Polynomial-Time Approximation Scheme for Data Broadcast

Claire Kenyon Nicolas Schabanel Neal Young

ABSTRACT

The data broadcast problem is to find a schedule for broadcasting a given set of messages over multiple channels. The goal is to minimize the cost of the broadcast plus the expected response time to clients who periodically and probabilistically tune in to wait for particular messages.

The problem models disseminating data to clients in asymmetric communication environments, where there is a much larger capacity from the information source to the clients than in the reverse direction. Examples include satellites, cable TV, internet broadcast, and mobile phones. Such environments favor the "push-based" model where the server broadcasts (pushes) its information on the communication medium and multiple clients simultaneously retrieve the specific information of individual interest. This sort of environment motivates the study of "broadcast disks" in Information Systems [1; 7].

In this paper we present the first polynomial-time approximation scheme for the data broadcast problem for the case when $W = O(1)$ and each message has arbitrary probability, unit length and bounded cost. The best previous polynomial-time approximation algorithm for this case has a performance ratio of 9/8 [6].

1. BACKGROUND AND RESULT

The input is a set $M = \{M_1, \ldots, M_m\}$ of messages, each with a probability p_i and cost c_i, and a parameter W — the number of channels. The output is (finitely described) infinite *broadcast schedule* S for the messages — specifying for each time $t = 0, 1, 2, \ldots$ and channel k, a message $S(t, k)$ (if any) to be broadcast at that time on that channel. The goal is to minimize the cost of the schedule, denoted $\mathrm{COST}(S)$ and defined as the *expected response time* plus the *broadcast cost* of S.

For a finite schedule S, the expected response time of S, denoted $\mathrm{ERT}(S)$, is defined as follows. At each time unit, each message is requested by some client with probability p_i. Once a message is requested, the client waits until the next time at which the message is scheduled on any channel (or the end of the schedule, whichever comes first). $\mathrm{ERT}(S)$ is defined to be the expected waiting time for a random request at a random time. The broadcast cost of S, denoted $\mathrm{BC}(S)$, is defined to be the total cost of scheduled messages, divided by the length of the schedule.

Throughout the paper, if any real-valued function f is defined with respect to finite schedules, then we implicitly extend it to any infinite schedule S as follows: $f(S) = \limsup_{n\to\infty} f(S_n)$, where S_n denotes S restricted to the first n time slots. Thus, the above definitions of expected response time and broadcast cost implicitly extend to infinite schedules. All of the infinite schedules considered in this paper will be periodic, in which case this extension is particularly simple.

The data broadcast problem and special cases were studied in [2; 3; 4; 5; 6; 10; 16; 19; 23]. Works studying applications and closely related problems include [1; 7; 9; 11; 12; 13; 14; 15; 17; 18; 19; 20; 21; 22; 24]. Some of the above works study the generalization allowing messages to have arbitrary lengths, which we do not consider here.

Ammar and Wong [3; 4] proved that there always exists an optimal infinite schedule with finite period. They also formulated a natural relaxation of the problem that gives an explicit lower bound on the optimum; the performance guarantee in this paper is proven with respect to that lower bound. More recently, constant-factor polynomial-time approximation algorithms have been shown [5; 6], the best to date being a 9/8-approximation [6]. Although the problem itself is not known to be *NP*-hard, several variants are known to be [6; 16; 19].

Khanna and Zhou [18, §1.2] state that it is unknown whether the problem is MAX-SNP hard, even when $W = 1$ and without broadcast costs. In this paper, we show that it is not (unless P=NP). We present the first deterministic polynomial time approximation scheme for the problem, assuming the W and each cost is bounded by a constant. By "polynomial time", we mean that the time taken to output the finite description of the infinite schedule is polynomial in the number of messages m in the input.

2. SUMMARY OF APPROACH

Our algorithm is based on a simple new observation that works for a special case of the problem. We use fairly technical but to some extent standard techniques to extend it to the general case. We sketch the idea here, glossing over a fair amount of technical detail.

Ammar and Wong [3; 4] relax the optimization problem by allowing messages to (a) be scheduled at non-integer times and (b) to *overlap*, while still insisting that the total *density* of the scheduled messages is at most W, the number of channels (the extension to the multiple channel case is due to [6]). The density of a message (or set of messages) is the total number of scheduled times, divided by the length of the schedule. Standard calculus yields a solution to this relaxed problem. The solution specifies for each message M_i a *density* d_i^*, meaning that the message should be scheduled every $\tau_i^* = 1/d_i^*$ time units.

Ammar and Wong describe the following simple randomized rounding algorithm for producing a real schedule: *For* $t = 1, 2, \ldots,$ *for* $k = 1 \ldots W$, *choose a single message* M_i *randomly so that* $\Pr\{M_i \text{ selected}\}$ *is* $d_i^*/W = W/\tau_i^*$; *schedule* M_i *in schedule slot* $S(t, k)$. They observe that the expected waiting time for a random request for M_i is essentially τ_i^* in this schedule. Since the expected waiting time in the relaxed schedule is essentially $\tau_i^*/2$ (because an average request falls midway between two successive broadcasts of M_i), this yields a 2-approximation w.r.t. expected response time. Since the expected broadcast cost of S is the same as the broadcast cost of the relaxed solution, the algorithm is a 2-approximation algorithm w.r.t. the total cost. Ammar and Wong also describe a greedy algorithm that Bar-Noy, Bhatia, Naor and Schieber generalize in [6] to the multiple channel case and prove to be essentially a derandomization of the randomized algorithm, with the same performance guarantee.

Round-robin within groups. Since our goal is a PTAS, we naturally group messages that are essentially equivalent (i.e. have essentially the same cost and probability). Our simple idea is the following variation of Ammar and Wong's rounding scheme, which is most simply described as follows: Schedule the messages as Ammar and Wong do, but then, *within each group, rearrange the messages so that they are scheduled in round-robin (cyclic) order.* The broadcast cost is unchanged, but the expected response time improves as follows. Whereas before, a random request for a message M in a group G would have waited (in expectation) for $|G|$ messages from G until finding its message, in the round-robin schedule, a random request for M will wait (by symmetry) for $(1+2+\cdots+|G|)/|G|) = (|G|+1)/2$ messages from G. That is, the expected wait in the round-robin schedule is $(|G|+1)/(2|G|)$ times the expected wait in the Ammar-Wong schedule. Since the Ammar-Wong schedule has performance guarantee 2, the round-robin schedule has performance guarantee $\max_G(|G|+1)/|G| = 1 + 1/\min_G |G|$. Thus, when the groups are all large, the Ammar-Wong relaxation is essentially tight.

Extending to the general case. Recall that for our purposes a group is a collection of messages with approximately (w.r.t. ε) the same probability and cost. As long as each group has size at least $1/\varepsilon$, the round-robin schedule gives a $(1+\varepsilon)$-approximation.

To extend to the general case, we show the following. *Any* set of messages can be partitioned into three classes as follows:

- A — A constant number of *important* (high probability) messages.
- B — Messages belonging to *large groups.*
- C — Leftover messages, contributing *negligibly* to the cost.

The basic intuition for the existence of this partition is that, due to the rounding, the message-probabilities of the successive groups decrease exponentially fast. Thus, for all but a constant number of groups (where the message-probability is high), either the group is very large, or the total probability of the messages in the group is very small. Althouth the intuition is basic, obtaining the proof with the appropriate parameters is is somewhat involved and delicate.

Once we have the partition, we proceed as follows:

1. Find the density α of messages in A in a near-optimal schedule of A and B.
2. Compute an optimal "short" schedule S_A of A having density approximately α.
3. Schedule the messages in B in the slots not occupied by A, using the group-round-robin algorithm.
4. "Stretch" the schedule, interspersing empty slots every $1/\varepsilon$ time units, and schedule the messages for C in these empty slots.

Note that in order to "cut and paste" the schedules together, we have to explicitly control the density of A and B. This in itself requires little that is new. The main new difficulty is the following. In step 3, we are using the round-robin algorithm to schedule B, but in a schedule that is already partially filled by A. For the analysis of the round-robin algorithm to continue to approximately hold, we require that the empty slots in schedule S_A are sufficiently *evenly distributed* so that the scheduling of B is not overly delayed at any time (cost increases quadratically with delay).

A-priori, imposing this additional requirement on S_A might increase the cost of S_A too much. To show that this is not the case, we show (using a non-constructive probabilistic argument) that there is a schedule of A that has *constant-length period*, density approximately α, and cost approximately the cost of any optimal schedule of A with density α. Since the period of this schedule is small, the empty slots are *necessarily* evenly distributed.

The final output of the algorithm is a finite (size linear in the input size) description from which an infinite schedule with approximately optimal expected cost can be generated by a randomized algorithm in an "on-line" fashion, where each step requires $O(W)$ time to schedule.

The running time of the various steps is as follows. In step 1, only a constant number of densities α need to be

considered: we can try them all and take the best. For each α, the time for the remaining steps is as follows. Step 2 can be done in constant time since the schedule we are looking for has constant length. Step 3 can be done in randomized time in the size of the output. Step 4 can also be done in randomized linear time in the size of the output.
The final technical hurdle is showing that the algorithm can be derandomized (extending the analysis of the greedy algorithm by Bar-Noy, Bhatia, Naor and Schieber to this more complicated setting). The resulting deterministic algorithm outputs a polynomial-length schedule, the repetition of which gives the desired near-optimal infinite schedule.

3. GROUP ROUND ROBIN

Let the set of messages M be partitioned into groups $G_1, \ldots, G_q$ where group G_j has size g_j every message in G_j has the same probability p_j and broadcast cost c_j. Let α be the desired maximum density of M in the schedule. In this notation, Ammar and Wong's relaxation of the problem is:

$$\mathrm{LB}(M,\alpha) = \begin{cases} \min\limits_{\tau>0} \sum\limits_{j=1}^{q} \dfrac{p_j g_j^2 \tau_j}{2} + \dfrac{c_j}{\tau_j} \\ \text{Subject to: } \sum\limits_{j=1}^{q} \dfrac{1}{\tau_j} \leqslant \alpha W \end{cases}$$

Lemma 1 (Lower Bound [3]) *The minimization problem* $\mathrm{LB}(B,\alpha)$ *is a lower bound to the contribution of the messages of B to the cost of any schedule S over W channels, in which B has density* $\leqslant \alpha$.
The problem has a unique solution τ^* satsifying: $g_j\tau_j^* = \sqrt{(2c_j+\lambda^*)/p_j}$, *for some* $\lambda^* \geqslant 0$. *If* $\sum_{j\in B} g_j\sqrt{p_j/(2c_j)} \leqslant \alpha W$, *then* $\lambda^*=0$*; otherwise,* λ^* *is the unique solution to:* $\sum_{j\in B}\sqrt{p_j/(2c_j+\lambda^*)} = \alpha W$.

Lemma 2 (Randomized Approximation) *In the setting of this section, the randomized algorithm 1 constructs a one-channel schedule S whose cost satisfies:*

$$\mathbb{E}[\mathrm{COST}(S)] = \sum_{j=1}^{q}\left(p_j \frac{g_j(g_j+1)}{2}\tau_j + \frac{c_j}{\tau_j}\right) - \frac{1}{2}$$

If τ is chosen in order to minimize $\mathrm{LB}(M,1)$*, then algorithm 1 is a* $\max_j(1+1/g_j)$*-approximation.*

Algorithm 1 Group round-robin algorithm

if $\sum_{j=1}^{q} 1/\tau_j < 1$ **then**
. Add a dummy group G_0 with $p_0 = c_0 = g_0 = 0$ and $1/\tau_0 = 1 - \sum_{j=1}^{q} 1/\tau_j$.
for $t = 1..\infty$ **do**
. Draw at random a group G_j with probability $1/\tau_j$. Schedule in time slot t, the next message of group G_j in Round Robin order, if $j \neq 0$; and stay idle otherwise.

PROOF. A message of G_j is broadcast during a time slot with probability $1/\tau_j$, then the average density of the group G_j is then $1/\tau_j$. Then: $\mathbb{E}[BC(S)] = \sum_j c_j/\tau_j$. As explained above, a request for a message in G_j waits on average $1/2$ until the end of the current time slot and then $(g_j+1)/2$ broadcasts of a message in G_j on average. Then: $\mathbb{E}[\mathrm{ERT}(S)] = \frac{1}{2} + \sum_j p_j g_j \tau_j \frac{g_j+1}{2}$.
Finally if $\tau = \tau^*$, then since $\sum_j p_j g_j = 1$ and $\tau^* \geqslant 1$, we get that: $\mathbb{E}[\mathrm{COST}(S)] \leqslant \sum_j (p_j g_j \frac{g_j+2}{2}\tau_j^* + c_j/\tau_j^*)$, hence the claimed performance ratio. □

Remark 1 Note that the law of large numbers implies that the expected cost is obtained with probability 1.

4. SCHEDULING A AND B

Next we treat the case where the set of the messages M can be partitioned into two sets A and B such that

- A consists of a constant number of messages
- B is partitioned into groups as in the previous section, such that each group has size at least $\kappa(\varepsilon)|A|^2$, where $\kappa = \kappa(\varepsilon)$ will be defined later.

Recall from the discussion in the introduction that the challenge at this point is to show that there is a near-optimal schedule of A with the appropriate density α and in which the empty slots are relatively uniformly distributed. If so, then we can find the desired schedule for A by exhaustive search, and then schedule B into the empty space in the schedule using the round-robin algorithm previously described.
To show the existence of the desired schedule for A, we show there is a near-optimal schedule of A with the appropriate density and with *constant period* (independent of α).

Lemma 3 *Given a set of messages A, with cost at most $\mathcal{C}$, some constant $0 < \varepsilon < 1$ and a density $0 < \alpha < 1$, there exists a periodic schedule S satsifying:*

1. *The density of empty slots is S is approximately $(1-\alpha)$:*
   $1-\alpha(S) \geqslant (1-\varepsilon)(1-\alpha)$, *and* $\alpha(S) \in]0,1[$
2. *The cost of S is approximately optimal:*
   $$\mathrm{COST}(S,A) \leqslant (1+\varepsilon)\,\mathrm{OPT}(A,\alpha) + \varepsilon/2$$
3. *The period T_S of S can be bounded:*
   $$T_S \leqslant \frac{40\ln(1+4/\varepsilon)}{\varepsilon^4(1-\varepsilon/6)}\max(\mathcal{C},1)\cdot|A|^2$$

PROOF SKETCH. Our proof uses the probabilistic method. The main new, simple idea, is in the construction, which efficiently smoothes the cost function by erasing its possible wide variations over time in the particular schedule under study.
Let T be a parameter to be determined later. Let S^* be a periodic schedule of A with density α and which is nearly optimal: basically, $\mathrm{COST}(S^*,A) \leqslant (1+$

$\varepsilon)\,\mathrm{OPT}(A,\alpha)$. Let T^* denote the period of S^*, which w.l.o.g. is a multiple of T.
From S^*, construct another periodic schedule S_2 by inserting in S^*, every T steps from a random starting point, all the messages of A in a fixed order. S_2 is thus structured into blocks of length $T+|A|$. Let S_n be the random schedule obtained by concatenating n blocks chosen at random from S_2. For suitable values of T and n, we can prove that with positive probability, S_n satisfies the first two statements of the Lemma. The period of S_n is clearly $n(T+|A|)$, which together with the choice of T gives the third statement of the Lemma. □

Notation 1 Let $T(\varepsilon) = \dfrac{40\ln(1+4/\varepsilon)}{\varepsilon^4(1-\varepsilon/6)}\max(\mathcal{C},1)$ denote the bound of the lemma for $\mathrm{period}(S)/|A|^2$. Parameter κ will be defined as $\kappa(\varepsilon) =_{\mathrm{def}} 2WT(\varepsilon)/\varepsilon$.

The algorithm for scheduling $A\cup B$ is given below, and is an ε-approximation for the cases studied in this section. Since this is the critical case, the analysis is promoted from a lemma to a proposition.

Proposition 1 *Let $\varepsilon < 1/7$. In the setting of this section, Algorithm 2 yields a schedule S, such that:*

$$\mathbb{E}[\mathrm{COST}(S)] \leqslant (1+5\varepsilon)\,\mathrm{OPT}(A\cup B)$$

Algorithm 2 Scheduling A and B

for $x = 1..(W\cdot T(\varepsilon)|A|^2 - 1)$ **do**
- Compute an optimal periodic schedule S_α of A with density $\alpha = x/(W\cdot T(\varepsilon)|A|^2)$ and period $\leqslant T(\varepsilon)|A|^2$.

- Choose α_0 which minimizes:
$$\mathrm{COST}(S_{\alpha_0},A) + \mathrm{LB}(B, 1-\alpha_0)$$
- Compute the τ^* that minimizes $\mathrm{LB}(B,1-\alpha_0)$; Add a dummy group G_0 with $p_0 = c_0 = 0$ and τ_0^*, such that: $1/\tau_0^* = (1-\alpha_0)W - \sum_{j\in B} 1/\tau_j^*$.

Output:
for $t = 1..\infty$ **do**
- Schedule during time slot t, the same messages of A on the same channels, as in S_{α_0}.
for all empty slot s during time slot t **do**
  - Draw a group G_j of B with probability $1/(\tau_j^*(1-\alpha_0)W)$. Schedule in slot s, the next message of G_j in Round Robin order, if $j\neq 0$; and stay idle otherwise.

Proof sketch. The proof works in two steps:

1. Scheduling the messages of B with the randomized algorithm 1 in the empty slots achieves a good approximation of $\mathrm{COST}(S_{\alpha_0},A) + \mathrm{LB}(B,1-\alpha_0)$ (Using the mapping lemma 10).
2. $\mathrm{COST}(S_{\alpha_0},A) + \mathrm{LB}(B,1-\alpha_0)$ is a good approximation of the optimal cost (Using Lemma 3).

□

5. C — THE NEGLIGIBLE MESSAGES

In this section, to show how to incorporate the "negligible" messages into the schedule. We assume that the set of messages M is partitioned into two sets AB and C, where C has a "negligible contribution" to the cost. (This section can be skipped by the reader who is in a hurry).

Definition 1 A subset of messages $C\subseteq M$ has *negligible contribution* if its contribution to the lower bound is $O(\varepsilon)$, when it is scheduled on one channel with density $O(\varepsilon/\mathcal{C})$, *i.e.*:
$$\mathrm{LB}_{W=1}(C,\varepsilon/(10\mathcal{C})) \leqslant 3\varepsilon\,\mathrm{OPT}(M)/10$$
The constants $1/10$ and $3/10$ are arbitrary and are chosen in order to improve readability in the following results.

Basically, a subset of messages C is negligible if its contribution to the cost is small, in the schedule constructed by inserting its messages are inserted from time to time (every $O(1/\varepsilon)$ steps) into a schedule of the rest of the messages.

Lemma 4 *Consider a set of messages M, partitioned into two sets AB and C, where C has a negligible contribution to the cost. Then, for any schedule S of AB, one can construct in linear time a random schedule S' of M, such that:*
$$\mathbb{E}[\mathrm{COST}(S')] \leqslant (1+\tfrac{\varepsilon}{10})\,\mathrm{COST}(S,AB) + \tfrac{9\varepsilon}{10}\,\mathrm{OPT}(M)$$
If $\mathrm{COST}(S,AB) \leqslant \mathrm{OPT}(M)$, this is a $(1+\varepsilon)$-approximation.

Proof. We first construct from S another schedule S_1 of AB by inserting an empty slot, on all the channels, every $10\mathcal{C}/\varepsilon - 1$ slots, starting at a random point in $\{0,\dots,10\mathcal{C}/\varepsilon-2\}$. The stretching lemma 8 ensures that: $\mathbb{E}[\mathrm{COST}(S_1)] \leqslant (1+\varepsilon/10)\,\mathrm{COST}(S,AB)$. Let τ^* be the solution to the minimization problem $\mathrm{LB}_{W=1}(C,\varepsilon/(10\mathcal{C}))$. We obtain S' by scheduling the messages of C on the first channel in the empty slots of S_1, according to the randomized algorithm 1 with $\tau = 10\tau^*\mathcal{C}/\varepsilon$. Lemma 2 and the scaling lemma 9 ensure that the expected contribution of C is bounded by $3\,\mathrm{LB}(C,\varepsilon/(10\mathcal{C})) \leqslant 9\varepsilon\,\mathrm{OPT}(M)/10$. □

Remark 2 The algorithm above can easily be derandomized by trying all the starting point and choosing the one that minimizes the over cost for the messages of AB and use the greedy algorithm 4 to schedule C.

6. PTAS FOR DATA BROADCAST

We now assume that we are in the general case. The aim of the section is to prove the following theorem, which is the main result of the paper.

Theorem 1 (PTAS) *Given $\varepsilon < 1/7$ and a set M of messages, with message costs bounded by $\mathcal{C}$, Algorithm 6* *constructs in $O(m^2)$ time a periodic schedule S with period $\leqslant (m^2 + m\max(1,\mathcal{C}))/\varepsilon$, so that:*

$$\mathrm{COST}(S) \leqslant (1+11\varepsilon)\,\mathrm{OPT}(M)$$

We will first derive a PTRAS that will be derandomized in Section 6.2.

6.1 Randomized

We now need to put together the ideas developed for the special cases of the previous sections. As a preliminary treatment, we use standard rounding techniques to reduce the number of different messages.

Lemma 5 (Rounding) *Without loss of generality, we can assume that the request probabilities p_i are a multiple of powers of $1/(1+\varepsilon)$ and the broadcast costs are multiples of ε/W:*

$$\begin{aligned} p_i &= r/(1+\varepsilon)^j, \text{ for some } j \geqslant 1 \\ c_i &= k\cdot\varepsilon/W, \text{ for some } k \in \{0,\dots,C\cdot W/\varepsilon\} \end{aligned}$$

where $1 < r \leqslant 1+\varepsilon$.

PROOF. Standard and omitted. □

The following lemma is the main tool for putting together the various special cases studied so far, and is thus a key part of our construction. We would like to claim that similar ideas could be applied to other problems as well, however we were unable to abstract simple and general ideas from the technical proof. Perhaps, if one believes that every approximation scheme rests on one "structural lemma", it can be seen as the structural lemma for this problem.

Lemma 6 (Partition) *Given $\varepsilon > 0$ and $\kappa > 0$, one can construct, in linear time in m, a partition of the groups $(G_{j,k})$, of messages with probability $r/(1+\varepsilon)^j$ (where r is the normalizing constant such that $\sum_{\{j\geqslant 1; k=0..\mathcal{C}W/\varepsilon\}} rg_{j,k}/(1+\varepsilon)^j = 1$) and cost $k\varepsilon/W$, into three sets A, B, C so that:*

1. *The groups of A have total size constant: $|A| =_{\mathrm{def}} \sum_{G_{j,k}\in A} g_{j,k} = O_{\varepsilon,\kappa,\mathcal{C},W}(1)$, independent of m.*
2. *The groups of B are all large:* $$\forall (G_{j,k}\in B), g_{j,k} \geqslant \kappa |A|^2$$
3. *The messages in C have negligible contribution if they are scheduled rarely (with density $O(\varepsilon/\mathcal{C})$):* $$\mathrm{LB}_{W=1}(C, \tfrac{\varepsilon}{10\mathcal{C}}) \leqslant \tfrac{3\varepsilon}{10}\,\mathrm{OPT}(M)$$

PROOF SKETCH. Since the proof is rather technical, we will only in this extended abstract give the construction of the partition into A, B and C in the case when there are no costs ($\mathcal{C} = 0$) and there is only one broadcast channel ($W = 1$); this already contains the gist of the proof.

Let $a = (1+\varepsilon)^{-1} < 1$. In the case where there are no costs, the lower bound can be solved explicitly (see [3; 6]) even when there is a density constraint, to yield, for any subset X of the message set:

$$LB(X,\alpha) = \tfrac{r}{2\alpha}\left(\textstyle\sum_{G_j\in X} g_j a^{j/2}\right)^2$$

The construction. The construction is best understood by referring to figure 1. We first deal with indices such that $g_j \leqslant a^{-j/4}$. Let j_0 be some constant to be defined later, and define $C_1 = \{(j : j > j_0,\ g_j \leqslant a^{-j/4}\}$, and $A_1 = \{j : j \leqslant j_0,\ g_j \leqslant a^{-j/4}\}$. (One can observe already that since the contributions of the messages of C_1 form the tail of a geometrically decreasing series, they will be negligible, and so they will end up in C; moreover, since j and g_j are both bounded for the definition of A_1, set A_1 can only contain a small number of messages and so these messages will end up in A).

We now consider the more delicate case of the groups for which $g_j > a^{-j/4}$, for which we will need to use the pigeon hole principle. We partition their indices into $(20/\varepsilon)$ blocks as follows:

$$\begin{aligned} \Lambda_1 &= \{(j,k) : \mu \leqslant j < \mu^2\}, \\ \Lambda_2 &= \{(j,k) : \mu^2 \leqslant j < \mu^3\}, \dots \\ \Lambda_{20/\varepsilon} &= \{(j,k) : \mu^{20/\varepsilon} \leqslant j < \mu^{1+20/\varepsilon}\}, \end{aligned}$$

where μ is some constant to be defined later. According to [3], we can then rewrite the lower bound on the expected response time as $\sqrt{2LB(M)/r} = \sum_j g_j a^{j/2} \geqslant \sum_h \sum_{j\in\Lambda_h} g_j a^{j/2}$, and the pigeon hole principle tells us that there exists at least one h such that $\sum_{j\in\Lambda_h} g_j a^{j/2} \leqslant \frac{\varepsilon}{20}\sqrt{2LB(M)/r}$. We now define $A_2 = \{j : j < \mu^h,\ g_j > a^{-j/4}\}$, $C_2 = \Lambda_h$, and $B = \{j : j > \mu^h,\ g_j > a^{-j/4}\}$.

Finally we set $A = A_1 \cup A_2$ and $C = C_1 \cup C_2$ as shown on Figure 1.

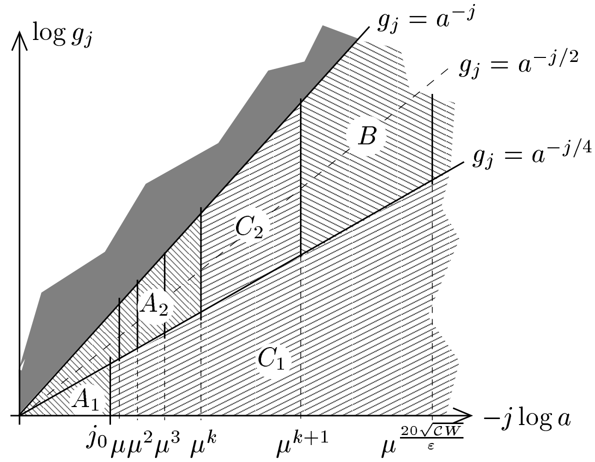

Figure 1: The partition.

□

It is now a simple matter to take our building blocks and deduce a randomized approximation scheme for the general Data Broadcast problem.

Proposition 2 (PTRAS) *Given $0 < \varepsilon < 1/7$, the randomized algorithm 3 yields a random schedule S with cost:*

$$\mathbb{E}[\mathrm{COST}(S)] \leqslant (1+10\varepsilon)\,\mathrm{OPT}(M)$$

Algorithm 3 A PTRAS

. Round the probabilities and costs of the messages in M, and partition the set of messages M into three sets A, B, C, according to Lemma 6 with $\kappa = \kappa(\varepsilon)$.
. Schedule A and B with algorithm 2.
. Insert the messages of C into the schedule of A and B, with the algorithm described in Lemma 4.

PROOF. Consider the rounded instance $\dot{M}$ of the set of messages. According to the previous Proposition 1 and Lemma 4, we have:
$$\mathbb{E}[\mathrm{COST}(S)] \leqslant (1+\varepsilon)(1+5\varepsilon)\,\mathrm{OPT}(\dot{M})$$
But Lemma 5 ensures that:
$$\mathrm{OPT}(\dot{M}) \leqslant (1+3\varepsilon)\,\mathrm{OPT}(M)$$
which yields the result. □

Note 1 The insertion of C can be done at the same time than the broadcast of A and B in Algorithm 3.

6.2 Derandomization

The PTRAS has one slight problem, namely, that it is not periodic, hence may be somewhat awkward to implement in some settings. In this section we derandomize it using greedy choices, and show how to control the period of the resulting algorithm.

Definition 2 (State) We define the *state* $(s^t_{j,k})_{1\leqslant j\leqslant q, 1\leqslant k\leqslant g_j}$ at slot t as the time period elapsed from the beginning of the k^{th} of the g_j last broadcasts of group G_j to the end of slot t, as shown Figure 2.

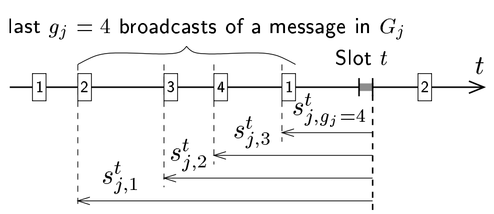

Figure 2: Definition of the state at time slot t.

Lemma 7 (Derandomization of Algorithm 1) *Given a set of messages partitioned into groups $G_1, \ldots, G_q$ of size g_j, and a set of reals $\tau_j > 0$ so that $\sum_j 1/\tau_j \leqslant 1$, the greedy algorithm 4 yields a one-channel schedule S whose cost satisfies:*

$$\mathrm{COST}(S) \leqslant \sum_{j=1}^{q}\left(p_j \frac{g_j(g_j+1)}{2}\tau_j + \frac{c_j}{\tau_j}\right) - \frac{1}{2}$$

If τ minimizes $\mathrm{LB}(M,1)$*, we get a* $\max_j(1+1/g_j)$*-approximation.*

Algorithm 4 Greedy Algorithm

. Add a dummy group G_0, if needed.
for $t = 1..\infty$ **do**
  . Let $(s_{j,k})$ be the state at time slot $t-1$.
  . Let $j \in \{0, \ldots, q\}$ which minimizes:
$$(c_j - p_j\tau_j \textstyle\sum_{k=1}^{g_j} s_{j,k})$$
  . Schedule during slot t, the next message of G_j in the Round Robin order, if $j \neq 0$, and stay idle otherwise.

PROOF SKETCH. The greedy choice at time slot t is made in order to minimize the expected cost of the already allocated slots $1, \ldots, (t-1)$, if the schedule continues with the randomized algorithm 1 after time t; this property ensures that the greedy schedule is at least as good as the randomized one. □

The above greedy algorithm could conceivably have very large period. The lemma below shows that we can truncate it so as to obtain a periodic schedule of polynomial length.

Corollary 1 (Greedy periodic schedule) *Given a set of messages partitioned into groups $G_1, \ldots, G_q$ of size g_j, a set of reals $\tau_j > 0$ such that $\sum_j 1/\tau_j \leqslant W$, and any $T \geqslant (8m^2 + (4\mathcal{C}-1)m)$, Algorithm 5 yields a one-channel schedule S with period $(T+2m)$, whose cost is bounded by:*

$$\mathrm{COST}(S) \leqslant \sum_{j=1}^{q}\left(p_j \frac{g_j(g_j+1)}{2}\tau_j + \frac{c_j}{\tau_j}\right)$$

Algorithm 5 A periodic greedy algorithm

. Schedule during slot $t = 1..m$ message M_t.
. Execute the greedy algorithm during slots $t = (m+1)..(T+m)$.
. Sort in increasing order the set $\{k\tau_j : 1 \leqslant j \leqslant q; 1 \leqslant k \leqslant g_j\}$ and Schedule in slots $t = (T+m+1)..(T+2m)$ in order of increasing $k\tau_j$, the k^{th} message of group G_j in the Round Robin order.

PROOF. Omitted. □

Our main algorithm can now be found in Algorithm 6.

PROOF OF THEOREM 1. Theorem 1 is proved by analyzing the algorithm 6. The analysis is derived from the analysis of the PTRAS. The six first steps are exactly the same, except that the periodic greedy algorithm 5 is used instead of the randomized algorithm 1. Since the performance ratio in Algorithm 5 is better, the schedule S obtained Step 6 is at least as good, and is periodic with period $O(m^2)$:
$$\mathrm{COST}(S) \leqslant (1+10\varepsilon)\,\mathrm{OPT}(M)$$
We finally reduce the period in Steps 7-8 by using stretching lemma 8, which ensures that at an increase of $(1+O(\varepsilon))$ of the cost, we can extract from S a block S^* with length $\leqslant \frac{m^2+m\max(1,\mathcal{C})}{\varepsilon}$ and:

Algorithm 6 The PTAS

1. Round the probabilities and costs, and partition M into A, B, C as in the PTRAS.
2. Compute τ^* and the density α_0 and periodic schedule S_{α_0} of A to minimizes $\mathrm{LB}(B,(1-\alpha_0)$, as in Algorithm 2.
3. Compute the greedy periodic schedule S_B of B with $\tau = (\tau^*(1-\alpha_0)W)$ and with period $\{T(\varepsilon)|A|^2(1-\alpha_0)W(8m^2+(4\mathcal{C}+1)m)\} = \Theta(m^2)$.
4. Concatenate $(8m^2+(4\mathcal{C}+1)m)$ periods of S_{α_0} and map S_B into the empty slots in the natural order.
5. Compute the greedy periodic schedule S_C of C with $\tau = (10\mathcal{C}\tau^*/\varepsilon)$ where τ^* minimizes $\mathrm{LB}(C,\varepsilon/10\mathcal{C})$, and with period $\{T(\varepsilon)|A|^2(8m^2+(4\mathcal{C}+1)m)\varepsilon/10\mathcal{C}\} = \Theta(m^2)$.
6. Choose the best starting point in $\{1,\dots,\frac{10\mathcal{C}}{\varepsilon}-1\}$ and stretch the schedule of A and B by inserting a slot of S_C on the first channel every $(\frac{10\mathcal{C}}{\varepsilon}-1)$ and an empty slot on the other channels at that time. Let S be the resulting schedule.
7. Choose the best starting point in $\{1,\dots,\frac{m^2+m\mathcal{C}}{\varepsilon}-m\}$ and construct S' by stretching S by inserting the m messages in fixed order on the first channel every $(\frac{m^2+m\mathcal{C}}{\varepsilon}-m)$.
8. S' is then structured into independent blocks of length $\frac{m^2+m\mathcal{C}}{\varepsilon}$. The cheapest block S^* will be the period of our approximation.

$$\mathrm{COST}(S^*) \leqslant (1+11\varepsilon)\,\mathrm{OPT}(M)$$

□

7. TECHNICAL LEMMAS

The lemmas in this sections are useful for analyzing several of our constructions. The stretching lemma states that changing a schedule by inserting a few empty slots once in a while does not affect the expected response time.

Lemma 8 (Stretching) *Given a schedule S on W channels of M and a positive integer y, let $\kappa \geqslant \frac{y^2+y}{\varepsilon} - y$. Consider the schedule S' obtained from S by inserting y empty slots just before the time slots $x, x+\kappa, \dots, x+i\cdot\kappa, \dots$, where x is a random time in $\{1,\dots,\kappa\}$. Then:*

$$\mathbb{E}[\mathrm{ERT}(S')] \leqslant (1+\varepsilon)\,\mathrm{ERT}(S)$$

PROOF. Omitted. □

The scaling lemma is immediate.

Lemma 9 (Scaling) *Given a set of messages M and a schedule S, let S_α the schedule obtained by scaling S by a factor $1/\alpha$: S_α schedule at time t/α on some channel the same message as S at time t, and stays idle otherwise. Then:*

$$\begin{aligned}\mathrm{ERT}(S_\alpha, A) &= \frac{1}{\alpha}\cdot\mathrm{ERT}(S,A)\\ \mathrm{BC}(S_\alpha, A) &= \alpha\cdot\mathrm{BC}(S,A)\end{aligned}$$

PROOF. Immediate. □

The mapping lemma is used for analyzing the effect of inserting the messages from B into the slots left empty in the density-constrained schedule of A; these slots may be spaced irregularly.

Lemma 10 (Mapping into reserved empty slots) *Given a set of messages M, partitioned into groups of identical messages, such that all groups are larger than TW, consider a one-channel schedule S of M scheduling each group in Round Robin order, and a periodic sequence of reserved time-slots over W channels with density α and period T. Let S' be the schedule obtained by mapping the schedule S into the reserved empty slots from left to the right, then:*

$$\begin{aligned}\mathrm{ERT}(S', A) &\leqslant \frac{1}{\alpha W}\cdot\mathrm{ERT}(S,A) + T\sum_{M_i\in A} p_i\\ \mathrm{BC}(S', A) &= \alpha W\cdot\mathrm{BC}(S,A)\end{aligned}$$

PROOF. Omitted. □

Corollary 2 (Case of large groups) *In the case where M is partitioned into groups of size $\geqslant 2T/\varepsilon$, S' has a cost bounded by:*

$$\mathrm{COST}(S', A) \leqslant (1+\varepsilon)\left\{\frac{1}{\alpha W}\cdot\mathrm{ERT}(S,A) + \alpha W\cdot\mathrm{BC}(S,A)\right\}$$

PROOF. Simple calculation. □

8. REFERENCES

[1] ACHARYA, S. *Broadcast Disks: Dissemination-based Management for Asymmetric Communication Environments.* PhD thesis, Brown University, 1998.

[2] AMMAR, M. H. Response time in a Teletext system: An individual user's perspective. *IEEE Transactions on Communications COM-35,11* (Nov. 1987), 1159–1170.

[3] AMMAR, M. H., AND WONG, J. W. The design of Teletext broadcast cycles. In *Performance Evaluation* (1985), vol. 5(4), pp. 235–242.

[4] AMMAR, M. H., AND WONG, J. W. On the optimality of cyclic transmission in Teletext systems. In *IEEE Trans. on Comm.* (1987), vol. COM-35(11), pp. 1159–1170.

[5] ANILY, S., GLASS, C. A., AND HASSIN, R. The scheduling of maintenance service. To appear (http://www.math.tau.ac.il/~hassin).

[6] BAR-NOY, A., BHATIA, R., NAOR, J. S., AND SCHIEBER, B. Minimizing service and operation costs of periodic scheduling. In *Proc. of the 9th Annual ACM-SIAM Symp. on Discrete Algorithms (SODA)* (1998), pp. 11–20.

[7] Bestavros, A., and Cunha, C. Server-Initiated Document Dissemination for the WWW. *IEEE Data Engineering Bulletin*, September 1996.

[8] Bar-Noy, A., and Shilo, Y. Optimal broadcasting of two files over an asymmetric channel. In *Proc. of Infocom* (1999).

[9] Chan, M. Y., and Chin, F. Schedulers for larger classes of pinwheel instances. *Algorithmica 9* (Jun. 1993), 425–462.

[10] Gecsei, J. The architecture of videotex systems. *Prentice Hall, Englewood Cliffs, N.J.* (1983).

[11] Hadley, G., and Whitin, T. M. *Analysis of inventory systems*. Prentice-Hall, 1963.

[12] Hofri, M., and Rosberg, Z. Packet delay under the golden ratio weighted TDM policy in a multiple-access channel. In *IEEE Trans. on Information Theory* (May 1987), vol. 11(33), pp. 341–349.

[13] Holte, R., Mok, A., Rosier, L., Tulchinsky, I., and Varvel, D. The pinwheel: A real-time scheduling problem. In *Proc. 22nd Hawaii Int. Conf. Syst. Sci.* (Jan. 1989), pp. 693–702.

[14] Imielinski, T., Viswanathan, S., and Badrinath, B. Energy efficient indexing on air. In *SIGMOD* (May 1994).

[15] Itai, A., and Rosberg, S. A golden ratio control policy for a multiple-access channel. *IEEE Transactions on Autom. Contr. AC-29* (Aug. 1984).

[16] Kenyon, C., and Schabanel, N. The data broadcast problem with non-uniform transmission times. In *Proc. of the 10th SODA* (1999), pp. 547–556.

[17] Khanna, S., and Liberatore, V. On broadcast disk paging. In *Proceedings of 30th STOC* (1998), vol. 30, pp. 634–643.

[18] Khanna, S., and Zhou, S. On indexed data broadcast. In *Proceedings of 30th STOC* (1998), vol. 30, pp. 463–472.

[19] Schabanel, N. The data broadcast problem with preemption. In *LNCS Proc. of the 17th Symp. on Theor. Aspects of Comp. Sc. (STACS 2000)* (Feb. 2000, to be published).

[20] Shekhar, S., and Liu, D. Genesis: An approach to data dissemination in Advanced Traveller Information Systems (ATIS). In *IEEE Data Engineering Bulletin* (Sept. 1996), vol. 19(3).

[21] Su, C.-J., and Tassiulas, L. Broadcast scheduling for information distribution. In *Proc. of INFOCOM* (1997).

[22] Tan, K., and Xu, J. Energy efficient filtering of nonuniform broadcast. In *Proc. of the 16^{th} Int. Conf. in Distributed Computing System* (1996), pp. 520–527.

[23] Vaidya, N., and Hameed, S. Log time algorithms for scheduling single and multiple channel data broadcast. In *Proc. of the 3rd ACM/IEEE Conf. on Mobile Computing and Networking (MOBICOM)* (Sep. 1997).

[24] Vishwanath, S. *Publishing in wireless and wireline environments*. PhD thesis, Rutgers University, 1994.