

Geolocation-based Routing in Wireless Ad-hoc Networks

Bin Zhou and Mostafa Bassiouni

University of central Florida



Outline

- Introduction
- Routing algorithm
- Simulation results
- Conclusions



Goals of Geolocation-based routing

- Routing algorithm's aim: improving the speed and/or reliability of the routing module
- Using positioning information services to improve the reachability of routing algorithms in ad-hoc networks



Related works

- R. Jain et al, 2001, GRA (geographical routing algorithm)
 - Used geographical information to guide the routing process
- Y. Ko, et al 2000 (LAR)
 - Used the location information to improve the performance of ad-hoc routing
- B. Karp, 2000, Greedy Perimeter Stateless Routing (GPSR)
 - Finds nodes progressively closer to the destination



Flooding algorithm

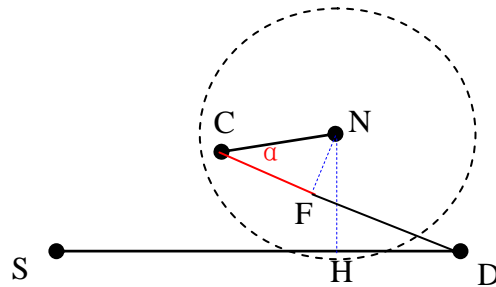
- The routing message is sent to all neighbors and the process is repeated until the destination is reached or the algorithm fails to do so
- The flooding protocol gives the best reachability
- This algorithm is used as our test benchmark



Geolocation-based Model environment

- Assuming every mobile can determine its position information with reasonable accuracy
- Assuming the source node knows the position of the destination node
- Every mobile periodically sends summary information about its status to its neighbors

Some definitions



Direct distance (DD_N)

$$DD_N = |CF| = |CN| \cdot \cos(\alpha) = \frac{|CN| \cdot \overrightarrow{CN} \cdot \overrightarrow{CD}}{|CN| \cdot |CD|} = \frac{\overrightarrow{CN} \cdot \overrightarrow{CD}}{|CD|}$$

Deviation angle (α_N)

$$\alpha_N = a \cos\left(\frac{\overrightarrow{CN} \cdot \overrightarrow{CD}}{|CN| \cdot |CD|}\right)$$



Routing weight (W_N)

- $W_N = DD_N * \delta(UDeg_N) / L + UDeg_N + \text{aweight}(\alpha_N)$

$$\delta(UDeg_N) = \begin{cases} 1, & UDeg_N \geq 1 \\ 0, & \text{otherwise} \end{cases}$$

- Useful degree ($UDeg_N$)
 - Deviation angle (α_N)
 - Direct distance (DD_N)
 - parameter L is tunable and is used to scale the contribution of the direct distance with respect to the useful degree
 - “*aweight*” is the angle weight function and is used to scale the contribution of the deviation angle
- The neighbor (unvisited) with maximum W_N should be chosen as the next hop.

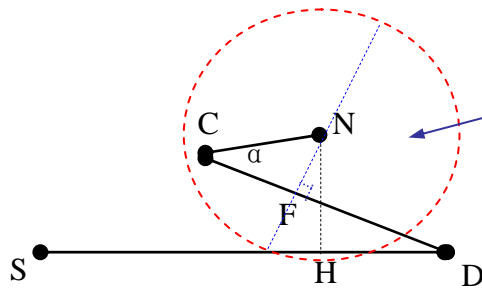


Useful degree

- Method 1 (M1)
 - The number of neighbors of N which have larger distances from C than $|CN|$
- Method 2 (M2):
 - The useful degree of N is half the actual degree of N.
$$UDeg_N = 0.5 * Deg_N$$
 - This method is noted as M2a when `aweight()` is considered

Useful degree (cont')

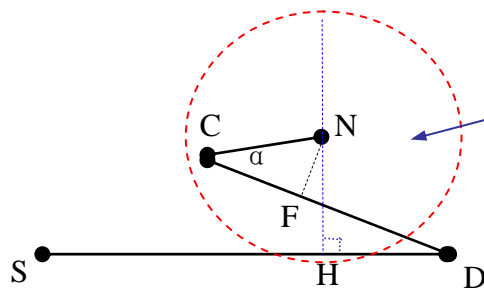
- Method 3 (M3)



Number of neighbors
of N in this half cycle

Useful degree (cont')

- Method 4 (M4)



Number of neighbors
of N in this half cycle

Method 2 is the simplest and the most useful way



Choice of L

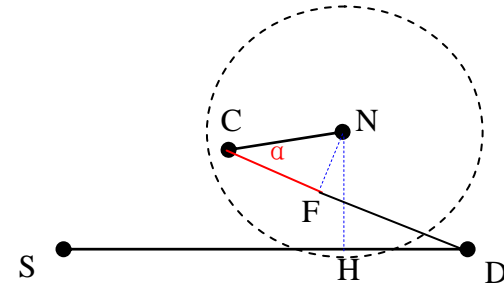
- It is better if the direct distance DD , the useful degree $UDeg$ and the deviation angle have the same effective weight
- For an area A has K mobiles, each mobile has the transmission range R

$$L = \frac{0.8 * A}{K * \pi * R}$$

weight*() value

- Depends on useful degree
- If the average value of UDeg is 3 , for an angle x

$$\text{weight}(x) = \begin{cases} 1 & x > 1.0 \\ 2 & 1.0 \geq x > 0.5 \\ 3 & 0.5 \geq x \end{cases}$$



- In our test report, we investigated the impact of this function. When this is dropped, the L has little impact on the total performance



Simulation Results

$$\text{reachability_difference} = \frac{Diff_L}{Total_L} \times 100\%$$

- $Diff_L$ is the difference between the number of cases of failure of tested algorithm and that of flooding algorithm
- $Total_L$ is the total number of tests for a giving scenario



Simulation results

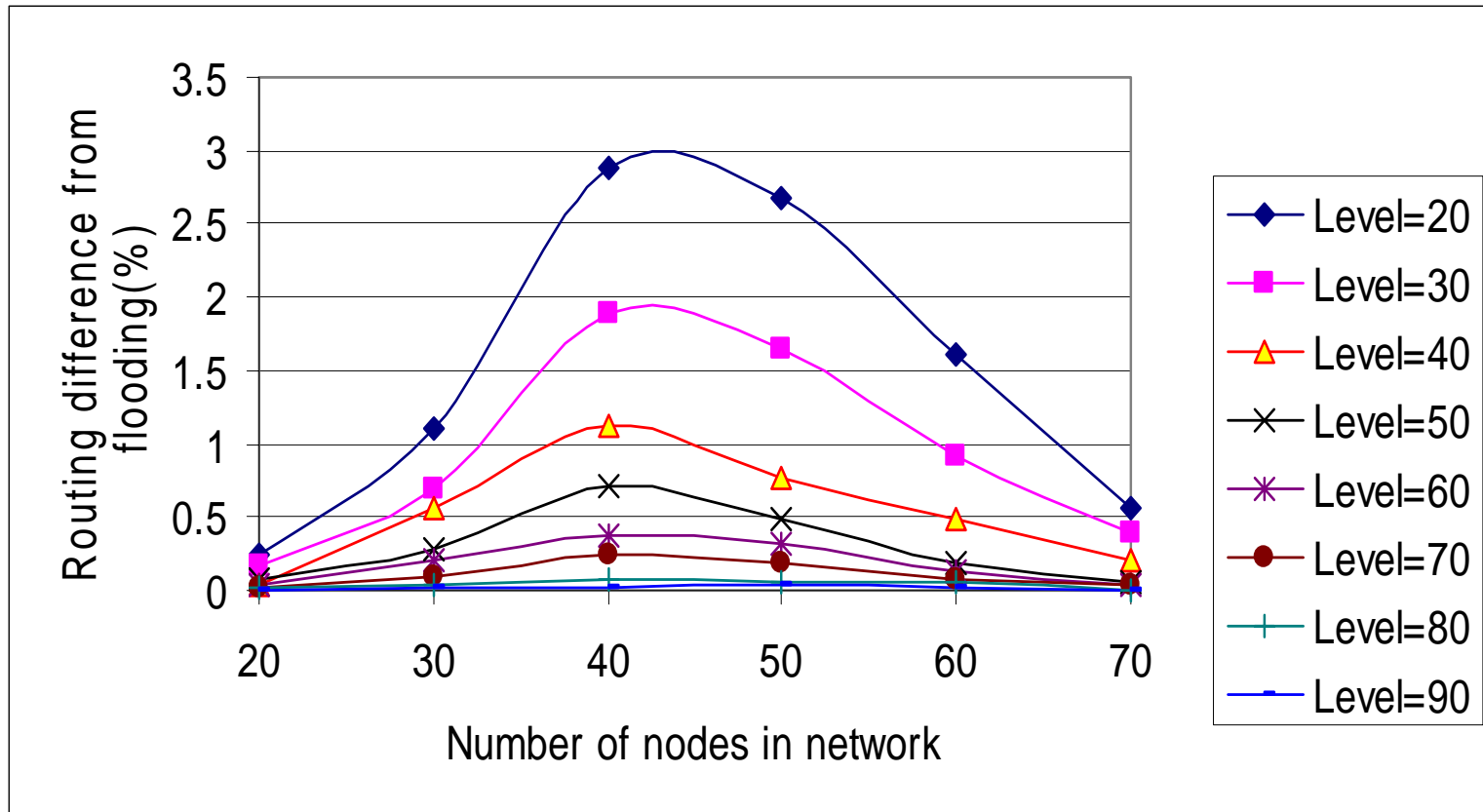
- *Search nodes: number of nodes searched in an algorithm*
- *Error nodes: number of nodes searched in an algorithm, however these nodes do not appear in the final route*



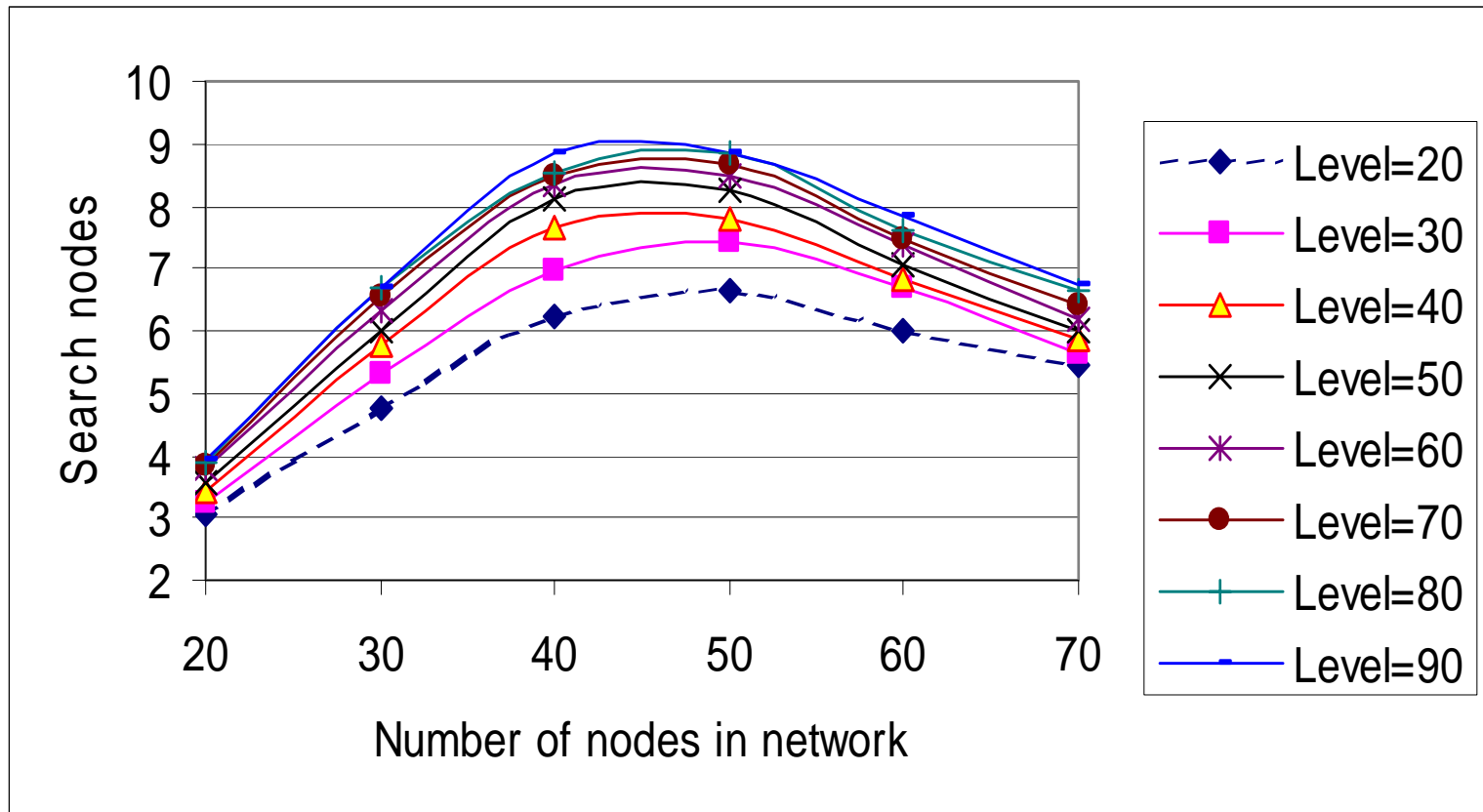
Simulation environments

- Mobiles distribute randomly in an 1000x1000 area
- Number of mobiles ranges 20 to 70
- Transmission range is set to 200, 300, 400, 500
- 10000 randomly distributed scenarios are tested

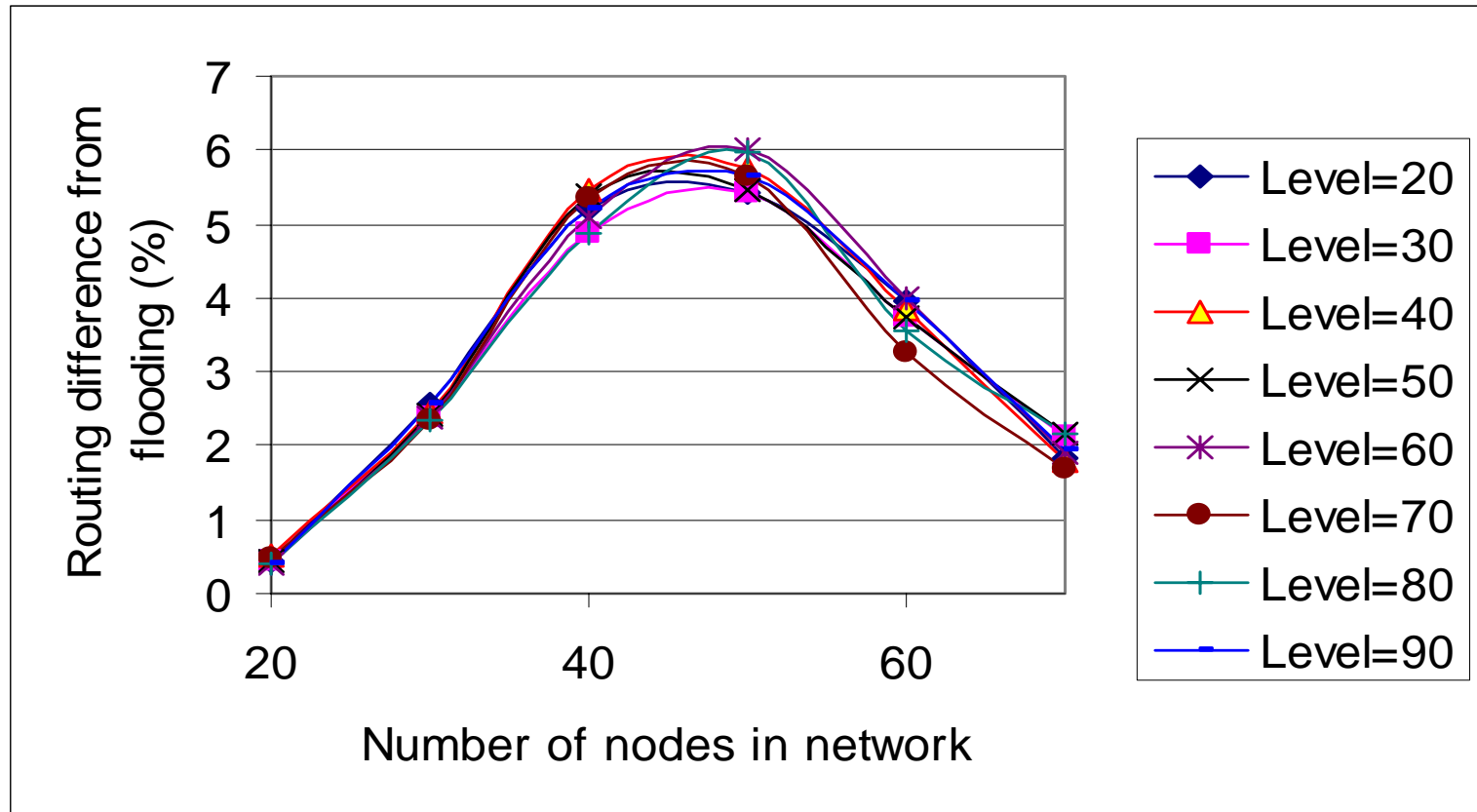
Performance results at different Parameter L values



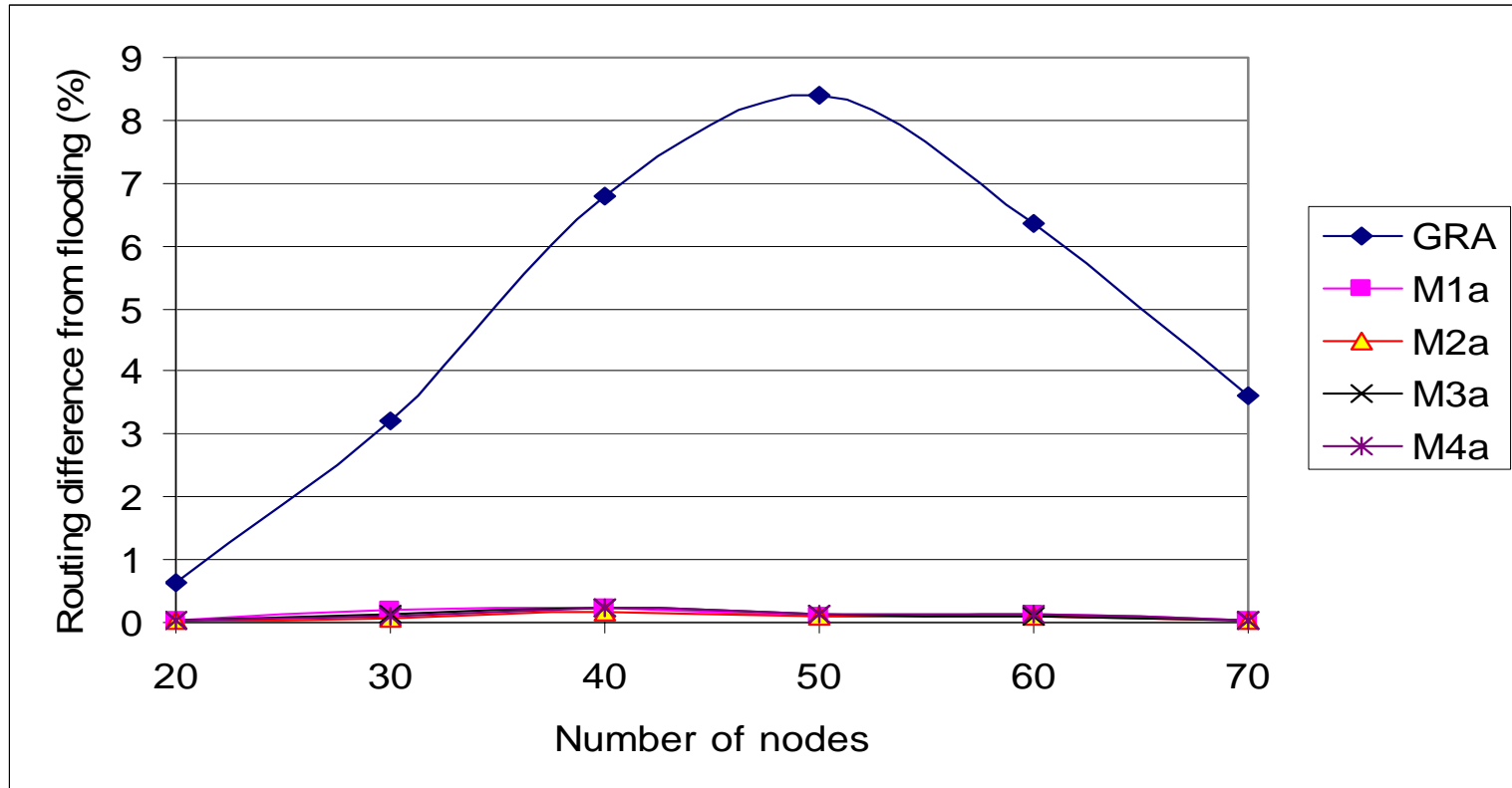
Performance results at different Parameter L values (cont')



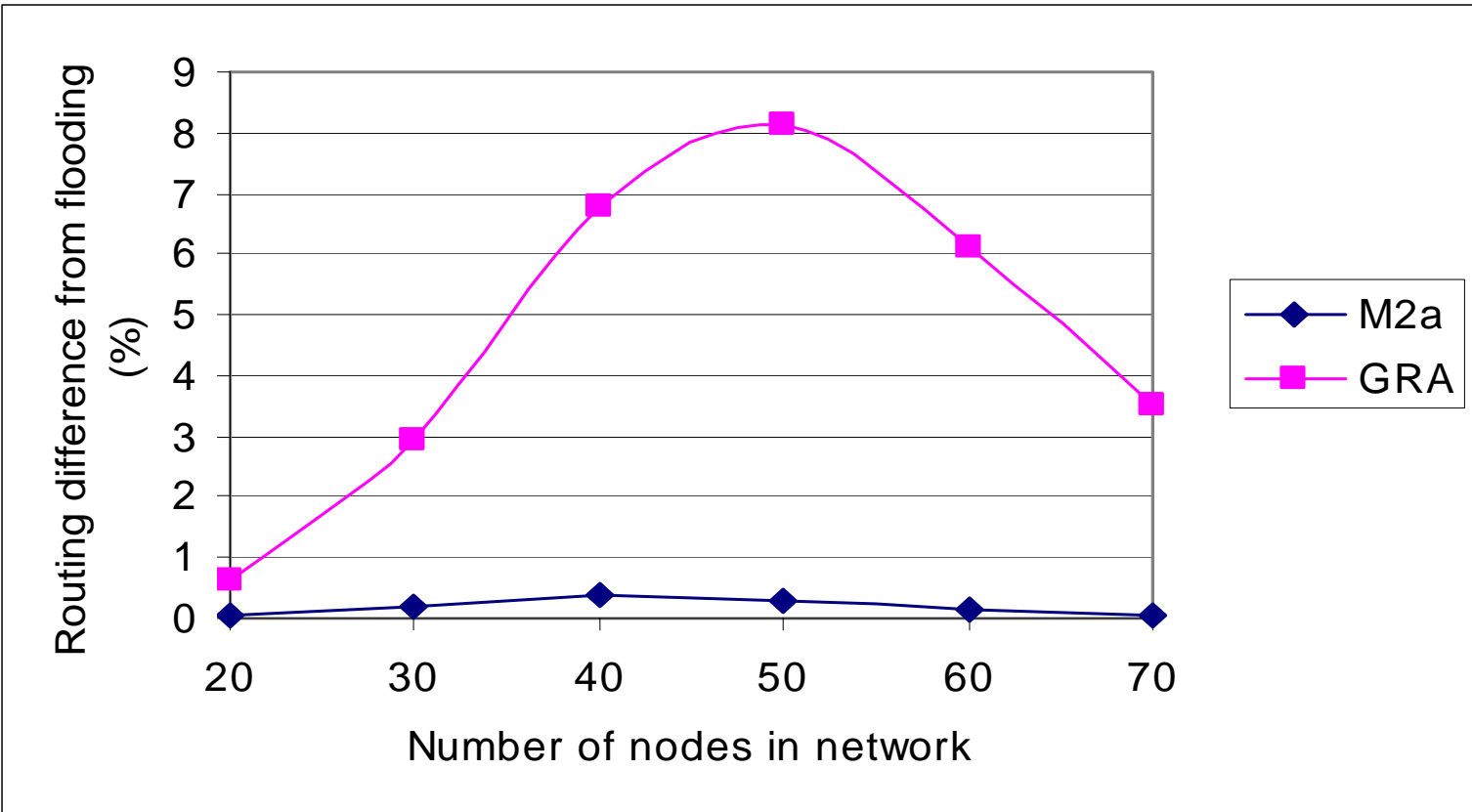
Performance results when $\text{aweight}()$ is not in W_N



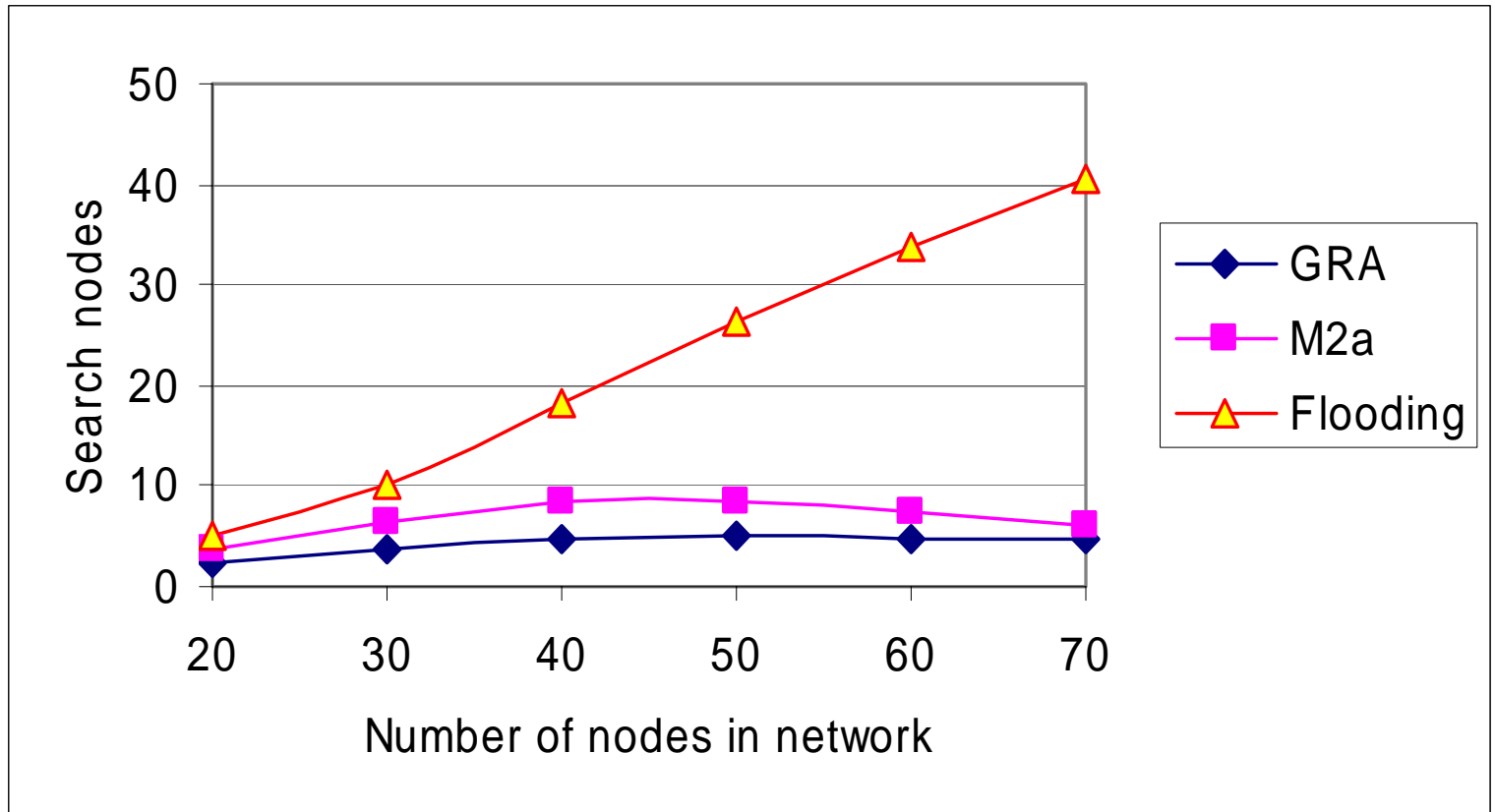
Performance comparison about different UDeqs



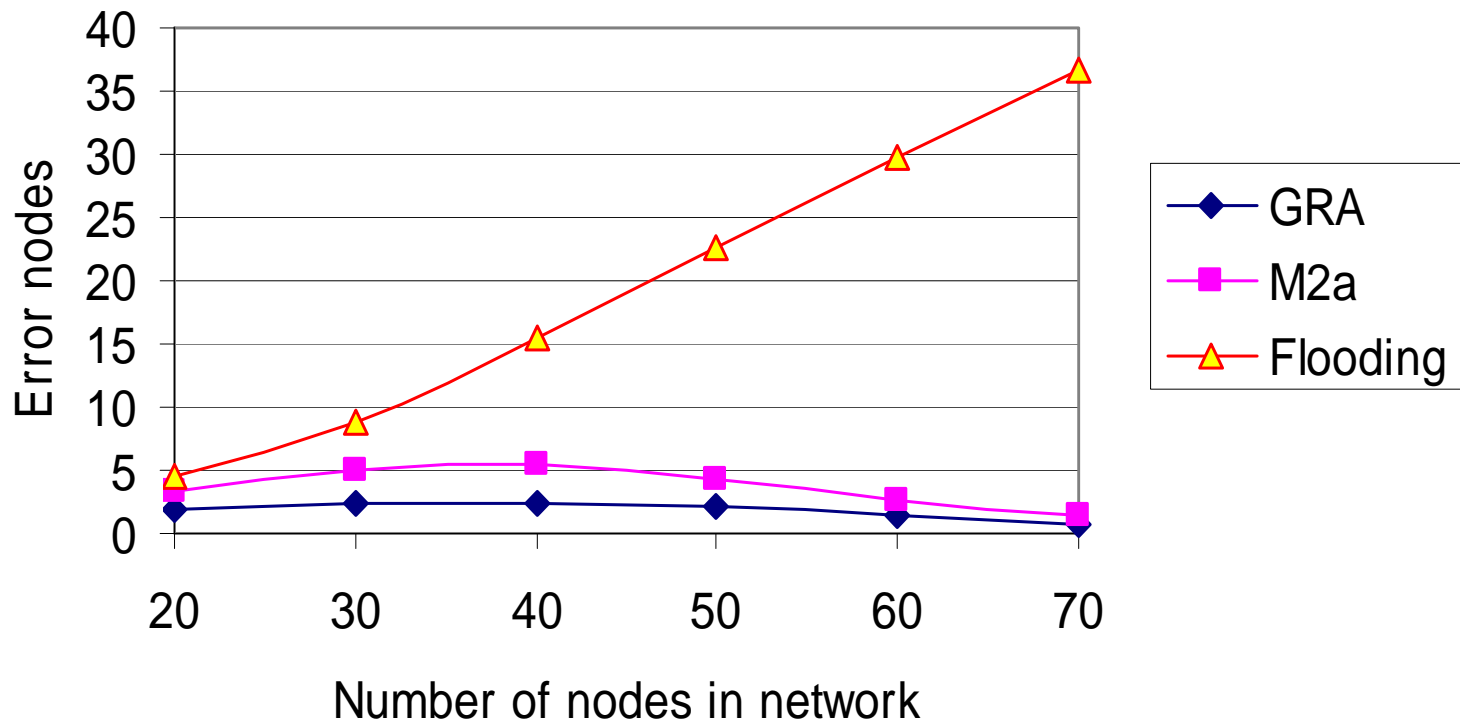
Reachability



Routing search nodes



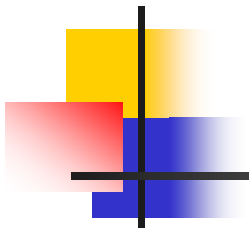
Routing error nodes





Conclusion

- A geolocation-based routing algorithm is proposed
- Waste hops are reduced greatly, small reduction in reachability compared with flooding algorithm
- Reachability is greatly improved compared with GRA, with the cost of slightly increasing the search nodes and error nodes



Thanks!