Finite automata and morphisms in assisted musical composition

(Short title: Finite automata in musical composition)

Jean-Paul Allouche[1] and Tom Johnson[2]

Abstract: Finite automata originated in computer science and linguistics. They have applications in mathematics and physics and they have already been used in assisted musical composition. We show in this paper how finite automata (or more generally locally catenative formulas) underlie the structure of the *Formulas for String Quartet* of Tom Johnson.

[1] C. N. R. S., LMD, Luminy Case 930, F-13288 Marseille Cedex 9, France
allouche@lmd.univ-mrs.fr
[2] 75 rue de la Roquette, F-75011 Paris, France

1. Introduction

As the second author was composing his *Formulas for String Quartet* he was using some kind of heuristical rules, which he wanted to make more explicit, or more formally defined. After having had some hints by two mathematicians, indeed M. Waldschmidt in Paris and D. Feldman at the University of New Hamsphire, he met the first author who proposed the notion of finite automaton, or equivalently of morphism (of the free monoid): this tool permitted a unified description of the intuitive rules which were used but also helped the composer to clarify and finish some of the pieces. We want to explain quickly what these finite automata are, in what kind of fields they are used (in mathematics, computer sciences or physics), and to give some examples of their possible use in musical composition.

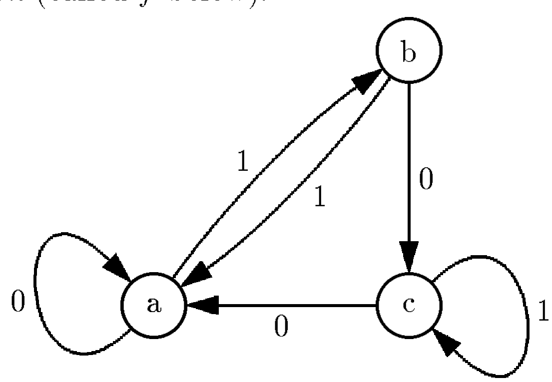
2. A quick overview on finite automata

A finite automaton is a sort of abstract machine (see figure below) consisting of:

- a finite set of *states*, (here the set of states is $\{a, b, c\}$), one of the states (say a here) being called the *initial state*,

- two families of arrows labelled respectively 0 and 1, the *transitions*, starting from each state and going to some other state (see picture),

- an *output function* (called f below).



$$f(a) = 0, \quad f(b) = 0, \quad f(c) = 1.$$

This automaton works as follows: for each integer n, say $n = 13$, one writes n base 2, here 1101, then one feeds the automaton with the digits of n read backwards, starting from the initial state and following the arrows according to the digits. Here one obtains successively:

$$1101.a = 110.b = 11.c = 1.c = c.$$

The state obtained after reading all the digits is then sent to some value by means of the output function, here $c \to 1$.

If one does this job for every integer (written base 2) i. e.

$$
\begin{aligned}
0: &\quad 0.a &= a \longrightarrow 0,\\
1: &\quad 1.a &= b \longrightarrow 0,\\
2: &\quad 10.a &= b \longrightarrow 0,\\
3: &\quad 11.a &= a \longrightarrow 0,\\
4: &\quad 100.a &= b \longrightarrow 0,\\
5: &\quad 101.a &= c \longrightarrow 1,\\
6: &\quad 110.a &= a \longrightarrow 0,\\
7: &\quad 111.a &= b \longrightarrow 0,\\
8: &\quad 1000.a &= b \longrightarrow 0,\\
9: &\quad 1001.a &= b \longrightarrow 0,\\
10: &\quad 1010.a &= c \longrightarrow 1,\\
11: &\quad 1011.a &= b \longrightarrow 0,\\
&\quad \cdots,
\end{aligned}
$$

one obtains a sequence of 0's and 1's, indeed $0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0 \cdots$, which is said to be generated by this automaton, or for short "automatic".

Another notion which we would like to allude to is the notion of *morphism of constant length*, also called *substitution of constant length*. Let $\mathcal{A}$ be an *alphabet*, i. e. a finite set, say $\mathcal{A} = \{a, b, c\}$. A morphism of say length 2 on $\mathcal{A}$ is a map σ which associates with each letter in $\mathcal{A}$ some two-letter word on $\mathcal{A}$. For example:

$$
\begin{aligned}
\sigma(a) &= ab,\\
\sigma(b) &= ca,\\
\sigma(c) &= ba.
\end{aligned}
$$

The image of a word, say $abcca$, is obtained by concatenating the images of its letters: $\sigma(abcca) = \sigma(a)\sigma(b)\sigma(c)\sigma(c)\sigma(a) = abcababaab$.

If the image of some letter begins with this letter, one notices that iterating the morphism starting from this letter yields longer and longer words, each of which begins with the previous one:

$$
\begin{aligned}
\sigma^1(a) &= \sigma(a) &&= ab,\\
\sigma^2(a) &= \sigma(\sigma(a)) &&= \sigma(ab) &&= abca,\\
\sigma^3(a) &= \sigma(\sigma^2(a)) &&= \sigma(abca) &&= abcabaab,\\
&\cdots
\end{aligned}
$$

and so the sequence of words $\sigma^n(a)$ converges to a limit,

$$abcabaabcaababca\cdots$$

which is, by construction, a fixed point of σ.

If one allows moreover to take a pointwise image of this sequence, say $a \to 0$, $b \to 1$, $c \to 1$, one obtains the sequence:

$$0110100110010110\cdots$$

which is said to be the image of a fixed point of a (this) morphism of constant length.

Remarks

- The terminology *constant length* comes from the fact that the images of the different letters under the substitution σ all have the same length (here length 2).

- One can replace the length 2 by a length k, where k is any integer greater than or equal to 2. As far as automata are concerned the above definition is actually the definition of a 2-automaton. One can also define a k-automaton, the only difference being that it reads integers written in base k instead of base 2 and that there are k families of arrows, labelled $0, 1, \cdots, k-1$; a sequence generated by a k-automaton is called k-automatic.

- A theorem of Cobham (1972) asserts that these two ways of generating sequences are equivalent: *a sequence is k-automatic if and only if it is the pointwise image of a fixed point of a morphism of length k.*

- **Locally catenative sequences**. An apparently different way of constructing sequences is the following: starting from a word w, one applies a certain function f, for instance $f(w) = w1\overline{w}$ where $\overline{w}$ is obtained from w by interchanging 0's and 1's. Starting from 0 and iterating f gives the words

$$0 \longrightarrow 011 \longrightarrow 0111100\cdots,$$

hence the infinite sequence:

$$011110011000011\cdots$$

Such a sequence is called a *locally catenative sequence*, and it has been proved by Shallit (1988) that, under some technical hypotheses, the locally catenative sequences are the sequences which can be constructed by taking the pointwise image of a fixed point of a morphism (possibly of non-constant length).

As an illustration of this last fact, let us define the words w_n by:

$$w_0 = 0, \quad w_1 = 01, \quad \forall n \geq 0, \; w_{n+2} = w_{n+1}w_n.$$

Hence

$$w_2 = w_1 w_0 = 010, \quad w_3 = w_2 w_1 = 01001, \cdots$$

The reader can check (and prove) that the sequence of words $(w_n)_n$ converges to an infinite sequence of 0's and 1's, which can be also obtained as the fixed point of the following morphism (which is not of constant length):

$$0 \to 01, \quad 1 \to 0.$$

This morphism is called the *Fibonacci morphism* and we will see below that it occurs in Physics.

3. Automata in mathematics, computer science and physics

Finite automata have been introduced in computer science and linguistics to give a (simple) model of extremely simple computers, but also in relation with formal grammars to answer questions like: is it possible to generate automatically all syntactically correct sentences in English (see Chomsky et al., 1963)?

We will survey here very quickly some applications of finite automata in mathematics, theoretical computer science and physics.

- An example in theoretical computer science: is it possible to construct a very long (infinite) sequence on a two-letter alphabet having no cubes (i. e. no three consecutive identical blocks) in it? A positive answer has been given by Thue in 1906: take the fixed point of the morphism $0 \to 01$, $1 \to 10$, yielding the (Prouhet-Thue-Morse) sequence,

$$0\,1\,1\,0\,1\,0\,0\,1\,1\,0\,0\,1\cdots$$

- An example of simply generated transcendental number: remember that a transcendental number is a number which is not a root of a polynomial with integer coefficients, (e and π are transcendental numbers). Now take the above binary sequence, insert a decimal point, yielding the real number

$$0.11010011001\cdots$$

It can be proved that *this number is transcendental.*

- In 1984, Shechtman, Blech, Gratias and Cahn discovered an alloy of aluminium and manganese (Shechtman et al., 1984) which is neither disordered nor periodic, i. e. neither a glass nor a crystal. They called this kind of alloy a *quasi-crystal*, and observed that its underlying structure is given by the Fibonacci morphism (see above): $0 \to 01$, $1 \to 0$. Then the automatic sequences defined above have been used to produce structures with a kind of "controlled disorder".

For more information on these different subjects one can read (Allouche 1987) and (Allouche 1989).

4. Music entering the picture

Some musicians have already used finite automata to produce musical shapes: for instance A. Mouret used morphisms to obtain rhythmic structures (see Allouche et al., 1989) and M. Frémiot (1993 and 1994) composed a mass using the *towers of Hanoi*, which can be shown to involve finite automata. These automata have also been used to generate Indian *ragas* (see Upadhye et al., 1992). On the other hand B. Bel (1987 and 1992) used some kinds of more general automata (actually formal grammars) to generate Indian *qa'idas*.

The second author of this paper has used automata, morphisms or locally catenative sequences in his *Formulas for String Quartet*, and we give below a survey of the explanatory pages of the score.

5. The Formulas for String Quartet

To compose the *Formulas for String Quartet* the second author used first intuitively, then more explicitly, automata, morphisms or locally catenative sequences. We would like to give a quick description of the tools which were used for composing four of the eight movements. For more details or explanations, as well as to see the music (!), one can read the complete score (Johnson 1994).

- Movement I is based on the following morphism of length 3 defined on the alphabet $\{-,+\}$,

$$+ \longrightarrow + - +$$
$$- \longrightarrow - - +.$$

The formula is followed simultaneously by the four instruments with tempo proportions $1:3:9:27$ and interval proportions of $1:2:3:4$. The symbol $+$ indicates a melodic ascent, $-$ a melodic descent.

VN1	+1	−1	+1	−1	−1	+1	+1	−1	+1	−1	−1	+1	−1	−1	+1	+1	−1	+1	+1	−1	+1	−1	−1	+1	+1	−1	+1
VN2	+2			−2			+2			−2			−2			+2			+2			−2			+2		
VLA	+3									−3									+3								
VCL	+4																										

- Movement II is based on a locally catenative sequence: define f as the function which transposes each note two scale degrees higher. Then define a sequence of words w_n by

$$w_1 = 12, \quad w_2 = 234, \quad \forall n \geq 0, \; w_{n+2} = f(w_n) f(w_{n+1}).$$

This formula is scored as a canon for the four instruments, with an additional rhythmic element which embeds the new words into the old. Beginning in measure 7, with all four instruments present, one can observe four levels at once:

VN1	(w_4)				4		5	6				5	6			6		7	8
VN2	(w_3)			3		4					4				5		6		
VLA	(w_2)		2							3				4					
VCL	(w_1)	1							2										

Note that this alphabet seems to be infinite, though it too could be analyzed as a finite number of specific transformations. Note also that Movement IV produces a very different kind of music although the underlying mathematical formula is identical.

- Movement VII is constructed by means of the "reverse paperfolding formula", (indeed the operation of repeatedly folding a piece of paper can be simulated by means of a finite automaton or of a locally catenative sequence):

$$w_1 = 1, \quad w_2 = 011, \quad \cdots \quad \forall n \geq 0, \; w_{n+1} = \overline{w_n} \; 1 \; h(w_n),$$

where the word $\overline{w}$ is the word w where the 0's have been replaced by 1's and *vice versa* and $h(w)$ is the word w read backwards.
In this case the first note in each part is a neutral point of departure, not counted in the sequence. The instruments proceed in tempos of 1 : 2 : 4 : 8, moving up a semi-tone for each 1 and down a semi-tone for each 0. The first violin and the cello play the subsequence with odd indices $(w_{2n+1})_n$. This has the effect of beginning 100 instead 011, which is musically an inversion. As a consequence all simultaneous moves in the instruments are in the same direction.

VN1	x		1		0		0		1		1		1		0		0		1		0		0		0		1		1		0	
VN2	x				0				1				1				0				0				0				1			
VLA	x	0	1	1	0	0	0	1	1	0	1	1	1	0	0	1	0	0	1	1	0	0	0	1	0	0	1	1	1	0	0	1
VCL	x								1								0								0							

- Movement VIII (the final movement) is constructed as follows: let g be the function which replaces the word w by the word obtained from it after letting each note be followed by the note three semi-tones higher, and let h be the function which replaces the word w by letting each note be followed by the note two semi-tones higher. Now define

$$w_1 = 023, \quad \forall n \geq 1, \ w_{2n} = g(w_{2n-1}), \ w_{2n-1} = h(w_{2n-2}),$$

hence one has:

$$w_2 = 032536, \quad w_3 = 023524573568, \ \cdots$$

The following illustration, beginning in measure 6, shows the formula moving in three instruments simultaneously:

VN2	(w_5)	0	2	3	5	2	4	5	7	3	5	6	8	5	7	8	10
VLA	(w_4)	0		3		2		5		3		6		5		8	
VCL	(w_3)	0				2				3				5			
		2	4	5	7	4	6	7	9	5	7	8	10	7	9	10	12
		2		5		4		7		5		8		7		10	
		2				4				5				7			
		3	5	6	8	5	7	8	10	6	8	9	11	8	10	11	13
		3		6		5		8		6		9		8		11	
		3				5				6				8			

6. Conclusion

These particular sequences might be used further in music to help the composer to build musical shapes. The properties of theses sequences, which are already used in mathematics, computer science and physics, are that (except for those which are ultimately periodic) they lie somewhere between regularity and chaos, between order and disorder, between periodicity and randomness.

However we would like to say that this can only be a tool and certainly not a way ("the" way) of composing music in a purely "automatic" fashion. The introduction of the score for the *Formulas for String Quartet* ends as follows (Johnson 1994),

I should add that this final movement had a long and trouble evolution. The first sketches go back to 1982, *and there must have been at least ten subsequent versions. Why, in January* 1993, *as I was completing the composition, did I finally decide that one of the versions was really better than the others, and that I would conclude the* Formulas *with this little piece? Was the logic really clearer? Were the harmonies really richer? I must say that the decision was clearly subjective, despite my obsession for objectivity and logic. But of course, even in activities as logical as choosing a chess move or finding the best notation for a mathematical formula, one talks often of style, taste, elegance, and other things that cannot be deduced from premises. I too have to rely on taste and instincts, and I can never prove that this version is better than the others, and finally this piece is not so much* Formulas *as simply music.*

References

Allouche, J.-P. (1987). Automates finis en théorie des nombres. *Expositiones Mathematicae*, 5, 239–266.

Allouche, J.-P. (1989). Finite automata in 1-D and 2-D Physics. In *Number Theory and Physics, Proceedings of the Winter School, Les Houches, France, March 7–16 1989*, J.-M. Luck, P. Moussa and M. Waldschmidt editors, Springer Proceedings in Physics, volume 47, Springer.

Allouche J.-P. and Mouret A. (1989). Libertés non anarchiques, automates finis et champs matriciels. In *Actes du Colloque "Structures Musicales et Assistance Informatique", Composition et Analyse, Marseille 1988.* M.I.M., Marseille.

Bel B. (1987). Grammaires de génération et de reconnaissance de phrases rythmiques. In *6e congrès de l'AFCET. Reconnaissance des Formes et Intelligence Artificielle*, (pp. 353–366). INRIA, Antibes.

Bel B. (1992). Modelling improvisatory/compositional processes. In *Languages of Design – Formalisms for Word, Image and Sound Synthesis, 1*, (pp. 11–26). Elsevier Science Publishers, Amsterdam.

Chomsky N. and Schützenberger M. P. (1963). The algebraic theory of context-free languages. In *Computer programming and formal languages* (pp. 118–161). North Holland, Amsterdam.

Cobham A. (1972). Uniform tag sequences. *Mathematical Systems Theory*, 6, 164–192.

Johnson T. (1994). Formulas for String Quartet. *Score published by* Éditions 75, 75 rue de la Roquette 75011 Paris, and Two-Eighteen Press, P. O. Box 218, Village

Station, New York, NY 10014. *The* Formulas *were premiered by the Brindisi Quartet on March 6, 1994 in a concert sponsored by the Swiss radio in Bern.* Performance materials are available from both Éditions 75 and the Two-Eighteen Press.

Frémiot M. (1993). *Messe pour orgue à l'usage des paroisses* (Compact Disc Coriolan COR 318 301). ARCAM Provence Alpes Côte d'Azur.

Frémiot M. (1994). *Messe pour orgue à l'usage des paroisses* (score including a detailed analysis). CY, Marseille.

Shallit J. (1988). A generalization of automatic sequences. *Theoretical Computer Science*, 61, 1–16.

Shechtman D., Blech I. A., Gratias D. and Cahn J. W. (1984). Metallic phase with long-range orientational order and no translational symmetry, *Physical Review Letters*, 53, 1951–1953.

Upadhye R. and Saharasbuddhe H. V. (1992). The role of consonances in Hindustan classical music. In *Workshop on Artificial Intelligence and Music, 10^{th} European Conference on Artificial Intelligence and Music (ECAI)*, Vienne.