Robust and lightweight routing with attractor selection

Naotaka Onzuka, Naoki Wakamiya, Masayuki Murata
Osaka University, Japan

Background

- Rapid growth of information networks makes traditional mechanisms unsuccessful and unfeasible
- Traditional routing suffers from increased computational complexity and continuous change

Traditional routing (OSPF)

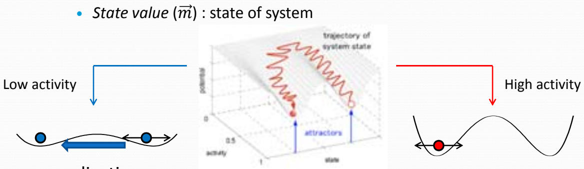
- Shortest path routing with Dijkstra algorithm based on whole network information → Low scalability
- Use of stable metrics → Low adaptability

Ideal routing for future networks

- Autonomous distributed routing → High scalability
- Use dynamic metrics → High adaptability

Learning from biology

- Attractor selection model [3]
  - $\frac{d\vec{m}}{dt} = f(\vec{m}) \times \alpha + \vec{\eta}$ ($f(x)$: potential function, $\vec{\eta}$: Gaussian noise)
    - *Activity* (α) : goodness of current condition
    - *State value* ($\vec{m}$) : state of system



- application
  - routing protocol in mobile ad-hoc networks [5]

[3] A. Kashiwagi, I. Urabe, K. Kaneko, and T. Yomo, "Adaptive response of a gene network to environmental changes by fitness-induced attractor selection," PLoS ONE, vol. 1, p. e49, Dec. 2006.
[5] N. Asvarujanon, K. Leibnitz, N. Wakamiya, and M. Murata, "Robust and adaptive mobile ad hoc routing with attractor selection," in *Proceedings of Adaptive and DependAble Mobile Ubiquitous Systems*, July 2010.

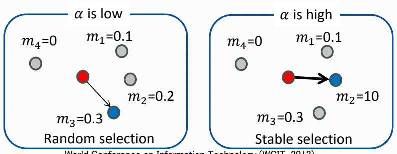
Objective

- Realize adaptive and robust routing in wired networks by adopting attractor selection model
  - Selection of nutrient to synthesis
    -> Selection of next-hop to forward packets

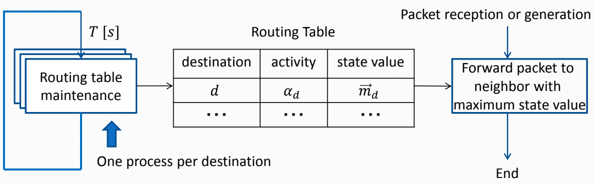
$$\frac{dm_i}{dt} = \frac{\alpha(\beta\alpha^\gamma + \phi^*)}{1 + \max_{1 \le j \le M} m_j^2 - m_i^2} - \alpha m_i + \eta_i$$

α : goodness of path
m_i : goodness of next-hop
η_i : Gaussian noise
M : number of neighbors
β, γ, ϕ^* : constant value

- Hop-by-hop routing



Overview of our routing mechanism

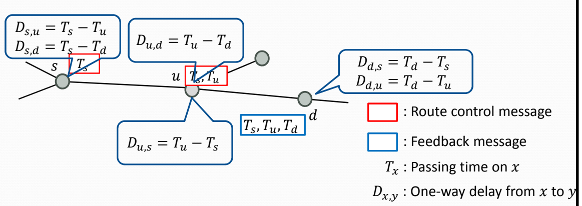


destination	activity	state value
d	α_d	$\vec{m}_d$
・・・	・・・	・・・

- Routing Information for destination d
  - *Activity* (α_d) : goodness of current next-hop selection for destination d
  - *State value* ($\vec{m}_d$) : goodness of neighbors as next-hop for destination d
- Routing table maintenance process
  - Collect path quality information
  - Update routing information based on attractor selection

Collection of path quality information

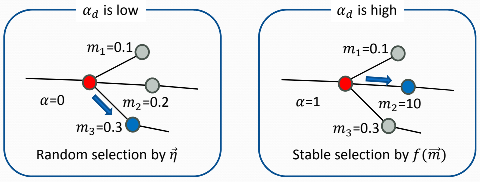
- Measure path quality (one-way delay) to destination d
  - Node sends *route control message* to destination d and d sends back *feedback message*
  - Calculate one-way delay as many as possible by using single pair of control and feedback messages



Updating routing information

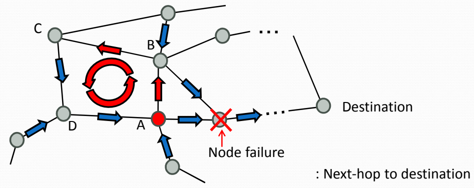
- Routing information is updated based on calculated delay
  - $\alpha = \frac{\text{Minimum Delay}}{\text{Latest Delay}}$
  - $\frac{dm_i}{dt} = \frac{\alpha(\beta\alpha^\gamma+\phi^*)}{1+\max_{1\le j\le M} m_j^2-m_i^2} - \alpha m_i + \eta_i$

α : goodness of path
m_i : goodness of next-hop
η_i : Gaussian noise
M : number of neighbors
β, γ, ϕ^* : constant value



Potential problem of hop-by-hop routing

- Independent selection of next-hop by each node may create looping path



- Loop detection and resolution
  - Check existence of loop and if exits find non-looping path

Simulation settings

- Simulator
  - Omnet++ [7]
- Network model
  - Waxman model [8]
  - Number of nodes : 50~400
  - Number of links : 2 x the number of nodes
- Parameters
  - Proposal
    - Control interval : 100 [s]
  - OSPF
    - Default parameters
    - LSA exchange and Dijkstra computation per T [s]

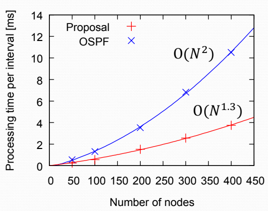
Waxman model network

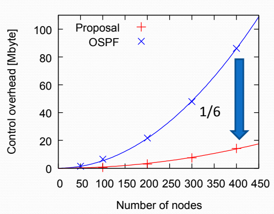
[7] "OMNeT++ Network Simulation Framework", http://www.omnetpp.org/
[8] B. M.Waxman, "Routing of multipoint connections," IEEE Journal of Selected Areas in Communications, vol. 6, pp. 1617-1622, Dec. 1988.

Scenario 1: Evaluation of overhead

- Scenario
  - No change in topology and traffic
  - Number of nodes : 50~400
- Evaluation metrics
  - Processing time
    - Total of realistic processing time in updating routing information during T [s] (Intel Core i7-2600 3.4 [GHz], the memory of 16 [Gbyte])
  - Control overhead
    - Total size of forwarded messages during T [s]
      - Proposal: route control messages and feedback messages
      - OSPF: DD, LSR, LSU, and LSAck messages

Processing time and control overhead



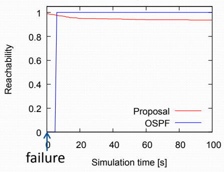


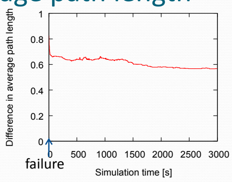
- Processing time
  - OSPF : Proposal = $O(N^2) : O(N^{1.3})$
- Control overhead
  - OSPF : Proposal = 6 : 1
- Higher efficiency and scalability

Scenario 2: Evaluation of robustness

- Scenario
  - Number of nodes : 100
  - Removal of one randomly selected node on one randomly selected path
- Evaluation metrics
  - Reachability
    - Probability that message reaches destination
    - Reachability $P_R(t) = \sum_{k=1}^{S_n}(P_k(t))/S_n$
      - $P_i(t) = \begin{cases} 1, & \text{if source has valid path to destination at } t \text{ in } i\text{th simulation} \\ 0, & \text{otherwise} \end{cases}$
      - S_n : number of simulations
  - Average path length

Reachability and average path length





- Reachability
  - OSPF : Drop to zero right after a failure
    Recovers to 1 after a while
  - Proposal : Nearly one right after a failure
    Slightly decrease by stochastic searches
- Difference in average path length
  - Gradually improved from 0.82 (0 s) to 0.56 (3000 s)

Conclusion and future work

- Conclusion
  - We propose a robust, scalable, and adaptive routing mechanism for wired networks
    - Shorter processing time and lower control overhead than OSPF
  - But, proposal suffers from slightly lower reachability
    - Disadvantage of stochastic behavior of proposal
- Future work
  - Extension to multi-path routing
    - Each node adaptively selects one of alternative paths by using attractor selection model
    - 100% reachability is expected