Read the attached paper and answer the following questions. Note that the time is limited, and you should budget your time carefully. It is suggested that you spend roughly 50 minutes in reading the paper and another 50 minutes in answering the questions. Note that 1) this is a long paper but you need not have to read it all. The content in section 5 is about experiments which can be ignored initially. 2) The ants that discuss in the paper include both "real" and "artificial" ants. 3) the Chinese meaning of the word "pheromone" is 費落蒙, a kind of 賀爾蒙 in live creature.

1. Please describe what the paper does, and the kind of problems it solves.

2. In the paper abstract, the author states that it is "a combination of distributed computation, positive feedback and constructive greedy heuristic" to stochastic optimization and problem solving. Please explain what "positive feedback" means here and how it is applied in the proposed algorithms.

3. Next, please explain what "distribution computing" means here and how it is applied in the proposed algorithms.

4. The last one, please explain what "constructive greedy" means here and which part in the proposed algorithms utilizes the greedy method.

Ant System: An Autocatalytic Optimizing Process

Technical Report 91-016

M. Dorigo , V. Maniezzo and A. Colorni
Dipartimento di Elettronica e Informazione
Politecnico di Milano
Piazza Leonardo da Vinci 32
20133 Milano, Italy
dorigo@elet.polimi.it

Abstract

A combination of distributed computation, positive feedback and constructive greedy heuristic is proposed as a new approach to stochastic optimization and problem solving. Positive feedback accounts for rapid discovery of very good solutions, distributed computation avoids premature convergence, and greedy heuristic helps the procedure to find acceptable solutions in the early stages of the search process. An application of the proposed methodology to the classical travelling salesman problem shows that the system can rapidly provide very good, if not optimal, solutions. We report on many simulation results and discuss the working of the algorithm. Some hints about how this approach can be applied to a variety of optimization problems are also given.

1. Introduction

In this paper we explore the emergence of global properties from the interaction of many simple agents. In particular, we are interested in the distribution of search activities over so-called "ants", i.e., agents that use very simple basic actions in order to ease the parallelization of the computational effort. Our work has been inspired by researches on the behavior of real ants ([5],[6],[12]), where one of the problems of interest is to understand how almost blind animals like ants can manage to establish shortest route paths from their colony to feeding sources and back. It was found that the media used to communicate among individuals information regarding paths and used to decide where to go consists of *pheromone trails*. A moving ant lays some pheromone (in varying quantities) on the ground, thus marking the path it follows by a trail of this substance. While an isolated ant moves essentially at random, an ant encountering a previously laid trail can detect it and decide with high probability to follow it, thus reinforcing the trail with its own pheromone. The collective behavior that emerges is a form of *autocatalytic* behavior[1] where the more the ants are following a trail, the more attractive that trail becomes for being followed. The process is thus characterized by a positive feedback loop, where the probability with which an ant chooses a path increases with the number of ants that previously chose the same path.

The algorithms that we are going to define in the next sections are a model deriving from the study of artificial ant colonies and therefore we call our system *Ant system* and the algorithms we introduce *Ant algorithms*. As we are not interested in simulation of ant colonies, but in the use of artificial ant colonies as an optimization tool, our system will have some major differences with a real (natural) one: artificial ants will have some memory, they will not be completely blind and will live in a environment where time is discrete. Nevertheless we believe that the ant colony metaphor can be useful as a didactic tool to explain our ideas.

[1] An autocatalytic [7], i.e. positive feedback, process is a process that reinforces itself, in a way that causes very rapid convergence and, if no limitation mechanism exists, leads to explosion.

A result obtained running experiments with the Ant systems was the observation of the presence of synergetic effects in the interaction of ants. In fact, the quality of the solution obtained increases when the number of ants working on the problem increases, up to reach an optimal point (see section 5.2 for details).

A major point in defining any distributed system is the definition of the communication procedure. In our algorithms a set of ants communicate by modifications of the problem representation, since at any step during the problem solving each ant gives a sign of its activity that will change the probability with which the same decision will be taken in the future. The idea is that if at a given point an ant has to choose between different options and the one actually chosen results in being particularly good, then in the future that choice will appear more desirable than it was before. We also give ants a heuristic to guide the early stages of the computational process, when experience hasn't yet accumulated into the problem structure. This heuristic automatically loses importance as the experience gained by ants, and memorized in the problem representation, increases.

A second, but no less important, result presented in this paper is the viability of autocatalytic processes as a methodology for optimization and learning. A "single-ant" autocatalytic process usually converges very quickly to a bad suboptimal solution. Luckily, the interaction of many autocatalytic processes can lead to rapid convergence to a subspace of the solution space that contains many good solutions, causing the search activity to find quickly a very good solution, without getting stuck in it. In other words, all the ants converge not to a single solution, but to a subspace of solutions; thereafter they go on searching for improvements of the best found solution.

At the present stage of understanding we do not have any proof of convergence or bound on the time required to find the optimal solution. Nevertheless simulations strongly support the above speculations. We also believe our approach to be a very promising one because of its generality (it can be applied to many different problems) and because of its effectiveness in finding very good solutions to difficult problems.

The paper is organized as follows: section 2 contains the description of the algorithm as it is currently implemented together with the definition of the application problem: as the algorithm structure partially reflects the problem structure, we introduce them together. Sections 3 and 4 describe three slightly different ways to apply the proposed algorithm to the chosen problem. Section 5 reports diffusely on experiments carried out with the algorithms previously introduced. In section 6 we discuss the properties of the distributed algorithm and show how it could be applied to other optimization problems. Conclusions, related and further work are contained in section 7.

2. The Ant system

We introduce in this section our approach to the distributed solution of difficult problems by many locally interacting simple agents. We call *ants* the simple interacting agents and ant-algorithms the class of algorithms we define. We first describe the general characteristics of the ant-algorithms and then introduce three of them, called *Ant-density*, *Ant-quantity* and *Ant-cycle*.

To test the ant algorithms, we decided to apply them to the well-known travelling salesman problem (TSP) [15], to have a comparison with results obtained by other heuristic approaches [11]: the model definition is influenced by the problem structure, however we will hint in section 6 that the same approach can be used to solve other optimization problems. We stress that the choice of TSP is due to its ubiquity as a benchmark for heuristics: we are interested in

the proposal of a new heuristic and in its comparison with other ones, not directly in proposing – at least in this paper – a more efficient approach to the solution of TSP (which in fact has been solved optimally for problems of much higher order than those presented here).

Given a set of n towns, the TSP problem can be stated as the problem of finding a minimal length closed tour that visits each town once.

We call d_{ij} the length of the path between towns i and j; in the case of euclidean TSP d_{ij} is the Euclidean distance between i and j (i.e., $d_{ij}=[(x_i-x_j)^2 + (y_i-y_j)^2]^{1/2}$). An instance of the TSP problem is given by a weighted graph (N,E), where N is the set of towns and E is the set of edges between towns, weighted by the distances.

Let $b_i(t)$ (i=1, ..., n) be the number of ants in town i at time t and let

$$m = \sum_{i=1}^{n} b_i(t)$$

be the total number of ants.

Each ant is a simple agent with the following characteristics:

- when going from town i to town j it lays a substance, called *trail*, on edge (i,j);
- it chooses the town to go to with a probability that is a function of the town distance and of the amount of trail present on the connecting edge;
- to force ants to make legal tours, transitions to already visited towns are inhibited till a tour is completed (see the tabu list in the following).

Let $\tau_{ij}(t)$ be the *intensity of trail* on edge (i,j) at time t. At each iteration of the algorithm trail intensity becomes

$$\tau_{ij}(t+1)=\rho\cdot\tau_{ij}(t)+\Delta\tau_{ij}(t,t+1) \qquad (1)$$

where ρ is a coefficient such that $(1 - \rho)$ represents the evaporation of trail,

$$\Delta\tau_{ij}(t,t+1) = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t,t+1)$$

$\Delta\tau_{ij}^{k}(t,t+1)$ is the quantity per unit of length of trail substance (pheromone in real ants) laid on edge (i,j) by the k-th ant between time t and t+1.

The coefficient ρ must be set to a value <1 to avoid unlimited accumulation of trail (see note 1). The intensity of trail at time 0, $\tau_{ij}(0)$, should be set to arbitrarily chosen small values (in our experiments the same value is chosen for every edge (i,j)).

In order to satisfy the constraint that an ant visits n different towns (a n-town tour), we associate to each ant a data structure, called *tabu list*[2], that memorizes the towns already visited up to time t and forbids the ant to visit them again before a tour has been completed. When a tour is completed the tabu list is emptied and the ant is free again to choose its way. We define

[2] Even though the name chosen recalls tabu search, proposed in [9] and [10], there are substantial differences between our approach and tabu search algorithms. We mention here: (i) the absence of any aspiration function, (ii) the difference of the elements recorded in the tabu list, permutations in the case of tabu search, nodes in our case (our algorithms are constructive heuristics, which is not the case of tabu search).

$\mathbf{tabu_k}$ a vector containing the tabu list of the k-th ant, and $\mathbf{tabu_k}(s)$ the s-th element of the tabu list of the k-th ant (i.e., the s-th town visited by ant k in the current tour).

We call *visibility* η_{ij} the quantity $1/d_{ij}$, and define the transition probability from town i to town j for the k-th ant as

$$p_{ij}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{j \in \mathbf{allowed}} [\tau_{ij}(t)]^{\alpha} [\eta_{ij}]^{\beta}} & \text{if } j \in \mathbf{allowed} \\ 0 & \text{otherwise} \end{cases} \qquad (2)$$

where $\mathbf{allowed} = \{j: j \in \mathbf{tabu_k}\}$ and where α and β are parameters that allow a user to control the relative importance of trail versus visibility. Therefore the transition probability is a trade-off between visibility (which says that close towns should be chosen with high probability, thus implementing a greedy constructive heuristic) and trail intensity (that says that if on edge (i,j) there has been a lot of traffic then it is highly desirable, thus implementing the autocatalytic process).

Different choices about how to compute $\Delta\tau_{ij}^{k}(t,t+1)$ and when to update the $\tau_{ij}(t)$ cause different instantiations of the ant algorithm. In the next two sections we present the three algorithms we used as experimental test-bed for our ideas, namely Ant-density, Ant-quantity and Ant-cycle.

3. The Ant-density and Ant-quantity algorithms

In the Ant-density model a quantity Q_1 of trail for every unit of length is left on edge (i,j) every time an ant goes from i to j; in the Ant-quantity model an ant going from i to j leaves a quantity Q_2/d_{ij} of trail for every unit of length.

Therefore, in the Ant-density model

$$\Delta\tau_{ij}^{k}(t,t+1) = \begin{cases} Q_1 & \text{if k-th ant goes from i to j between t and t+1} \\ 0 & \text{otherwise} \end{cases} \qquad (3)$$

and in the Ant-quantity model we have

$$\Delta\tau_{ij}^{k}(t,t+1) = \begin{cases} \dfrac{Q_2}{d_{ij}} & \text{if k-th ant goes from i to j between t and t+1} \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

From these definitions it is clear that the increase in trail intensity on edge (i,j) when an ant goes from i to j is independent of d_{ij} in the Ant-density model, while it is inversely proportional to d_{ij} in the Ant-quantity model (i.e., shorter edges are made more desirable by ants in the Ant-quantity model, thus further reinforcing the visibility factor in equation (2)).

The Ant-density and Ant-quantity algorithms are then

```
1  Initialize
       Set t:=0                                   {t is the time counter}
       For every edge (i,j) set an initial value τij(t) for trail intensity and Δτij(t,t+1):= 0
       Place bi(t) ants on every node i           {bi(t) is the number of ants on node i at time t}
       Set s:=1                                   {s is the tabu list index}
       For i:=1 to n do
             For k:=1 to bi(t) do
             tabuk(s):=i                          {starting town is the first element of the tabu list of the k-th ant}

2  Repeat until tabu list is full  {this step will be repeated (n-1) times}
   2.0 Set s:=s+1
   2.1 For i:=1 to n do                           {for every town}
       For k:=1 to bi(t) do                       {for every k-th ant on town i still not moved}
             Choose the town j to move to, with probability pij(t) given by equation (2)
             Move the k-th ant to j  {this instruction creates the new values bi(t+1)}
       Insert node j in tabuk(s)
       Set   Δτij(t,t+1):= Δτij(t,t+1) + Q1 in case of the Ant-density model or
             Δτij(t,t+1):= Δτij(t,t+1) +Q2/dij in case of the Ant-quantity model
   2.2 For every edge (i,j) compute τij(t+1) according to equation (1)

3  Memorize the shortest tour found up to now
   If (NC < NCMAX) or (not all the ants choose the same tour)  {NC is the number of cycles}
   then
       Empty all tabu lists
       Set s:=1
       For i:=1 to n do
             For k:=1 to bi(t) do
                   tabuk(s):=i                    {after a tour the k-th ant is again in the initial position}
       Set t:=t+1
       For every edge (i,j) set Δτij(t,t+1):=0
       Goto step 2
   else
       Print shortest tour and Stop
```

In words the algorithms work as follows. At time zero an initialization phase takes place during which ants are positioned on different towns and initial values for trail intensity are set on edges. The first element of each ant tabu list is set to be equal to the starting town. Thereafter every ant moves from town i to town j choosing the town to move to with a probability that is given as a function (with parameters α and β) of two desirability measures: the first (called trail - τ_{ij}) gives information about how many ants in the past have chosen that same edge (i,j), the second (called visibility - η_{ij}) says that the closer a town the more desirable it is (setting $\alpha = 0$ we obtain a stochastic greedy algorithm with multiple starting points, setting $\alpha = 0$ and $\beta \to \infty$ we obtain the deterministic classical one).

Each time an ant makes a move, the trail it leaves on edge (i,j) is summed to trail left on the same edge in the past. When every ant has moved, transition probabilities are computed using new trail values, according to formulae (1) and (2).

After n-1 moves the tabu list of each ant will be full: the shortest path found by the m ants is computed and memorized and all tabu lists are emptied. This process is iterated until the tour counter reaches the maximum (user-defined) number of cycles NC_{MAX} or all ants make the same tour (we call this last case *uni-path* behavior: it denotes a situation in which the algorithm stops searching for alternative solutions, see section 5.1).

By examination of the algorithms we see that the computational complexity expressed as a function of the number of ants m, the number of towns n and the number of cycles NC (where

a cycle is a complete tour) is $O(NC\cdot(m\cdot n^2+n^3))$. In fact, for both the Ant-quantity and Ant-density algorithms we have that:

Step 1 is $O(n^2+m)$.

Step 2 is $O(n\cdot m+n^3)$ (step 2.1 is $O(m)$, step 2.2 is $O(n^2)$, and they are repeated n times),

Step 3 is $O(n\cdot m+n^2)$

In our experiments, see section 5, we found that the optimal number of ants is $m=c_1\cdot n$, where c_1 is a small constant (the experimental best value is $c_1=1$). Therefore the overall complexity is $O(NC\cdot n^3)$. NC could well be a function of n (NC=NC(n)). We have investigated the form of the NC function in section 5.7.

4. The Ant-cycle algorithm

In this case we introduced a major difference with respect to the two previous systems. Here $\Delta\tau_{ij}^k$ is not computed at every step, but after a complete tour (n steps). The value of $\Delta\tau_{ij}^k(t,t+n)$ is given by

$$\Delta\tau_{ij}^k(t,t+n)=\begin{cases}\dfrac{Q_3}{L^k} & \text{if k-th ant uses edge (i,j) in its tour}\\ 0 & \text{otherwise}\end{cases} \qquad (5)$$

where Q_3 is a constant and L^k is the tour length of the k-th ant. This corresponds to an adaptation of the Ant-quantity approach, where trails are updated at the end of a whole cycle instead of after each single move. We expect this algorithm to have a better performance than the previously defined Ant-density and Ant-quantity because here global information about the value of the result (i.e., the tour length) is used.

The value of the trail is also updated every n steps according to a formula very similar to (1)

$$\tau_{ij}(t+n) = \rho_1\cdot\tau_{ij}(t)+\Delta\tau_{ij}(t,t+n) \qquad (1')$$

where $\Delta\tau_{ij}(t,t+n) = \sum_{k=1}^{m}\Delta\tau_{ij}^k(t,t+n)$

and ρ_1 is different from ρ because the equation is no more updated at every step but only after a tour (n steps).

The Ant-cycle algorithm is then

```
1  Initialize:
     Set t:=0                                  {t is the time counter}
     For every edge (i,j) set an initial value τij(t) for trail intensity and Δτij(t,t+n):= 0
     Place bi(t) ants on every node i          {bi(t) is the number of ants on node i at time t}
     Set s:=1                                  {s is the tabu list index}
     For i:=1 to n do
       For k:=1 to bi(t) do
           tabuk(s):=i                         {starting town is the first element of the tabu list of the k-th ant}
```

```
2  Repeat until tabu list is full          {this step will be repeated (n-1) times}
   2.0 Set s:=s+1
   2.1 For i:=1 to n do                    {for every town}
         For k:=1 to b_i(t) do             {for every k-th ant on town i still not moved}
           Choose the town j to move to, with probability p_ij(t) given by equation (2)
           Move the k-th ant to j          {this instruction creates the new values b_j(t+1)}
           Insert node j in tabu_k(s)
3  For k:=1 to m do                        {for every ant}
     Compute L^k, as results from its tabu list
     For s:=1 to n-1 do                    {scan the tabu list of the k-th ant}
       Set (h,l):=(tabu_k(s),tabu_k(s+1))
                                           {(h,l) is the edge connecting town (s,s+1) in the tabu list of ant k}
       Δτ_hl(t+n):=Δτ_hl(t+n) + Q_3/L^k
4  For every edge (i,j) compute τ_ij(t+n) according to equation (1')
   Set t:=t+n
   For every edge (i,j) set Δτ_ij(t,t+n):=0
5  Memorize the shortest tour found up to now
   If (NC < NC_MAX) or (not all the ants choose the same tour)      {NC the number of cycles}
     then
       Empty all tabu lists
       Set s:=1
       For i:=1 to n do
         For k:=1 to b_i(t) do
             tabu_k(s):=i                  {after a tour the k-th ant is again in the initial position}
       Goto step 2
     else
       Print shortest tour and Stop
```

The complexity of the Ant-cycle algorithm is $O(NC \cdot n^2 \cdot m)$ if we stop the algorithm after NC cycles (remember NC could be NC(n)). In fact we have that:

Step 1 is $O(n^2+m)$
Step 2 is $O(n^2 \cdot m)$
Step 3 is $O(n \cdot m)$
Step 4 is $O(n^2)$
Step 5 is $O(n \cdot m)$

Also for the Ant-cycle algorithm we found a linear relation between the number of towns and the best number of ants. Therefore the complexity of the algorithm is $O(NC \cdot n^3)$.

5. Computational results

We implemented the three algorithms and investigated their relative strengths and weaknesses by experimentation. Since we have not yet developed a mathematical analysis of the models, which would yield the optimal parameter setting in each situation, we ran several simulations to collect statistical data for this purpose. During this phase we found Ant-cycle to be superior to the other two algorithms. We therefore tried to deepen our understanding of it by analyzing the effects of several variations of the basic algorithm, like defining a unique versus different starting points for every ant, increasing the importance of the ant that found the best tour or adding noise to the computation of the probabilities. These results are described in the next subsections, together with a brief comparison with alternative heuristics for the TSP.

5.1 Parameters setting

The parameters considered here are those that affect directly or indirectly the computation of the probability in formula (2): α, β, ρ, Q_h (h=1, 2, 3). The number m of ants has always been set equal to the number n of cities. We tested several values for each parameter, all the other being constant (the default value of the parameters was α=1, β=1, ρ=0.7, Q_h=100; in each experiment only one of the values was changed), over ten simulations for each setting in order to achieve some statistical information about the average evolution. The values tested were: $\alpha \in \{0, 0.5, 1, 5\}$, $\beta \in \{0, 1, 2, 5, 10, 20\}$, $\rho \in \{0.3, 0.5, 0.7, 0.9, 0.999\}$ and $Q_h \in \{1, 100, 10000\}$; α and β have been tested over different sets of values because of the different sensitivity the algorithms have shown for them. Preliminary results, obtained on small-scale problems, have been presented in [3] and [4]; the tests reported here are based, where not otherwise stated, on the Oliver30 problem, a 30-cities problem described in [13], for which a tour of length 424.635 was found using genetic algorithms[3]. The same result is also often obtained by the Ant system, which can also yield better outcomes. In order to allow the comparison with other approaches (see section 5.6) the tour lengths have been computed both as real numbers and as integers[4]. All the tests have been carried out for NC_{MAX} = 5000 cycles and were averaged over ten trials.

Beside the tour length, we were interested also in investigating the *uni-path behavior*, i.e., the situation in which all the ants make the same tour: this would indicate that the system has ceased to explore new possibilities and therefore the best tour achieved so far will not be improved any more. With some parameters' settings in fact we observed that, after several cycles, all the ants followed the same tour despite the stochastic nature of the algorithms: this was due to a much higher trail level on the edges composing that tour than on all the others. This high trail level makes the probability that an ant chooses an edge not belonging to the tour very low.

The three algorithms show a different sensitivity to the parameters.

Ant-density shows for β a monotonic decrease of the tour length up to β=10. After this value the average length starts to increase.

β	0	1	2	5	**10**	20
avg. length	881.56	456.98	455.52	431.37	426.74	428.79

The tests on the other parameters show that α has an optimum around 1

α	0	0.5	**1**	2	5
avg. length	578.52	464.99	456.98	508.44	695.25

that ρ should be set as high as possible

[3] Genetic algorithms [13] seem to be a very good general heuristic for combinatorial optimization problems, at least as good as simulated annealing [17].

[4] In this case distances between towns are integer numbers and are computed according to the standard code proposed in TSPLIB 1.0 (Reinelt G., TSPLIB 1.0, Institut für Mathematik, Universität Augsburg, 1990).

ρ	0.3	0.5	0.7	0.9	[0.999]
avg. length	649.86	530.58	456.98	431.31	429.09

and that the system is little affected by Q_1, never being able to improve significantly the quite unsatisfactory solution obtained in standard condition. The importance of the quantity Q_h (h=1,2,3) resulted to be uninfluential in all the three algorithms.

The experiments with Ant-density show that this system enters the uni-path behavior only for $\beta \geq 2$, usually within the first 200-300 cycles.

Ant-quantity shows a different sensitivity to the parameters; β is still very important, as shown in the following table

β	0	1	2	5	10	[20]	30
avg. length	454.72	441.85	436.77	431.60	428.52	427.95	438.83

where a decrease of the average tour length with increasing β is still exhibited up to β=20.

Also in this case we have that a too high or a too low α can worsen the performance, but a minor sensitivity to trails (α=0.5) could improve the result obtained with standard configuration (α=1).

α	0	[0.5]	1	5
avg. length	649.45	430.70	441.85	478.48

The tests on ρ show that keeping a strong memory of past experience is a good policy here, as it was in Ant-density, since again higher ρ yield better results.

ρ	0.3	0.5	0.7	0.9	[0.999]
avg. length	555.24	451.51	441.85	426.48	426.25

Ant-quantity is more prone to uni-path behavior than Ant-density, in fact we observed uni-path behavior for $\alpha \geq 1$ and for $\beta \geq 1$. In all the other cases the system kept on the exploration activity.

Results obtained with Ant-cycle show that α presents an optimal range around 1, β between 2 and 5 and ρ around 0.5.

The average of the tests are, for β:

β	0	0.5	1	2	[5]	10	20
avg. length	848.31	452.62	427.44	424.63	424.25	428.35	438.88

for α:

α	0	0.5	[1]	2
avg. length	651.27	533.49	427.44	456.11

and for ρ:

ρ	0.3	0.5	0.7	0.9
avg. length	427.85	426.86	427.44	428.28

The Ant-cycle enters the uni-path behavior only for $\alpha \geq 2$; in all the other cases we always observed an ongoing exploration of different alternatives, even after more than $NC_{MAX}=5000$ cycles.

In the three algorithms ρ is the only parameter that has qualitatively different behaviors. Its best value is as high as possible (i.e., very close to 1) in Ant-density and Ant-quantity, and ρ=0.5 in Ant-cycle. This fact can be explained as follows: Ant-density and Ant-quantity use only strictly local information, i.e., the visibility η_{ij} (local by definition) and τ_{ij} that in these cases is also local because doesn't contain any information about the final tour length of the ants that laid it in the past. The amount of trail on each edge is therefore a direct consequence of the local greedy rule. Conversely, in the Ant-cycle algorithm trail actually laid is a function of the total tour length (see formula (5)) and the algorithm uses therefore global information on the result of sequences of local moves. In other words what happens is that in the first two algorithms τ_{ij} is only a reinforcement of η_{ij}, while in Ant-cycle, as global information about total tour length is used to compute the amount of trail to deposit, τ_{ij} represents a different type of information with respect to η_{ij}.

All the three algorithms mainly use the greedy heuristic to guide search in the early stages of computation, but it is only Ant-cycle that as computation runs can start exploiting the global information contained in the values τ_{ij} of trail. This explains both the better performance of Ant-cycle and the value ρ=0.5: the algorithm needs to have the possibility to forget part of the experience gained in the past in order to better exploit new incoming global information. In Ant-density and Ant-quantity this forgetting capability is neither necessary nor useful because there is no "second stage" in the search process in which global information can be exploited: the two algorithm always continue to use the same strategy based on the greedy heuristic.

We investigated also the behavior of the Ant-cycle algorithm for different combination of parameters α and β (in this experiment every run was stopped after 2500 cycles, i.e., after every ant had made 2500 tours). The results are summarized in Fig.1. We identified three different zones: for high values of α and not too high values of β the algorithm enters very quickly the uni-path behavior without finding very good solutions (this situation is represented by the symbol ■ in Fig.1); if we don't give enough importance to trail (i.e., we set α to a low value) or we give too a high importance to the greedy rule (high values to β) then the algorithm doesn't find very good solutions in the number of cycles used in the experiment (the symbol used for this situation is ∞). Very good solutions are found for α and β values in the central area (where the symbol used is ✖). In this case we found that different parameter combinations (i.e., (α=1, β=1),(α=1, β=2),(α=1, β=5),(α=0.5, β=5)) resulted in the same performance level: same result (the shortest tour known on the Oliver30 problem) in approximately the same number of cycles.

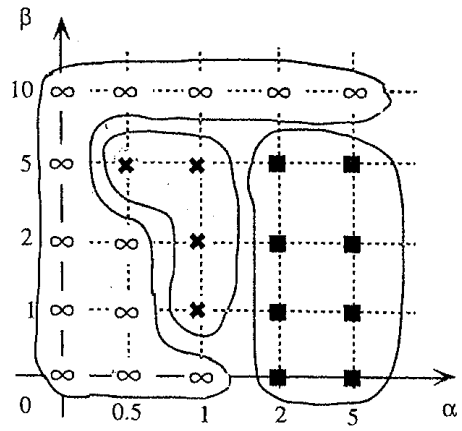


Fig.1 - Ant-cycle behavior for different combinations of α-β parameters
✖ - the algorithm finds very good solutions without entering the uni-path behavior
∞ - the algorithm doesn't find very good solutions without entering the uni-path behavior
■ - the algorithm doesn't find very good solutions and enters the uni-path behavior

The results obtained in this experiment are consistent with our understanding of the algorithm: a high value for α means that trail is very important and therefore ants tend to choose edges chosen by other ants in the past. This is true until the value of β becomes very high: in this case even if there is a high amount of trail on a edge, an ant always has a high probability of choosing another town that is very near.

High values of β and/or low values of α make the algorithm very similar to a stochastic multigreedy algorithm.

In Fig.2 we present the new optimal tour[5] found using the - experimentally determined - optimal set of parameters values for the Ant-cycle algorithm (α=1, β=2, ρ=0.5, Q_3=100). This tour results to be of length 423.74 and presents two inversions with respect to the best tour published in [20].

[5] This result is not exceptional since big dimensional TSP problems are now solved to optimality with special-purpose algorithms. Nevertheless, it is interesting to note that our algorithm can consistently find very good solutions to published problems (see also section 5.7)

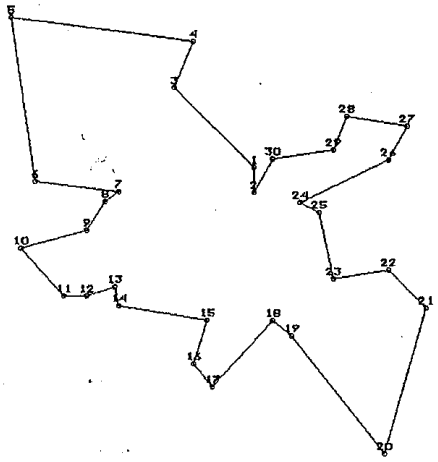


Fig.2 - The new best found tour[6] obtained with 342 iterations of Ant-cycle for the Oliver30 problem (α=1, β=2, ρ=0.5, Q_3=100), real length = 423.74, integer length = 420 (cfr. note 4).

The major strengths of the Ant-cycle algorithm can be summarized in the following points:

- within the range of parameter optimality the algorithm always finds a very good solution, most of the times one that is better than the best found using genetic algorithms;
- the algorithm finds good solutions very quickly (see Fig.3); nevertheless it doesn't enter the uni-path behavior (in which all ants choose the best found tour), viz. the ants continue to search for new possibly better tours;
- we tested the Ant-cycle algorithm to problems with increasing dimensions and we found the sensitivity of the parameters optimal values to the problem dimension to be very low.

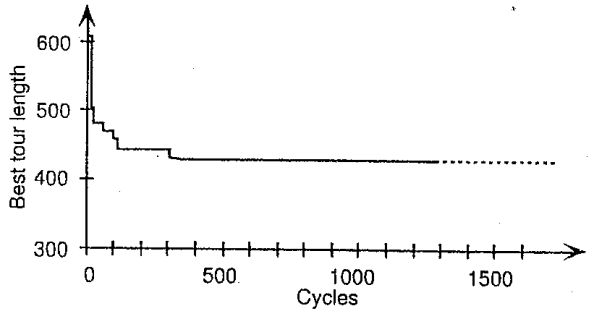


Fig.3 - The algorithm finds very good values for Oliver30 very quickly and the new optimal value (423.74) after 342 cycles. In this figure cycles correspond to complete tours.

On the Oliver30 problem Ant-cycle reached the former best-known result [20] with a frequency statistically higher than that of the other two algorithms and, on the whole, identification of good tours is much quicker; moreover, it is the only one which can identify the new optimal tour.

---

[6] The previous best found tour, published in [20], has real length of 424.63 and integer length of 421.

We partially tested the algorithm on the Eilon50 and Eilon75 problems [8] on a limited number of runs and with the number of cycles constrained to NC_{MAX}=3000. Under these restrictions we never got the best-known result, but a quick convergence to satisfying solutions was maintained for both the problems.

5.2 Number of ants

A set of experiments was run, in order to assess the impact of the number of ants on the efficiency of the solving process. In this case, the test problem involved finding a tour in a 4x4 grid of evenly spaced points: this is a problem with a priori known optimal solution (160 if we put to 10 the edge length, see Fig.4).

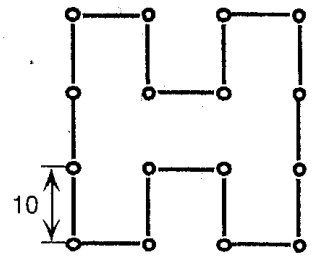


Fig.4 - An optimal solution for the 4x4 grid problem

In this case we determined the average number of cycles needed in each configuration to reach the optimum, if the optimum could be reached within 2000 cycles. As already said, this test, as the following ones, was conducted only for the Ant-cycle algorithm. The results are shown in Fig.5: on the abscissa there is the total number of ants used in each set of runs, on the ordinate the so-called *one-ant cycles*, i.e., the average number of cycles required to reach the optimum, multiplied by the number of ants used (in order to evaluate the efficiency per ant, hence the name, and to have comparable data). It is interesting to note that:

- the algorithm has been consistently able to identify the optimum with any number $m \geq 4$ of ants;
- there is a synergetic effect in using more ants, up to an optimality point given by n=m; the existence of this optimality point is due to the computational load caused by the management of progressively more ants that causes the overall efficiency, measured in one-ant cycles, to decrease when increasing number of ants;
- tests on a set of r x r grid problems (r = 4, 5, 6, 7, 8) have shown that the optimal number of ants is close to the number of cities (n≈m): this property was used in the assessment of the computational complexity (see sections 3 and 4).

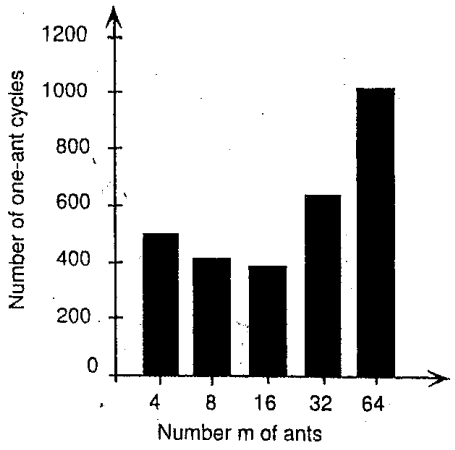


Fig.5 - Number of one-ant cycles required to reach optimum as a function of the total number of ants for the 4x4 grid problem

A second set of tests has been carried out with 16 cities randomly distributed (16 cities random graph). Again we found that the optimal performance was reached with 8-16 ants, a number of ants comparable with the dimension (measured in number of cities) of the problem to be solved.

5.3 Which town should ants start from?

We tested whether there is any difference between the case in which all ants at time t=0 are in the same city and the case in which they are *uniformly* distributed[7]. We used Ant-cycle applied to the 16 cities random graph and the 4x4 grid described in the previous subsection, and to the Oliver30 problem. In all cases uniformly distributing ants resulted in better performance.

In the case of the 16 cities random graph, we run 16 experiments (each one repeated five times) in which all the ants were positioned, at time t=0, on the same city (in the first experiment all ants were on town 1, in the second on town 2, and so on). We obtained that in all the cases the ants were able to identify the optimum, but they needed an average of 64.4 cycles vs. the average 57.1 needed in case of uniformly distributed ants.

In the case of the 4x4 grid problem experiments showed that an average of 28.2 cycles were needed to identify the optimum vs. 26.9 in case of uniformly distributed ants (in order to compare this data with those of Fig.5, the number of cycles must be multiplied by 16 to obtain the number of one-ant cycles).

In the case of the Oliver30 problem with 30 ants starting from the same city (runs were done using optimal parameters values), we noticed that the ants were never able to identify the optimum (average value of the best tour found: 438.43), and that after some hundreds of cycles all the ants followed one of a very small set of tours.

We also tested whether an initial uniform distribution of the ants over the cities performed better than a random one; results show that there is little difference between the two choices, even though the random distribution obtained slightly better results.

---

[7] We say ants are uniformly distributed if there is, at starting point, the same integer number of ants on every town (this excludes the possibility to have m which is not a multiple of n).

5.4 Elitist strategy

We call elitist strategy (because in some way it resembles the elitist strategy used in genetic algorithms, reproducing with probability 1 the best individual found) the modified algorithm in which at every cycle the trail laid on the edges belonging to the so far best found tour is reinforced by a quantity $e \cdot Q_3/L^*$, where e is the number of elitist ants[8] and L^* is the length of the best found tour. The idea is that the trail of the best tour so far identified (L^*), so reinforced, will direct in probability the search of all the other ants towards the edges that compose it.

The test were carried out again on the Oliver30 problem (the run was stopped after NC_{MAX} = 2500 cycles) and indicated that there is an optimal range for the number of elitist ants: below it, increasing their number results in better tours discovered and/or in the best tour being discovered earlier, above it, the elitist ants force the exploration around suboptimal tours in the early phases of the search, so that a decrease of performance results. Fig.6 shows the outcome of a test where this behavior is evident.

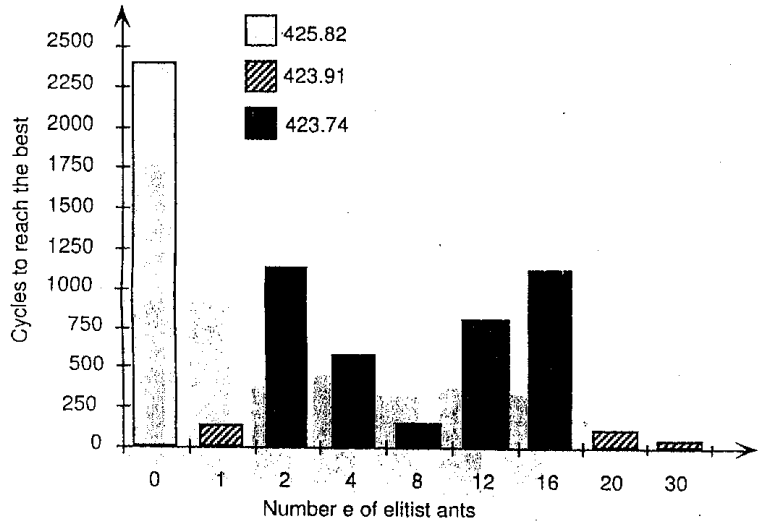


Fig.6 - Number of cycles required to reach a local optimum related to the number of elitist ants used

5.5 Noisy transition probability

We used in some experiments a slightly different transition rule including noise, given by the following formula:

[8] In our case the effect of an individual, i.e. an ant with its tabu list defining a tour, is to increment the value of the trail on edges belonging to its tour; therefore the equivalent of saving an individual is in our case to sum its contribution to the contribution of all other ants in the following cycle.

$$p_{ij}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)\cdot(1+\varepsilon_{ij}(\sigma))]^{\alpha}\cdot[\eta_{ij}]^{\beta}}{\sum_{j\in \mathbf{allowed}} [\tau_{ij}(t)\cdot(1+\varepsilon_{ij}(\sigma))]^{\alpha}\cdot[\eta_{ij}]^{\beta}} & \text{if } j\in \mathbf{allowed} \\ 0 & \text{otherwise} \end{cases} \qquad (2')$$

where **allowed** = {j: j∈ **tabu**$_k$} and $\varepsilon_{ij}(\sigma)$ is a noise function (random variable with zero mean and standard deviation σ).

This set of tests was carried out in order to assess the usefulness of formula (2') compared to formula (2), to calculate transition probabilities. The original idea was to evaluate the robustness of the procedure with respect to the intensity level of the trail, and to see whether early convergence could be avoided. The second objective turned out to be intrinsically met by the Ant-cycle algorithm, which naturally tend not to present the uni-path behavior. As test problem we used the 4x4 grid problem and we noticed that low noise values do not hamper the identification of the optimum – although they do not help either – while higher values lead to a significant worsening of the performance. In Fig.7 we display the average number of cycles, over 10 tests for each value, needed to reach the optimum with different values of noise.

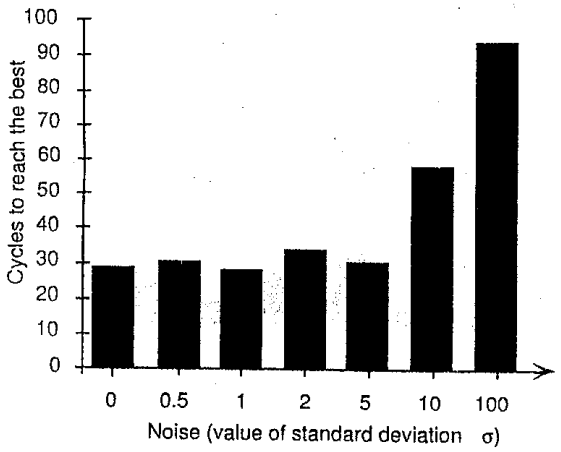


Fig.7 - Number of cycles required to reach optimum related to the different values of noise

5.6 Time required to find optimal solutions

The algorithm complexity presented in section 4, $O(NC\cdot n^3)$, doesn't say anything about the actual time required to reach the optimum. The experiment presented in this section is devoted to investigate the relation between NC and n, i.e., NC=NC(n). Results are reported in Table I for the case of similar problems with increasing dimensions (r x r grids with the edge length set to 10 as in Fig.4). It is interesting to note that, up to problems with 64 cities, Ant-cycle always found the optimal solution. Moreover both the number of cycles required to find it and the computational time actually used grow more slowly than the dimension of the search space, suggesting again that the algorithm uses a very effective search strategy.

Table I - Time required to find optimum as a function of problem dimension

Problem	Best solution	Relative dimension of search space	Average number of cycles to find optimum	Time required to find optimum[9] (seconds)
4 x 4	160	1	5.6	8
5 x 5	254.1	$\approx 10^{11}$	13.6	75
6 x 6	360	$\approx 10^{28}$	60	1020
7 x 7	494.1	$\approx 10^{49}$	320	13440
8 x 8	640	$\approx 10^{75}$	970	97000

Observing Table I we note that the time required to find optimum is proportional to the the number of cycles necessary to find it multiplied by n^3 (in this case n = 16, 25, 36, 49, 64) confirming in this way the complexity analysis presented in section 4.

5.7 Comparison with other approaches

We compared the results of Ant-cycle with those obtained, on the same Oliver30 problem, by the other heuristics contained in the package "Travel" [2]. This package represents the distances among the cities as an integer matrix and so, in order to enable the comparison, we implemented an analogous representation in our system.

The results are shown in Table II, where in the first column there is the length of the best tour identified by each heuristic, in the second and third columns the improvement on the corresponding first column solution as obtained by the 2-opt (exhaustive exploration of all the permutations obtainable from the basic one by exchanging 2 cities) and the Lin-Kernighan heuristics [16], respectively.

Note how Ant-cycle consistently outperformed 2-opt, while its efficacy – i.e., the effectiveness it has in finding very good solutions – can be compared with that of Lin-Kernighan (even if our algorithm requires a longer computational time).

Table II - Performance of Ant-cycle compared with other approaches

	basic[10]	2-opt	Lin-Kernighan[11]
Ant-cycle	420	-	-
Near Neighbour	587	437	420/421
Far Insert	428	421	420/421
Near Insert	510	492	420/421
Space Filling Curve	464	431	420/421
Sweep	486	426	420/421
Random	1212	663	420/421

As a general comment of all the tests, we like to point out that, given a good parameter setting (for instance α=1, β=2, ρ=0.5, Q_3=100, e=5), our algorithm consistently finds a very

[9] Tests were run on a IBM-compatible PC.
[10] The name "basic" means the basic heuristic, with no improvement.
[11] The Lin-Kernighan algorithm found solutions of length 420 or 421 depending on the starting solution provided by the basic algorithm.

good solution, the optimal one in case of BAYG29[12] or the new best known one in case of Oliver30, and finds rather quickly satisfying solutions (it usually identifies for Oliver30 the new best-known solution of length 423.74 in less than 400 cycles, and it takes only ≈100 cycles to reach values under 430). In any case exploration continues, as it is testified by the non-zero variance of the lengths of the tours followed by the ants in each cycle and by the fact that the average of the ants tour lengths gets never equal to the best tour found, but remains somewhat above it, thus indicating that tours around the best found are tested.

6. Discussion

Results presented in the preceding section suggest that the algorithm could be an effective optimization tool. In this section we try to give some insights about the way the algorithm works and to show how it could be applied to other NP-hard combinatorial problems.

A first way to explain the effect of applying the algorithm to the TSP problem is the following.

Consider the transition matrix **p**(t): every element $p_{ij}(t)$ is the transition probability from town i to town j at time t as defined by equation (2). At time t=0 each $p_{ij}(0)$ is proportional to η_{ij}, i.e., closer towns are chosen with higher probability. As the process evolves, **p**(t) changes its elements according to (1) and (2). The process can therefore be seen as a space deformation, in which distance is reduced between towns which are connected by edges with a high amount of traffic, and, conversely, distance is incremented between towns connected by edges with low traffic levels. From simulations we observed that the matrix **p**(t), at least in the range of optimality for our parameters, converges to a state[13] that is very close to stationary (i.e., variations in the transition matrix **p**(t) are very small). When this state is reached the behavior of the ants is dependent on the kind of transition matrix obtained. We observed two situations: in the most rare one, occurring - as we saw in section 5 - only for particular parameter settings, only one transition probability is significantly higher than zero in every row and therefore all the ants choose the same edge at each step and no new tour is searched. In the most common situations instead, most of the rows have two (sometimes more) transition probabilities with a value higher than zero. In these cases search never stops, even if the dimension of the space searched is highly reduced, with respect to the initial situation. Consider for example Fig.8, obtained as steady-state transition matrix for a randomly generated 10-town problem, where the area of each circle is proportional to the corresponding value of the transition probability. An ant in town 1 has a very high probability to go either to town 5 (near 50%) or to town 2 (near 35%), and a low probability of choosing any other edge (a similar analysis holds for ants in any other town; from town 9 and 0, for example, any destination is equally probable).

---

[12] This is a 29 cities problem proposed in TSPLIB 1.0 (cfr. note 4).

[13] The stochastic process that rules the evolution of the matrix **p**(t) is a Markov process with infinite memory.

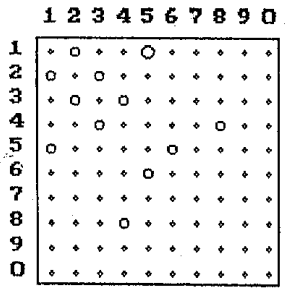


Fig.8 - Steady-state transition matrix for a randomly generated 10-town problem (Ant-cycle)

Another way to interpret how the algorithm works is to imagine to have some kind of probabilistic superimposition of effects: each ant, if isolated (i.e., if α=0), would move with a local, greedy rule. This greedy rule guarantees only locally optimal moves and will practically always lead to bad final results. The reason the greedy rule doesn't work is that greedy local improvements lead to very bad final steps (an ant is constrained to make a closed tour and therefore choices for the final steps are constrained by early steps). So the tour followed by an ant ruled by a greedy policy is composed by some parts that are very good and some others that are not. If we now consider the effect of the simultaneous presence of many ants, then each one contributes to a part of the trail distribution: good sets of paths will be followed by many ants and therefore they receive a great amount of trail; bad paths chosen only because obliged by constraints satisfaction (remember the tabu list) will be chosen only by few ants and therefore the trail over them remains low.

We have seen that the ants cooperate by exchanging a particular kind of information, trail, that is memorized in the problem structure. As no direct communication is necessary, and only local information is used to take decisions, the algorithm is very well suited to parallelization. We are currently designing the parallel version of Ant-cycle for a transputer architecture: we intend to give a set of ants to each transputer and to merge, every n steps, the trail left by each set of ants obtaining in this way a new trail matrix. We then redistribute this matrix to all the nodes.

Let's now consider the generality of our approach. We believe that many combinatorial problems can be faced by the Ant system. In order to apply the autocatalytic algorithm to another combinatorial problem, we must find an appropriate representation for:

1) the problem (to be represented as a graph searched by many simple agents);
2) the autocatalytic process;
3) the heuristic that allows a constructive definition of the solutions (the "*greedy force*");
4) the constraint satisfaction method (viz. the tabu list).

This has been done for three well-known combinatorial optimization problems – Satisfiability (SAT), Quadratic Assignment (QAP) and Job-Shop Scheduling (JSP) – each time obtaining an adapted version of the Ant-system that could effectively handle the relative problem. The most difficult (and *ad hoc*) tasks to face when trying to apply the Ant-system are to find an appropriate graph representation for the problem to be solved and a greedy force as heuristic. This required the introduction of many edges between each pair of nodes in the case of QAP, of one more node in the case of JSP, of suitable constraints between each pair of nodes in the case of SAT.

7. Conclusions, related work and future investigations

This paper introduces a new search methodology based on a *distributed autocatalytic process* and its application to the solution of a classical optimization problem. Our main contributions are:

(i) we introduce positive feedback as a powerful search and optimization tool
(ii) we show how synergetic effects can arise in distributed systems.

The Ant system uses many simple interacting agents and a fast search algorithm based on positive feedback,without getting trapped in local minima; moreover augmenting the number of agents has a synergetic effect on the system performance (until an upper limit is reached).

We reported many simulation results that illustrate the power of the approach. Moreover we presented an example of how the system can be applied to other optimization problems: we believe the approach can be extended to a broader class of problems.

The general idea underlying this model is that of a population of agents each one guided by an autocatalytic process pressed by a greedy force. Were an agent alone, both the autocatalytic process and the greedy force would tend to make the agent converge to a suboptimal tour with exponential speed. When agents interact it looks like the greedy force can give the right suggestions to the autocatalytic process and let it converge on very good, often optimal, solutions very quickly, without getting stuck in local optima. We have speculated that this behavior can be due to the fact that information gained by agents during the search process is used to modify the problem representation and in this way to reduce the dimension of the space considered by the search process. Even if no tour becomes unfeasible, bad tours become highly improbable, and the algorithms search only in the neighbourhood of good solutions.

Related work can be classified in the following major areas:

(i) studies of social animals behavior;
(ii) research in "natural algorithms";
(iii) stochastic optimization.

As already pointed out the research on behavior of social animals is to be considered as a source of inspiration and as a useful metaphor to explain our ideas. We believe that, especially if we are interested to design inherently parallel algorithms, observation of natural systems can be an invaluable source of inspiration. Neural networks [19], genetic algorithms [13], evolution strategies [18], immune networks [1], simulated annealing [14] are only some of the proposed models with a "natural flavour". Main characteristics, at least partially shared by members of this class of algorithms, are the use of a natural metaphor, the inherent parallelism, the stochastic nature and adaptativity, the use of positive feedback, the capacity to learn (i.e., to improve performance on the basis of past experience). Our algorithm can then be well considered to be a new member of this class. All this work in "natural optimization" can be inserted in the more general research area of stochastic optimization, in which the quest for optimality is traded for computational efficiency.

We believe that further work can be done along the following main research directions:

- theoretical investigation of the proposed model (properties of convergence, complexity, ...);
- evaluation of the generality of the approach, through an investigation on which classes of problems can be solved by this algorithm;
- evaluation of the scalability of the approach by testing it on bigger problems using a parallel computer;

- study of the implications that our model can have on artificial intelligence, particularly in the pattern recognition and machine learning fields.

Acknowledgments

We would like to thank Thomas Bäck, Hughes Bersini, Frank Hoffmeister, Mauro Leoncini, Francesco Maffioli, Bernard Manderik, Giovanni Manzini, Daniele Montanari, Hans-Paul Schwefel and Frank Smieja for useful comments on a early version of this paper.

References

[1] H.Bersini, F.J.Varela, "The Immune Recruitment Mechanism: a Selective Evolutionary Strategy," *Proc. of the Fourth Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, 1991.

[2] S.C.Boyd, W.R.Pulleyblank, G.Cornuejols, "Travel," Software Package, Carleton University, 1989.

[3] A.Colorni, M.Dorigo, V.Maniezzo "Distributed Optimization by Ant Colonies," *Proc. of the First European Conference on Artificial Life*, Paris, France, December 11-13, 1991.

[4] A.Colorni, M.Dorigo, V.Maniezzo, "Positive feedback as a search strategy," Technical Report n. 91-016 Politecnico di Milano, 1991.

[5] J.L.Denebourg, J.M.Pasteels, J.C.Verhaeghe, "Probabilistic Behaviour in Ants: a Strategy of Errors?," *J. Theor. Biol.*, vol.105, pp. 259-271, 1983.

[6] J.L.Denebourg, S.Goss, "Collective patterns and decision-making," *Ethology, Ecology & Evolution*, vol.1, pp. 295-311, 1989.

[7] M.Dorigo, *Optimization, Learning and Natural Algorithms*, Ph.D. Thesis, Politecnico di Milano, in press.

[8] S.Eilon, T.H.Watson-Gandy, N.Christofides, "Distribution management: mathematical modeling and practical analysis," *Operational Research Quarterly*, vol.20, pp.37-53, 1969.

[9] F.Glover, "Tabu Search — Part I," *ORSA Journal on Computing*, vol.1, no.3, pp.190-206, 1989.

[10] F.Glover, "Tabu Search — Part II," *ORSA Journal on Computing*, vol.2, no.1, pp.4-32, 1990.

[11] B.Golden, W.Stewart, "Empiric analysis of heuristics," in *The Travelling Salesman Problem*, E.L.Lawler, J.K.Lenstra, A.H.G.Rinnooy-Kan, D.B.Shmoys eds., New York:Wiley, 1985.

[12] S.Goss, R.Beckers, J.L.Denebourg, S.Aron, J.M.Pasteels, "How Trail Laying and Trail Following Can Solve Foraging Problems for Ant Colonies," in *Behavioural Mechanisms of Food Selection*, R.N.Hughes ed., NATO ASI Series, Vol. G 20, Berlin:Springer-Verlag, 1990.

[13] J.H.Holland, *Adaptation in Natural and Artificial Systems*, Ann Arbor, MI: The University of Michigan Press, 1975

[14] S.Kirkpatrick, C.D.Gelatt, M.P.Vecchi, "Optimization by Simulated Annealing" *Science* , vol.220, pp.671-680, 1983.

[15] E.L.Lawler, J.K.Lenstra, A.H.G.Rinnooy-Kan, D.B.Shmoys eds., *The Travelling Salesman Problem*, New York:Wiley, 1985.

[16] S.Lin, B.W.Kernighan, "An effective Heuristic Algorithm for the TSP," *Operations Research*, vol.21, pp.498-516, 1973.

[17] C.Peterson, "Parallel Distributed Approaches to Combinatorial Optimization: Benchmark Studies on Traveling Salesman Problem," *Neural Computation*, vol.2, pp.261-269, 1990.

[18] I.Rechenberg, *Evolutionsstrategie*, Fromman-Holzbog, Stuttgart, Germany, 1973.

[19] D.E.Rumelhart, J.L.McLelland, *Parallel Distributed Processing: Explorations in the Microstructure of Cogniton*, Cambridge, MA: MIT Press, 1986

[20] D.Whitley, T.Starkweather, D.Fuquay, "Scheduling Problems and Travelling Salesmen: the Genetic Edge Recombination Operator," *Proc. of the Third Int. Conf. on Genetic Algorithms*, Morgan Kaufmann, 1989.

For this question, please refer to the attached paper, "An exploratory study of customers' perception of company web sites offering various interactive applications: moderating effects of customers' Internet experience" by H. Nysveen and P.E. Pedersen.

a. Please briefly summarize the paper's theoretical foundation, research design, and findings.
b. Please elaborate on the theoretical relationship between the paper's research model and the Theory of Planned Behavior (Ajzen 1985).
c. In this research, experience is considered as a moderating variable. Based on the Social Cognitive Theory (Bandura 1986), what can be done to improve the research model and design reported in this paper?

Ajzen, I. "From Intentions to Actions: A theory of Planned Behavior," in *Action-Control: From Cognition to Behavior*, J. Kuhl and J. Bechmann(ed.), Springer, Heidelberg, 1985.

Bandura, A., *Social Foundations of Thought and Action*, Prentice Hall, Englewood Cliffs, NJ, 1986.

Available online at www.sciencedirect.com
SCIENCE DIRECT®
Decision Support Systems 37 (2004) 137–150
Decision Support Systems
www.elsevier.com/locate/dsw

An exploratory study of customers' perception of company web sites offering various interactive applications: moderating effects of customers' Internet experience

Herbjørn Nysveen[a,*], Per E. Pedersen[b]

[a] Norwegian School of Economics and Business Administration, Breiviksveien 40, 5045 Bergen, Norway
[b] Agder University College, Grooseveien 36, 4876 Grimstad, Norway

Received 1 October 2002; accepted 4 December 2002

Abstract

Often interactive applications as customer communities and personalization are implemented on company web sites to support the customers. This article is based on the assumption that users' general experience with the Internet moderates the effects of these applications. The focus of the article is on the following variables: (1) customers' perception of how easy it is to use a web site, (2) customers' perception of the usefulness of a web site, and (3) customers' attitude to using a web site when the web site contains interactive applications such as personalization or customer community services. Results from an experiment show how general Internet experience moderates the effects of interactive applications.

Keywords: Internet experience; Technology acceptance; Community; Personalization

1. Introduction

An important research issue is how the content and presentation of product information affect consumers' willingness to make choices in on-line environment [3]. Even though a rather generic picture has been given of how the content and the information should be presented on company web sites, the importance of including interactive applications such as personalization and customer community services has been discussed by several authors [5,25]. A precondition for customers' use of a company web site is that it offers services that are valued and preferred by potential customers. These services should be professional and useful, and they have to be presented in an intuitive and easy to use manner. A study by Liang and Huang [24] found that experienced and inexperienced Internet customers made different considerations when shopping electronically. Furthermore, Bruner and Kumar [7] found positive effects of Internet experience on users' attitudes to web sites. Therefore, it seems reasonable to argue that the perception and use of a company web site including interactive applications may depend on the individual characteristics of the web site users. Consequently, our purpose is to

* Corresponding author. Tel.: +47-55-95-95-37; fax: +47-55-95-95-40.
E-mail addresses: herbjorn.nysveen@nhh.no (H. Nysveen), per.pedersen@hia.no (P.E. Pedersen).

0167-9236/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/S0167-9236(02)00212-9

study the effects of general Internet experience on the perception of web sites containing different types of interactive applications. More specifically, we focus on company web sites containing personalized and community services.

A general model applied to studies of individual perceptions, attitudes and behavior when using information systems is the technology acceptance model (TAM) developed by Davis [11,12]. The presumption of this model is that perceived ease of use and perceived usefulness are important determinants of the attitudes to using information systems, and consequently, the use of information systems. The technology acceptance model is used as a framework in this study. First, we discuss the general effects of user experience with information technology on the effectiveness of information systems. This discussion also reviews studies of the effects of Internet experience. We then discuss interactive applications as value-added services on company web sites. Next, the technology acceptance model is introduced and used as an evaluation framework for this study, and hypotheses are presented based on this framework. Finally, we present the results of an experimental study testing these hypotheses, and the implications of the results are discussed. Thus, the main contribution of this article is to study effects of value-added services offered on specific company web sites among users with high and low general Internet experience, effects that we so far do not know much about, within an established theoretical framework (TAM).

2. Internet experience

In this article, Internet experience is defined in a broad sense. By visiting several web sites and using various value-added services offered on a broad range of web sites, users get a broad and general Internet experience. Internet experience is defined here as such general experience with web sites and not as experience with one particular web site.

We argue that the acceptance of company web sites will vary across user segments. When designing a communication system, marketers should pay attention to the customers' experience with the same and similar systems [4]. Studies undertaken in organizational contexts indicate that education and training in using information technology have positive effects on the users' attitude to information systems and performance [9]. This suggests that increasing user experience makes users more capable of taking advantage of an information system. Experience is assumed to increase users' confidence in their ability to master and use computers supporting their task performance [15,22]. However, Agarwald et al. [2] argue that effect of experience may not be universal. They found that structured learning experience were more beneficial than self-training experience among individuals with less than 4-year college degrees.

Although not universal, results from information system studies generally indicate a positive relationship between information system experience and ease of use, usefulness, and attitude to using an information system. Even though one should be careful in generalizing these results to Internet-based applications, Bruner and Kumar [7] argue that web sites that appear complicated to customers with low Internet experience are probably not that difficult to handle for customers with high Internet experience. They also found empirical support for positive effects of general Internet experience on users' attitudes to the web site. Liang and Huang [24] used a transaction cost model and found that inexperienced on-line customers were concerned with uncertainty and asset specificity while experienced on-line customers were concerned with uncertainty when purchasing electronically. The results from both these studies indicate that Internet experience is important in understanding customers' perceptions, attitudes, and behavior in on-line environments.

3. Interactive applications

Company web sites offer a wide range of information services and interactive applications to their customers. Examples of such applications are decision support, customer communities, personalized services, and push-based services. In this article, we focus on company web sites offering interactive applications in the form of personalization and customer community services. However, static web sites will be used for benchmarking the effectiveness of web sites offering interactive applications. Therefore, we also give a brief description of static web sites.

3.1. Static web sites

Many company web sites today do not offer any interactive services. Such web sites are often called static web sites, in contrast to web sites offering interactive applications. Static web sites are characterized by static text and pictures. Often the postal address, the telephone number and the fax number of the company offering the web site are presented on the site. In addition, an e-mail address is often offered as contact information. Static web sites can be described as a brochure distributed on the Internet. If shopping is made possible through static company web sites, this is often implemented as an Internet version of a mail order catalogue.

3.2. Personalized web site

"Web-based personalization involves delivering customized content for the individual, through web pages, e-mail, or push technology" [8,p.299]. Personalized services are often based on machine interactivity. This means interactivity *with* the medium [18]. The personalization is undertaken automatically by some technology, not by a person. Mainly, personalized services are based on a user profile or on user identification. By asking customers about their preferences for services the first time the customers visit the web site, it is possible for the company—for example by registration of username and password—to offer services that are in accordance to the individual visitor's personal preferences. Another method is to recognize the customers when they visit the web site—for example by the use of cookies. Based on the individual visitors' behavior history, it is possible to offer matching services. This implies offering services similar to previously preferred services.

By personalizing services, company web sites offer information and services that are more relevant to their individual customers' preferences and profiles. Personalized web sites help customers find the products and services they prefer. A personalized web site is like a dedicated assistant who knows your taste well and makes your choice more effective [17]. An important reason why customers revisit a web site is that it represents an instrument for maintaining company relationships. Much of the perceived value of these relationships comes from personal and customised service [21]. Personalized services also help customers navigate through myriads of content and shopping options [27] because this content is customized for the individuals' preferences and needs.

3.3. Web site with community

A community is "an Internet-based forum for special interest groups to communicate" [8,p.313]. Communities are based on what Hoffman and Novak [18] call person interactivity. This means interactivity between customers or between customers and a company *through* the Internet. Communities are supposed to serve customers' need for communication, information, and entertainment [5]. Usually, communities allow for communication both between customers and between customers and the company offering the community. This makes communities suitable for customers seeking the advice of other customers before they buy anything from the company web site. In addition, communities make it possible to communicate with the company offering products on the web site, asking questions about the products quality, and so on.

When customers shop for products and services, they often seek advice from others before they buy [5]. From a user perspective, direct communication between users therefore increases the perceived value of the web site [26]. Offering a customer community is believed to be among the main determinants of capturing and retaining customers on a company web site. Furthermore, a community is of great value in generating revisits to a web site [8]. A study by Dellaert [14] found that tourists' valuations of other tourists' contributions to travel web sites were relatively high compared to other information and services presented on the web site. In general, this makes communities a valuable application on company web sites.

4. Technology acceptance model

The technology acceptance model [11,12] has been used by several researchers to explain the attitudes and behavior of information system users. Although the model is mainly applied to explaining the adoption of technology within organizations, the constructs of

4

the model are meant to be fairly general [16]. Davis et al. [13] described the variables of the model as universal to different types of computer systems and user populations. The model may also be applied to explaining individuals' attitudes to using web sites [23]. Consequently, we find the model suitable for studying the effects of introducing interactive applications on company web sites among users with different Internet experience. A modified technology acceptance model for studying these effects is illustrated in Fig. 1.

Perceived usefulness is defined as "the degree to which a person believes that using a particular system would enhance his or her job performance" [11,p.320]. Ease of use is defined as "the degree to which a person believes that using a particular system would be free of effort" [11,p.320]. Attitude to using the system is defined as "the degree of evaluative affect that an individual associate with using the target system in his job" [12,p.476]. Doll et al. [16,p.847] have modified these definitions somewhat. They argue that information systems will be useful in general if they "contribute to accomplishing the end-user's purpose". Another perspective on the meaning of the "usefulness" construct is that an information system is useful "to which a potential adopter views the innovation as offering value over alternative ways of performing the same task" [1,p.365]. Both these modifications indicate that the model is suitable for studying different kinds of information systems, including systems that are not directly related to job situations. A study by Jung and Butler [20] found that useful information and ease of use were among the most important variables for a successful web site design in the tourism industry. Consequently, the technology acceptance model is considered relevant in studying the acceptance, adoption and use of company web sites.

As can be seen from Fig. 1, the content of company web sites are assumed to have an effect on the users perception of how easy the web site is to use. Furthermore, the model postulates that the perception of a web sites' usefulness will vary across different web sites. Perceived ease of use and perceived usefulness are assumed to have an effect on the attitude to using the web site. These effects are found to be positive, stressing the importance of developing company web sites that are useful and easy to use. The final relation postulated in the model is the positive effect of perceived ease of use on perceived usefulness, indicating that if a company web site is easy to navigate, the user is more likely to take advantage of the services offered on the site (perceived usefulness).

5. Hypotheses

In this article, company web sites with interactive applications are compared to static web sites. Although we have discussed two sorts of interactive application representing different types of interactivity [18], we do not discriminate theoretically between the two types in our hypotheses. However, we discuss the implications of introducing applications with different types of interactivity in the Results section of the article.

Static web sites represent web sites that do not offer any kind of interactive applications except e-mail. They are more or less "brochures" distributed on the Internet. By introducing interactive services such as communities and personalization, the users have to invest more effort to fully take advantage of the services offered on the company web site. This

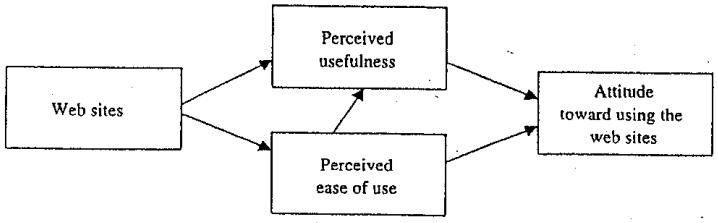


Fig. 1. Technology acceptance model.

includes reporting personal preferences, navigating a community, and often taking care of usernames and passwords. However, the effort required for using personalized services and communities on a company web site is relatively small. The use of interactive applications is voluntary, and other services on the web site can be utilized without using the interactive applications. Windham and Orton [27] argue that personalized services will help customers navigate a web site because the site is adapted to the individual preferences and needs. Furthermore, personalized web sites help customers find the products and services they prefer [17]. Thus, including these interactive applications should have a positive effect on perceived ease of use.

As discussed in the theory section of this article, both community and personalized services are supposed to offer value-added services to the customers. This includes delivering customized content for the individual, and the possibility for customers to seek advice from other customers before they buy anything from the company web site. In addition, Dellaert [14] found that information from communities were valued relatively high compared to other information presented on a web site. Therefore, it is reasonable to assume that these services are perceived as useful among customers. Consequently, interactive applications will make web sites more easy to use and more useful. Perceived ease of use and perceived usefulness are predicted to have positive effects on attitude to using a web site. Thus, we propose that interactive applications, through the mediating variables perceived ease of use and perceived usefulness, will have a positive effect on the attitude to using a web site. This is summarized in the following hypothesis.

Hypothesis 1. Customers using web sites with interactive applications will perceive the company web site as easier to use, will perceive the web site as more useful, and will have a more positive attitude to using the web site than customers using a static web site.

By visiting several web sites and using value-added services offered on a broad range of web sites, the user gets a broad and general Internet experience. It is often argued that prior learning or general experience can influence the performance of later activities on related areas, often referred to as transfer of training [19]. Web sites accessible on the Internet today are often based on some of the same applications. Although the web sites differ in design, the value-added services and the structure of navigation can to some extent be recognized across various web sites. Thus, it can be argued that users with a general Internet expertise are more able to use new web sites that they visit and to take advantage of value-added services offered on these web sites due to the experience they have on using other web sites. "Responses to new situations are assumed to be based on assimilating the new to a previously learned situation, and giving a response based on the similarity or analogy of the two situations" [6,p.28]. In general, therefore, Internet experience contributes to more effective use of web site applications. Thus, it is expected that experienced Internet users perceive a company web site as easier to use, that experienced Internet users are more capable of taking advantage of the information and services offered on a web site (usefulness), and consequently, have more positive attitudes to using a web site. Based on these arguments, we present the following general hypothesis.

Hypothesis 2. Customers with high Internet experience will perceive a web site as easier to use, will perceive the web site as more useful, and will have more positive attitudes towards using the web site than customers with low Internet experience.

Web sites offering community and personalized services have the potential to offer easier navigation and more useful information than static web sites. However, due to the effort needed to take advantage of these applications, they may be more effective for customers with high Internet experience than for customers with low Internet experience. As argued by Ariely [4], giving users increasing control of the information flow on a web site "creates demands on processing resources and therefore under some circumstances can have detrimental effects on consumers' ability to utilize information" [4,p.233]. One of the requirements for taking advantage of increasing information control is interface experience [4,p.235]. As users with more Internet experience will have more experience with the interface offered by interactive applications, they should be better able to take advantage of web sites offering interactive applications than inexperienced users. We therefore argue

6

that Internet experience moderates the effects of interactive applications on the variables included in the technology acceptance model. This is summarized in the following hypothesis.

Hypothesis 3. Internet experience moderates the effects of introducing interactive applications in company web sites on customers' perception of ease of use, customers' perception of usefulness, and customers' attitudes towards using the web site.

6. Methodology

To test the causal relations of these hypotheses, an experimental study was chosen. We study the effects of interactive applications, specifically customer communities and personalized services. It was decided to benchmark the effects of community services and personalized services against a static web site. The effects of the interactive applications were studied for two levels of Internet experience, high Internet experience and low Internet experience. Thus, a 3*2 between subjects design was used for testing the hypotheses. The whole experiment was undertaken on the Internet.

To ensure valid results across company product categories, the hypotheses were tested on both an airline and a restaurant web site. The product brand names were designed for the purpose of the experiment. The airline was named Blue&Gold Air while the restaurant was named The Blue&Gold. Even though the names were made up for the purpose of the experiment, the subjects were told that these were new companies recently introduced on the market, and that the brands would be available for them in the near future.

Fig. 2 illustrates how the web sites of the two companies were presented to the subjects. As can be seen, the company web sites were designed as similar as possible to prevent web site design from affecting the results.

A total of six company web sites were developed for the purpose of the experiment. The web sites of both the airline and the restaurant were developed in three versions. The first version was a static site where e-mail was the only possibility for two-way communication between the customer and the company. The second version of the web site included personalized services in addition to the services included in the static version. The third version of the web site included community services in addition to the services offered by the static version. None of the web sites offered both community services and personalized services. All web sites included "shopping" functionality in the form of a flight reservation system for the airline and a table reservation system for the restaurant.

The only difference between the versions of the web sites is that the menu button "Discussion" at the web site offering community services was termed "My Blue&Gold" at the web site offering personalized services. For the static web site, neither the "Discussion" nor the "My Blue&Gold" menu buttons were offered. In Fig. 3, the design of the community services and the personalized services are superficially illustrated. Except from these differences, the three versions of the web site for each company were identical.

6.1. Procedure

To ensure variation in Internet experience, the subjects were recruited among tourism students and employees at travel agencies. A somewhat skewed distribution of male and female was found in the sample with 35% men among the subjects. The age of the subjects varied from 18 to 64 years with an average age of 30 years. However, neither age nor sex had any significant effect on the dependent variables measured in the technology acceptance model, indicating no validity threats from these variables.

The experiment lasted for 10 days, and included several events that required interaction between the company and the subjects. The events are illustrated in Fig. 4.

All subjects were briefed about the experiment before it started. In the brief, the importance of visiting the web site at least once every day during the experiment was stressed. Furthermore, the subjects were told that events would be implemented, and that all instructions necessary to participate in the experiment would be given on the web site during the experiment. This brief was also used for the allocation of subjects into the six groups of the experimental design. The subjects were given an envelope containing an URL for their

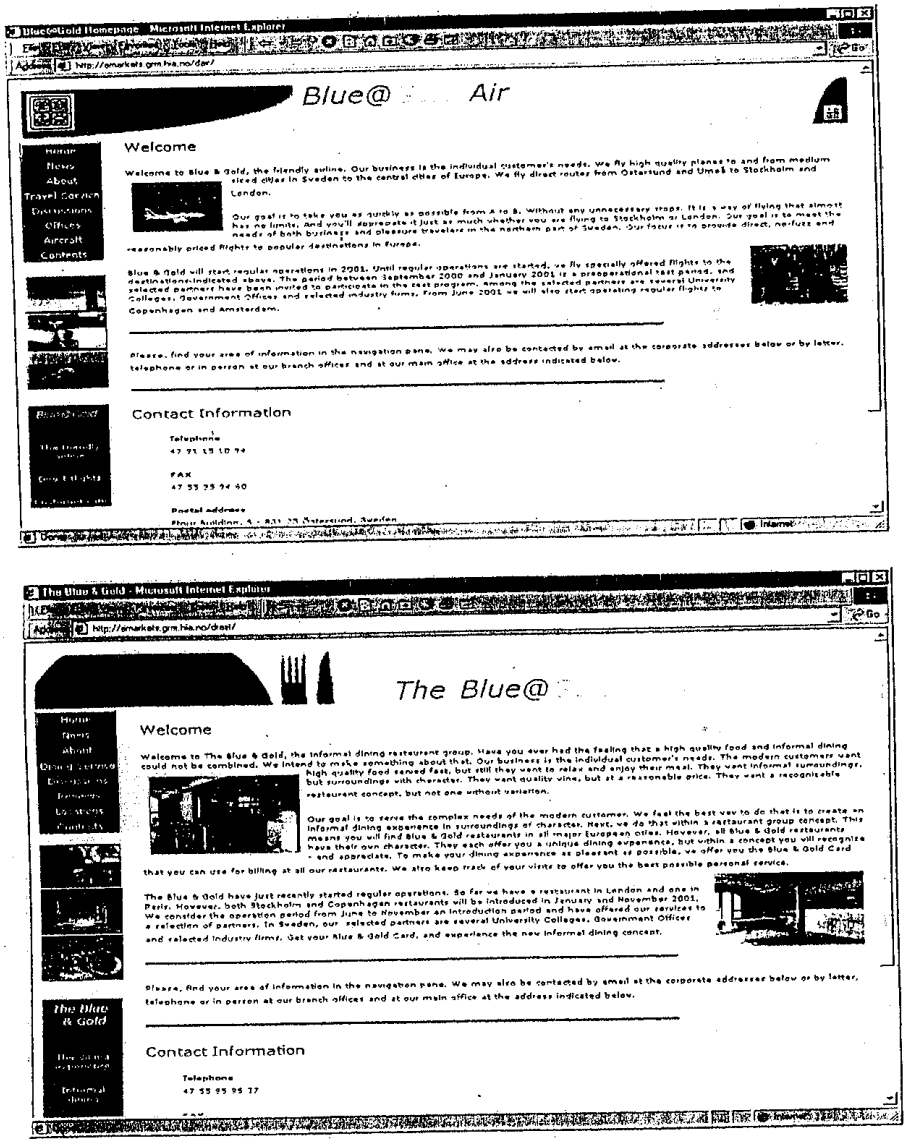


Fig. 2. Illustration of web sites for Blue&Gold Air and The Blue&Gold.

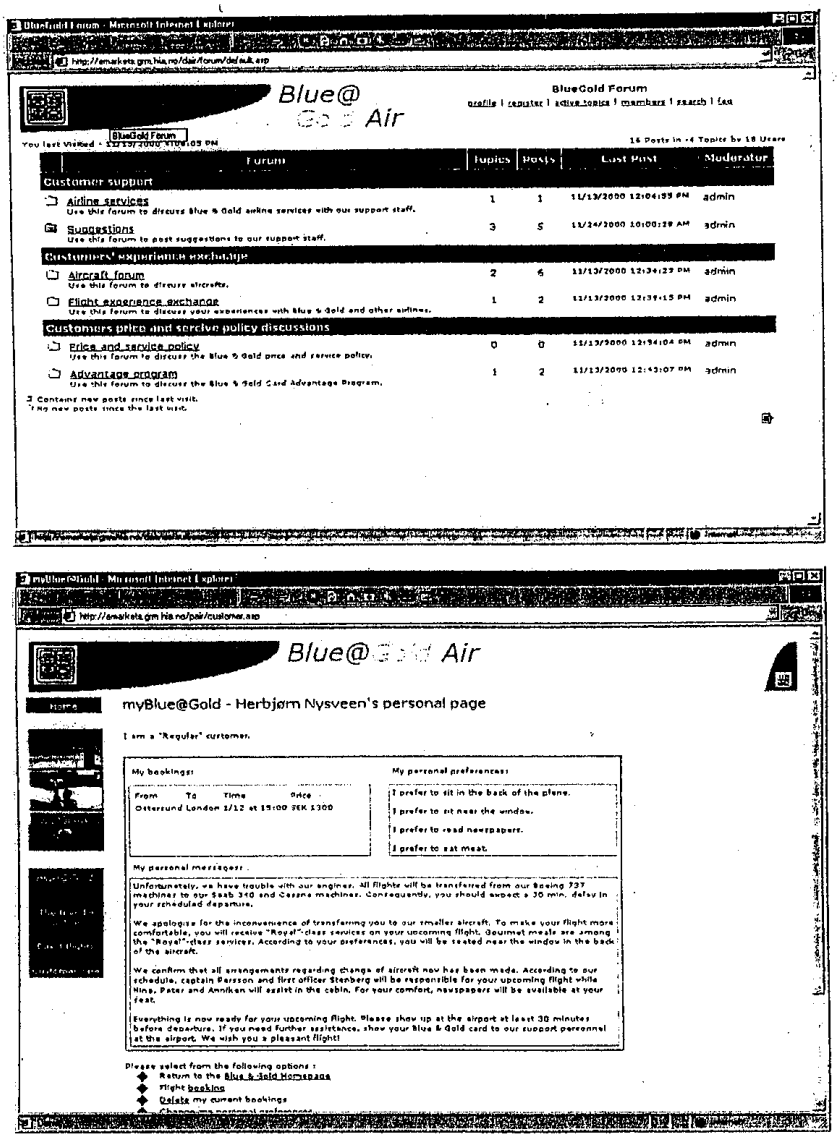


Fig. 3. Illustration of community and personalization at Blue&Gold Air.

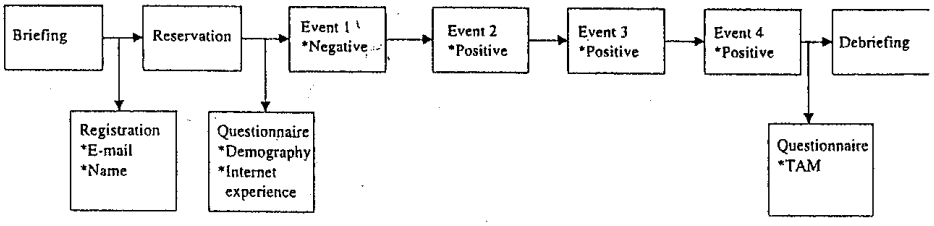


Fig. 4. Experimental procedure.

allocated company web site, a username, a password, and a personal account number for the purpose of reserving flights at the airline or tables at the restaurant. In addition, the envelope contained an explanatory letter. For the airline groups, the letter instructed the subjects to make a reservation with Blue&Gold Air for a roundtrip to London. In the restaurant groups, subjects were instructed to make a reservation at the Blue&Gold restaurant in London. The envelopes were distributed by a procedure ensuring random allocation of subjects in each of the six groups. The usernames and passwords prohibited the subjects from visiting other web sites than the one received in the envelope. Furthermore, the subjects were instructed not to discuss the experiment with other potential subjects for the duration of the experiment. Manipulation check indicated little understanding of the true purpose of the experiment among subjects.

The first time the subjects visited the web sites they were instructed to register by e-mail and name. They were invited to explore the company web site thoroughly, and to make the reservation described in the letter given to them. This reservation had to be confirmed with the correct account number. Subjects in the two groups with personalized services also had to fill in a form registering personal preferences. The preference questions are shown in Table 1. After the reservation was made, the subjects answered the first questionnaire measuring demographic characteristics and Internet experience.

In the series of events, the first was of a negative character. The purpose of this event was to generate activity among the subjects making them utilize the interactive applications offered on the company web sites. The next three events had a positive character. The messages related to events two and three were personalized for the subjects in the two personalized groups. All messages are found in Table 2. After the fourth event, a questionnaire was presented to the subjects measuring the variables of the technology acceptance model. Any questions or other responses sent to the companies by e-mail were answered by the authors as "representatives" of the companies using predefined neutral templates.

6.2. Measures

The measures of ease of use, usefulness and attitude to use were based on the original items used by Davis [11], and adapted for this web based experiment. As shown in the factor analysis of Table 3, the items used for measuring the variables in the technology acceptance model loaded as expected. All measures were based on 7-point scales. Cronbachs alpha exceeded 0.94 for all three variables, indicating satisfactory reliability. Crossloadings were under 0.36, indicating satisfactory convergent and discriminant validity. As shown in Table 3, significant correlations

Table 1
Questions for personalized profile

Airline
(1) Where do you prefer to be seated? (Back, Middle, Front)
(2) Window or aisle seating? (Window, Aisle)
(3) What do you prefer to read? (Newspaper, Business magazines, Lifestyle magazines)
(4) What is you dining preference? (Meat, Seafood, Vegetarian)
Restaurant
(1) Smoking preferences? (Non-smoking, Smoking)
(2) Music preferences? (Background classics, Silent dining)
(3) What kind of apertif do you prefer? (Gin&Tonic, Bitter, Wine)
(4) Are you particularly allergic to any ingredients (None, Seafood, Nuts, Milk or Eggs)

Table 2
Events

Airline	Restaurant
Event 1 Unfortunately, we have trouble with our engines. All flights will be transferred from our Boeing 737 machines to our Saab 340 and Cessna machines. Consequently, you should expect a 30-min delay in your scheduled departure.	Unfortunately, there is a problem with your upcoming dining reservation. Your table preferences could not be met, and you should expect a 30-min delay in your scheduled reservation time.
Event 2 We apologize for the inconvenience of transferring you to our smaller aircraft. To make your flight more comfortable, you will receive "Royal"-class service on your upcoming flight. Gourmet meals are among the "Royal"-class services. Personalized group: According to your preferences, you will be seated near the window in the back of the aircraft.[a]	We apologize for the problems with your upcoming reservation. To make your dining experience as pleasant as possible, seats will be reserved for your party in the bar where we will serve you a free apertif while you wait to be seated. Personalized group: According to your preferences, Gin&Tonic will be served and seats will be available in the non-smoking area of the bar.
Event 3 We confirm that all arrangements regarding change of aircraft now has been made. According to our schedule, captain Persson and his first officer Stenberg will be responsible for your upcoming flight, while Nina, Peter and Anniken will assist in the cabin. Personalized group: For your comfort, newspaper will be available at your seat.	We confirm that all arrangements regarding change of reservation now has been made. According to our schedule, managing chef will be Christian Courtot, while Nina and Peter will serve your table. Personalized group: For your comfort, you will be seated in the non-smoking area of the restaurant.
Event 4 Everything is now ready for your upcoming flight. Please show up at the airport at least 30 min before departure. If you need further assistance, show your Blue&Gold card to our support personnel at the airport. We wish you a pleasant flight.	Everything is now ready for your upcoming dining arrangement. The restaurant can be reached using both bus and Underground. Use the London Bridge over Tower Hill Stations. We will do our best to make your visit an unforgettable experience.

[a] The personalized messages varied according to the customers profile. The example messages shown are for customers who prefer to sit near the window in the back of the aircraft.

were found among the three variables. However, similar correlations seem to be common in studies based on the TAM (see for example Ref. [2], Study 1).

Internet experience was measured by the subjects' response to the expression: "I feel that I am an experienced user of Internet". There were no differences between the level of Internet experience among respondents exposed to the airline and restaurant web sites ($F=0.07/p=0.80$), between the two groups of respondents, students and employees at travel agencies ($F=2.96/p=0.09$), or between respondents exposed to the three versions of the web site-static, community, and personalization ($F=0.46/p=0.63$). The mean value of Internet experience among the subjects was 5.07. A binary scale was designed with values from 1 to 4 categorized as low experience and values from 5 to 7 as high experience.

7. Results

Hypothesis 1 focused on the main effects of the interactive applications on the users' perception of ease of use, usefulness, and attitudes to using the company web site. The MANOVA-test showed no main effects of the interactive application on subjects' perception of how easy it is to use the web site, their' perception of the web site usefulness, or their attitudes to using the web site. The results are presented in Table 4.

The results imply lack of support for Hypothesis 1. Interactive applications implemented on a company web site do not seem to have any main effect on the perception of the web site or users' attitudes to using it.

Hypothesis 2 predicted main effects of Internet experience on the users' perception of ease of use,

Table 3
Factor analysis

Eigen value	Usefulness (10.24)	Ease of use (2.09)	Attitude to use (1.42)
*Using the Blue&Gold web site enables me to establish my relation to The Blue&Gold more quickly	0.79		
*Using the Blue&Gold web site improves my relation to the Blue&Gold	0.79		
*Using the Blue&Gold web site increases the quality of my relation to The Blue&Gold	0.82		
*Using the Blue&Gold web site enhances the effectiveness of my relation to the Blue&Gold	0.81		
*Using the Blue&Gold web site makes it easier for me to have a relation to The Blue&Gold	0.79		
*The Blue&Gold web site is useful in my relation to The Blue&Gold	0.77		
*Learning to operate The Blue&Gold web site is easy for me		0.77	
*I find it easy to get The Blue&Gold web site to do what I want it to		0.74	
*My interaction with The Blue&Gold is clear and understandable		0.78	
*I find The Blue&Gold web site flexible to interact with		0.71	
*It is easy for me to become skillful at using The Blue&Gold web site		0.70	
*I find The Blue&Gold web site easy to use		0.78	
*All things considered, using The Blue&Gold web site in my relation to The Blue&Gold is			
Good–bad			0.75
Wise–foolish			0.83
Favorable–unfavorable			0.91
Beneficial–harmful			0.75
Positive–negative			0.82
Chronbachs alpha	0.95	0.94	0.94

Correlations

	Ease of use	Usefulness	Attitude to use
Ease of use			
Usefulness	0.69**		
Attitude to use	0.61**	0.53**	

**$p<0.01$.

Table 4
Results—Hypotheses 1, 2 and 3 (MANOVA)

	F	p
Application		
Ease of use	0.87	0.42
Usefulness	1.56	0.21
Attitude to use	0.25	0.78
Internet experience		
Ease of use	1.30	0.26
Usefulness	0.02	0.89
Attitude to use	0.07	0.79
*Application*Internet experience*		
Ease of use	2.11	0.13
Usefulness	5.09	0.01
Attitude to use	0.18	0.83

usefulness, and attitude to using a web site. As can be seen from Table 4, the results show no main effects of Internet experience on the perception of how easy it is to use a web site, its usefulness, or the attitudes to using the web site. These findings indicate that Internet experience in general does not have an effect on users' perception of company web sites and the attitudes to using such sites. The results do not support Hypothesis 2.

In Hypothesis 3, we predicted that Internet experience moderates the effects of interactive applications offered on company web sites. Internet experience was proposed to be more important for taking advantage of web sites with interactive applications than for static web sites. The results presented in Table 4 show

12

moderating effects of Internet experience on one of the three variables; perception of web site usefulness. Among users with low Internet experience, web sites offering community is perceived as more useful than static web sites ($p=0.01$). The prediction in Hypothesis 3 was that web sites with interactive applications should be more useful for customers with high Internet experience than for customers with low Internet experience. Thus, the result contrasts what was predicted in Hypothesis 3.

Further analyses were undertaken to study the differences between high and low general Internet experience within each of the three types of web sites using separate one-way ANOVA tests of differences between means (Table 5). For static web sites, the analyses revealed that customers with high general Internet experience perceived static web sites used in this study as easier to use ($p=0.02$) and more useful ($p=0.01$) than customers with low general Internet experience. No differences were found for attitude towards using a web site. For the web sites offering community services, the analyses indicated that customers with low general Internet experience perceived web sites with community services used in this study as more useful than customers with high general Internet experience ($p=0.09$). No differences were found for perceived ease of use and attitudes to using the web sites. For web sites offering personalized services, no differences were found between customers with high general Internet experience and customers with low general Internet experience.

Table 5
Comparison across Internet experience (one-way ANOVA)

	Static	Community	Personalized
Ease of use			
High experience	5.63	5.39	5.50
Low experience	4.82	5.43	5.57
F	5.65	0.11	0.06
p	0.02	0.92	0.81
Usefulness			
High experience	5.13	4.78	4.81
Low experience	4.20	5.46	4.97
F	7.00	2.96	0.23
p	0.01	0.09	0.64
Attitude to use			
High experience	5.40	5.49	5.56
Low experience	5.48	5.28	5.40
F	0.06	0.26	0.01
p	0.81	0.61	0.93

8. Implications

Even though our experimental analysis is exploratory, the results indicate that the implementation of interactive applications on a company web site do not have any main effect on customers' perception of how easy it is to use the site, their perception of the usefulness of it, or their attitude to using such a web site. Introducing such services does not, in general, make the company web site easier to use. It does not make the web site more useful to all customers, and it does not, in general, make the customers' attitudes towards using the web site more positive. The implication of this is that investment in interactive application on a web site does not seem to have any uniform and general added value for all customers.

Furthermore, the results indicate that general Internet experience does not have any main effect on customers' perception of how easy it is to use the company web site, their perception of the usefulness of the site, or their attitudes towards using the web site. These results contrasts earlier studies that have found support for a general positive effect of information system experience on the variables included in the technology acceptance model. However, it should be noted that Internet experience typically is based on self-training experience, a type of experience found to be less beneficial than structured learning experience [2]. This may explain the lack of support for the predicted effect of Internet experience. It may also be argued that users with high Internet experience are more likely to detect drawbacks of a web site than users with low Internet experience. Thus, this effect can outweigh the positive effect of Internet experience argued for in this article and explain the lack of support for positive effects of Internet experience on TAM.

From these findings, one must not conclude that interactive applications implemented on web sites or customers' Internet experience are of no importance for the effectiveness of company web sites. Results from our study clearly show moderating effects of general Internet experience on the effect of interactive application on the variables of the technology accept-

ance model. Even if static company web sites are the simplest forms of web sites, it is no surprise that customers with high Internet experience perceive these web sites as easier to use and more useful than customers with low Internet experience. However, it is surprising that customers with low Internet experience perceive web sites offering community services as more useful than customers with high Internet experience. A possible explanation of this result is that customers with high Internet experience search for supplementary information on other web sites on the Internet rather than using the community, while customers with low Internet experience perceive the community as a useful application for supplementary information. Anyway, the implication of the result is that companies should study the characteristics of their customers' general Internet experience carefully before deciding which interactive application to introduce on their company web site. More specifically, a community service should be implemented on the web site if the company serves a segment characterised by low Internet experience.

9. Further research

According to Hanson [17], excessive personalization is cumbersome, confusing and wastes consumer time. Personalization must provide added value to the customer, but for routine and simple products, standardization may suffice and personalization provides little added value. Reservation of airline tickets and tables at a restaurant may be simple products. The average product experience of the subjects of this experiment was 4.99 on a 7-point scale. This indicates that the subjects were familiar with these products. This may explain why community and personalized services did not give users added value when compared to a static web site. Further research should focus on the effects of introducing interactive applications on the company web sites of providers of more complex products.

Results from the experiment reported here show moderating effects of Internet experience on interactive applications' effect on the variables of the technology acceptance model. These results should be noticed by Internet researchers. The significant effect of Internet experience means that this variable should be included as a potential moderating variable in studies focusing the effectiveness of company web sites. For research focusing the effects of other variables than Internet experience on web site effectiveness, Internet experience should be included as a covariate.

Using experiments for studying the effects of stimuli on dependent variables is superior to other methods because of experimental control [10]. The results reported in this article are based on an experiment undertaken on the Internet, and have given priority to experimental control and internal validity. Thus, the experimental setting may have affected the subjects' perception of the company web sites' usefulness and their attitudes to using the web site, due to the artificial brands used in the experiment. An interesting path for research is to study the hypotheses in a more realistic setting, using the personalized and community services on the company web sites of real brands.

Acknowledgements

This research has been funded by the European Tourism Research Institute (ETOUR) in Östersund, Sweden. The authors also want to thank the reviewers of this article for useful comments.

References

[1] R. Agarwald, J. Prasad, Are individual differences germane to the acceptance of new information technologies? Decision Sciences 30 (2) (1999) 361–391.
[2] R. Agarwald, J. Prasad, M.C. Zanino, Training experiences and usage intentions: a field study of a graphical user interface, International Journal of Human Computer Studies 45 (1996) 215–241.
[3] J. Alba, J. Lynch, B. Weitz, C. Janiszewski, R. Lutz, A. Sawyer, S. Wood, Interactive home shopping: consumer, retailer, and manufacturer incentives to participate in electronic marketplaces, Journal of Marketing 61 (1997, July) 38–53.
[4] D. Ariely, Controlling the information flow: effects on consumers' decision making and preferences, Journal of Consumer Research 27 (2000, September) 233–248.
[5] A. Armstrong, J. Hagel, The real value of on-line communities, Harvard Business Review (1996, May–June) 134–141.
[6] G.H. Bower, E.R. Hilgard, Theories of Learning, Prentice-Hall, Englewood Cliffs, NJ, 1981.
[7] G.C. Bruner, A. Kumar, Web commercials and advertising hierarchy-of-effects, Journal of Advertising Research 4 (1/2) (2000) 35–43.

[8] D. Chaffey, R. Mayer, K. Johnston, F. Ellis-Chadwick, Internet Marketing, Financial Times Prentice-Hall, London, 2000.
[9] P.H. Cheney, R.I. Mann, D.L. Amoroso, Organizational factors affecting the success of end-user computing, Journal of Management Information Systems 3 (1) (1986) 65–80.
[10] T.D. Cook, D.T. Campbell, Quasi-experimentation, Design and Analysis Issues for Field Settings, Houghton Mifflin, Boston, 1979.
[11] F.D. Davis, Perceived usefulness, perceived ease of use, and user acceptance of information technology, Management Information Systems Quarterly, (1989, September) 319–339.
[12] F.D. Davis, User acceptance of information technology: system characteristics, user perceptions and behavioral impacts, International Journal of Man–Machine Studies 38 (1993) 475–487.
[13] F.D. Davis, R.P. Bagozzi, P.R. Warshaw, User acceptance of computer technology: a comparison of two theoretical models, Management Science 35 (8) (1989) 982–1003.
[14] B.G.C. Dellaert, Tourists' valuation of other tourists' contributions to travel web sites, in: D.R. Fesenmaier, S. Klein, D. Buhalis (Eds.), Information and Communication Technologies in Tourism 2000, Springer-Verlag, Wien, 2000, pp. 293–302.
[15] W.H. DeLone, Determinants of success for computer usage in small business, MIS Quarterly 12 (1) (1988) 51–61.
[16] W.J. Doll, A. Hendrickson, X. Deng, Using Davis's perceived usefulness and ease-of-use instruments for decision making: a confirmatory and multigroup invariance analysis, Decision Sciences 29 (4) (1998) 839–869.
[17] W. Hanson, Principles of Internet Marketing, South-Western College Publishing, Thompson Learning, USA, 2000.
[18] D.L. Hoffman, T.P. Novak, Marketing in hypermedia computer-mediated environments: conceptual foundations, Journal of Marketing 60 (1996, July) 50–68.
[19] D.H. Holding, Transfer of training, in: J.E. Morrison (Ed.), Training for Performance: Principles of Applied Human Learning, Wiley, New York, USA, 1991, pp. 93–125.
[20] T.H. Jung, R. Butler, The measurement of the marketing effectiveness of the Internet in the tourism and hospitality industry, in: D.R. Fesenmaier, S. Klein, D. Buhalis (Eds.), Information and Communication Technologies in Tourism 2000, Springer-Verlag, Wien, 2000, pp. 460–472.
[21] P. Keen, M. McDonald, The eProcess Edge, Osborne/McGraw-Hill, Berkeley, 2000.
[22] L. Kraemer, J.N. Danzinger, D.E. Dunkle, J.L. King, The usefulness of computer-based information to public managers, MIS Quarterly 17 (2) (1993) 129–148.
[23] A.L. Lederer, D.J. Maupin, M.P. Sena, Y. Zhuang, The technology acceptance model and the world wide web, Decision Support Systems 29 (2000) 269–282.
[24] T.-P. Liang, J.-S. Huang, An empirical study on consumer acceptance of products in electronic markets: a transaction cost model, Decision Support Systems 24 (1998) 29–43.
[25] N. Wells, J. Wolfers, Finance with a personalized touch, Communication of the ACM 43 (8) (2000) 31–34.
[26] G. Wiegran, H. Koth, Custom Enterprise.com, ft.com Pearson Education, London, 2000.
[27] L. Windham, K. Orton, The Soul of the New Customer, Windsor Books, Oxford, UK, 2000.

Herbjørn Nysveen (PhD) is an Associate Professor at the Norwegian School of Economics and Business Administration in Bergen, Norway. His research activities are in the field of marketing communication, customer behavior, relationship marketing, and e-marketing. Nysveen holds a part time position at European Tourism Research Institute (ETOUR) in Östersund, Sweden.

Per E. Pedersen (PhD) is a Professor at Agder University College and Adjunct Professor at the Norwegian School of Economics and Business Administration. His current research interests include consumer behavior and mobile commerce.

15

國立中山大學九十三學年度博士班招生考試試題

科目：資訊管理論文評述(第二節)【資管系選考】

請閱讀所附論文，並回答下列問題：

(1) 請就本篇的主題、研究架構/假說、理論基礎、資料分析、結論與建議中的一些主要問題與缺點，提出你的評論。(25%)

(2) 如果要你就這個相關主題來做一篇研究，你的研究架構（包括模式、假說、資料收集與分析）會如何設計？（並請說明你的邏輯）(25%)

Proceedings of the Third Conference on Sustainable
Development for Management 2001, pp. 233-242

電子商務之消費者行爲研究-以網路書店爲例

摘要

隨著網際網路的風行，「電子商務」一詞備受矚目，已成爲最熱門的管理課題，資訊科技不斷精益求精，網際網路造就了無止盡的網路世界，人們開始使用網路來滿足需求，網路科技快速地進入人們的生活當中，讓人無法漠視網際網路的存在，也使得網路行銷日趨重要。電子商務的服務品質優劣不一，要如何留住消費者，是相當重要的。本研究以網路書店爲例，根據調查顯示，適合在網路銷售的商品中，又以書籍市場的銷售潛力最大。網路書店所銷售之書籍同質性較高，品質較能夠確定，因此適合在網路銷售。但網路書店也有許多缺點亟待改進，除了交易上安全性較容易發生爭議外，尚有運送太慢等問題。且從目前國內網路商店業者經營成效看來，似乎不如預期地成功。本研究希望從消費者的角度出發，探討消費者的購買行爲及網路書店的構面，期能幫助網路商店業者在擬定行銷策略時有所參考的依據。
本研究主要的目的是爲了了解現今網際網路以及網路書店發展的現況，針對不同的消費者來作調查與研究。本研究旨在探討網路商店與消費者購買行爲之關係、探討消費者行爲之人口統計變數、心理變數、及行爲變數等對產品特性之影響。

關鍵字：電子商務、網路行銷、購買行爲、網路書店、消費者行爲

一、緒論

從電腦問世到現在的短短五十年間，人類正遭遇到有史以來最快速、最急劇的生活型態的改變，未來社會走向資訊化是必然的趨勢，也是跨進已開發國家行列的必要步驟。面對資訊化社會的來臨，實有必要了解大眾對電腦科技的接受度，尤其今日的電腦網路已愈來愈普及化，網際網路提供一個可以用來進行廣告、行銷、物品銷售及資訊服務等活動的低成本及快速的管道，而不會因時間與空間的藩籬而受到阻礙及干擾。具體來說，網際網路本身具有：(1)低成本、便捷、迅速通訊功能、(2)無遠弗屆的全球連線、(3)多媒體傳輸資料功能及(4)豐富的資訊資源等四項特質。網路不僅改變人的生活、工作方式、思考模式，它也將改變人們消費的習慣，網路行銷市場已隱然成形，全球的商家都想在這一個新的市場中佔有一席之地。

在電子商務中，線上購物是相當重要的一環，客戶藉由個人電腦撥接或公司提供的專線上網選擇想要的產品，可以享有快速的服務與多樣化的選擇，近年來網路上的電子商店，如雨後春筍般竄出，無論是唱片、書籍、模型、影音光碟等許多琳琅滿目的商品都藉由這個全新的商業環境更拉近了與消費者間的距離，身爲現代人不可不好好面對這樣的衝擊，才能跟上世界的脈動。
電子商務逐漸深入人們的生活，透過虛凝與實體通路的結盟，除了可以開發出新的經營模式外，也加入了人味，宅配的出現便是創造了另一種真人服務的可能，也能更貼近消費者的心。電子商務的服務品質優劣不一，要如何抓住消費者的口味，是相當重要的。以網路書店爲例，根據調查顯示，適合在網路銷售的商品中，又以書籍市場的銷售潛力最大。本研究希望從消費者的角度出發，探討消費者的購買行爲及網路書店的構面，期能幫助網路商店業者在擬定行銷策略時有所參考的依據。因此本研究主要的目的是了解現今網際網路以及網路書店發展的現況，針對不同的消費者來作調查與研究。本研究之研究目的爲(1)研究網路商店與消費者購買行爲之關係。上網購物者對購物網站操作流程的重視程度。(2)探討消費

者之人口統計變數，對產品的價格、屬性和購物環境之間的相關程度，以了解其對於網路商店及購買行為之影響。(3)探討消費者之心理變數，對產品的價格、屬性和購物環境之間的相關程度，以了解其對於網路商店及購買行為之影響。(4)探討消費者之行為變數，對產品的價格、屬性和購物環境之間的相關程度，以了解其對於網路商店及購買行為之影響。

二、文獻探討

資訊科技之普及帶動了資訊化時代的來臨，而網際網路的發展更加速人類文明的進步及工商業社會的競爭，從早期的電子資料交換(Electronic Data Interchange；EDI)、銷售點(Point Of Sales；POS)等技術支援的發展，到目前流行的電子商務話題，均為網路快速發展下的產物。面對電子商務(Electronic Commerce；EC)時代的風潮，不論是哪一種產業的 MIS 單位，都必須思考未來在電子商務環境中，如何由既有的資訊系統，結合最新的網路技術和整體企業資源，進行完善的系統改造，以拓展企業的競爭力，提升上、下游廠商間的合作效益。目前相當重要的企業資源規劃(Enterprise Resource Planning；ERP)、顧客關係管理(CRM)、資料倉儲與供應鏈管理(Supply-Chain Management；SCM)等系統建置，漸漸受到全球企業重視階段，其電子商務、顧客關係管理、供應鏈管理、與企業資源規劃等四個關係密切，環環相扣。其四者之關係圖，如圖 1 所示。

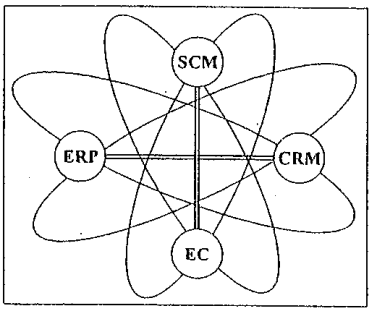


圖 1 EC、CRM、SCM、ERP 之關係圖

2.1 網路行銷

對一個公司而言，一個產品要能夠推廣出去，才能增加它的銷售量。我們很能接受以往叫賣式的推廣，也習慣了一對一的直接銷售，這樣簡明直接的方式，滿足了大部分消費者洽詢的需求。而現今在無國界的消費市場上，這種傳統的一對一行銷方式要如何轉移呢？坐在家裡把電腦打開，「它」會跟消費者一對一交談、「它」會好好的為消費者介紹一件商品、「它」可以為消費者回答所有疑難雜症，而這個「它」就是所謂的「網路行銷」。

「網路行銷」的好處：一、絕對不會讓消費者遭白眼；二、消費者不想聽它說話時，隨時可以把它關掉；三、隨時可以重複聽消費者覺得重要的那幾句話。所有的情況完全由消費者主控，消費者是非常受尊重的[寇世斌，1997]。但傳統的行銷通路在現今的商場上，也並非一點用處都沒有。「無國界世紀的廣告行銷，絕對是傳統行銷通路加上未來的網路行銷」，才是最完美的行銷組合。因為傳統的行銷通路本身，已有自己的一些通路；而網路行銷則是本身就已經等在那邊，等著消費者上門。因此透過一些行銷網路去促銷商品、網站、相關的產品，會有加成的效果。

網路行銷的五“I”原則[張文慧，1998]，分別為 1. 資訊(Information)：資訊是網際網路的主體，資訊的特點是(1)要夠新、(2)要夠充分、(3)內容要正確、(4)要能合乎興趣等、2. 個人化(Individuality)：網

際網路比其他傳播媒體賦予個人更多的控制權、3. 興趣(Interest)：資訊有趣與否，完全在於接收者的主觀認定，所以要想設計出有趣的資訊，首先要瞭解目標對象，要知道其關心的是什麼？感興趣的是什麼？在乎的是什麼？網路資訊是由對象主動找上門的，如果提供的資訊不合其興趣，怎麼也無法留住、4. 互動(Interactive)：全球資訊網的特性之一就是能與瀏覽者互動，亦即藉由與瀏覽者之間的「問」與「答」，讓來訪者產生一種熟悉感、親切感，並由問答回應中瀏覽者有一種參與感、主控感、5.融合(Integration)：網路行銷是整體行銷活動中的一環，它與傳統媒體上的活動應緊密結合，去達成一個共同的行銷目標。

2.2 消費者行爲之探討

經濟社會的快速變遷，每一個人每天都面臨購買與消費的事件，而此是生活的重心與共同點消費者的行爲是有目的，且是目標導向的，產品或服務之被接受與否，是基於與需求和生活方式關聯的程度。對企業而言，「消費者至上」是一無可避免的挑戰，產品與服務針對消費的需要與期望來設計提供，並以純熟的行銷技巧來影響消費者的動機與行爲。本研究以消費者行爲之人口統計變數、心理變數、及行爲變數等三構面來探討對產品特性之影響。而(1)人口統計變數包含了年齡、性別、所得、職業、教育程度等因素。由於消費者行爲和人口統計變數有極大的關連，因此人口統計變數最常被當成區隔消費市場的基礎。(2)心理變數包括了消費者上網動機[Richmond,1996]、消費者個人化服務[Eighmey，1997]、書評和文摘及知覺風險[古雅慧，1996]等因素。網路這種個人化色彩濃厚的傳遞資訊方式，已經會影響到人的心理層面。(3)行爲變數包含了過去經驗[林祖德，1997]、促銷敏感度、上網頻率、要求購物的便利性[徐椿輝，1997]、及每次消費金額[Richmond,1996]等因素。行爲性市場區隔是根據購買者對產品的知識、態度、使用與反應等行爲，將市場區分成不同群體。

除了上述網站消費者行爲因素之研究外，另一些研究乃針對網站產品特性整體評估，而產品特性又分爲產品價格、產品的屬性、及產品的購物環境等三方面來評估。而(1)產品價格包含價格是否比傳統書店便宜、運費的高低、產品的促銷活動、及商品規定加價金額等因素。(2)產品的屬性包括廠商之形象、信譽、書籍種類、送貨速度、及售後服務等因素。(3)產品的購物環境包含是否成爲會員才能訂購、訂購流程、所提供的功能、所提供的產品或服務、及傳輸的速度等因素。

2.3 網路書店特性

(一)陳列空間無限延伸，包容多元化書籍類型：以國內的博客來網路書店而言，十多萬筆的書目以業界平均值換算，也需要五百坪賣場的規模，加上歷年出版的書籍，沒有任何一家書店能陳列全部的圖書。相較之下，網路書店陳列空間可無限延伸，不受時間、空間限制。以 Amazon 爲例，目前超過三百萬的圖書書目(尚不包括音樂光碟類商品)，至少是全球最大書店陳列量的七倍，顧客幾乎可以一次購足所需的圖書。

(二)檢索功能與線上交易系統降低退貨率：以資料庫導向的網路書店可以提供多項準則同時進行的檢索服務，或是逆向檢索，利用關鍵字的排列組合找到相似主題的書籍。其次，退貨率高居不下是傳統出版業的大問題，爲了爭取新書上架的機會，出版商提供寬鬆的退貨條件，卻到書籍再版、三版時才發現原來存貨都堆積在零售商的倉庫裡。經由線上交易，書商可以控制存貨流量，記錄、學習消費者的採購行爲，透過事前、事後的有效規劃，不僅降低退貨率，也可緩和出版商與零售業者間的關係。

(三)高度依賴網路人口成長狀況：使用者必須先上網才有可能在線上購物，故網路書店最終是否能產生獲利，端視未來網路人口的成長狀況與消費者對線上購物的接受度。[林素儀，1998]

(四)提供消費者更貼心的服務：Amazon 記錄分析每一位顧客的消費習性，主動提供相關書訊的服務，滿足了所有愛書人以及專業人士的需求，更讓顧客無法忽略它的存在。網路書店所提供的書評或讀者討論區也是傳統通路所缺乏的功能。

(五)沒有營業時間的限制：利用網路書店，顧客可以在家購書，再也不必爲買書東奔西跑。由於網路書店沒有營業時間限制，可以彌補實體通路的不足。[楊志偉，1999]

三、研究設計

3.1 研究架構

本研究主要是以消費者特性及產品特性來探討消費者使用網路書店購物之意願，根據研究目的與文獻探討，本研究推衍出如圖 2 之研究架構圖。而消費者特性包含了人口統計變數、行爲因素、心理因素，等，產品特性包含了產品價格、產品的屬性、產品的購物環境。

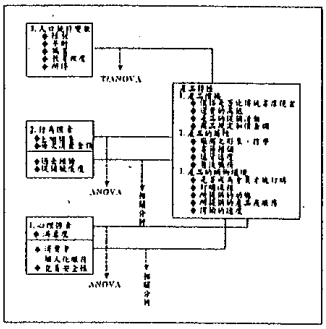


圖 2 研究架構圖

3.2 研究假設

本研究之研究假設如表 1 所示。

表 1 研究假設表(接下頁)

假設一：人口統計變數對於產品的價格並無顯著差異。 1-1 不同性別對於產品的價格並無顯著差異。 1-2 不同年齡對於產品的價格並無顯著差異。 1-3 不同職業對於產品的價格並無顯著差異。 1-4 不同教育程度對於產品的價格並無顯著差異。 1-5 不同所得對於產品的價格並無顯著差異。	假設九：心理因素對於產品的購物環境並無顯著差異。 9-1 消費者滿意度對於產品的購物環境並無顯著差異。
假設二：人口統計變數對於產品的屬性並無顯著差異。 2-1 不同性別對於產品的屬性並無顯著差異。 2-2 不同年齡對於產品的屬性並無顯著差異。 2-3 不同職業對於產品的屬性並無顯著差異。 2-4 不同教育程度對於產品的屬性並無顯著差異。 2-5 不同所得對於產品的屬性並無顯著差異。	假設十：行爲因素對於產品的價格無顯著性相關。 10-1 消費者過去的經驗對於產品的價格無顯著性相關。 10-2 消費者促銷的敏感度對於產品的價格無顯著性相關。
假設三：人口統計變數對於產品的購物環境並無顯著差異。 3-1 不同性別對於產品的購物環境並無顯著差異。 3-2 不同年齡對於產品的購物環境並無顯著差異。 3-3 不同職業對於產品的購物環境並無顯著差異。 3-4 不同教育程度對於產品的購物環境並無顯著差異。 3-5 不同所得對於產品的購物環境並無顯著差異。	假設十一：行爲因素對於產品的屬性無顯著性相關。 11-1 消費者過去經驗對於產品的屬性無顯著性相關。 11-2 消費者促銷的敏感度對於產品的屬性無顯著性相關。

表 1 研究假設表(續上頁)

假設四：行爲因素對於產品的價格並無顯著差異。 4-1 消費者上網頻率對於產品的價格並無顯著差異。 4-2 消費者每次消費金額對於產品的價格並無顯著差異。	假設十二：行爲因素對於產品的購物環境無顯著性相關。 12-1 消費者過去經驗對於產品的購物環境無顯著性相關。 12-2 消費者促銷敏感度對於產品的購物環境無顯著性相關。
假設五：行爲因素對於產品的屬性並無顯著差異。 5-1 消費者上網頻率對於產品的屬性並無顯著差異。 5-2 消費者每次消費金額對於產品的屬性並無顯著差異。	假設十三：心理因素對於產品的價格無顯著性相關。 13-1 消費者個人化服務對於產品價格無顯著性相關。 13-2 交易安全性對於產品價格無顯著性相關。
假設六：行爲因素對於產品的購物環境並無顯著差異。 6-1 消費者上網頻率對於產品的購物環境並無顯著差異。 6-2 消費者每次消費金額對於產品的購物環境並無顯著差異。	假設十四：心理因素對於產品的屬性無顯著性相關。 14-1 消費者個人化服務對於產品的屬性無顯著性相關。 14-2 交易安全性對於產品的屬性無顯著性相關。
假設七：心理因素對於產品的價格並無顯著差異。 7-1 消費者滿意度對於產品價格並無顯著差異。	假設十五：心理因素對於產品的購物環境無顯著性相關。 15-1 消費者個人化服務對於產品的購物環境無顯著性相關。

	13-2 交易安全性對於產品的購物環境無顯著性相關。
假設八：心理因素對於產品的屬性並無顯著差異。 8-1 消費者滿意度對於產品的屬性並無顯著差異。	

3.3 變數操作性定義與衡量

1.消費者特性

包含人口統計變數、行爲因素、心理因素等三項變素，茲說明如下。

(1)人口統計變數：本變數包含消費者的年齡、性別、個人所得、職業及教育程度等六項子變數。內容如下。

(2)行爲因素：本部份根據葉日武[民 86]消費者行爲與區隔理論加以設計，本變數包含上網頻率、過去經驗、每次平均消費金額、促銷敏感度等四個子變數，內容如下。

(3)心理因素：此變數根據第二章之文獻探討，將心理因素分爲上網動機、個人化服務、書評文摘、知覺風險等四個子變數。

2.產品特性

包含產品價格、產品屬性、產品的購物環境等三個變數，本部份以中 Likert 的五尺度量衡表來衡量，分爲「非常不滿意」、「不滿意」、「普通」、「滿意」、「非常滿意」，得分由 1 分到 5 分。1 分爲「非常不滿意」、5 分爲「非常滿意」。共分爲 20 題，其中產品價格方面的題目有 2 題，有關產品屬性方面的題目有 7 題，有關產品的購物環境方面的題目有 11 題。

3.消費者上網購物的意願

本研究由相關文獻探討得到十五個影響消費者上網購物的意願，在重視程度上由「非常重視」到「非常不重視」以 Likert 的五尺度量衡表，分別給予 5 分至 1 分，5 分代表「非常重視」，1 分代表「非常不重視」。

3.4 抽樣設計

本研究之抽樣設計就抽樣對象、時間、地點、及抽樣方式來探討。說明如下。

1.對象：本研究是以曾上網過的人爲抽樣的對象，只要曾經到過網站參觀或有購物經驗，都適合填寫這份問卷。本研究總共發出 250 份問卷，扣除無效問卷 17 份，共有 233 份問卷，回收率 93.2%。

2.時間：抽樣的時間大多利用星期假日，及平常下班、下課之人潮較多的時候進行訪問。

3.地點：發放問卷的地點分佈在北中南，包括台北、彰化、台南、高雄等地。例如台北的光華商場、彰化火車站、高雄的 NOVA、台南的北門路等，會在這些地帶逛街的受訪者，較有可能時常接觸電腦並上網過。

4.方式：問卷發放的方式採取現場訪問，視受訪者的意願爲主，若其沒有意願或沒時間接受訪問，並不會強迫其填寫此份問卷。

四、資料分析

4.1 基本資料分析

本研究受訪者基本資料，整理於表 2。問卷 1 表示有上網購物經驗，而問卷 2 表未曾在網路上購物。

消費者於網路書店購物之考量因素，分別就曾上網購物與未曾上網購物兩方面來分析，表 3 敘述曾上網購物之消費者考量因素分析結果，顧客之重視程度平均值在 4 以上，只有「24 小時購物環境」(Mean4.0900)、「提供多種商品搜尋方式」(Mean4.0500)較爲重視；其次重視程度較高的前三項依次爲「網站提供個人化服務與資訊」(Mean3.8800)、「節省時間」(Mean3.8100)、「網路書店的形象」(Mean3.7600)；相對而言，重視程度較低的三項依次爲「網路書店目前制度非常健全」(Mean2.8600)、「訊息回饋速度快」(Mean3.3500)、「提供良好的售後服務」(Mean3.3600)。

表 2 人口統計變數次數分配表

變項	項目	人數		百分比(%)	
		問卷 1	問卷 2	問卷 1	問卷 2
一、性別	男	70	55	70.0	41.4
	女	30	78	30.0	58.6
二、年齡	未滿 20 歲	5	8	5.0	6.0
	21-25 歲	55	85	55.0	63.9
	26-30 歲	17	20	17.0	15.0
	31-35 歲	15	10	15.0	7.5
	36-40 歲	2	1	2.0	0.8
	41-45 歲	4	2	4.0	1.5
	46 歲以上	2	7	2.0	5.3
三、職業	學生	53	73	53.0	54.9
	工	1	10	1.0	7.5
	商	12	8	12.0	6.0
	服務業	12	17	12.0	12.8
	資訊業	4	1	4.0	0.8
	自由業	5	4	5.0	3.0
	軍公教	9	10	9.0	7.5
	其它	4	10	4.0	7.5
四、教育程度	國小	0	1	0	0.8
	國中	0	1	0	0.8
	高中/職	3	17	3.0	12.8
	大專院校	84	111	84.0	83.5
	研究所以上	13	3	13.0	2.3
五、收入	15000 以下	47	66	47.0	49.6
	15001-25000	11	38	11.1	28.6
	25001-35000	17	18	17.0	13.5
	35001-45000	12	9	12.0	6.8
	45001-55000	9	2	9.0	1.5
	55001 以上	4	0	4.0	0

表 3 曾上網購物之消費者在網路書店購物之考量因素

題數	內容	平均值	標準差	排名
1	24 小時購物環境	4.0900	0.6831	1
2	網站提供個人化服務與資訊	3.8800	0.7691	3
3	網站的價格比傳統書店便宜	3.5500	0.9783	13
4	提供多種商品搜尋方式	4.0500	0.7160	2
5	網路書店書評	3.6200	0.8620	10
6	網路書店目前制度非常健全	2.8600	0.9214	20
7	訊息回饋速度快	3.3500	1.0384	19
8	時常舉辦促銷活動	3.5200	0.8817	14
9	提供市面上不易購得商品	3.4200	0.9763	16
10	提供最新的市場情報	3.7000	0.8933	8
11	網站容易導覽操作	3.7400	0.8718	7
12	網路書店的形象	3.7600	0.8180	5
13	退、換產品手續方便性	3.4200	1.1208	17
14	書籍種類多	3.5800	1.0068	12
15	送貨時間迅速	3.4600	1.1671	15
16	資訊內容即時更新	3.6400	0.9694	9
17	交易非常安全	3.6100	1.0815	11
18	個人資料保密	3.7500	1.0766	6
19	節省時間	3.8100	0.8492	4
20	提供良好的售後服務	3.3600	1.0686	18

表 4 敘述未曾上網購物之消費者考量因素分析結果，消費者重視程度之平均值在 3 以上，尤其對「交易非常安全」（Mean4.6466）更加重視；其次重視程度的前三項依次爲「個人資料保密」（Mean4.6241）、「提供良好的售後服務」（Mean4.4511）、「退、換產品的手續方便性」（Mean4.3609）。相對而言，消費

者對於「目前網路書店的制度非常健全」與否（Mean3.0451）就未如此重視了。

表 4 未曾上網購物之消費者在網路書店購物之考量因素(接下頁)

題數	內　　容	平均值	標準差	排名
1	24 小時購物環境	3.6692	0.8232	19
2	網站提供個人化服務與資訊	3.9699	0.7379	14
3	網站的價格比傳統書店便宜	4.2256	0.7649	8

表 4 未曾上網購物之消費者在網路書店購物之考量因素(續上頁)

4	提供多種商品搜尋方式	4.1203	0.8261	12
5	網路書店書評	3.6992	0.8527	18
6	網路書店目前制度非常健全	3.0451	0.9117	20
7	訊息回饋速度快	3.9398	1.0281	15
8	時常舉辦促銷活動	3.8797	0.8531	16
9	提供市面上不易購得商品	3.8421	0.9758	17
10	提供最新的市場情報	4.1053	0.8096	13
11	網站容易導覽操作	4.2105	0.7493	9
12	網路書店的形象	4.1278	0.9245	11
13	退，換產品手續方便性	4.3609	0.7720	4
14	書籍種類多	4.2707	0.7797	7
15	送貨時間迅速	4.2857	0.7936	6
16	資訊內容即時更新	4.3308	0.6823	5
17	交易非常安全	4.6466	0.7092	1
18	個人資料保密	4.6241	0.7241	2
19	節省時間	4.1955	0.7534	10
20	提供良好的售後服務	4.4511	0.7014	3

4.2 研究假設檢定與分析

根據本研究所提出的假設，分別以變異數分析及相關分析來驗證。爲了解人口統計變數之性別、年齡、職業、教育程度、所得對產品價格、產品屬性、及產品的購物環境之差異情形，本研究利用 T 檢定與 ANOVA 分析進行檢測；了解行爲因素與心理因素對產品價格、產品屬性、及產品的購物環境之相關情形，本研究使用 ANOVA 進行驗證，其結果整理於表 5。

表 5 本研究假設檢定之結果

研究假設	檢定結果	
	P 值	顯著差異
假設一：人口統計變數對於產品的價格並無顯著差異		
1-1 不同性別對於產品的價格並無顯著差異	0.280	
1-2 不同年齡對於產品的價格並無顯著差異	0.075	
1-3 不同職業對於產品的價格並無顯著差異	0.077	
1-4 不同教育程度對於產品的價格並無顯著差異	0.193	
1-5 不同所得對於產品的價格並無顯著差異	0.039	*
假設二：人口統計變數對於產品的屬性並無顯著差異		
2-1 不同性別對於產品的屬性並無顯著差異	0.268	
2-2 不同年齡對於產品的屬性並無顯著差異	0.275	
2-3 不同職業對於產品的屬性並無顯著差異	0.231	
2-4 不同教育程度對於產品的屬性並無顯著差異	0.409	
2-5 不同所得對於產品的屬性並無顯著差異	0.052	
假設三：人口統計變數對於產品的購物環境並無顯著差異		
3-1 不同性別對於產品的購物環境並無顯著差異	0.018	*
3-2 不同年齡對於產品的購物環境並無顯著差異	0.468	
3-3 不同職業對於產品的購物環境並無顯著差異	0.174	
3-4 不教育程度同對於產品的購物環境並無顯著差異	0.505	
3-5 不同所得對於產品的購物環境並無顯著差異	0.043	*
假設四：行爲因素對於產品的價格並無顯著差異		
4-1 消費者上網頻率對於產品的價格並無顯著差異	0.047	*
4-2 消費者每次消費金額對於產品的價格並無顯著差異	0.096	
假設五：行爲因素對於產品的屬性並無顯著差異		
5-1 消費者上網頻率對於產品的屬性並無顯著差異	0.361	
5-2 消費者每次消費金額對於產品的屬性並無顯著差異	0.030	*
假設六：行爲因素對於產品的購物環境並無顯著差異		
6-1 消費者上網頻率對於產品的購物環境並無顯著差異	0.426	

6-2 消費者每次消費金額對於產品的購物環境並無顯著差異	0.173	
假設七：心理因素對於產品的價格並無顯著差異		
7-1 消費者滿意度對於產品價格並無顯著差異	0.161	
假設八：心理因素對於產品的屬性並無顯著差異		
8-1 消費者滿意度對於產品的屬性並無顯著差異	0.041	*
假設九：心理因素對於產品的購物環境並無顯著差異		
9-1 消費者滿意度對於產品的購物環境並無顯著差異	0.344	

註：*表 P 值<0.05；**表 P 值<0.01；***表 P 值<0.001

本研究假設十至假設十五爲行爲與心理因素對產品特性的相關情形，本研究以相關分析方法來檢測，其分析結果整理於表 6。

表 6 行爲與心理因素對產品特性之相關分析

研究假設	檢定結果	
	相關係數	P 值
假設十：行爲因素對於產品的價格無顯著性相關		
10-1 消費者過去的經驗對於產品的價格無顯著性相關	0.306	0.221
價格是否比傳統書店便宜	0.044	0.663
運費之高低	0.415	0.000***
商品加價金額	0.458	0.000***
10-2 消費者促銷的敏感度對於產品的價格無顯著性相關	-0.084	0.055
價格是否比傳統書店便宜	0.169	0.094
運費之高低	-0.216	0.031*
商品加價金額	-0.205	0.041*
假設十一：行爲因素對於產品的屬性無顯著性相關		
11-1 消費者過去經驗對於產品的屬性無顯著性相關	-0.006	0.503
廠商之形象、信譽	-0.038	0.708
書籍種類	0.121	0.232
送貨速度	-0.125	0.214
售後服務	0.018	0.857
11-2 消費者促銷的敏感度對於產品的屬性無顯著性相關	0.322	0.038*
廠商之形象、信譽	0.455	0.000***
書籍種類	0.146	0.147
送貨速度	0.276	0.006**
售後服務	0.410	0.000***
假設十二：行爲因素對於產品的購物環境無顯著性相關		
12-1 消費者過去經驗對於產品的購物環境無顯著性相關	0.356	0.078
是否成爲會員才能訂購	0.121	0.233
訂購流程	0.535	0.000***
傳輸的速度	0.412	0.000***
12-2 消費者促銷的敏感度對於產品的購物環境無顯著性相關	-0.034	0.492
是否成爲會員才能訂購	0.066	0.515
訂購流程	-0.181	0.071
傳輸的速度	0.014	0.890
假設十三：心理因素對於產品的價格無顯著性相關		
13-1 消費者個人化服務對於產品價格無顯著性相關	0.174	0.457
價格是否比傳統書店便宜	0.451	0.000***
運費之高低	-0.009	0.933
商品加價金額	0.079	0.437
13-2 交易安全性對於產品價格無顯著性相關	0.001	0.024*
價格是否比傳統書店便宜	0.424	0.000***
運費之高低	-0.210	0.036*
商品加價金額	-0.211	0.035*
假設十四：心理因素對於產品的屬性無顯著性相關		
14-1 消費者個人化服務對於產品的屬性無顯著性相關	0.293	0.040*
廠商之形象、信譽	0.162	0.106
書籍種類	0.195	0.052
送貨速度	0.355	0.000***
售後服務	0.459	0.000***
14-2 交易安全性對於產品的屬性無顯著性相關	0.355	0.000***

廠商之形象、信譽	0.464	0.000***
書籍種類	0.358	0.000***
送貨速度	0.108	0.000***
售後服務	0.490	0.000***
假設十五：心理因素對於產品的購物環境無顯著性相關		
15-1 消費者個人化服務對於產品的購物環境無顯著性相關	0.114	0.637
是否成爲會員才能訂購	0.334	0.001**
訂購流程	-0.001	0.989
傳輸的速度	0.010	0.922
15-2 交易安全性對於產品的購物環境無顯著性相關	-0.235	0.02*
是否成爲會員才能訂購	-0.220	0.028*
訂購流程	-0.257	0.010*
傳輸的速度	-0.229	0.022*

五、結論與建議

5.1 研究結論

本研究在探討網路書店與消費者購買行爲之關係。上網購物者對購物網站操作流程的重視程度。就消費者之人口統計變數、心理因素、與行爲因素，對產品特性之產品的價格、屬性和購物環境之間的相關程度之研究，以了解其對於網路書店及購買行爲之影響。其主要的研究結論如下：

1. 除「人口統計變數對於產品屬性」、「行爲因素對產品的購物環境」、「心理因素對於產品價格」、與「心理因素對產品購物環境」無法判斷有顯著影響外，其餘各因素均呈現顯著差異的影響。(可參考表 5)
2. 就相關分析方面，「行爲因素對產品屬性」、「心理因素對產品屬性」、與「心理因素對產品的購物環境」等驗證結果呈現正相關，其餘因素無法判斷有相關。(可參考表 6)

5.2 研究建議

本研究之結論可提醒網站業者，應特別注重「消費者上網頻率」、「每次消費金額」、「消費者滿意度」「個人化服務」、「交易安全性」、「促銷敏感度」等這些因素，因這些因素對產品屬性中各要點有著正面之效果，期能幫助網路商店業者在擬定行銷策略時有所參考的依據。本研究提出幾點建議：1.減少交易安全的問題、2.速度與服務品質之改進，以達到個人化服務、及 3.增加消費者對網路書店的認同。

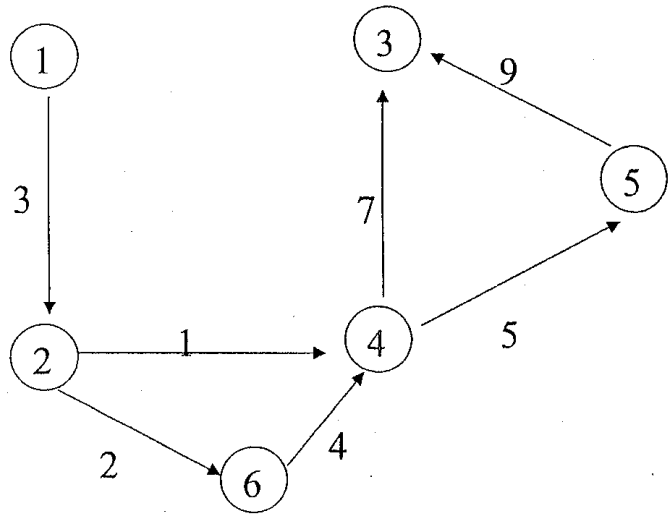
參考文獻

1. 王志剛、謝文雀(1998)，消費者行爲，華泰。
2. 王茂晃(2000)，連鎖書店消費者購買行爲之研究-以台北市兩大連鎖書店爲例，國立東華大學企業管理碩士班研究論文，6 月。
3. 古雅慧(1996)，資訊呈現對網路行銷廣告效果之研究-實驗法探討 WWW 網路購物情境，中央大學資管所碩士論文。
4. 李振妮(1999)，網路消費者購買決策行爲之研究，國立中山大學企業管理研究所碩士論文，6 月。
5. 林祖德(1997)，以實驗室實驗法探討等待對網際網路評估之影響-以 WWW 網路購物站台爲例，中央大學資管所碩士論文。
6. 林素儀(1998)，網路書店現況暨未來，光碟月刊，第 47 卷，頁 91-99。
7. 徐椿輝(1997)，網際網路線上服務服務品質評估模式之探討，台灣工業技術學院管理技術所碩士論文。
8. 寇世斌(1997)，網路上身，耶魯國際文化。
9. 張文慧(1998)，如何利用 Internet 行銷，聯經。
10. 張真誠(1998)，電子商務安全-數位交易金鐘罩，松崗。

11. 張瓊文(2000)，以消費者爲師－電子商務成功的關鍵就是消費者，電子商務時報，www.ectimes.org.tw
12. 張瓊文(2000)，由網店瀏覽到實際消費者購物習慣改變的大躍進，電子商務時報，www.ectimes.org.tw
13. 陳世運(2000)，台灣 B2C 電子商務個案探討(四)網路書店-博客來，資策會。
14. 黃智強(2000)，影響採用網路購物因素之影響－以網路書店爲例，國立中央大學資訊管理學系碩士論文，6 月。
15. 楊志偉(1999)，網路書店無可匹敵，突破雜誌，第 163 期，頁 73-75。
16. 趙培華(2000)，電子商務以人爲本，由「數位達爾文主義」談起，電子商務時報。www.ectimes.org.tw
17. 鄭聰華(2000)，網路購物消費者滿意度之研究-以台灣網路書店爲例，國立中山大學企業管理研究所碩士論文，6 月。
18. 簡貞玉譯(1996)，消費者行爲學，五南圖書出版公司。
19. Eighmey, J. (1997), "Profiling User Responses to Commercial Web Sites", Journal of Advertising Research, pp.21-35, May/June.
20. Richmond,A. (1996), "Enticing Online Shoppers to Buy: A Human Behavior Study", Computer Network and ISDN Systems, Vol.28,pp1469-1480.

Read the attached paper and answer the following questions. Note that the time is limited, and you should budget your time carefully. You are suggested to spend 50 minutes in reading the paper and another 50 minutes in answering the questions.

1. Please describe the work and the contributions the paper has done.
2. Given a dependency graph below, please construct the broadcast order based on the MST algorithm.
3. Given the same dependency graph below, please construct the schedule based on the Heavy algorithm, where θ=4 and $\bar{\ell}=2$.
4. There are two kinds of scheduling strategies introduced in the paper. Please give an application for each one for which it is suitable. (That is, you need to give two applications, one for each scheduling strategy.)



Multicast Scheduling for List Requests

Vincenzo Liberatore

Abstract—**Advances in wireless and optical communication, as well as in Internet multicast protocols, make broadcast and multicast methods an effective solution to disseminate data. In particular, repetitive server-initiated broadcast is an effective technique in wireless systems and is a scalable solution to relieve Internet hot spots. A critical issue for the performance of multicast data dissemination is the multicast schedule. Previous work focused on a model where each data item is requested by clients with a certain probability that is independent of past accesses. In this paper, we consider the more complex scenario where a client accesses pages in blocks (e.g., a HTML file and all its embedded images), thereby introducing dependencies in the pattern of accesses to data. We present a sequence of heuristics that exploit page access dependencies. We measured the resulting client-perceived delay on multiple Web server traces, and observed an average speed-up over previous methods ranging from 8% to 91%. We conclude that scheduling for multi-item requests is a critical factor for the performance of repetitive broadcast.**

Index Terms—**Multicast, Scheduling, Web performance, Network Applications, Wireless networks.**

I. Introduction

SEVERAL emerging technologies and applications naturally lead to the adoption of broadcast or multicast as the primary method for data dissemination. Broadcast is the primary mode of operation of the physical layer in media such as satellites and optical networks. As a result, it is natural to develop broadcast applications for those media. Broadcast can also be used in networks other than wireless and optical as a method to solve scalability problems. For example, multicast methods can relieve the scalability problems of Web hot spots [1] and can support the operations of a content delivery network [2]. Multicast methods can be combined with other performance enhancing techniques, such as caching [3], [4]. Broadcast and multicast techniques have spawned research (e.g., [1], [5]) and commercial ventures [6], [7], [8] that aim at higher scalability.

A common data dissemination method is to use *repetitive server-initiated multicast* [9], [5], [3], whereby a server cyclically multicasts (or broadcasts) data to a large client population. As a general data management technique, repetitive broadcast can be used in both wired [5] and wireless [10] networks to disseminate a variety of resource types, including Web contents [1] and database records [11]. A critical issue for broadcast performance is its organization. Some broadcast items will be more popular than others, so it is natural to broadcast the hot items more frequently. Page popularity has been modeled in the literature in terms of the probability p_i that page i is requested by clients. For the model where the probabilities p_i are stationary and independent of past accesses, several algorithms have been proposed [12], [13], [14]. Extensions include the cases when broadcast pages have different sizes [15], [16] and when client objectives are described by polynomial utility functions [17].

In general, clients are seldom interested in individual data items, and attempt to download multiple items. For example, Web clients are seldom interested in only one HTML resource, but access almost always the HTML document along with all its embedded images [18]. Analogously, database clients often access multiple items to complete a read transaction [19]. In this paper, we will examine novel scheduling strategies that keep into account dependencies in the client accesses to resources. The objective is to reduce client-perceived latency when she downloads multi-item objects.

The presence of multi-item requests complicates the scheduling of the broadcast, as shown by the following examples.

Example 1: Suppose that E is an image embedded in page A. Consider a schedule that broadcasts E immediately after A. A request $\{A, E\}$ takes only slightly more than the time needed to retrieve A only. If E and A were broadcast in an arbitrary order that takes into account only their access frequencies, it is possible that one document is broadcast a long time after the other, thereby delaying the request completion time.

Example 2: Suppose that E is embedded in A as well as in another page B. Consider now a request for $\{B, E\}$. By the same token, E should be transmitted immediately after B. However, if E is broadcast after A *and* after B, the repeated transmission of E lengthens the broadcast cycles and could delay other pending requests. A different method is to send the three documents in the order $\ldots A, B, E \ldots$, which could potentially have better performance than repeating E.

Example 3: Consider a request for $\{A_1, A_2, \ldots, A_k\}$, where $A_1, A_2, \ldots, A_{k-1}$ are broadcast fairly often and A_k is seldom broadcast. The completion time of this request is tied up to the low transmission rate of A_k. In other words, frequent broadcast of hot items does not help the completion time of multi-document requests involving colder items.

In general, multi-item requests create complex dependencies in the document access pattern and can complicate the broadcast schedule. The paper will propose and analyze effective heuristics for the problem of multicast scheduling under dependencies in the request sequence. Algorithms are evaluated on multiple server logs of Internet hot spots.

The paper is organized as follows. In section II, we give background information on broadcast data management techniques and on broadcast scheduling. In sections III and V, we present a sequence of algorithms for broadcast scheduling and provide evidence of the limited applicability of known broadcast strategies. In section IV, we describe our experimental

Electrical Engineering and Computer Science Department, Case Western Reserve University, 10900 Euclid Avenue, Cleveland, Ohio 44106-7071. E-mail: vl@eecs.cwru.edu. URL: http://vorlon.cwru.edu/~vxl11/.
Research supported in part under NSF grant ANI-0123929.

0-7803-7476-2/02/$17.00 (c) 2002 IEEE.

共10頁第2頁

set-up. In section VI, we validate our algorithms on two more traces that we did not use to tune parameters. In section VII, we summarize work related to ours, and in section VIII we draw the conclusions of our investigation.

II. Background

Broadcast Environment: Cyclical server-initiated broadcast and multicast [9], [5], [3] can be used to execute data dissemination and are well suited for wireless and mobile environments, as well as for relieving Internet hot spots. A set of n pages is cyclically broadcast by a server to a large client population. The broadcast is initiated without client requests, i.e., it follows a "push" style of data dissemination. Furthermore, the broadcast is repetitive, that is, the server continuously cycles through its set of broadcast data. Examples of broadcast programs are illustrated in figure 1 and 2. When a client needs to read the contents of page i, it waits for the data source to broadcast i. The client does not need to listen continuously on the broadcast for page i to be sent if an appropriate index is broadcast as well; such index also allows the client to determine which contents are present in the broadcast data set [20], [21], [22], [23]. In its purest implementation, the server accepts no input from clients and simply cycles through its broadcast. In more complex schemes, the server accepts update transactions from clients [24], [19] or uses broadcast as a complement to other dissemination methods [25]. For example, a server can use broadcast to propagate hot documents, while it uses other methods for colder items [1]. A critical performance metrics for broadcast data dissemination is the amount of time that elapses between a client request and the time when the client has downloaded all requested pages.

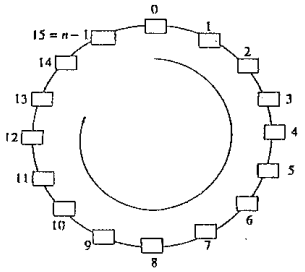


Fig. 1. An example of a flat broadcast program. Pages are numbered from 0 to $n-1$, and are cyclically transmitted by the server in that order.

The scope of the paper is to investigate algorithms to schedule the broadcast at the server site so as to reduce client-perceived latency. To focus on the scheduling problem, we make the following assumptions:

- The broadcast schedule is fixed by the server, and is known by clients.
- Pages are reliably received by the clients in the same order as they are broadcast.
- Pages are read-only, and cannot be updated by either the server or the clients.
- Clients receive pages only from a unique server broadcast over a single broadcast layer.

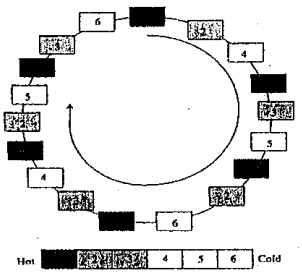


Fig. 2. A skewed broadcast schedule that is obtained by multiplexing on the same physical channel the logical channels $\{1\}, \{2,3\}, \{4,5,6\}$.

Term	Definition	Similar terms
Page	Broadcast transmission unit	-
Document	Data object that a client can identify by an id	Resource, ADU
Object	Collection of resources target of client requests	-

Dependency	Definition
Internal	between pages belonging to the same document
External	between documents

TABLE I
SUMMARY OF DEFINITIONS USED IN THE PAPER.

- The set of broadcast pages does not change.
- Data is broadcast at a constant rate.

All these restrictions can be removed in an actual implementation, as will be explained in the next sections. However, a more complex scenario would obscure the analysis of scheduling strategies, and so we do not consider it in the rest of the paper.

Broadcast Scheduling: The data source can schedule pages for broadcast according to a variety of strategies. The simplest broadcast strategy is to adopt a *flat broadcast schedule*, whereby each page is transmitted once every n ticks. A flat schedule is exemplified in figure 1. There are $(n-1)!$ distinct flat schedules, and section III will demonstrate that certain flat schedules have substantially better performance than others. Non-flat schemes are desirable when some pages are more popular than other, in which case hot pages should be devoted a larger fraction of available bandwidth. A simple way to differentiate pages is through *frequency multiplexing*: the data source partitions the data set across several physical channels according to their popularity. Differentiated treatment arises from aggregating a smaller amount of hot data on one channel and a larger amount of colder data on another channel. Since channel bandwidth is the same, the fewer hotter pages receive a proportionally larger amount of bandwidth than colder pages. Frequency multiplexing can be effective if multiple channels are available, but is unfeasible otherwise. An alternative is

0-7803-7476-2/02/$17.00 (c) 2002 IEEE.

time-division multiplexing, whereby pages are partitioned into a set of *logical channels*, and the logical channels alternate over the same physical channel [9]. The broadcast schedule is flat within a single logical channel, but hotter channels contain less pages or are scheduled for broadcast more frequently than colder channels. Thus, hot pages are transmitted more often than colder ones. Figure 2 gives an example of a broadcast schedule that is the time-multiplexed combination of three logical channels, each containing a different number of pages. Time-division multiplexing is potentially more flexible than frequency multiplexing in that it allows for a finer bandwidth partition. In particular, a logical channel can contain only one page, which results in a fine per-page transmission schedule. When the broadcast is scheduled on a per-page basis, pages are broadcast on the same physical channel with frequency proportional to their popularity.

A family of scheduling algorithm for time-multiplexed broadcast assumes that the data source has estimates of the probabilities with which clients need pages. The square-root law asserts that page i should be scheduled with frequency proportional to $\sqrt{p_i}$ [13], where p_i is the probability that i is requested by clients. A simple and practical 2-approximation algorithm is expressed by the *MAD* (Mean Aggregate Delay) rule [12], [13]. The MAD algorithm maintains a value s_i associated with each page i. The quantity s_i is the time since the last time page i was broadcast. The MAD algorithm broadcasts a page i with the maximum value of $(s_i + 1)^2 p_i$. MAD guarantees a cyclical schedule, and, in particular when all p_i's are equal, MAD generates a flat broadcast. The access probabilities p_i do not express dependencies between data items. Consider the following elementary example. Pages A and B are not accessed very frequently, but when A is accessed, page B is almost certainly accessed as well. In this scenario, the access probability p_B of page B is small, but the value of p_B is not fully expressive of the true access pattern to B.

We classify dependencies among pages in internal and external (table I). An *external dependency* arises when there is a dependency between the original resources in the client access pattern, as for example when the access probability of B is conditional to the previous occurrence of a request for A. An *internal dependency* arises when the underlying transport forces long documents to be broken in smaller portions. For example, IP-based data dissemination, such as [1], uses IP multicast to propagate data to a large client population. Since the broadcast can cross Internet links with different MTU's and reach clients with different reassembly buffer sizes, a document is fragmented at the source into *pages* of approximately equal size so that pages fit within the IP size limits [26], [27]. We make the distinction between *documents*, which can be identified and requested by clients through a resource identifier, and *pages* which are broadcast units of roughly equal size and in which original resources are partitioned. Documents can also be variously referred to as *resources* or *Application Data Units (ADU)*. We will call the dependency among pages from the same document an *internal dependency*. A common definition in the literature is that of an *object*, which is a collection of resources that are the target of a client request. With this terminology, documents within the same object show an external dependency, while pages within the same document have an internal dependency. Pages belong to only one document whereas documents can belong to an arbitrary number of other objects. As a result, internal dependencies are likely to create a simpler scenario than external dependencies. In this paper, we consider both internal and external dependencies. Client-perceived delays will be partitioned into two components: the *seek time* is the time that clients wait to receive the first page of an object, and the *transfer time* is the time that clients wait to receive the rest of the object.

III. Circular Arrangement

In this section, we examine whether transfer time can be reduced within the context of flat broadcast schedules. Skewed (i.e., non-flat) schedules will be examined later in section V. Although flat schedules do not transmit a page any more frequently than any other page, flat schedules performance can vary significantly when there are dependencies between page accesses. For example, suppose that page B is always requested after page A. A random flat schedule takes about n broadcast ticks in the expectation to retrieve the object $\{A, B\}$: $n/2$ ticks to retrieve A followed by $n/2$ ticks to retrieve B. A flat schedule that arranges B immediately after A takes only $n/2 + 1$ ticks to retrieve the same object $\{A, B\}$. Although the second schedule did not reduce the seek time (i.e., the time to retrieve A), it was extremely effective at reducing transfer time (i.e., the time to retrieve B after A has been downloaded). In general, a page can appear in multiple objects (e.g., a resource embedded in several other documents), and so object pages cannot always be grouped in a contiguous broadcast interval. A possible solution would be to replicate the page once for each containing object, but such approach could unnecessarily lengthen the broadcast schedule and result in longer seek times. In section V, we will explore a limited replication mechanisms that duplicates the hottest pages for a controllable number of times. First, however, we examine in which ways and to what extent transfer times can be reduced solely in the context of flat schedules.

Problem Model: We model the problem of reducing transfer time as the following graph optimization problem. We associate a node to each page and insert an arc (i, j) from page i to page j when there is a dependency between i and j that makes j more likely to be accessed after page i. We will also associate a weight to the arc (i, j) proportional to the strength of the dependency. We call such graph the *dependency graph* of a trace because its arcs express dependencies between pages and arc weight express the strength of the dependency. We then seek to arrange pages around a broadcast cycle so that the weighted arc length is minimized. More precisely, we define the *Minimum Circular Arrangement (MCA)* problem as

Instance: A directed graph $G = (N, A)$ and non-negative arc weights $w(e) \in \mathbf{N}$ for each $e \in A$.

Question: Find a one-to-one function $f : N \rightarrow \{0, 1, \ldots, n-1\}$ that minimizes

$$\sum_{e=(u,v)\in A} w(e)\ell(e) \qquad (1)$$

where $n = |N|$ and $\ell(e) = ((f(v) - f(u)) \bmod n)$.

0-7803-7476-2/02/$17.00 (c) 2002 IEEE.

共10页第4页

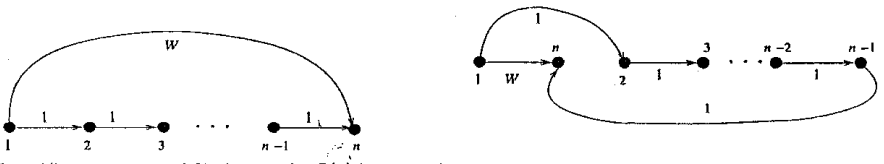


Fig. 3. A directed linear arrangement (left) whose cost is a $\Omega(n)$ factor away from the cost of the optimum circular arrangement (right). The value of h for the arc of weight W is $n-1$ in the topological arrangement and is only 1 in the circular arrangement. The value of h for the back arc in the circular arrangement is 2, and was 1 in the linear arrangement. Thus, the cost for the heavy arc can be substantially reduced with minimal changes in the cost of other arcs.

MST algorithm
Given a dependency graph $G = (N, A)$
Let P be a partition of the nodes of the graph G, initialized to n singleton sets.
(The algorithm maintains an ordering of each set in P)
for all arcs $e = (u, v)$ of G in non-increasing order of weight:
  Let P_u be the component of P that contains u and P_v be the component that contains v
  if $P_u \neq P_v$
    Insert e in the spanning tree T
    Unite P_v and P_u and append the ordering of P_v after the ordering of P_u
Concatenate all orderings of sets in P and
return such ordering.

Fig. 4. The MST algorithm.

The quantity $\ell(e)$ is the distance between arc endpoints in the circular arrangement f and will be said to be the *length* of arc e in f. In the MCA model, there is some degree of latitude in the choice of the objective function. We chose a linear objective (1) because it forces related pages to be clustered next to each other while the objective function remains relatively easy to analyze.

A question related to MCA is the *linear arrangement problem*, where the graph nodes are to be arranged along a line (instead of a circle) and the graph is assumed to be acyclic. The minimum linear arrangement has been extensively studied: it is NP-hard [28] and several approximation algorithms have been proposed [29], [30], [31], [32]. A simple example shows that the linear and circular arrangement problems are intrinsically and radically different, so that approximation algorithms for linear arrangement are most likely irrelevant in the context of circular arrangements. Specifically, we demonstrate that the optimum linear arrangement can cost $\Omega(n)$ times as much as a circular arrangement even when the underlying graph G is acyclic. Consider the graph in figure 3, and observe that it has a unique topological ordering $1, 2, \ldots, n$ at a cost of $(n-1)(W+1)$, whereas the circular arrangement $1, n, 2, \ldots, n-1$ has a cost of $W + n + 1$. We then take $W = \Omega(n)$ to make the cost ratio $\Omega(n)$. On the other hand, if we consider any circular arrangement, the cost due to an arc is at most $n-1$ times the cost that the optimum pays for the same arc. As a result, an $O(n)$-approximation algorithm is trivial. We conclude that circular and linear arrangement problems are in general unrelated.

Hardness: We consider the theoretical solvability for MCA, and to this end we introduce the following decision version of the optimum circular arrangement problem, which we call the *Circular Arrangement Problem* (CA):

Instance: A directed graph $G = (N, A)$, non-negative arc weights $w(e) \in \mathbb{N}$ for each $e \in A$, and a positive integer K.

Question: Is there a one-to-one function $f : N \rightarrow \{0, 1, \ldots, n-1\}$ such that

$$\sum_{e=(u,v)\in A} w(e)\ell(e) \leq K?$$

where $n = |N|$ and $\ell(e) = ((f(v) - f(u)) \bmod n)$.

Proposition 1: The Circular Arrangement Problem (CA) is NP-complete.

Proof: [Sketch] The proof is a reduction from the direct linear arrangement problem. ■

MST Heuristic: Although no polynomial-time algorithm is likely to solve MCA optimally, a reasonably good solution can be obtained through a heuristic applied to the dependency graph. Our procedure is based on a topological ordering of a maximum spanning tree (MST) of the dependency graph. Specifically, the algorithm maintains a partition of the node set and an ordering for the nodes within each partition. Initially, the node partition consists of n singleton, one for each node of the original graph. Then, our procedure computes the maximum spanning tree of the dependency graph with Kruskal's algorithm [33], and, when the algorithm combines two node sets, the heuristics also combines the two component orderings. On the whole, the algorithm is in figure 4.

The algorithm is greedy, in that it arranges nodes as close as possible if there is an arc with a large weight between them. At the beginning, the algorithm arranges the nodes (i.e., pages) u and v next to each other if the arc (u, v) has maximum weight (i.e., if v appears in the same object as u for the maximum number of times). As the algorithm progresses, the algorithm combines the ordering of P_u and P_v if the arc (u, v) has maximum weight among all remaining arcs (i.e., the page orderings are combined if there is a page v that appears in the same object as u for the maximum number of times). As a result, the algorithm can be viewed as producing a sequence of page clusters P and combining two clusters on the basis of dependencies between two pages; as clusters are combined, their orderings are

0-7803-7476-2/02/$17.00 (c) 2002 IEEE.

trace	begin (coord. univ. time)	end (coord. univ. time)	length		Broadcast trace				
			log	sanitized	n	clients	m	objfeq	sngobj
1	[30/Jun/1998:15:00:00]	[01/Jul/1998:00:00:00]	57639163	90.36%	225	127256	80046572	7626200	28.40%
2	[01/Jul/1998:15:00:00]	[02/Jul/1998:00:00:00]	9079767	94.29%	174	67261	9967240	896297	25.12%
3	[08/Jul/1998:20:00:00]	[09/Jul/1998:00:30:00]	15500700	90.95%	206	59479	20473599	2083034	28.98%
4	[09/Jul/1998:20:00:00]	[10/Jul/1998:00:00:00]	1986787	94.36%	200	17563	2449844	199169	21.29%

TABLE II

CHARACTERISTICS OF CLIENT WEB TRACES COLLECTED FROM THE WORLD CUP 98 SERVER TRACE.

concatenated as well. Therefore, the algorithm gives as a by-product a page clustering that depends on the frequency with which pages belong to the same object. The MST algorithm takes $O(n^2 \log n)$ time in the worst case and, on our simulations (section IV), it always ran in less than 600 ms on a Ultra 60 workstation with a 450Mhz CPU, 4MB L2 cache, 512 MB of memory, Solaris 8, g++ compiler, and LEDA data structures libraries [34].

IV. EVALUATION

Methodology: We limit our empirical analysis to Web logs of Internet of hot spots. Although the concepts in this paper should be applicable to both wired and wireless networks and to several applications, the restriction allows us to obtain more exhaustive results and to use several publicly available traces. Experiments were executed with traces that were extracted from the log of the HTTP servers for the Soccer World Cup 98. The World Cup trace includes more than one billion requests over a period of 1 1/2 month and is one of the largest trace analyzed to date [35]. Furthermore, the World Cup servers received up to 10 million requests per hour. As a result, the World Cup site is one of the most busy recorded so far, which makes it an ideal testbed for multicast data dissemination. Additional traces will be considered in section VI to execute a blind validation of algorithms.

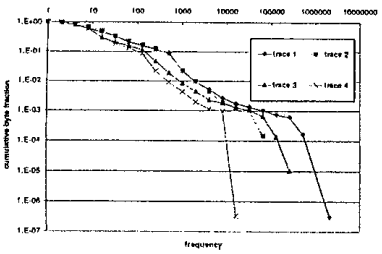


Fig. 5. Cumulative percentage of bytes requested with at least the given frequency.

The server logs report only requests that percolate to the origin server, and in particular, the logs do not report requests that are satisfied by intermediate caches. We extracted from the complete server logs a set of four subtraces that correspond to the four most active periods (table II). We kept requests that fell in the target busy interval, that are GET or HEAD methods for HTML, image, Java, or compressed resources, and that gave rise to 200 (ok) and 304 (not modified) response codes. Some document sizes changed during the course of the trace. Size change is due either to interrupted transfers or to document contents updates. Although updates can be incorporated in broadcast environments [19], [24], our logs do not allow us to determine the origin or nature of size changes, and so we restrict this study to fixed-size documents. After documents with changing sizes were eliminated, the resulting subtraces contain more than 90% of the transfers made during the chosen intervals (table II, column "sanitized").

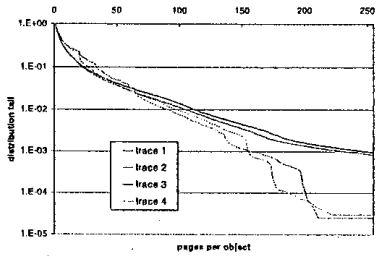
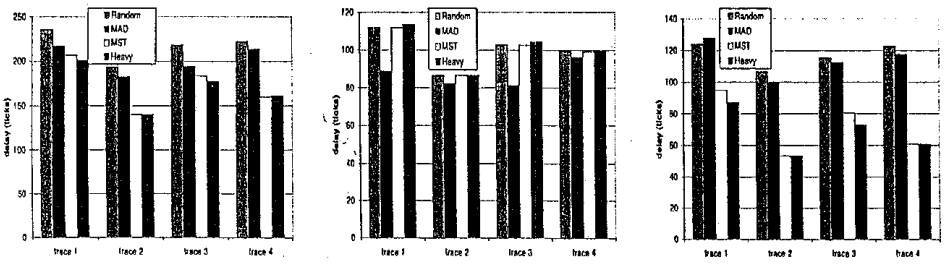


Fig. 6. Tail of the distribution of the number of pages per object. The vertical axis gives the frequency with which an object had at least the number of pages on the horizontal axis.

In Internet data delivery, documents that are referenced sporadically are not usually multicast [1], [36]. Figure 5 gives the cumulative size (as a fraction) of resources that were requested with at least a certain frequency. The distribution has a knee in correspondence of $\chi = 7 \cdot 10^{-4}$. In other words, if the hottest χ fraction of bytes is broadcast, the broadcast will contain extremely popular items, but a further increase of the broadcast size n would quickly result in significantly colder items occupying the broadcast schedule. Consequently, we inserted in the broadcast only the resources corresponding to the hottest χ fraction of bytes. An alternative choice is based on a dynamic assessment of document popularity [36]. We did not use any dynamic schemes in this paper because we were interested in isolating the performance of scheduling algorithms from that of other methods. We envisage that a real implementation would need to support both scheduling and dynamic document selection. As in [1], broadcast documents were divided into 512B pages. A GET method translates into a request for all document pages, a HEAD request for the appropriate number of pages at the beginning of the document, and a non-modified reply into a request for the first page of that document. In practice, clients

共10页第6页



(a) Total latency (b) Seek (c) Transfer

Fig. 7. Average latency expressed as number of broadcast ticks. *Seek* latency is the number of ticks required to retrieve the first document page. *Transfer* latency is the number of ticks to retrieve the remaining document pages.

are often interested in a collection of related documents, such as an HTML file and its embedded images. The HTTP protocol does not aggregate document requests into object requests. In these cases, a heuristic is that if two documents are requested within one second of each other in the trace, they belong to the same logical object [37]. In our experiments, the client triggers requests for all other pages in the object upon reception of the first object page; such scheme can be implemented through prefetching hints embedded in the first object page [38]. As cached resources do not appear in the server logs, the client requests are akin to GETLIST method invocations [18]. Table II gives the number n of pages in the simulated server broadcast, the number of unique IP addresses generating requests for broadcast pages, the number m of page requests, the number *objreq* of object requests, and the percentage *sngobj* of single page object requests. Some objects are big, but, depending on the trace, 21% to 29% of objects contained only one page. Figure 6 plots the distribution of the number of pages within objects.

Another design choice is the speed at which data is broadcast. Previous work suggests an optimal rate of 256 Kbps for IP multicast [1]. For comparison, if a single unicast-based server had been connected at the same rate, it would not have scaled to satisfy logged requests for broadcast resources in the World Cup traces. We simulated several rates ranging from 48 Kbps (to support most modems) to 1.544 Mbps (a T1 line). Most of these rates are also within the capacity of short-range or 3G wireless technology. A *broadcast tick* is the time needed to transmit a page. The duration of a broadcast tick clearly depends on the broadcast rate. Latencies were measured both in seconds and as number of ticks; in the latter case (latency as number of ticks), delays did not significantly depend on the broadcast rate. The simulations will employ only static values of available bandwidth. The more general problem of dynamically adjusting the transmission rate to the available bandwidth is explored in [39], [5], [40].

Results: Figure 7 compares a random flat broadcast, a MAD broadcast [12], [13], which is based solely on stationary access probabilities, and the MST heuristic. The figure shows the average delays expressed as number of broadcast ticks. For comparison, at the given broadcast rate, a unicast server would not have been able to satisfy the requests for broadcast resources. The reported delays are for requests to the origin server, and so any performance improvement is in addition to those due to caching. In these experiments, MAD schedules lead to improvements over flat schedules of the order of 6% to 13%. MAD schedules were particularly effective at reducing seek time (from 4% to 26%), but did not provide a clear advantage in terms of transfer time (from -3% to 6%). The MST algorithms outperformed MAD by 5% to 34%. Not surprisingly, MST seek time was comparable to that of a random schedule and it was roughly $n/2$. However, MST resulted in a substantial reduction of transfer times, which in two traces was half as much as MAD's. The waiting time reduction is more marked for trace 2 and 4; we believe that the more pronounced improvement is due to the smaller percentage of single objects request in those traces (table II).

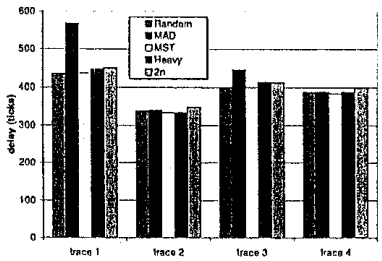


Fig. 8. Total latency in the 99th percentile expressed as number of broadcast ticks.

The distribution of the delays was collected as well and figure 8 shows the 99th percentile of the delays. Flat schedules, and MST in particular, never take more than $2n$ ticks to download an object, whereas no worst-case bound holds for MAD. Correspondigly, figure 8 shows that the delay of MST and random is always below $2n$, whereas MAD exceed $2n$ on both trace 1 and 3. It can be also noticed that if in a trace MAD did better on the average, it did more poorly in the 99th percentile. Intuitively, MAD attempts to optimize for the average case at the expenses of the tail whenever possible.

In summary:

Random flat broadcast is not a particularly good strategy on

0-7803-7476-2/02/$17.00 (c) 2002 IEEE.

Heavy algorithm
Construct an MST schedule f and determine all θ-heavy arcs
for each page v that is the head of a heavy arc
  Locate v's position in the MST ordering and, starting from this position,
  do
    Scan the MST ordering in the reverse direction of the broadcast order until the tail u of a heavy arc is (u, v) is found
    if u is more than ℓ ticks away from the next time v will be broadcast,
      then Insert v in the broadcast after u
  until the scan returns to v's original position
return the resulting schedule.

Fig. 9. The Heavy algorithm.

the average, but it provides a worst-case bound ($2n$).

MAD improved on a random flat broadcast by 6% to 13% on the average, but it provides no worst-case guarantees.

MST improved over MAD by 5% to 34% on the average by exploiting access pattern dependencies, and, since it follows a flat broadcast, it has the same worst-case bound ($2n$) as the random flat schedule.

V. Beyond Flat Schedules

We have dealt so far with two types of scheduling strategies: skewed schedules that exploit stationary access probabilities (i.e., MAD) and flat schedules that exploit page dependencies (i.e., MST). Up to this point, we viewed the two approaches as antithetic. In this section, we examine ways to combine the two paradigms to reduce client-perceived latencies. The resulting schedule should be skewed to favor hotter pages over colder ones and should be arranged in such a way as to reduce user-perceived latency.

Heavy Arcs: An MCA solution f can contain arcs that have a large value of $w(e)\ell(e)$ and that consequently contribute to a large fraction of the CA objective value.

Definition V.1: An arc is *θ-heavy* if $w(e)\ell(e) \geq \theta$ for a threshold value θ. If the value of the threshold θ is clear from the context, we will simply designate such arcs as *heavy*.

A flat schedule is inherently limited in its ability to eliminate heavy arcs. For example, suppose that the dependency graph has an arc from every node u to a designated vertex v and that the weights of the arcs $e = (u, v)$ are large, e.g., $w(e) > 2\theta/n$. Then, there are $\Omega(n)$ arcs with $w(e)\ell(e) \geq w(e)n/2 \geq \theta$.

Definition V.2—[33]: Let $G = (N, A)$ be a directed graph. The *head* of an arc (u, v) is node v and the *tail* is node u.

A method to reduce the impact of heavy arcs is to broadcast the page corresponding to the arc head soon after the tail. As a result, the arc length $\ell(e)$ is reduced and so is its contribution $w(e)\ell(e)$ to the CA objective value. The drawback is that the schedule contains more pages, and so the length of other arcs can increase. Therefore, a schedule should not replicate an excessive number of arc heads for an excessive number of times. The degree of replication can be controlled by tuning two parameters. The first parameter is the threshold θ for an arc to be considered heavy. The second parameter is the maximum length $\bar{\ell}$ of a heavy arc in the new schedule. A larger value of $\bar{\ell}$ allows a heavy arc to be longer in the new schedule, and thus the arc head v to be replicated a smaller number of times. To simplify the implementation, we ignored outgoing arcs from a replicated node. A simple greedy strategy calculates the smallest number of times a page is replicated along a schedule so that heavy arc length is no more than $\bar{\ell}$. The resulting algorithm is in figure 9.

We found that good parameter values are $\theta = 1$ and $\bar{\ell} = 32$ across all traces; such algorithms will be denoted as the *heavy* algorithm. The heavy algorithm is fast (no more than 640 ms on the same machine and traces as in section III). Figure 7 gives the latency for the resulting strategy for the four traces; MAD was outperformed from 8% to 33% by the heavy algorithm.

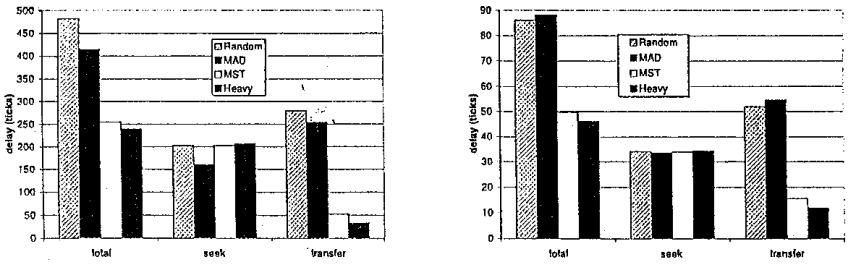
Square-Root Scheduling: We have attempted several methods to integrate MST and heavy scheduling with other methods based on independent probabilities and on the square root law, but we were not able to achieve any appreciable performance improvement over the heavy algorithm. We believe that this is due to the page access distribution in our traces. Figure 10 plots page access frequency as a function of page rank. The most popular page is assigned a rank of 1 and the least popular page a rank of n. In a log-log plot, a Zipf distribution $\Pr[i] \propto i^{-\alpha}$ would appear as a line with slope $-\alpha$. Page popularity can be explained by a Zipf distribution, although with rather low confidence ($0.65 \leq R^2 \leq 0.76$). Furthermore, the fitted Zipf distribution is only mildly skewed, with $0.65 \leq \alpha \leq 0.76$. Such result is consistent with the analysis in [35], where lack of skewness was attributed to factors such as the use of the most popular embedded images across most of the site objects and the presence of caches interposed between clients and servers. At any rate, low values of α lead to an almost flat square-root broadcast and so they hamper the potential for improvement of schedules based on independent probabilities. We conclude that square-root schedules are inherently limited in their ability to address satisfactorily this type of workloads.

VI. Validation

Up to this point, we have used the same traces both to measure scheduling performance and to tune parameters, such as χ, θ, and $\bar{\ell}$. We then validated our methods on two additional traces with *no* further algorithm modification or parameter tuning. Our objective was to perform a "blind" test of our methods. We collected the trace of HTTP requests to the main Web server of the Computer Science department at Rutgers University between 12 p.m. and 4 p.m. on December 18, 1999. The other

0-7803-7476-2/02/$17.00 (c) 2002 IEEE.

艾10页第8页



(a) cs.edu (b) NASA

Fig. 11. Latency on the cs.edu and NASA traces expressed as number of broadcast ticks.

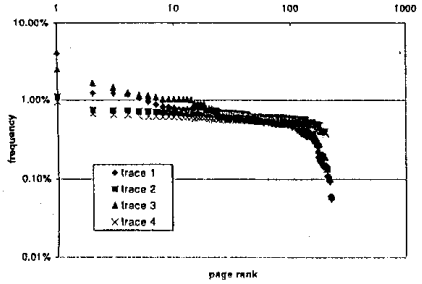


Fig. 10. Concentration of references (reference count plotted against page rank).

trace is the NASA server trace collected between 12 p.m. and 4 p.m. on August 3, 1995. Similarly to the preprocessing of the World Cup traces, we eliminated all but the $\chi = 0.07\%$ of the hottest bytes and simulated a cyclical broadcast at the rate of 256 Kbps. User-perceived latency in term of broadcast ticks is shown in Figure 11. The heavy algorithm outperformed MAD by 73% to 91%. The heavy and MST methods had seek time comparable to that of a random schedule, but reduced transfer time by a factor ranging from 4 to 7.

VII. Related Work

Broadcast scheduling has been the subject of extensive investigation, mostly under the hypothesis that pages have stationary access probabilities [12], [13], [14]. Furthermore, Ammar gives analytical expressions to estimate client waiting time in the presence of conditional access probabilities and Poisson OFF periods [38]. To the best of our knowledge, no previous work examines the case where a client requests simultaneously multiple resources. Broadcast with non-uniform page sizes has been considered as well [15]. Variable-size pages cannot be directly identified with objects in that multiple objects can have common pages. Alternative scheduling methods use a pyramid scheme that is particularly suited to streaming media [41]. Clustering through spanning trees is a well-known technique [33], which we adapt to broadcast scheduling in the presence of access pattern dependencies. Additional background was summarized in Section II and IV.

VIII. Conclusions

Discussion: Broadcast is the primary mode of operation for several wireless and optical media and leads naturally to transport and application solutions. Analogously, multicast has the potential of effectively relieving Internet hot spots, thereby leading to scalable applications. Multicast can also be used in CDN backbones and it can be employed in conjunction with caching. A critical issue in broadcast management is the organization of the broadcast. Typical broadcast schedules, such as MAD, transmit hot pages more often than colder ones. Meanwhile, clients are often interested in downloading lists of pages from the server. This scenario leads to strong dependencies in the access pattern, and if such dependencies are taken into account, substantial performance improvements are possible. We have a proposed a simple greedy algorithm (MST) for flat schedules and a refinement (Heavy) that leads to a (slightly) skewed schedule. The algorithms are fast in theory and in practice and it is easy to update their schedule if underlying dependencies change. Moreover, the algorithms give as a by-product a clustering of pages according to their access dependencies.

The algorithms were extensively analyzed on multiple Web traces. Furthermore, an additional set of two more traces was considered in order to perform a "blind" validation of algorithms, i.e., an algorithm validation without any further parameter tuning. Our final algorithm (Heavy) leads to improvements in client-perceived latency ranging from 8% to 33% on the World Cup traces, and up to 91% on the blind validations. The MST algorithm generates a flat schedule and so, in addition to improving average access time, it offers a worst-case bound on the acces time. We conclude that substantial performance improvements can be efficiently obtained by considering dependencies in the access pattern.

Methodological Implications: From a methodological perspective, we observe that most broadcast data management research has been conducted under the *independent reference* assumption: page i is requested at time t according to a stationary probability p_i that is independent of past accesses. Much research in broadcast scheduling and caching assumes independent references with stationary probabilities. We believe that the independent reference assumption is in most cases a good first order approximation that leads to valuable algorithms and to a first conceptual clarification of the problem at hand. We

0-7803-7476-2/02/$17.00 (c) 2002 IEEE.

also venture that substantial performance improvements and a better understanding can be derived from bringing to light the complex dependencies in data access patterns.

Future Work: The techniques in this paper have been validated in the context of multicast dissemination of Web contents. It is natural to conjecture that access pattern dependencies exist in other contexts as well, and so the algorithms in this paper should be applicable to a variety of other scenarios, as for example, wireless information stations or satellite-supported CDN's. We are actively working at extending the scope of our measurements. Furthermore, we plan to implement a platform to support data management issues for Internet data dissemination [42], which would, among other objectives, allow us to obtain more direct measurements for the performance of our algorithms and their interaction with document selection, caching, congestion control, and layering.

REFERENCES

[1] K. C. Almeroth, M. H. Ammar, and Z. Fei, "Scalable delivery of Web pages using cyclic best-effort (UDP) multicast," in *Proceedings of the Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 1998)*, 1998.

[2] Michael Rabinovich, "Resource management issues in content delivery networks (CDNs)," in *DIMACS Workshop on Resource Management and Scheduling in Next Generation Networks*, 2001.

[3] Sanjeev Khanna and Vincenzo Liberatore, "On broadcast disk paging," *SIAM Journal on Computing*, vol. 29, no. 5, pp. 1683–1702, 2000.

[4] Vincenzo Liberatore, "Caching and scheduling for broadcast disk systems," in *Proceedings of the 2nd Workshop on Algorithm Engineering and Experiments (ALENEX 00)*, 2000, pp. 15–28.

[5] John W. Byers, Michael Luby, Michael Mitzenmacher, and Ashutosh Rege, "A digital fountain approach to reliable distribution of bulk data," in *Proc. Sigcomm*, 1998.

[6] "http://www.digitalfountain.com/," .

[7] "http://www.hns.com/," .

[8] "http://www.panamsat.com/," .

[9] Swarup Acharya, Rafael Alonso, Michael Franklin, and Stanley Zdonik, "Broadcast disks: Data management for asymmetric communication environments," in *Proceedings of the 1995 ACM SIGMOD Conference International Conference on Management of Data*, 1995, pp. 199–210.

[10] Qinglong Hu, Wang-Chien Lee, and Dik Lun Lee, "Performance evaluation of a wireless hierarchical data dissemination system," in *Proc. MobiCom*, 1999.

[11] Gary Herman, Gita Gopal, K. C. Lee, and Abel Weinrib, "The datacycle architecture for very high throughput database systems," in *Proceedings of the 1987 ACM SIGMOD Conference International Conference on Management of Data*, 1987, pp. 97–103.

[12] C. J. Su and L. Tassiulas, "Broadcast scheduling for information distribution," in *Proceedings of the Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 1997)*, 1997.

[13] A. Bar-Noy, R. Bhatia, J. Naor, and B. Schieber, "Minimizing service and operation costs of periodic scheduling," in *Proceedings of the Ninth ACM-SIAM Symposium on Discrete Algorithms*, 1998, pp. 11–20.

[14] Claire Kenyon, Nicolas Schabanel, and Neal Young, "Polynomial-time approximation scheme for data broadcast," in *Proceedings of the Thirtisecond ACM Symposium on the Theory of Computing*, 2000.

[15] Claire Kenyon and Nicolas Schabanel, "The data broadcast problem with non-uniform transmission times," in *Proceedings of the Tenth ACM-SIAM Symposium on Discrete Algorithms*, 1999, pp. 547–556.

[16] Nicolas Schabanel, "The data broadcast problem with preemption," in *LNCS 1770 Proc. of the 17th International Symposium on Theoretical Aspects of Computer Science (STACS 2000)*, 2000, pp. 181–192.

[17] Amotz Bar-Noy, Boaz Patt-Shamir, and Igor Ziper, "Broadcast disks with polynomial cost functions," in *Proceedings of the Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000)*, 2000.

[18] Balachander Krishnamurthy and Jennifer Rexford, *Web Protocols and Practice*, Addison-Wesley, Boston, 2001.

[19] Jayavel Shanmugasundaram, Arvind Nithrakashyap, Rajendran Sivasankaran, and Krithi Ramamritham, "Efficient concurrency control for broadcast environments," in *ACM SIGMOD International Conference on Management of Data*, 1999.

[20] R. Agrawal and P. K. Chrysanthis, "Efficient data dissemination to mobile clients in e-commerce applications," in *Proceedings of the Third IEEE Int'l Workshop on Electronic Commerce and Web-based Information Systems*, June 2001.

[21] T. Imielinski, S. Viswanathan, and B.R. Badrinath, "Energy efficient indexing on air," in *Proc. of the SIGMOD Conference*, 1994, pp. 25–36.

[22] S. Viswanathan T. Imielinski and B.R. Badrinath, "Data on air: Organization and access," *IEEE Transactions on Knowledge and Data Engineering*, vol. 9, no. 3, pp. 353–372, 1997.

[23] Sanjeev Khanna and Shiyu Zhou, "On indexed data broadcast," in *Proceedings of the Thirtieth ACM Symposium on the Theory of Computing*, 1998, pp. 463–472.

[24] E. Pitoura and P. K. Chrysanthis, "Exploiting versions for handling updates in broadcast disks," in *Proc. of the 25th Int'l Conference on Very Large Data Bases*, Sept. 1999, pp. 114–125.

[25] Swarup Acharya, Michael Franklin, and Stanley Zdonik, "Balancing push and pull for data broadcast," in *ACM SIGMOD International Conference on Management of Data*, 1997.

[26] W. Richard Stevens, *Unix Network Programming*, PTR PH, 1998.

[27] S. Deering and R. Hinden, "Internet Protocol, version 6 (IPv6) specification," RFC 2460, 1998.

[28] Michael R. Garey and David S. Johnson, *Computers and intractability*, W. H. Freeman and Co., San Francisco, Calif., 1979, A guide to the theory of NP-completeness, A Series of Books in the Mathematical Sciences.

[29] Mark D. Hansen, "Approximation algorithms for geometric embeddings in the plane with applications to parallel processing problems," in *Proceedings of the 30th Annual Symposium on Foundations of Computer Science*, 1989, pp. 604–609.

[30] R. Ravi, Ajit Agrawal, and Philip Klein, "Ordering problems approximated: single-processor scheduling and interval graph completion," in *Automata, languages and programming (Madrid, 1991)*, pp. 751–762. Springer, Berlin, 1991.

[31] Guy Even, Joseph (Seffi) Naor, Satish Rao, and Baruch Schieber, "Divide-and-conquer approximation algorithms via spreading metrics," in *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, Oct. 1995, pp. 62–71.

[32] Satish Rao and Andréa W. Richa, "New approximation techniques for some ordering problems," in *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (San Francisco, CA, 1998)*, New York, 1998, pp. 211–218, ACM.

[33] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin, *Network flows*, Prentice Hall Inc., Englewood Cliffs, NJ, 1993, Theory, algorithms, and applications.

[34] Kurt Mehlhorn and Stefan Näher, *LEDA*, Cambridge University Press, Cambridge, 1999, A platform for combinatorial and geometric computing.

[35] Martin Arlitt and Tai Jin, "Workload characterization of the 1998 World Cup web site," Tech. Rep. HPL-1999-35R1, HP Labs, 1999.

[36] K. Stathatos, N. Roussopoulos, and J. S. Baras, "Adaptive data broadcast in hybrid networks," in *Proc. 23rd International Conference on Very Large DataBases*, 1997, pp. 326–335.

[37] Paul Barford and Mark Crovella, "A performance evaluation of hypertext transfer protocols," in *Proceedings of the ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, 1999, pp. 188–197.

[38] M. H. Ammar, "Response time in a teletext system: An individual user's perspective," *IEEE Transactions on Communication*, vol. COM-35, no. 11, pp. 1159–1170, Nov. 1987.

[39] S. Battacharyya, J. F. Kurose, D. Towsley, and R. Nagarajan, "Efficient rate controlled bulk data transfer using multiple multicast groups," in *Proceedings of the Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 1998)*, 1998.

[40] L. Vicisano, L. Rizzo, and J. Crowcroft, "TCP-like congestion control for layered multicast data transfer," in *Proceedings of the Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 1998)*, 1998.

[41] S. Vishwanath and T. Imielinski, "Pyramid broadcasting for video on demand service," Tech. Rep. DCS-TR-311, Rutgers, 1994.

[42] Panos K. Chrysanthis, Vincenzo Liberatore, and Kirk Pruhs, "Middleware support for multicast-based data dissemination: A working reality," White paper, 2001.

共10頁 第10頁