Distributed quota handling in the Grid by a diffusive based quota management protocol

Adrián Tóth, Ph.D. student

My tutor: dr. Péter Stefán, NIIF Institute
Contents

1. Quota management in the Grid
2. Diffusive load balancing
3. Decentralized diffusion-based quota protocol
4. Implemented test system
5. Conclusions, future work
Quota management in the Grid

- Grid interconnects several different kinds of resources
  - can be spanning over different countries or continents
  - provided by different institutions
  - shared between users
    - collaborating for solving large scale problems
    - grouped into virtual organizations
Quota management in the Grid

- The number of resources can be huge, but they are still limited!

- their usage has to be managed and restricted
  - Security: prohibit malicious use
  - Fail safe: prevent resource leaks (bugs)
  - Financial: moderate use per paid share

- Solution: Quotas
Quota management in the Grid: Resource type, resource demands

- Two main types of resources:
  - Refundable – can be reused (like memory, disk space, bandwidth, etc.)
  - Non-refundable – can be consumed only once (like CPU cycles, transfer volume, etc.)

- Resource demands can be:
  - Priory – resource accounting and quota enforcement can be done by a Grid broker (very rare)
  - Dynamic – handled by a separate quota management service
Quota management in the Grid: Central quota management

- Central quota manager allocates resources to applications via resource leases:
  - Quota manager issues fractions of quota to the clients
  - Client can acquire resources up to the quota

- Has several drawbacks
  - single point of failure
  - scalability (can be bottleneck in large-scale Grids)
  - lease time problem
    - small → frequent requests for additional resources
    - high → unused resource can remain (blocking other application)
Quota management in the Grid: Decentralised quota management

- No central quota service
  - scalability
  - adoptable for resource fluctuation

- Distributed quota information
  - nodes have local knowledge of free quotas
    - fast resource allocation/deallocation
    - improves fault-tolerance
Diffusive Load Balancing

- Load balancing for distributed multiprocessor systems (Cybenko, 1998)
- Nodes are organized in small overlapping groups
- If there is an imbalance, load is migrated
  - $A$, $B$ represent nodes
  - $w_A$, $w_B$ denote the current load of nodes $A$ and $B$
  - If $w_A < w_B$ then move load $(w_A - w_B) / 2$ from $B$ to $A$ and vice versa if $w_A > w_B$
Diffusive Quota Balancing

- Instead of load we balance the free quota
- Free quota is distributed and balanced among all nodes
  - fast resource allocation
  - resource demand may be fulfilled locally
Decentralised diffusion-based quota protocol

- The original protocol was designed by Hans P. Reiser and Rüdiger Kapitza (University of Ulm and Erlangen)
  - Handles refundable and non-refundable resource quotas
  - Paying attention on dynamic resource changes of the node set
  - Has extensions for node failures
  - Managing and controlling the resource usage is not addressed by the protocol (like the actions in case of quota exceeding)
Decentralised diffusion-based quota protocol

Quota Manager:
- Initial node
- Injection point of initial quota
- Not a central component

Initial set:
- Connected to the quota manager
- Connected to some other nodes of the initial set
- Forming an unstructured network

Additional nodes:
- Connect to a random subset of the nodes
Decentralised diffusion-based quota protocol

- Each node stores its local free quota
- Global quota is injected into the quota manager
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated without losing quota information
- Solution: link counter
Implemented test system

- **Quotas**
  - CPU cycles (consumable)
  - Bogo MIPS (refundable)
  - Using impact factor for quotas
    - for preventing frequenting quota exceeding
    - it counts an exponential moving average of the resource usage
    - it is slowing down and speeding up the applications depending on the impact value
Implemented test system

○ Test infrastructure
  • Grid middleware: Grid Underground (NIIF, Hungary)
  • 1 resource domain
    ○ contains ~ 10 nodes
    ○ linux based resources
  • simple resource usage control and management
    ○ with process signals and nice levels

○ Test application: FEM 2.5D
  • computing intensive geophysical application
  • it investigates the frequency domain EM responses of the under surface inhomogeneities
Conclusions, future work

- The decentralized diffusion-based quota protocol is scalable and fault-tolerant → we can use in Grid systems also
- The further improvements of the quota protocol is under investigation:
  - how can we apply to hierarchical, multi-domain controlled networks (nowadays Grid systems)
  - re-implementing to large-scale Grid middlewares (like knowARC, EGEe)
  - adopting statistical based diffusion load (quota) balancing techniques that are designed for heterogeneous systems
Conclusions, future work

- Investigate sandboxing techniques for resource usage control and management
  - Application sandboxing (like Novell’s AppArmor)
  - Virtualization, para-virtualization techniques

- Protocol usage in different kind of computing or storage networks
  - P2P file-sharing
  - BOINC based computing networks
Thank You for your attention!
Questions?