On Formulations of Firing Squad Synchronization Problems

Kojiro Kobayashi[1] and Darin Goldstein[2]

[1] Department of Information Systems Science, Faculty of Engineering, Soka University, 1-236 Tangi-cho, Hachioji-shi, Tokyo 192-8577, Japan
kobayasi@t.soka.ac.jp
[2] Department of Computer Engineering and Computer Science, California State University, Long Beach
daring@cecs.csulb.edu

Abstract. We propose a novel formulation of the firing squad synchronization problem. In this formulation we may use more than one general state and the general state to be used is determined by the boundary condition of the general. We show that the usual formulation and the new formulation yield different minimum firing times for some variations of the problem. Our results suggest that the new formulation is more suited for the general theory of the firing squad synchronization problem.

1 Introduction

The firing squad synchronization problem, or FSSP for short, is the following problem raised by J. Myhill in 1957 ([14]). Consider a finite automata A that has two input terminals, one from the left and the other from the right, and two output terminals, one to the left and the other to the right. The value of each output terminal at time t is the state of A at that time t. The state of A at a time $t+1$ is completely determined by the state and the values of the input terminals of A at time t. The set of the states of A includes at least three different states G, Q, F, called the *general state*, the *quiescent state*, and the *firing state*, respectively. For a number n (≥ 1) let N_n be the one-dimensional array of n nodes $p_1, p_2, \ldots, p_n$. Each node p_i is a copy of the automaton A, and the input terminals and the output terminals of adjacent nodes p_i, p_{i+1} are connected mutually ($1 \leq i < n$). See Figure 1. The values of the input terminal from the left of the leftmost node p_1 and the input terminal from the right of the rightmost node p_n are the special symbol # that indicates that the input terminal is open. The transition function of A must satisfy the following condition: if the state of A is Q and the value of each of the input terminals is either Q or # at a time t, then the state of A at the next time $t+1$ must be Q. We call the leftmost node p_1 the *root* of N_n. At time 0, the state of a node p_i is the general state G if the node is the root ($i = 1$) and the quiescent state Q otherwise ($i \geq 2$). Then, for each t (≥ 0) and i ($1 \leq i \leq n$), the state of the node p_i at time t is uniquely determined. The problem is to design a finite automaton A, a *solution* of FSSP, such that, for any n, all the nodes of N_n enter the firing state F simultaneously for the first time.

C.S. Calude et al. (Eds.): UC 2005, LNCS 3699, pp. 157–168, 2005.

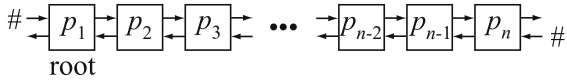

Fig. 1. The original FSSP

We can easily construct a solution having the firing time $3n$ for N_n. Moreover, we can easily show that the firing time of any solution for N_n cannot be smaller than $2n-2$. Hence, if a solution has the firing time $2n-2$ for all N_n, we may call it a *minimal-time solution*. Existence of minimal-time solutions was first shown by Goto ([7]), and later by Waksman ([20]).

After this original FSSP was introduced, many variations of FSSP have been proposed and studied ([13]). Suppose that a variation V of FSSP has a solution. For each problem instance N of V, by the *minimum firing time* of N of the variation V we mean the minimum of the firing times of solutions A of V for N, where A ranges over all solutions of V. If the firing time of a solution $\tilde{A}$ of V for N is the minimum firing time of N of V for all problem instances N of V, we call $\tilde{A}$ a *minimal-time solution* of the variation V.

In this paper we propose a modification of the formulation of FSSP and study how the modification influences the minimum firing times of various variations of FSSP. The modification is as follows. First, instead of having one unique general state G, we allow a finite automaton to have more that one general state $\mathrm{G}_1, \mathrm{G}_2, \ldots, \mathrm{G}_s$. The general state to be used is uniquely determined by the boundary condition of the root. Here, by the *boundary condition* of a node of a problem instance, we mean the information of which input terminals and output terminals of the node are open. Second, instead of having one unique firing state F, we specify a set $\mathcal{F}$ of states as the set of firing states. For a finite automaton to be a solution, all nodes of the network must enter some firing state simultaneously for the first time. Different nodes may enter different firing states. A general state G_i may be also a firing state.

This modification implies the following for designing solutions of FSSP. First, the root can send its boundary condition to adjacent nodes at time 0. Hence a node adjacent to the root can use the boundary condition of the root in determining its state at time 1. Second, the general state that should be used when the boundary condition of the root is "all terminals are open" may be a firing state. Hence, if the problem instance has only one node, the root can fire at time 0 and hence the firing time can be be 0.

We call the usual formulation and the new formulation of FSSP the *traditional model* and the *boundary sensitive model*, respectively. There are two motivations for using the boundary sensitive model.

The main motivation is that the boundary sensitive model simplifies the analysis of minimum firing time and allows us to understand the essential structures of minimal-time solutions. We explain why the boundary sensitive model is more suitable through examples.

Another reason concerns the recent change of the motivation for studying FSSP. Recently, the FSSP for directed networks has been utilized as one of the basic protocols for designing network algorithms (for example, [6]). In such applications, a node is a circuit or a computer in a network, and connections between nodes are network connections. In this case, the time for a node to check its boundary condition is negligibly smaller than the time for information exchange between nodes. Hence it is natural to assume that the root (the initiator of the protocol) can send its boundary condition at time 0. The "firing" of the network generally means that the network simultaneously takes some action. If the network has only one node the root can know this at time 0 and start the action promptly. Hence it is natural to assume that if the network has only one node the firing time is 0.

For a variation V of FSSP and a problem instance N of V, let $\mathrm{mft}_{V,\mathrm{tr}}(N)$ and $\mathrm{mft}_{V,\mathrm{bs}}(N)$ denote the minimum firing times of V for N of the traditional model and the boundary sensitive model, respectively. We are interested in the relation between $\mathrm{mft}_{V,\mathrm{tr}}(N)$ and $\mathrm{mft}_{V,\mathrm{bs}}(N)$. We always have $\mathrm{mft}_{V,\mathrm{tr}}(N) > \mathrm{mft}_{V,\mathrm{bs}}(N)$ for the problem instance N that has only one node because the first value is at least 1 and the second value is 0. Hence, in the remainder of the paper we consider only problem instances that have at least two nodes.

The main technical results of the paper are twofold. First we show that for many variations the two models give the same minimum firing time. Second we show that for some variations the two models give different minimum firing times and in the traditional model the determination of the minimum firing time is unnecessarily complicated due to unnaturalness of the model.

We should mention that the formulation of FSSP that uses more than one firing state has been used by Imai and Morita ([8]) to study FSSP by reversible cellular automata.

2 FSSP That Have Known Minimal-Time Solutions

In this section we consider variations of FSSP for which we know minimal-time solutions. The following lists some of these variations:

- The original FSSP of the one-dimensional line of length n (Fig. 1): The minimum firing time is $2n-2$ ([7], [20], [1]);
- The one-dimensional line of length n such that the root may be at any position: The minimum firing time is $2n-2-\min\{p-1, n-p\}$, where p is the position of the root ($1 \le p \le n$) ([15]);
- The one-dimensional line of length n with k roots such that the roots may be at any positions: The minimum firing time is $2n-2-\min\{\max_i(p_i-1), \max_i(n-p_i)\}$, where p_i is the position of the ith root ($1 \le p_i \le n$, $1 \le i \le k$) ([18]);
- The square of size $n \times n$: The minimum firing time is $2n-2$ ([19]);
- The rectangle of size $m \times n$: The minimum firing time is $m+n+\max\{m,n\}-3$ ([19]);
- The cube of size $n \times n \times n$: The minimum firing time is $3n-3$ ([19]);

- The ring of size n: The minimum firing time is n ([3], [2]);
- The ring of size n with one-way information flow: The minimum firing time is $2n - 1$ ([10], [12]).

For all of these variations V we can show $\mathrm{mft}_{V,\mathrm{tr}}(N) = \mathrm{mft}_{V,\mathrm{bs}}(N)$ for any N. The proofs are essentially the same. As an example, suppose that V is the original FSSP. For this V there exists a solution for the traditional model that shows $\mathrm{mft}_{V,\mathrm{tr}}(N_n) \leq 2n - 2$. Moreover, we can prove that the firing time of any solution A of V for N_n cannot be smaller than $2n-2$ by formalizing the intuitive reasoning that it takes at least $2n - 2$ time for the root to know the position of the rightmost node. But this proof is also true for the boundary sensitive model. Hence we have $2n - 2 \leq \mathrm{mft}_{V,\mathrm{bs}}(N_n) \leq \mathrm{mft}_{V,\mathrm{tr}}(N_n) \leq 2n - 2$.

3 FSSP of General Networks

Next we consider variations of FSSP for general networks. Of these variations two are the most basic. One is the FSSP of directed networks and the other is the FSSP of bilateral networks. We abbreviate these two variations to DN and BN respectively.

In DN, an automaton A has a input terminals and b output terminals, where a, b (≥ 1) are implicit parameters. A problem instance N of *DN* is a network that is obtained from copies of A by connecting some of the outputs to some of the inputs. Each output of a node is either open or is connected to a single input of another node, and hence the "fan-out" is at most one. Each automaton A knows whether its jth output is open or not for each j ($1 \leq j \leq b$). One node is specified as the root. Moreover, the network N must be strongly connected, that is, there must be a directed path of connections from v to v' for each pair (v, v') of nodes.

A network N of DN is a *bilateral* network if $a = b$ and the following condition is satisfied: if the ith output of a node v is connected to the jth input of a node v' then the jth output of v' is connected to the ith input of v. BN is the the variation such that the problem instances are all bilateral networks. Note that all the variations mentioned in Section 2 are subproblems of BN except the last one, the FSSP of rings with one-way information flow.

For both DN and BN we do not know minimal-time solutions. The best known solution of BN is by Nishitani and Honda ([16]) and its firing time is $3r - 1$, where r is the radius of the network. A solution of DN was first found by Kobayashi ([9]). Its firing time was an exponential function of the number n of nodes. The firing time has been improved to $O(n^2)$ by Even, Litman and Winkler ([4]) and then to $O(nd)$ by Ostrovsky and Wilkerson ([17]), where d is the diameter of the network.

Claim 1. *For both of* BN *and* DN *the two formulations have the same minimum firing time.*

Proof. Let $N = (V, E)$ be a directed network or a bilateral network, where V is the set of nodes and E is the set of connections. For $v, v' \in V$ and $e \in E$, let

$d(v, v')$ and $d_e(v, v')$ respectively denote the length of a shortest path from v to v' (or between v and v' if N is bilateral) and the length of a shortest path from v to v' (or between v and v' if N is bilateral) that passes through e, respectively. Moreover let $f(N)$ denote the value $\max_{e\in E, v\in V} d_e(v_{\mathrm{g}}, v)$, where v_{g} denotes the root. Then we have

$$\mathrm{mft}_{\mathrm{DN,tr}}(N) = \mathrm{mft}_{\mathrm{DN,bs}}(N) = f(N),$$
$$\mathrm{mft}_{\mathrm{BN,tr}}(N) = \mathrm{mft}_{\mathrm{BN,bs}}(N) = f(N).$$

We will very briefly explain the idea for proving these characterizations of minimum firing time only for DN.

First we show $\mathrm{mft}_{\mathrm{DN,bs}} \geq f(N)$ using the network N shown in Fig. 2 as an example. For this network N we have $f(N) = 6$ and the e, v that realize this maximum value 6 is $e = (p_3, 1, 1, p_4), v = p_3$, where the symbol (v, i, j, v') denotes the connection from the ith output of v to the jth input of v'. Let N' be the network shown in Fig. 2 and let $\tilde{t}$ be any time such that $\tilde{t} \leq 5$. Then, at time $\tilde{t}$, the states of p_3 in N and p_3 in N' are the same and the the state of p_9 in N' is the quiescent state Q. Hence, if a solution A of DN of the boundary sensitive model fires at $\tilde{t}$ on N, at that time the states of nodes in N' contain both of a firing state and Q. This contradicts our assumption that A is a solution of DN. Hence the firing time of A cannot be $\tilde{t}$. This proves $\mathrm{mft}_{\mathrm{DN,bs}}(N) \geq 6 = f(N)$.

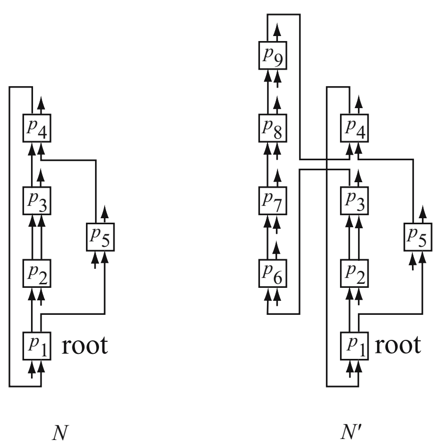

Fig. 2. FSSP of directed networks

Next we show $\mathrm{mft}_{\mathrm{DN,bs}}(N) \leq f(N)$. We select one directed network N and fix it. We show how to construct a solution A_{bs} of DN of the boundary sensitive model whose firing time for N is at most $f(N)$. The structure of A_{bs} essentially depends on the fixed network N.

A_{bs} simulates two finite automata $A_{1,\mathrm{bs}}$, $A_{2,\mathrm{bs}}$ of the boundary sensitive model and fires when at least one of them fires. $A_{1,\mathrm{bs}}$ may be any solution of

DN. $A_{2,\text{bs}}$ is a finite automaton such that all the nodes collaborate to check that the given network is N. If the given network is N then all the nodes know it before or at $f(N)$, and fire at $f(N)$. Otherwise each node never fires. Hence A_{bs} is a solution.

The details of the behavior of $A_{2,\text{bs}}$ are as follows. For each node $v \in V$ we fix one shortest path from v_{g} to v and use that path to uniquely specify v. For example, if we select the path $(p_1, 1, 1, p_2)$, $(p_2, 2, 2, p_3)$ for p_3 in the network N shown in Fig. 2, all the nodes refer to p_3 of N as "the node that is arrived at when we proceed from v_{g} along connections $(p_1, 1, 1, p_2)$, $(p_2, 2, 2, p_3)$."

For each pair (v', v) of nodes of N, $A_{2,\text{bs}}$ uses a signal to teach v that v' really exists in the network, and also the boundary condition of v'. The time needed for this is $d(v_{\text{g}}, v') + d(v', v)$ because we are using the boundary sensitive model. Moreover, for each pair (e, v) of $e \in E$ and $v \in V$, $A_{2,\text{bs}}$ uses a signal to teach v that e really exists in the network. The time needed for this is $d(v_{\text{g}}, v') + 1 + d(v'', v) = d_e(v_{\text{g}}, v)$, where v', v'' are nodes such that e is from v' to v''.

Hence, if the given network is N, using these signals all the nodes know this before or at time

$$\max\{\max_{v', v \in V}(d(v_{\text{g}}, v') + d(v', v)), \max_{e \in E, v \in V} d_e(v_{\text{g}}, v)\} = \max_{e \in E, v \in V} d_e(v_{\text{g}}, v) = f(N)$$

and $A_{2,\text{bs}}$ at any node can fire at time $f(N)$. If the network is not N, $A_{2,\text{bs}}$ at any node never fires. Hence, the firing time of the solution A_{bs} for N is at most $f(N)$, and hence $\text{mft}_{\text{DN,bs}}(N) \le f(N)$.

The above idea cannot be used directly for the traditional model because in the model the time for v to know the boundary condition of v' is not $d(v_{\text{g}}, v') + d(v', v)$ for $v' = v_{\text{g}}$. This complicates the analysis of $\text{mft}_{\text{DN,tr}}(N)$. However, if we note that $\max_{v \in V} d(v_{\text{g}}, v) + 1 \le f(N)$, we can construct a solution A_{tr} of DN of the traditional model whose firing time for N is at most $f(N)$.

A_{tr} is obtained from A_{bs} by modifying its components $A_{1,\text{bs}}$ and $A_{2,\text{bs}}$ to automata $A_{1,\text{tr}}$ and $A_{2,\text{tr}}$ of the traditional model as follows. $A_{1,\text{tr}}$ may be any solution of DN of the traditional model. There are 2^{a+b} boundary conditions. $A_{2,\text{tr}}$ simulates the behaviors of $A_{2,\text{bs}}$ for all of these boundary conditions simultaneously. At the same time, at time 1 the root broadcasts the correct boundary condition to all nodes. A node fires when it has received the correct boundary condition and the simulated $A_{2,\text{bs}}$ for that boundary condition fires. The condition $\max_{v \in V} d(v_{\text{g}}, v) + 1 \le f(N)$ guarantees that each node knows the correct boundary condition before or at $f(N)$. Hence $A_{2,\text{tr}}$ fires at $f(N)$ if the given network is N.

Hence the firing time of A_{tr} for N is at most $f(N)$, and hence $\text{mft}_{\text{DN,tr}}(N) \le f(N)$. This, together with $f(N) \le \text{mft}_{\text{DN,bs}}(N) \le \text{mft}_{\text{DN,tr}}(N)$, shows $\text{mft}_{\text{DN,tr}}(N) = f(N)$.

The proof of $\text{mft}_{\text{BN,tr}}(N) = \text{mft}_{\text{BN,bs}}(N) = f(N)$ is similar. □

4 FSSP of Paths and Regions in $\mathbb{Z}^2$ and $\mathbb{Z}^3$

The final variations we consider are paths and regions in the two-dimensional grid space $\mathbb{Z}^2$ and the three-dimensional grid space $\mathbb{Z}^3$. First we explain the variations for $\mathbb{Z}^2$.

We say that two points $p = (x, y)$, $p' = (x', y')$ in $\mathbb{Z}^2$ are *adjacent* if either $x = x'$ and $|y - y'| = 1$ or $|x - x'| = 1$ and $y = y'$. By a *path* in $\mathbb{Z}^2$, or simply a *path*, we mean a sequence $p_1 p_2 \ldots p_n$ of points in $\mathbb{Z}^2$ $(n \geq 1)$ such that p_i and p_j are adjacent if and only if $|i - j| = 1$ $(1 \leq i \leq n,\ 1 \leq j \leq n)$.

The FSSP of paths in $\mathbb{Z}^2$, or 2PATH for short, is the FSSP such that problem instances are paths $p_1 p_2 \ldots p_n$ in $\mathbb{Z}^2$ and p_1 is the root of each path $p_1 p_2 \ldots p_n$. Another variation, the FSSP of generalized paths in $\mathbb{Z}^2$, or g-2PATH for short, is the FSSP such that problem instances are paths $p_1 p_2 \ldots p_n$ in $\mathbb{Z}^2$ and the root may be at any position. Finally, the FSSP of regions in $\mathbb{Z}^2$, or 2REG for short, is the FSSP such that problem instances are nonempty finite subsets X of $\mathbb{Z}^2$ such that any two points p, p' in X are connected with a path in X, and the root may be any point in X.

We can define similar variations 3PATH, g-3PATH, 3REG for $\mathbb{Z}^3$. In Fig. 3 (a) and (b) we show examples of problem instances of 2PATH and 2REG, respectively.

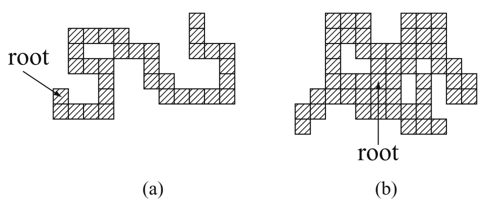

Fig. 3. Examples of 2PATH and 2REG

For each of these three variations, a finite automata A that is used to construct a solution has four inputs and four outputs, each corresponding to the direction of one of the four adjacent positions. Two copies of A at adjacent points p, p' are mutually connected with the corresponding input and output. Hence, all of these variations are subproblems of BN.

At present we know no minimal-time solutions for these six variations. However, in [5] we showed that if P $\neq$ NP then 3PATH, g-3PATH and 3REG have no minimal-time solutions. Hence these three variations are highly unlikely to have minimal-time solutions. (In [5] we showed the result only for 3PATH. However the proof also applies to g-3PATH and 3REG with slight modifications.)

It is evident that $\mathrm{mft}_{\mathrm{2PATH,tr}}(N) = \mathrm{mft}_{\mathrm{2PATH,bs}}(N)$ and $\mathrm{mft}_{\mathrm{3PATH,tr}}(N) = \mathrm{mft}_{\mathrm{3PATH,bs}}(N)$ for any path $N = p_1 p_2 \ldots p_n$ in $\mathbb{Z}^2$ and $\mathbb{Z}^3$. Hence we are interested in how $\mathrm{mft}_{V,\mathrm{tr}}(N)$ and $\mathrm{mft}_{V,\mathrm{bs}}(N)$ are related for $V =$ g-2PATH, 2REG, g-3PATH, 3REG. We consider the problem only for g-2PATH. All the results hold true for g-3PATH without any modification and for 2REG and 3REG with slight modifications.

In [11], a characterization of $\mathrm{mft}_{\mathrm{2PATH,tr}}(p_1p_2\dots p_n)$ was obtained. We elaborate on this result in detail below.

For $1 \le i < n$, let $e(p_1p_2\dots p_n, i)$ denote the length m of a longest extension of $p_1p_2\dots p_i$ of the form $p_1p_2\dots p_ip_{i+1}q_2\dots q_m$. The value $e(p_1p_2\dots p_n, i)$ may be ∞. For $i = n$, we define $e(p_1p_2\dots p_n, n)$ to be 0. Let i_0 be the value defined by $i_0 = \min\{i | 1 \le i \le n, i \ge e(p_1p_2\dots p_n, i)\}$. Let $f(p_1p_2\dots p_n)$ be $2i_0 - 1$ if $i_0 = e(p_1p_2\dots p_n, i_0)$ and $2i_0 - 2$ if $i_0 > e(p_1p_2\dots p_n, i_0)$.

Lemma 1 ([11]).

$$\mathrm{mft}_{\mathrm{2PATH,tr}}(p_1p_2\dots p_n) = \mathrm{mft}_{\mathrm{2PATH,bs}}(p_1p_2\dots p_n) = f(p_1p_2\dots p_n).$$

Proof. Only an outline of the proof is given. As we have already mentioned, we can easily show $\mathrm{mft}_{\mathrm{2PATH,tr}}(p_1p_2\dots p_n) = \mathrm{mft}_{\mathrm{2PATH,bs}}(p_1p_2\dots p_n)$. Hence we only show $\mathrm{mft}_{\mathrm{2PATH,bs}}(p_1p_2\dots p_n) = f(p_1p_2\dots p_n)$. We assume that $i_0 < n$. The proof for the case $i_0 = n$ is simpler.

First we show that the firing time of any solution A for $\alpha = p_1p_2\dots p_n$ cannot be smaller than $f(p_1p_2\dots p_n)$. Let $\tilde{t}$ be a time such that $\tilde{t} < f(p_1p_2\dots p_n)$.

Suppose that $i_0 = e(p_1p_2\dots p_n, i_0)$. Then $\tilde{t} < f(p_1p_2\dots p_n) = 2i_0 - 1$. There is a path of the form $\alpha' = p_1p_2\dots p_{i_0}p_{i_0+1}q_2\dots q_{i_0}$. At time $\tilde{t}$, the state of p_1 in α and the state of p_1 in α' are the same and the state of q_{i_0} in α' is Q. Hence A cannot fire on α at time $\tilde{t}$.

Suppose that $i_0 > e(p_1p_2\dots p_n, i_0)$. Then $0 \le \tilde{t} < f(p_1p_2\dots p_n) = 2i_0 - 2$ and hence $2 \le i_0$. We have $i_0 \le e(p_1p_2\dots p_n, i_0 - 1)$. Hence there is a path of the form $\alpha' = p_1p_2\dots p_{i_0-1}p_{i_0}q_2\dots q_{i_0}$. At time $\tilde{t}$, the state of p_1 in α and the state of p_1 in α' are the same and the state of q_{i_0} in α' is Q. Hence A cannot fire on α at time $\tilde{t}$.

Next we construct a solution A whose firing time for $p_1p_2\dots p_n$ is at most $f(p_1p_2\dots p_n)$. A simulates two finite automata A_1, A_2. The structure of A_2 essentially depends on the path $p_1p_2\dots p_n$. A fires when at least one of A_1, A_2 fires. A_1 may be any solution of 2PATH. A_2 checks that the given path starts with $p_1p_2\dots p_{i_0}p_{i_0+1}$. If the check succeeds, A_2 at any node fires at time $f(p_1p_2\dots p_n)$. If the check fails, A_2 at any node never fires. Hence A is a solution.

The details of the behavior of A_2 is as follows. At time 0, A_2 sends a check signal from the root to the node p_{i_0} along the path $p_1p_2\dots p_{i_0}$. If check succeeds, the check signal knows it at p_{i_0} at time i_0-1 and then the check signal broadcasts the order "fire at time $f(p_1p_2\dots p_n)$" to all the nodes. If $i_0 = e(p_1p_2\dots p_n, i_0)$ all the nodes receive the order before or at time $(i_0 - 1) + \max\{i_0 - 1, i_0\} = 2i_0 - 1 = f(p_1p_2\dots p_n)$. If $i_0 > e(p_1p_2\dots p_n, i_0)$ all the nodes receive the order before or at time $(i_0 - 1) + \max\{i_0 - 1, i_0 - 1\} = 2i_0 - 2 = f(p_1p_2\dots p_n)$. Hence, in any case, if the check succeeds all the nodes receive the order "fire at time $f(p_1p_2\dots p_n)$" before or at time $f(p_1p_2\dots p_n)$ and hence can fire at that time.

Hence the firing time of A for $p_1p_2\dots p_n$ is at most $f(p_1p_2\dots p_n)$. □

In Fig. 4 we show an example of a path. For this path $p_1p_2\dots p_{22}$ we have $e(p_1p_2\dots p_{22}, 18) = \infty$, $e(p_1p_2\dots p_{22}, 19) = 4$, and hence $i_0 = 19$, $19 >$

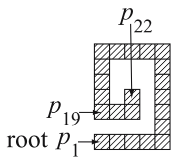

Fig. 4. An example of paths

$e(p_1p_2\ldots p_{22},19)$, $f(p_1p_2\ldots p_{22}) = 2{\cdot}19{-}2 = 36$. Hence $\mathrm{mft}_{\mathrm{2PATH,tr}}(p_1p_2\ldots p_{22}) = \mathrm{mft}_{\mathrm{2PATH,bs}}(p_1p_2\ldots p_{22}) = 36$.

A path $p_1p_2\ldots p_n$ such that p_1 is the root is also a problem instance of g-2PATH. For this problem instance we have the following results.

Theorem 1. $\mathrm{mft}_{\mathrm{g-2PATH,bs}}(p_1p_2\ldots p_n) = f(p_1p_2\ldots p_n)$.

Proof. In the proof of Lemma 1 the check signal of A_2 checked that the given path starts with $p_1p_2\ldots p_{i_0}p_{i_0+1}$. As a solution of g-2PATH, in addition to this the check signal should also check that the root is at the end. However, if we use the boundary sensitive model the check signal can check it without any additional time. Hence we can construct a solution A of g-2PATH of the boundary sensitive model whose firing time for $p_1p_2\ldots p_n$ is at most $f(p_1p_2\ldots p_n)$. □

Theorem 2. *If* $i_0 = e(p_1p_2\ldots p_n, i_0)$ *and there is a path of the form*

$$r_{2i_0+1}\ldots r_3r_2p_1p_2\ldots p_{i_0}p_{i_0+1}q_2\ldots q_{i_0},$$

then

$$\mathrm{mft}_{\mathrm{g-2PATH,tr}}(p_1p_2\ldots p_n) = f(p_1p_2\ldots p_n) + 1.$$

Proof. We have $\mathrm{mft}_{\mathrm{g-2PATH,tr}}(p_1p_2\ldots p_n) \le \mathrm{mft}_{\mathrm{g-2PATH,bs}}(p_1p_2\ldots p_n) + 1 = f(p_1p_2\ldots p_n) + 1$.

Suppose that there is a solution A of g-2PATH of the traditional model whose firing time $\tilde{t}$ for $p_1p_2\ldots p_n$ is at most $f(p_1p_2\ldots p_n) = 2i_0 - 1$.

Suppose that we run A on the three paths $\alpha = p_1p_2\ldots p_n$, $\alpha' = p_1p_2\ldots p_{i_0} p_{i_0+1}q_2\ldots q_{i_0}$, $\alpha'' = r_{2i_0+1}\ldots r_3r_2p_1p_2\ldots p_{i_0}p_{i_0+1}q_2\ldots q_{i_0}$. In all of these paths p_1 is the root. Consider the states of nodes in these three paths at time $\tilde{t}$. All the nodes in α are F because A on α fires at $\tilde{t}$. The state of p_1 in α and the state of p_1 in α' are the same. Hence the state of p_1 in α' is F and hence the state of q_{i_0} in α' is also F. But the state of q_{i_0} in α' and the state of q_{i_0} in α'' are the same because A is a solution of the traditional model. Hence the state of q_{i_0} is also F. But the state of r_{2i_0+1} in α'' is Q. This is a contradiction. Hence we have $\mathrm{mft}_{\mathrm{g-2PATH,tr}}(p_1p_2\ldots p_n) \ge f(p_1p_2\ldots p_n) + 1$. □

In Fig. 5 we show an example of paths $\alpha_1 = p_1p_2\ldots p_{108}$ that satisfies the condition of Theorem 2. For this path α_1 we have $e(\alpha_1, 106) = \infty$, $e(\alpha_1, 107) =$

107, $i_0 = 107$, and hence $f(\alpha_1) = 2i_0 - 1 = 213$. The equation $e(\alpha_1, 107) = 107$ was checked by the exhaustive search by computers. The path α_2 shown in Fig. 5 is one of the path of the form $\alpha_2 = p_1 p_2 \dots p_{107} p_{108} q_2 \dots q_{107}$ found by the search. From this α_2 we can easily construct a path of the form $r_{215} r_{214} \dots r_3 r_2 p_1 p_2 \dots p_{107} p_{108} q_2 \dots q_{107}$. Hence, by Theorem 2 we have $\mathrm{mft}_{\mathrm{g-2PATH,tr}}(\alpha_1) = f(\alpha_1) + 1 = 214$ while $\mathrm{mft}_{\mathrm{g-2PATH,bs}}(\alpha_1) = f(\alpha_1) = 213$.

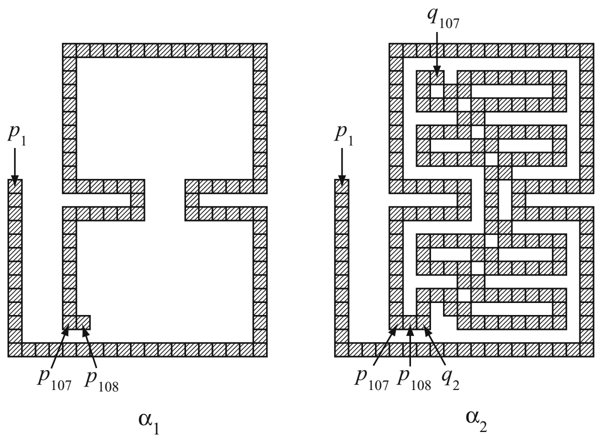

Fig. 5. Two paths α_1, α_2

The proofs of the following two theorems are not difficult and we omit them.

Theorem 3. *If* $i_0 - 1 = e(p_1 p_2 \dots p_n, i_0)$ *and there is a path of the form*

$$r_{2i_0} r_{2i_0-1} \dots r_3 r_2 p_1 p_2 \dots p_{i_0} p_{i_0+1} q_2 \dots q_{i_0-1},$$

then

$$\mathrm{mft}_{\mathrm{g-2PATH,tr}}(p_1 p_2 \dots p_n) = f(p_1 p_2 \dots p_n) + 1.$$

Theorem 4. *If* $i_0 - 2 \geq e(p_1 p_2 \dots p_n, i_0)$ *then*

$$\mathrm{mft}_{\mathrm{g-2PATH,tr}}(p_1 p_2 \dots p_n) = f(p_1 p_2 \dots p_n).$$

From Theorems 2, 3 we are tempted to conjecture that if $i_0 - 1 \leq e(p_1 p_2 \dots p_n, i_0)$ then $\mathrm{mft}_{\mathrm{g-2PATH,tr}}(p_1 p_2 \dots p_n) = f(p_1 p_2 \dots p_n) + 1$. However, this is not true. Suppose that we construct a path α_3 shown in Fig. 6 from the path α_1 shown in Fig. 5 by bending its beginning. For α_3, we have $i_0 = e(p_1 p_2 \dots p_n, i_0)$ and $\mathrm{mft}_{\mathrm{g-2PATH,tr}}(p_1 p_2 \dots p_n) = f(p_1 p_2 \dots p_n)$. In α_3, the check signal of the traditional model that starts at p_1 at time 0 knows the boundary condition of p_1 as soon as it arrives at p_{12}, and hence it needs no extra time to check the boundary condition of p_1.

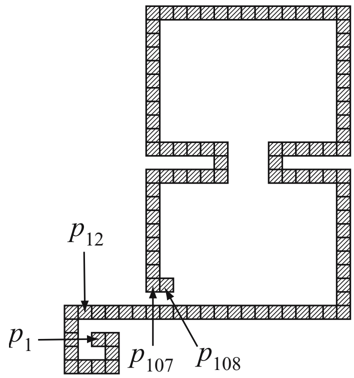

Fig. 6. A modified path α_3

5 Conclusions

The two models give the same minimum firing time for the variations considered in Sections 2, 3. However, they give different minimum firing time for g-2PATH (and 2REG, g-3PATH, 3REG). For these FSSP, the minimum firing time of the boundary sensitive model has a very simple characterization shown in Theorem 1 for paths with the roots at the end. However, Theorems 2, 3, 4 and the phenomenon mentioned in Fig. 6 show that the analysis of the minimum firing time of these FSSP of the traditional model is very complicated and moreover it is due to the unnaturalness of the model.

Hence, if our goal is to construct a general theory of minimum firing time of FSSP, our results suggest that the boundary sensitive model is "the" model to be used.

References

1. R. Balzer. An 8-state minimal time solution to the firing squad synchronization problem. *Information and Control*, 10:22-42, 1967.
2. A. Berthiaume, T. Bittner, L. Perković, A. Settle, and J. Simon. Bounding the firing synchronization problem on a ring. *Theoretical Computer Science*, 320:213-228, 2004.
3. K. Culik. Variations of the firing squad problem and applications. *Information Processing Letters*, 30:153-157, 1989.
4. S. Even, A. Litman, and P. Winkler. Computing with snakes in directed networks of automata. *J. Algorithms*, 24(1):158-170, 1997.
5. D. Goldstein and K. Kobayashi. On the complexity of network synchronization. In *Proc. of 15th International Symposium, ISAAC 2004*, Lecture Notes in Computer Science, vol. 3341, 496-507, Dec. 20-22, 2004.
6. D. Goldstein and N. Meyer. The wake up and report problem is asymptotically time-equivalent to the firing squad synchronization problem. In *Proceedings of the Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 578-587, San Francisco, CA, 6-8 January 2002.

7. E. Goto. A minimal time solution of the firing squad problem. *Course Notes for Applied Mathematics 298, Harvard University*, pages 52-59, 1962.
8. K. Imai and K. Morita. Firing squad synchronization problem in reversible cellular automata. *Theoretical Computer Science*, 165:475-482, 1996.
9. K. Kobayashi. The firing squad synchronization problem for a class of polyautomata networks. *J. Comput. Systems Sci.*, 17(3):300-318, 1978.
10. K. Kobayashi. A minimal time solution to the firing squad synchronization problem of rings with one-way information flow. *Tokyo Institute of Technology, Department of Information Sciences, Research Reports on Information Sciences*, No. C-8, September 1976.
11. K. Kobayashi. On time optimal solutions of the firing squad synchronization problem for two-dimensional paths. *Theoretical Computer Science*, 259:129-143, 29 May 2001.
12. S. LaTorre, M. Napoli, and M. Parente. Synchronization of one-way connected processors. *Complex Systems*, 10:239-255, 1996.
13. J. Mazoyer. An overview of the firing squad synchronization problem. In *Automata Networks* (C. Choffrut, ed.), Lecture Notes in Computer Science, vol. 316, 82-94, 1986.
14. E. F. Moore. *Sequential Machines, Selected Papers*, Addison Wesley, Reading, MA, 1962.
15. F. R. Moore and G. G. Langdon. A generalized firing squad problem, *Information and Control*, 12:212-220, 1968.
16. Y. Nishitani and N. Honda. The firing squad synchronization problem for graphs. *Theoretical Computer Science*, 14(1):39-61, 1981.
17. R. Ostrovsky and D. Wilkerson. Faster computation on directed networks of automata. In Proceedings of the Fourteenth Annual ACM Symposium on Principles of Distributed Computing, pages 38-46, Ottawa, Ontario, Canada, 2-23 August 1995.
18. H. Schmid and T. Worsch. The firing squad synchronization problem with many generals for one-dimensional CA. 3rd IFIP International Conference on Theoretical Computer Science, 111-124, 2004.
19. I. Shinahr. Two- and three-dimensional firing-squad synchronization problem, *Information and Control*, 24:163-180, 1974.
20. A. Waksman. An optimum solution to the firing squad synchronization problem. *Information and Control*, 9:66-78, 1966.