

---

# „FROM TUNING TO SPECTRUM“

BERECHNUNG VON SPEKTREN ZU TONLEITERN BEI MINIMIERUNG  
VON DISSONANZEN MITTELS EVOLUTIONÄRER ALGORITHMEN

Martin Schobert

<martin@schwingungsebene.de>

---



2008

# *Inhaltsverzeichnis*

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Bewertung der sensorischen Dissonanz</b>	<b>2</b>
2.1	Sensorische Dissonanz zweier Sinusschwingungen . . . . .	2
2.2	Wie funktioniert das Dissonanzmeter von Sethares? . . . . .	4
2.3	Grenzen des Modells . . . . .	7
<b>3</b>	<b>Anwendung Evolutionärer Algorithmen</b>	<b>10</b>
3.1	Modellierung der Individuen . . . . .	10
3.2	Modellierung der Tonleiter . . . . .	11
3.3	Bewertung der Fitness . . . . .	11
3.4	Einschränkungen und Auswahl . . . . .	12
3.5	Rekombination . . . . .	14
3.6	Mutation . . . . .	14
3.7	Integration . . . . .	15
<b>4</b>	<b>Schlussbemerkungen</b>	<b>17</b>

# 1 Einleitung

In den verschiedenen Musikkulturen dieser Welt ist das Konzept der Tonleiter in Theorie und Praxis durchweg gebräuchlich. Die Tonleiter definiert eine spielbare Menge von Frequenzen bzw. von Intervallen. In den einzelnen Epochen der Musikgeschichte und in den verschiedenen Regionen haben sich Tonleistersysteme entwickelt, die sich zwar in ihrem Charakter unterscheiden, aber dennoch eine Eigenschaft gemein haben, und zwar die Diskretisierung eines an sich kontinuierlichen Frequenzraumes.

Im vorliegenden Beitrag wird ein Modell der Wissenschaftler W. J. M. Levelt und R. Plomp aus den 1960er Jahren vorgestellt. Dieses geht stets vom Auftreten sensorischer Dissonanzen aus, sobald man wenigstens zwei Sinusschwingung unterschiedlicher Frequenz spielt. Demnach ist allen Tonleistersystemen, sowohl gebräuchlichen als auch selbsterstellten, inhärent, dass mehr oder weniger wahrnehmbare Dissonanzen vorhanden sind.

Bedingt durch die Bauart eines Instruments oder die Generierung eines Samples am Computer, weisen Töne, die Musiker verwenden, ein ganzes Spektrum von Partialtönen auf. Ein einzelner Ton hat nach dem Modell von Levelt und Plomp eine Eigendissonanz. Spielt man mehrere Töne des Instruments oder Samples parallel, treten zusätzliche Dissonanzen auf.

William A. Sethares präsentiert in seinem Buch *Tuning, Timbre, Spectrum, Scale* ausgehend vom Modell von Levelt und Plomp Wege, Samples zu berechnen, so dass bei der Verwendung mit einer Tonleiter möglichst wenig Dissonanzen auftreten. Dieser Beitrag geht auf die Grundlagen des Verfahrens ein und setzt sie praktisch mit der Software Matlab der Firma *The MathWorks* um.

## 2 Bewertung der sensorischen Dissonanz

Sethares stellt ein Messverfahren vor, um Dissonanzen qualitativ zu bewerten. Er nennt es Dissonanzmeter. Historisch gibt es eine Reihe von Erklärungen für die Begriffe Konsonanz und Dissonanz. Für die Realisierung seines Dissonanzmeters greift Sethares auf eine Studie von Levelt und Plomp zurück, die den Zusammenhang von Konsonanz und *kritischer Bandbreite* untersucht haben.

Die kritische Bandbreite ist ein Begriff aus der Psychoakustik. Zwei überlagerte Schwingungen annähernd gleicher Frequenz bilden eine Schwebung. Sie wird vom Ohr aufgrund der begrenzten Frequenzauflösung als einzelner rauher (dissonanter) Ton wahrgenommen, der in seiner Lautstärke schwankt. Diese Art der Dissonanzwahrnehmung führt zum Begriff *sensorische Dissonanz*. Die kritische Bandbreite gibt den Frequenzabstand zweier gleichzeitig präsentierter Schwingungen an, die das Ohr gerade noch als getrennte Töne wahrnimmt.

Die Bewertungen für Dissonanzen nach Levelt und Plomp wurden laut Sethares auch für andere Musikkulturen verifiziert. Sie gelten für die Beschreibung der Empfindung von Sinusschwingungspaaren als allgemein akzeptiert. [Set05, S. 47]

Ursprünglich haben Levelt und Plomp den Begriff der *tonalen Konsonanz* gewählt. [RP65, S. 548] Der Begriff *sensorische Dissonanz*, den auch Sethares verwendet, wurde von Ernst Terhardt nahegelegt. [Set05, S. 48]

### 2.1 Sensorische Dissonanz zweier Sinusschwingungen

Levelt und Plomp führten Experimente mit musikalisch nicht trainierten Personen durch. Diese sollten die Konsonanz bzw. Dissonanz von Sinusschwingungspaaren auf einer Skala von eins bis sieben bewerten. Dabei bedeutet sieben maximal konsonant und eine Bewertung von eins spiegelt maximale Dissonanz wider. Frühere Studien zum Thema Konsonanzbewertung mit musikalisch trainierten Versuchsteilnehmern zeigten, dass diese ein bereits stärker geprägtes Verständnis von Konsonanz haben als untrainierte Versuchspersonen. Deshalb bewerten trainierte Personen entsprechend eher nach ihrer musikalischen Prägung als nach ihrer Wahrnehmung während der Versuche. Um diesen Effekt zu reduzieren, wählten Levelt und Plomp unbefangene Versuchspersonen. Den Versuchsteilnehmern, die nach der Bedeutung des Begriffs Konsonanz fragten, wurde Konsonanz mit den Begriffen Schönheit und Wohlklang beschrieben. Die Angaben der Probanden wurden auch auf Konsistenz geprüft. [Set05, S. 92] [RP65, S. 553]

Die beiden Autoren kommen zu dem Ergebnis, dass zwei gleiche Frequenzen als maximal konsonant empfunden werden. Für weiter auseinander liegende Frequenzabstände empfanden die Probanden ebenfalls Konsonanz, die aber geringer war als für zwei gleiche Frequenzen. Für einen zwischenliegenden Bereich ermittelten die Autoren höhere Dissonanzwahrnehmung.

Die Bewertung der Konsonanzen kann in Kurven graphisch dargestellt werden. In Abbildung 2.1 wird die Konsonanz in Abhängigkeit von der kritischen Bandbreite dargestellt.

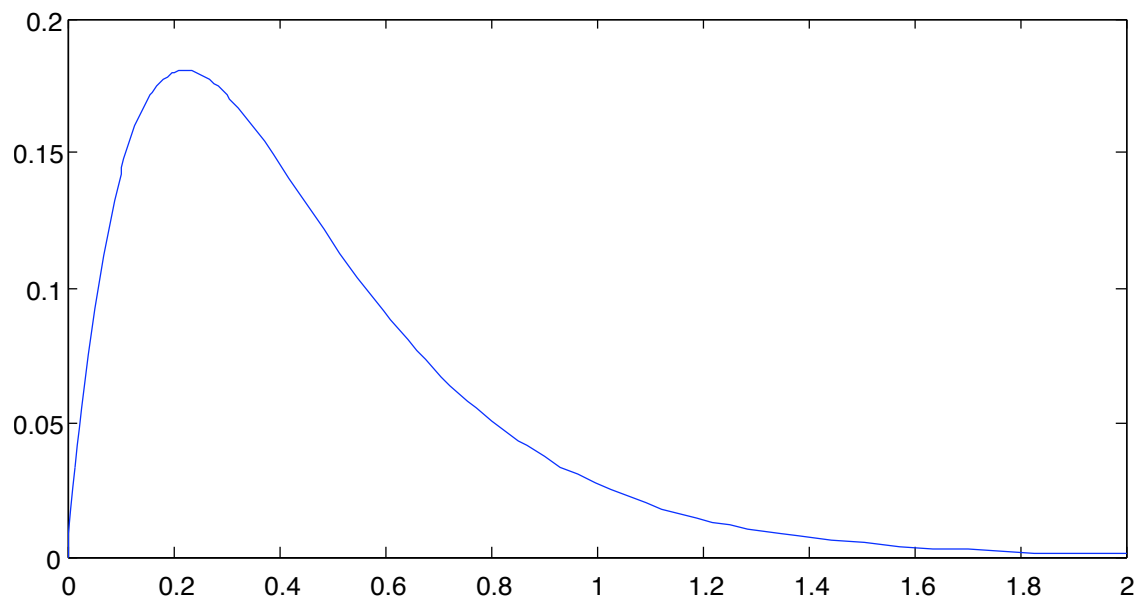


Abbildung 2.1: Darstellung der Ergebnisse von Levelt und Plomp. Die  $x$ -Achse gibt den Abstand zur Grundfrequenz in der Größe der kritischen Bandbreite an. Die  $y$ -Achse stellt das Maß für die sensorische Dissonanz dar. Der Wert *null* bedeutet maximal konsonant. Man kann die Kurve vertikal normalisieren, so dass maximale Dissonanz durch den Wert *eins* dargestellt wird. Darauf wurde in dieser Abbildung verzichtet.

Hermann von Helmholtz war der Ansicht, dass das Dissonanzmaximum unabhängig von der Mittelfrequenz der Sinusschwingungen ist. Levelt und Plomp widerlegen diese Einschätzung und ermitteln, dass die Stelle des Dissonanzmaximums von der kritischen Bandbreite abhängt und etwa bei einem viertel der kritischen Bandbreite liegt. Die kritische Bandbreite ist frequenzabhängig.

Sethares gibt als Modell für Levelt-Plomp-Kurven folgende Formel an [Set05, S. 345]:

$$d(x) = e^{-b_1 x} + e^{-b_2 x} \quad (2.1)$$

Dabei ist  $d(x)$  die Bewertung der sensorischen Dissonanz eines Frequenzabstandes. Dieser Frequenzabstand  $x$  wird aber nicht in der Einheit für die Frequenz, in Herz, dargestellt, sondern in Form der kritischen Bandbreite ausgedrückt.

Sethares bestimmt die Modellparameter  $b_1$  und  $b_2$  mit  $b_1 = 3.5$  und  $b_2 = 5.75$  an. Diese werden auch in der Abbildung 2.1 benutzt.<sup>1</sup>

## 2.2 Wie funktioniert das Dissonanzmeter von Sethares?

Das Bewertungsmodell von Levelt und Plomp basiert auf zwei gleichzeitigen und gleichlauten Sinusschwingungen. Das Modell, welches dem Dissonanzmeter von Sethares zugrunde liegt, geht aber von  $N$  gleichzeitigen Frequenzen mit verschiedenen Amplituden aus. Es bedarf also einer Erweiterung des ursprünglichen Modells.

Die Lautstärken  $l_1$  und  $l_2$  der beiden Schwingungen eines Frequenzpaares sollen als Parameter in die Bewertung der Dissonanz einfließen.<sup>2</sup> Dahinter steckt die Idee, dass eine Schwingungskomponente mit kleiner Amplitude weniger zur Dissonanz beitragen wird als eine Schwingung mit großer Amplitude. Die Komponente mit der kleineren Amplitude bestimmt dabei den Skalierungsfaktor für die Dissonanzbewertung.

$$d(x, l_1, l_2) = \min(l_1, l_2) * (e^{-b_1 x} + e^{-b_2 x}) \quad (2.2)$$

Der Parameter  $x$  ist wieder der Frequenzabstand im Sinne der kritischen Bandbreite. Im nächsten Schritt soll  $x$  explizit durch die Frequenzen  $f_1$  und  $f_2$  ausgedrückt werden.<sup>3</sup> Dabei soll die Dissonanzkurve bei der Grundfrequenz  $f_1$  beginnen und in der Einheit für die Frequenz, in Herz, erfolgen. Sethares gibt eine Formel an, die zu folgender arithmetisch äquivalent ist:

$$d(f_1, f_2, l_1, l_2) = \min(l_1, l_2)(e^{-b_1 s(f_2 - f_1)} + e^{-b_2 s(f_2 - f_1)}) \quad (2.3)$$

$$s = \frac{x^*}{s_1 f_1 + s_2} \quad (2.4)$$

Die Gleichung 2.2 ist unabhängig von der Frequenz. Die Frequenzabhängigkeit des Dissonanzmaximums und die Breite des Bereiches mit höherer Dissonanz ergeben sich nicht aus der Formel, sondern lediglich indirekt über die Frequenzabhängigkeit der kritischen Bandbreite. Aus Gleichung 2.3 muss sich dies aber explizit ergeben. Sethares führt deshalb den frequenzabhängigen Streckungsfaktor  $s$  in Gleichung 2.4 ein. Dieser soll bewirken, dass das Dissonanzmaximum bei einem Frequenzabstand auftritt, der in etwa einem viertel der kritischen

---

<sup>1</sup>Die Belegungen der Modellparameter sind für das Verständnis des Dissonanzmeters eigentlich nicht wichtig. Da diese Konstanten aber im Matlab-Code `dissmeasure.m` zu finden sind, soll sich deren Herkunft zum besseren Nachvollziehen aus dieser Arbeit ergeben.

<sup>2</sup> $l_i \in \mathbb{R} : l_i \geq 0$

<sup>3</sup>Es wird angenommen, dass  $0 < f_1 < f_2$  ist.

Bandbreite entspricht. Bildlich gesprochen wird der entsprechende Graph mit dem Faktor  $s$  horizontal gestaucht bzw. gestreckt.

Die Modellparameter sind wie folgt belegt:  $x^* = 0.24$ ,  $s1 = 0.0207$  und  $s2 = 18.96$ .

Wenn keine reine Sinusschwingung vorliegt sondern ein komplexerer Sound  $F$  bestehend aus  $N$  gleichzeitig auftretenden Partialtönen  $f_1, \dots, f_N$  mit den Lautstärken  $l_1, \dots, l_N$ , kann man dessen Eigendissonanz, die sogenannte *intrinsische Dissonanz*, berechnen. Dabei werden die Dissonanzen aller Paare von Partialtönen aufsummiert:

$$D_F = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N d(f_i, f_j, l_i, l_j) \quad (2.5)$$

Der Faktor 0.5 repariert dabei folgendes Problem: Aufgrund der Doppel-Summe in Gleichung 2.5 wird die Dissonanz vom Partialton  $[f_i, l_i]$  zu  $[f_j, l_j]$  aber auch von  $[f_j, l_j]$  zu  $[f_i, l_i]$  berechnet. Die Dissonanz zwischen beiden Tönen geht also doppelt in die Summe ein. Mit der Halbierung des Ergebnisses gleicht man den Effekt wieder aus.

Möchte man die Dissonanz eines komplexen Tons  $F$ , der im Intervall  $\alpha$ <sup>4</sup> gespielt wird, berechnen, so ergibt sich die Gesamtdissonanz aus der Summe der intrinsischen Dissonanzen der Töne  $F$  und  $\alpha F$ <sup>5</sup> sowie der Dissonanz zwischen allen Partialtönen aus  $F$  zu allen Partialtönen aus  $\alpha F$ :

$$D_F(\alpha) = D_F + D_{\alpha F} + \sum_{i=1}^N \sum_{j=1}^N d(f_i, \alpha f_j, l_i, l_j) \quad (2.6)$$

Anmerkung 1: Der Faktor 0.5 tritt im dritten Term in Gleichung 2.6 nicht auf, da die Partialtöne die über  $i$  und  $j$  indiziert werden, in „verschiedenen Tönen“ liegen.

Anmerkung 2: Die Gleichung gilt auch, wenn  $\alpha = 1$  ist. Die beiden Eigendissonanzen  $D_F$  und  $D_{\alpha F}$  sowie die Dissonanzen aller Partialtöne überlagern sich dann.

Auf der zu dem Werk von Sethares beiliegenden CD-ROM findet man die Implementation von Gleichung 2.5 in der Datei `dissmeasure.m`. Die Implementation von Gleichung 2.6 steckt in der Datei `dissmain.m`.<sup>6</sup> Der Code in der Datei `dissmain.m` variiert dabei den Skalierungs-

---

<sup>4</sup>Der Faktor  $\alpha$  soll  $\geq 1$  sein.

<sup>5</sup>Sethares benutzt diese Schreibweise. Sie bedeutet, dass jeder Partialton aus  $F$  mit dem Faktor  $\alpha$  multipliziert wird.

<sup>6</sup>Die beiden Dateien sind auch auf der Webseite des Autors unter <http://eceserv0.ece.wisc.edu/~sethares/comprog.html> gespeichert. (abgerufen am 5. November 2007)

faktor  $\alpha$  in kleinen Schritten, so dass eine Dissonanzkurve wie in Abbildung 2.2 entsteht.<sup>7</sup>

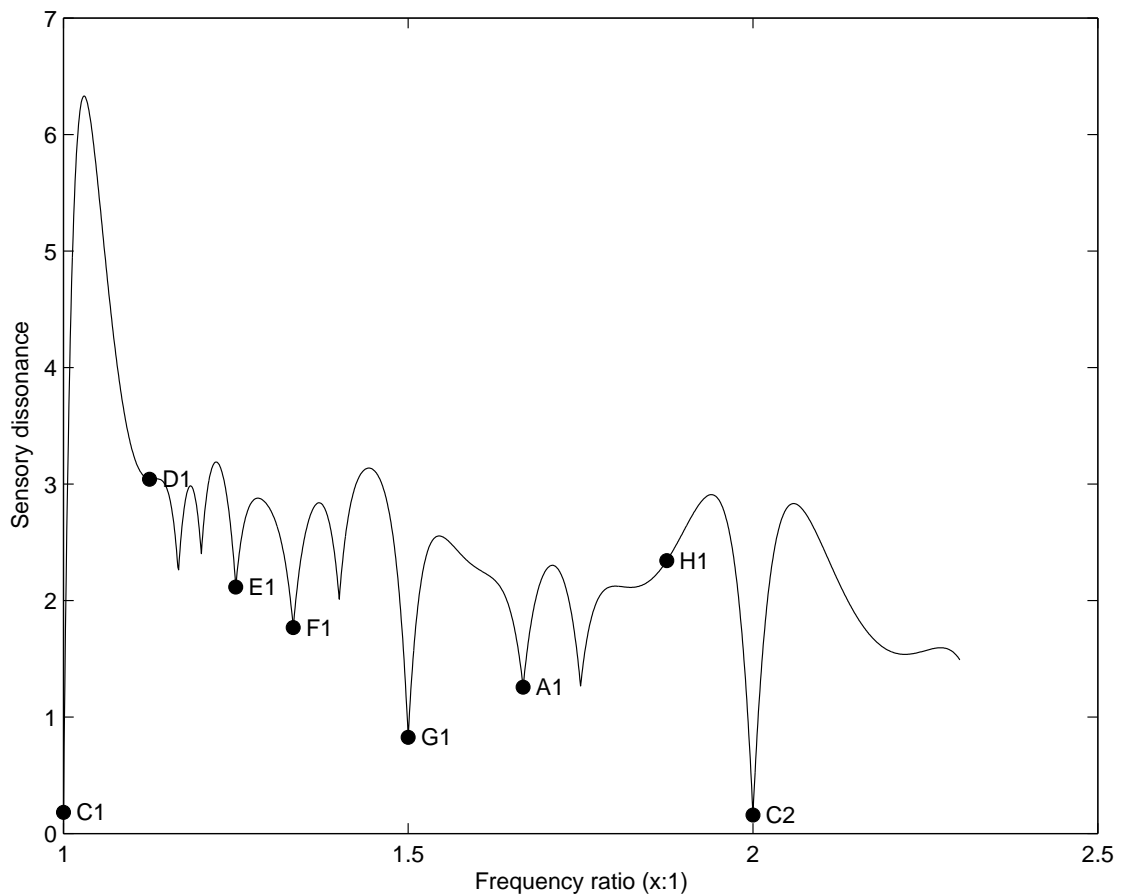


Abbildung 2.2: Dissonanzkurve für ein Sample  $F$ , bestehend aus den sieben Partialtönen mit den Frequenzen  $f = 500, 1000, 1500, 2000, 2500, 3000, 3500$  Hz. Jeder Ton hat die Amplitude 1. In den Graphen sind ebenfalls die Schritte einer C-Dur-Tonleiter eingezeichnet. An den Tonleiterschritten C, E, F, G, A und C (letzterer aus der nächsten Oktave) findet man Dissonanzminima. Es wird angenommen, dass beim Spielen des Grundtones C1, genau die Frequenzen aus  $F$  erklingen.

Sethares bedient sich in seiner Implementation des Dissonanzmeters eines Tricks. Statt  $D_F(\alpha)$  zu berechnen, konstruiert er einen neuen Ton  $F'$ , der sowohl den ursprünglichen Ton  $F$  als auch dessen Translation um den Faktor  $\alpha$  beinhaltet. Er berechnet dann die Eigendissonanz von  $F'$ . Sethares vereinfacht damit seine Implementation. Aufbauend auf dieser Idee, kann Gleichung 2.6 für gleichzeitig gespielte Intervalle einer Tonleiter  $R = [\alpha_1 = 1, \alpha_2, \dots, \alpha_m]$  verallgemeinert werden.<sup>8</sup>

Töne werden hier als Liste von Frequenzen und Lautstärken modelliert. Man kann nun eine Art Tonüberlagerung definieren, die eine Zusammenfassung von mehreren komplexen Tönen darstellt. Die Überlagerung ist eine geeignete Zusammenfassung von Listen-Elementen:

<sup>7</sup>In der Datei `dissmain.m` gibt es eine Vektor-Variable namens `diss`, die mit einem Element mit dem Wert 0 vorinitialisiert ist. Ich halte das für inhaltlich falsch und habe die Vorinitialisierung entfernt und lasse dafür den Parameter  $\alpha$  in der Schleife beim Wert 1 starten, so dass die Dissonanz  $D_F(\alpha = 1)$  durch den Algorithmus selbst berechnet wird.

<sup>8</sup>Die Modellierung der Tonleiter wird in 3.2 genauer erläutert.



$$F_A = [f_1^A, \dots, f_p^A, l_1^A, \dots, l_p^A] \quad (2.7)$$

$$F_B = [f_1^B, \dots, f_q^B, l_1^B, \dots, l_q^B] \quad (2.8)$$

$$F_A \oplus F_B = [f_1^A, \dots, f_p^A, f_1^B, \dots, f_q^B, l_1^A, \dots, l_p^A, l_1^B, \dots, l_q^B] \quad (2.9)$$

Mit der Operation  $\oplus$  kann man den Gesamttton  $F'$  bilden, für den dann lediglich die Eigendissonanz zu bestimmen ist:

$$F' = F \oplus \alpha_2 F \oplus \dots \oplus \alpha_m F \quad (2.10)$$

$$D_F(\alpha_1, \dots, \alpha_m) = D_{F'} \quad (2.11)$$

Den zusammengesetzten Ton  $F'$  kann man dann im Prinzip als Parameter an das Dissonanzmeter übergeben. Es ist dann sichergestellt, dass die Eigendissonanz von  $F$  nur einfach in die Gesamtdissonanz einfließt.

## 2.3 Grenzen des Modells

Das Modell von Levelt und Plomp und insbesondere das darauf aufbauende Dissonanzmeter von Sethares basieren auf einfachen Prinzipien. Im folgenden sollen einige Grenzen des Modells verdeutlicht werden:

In beiden Modellen wird nicht berücksichtigt, dass sich die spektralen Eigenschaften von Samples in Abhängigkeit von der Zeit ändern können. Das ist beispielsweise zu erwarten, wenn Saiten eines natürlichen Instrumentes mit Schwingungskomponenten erregt werden, die schnell abklingen oder bei elektronisch generierten Samples mit aufgeschalteten Effekten, beispielsweise einem Flanger- oder Phaser-Effekt.

Die Verwendung der Minimum-Funktion in Gleichung 2.2 ist nicht unproblematisch. Sethares räumt ein, dass die Berechnung der Lautstärke nicht ganz trivial ist. [Set05, S. 346] Er benutzt in Gleichung 2.2 die Minimum-Funktion, um den Beitrag der Lautstärke zur Dissonanz zu entscheiden. Ein genaueres Modell würde die psychoakustische Größe Lautheit berücksichtigen, denn die wahrgenommene Lautstärke ist nicht proportional zum Wert der Amplituden. Tatsächlich gibt es auch noch andere Möglichkeiten, den Beitrag der Lautstärke zur Dissonanz zu integrieren, beispielsweise über das Verhältnis der Amplituden.

Es ist möglich, Sounds mit gleicher Dissonanzbewertung zu konstruieren, die aber nicht gleichermaßen dissonant klingen. [Set05, S. 100, 101] Andererseits ist es möglich, nahezu gleich

konsonant klingende Sounds zu konstruieren, deren Dissonanzbewertungen sich im Bereich beispielsweise einer Zehnerpotenz unterscheiden. Da das eine Eigenschaft der Dissonanzbewertungsfunktion ist, soll das im Folgenden erklärt werden.

Die Abbildung 2.2 zeigt eine Dissonanzkurve für einen komplexen Ton mit den Partialtonfrequenzen  $f = 500, 1000, 1500, 2000, 2500, 3000, 3500$  Hz. Die Partialtöne haben die Amplitudenwerte  $l_{1..7} = 1$ . Man kann aus der Abbildung erkennen, dass die Dissonanzbewertung den Wert 7 für  $\alpha$  im Wertebereich  $[1, 2]$  nicht übersteigt.

Man kann zusätzliche Partialtöne mit Amplitudenwerten im Promillebereich der bereits vorhandenen Partialtöne einfügen und sich dann die Dissonanzbewertungen für verschiedene Belegungen von  $\alpha$  berechnen lassen. Für Abbildung 2.3 wurden zusätzliche 498 Störfrequenzen mit der Amplitude  $l = 0.0001$  bzw.  $l = 0.001$  eingefügt. Diese Störungen sind akustisch nicht wahrnehmbar. Da das Dissonanzmeter aber die Dissonanz zwischen allen Partialtönen berechnet und aufaddiert, fließen auch die Dissonanzen der nicht wahrnehmbaren Partialtöne in die Summe ein. Bei knapp 500 zusätzlichen Frequenzen sind das immerhin 125000 zusätzliche Dissonanzwerte. Diese sind jeweils extrem klein, bewirken aber in der Summe erhebliche Verfälschungen des Ergebnisses. Abbildung 2.3 zeigt zwei wesentliche Konsequenzen. Die Dissonanzwerte bewegen sich auf einem viel höheren Level. In der unteren Abbildung erreicht die Dissonanz sogar Werte bis etwa 140. Desweiteren sind Dissonanzminima in der Umgebung der Tonleiterschritte weniger stark ausgeprägt.

Die näherungsweise Umrechnung von Amplitudenwerten in Lautheit wird von Sethares angegeben [Set05, S. 346]. Die Berücksichtigung der Lautheit löst das Problem nicht, denn kleinere Amplituden tragen stärker zur Lautheit bei als größere Amplituden.

In Kapitel 7.1.3 erklärt Sethares anhand eines Beispiels, wie man relevante Partialtöne aus einem Spektrum extrahiert. [Set05, S. 134ff] Das ist auch ein brauchbarer Weg, um den Einfluss von Partialtönen mit geringer Amplitude zu unterdrücken.

Ursprünglich war für diesen Beitrag geplant, echte Sound-Samples als Ausgangsbasis für die Optimierung auf geringe Dissonanz hin zu verwenden. Mit einer Diskreten Fourier-Analyse sollten die einzelnen Frequenzen des Samples ermittelt werden. Bei diesem Schritt entstehen aber auch Phaseninformationen für die einzelnen Frequenzen. Diese sind für die Rekonstruktion der Zeit-Domäne essentiell. Optimiert man nun aber das Sample bezüglich der Konsonanz, ist aber nicht geklärt wie die Phaseninformationen weitergenutzt werden können, denn das Modell bezieht Phaseninformationen nicht mit ein. Insofern beschränkt sich die Anwendung auf wesentliche vorab extrahierte Partialtöne.

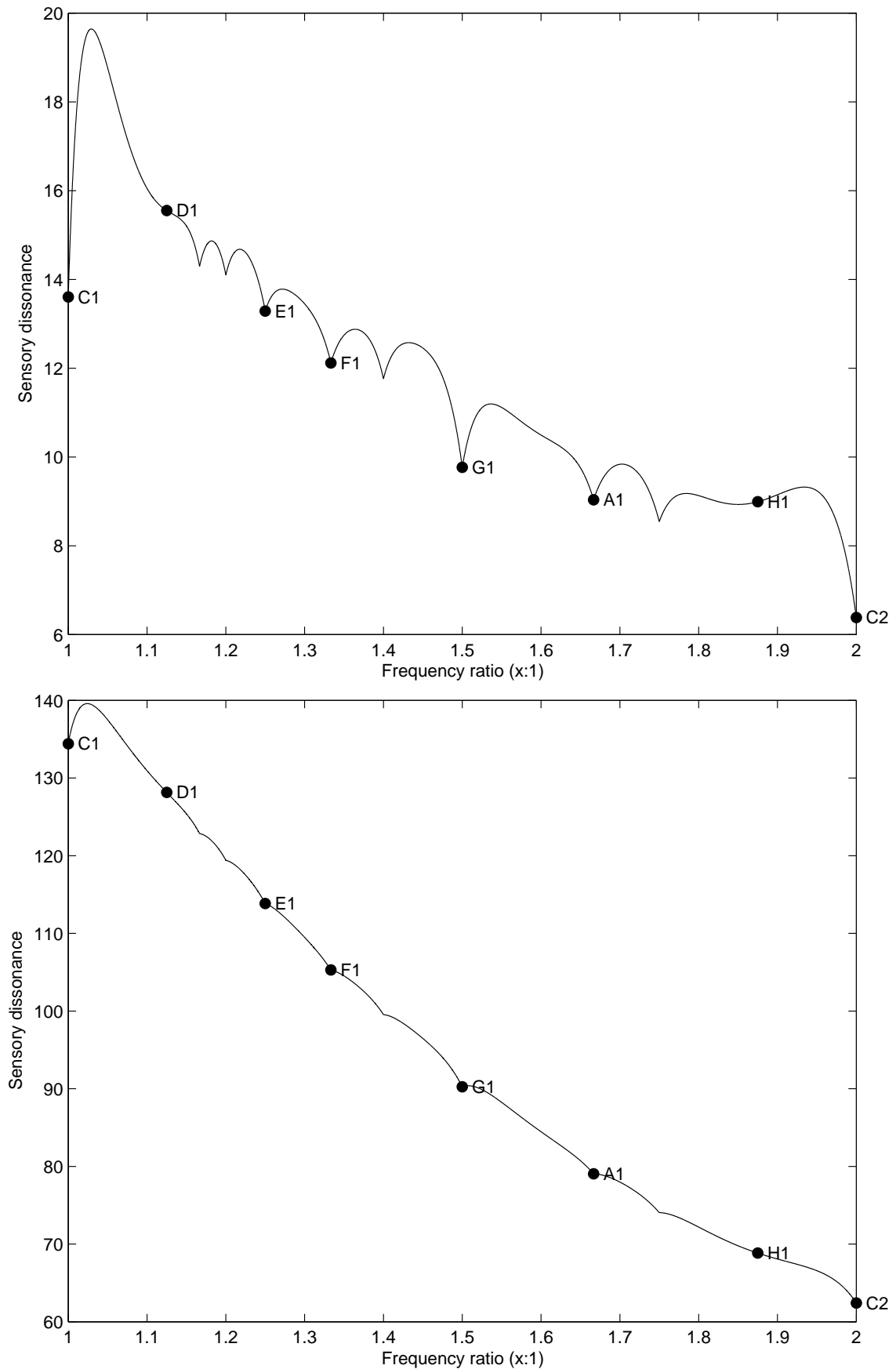


Abbildung 2.3: Einfluss von Störsignalen. Signal wie bei Abbildung 2.2 beschrieben. Zusätzliche Frequenzen  $f = 501 \dots 999$  Hz. Oben: mit Amplitude  $l = 0.0001$ . Unten: Amplitude  $l = 0.001$ .

### 3 Anwendung Evolutionärer Algorithmen

Evolutionäre Algorithmen stellen eine Klasse von Optimierungsverfahren dar, die anhand von Heuristiken Lösungen für nicht triviale Probleme suchen. Die Heuristik ist eine Zielfunktion, die die Qualität der Zwischenlösungen bewertet. Das Verfahren sucht nach Lösungskandidaten, die durch die Heuristik als optimal bewertet werden. Die Klasse heisst so, da sie abstrahierte Prinzipien der biologischen Evolution zur Grundlage hat.

Ausgangspunkt des Verfahrens sind eine Menge von *Individuen*. Jedes Individuum wird durch einen Satz von Parametern bestimmt. Die Menge der Individuen bildet eine *Population*. Die Qualität der Individuen als auch der Population wird über deren *Fitness* bewertet.

Das Verfahren arbeitet iterativ. In jedem Schritt werden Eigenschaften der „Eltern“ *rekombiniert* und *mutiert*, um eine neue Population von „Kindern“ zu erzeugen. Anhand der Fitnessbewertung wird aus der Menge der „Kinder“ die Population für die nächste Iteration *ausgewählt*. Ist die Fitness einer Population „gut“, ist sie als Lösung geeignet.

Sethares schlägt zur Suche von Spektren in Kapitel 12.4.2 die Verwendung Genetischer Algorithmen vor. [Set05, S. 252] Die Genetischen Algorithmen sind eine Unterklasse der Evolutionären Algorithmen. Sie unterscheiden sich von den Evolutionären Algorithmen durch den Wertebereich der Parameter. Während bei Evolutionären Algorithmen ein einzelner Parameter den Wertebereich der rationalen Zahlen  $\mathbb{R}$  hat, also der Parametersatz in den  $n$ -dimensionalen Raum  $\mathbb{R}^n$  abgebildet wird, ist der Wertebereich eines einzelnen Parameters Genetischer Algorithmen die Menge  $\{0, 1\}$  und der Wertebereich des Parametersatzes der Raum  $\{0, 1\}^n$ .

#### 3.1 Modellierung der Individuen

Die Individuen werden im vorliegenden Problem durch komplexe Töne repräsentiert, also durch Partialtöne bestehend aus Frequenzen und Amplituden. Frequenzen und Amplituden lassen sich einfacher durch reelle Zahlen darstellen. Es erscheint daher zweckmäßiger, die Töne mittels Evolutionärer Algorithmen als mit Genetischen Algorithmen zu finden, selbst wenn man dadurch den von Sethares vorgeschlagenen Weg zum Suchen mittels Genetischer Algorithmen verlässt. Letztendlich ist der Unterschied für die Lösung des vorliegenden Problems unwesentlich. Das Problem lässt sich aber durch Evolutionäre Algorithmen besser modellieren.

Ein Individuum  $i$  soll durch ein Tupel  $[f_1, \dots, f_n, l_1, \dots, l_n]$  repräsentiert werden, also durch  $n$  Frequenz- und  $n$  Amplitudenwerte. Das  $n$  kann prinzipiell individuell verschieden groß sein. Zur Beschleunigung des Algorithmus ist es sinnvoll, möglichst kleine  $n$  zu wählen. In der Implementation wird das  $n$  vorher über eine Konstante festgelegt. Unbenutzte Positionen

sind mit *null* belegt. Beispielsweise wird das Sample aus Abbildung 2.2 im Matlab-Code als folgendes  $2n$ -Tupel beschrieben:

$$I = [\underbrace{500, 1000, 1500, 2000, 2500, 3000, 3500, 0, 0, 0}_{\text{Frequenzen}}, \underbrace{1, 1, 1, 1, 1, 1, 1, 0, 0, 0}_{\text{Amplitudenwerte}}] \quad (3.1)$$

Die Individuen werden zu einer Population zusammengefasst. Im Matlab-Code wird das durch eine  $(N, M)$ -Matrix modelliert. Jede Spalte stellt ein Individuum dar.

## 3.2 Modellierung der Tonleiter

Eine Tonleiter kann man durch Frequenzen als auch durch Frequenzverhältnisse (Intervalle) zu einem Grundton darstellen. Das hier verwendete Modell besteht aus einer Menge  $R$  von Intervallen. Das Verhältnis des Grundtons zu sich selbst (1:1) wird aus praktischen Erwägungen ebenfalls gespeichert. Beinhaltet die Tonleiter  $m$  Töne, so ist  $R = [\alpha_1 = 1, \alpha_2, \dots, \alpha_m]$  die Menge der Intervalle zum Grundton. Eine C-Dur-Tonleiter ist demnach beispielsweise wie folgt beschreibbar:

$$f_0 = \underbrace{264\text{Hz}}_C \quad (3.2)$$

$$R = [\underbrace{1}_C, \underbrace{1.125}_D, \underbrace{1.250}_E, \underbrace{1.333}_F, \underbrace{1.500}_G, \underbrace{1.667}_A, \underbrace{1.875}_H] \quad (3.3)$$

Anmerkung: Die angegebenen Zahlenwerte in  $R$  sind hier z.T. gerundet. Das Verhältnis des Grundtons zum Grundton der nächsten Oktave ist nicht in  $R$ .

Die Angabe eines Grundtons in Herz bringt eine Unklarheit mit sich, denn was bedeutet es, z.B. einen komplexen Ton  $F$  mit den Partialtönen 500 Hz, 1000 Hz und 1500 Hz bei  $f_0 = 264$  Hz zu spielen? Wenn in der Modellierung die Tonleiter nur durch Frequenzverhältnisse aus  $R$  ausgedrückt werden, kann man diese Unklarheit vermeiden. Man kann dann definieren, dass jede Partialfrequenz um einen Faktor  $s$  aus  $R$  höher gespielt wird. Der komplexe Ton  $F$  gleicht dann dem Grundton.

## 3.3 Bewertung der Fitness

Sethares gibt in Kapitel 12.2 seines Werkes eine Kostenfunktion  $C$  an. Sie rechnet die Summe  $S$  der Dissonanzen an den Stellen der Tonleiterintervalle und die Anzahl der Minima  $M$  an

den Tonleiterintervallen gewichtet zusammen:

$$C = w_1 S + w_2 M \quad (3.4)$$

$$S = D_F(\alpha_1, \dots, \alpha_m) \quad (3.5)$$

Die Anzahl der Minima lässt sich nicht ohne Weiteres bestimmen. Die Bestimmung erfordert, dass nicht nur die Dissonanz an der Stelle eines Tonleiterintervalles berechnet wird, sondern auch in dessen Umgebung. Man kann ein (lokales) Minimum vermuten, wenn sich für die Frequenzverhältnisse links und rechts um den Tonleiterschritt herum höhere Dissonanzwerte ergeben. Beispielsweise kann man durch Auszählen der Minima in Abbildung 2.2 wenigstens fünf Minima ermitteln — C, E, F, G und bei A. Ob es bei D ein Minimum gibt, ist nicht ganz klar. Abbildung 2.3 (unten) muß man sehr wohlwollend interpretieren, denn da ist die Situation alles andere als eindeutig. Die Dissonanzbewertung an den Tonleiterschritten ist der Dissonanzbewertung in deren Umgebung sehr ähnlich. Es gibt hier keine „tiefen Täler“.

Sethares schlägt aufgrund numerischer Experimente vor, die Gewichtung  $w_1$  um das 100fache größer zu wählen als  $w_2$ . Da bei komplexeren Tönen mit mehreren Frequenzen vergleichsweise große Dissonanzwerte gegenüber der Anzahl  $M$  auftreten werden und da noch eine Gewichtung erfolgt, wirkt  $M$  kaum auf die Gesamtkosten ein. Deshalb wird  $M$  nebst Gewichtung im folgenden schlicht vernachlässigt. Damit benötigt man dann auch nicht mehr die Gewichtung  $w_1$ . Für die Kostenfunktion gilt dann  $C = S$ .

Das Dissonanzmaß ist nach unten begrenzt. Eine Dissonanz von *null* bedeutet maximale Konsonanz. Hingegen kann nach dem Modell von Sethares die Dissonanz beliebig groß sein. Es soll deshalb vereinbart werden, dass kleine Fitness-Bewertungen bessere Bewertungen sind. Es wird also auf eine kleine Fitness optimiert.

### 3.4 Einschränkungen und Auswahl

Bei der Suche nach möglichst konsonanten Tönen tendieren die Suchalgorithmen zu Triviallösungen. Dies muss durch geeignete Mechanismen verhindert werden. Sethares nennt in Kapitel 12.2 [Set05, S. 245, 246] zwei Probleme: Bei der Suche werden Frequenzen mit kleineren Amplituden bevorzugt, da kleine Amplitudenwerte geringere Dissonanz bedeuten. Die zweite Triviallösung, die Sethares anspricht, ist das Auseinanderdriften der Partialfrequenzen. Weit auseinanderliegende Frequenzen werden nach dem Levelt-Plomp-Modell als weniger dissonant wahrgenommen.

Das Problem der Amplitudenabschwächung lässt sich verringern, indem man in jeder Generation die Amplituden der Individuen normalisiert. Der Algorithmus multipliziert einen Faktor auf alle Amplituden, so dass die Summe der Amplituden für die Fitnessbewertung stets konstant ist. Das verhindert, dass alle Amplitudenwerte auf *null* absinken. Der Mechanismus verhindert zwar nicht, dass einzelne Amplitudenwerte ausklingen, dafür werden aber die Amplituden der anderen Partialtöne verstärkt. Im Mittel ergeben sich durch ausgeblendete Partialtöne keine Fitness-Vorteile.

Während die Optimierung läuft, können sich aber mitunter doch Individuen durchsetzen, bei denen nicht alle Amplitudenwerte größer *null* sind. Um diesen Fall zu verhindern, gibt es eine Modifikation im Auswahlprozess. Für die nächste Generationen werden nur Individuen ausgewählt, die bestimmten Mindestanforderungen genügen. Neben der Fitness, die prinzipiell Auswahlkriterium ist, werden auch nur Individuen gewählt, für die kein einziger Partialton unterhalb oder oberhalb eines Schwellwertes liegen. Die Schwellwerte dafür kann man im Hauptprogramm festlegen.

Das zweite von Sethares genannte Problem der auseinanderdriftenden Partialtöne kann man dadurch umgehen, dass es eine Minimal- und eine Maximalfrequenz gibt. Durch Mutation sollen keine tieferen bzw. höheren Frequenzen entstehen können. Tritt das dennoch ein, wird die Frequenz auf die untere bzw. obere Grenze zurückgestellt.

Alternativ könnte man auch nicht gewünschte Lösungsverläufe mittels Kostenfunktion schlechter bewerten. Allerdings stellt sich dann die Frage, mit welcher Gewichtung man die ungeeigneten Lösungskandidaten einbezieht. Diese Gewichtung müsste man sorgfältig durchdenken oder experimentell ermitteln.

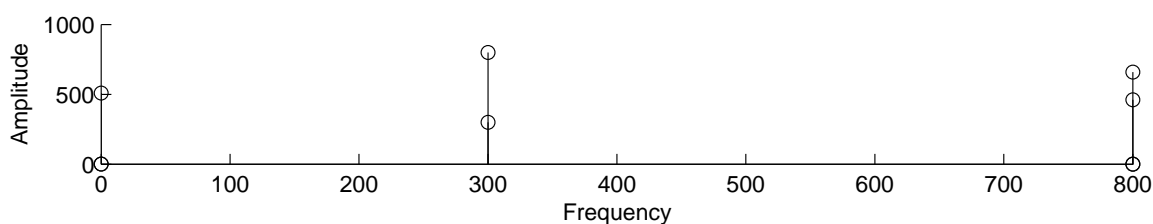


Abbildung 3.1: Hier ist ein Individuum mit minimaler Dissonanz entstanden, deren Frequenzen auffällig an den Intervallgrenzen des festgelegten Frequenzbereiches, und zwar 300 bis 800 Hz, liegen. Die Amplituden sind nicht normalisiert.

Diese Mechanismen verhindern jedoch keine fortgeschrittenen Trivallösungen. Bei umfangreichen Experimenten haben sich stets nach Modellierung von Einschränkungen neue Möglichkeiten für Trivallösungen ergeben. Abbildung 3.1 zeigt ein Beispiel: Nachdem die Amplitudennormalisierung und die Frequenzbegrenzung implementiert wurden, zeigten sich hauptsächlich Lösungen, bei denen die Frequenzen gerade noch im zulässigen Frequenzbereich lagen und diese Randfrequenzen im Individuum dann mehrfach auftraten. Die Darstellung der Individuen verhindert nicht, dass Frequenzwerte mehrfach gespeichert wurden. In diesem Fall konnten der-

artige Trivialsösungen dadurch herausgefiltert werden, dass man annähernd gleiche Frequenzen vereint.

### 3.5 Rekombination

Die Rekombination (engl. *cross-over*) soll Eigenschaften von Eltern vereinigen, um Kinder mit ähnlichen aber neuen Eigenschaften zu produzieren. Ähnlichkeit bedeutet hier bildlich ausgedrückt, dass die Nachfahren ähnliche Partialtöne wie ihre Eltern haben, also ähnliche Frequenzwerte und ähnliche Amplituden. Aus der Ähnlichkeit der Individuen sollen sich auch ähnliche Fitnessbewertungen ergeben.

Gegeben seien beispielsweise ein Elternteil  $A = [f_1^A, \dots, f_n^A, l_1^A, \dots, l_n^A]$  und ein Elternteil  $B = [f_1^B, \dots, f_n^B, l_1^B, \dots, l_n^B]$  mit jeweils  $n$  Partialtönen. Eine Form der Rekombination ergibt sich, wenn beispielsweise das Kind  $I$  weitgehend mit dem Elternteil  $A$  identisch ist, aber für einen zufällig gewählten Index  $i$  mit  $1 \leq i \leq n$  Eigenschaften vom Elternteil  $B$  übernommen werden:

$$I = [f_1, \dots, f_n, l_1, \dots, l_n] \quad (3.6)$$

$$f_j = \begin{cases} f_j^A, & \text{wenn } i \neq j \\ f_j^B, & \text{wenn } i = j \end{cases} \quad (3.7)$$

$$l_j = \begin{cases} l_j^A, & \text{wenn } i \neq j \\ l_j^B, & \text{wenn } i = j \end{cases} \quad (3.8)$$

Diese Form der Rekombination ist in der Matlab-Datei *crossover\_individuals\_at\_single\_pos.m* implementiert.

### 3.6 Mutation

Mutationen können als Einfügen, Löschen oder Ändern von Eigenschaften modelliert werden. Im vorliegenden Problem werden Partialtöne lediglich verändert. Die Funktion *mutate\_individual* wählt einen zufälligen Index im Vektor der gespeicherten Frequenzwerte aus und erhöht oder verringert den Wert der Frequenzkomponente um eine Schrittweite  $\Delta f$ . Auf die dazugehörigen Amplitude wird ein kleiner zufälliger Wert gerechnet.



### 3.7 Integration

Für die Anwendung von Genetischen und Evolutionären Algorithmen existieren eine Reihe von Frameworks, beispielsweise die „Genetic Algorithm and Direct Search Toolbox“ (GADS)<sup>1</sup> von The Mathworks oder „The Genetic and Evolutionary Algorithm Toolbox for Matlab“ (GEATbx)<sup>2</sup> von Hartmut Pohlheim. Beide Produkte sind kommerziell. Als freie und nichtkommerzielle Lösung existieren GPLAB<sup>3</sup> von Sara Silva in der GNU Library General Public License Version 2 und „The Genetic Algorithm Optimization Toolbox“ (GAOT)<sup>4</sup> als GNU General Public License Version 2 oder höher.

Es ist prinzipiell sinnvoll, auf derartige Frameworks zurückzugreifen, da die Grundkonzeption der Evolutionären Algorithmen stets gleich ist und sich die Probleme in der Auswahl und Ausgestaltung der Teilalgorithmen liegen. Desweiteren wäre bei Verwendung der existierenden Frameworks eine höhere Laufzeit-Performance zu erwarten. Da diese Frameworks dennoch hinreichend komplex sind und es einige Zeit bedarf, sich überhaupt damit zu beschäftigen, das Grundschema Evolutionärer Algorithmen aber sehr einfach ist, hat sich der Autor dazu entschieden, auf die Verwendung bereits existierender Frameworks zu verzichten und das Grundgerüst selbst zu implementieren. Das hat den entscheidenden Vorteil, dass man auf alle Aspekte direkten Einfluss nehmen kann, ohne sich damit auseinandersetzen zu müssen, wie jeweilige Teilprobleme innerhalb des Frameworks zu lösen sind.<sup>5</sup> Tatsächlich bestand dann auch der Hauptteil der Arbeit im Fein-Tuning der Algorithmen.

Das Hauptprogramm in *tuning\_to\_spectrum.m* besteht hauptsächlich aus Parameterbelegungen, der Generations-Schleife und Code zur Erstellung der Statistiken. Wichtige Zwischenergebnisse werden während des Programmlaufes regelmäßig graphisch wie in Abbildung 3.2 dargestellt. Die Grafik wird nach jeder Aktualisierung in der Datei *generationplot.pdf* gespeichert.

Die Hauptschleife endet nach einer Anzahl von Durchläufen oder nach Ablauf einer einstellbaren Zeitspanne. Die wesentlichen Datenstrukturen und Ablaufschritte sind im Hauptprogramm kommentiert.

Zwischendurch und nach dem Ende der Hauptschleife, spielt das Programm für ausgewählte Individuen die Tonleiter und mögliche Intervalle, und zwar für das bestbewertete Individuum, das bisher gefunden wurde, für das bestbewertete und schlechtestbewertete Individuum aus der

---

<sup>1</sup><http://www.mathworks.com/products/gads/>, Version vom 24. November 2007, 23:42

<sup>2</sup><http://www.geatbx.com/>, Version vom 24. November 2007, 23:42

<sup>3</sup><http://gplab.sourceforge.net/>, Version vom 24. November 2007, 23:42

<sup>4</sup><http://www.ise.ncsu.edu/mirage/GAToolBox/gaot/>, Version vom 24. November 2007, 23:42

<sup>5</sup>Unter Umständen eröffnet sich dadurch sogar eine größere Wahlmöglichkeit, unter welcher Lizenz die Implementation veröffentlicht werden kann.

aktuellen Generation. Nach Programmende verbleiben die Daten des Hauptprogrammes im Workspace von Matlab, so dass man gegebenenfalls Befehle über das Command Window in der Matlab-GUI absetzen kann, z.B. um noch einmal die Tonleiter mit einzelnen Individuen zu spielen. Das Sample des bestbewerteten Individuums wird nach Programmende als WAV-Datei gespeichert.

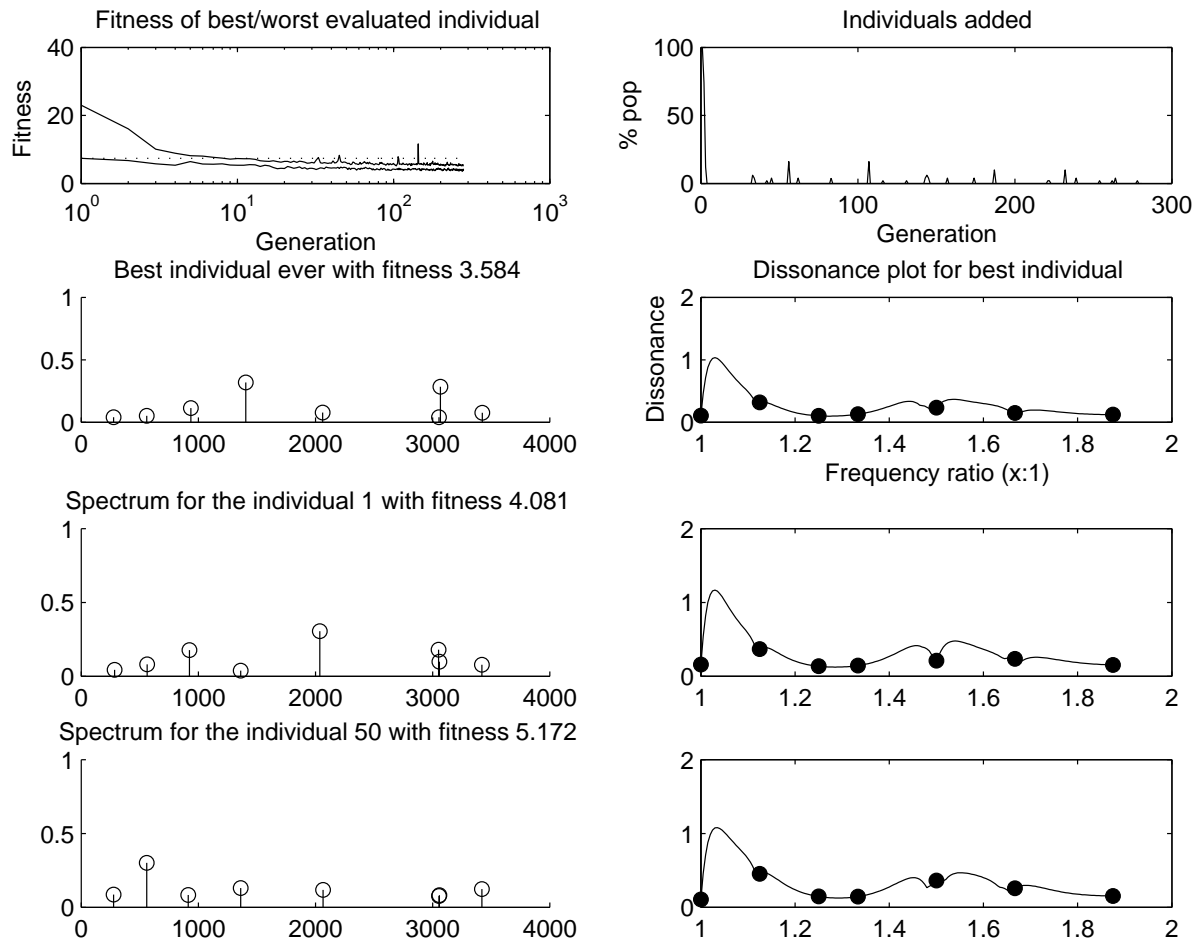


Abbildung 3.2: Darstellung der Zwischenergebnisse

## 4 Schlussbemerkungen

Ohne Betrachtung von Konsonanz und Dissonanz ist das Klang-Ergebnis ersteinmal ist in hohem Maße davon abhängig, wie die Parameter für den Suchlauf gewählt werden. Entscheidend sind der Frequenzbereich und die Anzahl der Partialfrequenzen. Dadurch ergibt sich bereits, ob eine Wellenform satt und weich klingt oder mehr quäkig wie ein Piezo-Pieper aus einer Grußkarte.

Sehr oft entstehen Spektren, in denen ein paar Frequenzen dominieren, weil ihre Amplitudenwerte vergleichsweise groß sind. Je nachdem in welchem Frequenzbereich die dominierenden Partialtöne liegen ergeben sich komplett verschiedene Klänge, die sich nur schwer vergleichen lassen.

Das Optimieren von Spektren für gegebene Tonleitern auf minimale Dissonanz basiert auf Operationen innerhalb des Modells von Sethares. Wie bereits in Abschnitt 2.3 dargestellt, hat das Modell einige Einschränkungen. Diese Einschränkungen übertragen sich nun auch auf das Ergebnis. Zwar ist es möglich, eine Menge von Partialtöne zu errechnen, so dass darauf aufbauende Klänge weniger dissonant klingen, allerdings stellt sich die Frage nach der praktischen Verwendbarkeit. Die ableitbaren Wellenformen klingen aufgrund ihres Sinus-Charakters stets nach elektronischer Orgel. Da die akustischen Testläufe, die das Programm ausführt auch ohne Effekte wie Anschlagdynamik und Abklingen arbeitet, klingen die Wellenformen besonders künstlich. Um die Frage nach der praktischen Verwendbarkeit zu klären, müsste man also schlicht ausprobieren, ob man Melodien mit einem potentiell geeigneten Individuum auf einem Synthesizer spielen kann.

Mitunter entwickeln sich auch Spektren, deren Konsonanzbewertung besser ist als andere Spektren, die aber doch als Wellenformen dissonant klingen, weil ein Partialton nicht zum Gesamtklang passt. Eventuell wird deshalb eine Nachbearbeitung des Spektrums notwendig sein.

Fazit: Das hier vorgestellte Verfahren erzeugt zwar gemäß dem Modell von Sethares konsonantere Klänge. Eine praktische Nutzung ist aber tendenziell nicht gegeben. Jede Form von Einflussnahme auf die Entwicklung der Spektren muss in das Verfahren implementiert werden. Jede Änderung am Verfahren muss man auf Brauchbarkeit testen. Für den Musiker ist das alles andere als praktisch. Dagegen wird ein Musiker bei Verwendung von speziellen Programmen, die für die Klangsynthese geeignet sind, beispielsweise Reaktor von *Native Instruments*, vergleichsweise sofort nutzbare Ergebnisse erzielen. Die praktische Umsetzung kann aber helfen, die derzeitigen Grenzen des Dissonanzmeters aufzuzeigen, denn sensorische Konsonanz bzw. Dissonanz komplexerer Klänge ist ein umfangreicheres Thema, als dass es sich mit einem vergleichsweise einfachen Modell erfassen lässt.

## *Literaturverzeichnis*

- [RP65] R. PLOMP, W. J. M. LEVELT: *Tonal Consonance and Critical Bandwidth*. The Journal of the Acoustical Society of America, 38(4):548–560, Oktober 1965. Der Artikel ist gegen Bezahlung digital unter <http://scitation.aip.org/getabs/servlet/GetabsServlet?prog=normal&id=JASMAN000038000004000548000001&idtype=cvips> abrufbar. Befreite Versionen des Artikels kursieren im Internet, z.B. unter [http://schubert.ece.drexel.edu/~ykim/classes/fall/fall/week4\\_files/Plomp1965.pdf](http://schubert.ece.drexel.edu/~ykim/classes/fall/fall/week4_files/Plomp1965.pdf) (OCR mit Layouterhaltung) oder <http://hdl.handle.net/2066/15403> (Scan). Abgerufen am 29. Oktober 2007 23:42 Uhr.
- [Set05] SETHARES, WILLIAM A.: *Tuning, Timbre, Spectrum, Scale*. Springer Verlag, 2. Auflage, 2005.