Power Assignment in Radio Networks with Two Power Levels*

Paz Carmi Matthew J. Katz

Department of Computer Science
Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel
{carmip,matya}@cs.bgu.ac.il

March 2, 2006

Abstract

We study the power assignment problem in radio networks, where each radio station can transmit in one of two possible power levels, corresponding to two ranges – short and long. We show that this problem is NP-hard, and present an $O(n^2)$-time assignment algorithm, such that the number of transmitters that are assigned long range by the algorithm is at most (11/6) times the number of transmitters that are assigned long range by an optimal algorithm. We also present an (9/5)-approximation algorithm for this problem whose running time is considerably higher.

1 Introduction

Assigning power levels (corresponding to transmission ranges) to the transmitters of a radio network, so that the total power consumption is as low as possible, is often an extremely important issue. Let $\mathcal{P}$ be a set of n points in the plane, representing n transmitters-receivers (or transmitters for short). We need to assign transmission ranges to the transmitters in $\mathcal{P}$, so that (i) the resulting communication graph is strongly connected; that is, the graph over $\mathcal{P}$ in which there exists a directed edge from p to q if and only if q lies within the transmission range r_p assigned to p, should contain a directed path from any transmitter $p \in \mathcal{P}$ to any other transmitter $q \in \mathcal{P}$, and (ii) the total power consumption (i.e., the cost of the assignment of ranges) is minimized, where the total power consumption is a function of the form $\sum_{p\in\mathcal{P}} r_p^c$, and $c > 0$ is a constant typically between 2 and 5.

This version of the power assignment problem is known to be NP-hard; Kirousis et al. [12] first proved this for 3-dimensional point sets and Clementi et al. [8] then proved this also for planar point sets. Kirousis et al. also present a 2-approximation algorithm, based on the minimum spanning tree of $\mathcal{P}$, which is the best approximation known.

In practice, it is usually impossible to assign arbitrary power levels (ranges) to the transmitters of a radio network. Instead one can only choose from a constant number of preset power levels,

*Research by P. Carmi is partially supported by a Kreitman Foundation doctoral fellowship. Research by M. Katz is partially supported by grant No. 2000160 from the U.S.-Israel Binational Science Foundation.

corresponding to a constant number of ranges. In this paper we consider the power assignment problem in radio networks, where each transmitter can transmit in one of two given power levels – low or high, corresponding to two possible ranges – short (r_1) and long (r_2). Since the cost of an assignment of power levels to the transmitters is a function of the form $n_1 r_1^c + (n - n_1) r_2^c$, where n_1 is the number of transmitters that are assigned range r_1 and $c \geq 1$ is some constant, the cost of an assignment is determined solely by the number of transmitters that are assigned range r_2. We are not aware of previous work on the power assignment problem under this model.

In Section 6 we prove that the power assignment problem with two power levels is NP-hard, by constructing a reduction from planar cubic vertex cover. More precisely, we show this for the special case where the initial components graph (see below) is a star. Recently we learned that the NP-hardness proof in [8] (that follows the NP-hardness proof in [12]) can be adapted to the case of two power levels. Our proof is similar to the second part of the proof in [8]; in the first part of this proof gadgets are constructed to force the optimal solution to use only three ranges, and it is easy to see that actually two ranges are enough.

Let m be the number of transmitters that are assigned range r_2 in an optimal assignment OPT. In Section 2 we describe an algorithm that assigns range r_2 to at most $(11/6)m$ transmitters, or, in other words, the algorithm computes an $(11/6)$-approximation (with respect to the number of transmitters that are assigned long range). The running time of this algorithm is $O(n^2)$ (see Section 4).

An immediate corollary of this result is that for *any* ranges r_1, r_2 and for *any* c, we can compute an assignment whose cost is at most $(11/6)$ times the cost of an optimal assignment. Usually though the cost of our assignment is much less than this, as is shown in Section 2.1. In this section we analyze the common case where $r_1 = 1$ and $r_2 = d$. Our algorithm computes in this case an assignment whose cost is at most $\frac{11d^c}{6d^c+5}$ times the cost of an optimal assignment. Plugging for example $d = 2$ we get a $44/29 \approx 1.52$ approximation, if $c = 2$, and a $22/17 \approx 1.29$ approximation, if $c = 1$.

A by-product of our range assignment algorithm is an algorithm for assigning ranges in the special case where the initial components graph is a tree. That is, consider the connected components of the communication graph that is obtained after assigning short range to all transmitters in $\mathcal{P}$. We draw an edge between two components C_1 and C_2 if and only if there exists transmitters $p_1 \in C_1$ and $p_2 \in C_2$, such that the distance between them is at most r_2. Now if this graph happens to be a tree then the algorithm described in Section 3 assigns long range to at most $(4/3)m$ transmitters, where m is the number of transmitters assigned long range by an optimal algorithm.

In Section 5 we describe another algorithm for the special case where the initial components graph is a tree. This algorithm assigns long range to at most $(6/5)m$ transmitters (rather than $(4/3)m$ transmitters), where m is the number of transmitters assigned long range by an optimal algorithm; however, its running time is extremely high. This algorithm allows us to achieve an approximation factor of $(9/5)$ instead of $(11/6)$ for our range assignment problem; however, this result is only interesting from a purely theoretical point of view because of the unrealistic running time.

More related work. Other variants of the power assignment problem have been studied. One such variant is the *symmetric* power assignment problem, where the corresponding communication graph is undirected and there exists an edge between two transmitters p and q if and only if both transmitters were assigned ranges greater than (or equal to) the distance between them; see [2, 3, 4]. Clementi et al. [6] consider the problem of assigning ranges to a set of transmitters on a common line, so that for any two transmitters p and q there exists a path from p to q of at most h *hops* in the corresponding (directed) communication graph. The case where $h = n - 1$ was also considered by [12]. An important related problem is the minimum-energy broadcast tree problem: Assign ranges to the transmitters so that a designated source transmitter can broadcast messages to all other transmitters; see, e.g., [5, 7, 13, 15, 17, 18, 19].

2 An (11/6)-Approximation

Let $\mathcal{P}$ be a set of n points in the plane representing n transmitters-receivers (or transmitters for short), and assume that each transmitter can transmit in one of two possible power levels – low or high, corresponding to short range (r_1) or long range (r_2). Further assume that if all transmitters in $\mathcal{P}$ are assigned long range, then the resulting communication graph is strongly connected. In this section we describe an algorithm for assigning ranges to the transmitters in $\mathcal{P}$, such that the number of transmitters that are assigned long range is at most $(11/6)m$, where m is the number of transmitters that are assigned long range by OPT. In Section 4 we show that the time complexity of the algorithm is $O(n^2)$.

Let G be the (undirected) graph of components. G is defined as follows. Assign to each transmitter in $\mathcal{P}$ short range and draw an edge between two transmitters p and q if $|p, q| \leq r_1$, where $|p, q|$ denotes the Euclidean distance between p and q. We think of each of the connected components in this graph as a subset of $\mathcal{P}$. These subsets are the nodes of the graph G; we shall call them components. We draw an edge between two components C_1 and C_2 of G if there exist transmitters $p \in C_1$ and $q \in C_2$, such that $|p, q| \leq r_2$. See Fig. 1.

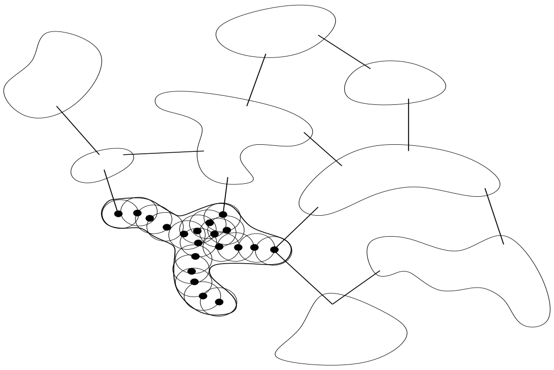

Figure 1: The components graph G.

Notice that we can easily obtain a 2-approximation. Simply compute a minimum spanning tree of G, and, for each edge (C_1, C_2) of the tree, assign long range to two transmitters $p \in C_1$ and $q \in C_2$, such that $|p, q| \leq r_2$.

Our range assignment algorithm consists of two stages. In the first stage we repeatedly find a cycle in G and reduce it to a single component by assigning long range to one transmitter in each of the components in the cycle. The second stage begins when there are no more cycles in G, i.e., when G is a tree. In this stage we assign long range to some more transmitters in order to complete our task.

We now describe the first stage in detail. While there is a cycle in G do the following. Let $C_1, C_2, \ldots, C_l, C_1$ be any cycle of size $l \geq 3$. Assign long range to any transmitter in C_1 that can reach a transmitter in C_2, assign long range to any transmitter in C_2 that can reach a transmitter in C_3, etc. All together we assign long range to l transmitters. Notice that after doing so any two transmitters in the union $C = C_1 \cup \cdots \cup C_l$ can talk with each other possibly through other transmitters in C. Thus these l components reduce to a single component C and the number of components decreases by $l - 1$. We update the graph G by replacing $C_1, \ldots, C_l$ with the single component C. After doing so we forget that some of the transmitters in C have already been assigned long range, and update the edges in G accordingly, see Fig. 2.

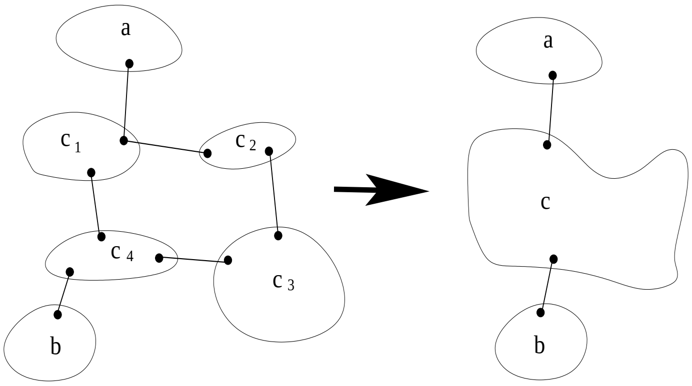

Figure 2: Reducing the cycle C_1, C_2, C_3, C_4, C_1 to the single component C.

At this point there are no cycles left in G, in other words G is a tree. In the next section we present a range assignment algorithm for the case where the components graph is a tree. This algorithm assigns long range to at most $\frac{4}{3}m_{tree}$ of the transmitters, where m_{tree} is the number of transmitters that are assigned long range by an optimal algorithm for this case. Thus in the second stage we apply the algorithm of the next section to G to complete the range assignment task. We now show that the overall number of transmitters that were assigned long range is bounded by $\frac{11}{6}m$.

Theorem 2.1 *The range assignment algorithm (described above) computes an* $(11/6)$*-approximation.*

Proof: Recall that in the first stage a loop is executed, such that, in each iteration a cycle in G of length at least three is found and replaced by a single component. Let i be the number of cycles that were found during the execution of the loop. We assume that all these cycles are of length exactly three, since this is the worst case for our analysis.

Let k be the initial number of components in G, i.e., right at the beginning of the first stage. Then m, the number of transmitters assigned long range by OPT, is at least k, since in each initial component at least one of the transmitters must be assigned long range. During the first stage the algorithm assigns long range to at most $3i$ transmitters, and the number of components in G at the end of the first stage is $k - 2i$.

At this point G is a tree and we distinguish between two cases.

Case 1: $i > k/2 - m/3$. In this case, instead of performing the second stage, we proceed in the most trivial way (for the purpose of the analysis only) and assign long range to $2(k - 2i - 1)$

transmitters. That is, for each edge in G connecting between two components C_1 and C_2, we assign long range to any transmitter in C_1 that can reach a transmitter in C_2 and vise versa. The total number of transmitters that were assigned long range is thus bounded by

$$3i + 2(k - 2i - 1) < 2k - i < 2k - (\frac{k}{2} - \frac{m}{3}) = \frac{3k}{2} + \frac{m}{3} \leq \frac{11}{6}m \ .$$

Case 2: $i \leq k/2 - m/3$**.** In this case we perform the second stage as described in Section 3 and assign long range to at most $(4/3)m_{tree}$ transmitters, where m_{tree} is the number of long range assignments needed to solve the tree G. But clearly $m_{tree} \leq m$, so the number of transmitters assigned long range in the second stage is at most $(4/3)m$. The total number of transmitter that were assigned long range is thus bounded by

$$3i + \frac{4m}{3} \leq 3(\frac{k}{2} - \frac{m}{3}) + \frac{4m}{3} = \frac{3k}{2} + \frac{m}{3} \leq \frac{11}{6}m \ .$$

Since in both cases we were able to bound the total number of long range assignments by $(11/6)m$, we conclude that our range assignment algorithm computes an (11/6)-approximation.

□

Recall that the cost of an assignment is $n_1 r_1^c + (n - n_1) r_2^c$, where n_1 is the number of transmitters that are assigned range r_1 and $c \geq 1$ is some constant typically between 2 and 5. An immediate corollary of Theorem 2.1 is that for *any* ranges r_1, r_2 and for *any* c, we can compute an assignment whose cost is at most (11/6) times the cost of an optimal assignment. Usually though the cost of our assignment is much less than this, as is shown below.

2.1 The Cost for Ranges 1 and d

Theorem 2.2 *If $r_1 = 1$ and $r_2 = d$, then one can compute a range assignment whose cost is at most $\frac{11d^c}{6d^c+5}$ times the cost of an optimal assignment. For $d = 2$ we get a (44/29)-approximation, if $c = 2$, and a (22/17)-approximation, if $c = 1$.*

Proof: The cost of an optimal algorithm is $d^c \cdot m + 1 \cdot (n - m) = n + (d^c - 1)m$, where m is the number of transmitters assigned long range. We apply both our algorithm and the naive algorithm which assigns range d to all the transmitters. Put $a = n/m$. We distinguish between two cases.
Case 1: $a \leq 11/6$**.** In this case we use the naive algorithm whose cost is $d^c n$. The ratio between the cost of the naive algorithm and the cost of an optimal algorithm is

$$\frac{d^c n}{n + (d^c - 1)m} \leq \frac{d^c n}{n + (d^c - 1)(6/11)n} = \frac{d^c}{(6/11)d^c + (5/11)} \ .$$

Case 2: $a \geq 11/6$**.** In this case we run our algorithm whose cost is at most

$$d^c \cdot (11/6)m + 1 \cdot (n - (11/6)m) = n + (11/6)(d^c - 1)m \ .$$

The ratio between the costs is

$$\frac{n + (11/6)(d^c - 1)m}{n + (d^c - 1)m} = \frac{a + (11/6)(d^c - 1)}{a + (d^c - 1)} \leq \frac{(11/6)d^c}{d^c + 5/6} = \frac{d^c}{(6/11)d^c + (5/11)} \ .$$

(By replacing a with the smallest value that it can receive, the expression to the left of the inequality is maximized.)

In both cases we got a $\frac{d^c}{(6/11)d^c+(5/11)}$-approximation on the cost. Thus for $d = 2$ we get a $(44/29)$-approximation, if $c = 2$, and a $(22/17)$-approximation, if $c = 1$. □

3 A (4/3)-Approximation for a Tree of Components

In this section we present a (4/3)-approximation algorithm for the case where the components graph G is cycle free, i.e., where G is a tree. In particular G may be the graph that is obtained at the end of the first stage of the general algorithm above.

We first pick an arbitrary component in G to be the root of G. Given a component C in G, we can now refer to its children components and to its parent component in the regular meaning.

For each component C we need to assign long range to some of the transmitters in C, so that for each child C' of C at least one of the transmitters in C assigned long range can reach (a transmitter in) C', and also at least one of these transmitters can reach the parent of C. A neighbor (i.e., one of the children or the parent) C' of C is *satisfied* if at least one of the transmitters in C that can reach it when assigned long range is assigned long range.

Initially all neighbors of C are *unsatisfied*. Our goal is to assign long range to a small number of transmitters in C so that all neighbors of C are satisfied. One can view this problem as a set cover problem: For each transmitter p in C let C_p be the subset of the neighbors of C that can be reached from p by assigning long range to p. It is easy to verify that the size of C_p is at most 5 (since no two components in C_p can be neighbors in G). Thus we could apply known results for k-set cover to achieve our goal; however, this would lead to a weaker result than the one that we obtain below.

We start with the leaf components. The case of a leaf component C is very simple; we assign long range to any transmitter in C that can reach the parent of C (when it is assigned long range). After considering all leaf components, we consider the internal components, where an internal component may be considered only if all its children have already been considered.

Let C be the internal component that is about to be considered. Let χ_C be the number of children of C. Clearly for each child C' of C, we must assign long range to at least one of the transmitters in C' that can reach C (after it is assigned long range). Let m'_C be the number of long range assignments (to transmitters in C) needed to satisfy all children of C. Then m_C, the number of long range assignments (to transmitters in C) assigned by OPT, is either m'_C, if the m'_C transmitters satisfying the children of C can be chosen so that one of them also satisfies the parent of C, or $m_C = m'_C + 1$, otherwise. The following inequalities are immediate: $\sum_C m_C = m$, where m is the overall number of long range assignments assigned by OPT, and $\sum_C \chi_C = k - 1 < m$, where k is the number of components in G. We will assign long range to at most $\frac{1}{3}\chi_C + m_C$ transmitters in C. Summing over all components in G we obtain

$$\sum_C (\frac{1}{3}\chi_C + m_C) \leq \frac{1}{3}m + m = \frac{4}{3}m \ .$$

For each transmitter p in C, let d_p (the *degree* of p) be the number of *unsatisfied children* of C that would be satisfied if p were assigned long range. Notice that d_p refers only to the children of

C and not to its parent. After assigning long range to a transmitter q in C we update the degrees d_p of all transmitters p in C (in particular d_q becomes 0).

We are now ready to describe our algorithm for assigning long range to transmitters in C. If $\chi_C \leq 2$, then we "solve" C optimally, that is, we find a minimum subset of transmitters in C that can reach all children of C and can also reach its parent (when assigned long range). We can do this since in this case $m_C \leq 3$.

Otherwise, as long as the number of unsatisfied children is at least 3 and there exists a transmitter of degree at least 3, we assign long range to any such transmitter q and update the degrees of all transmitters in C accordingly. By assigning long range to q we satisfy at least 3 of the children of C. Since for each of these 3 children, OPT assigns long range to one of their transmitters so that it can reach C, we charge the assignment to q to these 3 assignments of OPT. Thus in this loop we have used at most $\frac{1}{3}(\chi_C - x)$ long range assignments to transmitters in C, where $x \geq 0$ is the number of remaining unsatisfied children of C.

At this point either $x \leq 2$, or $x \geq 3$ and all transmitters in C have degree at most 2. In the former case we "solve" the remaining subproblem optimally (assigning long range to at most $3 \leq m_C$ transmitters in C). We have used in total at most $m_C + \frac{1}{3}\chi_C$ long range assignments.

In the latter case, where we are left with at least 3 unsatisfied children and transmitters of degree at most 2, we first assign long range to any transmitter in C that can reach C's parent (when assigned long range), and update the degrees of the transmitters in C. We charge this assignment to the at least 3 remaining unsatisfied children of C. Next we "solve" the remaining subproblem, which is an instance of 2-set cover, optimally. Again we have used in total at most $m_C + \frac{1}{3}\chi_C$ long range assignments.

Theorem 3.1 *If the components graph G is a tree, one can compute a range assignment that is a $(4/3)$-approximation.*

4 Running Time

In this section we show that the time complexity of our algorithm is $O(n^2)$. This bound holds for both stages of the algorithm described in Sections 2 and 3.

Computing the set of components of the initial graph of components G takes $O(n^2)$ time. Computing the edges between components also takes $O(n^2)$ time. For each such edge (C_1, C_2) we keep a pair of transmitters, one from each component, such that the distance between them is at most r_2.

In the first stage of the algorithm (Section 2), we repeatedly find a cycle in G and reduce it to a single component. A cycle in G (if exists) can be found in time proportional to the number of components in G, using a variant of *DFS* (see, e.g., [10]). The reduction of the cycle to a single component can be done in time proportional to the number of components in G plus the sum over the degrees of the components in the cycle. Thus the total time spent in this stage is $O(n^2)$.

Consider the second stage of the algorithm (Section 3). Let k be the number of components in the tree of components. We first analyze the running time for a single component C_i. The total time spent in this stage is then the sum of the running times for the k components.

Recall that χ_{C_i} is the number of children of C_i. We maintain two dynamic lists, the *high-degree* list and the *low-degree* list. If the degree of a transmitter $q \in C_i$ is greater than 2, we put it in the *high-degree* list, otherwise we put it in the *low-degree* list.

The following three cases are handled by the algorithm. If $\chi_{C_i} \leq 2$, then we "solve" C_i optimally in $O(|C_i|)$ time. If $\chi_{C_i} \geq 3$ and the high-degree list is not empty, then we assign long range to any transmitter q in the list and update the degrees of all transmitters in C_i accordingly. (The update phase can be done in $O(|C_i|)$ time, since the degree of q is bounded by some constant.) This case may repeat $O(\chi_{C_i})$ times, and thus the total running time due to this case is $O(\chi_{C_i}|C_i|)$. Finally, if $\chi_{C_i} \geq 3$ but the high-degree list is empty, then we solve an instance of 2-set cover optimally using matching techniques [14] in $O(\sqrt{\chi_{C_i}}|C_i|)$ time.

We conclude that the running time for C_i is $O(\chi_{C_i}|C_i|)$. Since we have k components, the total time spent in the second stage is

$$O(n^2) + O(\sum_{i=1}^{k}(\chi_{C_i}|C_i|)) \leq O(n^2) + O(\sum_{i=1}^{k}(\chi_{C_i})\sum_{i=1}^{k}(|C_i|)) \leq O(n^2) + O(kn) = O(n^2)$$

Theorem 4.1 *The range assignment algorithm (described in Sections 2 and 3) computes an (11/6)-approximation in $O(n^2)$ time.*

Remark: It is natural to consider the range assignment problem in the plane. However, our algorithm can be applied in any fixed dimension to compute an (11/6)-approximation in $O(n^2)$ time.

5 A (6/5)-Approximation for a Tree of Components

In this section we present an improved (6/5)-approximation algorithm for the case where the components graph G is a tree. This algorithm also allows us to achieve an approximation factor of (9/5) instead of (11/6) for our range assignment problem. The algorithm's top level is quite simple, but it includes calls to non-trivial (approximation) algorithms for 3-set cover and 4-set cover. Its analysis is rather difficult, and its time complexity is extremely high (though polynomial of course). Thus the results of this section are only interesting from a purely theoretical point of view.

We may assume that there exists a component in the components graph G that has more than 5 neighboring components. (Otherwise, we can compute an optimal solution by computing an optimal solution for each of the components). We pick any such component to be the root of the tree. We now apply the following three-stage algorithm to each of the components C in the tree. (At the root we only apply the first two stages of the algorithm.) Recall the χ_C denotes the number of children of component C, and that the degree of a transmitter in C is the number of (unsatisfied) children that it would reach if it were assigned long range.

In the first two stages we assign long range to a subset of the transmitters in C in order to satisfy all children of C. In the third stage, if still necessary, we assign long range to any transmitter in C that can reach C's father. Let π_C be the number of transmitters in C assigned long range by our algorithm, and let $\pi(\chi_C)$ be the number of transmitters in C assigned long range in the first two stages. Then either $\pi_C = \pi(\chi_C)$ or $\pi_C = \pi(\chi_C) + 1$.

We now describe the first two stages of our algorithm. In the first stage we perform the following loop. While there still remains a transmitter of degree at least 5: Pick an arbitrary transmitter of degree at least 5, assign to it long range, and update the degrees of all transmitters. Let l_C^5 be the number of transmitters that were assigned long range in the first stage, and let χ'_C be the number of the remaining unsatisfied children. Then $\chi'_C \leq \chi_C - 5l_C^5$.

In the second stage we apply two different procedures for satisfying the remaining χ'_C unsatisfied children. Our algorithm chooses the better of the two solutions. We denote by $\pi(\chi'_C)$ the number of transmitters that were assigned long range in this solution. Thus $\pi(\chi_C) = l^5_C + \pi(\chi'_C)$. The first procedure is simply the 4-set cover algorithm. Let $s_4(\chi'_C)$ be the size of the solution obtained. Then $\pi(\chi'_C) \leq s_4(\chi'_C) \leq 19/12m(\chi'_C)$, where $m(\chi'_C)$ is the size of an optimal solution for the remaining χ'_C children and 19/12 is the approximation ratio of the 4-set cover algorithm [9].

The second procedure repeatedly assigns long range to a transmitter of degree exactly 4 until the degree of all transmitters is at most 3. It then applies the 3-set cover algorithm. Let l^4_C be the number of transmitters of degree 4 that were assigned long range, and let χ''_C be the number of remaining unsatisfied transmitters. Then $\chi''_C = \chi'_C - 4l^4_C$. Also, let $s_3(\chi''_C)$ denote the size of the solution of the 3-set cover algorithm. Then $\pi(\chi'_C) \leq l^4_C + s_3(\chi''_C) \leq l^4_C + 4/3m(\chi''_C)$, where $m(\chi''_C)$ is the optimal solution for the remaining χ'' children and 4/3 is the approximation ratio of the 3-set cover algorithm [9].

We now prove that our algorithm computes a (6/5)-approximation; i.e., that

$$\frac{\sum_C \pi_C}{\sum_C m_C} \leq \frac{6}{5} \ .$$

The structure of the proof is as follows. In subsection 5.1 we show that it suffices to prove that for each component C with $\chi_C > 5$ it holds that

$$\frac{\pi(\chi_C) + \chi_C}{m_C + \chi_C - 1} \leq \frac{6}{5} \ ,$$

and in subsection 5.2 we prove that indeed for each component C with $\chi_C > 5$ this inequality holds.

Some additional notation. Recall that k denotes the number of components in the tree. We also denote the number of leaves in the tree by k_0, and the number of components in the tree for which $1 \leq \chi_C \leq 5$ by $k_{1:5}$.

Since the result of this section is purely theoretical, we may assume that for each component C, $m_C \geq 100$. Otherwise we can solve this component optimally by brute force and $\pi_C = m_C$.

5.1 A Sufficient Condition

In this subsection we show that it suffices to prove that for each component C with $\chi_C > 5$ it holds that

$$\frac{\pi(\chi_C) + \chi_C}{m_C + \chi_C - 1} \leq \frac{6}{5} \ .$$

We begin with two observations.

Observation 1.

$$\sum_C m_C = 1 + \sum_{C:\chi_C \geq 1} (m_C + \chi_C - 1)$$

Proof:

$$\begin{aligned}
\sum_C m_C &= \sum_{C:\chi_C=0} m_C + \sum_{C:\chi_C\geq 1} m_C = k_0 + \sum_{C:\chi_C\geq 1} m_C = 1 + (k-1) - (k-k_0) + \sum_{C:\chi_C\geq 1} m_C \\
&= 1 + \sum_{C:\chi_C\geq 1} \chi_C - \sum_{C:\chi_C\geq 1} 1 + \sum_{C:\chi_C\geq 1} m_C = 1 + \sum_{C:\chi_C\geq 1} (m_C + \chi_C - 1)
\end{aligned}$$

$\square$

Observation 2.

$$\sum_C \pi_C \leq \sum_{C:1\leq\chi_C\leq 5} (m_C + \chi_C - 1) + \sum_{C:\chi_C>5} (\pi(\chi_C) + \chi_C)$$

Proof:

$$\begin{aligned}
\sum_C \pi_C &= k_0 + \sum_{C:\chi_C\geq 1} \pi_C \leq k_0 + \sum_{C:1\leq\chi_C\leq 5} \pi_C + \sum_{C:\chi_C>5} (\pi(\chi_C)+1) - 1^1 \\
&= k_0 + \sum_{C:1\leq\chi_C\leq 5} m_C + \sum_{C:\chi_C>5} (\pi(\chi_C)) + (k - k_0 - k_{1:5}) - 1 \\
&= \sum_{C:1\leq\chi_C\leq 5} (m_C) - k_{1:5} + \sum_{C:\chi_C>5} (\pi(\chi_C)) + k - 1 \\
&= \sum_{C:1\leq\chi_C\leq 5} (m_C - 1) + \sum_{C:\chi_C>5} (\pi(\chi_C)) + (k-1) \\
&= \sum_{C:1\leq\chi_C\leq 5} (m_C + \chi_C - 1) + \sum_{C:\chi_C>5} (\pi(\chi_C) + \chi_C)
\end{aligned}$$

$\square$

We now use the two observations above to obtain the sufficient condition.

$$\begin{aligned}
\frac{\sum_C \pi_C}{\sum_C m_C} &\leq \frac{\sum_{C:1\leq\chi_C\leq 5}(m_C+\chi_C-1) + \sum_{C:\chi_C>5}(\pi(\chi_C)+\chi_C)}{1+\sum_{C:\chi_C\geq 1}(m_C+\chi_C-1)} \\
&= \frac{\sum_{C:1\leq\chi_C\leq 5}(m_C+\chi_C-1) + \sum_{C:\chi_C>5}(\pi(\chi_C)+\chi_C)}{1+\sum_{C:1\leq\chi_C\leq 5}(m_C+\chi_C-1)+\sum_{C:\chi_C>5}(m_C+\chi_C-1)} \\
&\leq \frac{\sum_{C:\chi_C>5}(\pi(\chi_C)+\chi_C)}{\sum_{C:\chi_C>5}(m_C+\chi_C-1)}
\end{aligned}$$

(In the last step we used the fact that if $a \geq b > 0$ and $c \geq c' \geq 0$, then $\frac{a+c}{b+c} \leq \frac{a+c'}{b+c'}$.) Now, in order to prove that the latter expression is bounded by 6/5, it suffices to prove that for each component C with $\chi_C > 5$ it holds that

$$\frac{\pi(\chi_C)+\chi_C}{m_C+\chi_C-1} \leq \frac{6}{5}\ .$$

5.2 The Sufficient Condition Holds

In this subsection we prove that for each component C with $\chi_C > 5$ it holds that

$$\frac{\pi(\chi_C)+\chi_C}{m_C+\chi_C-1} \leq \frac{6}{5}\ .$$

[1]Recall that we are assuming that the root has more than 5 children; hence the minus 1.

We do this by following the stages of the algorithm. The first stage of the algorithm assigns long range to transmitters of degree greater or equal 5. At the end of this stage we have (see above) $\pi(\chi_C) = l_c^5 + \pi(\chi'_C)$ and $\chi_C \geq 5l_c^5 + \chi'_C$. Therefore

$$\frac{\pi(\chi_C) + \chi_C}{m_C + \chi_C - 1} \leq \frac{\pi(\chi'_C) + \chi_C + l_c^5}{m_C + \chi_C - 1} \leq \frac{\pi(\chi'_C) + \chi'_C + 6l_c^5}{m_C + \chi'_C + 5l_c^5 - 1} \ .$$

Since we want to show that the latter expression is bounded by 6/5, it suffices to show that

$$\frac{\pi(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} \leq \frac{6}{5} \ .$$

At this point we are left with transmitters of degree less than 5. The second stage of the algorithm applies two different procedures in order to satisfy the remaining χ'_C children (see algorithm description at the beginning of section), and $\pi(\chi'_C)$ is the number of transmitters assigned long range by the one that finds a better solution. We may assume that $\chi'_C \geq 100$. (Otherwise we solve the problem with χ'_C children optimally, so $\pi(\chi'_C) = m(\chi'_C) \leq m_C$ and $\frac{\pi(\chi'_C)+\chi'_C}{m_C+\chi'_C-1} \leq \frac{m_C+\chi'_C}{m_C+\chi'_C-1} \leq \frac{6}{5}$.)

Recall that the first procedure is simply the 4-set cover algorithm, and the second procedure assigns long range to a transmitter of degree 4 as long as there is such a transmitter, and then applies the 3-set cover algorithm.

We list a few simple claims by which we justify the subsequent calculations.

1. $\pi(\chi'_C) \leq \chi'_C$; Trivial.
2. $\pi(\chi'_C) \leq s_4(\chi'_C)$; See above.
3. $\pi(\chi'_C) \leq l_c^4 + s_3(\chi''_C)$; See above.
4. $\chi'_C = 4l_c^4 + \chi''_C$; See above.
5. $l_c^4 < m_C$; Recall that l_c^4 is the number of transmitters of degree exactly 4 found by the second procedure. There is no transmitter in C that, when assigned long range, can transmit to more than 4 of the χ'_C children, since if there were such a transmitter it would have been chosen in the first stage. Hence it follows that $l_c^4 \leq m(\chi'_C)$. But if $l_c^4 = m(\chi'_C)$, then $\pi(\chi'_C) = m(\chi'_C) \leq m_C$, and the desired bound follows immediately.
6. $s_3(\chi''_C) \leq \chi''_C$; Trivial.
7. For some $q \geq 0$, $s_3(\chi''_C) = \frac{4}{3}m(\chi''_C) - q$; The 3-set cover approximation algorithm has an approximation ratio of $\frac{4}{3}$.
8. $\chi''_C \geq \frac{4}{3}m(\chi''_C) - q$; Follows from claims 6 and 7.
9. $\pi(\chi'_C) \leq s_4(\chi'_C) \leq \frac{19}{12}m(\chi'_C)$; The 4-set cover approximation algorithm has an approximation ratio of $\frac{19}{12}$.
10. $\chi''_C \geq m(\chi''_C)$; Trivial.

We shall assume that $m(\chi''_C) \geq 50$; this is justified by the following lemma.

Lemma 5.1 *If $m(\chi''_C) < 50$ then it holds that*

$$\frac{\pi(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} \leq \frac{6}{5} \, .$$

Proof:

$$l_c^4 \leq^{(5)} m_C \leq 2m_C - 6 \leq 6m_C - 4m(\chi''_C) - 6 = 6m_C - 5m(\chi''_C) + m(\chi''_C) - 6 \leq^{(10)} 6m_C - 5m(\chi''_C) + \chi''_C - 6 \, .$$

We thus got

$$(!) \qquad l_c^4 \leq 6m_C - 5m(\chi''_C) + \chi''_C - 6$$

We use the second procedure, but instead of applying the 3-set cover approximation algorithm to χ''_C, we compute an optimal solution of size $m(\chi''_C)$. Hence

$$\frac{\pi(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} \leq \frac{l_c^4 + m(\chi''_C) + \chi'_C}{m_C + \chi'_C - 1} =^{(4)} \frac{m(\chi''_C) + \chi''_C + 5l_c^4}{m_C + \chi''_C + 4l_c^4 - 1} \leq^{(!)} \frac{6}{5}$$

□

We now prove that if $m(\chi''_C) \geq 50$ (and $\chi'_C \geq 100$) then it holds that

$$\frac{\pi(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} \leq \frac{6}{5} \, .$$

We do this by distinguishing between two cases.

Lemma 5.2 *If $l_c^4 \leq \frac{2}{3}m(\chi''_C) - 6$ or $l_c^4 \geq \frac{4}{5}s_3(\chi''_C)$, then by applying the second procedure we obtain the desired ratio.*

Proof: If $l_c^4 \leq \frac{2}{3}m(\chi''_C) - 6$, then

$$\begin{aligned}\frac{\pi(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} &\leq^{(3,4)} \frac{l_c^4 + s_3(\chi''_C) + 4l_c^4 + \chi''_C}{m_C + 4l_c^4 + \chi''_C - 1} =^{(7)} \frac{l_c^4 + \frac{4}{3}m(\chi''_C) - q + 4l_c^4 + \chi''_C}{m_C + 4l_c^4 + \chi''_C - 1} \\ &\leq^{(8)} \frac{\frac{8}{3}m(\chi''_C) + 5l_c^4 - 2q}{m_C + \frac{4}{3}m(\chi''_C) + 4l_c^4 - q - 1} \leq \frac{\frac{8}{3}m(\chi''_C) + 5l_c^4 - 2q}{\frac{7}{3}m(\chi''_C) + 4l_c^4 - q - 1} \leq \frac{6}{5} \, .\end{aligned}$$

If $l_c^4 \geq \frac{4}{5}s_3(\chi''_C)$, then

$$\begin{aligned}\frac{\pi(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} &\leq^{(3,4)} \frac{l_c^4 + s_3(\chi''_C) + 4l_c^4 + \chi''_C}{m_C + 4l_c^4 + \chi''_C - 1} \leq \frac{s_3(\chi''_C) + 5l_c^4 + \chi''_C}{m_C + 4l_c^4 + \chi''_C - 1} \leq^{(5)} \frac{s_3(\chi''_C) + 5l_c^4 + \chi''_C}{5l_c^4 + \chi''_C} \\ &\leq^{(6)} \frac{2s_3(\chi''_C) + 5l_c^4}{s_3(\chi''_C) + 5l_c^4} \leq \frac{2s_3(\chi''_C) + 4s_3(\chi''_C)}{s_3(\chi''_C) + 4s_3(\chi''_C)} \leq \frac{6}{5} \, .\end{aligned}$$

□

Clearly $\frac{2}{3}m(\chi''_C) - 6 < \frac{4}{5}s_3(\chi''_C)$. The following lemma shows that if l_c^4 is in between these two values, then we also obtain the desired ratio.

Lemma 5.3 *If*

$$(*) \qquad \frac{2}{3}m(\chi''_C) - 6 < l_c^4 < \frac{4}{5}s_3(\chi''_C)\ ,$$

then we obtain the desired ratio.

Proof: We distinguish between two cases. If $\frac{7}{3}m_C \le \frac{11}{3}m(\chi''_C) - 30$, then by applying the first procedure we obtain the desired ratio.

$$\begin{aligned}
\frac{\pi(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} &\le^{(9)} \frac{\frac{19}{12}m(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} \le^{(4)} \frac{\frac{19}{12}m_C + 4l_c^4 + \chi''_C}{m_C + 4l_c^4 + \chi''_C - 1} \le^{(*)} \frac{\frac{19}{12}m_C + \frac{8}{3}m(\chi''_C) - 24 + \chi''_C}{m_C + \frac{8}{3}m(\chi''_C) - 24 + \chi''_C - 1} \\
&\le^{(10)} \frac{\frac{19}{12}m_C + \frac{8}{3}m(\chi''_C) - 24 + m(\chi''_C)}{m_C + \frac{8}{3}m(\chi''_C) - 25 + m(\chi''_C)} \le \frac{\frac{19}{12}m_C + \frac{11}{3}m(\chi''_C) - 24}{m_C + \frac{11}{3}m(\chi''_C) - 25} \\
&\le \frac{\frac{19}{12}m_C + \frac{7}{3}m_C + 30 - 24}{m_C + \frac{7}{3}m_C + 30 - 25} \le \frac{\frac{12}{3}m_C + 6}{\frac{10}{3}m_C + 5} \le \frac{6}{5}\ .
\end{aligned}$$

If $\frac{7}{3}m_C \ge \frac{11}{3}m(\chi''_C) - 30$, then by applying the second procedure we obtain the desired ratio.

$$\begin{aligned}
\frac{\pi(\chi'_C) + \chi'_C}{m_C + \chi'_C - 1} &\le^{(3,4)} \frac{l_c^4 + s_3(\chi''_C) + 4l_c^4 + \chi''_C}{m_C + 4l_c^4 + \chi''_C - 1} \le^{(7)} \frac{l_c^4 + \frac{4}{3}m(\chi''_C) - q + 4l_c^4 + \chi''_C}{m_C + 4l_c^4 + \chi''_C - 1} \\
&\le^{(8)} \frac{\frac{8}{3}m(\chi''_C) + 5l_c^4 - 2q}{m_C + \frac{4}{3}m(\chi''_C) + 4l_c^4 - q - 1} \le \frac{\frac{8}{3}m(\chi''_C) + 5l_c^4 - 2q}{\frac{11}{7}m(\chi''_C) - \frac{90}{7} + \frac{4}{3}m(\chi''_C) + 4l_c^4 - q - 1} \\
&\le \frac{\frac{8}{3}m(\chi''_C) + 5l_c^4 - 2q}{\frac{61}{21}m(\chi''_C) - \frac{90}{7} + 4l_c^4 - q - 1} \le \frac{6}{5}\ .
\end{aligned}$$

The last inequality follows from the assumptions that $l_c^4 < \frac{4}{5}s_3(\chi''_C)$ and that $m(\chi''_C)$ is large enough.
□

We summarize the result of this section concerning a tree of components.

Theorem 5.4 *If the components graph G is a tree, one can compute a range assignment that is a (6/5)-approximation in polynomial time.*

This result allows us to achieve an approximation factor of (9/5) instead of (11/6) for our range assignment problem. The algorithm is identical to the algorithm described in Section 2, except that, instead of computing a (4/3)-approximation for a tree of components, we compute a (6/5)-approximation.

Theorem 5.5 *One can compute in polynomial time a (9/5)-approximation for the range assignment problem with two power levels.*

Proof: The proof is identical to the proof of Theorem 2.1, except that Case 1 is performed if $i > k/2 - 3m/10 - 1/2$ and Case 2 is performed if $i \le k/2 - 3m/10 - 1/2$. The corresponding computations change accordingly. In Case 1 the total number of transmitters assigned long range is bounded by

$$3i + 2(k - 2i - 1) = 2k - i - 2 < 2k - (\frac{k}{2} - \frac{3m}{10} - \frac{1}{2}) - 2 < \frac{3k}{2} + \frac{3m}{10} \le \frac{9}{5}m\ ,$$

and in Case 2 the total number of transmitters assigned long range is bounded by

$$3i + \frac{6m}{5} \leq 3(\frac{k}{2} - \frac{3m}{10} - \frac{1}{2}) + \frac{6m}{5} < \frac{3k}{2} + \frac{3m}{10} \leq \frac{9}{5}m \ .$$

□

6 NP-Hardness

Let r_1 and r_2 be any two ranges, such that $r_1 < r_2$ (and r_2 is polynomial in r_1). In this section we show that the problem of finding an optimal range assignment for a given set $\mathcal{P}$ of points in the plane (representing transmitters-receivers) is NP-hard. One can think of the problem as follows: Assign short range (r_1) to each of the transmitters in $\mathcal{P}$. The goal now is to find a smallest subset $\mathcal{P}' \subseteq \mathcal{P}$ of transmitters, such that, after assigning long range (r_2) to each of the transmitters in $\mathcal{P}'$, one obtains a strongly connected graph.

Consider the components graph G that is obtained when each transmitter in $\mathcal{P}$ is assigned short range (see Section 2 for a precise definition of G). We show that even the special case where G is a star, i.e., G consists of one *central* component C that is connected to k *orbit* components (see Fig. 3) is NP-hard. In this case, the problem is to find a smallest subset of transmitters in C that satisfies all orbit components (when each of the transmitters in the subset is assigned long range).

We describe a reduction from minimum vertex cover in planar cubic graphs. Let $PCG = (V, E)$ be a planar cubic graph (i.e., each of the nodes in PCG has degree at most 3). A *vertex cover* for PCG is a subset U of V, such that, for each edge $(v_1, v_2) \in E$, either $v_1 \in U$ or $v_2 \in U$. The problem of finding a vertex cover of minimum size in planar cubic graphs is known to be NP-hard [1, 11].

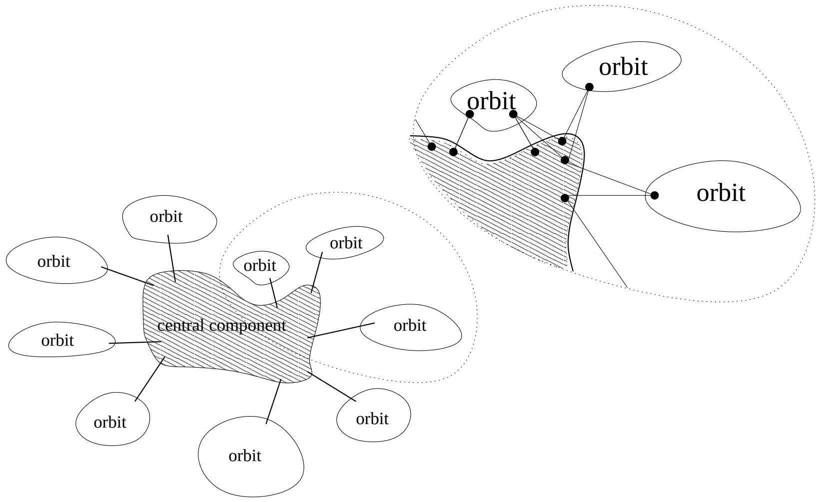

Figure 3: A star graph of components.

Valiant [16] showed that any planar cubic graph $PCG = (V, E)$ can be embedded in a rectangular grid of size $O(|V|^2)$ as follows. Each node $v \in V$ corresponds to some grid vertex, and each

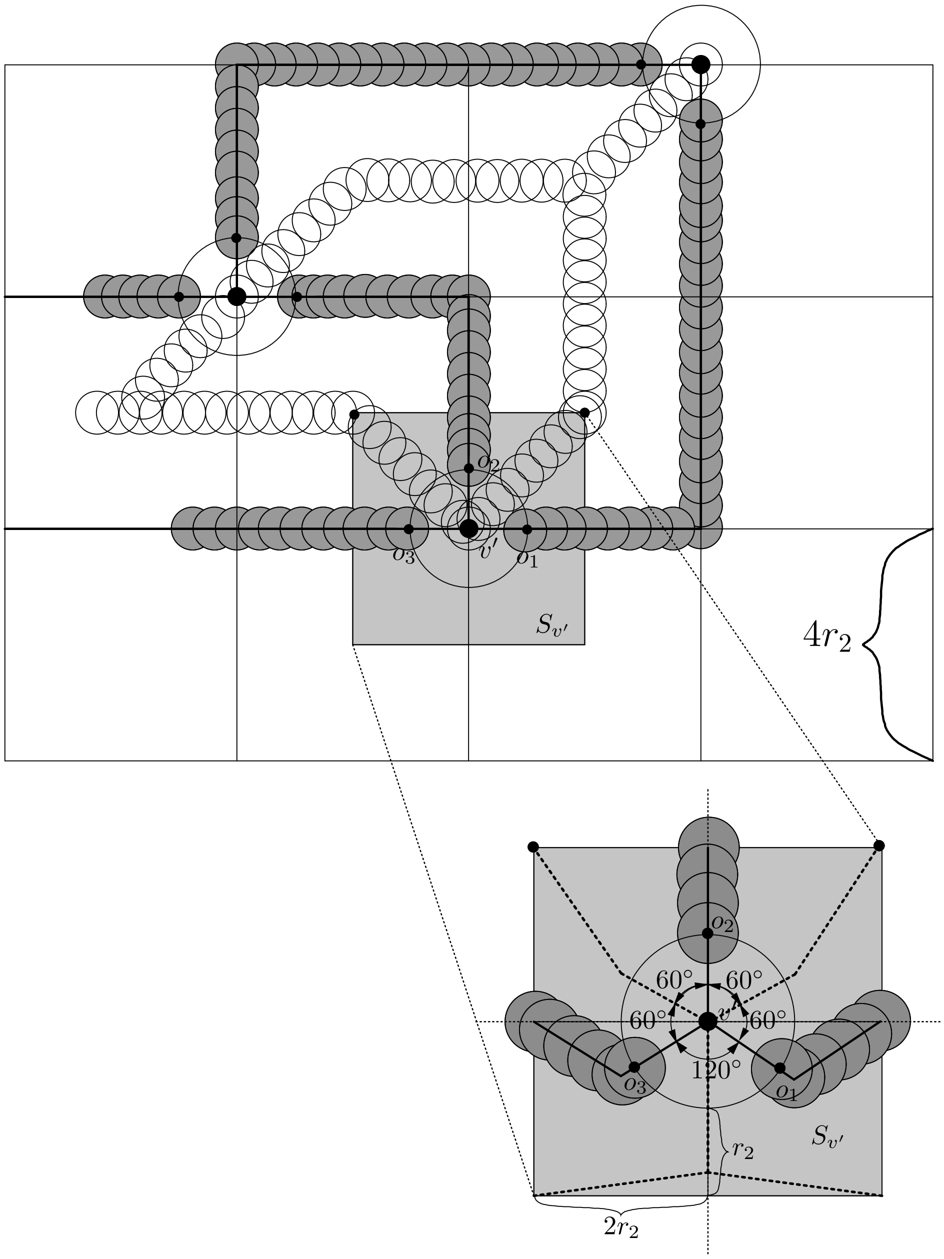

Figure 4: Top: Converting the embedded graph PCG' to a star graph of components. Bottom: Zooming in on the vicinity $S_{v'}$ of a vertex v'. One can connect v' to any of the four centers c of its adjacent grid cells by a sequence of transmitters starting at v' and ending at c, such that the transmitters lie on the dashed path between v' and c, the distance between any two consecutive transmitters is at most r_1, and none of the transmitters is within range r_1 of a point on an edge e' adjacent to v'.

edge $(v_1, v_2) \in E$ corresponds to a rectilinear path formed of grid edges, whose endpoints are the grid vertices corresponding to v_1 and v_2. Moreover the interiors of any two such paths are disjoint.

We now convert the embedded graph $PCG' = (V', E')$ into a star components graph G, see Fig. 4. We assume that the distance between adjacent grid vertices is $4r_2$. Each edge $e' \in E'$ is converted into an orbit component of G, and the set V' is converted into the central component of G. In general, we convert $e' = (v'_1, v'_2) \in E'$ into an orbit component by placing transmitters on the path e' as follows. Place transmitters along the path e' beginning at the point on e' at distance r_2 from v'_1 and ending at the point on e' at distance r_2 from v'_2, such that the distance between any two consecutive transmitters is at most r_1.

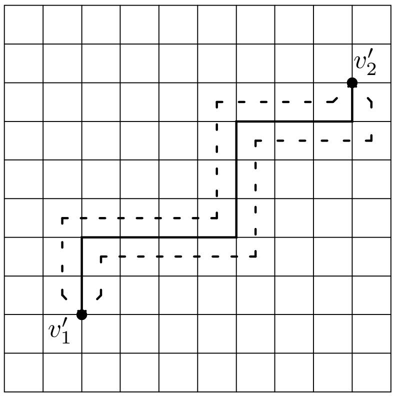

Figure 5: Connecting between v'_1 and v'_2.

In general, we convert the set V' into the central component by placing transmitters as follows. For each edge $e' = (v'_1, v'_2) \in E'$ we place transmitters at v'_1 and at v'_2 and along one of the two dashed paths between them (see Fig. 5), so that the distance between any two consecutive transmitters is at most r_1. However, we must be careful in the vicinity of a vertex $v' \in V'$. In order to avoid a situation where a transmitter on the portion of a dashed path connecting v' to the center of an adjacent grid cell is within distance r_1 of a transmitter belonging to an orbit component obtained from an edge of E' adjacent v', we slightly modify some of the paths in the vicinity of v' as depicted in Figure 4 (bottom). It is easy to see now that the transmitters belonging to the orbits obtained from the edges of E' adjacent to v' cannot cover a continuous portion of length greater or equal to r_1 of a dashed path connecting v' to the center of an adjacent cell. (Notice that we may assume that PCG is connected, since otherwise we could find a minimum vertex cover for each of its connected components and their union would be a minimum vertex cover for PCG.)

It is easy to verify that we obtained a star components graph G. That is (i) a transmitter in an orbit component C' that is assigned long range can either not reach any other component, or can only reach the central component (as is the case for the extreme transmitters in C'), and (ii) for each orbit component C' obtained from the edge $e' = (v'_1, v'_2)$ there exists a transmitter in the central component that can reach C', when assigned long range. The transmitters at v'_1 and at v'_2 are such transmitters.

Moreover, for any transmitter p in the central component, there exists a vertex v' that dominates it, in the sense that if both p and v' are assigned long range, then any orbit component that can be reached from p can also be reached from v'. Therefore when solving the range assignment problem, we may restrict ourselves to vertices v' in the central component. Also, assuming that r_2 is polynomial in r_1, the total number of transmitters that were used in the construction is polynomial in n. Finally, an optimal solution for the range assignment problem corresponds to a minimum vertex cover for the graph PCG.

Theorem 6.1 *Let r_1 and r_2 be any two ranges, such that r_2 is polynomial in r_1. Then the problem of finding an optimal range assignment (where r_1 and r_2 are the two possible ranges) for a given set $\mathcal{P}$ of points in the plane is NP-hard.*

7 Conclusion

As for the non-restricted version of the power assignment problem (described in the introduction), it remains open whether in 2-dimensional space the power assignment problem with two power levels admits a PTAS. Clementi et al. [8] showed that in 3-dimensional space the non-restricted version is APX-hard, and their proof also holds in our case, i.e., for the power assignment problem with two power levels. The gap however is big, so it would be interesting to try to close or reduce this gap.

Another natural and interesting problem that arises is the power assignment problem with $k > 2$ power levels, corresponding to k ranges $r_1 < \cdots < r_k$, for some fixed integer k. Notice that there is a significant difference between the case $k = 2$ and the case $k > 2$, since in the former case the goal is to minimize the number of transmitters assigned long range, while in the latter case a solution that minimizes the number of transmitters assigned the longest edge is not necessarily optimal.

Acknowledgement

We would like to thank an anonymous referee for bringing to our attention that the NP-hardness proof in [8] can be adapted to our model, and for stimulating us to remove a requirement that we formerly had in our NP-hardness proof on the ratio between r_2 and r_1.

References

[1] P. Alimonti and V. Kann. Hardness of Approximating Problems on Cubic Graphs. *Proc. 3rd Italian Conf. Algorithms and Complexity*, LNCS 1203, 288–298, 1997.

[2] E. Althaus, G. Călinescu, I. I. Măndoiu, S. Prasad, N. Tchervenski, and A. Zelikovsky. Power Efficient Range Assignment in Ad-hoc Wireless Networks. *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, 1889–1894, 2003.

[3] D. M. Blough, M. Leoncini, G. Resta, and P. Santi. On the Symmetric Range Assignment Problem in Wireless Ad Hoc Networks. *Proc. 2nd IFIP Int. Conf. Theoretical Computer Science*, 71–82, 2002.

[4] G. Călinescu, I. I. Măndoiu, and A. Z. Zelikovsky. Symmetric Connectivity with Minimum Power Consumption in Radio Networks. *Proc. 2nd IFIP Int. Conf. Theoretical Computer Science*, 119–130, 2002.

[5] A. E. F. Clementi, P. Crescenzi, P. Penna, and R. Silvestri. On the Complexity of Computing Minimum Energy Consumption Broadcast Subgraphs. *Proc. 18th Annual Sympos. Theoretical Aspects of Computer Science (STACS)*, LNCS 2010, 121–131, 2001.

[6] A. E. F. Clementi, A. Ferreira, P. Penna, S. Perennes, and R. Silvestri. The Minimum Range Assignment Problem on Linear Radio Networks. *Algorithmica* 35(2):95–110, 2003.

[7] A. E. F. Clementi, G. Huiban, P. Penna, G. Rossi, and Y. C. Verhoeven. On the Approximation Ratio of the MST-Based Heuristic for the Energy-Efficient Broadcast Problem in Static Ad-Hoc Radio Networks. *Int. Parallel and Distributed Processing Sympos. (IPDPS)*, 222, 2003.

[8] A. E. F. Clementi, P. Penna, and R. Silvestri. On the Power Assignment Problem in Radio Networks. *Mobile Networks and Applications* 9(2):125–140, 2004.

[9] R-c Duh and M. Fürer. Approximation of k-set cover by semi-local optimization. *Proc. 29th Annual ACM Sympos. Theory Computing (STOC)*, 256–264, 1997.

[10] S. Even. *Graph Algorithms*. Computer Science Press, 1979.

[11] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.

[12] L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc. Power Consumption in Packet Radio Networks. *Theoretical Computer Science* 243(1-2):289–305, 2000.

[13] W. Liang. Constructing Minimum-Energy Broadcast Trees in Wireless Ad Hoc Networks. *Proc. 3rd ACM International Sympos. on Mobile Ad Hoc Networking and Computing*, 112–122, 2002.

[14] S. Micali and V. V. Vazirani. An $O(\sqrt{V}E)$ algorithm for finding maximum matching in general graphs. *Proc. 21st Annual IEEE Sympos. Foundations of Computer Science (FOCS)*, 17–27, 1980.

[15] I. Papadimitriou and L. Georgiadis. Energy-aware Broadcasting in Wireless Networks. *Proc. Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, 267–277, 2003.

[16] L. Valiant. Universality Considerations in VLSI Circuits. *IEEE Trans. computers* 30:135–140, 1981.

[17] P.-J. Wan, G. Călinescu, X.-Y. Li, and O. Frieder. Minimum-Energy Broadcast Routing in Static Ad Hoc Wireless Networks. *ACM Wireless Networks* 8:607–617, 2002.

[18] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. On the Construction of Energy-Efficient Broadcast and Multicast Trees in Wireless Networks. *IEEE INFOCOM*, 585–594, 2000.

[19] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. Algorithms for Energy-Efficient Multicasting in Static Ad Hoc Wireless Networks. *Mobile Networks and Applications* 6(3):251–263, 2001.