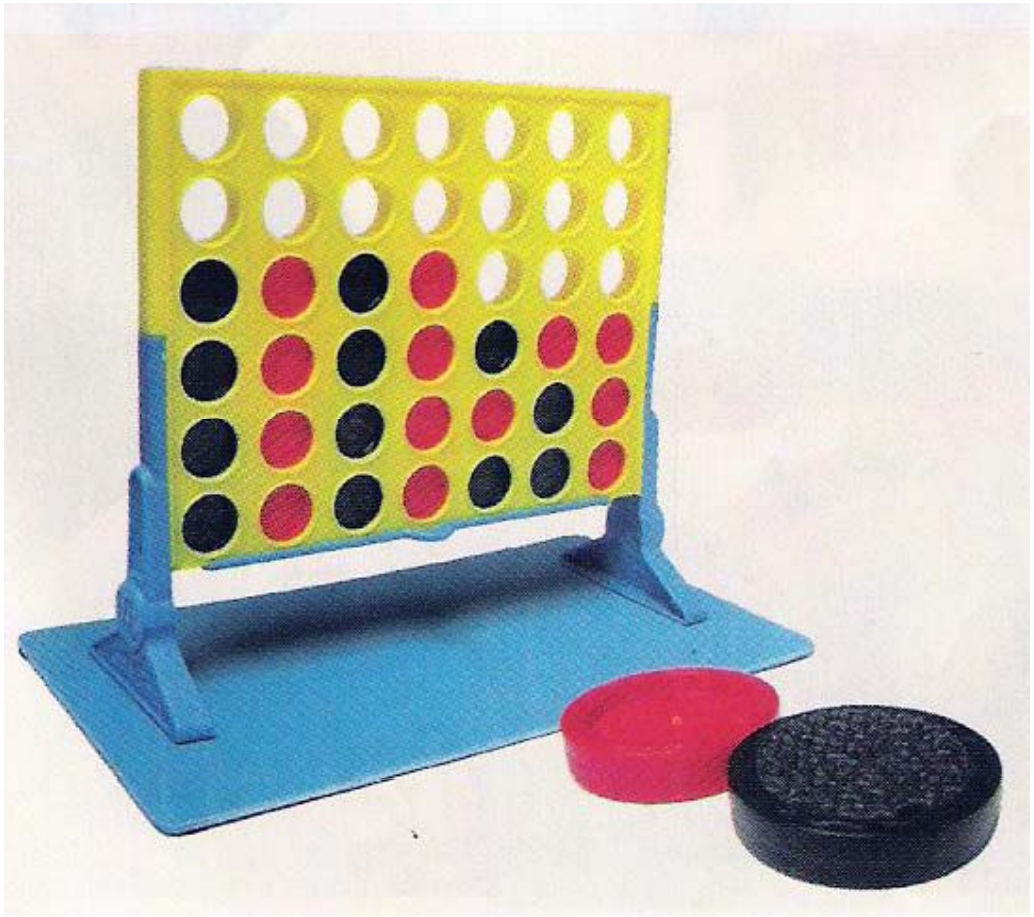
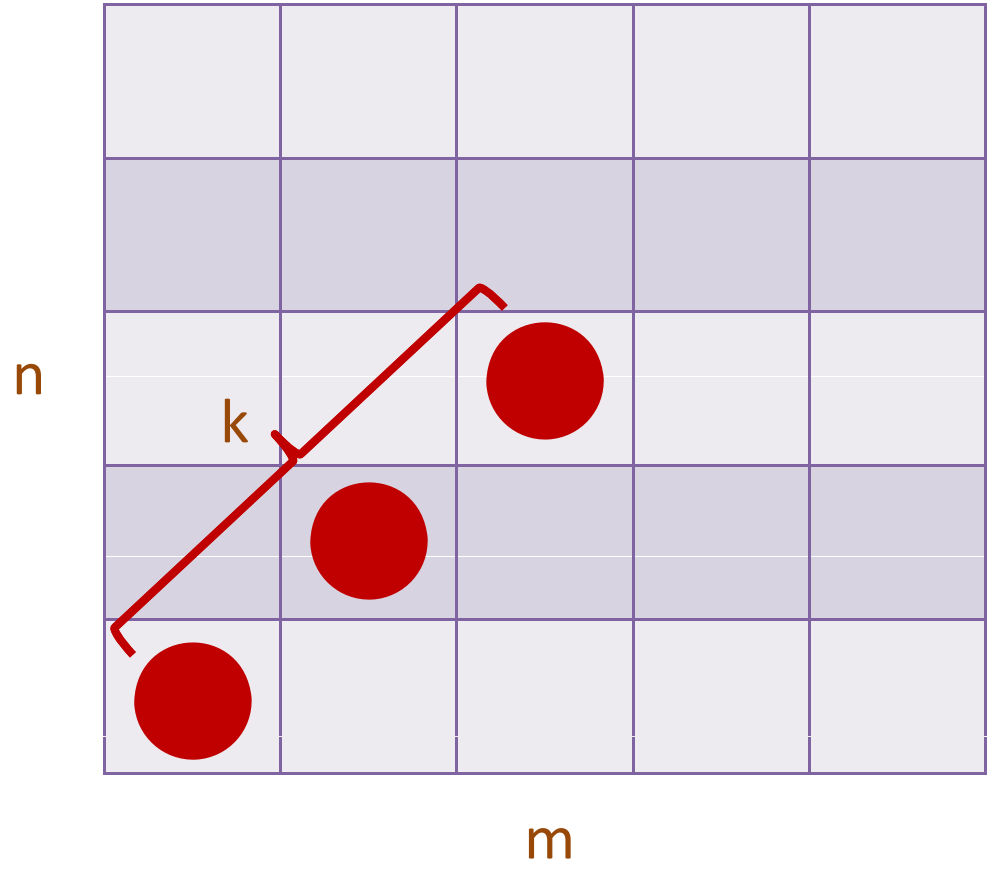
A Fast Thinking Connect Four Machine!



6.846 Final Project
Presented by Tina Wen

Goal

Write a fast connect k game on m by n grid from scratch

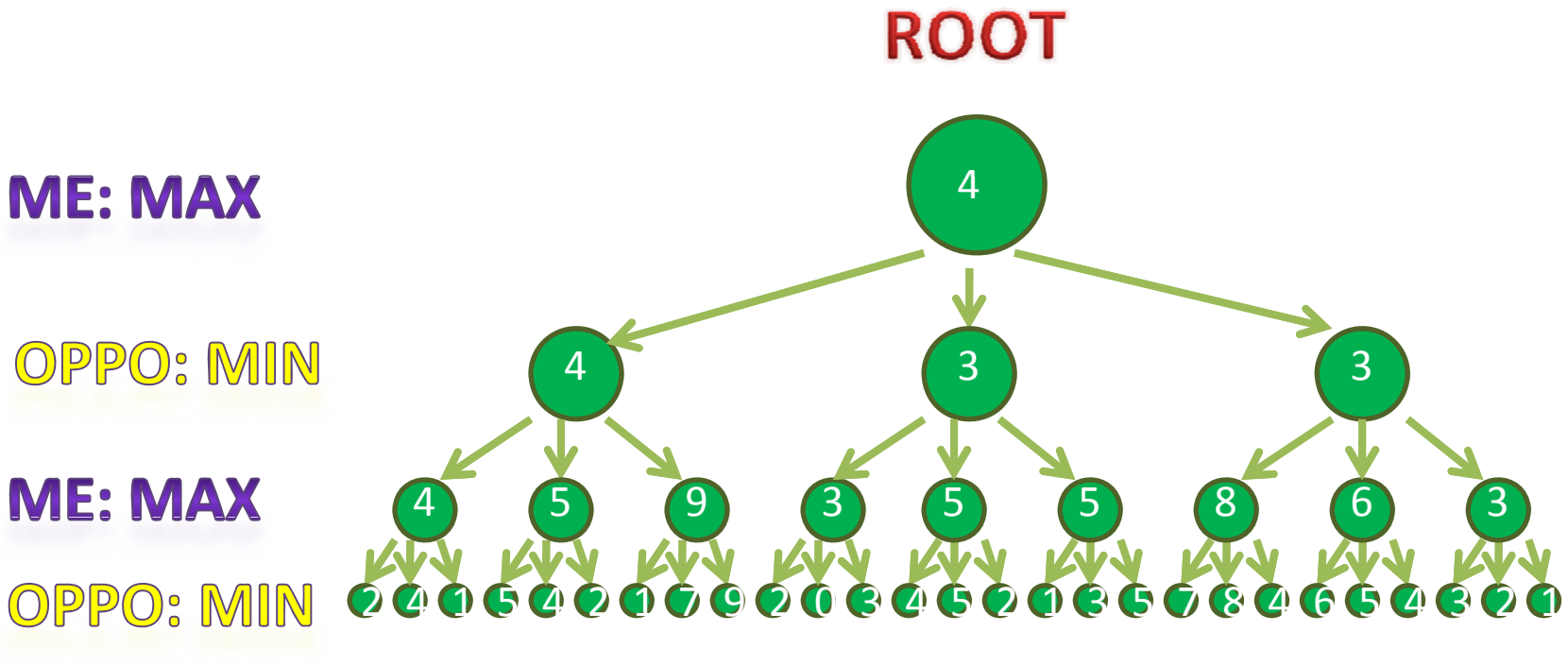


Small m and n:
exhaustive search

Big m and n: search to
a good depth

Basic Idea

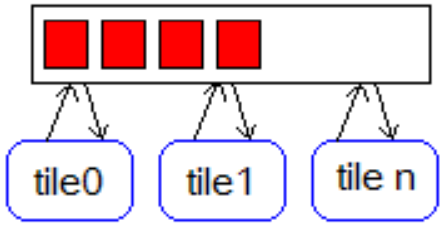
Negamax search (recursively call expand and combine)



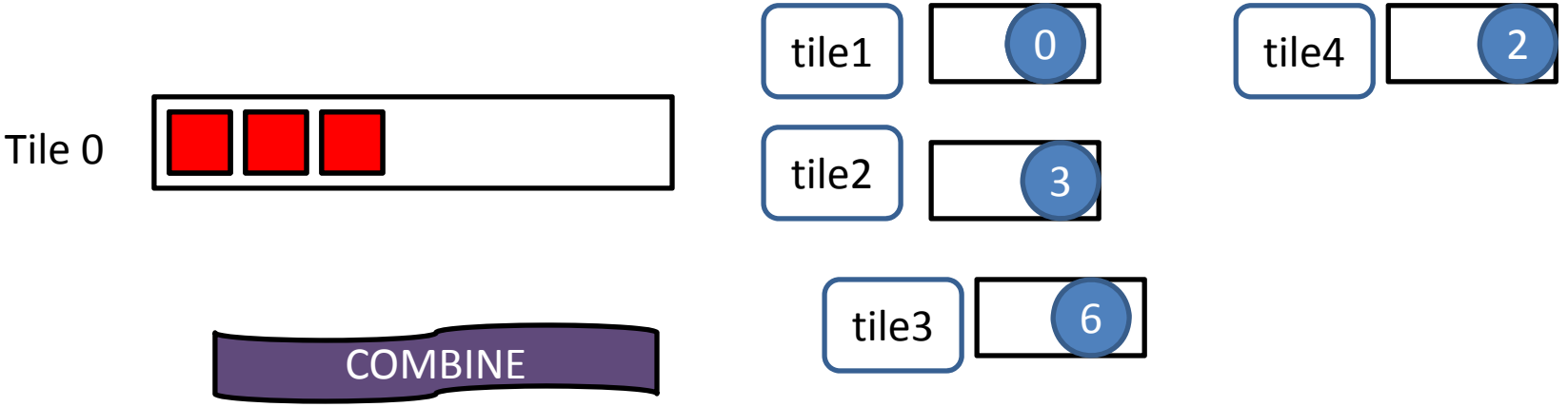
Approaches

1. Sequential exhaustive search

2. One dynamic queue in shared memory (TSP)

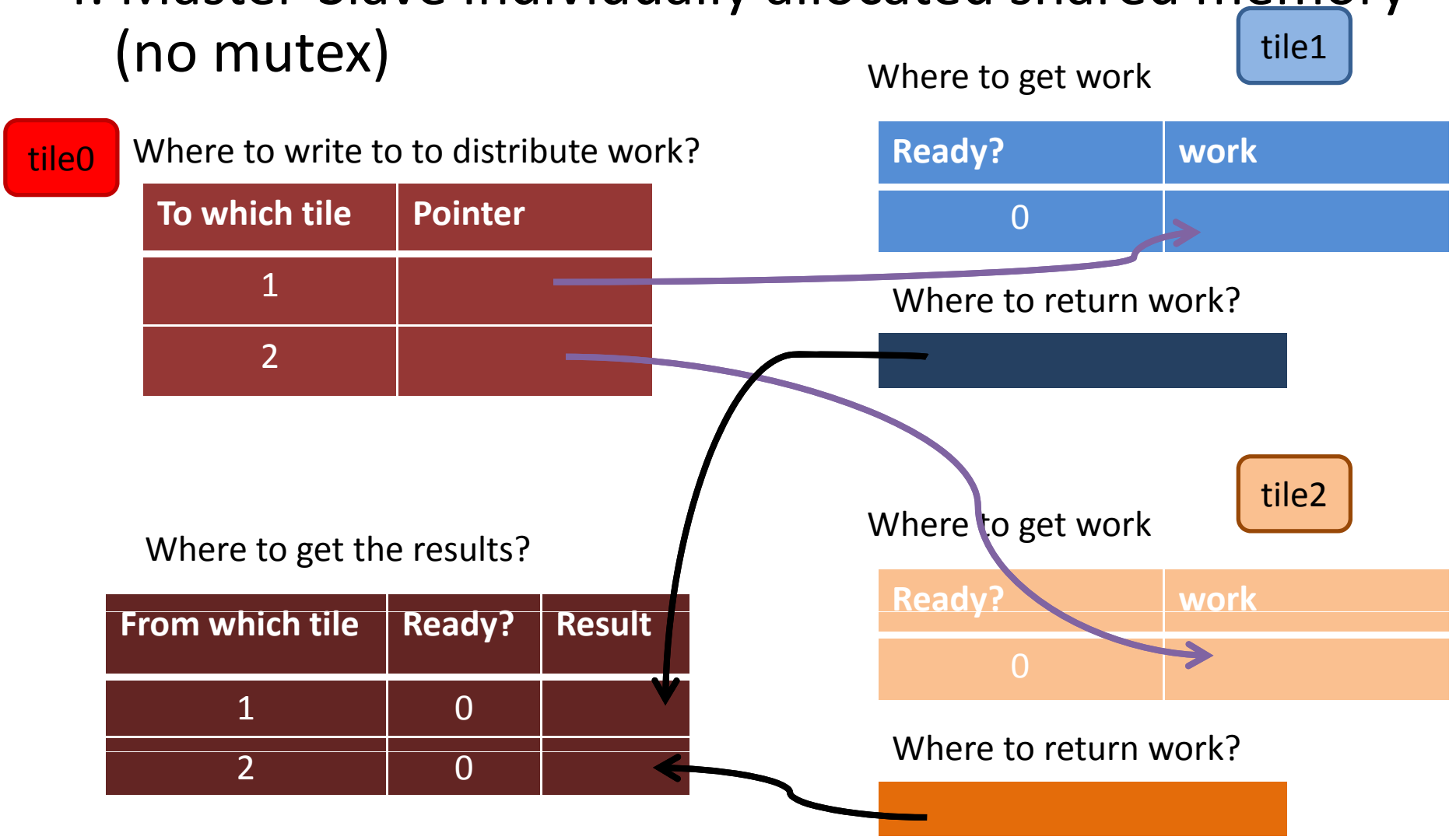


3. Master-Slave Message Passing



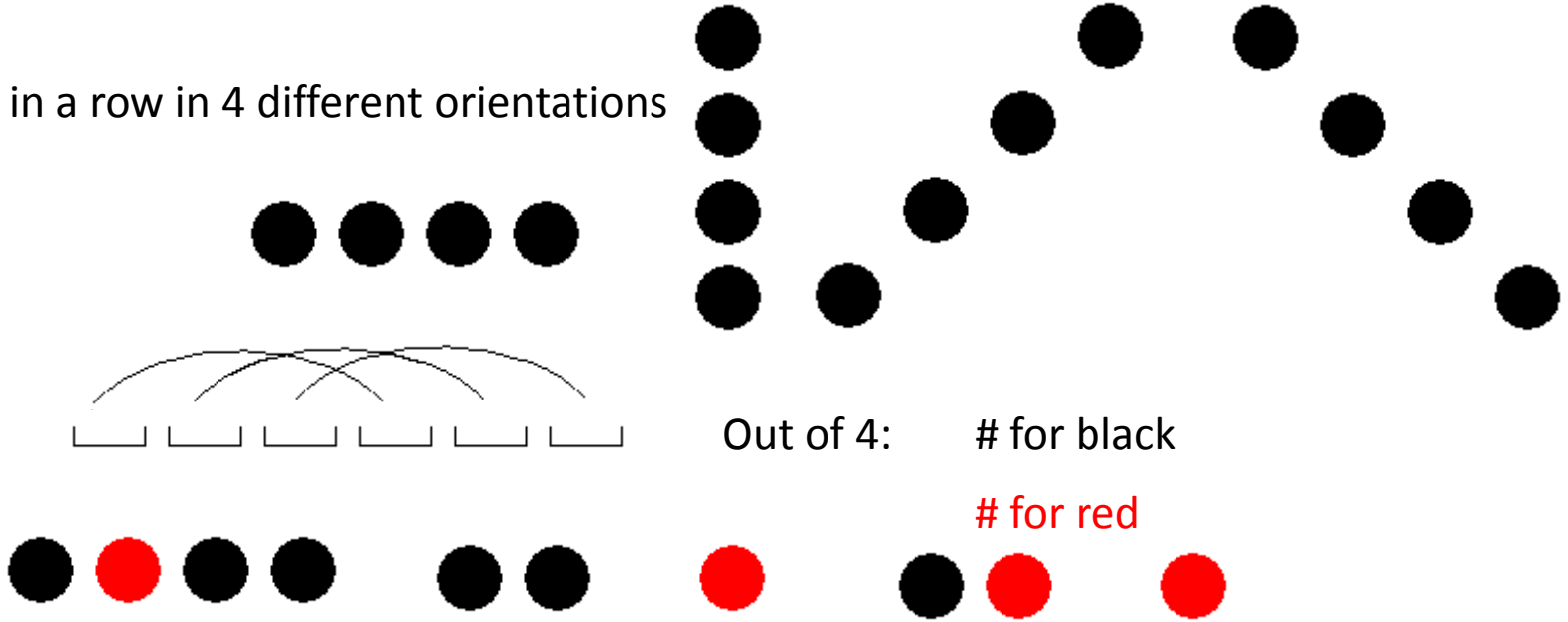
Approaches

4. Master-Slave Individually allocated shared memory (no mutex)



Heuristic

• Search for 4 in a row in 4 different orientations



Out of 4: # for black

for red

• Count only rows that consist of one color and space (either red or black == 0)

If we are red

• If one color has 4 in a row, then score =  (infinity + spaces)

Else score += 

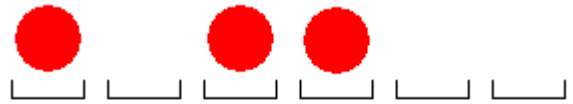
Heuristic

1 

score

9 + 4 + 1 = 14

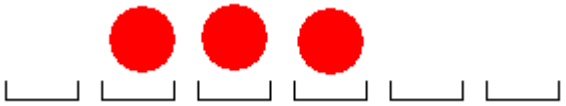
score += 

2 

9 + 4 + 4 = 17

3 

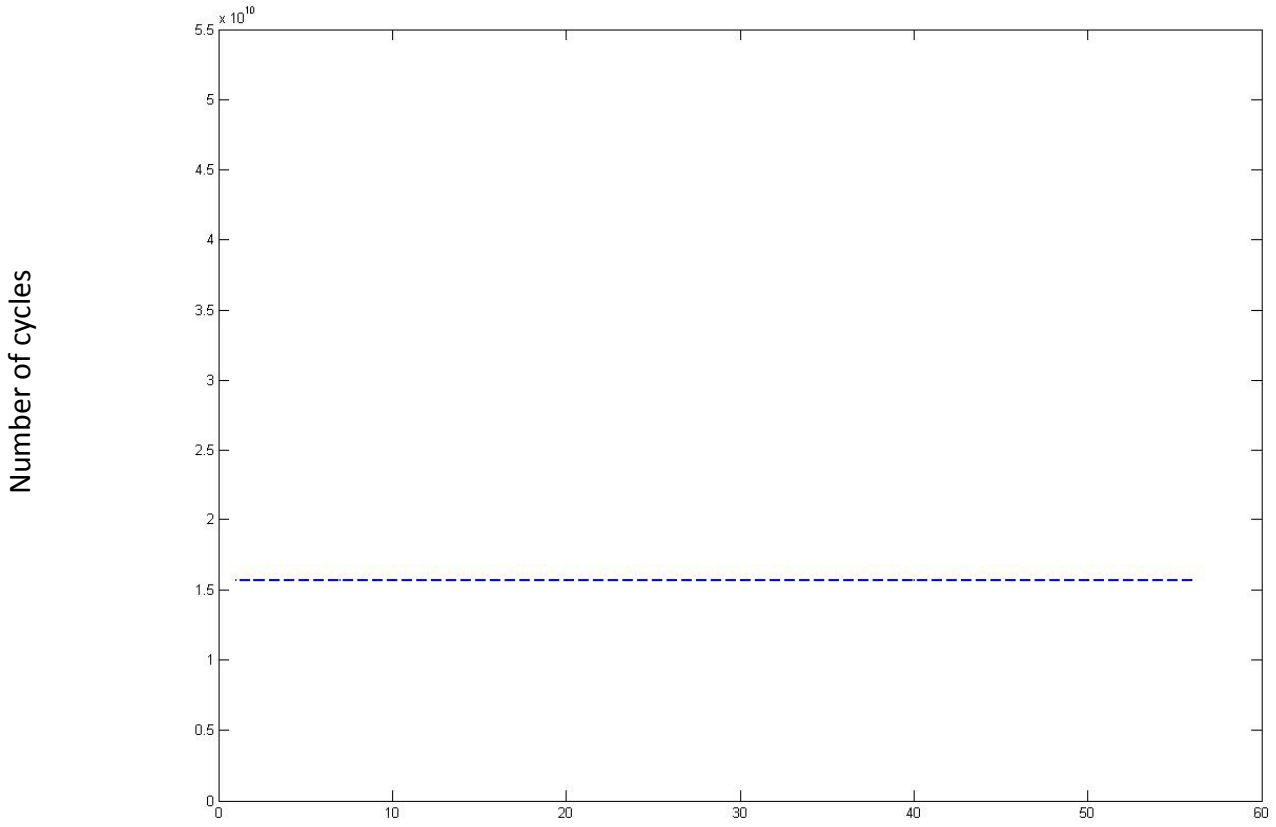
9 + 4 +1 = 14

4 

9 + 9 + 4 = 22

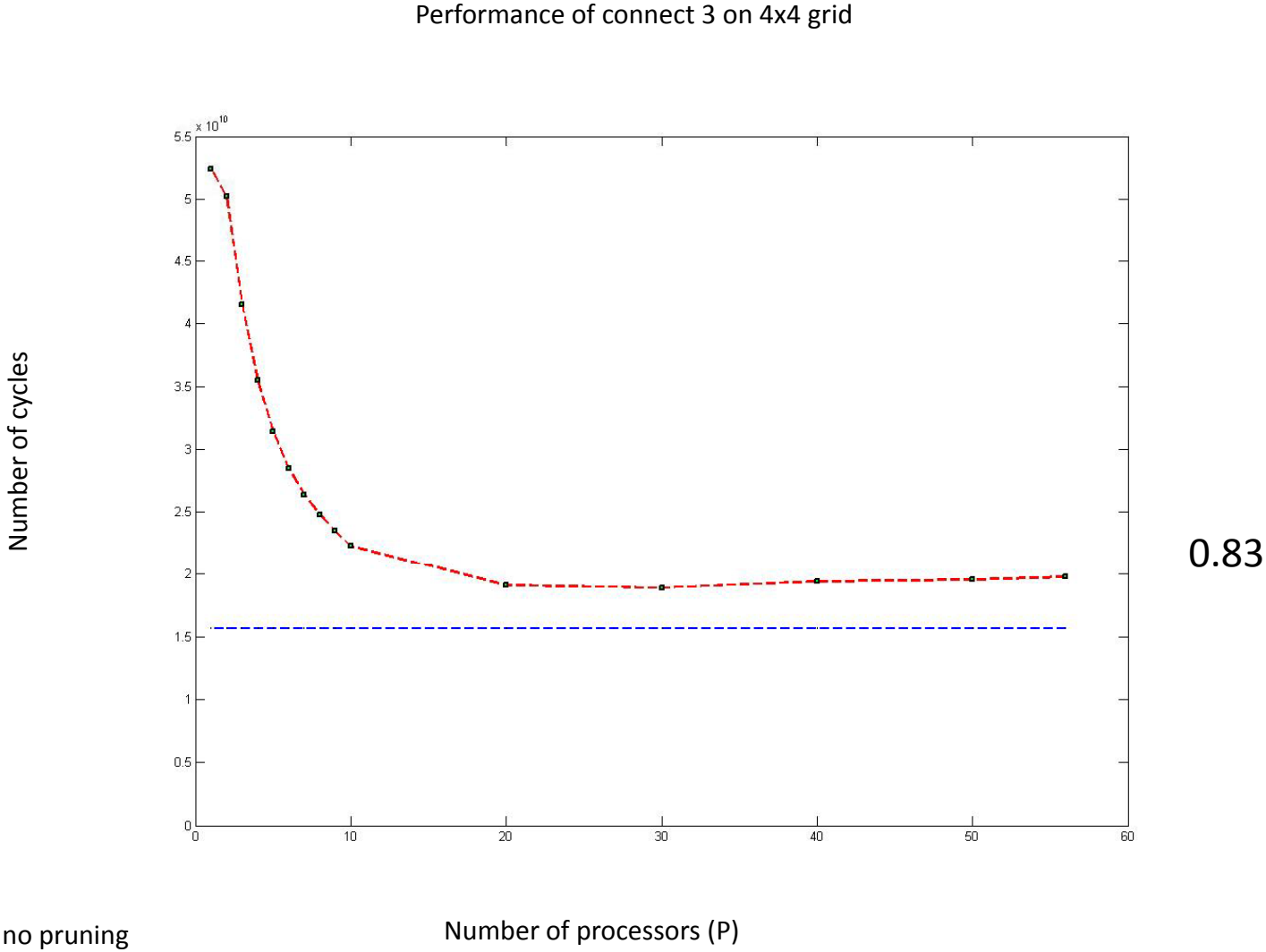
Performance Comparison (connect 3)

Performance of connect 3 on 4x4 grid



Purple: sequential no pruning

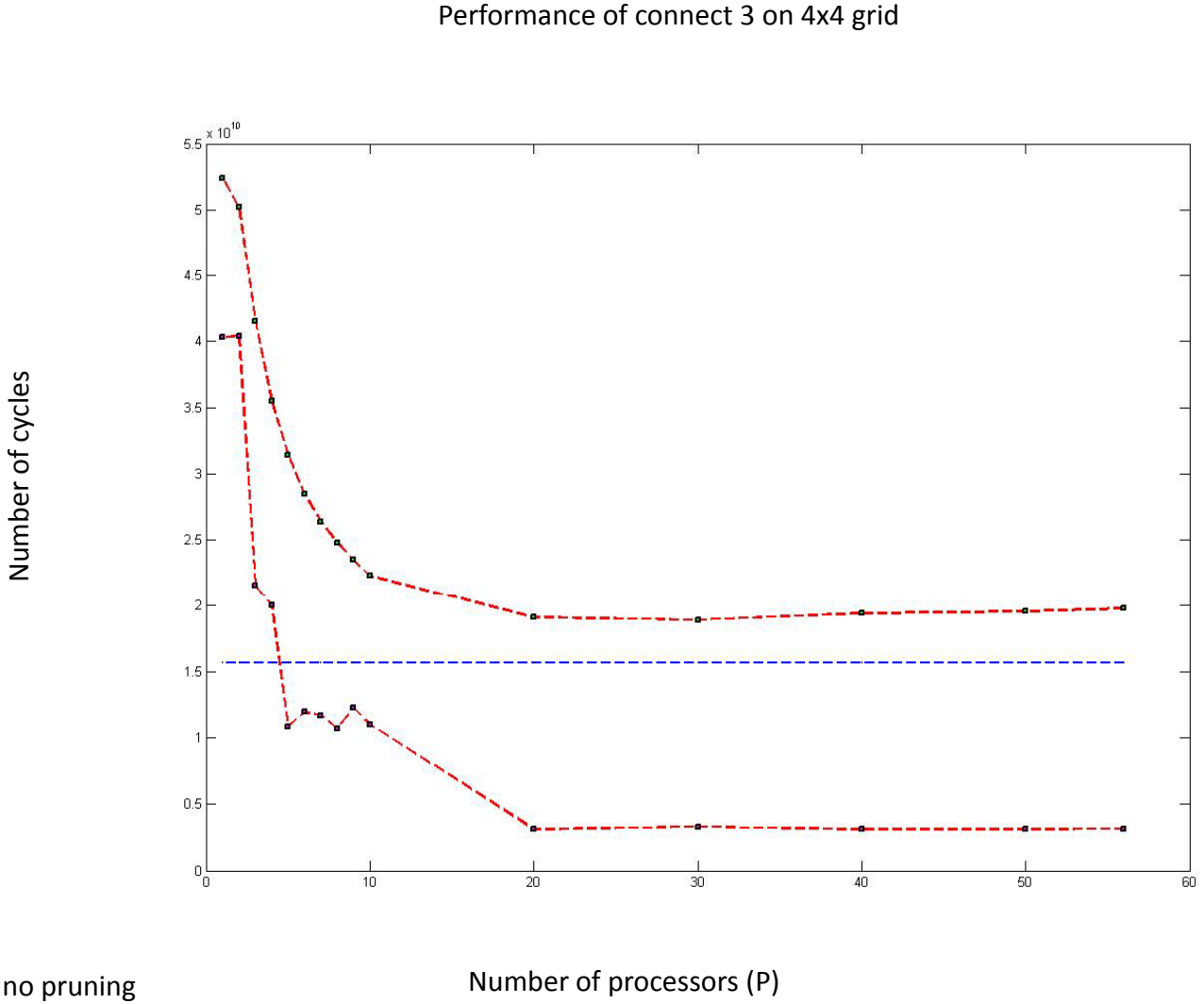
Performance Comparison (connect 3)



0.83

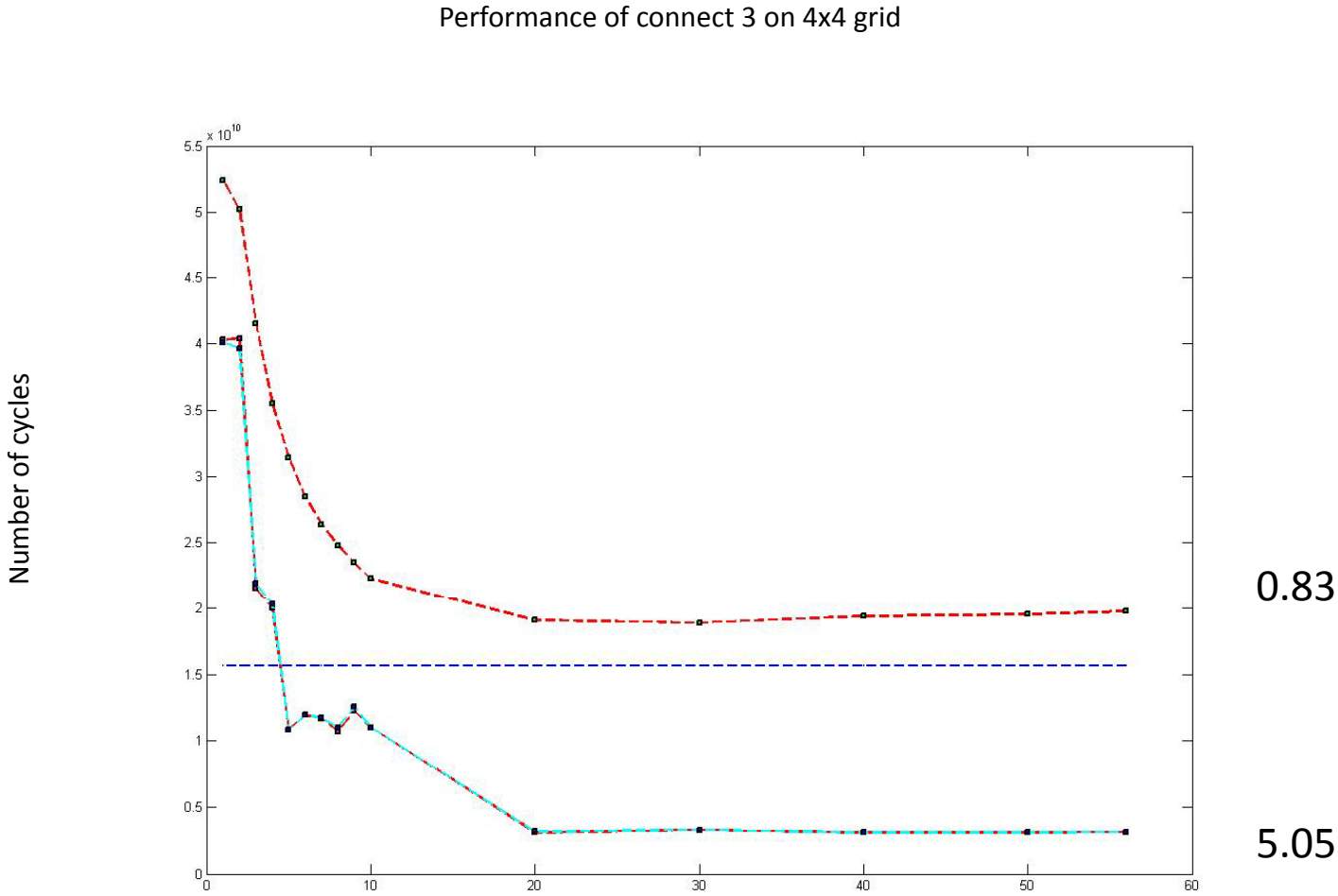
Purple: sequential no pruning
Red: dynamic queue

Performance Comparison (connect 3)



Purple: sequential no pruning
Upper Red: dynamic queue
Lower Red: message passing

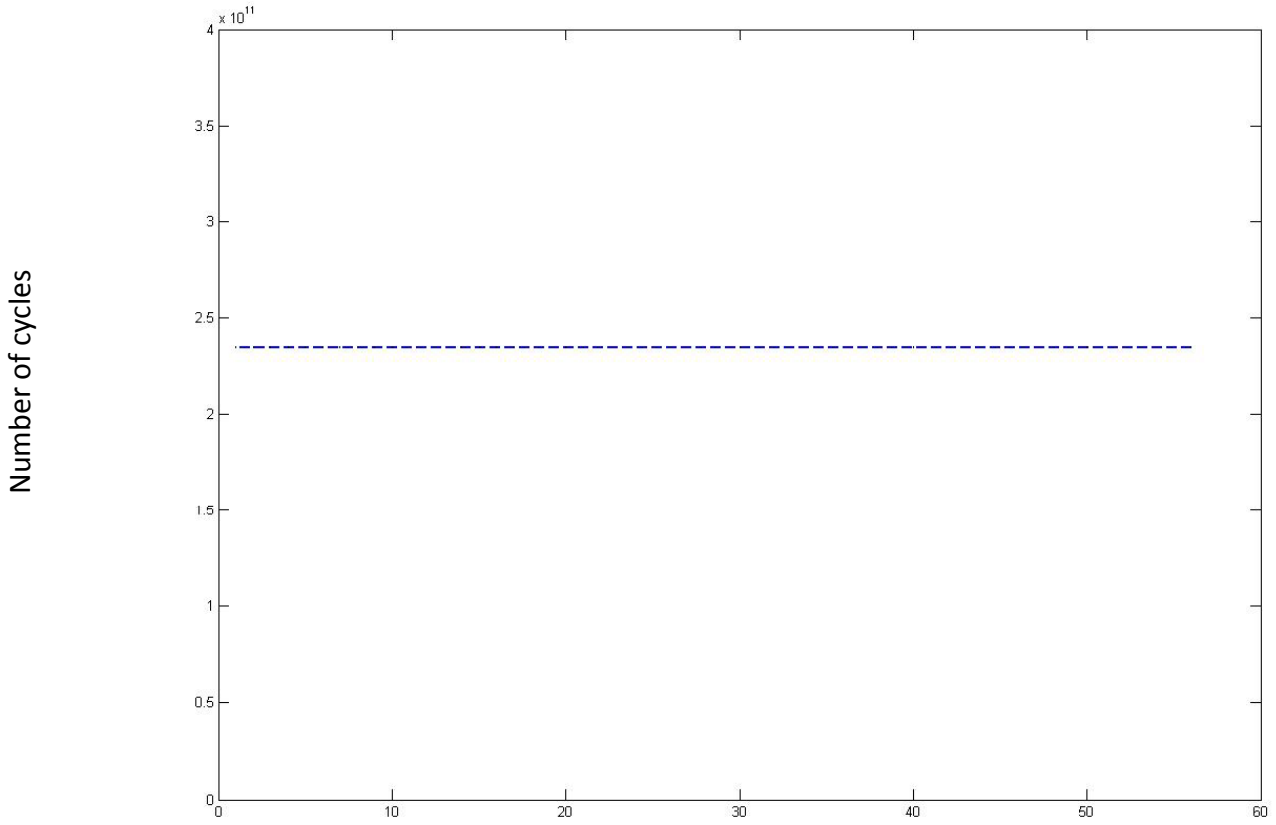
Performance Comparison (connect 3)



Purple: sequential no pruning
Upper Red: dynamic queue
Lower Red: message passing
Blue: individually allocated shared memory

Performance Comparison (connect 4)

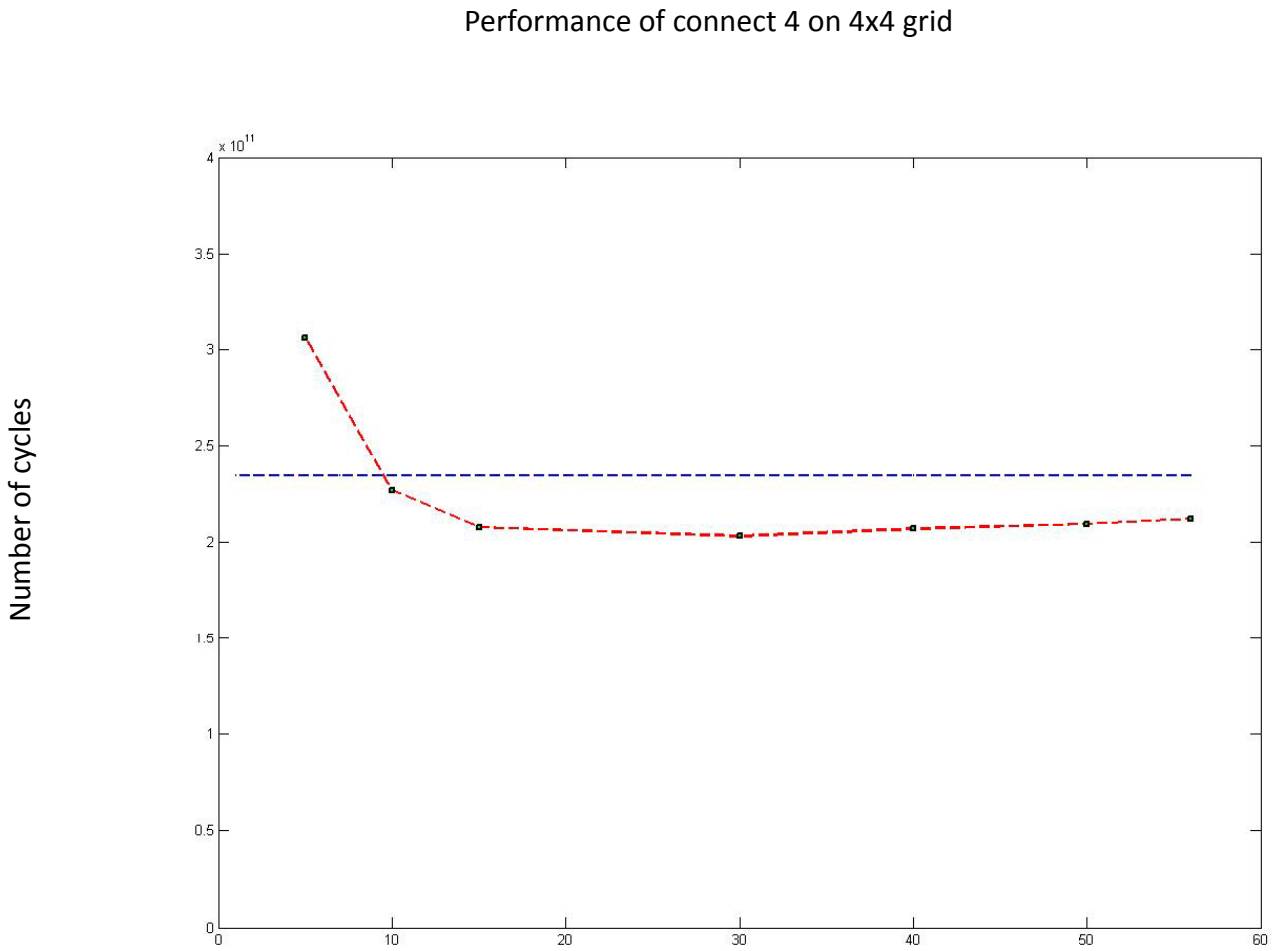
Performance of connect 4 on 4x4 grid



Purple: sequential no pruning

Number of processors (P)

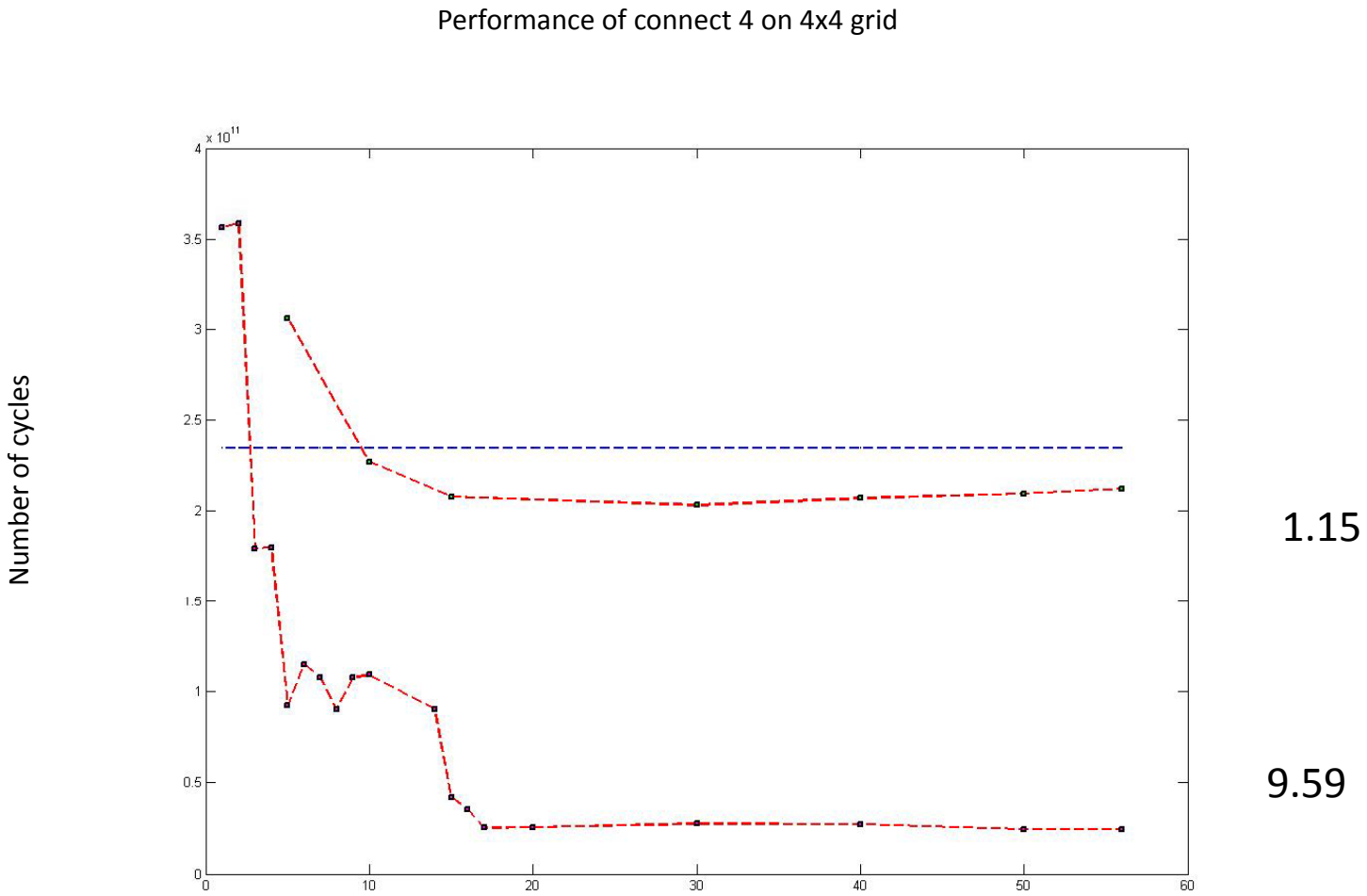
Performance Comparison (connect 4)



1.15

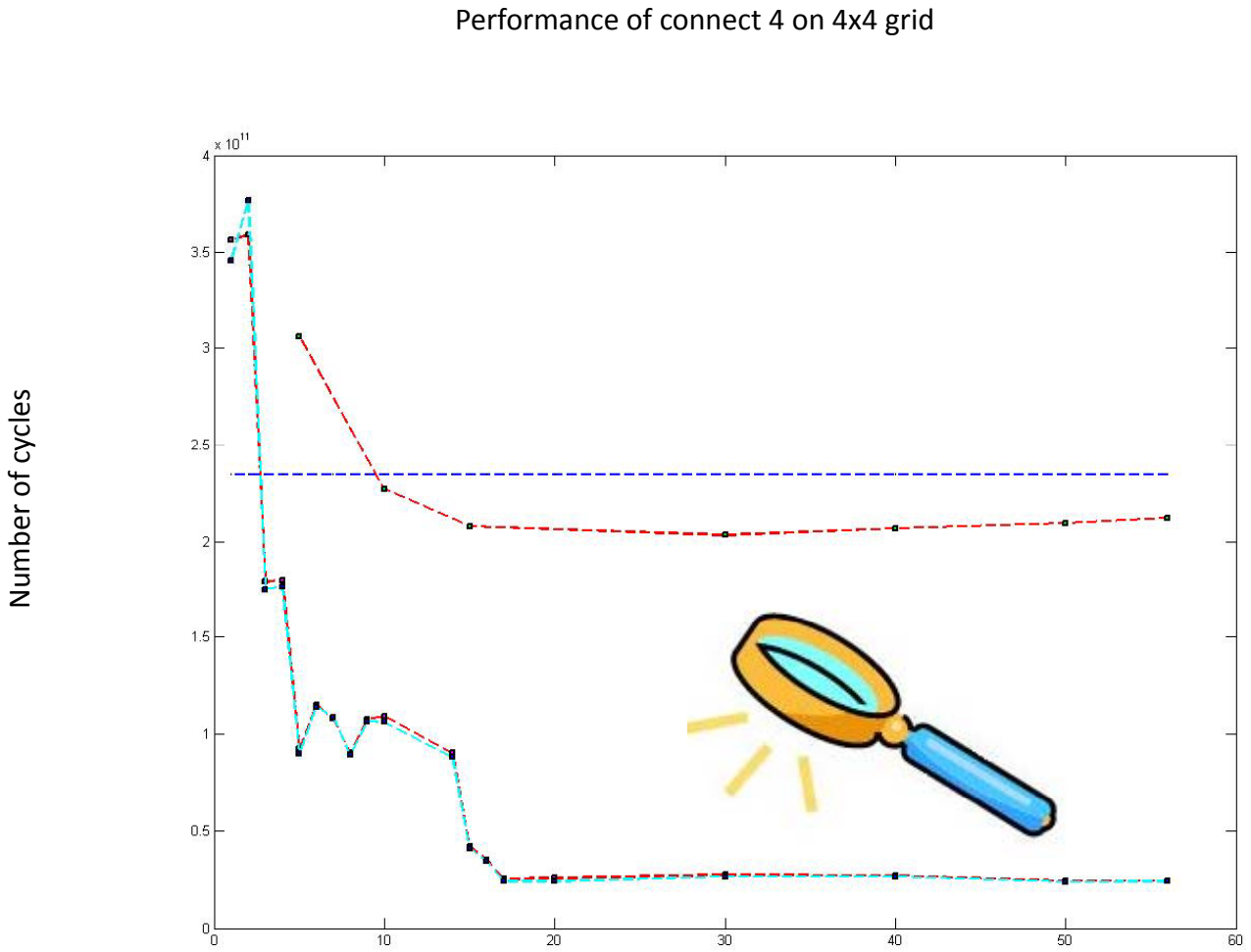
Purple: sequential no pruning
Red: dynamic queue

Performance Comparison (connect 4)



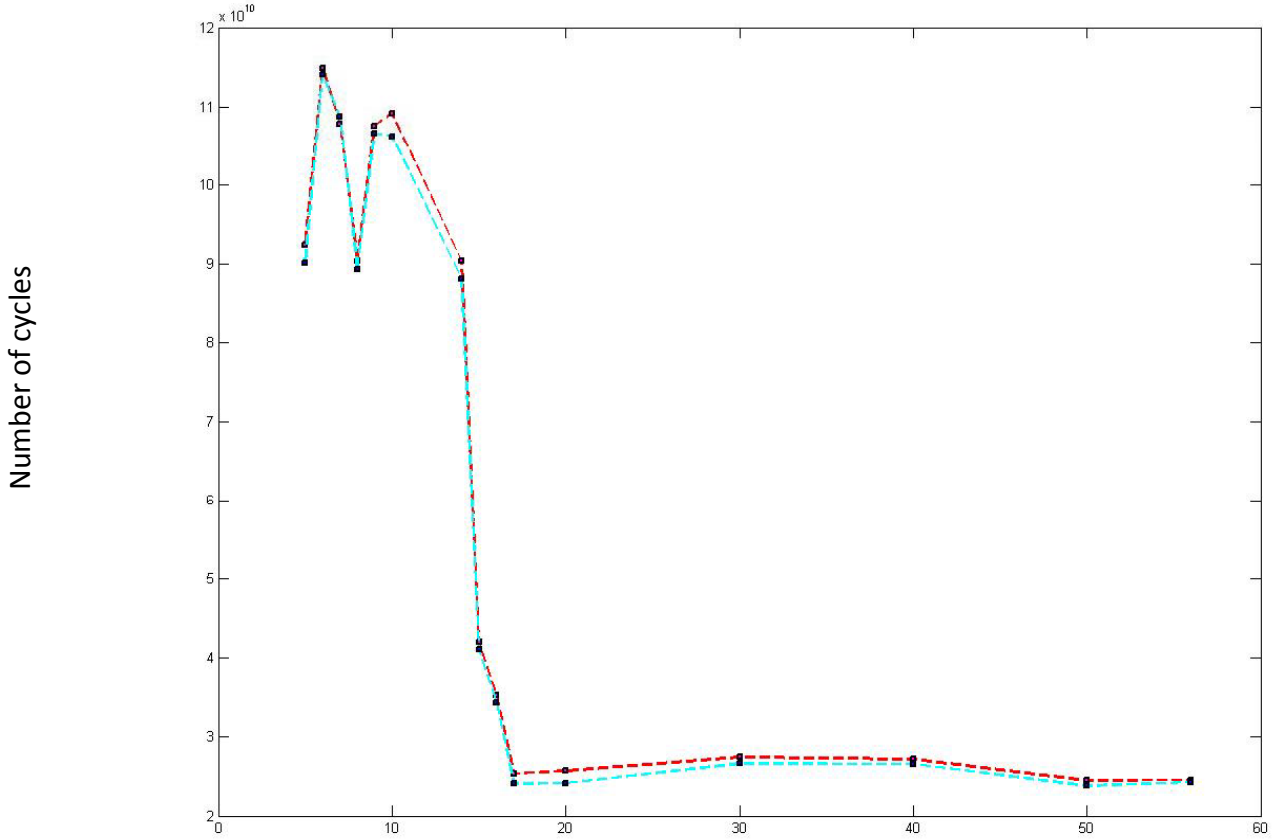
Purple: sequential no pruning
Upper Red: dynamic queue
Lower Red: message passing

Performance Comparison (connect 4)



Purple: sequential no pruning
Upper Red: dynamic queue
Lower Red: message passing
Blue: individually allocated shared memory

Performance Comparison (connect 4)

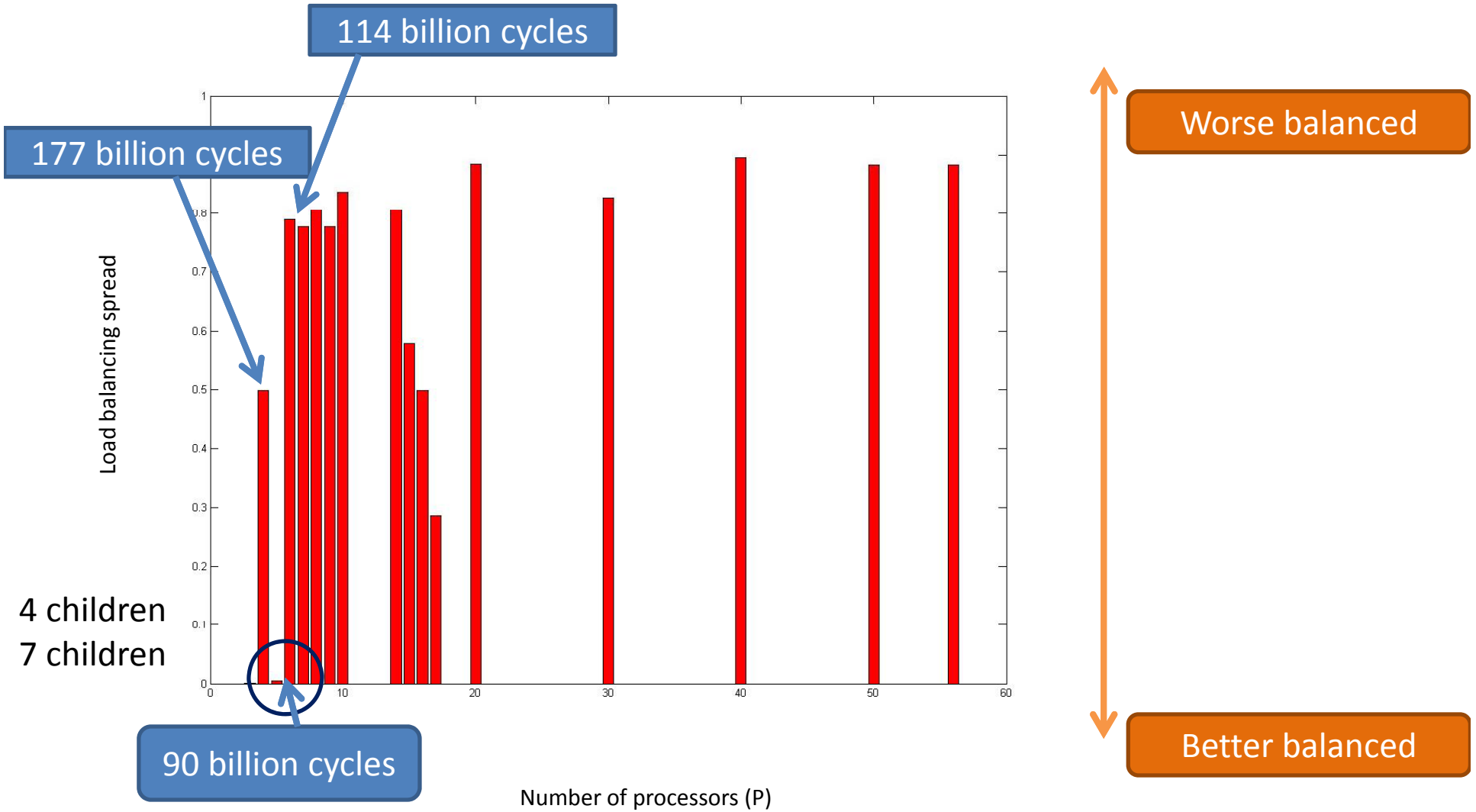


9.59

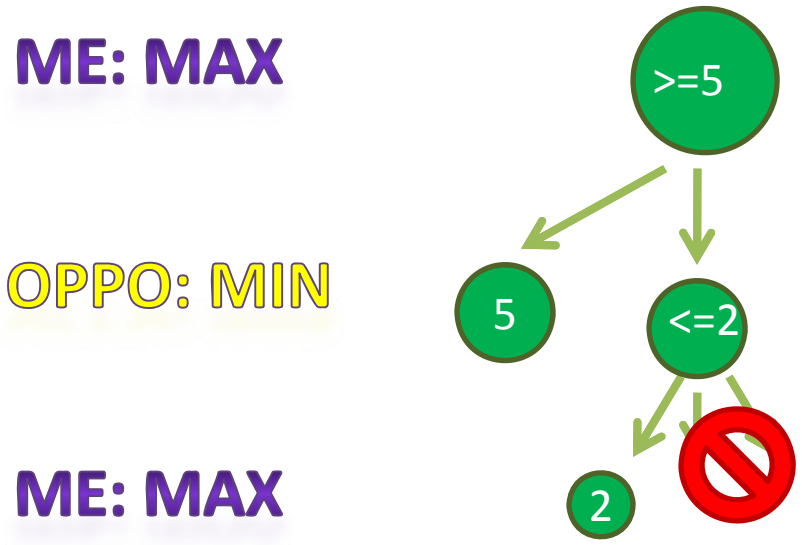
9.87

Blue: individually allocated shared memory
Red: message passing

Load Balancing

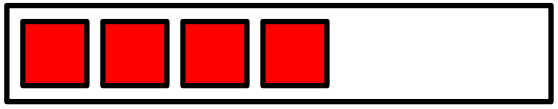


Sequential Pruning



Number of cycles	Sequential pruning	Individually allocated shared memory (min)
3x3 connect 3	0.157 billion	3 billion
3x3 connect 4	0.162 billion	23 billion

Final Version

Tile 0 

Other tiles: sequential pruning

Pros: Take advantage of both dynamic queue for load balancing and pruning
Cons: Can't do pruning between processors

Number of cycles	**Sequential pruning**	**Individually allocated shared memory (min)**	**Final Version (min)**
3x3 connect 3	0.157 billion	3 billion (P=50)	0.069 billion (P=20)
3x3 connect 4	0.162 billion	23 billion (P=50)	0.14 billion(P=20)

Interactive Mode

```
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

Welcome to the Connect-4 game! If you want to play first, key '0'; If you want me to play first, key '1'. Press ENTER to confirm
My turn now. Thinking...
score = -5 nextPos = 5
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | |o| |

time it took for first compute is 2710034100
It's your turn now. Please place your disc
You chose to place your disc in column 3
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |x| |o| |

In combine, My turn now. Thinking...
score = -6 nextPos = 3
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |o| | | |
| | | |x| |o| |

time it took for first compute is 7954251028
It's your turn now. Please place your disc
You chose to place your disc in column 2
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |o| | | |
| | |x|x| |o| |

In combine, My turn now. Thinking...
score = -3 nextPos = 1
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |o| | | |
| |o|x|x| |o| |

time it took for first compute is 19558239870
It's your turn now. Please place your disc
You chose to place your disc in column 3
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | |x| | | |
| | | |o| | | |
| |o|x|x| |o| ||

In combine, My turn now. Thinking...
score = -1 nextPos = 3
| | | | | | | |
| | | | | | | |
| | | |o| | | |
| | | |x| | | |
| | | |o| | | |
```

Conclusion

By using

- Pruning
- Master slave structure
- Individually allocated shared memory

My connect four

- searches fast
- has a good heuristic