# The Minimum Range Assignment Problem on Linear Radio Networks

A.E.F. Clementi* A. Ferreira† P. Penna* S. Perennes† R. Silvestri‡

**Abstract**

Given a set $S$ of radio stations located on a line and an integer $h$ ($1 \leq h \leq |S| - 1$), the Min Assignment problem is to find a range assignment of minimum power consumption provided that any pair of stations can communicate in at most $h$ hops. Previous positive results for this problem were known only when $h = |S| - 1$ (i.e. the unbounded case) or when the stations are equally spaced (i.e. the uniform chain). In particular, Kirousis, Kranakis, Krizanc and Pelc (1997) provided an efficient exact solution for the unbounded case and efficient approximated solutions for the uniform chain, respectively.

This paper presents the first polynomial time, approximation algorithm for the Min Assignment problem. The algorithm guarantees an approximation ratio of 2 and runs in time $O(hn^3)$.

We also prove that, for constant $h$ and for "well spread" instances (a broad generalisation of the uniform chain case), we can find a solution in time $O(hn^3)$ whose cost is at most an $(1 + \epsilon(n))$ factor from the optimum, where $\epsilon(n) = o(1)$ and $n$ is the number of stations. This result significantly improves the approximability result by Kirousis *et al* on uniform chains.

Both of our approximation results are obtained by new algorithms that exactly solves two natural variants of the Min Assignment problem that might have independent interest: the *All-To-One* problem (in which every station must reach a fixed one in at most $h$ hops) and the *Base Location* problem (in which the goal is to select a set of *Basis* among the stations and all the other stations must reach one of them in at most $h - 1$ hops).

Finally, we show that for $h = 2$ the Min Assignment problem can be solved in $O(n^3)$ time.

## 1 Introduction

At the present time, *Radio Networks* play a vital role in Computer Science and in several aspects of modern life. The proliferation of their applications, such as cellular phones and wireless local area networks, is due to the low cost of infrastructure and flexibility. In general, radio networks are adopted whenever the construction of more traditional networks is impossible or, simply, too expensive.

A *Multi-Hop Packet Radio Network* [14] is a finite set of radio stations located on a geographical region that are able to communicate by transmitting and receiving radio signals. In ad-hoc networks a transmission range is assigned to each station $s$ and any other station $t$ within this range can directly (i.e. by one *hop*) receive messages from $s$. Communication between two stations that are not within their respective ranges can be achieved by *multi-hop* transmissions. One of the main

*Dipartimento di Matematica, Università di Roma "Tor Vergata". E-mail: {clementi,penna}@mat.uniroma2.it.
†MASCOTTE Project, I3S-CNRS-INRIA Sophia Antipolis. E-mail: {ferreira,perennes}@sophia.inria.fr.
‡Dipartimento di Matematica Pura e Applicata, Università de L'Aquila. E-mail: silver@dsi.uniroma1.it.

benefits of ad-hoc networks is the reduction of the power consumption. This can be obtained by suitably varying the transmission ranges.

It is reasonably assumed [14] that the power $P_t$ required by a station $t$ to correctly transmit data to another station $s$ must satisfy the inequality

$$\frac{P_t}{d(t,s)^\beta} > \gamma \tag{1}$$

where $d(t,s)$ is the distance between $t$ and $s$, $\beta \geq 1$ is the *distance-power gradient*, and $\gamma \geq 1$ is the *transmission-quality* parameter. In an ideal environment (see [14]) $\beta = 2$ but it may vary from 1 to more than 6 depending on the environment conditions of the place the network is located. In the rest of the paper, we fix $\beta = 2$ and $\gamma = 1$, however, our results can be easily extended to any $\beta, \gamma \geq 1$.

Given a set $S = \{s_1, \dots, s_n\}$ of radio stations on the $d$-dimensional Euclidean space, a *range assignment* for $S$ is a function $r : S \rightarrow \mathcal{R}^+$ (where $\mathcal{R}^+$ is the set of non negative real numbers), and the *cost* of $r$ is defined as

$$\mathsf{cost}(r) \;=\; \sum_{i=1}^{n} r(s_i)^2.$$

Given an integer $h$ and a set $S$ of stations on the $d$-dimensional Euclidean space, the Min d-Dim Assignment problem is to find a minimum cost range assignment provided that the assignment ensures the communication between any pair of stations in at most $h$ hops. The Min 1-Dim Assignment problem (i.e. the linear case) will be simply denoted as Min Assignment. In this work we focus on the *linear* case, that is networks that can be modelled as sets of stations located along a line. As pointed out in [13], rather then a simplification, this version of the problem results in a more accurate analysis of the situation arising, for instance, in vehicular technology applications. Indeed, it is common opinion to consider one-dimensional frameworks as the most suitable ones in studying road traffic information systems. Indeed, vehicles follow roads, and messages are to be broadcasted along lanes. Typically, the curvature of roads is small in comparison to the transmission range (half a mile up to some few miles). Motivated by such applications, linear radio networks have been considered in several papers (see for instance [3, 8, 12, 13]).

## 1.1 Previous Results

Routing, broadcasting and scheduling problems on radio networks have been the subject of several papers over the last years [1, 5, 4, 9, 14, 16, 17]. Tradeoffs between connectivity and power consumption have been obtained in [7, 11, 15, 18].

As for the Min d-Dim Assignment problem, several complexity results have been obtained for the *unbounded case* (i.e. when $h = n - 1$). Under this restriction, Min 3-Dim Assignment is APX-complete (in [11], a polynomial time 2-approximation algorithm is given while in [6], the APX-hardness is proved); the Min 2-Dim Assignment problem is NP-hard [6]; finally, the Min Assignment problem is in P via an $O(n^4)$-time algorithm [11]. On the other hand, few results are known for the general case (i.e. for arbitrary $h$). In [11], the following bounds on the optimum cost have been proved for a strong restriction of Min Assignment.

**Theorem 1 (The Uniform Chain Case [11])** *Let $N$ be a set of $n$ collinear points equally spaced at distance $\delta > 0$; let $\mathsf{OPT}_h(N)$ be the cost of an optimum solution for* Min Assignment *on input $h$ and $N$. Then, it holds that*

- $\mathsf{OPT}_h(N) = \Theta\left(\delta^2 n^{\frac{2^{h+1}-1}{2^h-1}}\right)$, *for any fixed positive integer* $h$;
- $\mathsf{OPT}_h(N) = \Theta\left(\delta^2 \frac{n^2}{h}\right)$, *for any* $h = \Omega(\log n)$.

*Furthermore, the two above (implicit) upper bounds can be efficiently constructed.*

Although the constructive method of Theorem 1 yields approximated solutions for the uniform chain case, no approximation result is known for more general configurations. Moreover, the approximation ratio guaranteed by Theorem 1, for constant $h$, *increases with* $h$. We then observe that for *non constant* values of $h$ such that $h = o(\log n)$ no approximation algorithm *even for the uniform chain* restriction is known.

As for the Min 2-Dim Assignment problem, some upper and lower bounds on the optimum cost function, for constant values of $h$, have been derived in [7].

### 1.2 Our Results

We present the first polynomial time approximation algorithm for the Min Assignment problem. The algorithm guarantees an approximation ratio of 2 and runs in time $O(hn^3)$.

Then, we provide a better approximation that works on any family of *well spread* instances and for any constant $h$; in such instances, the ratio between the maximum and the minimum distance among adjacent stations is bounded by a polylogarithmic function of $n$ (see Sect. 4.2 for a formal definition). More precisely, we show that, for any well spread instance and for any constant $h$, it is possible to compute in time $O(hn^3)$ a solution whose cost is at most an $(1 + \epsilon(n))$ factor from the optimum, where $\epsilon(n) = o(1)$. Since uniform chains are a (very strong) restriction of well spread instances, our result strongly improves that in Theorem 1 in the case $h = O(1)$. Indeed, the obtained approximation ratio tends to 1 (while, as already observed, the approximation ratio achieved by Theorem 1 is an increasing function of $h$).

Our approximability results are obtained by exploiting exact solutions for two natural variants of the Min Assignment problem that might be of independent interest:

Min All-To-One Assignment Given a set $S$ of stations on the line, a *sink* station $t \in S$, and an integer $h > 0$; find a minimum cost range assignment for $S$ ensuring that any station is able to reach $t$ in at most $h$ hops.

Min Assignment with Basis Given a set $S$ of stations on the line, and an integer $h > 0$; find a minimum cost range assignment for $S$ such that, any station in $S$ is either a *base* (a station is a base if it directly reaches any other station in $S$) or it reaches a base in at most $h-1$ hops.

For each of the two above problems, we provide an algorithm, based on dynamic programming, that returns an optimum solution in time $O(hn^3)$.

Finally, we prove that for $h = 2$, the Min Assignment problem can be solved in time $O(n^3)$. This result is obtained by combining the algorithm for the Min Assignment with Basis problem with a simple characterisation of the structure of any optimal 2-hops range assignment.

**Organisation of the paper.** In Sect. 1.3 we provide some basic definitions and notation. An efficient solution for the Min All-To-One Assignment and the Min Assignment with Basis problem is given in Sect. 2 and in Sect. 3, respectively. The approximability results are contained in Sect. 4. In Sect. 5 we describe an exact algorithm for the case $h = 2$ and in Sect. 6 we discuss some open problems.

### 1.3 Preliminaries

Let $S = \{s_1, \dots, s_n\}$ be a set of $n$ consecutive stations located on a line. We denote by $d(i,j)$ the distance between station $s_i$ and $s_j$. We define $\delta_{\min}(S) = \min\{d(i,i+1) \mid 1 \leq i \leq n-1\}$, $\delta_{\max}(S) = \max\{d(i,i+1) \mid 1 \leq i \leq n-1\}$, and $D(S) = d(1,n)$.

Given a range assignment $r : S \rightarrow \mathcal{R}^+$, we say that $s_i$ directly (i.e. in one hop) reaches $s_j$ if $r(s_i) \geq d(i,j)$ (in short $i \rightarrow_r j$). Additionally, $s_i$ reaches $s_j$ in at most $h$ hops if there exist $h-1$ stations $s_{i_1}, \dots, s_{i_{h-1}}$ such that $i \rightarrow_r i_1 \rightarrow_r i_2, \dots, \rightarrow_r i_{h-1} \rightarrow_r j$ (in short $i \rightarrow_{r,h} j$). We will omit the subscript $r$ when this will be clear from the context. We will say that $r$ is an *$h$-assignment* $(1 \leq h \leq n-1)$ if for any pair of stations $s_i$ and $s_j$, $i \rightarrow_{r,h} j$. Notice that $h$-assignments are exactly the feasible solutions for the instance $(h,S)$ of Min Assignment. The cost of an optimal $h$-assignment for a given set $S$ of stations is denoted as $\mathsf{OPT}_h(S)$. Given a station $s_i$ we will refer to its *index* as $i$. Finally, we denote the set of stations $\{s_{i+1}, \dots, s_{j-1}\}$ by $(i,j)$ and we also use $[i,j]$, $[i,j)$ and $(i,j]$ as a shorthand of $(i-1, j+1)$, $(i-1,j)$ and $(i,j+1)$, respectively.

## 2 The Min All-To-One Assignment Problem

In this section, we present an efficient method for the Min All-To-One Assignment problem which is based on a suitable use of dynamic programming. To this aim, we introduce the following functions.

**Definition 1 (All-To-One)** *Given a set $S$ of $n$ stations and for any $1 \leq i \leq j \leq n$, we define*

$$\overleftarrow{\mathsf{ALL}}_h(i,j) = \min\{\mathsf{cost}(r) \mid \forall k \in [i,j], \quad k \rightarrow_{r,h} i\};$$

$$\overrightarrow{\mathsf{ALL}}_h(i,j) = \min\{\mathsf{cost}(r) \mid \forall k \in [i,j], \quad k \rightarrow_{r,h} j\}.$$

**Definition 2 (OR)** *Given a set $S$ of $n$ stations and for any $1 \leq i \leq j \leq n$, we define*

$$\mathsf{OR}_h(i,j) = \min\{\mathsf{cost}(r) \mid \forall k \in [i,j], \quad k \rightarrow_{r,h} i \ \vee \ k \rightarrow_{r,h} j\}.$$

Such functions will also be used in Section 3 in order to solve the Min Assignment with Basis problem.

**Lemma 1** *There is an algorithm that, for any set of $n$ stations on the line, for any $1 \leq i \leq j \leq n$, and for any $h \geq 1$, computes $\overleftarrow{\mathsf{ALL}}_h(i,j)$, $\overrightarrow{\mathsf{ALL}}_h(i,j)$, $\mathsf{OR}_h(i,j)$ in time $O(hn^3)$.*

Sketch of Proof. In order to prove the lemma, we need to define two further functions:

$$\overleftarrow{\mathsf{ALL}}^*_h(i,j) = \min\{\mathsf{cost}(r) \mid \forall k \in [i,j), \quad k \rightarrow_{r,h} i \ \wedge \ j \rightarrow_r i\};$$

$$\overrightarrow{\mathsf{ALL}}^*_h(i,j) = \min\{\mathsf{cost}(r) \mid \forall k \in (i,j], \quad k \rightarrow_{r,h} j \wedge \ i \rightarrow_r j\}.$$

Our next goal is to prove the following recursive equations:

$$\overrightarrow{\mathsf{ALL}}_h(i,j) = \min_{i \leq k < j} \{\overrightarrow{\mathsf{ALL}}^*_h(k,j) + \overrightarrow{\mathsf{ALL}}_{h-1}(i,k)\}; \tag{2}$$

$$\overleftarrow{\mathsf{ALL}}_h(i,j) = \min_{i < k \leq j} \{\overleftarrow{\mathsf{ALL}}^*_h(i,k) + \overleftarrow{\mathsf{ALL}}_{h-1}(k,j)\}. \tag{3}$$

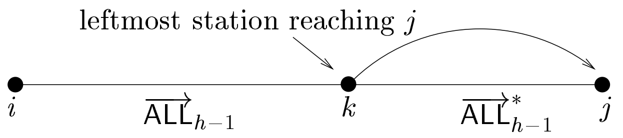


Figure 1: How to compute $\overrightarrow{\mathsf{ALL}}_h$.

In fact, consider the function $\overrightarrow{\mathsf{ALL}}_h(i,j)$ and consider any feasible range assignment $r$ for this function. Let $k$ be the index of the *leftmost* station reaching $j$ in one hop (see Fig. 1). For any station $s \in [i,k)$ it holds that $s \to_{r,h} j$ but it does not hold that $s \to_r j$ (by definition of $k$). It thus easily follows that, for any $s \in [i,k)$, it must be the case that $s \to_{r,h-1} k$. We also remark that no station in $[i,k]$ uses "bridges" in the interval $(k,j]$. This implies that $r$, restricted to $[i,k)$, is a feasible range assignment for $\overrightarrow{\mathsf{ALL}}_{h-1}(i,k)$. Furthermore, for any $s \in [k,j]$, $s \to_{r,h} j$ without using any "bridge" in $[i,k)$. Hence, $r$ restricted to $[k,j]$ is a feasible assignment for $\overrightarrow{\mathsf{ALL}}^{*}_h(k,j)$. It thus follows that

$$\mathsf{cost}(r) \;\geq\; \overrightarrow{\mathsf{ALL}}_{h-1}(i,k) + \overrightarrow{\mathsf{ALL}}^{*}_h(k,j)$$

Eq. 3 can be proved by a symmetric argument.

By using similar arguments, it is possible to prove the following other recursive equations

$$\overrightarrow{\mathsf{ALL}}^{*}_h(i,j) = \min_{i<k\leq j} \{\overrightarrow{\mathsf{ALL}}^{*}_h(k,j) + \mathsf{OR}_{h-1}(i,k)\} + d(i,j)^2; \tag{4}$$

$$\overleftarrow{\mathsf{ALL}}^{*}_h(i,j) = \min_{i\leq k<j} \{\overleftarrow{\mathsf{ALL}}^{*}_h(i,k) + \mathsf{OR}_{h-1}(k,j)\} + d(i,j)^2; \tag{5}$$

$$\mathsf{OR}_h(i,j) = \min_{i\leq k<j} \{\overleftarrow{\mathsf{ALL}}_h(i,k) + \overrightarrow{\mathsf{ALL}}_h(k+1,j)\}. \tag{6}$$

In what follows, we describe the correct "crossed" recursive computation that will return the outputs of the five functions. The overall computation goes over $h$ *phases*; in the $\ell$-th phase, all the functions will be computed for number of hops equal to $\ell$.

Phase $\ell = 1$ consists of computing the following values:

$$\forall i<j \;:\; \overleftarrow{\mathsf{ALL}}^{*}_1(i,j) = \overleftarrow{\mathsf{ALL}}_1(i,j) = \sum_{k=i+1}^{j} d(i,k)^2$$

$$\forall i<j \;:\; \overrightarrow{\mathsf{ALL}}^{*}_1(i,j) = \overrightarrow{\mathsf{ALL}}_1(i,j) = \sum_{k=i}^{j-1} d(k,j)^2$$

$$\mathsf{OR}_1(i,j) = \min_{i\leq k<j} \{\overleftarrow{\mathsf{ALL}}_1(i,k) + \overrightarrow{\mathsf{ALL}}_1(k+1,j)\}.$$

Notice that, in any Phase $\ell \geq 1$ and for any $i$, it easily holds that

$$\overleftarrow{\mathsf{ALL}}^{*}_\ell(i,i) = \overleftarrow{\mathsf{ALL}}_\ell(i,i) = \overrightarrow{\mathsf{ALL}}^{*}_\ell(i,i) = \overrightarrow{\mathsf{ALL}}_\ell(i,i) = \mathsf{OR}_\ell(i,i) = 0.$$

Now, assume that, at the end of Phase $\ell-1$, the algorithm has computed the values of the five functions for all possible segments in $[1,n]$ and for number of hops $\ell-1$. Then, the function $\overleftarrow{\mathsf{ALL}}^{*}_\ell$

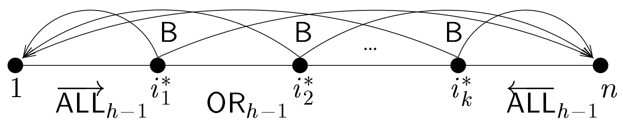


Figure 2: The structure of the optimum solution for the base location problem.

can be computed, by applying Eq. 5, for all the segments in the following order:

$$[1,1],[1,2],\ldots,[1,n],[2,2],[2,3],\ldots,[2,n],\ldots,[n-1,n],[n,n].$$

The opposite order is instead used for computing the values of $\overrightarrow{\mathsf{ALL}}^*_\ell$ by applying Eq. 4. The next two steps (in any order) are the computations of functions $\overrightarrow{\mathsf{ALL}}_\ell$ and $\overleftarrow{\mathsf{ALL}}_\ell$ for any interval in $[1,n]$ by applying Eq. 2 and Eq. 3. The last values computed at Phase $\ell$ are the $\mathsf{OR}_\ell(i,j)$ for all possible segments $(i,j)$ according to equation 6.

We finally observe that, at every Phase $\ell$, we need to compute $O(n^2)$ values, each of them requiring $O(n)$ time.

□

The above lemma easily implies the following theorem.

**Theorem 2** *The* Min All-To-One Assignment *problem can be solved in time $O(hn^3)$.*

## 3 The Min Assignment with Basis Problem

In order to provide exact solutions for the Min Assignment with Basis problem, we consider the following definitions.

**Definition 3 (base stations)** *Let $r$ be a feasible solution for the* Min Assignment *problem on input $h$ and $S$. A station $i$ is a* base *(in short* $\mathsf{B}$*) if $i \rightarrow_r 1 \wedge i \rightarrow_r n$. Moreover, $r$ is of type $\mathsf{B}^*$ if there is at least one base and any station which is not a base reaches some base in at most $h-1$ hops. Then, $\mathsf{BASES}_h(S)$ denotes the cost of an optimum assignment of type $\mathsf{B}^*$.*

Notice that $\mathsf{BASES}_h(S)$ is the optimum for the Min Assignment with Basis problem on input $h$ and $S$. The main contribution of this section can be stated as follows.

**Theorem 3** *For any set $S$ of $n$ stations on the line and for any $1 \le h \le n-1$, it is possible to construct an optimum $h$-assignment of type $\mathsf{B}^*$ for $S$ in time $O(hn^3)$. Thus, the* Min Assignment with Basis *problem is in* P.

Sketch of Proof. Let us first consider the indices $i_1^*, \cdots, i_k^*$ of the $k$ bases in the optimal solution (see Fig. 2). It is not hard to prove that, between two consecutive bases, any station must reach one of the two bases in at most $h-1$ hops. Additionally, the stations in $[1, i_1^*)$ (respectively, $(i_k^*, n]$) must reach in $h-1$ hops the base in $i_1^*$ (respectively, $i_k^*$).

Thus, given the indices of the bases in an optimal solution, we can use the functions described in Sect. 2 to find the optimal assignment. Notice that if $k$ would be always bounded by a constant then we could try all the possible indices for the bases. However, this is not the case, so a more tricky approach is needed. Basically, we will use the fact that every base "cuts" the instance into two *independent* intervals.

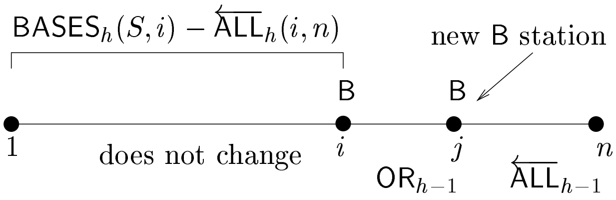


Figure 3: The proof of Theorem 3.

Let us define $\mathsf{BASES}_h(S,i)$ as the cost of the minimum $h$-assignment of type $\mathsf{B}^*$ subject to the rightmost base has index $i$. Clearly

$$\mathsf{BASES}_h(S) = \min_{1\leq i\leq n} \mathsf{BASES}_h(S,i).$$

Let us then see how to compute $\mathsf{BASES}_h(S,i)$ for any $i$. To this aim we need a function $\mathsf{CHANGE}_h(i,j)$ which, roughly speaking, corresponds to the change of the cost of $\mathsf{BASES}_h(S,i)$ when we set a *new* base $j > i$ (see Fig. 3).

It is easy to see that the new base $j$ is useless for the stations in $[1,i]$. Indeed, if a station in $[1,i]$ reaches $j$ in at most $h-1$ hops, then it also reaches $i$ within the same number of hops, thus making $j$ useless. Moreover, the cost of an optimal assignment for $\mathsf{BASES}_h(S,i)$ restricted to $(i,n]$ is $\overleftarrow{\mathsf{ALL}}_{h-1}(i,n)$. The "new cost" due to base $j$ in the interval $(i,n]$ is given by

$$\mathsf{OR}_{h-1}(i,j) + \overleftarrow{\mathsf{ALL}}_{h-1}(j,n) + \max\{d(1,j),d(j,n)\}^2.$$

We can thus define

$$\mathsf{CHANGE}_h(i,j) = \mathsf{OR}_{h-1}(i,j) + \overleftarrow{\mathsf{ALL}}_{h-1}(j,n) + \max\{d(1,j),d(j,n)\}^2 - \overleftarrow{\mathsf{ALL}}_{h-1}(i,n).$$

We can now compute $\mathsf{BASES}_h(S,\cdot)$. Let $\mathsf{BASE}^1_h(S,i)$ be the minimum cost of the range assignment in which there exists *only one* base and its index is $i$. Then, it holds that

$$\mathsf{BASE}^1_h(S,i) = \overrightarrow{\mathsf{ALL}}_{h-1}(1,i) + \max\{d(1,i),d(i,n)\}^2 + \overleftarrow{\mathsf{ALL}}_{h-1}(i,n). \qquad (7)$$

The algorithm starts by computing $\overrightarrow{\mathsf{ALL}}_{h-1}(i,j)$, $\overleftarrow{\mathsf{ALL}}_{h-1}(i,j)$, $\mathsf{OR}_{h-1}(i,j)$ by using the algorithm Lemma 1. Then, $\mathsf{CHANGE}_h(i,j)$ and $\mathsf{BASE}^1_h(S,i)$ can be computed by the above equations, for any $1 \leq i < j \leq n$.

The computation of $\mathsf{BASES}_h(S,j)$ goes over $n$ phases. Phase $j = 1$ corresponds to compute

$$\mathsf{BASES}_h(S,1) = \mathsf{BASE}^1_h(S,1). \qquad (8)$$

Assume that at phase $j-1$ $\mathsf{BASES}_h(S,i)$ for any $1 \leq i \leq j-1$ have been already computed. Then, the computation of $\mathsf{BASES}_h(S,j)$ can be carried out according to the following recursive equation:

$$\mathsf{BASES}_h(S,j) = \min\{\mathsf{BASE}^1_h(S,j), \min\{\mathsf{BASES}_h(S,i) + \mathsf{CHANGE}_h(i,j) \mid i = 1,\ldots,j-1\}\}. \qquad (9)$$

The correctness of the above equation follows from the fact that an optimal assignment for $\mathsf{BASES}_h(S,j)$ either contains only one base and its index is $j$, or it contains at least one base other than $j$. In the latter case, as previously observed, the assignment in $[1,i]$ is not used outside. Thus, such an assignment must be optimal for $\mathsf{BASES}_h(S,i)$.

Finally, it is easy to prove that the overall algorithm works in time $O(hn^3)$.

□

## 4 Approximation Algorithms

### 4.1 A 2-Approximation Algorithm for Min Assignment

Let us consider a feasible assignment $r$ for the instance $(h, S = \{s_1, \dots, s_n\})$ of Min Assignment; by definition, it should be clear that $r$ is a feasible assignment also for $\overleftarrow{\mathsf{ALL}}_h(1,n)$ and $\overrightarrow{\mathsf{ALL}}_h(1,n)$. This fact implies the following useful lower bound on the optima of Min Assignment.

**Lemma 2** *For any set of stations $S = \{s_1, \dots, s_n\}$ and for any $h \geq 1$, it holds that*

$$\mathsf{OPT}_h(S) \geq \max\left\{\overleftarrow{\mathsf{ALL}}_h(1,n), \overrightarrow{\mathsf{ALL}}_h(1,n)\right\}.$$

By combining the above lemma with the algorithm of Lemma 1, we get the following result (the proof is given in the Appendix).

**Theorem 4** *There is an algorithm that, for any set $S = \{s_1, \dots, s_n\}$ and any $h \geq 1$, computes in time $O(hn^3)$ an $h$-assignment $r_{apx}$ for $S$ such that $\mathsf{cost}(r_{apx}) \leq 2 \cdot \mathsf{OPT}_h(S)$.*

### 4.2 Well Spread Instances

This section is devoted to the construction of a better approximation for *well spread instances* of the Min Assignment problem.

**Definition 4 (well spread instances)** *A family $\mathcal{S}$ of* well-spread *instances is a family of instances $S$ such that*

$$\delta_{\max}(S) = O(\delta_{\min}(S)\mathrm{poly}(\log n)).$$

The following lemma is proved in the Appendix (by the way, the first item is an easy consequence of Theorem 1).

**Lemma 3** *a). For any set $S$ of $n$ stations on the line and for any constant $h \geq 1$, it holds that*

$$\mathsf{OPT}_h(S) = \Omega\left(\delta_{\min}(S)^2 n^{\frac{2^{h+1}-1}{2^h-1}}\right).$$

*b). Let $\mathcal{S}$ be a family of well-spread instances. Then, for any instance $S^w \in \mathcal{S}$ of $n$ stations and for any $h \geq 1$,*

$$\mathsf{OPT}_h(S^w) = \Omega\left(D(S^w)^2 \frac{n^{\frac{1}{2^h-1}}}{\mathrm{poly}(\log n)}\right).$$

**Theorem 5** *There is an algorithm that, for any family $\mathcal{S}$ of well spread instances, for any instance $S^w \in \mathcal{S}$, and for any constant $h > 0$, computes in time $O(hn^3)$ an $h$-assignment $r^{apx}$ of cost $\mathsf{cost}(r_{apx}) = (1+\epsilon(n)) \cdot \mathsf{OPT}_h(S^w)$, where $\epsilon(n) = o(1)$.*

SKETCH OF PROOF. We will prove that if $S^w \in \mathcal{S}$ then the optimum $\mathsf{BASES}_h(S^w)$ of the MIN ASSIGNMENT WITH BASIS problem on instance $(h, S^w)$ is such that $\mathsf{BASES}_h(S^w) = (1 + o(1)) \cdot \mathsf{OPT}_h(S^w)$. Let $r^{apx}$ be the range assignment yielded by the algorithm of Theorem 3 in time $O(hn^3)$. Wlog assume that $h \geq 2$. If there is an optimal $h$-assignment for $S^w$ of type $\mathsf{B}^*$ (see Def. 3) then $r^{apx}$ is optimal for $S^w$. So, assume that all the optimal $h$-assignments for $S^w$ are not of type $\mathsf{B}^*$. Consider an optimal $h$-assignment $r^{opt}$. Let $\mathsf{L}$ be the rightmost station reaching station 1 in one hop. It is not hard to see that all the stations that do not use a base to reach both the endpoints must reach $\mathsf{L}$. Define the range assignment $r^B$ in which all the stations different from $\mathsf{L}$ keep the range as in $r^{opt}$ while $\mathsf{L}$ has now a range sufficient to be a basis.

In this assignment any station reaches some base in at most $h-1$ hops, so $r^B$ is an $h$-assignment of type $\mathsf{B}^*$. It turns out that

$$\mathsf{cost}(r^{apx}) \leq \mathsf{cost}(r^B) \leq \mathsf{OPT}_h(S^w) + D(S^w)^2.$$

Since $\mathcal{S}$ is well spread, from Lemma 3(b) we have that, for some constant $c > 0$,

$$\mathsf{OPT}_h(S^w) \;\geq\; cD(S^w)^2 \frac{n^{1/(2^h-1)}}{\mathrm{poly}(\log n)}.$$

By combining the last two inequalities, we get

$$\frac{\mathsf{cost}(r^{apx})}{\mathsf{OPT}_h(S^w)} \leq 1 + \frac{D(S^w)^2}{\mathsf{OPT}_h(S^w)} \leq 1 + c\frac{\mathrm{poly}(\log n)}{n^{1/(2^h-1)}}.$$

For constant $h$ and for $n \to \infty$, the above value tends to 1. Hence the theorem follows. □

## 5 MIN 2-HOPS ASSIGNMENT is in P

A suitable application of the algorithms given in Lemma 1 and Theorem 3 will allow us to obtain an algorithm that solves MIN 2-HOPS ASSIGNMENT in time $O(n^3)$. To this aim, we need to distinguish three types of possible optimal 2-assignments.

**Definition 5** *Let $r$ be a 2-assignment. We say that $r$ is of type*

- $\mathsf{B}^*$ *if all the stations reach one base in one hop (this is the same as in Def. 3);*
- $\mathsf{LR}$ *if there are no base stations and $\mathsf{L}$ ($\mathsf{R}$) denotes the rightmost (leftmost) station reaching 1 ($n$) in one hop, and $\mathsf{L}$ is on the left of $\mathsf{R}$;*
- $\mathsf{B^*LRB^*}$ *if there is at least a base station and there is no base between $\mathsf{L}$ and $\mathsf{R}$, where $\mathsf{L}$ ($\mathsf{R}$) denotes the rightmost (leftmost) non-base station reaching 1 ($n$) in one hop, and $\mathsf{L}$ is on the left of $\mathsf{R}$.*

The proof of the following lemma is straightforward.

**Lemma 4** *Let $S$ be a set of stations on the line. If $r_{\mathsf{opt}_2}$ is an optimal 2-assignment for $S$, then it is either of type $\mathsf{B}^*$, $\mathsf{LR}$ or $\mathsf{B^*LRB^*}$.*

**Theorem 6** *The* MIN 2-HOPS ASSIGNMENT *problem can be solved in time $O(n^3)$.*

Sketch of Proof. From Lemma 4 the type of an optimal solution for Min 2-hops Assignment has to be one of the three types listed in Def. 5. As for the first type (i.e. $\mathsf{B}^*$) we apply the algorithm in Theorem 3. As for the second type (i.e. $\mathsf{LR}$), for each possible pair $\mathsf{L}$ and $\mathsf{R}$, we split the instance $[1, n]$ in three independent intervals $[1, \mathsf{L}]$, $(\mathsf{L}, \mathsf{R})$, $[\mathsf{R}, n]$. In $[1, \mathsf{L})$ we compute $\overrightarrow{\mathsf{ALL}}_1(1, \mathsf{R})$ in which the stations between $\mathsf{L}$ and $\mathsf{R}$ are removed. Symmetrically we compute the assignment for $(\mathsf{R}, n]$. In $(\mathsf{L}, \mathsf{R})$ we compute $\mathsf{AND}_1(\mathsf{L}, \mathsf{R})$. The third type (i.e. $\mathsf{B}^*\mathsf{LRB}^*$) can be solved by a simple combination of the previous two types.

An optimal 2-assignment can be thus obtained by selecting the minimum cost assignment among the all the assignments computed above. □

# 6 Open Problems

In 1997, Kirousis *et al* [11] wondered whether efficient solutions for the bounded hops case of the Min Assignment problem can be obtained. This paper provides a first positive answer to this question since, in most practical applications, an easy-to-obtain 2-approximated solution can be reasonably considered good enough.

On the other hand, the theoretical question whether Min Assignment $\in$ P still remains open. However, we believe that exact solutions can be obtained only by algorithms whose time complexity is $\Omega(n^h)$.

Finally, it is interesting to observe that a similar "state of the art" holds for another important "Euclidean" problem, i.e., the *Geometric Facility Location Problem* (in short, GFLP) [10]. Indeed, also in this case the Euclidean properties of the support space have been used to derive very good approximation results [2] (much better than those achievable on non-geometric versions of FLP). On the other hand, it is still unknown whether or not GFLP $\in$ P.

# References


[1] E. Arikan. Some complexity results about packet radio networks. *IEEE Transactions on Information Theory*, IT-30:456–461, 1984.

[2] S. Arora, P. Raghavan, and S. Rao. Approximation schemes for the euclidean $k$-medians and related problem. In *Proc. 30th Annual ACM Symposium on Theory of Computing (STOC)*, pages 106–113, 1998.

[3] M.A. Bassiouni and C. Fang. Dynamic channel allocation for linear macrocellular topology. *Proc. of ACM Symp. on Applied Computing (SAC)*, pages 382–388, 1998.

[4] I. Chlamtac and A. Farago. Making transmission schedules immune to topology changes in multihop packet radio networks. *IEEE/ACM Trans. Networking*, 2:23–29, 1994.

[5] B.S. Chlebus, L. Gasienec, A.M. Gibbons, A. Pelc, and W. Ritter. Deterministic broadcasting in unknown radio networks. *11th Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 861–870, 2000.

[6] A. Clementi, P. Penna, and R. Silvestri. Hardness results for the power range assignment problem in packet radio networks. *Proc. of RANDOM-APPROX'99, Randomization, Approximation and Combinatorial Optimization*, LNCS(1671):197 – 208, 1999.

[7] A. Clementi, P. Penna, and R. Silvestri. The power range assignment problem in radio networks on the plane. *17th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, LNCS(1770):651 – 660, 2000.

[8] K. Diks, E. Kranakis, D. Krizanc, and A. Pelc. The impact of knowledge on broadcasting time in radio networks. *Proc. 7th Annual European Symposium on Algorithms (ESA)*, LNCS(1643):41–52, 1999.

[9] A. Ephemides and T. Truong. Scheduling broadcast in multihop radio networks. *IEEE Transactions on Communications*, 30:456–461, 1990.

[10] D. Hochbaum. Heuristics for the fixed cost median problem. *Math. Programming*, 22:148–162, 1982.

[11] L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc. Power consumption in packet radio networks. *14th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, LNCS(1200):363 – 374, 1997.

[12] E. Kranakis, D. Krizanc, and A. Pelc. Fault-tolerant broadcasting in radio networks. *6th Annual European Symposium on Algorithms (ESA)*, LNCS(1461):283–294, 1998.

[13] R. Mathar and J. Mattfeldt. Optimal transmission ranges for mobile communication in linear multihop packet radio netwoks. *Wireless Networks*, 2:329–342, 1996.

[14] K. Pahlavan and A. Levesque. *Wireless Information Networks.* Wiley-Interscince, New York, 1995.

[15] P. Piret. On the connectivity of radio networks. *IEEE Trans. on Infor. Theory*, 37:1490–1492, 1991.

[16] S. Ramanathan and E. Lloyd. Scheduling boradcasts in multi-hop radio networks. *IEEE/ACM Transactions on Networking*, 1:166–172, 1993.

[17] R. Ramaswami and K. Parhi. Distributed scheduling of broadcasts in radio network. *INFOCOM*, pages 497–504, 1989.

[18] S. Ulukus and R.D. Yates. Stochastic power control for cellular radio systems. *IEEE Trans. Commun.*, 46(6):784–798, 1998.

# A Proof of Theorem 4

Let $r_{sx}$ and $r_{dx}$ be two optimal range assignments for $\overleftarrow{\mathsf{ALL}}_h(1,n)$ and $\overrightarrow{\mathsf{ALL}}_h(1,n)$, respectively. Define

$$r_{apx}(s_i) = \max\{r_{sx}(s_i), r_{sx}(s_i)\}, \quad \text{for any } s_i \in S$$

It is easy to see that $r_{apx}$ is a feasible $h$-assignment for $S$ since $s_i$ is able to reach both $s_1$ and $s_n$ (and thus any other station in $S$) in at most $h$ hops. Moreover, $\mathsf{cost}(r_{apx}) \leq \overleftarrow{\mathsf{ALL}}_h(1,n) + \overrightarrow{\mathsf{ALL}}_h(1,n)$. Thus, Lemma 2 implies $\mathsf{cost}(r_{apx}) \leq 2 \cdot \mathsf{OPT}_h(S)$.

# B Proof of Lemma 3

a). Without loss of generality, assume that $\delta_{\min}(S) = 1$. Consider any feasible assignment $r$ for $S$. It suffices to observe that that this assignment is also feasible for the instance $N$ of $n$ stations at unit distance.

b). From Item a). we obtain

$$\mathsf{OPT}_h(S^w) = \Omega\left(\delta(S^w)^2 n^{\frac{2^{h+1}-1}{2^h-1}}\right). \tag{10}$$

Since $S^w$ is well-spread then it holds that

$$\delta(S^w) \geq \frac{\delta_{\max}(S^w)}{\mathrm{poly}(\log n)}.$$

By replacing the above bound into Eq. 10, we get

$$\mathsf{OPT}_h(S^w) = \Omega\left(\frac{\delta_{\max}(S^w)^2}{\mathrm{poly}(\log n)} n^{\frac{2^{h+1}-1}{2^h-1}}\right) = \Omega\left(\frac{D(S^w)^2 n^{\frac{1}{2^h-1}}}{\mathrm{poly}(\log n)}\right),$$

where the last equality is a consequence of $\delta_{\max}(S^w)n \geq D(S^w)$.