Dynamic Layer Management in Superpeer Architectures

Presented by 曾胤燁
2006/04/27

IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS
NOVEMBER 2005
Outline

- Introduction
- Workload Model
- Dynamic Layer Management Algorithm (DLM)
- Performance Evaluation
- Conclusion
Introduction

- Superpeer unstructured P2P systems have been found to be very effective by dividing the peers into two layers, super-layer and leaf-layer.
  - Message flooding is only conducted among superpeer.
Problems

- What is the optimal size ratio of leaf-layer to super-layer?
  - Too many superpeers – pure P2P systems
  - Too few superpeers – centralized P2P systems
Problems(2)

- How can the optimal ratio be maintained?
- What types of peers should be elected to super-layer?
Outline

- Introduction
- Workload Model
- Dynamic Layer Management Algorithm (DLM)
- Performance Evaluation
- Conclusion
Workload Model

- $n$ peers, $n_l$ peers are leaf-peers
- $n_s$ peers are superpeers
- Each leaf-peer connects to $m$ superpeers.
- Each superpeer connects to $k_s$ other superpeers and $k_l$ leaf-peers.
- $\eta = n_l / n_s$ (layer size ratio)
- $W_{on}$ - the workload on the overall network
- $W_{sp}$ - the workload on a superpeer
- The workloads can be divided into three parts:
  - Connection Workload
  - Query Workload
  - Relay Workload
Connection Workload (CW)

- CW is defined as the traffic overhead incurred to maintain the connections to the neighboring peers.
- CW is related to the size and stability of the neighboring peer set.
Connection Workload

\[ W_{sp\_cw} = \frac{k_l}{t_l} + \frac{k_s}{t_s} = \frac{mn\eta}{t_l} + \frac{k_s}{t_s} \]

\[ W_{on\_cw} = \frac{n_l m}{t_l} + \frac{n_s}{t_s} (k_l + k_s) = \frac{mn\eta}{(1 + \eta)t_l} + \frac{n(m\eta + k_s)}{(1 + \eta)t_s} \]

- \( W_{sp\_cw} \) and \( W_{on\_cw} \): the portions of connection workload in \( W_{sp} \) and \( W_{on} \)
- \( t_l \) and \( t_s \): the average lifetimes of neighboring leaf-peers and superpeers
Query Workload (QW)

- QW is defined as the traffic overhead incurred for a peer to process the queries generated by its leaf neighbors and itself.
- QW is proportional to the number of leaf neighbors and the query frequency.
Query Workload

\[ W_{sp\_qw} = k_l f = m\eta f \]

\[ W_{on\_qw} = \frac{nm\eta f}{1 + \eta} \]

- \( W_{sp\_qw} \) and \( W_{on\_qw} \) : the portions of query workload in \( W_{sp} \) and \( W_{on} \)
- \( f \) : the query frequency of a peer
Relay Workload (RW)

- RW is defined as the traffic overhead incurred to process queries relayed form the superpeer neighbors.
To cover \( p \) peers, the number of superpeers that should be queried has a lower bound of \( \frac{p}{1+k_l} \) and an upper bound of \( \frac{mp}{m+k_l} \).

- A superpeer can be viewed to represent \( k_l+1 \) peers.

\[
\begin{align*}
    p_s k_l &= p_l m \\
    p_s k_l &= p_l
\end{align*}
\]

*Theorem 1*
\[ \frac{(p_s \times k_i)}{m} \leq p_i \leq p_s \times k_i \]
\[ \Rightarrow p - p_s \leq p_s \times k_i \leq (p - p_s)m \]
\[ \Rightarrow \frac{p}{(1 + k_i)} \leq p_s \leq \frac{mp}{(m + k_i)} \]

When \( p_s << n_s \), \( p_s \) is very close to \( \frac{p}{(1 + k_i)} \)
To cover $p_s = p/(1+k_l)$ superpeers, the number of query message range from $(p/(1+k_l))-1$ to $pk_s/(1+k_l)$

- The ideal search algorithm should only query each peer once. Therefore, it can only use $p_s-1$ message.
- For an inefficient search algorithm, each link relays the same query at most twice. The maximum number of links is $p_s*k_s/2$, so the maximum number of messages is $p_s k_s$
Relay Workload

- Each peer initiates $f$ queries per time unit and each superpeer receives $(1+k_l)f$ queries from itself and its leaf neighbors.

  the query frequency of the total network $(1+k_l)n_sf$

- From theorem 2, the number of messages used by a query ranges from

  $(p/(1+k_l))-1$ to $pk_s/(1+k_l)$
Relay Workload

\[ W_{on-rw(min)} = (1 + k_l)n_s f \left( \frac{p}{1 + k_l} - 1 \right) = n_s f (p - 1 - k_l) \]
\[ = \frac{n_f}{1 + \eta} (p - 1 - m \eta) \]

\[ W_{on-rw(max)} = n_s f p k_s = \frac{f p k_s n}{1 + \eta} \]

\[ W_{sp-rw(min)} = (p - 1 - m \eta) f \] and \[ W_{sp-rw(max)} = f p k_s \].

\[ W_{sp-rw} \text{ is } \frac{1}{n_s} \text{ of } W_{on-rw} \]
Optimal Layer Size Ratio

- \( W = W_{cw} + W_{qw} + W_{rw} \)

\[
W_{sp(max)} = \frac{mn \eta}{t_l} + \frac{k_s}{t_s} + m \eta f + f p k_s
\]
\[
= \left( \frac{1}{t_l} + f \right) m \eta + \frac{k_s}{t_s} + f p k_s,
\]

\[
W_{on(max)} = \frac{mn \eta}{(1 + \eta) t_l} + \frac{n(m \eta + k_s)}{(1 + \eta) t_s} + \frac{mn \eta f}{1 + \eta} + \frac{f p k_s n}{1 + \eta}
\]
\[
= \frac{n}{1 + \eta} \left( \frac{m \eta}{t_l} + \frac{m \eta + k_s}{t_s} + m \eta f + f p k_s \right).
\]
Optimal Layer Size Ratio

\[ W = \alpha W_{sp} + \beta \frac{W_{on}}{\eta} \]  \hspace{1cm} (1)

- Since both \( W_{sp} \) and \( W_{on} \) are functions of \( \eta \), by differentiating we can obtain optimal value \( \eta \) as

\[ \eta' = \sqrt{\frac{B - C}{A}} - 1, \]
where, for the most efficient search algorithm,

\[ A = \frac{m\alpha}{t_l}, \quad B = \left( \frac{k_s}{t_s} + fp - f \right) \beta, \quad \text{and} \quad C = \left( \frac{1}{t_l} + \frac{1}{t_s} \right) m\beta, \]

while, for the most inefficient search algorithm,

\[ A = \left( \frac{1}{t_l} + f \right) m\alpha, \quad B = \left( \frac{1}{t_s} + fp \right) k_s\beta, \quad \text{and} \]

\[ C = \left( \frac{1}{t_l} + \frac{1}{t_s} + f \right) m\beta. \]
Dynamic layer management algorithm

1. Information Collection
2. Maintaining Appropriate Layer-Size-Ratio
3. Scaled Comparisons of Capacity and Age
4. Promotion or Demotion
1. Information Collection

- Peers exchange information with their superpeers to know their leaf neighbor number.
- Peers report their *age* and *capacity* to their superpeers.
2. Maintaining Appropriate Layer-Size-Ratio

- Due to the randomness of the neighbor selection mechanism in superpeer systems, the current numbers of leaf neighbors of superpeers can reflect the current layer size ratio.
- $l_{nn}$: the leaf neighbors number
- $\mu = \log \left( \frac{l_{nn}}{k_1} \right)$.
- $\mu > 0$: too few superpeers
- $\mu < 0$: too many superpeers
3. Scaled Comparisons of Capacity and Age

- For each peer that runs DLM, it uses two counting variables, $Y_{\text{capa}}$, $Y_{\text{age}}$.

\[
\text{for all peer } d_i \text{ in } G(d) \hspace{1cm} \\
\text{if } (\text{capacity}(d_i) \times X_{\text{capa}} > \text{capacity}(d)) \\
Y_{\text{capa}} += 1/(\text{size of } G(d)); \\
\text{if } (\text{age}(d_i) \times X_{\text{age}} > \text{age}(d)) \\
Y_{\text{age}} += 1/(\text{size of } G(d));
\]

- The value of $X_{\text{capa}}$ and $X_{\text{age}}$ are adjusted according to the value of $\mu$. 
4. Promotion or Demotion

- We use two threshold variables $Z_{\text{capa}}, Z_{\text{age}}$ in the determination.
- For a leaf-peer, if $Y_{\text{age}}$ and $Y_{\text{capa}}$ are smaller than $Z_{\text{capa}}$ and $Z_{\text{age}}$, it will be promoted.
- For a superpeer, if $Y_{\text{age}}$ and $Y_{\text{capa}}$ are larger than $Z_{\text{capa}}$ and $Z_{\text{age}}$, it will be demoted.
## Performance Evaluation

### Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>50,000</td>
<td>Number of peers in the network</td>
</tr>
<tr>
<td>$n_l$</td>
<td>48,780</td>
<td>Number of preferred leaf-peers</td>
</tr>
<tr>
<td>$n_s$</td>
<td>1,220</td>
<td>Number of preferred super-peers</td>
</tr>
<tr>
<td>$\eta$</td>
<td>40.0</td>
<td>Layer size ratio</td>
</tr>
<tr>
<td>$m$</td>
<td>2</td>
<td>Number of super-peer neighbors of a leaf-peer</td>
</tr>
<tr>
<td>$k_l$</td>
<td>80</td>
<td>Average number of leaf-peer neighbors of a super-peers</td>
</tr>
<tr>
<td>$k_s$</td>
<td>3</td>
<td>Average number of super-peer neighbors of a super-peers</td>
</tr>
<tr>
<td>$t_l$</td>
<td>3.5</td>
<td>Average duration time of leaf-peers</td>
</tr>
<tr>
<td>$t_s$</td>
<td>50</td>
<td>Average duration time of super-peers</td>
</tr>
<tr>
<td>$f$</td>
<td>0.3</td>
<td>Average number of queries of a peer per minute</td>
</tr>
<tr>
<td>$p$</td>
<td>3,000</td>
<td>Number of covered peers to ensure some fixed success rate</td>
</tr>
</tbody>
</table>
Weighted workload of most efficient search \((\alpha = 0.5, \beta = 0.5)\).

\[
\eta' = \sqrt{\frac{B-C}{A}} - 1 \approx 38,
\]
Weighted workload of most inefficient search \((\alpha = 0.5, \beta = 0.5)\).

\[
\eta_1' = \sqrt{\frac{B-C}{A}} - 1 \approx 51.
\]
Layer Size Comparison

Size

Simulation Time (Minute)

Super Layer
Leaf Layer
Conclusion

- In this paper, we purpose a workload model by analyzing the workload on one superpeer as well as on the total network.
- Based on this model, we can obtain an optimal layer size ratio.
- By DLM, we can adaptively elect peers and adjust them between superlayer and leaf-layer.
Thank you😊