle Zusammenhänge zwischen Korrekturen und harmonisch „interessanten“ Wendungen erkannt werden.

12 = Common Music Notation. Ein MEI-Schema legt fest, welche Elemente und Attribute an welcher Stelle für die jeweilige Datei erlaubt sind. Das CMN-Schema ist für die „klassische“ westliche Musik gedacht und offengehalten. Daneben gibt es Schemata für Neumen und Mensuralnotation, s. <<https://github.com/music-encoding/music-encoding/tree/develop/schemata>>, 18.7.2018.

13 MEI-Codierungen können mit der Anzeigebibliothek Verovio gerendert und in klassischer Notation angezeigt werden, s. <www.verovio.org>, 27.6.2018.

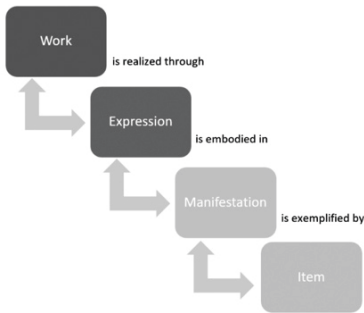


Abbildung 4: FRBR-Beziehungen

```

<workDesc>
  <!-- Connection to the Digital Catalogue of Works -->
  <work label="WAB99">
    <expressionList>
      <expression>
        <!-- Expression in the Kitzler-Studienbuch -->
        <relationList>
          <relation rel="hasEmbodiment" target="../kitzler_119.mei"/>
        </relationList>
      </expression>
    </expressionList>
    <identifier authority="WAB" authURI="http://bruckner-online.at/?page_id=959&Signatur=A-KR-C56-7"/>
  </work>

```

Abbildung 5: FRBR-Beziehungen im Kopfbereich einer MEI-Datei

Harmonische Analyse

Mit dem codierten Studienbuch als Datengrundlage sollen die Grenzen einer automatisierten, musikwissenschaftlichen Ansprüchen genügenden harmonischen Analyse ausgelotet werden. Das Ziel ist, eine Brücke zwischen editorischer Arbeit und späterer musikwissenschaftlicher Auswertung bzw. Forschung zu schlagen, um digitale Editionen in Zukunft nicht nur dynamisch anzeigen zu können, sondern diese auch auswerten zu lassen. Dabei ist klar, dass eine solche automatisierte Analyse an Grenzen stoßen wird. Spannend ist vor allem die Frage, wo diese Grenzen liegen und wie sie sich letztlich der Benutzerin und dem Benutzer vermitteln lassen, um diese zur eigenständigen Untersuchung der nicht eindeutigen harmonischen Phänomene anzuregen. Durch die mit MEI gegebene Möglichkeit, Analyse-Ergebnisse wiederum in die Codierungen zurückzuführen, können so die nicht erkannten Daten manuell nachgetragen werden.

Wie in der Einleitung bereits angedeutet, gibt es auf dem Gebiet der computergestützten Analyse bisher schon weitgehende Forschung und diverse unterschiedliche Ansätze, Musik mit dem Computer zu analysieren. Der Vollständigkeit halber soll erwähnt werden, dass die Pionierarbeit auf dem Gebiet der computerbasierten Analyse von Musik von Terry Winograd

aus dem Jahr 1968¹⁶ stammt, gefolgt von einem System von H. J. Maxwell aus dem Jahr 1992,¹⁷ die beide eine Stufenanalyse erzeugen.

Das in den 1980er Jahren von David Huron entwickelte Codierungsformat Humdrum¹⁸ bietet mit den von Craig Sapp implementierten Analyse-Funktionen der Humdrum extras¹⁹ ein Bündel an verschiedenen Analyse-Abfragen und Algorithmen. In der Kategorie „harmony“ findet man beispielsweise den Befehl „tsroot“, mit dem der von David Temperley 1997 entwickelte „algorithm for harmonic analysis“²⁰ aufgerufen wird, der 2001 als Melisma Music Analyzer veröffentlicht wurde, und dem User eine mit harmonischen Informationen angereicherte Humdrum-Datei ausgibt (s. Abb. 6).²¹ Temperleys Ziel ist, mit diesem Algorithmus den Prozess des „menschlichen“ Analysierens so gut wie möglich zu modellieren.²² Dafür teilt er die Analyse zunächst in Tonartbestimmung („key finding“) und Grundtonbestimmung („root finding“) auf, bei der das Stück in Segmente unterteilt wird.²³ Letztere verwendet Regeln, sog. „preference rules“, die u. a. die Bestimmung von Verzierungsnoten festlegen („Ornamental Dissonance Rule“²⁴), die Unterteilung der Segmente regelt („Strong-Beat Rule“²⁵) oder die Bestimmung von Akkordgrundtönen festlegt, wenn dies nicht eindeutig ist („Harmonic Variance Rule“²⁶). Dabei ist zu beachten, dass der von Temperley festgelegte Input Musik in Audioform, also MIDI, ist, und es deshalb z. B. keine enharmonische Unterscheidung gibt.

16 Terry Winograd, „Linguistics and the computer analysis of tonal harmony“, in: *Journal of Music Theory* 12 (1968), S. 2–49.

17 H. J. Maxwell, „An expert system for harmonic analysis of tonal music“, in: *Understanding music with AI*, Cambridge 1992, S. 335–354.

18 Vgl. <<http://www.humdrum.org>>, 30.7.2018.

19 Humdrum extras, <<http://extras.humdrum.org>>, 30.7.2018.

20 David Temperley, „An algorithm for harmonic analysis“, in: *Music Perception: An Interdisciplinary Journal* 15 (1997), S. 31–68.

21 Humdrum extras. tsroot manpage: <<http://extras.humdrum.org/man/tsroot/>>, 28.6.2018.

22 Vgl. Temperley, „An algorithm for harmonic analysis“, S. 33.

23 Vgl. Ebd., S. 34.

24 Ebd., S. 54.

25 Ebd., S. 52.

26 Ebd., S. 53, die „line of fifths“ führt Temperley in seinem Paper ein, es handelt sich um den Quintenzirkel, der als Linie in beiden Richtungen unendlich ist, vgl. ebd., S. 42.

```

!!HOTL: Prelude No. 12 in F minor
**tsrhm
*
*>[A,A,B,B]
*>norep[A,B]
*>A
*lcemba
*
*k[b-e;-a-d-]
*
*M2/4
*MM120
*
.
.
=1
.
V7
.
.
=2
i
.
.
=3
iib
.
V7b/V
.
=4
V
.
.
.
.
=5
ic

```

Abbildung 6: Humdrum-Codierung mit analytischen Informationen, Anfang des Präludiums in f-Moll aus dem *Wohltemperierten Klavier II* von J.S. Bach

Neben dem Melisma Music Analyzer bieten die Humdrum extras noch weitere Programme an. Mit den Befehlen „keycor“ und „mkeyscape“ wird z.B. ein Algorithmus aufgerufen, der die Tonarten bestimmt und graphisch anzeigt. Dieser Algorithmus wird unten näher erläutert.

Im Gegensatz zu den Codierungsformaten Humdrum, Lilypond²⁷, MusicXML²⁸ oder MEI ist das Python-basierte und am MIT entwickelte music21²⁹ auf musikalische Analysen und Abfragen diverser Art spezialisiert. Anstatt als Codierungssprache zu fungieren, stellt music21 einen „Werkzeugkasten“ („toolkit“) bereit, mit dem die von MIDI, Lilypond, MusicXML und auch MEI importierten Daten analysiert, verarbeitet und wiederum in das jeweilige Ausgabeformat exportiert werden können. Der Export von mit music21 verarbeiteten Daten nach MEI ist allerdings noch nicht möglich. Music21 wirbt damit, „simple to use but [...] also extremely powerful“³⁰ zu sein und mit wenigen Zeilen Code diese diversen Operationen und Abfragen zu stellen. In der Tat ist music21 zurzeit eins der mächtigsten Werkzeuge; zudem ist die Programmiersprache Python, mit der die Programme

27 Lilypond. Notensatz für Jedermann, <<http://lilypond.org/index.de.html>>, 30.7.2018.
 28 MusicXML, 2004 von Michael Good entwickelt, ist ein Vorgänger von MEI und basiert ebenfalls auf XML. Es ist vor allem als Austauschformat für unterschiedliche Codierungsformate in Gebrauch, vgl. <<https://www.musicxml.com>>, 30.7.2018.
 29 Homepage von music21, s. <<http://web.mit.edu/music21/>>, 28.6.2018.
 30 What is music21? Abschnitt „Learning music21“: <<http://web.mit.edu/music21/doc/about/what.html>>, 28.6.2018.

von music21 aufgerufen und ausgeführt werden, deutlich einfacher und simpler gehalten als bei anderen Programmen, und die Dokumentation ist sehr ausführlich und gut verständlich.³¹ Im Folgenden sind einige Zeilen Code zu sehen, mit denen eine Humdrum-Datei importiert und mit Stufen versehen wird, wobei die Tonart für das Stück für diese Analyse festgelegt werden muss:³²

```
# import humdrum file
bach = m21.converter.parse('bachfminor.krn')
# create chords by combining all voices
bach_chords = bach.chordify()
# define key f-minor
key = m21.key.Key('f')
for chord in bach_chords.recurse().getElementsByClass('Chord'):
# convert each chord to closed position
    chord.closedPosition(forceOctave=4, inPlace=True)
# obtain roman numeral in the context of the key
roman_numeral = m21.roman.romanNumeralFromChord(chord, key)
    # attach result as lyric
    chord.addLyric(str(roman_numeral.figure))
# add chord reduction with numerals to original score
bach.insert(bach_chords)
# export everything as musicxml for further processing
bach.write(fmt='musicxml', fp='bachfminor.xml')
```

Ein Nachteil bei einem MEI-Import ist, dass die analytischen Informationen nicht in die MEI-Datei zurückgeschrieben werden können. Als Export würde sich wieder MusicXML anbieten, das dann in einem Notensatzprogramm wie Sibelius graphisch dargestellt werden kann oder zurück nach MEI konvertiert werden müsste. Eine Analyse mit MEI als Input ist also nur über Umwege möglich und über diese Umwege bzw. Konvertierungen geht oftmals viel von den Inhalten verloren, wird verändert oder es kommen unnötige Informationen hinzu.

All diesen Systemen ist zudem gemeinsam, dass sie Benutzerinnen und Benutzer immer vor die Hürde der Einarbeitung in die jeweiligen Programmiersprachen und in die Arbeit mit der Kommandozeile stellt. Der Ansatz ist deshalb, einen Algorithmus, der zwar schon 1990 entwickelt wurde, aber in den Humdrum extras und auch bei music21 Verwendung findet,³³ für eine „kommandozeilenlose“ Nutzung mit MEI-Daten umzuarbeiten.

Dieser sogenannte key-finding-Algorithmus von Carol Krumhansl und Mark Schmuckler (im Folgenden KS-Algorithmus)³⁴ basiert auf der einfachen Idee des Auszählens von Tonhöhen und einem Abgleich mit Vergleichswerten. Dabei wird gezählt, wie häufig jede Tonhöhe (unabhängig von Oktave und enharmonischer Verwechslung) in einem Stück vorkommt. Dadurch entstehen sogenannte Histogramme, eine Reihe von Zahlenwerten. Diese Histogramme werden mit der sogenannten „Pearson-Korrelation“ mit Vergleichswerten für

31 music21 documentation: <<http://web.mit.edu/music21/doc/>>, 28.6.2018.

32 Die Zeilen mit einer Raute (#) kommentieren die nachfolgenden Code-Zeile.

33 Modulbeschreibung „music21.analysis.discrete“, <<http://web.mit.edu/music21/doc/moduleReference/moduleAnalysisDiscrete.html>>, 31.7.2018.

34 Carol L. Krumhansl, *Cognitive Foundations of Musical Pitch*, Oxford 1990, und Humdrum extras, keycor manpage, <<http://extras.humdrum.org/man/keycor/>>, 28.6.2018.

jede Tonart in Beziehung gesetzt, womit die Ähnlichkeit der beiden Histogramme errechnet werden kann:³⁵

$$R(x, y) = \frac{\sum(x_n - \bar{x})(y_n - \bar{y})}{\sqrt{\sum \bar{x}(x_n - \bar{x})^2(y_n - \bar{y})^2}}$$

Die Vergleichswerte entstammen kognitiven Experimenten, bei denen die Probandinnen und Probanden u. a. die Zugehörigkeit von Tönen zu Tonarten in verschiedenen musikalischen Kontexten bewerten sollten.³⁶ Das Histogramm der Probe wird mit einem Histogramm jeder Tonart verglichen, wobei die Vergleichswerte für C-Dur (im simple-Profil mit den Werten 2,0,1,0,1,1,0,2,0,1,0,1) für Cis-Dur als Grundton um eine Position nach rechts verschoben werden, für D-Dur zwei Positionen nach rechts etc.

Der Originalvergleichswert des Algorithmus ist das Krumhansl-Kessler-Profil. Im Laufe der Zeit haben andere Wissenschaftlerinnen und Wissenschaftler weitere Profile aus unterschiedlichen Musikcorpora abgeleitet, die verschiedene Ausprägungen in ihrer Aussagekraft besitzen.³⁷ Beispielsweise tendiert das originale Krumhansl-Kessler-Profil dazu, die Dominante als Tonika zu erkennen, das Aarden-Esser-Profil neigt leicht dazu, die Subdominante als Tonika zu erkennen. Das simple-Profil hingegen hat keine besondere Ausrichtung, wird bei kleinen zu untersuchenden Regionen aber sehr ungenau (s. Abb. 7).

**kern	**number	**kern	**number	**kern	**number
C	6.35	C	17.7861	C	2
C#	2.23	C#	0.145624	C#	0
D	3.48	D	14.9265	D	1
D#	2.33	D#	0.160186	D#	0
E	4.38	E	19.8049	E	1
F	4.09	F	11.3587	F	1
F#	2.52	F#	0.291248	F#	0
G	5.19	G	22.052	G	2
G#	2.39	G#	0.145624	G#	0
A	3.66	A	8.15494	A	1
A#	2.29	A#	0.232998	A#	0
B	2.88	B	4.95122	B	1
c	6.33	c	18.2645	c	2
c#	2.68	c#	0.737619	c#	0
d	3.52	d	14.0499	d	1
d#	5.38	d#	16.8599	d#	1
e	2.60	e	0.702494	e	0
f	3.53	f	14.4362	f	1
f#	2.54	f#	0.702494	f#	0
g	4.75	g	18.6161	g	2
g#	3.98	g#	4.56621	g#	1
a	2.69	a	1.93186	a	0
a#	3.34	a#	7.37619	a#	0.5
b	3.17	b	1.75623	b	0.5
*.	*.	*.	*.	*.	*.

Abbildung 7: Profile v.l.n.r.: Krumhansl-Kessler, Aarden-Essen, simple, in: <<http://extras.humdrum.org/man/keycor/>>, 28.6.2018

Die Tonart des aus der Berechnung resultierenden höchsten Korrelationswertes R wird als die „richtige“ Tonart des Ausschnitts gewertet. Damit kann die Tonart eines Stücks oder eines Ausschnitts mit hoher Wahrscheinlichkeit korrekt bestimmt werden. Am Beispiel von dem Lied „Von der schlummernden Mutter“ (s. Abb. 8) soll dies gezeigt werden: In dem Notenbeispiel befindet sich, in Achteln gezählt, zwölfmal der Ton c, unabhängig von Oktave und enharmonischer Verwechslung. Anschließend werden alle cis gezählt, dann alle d usw. Daraus entsteht folgendes Histogramm: 12, 1, 6, 5.5, 0, 34, 0, 16.5, 0, 15, 9, 0. Wendet man den KS-Algorithmus auf dieses Histogramm an, erhält man folgende Korrelationswerte: Cmaj: 0,45; C#/Dbmaj: -0,14; Dmaj: -0,17; D#/Ebmaj: 0,28; Emaj: -0,62;

35 Pearson-Korrelation mit x als Tonhöhen-Histogramm aus dem zu analysierenden Bereich und y als Vergleichswerte für jede Tonart. x-Strich und y-Strich sind die jeweiligen Durchschnittswerte.

36 Krumhansl, *Cognitive Foundations*, S. 25–31.

37 Vgl. keycor manpage, <<http://extras.humdrum.org/man/keycor/>>, 28.6.2018.

Fmaj: 0,76; F#/Gbmaj: -0,28; Gmaj: -0,02; G#/Abmaj: 0,11; Amaj: -0,44; A#/Bbmaj: 0,79; Bmaj: -0,69.

Man erkennt, dass in diesem dreitaktigen Ausschnitt die Tonarten F-Dur mit 0,76 und Ais bzw. B-Dur mit 0,79 die höchste Korrelation aufweisen und somit als eventuelle Tonarten in Frage kommen.

Abbildung 8: Verovio-Rendering der MEI-Codierung von „Von der schlummernden Mutter“ von Anton Bruckner, Anfang, Kitzler-Studienbuch, S. 22

Nicht selten treten Tonartänderungen innerhalb eines Stückes auf. Der KS-Algorithmus auf das gesamte Stück angewendet, könnte diese Änderungen logischerweise nicht erkennen. Eine notwendige Folge ist, die Musik in Segmente zu unterteilen, die dann jeweils mit dem KS analysiert werden. Die Frage ist, welche Abschnitte sinnvoll sind und das beste Ergebnis liefern. Bei ersten Tests lag die Spanne der Abschnitte zwischen acht Takten und jeder Zählzeit. Die Ergebnisse variierten bei Veränderung der Größe der Abschnitte jedoch stark, so dass hier keine eindeutige Lösung gefunden werden konnte. Das Problem hierbei ist, dass bei großen Abschnitten zu global gerechnet wird und Binnenänderungen der Tonarten nicht berücksichtigt werden. Bei kleinen Abschnitten besteht die Gefahr, dass zu wenig Informationen vorhanden sind, um ein passendes bzw. aussagekräftiges Ergebnis zu erhalten. Craig Sapp hat dieses Problem gelöst, indem er für die auf Humdrum basierenden Codierungen den KS-Algorithmus auf alle möglichen Segmentierungen eines Musikstücks anwendet und sie graphisch darstellt (s. Abb. 9 und 10).

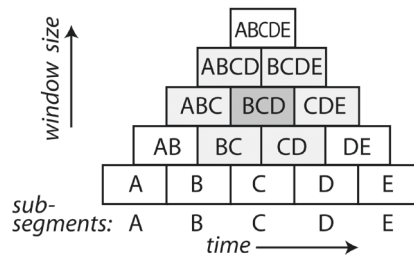


Abbildung 9: Segmentierung eines Musikstücks zur Berechnung der Tonarten, in: Craig Sapp, *Computational Methods for the Analysis of Musical Structure*, Stanford 2011, S. 96

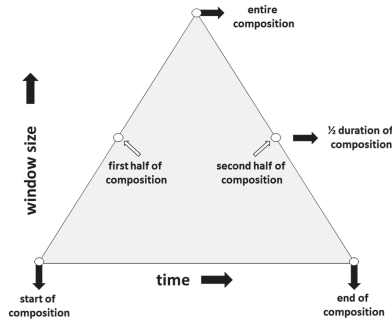


Abbildung 10: Schematische Darstellung einer Komposition, nach: Sapp, *Computational Methods*, S. 73

Die horizontale Achse beschreibt die Zeit vom Anfang des Musikstücks bis zum Ende. Die vertikale Achse beschreibt den Ausschnitt des Stücks, der dem Algorithmus übergeben wird. Somit repräsentiert die Spitze des Dreiecks das gesamte Stück und hat demnach nur einen Wert, eine Tonart. Auf der unteren Kante des Dreiecks werden die kleinsten Abschnitte, also Takte oder sogar Zählzeiten, dargestellt. Von oben nach unten werden die berechneten Ausschnitte immer zahlreicher und kleiner, also auch immer feiner in ihren Ergebnissen. Je nachdem, welcher Ausschnitt aus dem Dreieck angeschaut wird, können mehr oder weniger Details über die Tonartenstruktur erkannt werden. Sapp nennt dieses Diagramm „keyscape“, abgeleitet von „landscape“, weil es, wie in einer Landschaft, Phänomene im Vordergrund gibt, und welche, die im Hintergrund und globaler angesiedelt sind.³⁸ Dabei wird jeder Tonart eine Farbe zugewiesen. In Abbildung 11 ist das „keyscape“ der *Variation* Nr. 6 von Franz Schubert in Beziehung zu dem Notenausschnitt abgebildet.

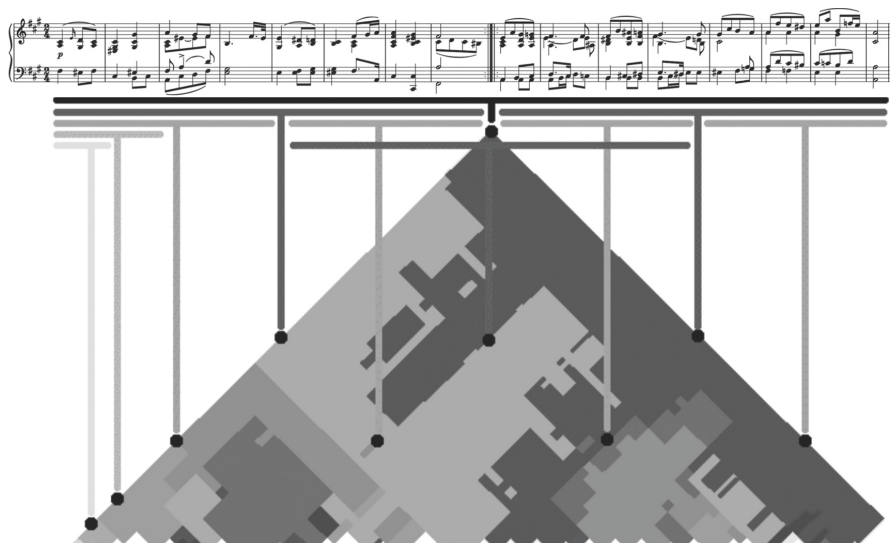


Abbildung 11: Keyscape von Schuberts *Variation* Nr. 6 im Vergleich zum jeweiligen Notenausschnitt, nach: Sapp, *Computational Methods*, S. 76.

38 Vgl. Sapp, *Computational Methods*, S. 74f.

Wie oben erklärt, gibt es verschiedene Profile des KS-Algorithmus, die zum Berechnen benutzt werden können und die sich in ihren Analyse-Ergebnissen immer leicht unterscheiden. Das eigentlich Komplizierte an Modulationen ist aber im Grunde nicht die Identifikation der verschiedenen Tonarten, sondern die „Grenzen“ der Tonarten, die Bereiche, in denen sich die Veränderungen abspielen. Ausschlaggebend ist die Bedeutung der Akkorde, die in verschiedenen Kontexten unterschiedliche oder mehrere Bedeutungen haben können. Grundlage der Akkordanalyse ist die Erkennung der Tonart bzw. des Grundtons. Mit der Segmentierung und der überlappenden Berechnung von Tonarten mit dem KS-Algorithmus werden diese Grenzen zumindest besser lokalisiert. Ambivalenzen und Doppeldeutungen erkennt man bei der händischen Analyse selber, weshalb auch jede Analyse anders ausfällt. Dieses Problem lässt sich kaum oder nur schwer mit einem Algorithmus lösen, obwohl Temperleys Algorithmus versucht, unterschiedliche Interpretationen der Analyse zu bewerten.

Der Ansatz hier ist, dem User kein eindeutiges Ergebnis der Tonarten zu liefern, sondern ihn dabei zu unterstützen, das für ihn „richtige“ Ergebnis zu finden. Bevor dem User eine Akkord- oder Stufenanalyse vorgelegt wird, muss er selbst entscheiden, wo ggf. die Tonartänderungen stattfinden. Dafür wird die Tonart des gesamten Stückes bestimmt, was mit dem KS-Algorithmus selten Probleme darstellt, solange es sich um Musik aus dem Dur/Moll-tonalen Bereich handelt. Daraufhin wird jede Ausschnittsgröße im Stück auf ihre Tonart hin untersucht. Das heißt, neben der globalen Betrachtung des Stückes werden die Abschnitte immer kleiner, von acht Takten zu sieben, sechs Takten usw., wobei diese Abschnitte auch jeweils mit Überlappung berechnet werden, um die Tonartänderungen noch besser lokalisieren zu können. Diese Masse an Informationen und Daten über die Tonarten sollen den Usern schließlich in einem Analyse-Tool zugänglich gemacht werden, in dem diese die Größe des Ausschnitts selbst wählen können und die Tonartänderungen visualisiert dargestellt bekommen. Sie können also nach eigenem Ermessen festlegen, welche Tonarten an welcher Stelle sinnvoll sind und auch, wo und wie die Tonartänderungen stattfinden, welche Tonarten sich wo überlappen und sich dann für eine „Lösung“ entscheiden.

Aufbauend darauf wäre es dann möglich, nach Bestätigung der Tonarten-Abschnitte durch den User, den von Temperley entwickelten Algorithmus zur Grundton- bzw. Akkorderkennung heranzuziehen und mit diesen beiden – KS-Algorithmus und dem Melisma Music Analyzer – auch eine Stufenanalyse durchzuführen. Die bestehenden Programme sollen aber weder verbessert werden, noch soll ein genaueres Analyse-Werkzeug programmiert werden. Sie sollen lediglich in einer benutzerfreundlichen Web-Oberfläche nachgenutzt werden. Ein Alleinstellungsmerkmal dieses Ansatzes ist, dass die Datengrundlage der Codierungsstandard MEI ist und immer wieder in die Ausgangsdaten zurückgegangen werden kann.

Abstract

Studying the harmonic structures of a musical work and exploring its origins is one of the main tasks of traditional musicology. Since the advent of computer technologies, new tools for musical analysis emerged to gain new perspectives on well-known compositions. In the field of digital musical editions, the markup language MEI (Music Encoding Initiative) plays a prominent role for encoding musical notation with a musicological demand. This paper presents the current state of the project “Digital Music Analysis with MEI using the Example of Anton Bruckner’s Compositional Studies”. Its aim is to encode the “Kitzler Study book” written by Bruckner and to present it in a digital Edition. Also, the project explores the capability of MEI for an automatic or half-automatic harmonic analysis.