

Data Sharing Pattern Aware Scheduling on Grids

Yong Choon Lee and Albert Y.Zomaya (ICPP '06)

Bag-of-Tasks Application

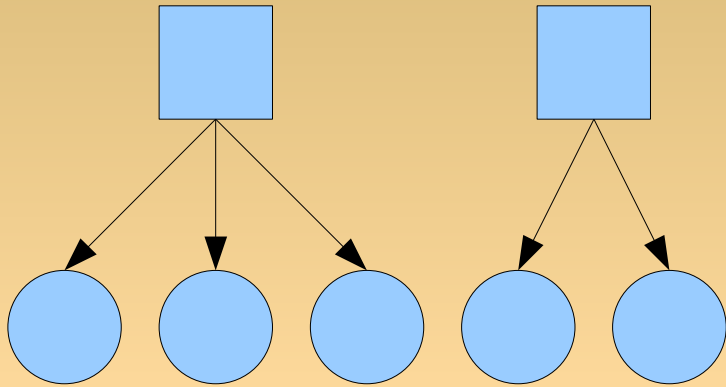
- independent tasks
- no specific order of task execution

Bag-of-Tasks Application

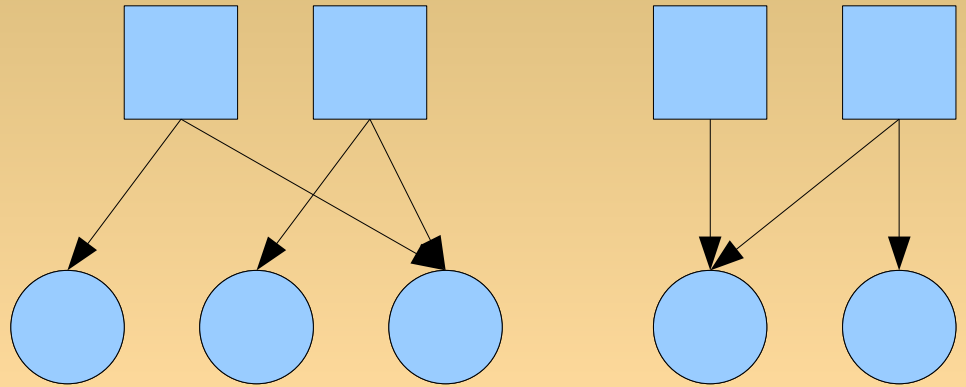
- compute-intensive
- data-intensive

Data Sharing Pattern of DBoT

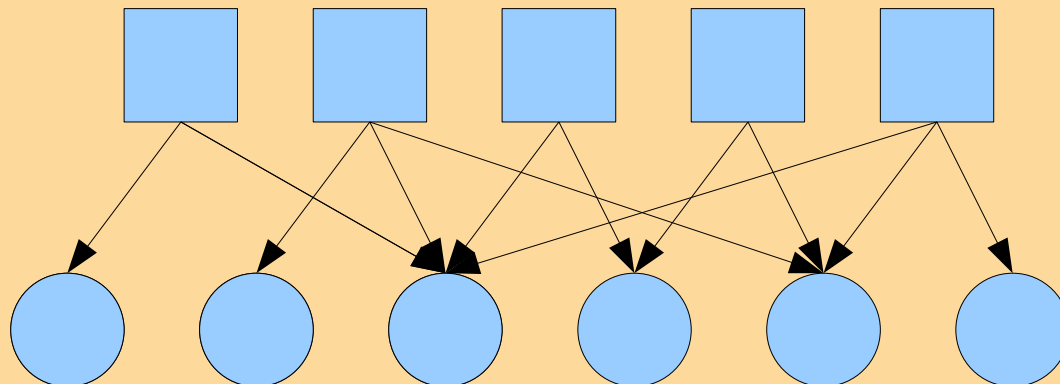
One-Many



Disjointed



Random



Grid Model

- $G = \{S_1, S_2, \dots, S_r\}$
- $S_i = \{H_{i,1}, H_{i,2}, \dots, H_{i,m}\} \cup D_i$
- S_i is i th site participating in G
- H_i and D_i are set of host machines and data repository/storage at S_i

Application Model

- Application J
- n heterogeneous independent task $\{T_1, T_2, \dots, T_n\}$ without inter-task communications and dependencies
- set I_i of input data objects $\{I_{i,1}, I_{i,2}, \dots, I_{i,d}\}$

Scheduling Algorithms

- Storage Affinity (SA)
- List scheduling with Round-robin order Replication (RR)
- The Shared Input data based Listing (SIL) Algorithm

Storage Affinity (SA)

- base on 'Storage affinity metric'
- amount of the task's input data already stored in the site
- dispatches the task with the largest storage affinity value
- If none of the task has positive value one of them is scheduled random

List scheduling with Round-robin order Replication (RR)

- first randomly assigns a task to each host
- on the completion of a task next unscheduled task is dispatched to the host
- once all the tasks are dispatched RR starts replicating running tasks hoping that some or all of these replicas finish earlier than their originals

The SIL Algorithm

- Grouping phase
- Scheduling phase