A VARIANT OF HOFSTADTER’S SEQUENCE AND FINITE AUTOMATA

JEAN-PAUL ALLOUCHE and JEFFREY SHALLIT

(Received 26 May 2011; accepted 16 January 2012)

Communicated by F. Pappalardi

In memory of Alf van der Poorten: colleague, connoisseur, raconteur, friend

Abstract

Following up on a paper of Balamohan et al. [‘On the behavior of a variant of Hofstadter’s \(q\)-sequence’, J. Integer Seq. 10 (2007)], we analyze a variant of Hofstadter’s \(Q\)-sequence and show that its frequency sequence is 2-automatic. An automaton computing the sequence is explicitly given.


Keywords and phrases: Hofstadter’s sequence, recurrence, automatic sequence, finite automaton.

1. Introduction

In his 1979 book Gödel, Escher, Bach [7], Douglas Hofstadter introduced the sequence \(Q(n)\) defined by the recursion

\[
Q(n) = Q(n - Q(n - 1)) + Q(n - Q(n - 2))
\]

for \(n \geq 2\) and \(Q(1) = Q(2) = 1\). Although it has been studied extensively (for example, [9]), still little is known about its behavior, and it is not mentioned in standard books about recurrences (for example, [6]). It is sequence A005185 in Sloane’s Encyclopedia [11].

Twenty years later, Hofstadter and Huber introduced a family of sequences analogous to the \(Q\)-sequence, and defined by the recursion

\[
Q_{r,s}(n) = Q_{r,s}(n - Q_{r,s}(n - r)) + Q_{r,s}(n - Q_{r,s}(n - s))
\]

for \(n > s > r\) (private communication cited in [4]). The case \(r = 1, s = 4\) is of particular interest.
1. The sequence \( V \).

| \( n \) | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| \( V(n) \) | 1 | 1 | 1 | 1 | 2 | 3 | 4 | 5 | 5 | 6 | 6 | 7 | 8 | 8 | 9 | 9 | 10 |

2. The sequence \( F \).

| \( n \) | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| \( F(n) \) | 4 | 1 | 1 | 1 | 2 | 2 | 1 | 2 | 2 | 1 | 3 | 2 | 1 | 2 | 2 | 1 | 3 |

Recently, Balamohan et al. [4] gave a nearly complete analysis of the sequence \( Q_{1,4} \) (called \( V \) in their paper). It is defined by

\[
V(1) = V(2) = V(3) = V(4) = 1 \\
V(n) := V(n - V(n - 1)) + V(n - V(n - 4))
\]

for \( n > 4 \). Table 1 is a short table of the sequence \( V \) (sequence A063882 in Sloane’s Encyclopedia [11]).

Among the results of Balamohan, Kuznetsov, and Tanny is a precise description of the ‘frequency’ sequence \( F \) defined by

\[
F(a) := \#\{n : V(n) = a\}.
\]

Table 2 is a short table of the sequence \( F \) (sequence A132157 in Sloane’s Encyclopedia [11]). In particular, they proved the following theorem [4, Lemmas 13–19 and Table 5].

**Theorem 1.1** (Balamohan, Kuznetsov, Tanny). There exist two (explicit) maps \( g, h \), with \( g, h : \{1, 2, 3\}^4 \to \{1, 2, 3\} \), such that, for all \( a > 3 \)

\[
F(2a) = g(F(a - 2), F(a - 1), F(a), F(a + 1)), \\
F(2a + 1) = h(F(a - 2), F(a - 1), F(a), F(a + 1)).
\]

(We note that in [4, Lemma 13], the quantifiers \( a \geq 3 \) for the equality \( F(2a) = 2 \) and \( a \geq 4 \) for the equality \( F(2a + 1) = 2 \) should have been mentioned.)

In this paper we prove that the sequence \( (F(n))_{n \geq 1} \) is 2-automatic, which means essentially that \( F(n) \) can be computed ‘in a simple way’ from the base-2 representation of \( n \). In particular, it can be computed in \( O(\log n) \) time. Furthermore, we give the automaton explicitly. For definitions and properties of automatic sequences, the reader is referred to [2]. For some recent related papers, see [5, 8, 10].

2. The main result

We begin this section with a general result on automatic sequences. Before stating the theorem we need some notation.
**Definition 2.1.** Let \( U = (U(n))_{n \geq 0} \) be a sequence. Let \( \alpha \) be an integer. We let \( U^\alpha \) denote the sequence defined by

\[
U^\alpha(n) := U(n + \alpha),
\]

where \( n \geq -\alpha \). For positive integers \( q, i, j \), let \( U_{q,i,j} \) be the subsequence of \( U \) defined by

\[
U_{q,i,j}(n) = U(q^i n + j),
\]

where \( n \geq 0 \).

For integers \( q \geq 2 \) and \( t \geq 1 \), let \( m(q, t) = q^{t+1} - 1 \) and \( s(q, t) = (q^{t+1} - 1)/(q - 1) \), so that \( s(q, t) \) counts the number of integer pairs \((i, j)\) with \( 0 \leq i \leq t \) and \( 0 \leq j \leq q^i - 1 \). In the proof below it will be convenient to fix some ordering of such pairs \((i, j)\) and then enumerate the corresponding sequences \( U_{q,i,j} \) as \( U_1 = U, U_2, \ldots, U_{s(q,t)} \).

**Theorem 2.2.** Let \( (U(n))_{n \geq 0} \) be a sequence with values in a finite set \( \mathcal{A} \) and let \( q \geq 2 \) be an integer. Then \( (U(n))_{n \geq 0} \) is \( q \)-automatic if there exist nonnegative integers \( t, a, b, n_0 \) and a family \( \{f_j\}_{j=0}^{m(q,t)} \) of functions from the set \( \mathcal{A}^{a+b+s(q,t)} \) to \( \mathcal{A} \) such that for all \( j \in [0, m(q, t)] \) and all \( n \geq n_0 \),

\[
U(q^{t+1}n + j) = f_j(U^{-a}(n), \ldots, U^{-1}(n), U^0(n), U^1(n), \ldots, U^b(n), U_2(n), U_3(n), \ldots, U_{s(q,t)}(n)).
\]

**Proof.** To prove that the sequence \( U = (U(n))_{n \geq 0} \) is \( q \)-automatic, it suffices to find a finite set of sequences \( \mathcal{E} \) that contains \( U \), such that if \( V = (V(n))_{n \geq 0} \) belongs to \( \mathcal{E} \), then, for any \( r \in [0, q - 1] \), the sequence \( (V(qn + r))_{n \geq 0} \) also belongs to \( \mathcal{E} \). Fix two positive integers \( K \) and \( L \) such that \( K \geq \max(n_0, q(a + 1)/(q - 1)) \) and \( L \geq q(b + 1)/(q - 1) \). Recall that the sequence \( U^a_k \) is defined by \( U^a_k(n) := U_k(n + k) \).

Let \( \mathcal{E} \) be the (finite) set of sequences defined by

\[
V \in \mathcal{E} \iff \exists \ell \in [1, s(q, t)], \exists k \in [-K, L], \forall n \geq A, V(n) = U^a_k(n).
\]

Now let \( V \) be a sequence in \( \mathcal{E} \). Take \( r \in [0, q - 1] \). There exist \( \ell \in [1, s(q, t)] \) and \( k \in [-K, L] \) such that for all \( n \geq A \),

\[
V(qn + r) = U^a_k(qn + r) = U_\ell(qn + r + k).
\]

Hence, for some \( i \leq t \) and \( j \in [0, q^i - 1] \),

\[
V(qn + r) = U(q^i(qn + r + k) + j).
\]

By the division algorithm, write \( q^i(r + k) + j = q^{i+1}x + y \), with \( x \in \mathbb{Z} \) and \( 0 \leq y \leq q^{i+1} - 1 \) (so that, in particular, \( y \leq m(q, t) \)). Then

\[
V(qn + r) = U(q^{i+1}(n + x) + y).
\]
Note that
\[ q^{i+1}x \leq q^{i+1}x + y = q^i(r + k) + j < q^i(r + k + 1) \]
and
\[ q^{i+1}x = q^i(r + k) + j - y > q^i(r + k) - q^{i+1}. \]
Hence,
\[ (r + k - q)/q < x < (r + k + 1)/q. \]

We distinguish two cases.

**Case 1.** \( i < t. \) Then \( i + 1 \leq t. \) Thus there exists \( \ell' \in [1, s(q, t)] \) such that, for \( n \geq A, \)
\[ V(qn + r) = U(q^{i+1}(n + x) + y) = U^\ell(n + x) = U^\ell(n). \]

Now
\[ x > (r + k - q)/q \geq (r - K - q)/q \geq (-K - q)/q \geq -K \]
(since \( K \geq q(a + 1)/(q - 1) \geq q/(q - 1) \)), and
\[ x < (r + k + 1)/q \leq (q + L)/q \leq L \]
(since \( L \geq q(b + 1)/(q - 1) \geq q/(q - 1) \)). This shows that the sequence \((V(qn + r))_{n \geq 0}\)
belongs to \( \mathcal{E} \).

**Case 2.** \( i = t. \) Then \( i + 1 = t + 1. \) From the hypothesis and the condition \( K \geq n_0, \)
we can write, for \( n \geq K, \)
\[ V(qn + r) = U(q^{i+1}(n + x) + y) = f_y(U^{x-a}(n), \ldots, U^y(n), U^{x+1}(n), \ldots, U^{x+b}(n), U^x_2(n), U^x_3(n), \ldots, U^x_{s(q,t)}(n)). \]

To prove that the sequence \((V(qn + r))_{n \geq 0}\) belongs to \( \mathcal{E} \), it suffices to prove that all
sequences \( U^\beta \) for \( \beta \in [x - a, x + b] \) and all sequences \( U^\ell \) for \( \ell \in [1, (q^{i+1} - 1)/(q - 1)] \)
belong to \( \mathcal{E} \), and to use composition of maps. But
\[ \beta \geq x - a > (r + k - q)/q - a \geq (-K - q)/q - a \geq -K \]
(recall that \( K \geq q(a + 1)/(q - 1) \)) and
\[ \beta \leq x + b < (r + k + 1)/q + b \leq (q + L)/q + b \leq L \]
(recall that \( K \geq q(b + 1)/(q - 1) \)). This implies that all sequences occurring in the
arguments of \( f_y \) above belong to \( \mathcal{E}. \)

**Remark 2.3.** Theorem 2.2 above is similar to (but different from) [3, Theorem 6, p. 5] on \( k \)-regular sequences. That theorem implies Theorem 2.2 above in the case where the
maps \( f_j \) are linear.

**Corollary 2.4.** The sequence \( F = (F(n))_{n \geq 0} \) is 2-automatic.

**Proof.** It suffices to use the theorem recalled in the first section, after having extended
the sequence \( F \) by \( F(0) = 0. \)
3. An explicit automaton

In this section we provide an explicit automaton\(^1\) to calculate the sequence \(F\).

The automaton is constructed in two stages. First, we give an automaton \(A\) with the
property that reading \(n\) in base 2 takes us to a state \(q\) with the property that the four
values \(F(n + a)\) for \(-2 \leq a \leq 1\) are completely determined by \(q\). Next, we show that \(A\)
can be minimized to give an automaton \(B\) computing \(F(n)\). We remark that we assume
throughout that the automaton reads the ordinary base-2 representation of \(n\) from ‘left
to right’, ending at the least significant digit, although we do allow the possibility of
leading zeros at the start.

Let us start with the description of \(A = (Q, \Sigma, \Delta, \delta, q_0, \tau)\). The machine \(A\) has 33
states with strings as names; \(\Sigma = \{0, 1\}; \Delta = \{0, 1, 2, 3, 4\}\), \(q_0 = \epsilon\). The transition
function \(\delta\) and the output map \(\tau\) are given in Table 3.

We introduce some notation. Let \([w]\) denote the integer represented by the binary
string \(w\) in base 2. Thus, for example, \([00110] = [110] = 6\). Note that \([\epsilon] = 0\),
where \(\epsilon\) denotes the empty string. If \(F\) is our sequence defined above, then by
\(F(a.a + i − 1)\) we mean the string of length \(i\) given by the values of the function \(F\)
at \(a, a + 1, \ldots, a + i − 1\).

Our intent is that if \(w\) is a binary string, then \(\tau(\delta(q_0, w))\) is the string of length four
given by \(F(n − 2..n + 1)\), where \(n = [w]\). (Note: we define \(F(0) = F(−1) = F(−2) = 0\).

To prove that this automaton computes \(F(n)\) correctly, it suffices to show that:
\(\delta\) for each state \(q\) we have \(\tau(q) = F([q] − 2)F([q] − 1)F([q])F([q] + 1)\); and
\(\delta\) if \(p = \delta(q, a)\) for two states \(p, q \in Q\) and \(a \in \{0, 1\}\), then \(F([px]) = F([qax])\) for
all strings \(x\).

Part (a) can be verified by a computation, which we omit. For example,
since \([1110011111]\) = 463, the claim \(\tau([1110011111]) = 2133\) means \(F(461..464) = 2133\),
which can easily be checked.

Part (b) requires a tedious simultaneous induction on all the assertions, by induction
on \(|x|\). Not surprisingly, we omit most of the details and just prove a single
representative case.

Consider the transition \(\delta(100, 1) = 110\). Here we must prove that
\(F([1001x]) = F([110x])\) for all strings \(x\). We do so by induction on \(x\). The base case is \(x = \epsilon\), and we have
\(F([1001]) = F(9) = 2\) and \(F([110]) = F(6) = 2\).

For the induction step, we use the fact that \([4, Table 5]\) shows that \(F(2a)\) and
\(F(2a + 1)\) are completely determined by \(F(a − 2), F(a − 1), F(a),\) and \(F(a + 1)\). It
thus suffices to check that \(F([1001x] + a) = F([110x] + a)\) for \(-2 \leq a \leq 1\); doing so
will then prove (3.1) for \(x0\) and \(x1\), thus completing the induction.

The only cases that require any computation are when \([x] = 0\) and \(a = −1, −2,\)
or \([x] = 1\) and \(a = −2,\) or \(x\) is a number of the form \(2^j − 1\) for some \(j \geq 1\) and \(a = 1\).

\(^1\) In honor of Alf van der Poorten, we cannot resist quoting Voltaire: ‘Impuissantes machines/Automates
pensants mus par des mains divines’.
Case 1. $x = 0^j$ for some $j \geq 0$. If $j = 0$ then this is the assertion that $F([1001] + a) = F([110] + a)$ for $-2 \leq a \leq 1$, which is the same as the claim that $F(7..10) = F(4..7)$. But $F(7..10) = 1221 = F(4..7)$.

Otherwise $j \geq 1$. Then $[1001]x - 1 = [10010^j] - 1 = [10001]$, and $[110]x - 1 = [1100^j] - 1 = [1011]$. Now by induction we have $F([10001^j]) = F([100011^{j-1}]) = F([10111^{j-1}]) = F([1011^j])$, as desired.

Similarly, $[1001]x - 2 = [10010^j] - 2 = [10011^{j-1}0]$. Also, $[110]x - 2 = [1100^j] - 2 = [101]$. Then by induction we have $F([1001^{j-1}0]) = F([10011^{j-2}0]) = F([101^j0])$, as desired.
Case 2. \( x = 0^j1 \) for some \( j \geq 0 \). Then \([1001x] - 2 = [10010^j1] - 2 = [10001^j+1] \).
Also \([110x] - 2 = [1100^j1] - 2 = [101^j+2] \). By induction we have \( F([10001^j+1]) = F([10001^j]) = F([101^j+2]) \), as desired.

Case 3. \( x = 1^j \) for some \( j \geq 1 \). Then \([1001x] + 1 = [1010^j+1] \). Similarly \([110x] + 1 = [1101^j] + 1 = [1110^j] \). By induction we have \( F([1010^j+1]) = F([101000^j]) = F([11100^j]) \), as desired.

This completes the proof of correctness of a single transition.

Ultimately, we are not really interested in computing \( \tau(q) \), but only the image of \( \tau(q) \) formed by extracting the third component, which is the one corresponding to \( F(n) \). This means that we can replace \( \tau \) by \( \tau' \), which is the projection of \( \tau \) along the third component. In doing so some of the states of \( A \) become equivalent to other states. We can now use the standard minimization algorithm for automata to produce the 20-state minimal automaton \( B = (Q', \Sigma, \Delta, \delta', q_0, \tau') \) computing \( F(n) \). Table 4 gives the names of the states of \( Q \), and \( \delta' \) and \( \tau' \) for these states.

### Table 4. The automaton \( B \).

<table>
<thead>
<tr>
<th>( q )</th>
<th>( \delta'(q, 0) )</th>
<th>( \delta'(q, 1) )</th>
<th>( \tau'(q) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \epsilon )</td>
<td>( \epsilon )</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>11</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>101</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>110</td>
<td>111</td>
<td>1</td>
</tr>
<tr>
<td>100</td>
<td>101</td>
<td>110</td>
<td>1</td>
</tr>
<tr>
<td>101</td>
<td>1010</td>
<td>1011</td>
<td>2</td>
</tr>
<tr>
<td>111</td>
<td>1110</td>
<td>110</td>
<td>1</td>
</tr>
<tr>
<td>1010</td>
<td>1110</td>
<td>10101</td>
<td>1</td>
</tr>
<tr>
<td>1011</td>
<td>1011</td>
<td>1100</td>
<td>3</td>
</tr>
<tr>
<td>1100</td>
<td>1101</td>
<td>1110</td>
<td>2</td>
</tr>
<tr>
<td>1101</td>
<td>11010</td>
<td>11011</td>
<td>1</td>
</tr>
<tr>
<td>1110</td>
<td>11100</td>
<td>1011</td>
<td>2</td>
</tr>
<tr>
<td>10101</td>
<td>1110</td>
<td>10101</td>
<td>2</td>
</tr>
<tr>
<td>11010</td>
<td>1110</td>
<td>110</td>
<td>2</td>
</tr>
<tr>
<td>11011</td>
<td>111</td>
<td>101</td>
<td>2</td>
</tr>
<tr>
<td>11100</td>
<td>11010</td>
<td>111001</td>
<td>1</td>
</tr>
<tr>
<td>111001</td>
<td>1010</td>
<td>111001</td>
<td>2</td>
</tr>
<tr>
<td>110011</td>
<td>1011</td>
<td>1110011</td>
<td>3</td>
</tr>
<tr>
<td>11100111</td>
<td>11100</td>
<td>1110011</td>
<td>2</td>
</tr>
</tbody>
</table>

4. Concluding remarks

It would be interesting to know whether the first difference sequence of the variant of Hofstadter’s, that is, the sequence \( (V(n+1) - V(n))_{n \geq 0} \), is also 2-automatic. We already know that it takes only finitely many values [4, Theorem 1, p. 5]. Of course it might well be the case that this sequence is not automatic: in a very different
context, think of the classical Thue–Morse sequence which is 2-automatic, but whose run-length sequence is not [1]. It would be also interesting to determine for which sequences $Q_{r,s}$ (with the notation in the introduction) the frequency sequence is automatic.

**Acknowledgement**

We thank the referees for valuable comments.

**References**


JEAN-PAUL ALLOUCHE, CNRS, Institut de Mathématiques, Université Pierre et Marie Curie, Case 247, 4 place Jussieu, F-75752 Paris Cedex 05, France

e-mail: allouche@math.jussieu.fr

JEFFREY SHALLIT, School of Computer Science, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada

e-mail: shallit@cs.uwaterloo.ca