

Development of Graph Library and Optimization Algorithm for Order/Radix Problem

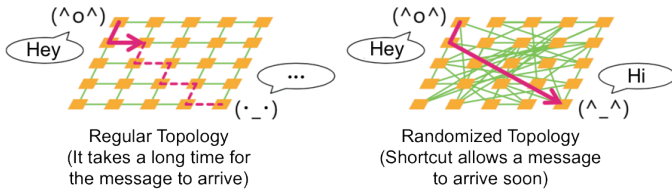


[†]Masahiro Nakao, [‡]Masaki Tsukamoto, [‡]Kosuke Kakizako, [‡]Yoshiko Hanada, [†]Keiji Yamamoto
[†]RIKEN Center for Computational Science, [‡]Kansai University

Order/Radix Problem

Background

- Network in distributed memory system is required to increase its scale and reduce its latency
- A good network topology has a small number of hops between hosts
- When connecting hosts randomly, the number of hops shrinks due to the small world effect, and its latency decreases [Koibuchi2013]



Definition of Order/Radix Problem

- A topology of an indirect network can be represented as an undirected graph by regarding its hosts and switches as vertices and its network cables as edges.
- Order/Radix Problem involves finding a graph with the minimum **diameter** and **h-ASPL** (host to host Average Shortest Path Length) from a set of undirected graphs that satisfy a given number of host **h** and degree **d**.

GraphGolf Competition

The purpose of the competition is to find graphs with the smallest **diameter** and **h-ASPL** in some problems with different **h** and **d** combinations, organized by National Institute of Informatics.

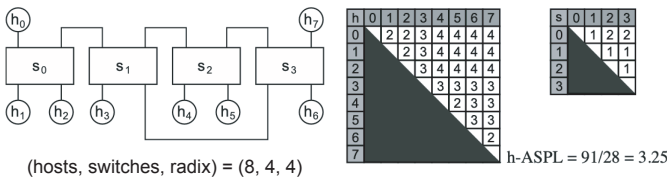


<http://research.nii.ac.jp/graphgolf/>

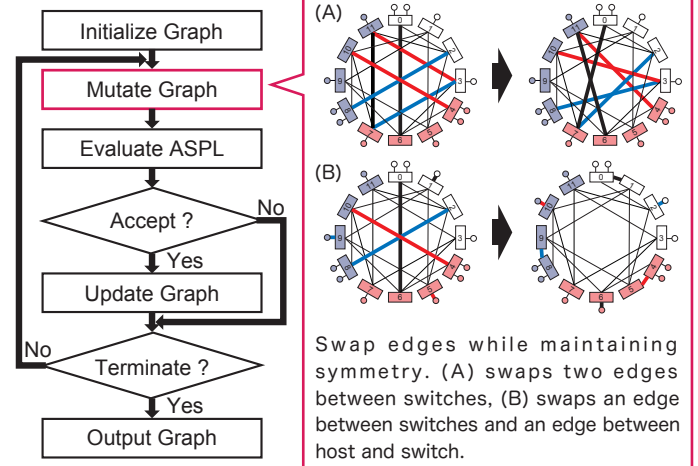
Approach

Overview

- Our algorithm uses Simulated Annealing (SA), its search performance is improved due to graph symmetry
- You can download our optimization algorithms and graph library in <https://github.com/mnakao/ORP>
- The h-ASPL is $(\sum_{0 \leq i < j < s} (l(s_i, s_j) + 2) \cdot w_i w_j + \sum_{0 \leq i < s} w_i (w_i - 1)) / \binom{h}{2}$
The distance between switch s_i and s_j is $l(s_i, s_j)$
The number of hosts adjacent to switch s_i is w_i



Make the network topology **symmetrical**

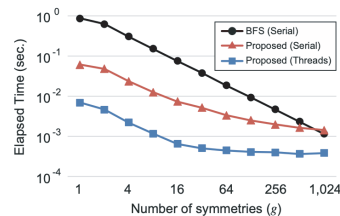


Evaluation

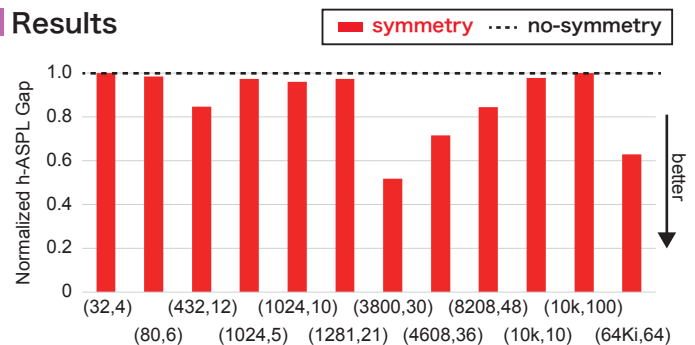
Calculation time on Cygnus

```
#include "orp.h"
...
ORP_Init_aspl(...);
for(int i=0; i<ITERATIONS; i++){
    /* Optimization */
    ORP_Set_aspl(...);
}
ORP_Finalize_aspl();
```

(hosts, switches, radix) = (65536, 4096, 64)

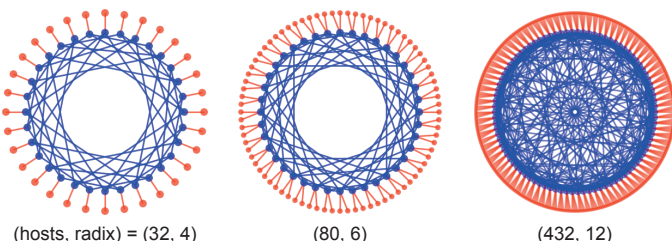


Results



Examples

● host ● switch



GraphGolf Competition

We won the award !!

