An Adaptive Algorithm for P System Synchronization

Michael J. Dinneen, Yun-Bum Kim, and Radu Nicolescu
Department of Computer Science, University of Auckland,
Auckland, New Zealand

CMC12, Fontainebleau, France
23 - 26 August 2011

Outline

Outline - Introduction

The Firing Squad Synchronization Problem (FSSP)

- The Firing Squad Synchronization Problem (FSSP) is one of the best studied problems for cellular automata, proposed by Myhill in 1957.
- The original problem involves constructing an one-dimensional cellular automaton, such that one of the end cell (general) causes all the other cells (soldiers) to enter a designated state (firing state), simultaneously and for the first time.

Membrane systems

- Extends an earlier version of neural P systems (non-spiking).
- System structure is a digraph, where structural arcs represent duplex communication channels (static structure—no cell creation or dissolution).
- Each cell has:
    - states and
    - multiset rewriting and transferring rules with priority (weak interpretation).
- Global clock (marks the time for all cells of a system).

Formulation of the FSSP in membrane systems

Design a P system (specify set of symbols, states and rules) that satisfies the following three conditions:

1. All cells start from an initial quiescent state, s_q.
2. Only the general can evolve from the quiescent state.
3. There exists the firing state, s_f, $s_f \neq s_q$, for all cells, such that during the last step of system evolution, all cells enter the firing state, simultaneously and for the first time.

One of the main criteria of this problem is to synchronize a system in the fewest number of steps.

Previous FSSP solutions in membrane systems

Bernardini, Gheorghe, Margenstern and Verlan (2008):

- for transition P systems with priority and polarity,
- a tree structure with the root cell as the general,
- synchronizes in $4n + 2h$ steps, where n is the number of membranes and h is the tree height.

Alhazov, Margenstern and Verlan (2008):

- for transition P systems with promoters and inhibitors,
- a tree structure with the root cell as the general,
- synchronizes in $3h + 3$ steps.

Dinneen, Kim and Nicolescu (2010):

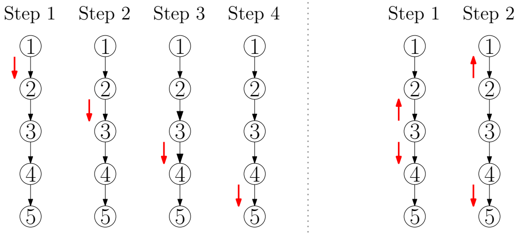
- for P systems with priority and cell states,
- a digraph structure with an arbitrary cell as the general,
- synchronizes in $3e + 10$ steps, where e is the eccentricity of the general.

Approach

- In principle, the general will send a firing order to all cells, which will prompt them to enter the firing state.
- However, in general, the general may not have direct communication channels to all cells, thus, the firing order has to be relayed through intermediate cells.
- Relaying the order through intermediate cells results in some cells receiving the order before other cells.
- To ensure that all cells enter the firing state simultaneously, each cell needs to wait until all other cells receive the order.

Approach (continued)

- In the previous solutions the order is sent from the general.
- The firing order broadcasted from the general (tree root) will reach all cells in e steps, where e is general's eccentricity.
- Sending the order from a tree center will reduce the broadcast steps by $e - r$ steps, where r is the radius of the tree.



Sending the firing order from the general (cell 1) takes four steps

Sending the firing order from a center cell 3 takes two steps

- Can the general find a more central cell to send the order?

Outline - New FSSP solution for tree-based systems

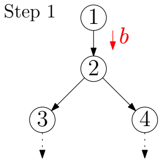
Main idea

The general finds a more central cell to send the firing order.
The general compares the heights of all its subtrees and finds the subtree that contains a center cell.

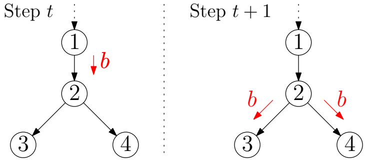
- The general sends down a signal in a BFS manner.
- The signal reflects back up the tree, once it reaches a leaf.
- The height of a subtree is half of the number of steps needed to receive a reflected signal from the subtree, minus one.
- If the height of the two of the highest subtrees differ by at most one, then the general is a center cell.
- If the height of one of the general's subtree is at least two greater than all other subtrees, then a center cell exists in that subtree.
- The root of that subtree takes the role of the general and continues the search.

Phase I: broadcast signal propagation

- The root (the general) sends down a broadcast symbol (*b*) to all its children.

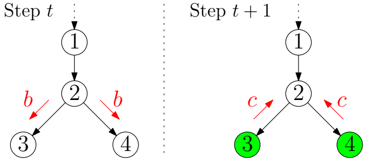


- Each cell forwards down the received symbol to its children, if any.

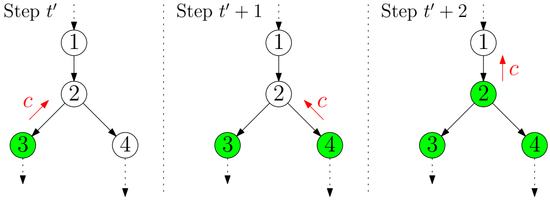


Phase I: reflected signal propagation

- Once a broadcast signal reaches a leaf, it sends up a "reflected" symbol (c) to its parent.



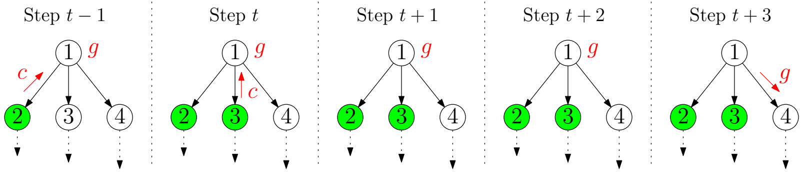
- Once a non-leaf cell receives the reflected symbols from all its children, it forwards up a reflected symbol to its parent, if any.



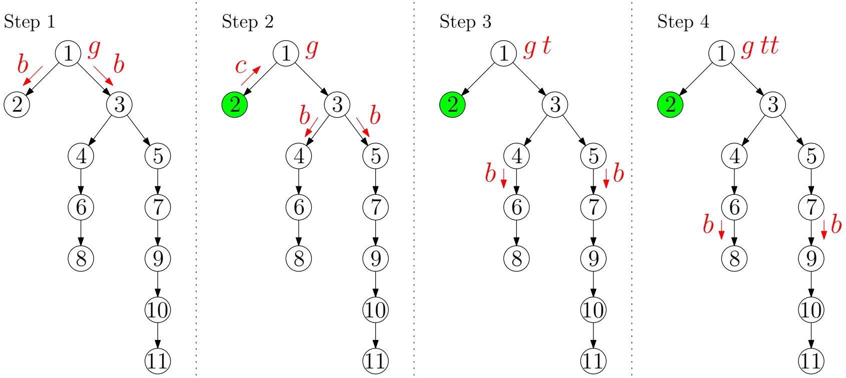
Phase I: comparing subtree heights

Starting from the general, symbol g is sent from a parent to one of its children, if that child contains a center cell.

- Let cell σ_k be the current cell that contains symbol g.
- Assume that, by step t, cell σ_k received the reflected signals from all its subtrees, except one (T).
- If cell σ_k does not receive a reflected signal from subtree T by step $t+2$, then the height of subtree T is at least two greater than all other subtrees of cell σ_k.
- In such case, at step $t+3$, cell σ_k sends symbol g to the root of subtree T.
- Here, we refer symbol g as the slow signal.

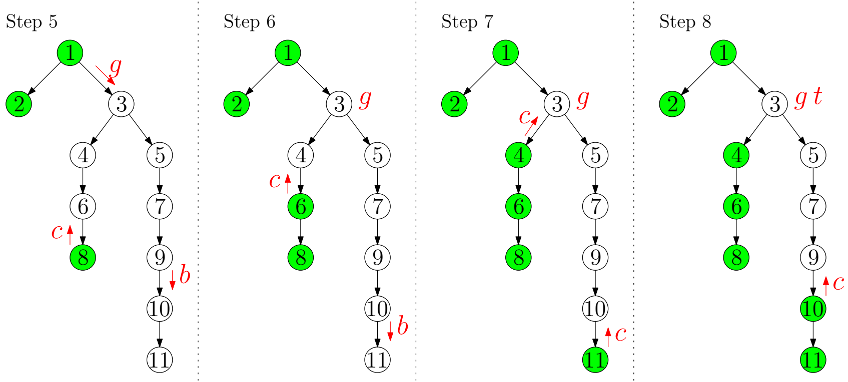


Phase I: visual description

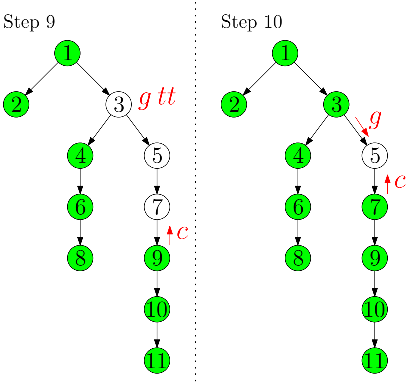


The multiplicity of symbol t represents the number of steps since the current cell that contains symbol g received the reflected signals from all its children but one.

Phase I: visual description (continued)



Phase I: visual description (continued)



An overview of Phase I

In Phase I, each cell

- starts a counter at the start of broadcast; represented by the multiplicity of a symbol.
- increments the counter by one in each step.
- ends the counter at the end of convergecast.

At the end of Phase I,

- The resulting counter equals the round trip time to a highest subtree.
- Determines its height by halving the counter.

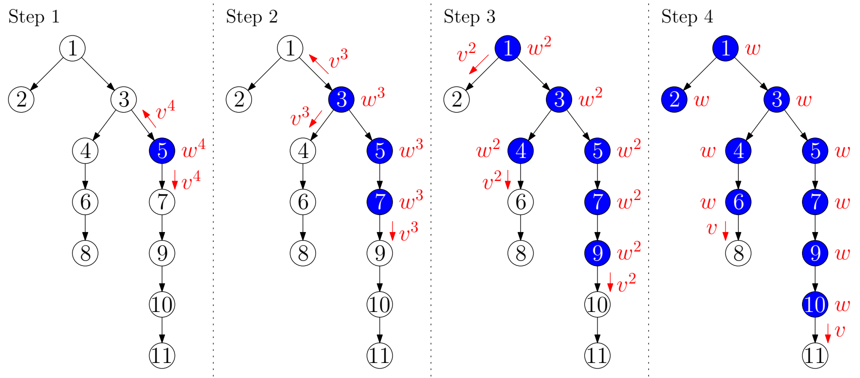
An overview of Phase I (continued)

- The slow signal and reflected signals meet at a center cell.
- The height of this center cell is the radius of the tree.
- Thus, this center cell's resulting counter corresponds to the tree radius.
- This center cell will initiate the second phase of the algorithm.

Phase II: second broadcast

- The center cell, found in Phase I, broadcasts a decrementing counter, which is initially set to its tree height, i.e. the radius of the tree.
- This counter decrements by one in each step.
- The current counter represents the remaining steps before the synchronization.

Phase II: visual description



The multiplicity of symbol *v* represents the current counter. The multiplicity of symbol *w* represents the remaining steps before entering the firing state.

Synchronization time of our solution

Synchronizes a tree-based membrane system in $h + 2r + 3$ steps, where h and r are the tree height and radius, respectively.

In Phase I:

- h steps to broadcast a signal from the general to all cells.
- r steps for the center cell to receive the reflected signals from all its subtrees.

In Phase II:

- r steps to broadcast the decrementing counter from the center cell.

Between phases:

- 3 overhead steps.

Outline - Empirical work

1. Introduction
2. New FSSP solution for tree-based systems
3. Empirical work
4. FSSP solution for digraph-based systems
5. Conclusions

Aim and method

Aim: To compare synchronization times of earlier solution ($3h$) and our solution ($h + 2r$), where h and r are the height and radius of a tree, respectively.

Method:

1. Generate random labeled trees of order $n = 1000, 2000, \ldots, 10000, 20000, \ldots, 100000$, using Prüfer correspondence.

For each tree of order n,

2. Compute the average tree height (h), i.e. the average eccentricity of all nodes, and the tree radius (r). Every node is assumed to be the location of the general.
3. Compute average gain by:

$$3h - (h + 2r)$$

and average gain % by:

$$((3h - (h + 2r))/3h) \cdot 100$$

Results

n	tree diameter	tree radius	average height	average gain	average gain %
1000	105	53	73.9	41.9	18.9
2000	126	63	95.4	64.9	22.7
3000	192	96	144.5	97.0	22.4
4000	199	100	152.6	105.1	23.0
5000	252	126	195.6	139.2	23.7
6000	268	134	198.8	129.7	21.7
7000	197	99	147.0	95.9	21.8
8000	312	156	237.8	163.6	23.0
9000	305	153	229.4	152.9	22.2
10000	354	177	263.5	173.0	21.9

Results (continued)

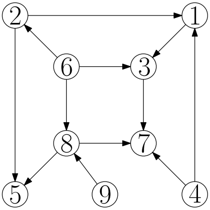
n	tree diameter	tree radius	average height	average gain	average gain %
10000	247	124	186.2	124.4	22.3
20000	418	209	321.8	225.6	23.4
30000	630	315	475.7	321.3	22.5
40000	685	343	502.3	318.7	21.1
50000	572	286	411.2	250.3	20.3
60000	668	334	501.9	335.8	22.3
70000	757	379	594.3	430.6	24.2
80000	991	496	738.7	485.5	21.9
90000	1175	588	881.3	586.7	22.2
100000	1257	629	972.4	686.9	23.5

Outline - FSSP solution for digraph-based systems

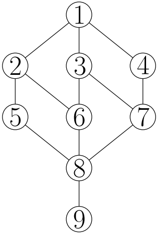
Main idea

For a given digraph, with the general located at an arbitrary node:

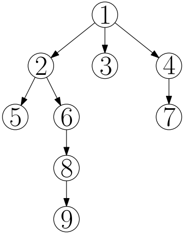
1. Construct a spanning tree, rooted at the general:
    - Perform a BFS from the general.
    - Each cell non-deterministically selects one of the BFS parents as its spanning tree parent.
2. On this spanning tree, apply the FSSP solution for tree structures.



A digraph



BFS from cell 1



A spanning tree rooted at cell 1

Remarks

- A center cell found in the spanning tree may not be a center of the digraph.
- Construction of a spanning tree requires cell identification mechanism (cell IDs):
  - Each cell σ_i appends its own ID (i) to a symbol o and sends the symbol o_i.
  - A receiver of this symbol can identify cell σ_i as the sender.
  - Each cell stores these symbols to keep track of its spanning tree parents.
- A solution based in this approach is weakly confluent; the system always reaches the firing configuration (all cells are in the firing state), but the synchronization time depends on the spanning tree.

Empirical work - aim and method

Aim: To compare synchronization times of our previous solution ($3e$) and our new solution ($e + 2r$) for digraph-based systems, where e is the eccentricity of the general and r is the radius of the spanning tree rooted at the general.

Method:

1. Generate connected random graphs of order n and size m.

For each random graph,

2. For every node v, construct a spanning tree rooted at node v.
3. Compute the average radius of all spanning trees (r) and the average eccentricity of all nodes (e).
4. Compute average gain by:

$$3e - (e + 2r)$$

and average gain % by:

$$((3e - (e + 2r))/3e) \cdot 100$$

Empirical work - results

n	m	graph radius	average spanning tree radius	average node eccentricity	average gain	average gain %
100	100	15	15.7	24.0	16.7	23.2
100	110	9	11.5	13.0	3.1	8.0
100	120	7	9.0	9.8	1.6	5.5
100	130	7	8.1	8.6	1.0	3.9
100	140	6	7.3	7.7	0.7	3.1
200	200	20	20.7	29.7	17.9	20.1
200	210	16	19.1	21.7	5.1	7.8
200	220	13	15.7	17.7	3.9	7.3
200	230	9	11.2	12.4	2.2	6.0
200	240	9	11.4	12.5	2.1	5.7

Empirical work - results (continued)

n	m	graph radius	average spanning tree radius	average node eccentricity	average gain	average gain %
300	300	25	25.0	36.2	22.3	20.6
300	310	17	19.0	22.9	8.0	11.6
300	320	16	18.6	22.8	8.3	12.1
300	330	12	15.0	16.7	3.4	6.7
300	340	12	14.0	15.3	2.5	5.4
400	400	24	24.6	36.6	24.1	22.0
400	410	22	24.8	28.7	7.7	9.0
400	420	19	21.9	25.5	7.1	9.3
400	430	15	17.9	19.2	2.8	4.8
400	440	13	15.9	17.0	2.3	4.5

Outline - Conclusions

1. Introduction
2. New FSSP solution for tree-based systems
3. Empirical work
4. FSSP solution for digraph-based systems
5. Conclusions

Summary

- We presented a new deterministic FSSP solution for a tree-based system, which synchronizes the system in $h + 2r + 3$ steps, where h and r are the height and radius of a tree, respectively.
- Our empirical results show at least 20% reduction in the number of steps needed to synchronize.
- We presented a solution for a digraph-based system, which first constructs a spanning tree and then applies the solution for tree-based systems.
- Open problem: find a solution that finds a center of a digraph.

Thank you

Thank you.