

BRUNO ABRANTES BASSETO

**UM SISTEMA DE COMPOSIÇÃO MUSICAL AUTOMATIZADA, BASEADO
EM GRAMÁTICAS SENSÍVEIS AO CONTEXTO, IMPLEMENTADO COM
FORMALISMOS ADAPTATIVOS.**

**Dissertação apresentada à Escola
Politécnica da Universidade de São
Paulo para a obtenção do título de
Mestre em Engenharia**

**São Paulo
2000**

BRUNO ABRANTES BASSETO

**UM SISTEMA DE COMPOSIÇÃO MUSICAL AUTOMATIZADA, BASEADO
EM GRAMÁTICAS SENSÍVEIS AO CONTEXTO, IMPLEMENTADO COM
FORMALISMOS ADAPTATIVOS.**

**Dissertação apresentada à Escola
Politécnica da Universidade de São
Paulo para a obtenção do título de
Mestre em Engenharia**

**Área de Concentração:
Computação e Sistemas Digitais**

**Orientador:
João José Neto**

**São Paulo
2000**

Basseto, Bruno Abrantes

Um Sistema de Composição Musical Automatizada, baseado em Gramáticas Sensíveis ao Contexto, Implementado com Formalismos Adaptativos. São Paulo, 2000.

143p.

Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Computação e Sistemas Digitais.

1. Composição Musical Automatizada 2. Linguagens 3. Formalismos Adaptativos – Aplicação. I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais II. t

*para Paula
e para meus Pais*

Agradecimentos

Prof. Dr. *João José Neto*
Prof. Dr. *Bruno Fregni Basseto*, da
Faculdade de Filosofia, Letras e
Ciências Humanas da USP, meu pai;
Prof. Dr. *Geber Ramalho* da UFPE;
Prof. Dr. *Benício José de Souza*, Prof.
Dr. *Paulo Sérgio Muniz Silva*, Prof^a. Dr^a.
Edith Ranzini, Prof. Dr. *Jaime Simão*
Sichman, Prof^a. Dr^a. *Maria Alice Grigas*
Varella Ferreira, da Escola Politécnica
da USP;
Prof. Dr. *Jair Minoru Abe*, do Instituto
de Estudos Avançados da USP;
Prof^a. Dr^a. *Maria Regina Godelli*, *Mônica*
Geraldi Valentim e *Lairtes Vidal*, do
Departamento de Psicologia
Experimental da USP;
Ruggero Ruschioni, da Escola de
Comunicações e Artes da USP;
Neusa Yoscimoto, do Serviço de
Bibliotecas da USP;
Elói Fritsch e *Roger Hoeffel*, da UFRGS.

Sumário

Sumário	i
Resumo	iii
Abstract	iv
1. Introdução e Motivação	1
1.1 Música	1
1.1.1 Organização Interna	2
1.1.2 O conceito de linguagem	4
1.1.3 A memória musical	6
1.2 A Música Contemporânea	8
1.3 O Computador e a Música	13
1.4 Este Trabalho	19
2. Representação Musical em Computadores	20
2.1 Representação musical tradicional	20
2.2 Representações computacionais	24
2.2.1 Representação em lista de eventos	25
2.2.2 Representação procedimental	28
2.2.3 Representação estruturada	31
2.2.4 Representações semânticas	34
3. Sistemas de Composição Algorítmica	39
3.1 Composição Musical	40
3.2 Compositores determinísticos	43
3.2.1 Gramáticas e sistemas de produção	45
3.3 Compositores estocásticos ou não-determinísticos	50
3.3.1 Redes de Markov	51
3.4 Alguns outros tipos de compositores automáticos	56
3.4.1 Compositores por analogia	57
3.4.2 Compositores baseados em exemplos	58
3.4.3 Compositores iterativos	60
3.5. Conclusões	61
4. Formalização Teórica	62
4.1 Linguagem	62
4.2 Gramática	62
4.3 Árvores de Derivação	64
4.4 Tipos de Linguagens	65
4.4.1 Gramáticas Livres de Contexto	65
4.4.2 Gramáticas Lineares	66
4.5 Reconhecedores	67
4.5.1 Autômatos Finitos	68
4.5.2 Autômatos de Pilha	68
4.6 Autômatos não-determinísticos	70

4.7 Redes de Markov	71
4.8 Autômatos Adaptativos	75
4.9 Máquinas de Markov Adaptativas	77
4.10 Sistemas de Markov Adaptativos	80
4.11 Conclusões	83
5. Apresentação da tecnologia empregada	85
5.1 Arquitetura do sistema	85
5.2 Representação dos dados	87
5.2.1 Representação de alturas	87
5.2.2 Representação de durações	88
5.3 Estrutura dos Sistemas de Markov Adaptativos	89
5.4 Descrição do Conhecimento Musical empregado	93
5.4.1 Linha melódica	93
5.4.2 Comportamento tonal	100
5.4.3 Comportamento temático	106
5.4.4 Interação entre linhas melódicas independentes	111
5.4.5 Interação de várias vozes	113
5.4.6. Controle sobre a estrutura musical	119
5.5 Conclusões	122
6. Discussão e Conclusões	123
6.1 Avaliação do trabalho e dos resultados obtidos	123
6.2 Aplicações do Sistema	126
6.3 Sistemas de Markov adaptativos como mecanismo para a solução de problemas complexos	130
6.4 Emulação Computacional da Criatividade	132
6.5 Música e Linguagem	134
6.6 Conclusões	136
7. Bibliografia	138

Resumo

Sistemas computacionais têm sido extensivamente empregados em Música, muito contribuindo para com uma maior sistematização e formalização do processo de composição musical. Este trabalho procura analisar alguns resultados obtidos na área da composição musical automatizada, verificando algumas limitações nos formalismos já existentes.

Considera-se que os sistemas não-determinísticos de composição musical são os mais interessantes, devido à natureza aleatória do processo de produção de material musical, a qual pode ser interpretada como uma emulação de um processo criativo.

Uma nova implementação é proposta, que, considerando o fenômeno musical como lingüístico, se propõe a modelar o conhecimento musical através de gramáticas. Verifica-se que as classes de linguagens regulares e livres de contexto são insuficientes para a representação do conhecimento musical necessário e que, portanto, devem ser empregadas gramáticas sensíveis ao contexto com essa finalidade.

Para propiciar a implementação computacional de gramáticas sensíveis ao contexto, empregadas como modelo, desenvolve-se um formalismo adaptativo, mediante uma variação dos modelos não-determinísticos em rede de Markov, tradicionalmente empregados em composição musical automatizada.

É ilustrada uma aplicação dos conceitos desenvolvidos, na resolução de alguns dos problemas clássicos da área da composição musical automatizada, comprovando a eficiência dos métodos propostos, apontando interessantes linhas de pesquisa futuras.

Abstract

Computer Systems are extensively employed in Music, making important contributions to a greater systemization and formalization of musical composition activity. The present work aim to analyze some results of the automated musical composition field, verifying some limitations on the existing formalisms.

We considered that non-deterministic composition systems are more interesting, due to the random nature of the music production scheme, what can be interpreted as machine-creativity emulation.

Considering the musical phenomenon as linguistic in its essence, a new implementation is proposed, modeling musical knowledge by means of grammars. Regular and context-free languages proved to be insufficient for describing such a knowledge, so context-sensitive grammars have to be used for this.

For the computer implementation of context-sensitive grammars, we developed an adaptive formalism, from a variation of non-deterministic models, based on Markov chains, which are usually employed in automated musical composition.

A particular application of those concepts is presented, in the solution of some classical problems of the automated musical composition domain, showing the proposed knowledge representation method efficiency, pointing to interesting future researches.

1. Introdução e Motivação

Na introdução deste trabalho, estamos apresentando algumas informações sobre seu escopo, principalmente considerando sua inerente interdisciplinaridade. Procuraremos apresentar alguns dos benefícios que uma pesquisa neste domínio pode trazer para o estado da arte.

1.1 Música

Nesta seção, procuramos desenvolver uma definição de Música que seja adequada aos propósitos do presente trabalho. A noção de Música é um tanto complexa e sujeita a diferentes interpretações; o conceito usualmente estende-se, tornando-se amplo e de difícil conceituação.

Inicialmente podemos situar a música como um fenômeno acústico; no entanto, isto é algo que corresponde a uma restrição, pois o fenômeno musical pode ser considerado de forma abstrata, sem existir necessariamente um correspondente sonoro. No entanto, é senso comum considerar a música como um fenômeno caracteristicamente auditivo, freqüentemente associado a elementos estéticos ou emocionais.

A música pode ser considerada como o fenômeno comunicativo de banda mais aberta conhecido pelo homem, valendo-se do conceito da teoria da informação. A quantidade de informação, que é comunicada em uma obra musical, é muito superior à conseguida pela linguagem falada, comparando-se apenas à transmissão de imagens [MOOG86].

No entanto, não é um sinal sonoro qualquer que é percebido como sendo música. Vamos discutir, em seguida, algumas de suas principais propriedades distintivas e demonstrar sua importância prática para o trabalho que está sendo desenvolvido. As características dos sinais musicais que mais nos interessam são, portanto (cf. [HILLER59]):

- Organização ou estruturação interna;
- Aplicação do conceito de linguagem;
- Importância da memória musical.

1.1.1 Organização Interna

Um princípio razoavelmente universal, aceito como elemento distintivo de sinais musicais, é a existência de uma ordenação ou organização interna que os caracterizam. Alguns definem o compositor como sendo aquele que organiza o caos de todos os elementos possíveis de existirem em uma composição musical. Alguns elementos são deliberadamente escolhidos em detrimento de outros, fato esse que está no cerne de nossa definição de música.

A origem do termo *compositor* está relacionada ao indivíduo que agrupa ou ordena elementos. Compor pode ser definido como “dispor lado a lado”, “ordenar”, o que nos leva naturalmente ao conceito de *organizar*. É interessante, inclusive, notar que os primeiros compositores de que se tem notícia na História da Música, não eram produtores de material musical inédito, mas sim organizadores desse material. Seu trabalho estava em escolher, dentre fragmentos ou modos disponíveis, os mais adequados a uma dada situação, e dispô-los em uma determinada seqüência conforme sua própria preferência ou conforme exigido pelas circunstâncias.

Freqüentemente, o valor estético ou funcional de uma obra musical está associado a sua ordem interna. A música é criada com base em um plano bem definido que visa a atingir algum objetivo.

Pode-se afirmar que um dos mais importantes níveis de organização musical que vamos encontrar, sendo a música uma arte essencialmente temporal, é a ordenação e disposição do som no tempo. Em uma primeira instância, tal organização se apresenta sob o que chamamos *ritmo*. Em níveis superiores de organização musical, temos o conceito de *forma*, referindo-se à maneira pela qual as estruturas musicais se comportam no decorrer da duração da obra.

Outros elementos organizadores, deliberadamente escolhidos pelas culturas, são aqueles que constituem as *escalas* musicais, dentre a infinita gama de alturas possíveis, como sendo os sons primordiais, com os quais

a música será, então, construída. Deve-se notar que a escolha desses elementos não segue apenas critérios físicos ou psicoacústicos, mas também aspectos fundamentalmente não universais, como fatores estéticos, culturais ou sociais [WISNIK85]. Em certas poéticas contemporâneas, substituem-se as escalas de sons de alturas definidas por sons abstratos, cuja característica fundamental é o *timbre*. De qualquer forma, essa é uma escolha arbitrária, análoga, por conseguinte, à definição de escalas.

A natureza da organização e estruturação musicais geralmente recai sobre aspectos estéticos, como simetria e similaridade. Muitos destes aspectos referem-se diretamente a fatores psicológicos, conforme observado por Wertheimer, fazendo a importante analogia entre eventos sonoros, considerados como um agrupamento em uma obra musical, com os agrupamentos visuais de figuras, na tradição psicológica da Gestalt [LEACH95] [LERDAHL83].

O fator mais complexo para a análise científica do fenômeno musical não se encontra propriamente na estruturação, mas sim na coexistência entre organização e caos. Da mesma forma que a um sinal de ruído dificilmente podem ser atribuídas propriedades musicais, uma obra musical, estruturada e organizada com perfeição matemática, pode ser considerada como destituída de qualquer valor estético. Tal paradoxo é debate constante entre especialistas; as obras mais brilhantes são aquelas em que há a combinação destes opostos, de forma a despertar no ouvinte impressões estéticas, ao mesmo tempo em que cria ansiedade e surpresa. Essa é uma relação freqüente entre sensibilidade e percepção estética.

A existência de uma estruturação interna resulta em uma redundância no sinal musical, limitando as seqüências possíveis de eventos sonoros com relação ao nível de diferença entre eles. Os estudos de Clarke e Voss, por exemplo, mostraram que a distribuição do grau de variação das alturas - tomada como uma relação em freqüência - aproxima-se da hipérbole $1/f$ (cf. [SMOLIAR91]). Tal estudo estatístico,

efetuado sobre os *Concertos de Brandenburgo* de J. S. Bach, despertou interesse pela análise musical com métodos estatísticos, baseados na teoria da informação. No entanto, foram precisos poucos experimentos para comprovar que nem todo sinal de espectro $1/f$ pode ser considerado como musical.

1.1.2 O conceito de linguagem

Pode-se suspeitar, devido à enorme amplitude de possibilidades de organização (banda larga), que a música corresponda a um fenômeno contínuo, porém a música constitui-se de uma seqüência temporal de eventos, sendo, portanto, discretizada no tempo. A existência de tal discretização e a possibilidade de discriminação dos eventos individuais nos permitem associar um conjunto de *símbolos* aos seus elementos constituintes, o que nos remete imediatamente à noção de *linguagem*, em um processo que nos parece consistente, uma vez que é possível a tradução de uma obra musical - pelo menos em parte - para uma outra linguagem, qual seja, a *partitura*.

Note-se que essa aproximação entre os conceitos de linguagem e de música nos permite supor que essa última possa ser adequadamente representada e manipulada por sistemas computacionais, o que é um de nossos objetivos.

A noção de música enquanto fenômeno lingüístico, já foi apontada na literatura [ROBINSON97][SCHURMANN90] e é também combatida por diversos teóricos (apud [CANDÉ78]). A dificuldade levantada muitas vezes encontra-se na associação natural entre o conceito de linguagem e um princípio comunicativo, ou seja, a idéia de uma *mensagem* emitida por um transmissor para um receptor. Muitos teóricos envidam esforços em negar a existência de tal mensagem, definindo, muitas vezes, a música como “uma linguagem que significa a si mesma” [CANDÉ78], ou seja, sem conotações exteriores ao fenômeno musical propriamente dito, estando ausentes, portanto, semânticas extra-musicais.

A despeito dessa discussão quanto a possíveis elementos extramusicais, o uso do conceito de linguagem como arcabouço formal para a descrição da organização interna do material musical é essencial para o nosso trabalho. Já foi sustentada a idéia de que a mente humana trabalha de forma lingüística, e tal tratamento não é diferente em relação à música. Mesmo um estudo não muito aprofundado da teoria musical tradicional pode revelar que esta possui características estruturais claramente hierarquizadas e associadas a contextos específicos [NARMOUR99], o que pode nos apontar um parentesco com as linguagens sensíveis ao contexto segundo o conceito de Chomsky [SMAILL93].

Contudo, há algumas particularidades que consideramos importantes de serem apresentadas como dificuldades a esse tipo de avaliação da linguagem musical. Considerando a linguagem gráfica própria para sua representação, ou seja, a partitura, observa-se que esse formato de comunicação (esse sim, não pode ser considerado como não-comunicativo) possui uma série de particularidades que estão relacionadas à natureza da música. Por exemplo, a representação musical tradicional não corresponde exatamente a uma tradução inequívoca do material musical de origem. Na maioria dos casos ela pode ser considerada uma representação abstrata de elementos musicais, conforme concebidos pelo compositor - mas ela não é a obra musical - sendo necessária a participação de outro especialista, o intérprete, para a sua execução final.

Dessa forma, alguns elementos não estão formalizados em sua descrição gráfica, sendo necessários conhecimentos adicionais, e, conforme apontado por vários autores, a cognição ou o aprendizado de outras obras musicais, segundo alguma forma de tradição (conforme [FIELD97], por exemplo). Este tipo de limitação, com freqüência obrigou compositores contemporâneos a adotarem formas alternativas para representação de suas composições.

A natureza dessas dificuldades leva muitos teóricos à suposição de que determinados aspectos da música não se encontram adequadamente

formalizados na notação musical tradicional, aspectos esses que possivelmente não correspondem a elementos do domínio lingüístico e, portanto, não são representáveis simbolicamente. Apesar destas opiniões, muitas formas alternativas de representação existem hoje, inúmeras delas computacionais [FIELD97].

Algumas representações, entre elas a notação musical tradicional, são essencialmente práticas, prontas para a execução, não evidenciando diretamente a estruturação musical última, que só pode ser obtida através de uma análise desse material por especialistas. São, portanto, representações de um nível semântico bastante baixo.

Várias formas de gramáticas e lógicas não-clássicas foram propostas para uma maior formalização, algumas com resultados interessantes (vide [BLEVIS89][ROADS87]). Lerdahl e Jackendorf afirmam que a linguagem musical não pode ser gerada por gramáticas semelhantes às que produzem a linguagem natural, pois essas não modelam uma série de características que são únicas da música [LERDAHL83]. Eles apontam, em seu trabalho, uma formalização gramatical alternativa, baseada em quatro grandes grupos de cláusulas gramaticais.

1.1.3 A memória musical

A memória tem um papel fundamental na percepção e cognição musical. Observando que a música é uma seqüência temporal de eventos, apenas uma associação a elementos passados pode assegurar alguma compreensão, ou seja, apelando à lembrança de eventos anteriores.

Assim, é freqüente a analogia entre o decurso de uma obra musical e a narrativa verbal. Da mesma forma que, na narrativa, elementos são apresentados segundo uma ordem lógica para assegurar a compreensão do ouvinte visando a uma conclusão, o discurso musical apresenta uma seqüência de eventos, de forma recorrente ou redundante, para que a obra tenha sentido; tal seqüência de eventos permite levar o ouvinte ao análogo da conclusão no discurso usual. Todos esses conceitos estão intimamente

relacionados a os de tempo e de ordem. Essas variáveis apenas podem ser combinadas através da memória.

Todos os conceitos da teoria musical estão de alguma forma relacionados com a memória. A idéia de ritmo, por exemplo, sustenta-se sobre a pulsação, que só pode ser percebida pela lembrança de eventos anteriores e pela suposição do momento de ocorrência do evento futuro. A diferença entre o esperado (previsto através da memória do passado) e o evento real acaba por constituir a percepção do ritmo. O sistema melódico também só pode ser concebido pela sucessão de intervalos musicais diferentes, os quais são, evidentemente, uma comparação entre as alturas de eventos musicais sucessivos.

A memória musical não tem apenas um aspecto local, interno à obra musical em questão, mas é freqüente remontar a elementos externos a ela própria. O mecanismo da *tonalidade*, por exemplo, é um claro apelo à memória musical. Conforme descrito por Schoenberg [SCHOENBERG74], ao ser ouvido um som de uma determinada altura, o ouvido automaticamente dá a esse som uma atribuição de repouso, ou tônica. Mas quando outro som aparece, uma série de relações entre os dois são supostas pelo ouvinte; em particular essas relações podem ser de concordância ou discordância, e, no caso desta última, há a tendência de levar o ouvinte para longe da tônica. Mas é apenas a memória dessa que faz com que o mecanismo de retorno a ela tenha sentido para o ouvinte.

Heinrich Schenker estende o conceito de relação tonal à totalidade da obra, esta composta por seções harmônicas bem definidas, sendo que, em sua totalidade, as relações harmônicas com a tônica - lembrada - mantêm-se [LERDAHL83]. A memória musical cumpre papel fundamental na especificação da tonalidade, sendo que esta é um fenômeno claramente hierárquico no conceito de Schenker [NARMOUR99].

A percepção e a memória de agrupamentos musicais em um sentido mais amplo que o da percepção imediata do evento corrente permitem a identificação de temas ou motivos, os quais são usados para caracterizar

seções ou subdivisões de uma estrutura musical qualquer. Esse procedimento permanece válido mesmo para a música atonal, na qual as relações de tonalidade são rompidas. De fato, na música serial, busca-se quebrar a relação de memória tonal, no entanto, o conceito de forma é levado às últimas conseqüências na *série*, que está sempre presente, memorizada consciente ou inconscientemente.

Analisando tais fatores, ainda considerando a música como um fenômeno lingüístico, podemos desenvolver a idéia de que a necessidade pelo uso da memória para a cognição do fenômeno musical implica necessariamente uma relação de contexto, uma vez que a compreensão dos eventos musicais presentes depende de uma relação de memória, possivelmente complexa, de eventos passados, e é freqüente também a necessidade por novos eventos, ocorrendo no futuro, para a compreensão do evento musical presente. Dessa forma, podemos levantar aqui a hipótese de que a representação lingüística da música deve ser, pelo menos, sensível ao contexto, conforme discutiremos mais adiante.

1.2 A Música Contemporânea

É oportuno situar o leitor no contexto da música contemporânea, um pouco de seu histórico e de suas necessidades atuais. Uma observação desse panorama pode demonstrar a importância da utilização de sistemas computacionais nesse meio. Nesta seção, podemos fazer referência a [GRIFFITHS94], [GRIFFITHS95] e [SIMMS86].

A linha evolutiva, que conduz a tradição musical erudita ocidental até nossos dias, passa por uma série de marcos importantes, que caracterizam grandes linhas de pensamento em relação à música e sua produção desde o estabelecimento do som de altura definida, aceito então como musical, provavelmente produzido pela voz humana, considerada como a primeira forma de manifestação musical.

O segundo marco histórico foi o surgimento da linha melódica [CANDÉ78]; o estabelecimento de linhas melódicas está em sua origem

associado à eleição de um conjunto fixo de sons preferenciais e à caracterização desse conjunto: são os *modos*. Cada *modo* possui seu próprio conjunto de sons e formas melódicas características, associado sempre a um dado *ethos* musical.

A evolução histórica, que se seguiu, passou à aglutinação de melodias soando simultaneamente, o que se chama *polifonia*. A polifonia é um marco distintivo da música ocidental [CANDÉ78], inexistente nas demais culturas. O desenvolvimento da polifonia resultou na preferência por determinados grupos de sons e na eleição de dois modos preferenciais: o modo maior e o modo menor.

Essa evolução conduz à *tonalidade*, onde o estabelecimento de *graus* e de *funções* em detrimento do modo e seus intervalos, assume a mais alta importância. A partir desse momento, o conceito de tonalidade é estendido de forma a abranger a totalidade da obra em suas seções, adquirindo máxima importância estrutural-hierárquica, com o conceito de *modulação*. Com o advento do Romantismo, o mecanismo tonal foi tendo seus limites alargados, muitas vezes por razões simplesmente expressivas, com modulações e tratamentos inesperados da tonalidade até a coexistência simultânea de mais de uma tonalidade [SIMMS86].

O fim do Romantismo é marcado por um sério conflito no mundo tonal, que teve seus fundamentos teóricos tão estendidos, que se encontrava dividido em diversas escolas com teorias antagônicas. A importância desse momento histórico está no alargamento dos horizontes dos artistas, os quais, a partir da crise do sistema tonal, passaram a estudar outras possibilidades de expressão musical [WISNIK85].

A abolição completa da tonalidade já havia sido sugerida, mas só pôde ser consumada formalmente através do trabalho de Arnold Schoenberg, que lançou o marco da *Música Serial* [GRIFFITHS94], onde a tonalidade era substituída pelo esquema mais genérico da *série*. Esse sistema é muito mais um arcabouço formal - quase algébrico - do que um estilo musical, o que atraiu muitas correntes associadas ao empirismo

científico da época. Esse processo conduz às correntes estéticas baseadas no *Expressionismo*.

A busca de novos meios expressivos inclui a decadência da tonalidade, mas não apenas isso. Teóricos ainda relacionados à doutrina tonal não descartaram a experimentação, em busca de expressão, freqüentemente através do interesse pela *sonoridade*, não apenas nas gigantescas e surpreendentes massas sonoras de Mahler e Strauss, mas também nas experiências timbrísticas e novas formas de colorido orquestral, que acabariam conduzindo à supervalorização do som, no Impressionismo de Debussy e mais adiante, na Música Concreta de Schaeffer.

A descoberta de que o mundo não é apenas ocidente também ocorre nessa época. O ouvido europeu, cansado de suas formas cadenciais já tão repetidas, volta-se ao som proveniente de outros povos e culturas, exóticos para a música ocidental. Novos instrumentos, e até novas escalas de sons serão, a partir de então, adicionados ao caldeirão da música contemporânea. A redescoberta do mundo modal, e mesmo da formação das escalas de sons primordiais, marca várias correntes estéticas do nosso século [WISNIK85].

A partir disto, foi possível negar inclusive a validade, suposta universal e indiscutível, do sistema de alturas fixo desde Guido d'Arezzo e apenas revisado no sistema de igual temperamento. O rompimento do conceito de "nota musical" é sugerido através do uso de escalas alternativas, intervalos menores do que o semitom, comuns a culturas não ocidentais, ou ainda no uso de timbres de espectros não harmônicos. Tal avanço é levado a suas últimas conseqüências na *Música Microtonal*, onde qualquer freqüência sonora é admissível.

Outras correntes estéticas procuraram romper com a tradição da forma e ordenação musicais, como por exemplo a *Música Aleatória*, de Cage, que explora, quase que exclusivamente, elementos extra-musicais, chegando, por exemplo, à *Música Ambiental*, onde a música é "composta"

no momento de sua interpretação. Várias correntes de vanguarda coexistem neste meio, mas a noção comum a todas é o alargamento do conceito de música em todas as direções possíveis.

O emprego de meios eletrônicos naturalmente não é estranho à grande maioria dessas correntes. O surgimento de dispositivos capazes de gravação de som provocou grande interesse por suas possibilidades, inicialmente utilizados para trazer, para a sala de concertos, sons que não poderiam ser produzidos pelos instrumentos convencionais, como trovões, vento ou canto de pássaros, e, mais tarde, no seu uso como manipulação dos sons gravados. A manipulação de sons gravados em fitas magnéticas e posteriormente reproduzidos por meios eletrônicos é o método básico de composição adotado por Pierre Schaeffer, idealizador da *Música Concreta*, por ele definida como o trabalho sobre os sons que existem na natureza [MENEZES96].

A síntese de som por dispositivos eletrônicos já havia sido sugerida e diversos “instrumentos musicais” eletrônicos foram construídos, como o *Ondes Martenot* e o *Thèrèmin*, para os quais diversas composições foram realizadas, como *Turangalîla* de Olivier Messiaen. A proposta de uma música produzida exclusivamente por meios eletrônicos, com a possível exclusão dos instrumentos musicais tradicionais foi proposta por Karheinz Stockhausen, da escola de Colônia, em sua *Música Eletrônica*. Mais tarde, poéticas de vanguarda como essas seriam reunidas em uma corrente denominada *Música Eletroacústica* [RUSCHIONI93].

Desse processo podemos acompanhar algumas deficiências e dificuldades que nos permitem introduzir o computador neste contexto. Num primeiro momento, o controle sobre as possibilidades expressivas dos meios musicais empregados é bastante complexo. Articulações e variações mínimas de dinâmica, requeridas pelas composições modernas, dificilmente podem ser esperadas mesmo dos melhores e mais experientes intérpretes. Muitas vezes a transcrição direta das idéias do compositor no

resultado sonoro, conforme produzido pelo intérprete, é praticamente impossível.

A especificação do anseio do compositor muitas vezes não pode ser sequer representada sob a forma tradicional da partitura, sendo que diversos compositores contemporâneos adotam formas alternativas de representação, com a conseqüência óbvia de perda da universalidade da interpretação.

Com a utilização de sistemas computacionais, não apenas a representação inequívoca das idéias do compositor pode ser efetuada em seus mínimos detalhes, como a sua condução, através do controle absolutamente preciso em todos os parâmetros e variáveis envolvidas, com uma resolução muito superior ao que seria possível para um intérprete humano [CHAMBERLIN85]. No caso de sistemas musicais com interação de intérprete humano, dispositivos sensores podem obter do intérprete uma vasta quantidade de informação sobre a execução da obra, de forma muito superior a instrumentos tradicionais [POLANSKY94].

Outro ponto que pode ser apresentado é o fato de que a música contemporânea, em seus aspectos teóricos, pode chegar a uma complexidade excessiva, muitas vezes intratável para o próprio compositor. O uso de ferramentas computacionais pode beneficiar a composição e formalização de obras musicais de muitas maneiras [LASKE89].

A capacidade de síntese de som por meios digitais é praticamente ilimitada. Teoricamente qualquer som pode ser produzido através de uma seqüência numérica enviada a um conversor digital-analógico e a um amplificador [MATHEWS69]. O compositor pode criar o timbre apropriado para sua composição, o que é um dos principais objetivos de diversas poéticas contemporâneas. Os meios eletrônicos não estão sujeitos ao estabelecimento de um conjunto fixo ou finito de alturas possíveis e tampouco à produção de sons com espectros harmônicos, como os instrumentos musicais tradicionais, que produzem som por meios

mecânicos, o que torna o uso do computador praticamente indispensável na produção de música microtonal.

A composição por computador ou composição algorítmica é possível em certo grau, obtendo-se resultados interessantes e musicalmente válidos. O compositor da obra pode fornecer ao computador as diretrizes básicas da peça e este pode produzir a música de forma autônoma ou guiado pelo próprio compositor, por intérprete ou intérpretes, ou ainda por quaisquer outros fatores. A criação de música ambiental, produzida no momento, como resposta a determinadas condições do ambiente, é facilmente realizável com o auxílio de sistemas computacionais. O computador pode prestar-se à produção de música aleatória de forma realmente eficiente.

O computador é capaz de reter informações de forma organizada e, portanto, pode ser empregado como poderosa base de dados para o compositor. A coleta de material e sua posterior organização e recuperação pode ser feita de forma bastante eficiente e produtiva [ROADS85].

Finalmente, o uso do computador como arcabouço de produção musical ou como elemento auxiliar na análise de material musical pode colaborar imensamente para a compreensão e o estabelecimento de teorias musicais e estéticas, bem como para o melhor entendimento do processo cognitivo da música pelos seres humanos. Além disso, o uso do computador como recurso didático permite inusitadas possibilidades de ensino, tanto da teoria quanto da prática musical.

1.3 O Computador e a Música

Está bastante fora do escopo deste trabalho apresentar um histórico detalhado da evolução dos sistemas de Computação e Música, principalmente devido à aplicação excessivamente ampla do computador em tarefas demasiadamente diferentes nesta área. Estamos interessados aqui em apresentar alguns marcos que consideramos importantes para o

estabelecimento de uma linha de raciocínio, que conduz nosso trabalho. Pode-se obter maiores informações através de [ROADS96], por exemplo.

Para muitos autores a primeira aplicação de sistemas computacionais no universo musical ocorre na década de 50, quando o engenheiro químico Lejaren Hiller passa a utilizar o Illiac, da universidade de Illinois, na composição de obras musicais utilizando métodos algorítmicos. A composição produzida segundo processos estocásticos, modelados em redes de Markov, foi intitulada *Illiatic Suite*, apresentada em 1957 por um quarteto de cordas.

Para atingir seu objetivo, Hiller observou uma série de propriedades do som musical, as quais procurou modelar computacionalmente. O cerne da produção “criativa” do evento musical é a geração de elementos pseudo-aleatórios, aos quais Hiller impõe uma série de restrições, como um processo de escolha (e não apenas sorteio) com base em um conjunto de regras, conjunto esse que era diferente para cada um dos quatro movimentos da Suite. Introduz-se no processo aleatório de escolha determinado grau de redundância através de um modelo em rede de Markov [HILLER59]. Voltaremos nossa atenção mais adiante neste trabalho para tais idéias.

O interesse pela produção aleatória de música encontrava-se bastante em voga na época, sobretudo com os trabalhos pioneiros de John Cage, que procuravam abandonar qualquer tipo de previsibilidade nas obras musicais compostas. Cage trabalhou com Hiller na produção de música estocástica, mas a diferença fundamental do sistema proposto em *Illiatic Suite* era justamente a existência do conjunto de restrições impostas à aleatoriedade no conjunto de regras de escolha. A poética decorrente deste estilo de modelamento geralmente é chamada de *Música Experimental*.

Sistemas de composição algorítmica semelhantes aos de Hiller foram utilizados pelo compositor grego Iannis Xenakis, em uma poética por ele definida como *Música Estocástica*, em que princípios aleatórios definidos

estatisticamente são aplicados na produção da obra musical. Xenakis empregava expressões matemáticas de distribuições de probabilidade para limitar o processo randômico e utilizou o computador como um auxiliar em seus cálculos, como na obra *Eonta*, para piano e quinteto de metais, publicada em 1964 [GRIFFITHS94].

A diferença fundamental entre as duas abordagens está em quem efetivamente toma a decisão estética ou “criativa” do material musical: enquanto Hiller atribui toda a responsabilidade ao computador, Xenakis acompanha o processo, tomando ele próprio as decisões composicionais oportunas. De certa forma, ele trata o computador como um auxiliar de cálculo, sendo o precursor dos sistemas de composição auxiliada por computador. Mais tarde, G. Koenig criará os programas PROJECT I e PROJECT II para auxiliá-lo em suas composições, a partir do fornecimento de regras de composição aos mesmos e da avaliação, pelo compositor, dos resultados obtidos [LASKE89].

Algumas observações importantes podem ser feitas a partir dessas pesquisas. Embora o modelo básico de produção musical seja o pseudo-randômico, utilizado na época para o modelamento matemático de diversos processos (lembrando inclusive que Hiller era engenheiro químico), uma série de características específicas da Música comparecem no processo de composição algorítmica, de forma a imitar o processo criativo de um compositor humano, como sendo um processo de *escolha*. Os resultados obtidos procuraram demonstrar a validade da música composta por meios computacionais e o estudo de suas possibilidades estéticas.

A interface dos sistemas de composição algorítmica era a mesma de um programa de computador qualquer, com a entrada de um programa codificado em alguma linguagem (FORTRAN, por exemplo), e o resultado do processamento uma listagem de símbolos, que deveria ser analisada pelo usuário para sua devida tradução para a forma tradicional de notação musical, que seria posteriormente fornecida aos intérpretes da obra.

A limitação entre composição e interpretação foi quebrada no final da década de 60 e início da década de 70 com o trabalho de Max V. Matthews da Bell Labs. Matthews, interessado no processamento digital de sinais, vislumbrando a enorme amplitude de potencialidades do uso da amostragem digital de sinais, sobretudo sonoros, sugeriu sua aplicação à música ao criar a série de programas MUSIC, dos quais o mais famoso foi o MUSIC V, criado em 1969.

Segundo Matthews, qualquer sinal pode ser obtido de uma seqüência numérica, desde que mantidas certas relações espectrais do sinal desejado, em relação à freqüência de amostragem (teorema de Nyquist). Assim, a forma mais genérica de composição musical seria escrever tal seqüência numérica arbitrariamente. Uma vez enviada a um conversor digital-analógico e convertida em variações de pressão do ar por um alto-falante, qualquer som imaginável pelo compositor poderia ser produzido [MATHEWS69].

A criação manual dessa seqüência de amostras de audio naturalmente estaria fora das possibilidades de qualquer compositor humano, e neste ponto entraria a ferramenta computacional no cálculo do som produzido. Surgia o primeiro sintetizador de som por meios exclusivamente digitais.

O programa de Mathews recebia como entrada dois arquivos independentes, um dos quais descrevia a *orquestra*, um conjunto de algoritmos derivados dos procedimentos do processamento digital de sinais, que seriam responsáveis pela especificação dos timbres, empregados na produção dos eventos sonoros que vinham especificados no outro arquivo, a *partitura*. A execução do programa consistia em algumas etapas, que iam da ordenação dos eventos e criação de tabelas internas de formas de onda à execução de operadores como osciladores, filtros e somadores, para cada evento desejado pelo usuário, escrevendo os resultados dessa computação em um rolo de fita magnética. Após a conclusão desse processo, a fita magnética, contendo as amostras de

audio obtidas, poderia ser executada, enviando esses valores para o conversor digital-analógico e ao sistema de audio subsequente.

O próprio autor aponta as inúmeras vantagens do sistema e suas possibilidades imensas de produção de música. Uma vez que o sistema poderia ser programado de forma flexível, um número enorme de timbres ou efeitos poderia ser programado; a precisão na especificação dos parâmetros na *partitura* era praticamente contínua, com precisão em milissegundos, frequência em décimos de Hertz e dinâmicas em valores contínuos de dB. Isso tudo aliado à polifonia virtualmente infinita oferecida pelo sistema.

Embora o resultado da execução da obra ocorresse *off-line*, esse resultado já era propriamente sonoro, sem a necessidade de tradução da saída do programa em alguma linguagem acessível ao intérprete e prescindindo, inclusive, do próprio intérprete. O sistema de Matthews estava orientado à síntese de som e não especificamente à composição algorítmica; no entanto, o cruzamento de ambas as abordagens é possível. O programa MUSIC V permitia a especificação de procedimentos do usuário, em FORTRAN, que poderiam ser utilizados para a manipulação do material fornecido, podendo, por exemplo, tornar a execução da obra não-determinística.

Diversos compositores adaptaram as idéias de Matthews para usufruir das inúmeras possibilidades de controle do som. O sonho de Edgard Varèse, vivamente impressionado pelas possibilidades musicais oferecidas pelos meios eletrônicos de produção e manipulação de sons, mas decepcionado pela qualidade dos resultados obtidos pela tecnologia disponível na época, pôde, com a tecnologia digital, ser efetivamente levado a termo. Criar um novo timbre, transformá-lo em outro, fazê-lo mover-se no espaço, controlar com precisão todos os seus parâmetros seria finalmente possível com o uso de computadores. Grande número de obras executadas totalmente por meios digitais existem no repertório

contemporâneo por programas descendentes do MUSIC, como na obra de Jean-Claude Risset [MENEZES96].

Os programas de Matthews introduzem também alguns conceitos importantes, como por exemplo a abstração dos dados sobre eventos musicais, informados ao computador agora como arquivos independentes da linguagem específica da máquina, o que abre caminho para a interpretação simbólica do material musical.

O próximo marco que devemos citar acontece na década de 70, quando Robert Moog emprega microprocessadores no controle de sintetizadores analógicos [ROADS96]. Esta nova situação marca não apenas uma popularização dos sistemas computacionais aplicados à música, mas principalmente uma nova posição do computador em relação à execução musical: *tempo real*. De fato, o papel do computador não estava, nessas aplicações, voltado à composição de música ou à síntese de som, mas sim no controle sobre dispositivos analógicos que o estariam produzindo, agindo de acordo com a atitude, monitorada pelo processador, do intérprete do instrumento.

A capacidade do computador de monitorar mínimas variações de grandezas analógicas, através de conversores analógico-digitais e taxas suficientemente altas de aquisição, poderia ser utilizada como meio expressivo nunca antes imaginado. O computador poderia interagir com dispositivos mecânicos, semelhantes aos instrumentos musicais tradicionais (órgãos elétricos, por exemplo), mas não necessariamente. A enorme possibilidade de controle expressivo, obtido com o uso de computadores, foi apontada e utilizada extensamente por Boulez, por exemplo, na composição *Répons* [GRIFFITHS94].

Alguns anos mais tarde a velocidade dos processadores já era suficiente para dispensar os dispositivos analógicos, e os sintetizadores digitais puderam ser produzidos, trazendo consigo características próprias dos computadores, como memória (*sequencers*, capazes de armazenar e reproduzir seqüências de eventos), execução de algoritmos ou programas e

comunicação entre máquinas (o protocolo MIDI surge no início da década de 80) [MOOG86].

Consideramos o principal marco tecnológico da década de 80, no campo da música por computador, a aplicação de conceitos da *inteligência artificial*. Os estudos de Curtis Roads [ROADS79], por exemplo, apresentam novos formalismos para a descrição musical, baseados em gramáticas (uma abordagem lingüística da música, portanto) e em modelos baseados em regras de produção e redes de restrições. O uso de sistemas especialistas voltados à música também é contemporâneo a esses desenvolvimentos. A pesquisa volta-se, agora, a aspectos caracteristicamente musicológicos.

Esse tipo de desenvolvimento conduz ao uso de *modelos* musicais. Quanto mais esses modelos se aproximam daqueles empregados pelos músicos e compositores, melhores resultados podem ser obtidos tanto na síntese de música como em sua análise, importante para o estudo dos fenômenos cognitivos da música. Chega-se cada vez mais perto dos níveis semânticos do fenômeno musical.

1.4 Este Trabalho

O restante deste trabalho apresenta uma aplicação desenvolvida neste meio, com os objetivos principais de representação computacional de música, representação computacional de conhecimento musical na forma de gramáticas e a composição musical automatizada, em tempo real, pelo computador. Os primeiros capítulos apresentam uma visão geral das tecnologias que são usualmente empregadas, conforme a literatura referenciada. A seguir são apresentadas contribuições específicas deste trabalho, que o capítulo de conclusão procura destacar e relacionar com outras pesquisas na área da Computação e Música.

2. Representação Musical em Computadores

Este capítulo trata de aspectos relacionados à representação do material musical em sistemas computacionais, apresentando alguns casos particulares de representação, bem como exemplos extraídos da literatura. Considera-se particularmente importante a questão da representação, uma vez que ela vai interferir na forma como o material é tratado para sua execução, bem como na possibilidade de definição de operações específicas sobre este material, considerado ou não como um novo tipo de dados que a máquina é capaz de manipular.

Três aspectos são fundamentais na consideração da forma como o material musical é representado em estruturas computacionais: facilidade de trabalho com esse material pela máquina, forma de transformação do material em som propriamente dito (interpretação) e interação com a linguagem musical do usuário.

O item seguinte apresenta sucintamente as formas tradicionais de representação musical, sem ambicionar uma exposição didática a leitores não familiarizados com tais representações, a fim de evitar prolongarmos excessivamente sobre esse tema.

2.1 Representação musical tradicional

Quando se fala em representação musical, usualmente se está referindo à representação musical tradicional (TMN) ou à “partitura”, como é denominada na prática. Essa é uma representação gráfica que descreve a sucessão de eventos musicais no tempo.

Cada evento musical é discretizado em um símbolo, que traz em si uma série de atributos que o caracterizam. O evento musical mais simples é denominado *nota*. Alguns dos atributos associados a uma nota musical estão exemplificados na figura 2.1 (conforme [FIELD97]).

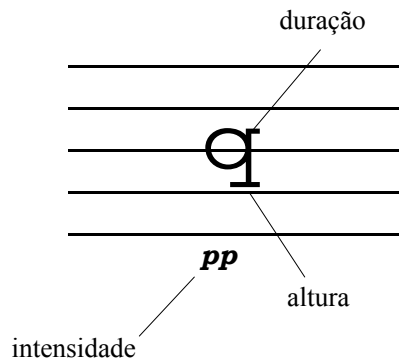


Figura 2.1 - Propriedades da nota musical na notação musical tradicional

Entre os principais atributos da nota musical encontramos a *altura*, que é um conceito relativo, tomado sempre em relação a um outro símbolo chamado *clave*, que especifica uma altura referencial, segundo a qual todas as notas terão suas alturas definidas. A altura de uma nota pode também ser alterada mediante à adição de símbolos chamados *acidentes*.

A *duração* da nota é especificada pelo seu desenho ou *figura*, de acordo com uma dada divisão do compasso e o tempo corrente. A duração de uma nota pode ser alterada pelo uso de símbolos especiais (pontos), sendo que alguns destes símbolos estão associados à articulação, comentada mais adiante.

A *intensidade* ou *dinâmica* é o atributo relacionado com a potência sonora do evento.

Pode-se observar imediatamente que os atributos citados estão sempre relacionados a um dado contexto. A *altura*, por exemplo, é sempre tomada em relação à *clave* e à *armadura da clave*; as *durações* exatas das notas não podem ser definidas sem a especificação de uma fórmula de compasso e um andamento.

Com isso sugere-se que alguns parâmetros musicais são válidos para um *agrupamento* de eventos ou notas musicais, e não

especificamente sobre cada evento individual. Alguns destes parâmetros não estão dispostos de forma idêntica para todos os eventos do agrupamento, sendo indicada uma variação progressiva do parâmetro ao longo de sua execução. Assim, indicações como *sforzando* e *rallentando* indicam operações sobre cada evento de um agrupamento, em sua intensidade e andamento, respectivamente. Dessa forma, pode-se observar que os elementos musicais estão sujeitos a um *contexto* específico em sua notação tradicional.

Alguns símbolos definem relações entre eventos consecutivos tais como as indicações de *articulação*. Informações como *legato* e *staccato*, entre outras, referem-se a alterações tanto na duração de um evento em relação ao seguinte, como especificam a maneira particular como deve ser o início de eventos (ataque) ou sua finalização (soltura). Esses símbolos também operam sobre grupos bem definidos de notas musicais.

Outros tipos de símbolos que aparecem na partitura referem-se à delimitação de trechos e instruções para a interpretação, como a imposição de repetições de trechos (*ritornelos*, duplas barras) e alteração na ordem em que os trechos são executados (*codas*, *segno*, *da capo*), *ornamentos* sobre notas (trinados, mordentes, trêmolos). Todas essas representações têm por finalidade a redução do nível de redundância da partitura para facilitar sua leitura, reduzindo o número de símbolos.

Alguns outros símbolos especiais podem estar presentes na partitura, específicos da técnica de execução do instrumento para o qual a parte é escrita. Símbolos, indicando pontuações e formas especiais de articulação, podem ser específicos a determinados instrumentos (movimento dos arcos, indicações de respiração), bem como a introdução de símbolos que requerem do intérprete a obtenção de um efeito específico (produção de harmônicos, por exemplo). De um modo geral, esses símbolos referem-se não somente ao evento musical, tomado dentro de um contexto léxico, mas ao próprio resultado sonoro ou timbrístico, conforme esperado pelo compositor.

É importante notar que, além da partitura, existe um jargão de comunicação musical que permite apontar elementos univocamente, desde a nomenclatura das notas (dó, ré, mi, fá, sol, lá, si nas línguas latinas, e as letras c, d, e, f, g, a, b nas anglo-saxônicas), alterações cromáticas ou acidentes (sustenido, bemol, bequadro ou natural), a posição absoluta da nota (“dó central”, “a sub-contraitava”), até indicações verbais como “o primeiro tema”, “a seção I” e assim por diante. Também existem sistemas de nomenclatura de acordes (maiores, menores) e funções harmônicas como “a dominante”, “a tônica”. Tudo isto constitui uma metalinguagem musical, que não aparece explicitamente no conteúdo da partitura.

Deve-se, portanto, deduzir que a representação musical tradicional (partitura) é uma representação de nível semântico baixo, estando voltada, quase que exclusivamente, à execução da obra pelo intérprete. Dificilmente o compositor considera o material musical diretamente nesse nível (o que já é sugerido pela observação do parágrafo anterior), possuindo modelos mentais diferenciados, dos quais ele se serve para a composição da música, de forma que tal processo de composição pode ser considerado como uma espécie de tradução de um modelo mental em uma forma mais imediata para a execução e tradução em som, que é a partitura propriamente dita.

Dessa maneira, pode-se esperar que as representações musicais em sistemas computacionais estejam situadas em níveis semânticos mais próximos aos da partitura, ou mais próximos ao jargão musical estruturado ou metalingüístico citado, ou mesmo de modelos semânticos ainda muito distantes da representação imediata dos eventos musicais, conforme a necessidade da aplicação em particular.

Geralmente a interação entre as duas formas de representação - tradicional e computacional - é necessária e mesmo essencial, a fim de propiciar uma interface homem-máquina mais adequada.

2.2 Representações computacionais

A possibilidade de codificação desses símbolos gráficos em elementos computacionais pode ser inferida pela existência de um conjunto finito de símbolos e de sua ordenação lógica. A existência de denominações para cada um dos símbolos da partitura já é um ponto de partida para este tipo de codificação.

É oportuno comentar que o uso de representações musicais algébricas não é estranho ao procedimento formal da música ocidental, como por exemplo, o baixo cifrado, onde representações do tipo

$$I_6 - V_7 - I$$

que, associadas às notas da linha do baixo, indicam, de forma relativamente unívoca, acordes e suas respectivas inversões. Outros números podem ainda ser inseridos para especificar quais notas devem estar dispostas na linha melódica em cada um dos acordes. Essa numeração é sempre associada a intervalos entre as notas que compõem o acorde, tomados sobre uma dada escala, ou ainda, com a adição de símbolos, em intervalos alterados (vide [SCHOENBERG74], por exemplo).

Outro exemplo, tomado da música do século XX, é o emprego dos números inteiros de 1 a 12, ou de 0 a 11, para a representação das doze classes de alturas na música serial. Cada número representa um som da escala cromática, ou, no caso do serialismo integral [SIMMS86], pode representar qualquer parâmetro, seja ele uma duração ou até alguma característica timbrística do som.

Discutiremos a seguir algumas técnicas de representação musical em sistemas computacionais. A classificação utilizada não é empregada com finalidades didáticas ou para enfatizar alguma evolução histórica, mas sim porque os casos isolados de representação servem a aplicações distintas e muitas vezes constituem formas de representação musical de níveis semânticos diferenciados:

- Representação em listas de eventos;

- Representações procedimentais;
- Representações estruturadas;
- Representações semânticas.

2.2.1 Representação em lista de eventos

Esta é a técnica mais simples e imediata de representação computacional de música. Cada parâmetro de cada evento da música ganha um símbolo ou valor próprio e é armazenado em uma estrutura de lista ordenada, na qual a ordem dos eventos é a mesma que aparece na representação tradicional.

Nesse caso, a execução da obra assim representada é bastante simples, bastando percorrer a lista do princípio ao fim, aguardando o tempo necessário para a transição entre um evento e outro, produzindo o som conforme especificado pela lista de argumentos de cada elemento da lista de eventos.

Evento:	0
Altura:	60
Intensidade:	120
Duração:	1250
Evento:	1
Altura:	61
Intensidade:	120
Duração:	1250
Evento:	2
Altura:	68
Intensidade:	100
Duração:	1460
⋮	

Figura 2.2.1 - A lista de eventos

Um exemplo de representação em lista de eventos pode ser vista na figura 2.2.1. Veja que nesta representação cada elemento da lista é um conjunto de parâmetros que especificam o evento musical por ele descrito. Cada parâmetro é devidamente codificado; no exemplo da figura, a altura do som é fornecida como na norma MIDI [FIELD97], um número inteiro na faixa 0 - 127, onde o dó central corresponde ao número 60, o dó central sustenido ao número 61, e assim por diante; também a intensidade do evento está codificada como MIDI, na faixa 0 - 127; a duração está especificada em milésimos de segundos.

Duas limitações são bastante óbvias nesse tipo de representação:

- (i) A representação da lista é estática, ou seja, seus valores estão fixados de forma absoluta. O caso mais drástico encontra-se na duração do evento, onde o valor absoluto em milésimos de segundo é fornecido. Portanto, informações como andamento e fórmula de compasso estão ausentes ou, mais especificamente, ocultas no parâmetro aqui chamado *duração*. Essa limitação pode tornar-se grave quando se desejam operações sobre o material codificado na lista. Uma simples mudança de andamento, por exemplo, requer uma série de cálculos sobre o valor do parâmetro duração;
- (ii) A representação não se presta à descrição de eventos não simultâneos, ou, pelo menos, que não estejam sincronizados. Assim, a descrição de mais vozes, soando em conjunto, não é possível com esse tipo de lista.

A primeira limitação pode ser contornada com o uso de parâmetros *relativos* e a adição de elementos especiais na lista, chamados *meta-eventos*, os quais são instantâneos (não possuem uma duração associada) e alteram o valor de tais propriedades globais. As propriedades dos eventos da lista são agora tomados de forma relativa a tais parâmetros globais. Um exemplo de propriedade global pode ser o andamento, em batidas por

minuto; um meta-evento poderia ser uma mudança de *adágio* para *presto*. Assim, as durações dos eventos estariam codificados em número de batidas, ao invés de milissegundos; observe-se que isso é mais semelhante ao que ocorre na notação musical tradicional.

A segunda limitação geralmente é contornada pela modificação da estrutura da lista para incluir o parâmetro que especifica o *início* do evento. Assim, dois ou mais eventos podem ocorrer simultaneamente, desde que possuam o mesmo instante de início. Este tipo de representação em lista insere maiores dificuldades no processamento e na execução do material codificado, uma vez que deve ser analisado qual o instante de início do próximo evento e aguardar o tempo adequado.

Outra forma de contornar o problema é a codificação de vozes independentes em listas separadas, sendo cada uma executada de forma autônoma. Obviamente, certos cuidados devem ser tomados, por exemplo, para efetuar o correto sincronismo entre tais vozes.

A combinação da técnica de lista de eventos com as soluções apresentadas anteriormente é empregada, por exemplo, na estrutura básica da norma MIDI 1.0 para arquivos de dados [FIELD97]. Note-se que essa representação é estritamente orientada à execução da música pela máquina, sendo difícil de ser tratada diretamente pelo usuário.

Na representação empregada no formato do arquivo .MID segundo a norma 1.0, definem-se diversos canais separados, chamados trilhas (tracks), que representam listas independentes de eventos musicais. Cada evento é representado, no arquivo, por um agrupamento de informações, que incluem parâmetros como instante de início (medido em uma unidade interna de relógio calculada com base no andamento em batidas por minuto), canal MIDI ao qual o evento se refere (0-15), altura e velocidade do evento. Os eventos mais comuns são o início da execução do som (NOTE ON) e o término da execução do som (NOTE OFF). Observe-se que, com esta codificação, o parâmetro referente à duração permanece oculto.

A representação dos arquivos SCORE nos programas MUSIC de Max Mathews é também a de lista de eventos, através do comando NOTE [MATHEWS69]. Os parâmetros são especificados de forma absoluta, ou seja, tempo em milésimos de segundo, alturas em hertz e intensidades em decibéis. A vantagem dessa representação está na sua flexibilidade, pois oferece inúmeras possibilidades, pelo controle quase contínuo dos parâmetros do evento, sendo, entretanto, de manuseio muito difícil para o usuário.

Dessa forma, o uso de representações musicais em listas de eventos geralmente requer mecanismos especiais de interface com o usuário para facilitar a tarefa de entrada de dados. Esse tipo de interface pode ser, inclusive, a gravação imediata dos eventos em tempo real, dada a correspondência entre a representação e a interpretação.

2.2.2 Representação procedimental

Outra maneira de pensar a execução de uma obra musical é observar que o comportamento do intérprete assemelha-se a um processo *algorítmico*, em que tal algoritmo é especificado pelo conteúdo da partitura. Assim, a seqüência de eventos executada pelo intérprete é definida conforme o estipulado pela partitura, de forma semelhante à execução de um programa por um computador.

Esse tipo de analogia pode ser ainda reforçada caso se considere que existem, dentro da própria representação musical tradicional, elementos nitidamente procedimentais, que fornecem instruções específicas ao intérprete, desde o controle sobre o fluxo da execução através de indicações como *ritornelos*, repetições de compasso ou grupos de compassos, *codas*, *trêmolos* e a indicação de ornamentos. Certas operações como alterações progressivas no andamento ou na dinâmica de certos trechos também podem, evidentemente, ser considerados como elementos procedimentais.

Assim, parece relativamente natural a representação de música através de uma seqüência de operações, de forma análoga a um programa de computador. Nessa linguagem de programação musical estarão presentes *comandos* que dão início à execução de um evento, sendo executados sucessivamente. Além desses comandos, vamos encontrar estruturas semelhantes às encontradas nas linguagens de programação usuais, como delimitação de blocos, comandos de repetição (loops), elementos condicionais (if, else) e de controle sobre a execução. Cada comando de produção de evento pode receber *parâmetros* que indicam a forma como o evento deverá ser produzido, ou então obter alguns desses parâmetros em *variáveis globais*, devendo, portanto, existir nesta linguagem também comandos capazes de alterar os valores de tais variáveis.

A codificação dos elementos musicais em tais programas é relativamente simples, muitas vezes oferecendo formas estruturadas de representação, que facilitam o trabalho do usuário, de forma que a representação pode ser tornada não-redundante. A especificação dos parâmetros não precisa ser determinística, com seus valores definidos *a priori*, de forma estática. Tais parâmetros podem ser obtidos em tempo de execução através de operações lógicas ou aritméticas, inclusive a partir de variáveis pseudo-aleatórias, oferecidas pelo sistema. Tal facilidade também permite que sistemas programados dessa forma sejam capazes de alterar parâmetros da música em tempo real, através da verificação do estado de variáveis obtidas do ambiente, por exemplo, monitorando a ação de um *intérprete*.

Note-se que aqui se apresenta a mesma limitação imposta ao número de vozes da lista de eventos mais simples. A execução de um programa é sempre linear, não cabendo, neste tipo de representação, formas imediatas de representação de eventos simultâneos e, principalmente, não sincronizados. A solução para este tipo de limitação é bastante simples, tomando emprestado ainda outro conceito da ciência da

computação, que é a noção de *processo*. Um processo é um programa em execução, podendo haver, para uma mesma unidade de processamento, diversos processos sendo executados simultaneamente. Desde que durante a maior parte do tempo de execução, um processo esteja aguardando a conclusão de um evento por ele produzido, um tratamento multitarefa é viável e de implementação relativamente simples. Cuidados especiais são necessários para o adequado sincronismo entre esses processos, e também deve-se observar que necessariamente o acesso a propriedades comuns (o andamento, por exemplo) deve ser permitido.

Uma outra limitação é mais grave e de solução apenas parcial. Boa parte do material está codificado na forma de um programa, o que torna sua alteração em tempo de execução difícil. Produzir uma variação, por exemplo, exige a reformulação de todo o corpo ou de um trecho do código, uma vez que os eventos musicais estão codificados como instruções do computador, e normalmente é vedado a um programa alterar a si próprio. Para que este tipo de problema seja amenizado, parece que uma combinação da técnica de representação musical através de lista de eventos com a representação procedimental pode ser adequada. Assim, ao invés de comandos individuais dispararem eventos (notas) isolados, podem existir comandos que iniciem seqüências pré-estabelecidas de eventos. A linguagem pode, então, incluir *operações* sobre tais seqüências, como por exemplo operações de inversão, espelhamento, transposição, etc, de forma semelhante às *funções* oferecidas pelas linguagens de programação usuais de alto nível. O uso de metodologias adaptativas também oferece alternativas para a solução do problema, por possibilitarem que determinados eventos ou seqüências de instruções alterem o próprio conteúdo do algoritmo que está sendo executado.

Várias linguagens foram propostas como representações procedimentais de música. Um exemplo é a linguagem FORMULA, desenvolvida por Anderson e Kuivila [ANDERSON91], que se baseia na linguagem FORTH e no conceito de multiprogramação, com a execução de

processos independentes. Tal linguagem oferece recursos como operações sobre controle de fluxo do programa, criação de novos processos e sincronização através de comunicação entre processos, com o estabelecimento de processos de controle que arbitram o valor dos parâmetros de execução dos demais, tais como, andamento, articulação e dinâmica.

Deve-se observar que esse tipo de representação musical é manipulável diretamente pelo usuário; essa manipulação pode não ser simples ou intuitiva para usuários leigos, mas, a princípio, é possível codificar-se música pela utilização direta dessas linguagens, de forma muito menos exaustiva do que a especificação de enormes listas de eventos e seus parâmetros individuais. No entanto, a tradução de outras formas de interface com o usuário (interfaces gráficas, ou a gravação de eventos produzidos por um intérprete, por exemplo) para esse tipo de representação absolutamente não é simples, e dificilmente chegar-se-á a uma solução eficiente.

As duas formas de representação musical apresentadas até agora são formalmente semelhantes à representação tradicional, ou seja, voltadas basicamente à execução imediata do material musical. Outras formas de representação podem incluir elementos semânticos mais sofisticados, possibilitando não apenas uma interface mais inteligente com o usuário, mas também oferecendo maiores recursos caracteristicamente computacionais, de forma a aproximar a representação musical de modelos semânticos, concebidos pelo usuário.

2.2.3 Representação estruturada

A principal vantagem da representação em lista de eventos é a sua simplicidade. No entanto, a lista é, conforme já foi comentado, de difícil manuseio para o usuário. A principal dificuldade encontra-se na incapacidade natural de ser visualizado todo o material, de uma forma abrangente. Estabelecer seções separadas, com parâmetros e

comportamentos próprios, é normalmente necessário, e, neste tipo de representação, essas seções não estão aparentes.

A representação procedimental em linguagens funcionais é interessante por oferecer mecanismos de controle sobre o material musical em tempo de execução e uma forma de interface com o usuário mais eficiente. Seguindo esse raciocínio, pode-se observar que linguagens *estruturadas* possibilitam a criação e aninhamento de blocos, os quais podem então ser manipulados diretamente, como uma unidade.

Uma limitação das representações procedimentais é a necessidade de o próprio usuário adaptar sua visão musical àquela oferecida e requerida pela linguagem, com poucas possibilidades para a representação de seu próprio modelo mental diretamente na composição.

Oferecer meios para a *estruturação* da lista de eventos é tarefa relativamente simples, já existente em muitas linguagens de programação, como por exemplo, o LISP. Essa estruturação consiste basicamente na possibilidade de que um elemento de uma lista seja, ele próprio, uma outra lista. Em última instância, o aninhamento ou hierarquização do material musical é obtido através da possibilidade de que um agrupamento de eventos musicais possa ser, igualmente, considerado como um evento e assim sucessivamente.

Além disso, a estruturação incorpora outro conceito extremamente importante e de grande utilidade prática, a *abstração* dos dados da representação. O princípio da abstração permite não apenas uma maior simplicidade de interação com o usuário, mas também possibilita a criação de operações específicas do material propriamente musical, bem como operações diferenciadas conforme o *tipo* do objeto musical do contexto.

Dessa maneira, o uso de representações musicais *estruturadas* engloba o conceito de *contexto* no aninhamento de estruturas, possibilitando não apenas formas mais eficazes de organização do material musical e a correspondente simplificação da representação, mas também permitindo ao usuário a criação de elementos musicais que sejam,

igualmente, estruturados hierarquicamente de forma natural. Engloba também o conceito de *abstração* dos dados associados a elementos musicais, levando em consideração que tais elementos representam entidades musicais ou tipos de dados específicos, diferenciados dos demais tipos de dados que a máquina é capaz de tratar, como strings, números em ponto flutuante, etc. [BUXTON78] apresenta uma série de situações concretas nas quais a descrição estruturada do material musical contribui significativamente.

O sistema *ScoreSynth*, por exemplo, desenvolvido por Haus e Sametti [HAUS91], é baseado em redes de Petri como implementação, mas introduz conceitos bastante interessantes em sua linguagem própria: o conceito de *objeto musical*, que pode ser um evento musical isolado ou uma seqüência de eventos e a definição de operações sobre tais objetos, através de uma *álgebra musical*. Tal sistema também é interessante por oferecer recursos de controle sobre a execução da partitura, com elementos em execução concorrente ou linear, de forma relativamente transparente.

A linguagem *Fugue*, desenvolvida por Dannenberg, Fraley e Velikonja [DANNENBERG91], baseia-se na linguagem LISP e permite uma espécie de combinação entre as representações procedimentais e de listas; desde que a própria linguagem origem não faça distinção nítida entre um procedimento e uma lista, ela é especialmente adequada a esta uniformização. Em *Fugue*, elementos musicais são definidos como listas, e podem ser atribuídos a símbolos definidos pelo usuário (como variáveis), o que constitui seu mecanismo básico de abstração da informação musical. O contexto é oferecido também de forma natural em LISP, uma vez que os parâmetros definidos para um dado símbolo passam a valer para todos os símbolos aninhados; o parâmetro vale para cada elemento de uma lista, por exemplo, e, no caso de um destes elementos ser associado a uma outra lista, também vai valer para esta, a menos que ela própria altere o valor de tal parâmetro, alteração essa que se apresenta como local a seu próprio contexto e ao de elementos dela derivados. A linguagem oferece uma série

de operações definidas sobre os elementos representativos dos eventos musicais descritos, como transposições, alterações de andamento, alterações da dinâmica, etc, as quais serão realizadas de forma diferenciada para cada tipo de elemento musical. A possibilidade de definição de operações, pelo próprio usuário, constitui um meio eficiente para a criação de entidades musicais autônomas, de maior nível semântico (temas, seções, etc), bem como da definição de operações mais específicas, conforme a necessidade, ou da própria forma de pensar o material musical por parte do usuário.

Um exemplo interessante e particularmente útil à interpretação de música em tempo real é o sistema *HMSL* (Hierarchical Music Specification Language) desenvolvido por Rosenboom e Polansky [POLANSKY94], que se baseia em um pequeno sistema operacional, desenvolvido pelos autores, no qual o controle sobre o material musical é dado em camadas individuais de software, à semelhança dos modelos estruturados como o ISO/OSI, por exemplo. Cada camada é responsável pelo tratamento de material musical em um dado nível semântico, desde o mais abstrato até a temporização e seqüenciação de eventos e o controle sobre variáveis contínuas, nos níveis mais básicos. Um dos objetivos do desenvolvimento de tal sistema é a possibilidade de interação do mesmo com intérpretes ou com o ambiente, em tempo real, e o controle do material musical em qualquer dos níveis da estrutura. Outra das diretrizes básicas dos autores era a possibilidade de execução do HMSL em máquinas com capacidade de processamento limitada e microcomputadores, devendo ser de simples adaptação a diferentes arquiteturas e facilmente extensível.

2.2.4 Representações semânticas

Com o objetivo de fornecer ao usuário do sistema um conjunto de mecanismos mais sofisticados, capaz de adaptar-se a seus modelos mentais e, principalmente, apto a efetuar processamentos muito mais específicos sobre o material musical, é necessária a inclusão de estruturas

que tratam não apenas de eventos musicais isolados, mas de elementos musicais abstratos; isso corresponde, claramente, a uma elevação do nível semântico da linguagem do sistema.

A implementação da semântica de um dado objeto musical consiste normalmente na definição de propriedades ou atributos, específicos para este dado objeto musical, bem como através do estabelecimento de relações determinadas entre os objetos musicais que foram definidos.

Esse tipo de representação musical nos conduz a técnicas semelhantes às redes semânticas, com suas associações, especializações e instanciações. Criar um novo objeto musical é uma tarefa que consiste em tomar uma instância de um determinado objeto musical abstrato, alterar-lhe as propriedades de forma conveniente, e associá-lo a outros objetos definidos no sistema.

Novos tipos de elementos musicais podem ser criados a partir da especialização de objetos já existentes. Os elementos musicais, aos quais estamos nos referindo aqui, podem ser traduzidos imediatamente em eventos musicais isolados, ou ainda consistirem de elementos abstratos, chamados *conceitos*. A livre definição de conceitos pelo compositor permite criar ferramentas musicais extremamente poderosas.

Um objeto musical definido em uma representação semântica consiste em uma lista de parâmetros que o caracterizam. Esta representação é similar a um *frame* nas redes semânticas, empregadas em inteligência artificial. Tais parâmetros podem ou não estar imediatamente especificados no *frame*. Caso não se encontrem especificados, devem ser obtidos através de algum caminho no interior da rede de relações deste objeto com os demais, o que é um mecanismo semelhante à estruturação e hierarquização, desejadas para o material musical.

Objetos e conceitos musicais relacionam-se de diversas formas. A mais comum e importante é a *instanciação* ou propriedade “é um” ou “um tipo de” (AKO - *a kind of*): geralmente um objeto musical é uma instanciação de algum conceito; por exemplo, o conceito “nota musical”

possui parâmetros comuns a todas as notas musicais, tais como altura, duração e assim por diante; a segunda nota do terceiro compasso é uma específica instância deste conceito. Uma relação semelhante à instanciação é a *especialização*, que cria um novo conceito, relacionado com o primeiro, com todas ou parte de suas características, com a adição de outras.

Outra relação importante é o *todo-parte*, ou “parte de”; no exemplo dado anteriormente, a segunda nota do terceiro compasso faz parte do terceiro compasso, um outro objeto do sistema. Normalmente, os sistemas de representação semântica permitem ao usuário a criação de suas próprias relações entre os objetos ou conceitos.

Veja no exemplo da figura 2.2.4.1 uma relação do tipo “parte de”, definindo uma determinada nota musical como pertencente a um acorde, o qual, por sua vez está inserido em uma dada escala e esta é, em última instância, um subconjunto da escala cromática:

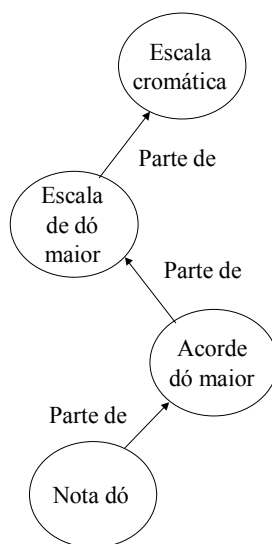


Figura 2.2.4.1 - Relações de todo-parte entre os objetos de uma representação musical

Conceitos como temas e seções podem ser definidos e utilizados, como no exemplo da figura 2.2.4.2, onde uma seqüência de objetos do tipo “nota” é associado a um tema:

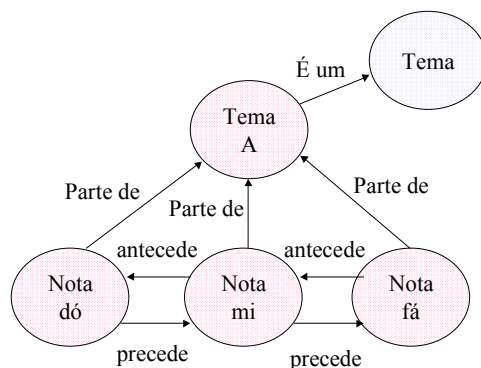


Figura 2.2.4.2 - Aplicação do conceito de tema na representação semântica

O conceito “tema” é instanciado pelo objeto “Tema A”, que é composto pelos objetos referentes às notas musicais isoladas. Observe-se que a seqüenciação de eventos é efetuada mediante o estabelecimento de relações entre tais objetos (“precede” e “antecede”).

As vantagens desse tipo de representação computacional são a flexibilidade e o tratamento semântico do material musical, em um nível de linguagem mais alto que as representações tradicionais. Esse tipo de representação emprega conceitos importantes da ciência da computação, como objetos e classes, e possibilita ao usuário a configuração de sua obra de forma abrangente e organizada, adaptando-se de forma razoavelmente flexível ao seu próprio modelo mental de composição. Sistemas de representação semântica podem possuir interfaces gráficas, com manuseio relativamente simples para o usuário. Uma característica fundamental deste tipo de representação é que ele separa a execução final da representação propriamente dita; dessa maneira, pode haver diferentes formas de execução para uma mesma rede semântica.

A desvantagem natural desse tipo de representação está, obviamente, na complexidade que o sistema possui do ponto de vista computacional, usualmente requerendo grande esforço de processamento para conduzir a representação à sua correspondente execução. A obtenção dos parâmetros indefinidos requer a execução de um algoritmo de busca

através da rede, o que algumas vezes pode limitar sua aplicação em tempo real.

Um exemplo de sistema de representação semântica é o HARP, desenvolvido por Camurri, Canepa, Frixione e Zaccaria [CAMURRI95], e definido como sendo um assistente inteligente de composição.

3. Sistemas de Composição Algorítmica

Neste capítulo, vamos resumir algumas técnicas, relatadas na literatura, que são empregadas em Composição Algorítmica, ou seja, na produção automática ou parcialmente automatizada de material musical por sistemas computacionais.

A princípio, deve-se distinguir os sistemas de composição algorítmica em três tipos básicos: sistemas de composição automática, sistemas para processamento sobre material musical previamente fornecido (também chamados de sistemas de improvisação automática) e assistentes de composição inteligentes.

O primeiro tipo de sistema efetivamente produz material musical inédito, sem o fornecimento de fragmentos musicais pelo usuário; esse fornece ao sistema apenas regras e diretrizes básicas para a composição desejada, que serão utilizadas pelo programa para a produção autônoma de música.

Outro tipo de sistema de composição algorítmica trabalha sobre material musical previamente fornecido (a “semente” da composição). Também chamado, de forma imprecisa, sistema de improvisação automatizada; esse tipo de sistema é semelhante ao anterior no que concerne ao estabelecimento de regras de manipulação e ao fornecimento de um conjunto de parâmetros, mas difere na granularidade do material musical empregado como elemento construtivo. Ao contrário do primeiro, este sistema não produz eventos musicais isolados, mas sim transformações específicas sobre o material fornecido como entrada.

O último tipo de sistema é utilizado de forma interativa, sendo uma ferramenta de auxílio à composição, tomando para si algumas das tarefas usuais do compositor humano, tendo seus resultados parciais avaliados e possivelmente reinterpretados por ele [COPE87][ROADS85]. Esse tipo de sistema difere dos anteriores por incluir aspectos interativos e fatores associados com aprendizagem.

Sistemas de composição totalmente automatizada tiveram o primeiro foco de atenção na comunidade científica. Sistemas desta natureza são de grande interesse por oferecerem subsídio para a pesquisa de diferentes estilos e o estudo da natureza do processo de composição. A disponibilidade de um sistema capaz de aceitar um conjunto de regras, representando o conhecimento necessário à própria composição, é interessante do ponto de vista formal e musicológico.

Sistemas de improvisação automática normalmente obtêm resultados práticos mais interessantes, por diversas razões. Em particular, o trabalho sobre elementos musicais pré-existentes os aproxima da forma natural do processo de composição tradicional. A capacidade de utilização de sistemas de improvisação automática em tempo real, inclusive com a possível interação com intérpretes humanos os torna atraentes como meios de expressão musical.

Os sistemas de composição auxiliada por computador reúnem características dos dois anteriormente apresentados, algumas vezes sem distinção clara entre os processos de composição e de interpretação do material musical (por exemplo o sistema “M” de [ZICARELLI87]).

Vamos considerar os sistemas de composição automática em dois níveis, de acordo com a maneira através da qual o material musical original é tratado: os sistemas determinísticos e os chamados sistemas estocásticos.

3.1 Composição Musical

Do ponto de vista computacional, o processo de composição musical não pode ser visualizado como um processo criativo ou expressivo, da forma como é concebido na atividade intelectual correspondente, uma vez que a máquina não é capaz de “expressão”, nem tampouco dotada da capacidade para criar algo novo. De fato, é paradoxal falar em *automatização* e *criação*, uma vez que os dois conceitos se contradizem pelas suas próprias definições.

Deve-se observar que o processo de composição algorítmica é sempre um processo de *busca de solução* em um espaço de problema particular como é o musical. Nesse caso, a definição precisa do que seja uma solução nesse espaço de problema é essencial, sendo, justamente, o principal objetivo da maioria dos sistemas de composição automática. A restrição da busca também é uma característica fundamental, pois a quantidade de soluções musicalmente válidas é virtualmente infinita, da mesma forma que a quantidade de elementos musicais inválidos também o é.

Outra característica importante desse particular espaço de problema está no fato de que normalmente não é possível diferenciar, de forma automática, uma *boa* solução de uma solução insatisfatória, da mesma forma que normalmente não é possível obter uma lista exaustiva das soluções.

A forma, pela qual o sistema se conduz através do espaço de problema em busca de alguma solução, também deve ser levada em consideração, de forma particular. Essa, certamente, é uma situação incomum a outros problemas computacionais, uma vez que o direcionamento na busca da solução não garante que ela seja encontrada; deve-se notar, por exemplo, que o conjunto teórico de decisões possíveis em um dado estado não é necessariamente consistente ou tampouco completo, o que quer dizer, na prática, que muitas vezes a solução não é passível de ser encontrada com o conjunto de regras ou algoritmo fornecido. Este fato parece corroborar a noção de que os compositores seguem regras teóricas quando isto é adequado, valendo-se muito mais freqüentemente de exceções às mesmas, na busca por melhores resultados para a composição.

Pode-se considerar o problema da composição algorítmica como sendo um processo de *seleção* ou *escolha*, segundo critérios e contextos específicos. Os critérios de seleção podem ser baseados simplesmente em aceitação ou recusa de uma solução parcial, mas, devido justamente às particularidades comentadas no parágrafo anterior, é conveniente que essa

aceitação seja efetuada segundo alguma gradação, de forma que seja possível distinguir de algum modo as melhores escolhas das piores, de maneira a garantir sempre a possibilidade de uma escolha, mesmo não sendo esta a escolha ótima, para que seja garantida a convergência do processo.

A questão do contexto possui um valor importantíssimo nos sistemas de composição algorítmica, uma vez que decisões composicionais são tomadas em níveis diferentes de uma composição; certas escolhas são locais, afetando elementos em algum sentido próximos ao atual, como por exemplo a escolha de uma certa harmonia, enquanto que outras escolhas afetam grandes seções, como por exemplo o estabelecimento de uma tonalidade ou de uma série, tornando-se um contexto global para toda uma seção da música que se está compondo. Deve-se levar em consideração este tipo de estruturação ou aninhamento do processo de composição a fim de se obterem melhores resultados práticos.

A questão em torno de ações criativas por parte da máquina pode ser tomada do ponto de vista da inclusão da imprevisibilidade inerente ao algoritmo de escolha. Segundo esse princípio, o algoritmo composicional seria não-determinístico em sua natureza, incluindo no processo de escolha uma característica de *sorteio*, ou seja, baseado em elementos estocásticos. Evidentemente, pode-se retomar a discussão pela impossibilidade teórica, que os sistemas computacionais apresentam, de produzirem seleções completamente aleatórias. No entanto, o uso de métodos numéricos pode produzir seqüências numéricas chamadas pseudo-aleatórias. Sistemas que incluem esse tipo de sorteio ou acaso em suas decisões composicionais são chamados estocásticos, diferindo dos demais, que não incluem a imprevisibilidade no processo de seleção, os quais são chamados determinísticos.

3.2 Compositores determinísticos

Nos sistemas determinísticos, o processo de tratamento do material musical segue um plano estabelecido e conhecido *a priori*, de forma que as mesmas operações, aplicadas sobre um material musical idêntico, venham a produzir exatamente os mesmos resultados. Pode-se dizer, no caso dos sistemas determinísticos, que não há elementos aleatórios ou dependentes de decisões que não sejam baseadas em critérios pré-estabelecidos.

Os sistemas de composição determinísticos são geralmente modelados como sistemas de produção, com regras do tipo *se - então - senão*. Não diferem, portanto, dos sistemas usuais de computação, modelados por formalismos determinísticos como, por exemplo, autômatos finitos.

Exemplo de compositor determinístico é um sistema capaz de escrever contrapontos para uma dada melodia. Esse compositor deverá escolher, para cada nota da melodia, uma segunda nota, que deverá soar simultaneamente. Tal escolha é baseada em um conjunto de regras que proíbem a ocorrência de certos intervalos em determinados contextos e a sucessão de intervalos especiais (oitavas paralelas, etc).

(Regra 1) Se está iniciando escolher a oitava
(Regra 2) Se é a última nota da melodia, escolher a oitava
(Regra 3) Não escolher quartas, segundas ou sétimas [dissonâncias]
(Regra 4) Escolher sempre a nota mais próxima da última [evitar saltos]
(Regra 5) Se o último intervalo escolhido foi a oitava, não escolher outra oitava [oitavas paralelas]
(Regra 6) Se o último intervalo escolhido foi a quinta, não escolher outra quinta [quintas paralelas]
(Regra 7) Se os dois últimos intervalos escolhidos foram terças, não escolher outra terça [dependência de vozes]
(Regra 8) Se os dois últimos intervalos escolhidos foram sextas, não escolher outra sexta [dependência de vozes]

Observe-se que este exemplo pode ser implementado de forma simples, através de um algoritmo na forma de autômato finito. Casos onde o contexto seja mais complexo do que o deste exemplo, a busca de solução

pode exigir um processo de escolha mais sofisticado, que inclua, por exemplo, retrocessos, quando mais de uma regra puder ser aplicada em um mesmo contexto e uma das regras vier a falhar. Neste caso específico e simples, a forma como o conjunto de regras foi escrito assegura qual regra deve ser aplicada em seguida, impossibilitando a inexistência de regras aplicáveis a um contexto qualquer. Em muitos casos práticos, todavia, em uma dada circunstância, várias das regras serão igualmente aplicáveis, sendo completamente arbitrária a escolha da próxima regra. Nestes casos, é possível manter-se ainda o determinismo do sistema de composição, através do uso de algoritmos de ordenação de regras (menor índice primeiro, por exemplo); no entanto, está claro que, nestas situações, a implementação aponta em direção a métodos não-determinísticos.

Um exemplo de resultado obtido com o conjunto de regras apresentado anteriormente é ilustrado na figura 3.2.



Figura 3.2 - Resultado do compositor determinístico

Nesse exemplo, a linha inferior foi fornecida ao programa, o qual definiu as notas para a linha superior, de acordo com as regras apresentadas. O algoritmo deve guardar como contexto as duas últimas escolhas intervalares entre as vozes e a última nota escolhida. O programa sempre faz uma escolha, dando preferência a não violar as regras de número menor.

Observe-se, neste exemplo, a limitação imposta pela inexistência de retrocesso, que se faz visível nas duas últimas notas escolhidas. A Regra 2 é escolhida por tratar-se da última nota da linha melódica, contrariando a Regra 5; talvez mais adequado fosse, em função da última escolha,

retroceder e alterar a penúltima nota escolhida. Mecanismos recursivos do tipo *backtracking* são relativamente simples de serem implementados, já existindo em forma nativa em algumas linguagens não procedimentais como, por exemplo, a linguagem PROLOG. Um exemplo de sistema especialista com retrocesso nas regras de composição, implementado nesta linguagem, é o C.A.A.M., desenvolvido por Fritsch [FRITSCH94].

A vantagem de um sistema sem retrocesso está na simplicidade de implementação (um simples autômato), bem como na velocidade do processamento. A recursão em muitos níveis pode consumir tempo de processamento excessivo e pode, inclusive, falhar totalmente para um dado conjunto de regras. O sistema apresentado, ao contrário, *sempre* obtém uma solução.

O fato de uma solução ser sempre encontrada é um diferencial importante. Tal solução pode não ser ótima, mas na grande maioria dos casos práticos obtém-se um resultado sonoro convincente, pois a maioria das regras foi seguida na maior parte das iterações. Isso é importante quando se deseja algum resultado, qualquer que seja, por exemplo, em situações que envolvam tempo real e interação com intérpretes humanos.

Um sistema de composição que trabalhe *off-line*, como por exemplo um assistente de composição inteligente, pode procurar validar todas as regras e, não encontrando uma solução, retornar ao usuário, para que este refine o conjunto de regras. Mesmo assim, um sistema, que não permita a violação de quaisquer regras, pode oferecer grandes dificuldades práticas ao usuário.

3.2.1 Gramáticas e sistemas de produção

Este tipo de aplicações nos sugere o uso de gramáticas como elementos geradores de material musical, como um caso particular dos sistemas de produção com regras se-então-senão, empregados, por exemplo, nos sistemas especialistas.

Uma gramática é um método bastante eficaz de descrição do material musical, uma vez que o uso de gramáticas livres de contexto permite a especificação de estruturas musicais aninhadas hierarquicamente. Já foi apontada, na introdução deste trabalho, a natureza lingüística do fenômeno musical. Supondo a música como pertencente a alguma classe de linguagem, o uso de gramáticas - meio tradicionalmente empregado como gerador de linguagens - parece ser adequado e formalmente correto para o processo de composição musical automatizada.

A aplicação desse tipo de metodologia foi proposta inicialmente por Roads [ROADS79], que sugere o uso de gramáticas ao considerar muitos dos elementos pertencentes à teoria musical como lingüísticos. Lerdahl e Jackendorf [LERDAHL83] apresentam um modelo lingüístico para a música tonal, que pode ser adequadamente representado por um conjunto de gramáticas.

A expressão de uma determinada característica musical na forma de uma gramática é relativamente intuitiva, o que pode facilitar o trabalho do usuário do sistema na especificação do conhecimento musical a ser empregado, o que vem a simplificar a interface homem-máquina.

Uma gramática é definida pelo agrupamento de um conjunto de símbolos, alguns chamados *não-terminais*, que não fazem parte da linguagem final, e outros, os *terminais*, que são efetivamente os elementos encontrados na linguagem gerada por essa gramática, um símbolo não-terminal inicial, chamado *raiz*, e um conjunto de *regras de substituição* que permitam a eliminação dos não-terminais. Roads introduz ainda nas gramáticas elementos procedimentais, que são executados quando da aplicação da respectiva regra de substituição [ROADS79].

Em uma gramática musical podemos ter, como terminais, notas ou agrupamentos de notas, enquanto que os não-terminais poderão ser elementos associados a uma metalinguagem musical, que vai incluir entidades musicais abstratas, tais como funções harmônicas, temas,

seções, etc. Dessa forma, estarão adequadamente modelados na gramática tanto a teoria musical, através da definição e da interação dos símbolos não-terminais, quanto o resultado final, em termos de símbolos terminais.

Um exemplo de um trecho de uma gramática geradora é apresentado em seguida, usado para produzir seqüências de seções temáticas, A, B e C. Os não-terminais são representados em letras maiúsculas e os terminais em minúsculas. O símbolo inicial é Σ . A seta indica substituição.

$\Sigma \rightarrow ABA$	(1)
$A \rightarrow AC$	(2)
$B \rightarrow BC$	(3)
$A \rightarrow defcA$	(4)
$A \rightarrow abcgA$	(5)
$A \rightarrow bcdaA$	(6)
$A \rightarrow defc$	(7)
$A \rightarrow abcg$	(8)
$A \rightarrow bcda$	(9)
$B \rightarrow gaB$	(10)
$B \rightarrow cdB$	(11)
$B \rightarrow ga$	(12)
$B \rightarrow cd$	(13)
$C \rightarrow bb$	(14)

Os terminais representam classes de alturas isoladas, na notação anglo-saxônica (a = lá, b = si, c = dó, etc), de forma que a regra (4), por exemplo, substitui o não-terminal referente ao tema A pela seqüência de classes ré-mi-fá-dó. Observa-se, pela primeira regra, que esta gramática vai gerar estruturas ternárias ABA, com uma possível introdução de elementos de ligação C, entre as seções. Assim, é válido o seguinte segmento melódico, gerado através dessa gramática:



Nesse exemplo, estão também representados os não-terminais referentes aos temas aqui chamados “A”, “B” e “C”, bem como a regra de substituição final que foi empregada para substituir cada um destes não-terminais. A seqüência de derivação, que produziu esta melodia segundo a gramática apresentada, poderia ter sido: $\Sigma \rightarrow ABA \rightarrow ACBA \rightarrow abcgACBA \rightarrow abcgdefcCBA \rightarrow abcgdefcbbBA \rightarrow abcgdefcbbgaBA \rightarrow abcgdefcbbgagaBA \rightarrow abcgdefcbbgagacdA \rightarrow abcgdefcbbgagacdbcdA$.

Deve-se notar ainda que a gramática fornecida é *ambígua*, ou seja, existem diversas formas de substituição de não-terminais, atendendo às regras da gramática, que permitem obter a melodia apresentada no exemplo. Isso se deve ao fato de que, para uma dada configuração de sentença, aplicar-se mais de uma regra. Isso pode ser notado facilmente, observando-se as regras (4), (5), (6), (7), (8) e (9), todas elas permitindo a substituição do não-terminal A. A substituição através de qualquer uma das regras é igualmente possível, o que sugere que essa escolha seja arbitrária.

Uma conseqüência importante dessa característica recai sobre a forma de implementação computacional de sistemas geradores baseados em gramáticas. Esse tipo de situação aponta para o uso de algoritmos não-determinísticos, em que um princípio de aleatoriedade pode ser introduzido no momento da escolha da próxima substituição. No entanto, é perfeitamente possível implementar um gerador determinístico, baseado em tais gramáticas, fazendo, para isso, com que o procedimento de escolha das regras de substituição de não-terminais siga um padrão definido *a priori*.

O tipo mais simples de padrão de aplicação das regras de substituição é o “top-most-first” ou “left-first”, ou seja, substitui-se um não-terminal sempre pela aplicação da primeira regra que se aplica ao não-terminal mais à esquerda. Este método não funciona com a gramática apresentada, uma vez que esta substituição vai conduzir a ciclos, por

exemplo, substituindo-se o não-terminal A por uma infinita sucessão do mesmo não-terminal. Para permitir substituições mais à esquerda, a gramática deveria ser reescrita adequadamente; outra técnica possível de substituição é o emprego da regra menos utilizada, quando várias regras se apliquem. Note-se que um sistema de composição implementado com esta técnica continua sendo determinístico.

O uso de gramáticas como a apresentada no exemplo anterior permite obter resultados musicais relativamente interessantes, mas o próprio Chomsky observou que a modelagem de música através de gramáticas, se é que é possível, deve ser feita através do uso de gramáticas gerais, considerando-se a música uma linguagem de nível 0, ou irrestrita. Esta classe de linguagens oferece grandes dificuldades computacionais, o que pode inviabilizar o uso de gramáticas como definição do conhecimento musical. Algumas gramáticas diferenciadas foram propostas, como as gramáticas multi-dimensionais, onde grupos de regras podem interagir, alterando o contexto das demais [ROADS79]. Um exemplo seria o emprego de gramáticas separadas para o estabelecimento da estrutura musical e da harmonia da composição. O modelo lingüístico de Lerdahl e Jackendorf [LERDAHL83] para a música tonal pode ser descrito como a interação de quatro gramáticas, cada uma modelando um aspecto particular da música.

Outra técnica semelhante ao emprego de gramáticas são as redes de restrições, um conceito empregado em sistemas especialistas para reconhecimento de linguagens naturais [ROADS79]. A questão do contexto é modelada nas redes de restrições através de uma imposição de restrições a determinadas substituições de elementos. Assim, por exemplo, se uma harmonia de lá maior é imposta, uma restrição é imposta à substituição por notas que não pertençam a essa harmonia.

3.3 Compositores estocásticos ou não-determinísticos

Nos sistemas de composição estocástica, a escolha de um determinado parâmetro da composição depende do valor de uma variável aleatória, ou seja, cujo valor não é conhecido *a priori*. Nesses sistemas, conforme se nota, substitui-se a noção de escolha pela noção de sorteio.

De certa maneira, associa-se à aleatoriedade do processo um certo conceito de *criatividade* por produzir-se algo novo (no sentido de inesperado ou imprevisto), com grande importância estética [BASSETO99]. Dessa forma, os sistemas de composição estocástica são os mais importantes, não apenas por razões históricas, mas pelo interesse em seus resultados práticos.

O emprego de sistemas não-determinísticos já nos foi sugerido anteriormente, quando observamos que, em muitas situações nos sistemas determinísticos, um número de regras é aplicável em um mesmo contexto, sendo que a escolha por alguma das regras é arbitrária. O uso de gramáticas também pode introduzir diferentes caminhos para a substituição de símbolos não-terminais.

A primeira idéia, associada a um sistema estocástico é a de *ruído*. Um processo de seleção aleatória de eventos musicais criaria uma sensação dificilmente associada à música, assemelhando-se ao ruído. Portanto, parece evidente que é necessário um mecanismo para reduzir o nível de aleatoriedade na seleção de eventos musicais, em função de um dado contexto, o que corresponderia a uma espécie de *filtragem* do nível de ruído presente.

Dessa maneira, um dado parâmetro musical possui uma determinada densidade de probabilidade associada a seus valores possíveis. Pode-se modelar um sistema de composição estocástico partindo-se de um sistema determinístico, mediante a extensão do conceito de autômato finito para o conceito de rede de Markov,

freqüentemente empregado em composição estocástica, conforme será descrito, com maiores detalhes, a seguir.

3.3.1 Redes de Markov

A rede de Markov foi uma das primeiras estratégias empregadas na composição algorítmica, por exemplo, em obras de Hiller, Cage e Varèse [HILLER59].

Uma rede de Markov é um sistema de estados e transições, como no autômato finito. O sistema estocástico, uma vez em um dado estado q , vai transitar para outro estado em função das relações de probabilidade entre as transições que partem desse estado. O que vai caracterizar a rede de Markov, chamada de primeira ordem, é o fato de que a probabilidade de o sistema atingir um dado estado depende apenas do estado atual e da probabilidade da ocorrência de uma transição entre estes dois estados [HOWARD71].

Um exemplo de sistema estocástico em rede de Markov aplicado à composição algorítmica é ilustrado pela figura 3.3.1.1. Esta rede gera melodias na escala pentatônica, de acordo com as relações de probabilidade de transição, fornecidas para cada um dos estados. Uma vez atingido um estado, o sistema produz uma nota, de forma que uma melodia é interpretada como sendo um caminho através dos estados da rede.

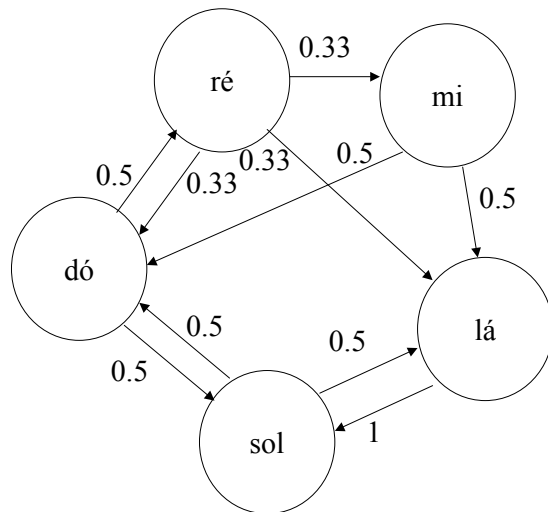


Figura 3.3.1.1 - Uma rede de Markov empregada como sistema de composição não-determinístico

Nesse exemplo, a probabilidade de seguir-se a nota *ré* a uma nota *dó* é de 50%, a de se seguir um *sol*, também 50% e assim por diante. Um exemplo de melodia gerada por esta rede pode ser visto na figura 3.3.1.2.

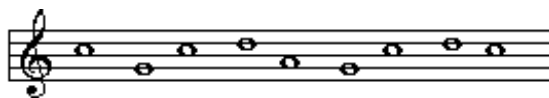


Figura 3.3.1.2 - Exemplo de linha melódica composta pela rede de Markov

Uma outra forma de representação de um modelo de Markov é através da Matriz de Transição de Estados, que, para esse exemplo em particular é a seguinte:

	dó	ré	mi	lá	sol
dó	0	0.5	0	0	0.5
ré	0.33	0	0.33	0.33	0
mi	0.5	0	0	0.5	0
lá	0	0	0	0	1
sol	0.5	0	0	0.5	0

Uma propriedade importante do modelo de Markov é que, após um número suficiente de iterações, a probabilidade de se atingir qualquer estado da rede é estatisticamente independente da escolha do estado inicial, o que permite que seja calculada uma probabilidade absoluta de o sistema encontrar-se num dado estado em um dado momento.

Esta propriedade nos mostra que, com algoritmos estocásticos modelados através de redes de Markov, é possível reduzir a aleatoriedade que anteriormente associamos ao ruído, no caso de aplicações em composição musical. Ou seja, se observarmos que em uma rede, cuja topologia permita atingir qualquer estado, com iguais probabilidades associadas a cada transição, concluímos que a probabilidade de o sistema encontrar-se em qualquer dos estados é a mesma e, conseqüentemente, o sistema é perfeitamente aleatório; por outro lado, caso a rede possua caminhos diferentes de transição, com diferentes probabilidades associadas, as probabilidades absolutas do sistema encontrarem-se em cada estado serão diferentes.

Esta diferença de relações de probabilidade absoluta, ou preferência estatística, está associada à natureza do processo modelado e aponta que, apesar do não-determinismo na escolha do próximo estado, o sistema possui um comportamento *previsível*, evidentemente derivado do conhecimento que temos sobre o processo que está sendo modelado. Aplicando-se este conceito à música, a preferência estatística por determinados estados ou caminhos no interior da rede será devido a algum tipo de imposição teórica estabelecida na definição da mesma, como por exemplo, mecanismos tonais ou imposições estruturais à música composta.

Analisando o resultado obtido no exemplo anterior, podemos observar que o comportamento da melodia gerada, embora apresente uma nítida preferência por determinados estados, ainda carece de muitas propriedades que normalmente associamos à música. Após observar o comportamento do sistema após uma quantidade suficiente de transições, observa-se que seu comportamento não segue a nenhum tipo de padrão estrutural, com um nível muito pequeno de redundância, que não seja aquela associada à nota musical isolada.

Este tipo de limitação é derivado, por um lado, da simplicidade da rede empregada e, por outro, da natureza do material musical produzido, no caso, notas musicais isoladas. A questão da granularidade do material composicional empregado já foi citada e comentada na literatura [SMOLIAR91].

Nota-se com isto que, apesar da grande potencialidade da aplicação de redes de Markov no modelamento de música, uma dificuldade é apontada quanto à estruturação do material musical produzido. Tal fato é derivado da natureza do processo de composição, onde uma decisão tomada no passado tende claramente a interferir nas escolhas feitas no momento presente, algo que não está imediatamente presente no formalismo de Markov, onde o tratamento dado ao estado atual independe dos estados pelos quais o sistema já tenha passado anteriormente.

Uma variação do formalismo markoviano permite contornar este problema parcialmente, que é o emprego dos chamados sistemas de Markov de ordem superior, dos quais o apresentado, de primeira ordem, seria um caso particular. Nessa extensão dos modelos de Markov, define-se uma rede de *ordem N* como sendo aquela em que a probabilidade de se atingir um determinado estado a partir do estado atual do sistema depende dos $N-1$ estados anteriores pelos quais a rede passou [HOWARD71]. Naturalmente, esse tipo de formalismo tende a ser muito mais complexo do que o sistema de primeira ordem, tanto do ponto de vista computacional quanto do ponto de vista prático da definição da rede.

De forma análoga ao caso anterior, sistemas de ordem N tendem a apresentar, após um grande número de iterações, uma probabilidade absoluta associada a um *conjunto* de estados, o que vem a caracterizar a existência de *ciclos* no interior da rede.

Este tipo de formalismo pode ser adequadamente empregado em composição musical automatizada, no qual podemos estabelecer um conjunto de estados fixos pelos quais o sistema deve passar, estabelecendo uma probabilidade absoluta para uma *seqüência* de estados, o que pode caracterizar uma seção específica (como um tema, por exemplo) no interior de uma composição musical.

O exemplo seguinte ilustra este tipo de idéia, aplicada a compositores estocásticos, representando um trecho de uma possível Matriz de Transição de Estados para um sistema de segunda ordem:

	dó	ré	mi	lá	sol
dó, ré	0	0	0.5	0	0.5
dó, mi	0.33	0.33	0	0	0.33
ré, ré	0	0	0	1	0
ré, lá	0	0	0.5	0	0.5

Observa-se nesse exemplo que a escolha do próximo estado depende estatisticamente dos dois últimos estados que foram escolhidos. Deve-se observar de imediato que a especificação completa de toda a rede é bastante complexa, requerendo a especificação de todas as possíveis combinações de estados, dois a dois, e suas respectivas probabilidades de transição. Note-se que é inclusive difícil de prever o resultado da execução de uma rede como essa.

Pode-se, contudo, modelar o comportamento de um sistema em que uma escolha do passado venha a interferir numa transição futura sem apelar para redes de Markov de ordem superior, utilizando para isso, por exemplo, uma rede equivalente de primeira ordem com N^2 estados [HOWARD71], realizando-se a combinação de dois estados através de uma

transição entre eles, possivelmente com probabilidade 1. Esse método pode produzir os mesmos resultados das redes de segunda ordem, com a vantagem de ser muito mais simples de se visualizar e se construir.

Outra técnica pode ser empregada sobre as redes de Markov, ainda mais eficiente para a solução da questão da estruturação musical, que consiste na associação dos estados do modelo não a notas musicais isoladas, mas a trechos musicais inteiros ou a operações a serem realizadas sobre tais trechos ou agrupamentos de notas, reduzindo-se com isto o nível de granularidade do material musical manipulado pelo modelo, mantendo-se, porém, sua simplicidade. Os resultados musicais obtidos com essa técnica tendem a ser muito superiores àqueles das redes operando sobre notas isoladas, qualquer que seja a ordem do sistema de Markov empregado.

Desta breve apresentação pode-se concluir que o emprego de redes de Markov, como modelo para compositores estocásticos, é apropriado e oferece resultados práticos úteis, embora em casos mais elaborados venha a oferecer problemas de interface, devido a questões de complexidade, envolvendo grande número de transições e probabilidades associadas, dificultando a sua construção e manutenção pelo usuário. No capítulo seguinte será apresentada uma nova visão do modelo de Markov, adicionando-lhe propriedades hierárquicas e adaptativas, que virão a aumentar-lhe grandemente a potencialidade como modelo para a construção de compositores não-determinísticos.

3.4 Alguns outros tipos de compositores automáticos

Na literatura pode ser encontrado um grande número de aplicações de composição musical algorítmica que seguem outros princípios, diferentes dos exemplos mostrados anteriormente. A título de ilustração, apresentaremos algumas destas técnicas, procurando destacar as mais importantes. Boa parte da nomenclatura que será empregada é de nossa autoria, devido à necessidade de uma classificação mais ou menos

sistemática, algumas vezes inexistente na literatura utilizada. Uma relação mais detalhada de técnicas de composição algorítmica pode ser verificada em [ROADS96].

3.4.1 Compositores por analogia

Os sistemas de composição algorítmica por analogia seguem a princípios diferentes dos apresentados até o momento, freqüentemente baseados em elementos essencialmente extra-musicais.

De fato, um compositor por analogia é uma espécie de *função de mapeamento* de uma entidade qualquer, que contenha uma certa estruturação, em uma estrutura musical correspondente. Uma vez definidos os elementos da estrutura que devem ser mapeados na música, o estabelecimento de tal função é possível, de forma que qualquer objeto representável matematicamente pode ser devidamente transformado em música.

Assim, baseando-se no comportamento de determinadas funções matemáticas, e fazendo-se associações convenientes, pode-se obter uma seqüência estruturada de sons, à qual podemos, eventualmente, atribuir propriedades musicais. Um exemplo disto é o AlgoRythms [JANZEN92], que emprega um conjunto de funções senoidais, sendo a periodicidade destas funções considerada como parâmetro estrutural fundamental.

Outro exemplo interessante é o emprego de fractais como elementos geradores de música [DODGE86]. Os fractais, como entidades matemáticas, apresentam algumas características estruturais que são de grande importância para a criação de música. A primeira dessas características é seu comportamento espectral, equivalente à hipérbole $1/f$, que, como já citado anteriormente, representa uma densidade espectral comum à grande quantidade de composições analisadas por Voss e Clarke [SMOLIAR91]. Outra característica é a similaridade interna, ou seja, um fractal é semelhante a si mesmo em níveis distintos de estruturação. Esta

estruturação que o fractal apresenta é interessante quando mapeada em uma estrutura musical, com níveis de repetição interna distintos.

O comportamento de sistemas não-lineares também pode ser empregado para ser mapeado em estruturas musicais, sobretudo daqueles sistemas que não possuem um comportamento muito bem definido, como por exemplo os sistemas caóticos. Compositores automáticos baseados em *caos* são encontrados na literatura (por exemplo [ZICARELLI87]). O comportamento dos sistemas caóticos apresenta algumas propriedades interessantes, como a parcial indeterminação do estado seguinte (ou pseudo-aleatoriedade), a ocorrência de ciclos-limite e a existência de estados perfeitamente determinísticos, tudo isso associado ao valor de alguns parâmetros, fornecidos ao modelo.

Segundo o princípio da composição por analogia, qualquer entidade dotada de organização interna pode ser convertida em música, como por exemplo imagens, estruturas sintáticas de alguma linguagem verbal, etc. Embora o resultado obtido possa ser bastante interessante do ponto de vista estético e prático, devemos observar aqui que os compositores por analogia trabalham sobre estruturas que não são musicais em sua essência, de forma que o resultado final não é exatamente previsível ou formalizável do ponto de vista estritamente musical. Em outras palavras, o compositor automático possui uma *controlabilidade* potencialmente baixa, para que um usuário especializado defina especificamente determinadas estruturas musicais por ele desejadas para as composições geradas.

3.4.2 Compositores baseados em exemplos

Nos chamados compositores por exemplos, um segmento musical é fornecido ao mesmo como um exemplo, de forma que ele possa adaptar-se a este e produzir um fragmento de música que guarde algum tipo de similaridade com a estrutura musical fornecida originalmente. Desta maneira, o sistema age de forma semelhante a um sistema com capacidade de *aprendizado*.

Os chamados compositores por estilo encontram-se entre os apresentados nesta categoria. Estes compositores fazem inicialmente uma análise do material fornecido, sendo esse uma obra ou várias obras de um ou mais compositores, e, a partir das estruturas sintáticas encontradas, o sistema produz estruturas semelhantes segundo algum princípio.

Um exemplo de compositor desse tipo é o criado por Vieira Lima [LIMA98], que extrai padrões melódicos e harmônicos de uma ou mais peças musicais fornecidas, estabelecendo um padrão estatístico que será seguido para a composição da nova peça musical.

Outros sistemas procuram encontrar temas individuais e padrões de sucessão harmônica, que procuram imitar na composição gerada. Em muitos casos, após a análise, é preenchida uma tabela de probabilidades internas de sucessões, de forma que o sistema final é, na verdade, um compositor estocástico, baseado nas relações estatísticas que foram obtidas da obra musical original.

Outras formas de implementação foram propostas, como redes neurais artificiais e autômatos celulares; a capacidade de “aprendizado”, de que é dotada uma rede neural, pode oferecer recursos interessantes à composição por exemplos, inclusive mesclando propriedades de exemplos musicais fornecidos, de forma a obter variações sobre algum deles.

Embora os resultados obtidos com este tipo de compositores possam ser considerados bastante interessantes, eles oferecem algumas dificuldades teóricas. Em primeiro lugar, apontamos que a qualidade do resultado final é fortemente dependente do sistema responsável pela análise do exemplo fornecido. Dessa forma, um analisador que apenas verifique padrões de sucessão de sons, ou ainda de seqüências harmônicas simples, pode produzir como resultado final uma composição que careça de muitos elementos estruturais, uma vez que a granularidade do material de análise foi muito baixa. Por outro lado, análises mais sofisticadas do material musical fornecido podem ser excessivamente complexas. A localização de temas individuais pode ser feita por algoritmos de

convolução, que localizem similaridades globais, mas muitas dessas similaridades são de difícil formalização e podem situar-se em níveis estruturais distintos.

A segunda deficiência que apontamos é semelhante à apresentada no item anterior, quanto à controlabilidade do processo de composição do ponto de vista formal. A menos que o sistema de análise possa fornecer ao usuário uma descrição inteligível do conhecimento adquirido com a análise do exemplo, este não é capaz de determinar exatamente o processo segundo o qual a música está sendo composta pelo sistema.

Sistemas desse tipo são de grande interesse musicológico, por incluir análise de material previamente fornecido, podendo ser empregados para estudo de elementos musicais, aos quais elementos cognitivos são associados pelos ouvintes.

3.4.3 Compositores iterativos

Estamos identificando aqui os compositores iterativos como sendo aqueles em que o processo de composição é realizado em passos específicos, nos quais é incluído algum tipo de realimentação. Assim, uma solução obtida é iterada de forma a ser refinada pelo usuário. Evidentemente, nesta categoria de algoritmos deve estar presente algum tipo de avaliação do material produzido, seja ela própria igualmente automatizada, ou ainda derivada da avaliação de ouvintes humanos.

Uma implementação interessante destes sistemas são os algoritmos de Gerar e Testar (GAT), nos quais um gerador qualquer de material musical é empregado (determinístico, estocástico, analogia, etc) e um analisador do resultado avalia esse material; essa avaliação pode ser realizada por elementos computacionais bem conhecidos, como gramáticas ou sistemas especialistas. Uma vez avaliado, o material criado pode ser descartado, corrigido segundo algum algoritmo fornecido, ou ainda ter determinados parâmetros realimentados ao programa de composição para a iteração seguinte.

O correto estabelecimento do mecanismo de análise do resultado é necessário, em primeiro lugar para que o processo venha a convergir e, em segundo lugar, para que essa convergência se faça sobre estruturas musicais válidas segundo o padrão estético desejado. Em terceiro lugar, essa avaliação precisa ser, tanto quanto possível, quantitativa em sua natureza. Essas condições são bastante difíceis de serem satisfeitas em sistemas práticos.

Outro tipo de compositores desta categoria são aqueles que empregam algoritmos genéticos. Um algoritmo genético pode adaptar-se a uma dada situação, mediante a variação de alguns de seus parâmetros a cada iteração e a comparação do resultado com o da iteração anterior. Há, na literatura, exemplos de compositores baseados nesta classe de algoritmos [JACOB95].

Essa classe de compositores automáticos apresenta um interesse especial por algumas vezes poder ser considerada como aproximação do modelo mental empregado por compositores humanos. Propondo-se a modelar esse tipo de comportamento, os compositores iterativos podem ser empregados como mecanismos de estudo do processo de composição musical.

3.5. Conclusões

Apresentamos neste capítulo um apanhado geral de diversas técnicas de composição musical algorítmica, que têm sido empregadas nesse domínio, conforme reportadas pela literatura. As comparações que foram realizadas são de grande relevância para este trabalho, permitindo a verificação de vantagens e desvantagens de cada técnica apresentada.

Com base neste apanhado, apresentaremos, nos capítulos subseqüentes, a técnica de composição algorítmica que foi desenvolvida especificamente para este trabalho, procurando relacioná-la aos aspectos tratados neste capítulo, de forma a demonstrar o seu potencial como tecnologia para a composição musical automatizada.

4. Formalização Teórica

O objetivo deste capítulo é apresentar o formalismo empregado no trabalho, o qual trata a Música como uma manifestação lingüística. Introduzem-se, resumidamente, conceitos clássicos importantes, como o de linguagem e gramática, fundamentais para a representação do conhecimento musical neste contexto. São analisados, em seguida, os reconhecedores e seu processo de construção a partir de uma gramática, com o objetivo de introduzir dispositivos geradores de material musical semelhantes a tais reconhecedores.

4.1 Linguagem

Definimos uma *Linguagem* L como sendo um conjunto de cadeias de símbolos de comprimento finito. Tais símbolos são elementos do conjunto alfabeto Σ , ou seja

$$L \subseteq \Sigma^*$$

onde Σ^* denota o fechamento recursivo e transitivo do conjunto Σ .

Cada elemento da linguagem L é uma *sentença* da linguagem. Tais sentenças são cadeias formadas pelos elementos de Σ , chamados *átomos* da linguagem.

Existem três formas usuais de definição de uma linguagem: a enumeração exaustiva de todas as cadeias componentes, a definição de regras de formação da linguagem através de uma *gramática* e a especificação de um *reconhecedor*, dispositivo capaz de identificar uma dada cadeia w como pertencente ou não à linguagem.

4.2 Gramática

Uma gramática G é um dispositivo gerador de uma linguagem, e pode ser definida como uma quádrupla do tipo

$$G = (V, \Sigma, P, S)$$

onde V é o *vocabulário* da gramática G . Os elementos de V são todos os símbolos que serão utilizados pela gramática para definir a linguagem. Σ

contêm alguns dos elementos de V , correspondendo aos símbolos ou átomos dos quais as sentenças da linguagem são constituídas. Denominam-se *não-terminais* os elementos de V que não pertencem a Σ , ou seja,

$$N = V - \Sigma$$

P é o conjunto das *produções* da gramática G , que constituem regras de formação, definindo algum não-terminal, sendo da forma

$$\alpha \rightarrow \beta$$

onde α é uma cadeia, possivelmente contendo algum não-terminal, e β uma cadeia arbitrária de elementos de V , eventualmente vazia. Define-se $\varepsilon \notin V$ a *cadeia vazia*, de comprimento nulo, que pode ser empregada na especificação de produções onde ocorre a eliminação de símbolos não-terminais.

Finalmente, S é um não-terminal dito *símbolo inicial* ou *raiz* da gramática, a partir do qual todas as sentenças da linguagem são derivadas.

O processo de geração ou derivação das sentenças da linguagem pela gramática G se dá da seguinte maneira:

(i) O símbolo S é o símbolo não-terminal inicial, e dá partida ao processo de geração;

(ii) Uma substituição de um não-terminal A é feita se e somente se existir uma produção em P , tal que seja possível a transformação da cadeia $\alpha A \beta$ para $\alpha \gamma \beta$, na qual ocorre a substituição de A por γ . Por exemplo, uma produção da forma $A \rightarrow \beta$ produz a cadeia $x\beta y$, a partir da cadeia xAy , indicando-se tal substituição por

$$xAy \Rightarrow x\beta y$$

As cadeias de símbolos de V obtidas a partir de S neste processo são chamadas *formas sentenciais*;

(iii) Chamam-se *sentenças* os elementos de Σ^* obtidos pela aplicação do passo (i) e um número arbitrário de vezes o passo (ii), desde que sejam formas sentencias que não contenham símbolos não-terminais. Indica-se

$$S \Rightarrow^* w$$

A cadeia w pertence à linguagem L se e somente se for uma sentença obtida pelos passos anteriores sobre a gramática G . Denota-se $L(G)$ a linguagem gerada pela gramática G segundo este processo:

$$L(G) = \{ w \in \Sigma^* \mid S \Rightarrow^* w \}$$

Definiu-se aqui a forma mais geral das gramáticas, conhecidas como gramáticas irrestritas. Em itens posteriores, são feitas simplificações nesta definição para a obtenção de linguagens mais simples do ponto de vista computacional.

4.3 Árvores de Derivação

Define-se a *árvore de derivação* associada à gramática G a árvore obtida da seguinte maneira:

- (i) A raiz da árvore corresponde ao não-terminal inicial S ;
- (ii) Um nó A possui filhos do tipo $\alpha_1, \alpha_2, \dots, \alpha_n$ se e somente se existir no conjunto de produções P da gramática uma produção da forma $A \rightarrow \alpha_1\alpha_2 \dots \alpha_n$, $n \geq 0$, onde os α_i são elementos do vocabulário V , terminais ou não-terminais, da gramática G .
- (iii) As folhas da árvore devem ser apenas terminais da gramática G .

Note-se que a árvore assim construída vai representar caminhos de substituição, de acordo com a ordem em que as produções são aplicadas sobre S , até chegar às sentenças que constituem a linguagem em suas folhas, visitadas seqüencialmente da esquerda para a direita.

A utilização de uma gramática como mecanismo de geração de sentenças de uma dada linguagem pode, portanto, ser considerada como um problema de busca em uma árvore de derivação correspondente.

4.4 Tipos de Linguagens

No conceito de Chomsky [NETO87], há quatro tipos de linguagens, classificadas conforme as gramáticas que lhes dão origem, e às quais também podemos associar um diferente grau de complexidade de computação de suas cadeias:

Linguagens do tipo 0 ou conjuntos recursivamente enumeráveis, obtidos pelas gramáticas irrestritas apresentadas anteriormente, onde qualquer tipo de substituição é permitida no conjunto de produções;

Linguagens do tipo 1 ou sensíveis ao contexto, em que podem existir produções nas quais a substituição de algum não-terminal esteja sujeita à presença de outros elementos da sentença (contexto), de tal forma que a ordem das substituições é importante;

Linguagens do tipo 2 ou livres de contexto, em que a substituição de um não-terminal por uma produção não é sujeita a um contexto, de forma que a ordem das substituições é irrelevante;

Linguagens do tipo 3 ou regulares, em que a substituição do não-terminal, além de incondicional, deve ocorrer obrigatoriamente na extremidade direita ou esquerda da forma sentencial, evitando-se que apareçam estruturas aninhadas nas sentenças da linguagem.

4.4.1 Gramáticas Livres de Contexto

As linguagens livres de contexto, geradas por este tipo de gramáticas, nos interessam mais diretamente devido à sua simplicidade e principalmente à existência de mecanismos práticos de implementação de reconhecedores bastante eficientes, os autômatos de pilha, também empregados na construção de compiladores.

Uma gramática $G = (V, \Sigma, P, S)$, livre de contexto, é definida conforme mostrado anteriormente em 4.2, impondo-se uma restrição sobre P , segundo a qual todos os seus elementos devem obrigatoriamente ser da forma

$$A \rightarrow \alpha$$

desde que $A \in N$ e $\alpha \in V^*$.

Uma Linguagem é livre de contexto se e somente se puder ser obtida através de uma gramática livre de contexto.

Pode-se demonstrar [NETO87] que as cadeias da linguagem L , gerada pela gramática livre de contexto, possuem a propriedade seguinte: existe em L um comprimento de sentença mínimo, acima do qual sempre é possível decompô-la na forma $uvxyz$ e, nessas condições, $uv^nxy^n z$ também fará parte de L .

A propriedade anterior ilustra a capacidade de linguagens livres de contexto apresentarem estruturas aninhadas, para as quais as cadeias v e y funcionam como delimitadores casados. Como este tipo de estrutura é muito importante nas linguagens de programação usuais, as linguagens livres de contexto têm especial interesse nestes casos, aliado ainda ao fato de os reconhecedores desse tipo de linguagem serem facilmente implementáveis.

4.4.2 Gramáticas Lineares

Se impusermos ainda mais uma restrição à forma das produções, chegamos a uma gramática ainda mais restrita, mas de grande importância prática, dada a simplicidade dos reconhecedores para linguagem obtida. Nas gramáticas lineares, os elementos de P devem obrigatoriamente ser todos da forma

$$A \rightarrow \alpha B \text{ (gramática linear à direita)}$$

ou todos da forma

$$A \rightarrow B\alpha \text{ (gramática linear à esquerda)}$$

onde A e B são cadeias arbitrárias de não-terminais (B pode eventualmente ser ϵ) e α é obrigatoriamente um terminal.

Mostra-se facilmente [NETO87] que as cadeias produzidas por esse tipo de gramática, cujo comprimento seja superior a um mínimo característico de cada gramática, podem ser expressos na forma xyz e que todas as cadeias da forma $xy^n z$ também pertencerão à linguagem, sempre que $n \geq 0$. Observe-se então que, embora a linguagem gerada não permita aninhamentos, ela se presta muito bem à representação da estrutura de elementos léxicos de linguagens mais complexas.

4.5 Reconhedores

Um *reconhedor* é uma forma alternativa de especificação de uma dada linguagem. Reconhedores são dispositivos capazes de classificar uma dada cadeia de entrada como pertencente ou não à linguagem. Um reconhedor é um dispositivo conceitual composto de

(i) Uma cadeia de entrada, composta de símbolos terminais da linguagem, um dos quais é apontado por um cursor de leitura. Esse será o próximo símbolo da cadeia a ser consultado pelo reconhedor;

(ii) Um controle finito, representado por uma máquina de estados, cujas transições são condicionadas pelo estado corrente e pelo símbolo presente na posição do cursor;

(iii) Eventualmente uma memória auxiliar, para o armazenamento de informações retiradas da parte já lida da cadeia de entrada.

A operação do reconhedor se inicia com o cursor posicionado na primeira posição da cadeia de entrada, e com a máquina de estados posicionada em seu estado inicial. O reconhedor aceita a entrada como pertencente à linguagem desde que todos os seus elementos sejam lidos, que o reconhedor termine sua operação em um estado final e que a sua memória auxiliar possua um conteúdo pré-definido (geralmente vazia) nessa ocasião.

4.5.1 Autômatos Finitos

Um *autômato finito* é um reconhecedor definido como uma quintupla da forma

$$M = (Q, \Sigma, P, q_0, F)$$

onde Q é o conjunto finito de estados do autômato, Σ o seu alfabeto de entrada, P uma função de transição na forma $Q \times (\Sigma \cup \{ \varepsilon \}) \rightarrow Q$, que determina o próximo estado do autômato, dados o estado e o símbolo na cadeia de entrada apontado pelo cursor; q_0 é o estado inicial da máquina de estados e $F \subseteq Q$, o seu conjunto de estados finais.

Note-se que um autômato finito não apresenta memória auxiliar. Prova-se que as linguagens aceitas por autômatos deste tipo são as linguagens regulares, obtendo-se assim uma equivalência com as gramáticas lineares [NETO87].

A configuração do autômato finito em cada instante será dada por $(q_i, z) \in Q \times \Sigma, zw$, onde o par ordenado (q_i, z) denota o estado atual (q_i) e a parte da cadeia ainda não consumida w , com zw a cadeia de entrada original.

Dada uma gramática G , linear, pode-se construir um autômato finito que reconheça a linguagem por ela gerada da seguinte forma:

- (i) Ao símbolo não-terminal S associa-se o estado inicial q_0
- (ii) Para cada produção do tipo $A_n \rightarrow \alpha A_{n+1}$ incluir o estado q_{n+1} , caso ainda não exista, e uma transição de q_n para o mesmo, consumindo o símbolo α (eventualmente vazio).

Observe-se que o autômato finito assim construído vai reconhecer apenas as sentenças produzidas pela gramática G fornecida [NETO87].

4.5.2 Autômatos de Pilha

Um *autômato de pilha* é um reconhecedor definido como uma sêxtupla

$$M = (Q, \Sigma, \Gamma, P, q_0, F)$$

onde Q é o conjunto finito de estados do autômato, Σ o seu alfabeto de entrada, Γ o alfabeto de pilha, P uma relação de transição na forma $Q \times (\Sigma \cup \{ \varepsilon \}) \times \Gamma \rightarrow Q \times \Gamma^*$, determinando o próximo estado e topo da pilha, a partir do estado atual, do símbolo apontado pelo cursor na cadeia de entrada e do topo da pilha correntes; q_0 é o estado inicial da máquina de estados e $F \subseteq Q$ o conjunto de estados finais do autômato.

O estado do autômato é definido por $(q_i, w, z) \in ((Q \times \Sigma) \times \Gamma)$, onde a tripla (q_i, w, z) denota o estado corrente, a parte da cadeia de entrada ainda não lida e o conteúdo corrente da pilha. O autômato de pilha reconhece uma cadeia de entrada como pertencente à linguagem que define, após percorrê-la totalmente, se e somente se estiver em um estado final $q_j \in F$ e estando a pilha vazia.

Note-se que o autômato de pilha possui uma memória auxiliar organizada como uma pilha. Prova-se que a classe das linguagens reconhecidas por estes dispositivos é a das linguagens livres de contexto [NETO87].

Pode-se construir um autômato de pilha que reconheça a linguagem gerada por uma gramática $G = (V, \Sigma, P, S)$, livre de contexto, da seguinte forma:

- (i) $M = (Q, \Sigma, \Gamma, P, q_0, F)$, $\Gamma=V$, $F=\{ q_f \}$;
- (ii) Em P deverão constar as transições $((q_0, \sigma, \varepsilon), (q_0, \sigma))$ para cada $\sigma \in \Sigma$, ou seja, empilha-se cada símbolo da cadeia de entrada;
- (iii) Para cada produção $A \rightarrow \alpha$ tem-se em P a transição $((q_0, \sigma, \alpha^R), (q_0, A))$, onde a operação α^R denota a cadeia reversa da cadeia α . A ocorrência dessa transição, portanto, substitui a cadeia α , no topo da pilha, pelo símbolo não-terminal A ;
- (iv) Inclui-se em P a transição $((q_0, \varepsilon, S), (q_f, \varepsilon))$ que encerra o processo, removendo o símbolo inicial S da pilha e saltando para o estado final, indicando o reconhecimento da cadeia de entrada.

O reconhecedor canônico assim obtido denomina-se *reconhecedor ascendente* [NETO87]. Observe-se que a linguagem reconhecida por esta classe de reconhecedores é livre de contexto. Pode-se demonstrar ainda que esta classe de reconhecedores não é capaz de reconhecer linguagens sensíveis ao contexto, as quais apenas podem ser tratadas por dispositivos mais genéricos, como as máquinas de Turing.

4.6 Autômatos não-determinísticos

Máquinas de estado não-determinísticas são aquelas em que, para uma mesma cadeia de entrada, estando a máquina em um dado estado, o estado seguinte não é único, ou seja, há diferentes caminhos possíveis de consumo dos símbolos da cadeia de entrada, alguns dos quais podendo conduzir a estados finais, com a cadeia reconhecida, e outros não. Os algoritmos canônicos apresentados, para a obtenção de reconhecedores a partir de gramáticas lineares e livres de contexto nos itens 4.5.1 e 4.5.2 podem, eventualmente, produzir autômatos não-determinísticos, devido à inserção de transições em vazio, sem o consumo de símbolos da cadeia de entrada, ou então de mais de uma transição consumindo um mesmo símbolo, partindo de um mesmo estado.

Embora do ponto de vista teórico a existência de não-determinismos nos reconhecedores não apresente qualquer problema, sua implementação em sistemas computacionais torna-se mais complexa e ineficiente, transformando-se num problema de busca, do tipo tentativa-e-erro.

Normalmente, a construção de reconhecedores não-determinísticos é indesejável e desnecessária, pois muitas vezes é possível, a partir de um reconhecedor contendo não-determinismos, obter um reconhecedor determinístico equivalente, possivelmente com um número maior de estados. No entanto, a idéia da existência de não-determinismos nos vai propiciar a criação de mecanismos de geração de cadeias da linguagem através de dispositivos derivados de seus reconhecedores.

4.7 Redes de Markov

Define-se uma rede de Markov como uma tripla

$$K = (Q, q_0, T)$$

onde Q é um conjunto finito com n estados, $q_0 \in Q$ o estado inicial da rede e T uma matriz $n \times n$, com as seguintes propriedades

$$0 \leq t_{ij} \leq 1, \text{ com } 0 \leq i < n \text{ e } 0 \leq j < n \text{ e}$$

$$\sum_{i=0 \dots n-1} t_{ij} = 1$$

para qualquer j , ou seja, soma dos valores de qualquer linha da matriz T é igual à unidade.

O valor do elemento t_{ij} da matriz T representa a probabilidade de ser atingido o estado q_j , a partir do estado q_i . Assim, a transição entre estados ocorre de forma aleatória, mas condicionada pelas relações de probabilidade definidas em T . A matriz de probabilidade de transição de estados assim definida garante que a probabilidade de um dado estado ser atingido depende exclusivamente do estado atual da rede, propriedade esta que define esta rede de Markov como *de primeira ordem* [HOWARD71].

Prova-se que, partindo-se do estado inicial q_0 , e após um número suficientemente grande de iterações da rede, a probabilidade de a rede encontrar-se em um de seus n estados é constante [HOWARD71]. Redes de Markov normalmente são empregadas em modelos matemáticos de processos físicos, mas vamos fazer uma adaptação, relacionando-as com os autômatos finitos não-determinísticos, o que nos vai possibilitar utilizá-las como dispositivos *geradores* de linguagens.

Definição: a máquina de Markov M é definida pela quintupla

$$M = (Q, \Sigma, T, P, q_0)$$

Onde os novos elementos Σ e P representam, respectivamente, o alfabeto de saída de M e um conjunto de símbolos de saída p_{ij} , associados às transições t_{ij} , para a anexação de elementos de Σ à lista de saída. O símbolo $p_{ij} \in \Sigma \cup \{ \varepsilon \}$ é inserido na cadeia de saída de M quando da

ocorrência da transição entre os estados q_i e $q_j \in Q$, cuja probabilidade de ocorrência, sendo o estado atual igual a q_i , é definida pelo valor do elemento t_{ij} da matriz T .

Teorema. A linguagem formada pelas possíveis cadeias de saída contendo símbolos produzidos por uma máquina de Markov M é linear.

Demonstração.

Vamos definir um conjunto arbitrário $F \subseteq Q$ de estados da máquina de Markov M , que passamos a chamar, por definição, estados finais de M .

Pode ser construído um autômato finito, eventualmente não-determinístico, $M' = (Q', \Sigma', P', q_0, F')$ com as seguintes propriedades:

- O conjunto de estados é equivalente ao da máquina de Markov M , possuindo o mesmo número de estados;
- O alfabeto do autômato, Σ' , é idêntico ao conjunto Σ ;
- O estado inicial q_0 de M' será o estado equivalente definido para a máquina de Markov M , bem como o conjunto de estados finais definido anteriormente;
- Definimos o conjunto de transições P' da seguinte forma:

Para todo $t_{ij} \in T$ com $p_{ij} \in P$ associado, tal que $t_{ij} > 0$, a transição $(q_i, p_{ij}) \rightarrow q_j \in P'$.

Para uma máquina de Markov M , a linguagem gerada a partir de um estado q_i , arbitrário, é dada por

$$L(q_i) = \{ p_{ij} L(q_j) \mid \exists t_{ij} \in T, \text{ com } t_{ij} > 0, p_{ij} \neq \varepsilon \}, \text{ se } q_i \notin F$$

e

$$L(q_i) = \{ p_{ij} L(q_j) \mid \exists t_{ij} \in T, \text{ com } t_{ij} > 0, p_{ij} \neq \varepsilon \} \cup \{ \varepsilon \}, \text{ se } q_i \in F$$

Por definição, a linguagem gerada pela máquina de Markov M , é dada por

$$L(M) = L(q_0)$$

Por outro lado, a linguagem reconhecida pelo autômato finito M' definido no parágrafo anterior, a partir de um estado q_i , arbitrário, é dada por

$$L'(q_i) = \{ \sigma L'(q_j) \mid \exists (q_i, \sigma) \rightarrow q_j \in P' \}, \text{ para todo } q_i \notin F$$

e

$$L'(q_i) = \{ \sigma L'(q_j) \mid \exists (q_i, \sigma) \rightarrow q_j \in P' \} \cup \{ \varepsilon \}, \text{ para todo } q_i \in F.$$

Em particular, para o estado q_0 , tal linguagem é a linguagem reconhecida pelo autômato,

$$L'(M') = L'(q_0)$$

Como M' é um autômato finito, a linguagem $L'(M')$ é regular.

Suponhamos que a cadeia de saída da máquina de Markov M seja a cadeia de entrada do autômato M' . Para qualquer estado q_i , verifica-se, portanto, que

$$L(q_i) = L'(q_i)$$

uma vez que, por construção de M' , as condições para o consumo do símbolo σ_{ij} da cadeia de entrada estarão satisfeitas se e somente se as condições para a anexação de σ_{ij} à cadeia de saída de M também forem satisfeitas.

Em particular, para o estado inicial q_0 , $L(q_0) = L'(q_0)$. Como tais estados iniciais coincidem por definição,

$$L(M) = L'(M')$$

Logo, $L(M)$ é regular. \square

Podemos utilizar o resultado anterior para comprovar a equivalência entre as máquinas de Markov e os autômatos finitos, e que aquelas poderão ser empregadas eficientemente na geração de linguagens regulares, de forma equivalente às gramáticas lineares.

Seja, então, uma linguagem L , regular, e M , um reconhecedor finito para suas cadeias. Vamos definir a máquina de Markov M' , que possui o mesmo número de estados que M . Para cada transição $p \in P$ do reconhecedor M da forma $(q_i, \sigma) \rightarrow q_j$, definimos para M' , na matriz T , o elemento t_{ij} , cujo valor é igual a $1/m$, onde m é o número de transições que partem do estado q_i em M . Além disso, associamos a esta transição particular da matriz T a geração do símbolo $p_{ij} = \sigma$ em P' , correspondente à anexação do símbolo σ à sua cadeia de saída. Todos os demais elementos da linha i da matriz T , que não possuam transições equivalentes $(q_i, \sigma) \rightarrow q_k$, $k \neq j$, em M vão conter o valor $t_{ik} = 0$. O estado inicial q_0 de M' corresponde ao estado inicial q_0 de M .

A máquina não-determinística M' assim construída vai, a cada transição entre dois estados definidos, produzir um novo símbolo σ a ser inserido em sua cadeia de saída. A cada passagem de M' por qualquer estado que corresponda a um estado final de M , a cadeia de saída conterà uma sentença válida da linguagem L .

Teorema. O dispositivo M' , assim construído, gera a linguagem L .

Demonstração.

De forma análoga à demonstração anterior, temos que, para um dado estado q_i , arbitrário, tal que $q_i \in Q$, a linguagem reconhecida pelo autômato M é dada por

$$L(q_i) = \{ \sigma L(q_j) \mid \exists (q_i, \sigma) \rightarrow q_j \}, q_i \notin F$$

e

$$L(q_i) = \{ \sigma L(q_j) \mid \exists (q_i, \sigma) \rightarrow q_j \} \cup \{ \varepsilon \}, \text{ para todo } q_i \in F.$$

A linguagem gerada pela máquina de Markov M' , definida anteriormente, a partir do estado q_i , é dada por

$$L'(q_i) = \{ p_{ij} L'(q_j) \mid \exists t_{ij} > 0 \text{ e } p_{ij} \neq \varepsilon \}, q_i \notin F$$

e

$$L'(q_i) = \{ p_{ij} L'(q_j) \mid \exists t_{ij} > 0 \text{ e } p_{ij} \neq \varepsilon \} \cup \{ \varepsilon \}, \text{ para todo } q_i \in F'.$$

Ora, novamente por construção, foi imposto que $t_{ij} = 1/m$, não nulo para todas as transições equivalentes $(q_i, \sigma) \rightarrow q_j$ em P , e nulo nos demais casos, e que $p_{ij} = \sigma$ para cada transição. Logo $L(q_i) = L'(q_i)$ e, em particular, $L(q_0) = L(M) = L'(q_0) = L'(M')$. \square

O método de produção de sentenças da linguagem por meio da máquina de Markov, equivalente ao reconhecedor, é formalmente equivalente ao das gramáticas, com a diferença básica de que sua implementação é mais simples do ponto de vista computacional. Uma máquina de Markov assim construída vai produzir, de forma aleatória, sentenças válidas da linguagem de origem, sem a necessidade de buscas exaustivas em árvores de derivação.

4.8 Autômatos Adaptativos

A definição dos autômatos adaptativos que está sendo feita neste trabalho é ligeiramente simplificada com relação ao formalismo original, conforme definido por [NETO93]. O leitor poderá verificar que tal simplificação não compromete a potência computacional do modelo original.

Um *autômato adaptativo* é um reconhecedor definido como uma sêxtupla da forma

$$M = (Q, \Sigma, P, q_0, F, \Phi)$$

onde Q é o conjunto de estados do autômato, Σ o seu alfabeto de entrada, P uma função de transição na forma $Q \times (\Sigma \cup \{\epsilon\}) \rightarrow Q \times \Phi^*$, indicando o próximo estado, dados o estado e o símbolo na cadeia de entrada atuais, e um conjunto de ações adaptativas a serem executadas, contido no fechamento recursivo e transitivo do conjunto Φ ; q_0 é o estado inicial da máquina de estados, $F \subseteq Q$, o conjunto de estados finais e Φ o conjunto de ações adaptativas definidas sobre P :

$$\Phi = \{ \text{insere } p_i \text{ em } P, \text{ remove } p_i \in P, \text{ consulta } p_i \in P \},$$

$0 \leq i < n$, onde n é o número de estados do autômato.

Uma ação adaptativa de *inserção* da transição p_i adiciona tal elemento ao conjunto P , correspondendo à definição de uma nova transição para o autômato. Analogamente, a ação adaptativa de *eliminação* exclui elementos deste conjunto. Tais operações permitem ao autômato adaptativo a reconfiguração de sua própria topologia.

Por outro lado, a ação adaptativa de *consulta* corresponde à validação da transição que está associada a esta ação, em função da existência do elemento p_i em P . Dessa maneira, a transição para a qual está definida a ação adaptativa de consulta apenas ocorre, executando outras eventuais ações adaptativas de inserção ou remoção, caso $p_i \in P$. Caso mais do que uma ação adaptativa de consulta esteja definida para uma mesma transição, tal transição apenas será validada caso *todas* as ações de consulta existentes resultem em sua validação.

Dessa forma, para o autômato adaptativo, o conjunto P não é constante, mas pode ter elementos adicionados ou removidos durante a operação do autômato, através da execução das ações adaptativas associadas a suas próprias transições. Assim, a situação ou configuração do autômato adaptativo é definida não apenas pelo seu estado corrente,

mas também pela topologia geral do reconhecedor, o qual representa o próprio algoritmo de reconhecimento, e que passa a evoluir conforme ocorrem as transições, dependendo da seqüência de símbolos na cadeia de entrada.

Demonstra-se que tal formalismo é equivalente às máquinas de Turing [NETO93], constituindo, portanto, reconhecedores de linguagens sensíveis ao contexto.

4.9 Máquinas de Markov Adaptativas

Uma máquina de Markov adaptativa é definida como a sétupla

$$M = (Q, \Sigma, q_0, T, P, A, \Phi)$$

onde Q é o conjunto de estados da máquina, Σ o alfabeto de saída, $q_0 \in Q$ o estado inicial, T é uma matriz $n \times n$, com n o número de elementos de Q , cujos elementos t_{ij} gozam das seguintes propriedades

$$0 \leq t_{ij} \leq 1, \text{ com } 0 \leq i < n \text{ e } 0 \leq j < n, \text{ e}$$

$$\sum_{i=0 \dots n-1} t_{ij} = 1$$

para qualquer j , ou seja, a soma dos valores de qualquer linha da matriz T é igual à unidade. P um conjunto de símbolos de saída, associados às transições t_{ij} , os quais são anexados à cadeia de saída da máquina M , quando de sua ocorrência. A matriz A , $n \times n$ tem seus elementos na forma

$$a_{ij} \subseteq \Phi^*$$

Φ é um conjunto de ações adaptativas sobre transições $t_{ij} \in T$, na forma

$$\Phi = \{ \text{altera } t_{ij}, \text{ consulta } t_{ij} \},$$

$$0 \leq i < n, 0 \leq j < n$$

O valor do elemento t_{ij} da matriz T representa a probabilidade de ser atingido o estado j , a partir do estado i , à semelhança do que ocorre no caso das máquinas de Markov, definidas em 4.7. Em função da ocorrência de uma particular transição entre os estados i e j , o conjunto associado de ações adaptativas $a_{ij} \in A$ é executado. As ações adaptativas definidas sobre

a máquina, representadas no conjunto Φ , são definidas de forma análoga ao que foi feito para os autômatos adaptativos, no item 4.8.

Uma ação adaptativa de *alteração* sobre T é definida como sendo a modificação da linha i da matriz, de forma que algum elemento t_{ij} assumira um novo valor, conforme estabelecido pela ação adaptativa, enquanto os demais elementos da linha i , $0 \leq k < n$, com $k \neq j$, são proporcionalmente reajustados de tal forma que as propriedades da matriz T original sejam mantidas:

$$t_{ik} = t_{ik} \cdot \left(\frac{1 - v}{1 - t_{ij}} \right), 0 \leq k < n, k \neq j$$

$$t_{ij} = v$$

onde v é o novo valor para o elemento t_{ij} , conforme estipulado pela ação adaptativa.

Operações de *consulta* são definidas como sendo a validação da transição que está associada a tal ação, em função do valor corrente de algum elemento t_{uw} . Dessa maneira, a transição para a qual está definida a ação adaptativa de consulta apenas ocorre, executando outras eventuais ações adaptativas de alteração, caso o valor do elemento consultado $t_{uw} > 0$.

Os símbolos de saída $p_{ij} \in P$ são anexados a uma cadeia de saída, de forma semelhante ao efetuado anteriormente.

Teorema. A linguagem L , gerada pela máquina de Markov adaptativa $M = (Q, \Sigma, q_0, T, P, A, \Phi)$, conforme definida acima, é sensível ao contexto.

Demonstração.

De forma semelhante à demonstração correspondente no item 4.7, vamos definir um autômato adaptativo que reconheça a linguagem gerada pela máquina de Markov adaptativa.

Seja F um conjunto arbitrário de estados da máquina de Markov adaptativa M , tal que $F \subseteq Q$, cujos elementos passamos a chamar estados finais de M .

Seja um autômato adaptativo $M' = (Q', \Sigma', P', q_0, F', \Phi')$, definido da seguinte forma:

- O conjunto de estados é equivalente ao da máquina de Markov adaptativa M , possuindo o mesmo número de estados;
- O alfabeto do autômato Σ' coincide com o alfabeto da máquina, Σ ;
- O estado inicial q_0 de M' será o estado equivalente, q_0 , definido para a máquina de Markov adaptativa M , o mesmo ocorrendo para os conjuntos de estados finais, F e F' ;
- Definimos o conjunto de transições P' da seguinte forma:
Para todo $t_{ij} \in T$ com $p_{ij} \in P$ associado, tal que $t_{ij} > 0$, a transição $(q_i, p_{ij}) \rightarrow (q_j, a_{ij}')$ $\in P'$, onde a_{ij}' é o conjunto de ações adaptativas, eventualmente vazio, equivalente a $a_{ij} \in A$;
- O conjunto Φ' é construído da seguinte forma:
 - Para os elementos de Φ que correspondem à alteração de t_{ij} ,
 - Se $t_{ij} > 0$ a operação correspondente em Φ' deverá ser uma operação de adição de transição entre os estados equivalentes de Q' ;
 - Se $t_{ij} = 0$ a operação correspondente em Φ' deverá ser uma operação de remoção de transição entre os estados equivalentes de Q' .
 - Para os elementos de Φ referentes a operações de consulta, operação equivalente é definida para Φ' .

Se a cadeia de saída de M for igual à cadeia de entrada do autômato adaptativo assim definido, M' , é fácil observar que a toda cadeia

produzida por M será reconhecida por M' , sempre que M encontrar-se em um estado $q_i \in F$, uma vez que, por construção de M' , transições consumindo o símbolo σ da cadeia de entrada de M' existem se e somente se existirem transições equivalentes em M produzindo o símbolo $p_{ij} = \sigma$. Tal equivalência permanece válida enquanto a topologia de M' for equivalente à topologia de M .

No entanto, a ocorrência de ações adaptativas, as quais podem alterar a topologia de M durante a sua execução, não invalida o resultado anterior, uma vez que M' possui ações adaptativas que, por construção, adicionam ou removem transições entre estados equivalentes de Q e Q' . De acordo com a construção de M' , as ações adaptativas definidas em Φ' garantem que a topologia do autômato adaptativo M' permaneça equivalente à topologia da máquina de Markov adaptativa M .

Desta forma, podemos concluir que a linguagem gerada por M é reconhecida por M' . Como M' é um autômato adaptativo, capaz, portanto, do reconhecimento de linguagens sensíveis ao contexto, e, como tais máquinas foram construídas de forma genérica, segue que a linguagem L , gerada por M e reconhecida por M' é sensível ao contexto.

□

A demonstração da equivalência dos formalismos adaptativos definidos neste capítulo pode ser efetuada através do mesmo raciocínio apresentado na demonstração do item 4.7.

4.10 Sistemas de Markov Adaptativos

Vamos definir um Sistema de Markov Adaptativo como sendo um conjunto de máquinas de Markov, que podem se relacionar através de ações adaptativas. Assim, pode-se estabelecer uma relação de escopo sobre as ações adaptativas definidas para cada máquina, classificando-as em: ações que modifiquem apenas a topologia local à máquina e ações que

podem interferir no comportamento de outras máquinas pertencentes ao conjunto.

Cada elemento de um sistema de Markov adaptativo é definido como a sétupla:

$$M^k = (\Sigma, Q^k, q_0^k, T^k, P^k, A^k, \Phi^k)$$

onde Σ é o alfabeto de saída, comum a todas as máquinas do sistema, Q^k é o conjunto de estados da máquina de índice k , $q_0^k \in Q^k$ seu estado inicial, T^k uma matriz $n^k \times n^k$, com n^k o número de elementos de Q^k , com as mesmas propriedades da máquina de Markov adaptativa,

$$0 \leq t_{ij}^k \leq 1, \text{ com } 0 \leq i < n^k \text{ e } 0 \leq j < n^k \text{ e}$$

$$\sum_{i=0 \dots n^k-1} t_{ij}^k = 1$$

para qualquer j , ou seja, a soma dos valores de qualquer linha da matriz T^k é igual à unidade. P^k é o conjunto dos símbolos anexados à cadeia de saída, associados às transições t_{ij}^k . A^k é outra matriz $n^k \times n^k$, cujos elementos a_{ij}^k são da forma

$$a_{ij}^k \subseteq \Phi^{k*}$$

Φ^k é um conjunto de ações adaptativas sobre transições $t_{ij}^k \in T^k$. Particiona-se Φ^k da forma

$$\Phi^k = \phi^k \cup \zeta^k, \text{ de maneira que } \phi^k \cap \zeta^k \text{ é vazio.}$$

O conjunto ϕ^k é um conjunto de ações adaptativas internas, aplicáveis sobre a máquina k , na forma

$$\phi^k = \{ \text{altera } t_{ij}^k, \text{ consulta } t_{ij}^k \}$$

$$0 \leq i < n^k, \quad 0 \leq j < n^k$$

e ζ^k um conjunto de ações adaptativas sobre outras máquinas pertencentes ao sistema, na forma

$$\zeta^k = \{ \text{altera } t_{ij}^w, \text{ consulta } t_{ij}^w, \text{ ativa } M^w, \text{ desativa } M^w \},$$

$$\text{com } w \neq k, \quad 0 \leq i < n^w \text{ e } 0 \leq j < n^w$$

A ação adaptativa de *alteração* sobre T^u é definida como a modificação da linha i da matriz, de forma que o elemento t_{ij}^u assuma o valor estabelecido pela ação adaptativa, enquanto que os demais elementos da linha i , $0 \leq k < n^u$, com $k \neq j$, são reajustados para que as propriedades da matriz T_u original sejam mantidas:

$$t_{ik}^u = t_{ik}^u \cdot \left(\frac{1 - v}{1 - t_{ij}^u} \right), 0 \leq k < n^u, k \neq j$$

$$t_{ij}^u = v$$

onde v é o novo valor para o elemento t_{ij}^u , conforme estipulado pela ação adaptativa.

Operações de *consulta* são especificadas como a validação da transição que está associada a essa ação, em função do valor corrente de algum elemento t_{ij}^u . Dessa maneira, a transição para a qual está definida a ação adaptativa de consulta apenas ocorre, executando outras eventuais ações adaptativas de alteração, caso o valor do elemento consultado t_{ij}^u seja diferente de zero. Observe-se que tanto a consulta como a alteração do valor do elemento t_{ij}^u estão definidas sobre qualquer máquina M^u do sistema.

As operações de desativação e ativação da máquina M^w por uma máquina M^k , com $k \neq w$, representam a suspensão e reabilitação de sua execução, respectivamente. Uma máquina *desabilitada* por outra permanece em seu estado mais recente, não executando qualquer ação adaptativa até que seja novamente habilitada, por uma ação adaptativa complementar de *ativação*.

Note-se que a definição dos sistemas de Markov adaptativos apenas serve a fins práticos, uma vez que a distribuição do controle em um sistema adaptativo não altera o potencial de computação do mesmo, o que pode ser verificado pela possibilidade de definição de uma *única* máquina de Markov adaptativa equivalente a um sistema, o que pode ser realizado

pelo mapeamento de todos os estados e ações adaptativas correspondentes de cada uma das máquinas do sistema original. As operações, referentes a ativação e desativação de máquinas individuais, são traduzidas em transições de e para estados de “inatividade” equivalentes, em tal máquina adaptativa única. Observando que tal mapeamento é possível, pode-se concluir que a classe de linguagens gerada pelos sistemas de Markov adaptativos também é sensível ao contexto.

4.11 Conclusões

Estamos considerando o uso de gramáticas como uma forma eficiente de tradução do conhecimento necessário à especificação de uma linguagem L qualquer, devido à sua inerente simplicidade e razoável compatibilidade para com modelos mentais que utilizamos para a representação de tais linguagens.

No entanto, a implementação direta de gramáticas em sistemas computacionais para a produção das cadeias pertencentes à linguagem definida é complexa, tornando-se um problema de geração de árvores de derivação, com implementações ineficientes do ponto de vista computacional. Por outro lado, a implementação de reconhecedores eficientes para a linguagem é bastante simples de ser realizada. A tradução das gramáticas em reconhecedores para a linguagem gerada é possível e muitas vezes direta, através de algoritmos que podem viabilizar a automatização de tal processo.

A idéia do emprego de dispositivos análogos a tais reconhecedores, quais sejam, redes de Markov com a anexação de símbolos a uma cadeia de saída, não com a finalidade de reconhecer, mas sim de gerar, cadeias de símbolos que sejam pertencentes à linguagem, permite que as vantagens apresentadas no parágrafo anterior, referentes aos reconhecedores, possam ser igualmente aplicadas à *geração* das sentenças da linguagem, sem a necessidade de recorrer-se à geração exaustiva de árvores de derivação.

Dessa forma, segundo o método aqui proposto, o conhecimento acerca da linguagem é formalizado por meio de uma gramática, que é empregada para a definição de um reconhecedor para a linguagem por ela gerada. Tal reconhecedor é, então, convertido para a forma de um gerador da linguagem, pela atribuição de relações de probabilidade de ocorrência de suas transições de estado e pela anexação de símbolos a uma cadeia de saída, em substituição ao consumo de símbolos de uma cadeia de entrada.

A utilização de formalismos adaptativos permite que sejam modelados aspectos referentes a linguagens sensíveis ao contexto, a cujo reconhecimento não se aplicam os modelos baseados em autômatos (finitos e de pilha). Isto é bastante interessante para o gerenciamento de linguagens mais genéricas e complexas. As máquinas de Markov adaptativas propostas cumprem esta finalidade na geração de cadeias pertencentes a linguagens sensíveis ao contexto, como equivalentes aos autômatos adaptativos, conforme demonstrado neste capítulo.

Introduzindo o conceito de sistema de Markov adaptativo é possível uma distribuição do controle entre diversas máquinas de Markov adaptativas individuais, que podem ser empregadas para modelar adequadamente, cada uma, aspectos específicos de uma linguagem complexa, distribuindo assim o conhecimento descrito na gramática equivalente entre um certo número de máquinas de Markov mais simples, facilitando tanto sua implementação quanto a descrição do conhecimento acerca da linguagem modelada.

No próximo capítulo é apresentada uma aplicação dos conceitos aqui definidos ao campo da composição musical automatizada, demonstrando, assim, o potencial do formalismo adaptativo proposto.

5. Apresentação da tecnologia empregada

Neste capítulo, vamos apresentar brevemente alguns detalhes sobre os experimentos realizados nesta pesquisa com o emprego dos sistemas de Markov adaptativos, descritos no capítulo anterior.

Os experimentos foram criados como forma de demonstração prática da potência da tecnologia desenvolvida para este trabalho, com o objetivo de resolver alguns dos problemas clássicos em composição musical automatizada:

- Composição de linhas melódicas;
- Interação entre linhas melódicas independentes (acompanhamento, por exemplo);
- Realização de composições envolvendo múltiplas variáveis musicais (corais a quatro vozes).

Vamos descrever progressivamente os experimentos realizados, apresentando alguns dos resultados obtidos em cada caso. Comentários mais específicos sobre tais resultados serão apresentados no próximo capítulo. Na realização dos exemplos descritos, foi empregado o conhecimento musical derivado da teoria da música tonal, sendo que algumas das referências bibliográficas empregadas serão citadas oportunamente. A seleção do arcabouço formal a ser empregado é completamente arbitrária, mas a música tonal foi escolhida devido à sua generalidade, tornando os segmentos musicais gerados pelo sistema passíveis de avaliação tanto por especialistas como por não-especialistas em música.

5.1 Arquitetura do sistema

Todos os exemplos foram criados e executados em micro computadores com plataforma Microsoft Windows, utilizando os serviços de multimídia oferecidos por esse sistema operacional. Os programas estão

codificados na linguagem de programação “C”, acessando os serviços definidos pela biblioteca MMSYSTEM.DLL para a geração de eventos MIDI.

Essa alternativa de implementação foi preferida não por razões técnicas, mas devido à grande disponibilidade de equipamentos com estas características, fato útil para simplificar a demonstração dos experimentos. Dada a simplicidade dos programas desenvolvidos, conforme será comentado mais adiante, praticamente qualquer arquitetura, com um mínimo de recursos computacionais, poderia ter sido empregada para a realização destes experimentos.

A arquitetura do software foi a mesma para todos os exemplos descritos, e consiste na integração de três elementos de controle: o *executor das máquinas de Markov adaptativas*, que verifica os estados atuais de cada máquina, com base nas transições por ele calculadas, em função das regras programadas; o *módulo de controle*, responsável pela ativação do módulo anterior, dependente de temporização atuante sobre o sistema, sendo também responsável pela interface com o usuário; e finalmente, o *canal MIDI*, implementado pelo sistema operacional, o qual traduz em som propriamente dito os eventos requisitados pela execução das máquinas e comunicados ao módulo de controle para sua execução nos instantes de tempo adequados.

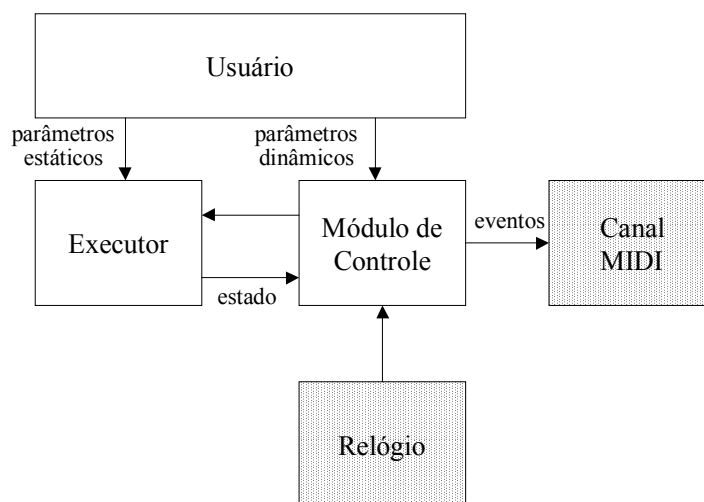


Figura 5.1 - Arquitetura do sistema implementado

Em alguns dos exemplos desenvolvidos, o usuário do sistema pode atuar, em tempo real, sobre o processo de composição, mediante uma interface gráfica com botões e barras de rolagem, alterando alguns parâmetros que virão a interferir no processo de composição, alteração esta que provocará um efeito imediato sobre a música que estiver sendo produzida pelo sistema. Desta forma, o sistema pode funcionar como um metainstrumento musical, uma vez que tais parâmetros podem estar associados a variáveis musicais abstratas, pertencentes ao domínio da semântica do compositor automático.

A figura 5.1 ilustra a arquitetura apresentada, mostrando as interações entre os módulos de programa, descritos anteriormente.

5.2 Representação dos dados

Neste item, é apresentado o modelo empregado para representação dos dados no sistema desenvolvido. Tal representação, aqui descrita, refere-se aos dados de saída do sistema, em termos de definição de eventos isolados, entendidos como alturas e durações das notas musicais individuais. O modelamento interno ao sistema, no que se refere ao modelamento de conhecimento musical, é descrito no item 5.3.

5.2.1 Representação de alturas

A representação das alturas para a saída do compositor estocástico segue o padrão MIDI, para o qual os sons gerados têm sua altura definida por números inteiros de semitons, na faixa 0-127, onde o número 57 representa a nota lá 440 Hz [FIELD97]. O número zero foi reservado internamente para representar a ausência de som (pausa).

O executor das máquinas de Markov efetua um mapeamento de seus estados internos em alturas MIDI. Note-se, entretanto, que um mesmo estado pode eventualmente acionar diversas notas MIDI. Tal mapeamento também é responsável pela seleção das escalas ou modos, sobre as quais a melodia ou melodias compostas pelo sistema estarão enquadradas. São

definidas operações sobre este mapeamento, delas resultando operações equivalentes a transposições ou modulações, pela alteração de uma dada altura de referência.

Dessa forma, internamente às máquinas de Markov adaptativas, a representação de alturas é essencialmente relativa. A saída do sistema para uma melodia, que, por exemplo, consiste em uma seqüência numérica como (0, 1, 2, 3), pode ser mapeada em (60, 62, 64, 65) , resultando na melodia dó-ré-mi-fá, ilustrada na figura 5.2.1.

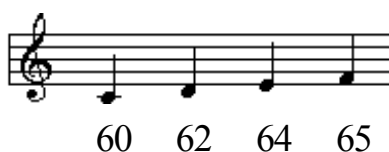


Figura 5.2.1 - Mapeamento em alturas MIDI

5.2.2 Representação de durações

A representação dos eventos, ilustrada no item anterior, não inclui a especificação das durações das notas individuais, representando apenas uma tradução momentânea do estado de determinada máquina de Markov para a forma de uma determinada altura. Tal representação, correspondente à saída do sistema, é dinâmica, produzindo um novo elemento a cada iteração da máquina, não especificando a duração absoluta do evento recém-produzido.

O controle sobre a geração de notas mais longas ou mais curtas é efetuado indiretamente pelas máquinas de Markov sobre a própria lista de saída, de forma que uma sucessão de elementos de mesma altura corresponda à permanência da máquina sobre um mesmo estado, representando um prolongamento do som anteriormente produzido. Caso uma interrupção deste seja necessária, sem que sua altura seja modificada, deve-se inserir explicitamente uma pausa para identificar esta situação.

Segundo esse modelo, cada elemento da lista de eventos, assim composta, possui uma duração fixa, a qual corresponde diretamente à

unidade métrica do sistema. Agrupando dois ou mais elementos sucessivos da lista, por possuírem a mesma altura, criam-se notas de durações maiores. Assim, supondo-se a unidade métrica como sendo a semínima, a seqüência formada por duas semínimas de igual altura vai corresponder a uma mínima. Isto é ilustrado pela figura 5.2.2.



Figura 5.2.2 - Representação de durações na lista de saída

O sistema de controle das máquinas de Markov mantém informação dinâmica sobre o tempo métrico, de forma que informações quanto à fórmula de compasso e disposição dos tempos fortes e fracos no interior do compasso podem ser relevantes para o sistema. São permitidas síncopas, desde que a máquina responsável pela seleção da nota da melodia mantenha-se sobre o mesmo estado entre o último tempo de um compasso e o primeiro tempo do compasso seguinte.

A unidade métrica não precisa ser necessariamente igual à unidade de pulsação ou tempo da composição gerada, de forma a permitir subdivisões da unidade de tempo em unidades menores, desde que estas sejam, evidentemente, múltiplas de uma unidade métrica fundamental. Por este princípio, divisões e ritmos compostos são possíveis de serem obtidos.

5.3 Estrutura dos Sistemas de Markov Adaptativos

O modelo de representação do conhecimento musical, empregado no sistema de composição automatizada aqui descrito, consiste na especificação de um conjunto de máquinas de Markov adaptativas especializadas, cada uma representando algum aspecto teórico (que pode ser descrito formalmente através de uma gramática equivalente)

razoavelmente independente, e das interrelações adaptativas entre tais máquinas.

Conforme definida no capítulo anterior, uma máquina de Markov adaptativa pode executar ações adaptativas quando da ocorrência de suas transições, ações estas que podem alterar a topologia geral da própria máquina ou de outras máquinas que componham o conjunto. As ações adaptativas básicas permitem a modificação das relações de probabilidade de uma transição ou de um conjunto de transições de estados da própria máquina ou de outras máquinas que pertençam ao conjunto.

A execução das funções adaptativas vai permitir a uma dada máquina de Markov a alteração de seu próprio funcionamento (algoritmo) ou a interferência direta sobre o funcionamento de outras máquinas do conjunto, podendo ser utilizadas para induzir ou forçar um determinado comportamento, para inibir, ativar ou reativar a execução de outras máquinas e assim por diante. Dessa forma, mesmo com um conjunto pequeno de máquinas de Markov adaptativas pode-se obter um comportamento extremamente complexo para o sistema.

De modo geral, podem ser definidas três formas básicas de interação entre duas máquinas de Markov adaptativas, o *controle hierárquico*, a *adaptação* e o *aprendizado*.

O *controle hierárquico* é a forma mais importante de relacionamento entre máquinas, representando o controle de uma máquina de Markov por outra, de nível semântico superior. Assim, o comportamento de uma dada máquina de Markov pode estar condicionado pelo estado de máquinas mais gerais, o qual modela elementos lingüísticos pertencentes a um nível semântico mais elevado da composição. Um exemplo disto está no condicionamento das regras de movimentação melódica, definidas por uma gramática própria, a regras gramaticais que vão definir progressões harmônicas válidas. A figura 5.3.1 ilustra este tipo de relacionamento entre máquinas de Markov adaptativas, em que o comportamento da

Máquina 2 depende do comportamento da Máquina 1, de hierarquia semântica superior.

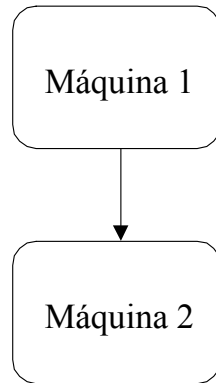


Figura 5.3.1 - Relação hierárquica entre duas máquinas de Markov adaptativas

A *adaptação* consiste na interferência mútua entre duas máquinas, cuja operação pode ser considerada como conflitante ou inconsistente em algum sentido, com o objetivo de se obterem soluções comuns, com um resultado global satisfatório. Esta interferência pode, por exemplo, evitar que uma máquina transite para determinados estados, que sejam considerados conflitantes com o estado corrente de uma outra máquina. A adaptação ocorre entre máquinas de Markov adaptativas que estão no mesmo nível semântico, fazendo-se desnecessária a elevação deste nível com a finalidade de contemplar um conhecimento adicional, o qual seria necessário para que estes conflitos fossem resolvidos no âmbito de uma única máquina de Markov, que certamente exibiria uma configuração muito mais complexa. Um exemplo prático interessante desta forma de interação é a acomodação de vozes, cujas melodias são geradas por máquinas independentes, de maneira que sejam evitados cruzamentos entre vozes e o aparecimento de intervalos harmônicos dissonantes indesejáveis. A figura 5.3.2 ilustra a relação de adaptação entre duas máquinas de Markov adaptativas no mesmo nível semântico.

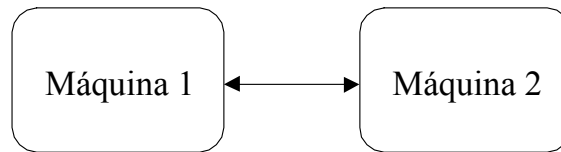


Figura 5.3.2 - Relação de adaptação entre duas máquinas de Markov adaptativas situadas no mesmo nível semântico

O *aprendizado* permite que o funcionamento de uma máquina seja monitorado por outra, o que pode vir a interferir no seu próprio funcionamento ou na forma de controle que tal máquina exerce sobre outras. A imitação de uma dada seqüência de estados, por exemplo, pode ser uma forma simples de introdução de ciclos no comportamento de uma máquina. Tornando mais ou menos rígido o controle sobre tais ciclos, é possível obter comportamentos semelhantes à imitação e à variação, respectivamente. A figura 5.3.3 ilustra a relação de aprendizado, no qual a Máquina 1 monitora ou supervisiona o funcionamento da Máquina 2, situada em nível semântico inferior.

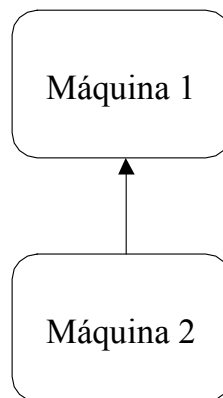


Figura 5.3.3 - Relação de aprendizado entre duas máquinas de Markov adaptativas

A utilização de máquinas de Markov adaptativas independentes em um sistema de Markov pode contribuir muito para a simplificação do processo de descrição do conhecimento musical necessário para a resolução do problema da composição musical automatizada, uma vez que cada aspecto isolado deste conhecimento pode ser modelado por meio de

uma gramática independente, posteriormente convertida em uma ou mais máquinas de Markov adaptativas. A integração de tais máquinas, e conseqüentemente, do conhecimento musical descrito pelas gramáticas originais, será efetuada pelo sistema. O resultado final vai corresponder aproximadamente à soma dos conhecimentos individuais. Com essa idéia, o conhecimento necessário à escrita de melodias pode ser tratado de forma razoavelmente independente do conhecimento sobre harmonia e tonalidade, sobre forma e estrutura e assim por diante.

Com a utilização desse paradigma, pode-se visualizar este sistema como uma forma eficiente de implementação de gramáticas multi-dimensionais, citadas na literatura como mecanismos interessantes para a descrição de conhecimento musical formal [ROADS79].

5.4 Descrição do Conhecimento Musical empregado

Os itens seguintes descrevem brevemente o modelamento empregado para a representação do conhecimento musical dos programas-exemplo, buscando a solução para os problemas clássicos mencionados no início deste capítulo.

5.4.1 Linha melódica

Para a definição de uma gramática para a composição de linhas melódicas, estabeleceremos, antes de mais nada, um conjunto de símbolos que vão compor a linguagem final. Esses símbolos vão representar, no caso, notas musicais isoladas a serem escolhidas para comporem as sentenças geradas por esta gramática.

$$\Sigma = \{ c, d, e, f, g, a, b, c_2 \}$$

Assumimos aqui como símbolos terminais as letras empregadas para a designação das notas musicais no sistema anglo-saxônico (c = dó, d = ré, etc). Como estamos utilizando uma oitava completa, o símbolo c_2 refere-se

ao dó, uma oitava acima da nota de mesmo nome, representada pelo símbolo c .

Uma vez definidos esses símbolos terminais, já foi resolvido o problema de estabelecer uma *escala*, já que qualquer melodia produzida segundo a aplicação de regras da gramática vai conter exclusivamente símbolos pertencentes a essa escala (dó maior, no exemplo).

Podemos definir uma linguagem para a linha melódica como sendo qualquer seqüência de elementos do conjunto de terminais assim definido; uma possível gramática geradora dessa linguagem poderia ser:

$$\begin{aligned} R &\rightarrow N \\ N &\rightarrow cN \\ N &\rightarrow dN \\ N &\rightarrow eN \\ N &\rightarrow fN \\ N &\rightarrow gN \\ N &\rightarrow aN \\ N &\rightarrow bN \\ N &\rightarrow c_2N \\ N &\rightarrow \varepsilon \end{aligned}$$

Nessa gramática, R representa o não-terminal inicial ou raiz da gramática. Pode-se observar que essa gramática produz sentenças contendo uma seqüência qualquer dos símbolos a , b , c , c_2 , d , e , f e g em qualquer ordem.

Uma máquina de Markov muito simples, capaz de produzir a mesma linguagem gerada por essa gramática, apresenta oito estados, todos identificados como estados finais, referentes a cada um dos símbolos terminais, com transições de probabilidade $1/8$ para qualquer estado, incluída a transição para o mesmo estado. A figura 5.4.1.1 ilustra essa máquina, representando as transições que partem do estado “ c ”. Transições similares partem também de cada um dos demais estados (não representado na figura).

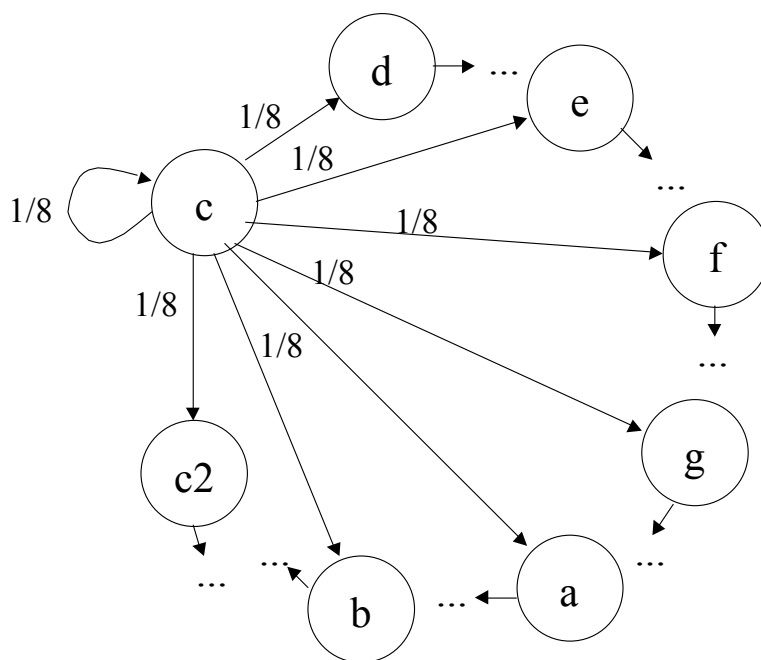


Figura 5.4.1.1 - Máquina de Markov para a composição de seqüências aleatórias dos terminais c, d, e, f, g, a, b e c₂ representando apenas as transições com origem no estado "c"

Um exemplo de resultado obtido com essa máquina individual é apresentado na figura 5.4.1.2, que ilustra a sentença *aeedcebdegfc₂dc₂gf*, produzida por uma seqüência de transições de estados da máquina descrita anteriormente. O agrupamento de símbolos em compassos individuais foi definido arbitrariamente.



Figura 5.4.1.2 - Linha melódica composta pela máquina da figura 5.4.1.1

Pode-se concluir imediatamente que, embora os elementos que formam a melodia em questão pertençam a uma escala única, o trecho é desprovido de qualquer elemento organizador e a aleatoriedade interna a esse fragmento musical pode ser associada ao ruído.

Podemos definir uma outra linguagem, para a solução do problema através de outra gramática, empregando certo conhecimento musical sobre linhas melódicas, com o objetivo de obter um resultado mais consistente, eliminando parte da imprevisibilidade do exemplo anterior, e, portanto, reduzindo o efeito de ruído produzido.

Na escrita de linhas melódicas, em geral uma nota sucede uma nota vizinha, de forma tal que o movimento das alturas em seu interior seja gradativo. Podemos escrever uma gramática, dependente de contexto, que restrinja a sucessão dos símbolos a elementos vizinhos, como a seguinte:

$cN \rightarrow cdN$	$eN \rightarrow eeN$	$aN \rightarrow abN$
$cN \rightarrow cbN$	$fN \rightarrow feN$	$an \rightarrow aaN$
$cN \rightarrow ccN$	$fN \rightarrow fgN$	$bN \rightarrow baN$
$dN \rightarrow dcN$	$fN \rightarrow ffN$	$bN \rightarrow bcN$
$dN \rightarrow deN$	$gN \rightarrow gfN$	$bN \rightarrow bbN$
$dN \rightarrow ddN$	$gN \rightarrow gaN$	$c_2N \rightarrow bc_2N$
$eN \rightarrow edN$	$gN \rightarrow ggN$	$c_2N \rightarrow c_2N$
$eN \rightarrow efN$	$aN \rightarrow agN$	

e mais as regras $R \rightarrow cN$ e $N \rightarrow \varepsilon$. R é o símbolo inicial.

Segundo esse sistema de produção, apenas são possíveis progressões nas quais o símbolo, que venha em seqüência, seja vizinho na escala de origem, ou seja, apenas permite que intervalos de segundas (maiores ou menores, dependendo do estado) estejam presentes na melodia composta. Note-se que a repetição do mesmo símbolo é permitida. Dessa forma, fazem parte da linguagem gerada pela gramática anterior as sentenças *cdedeeefgffgab*, *dddedcbaab* e *cdefgab*, mas não fazem parte as sentenças *abcfg*, *acg* e *abce*.

Uma possível linha melódica gerada por esta gramática está exemplificada a seguir (figura 5.4.2.3), onde novamente impusemos um agrupamento arbitrário de quatro elementos para formar um compasso.



Figura 5.4.2.3 - Linha melódica composta, com restrição de saltos sobre notas vizinhas

Desse exemplo pode-se concluir que a melodia produzida exibe uma consistência melódica bem superior à anterior. No entanto, linhas melódicas reais podem apresentar saltos, os quais foram excluídos da gramática assim definida.

Podemos adicionar os saltos à linguagem melódica que definimos, por exemplo, inserindo na gramática apresentada anteriormente as produções seguintes:

$$\begin{aligned} N &\rightarrow S \\ S &\rightarrow cN \\ S &\rightarrow dN \\ S &\rightarrow eN \\ S &\rightarrow fN \\ S &\rightarrow gN \\ S &\rightarrow aN \\ S &\rightarrow bN \\ S &\rightarrow c_2N \end{aligned}$$

Pode-se observar que essa nova gramática permite a ocorrência de saltos, desde que haja a substituição do não-terminal N pelo não-terminal S, o que permitirá que uma nota qualquer seja substituída em seguida.

É simples notar que essa última gramática gera exatamente a mesma linguagem que a gramática do início, onde qualquer seqüência de terminais pertence, igualmente, à linguagem. Dessa forma, podemos empregar para a sua implementação uma máquina de Markov com exatamente a mesma topologia daquela representada na figura 5.4.1.1.

A presença de dois grupos separados de regras caracteriza e evidencia dois conceitos diferentes, pertencentes ao conhecimento musical

sobre linhas melódicas: a seqüência de graus conjuntos e os saltos. A máquina de Markov pode ser implementada para efetuar a opção por estados vizinhos ou por saltos, segundo uma relação de probabilidade. Em outras palavras, se considerarmos que o número de saltos em uma linha melódica é estatisticamente inferior ao de trechos sobre escalas, então, traduzindo este fato nas ações adaptativas da máquina implementada, que passam a elevar a probabilidade das transições conduzindo a estados vizinhos, poderemos representar adequadamente esta idéia, de forma simples e transparente.

O exemplo a seguir ilustra esta idéia. Na figura 5.4.1.4a supomos a ocorrência da transição entre os estados “d” e “e” da máquina de Markov adaptativa:

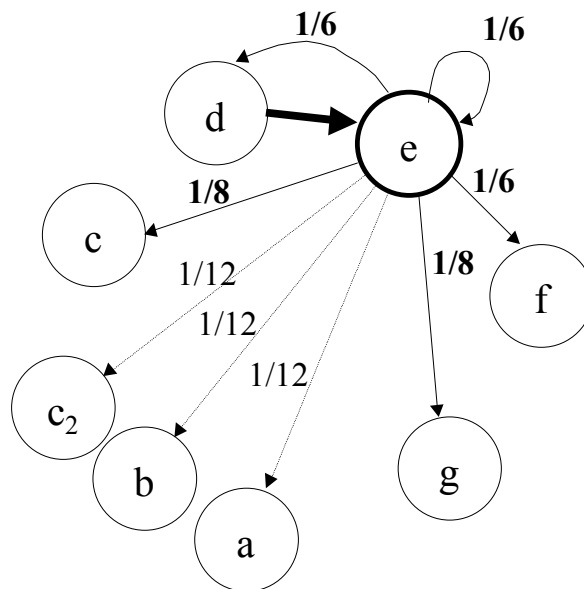


Figura 5.4.1.4a - Reconfiguração da máquina de Markov adaptativa

Associada a esta transição, há a ação adaptativa responsável pela elevação das probabilidades de ocorrência das transições que conduzem a estados vizinhos (de “e” para “d”, de “e” para “f”, etc), reduzindo a probabilidade de ocorrência de saltos (para os estados “c2”, “b” ou “a”). Por outro lado, supondo a ocorrência de um salto, conforme a figura 5.4.1.4b,

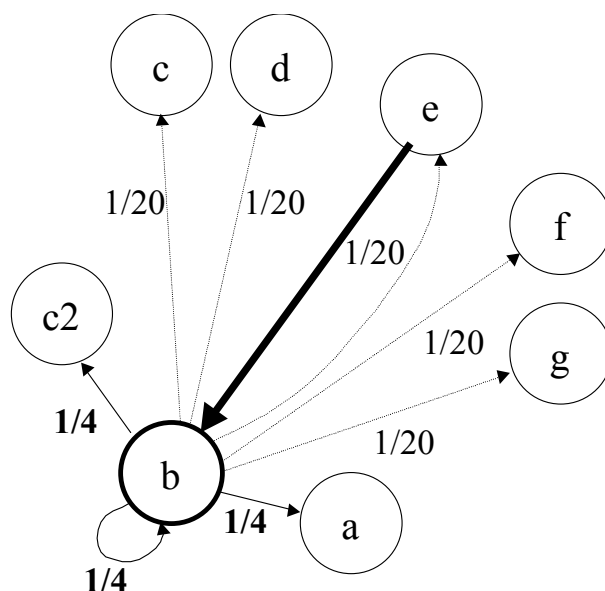


Figura 5.4.1.4b - Reconfiguração da máquina de Markov após um salto

Nesse caso, uma ação adaptativa reconfigura a máquina de forma diferente, com o objetivo de restringir a ocorrência de um novo salto, dando maior preferência estatística a transições para estados próximos (de “b” para “c2”, por exemplo).

Um exemplo de melodia obtida dessa forma está ilustrado a seguir, na figura 5.4.1.5.



Figura 5.4.1.5 - Linha melódica composta pela máquina de Markov adaptativa com restrição de intervalos

Outras características do conhecimento musical podem ser adicionadas à gramática que descrevemos, como por exemplo a identificação de seqüências intervalares, cujo resultado audível não seja satisfatório, saltos dissonantes, simples ou compostos, como os intervalos diminutos ou aumentados, e em alguns casos as sétimas, de forma que não seja permitida a sua escolha pela máquina de Markov que gera a linha melódica. O número de elementos em seqüências individuais, que não

incluam saltos, também pode ser limitado, para que a composição de seqüências escalares monótonas seja evitada.

5.4.2 *Comportamento tonal*

A análise do exemplo musical, obtido no item anterior para a geração de linhas melódicas, pode revelar que, embora se trate de um resultado interessante, este não apresenta qualquer ordem ou estrutura interna, o que, para frases musicais relativamente longas, pode produzir novamente a sensação de ruído.

Com o objetivo de garantir um maior nível de estruturação musical nos segmentos musicais gerados pelo programa, podemos pensar inicialmente em introduzir nas frases musicais compostas uma certa consistência tonal, que contribuiria, por exemplo, para o estabelecimento de limites de início e término de seqüências.

Da teoria da tonalidade (vide, por exemplo, [SCHOENBERG74]), verificamos que cada elemento da escala pode estar relacionado a uma determinada *função harmônica*, e que a sucessão de tais funções harmônicas segue um padrão sintático bem definido. Existem três funções harmônicas, a saber: a *tônica*, que apresenta um caráter de repouso, e as outras duas, a *dominante* e a *subdominante*, que se opõem à tônica, servindo para o estabelecimento de movimento.

Segundo a teoria harmônica tradicional, um segmento musical qualquer deve iniciar-se com a função de tônica (assim identificada naturalmente pelo ouvido), afastando-se em seguida desta função, em direção às outras funções, e finalmente retornando à tônica, o que conclui harmonicamente a passagem. Podemos chamar de *período* um encadeamento completo dessas funções, conforme apresentado.

Dessa maneira, um encadeamento de funções harmônicas, sintaticamente correto, pode apresentar os seguintes padrões: $T \rightarrow D \rightarrow T$, $T \rightarrow S \rightarrow T$ e $T \rightarrow S \rightarrow D \rightarrow T$, onde as letras T, D e S representam as funções harmônicas definidas, tônica, dominante e subdominante,

respectivamente. O último encadeamento apresentado inclui todas as três funções harmônicas, e por isso ganha importância por apresentar a tonalidade com o maior grau possível de clareza.

Podemos, com base no que foi apresentado, definir uma gramática para a geração de períodos musicais com sintática tonal, cujos símbolos terminais são as funções harmônicas definidas,

$$\begin{aligned} N &\rightarrow TDTN \\ N &\rightarrow TSTN \\ N &\rightarrow TSDTN \\ N &\rightarrow \varepsilon \\ T &\rightarrow tT \\ D &\rightarrow dD \\ S &\rightarrow sS \\ T &\rightarrow t \\ D &\rightarrow d \\ S &\rightarrow s \end{aligned}$$

Incluindo o não-terminal R , a raiz da gramática, e a produção $R \rightarrow N$, considerando que os símbolos terminais t , s e d representam a tônica, subdominante e dominante, respectivamente, temos uma gramática que gera progressões de funções harmônicas coerentes com o que foi descrito nos parágrafos anteriores. Observe-se que as seqüências geradas por tal gramática não correspondem a elementos musicais imediatos, mas a símbolos pertencentes a uma metalinguagem musical, comentada anteriormente neste texto.

Uma configuração possível para uma máquina de Markov adaptativa para a geração desta linguagem é exemplificada a seguir, na figura 5.4.2.1. O estado marcado com a letra “ t ” corresponde a um estado final.

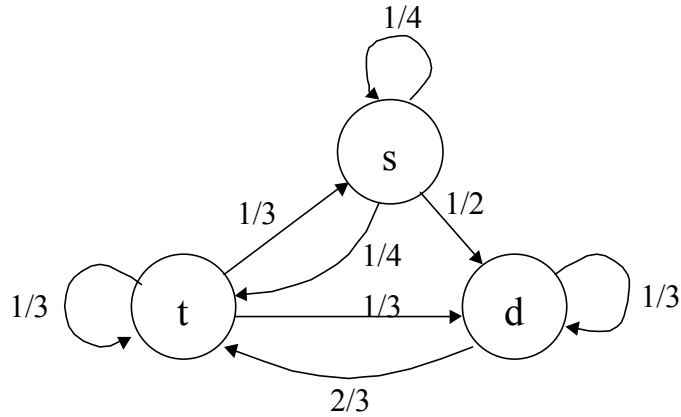


Figura 5.4.2.1 -Máquina de Markov para a geração de sintaxe tonal

O leitor pode observar que a distribuição de probabilidades para a transição dos estados desta máquina foi escolhida de forma a privilegiar o aparecimento de todas as funções harmônicas, quando da seleção de um período envolvendo a função subdominante.

Vamos agora estabelecer relações entre as notas musicais isoladas e as funções harmônicas definidas, com a finalidade de combinar a operação das duas máquinas de Markov, de forma a obtermos a composição de uma linha melódica com características tonais.

Para tanto, vamos explorar a relação que é estabelecida entre o primeiro grau da escala e a função tônica, entre o quinto grau e a função dominante e entre o quarto grau e a função subdominante. Para relacionarmos as demais notas da escala, faremos uso das relações de terças (formação de acordes). Nossa hipótese está representada na figura seguinte:

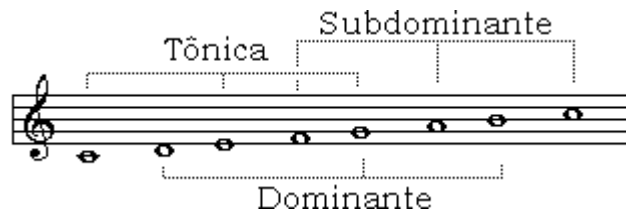


Figura 5.4.2.2 - Associação entre graus e funções harmônicas

Estamos relacionando os elementos do acorde de tônica, {c, e, g, c₂}, à função tônica, de símbolo t; os elementos do acorde de subdominante, {f, a, c, c₂}, ao símbolo s e, analogamente, os elementos do acorde de dominante, {g, b, d} ao símbolo d. Observe-se que a união destes conjuntos resulta no conjunto de terminais da gramática, ainda que certos elementos estejam repetidos, o que significa que tais elementos podem estar relacionados com funções harmônicas diferentes (contexto).

Vamos impor um relacionamento hierárquico entre a máquina de Markov que define as sentenças sintaticamente corretas de funções harmônicas e a máquina definida no item anterior. Ações adaptativas serão definidas nas transições da máquina de Markov que estabelece a função harmônica corrente, as quais vão impor alterações na topologia da máquina responsável pela composição da linha melódica. Assim, no exemplo da figura 5.4.2.3, o contexto imposto pelo estado “s” da máquina responsável pela harmonia, vai interferir nas transições partindo do estado corrente (“e”) e conduzindo aos estados correspondentes ao acorde de subdominante (“f”, “a” e “c”).

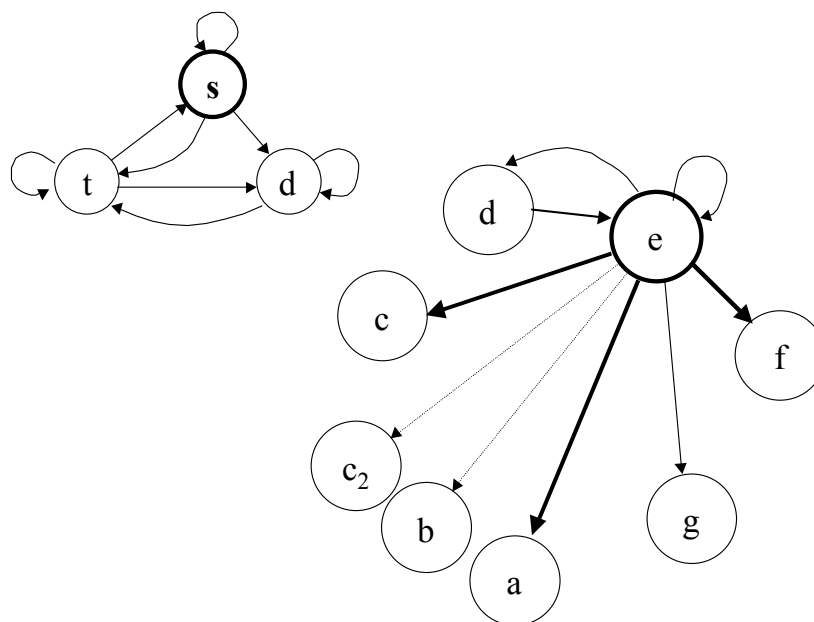


Figura 5.4.2.3 - Interação entre as duas máquinas de Markov

Em um primeiro momento, tais ações adaptativas excluirão as transições que conduzam a estados que não estejam relacionados à função harmônica (estado corrente da máquina de nível semântico mais elevado). Como resultado da execução dessa configuração de máquinas de Markov, podemos obter uma seqüência de acordes arpejados como linha melódica, conforme está ilustrado pela figura 5.4.2.4.



Figura 5.4.2.4 - Resultado da execução da combinação das máquinas anteriores

onde as letras denotam o estado da máquina adaptativa geradora de função harmônica, que, no exemplo, está produzindo um novo símbolo a cada quatro transições (um compasso) da máquina de produção de melodia.

Nesse exemplo, as características melódicas desapareceram para dar lugar a uma linguagem essencialmente harmônica. Embora isso possa ser interessante para a produção de linhas com função de acompanhamento, em um período melódico usualmente não estão presentes apenas acordes.

Alterando as ações adaptativas da máquina responsável pela função harmônica sobre a máquina geradora da linha melódica para, ao invés de eliminar transições, apenas reduzir as probabilidades de ocorrência associadas às transições que conduzem a estados não relacionados à função harmônica corrente, podemos obter um resultado com caráter melódico bem mais interessante. Vamos adicionar, ainda, outro grau de dependência de contexto, aumentando a probabilidade de ser atingido um estado correspondente a uma nota do acorde durante os tempos fortes do compasso (primeiro e terceiro).

Um possível resultado desta técnica é apresentado na figura 5.4.2.5.



Figura 5.4.2.5 - Combinação dos comportamentos melódico e harmônico

Observa-se, nesse exemplo, um comportamento musical muito mais coerente, e ouvintes humanos podem facilmente reconhecer uma estruturação tonal que permite identificar esse fragmento como musicalmente completo. É interessante observar o aparecimento de notas de passagem nos tempos fracos, em decorrência do relaxamento das suas probabilidades de ocorrência, conforme descrito anteriormente, combinado com a operação normal da máquina responsável pela linha melódica, evitando a ocorrência de saltos.

Nos exemplos criados para demonstração, foram incluídas ainda outras propriedades referentes à sintática da música tonal, como por exemplo a relação de sensível, elevando a probabilidade de ocorrência do movimento ascendente de segunda menor, característico das cadências perfeitas, quando esta nota aparece em contexto de função harmônica de dominante.

Podemos exemplificar a interação hierárquica entre as máquinas de Markov deste item com a figura 5.4.2.6.

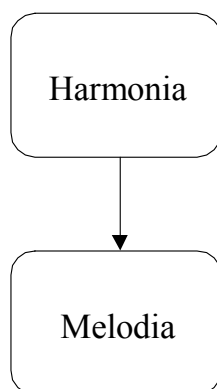


Figura 5.4.2.6 - Interação entre as máquinas de Markov

5.4.3 *Comportamento temático*

O exemplo obtido no item anterior mostra que o nível de estruturação imposto à linha melódica, pela adição de relações de tonalidade sobre os graus da escala, é capaz de produzir resultados que podem mais facilmente ser interpretados como sendo música. No entanto, outras formas de estruturação musical, igualmente importantes, estão ausentes.

Embora a sucessão consistente de elementos harmônicos contribua para a produção de seqüências musicais que, do ponto de vista tonal, aparentam ter um início e uma conclusão, as seqüências compostas usando apenas esse nível de estruturação apresentam pouca redundância interna, o que pode ser percebido pelo ouvinte após a análise de um número suficiente de compassos produzidos.

Para que uma composição seja compreensível, além de relações que estabeleçam os limites de um discurso harmônico (no caso da música tonal), usualmente são incluídos elementos que são deliberadamente repetidos ao longo do tempo [SCHOENBERG94], redundância esta que é essencial para o estabelecimento de uma consistência interna para a obra, denominada *estrutura* ou *forma*.

Para a introdução de tal redundância na linha melódica vamos empregar o conceito de *tema* ou *motivo*, conforme definido na literatura musical [SCHOENBERG93][SCHOENBERG94]. Há duas propriedades importantes que definem este conceito. A primeira especifica que um tema é sempre repetido, de forma literal ou ligeiramente variada, aparecendo várias vezes em instantes diferentes, e a segunda aponta o tema como uma unidade que pode ser identificada de forma clara e relativamente inequívoca, ainda que se trate de uma variação.

Essas duas propriedades estão relacionadas entre si, uma vez que a repetição do tema é uma das maneiras pelas quais ele pode ser identificado. Para que o ouvinte possa identificar a repetição do tema é

necessário que ele tenha condições de memorizá-lo, separando-o dos demais elementos presentes na composição.

Além da repetição, que permite ao ouvinte identificar a presença de padrões recorrentes e atribuir-lhes a função de temas, determinadas características da composição permitem ao ouvinte o agrupamento de elementos musicais, com a possível associação com o tema. Estas características, conforme apresentadas em [LERDAHL83], relacionam-se com a teoria psicológica da Gestalt, afirmando que elementos são agrupados pelo ouvido em função de sua similaridade e pelo estabelecimento de fronteiras com elementos contrastantes.

Os padrões rítmicos são os mais eficazes como formas temáticas, por serem facilmente identificáveis. Associar, por exemplo, um número de colcheias em um tema é simples, devido à similaridade rítmica de seus elementos, desde que existam fatores contrastantes, que venham a servir como delimitadores, como por exemplo, pausas ou sons cujas durações sejam muito diferentes.

O uso de padrões melódicos como células temáticas também é possível. Normalmente, os elementos melódicos que são considerados similares encontram-se próximos entre si, por exemplo, sobre uma dada escala. Saltos podem ser considerados como elementos contrastantes e, portanto, delimitadores da célula temática. A direção do movimento melódico pode ser facilmente reconhecida como elemento temático, por ser de fácil memorização, independentemente da natureza dos intervalos que compõem a passagem. Dessa forma, uma determinada seqüência de movimentos em direções definidas pode ser reconhecida como um tema.

Uma vez estabelecido um número de células temáticas, estas são organizadas entre si em determinada seqüência, estabelecendo a estrutura do período musical ou mesmo da composição completa. Os mecanismos para a definição de tal seqüência estão condicionados por padrões estéticos que vão caracterizar redundância, contraste e variação ao longo da peça musical. Excessiva redundância pode tornar a obra monótona e

previsível, diluindo o interesse do ouvinte. Por outro lado, o excesso de contraste entre elementos sucessivos pode fazer com que a obra perca sua consistência, aumentando a sensação de ruído para o ouvinte.

Com a finalidade de melhorar as composições produzidas automaticamente pelo sistema, aumentaremos o grau de redundância das seqüências melódicas produzidas, mediante a introdução de elementos repetitivos, que passarão a funcionar como células temáticas quando da especificação de seqüências de temas.

A nova máquina de Markov adaptativa define elementos temáticos pela seleção da direção do movimento melódico e pela definição de padrões rítmicos, mediante o controle de execução da máquina geradora de melodia, através de ações adaptativas entre as máquinas.

Um tema pode ser definido em uma gramática através de uma produção específica, conforme ilustrado nas regras seguintes:

$$\begin{aligned} A &\rightarrow ab \\ B &\rightarrow idi \end{aligned}$$

onde empregamos os símbolos terminais para identificar formas de movimentação melódica específicas:

- a = movimento melódico ascendente, intervalo curto
- b = movimento melódico descendente, intervalo grande (salto)
- d = movimento melódico descendente, intervalo curto
- i = ausência de movimento melódico (duplicação da duração da nota)

É bastante simples a criação de uma máquina de Markov adaptativa que gere a mesma linguagem desta gramática. A relação entre tal máquina e a máquina de composição da linha melódica deve ser estabelecida por funções adaptativas que correspondam à realização do movimento melódico desejado, conforme descrito no parágrafo anterior. A seleção do símbolo a, por exemplo, pode condicionar a máquina de geração da linha

melódica, através de ações adaptativas específicas, a direcionar sua próxima transição a um estado próximo, correspondendo a um intervalo ascendente de segunda ou terça, reduzindo a preferência estatística pelos demais estados. A seleção do símbolo “i”, por outro lado, suspende a execução da máquina responsável pela melodia por uma unidade de tempo, permanecendo sobre seu último estado, o que corresponde à duplicação da duração da nota gerada. Pela aplicação de ações adaptativas como essas, podemos obter, para a linha melódica, os comportamentos ilustrados na figura 5.4.3.1, quando da seleção da seqüência *ab* (não-terminal A da gramática):



Figura 5.4.3.1 - Melodias obtidas para a seqüência *ab*

Ou ainda, para a seqüência *idi* (não-terminal B da gramática):

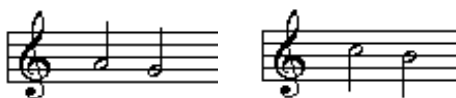


Figura 5.4.3.2 - Melodias obtidas para a seqüência *idi*

Observa-se que tais seqüências melódicas podem ser efetivamente reconhecidas como células temáticas no interior de linhas melódicas, desde que seja estabelecida a recorrência dos não-terminais A e B nas formas sentenciais obtidas da gramática correspondente ao comportamento temático, e garantida pelo contraste, tanto melódico quanto rítmico, entre tais seqüências.

Podemos introduzir na gramática referente ao comportamento temático uma produção do tipo

$$N \rightarrow AABA$$

e após construída a máquina adaptativa correspondente, obtemos como resultado a possível linha melódica da figura 5.4.3.3.



Figura 5.4.3.3 - Linha melódica obtida pela indução de células temáticas

Desse resultado observa-se um grau de redundância bem superior aos anteriores, com uma sensível redução do nível de ruído, de forma a apresentar uma estrutura que pode ser reconhecida como propriamente musical, com temas identificáveis. Regras gramaticais específicas podem ser escritas diretamente para que sejam criados períodos musicais completos, como no exemplo anterior, e, mediante uma gramática livre de contexto, permitindo aninhamentos, criar formas mais amplas, envolvendo um número maior de períodos e temas compostos.

A figura 5.4.3.4 ilustra a interação entre as máquinas que foram definidas até o momento.

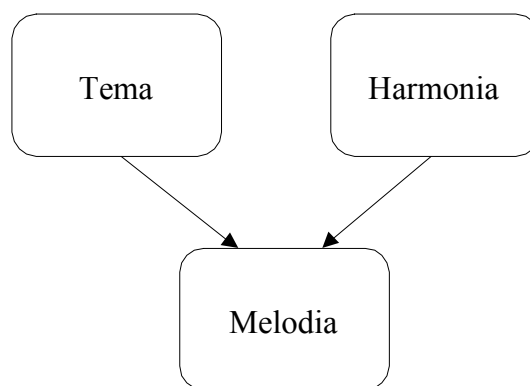


Figura 5.4.3.4 - Interação entre as máquinas de Markov

É possível, ainda, combinar a operação das duas últimas máquinas de Markov adaptativas que definimos, em um esquema de adaptação, para obtermos, por exemplo, uma correlação entre o comportamento tonal e o

comportamento temático, na criação de períodos musicais mais complexos.

Podemos observar que o comportamento obtido para o sistema até o momento é bastante sofisticado, mas que, no entanto, foi obtido segundo um modelamento muito simples e transparente, o que evidencia o potencial dos formalismos adaptativos que estamos empregando como mecanismos de descrição do conhecimento musical.

5.4.4 Interação entre linhas melódicas independentes

Procurando resolver problemas musicais mais complexos do que a produção de linhas melódicas individuais, vamos apresentar a interação de máquinas de Markov adaptativas responsáveis pela geração de linhas melódicas independentes. O primeiro caso, mais simples, envolvendo duas linhas melódicas em estilo melodia-acompanhamento, é resolvido com base em ações adaptativas entre máquinas semelhantes, no mesmo nível semântico, que se adaptam mutuamente, em função de restrições de contraponto. O segundo exemplo prescreve uma máquina de Markov, de hierarquia mais elevada, com essa responsabilidade, e será descrito em 5.4.5.

A execução simultânea de duas máquinas de Markov idênticas, responsáveis pela geração das linhas melódicas individuais, pode incorrer diretamente em situações de conflito, cujo resultado, em uma primeira análise, poderia ser a produção de intervalos harmônicos dissonantes.

Para evitar a ocorrência de tais dissonâncias, podem ser estabelecidas relações de adaptação entre as máquinas responsáveis pelas linhas melódicas em contraponto, de forma que o estado selecionado para uma das máquinas interfira diretamente na preferência estatística por determinados estados da outra máquina, considerados inconsistentes com relação a tal estado. Segundo a teoria do contraponto [TRAGTENBERG94], vamos considerar como dissonâncias os intervalos de segundas (maiores ou menores), quartas (justas ou aumentadas) e sétimas (maiores ou

menores), de maneira que as inconsistências entre os estados das máquinas serão consideradas como aquelas combinações que levem à produção de tais intervalos na música composta pelo sistema.

No entanto, deve-se observar que a presença de dissonâncias é interessante e mesmo recomendável em uma composição musical. É, portanto, necessário um certo controle sobre a presença e resolução de tais dissonâncias pelo sistema. Como uma solução simples, vamos impor uma restrição mais rígida ao aparecimento de uma dissonância nos tempos fortes (primeiro e terceiro tempos), relaxando tal restrição nos tempos fracos (segundo e quarto tempos). Além disso, restringimos a sucessão de duas dissonâncias, o que corresponderia à ausência de uma resolução.

Este algoritmo, correspondente a um conhecimento específico sobre contraponto, é implementado exclusivamente por meio de ações adaptativas entre as máquinas responsáveis pela composição das melodias individuais, agora reconhecidas como *vozes*. Outros aspectos desse conhecimento, como a correta resolução das dissonâncias e a limitação da ocorrência de movimentos paralelos, são mais convenientes de serem representados através de gramática própria, em nível semântico mais elevado.

Considerando que o comportamento tonal já foi definido anteriormente e implementado por máquina de Markov adaptativa específica, tal máquina passa a ter influência direta sobre ambas as máquinas de geração de melodias. Podemos impor uma influência semântica mais rígida sobre a máquina que implementa o baixo (voz mais grave, geralmente com uma função característica de acompanhamento), tornando mais clara a harmonia de todo o conjunto. Nota-se que, ao serem estabelecidas as relações adaptativas entre as três máquinas, ocorre a nítida preferência por intervalos de terça e sexta, correspondentes aos acordes referentes às funções harmônicas selecionadas pela máquina de controle da harmonia da composição.

Um exemplo de resultado empregando essa técnica é ilustrado pela figura 5.4.4.1, onde a linha do baixo foi arbitrariamente criada com um movimento mais lento em relação à linha melódica principal.



Figura 5.4.4.1 - Contraponto de duas melodias

A disposição das máquinas de Markov até este momento é esquematizada na figura 5.4.4.2.

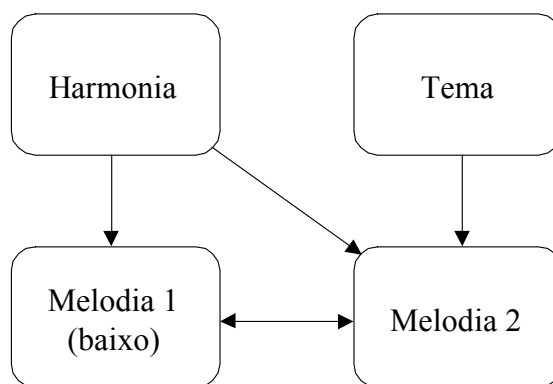


Figura 5.4.4.2 - Interação de máquinas de Markov

5.4.5 Interação de várias vozes

O exemplo anterior nos mostra a possibilidade de adaptação entre duas máquinas de Markov responsáveis pela composição de linhas melódicas (ou *vozes*) independentes, de forma que sejam estabelecidas relações de contraponto. No entanto, com a finalidade de se obterem composições mais sofisticadas, envolvendo um número maior de melodias soando em conjunto, como ocorre no problema dos corais a quatro vozes, torna-se necessário o uso de mecanismos de nível semântico mais elevado.

O tratamento dado pelas máquinas de Markov individuais, responsáveis pela produção de suas respectivas linhas melódicas,

corresponde a um conhecimento local sobre o estado atual e os possíveis próximos estados, que podem ser assumidos sem incorrer em inconsistência com a máquina responsável pela outra melodia em contraponto. Para conseguirmos um resultado convincente, que venha a envolver um conjunto mais complexo de máquinas adaptativas, faz-se necessário estabelecer relações de contexto, que podem depender de diversas máquinas a um tempo, bem como de informação referente ao comportamento de todo o conjunto, durante um determinado número de transições.

Na teoria musical clássica [TRAGTENBERG94], há uma série de circunstâncias de contexto que devem ser evitadas ao se escreverem partes independentes que devem soar em conjunto. Em particular, conforme já delineado no item anterior, o uso e resolução de determinados intervalos, denominados dissonantes, deve ser tratado com critério especial, de forma que a combinação de todas as linhas melódicas envolvidas seja harmoniosa, produzindo um resultado acústico agradável.

O grande objetivo da teoria do contraponto, além de manter um número de vozes soando harmoniosamente em conjunto, é manter e afirmar a independência melódica de cada voz como linha melódica individual. Para tanto, é necessário que as linhas possam ser distinguidas e acompanhadas, uma a uma, pelo ouvinte, sem serem confundidas umas com as outras, em prejuízo da percepção da polifonia.

Obviamente, uma primeira necessidade para uma composição atender a tais requisitos está em manter cada voz melódica em sua região própria, que, embora podendo avançar sobre o espaço das linhas vizinhas, não deverá cruzá-las, invertendo a posição de duas vozes, efeito este que provocaria imediata confusão sobre a real natureza do movimento melódico efetuado por ambas as linhas envolvidas no cruzamento.

O uso de movimentos idênticos por duas vozes também pode prejudicar esse objetivo básico, pois as linhas melódicas envolvidas podem ser confundidas entre si. Não apenas os movimentos paralelos em

uníssono (mesmas notas) devem ser evitados, como também movimentos paralelos de intervalos justos (oitavas e quintas, chamadas consonâncias perfeitas), aos quais o ouvido atribui naturalmente uma relação de equivalência. O uso de movimento paralelo, ainda que excluídos tais intervalos, também pode causar a perda da independência das vozes individuais, se mantido por período de tempo suficientemente longo.

Com a finalidade de impedir cruzamentos de vozes, empregamos ações semânticas para que as máquinas de Markov responsáveis pela composição de linhas melódicas vizinhas se adaptem entre si, da mesma forma como o fizemos para a restrição do aparecimento de dissonâncias nos tempos fortes do compasso. Para o controle sobre a progressão do contraponto, de forma que sejam resolvidas as dissonâncias e evitem-se a ocorrência de movimentos em intervalos paralelos, empregamos uma máquina adaptativa independente, responsável pelo conhecimento derivado da teoria do contraponto.

Vamos escrever uma gramática que produza uma linguagem composta de distribuições de intervalos, que atendam aos requisitos de contraponto comentados nos parágrafos anteriores.

Os símbolos terminais dessa gramática representam disposições relativas de acordes a serem formados sobre uma nota de baixo. Para o contraponto de quatro vozes, empregamos na especificação de cada terminal três números inteiros, que representam os intervalos harmônicos entre cada par de notas que formam o acorde. Assim, o terminal (3, 3, 4), por exemplo, representa o acorde formado pela adição de três notas à voz mais grave, sendo que a primeira se encontra uma terça acima da voz do baixo, a segunda nota uma terça acima da primeira, e, finalmente, a quarta voz é disposta uma quarta acima desta última. Esse acorde está representado na figura 5.4.5.1, sendo a nota “dó” a nota do baixo, sobre a qual o acorde é construído.



Figura 5.4.5.1 - Interpretação do terminal (3, 3, 4)

A gramática, que vamos escrever, vai definir as progressões válidas entre cada disposição dos acordes, de forma a excluir aquelas que contenham seqüências com intervalos perfeitos em movimento paralelo, as quais procuramos evitar. Por exemplo, a sucessão entre os símbolos (3, 3, 4) e (3, 4, 3) não deve ser permitida por essa gramática, pois na sucessão dos acordes que representam está presente a oitava em movimento paralelo entre as vozes extremas, como ilustrado pela figura 5.4.5.2.



Figura 5.4.5.2 - Ocorrência de oitava paralela na sucessão dos símbolos (3, 3, 4) e (3, 4, 3)

Podemos definir a seguinte gramática:

N → AM	P → AN
N → CM	P → BN
N → EM	P → CN
N → GP	P → DN
N → HP	P → EN
N → IP	P → FN
M → BN	P → AM
M → DN	P → BM
M → FN	P → CM
M → GP	P → DM
M → HP	P → EM
M → IP	P → FM

Verifica-se que essa gramática produz formas sentenciais, em que certos grupos dos símbolos não-terminais A, B, C, D, E, F, G, H e I nunca ocorrem em seqüência, devido às substituições que são permitidas para os não-terminais M, N e P.

As produções sobre os símbolos A, B, C, D, E, F, G, H e I correspondem efetivamente à seleção dos símbolos terminais, representando os intervalos que vão definir os acordes formados pelas vozes em contraponto:

$$\begin{aligned} A &\rightarrow (3, 3, 4) \\ B &\rightarrow (3, 3, 6) \\ C &\rightarrow (3, 4, 3) \\ D &\rightarrow (3, 4, 5) \\ E &\rightarrow (4, 3, 3) \\ F &\rightarrow (4, 3, 6) \\ G &\rightarrow (5, 4, 3) \\ H &\rightarrow (6, 3, 3) \\ I &\rightarrow (6, 3, 4) \end{aligned}$$

Finalmente, com as produções $N \rightarrow \varepsilon$, $M \rightarrow \varepsilon$ e $R \rightarrow N$, sendo R o símbolo inicial, temos a gramática definida.

Pode-se verificar que a sucessão dos acordes, correspondente à sucessão dos símbolos definidos pela gramática, não vai conter intervalos paralelos de oitava e quinta justas entre quaisquer das quatro vozes em contraponto.

A máquina de Markov adaptativa, que gera a mesma linguagem que essa gramática, deve, ao ser assumido um determinado estado, acionar funções adaptativas sobre as máquinas responsáveis pelas diversas linhas melódicas em contraponto, de forma a impor a preferência estatística correspondente para a formação do acorde desejado, em função da nota estabelecida como baixo do conjunto. A escolha da nota do baixo é independente desse processo, estando relacionada exclusivamente aos

processos melódicos referentes à sua própria voz e ao contexto harmônico que está em vigor.

Um possível esquema da interação das máquinas de Markov para o estabelecimento das relações de contraponto descritas neste item é apresentado na figura 5.4.5.3.

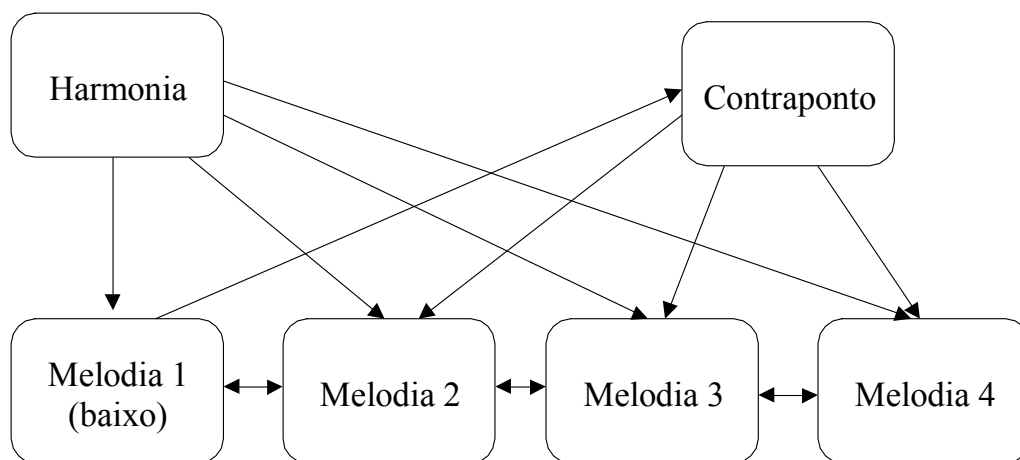


Figura 5.4.5.3 - Interação entre Máquinas de Markov

Da forma como a nova máquina está definida, ela não mantém nenhuma relação particular com a máquina que implementa a progressão das funções harmônicas, exceto pela dependência, indireta, da nota escolhida para o baixo. No entanto, observe-se que tal fato não prejudica o funcionamento do conjunto, uma vez que a função harmônica do trecho é mantida, ainda com a vantagem de aparecerem novos acordes, sobre outros graus da tonalidade.

Para ser obtida uma consistência harmônica maior, poder-se-ia unificar as máquinas de contraponto e de harmonia, de forma que se considerem funções harmônicas particulares para cada acorde, em cada inversão, ou, uma solução mais prática, mantendo-se a simplicidade obtida para ambas, permitir que tais máquinas, permanecendo como unidades de conhecimento musical independentes, interajam em um esquema de adaptação de comportamento, em um mesmo nível semântico.

Essa interação poderia servir para, por exemplo, obter cadências perfeitas ou imperfeitas em instantes adequados.

5.4.6. *Controle sobre a estrutura musical*

Todo o conjunto de máquinas descrito nos itens anteriores pode, eventualmente, estar sob o controle de máquinas de Markov adaptativas gerais, que vão definir o estado global da composição em cada instante. Este tipo de interação pode ser interessante para, por exemplo, a imposição de relações estruturais à peça musical composta, de forma que sejam compostas seções separadas, facilmente identificáveis pelo ouvinte.

A utilização das máquinas de Markov adaptativas apresentadas até o momento produz seqüências intermináveis de símbolos, permanecendo a compor linhas melódicas de duração indefinida, enquanto estiverem sendo executadas, até que ocorra a interferência do usuário do sistema. É interessante incluir um mecanismo capaz de estabelecer, pelo menos, seções caracterizáveis como início-desenvolvimento-fim, encerrando a composição pela desativação de todas as máquinas do sistema no momento apropriado, uma vez atingido um determinado estado final.

Elementos lingüísticos pertencentes a tal nível semântico podem definir elementos específicos para os outros níveis, visando à caracterização de cada seção, como por exemplo, em temas ou conjuntos de temas para a especificação da estrutura das frases musicais compostas, progressões harmônicas que conduzam a modulações, introduzindo tonalidades diferentes para seções separadas, alterações na forma como é efetuado o acompanhamento, ou até modificações na disposição de vozes individuais.

O controle sobre a estrutura da peça musical composta pode estabelecer formas musicais definidas, como por exemplo, através da produção

N → ABA

que pode ser usada para estabelecer uma forma ternária, em que o retorno de elementos da seção inicial, A, ao final, é facilmente perceptível pelo ouvinte, sendo interpretada como conclusão e retorno a um estado de repouso, preparando para o encerramento da composição.

Um controle mais rígido sobre a produção das frases musicais e sobre a tonalidade do segmento musical composto pode ser empregado para compor estruturas musicais mais complexas, como no exemplo a seguir:

$$\begin{aligned} N &\rightarrow IDRC \\ I &\rightarrow AB \\ D &\rightarrow AD \\ D &\rightarrow BD \\ D &\rightarrow \varepsilon \\ R &\rightarrow AB \end{aligned}$$

onde o símbolo N é substituído por uma seqüência de não-terminais descrevendo uma sucessão de várias seções, nas quais os elementos AB aparecem substituindo I, R e D, em seqüência nos dois primeiros, em qualquer ordem e repetindo-se indefinidamente no último. A forma sentencial é encerrada pelo símbolo C.

Pode-se verificar facilmente que esta estrutura, caso os símbolos não-terminais A, B e C sejam empregados na definição de células temáticas específicas, vai produzir algo semelhante à forma sonata, em seções de introdução, desenvolvimento, recapitulação e coda. Se a seleção dos símbolos I, referente à seção inicial, D, referente ao desenvolvimento, e R, referente à recapitulação, ainda intervier sobre a seleção de tonalidades diferentes para cada seção definida, por exemplo, para que o tema B na seção I se encontre na tonalidade da dominante, bem como os temas A e B em D, e ambos os temas A e B na seção R apresentem-se na tonalidade de tônica, tal analogia para com a forma sonata será ainda mais forte e interessante.

No exemplo realizado para a composição de corais a quatro vozes, foi empregada uma relação simples de início-desenvolvimento-conclusão, com a especificação de formas diferentes de início da composição, como por exemplo, a entrada gradual das vozes do coral, formas imitativas, etc. A seção correspondente ao desenvolvimento permite que as máquinas adaptativas transitem livremente, enquanto que a seção final força o aparecimento de cadências envolvendo todas as vozes, através da elevação gradativa da probabilidade de serem atingidas as notas do acorde de tônica, mantendo-se neste último estado pela duração inteira de um compasso, após o que todas as máquinas são desativadas, encerrando a composição.

A figura 5.4.6 ilustra o interrelacionamento de diversas máquinas de Markov, para um exemplo complexo de composição a quatro vozes, adicionado o controle sobre a estrutura global da música a ser composta. Observe-se a existência de múltiplos relacionamentos, tanto interações entre máquinas situadas em um mesmo nível semântico, quanto em níveis distintos. Observe-se também a existência de vários níveis semânticos, dispostos hierarquicamente, evidenciando um conhecimento musical estruturado.

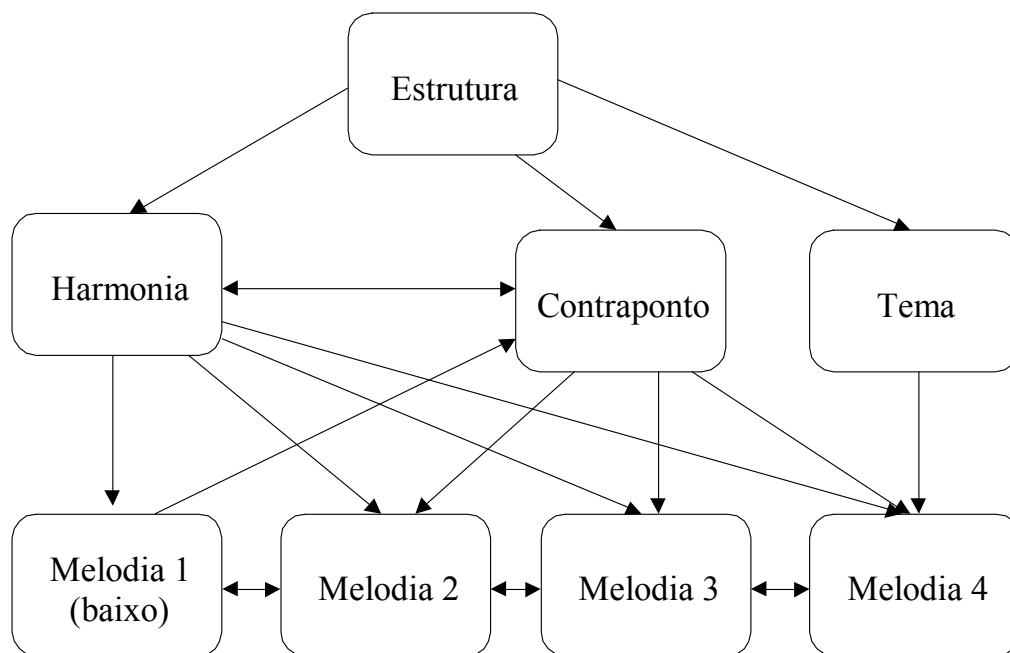


Figura 5.4.6 - Interação entre máquinas de Markov

5.5 Conclusões

Os modelos propostos para a representação do conhecimento musical, empregado pelo sistema para a composição automática de música, demonstram um potencial interessante tanto para a representação de conhecimento quanto para a estruturação dessa representação. A complexidade dos modelos empregados foi sendo intencionalmente aumentada no decorrer da apresentação deste capítulo. Pode-se observar, contudo, que os modelos mais simples (por exemplo a geração de linhas melódicas do item 5.4.1) não são substituídos por modelos mais complexos (por exemplo a interação do contraponto a quatro vozes, do item 5.4.5), mas sim a eles integrados. Tal modularidade conseguida para a representação do conhecimento musical é uma das características e vantagens mais importantes dos sistemas de Markov adaptativos desenvolvidos com essa finalidade. No capítulo seguinte, os resultados musicais obtidos com a tecnologia empregada serão discutidos em maior detalhe.

6. Discussão e Conclusões

Neste capítulo, discutimos os resultados obtidos, aspectos relacionados à tecnologia desenvolvida, e apresentamos conclusões, delineando propostas para pesquisas futuras. Dada a interdisciplinaridade do trabalho, diversas conclusões estão relacionadas a outros domínios de conhecimento, o mesmo ocorrendo em relação a futuras linhas de pesquisa, algumas das quais estão sendo propostas neste capítulo.

6.1 Avaliação do trabalho e dos resultados obtidos

O sistema implementado chega a surpreender pela sua simplicidade, frente aos resultados que puderam ser obtidos. Os arquivos executáveis mostraram-se muito compactos e a execução do programa muito eficiente, possibilitando sua execução em tempo real, outra característica distintiva do sistema implementado.

O tempo de execução total do sistema executor das máquinas entre iterações, para a definição dos próximos estados de cada uma das máquinas de Markov do sistema implementado, é proporcional ao número total de estados existentes em um dado instante. Dessa forma, podemos considerar a complexidade do algoritmo de execução do sistema de Markov adaptativo como linear, ou seja, $O(n)$. A não existência de retrocessos para a busca de solução é um diferencial importante, permitindo que o algoritmo seja executado de forma linear.

Com tais características, o algoritmo desenvolvido pode ser implementado facilmente em qualquer arquitetura que possua apenas um mínimo de recursos computacionais. Podem ser criados geradores de seqüências pseudo-aleatórias eficientes, que podem servir adequadamente como geradores de variáveis aleatórias com distribuições razoavelmente uniformes de probabilidade, necessárias à implementação do núcleo de execução de sistemas de Markov adaptativos.

O sistema desenvolvido opera em tempo real, reproduzindo a música à medida em que é composta, uma característica interessante, que pode ser explorada na criação de diferentes aplicações, conforme discutido mais adiante.

Ao analisarmos tanto a teoria quanto a implementação da técnica apresentada neste trabalho, podemos verificar alguns pontos de convergência com outros tipos de técnicas ou teorias. A existência de fontes independentes de conhecimento, cada uma referente a um determinado aspecto do conhecimento musical, é apontada por [LERDAHL83] em sua teoria sintática musical, que pode ser formalizada através de gramáticas multi-dimensionais propostas por Roads [ROADS79][ROADS96]. Podemos analisar o sistema implementado como uma possível forma de realização de um conjunto de gramáticas multi-dimensionais.

Também pode ser visualizado algum tipo de equivalência ou parentesco entre os sistemas de Markov adaptativos e os conjuntos nebulosos. Nestes últimos, a verificação da validade de uma dada hipótese é assumida após um conjunto de operações sobre variáveis que apresentam um grau de certeza associado; é possível imaginar uma analogia para com uma máquina adaptativa de Markov em um estado isolado, o que depende, em última instância, de um conjunto de transições com probabilidades de ocorrência associadas. Caso seja possível estabelecer relações entre probabilidade e grau de pertinência, é possível estabelecer um paralelo entre os dois domínios. Deve-se notar, contudo, a importante diferença que reside no fato de que as redes de Markov representam processamentos não-determinísticos, enquanto que tecnologias baseadas em conjuntos nebulosos são determinísticas.

A tarefa de avaliar os resultados obtidos com a aplicação dos conjuntos de Markov adaptativos, apresentados neste trabalho, pode ser considerada complexa, bem como a verificação sintática das cadeias musicais produzidas pelo sistema. Torna-se difícil avaliar uma

determinada cadeia produzida como sendo ou não uma *solução* para o problema da composição musical automatizada. A natureza de tal dificuldade está intimamente relacionada à complexidade do problema proposto e à ambigüidade inerente ao domínio lingüístico-musical.

Para efetuar-se uma tal avaliação, procedeu-se à apresentação dos fragmentos musicais gerados pelo programa a ouvintes humanos, sujeitando o resultado desta avaliação à apreciação daqueles fragmentos por tais ouvintes. Apesar da fragilidade desse tipo de avaliação, ela pode fornecer alguns dados importantes sobre os resultados obtidos com o sistema implementado.

A totalidade dos ouvintes entrevistados não apresentou dúvidas quanto ao reconhecimento de qualidades essencialmente musicais nos trechos compostos pelo sistema. Curiosamente, alguns ouvintes pareceram não ter percebido o fato de que a música havia sido produzida por um programa de computador, referindo-se a ela com expressões normalmente atribuídas a compositores humanos.

Observou-se que as características impostas para a estruturação das peças musicais compostas puderam ser reconhecidas facilmente pelos usuários do sistema, os quais apontaram segmentos específicos das composições como início e término de seqüências, reconheceram características tonais de sucessão de funções harmônicas, e alguns ouvintes apontaram inclusive determinadas características referentes a um suposto estilo musical, as quais obviamente não se encontravam descritas explicitamente em nenhum aspecto específico do modelo de conhecimento musical que foi empregado. Tais associações estilísticas podem ser justificadas por uma compreensão global da composição, de forma que foi possível aos ouvintes reconhecer uma estruturação relativamente rígida e associá-la a determinados estilos, que exibem uma rigidez formal comparável a esta.

6.2 Aplicações do Sistema

Podem ser vislumbradas diversas aplicações interessantes para a tecnologia desenvolvida neste trabalho. A relativa simplicidade de sistemas baseados nesta técnica, aliada à universalidade e formalismo no procedimento de representação de conhecimento, suas possibilidades de execução em tempo real, etc, conferem a ela inúmeras possibilidades de utilização em sistemas já existentes ou em novas idéias de software dedicado à música. Passamos a discutir, brevemente, algumas aplicações, em que a utilização desta tecnologia pode trazer vantagens.

Pela relativa simplicidade de sua implementação, a introdução de módulos de composição musical baseados nos sistemas de Markov adaptativos em dispositivos multimídia é bastante direta e simples. Seu uso em jogos e brinquedos eletrônicos, na geração de fundos musicais para apresentações multimídia, eventualmente sensível a contextos, e assim por diante, pode ser um recurso muito útil para a criação de aplicações mais sofisticadas e interessantes para os seus usuários.

Sistemas de composição automatizada, baseados nos sistemas de Markov adaptativos podem ser incluídos de forma modular em programas para improvisação musical, para geração automática de acompanhamentos, inclusive em tempo real. Um sistema assim construído ainda oferece a vantagem de poder ser externamente reconfigurado por meio de parâmetros ou mesmo através da adição de novas formas de conhecimento musical.

O uso dessa técnica em sistemas de Composição Auxiliada por Computador, também na forma de assistentes de composição, é outra aplicação interessante. Um sistema de Markov adaptativo como este pode ser empregado para o modelamento de conhecimento musical para a geração de fragmentos musicais que o usuário poderá analisar, alterar e reorganizar para a composição de uma nova obra. Em um sistema como esse, pode-se dispensar a execução do mesmo em tempo real, o que pode

vir a permitir que outras características lhe sejam adicionadas, como por exemplo, a interação com técnicas envolvendo algum processo de tentativa e erro.

A geração de composições musicais, baseadas em conhecimento musical formal, na forma de regras gramaticais, por exemplo, é uma interessante maneira pela qual esse conhecimento pode, de certa forma, ser validado. Seguindo este princípio, compositores formais do tipo que foi apresentado neste trabalho podem ser eficientemente empregados em sistemas de Musicologia Auxiliada por Computador. Utilizando tais sistemas, usuários podem verificar a validade de hipóteses estilísticas sobre determinadas poéticas musicais, auxiliando na descrição formal dessas hipóteses. As composições musicais produzidas pelo sistema poderão, então, ser confrontadas qualitativa ou estatisticamente com as obras musicais que se pretendem analisar.

No caso dos exemplos anteriores, pode ser necessária a adição de módulos encarregados de extraírem conhecimento musical de exemplos musicais fornecidos, evitando a codificação direta de regras gramaticais, funcionando segundo algum processo de análise musical automatizada. Tal procedimento seria uma espécie de verificação estatística e quantitativa de relações sintáticas existentes entre os elementos de uma composição, com o objetivo de obter automaticamente regras formais qualitativas, como aquelas que foram empregadas na codificação do conhecimento musical para os programas-exemplo apresentados neste trabalho. A aquisição de conhecimento musical através de análise pode ser empregada como uma entrada direta para o sistema, funcionando como um mecanismo de aprendizagem, ou ainda oferecendo-se à validação por parte de um especialista humano, no caso dos sistemas de Musicologia Auxiliada por Computador. Esse tipo de aplicação é uma interessante fonte para pesquisas futuras, envolvendo a tecnologia que estamos apresentando.

Nas aplicações citadas nos parágrafos anteriores, nas quais o usuário deve, necessariamente, trabalhar diretamente sobre o

conhecimento musical que está codificado no sistema, eventualmente inserindo, ele próprio, novas cláusulas ou regras sintáticas, é evidente a necessidade de algum modelo de interface homem-máquina mais adequada para que esse tipo vital de interação seja realizada de forma eficiente ou, no mínimo, prática. O presente trabalho não enfoca diretamente a questão da interface com o usuário no nível da codificação do conhecimento, uma vez que os programas-exemplo foram criados mediante a codificação direta do conhecimento musical, de forma que seja simulado o comportamento dos sistemas de Markov adaptativos correspondentes. A manipulação desses programas para a modificação da base de conhecimento empregada evidentemente não é simples para usuários não-especialistas. Dessa forma, a especificação de interfaces homem-máquina adequadas para aplicações, envolvendo a tecnologia aqui apresentada, consiste em uma linha de pesquisa bastante interessante.

Com todas estas características, softwares construídos com base em sistemas de Markov adaptativos podem ser empregados eficientemente na teoria e prática da música eletroacústica, que freqüentemente depende de formalismos lingüísticos [MENEZES98]. O sistema, que desenvolvemos aqui, trabalha com notas musicais isoladas, porém o tratamento é essencialmente simbólico, não interessando as reais entidades representadas pelos diversos símbolos. Assim, qualquer entidade ou parâmetro de uma composição musical pode ser representado. O fato de nosso sistema ser baseado em algoritmos não-determinísticos confere a ele, evidentemente, uma aplicação imediata na produção de música aleatória, à semelhança de outros programas desenvolvidos por compositores desse meio.

Outra aplicação muito interessante desse tipo de sistema é em educação musical. O ensino da teoria musical pode ser enfocada em softwares educacionais, através dos quais o estudante pode verificar a aplicação de regras teóricas nas composições musicais geradas, bem como os efeitos práticos da violação de determinadas regras ou de conjuntos de

regras. Programas educacionais podem ser tornados mais interessantes pela adição de sistemas como o aqui apresentado, para a produção de exemplos musicais dinâmicos, gerados no momento de sua execução, em substituição a exemplos estáticos. Uma versão simplificada da máquina de Markov adaptativa para a composição de linhas melódicas, aqui apresentada, foi empregada com sucesso em um software para a produção de ditados musicais, utilizado em disciplinas de percepção auditiva.

A possibilidade de execução dos sistemas de Markov adaptativos em tempo real, com a possível alteração de parâmetros durante a sua execução, por parte de algum intérprete, é um recurso importante para a sua utilização como núcleos de meta-instrumentos musicais. Por meta-instrumento musical designa-se um instrumento musical no qual o intérprete (ou intérpretes) atua(m) em níveis semânticos mais elevados da composição, e não apenas no nível léxico-sonoro, como ocorre com os instrumentos musicais tradicionais (veja, por exemplo, [POLANSKY94] ou [BARTLET79]). Além disso, um meta-instrumento musical pode, de forma ativa, interagir com o intérprete durante a execução da peça musical.

O núcleo de um meta-instrumento musical pode ser constituído por um sistema como o apresentado neste trabalho, com a adição de elementos de interface, os quais serão responsáveis pela interação com o intérprete. Tal interação irá refletir-se sobre um conjunto de parâmetros, que serão consultados pelo sistema de composição automatizada para o estabelecimento de novos padrões para a composição que está sendo gerada, a qual passará, portanto, a exibir um comportamento dependente da ação do intérprete sobre o sistema.

Os dispositivos de interface podem ser de qualquer tipo, como é o caso de sensores de posição, de pressão, de câmeras de vídeo associadas a programas de visão computacional, de microfones, de sensores de raios infravermelhos ou de ultra-som, etc. A correspondência entre as variáveis monitoradas pela interface e suas respectivas traduções em elementos

musicais é arbitrária, podendo interagir em qualquer nível semântico da composição efetuada pelo sistema.

Pode-se desenvolver esta linha de raciocínio, relacionada com os meta-instrumentos musicais, para chegar-se à conclusão de que qualquer objeto representável computacionalmente pode ser traduzido em música por tais sistemas. Sistemas funcionando como meta-instrumentos musicais podem ser empregados em diversas áreas, desde a realização de *performances* de música eletroacústica, instalações de música ambiental, até sua utilização em educação musical e musicoterapia.

6.3 Sistemas de Markov adaptativos como mecanismo para a solução de problemas complexos

Pode-se concluir, pela análise dos resultados obtidos, que a arquitetura do sistema, modelando o conhecimento musical desejado, pode ser considerada eficiente sob diversos aspectos. Considerando a relação entre a complexidade do problema e a do algoritmo proposto para a sua solução, pode-se observar que a solução adotada apresenta importantes vantagens sobre diversas técnicas tradicionais de solução de problemas.

O problema da composição musical automatizada parece ser um caso bastante particular, com um domínio extremamente específico. No entanto, problemas da mesma natureza podem ser identificados em qualquer aplicação, que envolva diretamente elementos lingüísticos e relações de contexto. Da mesma forma que o sistema aqui implementado lida com entidades que representam sons isolados, e ocupa-se da combinação de tais elementos de forma sintaticamente coerente segundo um conjunto de regras gramaticais, a mesma técnica pode ser eficientemente empregada para manipular outras unidades simbólicas, como fonemas ou elementos léxicos mais sofisticados, normalmente encontrados em linguagens naturais.

É interessante também observar que a técnica utilizada não busca soluções puramente analíticas para o problema proposto, mas segue por

um caminho diferente, procurando atender aos requisitos especificados, podendo, entretanto, eventualmente violá-los, sem que isto acarrete qualquer tipo imediato de inconsistência. Dessa forma, esperam-se não exatamente soluções absolutas, mas boas aproximações de soluções. Tal natureza de comportamento é muito interessante para problemas, cujas soluções ou não são de simples obtenção por meio de algoritmos convencionais, ou mesmo nem sequer possam ser obtidas pela aplicação de técnicas algorítmicas.

Algumas das características intrínsecas desse tipo de modelamento são bastante úteis para a resolução de problemas complexos, para os quais outras técnicas computacionais não apresentam a eficácia desejada. O exemplo apresentado neste trabalho ilustra um caso em que está claramente presente um espaço de problema para o qual o conhecimento necessário para resolvê-lo é incompleto, vago ou mesmo inconsistente. No entanto, nenhuma dessas limitações impede o sistema de encontrar soluções. A implementação das redes de Markov adaptativas traduz a incompletude e a inconsistência em relações internas de probabilidades de forma natural, como se pode verificar pela análise do modelo de conhecimento musical que foi empregado.

A incompletude está diretamente relacionada a um conhecimento insuficiente sobre as soluções para o problema, enquanto a inconsistência se relaciona mais com um excesso de informação, possivelmente proveniente de fontes de conhecimento diferentes, fator comum em problemas computacionais complexos. Esse tipo de tecnologia demonstra um grande potencial para a manipulação de grandes volumes de informação, sem que este fato incorra necessariamente na degeneração do sistema por excessivo processamento ou explosão combinatória.

Pode-se observar, também, que a entrada do sistema é justamente um conjunto de regras, que descreve o conhecimento necessário para a solução do problema. Como entradas para o sistema podem ser empregadas, também, novas informações, codificadas na forma de

parâmetros, recebidos concorrentemente à execução das regras, os quais podem ser utilizados para direcionar o comportamento do sistema.

A avaliação do comportamento do sistema implementado pode ser interpretada como indicativo de uma grande capacidade de adaptação, tanto a adaptação a novas regras fornecidas ao sistema, correspondendo a uma ampliação ou modificação da base de conhecimento existente, quanto uma adaptação a parâmetros fornecidos ao sistema durante a sua execução, que pode eventualmente corresponder a alguma forma de aprendizado.

As últimas propriedades apresentadas sugerem uma interessante possibilidade do sistema para adaptação a um ambiente, com o qual ele se relaciona através de um número qualquer (eventualmente grande) de variáveis ambientais. Dessa maneira, podem ser vislumbradas aplicações deste tipo de tecnologia em sistemas complexos de controle, como a robótica, por exemplo.

6.4 Emulação Computacional da Criatividade

Outro aspecto interessante a ser comentado refere-se à natureza do processo que se buscou simular através dos sistemas de Markov adaptativos. Mais do que um processo de busca de solução, análogo a outros problemas computacionais, a composição musical automatizada envolve uma série de questões, que normalmente não estão imediatamente associadas aos sistemas computacionais.

Não é necessário discutir a semiótica da palavra “composição” para reconhecer que o termo é entendido não apenas como um processo intelectual linear, procedimental, e, portanto, algorítmico, mas é também, e talvez principalmente, relacionado a fenômenos criativos. Por processo criativo (envolvendo *criatividade*), podemos entender uma atividade cujo produto ou processo possa, em algum sentido, ser considerado como inusitado ou inédito.

O domínio da criatividade não é normalmente abrangido pelos processos computacionais, posto que os computadores não podem apresentar decisões criativas sob nenhum aspecto, uma vez que, para um conjunto de entradas especificadas, a execução de algoritmos define, *a priori*, um comportamento específico e um conjunto de saídas ou resultados possíveis, em uma relação lógica direta de implicação – implicado.

A conceituação dos processos criativos é de difícil obtenção. Ela é alvo de discussões do domínio da Psicologia, dentre as quais podemos destacar a visão comportamentalista, que inclui a variabilidade de comportamento como uma fonte de recursos para a criatividade.

Apesar de tais dificuldades, uma observação interessante pode ser feita a respeito do sistema implementado. Uma das maiores curiosidades apresentadas pelos ouvintes das saídas do sistema apresentado referia-se ao mecanismo através do qual uma atividade criativa poderia estar sendo reproduzida, ou pelo menos emulada, pelo computador.

Embora corresponda obviamente a uma personificação, este reconhecimento intuitivo da presença, no sistema, de algo semelhante a uma criatividade computacional, parece estar relacionada à complexidade das interações entre as fontes de conhecimento musical empregadas na composição e também à natureza não-determinística deste processo de composição algorítmica, o qual segue um modelo de sistema estocástico, baseado em grandezas aleatórias.

Ao partirmos de um conjunto complexo de regras de composição, podemos perder de vista o processo algorítmico, que efetivamente conecta o problema à sua solução, fato que pode conferir a ilusão de um processo genuinamente criativo. Por outro lado, ainda, dada a natureza não-determinística, intrínseca ao algoritmo implementado, a própria solução do problema não é conhecida *a priori*, senão como uma relação de probabilidade. Este fato pode também ser interpretado como alegoria de

uma variabilidade comportamental, contribuindo assim para a ilusão de um suposto processo criativo.

6.5 Música e Linguagem

Caso aceitemos a opinião dos ouvintes, que avaliaram a saída do sistema como sendo propriamente música, podemos questionar quais tenham sido as características desses particulares resultados que possam ter contribuído diretamente para tal julgamento. A verificação da existência de uma organização interna nas seqüências musicais produzidas pelo sistema parece ser um fator primordial, uma vez que, quanto mais evidente se tornar uma tal organização, com maior grau de certeza a peça composta poderá ser classificada como sendo autenticamente musical.

A organização imposta às seqüências musicais produzidas é essencialmente sintática, no sentido de atender, tanto quanto possível, a regras gramaticais definidas com base no conhecimento musical. Ao que parece, a existência de uma organização sintática é um diferencial pelo qual se pode classificar o resultado do sistema como sendo ou não música.

Esse resultado parece estar de acordo com afirmações de diversos teóricos musicais, que tendem a afirmar que música é uma relação entre sons (ou entre “tons”), que pode muito bem ser considerada como uma relação sintática [LERDAHL83]. Uma questão natural aparece diante de tal afirmação, no que tange ao conceito de linguagem. As relações sintáticas servem para a definição de uma linguagem, à qual o senso comum atribui a função social de transmitir ou de comunicar algum tipo de informação, que se encontra codificada ou simbolizada pelas estruturas sintáticas empregadas (conforme [SCHURMANN90]). Se assim é, qual seria a mensagem, ou seja, o significado, e qual seria a semântica associada a um determinado fragmento musical?

A esse tipo de questionamento, os puristas costumam responder que a música abstrata não é uma linguagem, ou, pelo menos, que não é uma

linguagem no sentido de um código com alguma função comunicativa. Outros teóricos admitem a existência de uma semântica musical, mas reservam para ela um domínio exclusivamente musical, não associado a elementos exteriores ao universo musical (conforme [ROBINSON97]). Segundo tais interpretações, quaisquer associações que venham a ser estabelecidas nesse sentido devem ser consideradas exclusivamente como metáforas, as quais devem, inclusive, ser evitadas. Uma música que deliberadamente suscite tais associações, como a música programática, por exemplo, é considerada como de valor inferior.

Aparentemente, a teoria musical ocidental que temos à disposição atualmente parece concordar com esta postura, uma vez que encontramos extensa bibliografia sobre harmonia, contraponto, forma, análise; nenhum desses textos faz referência a qualquer metodologia para a codificação de elementos extra-musicais. Dessa forma, a existência de uma retórica musical é avaliada como desprovida de significado, se considerada exterior à linguagem musical propriamente dita.

Esse tipo de análise supervaloriza a sintática musical, relegando eventuais aspectos semânticos a um segundo plano. Segundo ela, a análise de uma composição musical tem por objetivo a observação das estruturas sintáticas empregadas por determinado compositor, mas jamais procurando interpretações quanto a significados extra-musicais.

Por outro lado, também são inúmeros os trabalhos, inclusive em áreas diversas, como a psicologia experimental, a filosofia cognitiva, a sociologia, etc, em que a existência de elementos semânticos extra-musicais é considerada incontestável [ROBINSON97]. Segundo eles, o estabelecimento de um significado para uma obra musical qualquer é possível e necessário para a sua compreensão, e que tal significado foi devida e intencionalmente codificado pelo compositor da obra, com a finalidade de representar, simbolizar, ou exemplificar uma determinada realidade expressiva.

Não é do escopo deste trabalho levantar este tipo de debate. Entretanto, voltando ao nosso exemplo, o compositor estocástico desenvolvido obviamente ignora qualquer tipo de semântica, produzindo seqüências sintaticamente corretas, conforme definidas através de regras gramaticais formais. Segundo o que foi discutido sobre a análise não-lingüística da música, considerada como música dita absoluta, pura ou abstrata, a saída produzida por ele pode ser efetivamente considerada como tal.

No entanto, pode-se questionar se a ausência de possíveis semânticas não possa ser percebida e identificada pelos ouvintes humanos que avaliarem os resultados obtidos. A resposta é afirmativa. Vários ouvintes observaram a falta de uma “expressão” na música produzida, por vezes verificando a presença de alterações no caráter da composição ao longo de sua execução, porém sem identificarem uma razão ou ordenação lógica aparente. Pode-se entender tais observações tanto do ponto de vista da estruturação sintática – segundo a qual níveis de estruturação mais elevados não estariam presentes nas composições geradas (tal fato foi percebido pelos ouvintes), quanto da constatação da carência de uma possível estruturação semântica discursiva.

Este tipo de discussão, suscitada por tais comentários dos ouvintes, é interessante e refere-se não apenas ao presente trabalho, mas à totalidade dos sistemas de composição musical automatizada. Se há interesse por uma semântica, ou um objetivo composicional específico, como este pode ser modelado? Quais aspectos sintáticos da composição devem ser selecionados para a exploração deste ou daquele aspecto semântico? Todas essas questões são relevantes para diversos trabalhos futuros na área.

6.6 Conclusões

O desenvolvimento da teoria descrita neste trabalho mostrou-se útil para a solução, ou aproximação de soluções, para os problemas envolvidos

com a composição musical automatizada. Resumindo o que foi comentado no presente capítulo, dentre as principais vantagens oferecidas por esta tecnologia, é interessante destacar:

- Descrição formal de conhecimento;
- Relativa simplicidade da implementação final;
- Tolerância a grandes volumes de informação;
- Aplicação na solução de problemas complexos.

Observa-se que a continuidade para trabalhos de pesquisa nesta área é um campo aberto, envolvendo diversas áreas de conhecimento. Dentre as linhas de pesquisa que poderiam relacionar-se mais ou menos diretamente à teoria proposta, destacamos:

- Estabelecimento de paralelos com outros formalismos;
- Integração com outros formalismos;
- Aplicação em outros domínios lingüísticos;
- Enriquecimento do conhecimento modelado com base em novos níveis semânticos;
- Estudo de aspectos semânticos envolvidos na comunicação musical;
- Estudo de aspectos lingüísticos envolvidos com a expressão musical, aplicações em musicologia;
- Modelamento de interfaces homem-máquina mais adequadas;
- Aplicações em sistemas de Composição Auxiliada por Computador, Musicologia Auxiliada por Computador e sistemas de acompanhamento automático;
- Aplicação em educação musical;
- Aplicação em sistemas multimídia;
- Criação e utilização de meta-instrumentos musicais.

7. Bibliografia

[ANDERSON91] ANDERSON, D. P., KUIVILA, R. Formula: a Programming Language for Expressive Computer Music. **Computer** v. 24, n. 7 pp.12-21, 1991.

[BARTLETT79] BARTLETT, M. Software for a Microcomputer-Controlled Synthesizer for Live Performance. **Computer Music Journal** v.3, n.1, pp.25-29, 1979 apud ROADS, C., STRAWN, J. **Foundations of Computer Music**. 1st Ed., Cambridge, The MIT Press, 1987.

[BASSETO99] BASSETO, B. A. NETO, J. J. A Stochastic Musical Composer Based on Adaptive Algorithms. /Simpósio Brasileiro de Computação e Música, Rio de Janeiro, 1999./

[BISCHOFF78] BISCHOFF, J., GOLD, R., HORTON, J. Music for an Interactive Network of Microcomputers. **Computer Music Journal** v.2, n.3, pp.24-29, 1978 apud ROADS, C., STRAWN, J. **Foundations of Computer Music**. 1st Ed., Cambridge, The MIT Press, 1987.

[BLEVIS89] BLEVIS E., JENKINS, M., ROBINSON, E. On Seeger's Music Logic. **Interface** v.18 pp.9-31, 1989.

[BUXTON78] BUXTON, W., REEVES, W., BAECKER, R., MEZEI, L. The Use of Hierarchy and Instance in a Data Structure for Computer Music. **Computer Music Journal** v.2 n.4 pp.10-20, 1978 apud ROADS, C., STRAWN, J. **Foundations of Computer Music**. 1st Ed., Cambridge, The MIT Press, 1987.

[CAMURRI95] CAMURRI A., CATORCINI, A., INNOCENTI, C., MASSARI, A. Music and Multimedia Knowledge Representation and Reasoning: the HARP System. **Computer Music Journal** v. 19 n. 2 pp.34-58, Cambridge, 1995.

[CANDÉ78] CANDÉ, R. **Historie Universelle de la Musique**, Éditions du Seuil, 1978.

[CHAMBERLIN85] CHAMBERLIN, H. **Musical Applications of Microprocessors**, 2.ed. New Jersey, Hayden Book Company, 1985.

[COPE87] COPE, D. An Expert System for Computer-assisted Composition. **Computer Music Journal** v.11 n.4 pp.30-46, 1987.

[DANNENBERG91] DANNENBERG, R. B., FRALEY, C. L. , VELIKONJA, P. Fugue: a Functional Language for Sound Synthesis. **Computer** v. 24, n. 7, pp. 36-41, 1991.

[DODGE86] DODGE, C., BAHN, C. R. Musical Fractals. **BYTE** v. 11, n. 6, pp. 185-196, 1986.

[DOWLING86] DOWLING W. J., HARWOOD, D. L. **Music Cognition**, Academic Press, 1986.

[FIELD97] SELFRIEDGE-FIELD, E. **Beyond MIDI: the handbook of Musical Codes**, Massachusetts, The MIT Press, 1997.

[FRITSCH94] FRITSCH, E. **Compositor Automático de Melodias Musicais – C.A.M.M.**, Dissertação de Mestrado, UFRGS, 1994.

[GRIFFITHS94] GRIFFITHS, P. **Modern Music**, 2ed., Malta, World of Art, 1994.

[GRIFFITHS95] GRIFFITHS, P. **Enciclopédia da Música do Século XX**, 1ed., São Paulo, Martins Fontes, 1995.

[HAUS91] HAUS, G., SAMETTI, A. Scoresynth: A System for the Synthesis of Music Scores Based on Petri Nets and a Music Algebra. **Computer** v. 24, n. 7, pp.56-60, 1991.

[HILLER59] HILLER, L., ISAACSON, L. **Experimental Music**, 1ed., New York, McGraw-Hill, 1959.

[HOWARD71] HOWARD, R. A. **Dynamic Probabilistic Systems**, v. 1 e 2, 1ed., New York, John Wiley & Sons, Inc., 1971.

[JACOB95] JACOB, B. L. Composing with Genetic Algorithms /International Computer Music Conference, Banff Alberta, Sept. 1995/

[JANZEN92] JANZEN, T. E. Algorhythms: Real-time Algorithmic Composition for a Microcomputer, apud BAGGI, D. **Computer Generated Music**, 1st. ed. California, IEEE Computer Society Press, 1992.

[LASKE89] LASKE, O. Composition Theory: An Enrichment of Music Theory. **Interface** v.18 pp.45-59, 1989.

[LEACH95] LEACH, J., FITCH, J. Nature, Music and Algorithmic Composition. **Computer Music Journal** v.19, n.2, pp.23-33, 1995.

[LERDAHL83] LERDAHL, F., JACKENDOFF, R. **A Generative Theory of Tonal Music**, 1ed., Cambridge, The MIT Press, 1983.

[LIMA98] VIEIRA LIMA, L. **Um Sistema de Composição Musical Dirigido por Estilo**. Tese de Doutorado, São Paulo, Escola Politécnica da Universidade de São Paulo, 1998.

[MATHEWS69] MATHEWS M. V. **The Technology of Computer Music**, 1.ed. Massachusetts, The MIT Press, 1969.

[MENEZES96] MENEZES, F. **Música Eletroacústica - História e Estéticas**, EDUSP, São Paulo, 1996.

[MENEZES98] MENEZES, F. **Atualidade Estética da Música Eletroacústica**, Editora da UNESP, São Paulo, 1998.

[MOOG86] MOOG, R. A. Digital Music Synthesis. **BYTE** v. 11, n. 6, pp. 155-168, 1986.

[NARMOUR99] NARMOUR, E. Hierarchical Expectation and Musical Style, apud DEUTSCH, D. **The Psychology of Music**, 2nd. ed., San Diego, Academic Press, 1999.

[NETO87] NETO, J. J. **Introdução à Compilação**, 1. ed. São Paulo, LTC, 1987.

[NETO93] NETO, J. J. **Contribuições à Metodologia de Construção de Compiladores**, Tese de Livre Docência, São Paulo, Escola Politécnica da Universidade de São Paulo, 1993.

[PAHLEN91] PAHLEN, K. **Nova História Universal da Música**, São Paulo, Melhoramentos, 1991.

[PENNYCOOK85] PENNYCOOK, B. W. Computer-Music Interfaces: a Survey. **ACM Computing Surveys** v.17 n.2 pp.267-289, 1985.

[POLANSKY94] POLANSKY, L. Live Interactive Computer Music in HMSL, 1984-1992. **Computer Music Journal** v.18 n.2 pp.59-77, 1994.

[ROADS79] ROADS, C. Grammars as Representations for Music. **Computer Music Journal** v.3 n.1 pp.48-55, 1979 apud ROADS, C., STRAWN, J. **Foundations of Computer Music**. 1st Ed., Cambridge, The MIT Press, 1987.

[ROADS85] ROADS, C. Research in Music and Artificial Intelligence. **ACM Computing Surveys** v.17 n.2 pp.163-190, 1985.

[ROADS87] ROADS, C., STRAWN, J. **Foundations of Computer Music**. 1st Ed., Cambridge, The MIT Press, 1987.

[ROADS96] ROADS, C. **The Computer Music Tutorial**, 1ed., Cambridge, The MIT Press, 1996.

[ROBINSON97] ROBINSON, J. – editor. **Music and Meaning**, 1st. ed., New York, Cornell University Press, 1997.

[RUSCHIONI93] RUSCHIONI, R. A. Música Digital: Histórico e Perspectivas /Workshop sobre Computação de Alto Desempenho para Processamento de Sinais, 1993/

[SCHOENBERG74] SCHOENBERG, A. **Tratado de Armonía**, 3ed, Madrid, Real Musical, 1974.

[SCHOENBERG93] SCHOENBERG, A. **Fundamentos da Composição Musical**, 2ed, São Paulo, Editora da Universidade de São Paulo, 1993.

[SCHOENBERG94] SCHOENBERG, A., NEFF, S. – editor. **Coherence, Counterpoint, Instrumentation, Instruction in Form**, Nebraska, University of Nebraska Press, 1994.

[SCHURMANN90] SCHURMANN, E. F. **A música como Linguagem**, 2ed, São Paulo, Brasiliense, 1990.

[SIMMS86] SIMMS, B. R. **Music of the Twentieth Century: Style and Structure**, 1ed., New York, Schirmer Books, 1986.

[SMAILL93] SMAILL, A., WIGGINS, G., HARRIS, M. Hierarquical Music Representation for Composition and Analysis. **Computer and the Humanities** n.27 pp.7-17, 1993.

[SMOLIAR91] SMOLIAR, S. W. Current Research in Computer-Generated Music. **Computer** v. 24, n. 7, pp.54-56, 1991.

[TRAGTENBERG94] TRAGTENBERG, L. **Contraponto: uma Arte de Compor**. São Paulo, Editora da Universidade de São Paulo, 1994.

[WIDMER95] WIDMER, G. Modeling the Rational Basis of Musical Expression. **Computer Music Journal** v.19, n.2, pp.76-96, 1995.

[WISNIK85] Wisnik, J. M. **O Som e o Sentido**, São Paulo, C. das Letras, 1985.

[ZICARELLI87] ZICARELLI, D. M and Jam Factory. **Computer Music Journal** v.11 n.4 pp.13-29, 1987.