控制集合在樹及森林的線上演算法
On-line Algorithms for the Dominating Set Problem for Trees and a Note for General Graphs

Gow-Hsing King (金國興)* AND Wen-Guey Tzeng (曾文貴)
Dept. of Computer and Information Science
National Chiao Tung University, Hsinchu 30050, TAIWAN

Keywords: on-line algorithm; dominating set; forest

中文摘要:

本文先提出一個樹(trees)和森林(forests)控制集合(dominating set)的線上演算法(on-line algorithm)。針對這個問題文中將提出一個逼近率(performance ratio)爲 2 的線上演算法 *TWO*, 並證明 2-Θ(1/n) 爲逼近上限。

本文另外改進 [2] 的結果, 提出一個一般圖型(general graphs)逼進率爲 $\sqrt{2n}+c$ 的線上演算法 *NEW* 改進過去逼進率爲 $1.5\sqrt{n}+c$ 的結果。

Abstract

In this paper, an on-line algorithm TWO is proposed and the performace ratio is 2 of the on-line dominating set problem of trees and forests. In addition, a new on-line algorithm NEW is presented and gets a better performance ratio, $\sqrt{2n}$+c, than [2].

1 Introduction

Let $G=(V,E)$ with $V=\{1,2,\ldots,n\}$ be a graph. A dominating set of G is a subset $V' \subseteq V$ such that for each vertex $u \in V-V'$ there is a vertex $v \in V'$ so that $(u,v) \in E$. We study the on-line version of dominating set problem in this paper. Some on-line dominating set problems have already been studied in [2] and [3].

*To whom all correspondences should be sent.

On-line algorithms deal the events as they arrive without knowing future events. How many informations are shown to an on-line algorithm at any certain time that determines the on-line setting. Two on-line settings are considered in this paper. Without loss of generality, we assumed that the adjacency conditions of vertices are given in sequence $1,2,\ldots,n$. The first on-line setting is that at time i, the adjacency condition of vertex i to the other vertices j, $1 \leq j \leq n$, is presented. The second on-line setting is that at time i, the adjacency condition of vertex i to the vertices j, $1 \leq j < i$, is presented.

Two main results of the on-line dominating set problem are presented in this paper. First, we propose an on-line dominating set algorithm for forests of performance ratio 2 and show that $2-\Theta(1/n)$ is a lower bound for the performance ratio that an on-line dominating set algorithm can possibly achieve under an adaptive adversary. Second, we modify the on-line dominating set algorithm in [2] of general graphs to get performance ratio $\sqrt{2n}+c_1$ under the first on-line setting that improves the previous performance ratio $1.5\sqrt{n}+c_2$, where c_1 and c_2 are some positive constants.

2 The on-line algorithm TWO

Let $S=(V,E)$ with $V=\{1,2,\ldots,n\}$ be a forest; that is, S is a special case of a general graph without cycles. Let $i_r=\{v \mid (i,v) \in E\}$ denote

Table 1: The algorithm TWO.

At time 0: $D_0 \leftarrow \emptyset$;
At time i, $1 \le i \le n$, i is included in D_i if

1. there are ≥ 2 "undominated" vertices among i_r **(j-vertex)**.
2. each vertex in i_r is "visited" and not included in D_{i-1} **(f-vertex)**.
3. $\exists j \in i_r$ such that vertex j and each vertex in $j_r - \{i\}$ are "visited" and "undominated" **(f-vertex)**.

---

the neighbors of vertex i. On-line algorithm TWO for finding a dominating set problem of a forest is presented in Table 1. Let D_i be the set of vertices that are selected by TWO at time i, $0 \le i \le n$. It is natrual that $D_0 = \emptyset$, $D_i \subseteq D_{i+1}$, and $D = D_n$ is the dominating set returned by TWO. At time i, vertex v is called "visited" if $v \le i$, or else it is called "unvisited". After time i, vertex v is labeled as "dominated" if $v \in D_i$ or there is a vertex $u \in D_i$ such that $(u, v) \in E$ and other vertices are labeled as "undominated".

The algorithm TWO puts vertex i in D at time i if one of the three conditions that is shown in Table 1 is met. We call vertex i is a *j-vertex* if it meets the first condition and a *f-vertex* if it fails to meet the first condition but meets the last two ones. Therefore, let J (F) be the set of *j-vertices* (*f-vertices*) selected in D such that $D = J \cup F$ and $J \cap F = \emptyset$.

Performance ratio. We now show that the performance ratio of TWO is 2. Let C be a minimum dominating set for S. Let $F' = F - C$ and $J' = J - C$. The performance ratio of TWO is 2 if $|F'| \le |D - C|$ and $|J'| \le |C|$.

Lemma 2.1 $|F'| \le |D - C|$.

Proof. $|D - C| = |F' \cup J'| = |F'| + |J'| \ge |F'|$. □

Since S is a forest, S is composed of a set of disjoint rooted trees $T_1, T_2, \ldots, T_p$ such that $S = T_1 \cup T_2 \cup \ldots \cup T_p$. Root vertices of every disjoint trees T_i, $1 \le i \le p$, are arbitrarily assigned. Therefore, the parent/child relationship of each vertex of S (or each rooted tree T_i) is defined.

A j'-tree is a subtree of some T_i, $1 \le i \le p$, that is composed of vertices $v \in J'$. Each j'-tree is rooted at vertex r' such that the distance between r' and r in $T - i$ is minimum, where r is the root of T_i.

Lemma 2.2 $|J'| \le |C|$.

Proof. Consider a j'-tree L that is rooted at vertex r'. We can see that for each j'-path beginning with vertex r', say r'-j'_1-j'_2-$\cdots$-j'_t, then each $j'_i \in J'$ has at least one vertex c_i such that parent$(c_i) = j'_i$ and $c_i \in C$. Therefore, each c_i is charged by j'_i, where $1 \le i \le t$. That is, only r' in the j'-tree is left to be discussed.

By the algorithm TWO, the last vertex $v \in J' \cap L$ that is selected in D at time v must be adjacent to at least two "undominated" vertices u_i in $V - J'$. Since each u_i is a child of $v \in J'$, either $u_i \in C$ or there exists $w_i \in C$ such that parent$(w_i) = u_i$. Let $C' = \{u_i | u_i \in C\} \cup \{w_i | u_i \notin C,\ \text{parent}(w_i) = u_i, w_i \in C\}$. Since $|C'| \ge 2$, vertex v is able to be charged to one vertex of C' and r' is able to be charged to the other vertex of C'.

Since j'-trees T_i are mutually disjoint and the parent/child relationship of vertices in S (or each T_i) are fixed, each $v \in C$ is charged by at most one vertex $u \in J'$, where $1 \le i \le p$.

If a j'-tree L contains only a single vertex r', it can be proved similary as above case. □

Lemma 2.3 *The performance ratio of* TWO *is at most* 2.

Proof. Since $F \cap J = \emptyset$, $F \cup J = D$, and by Lemma 2.1 and Lemma 2.2,

$$\begin{aligned} |D| &= |F \cup J| \\ &= |F' \cup J' \cup (C \cap D)| \\ &= |F'| + |J'| + |C \cap D| \\ &= |D - C| + |C| + |C \cap D| \end{aligned}$$

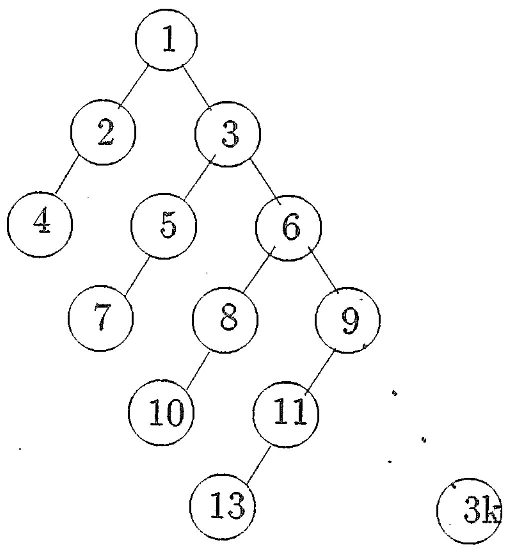

Figure 1: The structure of a forest against on-line dominating set algorithms

$$= \quad 2 \cdot |C|.$$

□

Lower bound. We now use an off-line adaptive adversary to prove that $2-\Theta(n)$ is a deterministic lower bound for the performance ratio of the on-line dominating set problem for forests in the first on-line setting.

Theorem 2.4 $2-\Theta(n)$ *is a lower bound for the performance ratio of the on-line dominating set problem for forests in this setting.*

Proof. Let Q be an on-line algorithm and R be an adaptive off-line adversary. The algorithm of R is as follows. At time 1, the adjacency condition of vertex 1 is $(1,2)$ and $(1,3)$. If vertex 1 is not selected in D by Q, vertices 2 and 3 are only adjacent to vertex 1 such that Q is forced to include vertices 2 and 3 in D. In this case, since vertices 2 and 3 can be replaced by vertex $1 \in C$ such that vertex 1 is charged by two vertices. If vertex 1 is selected in D, edge $(2,4)$ is given at time 2. Q must include either vertex 2 or 4 into D to dominate vertex 4 at time 2 or 4. In this case, since vertices 1 and 2 or vertices 1 and 4 can be replaced by vertex $2 \in C$ such that vertex 2 is charged by two vertices.

In the first case above, edges $(4,5)$ and $(4,6)$ are given at time 4 and vertices 4, 5 and 6 play the similar roles as vertices 1, 2 and 3 discussed above respectively. In the second case above, edges $(3,5)$ and $(3,6)$ are given at time 3 such that vertices 3, 5 and 6 play the similar roles as vertices 1, 2 and 3 discussed above respectively. R constructs the adjacency condition of S in the same way, and the structure looks like Figure 1. The adjacency conditions of remaining vertices are given in the similar structure.

In general, once a vertex $3m$ (or 1) is not included in D at time $3m$ (or 1), R lets the latest connected vertices be formed as a disjoint tree T of S, where m is a positive integer. It is trivial to verify that the performance ratio is 2 of T. Therefore, it causes the performance ratio closer to 2.

If vertices $3m$ (and 1), $1 \leq m \leq \lfloor \frac{n}{3} \rfloor$, are always included in D of the above adjacency conditions given by R, the performance ratio is about $\frac{2n/3-1}{n/3} = 2 - \frac{3}{n}$. □

Theorem 2.5 $1.5-\epsilon(n)$ *is a lower bound for the performance ratio of the on-line dominating set problem for a connected tree in this setting.*

Proof. The algorithm of R is given as follows. At time 1, edges $(1,2)$, $(1,3)$, and $(1,4)$ are given. If vertex 1 is not selected in D then let vertices 2 and 3 be only adjacent to vertex 1. Viewing locally, the performance ratio is 2 since vertices 2 and 3 can be replaced by vertex 1. If vertex 1 is selected in D then let edges $(2,5)$, $(3,6)$, $(4,7)$, $(4,8)$ and $(4,9)$ are given from time 2 to time 6. One of vertex 2 or 5 and one of vertex 3 or 6 are forced to be included in D to dominate vertices 5 and 6. Viewing locally, the performance ratio is 1.5 since vertex 1 is not necessary to be included in D and charged to vertices 2 and 3 that are in C. Obviously, Q should be in favor of including vertex 1 in D against R.

The remaining proof is the same as the above theorem. □

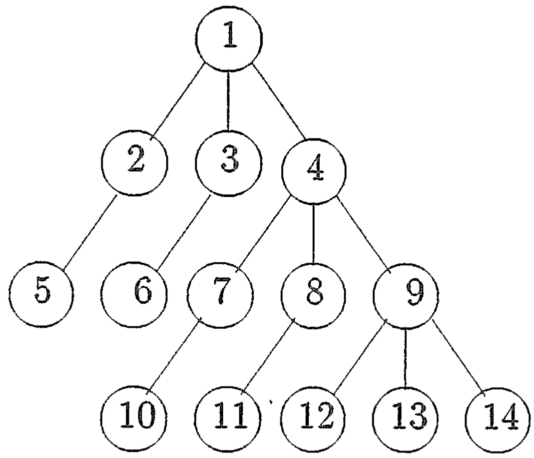

Figure 2: The structure of a connected tree against on-line dominating set algorithms

The second on-line setting. We show that $n-1$ is the tight bound under the second on-line setting with respect to an adaptive off-line adversary. We present an on-line algorithm for this setting for completeness although it is straightforward. At time i, the algorithm puts vertex i into D if vertex i is not dominated by other vertices $j \in D$, $1 \leq j < i$. We now show that the lower bound is $n-1$.

Theorem 2.6 *$n-1$ is a deterministic lower bound for the performance ratio of the on-line dominating set problem in the second on-line setting.*

Proof. An adaptive adversary is given as follows. Let A be an on-line algorithm. At time i, if vertex i is not selected in D by A then let vertices j, $i < j \leq n$, be adjacent to i at time $i+1$ to n. If vertices 1 to $n-1$ are all included in D, let vertex n is adjacent to all other vertices at time n. It is trivial to verify that $n-1$ is a deterministic lower bound for the performance ratio. □

3 A note for general graphs

Algorithm NEW uses the threshold value $\lceil\sqrt{n/2}\rceil$ to select j-vertices in D, while the algorithm JUMP in [2] uses $\lceil\sqrt{n}\rceil$. Since the threshold value of selecting j-vertices is changed to $\lceil\sqrt{n/2}\rceil$, the following lemmas in [2] are changed.

Lemma 3.1 *There are at most $\lfloor\sqrt{2n}\rfloor$ j-vertices in D.*

Lemma 3.2 *For each vertex $v \in C - D$, v is charged by at most $\lceil\sqrt{n/2}\rceil - 1$ f-vertices.*

Lemma 3.3 *If $|C| = 1$ then the performance ratio of algorithm NEW is at most $\lfloor\sqrt{2n}\rfloor + 1$.*

Theorem 3.4 *The performance ratio of NEW is at most $\sqrt{2n} + c_1$ for some constant c_1.*

Proof. Consider the case $|C| \geq 2$. By Lemma 3.1 and 3.2, the performance ratio is

$$\begin{aligned} & \frac{\lfloor\sqrt{2n}\rfloor + 1 + |C| \cdot (\lceil\sqrt{n/2}\rceil - 1)}{|C|} \\ \leq \; & \sqrt{2n} + c_1. \end{aligned}$$

□

Therefore, NEW has the performance ratio $\sqrt{2n} + c_1$ for general graphs. This improves the previous performance ratio $1.5\sqrt{n} + c_2$ [2] under the first on-line setting, where c_1 and c_2 are some positive constants.

References

[1] E. Cockayne, S. Goodman, S. Hedetniemi, A linear algorithm for the domination number of a tree, *Inform. Process. Lett.* 4 (1975) 41-44.

[2] G.H. King, W.G. Tzeng, On-line algorithms for the dominating set problem. *Inform. Process. Lett.* **61** (1997) 11-14.

[3] W.G. Tzeng, G.H. King, Tight Bound for the On-line Dominating Set Problem of Permutation Graphs, *submitted*.