Generating Chants Using Mnemonic Capabilities

Sudharsan Iyengar
Winona State University
siyengar@winona.edu

Abstract:

A chant is a simplistic repetitive song in which syllables may be assigned to a single tone. Additionally chants may be rhythmic and include simple melody. Chants can be considered speech or music that can convey emotion. Additionally, a chant can be monotonous, droning, and tedious. Fundamental to a chant is the notion of timing and note patterns.

We present here a framework for the synthesis of chants of music notes. These are chants without syllables from spoken language. We introduced Mnemonic capabilities in [1] and utilize these to systematically generate chants. We illustrate our ideas using examples using the notes set{C, D, E, G, A, ϕ (or silence)} - a Pentatonic Scale (perfect fifths).

First, we define, and structure, the use of timing and notes to develop the chant strings. Then we propose the use of mnemonics to develop different styles of musical chants. Finally we suggest the adoption of intonations and syllables for controlled generation of musical chants.

Keywords: Chants, Music Generation, Mnemonic Capabilities.

1. Introduction:

Artificial Intelligence is an area of Computer Science which deals with the development of 'learning machines'. It is generally accepted that learning is inherent to the notion of intelligence – with the expectation that learning contributes to accuracy, adaptability, and extension of abilities.

An intelligent system must have cognitive capabilities. Cognition is awareness and ability to perceive objects and/or ideas. Using sensory input one has to recognize a collection, or a stream, of input with a previously recognized and assimilated object or idea. Learning is not just rote recording and matching of inputs but a method of selectively recording recognized objects for effective reuse for cognition.

Eduardo Reck Miranda [2] present the notion that music is an intellectual activity; that is, the ability to recognize patterns and imagine them modified by actions. Miranda [3] presents computational models based on Grammars. Mira Balaban [4] presents a treatise on AI perspectives on Music Cognition. Specter and Alpern [5] present systematic approaches to explore the musical structures for the purposes of cognition.

In [1], using a 2-notes {C, D} we proposed a theory for abstraction (learning) and subsequent recognition of the abstracted musical elements. We also developed a framework based on fundamental mnemonic capabilities a methodology for cognition of newer elements. Learning Machines based on these capabilities were suggested. In essence, learning, and subsequent cognition, starts with primitive information/knowledge, and proceeds using capabilities and abstraction over time – an out come of repeated stimulation with inputs.

Generation of music – or speech in general using vocal and/or instrument – is a collective outcome of note pattern selection, timing pattern selection, and the resulting unification of the chants thereof with syllables. We present this as a bottom up synthesizing process.

We first develop the concept of timing and note patterns, and present a chant synthesizing system (CSS). We then present the use of mnemonics capabilities, from [1], for the development of creative chants. Lastly, we propose the adoption of intonations and syllables to develop musical chants. Illustrative examples using note alphabet - *{C, D, E, G, A, ϕ (or silence)}* are presented throughout.

2. Timing, Notes and Chant Synthesis.

It is quite often we hear a musical piece for the first time and we taken to like it. What is it that makes us like the piece? Fundamental to liking a piece is the familiarity and commonness of beats – which correspond to timing of the phrases and melody. Also, important is the familiarity of style which relates to composition characteristics.

We argue that three crucial characteristic are elementary to musical pieces.

The first characteristic is timing. Timing corresponds to a sequence of beats – with various durations. The pattern of beats is fundamental and necessary for coherent and recognizable musical piece.

The second characteristic are notes or note elements. The usage of notes belonging to a defined a set is used for a composition. These are audible elements of different pitches and/or pitch characteristics. The proper and simple arrangement of these notes is essential to the musicality of a piece.

The third characteristic is amplitude and/or instrumentation. Every note element and note element sequences can be further emphasized by the use of different instruments and volume. These assist in the aesthetics of the composition.

Timing element can be formed using a sequence of beats. Every beat has duration. We will work with beats of durations 1/8, ¼, ½ and 1 - denoted by δ, Δ, d, and F, respectively.

Although, timing patterns do not have to have commonality – use of distinct measures is prevalent musical pieces.

For illustration purposes we will use a measure of 1 for timing elements. The ideas developed in this paper are easily applicable to timing elements of other durations as well as varying durations.

Definition 1: A timing element $\mathcal{T}$ is a string of beats whose duration totals one measure.

Given the beats described above we can create a combination of 32 different timing patterns within duration of 1. Some timing patterns are shown in Table 1.

Definition 2: A note pattern $\mathcal{N}$ is a string of notes from the alphabet chosen.

We choose the alphabet *{C, D, E, G, A, ϕ}* where ϕ is silence. *The ϕ note is the universal complement.*

Given the alphabet in definition 3 we can create different strings. Some note patterns are shown in Table 2.

F
d d
d δ δ Δ
d δ d δ
δ δ Δ d
d δ δ δ δ
δ Δ δ d
δ δ Δ Δ Δ
Δ d Δ
δ δ δ δ δ δ δ δ

Table 1. Some timing patterns.

CD ϕ E
CDE
DCD
D

Table 2. Some Note Patterns

Definition 3: We play a note, n, on a beat, b, if the note is played for the duration of the beat. This is denoted (nb).

For example, the playing of the note *D* on a beat Δ is denoted ($D\Delta$).

Definition 4: A single chant element, $\mathbb{C}$, is a 3-tuple, $\mathbb{C} = (\mathcal{T}, \mathcal{N}, \mathcal{A})$. $\mathcal{T}$ is the timing element, $\mathcal{N}$ is the note pattern, and $\mathcal{A}$ is the amplitude/instrument pattern. A chant is then $\{\mathbb{C}\}^*$.

For the sake of simplicity we will assume uniform amplitude/instrument for the rest of the paper. Thus, a chant element is $\mathbb{C} = (\mathcal{T}, \mathcal{N})$. Note that silence is an empty chant!

Definition 5: Playing a chant element is the playing of a note pattern $\mathcal{N}$ using a timing element $\mathcal{T}$

Table 3.1 shows the use of beat pattern, $\mathcal{T} = \delta \delta \Delta d$, and various note patterns from Table 2.

CD ϕ E	$(C\delta)(D\delta)(\phi\Delta)(E\delta)$
CDE	$(C\delta)(C\delta)(D\Delta)(E\delta)$
DCD	$(D\delta)(C\delta)(D\Delta)(D\delta)$
D	$(D\delta)(D\delta)(D\Delta)(D\delta)$

Table 3.1. Some Chant elements.

Table 3.2 shows the use of a note pattern, $\mathcal{N} = C\ D$, played using different timing elements from Table 1.

(Cd) (Cδ) (Dδ) (DΔ)
(Cd) (Cδ) (Cd) (Dδ)
(Cδ) (Cδ) (DΔ) (Dd)
(CΔ) (Cδ) (D δ) (Dd)
(CΔ) (Cd) (DΔ)

Table 3.2. Some Chant elements (continued)

The synthesis of $\mathcal{N}$ over $\mathcal{T}$ is the generation of the chants $\mathbb{C}$. One can observe that it may be possible to play a note pattern over a timing element in more than one ways.

3. Classification of Timing and Note Patterns.

As described in [1] cognition is a function of the capabilities within a system. System with fundamental capabilities – recognize negation, repetition, and bifurcation. Higher order capabilities of homogeneous and heterogeneous combinations yield Intelligent Learning Systems [1].

We first propose our taxonomy of Timing Patterns.

B1: A single beat comprising the measure
B2: Measure made be repetition of a beat
B3: The measure is made up of a repetition of a beat followed by a repetition of a different beat.
B4: Repeated beat sequences comprise the measure.
B5: Beats sequence not conforming to the above levels of taxonomy.

Given that the notes are distinguishable by their pitch, we use the pitch of the notes to order the notes. We use the alphabet {*ϕ, C, D, E, G, A}*. For the purpose of this paper, we confine ourselves to these notes in a selected octave.

We suggest development of note patterns based on our taxonomy of note patterns (arrangements) as given below.

A1: Single note.
A2: Notes in monotonic ascending or descending order.
A3: A2 patterns where notes ascend after an initial skip; A2 pattern where notes descend after an initial skip.
A4: Ascending A2 pattern followed by a descending A2 pattern – and vice versa.
A5: A2 or A4 with one or more skips.

Table 4 and Table 5 show some timing and note patterns based on our taxonomy.

B1	F
B2	d d; Δ Δ Δ Δ; δ δ δ δ δ δ δ δ
B3	d Δ Δ; δ δ δ δ d
B4	Δ δ δ Δ δ δ; δ Δ δ δ Δ δ
B5	d δ δ Δ; d δ δ δ δ; Δ δ d δ

Table 4: Timing patterns based on timing taxonomy.

A1	C; E;
A2	DEG; CCDEGG CDEGA
A3	CEGA; ADC
A4	CDEGED; GAG
A5	DEAGED; CEGAEC

Table 5: Note patterns by arrangement taxonomy.

4. Synthesizing Chants:

Chants may be synthesized based on a given pair of timing and note pattern. Obviously, certain pairs do not yield chants (e.g. A4 cannot be synthesized on B1).

Familiarity with chants – as is with music – depends on the cultural upbringing and repeated exposure to combinations of notes patterns and timing patterns.

Simplest form of chant is the A1B1 combination – a single note on a single beat.

Another simple form of chant is the A1B2 combination – where a single note is repeated for equal intervals over a measure. In fact, A1B2 is a repetition of A1B1!

Similarly, A1B3 is distinguished from A1B2 by the fact that it is made up of two different A1B2 – with the requirement that the beats in the first and the second are different.

A1B4 is a combination of the previously described combinations.

A1B5 finally, is simply A1 applied on a plain combination of beats.

Definition 6: Chants generated using A1 note patterns are termed **primitive chants**.

Definition 7: The cardinality of a timing pattern, $\mathcal{T}$, is the number of beats in it, denoted $X(\mathcal{T})$. The cardinality of a note pattern, $\mathcal{N}$, is the number of notes prescribed in it, denoted $X(\mathcal{N})$.

Let us now consider chants with other note patterns. In order to synthesize a chant we need a timing pattern with cardinality equal to or greater than the cardinality of the note pattern.

When the cardinality of the timing and note patterns are equal then the synthesis of the chant is a 1-1 mapping of notes to the beats.

When the cardinality of the timing pattern exceeds that of the note pattern then we will utilize repetition of notes.

We propose that the cardinality of a note pattern be modified using repetitions, or by inserting ϕ, so as to equal its cardinality of the timing pattern. Note that tailoring should preserve the characteristic of the note pattern. Synthesizing a chant given a timing pattern may thus require tailoring the note pattern.

Definition 8: The modification of a note pattern, $\mathcal{N}$, to achieve a desired cardinality is termed tailoring. This denoted $X(\mathcal{N}) \Xi X(\mathcal{T})$.

Definition 9: Tailoring a note pattern by modifying its start using repetition (capability C2) is termed premature tailoring.

Definition 10: Tailoring a note pattern by modifying its end using repetition (capability C2) is termed tardy tailoring.

Definition 11: Tailoring a note pattern by placing ϕ, as needed, is termed lazy tailoring.

Example: $\mathcal{N}$ = DEG; $\mathcal{T} = \delta\ \delta\ \delta\ \delta$ d

$X(\mathcal{N}) = 3$; $X(\mathcal{T}) = 5$.

Premature tailoring:

$X(\mathcal{N}) \Xi X(\mathcal{T}) \Rightarrow \mathcal{N}'$ = DDDEG

Tardy tailoring:

$X(\mathcal{N}) \Xi X(\mathcal{T}) \Rightarrow \mathcal{N}'$ = DEEGG

Lazy tailoring:

$X(\mathcal{N}) \Xi X(\mathcal{T}) \Rightarrow \mathcal{N}'$ = DϕEϕG

Given an alphabet a chant synthesizing system (CSS) – Figure 1. The choice of chants to be generated is determined by the alphabet provided, the selection of control for note patterns and timing patterns and the choice of tailoring.

Thus based on the choice of timing, note, and tailoring used we can generate various types of chants.

Definition 12: Chants synthesized using only A2 patterns are termed **green chants**.

Definition 13: Chants synthesized using only A3 patterns are termed **motivating chants**.

Definition 14: Chants synthesized using only A4 patterns are termed **affecting chants**.

Definition 15: Chants synthesized using all types of note patterns are termed **appealing chants**.

Chant type	Note Pattern Used
Primitive chants	A1
Green chants	A2
Motivating chants	A3
Affecting chants	A4
Appealing chants	{A1 .. A5}

Table 6: Chant Types based on Note Pattern Types

Chant type	Timing Pattern Used
Simple	B1
Monotonous	B2
Permuted	B3
Assorted	B4
Generic	B5

Table 7: Chant Types based on Timing Pattern Types

Chant type	Tailoring used
Hasty	Premature
Sluggish	Tardy
Broken	Lazy

Table 8: Chant Types based on Tailoring Schemes

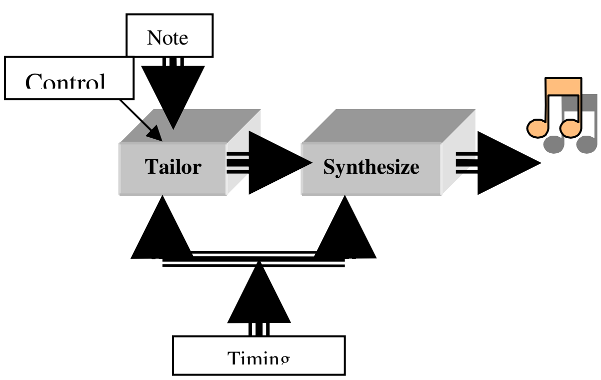

Figure 1. Architecture of Chant Synthesizing System (CSS).

5. Use of Mnemonic Capabilities [1].

In [1] the authors develop an abstraction mechanism that assimilates recognized input into the system knowledgebase. The abstracted patterns are used fundamental mnemonic to extend the cognitive powers of the system.

5.1. Fundamentals:

An element, e, is a unit of information – a string of musical notes. An element e can be viewed, if applicable, as a string of substrings.

A capability is the application of an operation to an element e, or any sub-element of e, that yields sub-elements or a new string.

An element e, is recognized, when e matches with known elements in the knowledgebase or the application of capabilities lead to recognition of elements obtained.

A learning capable system, S, is a triple, S = {C, A, H}, where, C is a set of capabilities; A is a set of abstracted elements or knowledge, and H is the history set.

5.2. Capabilities:

C1: Negation. Given an element e, the application C1(e) => e', where e' != e.
$C1(e) = C1(e_1e_2..e_k) \Rightarrow C1(e_1)C1(e_2)\ldots C1(e_k)$.

C2: Repetition. Given an element e, an application C2(e, n) => ee..ee (n repetition of e).

The history of recognized inputs, are used against a threshold to imbibe newer elements knowledgebase so they can be recognized immediately.

C3: Bifurcation. Given an element $e = e_1e_2\ldots e_m$, an application C3(e, j, k) => e' e''' e"; 1 <= j <= i <= m,

where $e' = e_1e_2..e_j$;
$e''' = e_{(j+1)}e_{(j+2)}..\ e_i$; and
$e'' = e_{(i+1)}e_{(i+2)}\ldots e_m$.

In other words e' is the first j elements, e" is the last k elements and e'" is the elements between them.

Rules for bifurcation based on the immediate recognition of repetition.

BF1: Bifurcate after the largest repetition of matched element from the start of string.
BF2: Bifurcate before the largest repletion of matched element from end of string.
BF3: Bifurcate before and after the largest repetition of matched element.

The granularity of an element e, denoted \e\, is expressed by the number of matching elements applied to recognize it. The size of an element e, denoted e_z, is defined as the minimum granularity with immediate matching. The expressiveness of e_x is the set of elements matched to recognize e. The shape of e is the minimum e_x of e, and is denoted e_s.

The shape, the expressiveness, the size, and the granularity may changes with changes in knowledgebase –A.

Given e = e_1 e_2; e = **H1**(e1, e2), a homogeneous combination I of e_1 and e_2, if e1 and e2 have the same shape.

Given e = e_1 e_2, e = **H2**(e1, e2), a homogeneous combination II of e_1 and e_2, if e1 and e2 have the equal granularity and have common matching elements.

Given e = e_1 e_2, e = **R1**(e1, e2), a heterogeneous combination of e_1 and e_2 if neither M1 nor M2 applies to e, e1, and e2.

H1 recognizes a input as symmetrical in content when bifurcated. That is, there is a bifurcation that yields sub-elements that are recognized using one set of known elements. The parts obtained by bifurcation don't have to be identical. They are composed of known elements from this.

H2 recognizes a input as similar in content when bifurcated. That is, there is a bifurcation that yields sub-elements that are recognized using different set of know elements – with same granularity and share common elements. The parts obtained by bifurcation are different but have same granularity (effort to recognize) and have similar characteristics.

R1 combinations are when bifurcation does not yield symmetric or similar parts, but do assist in recognition as a composition of diverse parts.

The following frameworks of learning systems were proposed.

Intelligent Learning System I (ILS1): System with capabilities {C1, C2, C3, H1}.
Intelligent Learning System II (ILS2): System with capabilities {C1, C2, C3, H1, H2}.
Intelligent Learning System III (ILS3): System with capabilities {C1, C2, C3, H1, H2, R1}.

The different levels of learning capable system illustrate the potential for various grades and scalable levels of learning and intelligence. Illustrative examples of the use of the capabilities and the cognition by the different learning systems are provided in [1].

6. Chants by Intelligent Learning Systems.

Lazy tailoring uses ϕ - where ϕ is a direct result of capability C1 – C1(note) = ϕ, for all notes.

Premature tailoring is a complement of BF1. Tardy tailoring is a complement of BF2.

The timing patterns provide bifurcation points. As the system evolves bifurcation may be appropriate over a sequence of timing patterns.

C3 can be used as the basis for generating variety of chants. Additionally, intelligent systems may use H1, H2, and R1 to generate complex chants..

6.1. Generating Homogeneous Chants.

We can start with a note string N1. Obtain its shape. Re-order the notes in N1 and obtain N2. N2 has the same shape as of N1.

Now generate chant C1 using N1 and then chant C2 using N2. The resulting chant C = [C1C2] is a homogeneous combination.

Let N1 yields a green chant (using A2 note pattern). N2 can be made to yield a green chant (N2 obtained by reversing N1). The resulting chant is an affecting chant!

Similarly, using N1 that yields a motivating chant, we can generate an appealing chant!

If N1 is an A2 pattern, and using its shape, we develop N2 which is an A3 pattern. Then the resulting chant using N1 followed by N2 is an appealing chant!

Thus appealing chants are generated by the use of a fixed shape. The resulting chants are homogeneous combinations. If the cardinality of the timing pattern used with N1 and N2 are equal the resulting chant is H1. Else, it is H2.

6.2. Generating Heterogeneous Chants.

We can start with a note string N1. Obtain its shape. We can generate a different shape by simple translation of notes, or addition/deletion of note(s).

The simple translation of an A2 notes pattern – say from CDE to DEG – yields N2 with different shapes but still of A2.

If timings with same cardinality are used to generate chants using N1 and N2, the resulting chant sequence will be H2. Else, the resulting chant will be R1.

If we add a note at the start (or at the end) of an A2 notes pattern – say from DEG to CDEG – we get a note pattern N2 with different shapes but still of A2.

When timings of same cardinality are used to generate chants using N1 and N2, the resulting chant sequence will be H2. Else, the resulting chant will be R1.

Skipping a note at the start or end again permits generation of chants as above.

Upon insertion/deletion of notes anywhere else we obtain R1 chants that are generic-appealing chants.

Thus, with the use of capabilities together with simple string manipulation, XCSS can be made to generate chants that range from simple to generic-appealing.

CSS and XCSS are in the implementation stage. Currently we aim to automate the development of chants using a select timing and note patterns.

Our ultimate goal is to make these systems real-time interactive.

7. Musical chants

Musical chants are extensions of chants with intonations, inflections, and harmonic notes together with syllables from spoken language.

Definition 16: A note n' that is an intonation, or an inflection, or in harmony with, a note n is said to be substitutable note for n.

Definition 17: A note n is said to be enhanced when a harmonic note/notes are superimposed with it.

Definition 18: A syllable (or a syllable sequence) is stretched by tailoring it to the timing pattern.

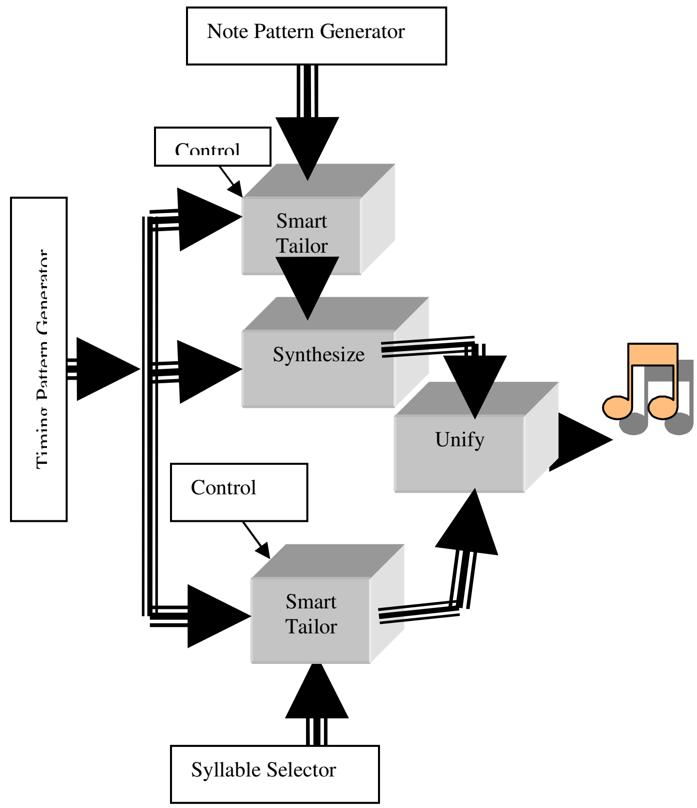

Figure 2: The architecture of a music generation system (MGS).

The substitutability principle is significant because it preserves the note pattern type after substitution. That is, using substitutable notes do not alter the shape of a pattern. Substitutability is used extensively to create melodies in Eastern music systems. The principle of enhancement is of essence in Western music systems where harmony is primal. As before, enhancing notes does not affect the note pattern types.

It is not necessary to mimic the tailoring of syllable to the tailoring of note patterns.

Definition 19: Stretching of a syllable is an extended use of a syllable beyond a chant.

Definition 20: The unification of stretched syllables on a chant is termed the singing of a chant.

When no stretching of syllables is required then we have elementary music. Otherwise we have advanced or improvised music.

A single syllable, single beat, single note can yield a primitive music like the crowing by a crow.

Knowledge of substitutable, enhanced, and harmonic notes together with the judicious use of different note and timing patterns can lead to the creation of orchestral compositions!

7. Conclusion

Based on the premise that timing and notes are fundamental to chant generation – we have proposed the concept of timing and note patterns. Based on these, we develop a chant synthesizing system (CSS). Mnemonic capabilities are applied by Intelligent Learning Systems to generate variety of chants. The use of musical features of intonations and harmony together with the use of syllables are integrated into a music generation system (MGS).

Our proposed framework can be implemented to generate trivial chants or to generate complex and composite music.

The authors are making a case that the generation of music – or speech in general using vocal and/or instrument – is a collective outcome of note pattern selection, timing pattern selection, and the resulting unification of the chants thereof with syllables.

References

[1] Sudharsan Iyengar, "A Theory on the Abstraction and Cognition Based on Musical Note Patterns", IC-AI-04, Las Vegas, June 2004.

[2] Eduardo Reck Miranda: Regarding Music, Machines, Intelligence and the Brain: An Introduction to Music and AI. Readings in Music and Artificial Intelligence 2000: 1-13

[3] Miranda, E. R., Kirby, S., and Todd, P. M. (2003). On Computational Models of the Evolution of Music: From the Origins of Musical Taste to the Emergence of Grammars. *Contemporary Music Review*, 22(3):91--111.

[4] Mira Balaban, Kemal Ebcioglu, and Otto Laske, editors, "Understanding Music with AI: Perspectives on Music Cognition", MIT Press, Cambridge, MA, 1992. 512 pages, ISBN 0-262-52170-9.

[5] Spector, L., Alpern, A., "Induction and Recapitulation of Deep Musical Structures," Proceedings of the IJCAI-95 Workshop on *Music* and AI.