AN INTRINSICALLY NON MINIMAL-TIME MINSKY-LIKE 6-STATES SOLUTION TO THE FIRING SQUAD SYNCHRONIZATION PROBLEM

JEAN-BAPTISTE YUNÈS

Abstract. Here is presented a 6-states non minimal-time solution which is intrinsically Minsky-like and solves the three following problems: unrestricted version on a line, with one initiator at each end of a line and the problem on a ring. We also give a complete proof of correctness of our solution, which was never done in a publication for Minsky’s solutions.

1991 Mathematics Subject Classification. 65Y05,68Q25,68Q80,68W10.

1. Introduction

Among the variety of problems in cellular automata, the Firing Squad Synchronization Problem (FSSP) plays a very special role. It is one of the oldest and most vivid questions. Despite the fact that numerous solutions actually exist, it is very interesting to build new ones either to solve the problem in some new conditions (new communication graphs, faulty environments, etc), or to study how currently used algorithms really work, in order to better understand how we can compute with such a computation model. So is the quest for few states FSSP algorithms, for which the best records are the famous 6-states minimal-time solution on a line of cells (see [8]), the 6-states minimal-time solution on a rectangle (see [20]) and some other tricky 6-states solutions on a line (see [17]).

1.1. The problem

A one-dimensional cellular automata, CA, consists of an array (a line) of identical finite automata, each one being wired to its two direct neighbors. The whole
machine operates at discrete time-steps; every automaton reads its inputs (states of its neighbors and its own state) and changes state according to a given transition function. Its new state is then available to its neighbors at the next step. We define the quiescent state such that any quiescent cell remains quiescent when its neighborhood is quiescent. Then, we can define the Firing Squad Synchronization Problem, in short FSSP, as follows. Consider arrays of identical automata which are initially all in the quiescent state except the initiator (located at, say, the left end of the array and set up in a special initiator state). The FSSP is to design a transition function such that after some steps all cells are in the same state (firing state) which never occurred before. What is really challenging is that such a transition function mustn’t depend on the length of the line, and that the same function must be used for every possible finite line.

That problem has been studied long ago and has been first stated by J. Myhill in 1957 (as reported by E. Moore in 1964, see [11]) to synchronously start a copy of an original machine in a self-reproducing CA. Since then, a very rich set of solutions has grown up. Solutions to the original problem have been set up for a long time and the very first published solution is due to J. McCarthy & M. Minsky (in 1967, see [10]). E. Goto (in 1962, see [3]) has built a so-called minimal-time solution: one which takes only \(2n - 2\) steps to synchronize a line of \(n\) automata (one can easily realize that it is the minimal time required for an information to be sent from the initiator to the other end and get the answer back). After that and independently, A. Waksman and R. Balzer (in 1966 and 1967, see [1, 23]) have got minimal-time solutions with very few states (resp. 16 and 8 states). R. Balzer also proved that no 4-state minimal-time solution exists. A famous record is held by J. Mazoyer who exhibited a 6-states minimal-time solution (in 1987, see [8]) to the FSSP. Since then the design of a 5-states solution remains as an open problem. One can note that nobody knows how to solve that question: building a solution with few states seems to require new ideas.

Of course many very different variants of the original problem have been considered. One can change the underlying network of cells, extend the line to higher dimensions giving grids, cubes, etc (see [18–20]), generate different regular graphs like fireflies-graphs, Cayley graphs, etc (see [14, 15]), or even free many constraints (see [4–6, 16]) like communication delays, fan-outs, etc. One can also limit the bandwidth down to a single bit (see [9]). Another common modification to the original problem is to consider faulty environments, i.e. to consider that some cells are unable to compute the transition function, see [7, 21, 25]

1.2. Our contribution

In this paper we present a 6-state non minimal-time solution to the unrestricted problem inspired by a solution given by L. Pierre [13]. By unrestricted we mean that the initiator can be located at any end of the line. We will also show that the same transition function also solves the problem with initiators at both ends and on a ring. What is new in this solution is that it does not involve some “trick” in extending the 6-state minimal-time Mazoyer’s solution as does that in the paper
of A. Settle & J. Simon (see [17]). As illustrated in figure 1, their strategy is to delay the original Mazoyer’s solution when the initiator is located at the right: a right to left traversal is done before starting the original solution from the left. In case the initiator is at the left the original solution is launched as usual. What is tricky (and hard to do) in this solution is that some unused transitions of the original solution were used to produce the desired effect: the right-to-left signal is not obvious (see figure 1). This solution is thread-like and use $O(n \log n)$ state changes which makes it different from a 6-states solution designed by H. Umeo et al. (see [22]) which use $O(n^2)$ state changes.

![Figure 1. The Settle/Simon’s 6-states non minimal-time solution](image)

In contrast, our solution computes the Minsky strategy (see figure 2). This contradicts what Mazoyer sometimes claimed, namely that to minimize the number of states it is necessary to synchronize in minimal-time and to break the symmetry of the solution. Though very interesting, Settle and Simon’s solution did not violate this last assertion. Our solution does since it is intrinsically non minimal-time and symmetric. In this paper we will prove the correctness of our solution for the following problems:

- the restricted problem,
• the unrestricted problem,
• the problem with initiators at both ends,
• the problem for the ring.

As far as we know, there are only two formal proofs of the correctness of solution to the FSSP (see [8, 12]), moreover, Mazoyer’s proof has been verified with the help of the theorem prover Coq by J. Duprat (see [2]). This one is the first proof of a Minsky based solution.

2. A 6-state solution to the problem

We first remind the reader that Minsky’s strategy is to solve the problem with a “divide-and-conquer” method: cut the line into two equals parts and recursively compute it on each part (as illustrated in figure 2(a)).

![Figure 2. The Minsky’s solution](image)

Cutting the line into two equal parts is done using two basic signals launched from the initiator. The first one evolves at maximum speed from the initiator up to the other end of line and bounces back. Simultaneously, the second signal evolves from the initiator at \( \frac{1}{2} \) the maximum speed up to the moment it encounters the first one. This meeting takes place at the middle of the line, and that process is repeated again on each of the two new sub-lines. The end of the process is reached when automata detects that it is unnecessary or impossible to cut the line (one cannot cut a single automaton).
The complete transition function of our solution is given in figure 3. To ease the read of the table, we used the following conventions: the initiator is denoted by “A”, the quiescent state by “•”, the border by “$” and the firing state by “E”. The number of transitions is 132, which can be compared to the 119 rules solution by Mazoyer or Nogushi (see [8, 12]). Figure 2(b) shows how the process takes place on a line of 30 automata, and for convenience a different grey tone has been attributed to each significant state.

Figure 3. The 6-state Minsky-like transition function

3. The Proof of Correctness of the solution

In the following, we shall use $t : s_1...s_l$ to denote that automaton $i \in [1, l]$ at time $t \geq 0$ is in state $s_i \in \{•, A, B, C, D, E\}$. For convenience, we shall use $[i,j] \overrightarrow{s...s}$ to denote that each automaton $k \in [i,j]$ is in state $s$, and $⋆$ (the wildcard) to denote any suitable state.

**Definition 3.1.** The starting configuration of a Firing Squad is a line of $l$ automata so that

$0 : A \overrightarrow{\cdots}$

**Lemma 3.2 (First lines).** Our transition function synchronizes every line of length 1 ≤ $l$ ≤ 5.

**Proof.** The 5 runs (see figure 4) of the automaton constitute the proof. The rules used in these runs are: $A$ → $E$, $A • \rightarrow B$, $A • \rightarrow C$, $B C \rightarrow A$, $B C \rightarrow B$, $A B \rightarrow A$, $A B \rightarrow E$, $A A \rightarrow E$, $A \rightarrow C$, $A \rightarrow C$, $B C \rightarrow B$, $C B \rightarrow B$, $A B \rightarrow C$, $B \rightarrow B$, $A C \rightarrow C$, $A C \rightarrow D$, $C B \rightarrow A$, $C D \rightarrow C$, $C D \rightarrow B$, $D A \rightarrow C$, $C B \rightarrow B$, $C B \rightarrow A$, $B A \rightarrow A$, $A A \rightarrow E$, $\cdots \rightarrow E$, $C \rightarrow \cdots$, $B \rightarrow \cdots$, $C \rightarrow \cdots$. 
Lemma 3.3 (Quiescent upright triangle). From a starting configuration of length \( l \), every configuration at time \( 0 \leq t < l - 1 \) is a pattern of the form

\[ [1, t+1) [t+2, l] \rightarrow \cdots \]

Proof. Obvious from the definition of the quiescent state. \( \square \)

Lemma 3.4 (Starting). If \( 4 \leq l \) then from the starting configuration of length \( l \) we have

\[ 0 : A \rightarrow 4 : CDDB \rightarrow \cdots \]

Proof. It is easy to check that it is true for \( l = 4 \) and \( l = 5 \) (see lemma 3.2 and figures 4(d-e)). The general case (\( l \geq 5 \)) is then a corollary of lemma 3.3. \( \square \)

Lemma 3.5 (Upright triangle). If \( 3 \leq l \) then

\[ t : \cdots DB \rightarrow t + 1 : \cdots DB \]

Proof. The following transition rules are sufficient: \( DB \rightarrow D, B \rightarrow B, \) and at boundaries \( (i + 3 = l) \): \( B \rightarrow B \). \( \square \)
Lemma 3.6 (Forward propagation). Configurations at time $t$ when $5 \leq t \leq l$, are of the kind

$$t : \cdots \bullet A \cdots \bullet B \cdots \bullet \quad \text{if } t = 5 + 3k$$

(1)

$$t : \cdots \bullet \bullet \bullet \bullet \bullet \bullet \bullet \quad \text{if } t = 6 + 3k$$

(2)

$$t : \cdots \bullet \bullet \bullet \bullet \bullet \bullet \bullet \quad \text{if } t = 7 + 3k$$

(3)

Proof. We will prove the correctness by recursion.

Initial case $k = 0$, ($t = 5 + 3k$): from lemmas 3.4 and 3.5 and with the use of the rules $CD \rightarrow C$, $CDD \rightarrow A$, and $DDB \rightarrow \bullet$, we have

$$5 : \bullet \bullet \bullet \bullet \bullet \bullet \bullet$$

Induction step. We will now prove that if $l \geq t + 1$, then Eq. (1) at time $t$ yields Eq. (2) at $t + 1$, Eq. (2) at time $t$ yields Eq. (3) at $t + 1$ and Eq. (3) at time $t$ yields Eq. (1) at $t + 1$.

Suppose that it is true for any $0 \leq k$. If $6 \leq l$ then with the help of lemma 3.5 and the rules $\bullet \bullet C \rightarrow \bullet$, $\bullet CA \rightarrow D$, $CA \bullet \rightarrow A$, $A \bullet \bullet \rightarrow C$ (and for the boundary cases $i.e.$ when $k = 0$: $\bullet CA \rightarrow D$, $A \bullet D \rightarrow C$ and when $k = 1$: $\bullet C \rightarrow \bullet$), we get

$$t + 1 : \bullet \bullet \bullet \bullet \bullet \bullet \bullet$$

If $7 \leq l$, then with the rules $\bullet \bullet D \rightarrow \bullet$, $\bullet DA \rightarrow \bullet$, $DAC \rightarrow C$, $AC \bullet \rightarrow D$, $C \bullet \bullet \rightarrow \bullet$ (and for boundary cases $k = 0$: $\bullet DA \rightarrow \bullet$, $C \bullet D \rightarrow \bullet$, and $k = 1$: $\bullet D \rightarrow \bullet$) we get

$$t + 2 : \bullet \bullet \bullet \bullet \bullet \bullet \bullet$$

And if $8 \leq l$, then with $\bullet CD \rightarrow C$, $CD \bullet \rightarrow A$, $D \bullet \bullet \rightarrow \bullet$ (and for boundary cases with $k = 0$: $\bullet C \rightarrow \bullet$, and if $k = 1$: $\bullet \bullet \rightarrow \bullet$) we get

$$t + 3 : \bullet \bullet \bullet \bullet \bullet \bullet \bullet$$

which is exactly the property for $k + 1$. \qed

As a consequence we have the following lemmas:
Lemma 3.7 (End of forward propagation). The configuration at time $5 \leq t = l$ is

\[ l : \bullet \ldots \bullet DAC \bullet \ldots \bullet DB, \text{ if } l = 0 \mod 3 \]
\[ l : \bullet \ldots \bullet CD \bullet \ldots \bullet DB, \text{ if } l = 1 \mod 3 \]
\[ l : \bullet \ldots \bullet CA \bullet \ldots \bullet DB, \text{ if } l = 2 \mod 3 \]

Proof. This is a direct application of lemma 3.6. \qed

Lemma 3.8 (Bouncing). The configuration at time $l + 1$ when $5 \leq l$ is

\[ l + 1 : \bullet \ldots \bullet CD \bullet \ldots \bullet B, \text{ if } l = 0 \mod 3 \]
\[ l + 1 : \bullet \ldots \bullet CA \bullet \ldots \bullet B, \text{ if } l = 1 \mod 3 \]
\[ l + 1 : \bullet \ldots \bullet DAC \bullet \ldots \bullet B, \text{ if } l = 2 \mod 3 \]

Proof. With the help of the rule DB$\rightarrow$B and the preceding lemma. \qed

Lemma 3.9 (Backward propagation). The configurations at times $l + 2 \leq t < l + \frac{1}{2}$ when $6 \leq l$ are of the kind

\[ l + 2 + 3k : \bullet \ldots \bullet CA \bullet \ldots \bullet BD \bullet \ldots \bullet , \text{ if } l = 0 \mod 3 \]
\[ l + 2 + 3k : \bullet \ldots \bullet DAC \bullet \ldots \bullet BD \bullet \ldots \bullet , \text{ if } l = 1 \mod 3 \]
\[ l + 2 + 3k : \bullet \ldots \bullet CD \bullet \ldots \bullet BD \bullet \ldots \bullet , \text{ if } l = 2 \mod 3 \]

Proof. From lemma 3.8 and with the rules $\bullet \bullet B \rightarrow B$ and $\bullet \bullet \rightarrow D$. Note that only the maximum speed signal moves backwards, so the $CA \rightarrow DAC \rightarrow CD$ transformation is as in lemma 3.6. \qed

We define the end of the backward propagation process as the time when the quiescent middle segment is of length 1.
Lemma 3.10 (End of backward propagation). If $6 \leq l$ then we have

$$
\begin{cases}
  l + \frac{t}{2} - 1 : \cdots \bullet CA \bullet BD \cdots & \text{if } l = 0 \mod 2 \\
  l + \frac{t}{2} - 1 : \cdots \bullet DAC \bullet BD \cdots & \text{if } l = 1 \mod 2
\end{cases}
$$

Proof. From lemma 3.9 we have:

**Case $l = 0 \mod 3$.**

We first show that the length $\Delta$ of the quiescent middle segment (when surrounded by CA and BD) is either $4k + 1$ or $4k + 3$. As $l = 3n$, we have

$$
\Delta = (l - 2 - 3k) - \left(\frac{l + 6}{3} + k\right) + 1 = 3n - 2 - 3k - n - 2 - k + 1 = 2n - 4k - 4 + 1
$$

So

$$
\begin{cases}
  \text{if } n = 2m, & \Delta = 4m - 4k - 4 + 1 = 4(m - k - 1) + 1 \\
  \text{if } n = 2m + 1, & \Delta = 4m + 2 - 4k - 4 + 1 = 4(m - k - 1) + 3
\end{cases}
$$

(4)

Subcase $l = 0 \mod 6$: i.e. $l = 3n = 6m$. Observe that for $k = m - 1$ we have $t = l + 2 + 3k = l + 2 + 3\left(\frac{1}{6} - 1\right) = l + \frac{t}{2} - 1$ and $\Delta = 1$:

$$
l + \frac{t}{2} - 1 : \cdots \bullet CA \bullet BD \cdots
$$

Subcase $l = 3 \mod 6$: i.e. $l = 3n = 3(2m + 1) = 6m + 3$. Observe that for $k = \frac{1}{3} - 1$ we have $t = (l + 1) + 2 + 3k = l + 2 + \frac{1}{3} - 3 = l + \frac{t}{2} - 2$ and $\Delta = 3$ then

$$
l + \frac{t}{2} - 2 : \cdots \bullet CA \bullet BD \cdots
$$

Case $l = 1 \mod 3$.

As in the previous case, we can see that the length of the middle segment (when surrounded by DAC and BD) is either $4p + 1$ or $4p + 3$. As $l = 3n + 1$ we have

$$
\Delta = l - 2 - 3k - \frac{l + 8}{3} - k + 1 = 2n - 4k - 4 + 1
$$

and then the property of equation 4.

Subcase $l = 1 \mod 6$: i.e. $l = 3n + 1 = 6m + 1$. Observe that for $k = m - 1$ we have $t = l + 2 + 3k = l + 2 + \frac{1}{2} - 3 = l + \frac{t}{2} - 1$ and $\Delta = 4m - 4(m - 1) - 4 + 1 = 1$

then

$$
l + \frac{t}{2} - 1 : \cdots \bullet DAC \bullet BD \cdots
$$
Subcase $l = 4 \mod 6$: i.e. $l = 3n + 1 = 6m + 4$. Observe that for $k = m - 1$ we have $t = l + 2 + 3k = l + 2 + \frac{l+1}{2} - 3 = l + \frac{l}{2} - 3$ and $\Delta = 4m + 2 - 4m + 4 - 1 + 1 = 3$ then

\[
\begin{align*}
  l + \frac{l}{2} - 3 & : \quad \bullet \ldots \bullet DAC \bullet \bullet \bullet BD \bullet \ldots \bullet \\
  l + \frac{l}{2} - 2 & : \quad \bullet \ldots \bullet CD \bullet \bullet \bullet BD \bullet \ldots \bullet \\
  l + \frac{l}{2} - 1 & : \quad \bullet \ldots \bullet CA \bullet \bullet \bullet BD \bullet \ldots \bullet 
\end{align*}
\]

Case $l = 2 \mod 3$.
We first see that the length of the middle segment (when surrounded by CD and BD) is either 4\(p\) or 4\(p\) + 2. As $l = 3n + 2$ we have

$$\Delta = l - 2 - 3k - \frac{l+7}{3} - k + 1 = 3n + 2 - 2 - 3k - n - 3 - k + 1 = 2n - 4k - 2$$

So

\[
\begin{align*}
  \text{if } n = 2m, & \quad \Delta = 4m - 4k - 2 = 4(m - k) - 2 \\
  \text{if } n = 2m + 1, & \quad \Delta = 4m + 2 - 4k - 2 = 4(m - k)
\end{align*}
\]

Subcase $l = 2 \mod 6$: i.e. $l = 3n + 2 = 6m + 2$. Observe that for $k = m - 1$, we have $t = l + 2 + 3k = l + 2 + 3\left(\frac{m-2}{6} - 1\right) = l + \frac{l}{2} - 2$ and $\Delta = 4(m - k) - 2 = 4(m - m + 1) - 2 = 2$ then

\[
\begin{align*}
  l + \frac{l}{2} - 2 & : \quad \bullet \ldots \bullet CD \bullet \bullet \bullet BD \bullet \ldots \bullet \\
  l + \frac{l}{2} - 1 & : \quad \bullet \ldots \bullet CA \bullet \bullet \bullet BD \bullet \ldots \bullet 
\end{align*}
\]

Subcase $l = 5 \mod 6$: i.e. $l = 3n + 2 = 6m + 5$. Observe that for $k = m - 1$ we have $t = l + 2 + 3k = l + 2 + 3\left(\frac{m-5}{6} - 1\right) = l + \frac{l+1}{2} - 3$ and $\Delta = 4(m - k) = 4$ then

\[
\begin{align*}
  l + \frac{l+1}{2} - 3 & : \quad \bullet \ldots \bullet CD \bullet \bullet \bullet BD \bullet \ldots \bullet \\
  l + \frac{l+1}{2} - 2 & : \quad \bullet \ldots \bullet CA \bullet \bullet \bullet BD \bullet \ldots \bullet \\
  l + \frac{l+1}{2} - 1 & : \quad \bullet \ldots \bullet DAC \bullet \bullet \bullet BD \bullet \ldots \bullet 
\end{align*}
\]

$\Box$

Lemma 3.11 (Cuts). If $6 \leq l$ then

\[
\begin{align*}
  l + \frac{l}{2} & : \quad \bullet \ldots \bullet DABD \bullet \ldots \bullet \quad \text{if } l = 0 \mod 2 \\
  l + \frac{l+1}{2} - 1 & : \quad \bullet \ldots \bullet CDAD \bullet \ldots \bullet \quad \text{if } l = 1 \mod 2
\end{align*}
\]
Proof. Obvious from lemma 3.10 and the rules $A\bullet B \to B$ and $C\bullet B \to A$. □

Lemma 3.12 (Recursive start). If $6 \leq l$ then

\[
\begin{align*}
&l + \frac{l}{2} + 1 : \begin{cases} 
[1, \frac{l-1}{2}] \\
[\frac{l}{2}, l]
\end{cases} AA \ldots \bullet 
\end{align*}
\]
if $l = 0 \mod 2$

\[
\begin{align*}
&l + \frac{l-1}{2} + 1 : \begin{cases} 
[1, \frac{l-3}{2}] \\
[\frac{l-1}{2}, l-1]
\end{cases} \bullet CBC \ldots \bullet 
\end{align*}
\]
if $l = 1 \mod 2$

and then

\[
\begin{align*}
&l + \frac{l}{2} + 2 : \begin{cases} 
[1, \frac{l-2}{2}] \\
[\frac{l}{2}+1, l]
\end{cases} CBB \ldots \bullet 
\end{align*}
\]
if $l = 0 \mod 2$

\[
\begin{align*}
&l + \frac{l-1}{2} + 2 : \begin{cases} 
[1, \frac{l-3}{2}] \\
[\frac{l-1}{2}+1, l]
\end{cases} \bullet CBC \ldots \bullet 
\end{align*}
\]
if $l = 1 \mod 2$

Proof. From lemma 3.11 and with the help of the rules DAB $\to A$, ABD $\to A$, CDA $\to B$, DAD $\to C$, and $\bullet \bullet \to C$, $\bullet A \to C$, $A \bullet \bullet \to C$. (And for boundary case $l = 6$, $\bullet A \to C$ and $A \bullet \bullet \to C$). □

Lemma 3.13 (Virtual borders). If $6 \leq l$ and $l = 0 \mod 2$ (respectively when $l = 1 \mod 2$) then each segment $[1, \frac{l}{2}]$ and $[\frac{l}{2}+1, l]$ (resp. $[1, \frac{l-1}{2}]$ and $[\frac{l-1}{2}+1, l]$) evolves independently and symmetrically.

Proof. One can easily check that the transition function is symmetric, i.e. for every states $s, s', s''$, if $ss's'' \to x$ is a transition then $s''s's \to x$ also. Then, obviously if a configuration is symmetric, its evolution is symmetric too. The recursive starting configurations (lemma 3.12) being symmetric, so is their evolution.

Finally, observing that after any cut, the middle of the configurations is either of the form $ss's's$ (for even length) or $ss's$ (for odd length), independence of the configurations of the two sub-lines is obtained with the following properties of the transition rules (and their symmetric counterpart):

if $ss's\to x$ is a transition, then so is $ss's\to x$
if $ss's'\to x$ is a transition, then so is $ss's'\to x$

which insures that any such state $s$ plays the role of a border. □

Theorem 3.14 (FSSP). The transition function synchronizes every line of $l > 1$ cells in time $T_s(l) = 3(l - 1) + \lceil \log_2(l) \rceil$.

Proof. We already know that $T_s(2) = 4$ and from preceding lemmas we have:

<table>
<thead>
<tr>
<th>Length</th>
<th>Cut at time</th>
<th>Subline length</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2p$</td>
<td>$3p + 1$</td>
<td>$p$</td>
</tr>
<tr>
<td>$2p + 1$</td>
<td>$3p + 1$</td>
<td>$p + 1$</td>
</tr>
</tbody>
</table>
Suppose it is true for \( l \) then we have
\[
T_s(2l) = 3l + 1 + T_s(l) = 3l + 3(l - 1) + 1 + \lceil \log_2(l) \rceil
= 3(2l - 1) + \lceil \log_2(2l) \rceil
\]
and as \( x > 1 \) yields \( \lceil \log_2(2x) \rceil = \lceil \log_2(2x - 1) \rceil \) we also have
\[
T_s(2l - 1) = 3(l - 1) + 1 + T_s(l) = 3(l - 1) + 3(l - 1) + 1 + \lceil \log_2(l) \rceil
= 3[(2l - 1) - 1] + \lceil \log_2(2l) \rceil
\]
\[
= 3[(2l - 1) - 1] + \lceil \log_2(2l - 1) \rceil
\]

\[\square\]

**Theorem 3.15** (Unrestricted FSSP). *The transition function solves the unrestricted problem, i.e. synchronizes every line of \( l > 0 \) cells with the initiator at the left or the right.*

*Proof.* The transition function is symmetric, so computing the transition function on a line as \( t : s_1s_2\ldots s_l \) is the same as computing the function on the line \( t : s_l\ldots s_2s_1 \). The figure 5(b) shows such a run on a line of length 14. \[\square\]

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure5.png}
\caption{Various runs}
\end{figure}

4. **Some more solved problems**

Our transition function also solves some more variants of the problem. It does synchronize lines with two initiators on the line (one at each end), and it also synchronizes rings of cells as stated in the following theorems.
Theorem 4.1 (left-right-FSSP). The transition function synchronizes every line of \( l > 0 \) cells in the configuration \( A \bullet \ldots \bullet A \).

**Proof.** From lemma 3.13 and theorem 3.15. The figure 5(c) shows a sample of a such a computation on a line of length 28. \( \Box \)

Theorem 4.2 (Ring). The automata synchronizes every ring of perimeter \( l > 0 \).

**Proof.** It is sufficient to remark that taking a linear configuration with initiator at both ends and sticking the first cell on the last gives a ring. Then synchronizing a ring of perimeter \( l > 0 \) is the same as synchronizing a line of length \( l + 1 \) with initiators at both ends. \( \Box \)

5. Conclusion

In this paper we have exhibited a 6-state symmetric non time-optimal solution to some firing squad synchronization problems: unrestricted, both initiators and ring variants. This solution has two interesting features. First, it is intrinsically non minimal-time, \textit{i.e.} not built on top of some minimal-time solution, charging down the intuitive argument stating that \textit{minimal-state solution implies minimal-time}. Second, it is symmetric which also contradicts another intuition about minimal-state solutions.

We also gave a complete proof of the correctness of the solution, which was never done since Mazoyer’s solution and Noguchi’s (see [8, 12]). As the nature of their solutions is very different, that proof is significant, and not only because proofs are \textit{sine qua non} in science, but also because we think that we can extract, from it, some very interesting information about the solution.

I would like to extend my grateful thanks to Professor Serge Grigorieff. I met him during the late years of my student time when he was teaching “Calculability and Models of Computation”. To my astonishment, that great teacher and fount of knowledge, was always well-disposed towards me, and so much far beyond what could be expected in such situation. He introduced me to the firing squad synchronization problem and encouraged me to make a doctorate in the field of Cellular Automata – he was a great director, I felt and indeed was free. Later on, he pushed me to compete for an “Maitre de Conférence” position in the University, and now that I am in a very comfortable position, I had some time to observe what happened during those years and what is due to Serge – absolutely invaluable. Folklore says that “we don’t have more true friends than fingers in one hand”, and according to this I can say that Serge is one of the few person whom I can really reckon with. I didn’t know so much people so peaceful and full of kindness and, I hope that the time which passes will never change anything in our mutual respect and humble admiration. Sincerely.
References


Communicated by (The editor will be set by the publisher).
(The dates will be set by the publisher).