

Compunerea de muzică asistată de calculator

Constantin Prică

Universitatea Politehnica București
Bd. Splaiul Independentei, nr. 313 060032, București
E-mail: constantin.prica@cti.pub.ro

Rezumat. Păreră conform căreia calculatoarele nu pot avea un rol central în creația muzicală este greșită. Cercetări se fac în acest domeniu de cel puțin 50 de ani. În această lucrare, propunem o aplicație ce poate să ajute decisiv un compozitor în procesul de compunere muzicală. Genul muzical ales este unul puternic formalizat și anume contrapunctul strict. Regulile și constrângerile, specifice genului, se pretează pentru o implementare recursivă, de generare și testare a candidaților. Folosind această metodă, aplicația compune până la 10.000 de linii contrapunctice valide. Rezultatele obținute sunt promițătoare, fiind asemănătoare cu cele date de un compozitor uman. Ajutorul dat de calculatoare, dacă este acceptat, poate să transforme compunerea muzicală.

Cuvinte cheie: compunere de muzică, contrapunct, backtracking, evaluare muzicală.

1. Introducere

Creația muzicală nu este un domeniu asociat în mod natural cu calculatoarele. Încercarea de a defini muzica, într-un mod în care să poată fi apoi algoritmică, pare sortită eșecului. Creativitatea și originalitatea, două caracteristici importante ale oricărei piese muzicale, sunt lipsite de sens pentru un calculator. Sensibilitatea, centrală compoziției muzicale, nu poate fi tradusă în limbaj mașină.

Astfel de motive sunt cel puțin superficiale. De altfel, toate domeniile comune ale informaticii de astăzi s-au dezvoltat pe baza soluțiilor la probleme dintre cele mai diverse. Interesul în dezvoltarea de aplicații care să ajute compozitorii sau chiar să compună de la zero melodii a existat și încă există. Dovada cea mai grăitoare este numărul mare de cercetări și rezultate obținute până acum.

Fără îndoială, un calculator poate să facă anumite lucruri mult mai repede și mai corect decât un compozitor. O piesă muzicală trebuie gândită de la cel mai de jos până la cel mai înalt nivel structural. Compozitorul se concentrează pe imaginea de ansamblu, pe modul în care mișcările și

melodiile se întrepătrund pentru a forma un întreg organic (Roads 1996, 852). Munca de jos, la nivelul notelor și acordurilor, este de cele multe ori laborioasă și cu o doză mare de inerție. Aplicațiile de compunere muzicală încearcă să-l ajute pe utilizator tocmai la acest nivel, să-l scutească de aceste preocupări.

Folosirea programelor care compun piese sau oferă asistență ridică anumite probleme. Un program dă aceleași rezultate pentru aceleași date de intrare, indiferent de utilizator. Un astfel de produs muzical nu mai poate fi atribuit compozitorului, ce devine un simplu instrument care pornește programul și preia rezultatele. Piese rezultate vor fi cu greu acceptate de comunitatea muzicală, tocmai pentru că nu au individualitate. Rolul care îi rămâne calculatorului este cel pe care l-a avut de la creare, de a înlesni munca omului.

Această lucrare introduce o aplicație numită Expert Fux, ce oferă asistență prin compunerea de linii melodice contrapunctice. Ca gen muzical a fost ales contrapunctul pentru două voci, deoarece dispune de un formalism aproape complet, satisfăcător din punct de vedere computațional. Din cele cinci specii diferite, patru au fost implementate.

Scopul lucrării nu este compunerea unei piese muzicale de contrapunct de la un capăt la altul. De altfel și rezultatele din trecut, la astfel de încercări, nu au fost importante din punct de vedere muzical. O privire mai detaliată asupra cercetărilor anterioare este dată în secțiunea următoare.

2. Cercetări anterioare

Odată ce calculatoarele au devenit accesibile compozitorilor, aplicațiile nu au întârziat să apară. Acestea ofereau posibilitatea de testa strategii diferite, de a modifica parametrii doriți și de a observa imediat efectele (Roads 1996, 852). Mai mult, experimentele erau ușor de repetat. Programele de compunere și asistență permiteau o micșorare a timpului de lucru, asigurând metode facile de a reface și de a corecta produsul muzical.

Abordarea bazată pe formalism muzical este cea mai apropiată de felul în care un compozitor gândește și lucrează. Metoda ce constă în a genera note candidate ce sunt apoi supuse unor teste este una naturală pentru un calculator. Un număr mare de calcule este necesar pentru explorarea întregului spațiu de soluții. Pentru contrapunct, oferă cele mai bune rezultate. Regulile muzicale pot fi codificate și folosite atât în teste cât și la

generarea candidaților. Toate aceste motive au făcut ca aceasta să fie metoda de implementare aleasă pentru aplicația Expert Fux.

Primele încercări și reușite în a formaliza actul de a compune au fost făcute în anii '60 de către Lejaren Hiller și Leonard Isaacson (1959). Aplicația dezvoltată de aceștia genera o notă aleatoare care apoi era supusă anumitor teste. Dacă trecea de aceste teste eliminatorii, era adăugată la compoziția curentă. Dacă nu, altă notă candidată era creată și procesul se repeta. Această metodă este una de satisfacere a constrângerilor, fără optimizări. Hiller este cel care a publicat prima compoziție complet creată de calculator, suita Illiac în 1959. Sistemele de acest tip pot să aibă definite și anumite margini între care o notă este acceptabilă (Järveläinen 2000, 8). Un alt pionier, Ebcioglu (1990), a obținut rezultate surprinzătoare cu un sistem complex cu peste 350 de reguli interne, bazat tot pe metode de satisfacere a constrângerilor. Alte exemple de programe demne de menționat pentru această abordare sunt Henz, Lauer și Zimmerman (1996) și Ovans (1992).

Provenind din domeniul inteligenței artificiale, sistemele expert se folosesc de cunoștințe sub formă de reguli, fapte și euristici pentru generarea notelor. Faptele reprezintă parametrii locali sau globali. Regulile sunt condiționale, de genul „if clause then action” (Roads 1996, 902-904). Euristicile sunt pentru a selecta o soluție dintr-un spațiu mare de soluții posibile. Sunt de menționat aplicațiile dezvoltate de Schottstaedt (1984) și D.Cope (1987). O alta, mai nouă, este cea dezvoltată de Dion (2006). Toate sunt reușite în procesul de formalizare a limbajului muzical, însă, din punct de vedere strict estetic, nu depășesc nivelul de capabilitate al unui student al Conservatorului (Loy 1991, 26). În plus, baza de cunoștințe a aplicațiilor conține sute de reguli și fapte, fiind greu de extins și adaptat pentru alte genuri. Nu se pune problema ca sistemul să învețe încontinuu, din alte exemple.

3. Descrierea conceptelor muzicale

Aplicația se adresează unui utilizator familiarizat cu notația muzicală și teoria muzicală (Zeitlin and Golberger 2001). Domeniul este unul vast și complex. Din aceasta cauză, în această secțiune, doar anumite noțiuni elementare de teorie muzicală sunt prezentate.

3.1 Noțiuni de contrapunct

Contrapunctul este o tehnică muzicală de construire și combinare a două sau mai multe linii melodice (Dicționarul de termeni muzicali 2010). Termenul provine din expresia latină „punctus contra punctus”, însemnând notă contra notă. Contrapunctul poate fi foarte ușor confundat cu armonia. Și în tehnica armonică mai multe voci sunt combinate pentru a forma intervale consonante, disonante sau acorduri.

Spre deosebire de armonie, în contrapunct, vocile au o importanță egală (Miller 2005, 194-195). Ambele trebuie să fie melodii de sine stătătoare. De asemenea, trebuie să sune diferit, uneori contrar, creând un conflict sesizabil. Faptul că acest conflict este plăcut auzului, datorită unei îmbinări armonioase, este un aspect definitoriu al acestei tehnici melodice. Acordurile care apar într-o compoziție contrapunctică sunt de o importanță secundară, dar nu trebuie neglijate dacă se dorește o combinație organică. O simplă suprapunere sonoră de două sau mai multe voci total independente nu este contrapunct.

3.2 Specii de contrapunct

Preocuparea acestei lucrări este contrapunctul strict, formalizat de Johann Joseph Fux în lucrarea sa „Gradus ad Parnassum”(1725) (Mann 1971).

Contrapunctul este pentru două, trei, patru sau mai multe voci. Este implementat cel pentru două voci. De aceea toate regulile și observațiile se vor axa pe acest tip de contrapunct.

Tabelul 1. Definiția speciilor de contrapunct

Specie	Definiție
Specia I	Notă contra notă
Specia II	Două sau trei note contra o notă
Specia III	Patru note contra o notă
Specia IV	Două note sincopate contra o notă
Specia V	O combinație a celorlalte patru specii

Contrapunctul de tip strict, cântat fie de interpreți, fie la instrumente, pornește de la o melodie fixată, dată, numită „cantus firmus” (prescurtat

CF). Un sistem de reguli îi permite unui compozitor pornind de la această melodie, de multe ori simplă, să construiască celelalte voci. Cea de-a doua voce este numită contrapunct (CPT). Valențele armonice, de îmbinare a vocilor, sunt și ele considerate. Regulile, așa cum se va vedea în secțiunea următoare, sunt de obicei sub forma constrângerilor. A alege una dintre mai multe melodii posibile și corecte formal revine compozitorului și este o măsură a talentului său.

În cartea sa, Fux (Mann 1971) definește cinci specii sau tipuri de contrapunct, criteriul fiind numărul de note din ceea de-a doua voce pentru fiecare notă din cantus firmus.

3.3 Tipuri de reguli

Toate regulile au ca principiu fundamental variația sunetului (Mann 1971, 21). Liniile melodice nu trebuie să fie plictisitoare sau repetitive, sau fără individualitate.

În funcție de rolul pe care îl au în compunere, se pot diferenția două tipuri de reguli:

- Reguli sintactice – acestea se găsesc în tratate de muzică și definesc ceea ce este considerat valid în practica contrapunctică. Sunt pozitive sau negative. Cele pozitive sunt pentru crearea intervalelor și acordurilor, cele negative, adică constrângerile, sunt pentru eliminarea soluțiilor inadecvate.
- Reguli semantice – astfel de reguli asigură o evaluare estetică, muzicală a soluțiilor valide.

Regulile contrapunctului sunt regulile unui anumit stil. Pe lângă regulile din cartea „Gradus ad Parnassum” de către Johann Joseph Fux au mai fost folosite și cele din următoarele lucrări: (Belkin 2000), (Hitson 1962), (Nordon 1969) și (Kennan 1999).

Înainte de prezenta câteva dintre regulile care definesc contrapunctul strict anumite explicații și definiții sunt necesare.

Modurile sunt game diatonice formate numai din note naturale (Miller 2005). Sunt 7 la număr, fiecare pornind de la o notă a gamei numită tonică. Notele naturale corespund clapelor albe ale unui pian. Cele două linii melodice au note dintr-un singur mod muzical. Modul locric, din cauza unor neajunsuri muzicale, nu este folosit.

Intervalele care există între cele două linii melodice sunt de două tipuri: consonanțe și disonanțe. Consonanțele sunt stabile, pe când disonanțele nu, rezolvându-se într-o consonanță (Mann 1971).

Mișcarea, în muzică, reprezintă trecerea de la un interval la alt interval într-o anumită direcție, în sus sau în jos (Mann 1971). Sunt trei tipuri de mișcare pentru două voci: mișcare directă, mișcare contrară și mișcare oblică. Numele fiecăreia reprezintă modul în care prima linie melodică se dezvoltă față de cea de-a doua.

Nu vor fi prezentate toate regulile implementate, sunt peste 32 ca număr, ci doar câteva exemple:

- Singurele intervale permise sunt consonanțele.
- Singurele note permise sunt notele naturale ale modului muzical, cu unele excepții la terminarea contrapunctului.
- Mișcarea trebuie să fie contrară sau oblică între două consonanțe perfecte.
- Între o primă consonanță imperfectă și o a doua perfectă, mișcarea trebuie să fie contrară sau oblică.
- Pentru celelalte 2 combinații de intervale, toate tipurile de mișcări sunt acceptate.
- Intervalele de o cvintă și de o octavă paralele sunt interzise. Aceasta este o consecință a regulilor de mișcare.
- Contrapunctul trebuie să fie în același mod muzical cu cantus firmus-ul.
- Între note succesive ale contrapunctului, anumite intervale sunt interzise, de exemplu tritonul (interval muzical compus din trei tonuri).
- Intersectarea vocilor este, în general, de evitat.

4. Descrierea sistemului

Aplicația va prelua notele de început dintr-un fișier de tip MIDI și le va memora într-o structură de date (un vector). Acest tip de fișiere (MIDI File Format n.d.) conține toate informațiile muzicale necesare pentru compunerea contrapunctului. Notele din vector, în ordinea dată, reprezintă cantus firmus-ul. Numai valoarea notelor este de interes.

4.1 Motorul de generare

Generarea de contrapunct pornește de la vectorul de note. Motorul de generare a contrapunctului este recursiv, prin backtracking. Se caută, incremental, toate soluțiile posibile pentru o problemă. Candidații care nu pot forma o soluție validă sunt abandonați, algoritmul revenind la un candidat care este valid. Se încearcă apoi completarea acestuia.

Expert Fux găsește combinațiile valide în funcție de regulile speciilor. Peste un anumit număr de note, la unele specii, generarea tuturor combinațiilor posibile ar dura foarte mult timp. De aceea, backtrackingul se oprește după primele maxim 10.000 de soluții.

Inițial, se generează toate notele candidat, în funcție de anumite reguli formatoare, specifice speciilor. O notă candidat trebuie să treacă anumite teste ce țin cont de măsura curentă și de cea anterioară acesteia, ce este deja completată. Practic, contrapunctul este completat notă cu notă. Nota acceptată este memorată într-o linie melodică. Altă notă acceptată va fi memorată în altă linie melodică. Liniile melodice vor fi diferite între ele. Dintre acestea unele se vor termina, altele pentru că nu mai sunt note candidat valide vor fi abandonate. Vectorul de linii melodice este apoi evaluat. Funcțiile de evaluare sunt specifice speciilor și sunt obținute experimental. Se dorește ca cea mai plăcută linie melodică din punct de vedere estetic, muzical, să aibă cel mai mare grad. Aceasta va fi afișată pe portativ.

Începutul și sfârșitul unei melodii generate este fixat. Sunt mai multe începuturi posibile la anumite specii. Odată completată prima măsură, se pot genera restul.

CF-ul trebuie să fie într-unul din cele 7 moduri muzicale. Se acceptă și modul locric deși rezultatele vor fi deficitare. Melodiile trebuie să aibă doar note naturale. Astfel este implementată cerința impusă de Johann Joseph Fux. Determinarea modului se face la început prin verificarea primei note. Aceasta trebuie să fie tonica modului. Dacă melodia este într-o altă gamă sau chiar mai multe game, rezultatele sunt imprevizibile și greșite din punct de vedere al regulilor. Muzical pot suna plăcut, dar nu sunt contrapunct.

Și cantus firmus-ul trebuie să respecte anumite reguli. Ca de exemplu, nota de început și nota de sfârșit trebuie să fie tonica modului. Aplicația vine cu un număr de CF-uri valide.

Pentru tipul de contrapunct ales, cel puțin o notă este generată pentru fiecare notă din CF. Nu vor fi pauze între note.

CPT-urile sunt construite pe baza intervalelor verticale simple, adică mai mici decât o octavă. CPT-urile vor fi peste CF pe portativ. Expert Fux nu construiește contrapunct sub CF. Implementarea intervalelor negative este posibilă prin modificarea setului de reguli.

Nu sunt implementate mecanisme de verificare a consistenței arcelor, specifice problemelor de satisfacere a constrângerilor. Motivul este că notele candidat nu sunt memorate pentru mai mult de o iterație. Dacă se încearcă acest lucru, ținând cont de numărul mare de note candidat pentru fiecare notă din CF, memoria ocupată crește astronomic. În plus, constrângerile la speciile II, III și IV nu se pretează pentru astfel de tehnici. Unele dintre acestea au 8 variabile.

4.2 Arhitectura generală

Arhitectura este una orientată pe obiecte, flexibilă la adăugarea de noi funcționalități. Modul de organizare a claselor este prezentat prin intermediul următoarelor diagrame.

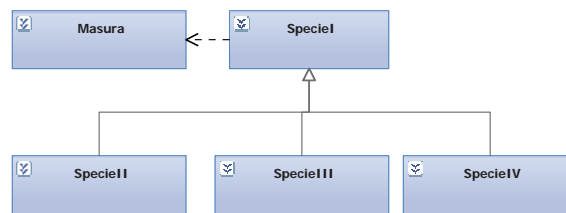


Figura 1. Clasele pentru speciile de CPT

Motorul de generare este implementat prin clasa Engine. Clasele SpecieI, SpecieII, SpecieIII și SpecieIV implementează testele eliminatorii. Clasa Masura conține informațiile despre notele CF-ului și CPT-ului. Interfața cu utilizatorul este clasa principală a aplicației, asigurând controlul asupra motorului de generare și a testelor eliminatorii.

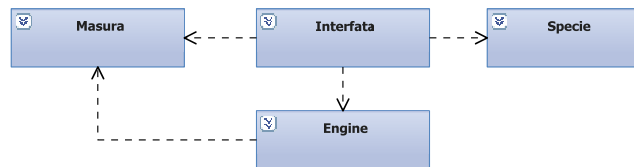


Figura 2. Clasele de care depinde interfața grafică

4.3 Interfața cu utilizatorul

Funcționalitățile interfeței grafice vin în satisfacerea necesităților utilizatorului. Aceasta permite încărcarea și salvarea melodiilor sub forma fișierelor MIDI, afișarea notelor pe portativ și redarea melodiilor în timp real.

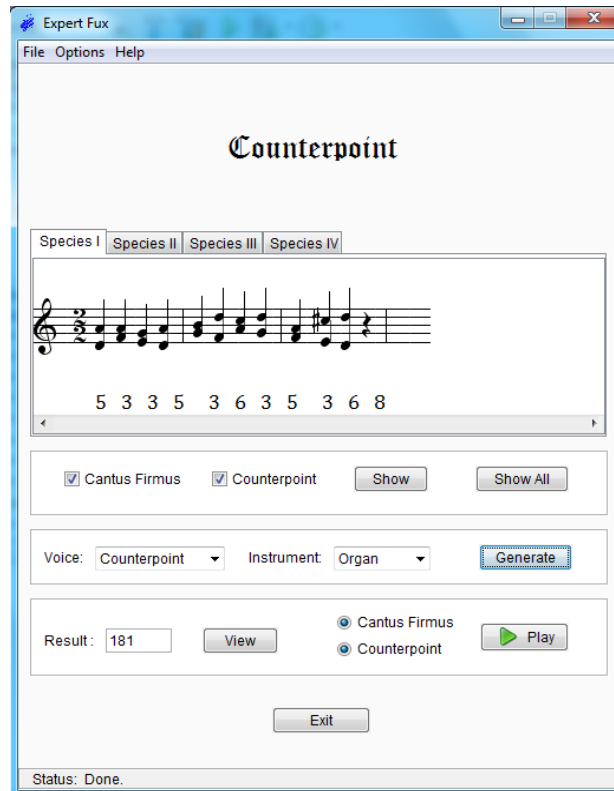


Figura 3. Interfața grafică principală

Se pot genera toate cele 4 specii diferite de contrapunct. Utilizatorul poate afișa sau salva oricare din soluțiile găsite. De asemenea, el poate seta instrumentele muzicale specifice liniilor melodice.

Este de preferat, din punctul de vedere al vizualizării portativului, ca notele CF-ului să fie între octava 3 și 5, octava 4 fiind ideală. Portativul muzical, cu notele dorite, este memorat ca un șir de caractere, conform specificațiilor notației abc (Gonzato 2003).

Exemplu de șir de reprezentare a notelor pe portativ.

X:1
 T: Contrapoint
 M:2/2
 K:C
 E2 C2 D2 C2 |A,2 A2 G2 E2 |F2 E2 z2z2 |

4.4 Interfețe secundare

Aplicația Expert Fux are două interfețe secundare, pentru personalizarea atât a motorului de generare cât și a fișierelor MIDI salvate.

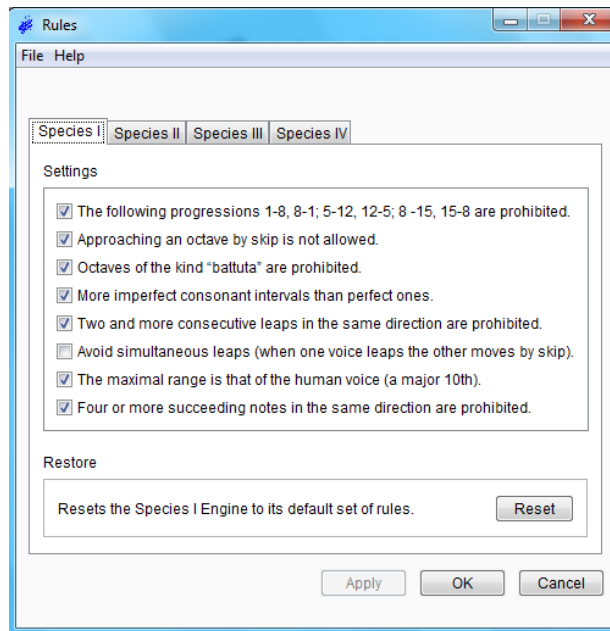


Figura 4. Interfața Rules

Opțiunile de personalizare sunt 32 la număr, fiind disponibile la rulare. Se pot activa sau dezactiva 8 reguli la fiecare specie. Efectele pot să fie majore, în ce privește numărul de soluții generate și calitatea acestora. Astfel, personalizarea motorului trebuie făcută cu mare grijă și în cunoștință de cauză. Se pune la dispoziția utilizatorului posibilitatea de a reveni în orice moment la configurația inițială.



Figura 6. Speciile I,II,III și IV de contrapunct pentru modul D

Intervalele generate sunt:

Tabel 2. Intervale generate

Specia I	5 3 3 5 3 6 3 3 6 6 8
Specia II	8 3 4 6 1 3 8 6 1 3 8 6 1 3 4 6 1 5 6 8
Specia III	2 3 4 3 4 5 3 3 4 3 6 6 7 8 5 3 2 3 4 6 7 8 6 3 4 3 6 6 7 5 6
Specia IV	8 7 8 1 3 4 5 6 8 5 3 6 7 6 7 6 3 5 6 1 -3 5 6 6 7 6 7 6 8

Intervalele prezentate de Fux sunt:

Tabel 3. Intervale Fux

Specia I	5 3 3 5 3 5 3 3 6 6 8
Specia II	8 3 4 6 3 5 8 3 1 3 4 3 1 5 3 5 3 5 6 8
Specia III	2 3 4 3 4 5 6 8 7 5 6 8 7 6 5 3 4 5 6 8 1 3 4 3 1 2 3 3 2 1 3 3
Specia IV	3 2 1 3 4 5 6 8 5 3 6 7 6 7 6 3 1 3 5 3 6 7 6 7 6 7 6 8

După cum se poate observa, rezultatele sunt asemănătoare la toate speciile.

La Specia I, contrapunctul urmărește de aproape CF-ul, ceea ce este un lucru bun. O singură notă diferențiază soluția aplicației de cea prezentată de Fux.

La Specia II, aplicația folosește de multe ori, pe cea de a doua bătaie, unisonul, folosit destul de rar la Fux. Acest lucru se datorează funcției de evaluare, ce caută un CPT cât mai apropiat de CF, dar melodic diferit. CPT-

ul lui Fux folosește intervale mici, ca terța sau cvinta. Salturile dintre notele CPT-ului generat sunt mai mari decât cele de la Fux, mai ales la schimbarea direcției de mișcare, ceea ce este un neajuns al funcției de evaluare. Varietatea de intervale este bună la ambele exemple.

La Specia III, ambele melodii sunt aglomerate. Specia III nu are implementat unison-ul pe toate bătăile și de aceea intervalele sunt mai mari decât la Fux. Melodia este destul de repetitivă și nu pare a avea un punct culminant. Pe de altă parte și melodia lui Fux este, pe portativ, repetitivă. Specia III nu are rezultate bune pentru un număr mare de note. Ea asigură culoarea unei secțiuni, o îmbogățește armonic. Mișcarea și tensiunea sunt secundare, urcând și coborând încontinuu. Este de observat că se obțin aceleași note la final.

La Specia IV, aplicația a generat mai multe soluții doar cu note sincopate, adică legate peste barele de măsură. Exemplul lui Fux nu este Specia IV pură, pentru că folosește și note din Specia II. În acest sens, generarea de muzică pe calculator își arată avantajele. Din punct de vedere melodic, Specia IV pură dă cele mai interesante rezultate. Este și cea mai dificilă de compus, existând puține variante posibile. Expert Fux a generat doar 4, pe când la Specia II sunt peste 10.000. Se observă asemănarea atât la început cât și la sfârșit între cele două CPT-uri.

Numărul de rezultate valide depinde de CF-ul inițial. O statistică pentru toate modurile este următoarea:

Tabel 4. Numărul de rezultate generate

Mod	Specia I	Specia II	Specia III	Specia IV(pură)
C	720	>10.000	>10.000	7
D	182	>10.000	6.132	4
E	249	494	282	6
F	408	>10.000	>10.000	2
G	1.999	>10.000	>10.000	229
A	728	>10.000	>10.000	2

Mai multe observații se pot face:

- În general, numărul de variante crește cu numărul de note din CF, a se observa modul G care are 14 note, restul având 10 sau 11.
- Sunt generate puține rezultate dacă CF-ul are caracteristici speciale la început sau sfârșit, salturi mai mari de o cvintă între note, etc. Se observă acest lucru la modul E care are un salt de o octavă la a șasea notă.

- Generarea durează mai mult la Specia III, deși ca număr de soluții aceasta nu diferă de Specia II, în multe cazuri. Acest lucru se datorează modului în care sunt construite notele candidat. La Specia III, la fiecare pas, sunt 368 de note candidat, pe când la Specia II sunt doar 20.

Este important de specificat că se generează doar primele 10.000 de rezultate posibile. Dacă generarea ar fi lăsată să continue, în unele cazuri ar bloca programul, în altele ar umple toată memoria sistemului de calcul. Acest lucru s-a întâmplat la Specia III unde numărul de soluții verificate este de ordinul miliardelor.

Sunt implementate mai multe începuturi de contrapunct, în cazul în care primul nu dă nici un rezultat. Ca exemple sunt modurile F și G care nu au rezultate valide pentru Specia III. Odată găsit un început de contrapunct cu soluții, nu se mai încearcă altul.

Contrapunctul afișat la toate speciile după apăsarea butonului de generare este cel mai bun, conform unei evaluări. Funcția folosită este foarte importantă. O mare parte din soluții sunt monotone și repetitive. Inteligența umană, care poate ușor aprecia estetic o melodie, se traduce în limbajul calculatorului în relații între intervale, note și mișcări. Aceste relații pot fi descoperite doar experimental, nu sunt formale. Bineînțeles că este de dorit ca cele mai interesante și melodioase soluții să fie găsite. Dar munca pentru descoperirea criteriilor care să permită alegerea corectă dintr-o mulțime de zeci sau chiar sute de mii de soluții este mai laborioasă decât compunerea de mână a contrapunctului. În plus, nu are un rezultat asigurat. Totuși, aceste observații nu trebuie să descurajeze. Descoperirea unui set de criterii de evaluare poate fi comparată nu cu compunerea câtorva melodii contrapunctice ci cu dezvoltarea unei tehnici complete de către un compozitor, însemnând o perioadă de câțiva ani.

5.2 Rezultate computaționale

Aplicația a fost dezvoltată pe un calculator cu un procesor AMD 3200+ și 2 GB memorie. Pentru toate operațiunile, acesta a fost mai mult decât suficient. Faptul că procesorul a fost single-core nu a însemnat o scădere evidentă a performanței, deși mai multe operații se execută în paralel.

Timpul de generare a soluțiilor este la speciile I, II și IV sub 15 secunde. Specia III, la unele moduri, are un timp mai mare, dar acesta nu depășește un 1 minut.

Caracterul melodic al cantus firmus-ului are cea mai mare influență asupra timpului de generare, determinând forma spațiului soluțiilor. Dacă în explorarea acestuia este nevoie de multe reveniri timpul crește. Dimensiunea spațiului este dată de numărul de note al CF-ului. Rolul acestui factor este unul secundar.

6. Îmbunătățiri

Mai multe îmbunătățiri se pot aduce aplicației atât ca funcționalitate cât și ca aspect.

Afișarea portativului poate fi îmbunătățită. S-a încercat ca fiecare linie melodică să fie pe un portativ propriu. Din păcate, biblioteca folosită nu are implementată, încă, această opțiune. Nu are implementată nici opțiunea de versuri la melodii astfel încât intervalele dintre CPT și CF sunt afișate separat.

Îmbunătățiri minore se pot aduce la generarea efectivă, prin implementarea unui număr mai mare de începuturi de contrapunct. Se pot implementa și mai multe reguli, deși numărul actual este suficient. Faptul că sunt generate primele 10.000 de soluții și nu cele mai bune 10.000 de soluții este determinat de spațiul imens de soluții posibile, chiar pentru un număr mic de note. Implementarea unor mecanisme de verificare a consistenței arcelor, ca în problemele de satisfacere a constrângerilor, înseamnă modificări majore în motorul de generare, dar este posibilă.

Evaluarea soluțiilor poate fi îmbunătățită. Cum am mai spus, descoperirea de criterii care să permită selectarea melodiilor interesante și plăcute auzului este experimentală. Cunoștințele de teorie muzicală și contrapunct, necesare, trebuie să fie vaste. De asemenea, influența melodiei inițiale nu trebuie neglijată. Dacă aceasta este monotonă, sunt șanse mari ca orice melodie generată să fie nesatisfăcătoare. Criteriile de evaluare sunt importante dar nu sunt singurul factor de care trebuie să se țină cont în generarea de contrapunct.

O metodă de a depăși această problemă de evaluare a rezultatelor este prin implementarea unor tehnici de învățare automată. Nu ar mai fi necesare cunoștințe avansate de teorie deoarece aplicația își va dezvolta propriile

criterii. Rolul utilizatorului se va transforma în cel al unui mentor care corectează rezultatele și ghidează aplicația în direcția dorită. Cel puțin două tehnici de învățare automată au fost folosite în compunerea muzicală: rețelele neurale și algoritmi genetici. O implementare cu rețele neurale este ceea ce a lui (Mozer 1994). Pentru algoritmi genetici a se vedea (Burton and Vladimirova 1999), (Johanson and Poli 1998) și (Acevedo 2004).

O ultimă îmbunătățire ar fi implementarea ultimei specii de contrapunct.

7. Concluzii

Aplicația propusă are ca idee de bază compunerea de muzică, dar scopul nu este de a genera compoziții originale de zeci de măsuri și sute de note. Un student la compoziție sau chiar un compozitor o poate folosi pentru rezolvarea unei probleme de contrapunct sau de armonie. Expert Fux oferă foarte multe soluții pentru un utilizator în pană de idei. Compunerea muzicală profesionistă, în muzica clasică și nu numai, pornește de jos. O întregă melodie pornește de la un motiv muzical de doar câteva note. Faptul că muzica este tot timpul variată, dar aceeași structural, este calitatea ce o separă de simplu zgomot. Însă, această muncă de jos este anevoioasă și fără satisfacții imediate. Aplicația Expert Fux poate să ajute, micșorând efortul depus în găsirea și dezvoltarea temelor muzicale.

În prezent, există aplicații mai vechi care generează contrapunct dar nu sunt disponibile utilizatorului comun. Se găsesc documentațiile dar nu programele efective. Expert Fux vine contra acestui curent, punând la dispoziție o aplicație gata să fie folosită.

Contrapunctul nu mai este o tehnică actuală de compoziție. A rămas însă o introducere foarte bună în această artă, dorită caracterului logic, aproape matematic. Epoca contemporană a văzut renașterea multor genuri și tehnici, astfel încât viitorul poate să rezerve surprize plăcute. Nu trebuie uitat că muzica barocului și mai ales aceea compusă de Bach este mai apreciată astăzi decât acum un secol.

Revenind la aplicație, rezultatele la testele făcute au fost foarte promițătoare. Contrapunctul generat este asemănător cu cel prezentat de Johann Joseph Fux. Faptul că în loc de o singură variantă, ca în cartea acestuia, aplicația poate să genereze sute sau mii în câteva secunde demonstrează în mod sigur validitatea compunerii muzicale automate.

Flexibilitatea datorată arhitecturii orientate pe obiecte permite implementarea facilă a îmbunătățirilor prezentate în secțiunea precedentă.

Muzica nu este numai sunet ci și note pe portativ. Apariția unei notații unificate și complete a însemnat un uriaș câștig în răspândirea și dezvoltarea muzicii. O aplicație de compunere muzicală trebuie să poată să afișeze melodiile, în felul în care un compozitor lucrează de obicei cu ele și anume pe un portativ. Expert Fux afișează atât cele două linii melodice ale contrapunctului cât și intervalele dintre acestea, tocmai pentru a ușura înțelegerea rezultatelor.

MIDI este un format cu multe aplicații în industria muzicală. Caracteristica sa cea mai importantă, faptul că toate informațiile de tempo, note, ritm nu se pierd, îl fac ideal pentru fișierele de intrare și ieșire. Dimensiunea redusă și suportul pe care majoritatea aplicațiilor muzicale îl oferă sunt două alte motive pentru folosirea formatului. Redarea și răspândirea contrapunctului generat nu poate fi mai simplă.

Compunerea de muzică a fost pentru o lungă perioadă de timp apanajul exclusiv al compozitorilor. Așa cum am văzut, apariția calculatoarelor nu a schimbat această realitate. Este greu să demonstrezi că o piesă generată de calculator, printr-un algoritm, este echivalentă unei sonate sau simfonii compuse de un om. Dar poate nici nu trebuie. Compoziția este, la bază, o activitate ca toate celelalte. Cere un efort considerabil, cu multe greșeli, refaceri și în general muncă la toate nivelele. Calculatoarele nu greșesc și pot să facă foarte multe lucruri într-un timp foarte scurt. Familiarizarea compozitorilor cu acestea și folosirea lor, ca ajutoare de încredere, nu poate să reprezinte decât viitorul.

Referințe

<http://code.google.com/p/abc4j/>.

Acevedo, Andres Garay. „Fugue Composition with Counterpoint Melody Generation Using Genetic Algorithms.” 2004.

Belkin, Alan. *Principles of Counterpoint*. 2000.

Burton, Anthony R., Tanya Vladimirova. „Generation of Musical Sequences with Genetic Techniques.” *Computer Music Journal*, 1999: 59–73.

Cope, David. „An Expert System for Computer-assisted Composition.” *Computer Music Journal*, 1987.

Dicționarul de termeni muzicali. Editura Enciclopedică, 2010.

Dion, John A. „Automated Music Composition: An Expert Systems Approach.” 2006.

- Ebcioğlu, Kemal. „An Expert System For Harmonizing Chorales In The Style Of J.S.Bach.” *The Journal of Logic Programming*, 1990.
- Gonzato, Guido. *Making Music with ABC PLUS*. 2003.
- Henz, Martin, Stefan Lauer, Detlev Zimmermann. „COMPOzE — Intention-based Music Composition through Constraint Programming.” 1996.
- Hiller, Lejaren A., Leonard M. Issacson. *Experimental Music: Composition with an Electronic Computer*. McGraw-Hill, 1959.
- Hitson, C.H. *Counterpoint for Beginners*. London: Oxford University Press, 1962.
- Järveläinen, Hanna. „Algorithmic Musical Composition.” 2000.
- Johanson, Brad, Riccardo Poli. „GP-Music: An Interactive Genetic Programming System for Music Generation with Automated Fitness Raters.” 1998.
- Kennan, Kent. *Counterpoint Based on Eighteenth-Century Practice*. New Jersey: Prentice Hall, 1999.
- Loy, D. Gareth. „Connectionism and Musiconomy.” În *Music and Connectionism*, de Peter M. Todd, D. Gareth Loy. MIT Press, 1991.
- Mann, Alfred. *The Study of Counterpoint from Johann Joseph Fux's Gradus ad Parnassum*. London: W.W. Norton & Company, 1971.
- MIDI File Format*. <http://www.sonicspot.com/guide/midifiles.html>.
- Miller, Michael. *The Complete Idiot's Guide to Music Theory*. Penguin Group, 2005.
- Mozer, Michael C. „Neural network music composition by prediction: Exploring the benefits of psychoacoustic constraints and multiscale processing.” *Connection Science*, 1994.
- Nordon, Hugo. *Fundamental Counterpoint*. Boston: Crescendo Publishing Company, 1969.
- Ovans, Russell, Rod Davidson. „An Interactive Constraint-Based Expert Assistant For Music Composition.” 1992.
- Roads, Curtis. În *The Computer Music Tutorial*, de Curtis Roads. 1995: MIT Press, 1996.
- Schottstaedt, Bill. „Automatic Species Counterpoint.” 1984.
- Zeitlin, Poldi, David Golberger. *Understanding Music Theory*. Omnibus Press, 2001.