A Graph Partitioning-Based Work Distribution Method for Parallel Best-First Search
Yuu Jinnai and Alex Fukunaga (RIKEN, The University of Tokyo)

Background: The state-of-the-art strategy of work distribution for parallel A* is static load balancing: assigning each state to a process by a global hash function. However, there was no quantitative analysis on what kind of hash function is optimal.

Main contribution: We deploy a model of parallel A* to examine the effectiveness of hash functions. Using the model, we propose graph partitioning-based approach for work distribution method. Our experimental results show that our method significantly outperforms previous methods.

Why Parallel Search?
Both time and space are bottleneck of A* search. Both can be addressed by parallel search on distributed environment.

Model of Parallel A* with Static Load Balancing
![Model Diagram]

1. Expand a node owned by the process \( t = t_{\text{proc}} \)
2. Send child nodes to their owner \( t = t_{\text{com}} \)
3. Terminates when all nodes are expanded and sent (to ensure optimality)

- **CO**: number of edges which require communication \( \frac{4}{6} = 0.66 \)
- **LB**: maximum number of nodes owned by a process \( \frac{3}{2.5} = 1.2 \)

- **Communication Efficiency**
  - Assume communication cost for every pair of processors are identical
  - The degradation of walltime efficiency by communication
    \[
    \text{eff}_c := \frac{1}{c \cdot \text{CO}} \quad \text{where} \quad c := \frac{t_{\text{com}}}{t_{\text{proc}}}
    \]

- **Search Efficiency**
  - The ratio of the increase of the number of nodes expanded compared to sequential search
  - The degradation of walltime efficiency by search overhead
    \[
    \text{eff}_s := \frac{1}{1 + \text{SO}} \quad \text{where} \quad \text{SO} = p(LB - 1)
    \]

- **Model Efficiency**
  - Assume communication and search overheads are the dominant overhead
    \[
    \text{eff}_{\text{esti}} := \text{eff}_c \cdot \text{eff}_s = \frac{1}{1 + c \cdot \text{CO}} \cdot \frac{1}{1 + p(LB - 1)}
    \]

Using CO and LB we can model the walltime efficiency of the Parallel A* on the graph.

- **Experimental Results**
  - Evaluated on a 48 core cluster with 6 hashing functions
  - merge&shrink heuristic

- **Comparison of Model Efficiency**
  - Evaluated on a 48 core cluster with 6 hashing functions

- **Summary**
  - Background: The state-of-the-art strategy of work distribution for parallel A* is static load balancing: assigning each state to a process by a global hash function. However, there was no quantitative analysis on what kind of hash function is optimal.
  - Main contribution: We deploy a model of parallel A* to examine the effectiveness of hash functions. Using the model, we propose graph partitioning-based approach for work distribution method. Our experimental results show that our method significantly outperforms previous methods.