A Super-Peer Model for Building Resource Discovery Services in Grids: Design and Simulation Analysis

Carlo Mastroianni (ICAR-CNR , Italy)
mastroianni@icar.cnr.it

Domenico Talia (DEIS, Univ. Of Calabria , Italy)
talia@deis.unical.it

Oreste Verta (DEIS, Univ. Of Calabria , Italy)
verta@deis.unical.it

Outline

- Convergence between Grid and P2P models
  - Use of P2P techniques to manage basic Grid services
- A Super-Peer model for Grids
  - Proposal of a resource discovery protocol
  - Implementation in the OGSA framework
- Performance evaluation (with simulation) in
  - Grid networks having a fixed overall size and a variable cluster size
  - Grid networks having a fixed cluster size and a variable overall size

Convergence between Grids and P2P

- **Grids** currently adopt a centralized/hierarchical approach for node and resource management (e.g. the Globus Toolkit)
  - this approach can be ineffective in highly dynamic, large-scale Grids
- **P2P systems** adopt a decentralized approach which is more flexible and scalable in large size and dynamic networks

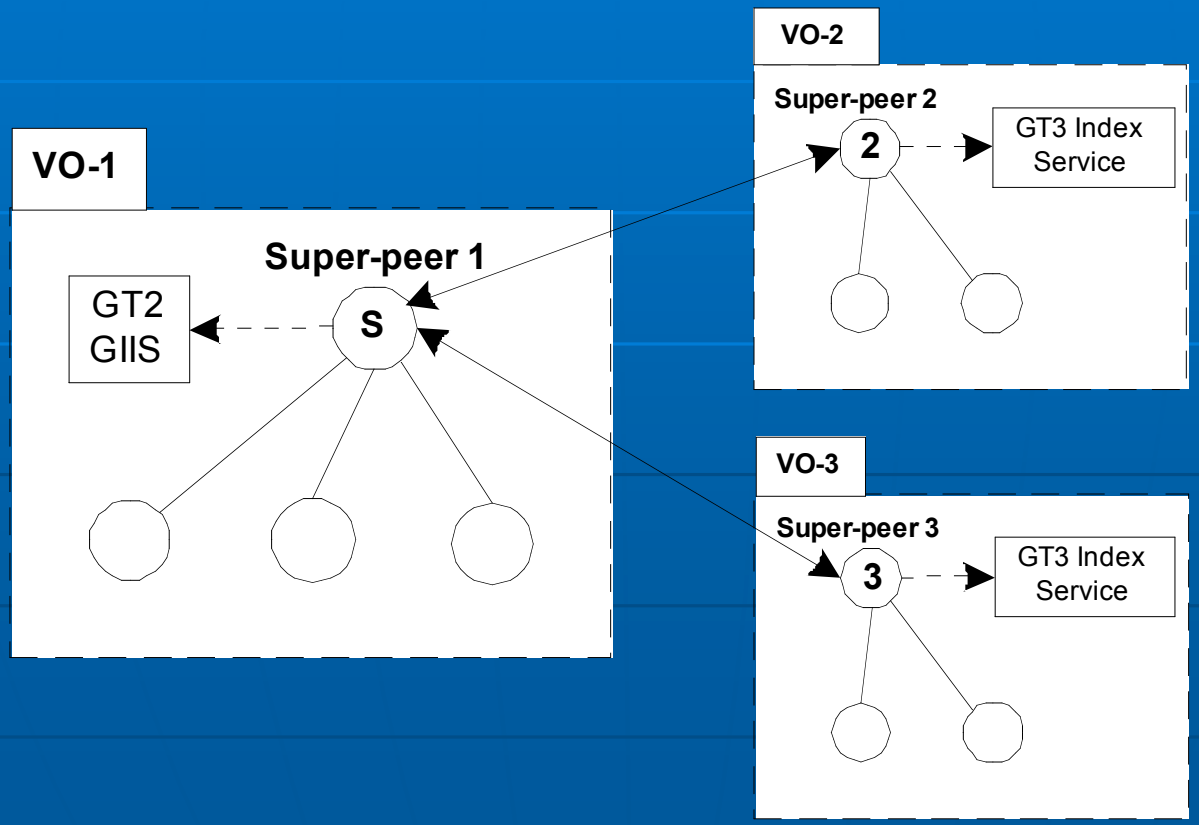
- The super-peer model achieves a balance between:
  - the inherent efficiency of centralized/hierarchical search, and
  - the autonomy, load balancing and fault-tolerant features offered by distributed (P2P) search.

The super-peer model

- The super-peer model is well-suited for large-scale Grids
  - a Grid can be viewed as a network composed of small-scale, proprietary Grids, called Virtual Organizations (VOs).
  - a super-peer acts as a centralized server for the peers of a VO and manages their resources.
  - super-peers connect to each other to form a P2P network at a higher level.
- The super-peer model can be used in Grid networks to implement:
  - the **membership management service**, used to add a node to the network and assign a set of neighbours to it
  - the **resource discovery service**, used to search Grid resources having a given set of properties

The super-peer model in Grids

- The super-peer model exploits the information services provided by the Grid infrastructures of the local VOs
  - e.g. the MDS-2 service of GT2 or the Index Service of GT3



Resource Discovery Protocol

Query messages generated by a Grid node are forwarded to the local super-peer, that:

- searches the local VO information service for the requested resources
- forwards a copy of the query to a selected number of neighbour super-peers, which in turn contact the respective information systems and so on

Whenever resources are found in a remote VO

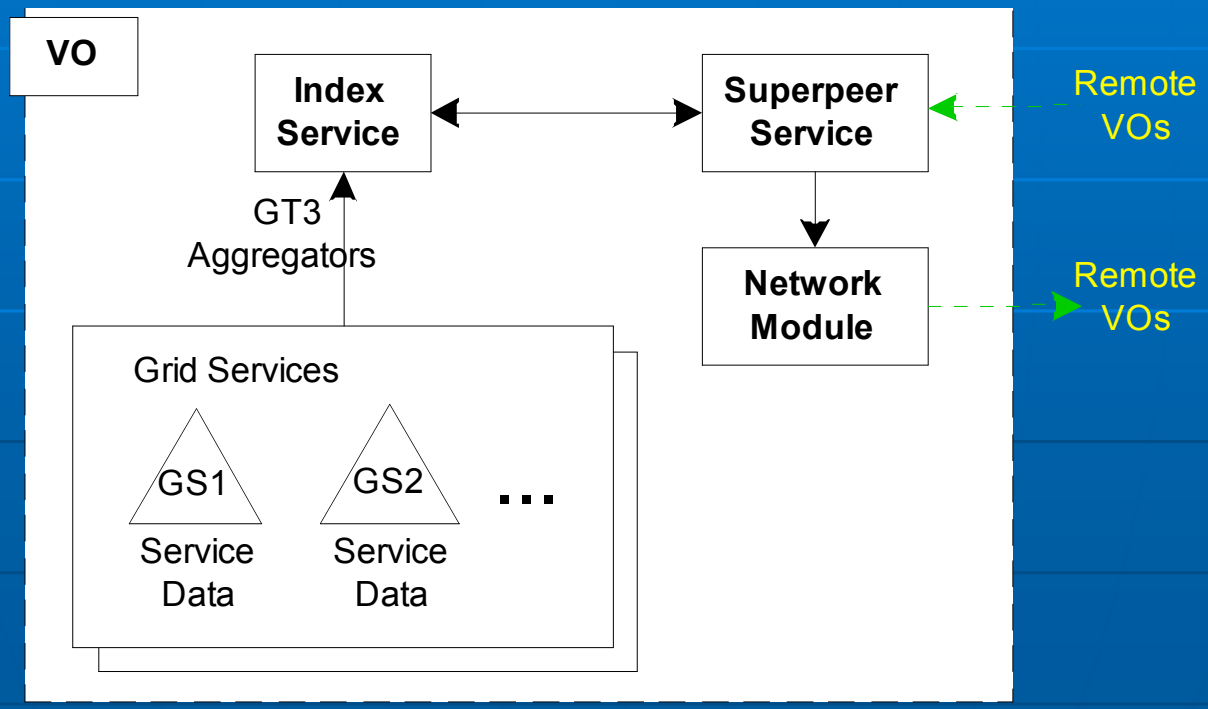
- a queryHit, containing the IDs of the nodes which possess the resourceis, is sent back to the requesting node
- a notification message is sent by the remote super-peer to the nodes that handle the discovered resource

Techniques for decreasing the network load

- A TTL parameter to limit the number of hops between super-peers
- Each query message annotates the nodes that it traverses along its path.
  - a super-peer does not forward a query to a neighbour super-peer that has already received it.
- Each super-peer caches the IDs of the last received queries.
- If the super-peer finds several resources that satisfy the query, it constructs and forwards *only one* queryHit message containing the IDs of the nodes that own those resources.

The super-peer model in GT3

- The GT3 Index Service collects Service Data elements provided by Grid services.
- The static Superpeer Service uses the Index Service to manage metadata information related to the Grid services of a VO.



Resource Discovery Algorithm

Algorithm executed by the Superpeer Service

```
// v = max number of neighbours
// q.list: list of hosts traversed by the query q
// q.sender: neighbour super-peer from which q has been received
// q.id: query identifier
// q.ttl: current value of ttl
For each incoming query q:
  If <q.id is in the cache> then queryInCache:=true;
  Else <put q.id in the cache>
  q.ttl -= 1;
  if ((q.ttl>0) and not queryInCache)
  {
    select at most v best neighbours
    for each selected neighbour n:
      if <n is not in q.list> {
        <Add this super-peer to q.list>
        forward a copy of q to n
      }
  }
  <ask the local information service for resources matching q>
  if <there are such resources> {
    send to q.sender a queryHit containing the IDs of the nodes owning
      the discovered resources;
    send notifications to the hosts owning the resources;
  }
```

Simulation Analysis

- An **event-based object-oriented simulator** was used to:
  - Assess the effectiveness of the resource discovery protocol in large Grid networks
  - Evaluate the influence of protocol and network parameters on performance
- **Distribution of resources**
  - The number of nodes was varied from 10 to 10000
  - The cluster size was varied from 1 to 5000 nodes
  - The overall number of resources is proportional to the network size
  - A resource class is a set of resources that satisfy given constraints on resource properties
  - The number of resource classes offered by a Grid with N nodes is equal to $5*(\log_2 N)^2$

Performance Measures

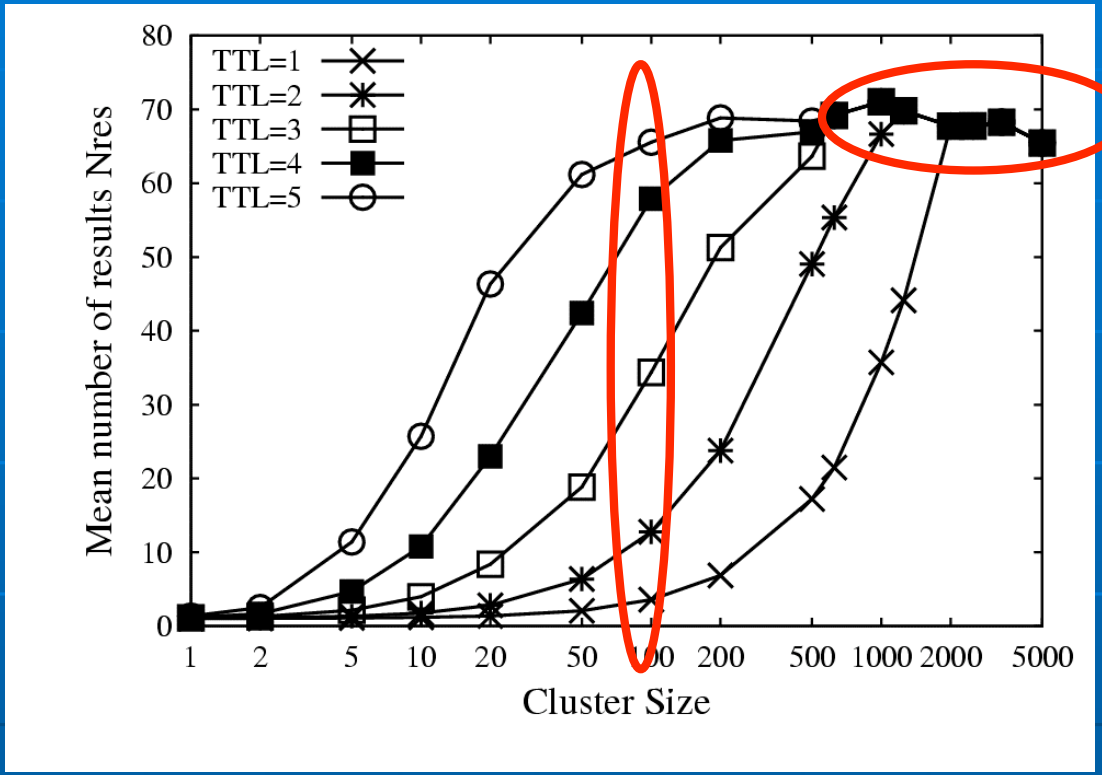
- To evaluate the success of a query, we calculated:
  - The Probability of success and the Mean number of results
  - *the latter is more significant to evaluate user satisfaction*
- The message load is the frequency of messages processed by a node
- The message load counterbalances the success indices…
  - The queryHits/messages ratio is an index of protocol efficiency
- The time to satisfaction is evaluated through…
  - Response times related to: a *generic* queryHit, the *kth* queryHit, the *last* queryHit received for a given query.

Two sets of simulations

- Grid networks having:
  - fixed number of nodes (10000, including peers and super-peers).
  - a variable mean cluster size, ranging from 1 (a fully decentralized P2P network) to 5000 (a network composed of two clusters).
  - different values of v (no. of neighbours for a generic node) and TTL to analyze how these parameters can be tuned to improve performance.
- Grid networks having:
  - A fixed mean cluster size, set to 10.
  - A variable number of nodes, ranging from 10 (a super-peer network having only one cluster) to 10000.

Mean number of results at super-peers

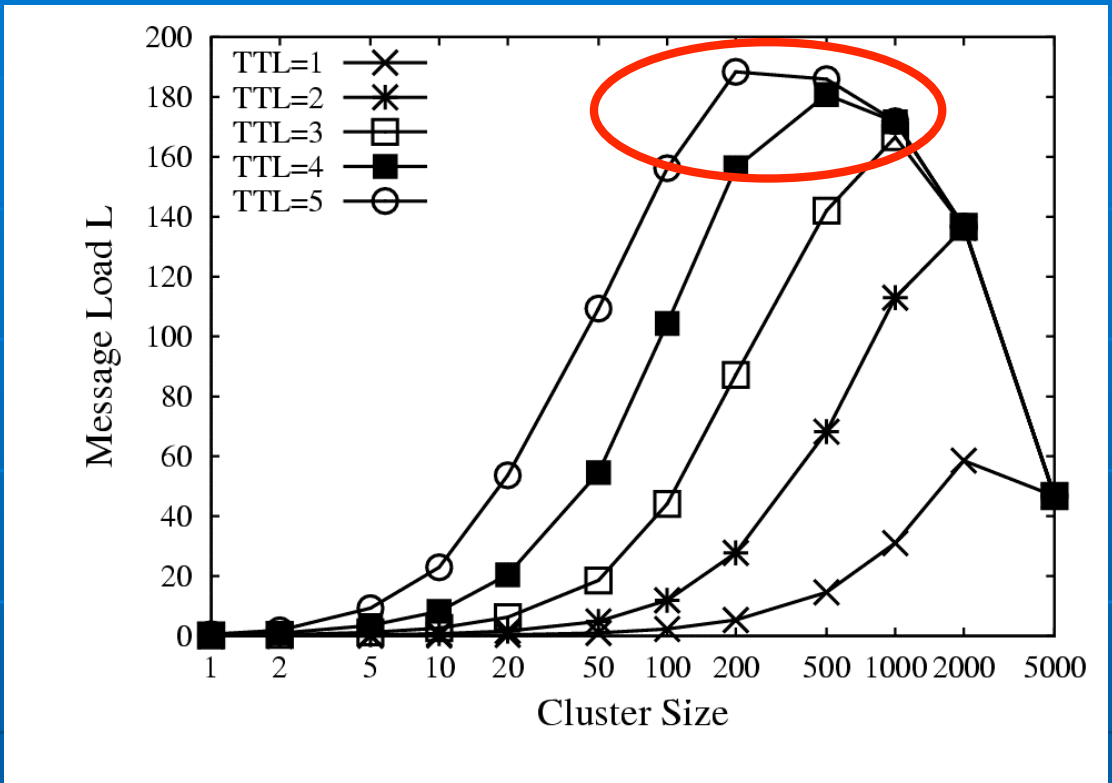
- Number of results versus cluster size C; TTL from 1 to 5, v = 4.



- performance increases with the TTL value as long as C is lower than 1000.
- beyond this threshold, curves tend to converge.
- these results can be exploited when tuning the value of TTL

Message load at super-peers

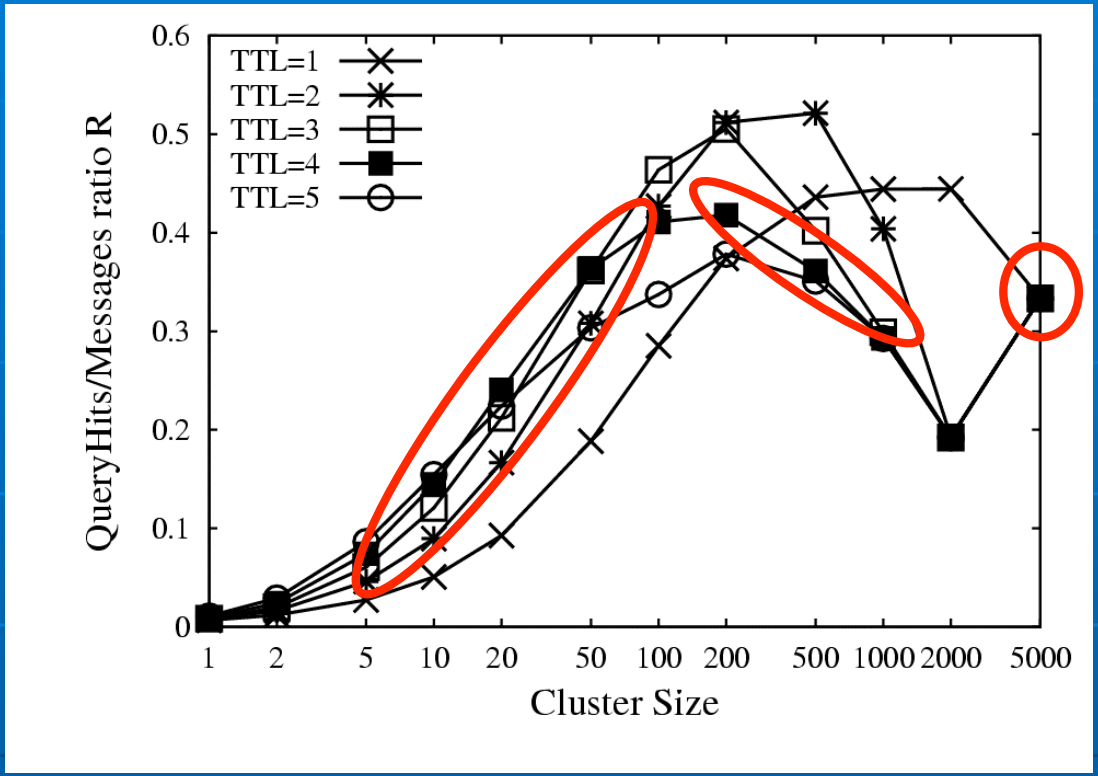
- Message load versus cluster size C; TTL from 1 to 5, v = 4.



- a high processing load is experienced at super-peers if a high number of results is desired
- a trade-off should be reached between maximizing the number of results and minimizing the processing load

QueryHits/messages ratio R at super-peers

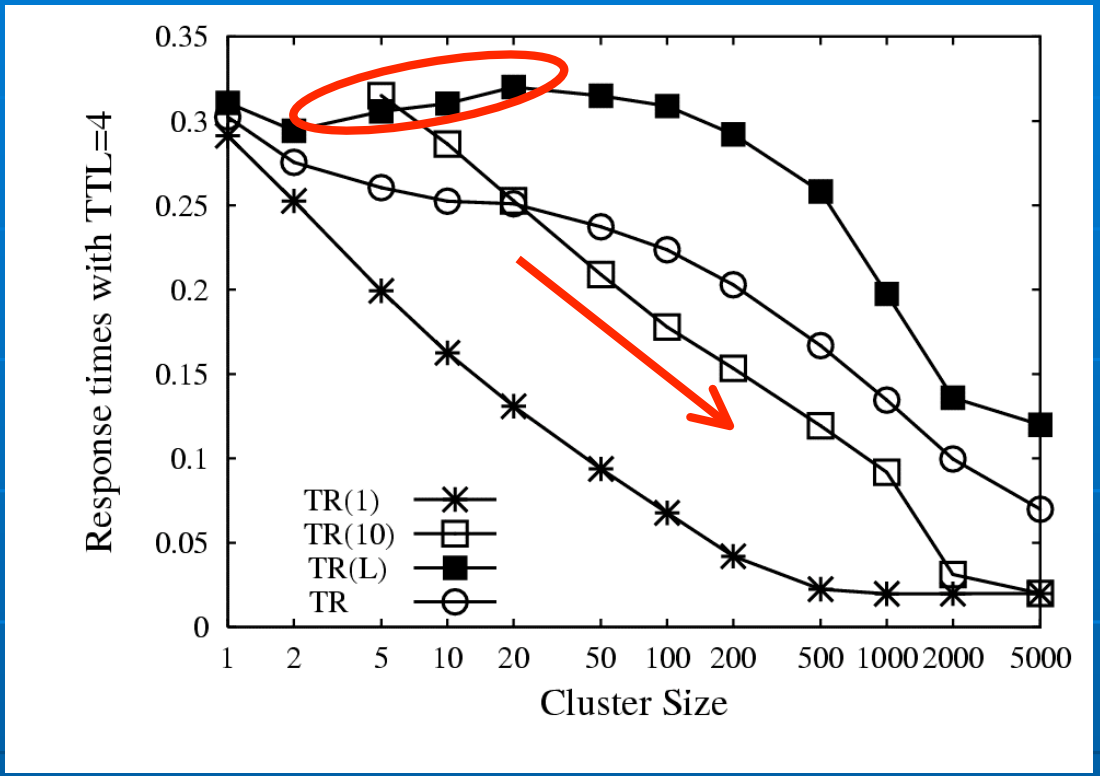
- Ratio R versus cluster size C; TTL from 1 to 5, v = 4.



- for a fixed value of TTL, R initially increases with C
- beyond a threshold value of C, the number of received queryHits falls down
- R converges to 1/3 with C=5000 because, with only two clusters:
  each super-peer receives comparable numbers of internal queries, external queries, and queryHits

Response times

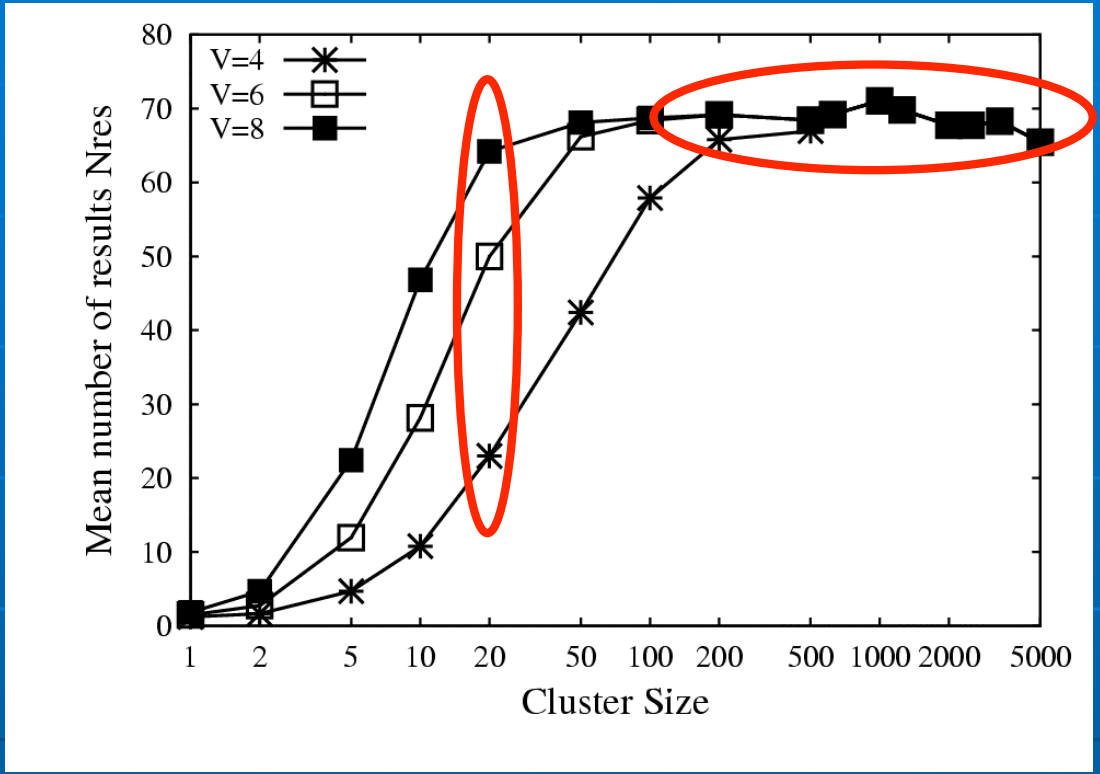
- Tr (average), Tr(1), Tr(10), Tr(Last) versus cluster size C; TTL = 4, v = 4.



- Response times decrease as the cluster size increases because:
  - queries and queryHits traverse a smaller number of super-peers
  - a higher fraction of queryHits are internal
- However Tr(Last) slightly increases as the cluster size increases form 2 to 100.

Influence of the number of neighbours (1)

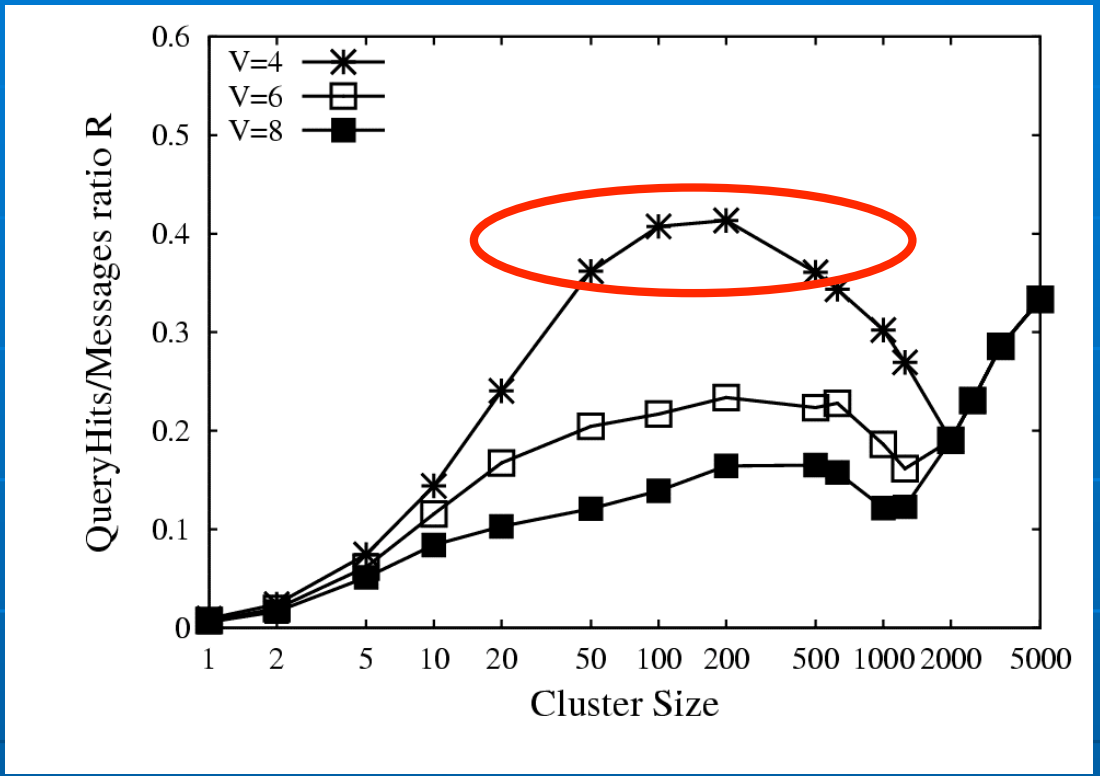
- Number of results versus cluster size C; TTL = 4, v from 4 to 8



- The number of results significantly increases with the value of the number of neighbours v only if the cluster size is lower than 100;
- With larger clusters, a value of v equal to 4 is sufficient to achieve a high number of results.

Influence of the number of neighbours (2)

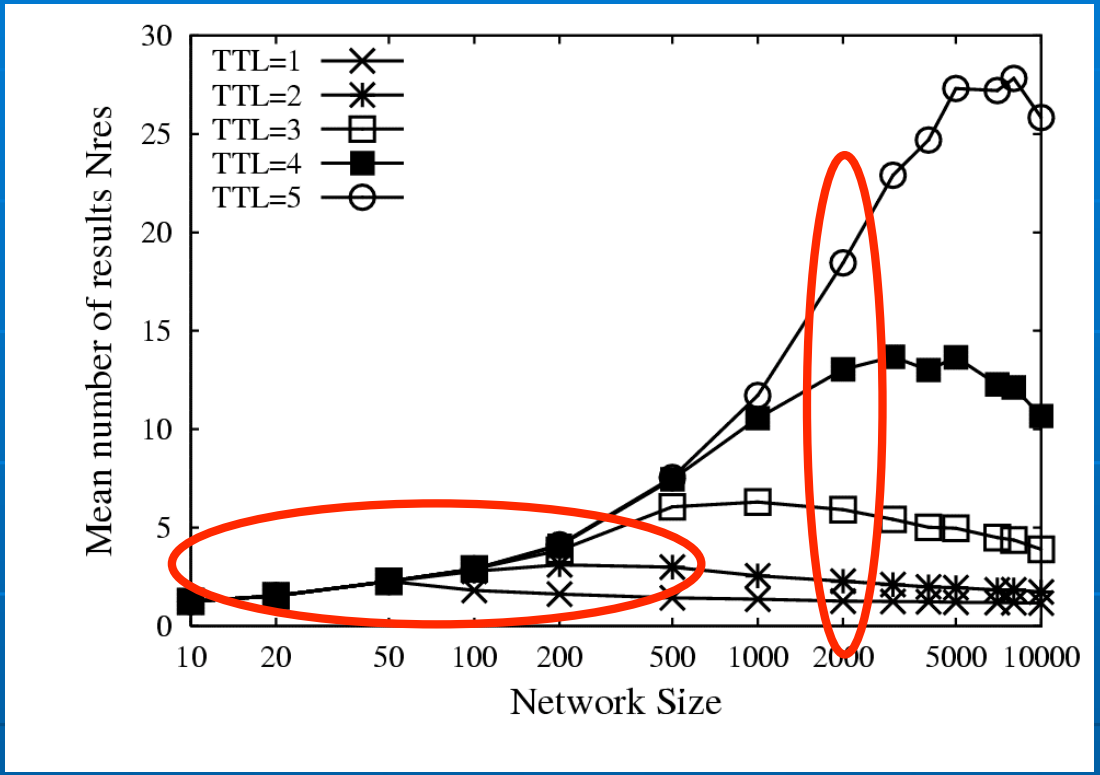
- Ratio R versus cluster size C; TTL = 4, v from 4 to 8



- The values of R are maximized with v equal to 4;
- We can conclude that it is not convenient to set v to a value higher than 4 if the cluster size exceeds 100: we would increase the load without a significant increase in the number of results.

Number of results with different network sizes

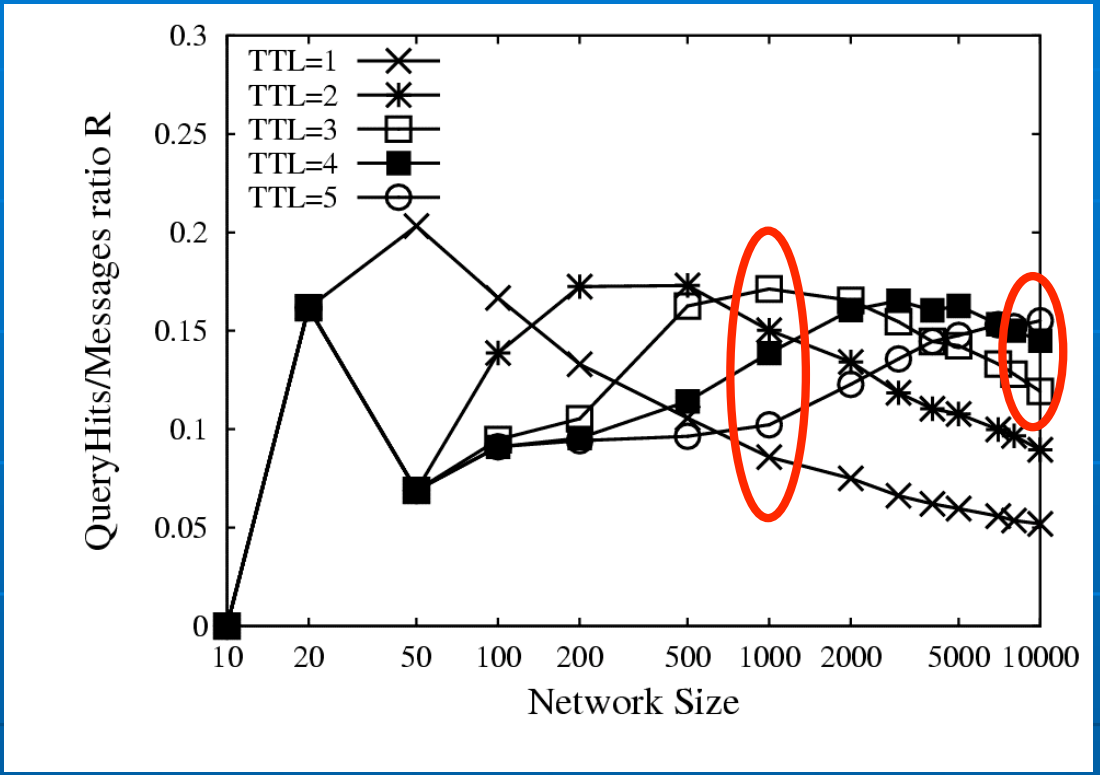
- Number of results versus network size; C = 10, v= 4, TTL from 1 to 5



- The number of results increases with TTL only if the network has more than 1000 nodes.
- For smaller networks, TTL has little effect.

QueryHits/messages with different network sizes

- Ratio R versus network size; C = 10, v= 4, TTL from 1 to 5



- The value of TTL that maximizes the ratio R increases with the network size.
- For example, in a network with 1000 nodes, R is maximized with TTL=3.
- TTL value should be 5 or higher for very large networks.
- In smaller networks a high TTL would unnecessarily increase the network load.

Conclusions

- The super-peer model facilitates the adoption of P2P techniques in highly dynamic, large-scale Grid networks.
- We proposed a resource discovery protocol based on the super-peer model.
- We analyzed the resource discovery protocol in different scenarios and network configurations.
- We evaluated the influence of protocol parameters (number of neighbours, time to live) on performance measures.
  - Results can be used to tune protocol parameters and improve performance for a given network configuration.