



# An algorithm for the $\langle r, s \rangle$ -domination number of a tree

EJ Cockayne\*

*Received: 7 March 2007; Revised: 6 April 2007; Accepted: 16 April 2007*

## Dedication to Emeritus Professor Gerhard Geldenhuys

*Dedicated to my friend and colleague Professor Gerhard Geldenhuys, a gentleman and scholar, on the occasion of his 70<sup>th</sup> birthday. This friendship has led to twenty years of fruitful research collaborations on practical aspects of graph theory involving scientists from our respective universities.*

## Abstract

Suppose that at most  $r$  units of some commodity may be positioned at any vertex of a graph  $G = (V, E)$  while at least  $s$  ( $\geq r$ ) units must be present in the vicinity (*i.e.* closed neighbourhood) of each vertex. Suppose that the function  $f : V \mapsto \{0, \dots, r\}$ , whose values are the numbers of units stationed at vertices, satisfies the above requirement. Then  $f$  is called an  $s$ -dominating  $r$ -function. We present an algorithm which finds the minimum number of units required in such a function and a function which attains this minimum, for any tree.

**Key words:** Domination, dominating function,  $s$ -dominating  $r$ -function.

## 1 Introduction

We are concerned with situations in which units of some commodity are to be placed at nodes of a network which may be modelled by a simple undirected  $n$ -vertex graph  $G = (V, E)$ . Units positioned at a vertex  $v$  may be used in the vicinity of  $v$ , *i.e.* in the *closed neighbourhood*  $N[v]$  of  $v$ , which consists of  $v$  and all vertices adjacent to  $v$ . For various reasons (*e.g.* economy, space constraints, environmental issues) at most  $r$  units may be located at each vertex  $v$ , while at least  $s$  units may be required in its vicinity. Specific practical examples of these situations are distributions of guards, parking places, retail items, cash registers or hospital beds among the nodes of some network. The minimum

---

\*Department of Mathematics and Statistics, University of Victoria, PO Box 3045, STN CSC, Victoria, BC V8W 3P4, Canada, email: [ernie.cockayne@gmail.com](mailto:ernie.cockayne@gmail.com)

number of units which must be distributed throughout the entire network so as to satisfy the above constraints, is obviously an important parameter.

Formally, positive integers  $r, s$  ( $r \leq s$ ) are given. An  $r$ -function of  $G$  is a function  $f : V \mapsto \{0, \dots, r\}$ . For  $v \in V$ , let  $f[v] = \sum_{u \in N[v]} f(u)$ . The  $r$ -function  $f$  is said to be  $s$ -dominating if, for each  $v \in V$ ,  $f[v] \geq s$ . The *weight*  $|f|$  of an  $r$ -function  $f$  is defined by  $|f| = \sum_{v \in V} f(v)$ .

The  $\langle r, s \rangle$ -domination number of  $G$ , denoted by  $\gamma \langle r, s; G \rangle$ , is the minimum weight of an  $s$ -dominating  $r$ -function of  $G$ . The study of  $\langle r, s \rangle$ -domination was initiated in [2]. In the special case  $r = s = 1$ ,  $\gamma \langle r, s; G \rangle$  is the well-studied domination number  $\gamma(G)$  [6]. A linear algorithm for  $\gamma(T)$  was given in [3]. Other recently studied special cases are  $r = s = k$  (called  $\{k\}$ -domination [1, 4]) and  $r = 1, s = k$  (termed  $k$ -tuple domination [1, 5, 7]).

We present an efficient algorithm here for computing  $\gamma \langle r, s; T \rangle$  and a function attaining the minimum, for any tree  $T$ .

## 2 Theoretical justification for the algorithm

We begin with a simple property of integers. Let  $0 < r \leq s \leq 2r$  and let  $P$  be the set of all ordered pairs  $(a, b)$  of non-negative integers satisfying

$$0 \leq a \leq r, \tag{1}$$

$$0 \leq b \leq s \tag{2}$$

$$\text{and } 0 \leq b - a \leq r. \tag{3}$$

**Lemma 1** *If  $(x_i, y_i) \in P$  for each  $i = 1, \dots, k$ ,  $M = \max_i(y_i - x_i)$ ,  $\Sigma = \sum_{i=1}^k x_i$ ,  $y = \max\{M, s - \Sigma\}$  and  $x = \max\{M, y - r\}$ , then  $(x, y) \in P$ .*

**Proof:** By hypothesis and the definitions of  $x, y, M$  and  $\Sigma$ ,

$$\left. \begin{array}{l} 0 \leq M \leq r \leq s \\ \text{and } s - \Sigma \leq s. \end{array} \right\} \tag{4}$$

Therefore

$$0 \leq y \leq s, \tag{5}$$

*i.e.*  $(x, y)$  satisfies (2). Also,

$$x \geq M \geq 0. \tag{6}$$

There are now two cases.

**Case 1**  $x = M$ . By (4) and (6),  $(x, y)$  satisfies (1). If  $y = M$ , then (3) is satisfied and so  $(x, y) \in P$ . If  $y = s - \Sigma$ , then  $y \geq M$  so that  $y - x \geq 0$ . Moreover,  $x = M$  implies that  $M \geq y - r$ , *i.e.*  $y - M = y - x \leq r$  and (3) is satisfied. Again  $(x, y) \in P$  and Case 1 is complete.

**Case 2**  $x = y - r$ . In this case, using (5),  $x = y - r \leq s - r \leq r$ . Hence, using (6),  $(x, y)$  satisfies (1). Also,  $y - x = r$  and so (3) holds. Therefore  $(x, y) \in P$  as required. ■

In order that the  $r$ -function  $f$  satisfies  $f[v] \geq s$  for a leaf  $v$  of a tree, it is necessary that  $s \leq 2r$ . Hence the preceding lemma is applicable to our situation.

The algorithm, in fact, solves a more general problem than the one stated in the introduction. Let  $\mathcal{L}(T)$  be the set of leaves of  $T$  and let  $h$  be any function from  $\mathcal{L}(T)$  to  $P$ . Suppose that for  $u \in \mathcal{L}(T)$ ,  $h(u) = (x_u, y_u)$ . Our procedure will calculate  $\mu(T, h)$ , the minimum weight of an  $r$ -function  $f$  satisfying

$$f(u) \geq x_u \quad (u \in \mathcal{L}(T)), \quad (7)$$

$$f[u] \geq y_u \quad (u \in \mathcal{L}(T)) \quad (8)$$

$$\text{and } f[v] \geq s \quad (v \in V - \mathcal{L}(T)). \quad (9)$$

It also finds an  $r$ -function of minimum weight satisfying (7), (8) and (9), which will be called a  $\mu(T, h)$ -function.

It is easily seen that any tree  $T$  (of order  $n \geq 3$ ) has a minimum weight  $s$ -dominating  $r$ -function in which each leaf has value  $s - r$ . Hence

$$\gamma \langle r, s; T \rangle = \mu(T, h^*),$$

where  $h^*(u) = (s - r, s)$  for each  $u \in \mathcal{L}(T)$ . Thus, running the algorithm using  $h = h^*$  solves our original problem.

The following concept is required for the procedure. Suppose that the tree  $T$  (of order at least 3) is rooted at  $w$  and that  $\mathcal{S}(T)$  is the set of support vertices (*i.e.* vertices adjacent to leaves)  $v$  such that the set  $C(v)$  of children of  $v$  satisfies  $C(v) \subseteq \mathcal{L}(T)$ . For  $v \in \mathcal{S}(T)$ , let  $T' = T - C(v)$ . Observe that

$$\mathcal{L}(T') = (\mathcal{L}(T) - C(v)) \cup \{v\}$$

and that

$$\mathcal{S}(T) = \emptyset \Rightarrow T \text{ is a star with centre } w.$$

We now present an informal overview of the algorithm, which has two principal parts. Full details, theorems proving correctness and a formal statement of the procedure will follow.

Step 1 sequentially assigns a *label* (*i.e.* an element of  $P$ ) to each vertex of  $T - w$ . Initially each  $u \in \mathcal{L}(T)$  is given the label  $h(u)$ . Then a label for some  $v \in \mathcal{S}(T)$  is calculated from the labels of  $C(v)$ , and  $C(v)$  is removed. The set of labels on  $\mathcal{L}(T')$  is the function  $h'$ . The calculation is performed in such a way that a simple equation relates  $\mu(T, h)$  and  $\mu(T', h')$ . We now replace  $(T, h)$  by  $(T', h')$  and repeat the process. Iterations cease when  $\mathcal{S}(T') = \emptyset$ , at which time all vertices of  $T - w$  are labelled.

For  $u \in V(T) - \{w\}$ , let  $f(u)$  be the first coordinate of the label of  $u$ . A  $\mu(T, h)$ -function is found from  $f$  by computing a value for  $f(w)$  and possibly changing some values of  $f$  on  $C(w)$ , as dictated by Step 2 of the algorithm.

We now return to the formal description. The following two theorems show that the algorithm (given in Section 3) for the calculation of  $\mu(T, h)$  and the construction of a  $\mu(T, h)$ -function, is correct.

For  $v \in \mathcal{S}(T)$ , let  $C(v) = \{u_1, \dots, u_k\}$ , let  $h(u_i) = (x_i, y_i)$  for  $i = 1, \dots, k$  and let  $x, y, M$  and  $\Sigma$  be defined as in Lemma 1 (which proves that  $(x, y) \in P$ ). Recall that  $v$  is a leaf of  $T' = T - C(v)$ . Define  $h' : \mathcal{L}(T') \mapsto P$  by

$$\begin{aligned} h'(v) &= (x, y) \\ h'(u) &= h(u), \quad u \in \mathcal{L}(T') - \{v\}. \end{aligned}$$

**Theorem 1**  $\mu(T, h) = \Sigma + \mu(T', h')$ .

**Proof:** Let  $f'$  be a  $\mu(T', h')$ -function and define  $f$  on  $V(T)$  by

$$\begin{aligned} f(u_i) &= x_i, & i &= 1, \dots, k \\ f(u) &= f'(u), & u &\in V(T) - C(v). \end{aligned}$$

We show that  $f$  satisfies (7), (8) and (9). Since  $(x_i, y_i) \in P$ ,  $f$  satisfies (7). Moreover, for all  $i = 1, \dots, k$ ,

$$\begin{aligned} f[u_i] &= f(u_i) + f(v) \\ &= f(u_i) + f'(v) \\ &\geq x_i + x \\ &\geq x_i + M \\ &\geq x_i + (y_i - x_i) = y_i, \end{aligned}$$

so that  $f$  satisfies (8). Finally,

$$\begin{aligned} f[v] &= f'[v] + \Sigma \\ &\geq y + \Sigma \\ &\geq (s - \Sigma) + \Sigma = s, \end{aligned}$$

so that  $f$  satisfies (9). Therefore

$$\mu(T, h) \leq |f| = |f'| + \Sigma = \mu(T', h') + \Sigma. \quad (10)$$

Now let  $f$  be a  $\mu(T, h)$ -function and let  $f' = f|V(T')$ . It is immediate that  $f'$  and  $T'$  satisfy (9). We show that they also satisfy (7) and (8).

**Case 1**  $x = M$ . Suppose that  $f'(v) (= f(v)) < x$  and choose  $j$  so that  $M = y_j - x_j$ . Then

$$\begin{aligned} f[u_j] &= x_j + f(v) \\ &< x_j + M \\ &= x_j + (y_j - x_j) \\ &= y_j. \end{aligned}$$

This contradiction shows that  $f'(v) \geq x$  in this case.

**Case 2**  $x = y - r$ . In this case  $y - r \geq M$ . Therefore  $y > M$  and hence  $y = s - \Sigma$ . Suppose that  $f'(v) = f(v) < x = y - r$ . Then  $f(v) < (s - \Sigma) - r$ . Now  $f[v] = \Sigma + f(v) + f(p(v))$  (where  $p(v)$  is the parent of  $v$ ), hence

$$f[v] < \Sigma + (s - \Sigma - r) + r = s.$$

This contradiction shows that  $f'(v) \geq x$  and so (7) is satisfied in both cases.

To establish (8), firstly assume that  $y = M$ . Then  $f'[v] \geq f'(v) \geq x \geq M = y$  by (6). Secondly, let  $y = s - \Sigma$  and suppose, to the contrary, that  $f'[v] < y$ . Then

$$\begin{aligned} f[v] &= f'[v] + \Sigma \\ &< y + \Sigma \\ &= (s - \Sigma) + \Sigma = s, \end{aligned}$$

a contradiction which shows that  $f'[v] \geq y$  in this case.

The above implies that  $f'$  and  $T'$  satisfy (7), (8) and (9). Hence

$$\mu(T', h') \leq |f'| = |f| - \Sigma = \mu(T, h) - \Sigma. \quad (11)$$

The result now follows from (10) and (11).  $\blacksquare$

Let  $S$  be a star with centre  $w$ , let  $\mathcal{L}(S) = \{u_1, \dots, u_k\}$  where  $k \geq 2$ , let  $h(u_i) = (x_i, y_i)$  for  $i = 1, \dots, k$ , and let  $M, \Sigma$  be defined as above.

**Theorem 2**  $\mu(S, h) = \max\{M + \Sigma, s\}$ .

**Proof:** If  $M + \Sigma \geq s$ , then it is easily seen that the function  $f$  defined by

$$\begin{aligned} f(u_i) &= x_i, \quad i = 1, \dots, k \\ f(w) &= M \end{aligned}$$

is a  $\mu(S, h)$ -function with  $|f| = M + \Sigma$ . If  $M + \Sigma < s$ , then a  $\mu(S, h)$ -function  $g$  is found as follows. Set  $g(w) = M$  and choose  $g(u_1), \dots, g(u_k)$  so that

$$\left. \begin{aligned} x_i \leq g(u_i) \leq r, \quad i = 1, \dots, k \\ \text{and } g[w] = \sum_i g(u_i) + M = s. \end{aligned} \right\} \quad (12)$$

It remains to establish that choices are available to satisfy (12). Let  $g(u_i) = x_i + \alpha_i$  for  $i = 1, \dots, k$ . Then (12) requires that

$$\alpha_i \leq r - x_i \quad \text{and} \quad \sum_i (x_i + \alpha_i) + M = s. \quad (13)$$

To satisfy (13) we require that

$$\sum_i \alpha_i = s - M - \Sigma \quad \text{and} \quad \sum_i \alpha_i \leq kr - \Sigma,$$

*i.e.*  $s \leq M + kr$ . But this last inequality is true, because  $k \geq 2$  and  $s \leq 2r$ .  $\blacksquare$

### 3 The algorithm

The following procedure calculates  $\mu = \mu(T, h)$  and constructs a  $\mu(T, h)$ -function  $x : V \mapsto \{0, \dots, r\}$ . Steps 1 and 2 are justified by Theorems 1 and 2 respectively.

**Algorithm TREEDOM**  $\langle r, s \rangle$

**Input:**  $r, s, T$  (order  $n \geq 3$ ), root  $w \notin \mathcal{L}(T)$ ,  $h$

**Step 0** (initialize)  $\mu \leftarrow 0$

(Label each  $u \in \mathcal{L}(T)$ )  $(x(u), y(u)) \leftarrow h(u)$

**Step 1** (successively label vertices of  $V - \mathcal{L}(T) - \{w\}$ )

While  $T$  is not a star

Do Pick support vertex  $v$  with  $C(v) = \{u_1, \dots, u_k\} \subseteq \mathcal{L}(T)$

$M \leftarrow \max_i (y(u_i) - x(u_i))$

$\Sigma \leftarrow \sum_i x(u_i)$

$y(v) \leftarrow \max\{M, s - \Sigma\}$

$x(v) \leftarrow \max\{M, y(v) - r\}$

$\mu \leftarrow \mu + \Sigma$

$T \leftarrow T - C(v)$

od

(At this stage  $w$  is the only unlabelled vertex,  $T$  is a star with centre  $w$ ,  $C(w) = \{u_1, \dots, u_k\}$ .)

**Step 2**

$M \leftarrow \max_i (y(u_i) - x(u_i))$

$\Sigma \leftarrow \sum_i x(u_i)$

$x(w) \leftarrow M$

If  $(M + \Sigma < s)$  then

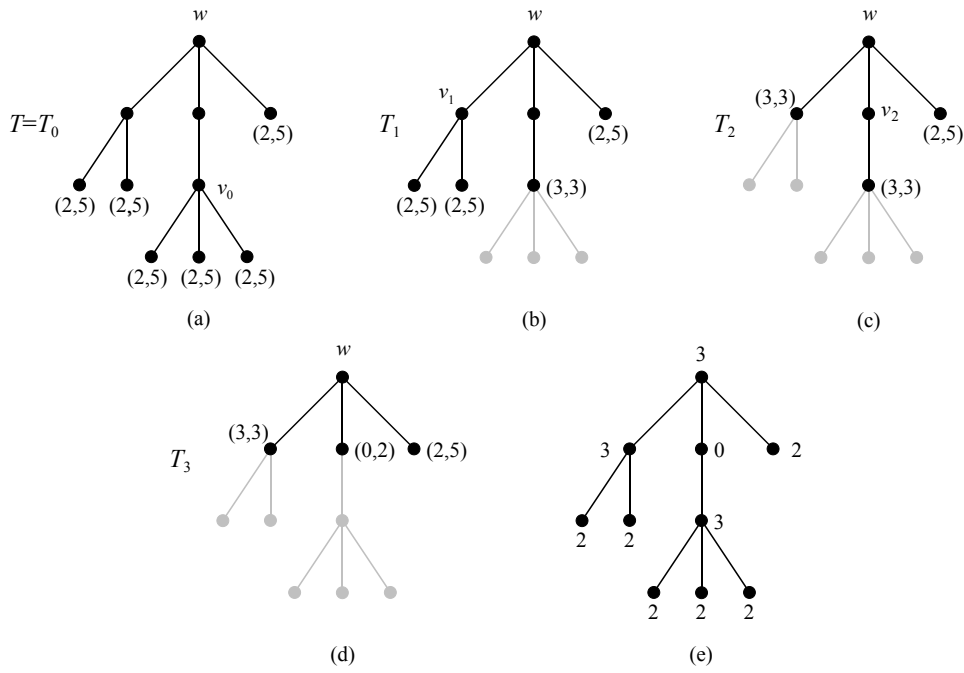
increase the  $x(u_i)$ s as necessary so that each  $x(u_i) \leq r$  and  $\Sigma + M = s$ . (14)

$\mu \leftarrow \mu + \Sigma + M$

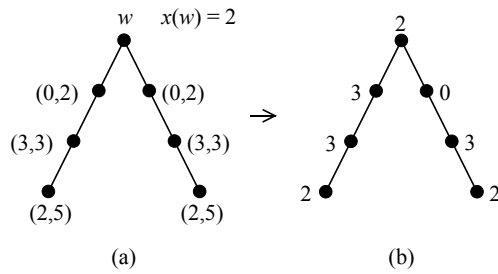
**Output:** function  $x$  and  $\mu$

**Stop.**

We have omitted the obvious details of (14). The algorithm may be implemented in  $O(n^2)$  time and it is likely that suitable data structures and possible refinements could provide a linear algorithm. We leave this question for future consideration.



**Figure 1:** Computation of  $\gamma\langle 3, 5; T \rangle$ .



**Figure 2:** Computation of  $\gamma\langle 3, 5; T \rangle$ .

## 4 Examples

We conclude with two examples in which the algorithm is used to find  $\gamma \langle 3, 5 \rangle$  for two small trees.

**Example 1** Let  $T = T_0$  be the tree (rooted at  $w$ ) depicted in Figure 1(a). Initially each leaf  $u$  is labelled with  $h^*(u) = (2, 5)$ . The edges of the subtrees  $T_1$ ,  $T_2$  and  $T_3$  produced by the successive iterations of Step 1 are depicted by means of black lines, with deleted vertices and edges in grey, in Figure 1(b), (c) and (d). For each  $i = 0, 1, 2$ , the vertex  $v_i$  will receive the next label and  $C(v_i)$  will be deleted. The computed label for  $v_i$  is shown in  $T_{i+1}$ . In Step 2  $x(w)$  is set to 3. This value together with the first coordinates of the labels of  $V(T) - \{w\}$  forms the 3-function with weight  $\gamma \langle 3, 5; T \rangle$  depicted in Figure 1(e).  $\square$

**Example 2** Let  $T$  be the path  $P_7$  rooted at the central vertex  $w$ . When the algorithm is used as in Example 1, the labels at the termination of Step 1 are depicted in Figure 2(a) and Step 2 sets  $x(w)$  to 2. In this case the first coordinates of the labels and  $x(w)$  do not 3-dominate  $w$  and Step 2 increases the first coordinates of  $C(w)$  accordingly. A solution is given in Figure 2(b).  $\square$

### Acknowledgement

The author thanks the referees for their speedy and thorough work and for the helpful suggestions for improving the presentation of the paper. He also acknowledges financial support by the Natural Sciences and Engineering Research Council of Canada.

### References

- [1] BREŠAR B, HENNING MA & KLAVŽAR S, 2006, *On integer domination in graphs and Vizing-like problems*, Taiwanese Journal of Mathematics, **10**, pp. 1317–1328.
- [2] COCKAYNE EJ, *Towards a theory of  $\langle \mathbf{r}, \mathbf{s} \rangle$ -domination in graphs*, submitted.
- [3] COCKAYNE EJ, GOODMAN S & HEDETNIEMI ST, 1975, *A linear algorithm for the domination number of a tree*, Information Processing Letters, **4**, pp. 41–44.
- [4] DOMKE GS, HEDETNIEMI ST, LASKAR T & FRICKE GH, 1991, *Relationships between integer and fractional parameters of graphs*, pp. 371–387 in KALAMAZOO MI (Ed), *Graph theory, combinatorics and applications*, **1**, Wiley-Interscience Publications, Wiley, New York (NY).
- [5] HARARY F & HAYNES TW, 2000, *Double domination in graphs*, Ars Combinatoria, **55**, pp. 201–213.
- [6] HAYNES TW, HEDETNIEMI ST & SLATER PJ, 1998, *Fundamentals of domination in graphs*, Marcel Dekker, New York (NY).
- [7] MEHRI J, MIRNIA M & SHEIKHOESLAMI SM, 2006, *3-Tuple domination number in complete grid graphs*, International Mathematical Forum, **(23)1**, pp. 1099–1112.