Criteria to aid in solving the problem of allocating copies of a file in a computer network

J. G. Kollias* and M. Hatzopoulos†

The paper presents three criteria which aid in solving the problem of allocating copies of a file in a computer network. The criteria simplify the problem because (*a*) they can be applied *a priori* to determine that certain sites will (or will not) be included in the optimal allocation, and (*b*) they can become an integral part and accelerate a search procedure for finding an optimal allocation.

(Received October 1979)

1. Introduction

Casey (1972) considered the following problem: 'Given a fully connected computer network having n sites find the optimal set of network sites at which to locate copies of a file'. This problem is frequently referred to as the file allocation problem. Casey assumed that the overall file usage over a planning period is known and that this usage is expressed by the query and the update traffics originating at every site of the network. The problem was formulated as the 'plant location' problem in OR (Spielberg, 1969; Khumawala, 1972) with objective to find the particular allocation which minimises the total communication cost. The formulation shows that the problem has 2^n possible solutions and a heuristic was developed which reduces the computational complexity of finding an acceptable solution to a reasonably sized problem.

Grapa and Belford (1977) also considered the file allocation problem and they proved three theorems which apply to it. The theorems establish properties to aid in solving the problem by knowing in advance that certain sites must or must not have a copy of the file in the optimal allocation. We present stronger criteria than those introduced by the theorems of Grapa and Belford, and show how the criteria can be incorporated in Casey's (or other) heuristic when searching for the optimal allocation. It is noted that there is considerable literature concerning variations of the file allocation problem much of which is reviewed by Rothnie and Goodman (1977).

2. The criteria

We use notation similar to that used by Casey (1972) and Grapa and Belford (1977). Let:

n = number of sites in the network,
λ_j = query load originating at site j,
ψ_j = update load originating at site j,
σ_k = storage cost of file at site k,
d_{kj} = cost of communication of one query unit from site k to site j,
d'_{kj} = cost of communication of one update unit from site k to site j,
I = index set of sites with a copy of the file,
I^n = index set of all the n sites,
N_j = index set of sites that can communicate with site j,
I° = index set of sites without a copy of the file.

The file allocation problem is: Find the index set I which minimises the cost function

$$C(I) = \sum_{j=1}^{n} \Big[\sum_{k \in I} \psi_j d'_{kj} + \lambda_j \min_{k \in I} d_{kj}\Big] + \sum_{k \in I} \sigma_k .$$

The cost of storing and updating a copy of the file at site i is

$$Z_i = \sigma_i + \sum_{j=1}^{n} \psi_j d'_{ij} \tag{1}$$

The first criterion determines a minimum bound for allocating a file at a site.

Criterion 1:
Let

$$I^c = I^n - (I \cup I^\circ)$$
$$s_{ij} = \min_{\substack{k \in N_j \cap (I^c \cup I) \\ k \neq i}} \{\max(\lambda_j d_{kj} - \lambda_j d_{ij}, 0)\}$$
$$R_i = \sum_{j \in N_i} s_{ij} - Z_i \tag{2}$$

If $R_i > 0$ then a copy of the file must be allocated at site i.

The s_{ij} is a lower bound for the cost saving for site j that can be made by allocating a copy of the file at site i. Clearly if the sum of savings over all sites served by i exceeds the fixed cost Z_i then it is profitable to allocate a file at site i. It should be noted that this result is stronger than the criterion introduced by Theorem 1 in Grapa and Belford (1977).

The second criterion provides the means of reducing the sets N_i.

Criterion 2:
If for $i \in I$, $j \in N_i$

$$\min_{k \in I \cap N_j} (\lambda_j d_{kj} - \lambda_j d_{ij}) < 0 \tag{3}$$

then (site) j is dropped from N_i.

Of course if (3) holds for all $j \in N_i$ then $N_i = \emptyset$ and the i-th site will not have a copy of the file.

The third criterion determines a maximum bound on the cost reduction for allocating a file.

Criterion 3:
Let $i \in I^c$, $j \in N_i$

$$t_{ij} = \min_{k \in N_j \cap I} \{\max(\lambda_j d_{kj} - \lambda_j d_{ij}, 0)\}$$
$$T_i = \sum_{j \in N_i} t_{ij} - Z_i \tag{4}$$

If $T_i < 0$ then a copy of the file will not be allocated at site i.

The quantity t_{ij} is similar to s_{ij} in Criterion 1, but the comparisons are now made over all the sites which have been allocated a copy of the file. Clearly, if the sum of the savings, t_{ij}, is not greater than the fixed costs introduced by having a copy of the file at site i then it is not profitable to allocate a copy at site i. Criterion 3 is stronger than Theorem 3 in Grapa and Belford (1977) because it can become an integral part of any algorithm.

*School of Engineering, Department of Computer Science, University of Patras, Patras, Greece.
†Unit of Applied Mathematics, University of Athens, Athens, Greece.

CCC-0010-4620/81/0024-0029 $01.00

3. An example

To demonstrate how the criteria can be applied to solve the file allocation problem we take the communication costs from Casey's paper. These costs are symmetrical and are the same when serving either a query or an update (i.e. $d_{kj} = d_{jk} = d'_{kj} = d'_{jk}$).

Sites	1	2	3	4	5
1	0	6	12	9	6
2	6	0	6	12	9
3	12	6	0	6	12
4	9	12	6	0	6
5	6	9	12	6	0

We assume that the query, λ_j, and the update, ψ_j, traffic is as follows:

Sites	*Query load*	*Update load*
1	10	2
2	15	3
3	20	4
4	30	6
5	40	8

The overall query costs are:

Sites	1	2	3	4	5
1	0	90	240	270	240
2	60	0	120	360	360
3	120	90	0	180	480
4	90	180	120	0	240
5	60	135	240	180	0

and the overall update costs are:

Sites	1	2	3	4	5
1	0	18	48	54	48
2	12	0	24	72	72
3	24	18	0	30	96
4	18	36	24	60	48
5	12	27	48	36	0

By taking $\sigma_i = 0$ (i.e. we neglect the storage costs), from (1) we get:

$$Z_1 = 0 + 18 + 48 + 54 + 48 = 168,$$

$$Z_2 = 180, Z_3 = 174, Z_4 = 126 \text{ and } Z_5 = 123.$$

From (2) we get:

$$R_1 = 60 - 168 < 0, R_2 = 90 - 180 < 0,$$

$$R_3 = 120 - 174 < 0, R_4 = 180 - 126 > 0 \text{ and}$$

$$R_5 = 240 - 123 > 0.$$

Sites 4 and 5 satisfy Criterion 1 and therefore must both have a copy of the file in the optimal solution (i.e. $I = \{4, 5\}$). Given I and by applying Criterion 2 the sets N_i are reduced as follows:

$$N_1 = \{1, 2\}, N_2 = \{1, 2, 3\} \text{ and } N_3 = \{2, 3\} .$$

Finally, from (4) we get:

$$T_1 = 60 + 45 - 168 < 0,$$

$$T_2 = 0 + 135 + 0 - 180 < 0 \text{ and}$$

$$T_3 = 0 + 45 + 120 - 174 < 0 .$$

From Criterion 3 we have that sites 1, 2 and 3 should not have a copy of the file. (i.e. $I^\circ = \{1, 2, 3\}$). Every site is now included in either I or I°, so an optimal solution is now known. (It should be noted that the theorems in Grapa and Belford also produce $I = \{4, 5\}$ but fail to verify the optimality of this allocation).

4. Further discussion

In the previous section we show how the three criteria can be applied to determine *a priori* if certain sites will (or will not) have a copy of the file. When the network is large the criteria may fail to reduce significantly the size of the problem. In this case Casey's (1972) heuristic approach is the only way to attack the file allocation problem. The heuristic may be divided into two steps. The first step applies to the first few levels of the cost graph, which corresponds to the 2^n possible solutions of the problem, where a complete path tracing is performed. In the second step only the most promising paths are followed. Below we show how the criteria can improve both the steps of the heuristic and therefore improve its performance when they become an integral part of it.

For the first step it is clear that Criteria 1, 2 and 3 consider only certain paths and ignore the rest. This avoids the complete path tracing of the heuristic. For example, Casey (in his nineteen-node example which corresponds to the ARPA network) checks 171 nodes at the second level of the cost graph. When using Criterion 3 it is necessary to check only 113 nodes. The second step of Casey's heuristic considers only certain paths each time. Below we introduce two rules which locate promising paths.

R-rules

Criterion 1 states that a site i must be allocated a copy of the file if $R_i > 0$. The other sites which have the largest R_i values are most likely to have a copy of the file in the optimal solution. Therefore, promising sites for allocation are those with large R_i values. Similarly, sites with small R_i values may be excluded from having a copy of the file.

T-rules

Criterion 3 states that if $T_i < 0$ then site i should not be allocated a copy of the file. As before, the size of T_i may be useful in deciding which of the other sites are promising.

Acknowledgement

The authors thank Dr F. W. Burton for his helpful suggestions.

References

Casey, R. G. (1972). Allocation of copies of a file in an information network, *Proc. AFIPS SJCC*, Vol. 40, pp. 617-625.

Grapa, E. and Belford, G. G. (1977). Some theorems to aid in solving the file allocation problem, *CACM*, Vol. 20 No. 11, pp. 878-882.

Khumawala, B. M. (1972). An efficient branch and bound algorithm for the warehouse location problem, *Management Science*, Vol. 18 No. 12, pp. B-718-B-731.

Rothnie, J. B. and Goodman, N. (1977). A survey of research and development in distributed database management, *Proc. 3rd Int. Conf. on Very Large Databases*, Tokyo, Japan.

Spielberg, K. (1969). Algorithms for the simple plant location problem with some side conditions, *Oper. Research*, Vol. 17, pp. 85-111.