3U-04　PC クラスタを用いた並列全文検索エンジン(2) — 動的負荷分散方法 —

河合英紀†, 森永聡‡, 小西弘一‡, 相場雄一‡, 赤峯享†
†NEC ヒューマンメディア研究所
‡NEC C&C メディア研究所

1. はじめに

近年、急速な情報化に伴って電子化テキストが大量に流通している。そのため、日々増加する情報に対応して、所望の情報をいつでも素早く手に入れる技術として、高速性・高拡張性・高信頼性を備えた全文検索エンジンの必要性が高まっている。

PC クラスタを用いた全文検索エンジン[1]には、並列処理による高速性、計算機の追加により性能向上できる高拡張性などの利点がある。一方、各計算機に負荷の偏りが存在すると、並列処理の効率が悪くなる問題がある。また、1 台のノードが停止すると、完全な検索結果を得られないという問題もある。

そこで、本稿ではインデックスを複製して異なるノードに分散冗長格納し、高負荷のノードや停止したノードの処理を別のノードに肩代わりさせることによって、負荷の均一化と耐障害性を実現する動的負荷分散方式を提案し、評価結果を報告する。

2. インデックスの分散冗長格納

図 1 にインデックス分散冗長格納方法の一例を示す。

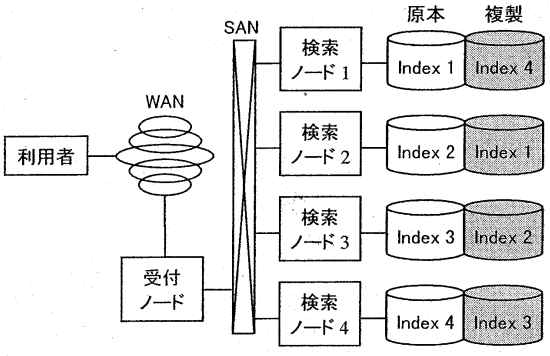

図 1 インデックスの冗長格納

図 1 では、まず検索対象となる大規模文書群を 4 つに分割し、原本インデックス 1〜4 を作成する。次に、原本インデックスを複製したインデックスを、同一ノードに重複しないように冗長に格納する。検索の際には、受付ノードがクエリの他に検索対象とするインデックスを指定して各検索ノードに処理を依頼し、Index1〜4 の全ての検索結果をマージして利用者に返答する。

これにより、ある検索ノードの負荷が増大しても、複製を持つ別のノードが処理を肩代わりできるので、負荷の偏りを均一化できる。また、1 台の検索ノードが停止しても、複製を持つ別のノードから検索結果を得ることができるため、耐障害性を実現することができる。

3. 負荷分散方式

並列全文検索エンジンにおける負荷分散を行う上では、以下に挙げる問題点が存在する。

(1) 各検索ノードの負荷は二次記憶装置からの読み出し時間に依存するため、CPU 利用率では各検索ノードの負荷を正しく把握できない。

(2) 二次記憶からの読み出し時間は検索結果の数に依存しているため、インデックスサイズを均一化[2]したり、ラウンドロビンで依頼先を変更しても、各検索ノードの処理時間は均一化されない。

(3) 検索ノード間での読み出しサイズを均一化するために、各インデックスキーについて検索結果を等分してインデックスを配置したり、負荷に応じて実行時に配置を変更したりする[3]と、1 台の検索ノード内で AND 検索できないため、通信オーバーヘッドが増大してしまう。

そこで、本稿では負荷の指標として検索ノードのキューに蓄積されている未処理のクエリの数を用いた。さらに、高負荷のノードへの依頼回数を減らし低負荷のノードへの依頼回数を増やすことによって、単位時間当たりの各検索ノードの処理時間が均一化されるようにした。

具体的には、クエリが入力された時点で各インデックスを処理可能な検索ノードのうち、負荷が最小のノードを選んで処理を依頼することとした。図 1 の場合では、Index1 を対象とした処理を検索ノード 1、2 のうち負荷

A Parallel Full Text Retrieval Engine Using a PC Cluster (2) — Dynamic Load Balancing —, Hideki Kawai, Satoshi Morinaga, Koichi Konishi, Yuichi Aiba and Susumu Akamine. NEC Corporation.

が軽い方を選んで依頼する。以下、Index2～4 を対象とした処理も同様にして依頼先の計算機を決定する。

4. 性能評価

各検索ノード間の負荷の偏りを変化させて、負荷分散しない場合と、提案方式との比較評価を行った。検索ノード 1 台当たりに 35 万件の文書をインデックス化し、1 台のノードのみインデックス化する文書を 1.5 倍または 2 倍することによって、負荷の偏りを発生させた。また、1 台のノードが停止した場合については、サービスの継続が可能である提案方式のみの性能を測定した。

クラスタを構成するノードには、450MHzPentiumII の CPU と 512MB のメモリを載せた PC を用い、検索ノード 12 台、受付ノード 1 台を Myrinet で接続して動作させた。キャッシュの効果を考慮して、システムをリブート後 5 万クエリ処理し、定常に達した最後の 1 万クエリでの平均スループットを性能の指標として用いた。クエリは、検索エンジン NETPLAZA[4]におけるログを使用した。

表 1 に、スループットの比較結果を示す。また、偏りが 1.5 倍の場合における各検索ノードの総処理時間の分布を図 2、3 に示す。

表 1 スループットの比較（単位は query per sec.）

	偏りなし	1.5 倍	2 倍	1 台停止
負荷分散なし	113	72	52	-
提案方式	74	69	64	47

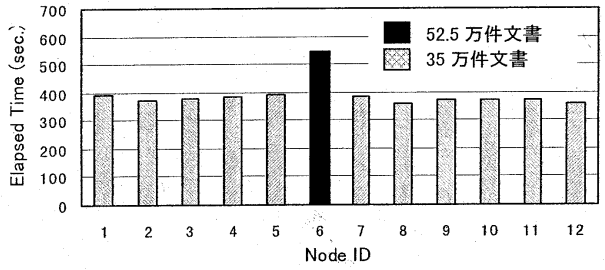

図2 検索ノードの処理時間分布（負荷分散なし）

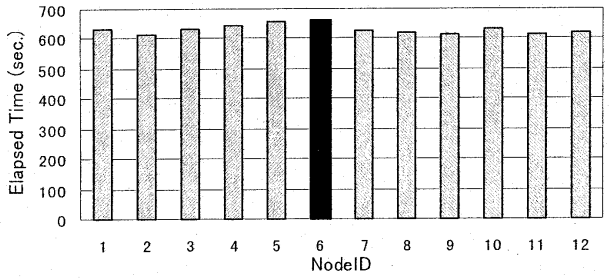

図3 検索ノードの処理時間の分布（提案方式）

5. 考察

表 1 より、検索ノード間の負荷の偏りが 2 倍以上大きい場合は、提案方式による負荷分散の方が優れているといえる。これは、高負荷のノードの処理を負荷分散によって他のノードに分散させることができたからである。

一方、負荷の偏りが 1.5 倍以下の場合には、負荷分散をしない場合のスループットの方が良い。この原因について以下に考察する。

図 2 では、負荷分散しないために、他の検索ノードより 1.5倍多い 52.5 万件の文書を所有するノード 6 がボトルネックとなって性能を低下させている。一方、図 3 では、提案方式によって各検索ノードの負荷が均一化されている。しかし、同時に各検索ノードの処理時間が図 2 の場合に比べて増大していることがわかる。

各検索ノードの処理時間が増大するのは、原本と複製インデックスによって、同一クエリに対する同じ内容の検索結果が異なるノードのメモリを占有し、キャッシュのヒット率を低下させていることが原因であると考えられる。

したがって、負荷の偏りが小さい時は原本優先でスケジューリングしてキャッシュの競合を緩和し、偏りが大きくなったり、停止ノードが出現した場合に、提案方式のスケジューリングに切り替えるなどの対策が考えられる。

6. おわりに

インデックスを複製して異なる計算機に分散冗長格納することによって、負荷の均一化と耐障害性を実現する動的負荷分散方式を提案・実装し、性能評価を行った。その結果、文書量の偏りが 1.5 倍以下では、負荷分散を行わない場合の性能が良く、2 倍以上の偏りがある場合は、提案方式が優れていることがわかった。

今後は、原本と複製インデックスのキャッシュ上での競合を緩和した負荷分散方式の検討を行う。

参考文献

[1]赤峯ほか、PC クラスタを用いた並列全文検索エンジン(1) -概要-、情処第 60 回全国大会,3U-03(2000).
[2]R.L.Haskin, Tiger Shark -A scalable file system for multimedea,IBM J. Res Develop.Vol.42 No.2,185(1998).
[3]P. Scheuermann, et al., "Disk Cooling" in Parallel Disk Systems. , IEEE Data Engineering Bulletin, Vol. 17 No. 3, 29(1994).
[4]検索エンジン NETPLAZA, http://netplaza.biglobe.ne.jp/index.html