Bounding the Firing Synchronization Problem on a Ring

André Berthiaume, Todd Bittner, Ljubomir Perković, Amber Settle[a] and Janos Simon[b]

[a]DePaul University, School of Computer Science, Telecommunications and Information Systems, 243 S. Wabash, Chicago, IL, 60604, berthiaume@cti.depaul.edu, tbittner@students.depaul.edu, {lperkovic, asettle}@cti.depaul.edu

[b]University of Chicago, Department of Computer Science, 1100 E. 58th Street, Chicago, IL 60637, simon@cs.uchicago.edu

In this paper we improve the upper and lower bounds on the complexity of solutions to the firing synchronization problem on a ring. In this variant of the firing synchronization problem the goal is to synchronize a ring of identical finite automata. Initially, all automata are in the same state except for one automaton that is designated as the initiator for the synchronization. The goal is to define the set of states and the transition function for the automata so that all machines enter a special fire state for the first time and simultaneously during the final round of the computation. In our work we present two solutions to the ring firing synchronization problem, an 8-state minimal-time solution and a 6-state non-minimal-time solution. Both solutions use fewer states than the previous best-known minimal-time automaton, a 16-state solution due to Culik. We also give the first lower bounds on the number of states needed for solutions to the ring firing synchronization problem. We show that there is no 3-state solution and no 4-state, symmetric, minimal-time solution for the ring.

1. Introduction

In the original firing synchronization problem we consider a one-dimensional array of identical finite automata. Initially all automata are in the same state except for one automaton that is designated as the initiator for the synchronization. The machines operate in lock-step, and the transitions of each automaton depend on the state of the automaton and the states of its two neighbors. The goal is to define the set of states and transition rules for the automata so that all machines enter a special fire state for the first time and simultaneously during the final round of the computation. A great deal of work has been done on the original firing synchronization problem [1,6,10–12,14]. It is interesting to note that the vast majority of this work has focused on finding solutions to the problem and that few lower bounds on the number of states needed for solutions to the original problem are known.

There are many variations of the firing synchronization problem that involve networks of automata other than the one-dimensional array [2,4,5,7–9,12,13]. We consider the problem of synchronizing rings of finite automata. In this problem each automaton has exactly two neighbors and there are no endpoints in the system. The goal is the same as the original problem, namely the synchronization of all automata in the final round of the computation.

Initial work on the ring variant of the firing synchronization problem focused on finding correct solutions to the problem without considering the number of states or even the minimal time required to solve the problem [4,5,8,9]. The main reason for this is that the solutions to the ring were given as an initial step in solving a more general problem, that of synchronizing general, connected graphs. The solution to the ring was not the goal, but a necessary first step.

The first work directly considering the number of states needed to solve ring synchronization was done by Culik. He established the minimal time necessary for synchronization, n steps for a ring with n automata, and produced a minimal-time solution using 16 states. He did so in the context of solving a related synchronization problem in which there are multiple initiators [3].

Since the original interest in the firing squad problem on the ring was in finding solutions, little work has been done on finding state lower bounds for the ring. We are unaware of any results giving state lower bounds for the firing synchronization problem on the ring prior to this work.

In this paper we improve Culik's result for the firing synchronization problem on a ring by giving an 8-state, symmetric, minimal-time solution. Not only does this solution use 8 fewer states than Culik's solution, it is symmetric, meaning that automata do not need to distinguish their left and right neighbors when changing state. Few symmetric solutions for any variant of the firing synchronization problem exist [2,13], yet symmetric solutions are particularly interesting. For one, symmetric solutions eliminate any directional information provided to the automaton, intuitively making the problem more difficult. Symmetric solutions are also more elegant, generally producing automata with simpler and easier to understand transition functions. We also give a 6-state, non-minimal-time solution for the ring, the first known non-minimal-time solution for the ring.

We also establish the first known lower bounds on the number of states needed to solve the firing synchronization problem on a ring. We show that there is no 3-state solution, regardless of the time provided for synchronization. We also show that no 4-state, symmetric, minimal-time solution to the ring exists. The 3-state bound is the first lower bound result for any version of the firing squad problem that makes no assumption about the time needed for synchronization. This provides a stronger result, since it applies to both minimal-time and non-minimal-time solutions. The proofs of these results are also far simpler than those for any known existing state lower bound for any variant of the problem [1,2,10,12].

2. Preliminaries

We now outline the definitions for the ring version of the firing synchronization problem, sketch the previous work done on the problem, and state our results.

2.1. Definitions

One of the oldest variants of the firing synchronization problem is one in which the underlying network is not a one-dimensional array but a ring. As in the original firing squad problem there is a single initiator that may be located anywhere in the ring. The initiator is responsible for beginning the synchronization process. In order to ensure that this is true, the remainder of the automata begin in a quiescent state. The automata change state once during each round based on their current state and their two neighbors' current states. Since the initiator begins the process of synchronization, the problem definition requires that automata in the quiescent state with neighbors in the quiescent state must remain in that state. The problem is to define the set of states and the transition function for the automaton so that all machines fire for the first time and simultaneously in some round $t(n)$. It should be noted that the initial configuration is usually counted as round 0 of any simulation. There are also restrictions on the value of $t(n)$, which are discussed in the next section.

The transition function for each automaton can be given as a set of 4-tuples. The 4-tuple (U,V,W,X) represents the rule that an automaton currently in state V, with left neighbor in state U and right neighbor in state W will enter state X at the next time step. We will denote this by UVW $\rightarrow$ X. By definition, automata solving the firing synchronization problem are deterministic so there is at most one 4-tuple (U,V,W,X) for any triple of states U,V,W. As mentioned above, one triple is required by the definition of the problem, namely ZZZ $\rightarrow$ Z, where Z is the quiescent state.

A *symmetric* automaton is one which has a symmetric transition function, that is, whenever a transition UVW $\rightarrow$ X is defined, the transition WVU $\rightarrow$ X must also be defined. This means that the automata do not distinguish their left and right neighbors. As mentioned before, symmetric solutions are intuitively more difficult to produce since directional information cannot be used.

2.2. Previous work

As we have said, initial work on the synchronization of the ring was done while developing solutions general, connected graphs [4,5,8,9]. The solution for the ring was not the goal, but a necessary first step for the solutions for more general graphs. Note that in none of the work cited do the authors consider the number of states needed to synchronize the ring. The solutions are given at a high level, and transition functions are not provided.

The first work directly considering the number of states needed to solve ring synchronization was done by Culik [3]. In his paper Culik considered a variation of the firing synchronization problem in which there are multiple initiators. He showed that any solution to the problem for the one-dimensional array of length n with two initiators located at the endpoints requires $n-1$ steps to synchronize. The following theorem is a direct corollary of this result.

Theorem 2.1 (Culik) *Any solution to the firing synchronization problem for the ring with n automata requires n time steps to synchronize.*

If we take an array of length $n+1$ with two initiators at the endpoints and consider it as a ring with a single initiator we obtain Theorem 2.1. It should be noted that Culik incorrectly gave a time bound of $n-1$ time steps in his paper, as he neglected to account for the fact that the length of the ring is shorter by one than the equivalent array since the two initiators are merged into one.

We will call any solution that synchronizes a ring of n automata in $t(n) = n$ time steps a *minimal-time solution*. Any solution that requires $t(n) > n$ time steps to synchronize will be called a *non-minimal-time solution*.

In addition to giving a time bound, Culik also described a minimal-time solution to the ring version of the problem. He used a modified version of Waksman's solution [14], producing an automaton that uses 16 states. To our knowledge, no prior work has been done on improving the number of states used by Culik's solution.

2.3. Motivation

In understanding the history of the work on lower bounds for the ring firing synchronization problem, it is instructive to consider state lower bounds for the firing synchronization problem on the array. Despite the large body of work that exists for the firing synchronization problem, very little work has been done on finding lower bounds for the problem. The only known state lower bound for the problem was claimed in 1967 by Balzer [1] and confirmed in 1994 by Sanders [10]. Sanders showed that there is no 4-state minimal-time solution to the restricted original problem, the problem where the initiator must be located at the right endpoint of the array.

The technique used for finding the state lower bound was a modified exhaustive search. Balzer wrote a simulation program that examined all possible 4-state solutions in an attempt to demonstrate that none of the solutions correctly solve the firing synchronization problem. Sanders showed that Balzer's program was incomplete but was able to write a correct implementation of the program to confirm the 4-state lower bound. It is crucial for both programs that only minimal-time solutions are examined. This assumption allows the programs to discard any solution that would require more than the minimal number of rounds for synchronization, making the search feasible.

Disappointingly, Sanders showed that this technique of exhaustive search does not scale. In particular, it cannot be used to determine if there is a 5-state solution to the firing synchronization problem on the array, as the search space grows too quickly. This leaves a gap between the best known solution, a 6-state automaton by Mazoyer [6], and the best known lower bound of 4-states [1,10]. This problem has remained open for nearly 20 years.

In an attempt to tackle this long-standing open problem, we consider the number of states needed to solve the ring variant of the firing squad problem. It is our hope that by gaining insight into state lower bounds for the ring, progress can be made in finding lower bounds on the number of states needed for the original problem. Finding a relationship between the structure of solutions for the ring and solutions for the array may aid us in producing the long-sought lower bound. To the best of our knowledge this avenue of research has not been explored, and no state lower bounds for the ring firing synchronization problem exist prior to our work [2,12].

2.4. Our contributions

We present a 8-state, symmetric, minimal-time solution to the firing synchronization problem on the ring. This solution is adapted from Szwerinski's solution to the original firing synchronization problem [13]. To synchronize an array of length n the ring solution requires time n. It uses 8 fewer states than the best-known existing solution for the ring, and it is the first symmetric solution for the ring.

We also give a 6-state, non-minimal-time solution for ring synchronization. This solution is an extension of Mazoyer's solution to the original problem [6]. It requires $2n - 2$ steps to synchronize an array with n automata. This is the first non-minimal-time solution for the firing synchronization problem on the ring, and it uses 2 fewer states that our minimal-time solution and 10 fewer states than Culik's solution.

We also give the first known state lower bounds for the firing synchronization problem on the ring. Our first state lower bound is given in the following theorem:

Theorem 2.2 *There is no* 3*-state solution to the firing synchronization problem for the ring.*

This is the first known state lower bound for the firing synchronization problem that places no restrictions on the time required to synchronize. Such a result would not have been possible using previously existing techniques, as the exhaustive search programs depend on the fact that solutions requiring more than the minimal time can be discarded.

With two additional conditions, we can extend the theorem to the following:

Theorem 2.3 *There is no 4-state, symmetric, minimal-time solution to the firing synchronization problem for the ring.*

The ring provides an inherently symmetric setting, as the endpoints in the array are frequently the place where asymmetry is introduced in existing solutions to the firing synchronization problem. The simplest solutions to the ring synchronization problem are often symmetric. In fact, we conjecture that if an asymmetric, minimal-time solution exists for the ring, then there is a symmetric, minimal-time solution using the same number of states.

3. The minimal-time solution

Our 8-state, minimal-time solution is adapted from Szwerinski's 8-state, symmetric solution to the firing synchronization problem on the one-dimensional array [13]. The construction of the solution requires the addition of some transitions to the solution, as well as the removal of transitions that are not needed for the ring, but the solution behaves in the same manner as Szwerinski's.

In Szwerinski's solution, the array is repeatedly subdivided into halves as new initiators are placed in the center(s) of each of the intervals. The simulation ends when all automata become initiators and then fire. The synchronization begins when the first initiator sends out a signal, the purpose of which is to produce a second initiator when it reaches the opposite end of the array. What is meant by the term signal is a state that tends to propagate toward neighboring automata and whose purpose it is to carry information from one part of the array to another. When this wake-up signal is reflected back by the new initiator, it intersects with markers created in the wake of the first signal and produces a third initiator (or pair of initiators depending on the parity of the original array) located at the center of the array. The term marker indicates a state that remains stationary until it comes into contact with certain signals and whose purpose it is to indicate significant positions in the array. This division of the array into halves continues

0 :										**G**							
1 :									*A*	**G**	*A*						
2 :								*A*	R	**G**	R	*A*					
3 :							*A*	P	B	**G**	B	P	*A*				
4 :						*A*	R	P	B	**G**	B	P	R	*A*			
5 :					*A*	P	Q	R	B	**G**	B	R	Q	P	*A*		
6 :				*A*	R	P		B	B	**G**	B	B		P	R	*A*	
7 :			*A*	P	Q	R		B	B	**G**	B	B		R	Q	P	*A*
8 :	*A*	*A*	R	P		Q	R	B	B	**G**	B	B	R	Q		P	R
9 :	**G**	**G**	Q	R			B	P	B	**G**	B	P	B			R	Q
10 :	**G**	**G**	*A*	Q	R		B	P	B	**G**	B	P	B		R	Q	*A*
11 :	**G**	**G**	R	*A*	Q	R	B	P	B	**G**	B	P	B	R	Q	*A*	R
12 :	**G**	**G**	B	P	*A*	B	Q	P	B	**G**	B	P	Q	B	*A*	P	B
13 :	**G**	**G**	B	P	R	**G**	Q	R	B	**G**	B	R	Q	**G**	R	P	B
14 :	**G**	**G**	B	R	*A*	**G**	*A*	B	B	**G**	B	B	*A*	**G**	*A*	R	B
15 :	**G**	**G**	B	**G**	R	**G**	R	**G**	B	**G**	B	**G**	R	**G**	R	**G**	B
16 :	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**	**G**
17 :	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F

Table 1
A simulation of the 8-state solution for a ring with $n = 17$. The initiator is located in position 10 and all quiescent states are indicated by blank spaces.

until every other automaton is an initiator. At the next step in the simulation every automaton becomes an initiator, and at the next time step all automata fire. For a more detailed description of Szwerinski's solution, see his paper [13]. Important here is that the initiator state is G, Q and P serve as parity markers, B is a marker used to create new initiators, R is the state used to create and advance the B markers, Z is the quiescent state, and A serves as both the wake-up signal and the state that interacts with automata in state B to produce new initiators.

Because the eight-state solution is symmetric, it can be adapted to the ring in a straightforward manner. Instead of a single wake-up signal, two signals are sent from the initiator. These intersect on the opposite side of the ring, creating either one or two new initiators depending on the parity of size of the ring. The process then continues as described above until every automaton becomes an initiator and fires.

To understand the process in more detail, consider a simulation of the 8-state solution for a ring with length $n = 17$ where the initiator is located in position 10. A simulation for such a ring is given in Table 1. Note that synchronization occurs at time step 17, that the time step is listed in the leftmost column, and that the automaton in the first position is assumed to be adjacent to the one in the 17th position. Also, all automaton in the quiescent state are indicated by blank spaces for readability.

In the following description, all time steps refer to the simulation in Table 1. The first initiator in state G sends out two A-signals to the other side of the ring. These signals serve as a wake-up for the remaining automata. In the simulation the A-signals are produced at time step 1. Each A-signal moves at a rate of one automaton per time step. As an A-signal advances away from an automaton, it leaves the automaton in one of two states, either R or P. An R is produced at all even time steps and a P at all odd time steps. Thus the parity of the ring segment the A-signal crosses can be determined by the state appearing behind the A-signal.

The R-signals produced by automata in state A move back in the direction from which the A-signal came at the rate of one automaton per time step. The first R-signals are produced at time step 2 of the sample simulation. When an R-signal collides with an initiator it produces a B-marker. This occurs in the simulation on both sides of the initiator at time step 3. The new B-marker moves away from the initiator one position each time it encounters a new R-signal. For example, the first B-marker advances at time step 6 of the simulation.

The first new initiator is produced when the A-signals meet opposite the original initiator. In the sample simulation this occurs at time step 8. There are two possible outcomes for this collision. If the A-signals are separated by one automaton when they reach the opposite side of the ring, the ring has odd length and can be split into two equal pieces. In this case, a single new initiator opposite the original will be produced. If, on the other hand, the A-signals directly meet, then this means that the ring has odd length and cannot be split into two equal pieces. In this case, two initiators will be produced opposite the original initiator. Since $n = 17$ in the simulation, the latter case occurs at time step 9.

Additional initiators are produced by the interaction of A-signals and B-markers. When an A-signal reaches a B-marker, it produces a new initiator. A single new initiator is produced if the state behind the A-signal is a P since this indicates that the segment traversed by the A-signal is odd and can be split into two pieces. Two new initiators are produced if the automaton behind the A-signal is in state R since in this case the segment is even and there are two central positions. The former case is true in the simulation and the production of the third set of initiators can be seen at time step 13.

This process of recursively dividing the ring continues until the point at which every other automaton is an initiator. This occurs in the simulation at time step 15. At the next time step every automaton becomes an initiator and at the next time step firing occurs.

To produce the above solution for the ring, two types of changes to Szwerinski's 8-state solution to the original problem had to be made. First, all unnecessary transitions were eliminated. A transition is unnecessary if it involves a triple that does not appear in any simulation. Clearly, any transition involving the end marker is unnecessary, as the end marker is used in solutions to the original problem to indicate the end of the array. The marker allows the definition of a single transition function instead of three different types of transition functions, one for the central automata and one for each of the left and right end machines. Since there are no endpoints in the ring, these transitions can be removed. In addition, the transitions ARA $\rightarrow$ Q, PRQ $\rightarrow$ Q, QRP $\rightarrow$ Q, and QRQ $\rightarrow$ Q were eliminated. Each corresponds to a configuration produced only for arrays.

Next, additional transitions had to be defined for configurations that appeared in simulations on the ring but did not appear in any simulations in the array. These transitions are AZA $\rightarrow$ G, AAR $\rightarrow$ G, RAA $\rightarrow$ G, AAP $\rightarrow$ G, PAA $\rightarrow$ G, AAG $\rightarrow$ G, GAA $\rightarrow$ G, QGG $\rightarrow$ G, and GGQ $\rightarrow$ G. These configurations are of two types. The first type is

neighbors' states	present state Z	A	B	R	P	Q	G
Z–Z	Z	Z	B				G
Z–A	A	Z	G	A			
Z–B	Z	G	B		P		
Z–R	R	P	P	Q	R	Z	G
Z–P	Z	R	B	Q			
Z–Q	Z		B	Q	R		
Z–G	A	R		B	A		G
A–A	G	G					G
A–B	A	G	G	G	P		
A–R	P	G				A	
A–P	R	G	G	Q			
A–Q	A		G		P		
A–G	R	G	G	B			G
B–B	Z				P		G
B–R	R	P	P		R	Z	G
B–P	Z	R				Q	
B–Q	Z			B	R		
B–G	A	R	B		A		G
R–R		P					G
R–P		R	Q			Z	
R–Q		P				Z	G
R–G		R	B	A		A	G
P–P							A
P–Q	Z	R				Z	
P–G			B	A		A	A
Q–G	A	R			A		G
G–G	G		G	G			F

Table 2
The transition function for the 8-state automaton

triples produced immediately following the creation of the center initiator or initiators, and the second is triples that occur just prior to synchronization. The state A is used in Szwerinski's solution as a pseudo-initiator to break symmetry in these places in the simulation. Because the ring produces more symmetric behavior than the array, more transitions designed for this purpose were needed.

Table 2 shows the transition function for the 8-state automaton. The state of an automaton at the next time step can be found by looking at the entry in the column corresponding to the automaton's present state and the row corresponding to the states of its neighbors. Since the automaton is symmetric, the orientation of the neighbors is irrelevant.

4. A non-minimal-time solution

The 6-state, non-minimal-time ring solution is an extension of Mazoyer's 6-state solution to the restricted firing synchronization problem [6]. The restricted version of the problem requires that the initiator to be located at the left endpoint of the array. The construction of a ring solution requires a slight modification of the transition function, but the solution behaves in the same manner as Mazoyer's. Since it was adapted from Mazoyer's solution, our solution requires $2n - 2$ rounds to synchronize a ring of n automata.

Mazoyer's solution works by dividing the line of n automata into unequal parts, one of length $\frac{2}{3}n$ and the other of length $\frac{1}{3}n$. An initiator is placed at the left end of the shorter segment, and each segment is then recursively subdivided. After every automaton becomes an initiator, the automata fire and the synchronization ends. A simulation of the 6-state solution for a ring with length $n = 17$ where the initiator is located in position 5 is given in Table 3. Note that synchronization occurs at time step 32, that the time step is listed in the leftmost column, and that the automaton in the first position is assumed to be adjacent to the one in the 17th position. For a more detailed description of the solution see Mazoyer's paper [6].

In order to extend Mazoyer's solution to the ring, two types of changes had to be made. First, all transitions involving the end marker were eliminated, as in the 8-state solution described above. Then transitions were added to preserve the behavior of Mazoyer's solution. These transitions prevent the wake-up signal from propagating to the left of the first initiator and keep all of the automata to the left of the first initiator quiescent.

Table 4 gives the transition function for the 6-state non-minimal-time automata. The state of an automaton at the next time step can be found by looking at the table corresponding to the automaton's present state. The state that the automaton should enter at the next time step is the one in the row and column corresponding to the states of its left and right neighbors respectively.

The fact that a solution to the restricted firing synchronization problem could be adapted to work on a ring is remarkable. Particularly interesting in this case is that Mazoyer's solution is distinctly non-symmetric. He relied on asymmetry to help him reduce the number of states needed for the solution, which is why the solution only works for the restricted version of the original problem. Despite this, the solution could be modified to work on the ring where symmetry is inherent. We conjecture that this is a consequence of the structure of Mazoyer's solution, and that not all asymmetric solutions can be modified for the ring.

5. Lower bounds

As mentioned previously, there are no known lower bounds for the firing synchronization problem on a ring. In this section we show that there is no 3-state solution to ring synchronization. We also show that there is no 4-state, symmetric, minimal-time solution.

5.1. Three-state bound

We now prove Theorem 2.2, a result stating that there is no 3-state solution to the firing synchronization problem on the ring.

5.1.1. Proof of Theorem 2.2

Proof: Denote the three states of the solution by G, Z, and F. Assume that F is the firing state, G is the initiator state, and Z is the quiescent state. Since there are only three states for the solution and the fire state cannot be used prior to the final round, there are only eight possible triples of states that may be used prior to the last round. These are: ZZZ, GZZ, ZGZ, ZZG, GZG, GGZ, ZGG, and GGG. We know that ZZZ $\rightarrow$ Z must be defined by the definition of the firing synchronization problem. We partition the triples into four classes, based on the number of initiators.

Class 0	ZZZ
Class 1	GZZ, ZGZ, ZZG
Class 2	GGZ, GZG, ZGG
Class 3	GGG

Consider the ring of length 3. By assumption, the initial configuration is ZGZ. In order to produce the next configuration we must apply three class 1 rules. The next configu-

0 :					G												
1 :					A	C											
2 :					G	B	A										
3 :					G	C	G	G									
4 :					G	B	A	B	C								
5 :					G	C	G		C	A							
6 :					G	B	A		A	A	G						
7 :					G	C	G		A	B	B	C					
8 :					G	B	A			B	C	C	A				
9 :					G	C	G	G			C	A	A	G			
10 :					G	B	A	B	C		A	A	B	B	C		
11 :					G	C	G		C		A	B	B	C	C	A	
12 :					G	B	A		C			B	C	C	A	A	G
13 :	C				G	C	G		C	A			C	A	A	B	B
14 :	C	A			G	B	A		A	A	G		A	A	B	B	C
15 :	A	A	G		G	C	G		A	B	B		A	B	B	C	C
16 :	A	B	B	A	G	B	A			B	G			B	C	C	A
17 :	B	B	A	C	G	C	G	G		B	B	C			C	A	A
18 :	B	A	C	B	G	B	A	B		B	C	C	A		A	A	B
19 :	A	C	B		G	C	G	G			C	A	A		A	B	B
20 :	C	B			G	B	A	B	C		A	A	A			B	A
21 :	B				G	C	G		C		A	A	A	G		G	C
22 :					G	B	A		C		A	A	B	B	A	G	B
23 :					G	C	G		C		A	B	B	A	C	G	C
24 :	A				G	B	A		C			B	A	C	B	G	B
25 :	G	G			G	C	G		C	A		G	C	B		G	C
26 :	A	B	C		G	B	A		A	A	C	G	B			G	B
27 :	G		C	G	G	C	G		A	C	B	G	C			G	C
28 :	A		G	A	G	B	A		G	B		G	B	A		G	B
29 :	G	C	G	C	G	C	G	C	G	C		G	C	G	C	G	C
30 :	G	B	G	B	G	B	G	B	G	B	G	G	B	G	B	G	B
31 :	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G	G
32 :	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F

Table 3

A simulation of the 6-state solution for a ring with $n = 17$. The initiator is located in position 5 and all quiescent states are indicated by blank spaces.

A	Z	A	B	C	G
Z		A	Z	G	
A	A	A	B	C	B
B	G		G	C	C
C	A	A			
G				C	C

B	Z	A	B	C	G
Z		G	B	Z	B
A	G	B	B	Z	
B	G	A	B	C	B
C	Z	A			Z
G	C	C		B	G

C	Z	A	B	C	G
Z	C	A	G	C	G
A	B		B		B
B	C			C	G
C	C	A	B	C	B
G	B		B		B

Z	Z	A	B	C	G
Z	Z	Z	Z	Z	Z
A	G	Z	Z	Z	C
B	Z	Z	Z	Z	Z
C	A	Z	Z	Z	G
G	C	Z	Z	Z	A

G	Z	A	B	C	G
Z	A	G	G	G	
A	B		G	G	
B	B		G	G	G
C	A		G	G	A
G	B		G	G	F

Table 4
The transition function for the 6-state automaton

ration, however, must have at least two initiators, since otherwise it would duplicate the initial configuration.

This means that there are two cases to consider:

1. Class 1 rules have all G's on the right hand side, or

2. Exactly two of the class 1 rules have a G on the right hand side.

In the first case, we must have GGG $\rightarrow$ F. This yields a contradiction for the ring of length four, where after one round we produce the configuration GGGZ.

In the second case, we must have all class 2 triples defined to have G on the right hand side, or we produce an infinite loop for the length three ring. This is because in the length three ring, the first configuration has two Z's and the second configuration has one Z. Since it is not possible by the definition of the problem to have three Z's, the next configuration must have no Z's.

So in the case where exactly two of the class 1 triples are defined to transition to G, all class 2 rules must use G on the right hand side. Further, we must define GGG $\rightarrow$ F, since it is the only remaining undefined triple. Consider the ring of length five. ZZZGZ yields ZZs, where s is a string with 2 initiators. If the initiators are adjacent in s, then we are done since in the next round we get the triple GGG from the triples ZGG, GGZ, and either ZZG or GZZ since two out of three of the class 1 triples are defined to transition to G. This produces a firing prior to the final round. If the string s is of the form GZG, so that the length 5 ring in round 2 looks like ZZGZG, we must have had ZGZ $\rightarrow$ Z, ZZG → G, and GZZ → G. This produces the configuration GGZGZ and then GGGZG, causing a partial firing of the ring contrary to the definition of the problem. ♢

Although this proof only considers rings of length less than or equal to 5, it is valid. A correct solution to the firing synchronization problem must synchronize all rings of length greater than or equal to 2. This assumption is motivated by the existing work on state lower bounds for the original firing synchronization problem. In fact, there are solutions to the firing squad problem on the array using only 4 states that work for small arrays. These semi-solutions synchronize arrays with 10 or fewer automata but fail to synchronize arrays of larger lengths [10]. Automata that do not synchronize all arrays of length greater than or equal to 2 are not considered genuine solutions. It is reasonable to make the same type of assumption about solutions on the ring.

There is, however, an alternate proof for Theorem 2.2 that is not based on arguments about short rings.

5.1.2. Alternate proof of Theorem 2.2

Proof: In the alternative proof, we first show that GGG is the only triple that may fire. We then show this forces certain constraints on how class 1 and class 2 triples may be used during the next to the last round of the synchronization. Finally, we show that these constraints do not allow any possible solution.

Recall that class i rules are those with i initiators on the left side of the transition for $0 \leq i \leq 3$. A misfire is the term used to describe either a configuration where at least one automaton is in the firing state while one or more automata are not firing or a correct firing configuration that occurs prematurely, that is, before round n when the ring contains n automata.

Claim 5.1 *No class 1 rule may fire.*

Proof: By definition, if a class 1 rule was defined to fire, then there would be a misfire during round 1. ♢

Claim 5.2 *Class 2 rules are either all defined to fire or none are defined to fire.*

Proof: Assume that a ring of automata synchronizes during round k and consider the ring configuration at round $k-1$. By Claim 5.1, no class 1 triples can be found in this configuration. If none of the class 2 triples are defined to fire, we are done. Assume then, that one of the class 2 triples is defined to fire. Without loss of generality, assume that the triple GGZ is found in the configuration and consider the state of the next two automata:

$$\begin{array}{l|ccccccc} \text{round } k-1 & \dots & \text{G} & \text{G} & \text{Z} & x & y & \dots \\ \text{round } k & \dots & \text{F} & \text{F} & \text{F} & \text{F} & \text{F} & \dots \end{array}$$

Since no class 1 triple may be present, x must be G, forcing GZG → F. By the same argument, y must also be G, forcing ZGG → F. ♢

Claim 5.3 *At least two class 1 triples must produce a G.*

The proof is left to the reader.

Corollary 5.1 *The only possible firing rule is GGG $\rightarrow$ F.*

Proof: Consider a ring of arbitrary length. By Claim 5.1 and Claim 5.3, either GZG or GGZ appear during round 1. But by Claim 5.2, if any class 2 rules fire, then both triples will cause a misfire. $\diamondsuit$

Corollary 5.2 *At most two class 1 triple can produce a G*

Proof: Otherwise, by Corollary 5.1, we would misfire at round 2. $\diamondsuit$

Clearly, Corollary 5.1 implies that if a ring is to synchronize at round k, then its configuration at round $k-1$ must be composed of automata in state G. We now concern ourselves with the rules used to generate this all-G configuration and show that they must obey certain constraints.

Lemma 5.1 *If a rule $xyz \rightarrow G$ is defined and is used two rounds prior to synchronization then $zyx \rightarrow G$ must be defined and it must used during the same round.*

Proof: Assume that the ring synchronizes during round k. If the ring is to fire properly, then triples ZZZ and GGG cannot appear in the ring configuration at time $k-2$. Therefore, the $k-2$ configuration must be composed only of triples of the form xxy or xyx, where x and y are either Z or G and $x \neq y$. Triples of the form xyx are symmetric and thus satisfy the theorem. Consider a triple of the form xxy together with the state a of the automaton immediately preceding this triple in the ring:

$$\begin{array}{r|cccccc} \text{round } k-2 & \ldots & a & x & x & y & \ldots \\ \text{round } k-1 & \ldots & \text{G} & \text{G} & \text{G} & \text{G} & \ldots \end{array}$$

If $a = x$, we either have the triple ZZZ or GGG, neither of which produce the required G at round $k-1$. Therefore $a = y$, forcing $yxx \rightarrow$ G. $\diamondsuit$

Lemma 5.2 *The rule ZGZ $\rightarrow$ G is used two rounds prior to synchronization if and only if the rule GZG $\rightarrow$ G is also used during that round.*

Proof: Assume that the synchronization occurs during round k. Without loss of generality, assume that the triple ZGZ is present in the configuration at round $k-2$. Consider the following 5 consecutive automata in the ring:

$$\begin{array}{r|ccccccc} \text{round } k-2 & \ldots & x & \text{Z} & \text{G} & \text{Z} & y & \ldots \\ \text{round } k-1 & \ldots & \text{G} & \text{G} & \text{G} & \text{G} & \text{G} & \ldots \end{array}$$

By Corollary 5.2, since ZGZ $\rightarrow$ G then at least one of x or y must be G, forcing GZG $\rightarrow$ G. $\diamondsuit$

We now show that under these constraints, there is no possible way to define class 1 triples that would lead to the synchronization of the ring.

Lemma 5.3 *It is not possible to produce the all-G configuration at round $k-1$ for rings of even lengths.*

Proof: Consider the round $k-2$ configuration. It cannot contain the triples GGG or ZZZ as neither are defined to produce G. Furthermore, from Corollary 5.2, class 1 triples can be defined in two ways:

- Case 1: ZGZ $\rightarrow$ Z while the other two triples produce G. By Lemma 5.2, neither ZGZ nor GZG can appear in the ring configuration at round $k-2$. This implies that the $k-2$ ring configuration is

$$\text{round } k-2 \,\Big|\, \ldots \quad \mathrm{G} \quad \mathrm{G} \quad \mathrm{Z} \quad \mathrm{Z} \quad \mathrm{G} \quad \mathrm{G} \quad \mathrm{Z} \quad \mathrm{Z} \quad \ldots$$

- Case 2: One of GZZ or ZZG produces Z while the other two produce G. Consider rounds 2 and 3:

$$\begin{array}{r|cccccccc} \text{round } 2 & \ldots & \mathrm{Z} & \mathrm{Z} & \mathrm{G} & \mathrm{G} & \mathrm{Z} & \mathrm{Z} & \ldots \\ \text{round } 3 & \ldots & \mathrm{Z} & a & x & y & b & \mathrm{Z} & \ldots \end{array}$$

  where either $a = \mathrm{G}$ and $b = \mathrm{Z}$ or vice versa. Clearly, x and y must not be both G as that would cause a misfire at round 4. But this implies by Lemma 5.1 that neither the pair ZGG and GGZ nor the pair GZZ and ZZG can appear within the ring configuration at round $k-2$. Consequently, the $k-2$ configuration must be:

$$\text{round } k-2 \,\Big|\, \ldots \quad \mathrm{Z} \quad \mathrm{G} \quad \mathrm{Z} \quad \mathrm{G} \quad \mathrm{Z} \quad \ldots$$

In either case, the configuration at round $k-2$ is only possible for rings of even lengths. Odd lengths would involve produce one or more Z to appear at round $k-1$. $\diamondsuit$

Theorem 2.2 is of course a direct corollary of Lemma 5.3. It is worth mentioning that Lemma 5.3 only rules out the synchronization of odd-length rings, which is sufficient for our purposes. To prove the impossibility of even lengths as well, one merely needs to consider the options for defining class 2 triples in the context of the proof of Lemma 5.3. $\diamondsuit$

5.2. Four state bound

Recall that Theorem 2.3 states that there is no 4-state, symmetric, minimal-time solution to the ring synchronization problem. We now give the proof of the theorem.

Proof of Theorem 2.3

This result follows from the result due to Balzer [1] and Sanders [10] that there is no 4-state minimal-time solution to the firing synchronization problem on an array, and the following lemma:

Lemma 5.4 *If there exists a symmetric, minimal-time k-state solution to the firing squad problem on a ring with $2n-2$ automata, then there exists a symmetric, minimal-time k-state solution to the firing squad problem on an array of n automata.*

To see intuitively why this lemma is true, we describe the special case of how to construct a simulation of an array of $n = 6$ automata from a simulation on a ring of $2n - 2 = 10$ automata. We first run a simulation on the ring, using the symmetric, minimal-time k-state solution to the firing synchronization problem on a ring:

0	G	Z	Z	Z	Z	Z	Z	Z	Z	Z
1	?	?	Z	Z	Z	Z	Z	Z	Z	?
2	?	?	?	Z	Z	Z	Z	Z	?	?
3	?	?	?	?	Z	Z	Z	?	?	?
4	?	?	?	?	?	Z	?	?	?	?
5	?	?	?	?	?	?	?	?	?	?
6	?	?	?	?	?	?	?	?	?	?
7	?	?	?	?	?	?	?	?	?	?
8	?	?	?	?	?	?	?	?	?	?
9	?	?	?	?	?	?	?	?	?	?
10	F	F	F	F	F	F	F	F	F	F

We obtain the simulation on an array of $n = 6$ automata by simply removing the last four columns.

In the arguments below, we will assume that $2n - 2$ automata on a ring are numbered 1 through $2n - 2$ in counter-clockwise order, with the initiator being numbered 1. Before we formally prove Lemma 5.4, we first show the following holds:

Claim 5.4 *Suppose we run a simulation of a symmetric solution on a ring of $2n - 2$ automata. Then, in any round r, automata i and $2n - i$ must be in the same state, for $i = 2, 3, ..., n - 1$.*

Proof: We use induction on r. If $r = 0$, the claim holds trivially since all relevant automata are quiescent. Consider now round $r \geq 1$ and choose some i between 2 and $n - 1$. By induction, automata $i - 1$ and $2n - i + 1$ (or 1 if $i = 2$) are in the same state in round $r - 1$, as are automata i and $2n - i$, and automata $i + 1$ and $2n - i - 1$. Since the solution is symmetric, this implies that automata i and $2n - i$ must be in the same state in round r. This is true for any i between 2 and $n - 1$, which completes the induction step. $\diamondsuit$

We are now ready to complete the proof of Lemma 5.4.

Proof: From our example, it should be clear that all we need to do is define the additional transitions for the array solution that involve the left or right end markers. Let δ_1 be the set of all the transitions of the k-state, symmetric, minimal-time ring solution that are used by automaton 1 in a ring of size $2n - 2$, for any $n \geq 2$. Each transition in δ_1 must be of the form $XYX \to W$, since automata 2 and $2n-2$ are always in the same state, by the above claim. For each such transition, we define a new, array transition $*YX \to W$. Next, we consider the set δ_n of transitions of the k-state, symmetric, minimal-time ring solution that are used by automaton n in a ring of size $2n - 2$, for any $n \geq 2$. Each transition in δ_n is also of the form $XYX \to W$. For each such transition, we define a new array transition $XY* \to W$. $\diamondsuit$

It should be noted that the requirement of symmetry in Theorem 2.3 and Lemma 5.4 is stronger than necessary. In the proof of Lemma 5.4, all we really used is that the following two conditions are satisfied by the k-state solution for the firing synchronization problem on a ring:

(a) For any simulation on a ring of even length $2n-2$, automaton 1 does not use two transitions $X_1YZ \to W_1$ and $X_2YZ \to W_2$ where $X_1 \neq X_2$ and $W_1 \neq W_2$.

(b) For any simulation on a ring of even length $2n-2$, automaton n does not use two transitions $XYZ_1 \to W_1$ and $XYZ_2 \to W_2$ where $Z_1 \neq Z_2$ and $W_1 \neq W_2$.

The proof of Lemma 5.4 contains a procedure that constructs a k-state solution for the firing synchronization problem on an array from a k-state solution on a ring that satisfies **(a)** and **(b)**, whether or not the ring solution is symmetric or minimal-time. If the ring solution is minimal-time, then the array solution constructed from it is minimal-time. If the ring solution is symmetric, then the array solution is symmetric as well. This symmetric array solution has a few additional properties. First, if $*YZ \to W_1$ and $ZYZ \to W_2$ are defined, then $W_1 = W_2$. Second, if $XY* \to W_1$ and $XYX \to W_2$ are defined, then $W_1 = W_2$.

6. Conclusion

In this paper we presented improved bounds on the complexity of solutions to the firing synchronization problem on the ring. We gave a symmetric, minimal-time solution and a non-minimal-time solution to the firing synchronization problem on the ring, both of which use fewer states than the only known ring solution by Culik [3]. We also gave the first lower bounds for the synchronization of the ring. These results are the first lower bounds for any variant of the firing synchronization problem that do not rely on exhaustive search. The fact that the proofs are straightforward is a surprising bonus.

This work leaves a gap between the best-known upper bounds and lower bounds for ring synchronization. For minimal-time solutions this gap is 4 states in the symmetric case and 5 states in general. For non-minimal-time solutions the gap is only 3 states. Reducing this gap, either by producing a smaller solution for the ring or by improving the lower bounds, is an important direction for future work.

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, L. Perković, A. Settle and J. Simon. New Bounds for the Firing Squad Problem on a Ring. In SIROCCO 9: Proceedings of the 9th International Colloquium on Structural Information and Communication Complexity, Andros, Greece, pp. 17-31, Carleton Scientific, 2002.
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. *Journal of Algorithms*, 24(1):158–170, 1997.
5. K. Kobayashi. The firing squad synchronization problem for a class of polyautomata networks. *Journal of Computer and System Sciences*, 17(3):300–318, 1978.

6. J. Mazoyer. A six-state minimal time solution to the firing synchronization problem. *Theoretical Computer Science*, 50:183–238, 1987.
7. F. R. Moore and G. G. Langdon. A generalized firing squad problem. *Information and Control*, 12:212–220, 1968.
8. Y. Nishitani and N. Honda. The firing squad synchronization problem for graphs. *Theoretical Computer Science*, 14(1):39–61, 1981.
9. R. Ostrovsky and D. Wilkerson. Faster computation on directed networks of automata. In *Proceedings of 14th Annual ACM Symposium on Principles of Distributed Computing*, pages 38–46, 1995.
10. P. Sanders. Massively parallel search for transition-tables of polyautomata. In C. Jesshope, V. Jossifov, and W. Wilhelmi, editors, *Proceedings of the VI International Workshop on Parallel Processing by Cellular Automata and Arrays*, pages 99–108. Akademie, 1994.
11. A. Settle and J. Simon. Smaller solutions for the firing squad. *Theoretical Computer Science*, 276(1):83–109, 2002.
12. T. A. Settle. New bounds for the distributed firing synchronization problem. Ph.D. thesis, Department of Computer Science, University of Chicago, 1999.
13. H. Szwerinski. Time-optimal solution of the firing-squad-synchronization-problem for n-dimensional rectangles with the general at an arbitrary position. *Theoretical Computer Science*, 19:305–320, 1982.
14. A. Waksman. An optimum solution to the firing squad synchronization problem. *Information and Control*, 9:66–78, 1966.