Distributed Battle Management

Distributed Leader Election

Timothy Woods

25 November 2014

EECE 506

Outline

- Problem
  - Background
  - Refinement
- Solution
  - Algorithm evaluation
  - Mathematical evaluation
- Modeling and simulation
- Results and conclusions

Origin

Battle management has traditionally been a manual process relying on high-availability communications networks with reach back to wide body Battle Management Command and Control (BMC2) aircraft and ground stations via satellite communications (SATCOM) and tactical data links

<u>While successful in past conflicts, such an approach is no longer sufficient!</u>

In the contested environment availability of communications connectivity is not assured (e.g., due to jamming).

DBM will develop decision aids for airborne battle managers and pilots in tactical aircraft, as well as autonomy for unmanned systems, to manage complex kill chains for air-to-air and air-to-ground missions in a contested environment.

DARPA

Origin

Battle management has traditionally been a manual process relying on high-availability

comm … and

WHEN COMMUNICATIONS DROP OUT HOW DO WE COMPLETE THE MISSION?

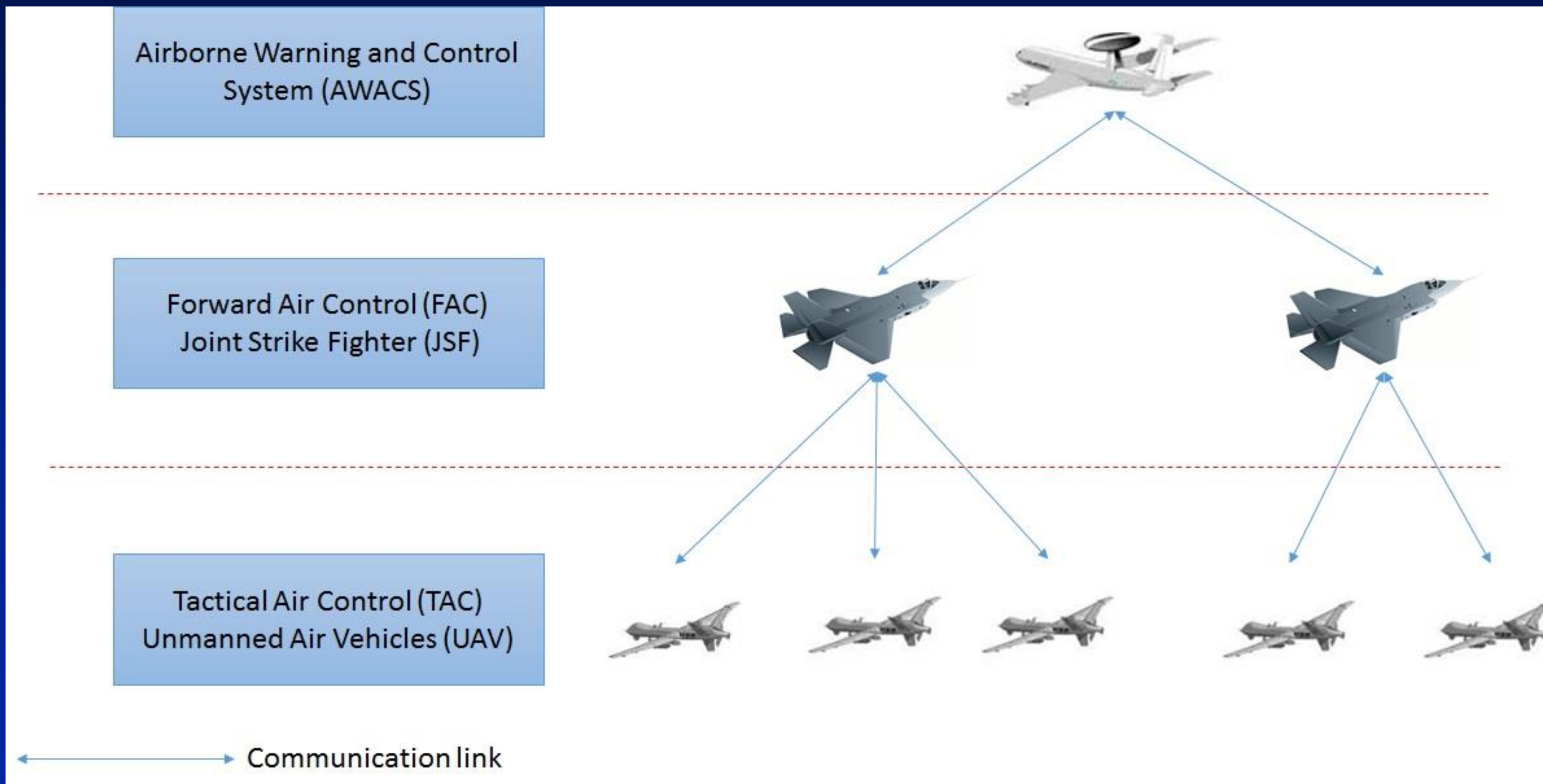
In th … ured

D … l

aircr … for

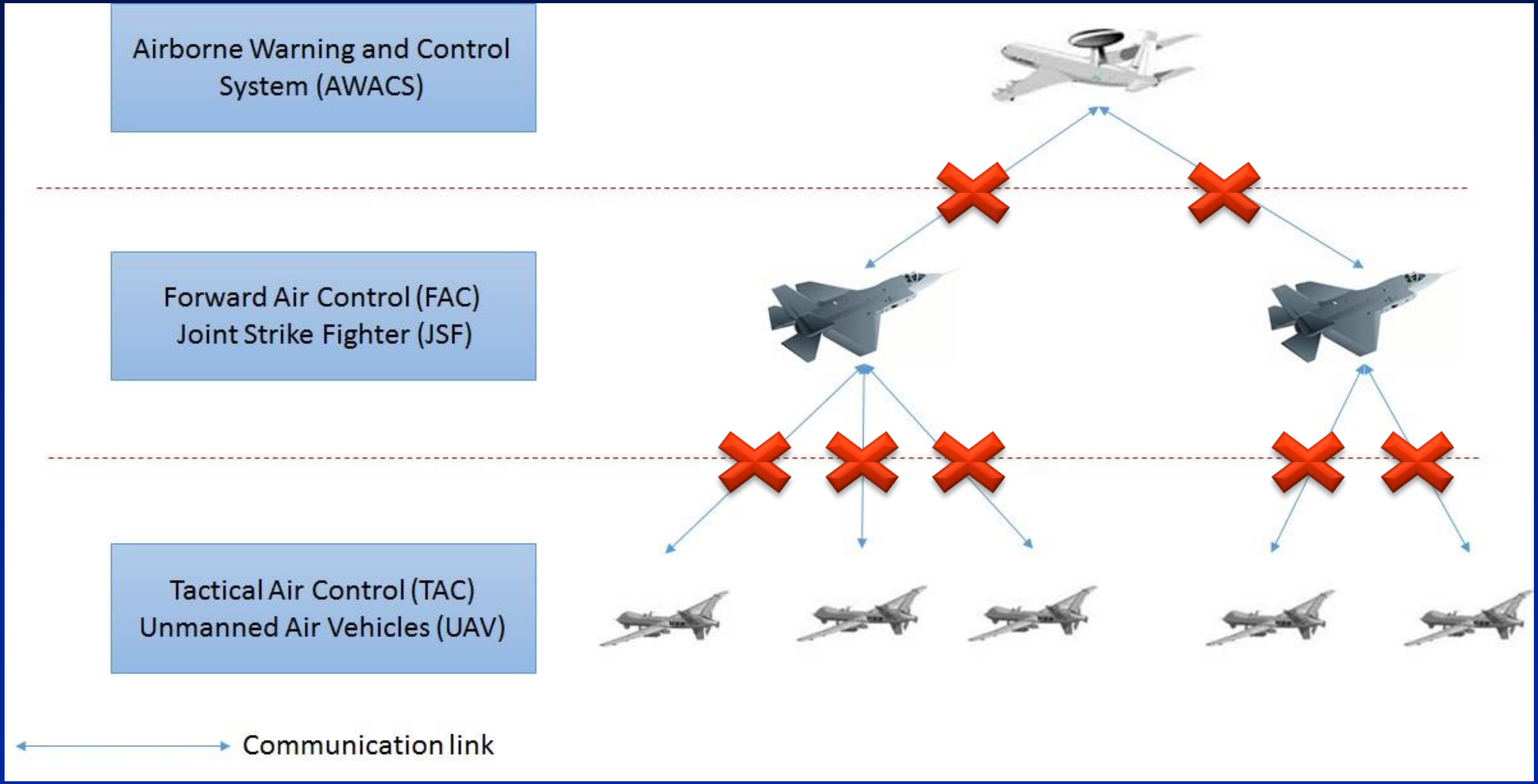
air-to-air and air-to-ground missions in a contested environment.

Background

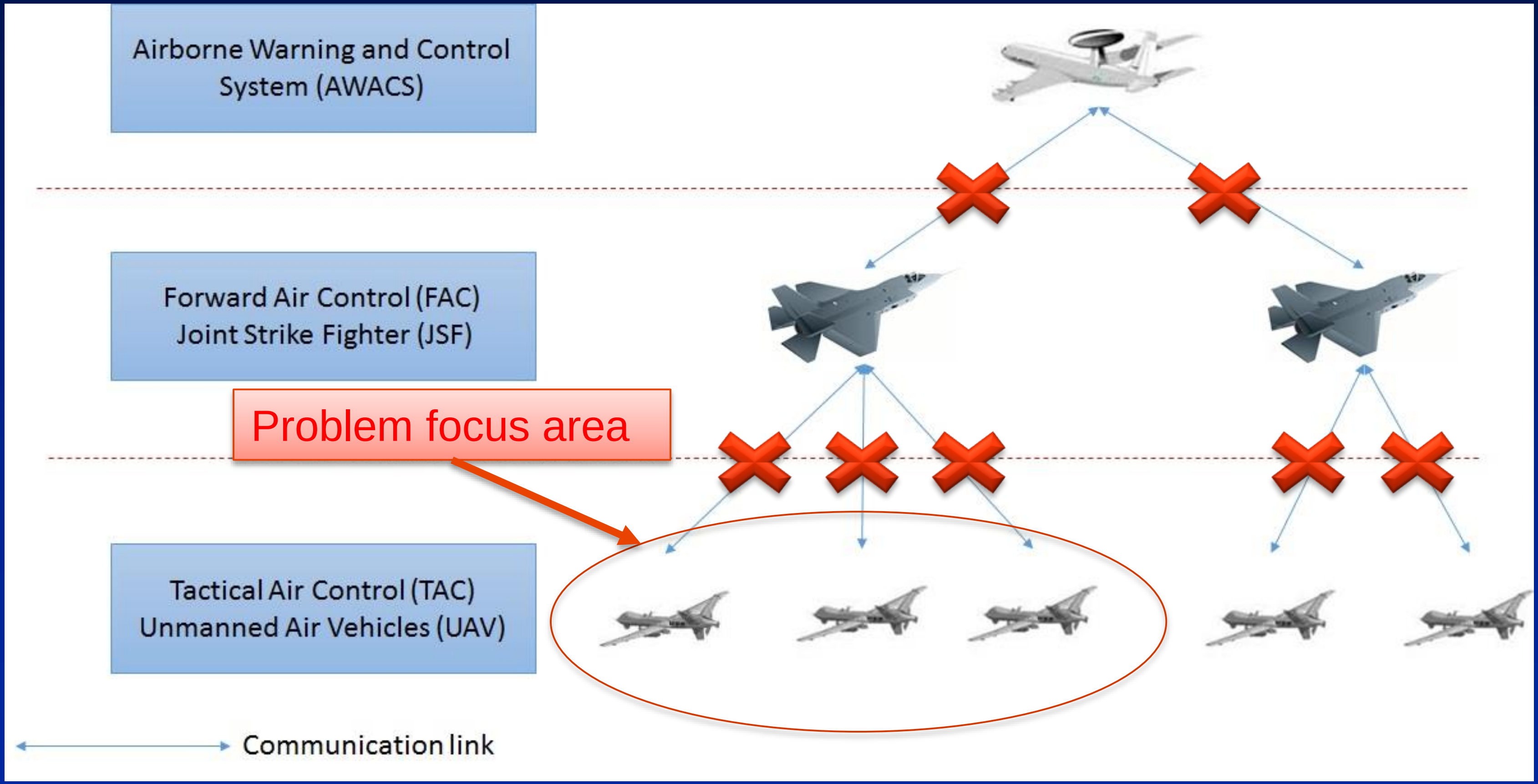


Airborne Warning and Control System (AWACS)
Forward Air Control (FAC) Joint Strike Fighter (JSF)
Tactical Air Control (TAC) Unmanned Air Vehicles (UAV)
Communication link

Background

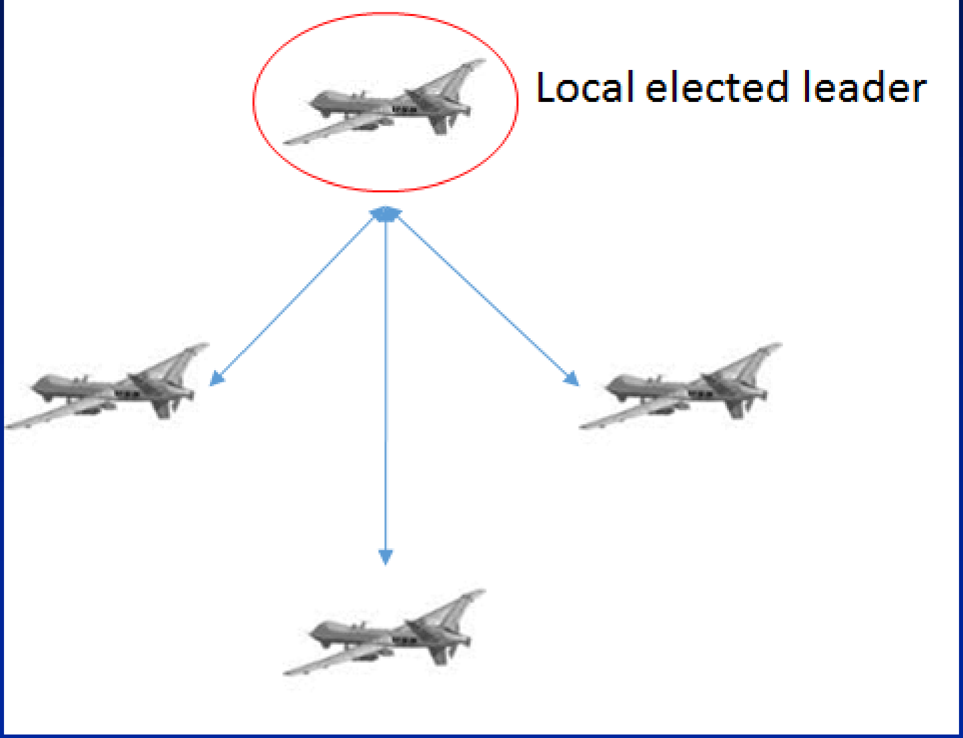

Airborne Warning and Control System (AWACS)
Forward Air Control (FAC) Joint Strike Fighter (JSF)
Tactical Air Control (TAC) Unmanned Air Vehicles (UAV)
Communication link

Problem Focus



UNCLASSIFIED

Problem Focus



UNCLASSIFIED

Problem Statement

What is the “best” way to perform distributed leader election within the lowest tier of the air theatre when tier to tier communications are lost?

Problem Statement

What is the “best” way to perform distributed leader election within the lowest tier of the air theatre when tier to tier communications are lost?

Distributed leader election algorithms

Bully and Ring Algorithms

Distributed Election Algorithm Assumptions

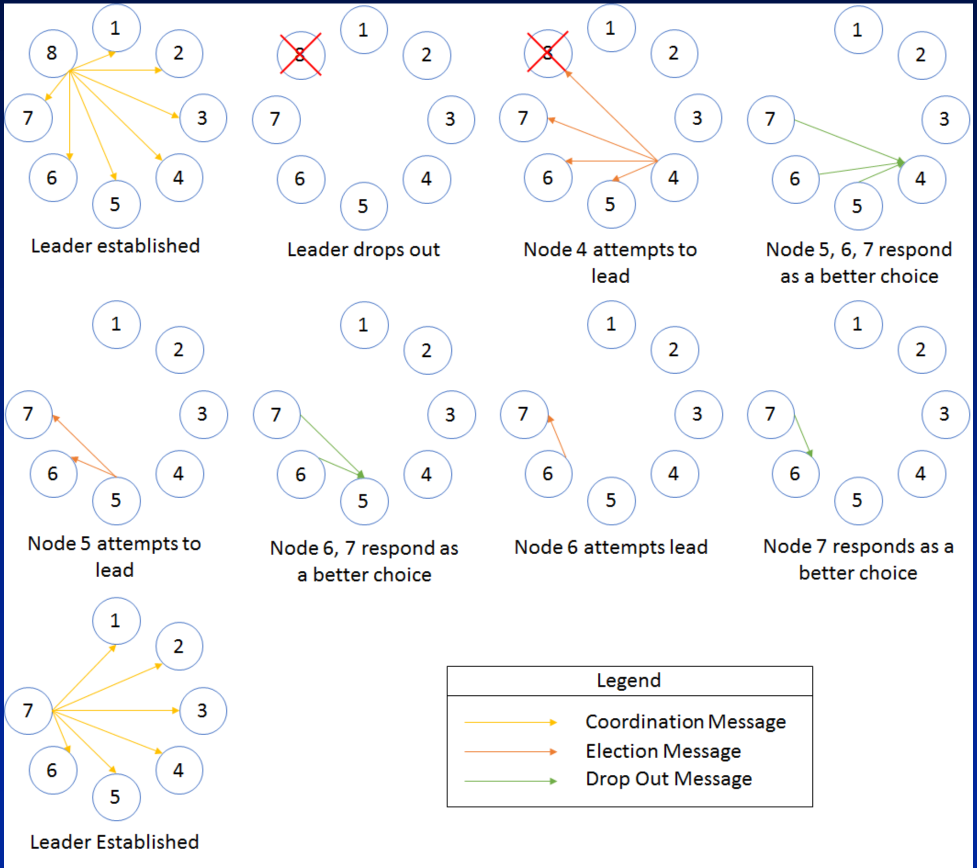
- Distributed
  - No central controller
  - Assets know that other assets exist
  - Assets do not know each others status, online or offline
  - Each asset is assigned a unique identifier
- Cooperative
  - Assets must agree on the election protocols

Distributed Election Algorithm Performance Evaluation

1. **How long does it take to successfully establish a new leader in the event of a lost leader?**

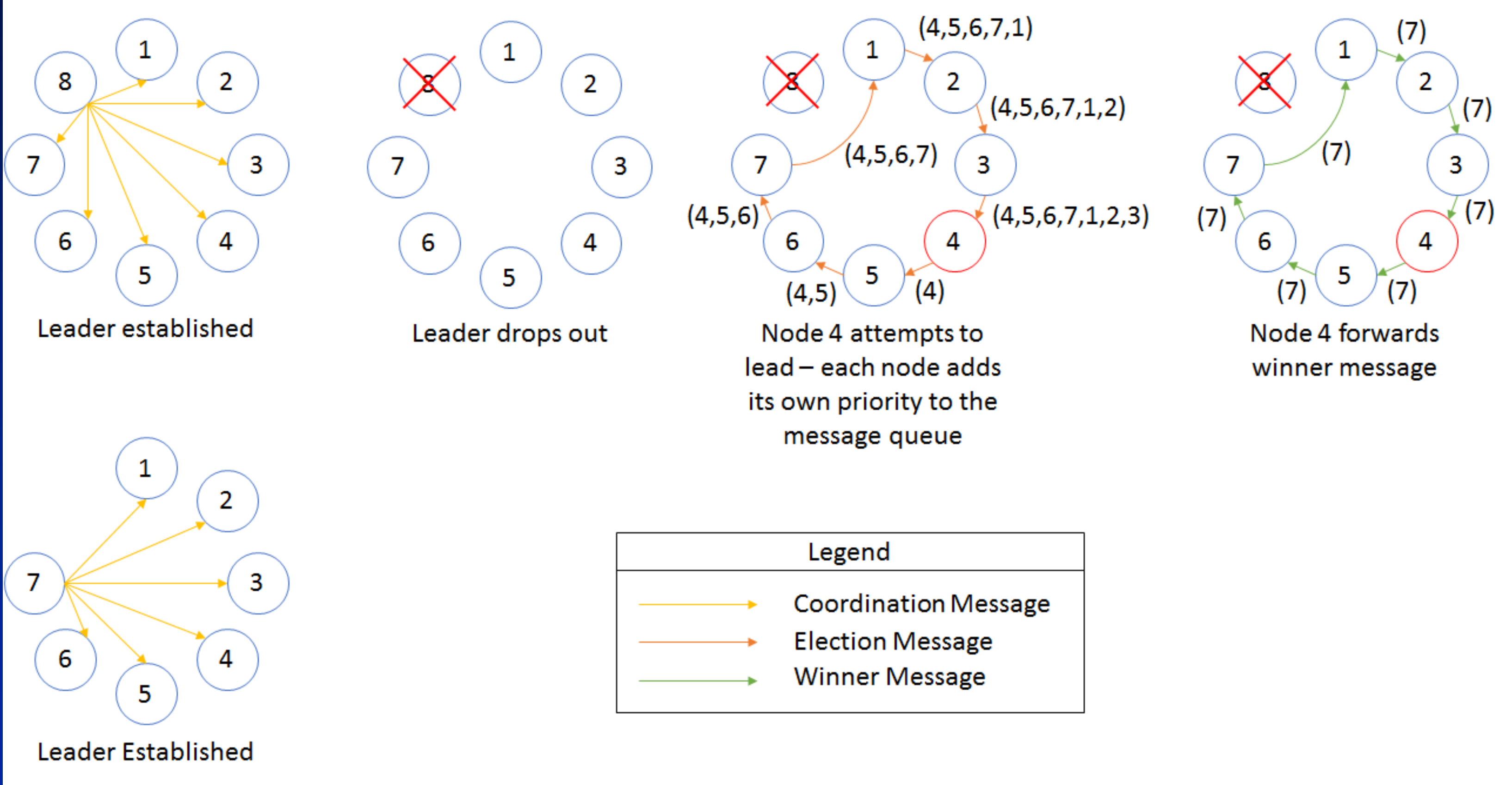
2. **How does network size affect re-election time?**

Bully Algorithm



UNCLASSIFIED

Ring Algorithm



UNCLASSIFIED

Mathematical Evaluation

Bully Algorithm

To develop a mathematical relationship for the Bully algorithm consider the following.

$$n = total\ number\ of\ assets\ in\ the\ network$$

$$r = the\ asset\ that\ is\ starting\ an\ election$$

$$n - r = \#\ of\ neseccary\ election\ messages$$

$$n - r - 1 = \#of\ drop\ out\ messages$$

$$n - 2 = \#\ of\ coordination\ messages$$

$$N(r) = \#\ of\ total\ messages$$

Therefore, to determine how many messages that are needed in an election let:

$$N(r) = (n - r - 1) + (n - r) + (n - 2)$$

Now to evaluate the best case scenario when the highest available assets initiates the election let $r = n - 1$.

$$N(n - 1) = n - 2$$

To evaluate the worst case scenario when the lowest potential assets consecutively attempt election.

$$\left(\sum_{r=1}^{n-1}(n - r - 1) + (n - r)\right) + (n - 2)$$

Ring Algorithm

To develop a mathematical relationship for the Ring algorithm consider the following.

$$n = total\ number\ of\ assets\ in\ the\ network$$

$$n - 2 = \#\ of\ coordination\ messages$$

$$N(r) = \#\ of\ total\ messages$$

The best and worst case scenarios are identical in regards to the Ring algorithm.

$$N(r) = 2 * (n - 1)$$

Mathematical Evaluation

Bully Algorithm

To develop a mathematical relationship for the Bully algorithm consider the following.

$$n = total\ number\ of\ assets\ in\ the\ network$$

$$r = the\ asset\ that\ is\ starting\ an\ election$$

$$n - r = \#\ of\ neseccary\ election\ messages$$

$$n - r - 1 = \#of\ drop\ out\ messages$$

$$n - 2 = \#\ of\ coordination\ messages$$

$$N(r) = \#\ of\ total\ messages$$

Therefore, to determine how many messages that are needed in an election let:

$$N(r) = (n - r - 1) + (n - r) + (n - 2)$$

Now to evaluate the best case scenario when the highest available assets initiates the election let $r = n - 1$.

$$N(n - 1) = n - 2$$

To evaluate the worst case scenario when the lowest potential assets consecutively attempt election.

$$\boxed{\left(\sum_{r=1}^{n-1}(n - r - 1) + (n - r)\right) + (n - 2)}$$

Ring Algorithm

To develop a mathematical relationship for the Ring algorithm consider the following.

$$n = total\ number\ of\ assets\ in\ the\ network$$

$$n - 2 = \#\ of\ coordination\ messages$$

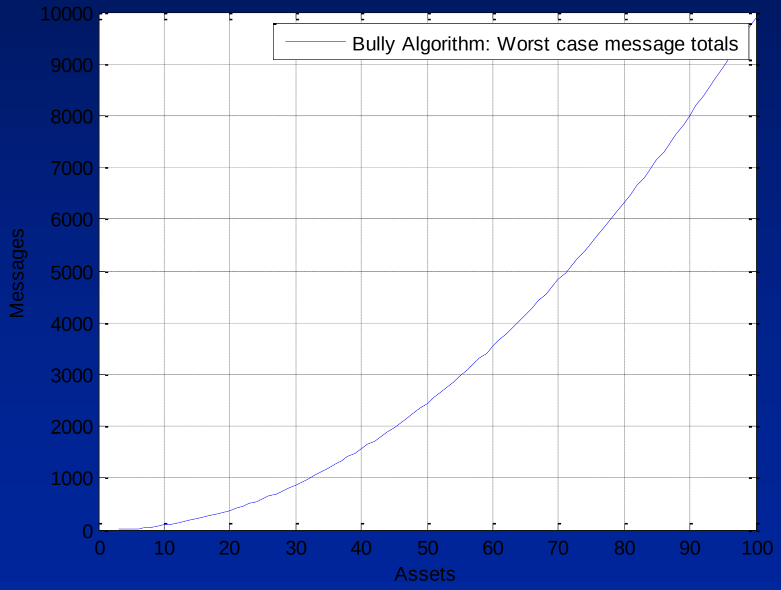
$$N(r) = \#\ of\ total\ messages$$

The best and worst case scenarios are identical in regards to the Ring algorithm.

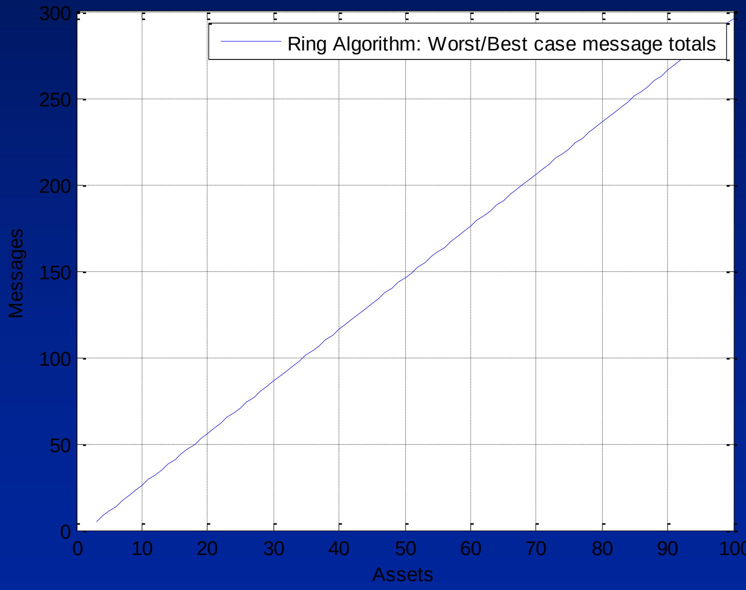
$$\boxed{N(r) = 2 * (n - 1)}$$

Message Passing

Bully – Worst Case



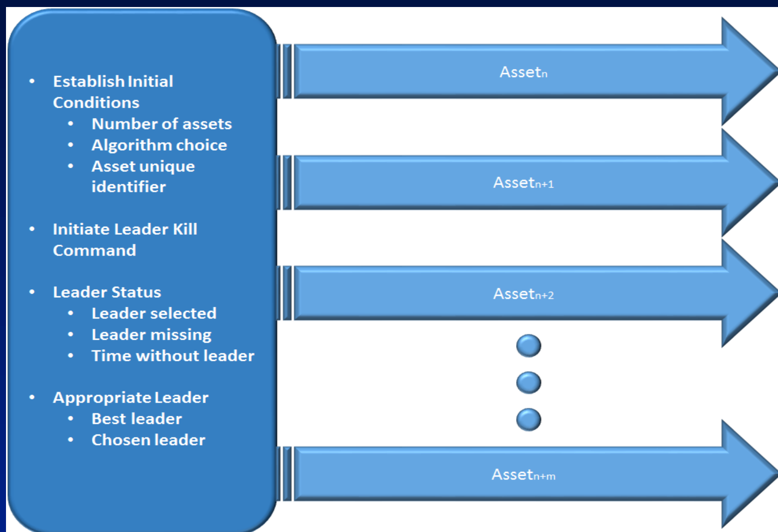
Ring – Best/Worst case



Algorithm Performance - Big O

	Bully Algorithm	Ring Algorithm
Best Case	O(n)	O(n)
Worst Case	O(n^2)	O(n)

Simulation Implementation



- Each assets is started as its own thread using POSIX
- Each asset runs the distributed leader election algorithm via a finite state machine implementation

Implementation Procedures

- Assign the leader, “build” the network, and begin coordination messages.
- A leader kill message will be sent to the leader and a timer will begin.
- The algorithm will autonomously perform leader assignment.
- The moment a new leader is assigned the time is benchmarked and recorded.
- This process will be performed 10 times for each increment of number of assets.
- The process will be performed for 3-25, 50, 75, 100 assets at a time.

Bully Algorithm Pseudo

- **LEADER STATE**
  - Send Coordination Message
  - If received Coordination Message or Election Message or Drop Out Message
    - Error
  - If received KILL Message
    - Terminate the asset

- **ELECTION STATE**
  - Send Election Message to all assets with value > self
    - While (not timed out)
      - If receive Drop Out Message
        - Lost election → go to FOLLOWER STATE
      - If receive Election Message
        - If Election Message < self
          - Send Drop Out Message
        - Else
          - Lost election → go to FOLLOWER STATE
      - Increment timer
  - Won election → go to LEADER STATE

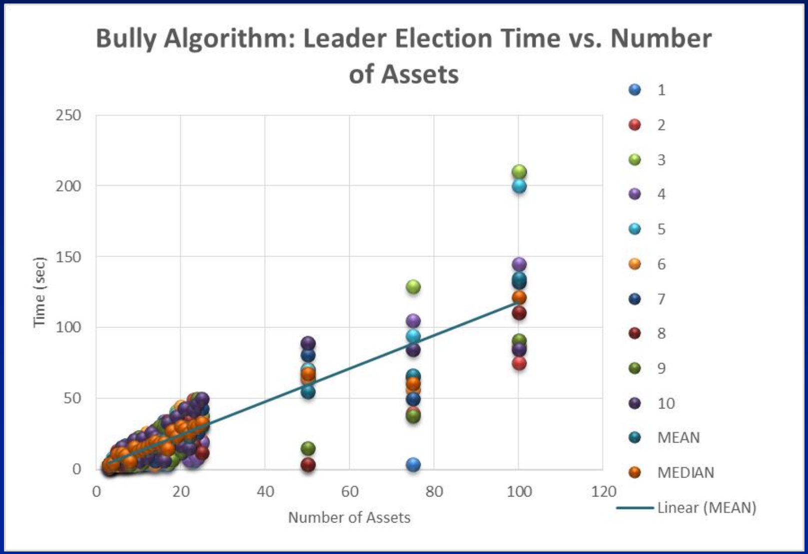
- **FOLLOWER STATE**
  - While (not timed out)
    - If receive Election Message
      - If node sending Election Message > self
        - Send Drop Out Message
        - Start election → go to ELECTION STATE
      - Else
        - Error
    - If receive Coordination Message
      - Restart timer
    - Increment timer
  - Leader disconnect → go to ELECTION STATE

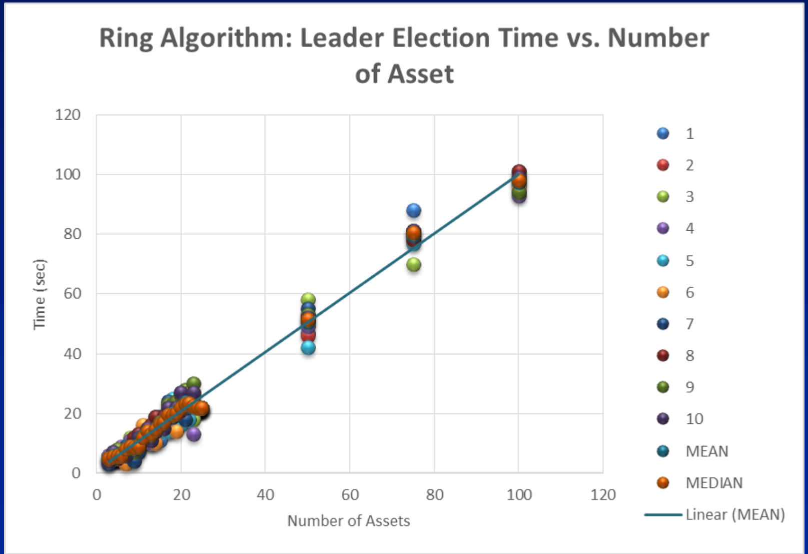
Ring Algorithm Pseudo

- **LEADER STATE**
  - Send Coordination Message
  - If received Coordination Message or Election Message or Winner Message
    - Error
  - If received KILL Message
    - Terminate the asset
- **ELECTION STATE**
  - Send Election Message to next highest asset
    - For (ever)
      - If receive Election Message
        - If Election Message < (initial size of network - # of dropped leaders)
          - Append Election Message
          - Forward Election Message to next highest asset
        - else
          - Set winner = largest value (Election Message)
          - Forward Winner Message
      - If receive Winner Message
        - If Winner Message == self
          - Won election → go to LEADER STATE
        - Else
          - Lost election → go to FOLLOWER STATE
      - Increment timer
- **FOLLOWER STATE**
  - While (not timed out)
    - If receive Election Message
      - Append Election Message with self
      - Forward Election Messages to next highest asset
    - If receive Winner Message
      - If Winner Message == self
        - Won election → go to LEADER STATE
      - Else
        - Lost election → go to FOLLOWER STATE
    - If receive Coordination Message
      - Restart timer
    - Increment timer
  - Leader disconnect → go to ELECTION STATE

Live Demonstration of Algorithm Simulation

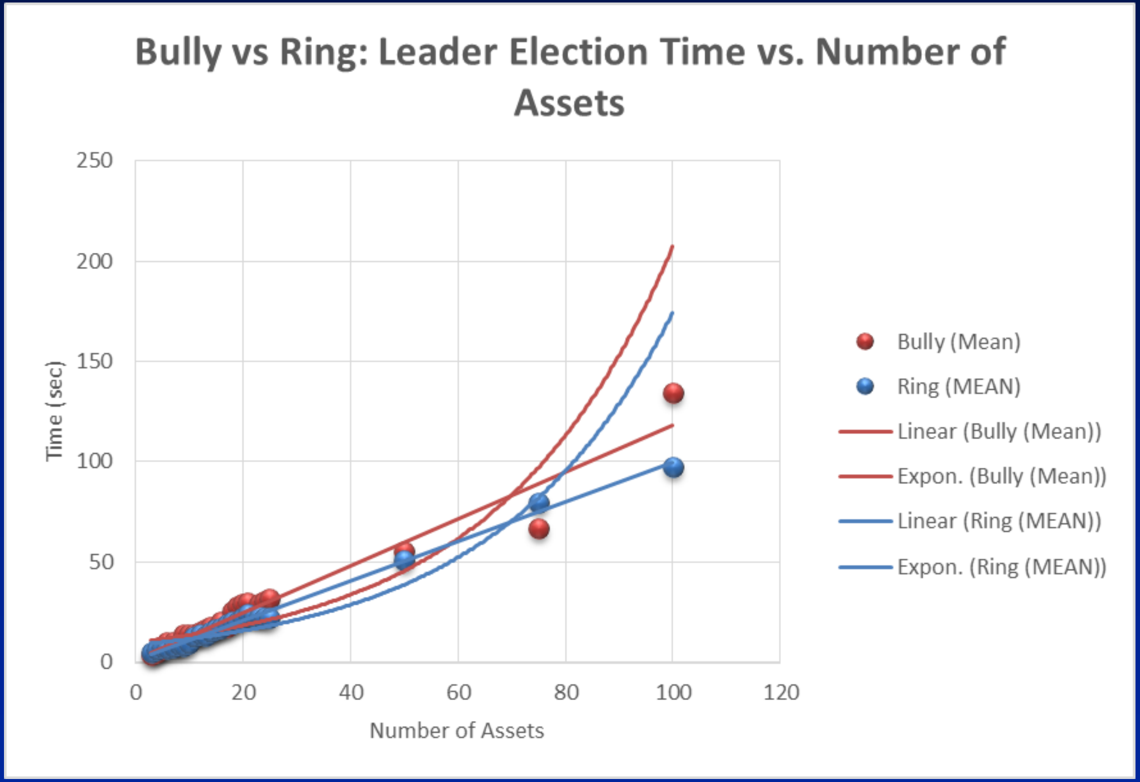
Simulation Results





UNCLASSIFIED

Simulation Results



UNCLASSIFIED

Conclusion

- Mathematical analysis was correct
  - Total messages directly relates to re-election time
- Algorithm performance
  - On average the Bully algorithm performs better than the Ring algorithm in networks smaller than 5 or less assets.
  - On average the Ring algorithm performs better than the Bully algorithm in networks larger than 10 or more assets.

Further Research

- Algorithm Research
  - Algorithm fencing
  - New algorithms
- Distributed Battle Management Research
  - Protocols for asset priority weighting

References

[1] DARPA. (2014). *Broad Agency Announcement Distributed Battle Management* [document]. Available: https://www.fbo.gov/index?s=opportunity&mode=form&id=3639054acc4cb20f5e979c1106075807&tab=core&_cview=1

[2] Garcia-Molina "Elections in a distributed computing system", IEEE transactions on computers, vol C-31, No 1,pp 48-59.,1982.

[3] Fredrickson and Lynch, Fredrickson, and Lynch, "Electing a Leader in a synchronous Ring", journal of the ACM, Vol 34, pp 98-115, 1987.

[4] Hector Garcia-Molina, *Elections in a Distributed Computing System*, IEEE Transactions on Computers, Vol. C-31, No. 1, January (1982) 48-59

Questions?

Backup Slides

Bully Algorithm – Time (sec) vs. Number of Assets

		Number of assets																								
		3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	50	75	100
Trial	1	4	5	11	14	10	5	6	4	13	24	21	19	4	34	4	26	39	40	29	21	6	13	38	89	4	210
	2	4	4	13	9	13	17	6	13	5	5	7	23	24	15	25	28	24	20	20	31	49	39	20	71	40	75
	3	3	4	10	15	4	5	4	20	19	19	8	12	23	22	5	32	41	44	35	8	37	15	38	4	129	210
	4	4	4	11	4	4	4	16	10	19	12	4	24	9	16	14	17	19	33	16	7	11	8	19	64	105	145
	5	2	8	4	4	17	17	20	14	6	5	14	14	22	12	15	34	39	40	28	34	30	26	31	70	94	200
	6	4	6	4	13	13	5	15	5	23	26	16	4	16	14	12	27	21	44	18	33	23	35	34	65	56	87
	7	4	4	11	6	17	6	20	22	16	8	25	6	8	22	26	27	14	12	43	33	44	26	43	81	50	132
	8	2	6	12	4	4	17	21	23	11	22	22	28	14	32	34	29	23	27	21	35	17	46	12	4	65	111
	9	4	4	10	14	10	4	6	23	5	19	19	20	30	31	8	7	22	13	43	16	30	50	32	15	38	91
	10	4	6	6	16	6	13	21	6	22	9	26	23	21	6	33	24	37	17	43	15	42	46	50	89	85	85
MEAN		3.5	5.1	9.2	9.9	9.8	9.3	13.5	14	13.9	14.9	16.2	17.3	17.1	20.4	17.6	25.1	27.9	29	29.6	23.3	28.9	30.4	31.7	55.2	66.6	134.6
MEDIAN		4	4.5	10.5	11	10	5.5	15.5	13.5	14.5	15.5	17.5	19.5	18.5	19	14.5	27	23.5	30	28.5	26	30	30.5	33	67.5	60.5	121.5
MODE		4	4	11	4	4	5	6	23	5	5	#N/A	23	#N/A	22	#N/A	27	39	40	43	33	30	26	38	89	#N/A	210

Ring Algorithm – Time (sec) vs. Number of Assets

		Number of assets																									
		3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	50	75	100
Trial	1	6	6	6	6	9	9	12	11	12	13	14	11	11	16	20	15	20	17	22	23	21	22	22	47	88	97
	2	6	6	5	8	8	7	9	8	11	15	13	15	19	16	15	17	20	20	26	20	20	22	22	46	80	100
	3	6	5	7	7	9	12	4	10	12	15	10	14	14	19	15	18	20	21	27	19	18	22	22	58	70	99
	4	6	5	5	9	7	11	10	8	14	14	10	14	16	17	16	21	17	23	21	21	13	22	22	49	81	93
	5	4	7	6	5	9	8	6	9	12	12	13	19	17	14	14	25	21	16	27	18	22	22	22	42	77	101
	6	5	5	8	6	3	8	11	9	16	14	16	10	15	17	21	19	14	25	24	24	26	22	22	53	78	96
	7	3	5	6	7	9	6	4	7	13	12	11	19	19	19	24	19	24	24	18	24	23	22	22	55	81	94
	8	4	7	5	6	8	9	12	13	11	12	16	19	17	19	19	23	20	21	22	24	24	22	22	51	78	101
	9	5	5	8	6	6	9	7	9	12	14	14	15	17	17	23	23	24	26	28	27	30	22	22	53	81	94
	10	4	7	6	6	6	9	8	12	12	15	11	14	16	15	22	20	22	27	22	24	27	22	22	52	81	99
MEAN		4.9	5.8	6.2	6.6	7.4	8.8	8.3	9.6	12.5	13.6	12.8	15	16.1	16.9	18.9	20	20.2	22	23.7	22.4	22.4	22	22	50.6	79.5	97.4
MEDIAN		5	5.5	6	6	8	9	8.5	9	12	14	13	14.5	16.5	17	19.5	19.5	20	22	23	23.5	22.5	22	22	51.5	80.5	98
MODE		6	5	6	6	9	9	12	9	12	15	14	14	17	19	15	19	20	21	22	24	#N/A	22	22	53	81	99